

# HP Service Manager Exchange with SAP Solution Manager

Software Version: 1.10 patch 1

Service Manager Version: 9.x

---

## Installation and Administration Guide

Document Release Date: February 2014  
Software Release Date: February 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 Notices

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

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>). Portions Copyright © 1999-2003 The Apache Software Foundation. All rights reserved. SMSSMEX uses software from the Apache Jakarta Project including Apache Axis2, Apache Tomcat 5.5, Log4j Apache License, and Spring Framework.

### Trademark Notices

Java and Oracle® are registered trademarks of Oracle and/or its affiliates.

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

Adobe® and Acrobat® are trademarks of Adobe Systems Incorporated.

Intel®, Itanium® and Pentium® are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

SAP® is a registered trademark of SAP AG in Germany and in several other countries.

## Documentation Updates

This guide's title page contains the following identifying information:

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

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

**<http://h20230.www2.hp.com/selfsolve/manuals>**

This site requires that you register for an HP Passport and sign-in. To register for an HP Passport ID, go to:

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

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

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

## Support

You can visit the HP Software Support Online web site at:

**<http://www.hp.com/go/hpsoftwaresupport>**

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

HP Software Support Online 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 HP Software Support web site to:

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

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract.

To find more information about access levels, go to:

**[http://h20230.www2.hp.com/new\\_access\\_levels.jsp](http://h20230.www2.hp.com/new_access_levels.jsp)**

To register for an HP Passport ID, go to:

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

# Contents

<b>Introduction</b> .....	<b>9</b>
HP Incident Exchange .....	9
Existing Fragmented Incident Management Workflow .....	9
SAP Solution Manager Service Desk .....	10
HP Service Manager .....	10
Purpose of Document .....	10
Use Cases .....	11
Use Case 1: Incident Originates from Solution Manager .....	11
Use Case 2: Incident Originates from Service Manager .....	13
<b>Deployment Scenarios</b> .....	<b>15</b>
High Level Overview .....	15
Components .....	15
Support Matrix .....	16
<b>Installing and Configuring SMSSMEX on Tomcat</b> .....	<b>17</b>
Installing SMSSMEX .....	17
Prerequisites .....	17
Install SMSSMEX .....	17
Uninstall SMSSMEX .....	18
SMSSMEX Installed Files .....	18
Configuring Tomcat .....	19
Setting up Database .....	19
Oracle .....	20
MS-SQL 2005 .....	21
MS-SQL 2008 .....	23
Configuring ovictex.properties .....	23
Configuring File ovictexInternal.properties .....	24
External Helpdesks .....	24
Configuring FieldMapping.xml .....	25
Verifying Configuration .....	25
Starting/Stopping SMSSMEX .....	26
<b>Installing and Configuring SMSSMEX on WebLogic</b> .....	<b>27</b>
Installing SMSSMEX .....	27
Setting up Database .....	27
Configuring ovictex.properties .....	27
Configuring File ovictexInternal.properties .....	27
External Helpdesks .....	27

Configuring FieldMapping.xml . . . . .	27
Verifying Configuration . . . . .	27
Deploying on WebLogic . . . . .	28
<b>Applying SMSSMEX V1.10 Patch 1 . . . . .</b>	<b>29</b>
<b>Customizing HP Service Manager in Non-PD Environment . . . . .</b>	<b>31</b>
Creating a Service Manager User for Web Service . . . . .	31
Importing Customizations Using Unload . . . . .	32
Core Unload . . . . .	32
Demo Unload . . . . .	33
Customizing Demo Unload Manually . . . . .	34
Adding Integration Name to the Info Table . . . . .	34
Incident Custom Fields and Web Service Exposure . . . . .	34
Expose Custom Fields in Web Service Interface IncidentManagement . . . . .	35
Contacts Web Service Exposure . . . . .	37
Journal Separator Line Format . . . . .	38
Template . . . . .	39
Incident Process . . . . .	40
Incident Form . . . . .	41
Status and Hidden Metadata . . . . .	41
Exchange History . . . . .	41
Trigger Buttons . . . . .	43
Selection of External Helpdesk System . . . . .	45
SAP Configuration Item handling . . . . .	45
Overview . . . . .	45
Implementation . . . . .	45
Example Implementation via New Device Type and fill.fc . . . . .	45
Implementation Alternatives and Enhancements . . . . .	60
Configuring WSDL Mapping . . . . .	60
Adding Instance in SMIS and Configuring Parameters . . . . .	61
<b>Customizing HP Service Manager in PD Environment . . . . .</b>	<b>63</b>
Creating a Service Manager User for Web Service . . . . .	63
Importing Customizations Using Unload . . . . .	64
Core Unload . . . . .	64
Demo Unload . . . . .	65
Customizing Demo Unload Manually . . . . .	65
Adding Integration Name to the Info Table . . . . .	65
Incident Custom Fields and Web Service Exposure . . . . .	65
Expose Custom Fields in Web Service Interface IncidentManagement . . . . .	65
Contacts Web Service Exposure . . . . .	66
Customize Incident Workflow . . . . .	66
SAP Configuration Item Handling . . . . .	72
Overview . . . . .	72
Implementation . . . . .	73
Example Implementation via New Device Type and fill.fc . . . . .	73
Implementation Alternatives and Enhancements . . . . .	75

Configuring WSDL Mapping . . . . .	75
Adding Instance in SMIS and Configuring Parameters . . . . .	75
<b>Configuring SAP Solution Manager . . . . .</b>	<b>77</b>
Prerequisites . . . . .	77
Configuring SAP Solution Manager External Service Desk Interface . . . . .	78
Release Web Service . . . . .	79
Assign Roles to the Communication User . . . . .	80
Create HTTP Connection . . . . .	81
Create a Logical Port . . . . .	82
Configure Interface to SAP Solution Manager Service Desk . . . . .	84
Define Value Mapping for the Service Desk Interface . . . . .	84
Define Extended Interface Mapping for Service Desk . . . . .	84
Get SAP Solution Manager Service Port . . . . .	84
Solution Manager Tracing . . . . .	85
Enable tracing . . . . .	85
Download Trace File . . . . .	85
<b>Configuring Security . . . . .</b>	<b>87</b>
Security between SAP Solution Manager and Tomcat . . . . .	87
Configure SAP Solution Manager for SSL . . . . .	87
Checking SAP SSL Configuration . . . . .	87
Creating a Client PSE in Trust Manager . . . . .	89
Setting Up an Outgoing Connection in SAP Solution Manager . . . . .	91
Set up an Incoming Connection in SAP Solution Manager . . . . .	92
Set up SSL between SAP and SMSSMEX . . . . .	94
Create Keystore and Truststore . . . . .	94
Configure Tomcat SSL Use . . . . .	95
Configure Property Files . . . . .	95
Security Between HP Service Manager and SMSSMEX . . . . .	97
Configure HP Service Manager for SSL . . . . .	97
Generate a Private/Public Key Pair for Root Certificate Authority . . . . .	97
Generate a Private/Public Key Pair for Service Manager Server . . . . .	97
Generate the Client Keystore for Service Manager Client . . . . .	98
Generate the Client Keystore for SMSSMEX . . . . .	99
Generate the Trust-List Keystore for Service Manager Server . . . . .	100
SSL Configuration in Service Manager Server . . . . .	100
SSL Configuration in Service Manager Client . . . . .	101
Service Manager Web Client SSL Configuration . . . . .	102
Configure SMSSMEX for SSL Communication with Service Manager . . . . .	103
<b>Licensing . . . . .</b>	<b>105</b>
License Types . . . . .	105
Autopass License Management . . . . .	105
<b>Status Page . . . . .</b>	<b>107</b>
<b>Troubleshooting . . . . .</b>	<b>109</b>
checker.bat and encryptPasswords.bat Fail . . . . .	109

Incident not Sent to SAP AGS .....	109
java.lang.OutOfMemoryError .....	109
Record in EventIn is not Executed .....	110
Incident Update or Process Action Fails .....	112
Information is not Updated in SAP Solution Manager .....	114
Error Occurs When Installing SMSSMEX on Windows 2008 R2 (64-bit) Server .....	114
<b>Upgrading SMSSMEX .....</b>	<b>117</b>
Upgrading SMSSMEX from V1.00 to V1.01 .....	117
Upgrading SMSSMEX from V1.01 to V1.02 .....	120
Upgrading SMSSMEX from V1.02 to V1.10 .....	120
Upgrading SMSSMEX from V1.10 to V1.10 Patch 1 .....	128
<b>Incident Exchange Details .....</b>	<b>129</b>
Database Tables .....	129
Tools .....	130
Field Mapping Configuration .....	131
Types of Mapping .....	131
Structure of FieldMapping XML file .....	131
Composite Field Mapping .....	132
Field Value Mapping .....	133
Field Mapping Schema .....	133
Default Field Mapping File and Customization .....	136
Prerequisites .....	136
Adding Fields to fieldMapping.xml .....	137
Additional Information .....	138
Changeable Mappings .....	140
Person Synchronization Details .....	140
SAP Solution Manager to Service Manager .....	140
SMSSMEX Version .....	141
<b>Installing and Configuring SAPCRYPLIB .....</b>	<b>143</b>
<b>Logging .....</b>	<b>145</b>
<b>Deploying Button Icons .....</b>	<b>147</b>
Windows Client .....	147
Web Client .....	147
<b>SAP System Landscape Directory Registration .....</b>	<b>149</b>
Prerequisites .....	149
Registering System Landscape Directory .....	149



# 1 Introduction

This HP integration product implements HP Service Manager Exchange with SAP Solution Manager. This version only implements Service Manager Incident Exchange with SAP Solution Manager. Therefore, you can refer to this document for HP Incident Exchange whether the Service Manager Process Designer (PD) Content Pack is applied or not.

## HP Incident Exchange

Businesses today increasingly rely on their mission-critical SAP applications. Disruptions in the SAP environment have a severe business impact. Keeping the system continuously available has never been more vital for success. In any SAP landscape, business process disruptions caused by an application or infrastructure incident must be proactively prevented. If disruptions do occur, they need to be quickly and efficiently resolved. HP and SAP have teamed up to solve this issue.

Incident management in enterprises today consists of disconnected incident management systems that often implement divergent processes. This situation diminishes collaboration within IT operations, lowers quality of service and productivity.

The integration of SAP Solution Manager Service Desk with HP Service Manager provides a cohesive Incident and Service Request Management solution for the entire enterprise, resulting in higher enterprise availability, improved service quality and reduced IT costs.

HP Incident Exchange builds a dynamic link between HP Service Manager Software and SAP Solution Manager Service Desk and improves the Incident and Service Request Management Process throughout the entire enterprise. HP Incident Exchange offers dynamic integration between HP Service Manager and SAP Solution Manager Service Desk for improved incident workflow.

The interface to exchange support messages between HP Service Manager and SAP Solution Manager Service Desk was designed and developed jointly by HP and SAP and is certified by SAP.

## Existing Fragmented Incident Management Workflow

Performance monitoring of an SAP environment must include SAP and non-SAP applications.

## SAP Solution Manager Service Desk

To monitor and manage SAP environments, IT operations management uses the SAP Solution Manager Service Desk to collect information about SAP systems and serves as an internal help desk for SAP installations. Users and administrators can create support messages from any SAP system. The messages are processed centrally in the Solution Manager Service Desk.

If the support message involves an SAP application, a solution may be available in the SAP Service Marketplace or from SAP Active Global Support or from the in-house SAP support team. But if the issue is not caused by the SAP application, the message will be forwarded to the administrators responsible for the non-SAP systems. The support call needs to be entered in a second or third service desk and tracked until resolved. In the meantime, the SAP Service Desk team waits for feedback before closing the call and informing the originator, who is temporarily left “in the dark”.

## HP Service Manager

An incident can also be reported to the service desks monitoring non-SAP applications and infrastructure hardware and software. Many SAP customers have integrated these tasks in the HP Service Manager, which is able to support nearly all IT application and infrastructure components.

If a support call, for example, pertains to a “printing issue from an SAP application” and the HP Service Manager team detects no issue with the printer hardware or software, the call will be forwarded to the SAP service desk team to check whether it is related to the SAP application. Again the service call must be re-entered in a service desk, in this case in the SAP Solution Manager Service Desk. Additional information or attachments regarding the error or error resolution must be forwarded manually. The HP Service Manager team has to wait for feedback before informing the requesting user and closing the call.

In both cases the disconnected service desks and the fragmented incident management workflow impede the service desk team’s ability to resolve problems. Disadvantages of this non-integrated workflow are

- Only limited and often inconsistent information about the incident is available.
- It is difficult to monitor, track and report incidents or to work together toward resolution.
- Manual workarounds are required for the handover of incidents between the SAP and non-SAP service desks and for information updates.
- There is insufficient synchronization. The same incident may get reported, recorded and tracked in separate service desks, or the incidents may get lost or ‘dropped’.
- Expertise about the interrelationships of SAP applications with non-SAP applications and other IT components is lost.

This results in productivity loss and reduced quality of service.

## Purpose of Document

This document describes installation, configuration, administration and maintenance of HP Incident Exchange and the HP Incident Exchange web service. This guide is intended for use by HP consultants and application administrators that install and maintain HP Incident Exchange. This document is not an end user document.

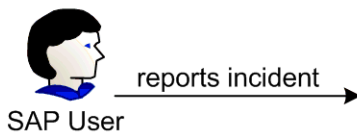
# Use Cases

This section discusses two use cases for HP Service Manager Exchange with SAP Solution Manager that demonstrate the integration scenarios.

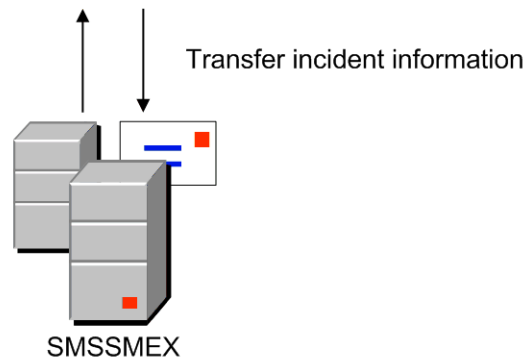
## Use Case 1: Incident Originates from Solution Manager

In this use case, a user reports an issue to SAP Solution Manager. The Solution Manager generates a new incident and sends the incident to Service Manager to request a solution for the issue.

1. SAP end user:  
encounter issue when using SAP application

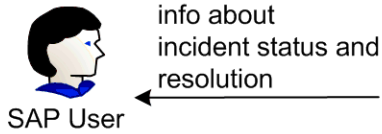


2. SAP Solution Support:
  - Analyse received incident and determine it as a infrastructure issue.
  - Send the incident to Service Manager.
4. SAP Solution Support:
  - Add more information for the incident
  - Send the incident to Service Manager again.



3. Infrastructure Support:
  - Record/classify received incident transferred from Solution Manager.
  - Find that the information of the incident is not enough for providing a solution.
  - Send incident back to Solution Manager.

9. SAP end user:  
The issue is resolved

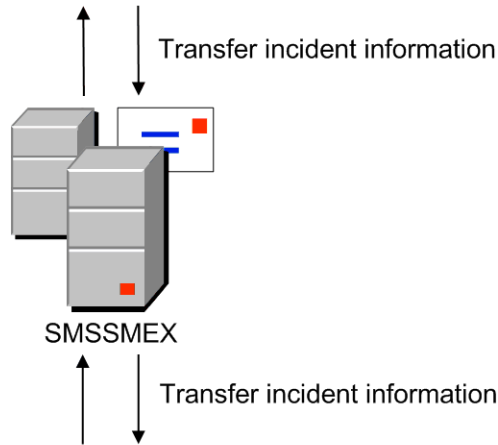


6. SAP Solution Support:

- The solution provided by Service Manager does not solve the issue of user.
- Reject the Solution to Service Manager

8. SAP Solution Support:

- The solution solve user's issue
- Close the incident



5. Infrastructure Support:

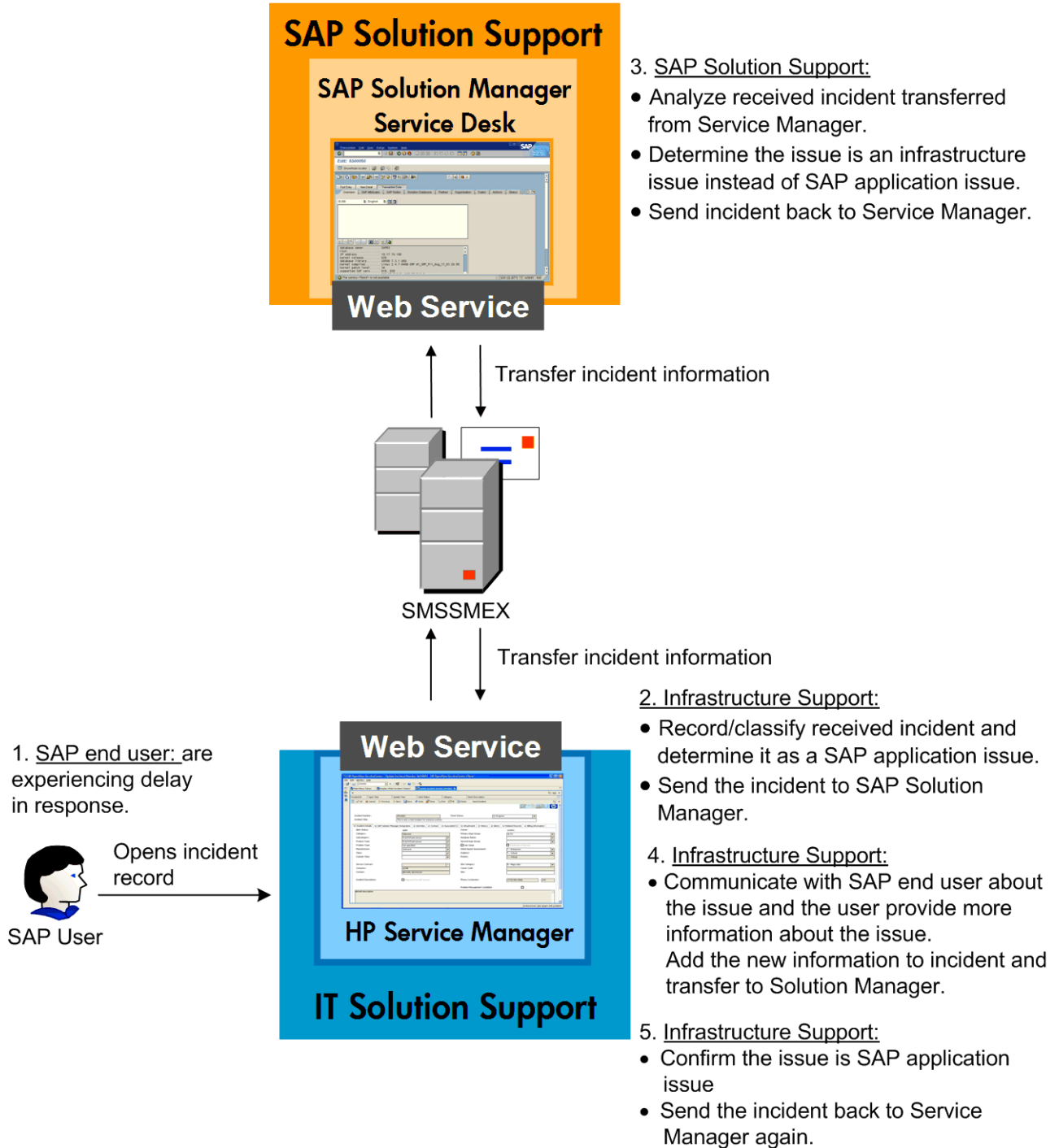
- Get additional information for the incident.
- Solve the issue.
- Send solution to SAP Solution Manager.

7. Infrastructure Support:

- Get to know that Solution Manager Support rejected the solution.
- Solve the issue with a new scheme.
- Send new solution to SAP Solution

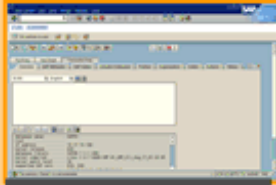
## Use Case 2: Incident Originates from Service Manager

In this use case, the user issue is captured and sent to Service Manager. An incident is generated in Service Manager and is sent to Solution Manager to request a solution for the issue.



## SAP Solution Support

### SAP Solution Manager Service Desk



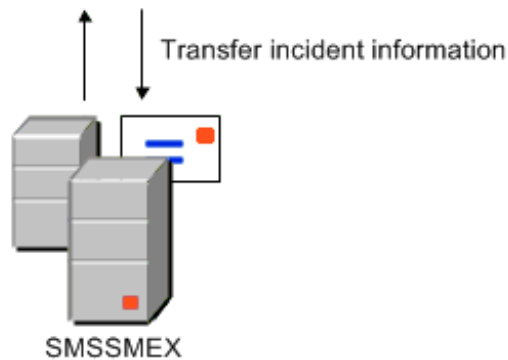
### Web Service

#### 6. SAP Solution Support:

- Get additional information for the incident.
- Solve the issue.
- Send solution to Service Manager.

#### 8. SAP Solution Support:

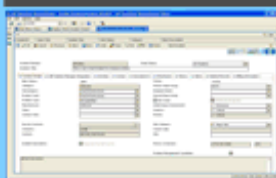
- Get to know that infrastructure Support rejected the solution.
- Solve the issue with a new scheme.
- Send new solution to Service Manager.



SMSSMEX

Transfer incident information

### Web Service



### HP Service Manager

## IT Solution Support

#### 7. Infrastructure Support:

- Find that the solution provided by Solution Manager is only a work around instead of solution for the issue.
- Send back the Solution to Solution Manager

#### 9. Infrastructure Support:

- The solution solve user's issue
- Close the incident

#### 10. SAP end user:

Issue of delay in user response is resolved



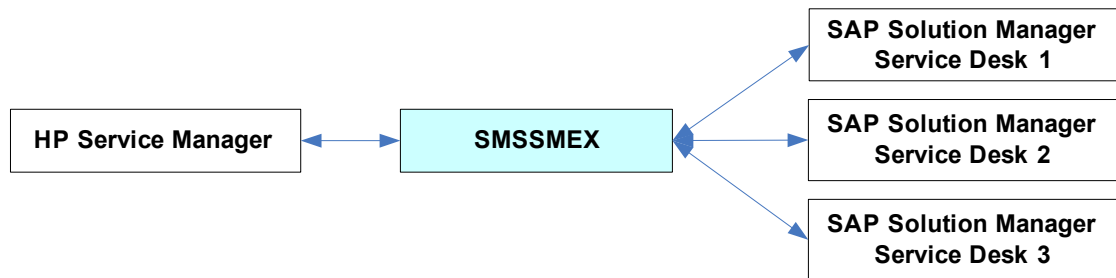
SAP User

Info Incident solved

## 2 Deployment Scenarios

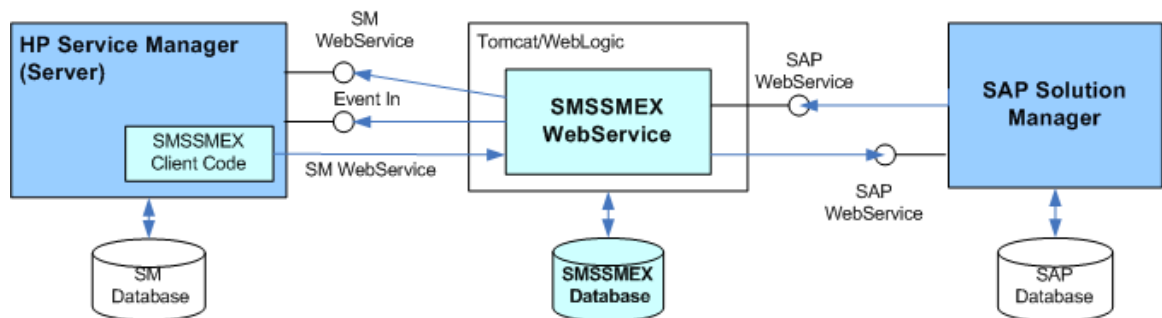
### High Level Overview

SMSSMEX integrates a single Service Manager server with multiple external helpdesk systems.



### Components

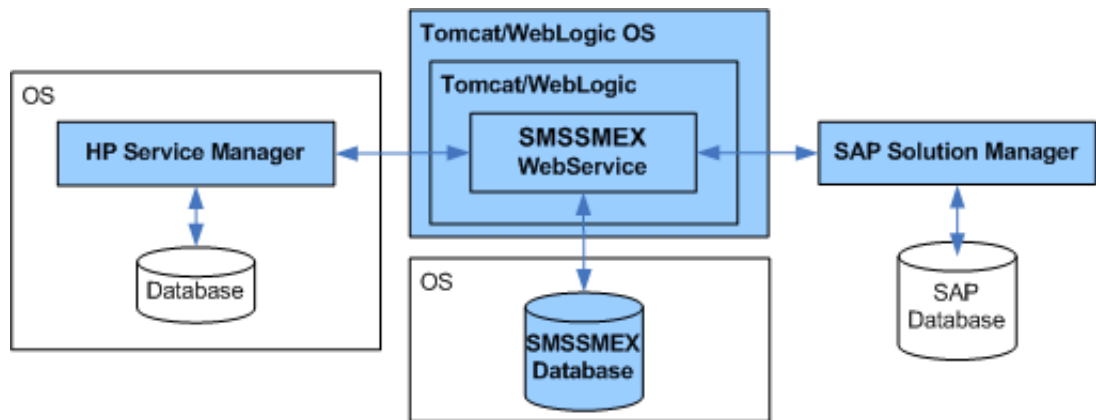
The following diagram shows the component details.



- HP Service Manager Server is the HP service desk system.
- Service Manager DB provides persistent storage for HP Service Manager.
- SMSSMEX Client Code consists of RAD and Java scripts, table definitions and GUI formats. The SMSSMEX webservices are called from this client code.
- WebServer is a Tomcat Web Application Server or WebLogic Application Server that hosts the SMSSMEX WebService (deployed as a .war file).
- SMSSMEX WebService exposes the incident webservice of HP Service Manager in the SAP format and transfers client requests to SAP Solution Manager webservices.
- SMSSMEX Database provides persistent storage for the SMSSMEX WebService.
- SAP Solution Manager is the Service Desk.

# Support Matrix

The following diagram shows the supported components.



The following table shows the supported component versions.

**Table 1 Supported component versions**

Platform	Component	Versions
Service Manager / ServiceCenter	Service Manager	See the Support Matrix section on <b>ID:201 Incident Exchange (SM-SAP Solution Manager)</b> on the HP Software Support Online website.
	ServiceCenter	
SMSSMEX OS	Windows Server	2003, 2003 R2 (32-bit) 2008, 2008 (32-bit) <sup>a</sup>
	Linux	SUSE10
SMSSMEX Database	Microsoft @ SQL Server	2005, 2008
	Oracle Standard and Enterprise Edition	9.2, 10.2, 11
SAP Solution Manager	Solution Manager 7.0	>= SP12
	Solution Manager 7.1	
WebLogic Server		10.3.2

a. To install SMSSMEX on Windows server 2008 R2 (64-bit), refer to *Error Occurs When Installing SMSSMEX on Windows 2008 R2 (64-bit) Server* on page 114.



# 3 Installing and Configuring SMSSMEX on Tomcat

## Installing SMSSMEX

The HP Service Manager Exchange with SAP Solution Manager product CD includes an autorun program for installation.

### Prerequisites

- It is NOT recommended to install SMSSMEX and Service Manager/ServiceCenter on the same server.
- For installation on Unix, install HP OpenView AutoPass manually before the installation of HP Service Manager Exchange with SAP Solution Manager.

The AutoPass installer is available from:

```
<SMSSMEX1.10 Release  
Package>\AutoPassInstaller\Linux\HPOvLic-05.40.010-Linux2.6-release.r  
pm
```

### Install SMSSMEX

- 1 Log in to the operation system as a super user.
- 2 The installer is in:
  - <SMSSMEX1.10 Release Package>\InstData\Windows\VM\install.exe (Windows 2003 Release 2 and Windows 2008)
  - <SMSSMEX1.10 Release Package>\InstData\Linux\VM\install.bin (Linux)
- 3 Run install.bin or install.exe. The Introduction dialog appears.
- 4 Click **Next**. The license agreement appears.
- 5 Select **I Accept the terms of License Agreement**.
- 6 Click **Next**. The Choose Install Folder page displays. For example, the default installation folder on Windows 2003 is C:\Program Files\HP\SMSSMEX.
- 7 Click **Next**. Review the summary information.
- 8 Click **Install**. The files are installed. For Windows installation, HP AutoPass Licensing installs the HP OpenView component automatically.  
The Install Complete dialog appears.
- 9 Click **Done** to close the installer.

## Uninstall SMSSMEX

To uninstall SMSSMEX on Windows, execute

`<SMSSMEX_installDir>\Uninstall SMSSMEX\Uninstall SMSSMEX.exe`

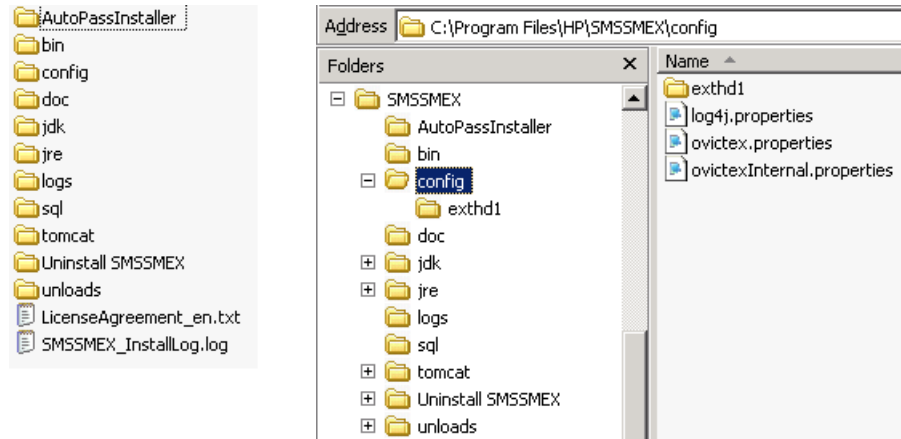
Or simply go to **Start** → **Programs** → **SMSSMEX** → **Uninstall SMSSMEX**.

To uninstall SMSSMEX on Unix, execute

`<SMSSMEX_installDir>/Uninstall SMSSMEX/Uninstall_SMSSMEX`

## SMSSMEX Installed Files

After installation, the SMSSMEX folder has the following contents.



**Table 2** Contents of \SMSSMEX

Directory	Content
bin	Executable commands and product description file
config	Web service configuration files
config\exthd1	Template for an external helpdesk configuration
sql	Database table creation/deletion scripts
unloads\SC6.2	ServiceCenter 6.2 customization unload files <b>Note:</b> SMSSMEX 1.10 patch 1 unload files are available at <code>&lt;SMSSMEX1.10p1 Release Package&gt;\unloads\ directory</code> .
unloads\SM9.2	Service Manager 9.2x customization unload files
unloads\SM9.3	Service Manager 9.3x customization unload files
logs	Log files
jdk	Internal JDK 5
tomcat	Tomcat 5.0.28

**Table 2** Contents of \SMSSMEX

jre	Internal JRE by InstallAnywhere
AutoPassInstaller	HP AutoPass Licensing component installer (Windows only)
Uninstall SMSSMEX	Executable file for uninstallation

## Configuring Tomcat

The connector for deploying the web service must be enabled. Uncomment the port specification in `<SMSSMEX_installDir>\tomcat\conf\server.xml`. For example:

```
<Connector port="8080"
  redirectPort="8443"
  maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
  enableLookups="false" acceptCount="100" debug="0"
  connectionTimeout="20000" disableUploadTimeout="true" />
```

You can modify the ports if necessary.

## Setting up Database

This section describes how to setup the database.



The SMSSMEX web service uses a database to store metadata. The SMSSMEX web service must be able to read table `v$database` (Oracle) or execute function `SERVERPROPERTY('ProductVersion')` (SQLServer). These system tables are queried when validating the database connections.

## Oracle

To setup the Oracle database do the following:

- 1 Create a user.

```
CA Select C:\WINDOWS\system32\cmd.exe - sqlplus
C:\TEMP>sqlplus
SQL*Plus: Release 11.2.0.1.0 Production on Tue Oct 9 06:43:42 2012
Copyright (c) 1982, 2010, Oracle. All rights reserved.
Enter user-name: /as sysdba
Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Pro
With the Partitioning, OLAP, Data Mining and Real Application Testing
SQL> create user ovictex identified by password;
User created.
SQL> grant connect,dba,resource to ovictex;
Grant succeeded.
SQL> █
```

- 2 Give the user the right to do a select on table v\$database. This system table is queried by the SMSSMEX web service to validate database connections.
- 3 Login as the user and run the script create\_tables\_oracle.sql (log in from path <SMSSMEX\_installDir>\sql so that the script is found). This creates all required tables.

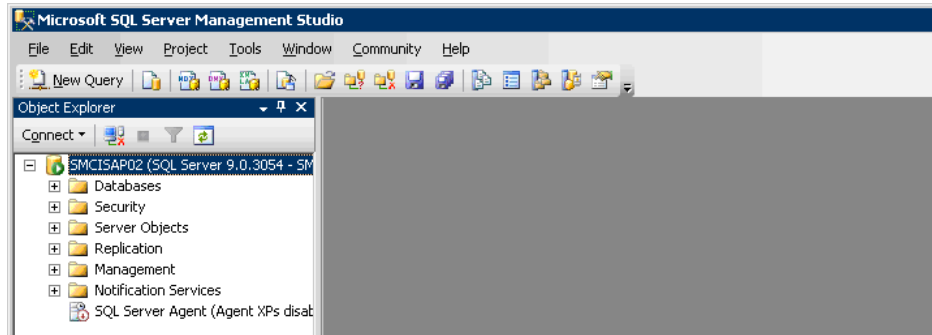
```
CA C:\WINDOWS\system32\cmd.exe - sqlplus
C:\TEMP>sqlplus
SQL*Plus: Release 11.2.0.1.0 Production on Tue Oct 9 06:51:04 2012
Copyright (c) 1982, 2010, Oracle. All rights reserved.
Enter user-name: ovictex
Enter password:
Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Pro
With the Partitioning, OLAP, Data Mining and Real Application Testing
SQL> @create_tables_oracle
Table created.
Table created.
Table created.
Table created.
Table created.
SQL> _
```

These tables are created within the schema of the database user (the tables are logically separated and do not interfere with each other).

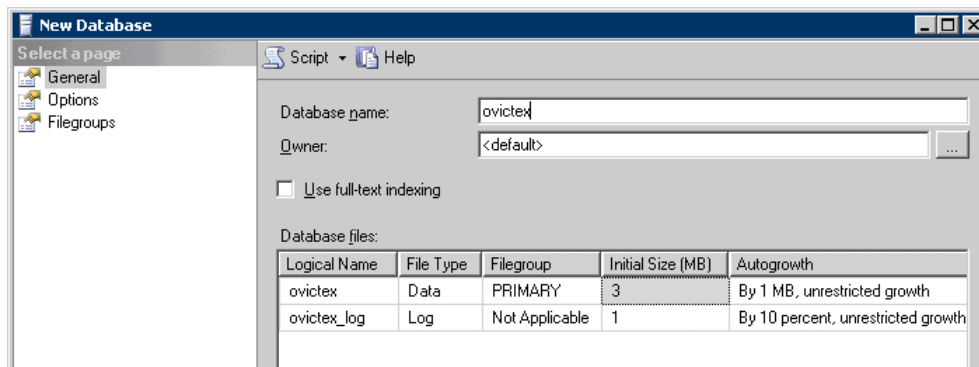
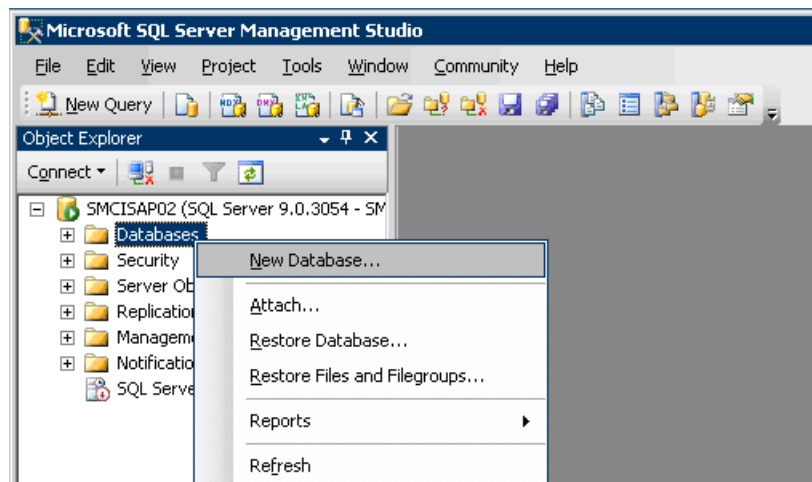
## MS-SQL 2005

Do the following to create the required separate database for SMSSMEX tables:

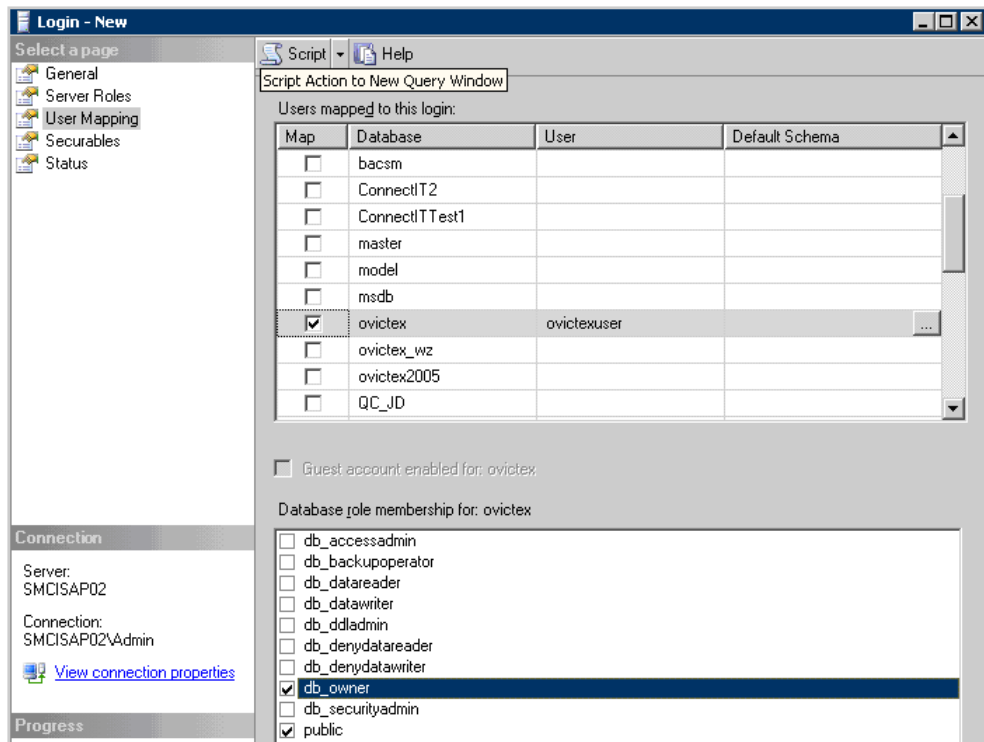
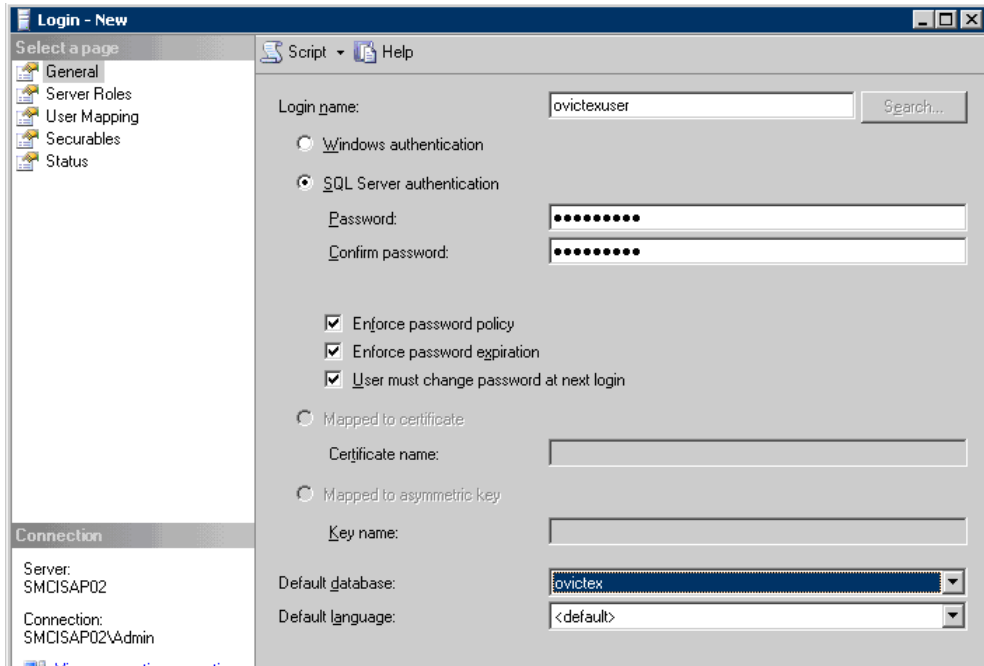
- 1 Launch SQL Server Management Studio.



- 2 Create a new database (ovictex). Right-click on **Databases** and choose **New Database**.



- 3 Create a database user (ovictexuser) with permission for database ovictex. Right-click on **Security/Login** and select **New Login**.



- 4 Create the SMSSMEX tables.
  - a Click **New Query** on the toolbar and select database **ovictex**.
  - b Copy and execute the SQL scripts under folder
 

```
<SMSSMEX_installDir>\sql\create_tables_sqlserver.sql.
```

## MS-SQL 2008

The DB setup for MS-SQL 2008 is similar to the MS-SQL 2005 setup. Refer to [MS-SQL 2005](#) on page 21 for detail information.

## Configuring ovictex.properties

File `<SMSSMEX_installDir>/config/ovictex.properties` must specify the local helpdesk installation. The file comments describe how to do this.



To configure the passwords, use command line application `<SMSSMEX_installDir>/bin/encryptPasswords.bat | sh` (do not enter the password directly in the file; passwords are stored in encrypted format). There are several sensitive fields that must be encrypted. These fields are discussed below. For more information about using `encryptPasswords.bat | sh`, see [Tools](#) on page 130.

The following parameters must be configured:

- Service Manager web service endpoint

- To connect to a Service Manager:

```
sc.webservice.endpoint = http://<ServiceManager host>:<Port>/  
sc62server/PWS
```

- To connect to a ServiceCenter:

```
sc.webservice.endpoint = http://<ServiceCenter host>:<Port>/sc62server/ws
```

- The following are required parameters:

```
sc.user=<web service endpoint access user name>  
sc.password=<encrypted password>
```

▶ `sc.password` must be filled by `encryptPasswords.bat | sh`. SMSSMEX supports SSL connections to Service Manager, but the parameter values are different than above and additional parameters must be set (see [Security between SAP Solution Manager and Tomcat](#) on page 87).

- SMSSMEX database configuration information:

```
ovictex.db.type= <oracle | sqlserver>  
ovictex.db.host=<database server address>  
ovictex.db.port=<database server port number>  
ovictex.db.instance=<sqlserver database server instance>  
ovictex.db.name=<database name or oracle DB SID>  
ovictex.db.user=<database user name>  
ovictex.db.password=<database password>
```

▶ `ovictex.properties` contains examples. `ovictex.db.password` must be filled by `encryptPasswords.bat | sh`.

- One or more External Helpdesk instance names.

- Parameters are **exthd.instances.id.<number>**, where <number> is a number {1,...,n}.
- First number must be 1 and each number must be greater than the previous.
- ExtHdInstanceName differentiates multiple External Helpdesks and is the name of the subfolder in <SMSSMEX\_installDir>/config and the ExtHd configuration file.
- Incident category in Service Manager.
- Property **ovhd.incident.informationlog.entry.separator** should be configured to a unique value that is not contained in messages exchanged between Helpdesks. By default it is configured to “----”. Service Manager must be configured to use this separator to append information to the Journal. If this separator is contained in a message then duplicate information could be sent to the external Helpdesk (no data is lost).

## Configuring File ovictexInternal.properties

The property file for internal configurations is in the <SMSSMEX\_installDir>/config directory of the SMSSMEX installation. There is typically no need to configure this file.

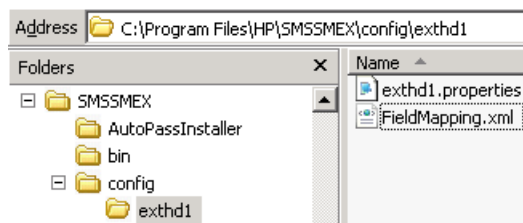
## External Helpdesks

Main configuration file <SMSSMEX\_installDir>/config/ovictex.properties must define all External Helpdesk Instances. For example:

- **exthd.instances.id.1 = exthd1**
- **exthd.instances.id.2 = SAP\_exthd2**
- **exthd.instances.id.3 = NY200BM**

Each external helpdesk has the following configuration files:

- <ExtHdInstanceName>.properties
- FieldMapping.xml



- ▶ The same names (such as exthd1, SAP\_exthd2, NY200BM) must be used for the names of subfolders with specific configuration file names. The names must not contain spaces or special characters. The default configuration comes with a defined exthd1 sample External Helpdesk configuration.

To create a new instance:

- 1 Add a new line in ovictex.properties for the new ExtHd.  
exthd.instances.id.2 = exthd2



- 2 Create the new subfolder `<SMSSMEX_installDir>/config/exthd2`.
- 3 Copy the configuration files for `exthd1` to `exthd2`.
- 4 Rename `<SMSSMEX_installDir>/config/exthd2/exthd1.properties` to `<SMSSMEX_installDir>/config/exthd2/exthd2.properties`.
- 5 Make the required changes to the new files.
- 6 The following parameters must be configured in `<ExtHdInstanceName>.properties`:

```
exthd.webservice.endpoint = http://<SolutionManager host>:<Port>/sap/bc/srt/rfc/sap/ICT_SERVICE_DESK_API?SAP-CLIENT=<SAP client number>
exthd.webservice.authentication.scheme = BASIC
exthd.webservice.authentication.username = <SAP client user name>
exthd.webservice.authentication.password = <encrypted SAP client user password>
```

IMG activity guides you to SAP transaction `/nsmicm`. Select the activity in menu **Goto** → **Services**.

### ICM Monitor - Service Display

Active Services							
No.	Log	Service Name/Port	Host Name	Keep Alive	Proc.Timeo	Actv	External Bind
<input type="checkbox"/>	1	HTTP 8003	gomorrah.deu.hp.com	30	60	✓	
<input type="checkbox"/>	2	HTTPS 8001	gomorrah.deu.hp.com	30	60	✓	

This transaction shows the host and port for access to the SAP Solution Manager Service Desk web service. Specify the host/port in `<ExtHdInstanceName>.properties` as the endpoint entry.

▶ `exthd.webservice.authentication.password` must be filled by `encryptPasswords.bat | sh`.

## Configuring FieldMapping.xml

The files `<SMSSMEX_installDir>/config/<ExtHdInstanceName>/FieldMapping.xml` must be adjusted to send/receive special/customized fields to/from the external Helpdesk. For detailed information see [Field Mapping Configuration](#) on page 131.

## Verifying Configuration

Verify the configuration with the checker tool before trying to exchange incidents between Service Manager and SAP Solution Manager. The checker error messages are much more helpful for troubleshooting than Service Manager and Solution Manager error messages.

To execute the checker run

```
<SMSSMEX_installDir>/bin/checker.bat | sh
```

Checker checks the environment, database and HTTP connections and configuration of Service Manager. No incidents are exchanged. The following are the possible results:

- OK

- ERROR (partial failure; checks that the check depends on have failed)
- FAIL (with troubleshooting recommendations)

You can re-run a check by passing the number of the check to the executable. You can also examine the Incident Exchange log messages or run a trace. For more information about using `checker.bat | sh`, see *Tools* on page 130.

## Starting/Stopping SMSSMEX

Starting from Windows:

```
<SMSSMEX_installDir>\bin\setup startup
```

Stopping from Windows:

```
<SMSSMEX_installDir>\bin\setup shutdown
```

Starting from Linux:

```
<SMSSMEX_installDir>/bin/setup.sh startup
```

Stopping from Linux:

```
<SMSSMEX_installDir>/bin/setup.sh shutdown
```

---

## 4 Installing and Configuring SMSSMEX on WebLogic

### Installing SMSSMEX

See *Installing SMSSMEX* on page 17 for detailed instructions.

### Setting up Database

See *Setting up Database* on page 19 for detailed instructions.

### Configuring ovictex.properties

See *Configuring ovictex.properties* on page 23 for detailed instructions.

### Configuring File ovictexInternal.properties

See *Configuring File ovictexInternal.properties* on page 24 for detailed instructions.

### External Helpdesks

See *External Helpdesks* on page 24 for detailed instructions.

### Configuring FieldMapping.xml

See *Configuring FieldMapping.xml* on page 25 for detailed instructions.

### Verifying Configuration

See *Verifying Configuration* on page 25 for detailed instructions.

## Deploying on WebLogic

- 1 Before starting the WebLogic server, set an environment variable named "SMSSMEX\_HOME" to the pathname where this application is installed.  
  
For example, if the WebLogic server is installed in the /opt/HP/SMSSMEX directory, set the environment variable to the following:  

```
$ export SMSSMEX_HOME=/opt/HP/SMSSMEX
```
- 2 Start the WebLogic server and launch the WebLogic administration console.
- 3 Deploy the ovictex.war file in the /opt/HP/SMSSMEX/war directory. See the following steps for an example:
  - a Select **Domain Structure > Deployments** and click **Install**.
  - b Use the Install Application Assistant to locate the ovictex.war file.
  - c Select **Install this deployment as an application** and click **Next** until last step.
  - d Click **Finish** to exit the installation wizard.

For advanced configuration, refer to *BEA WebLogic Server Administration Console Online Help* for more information.

# 5 Applying SMSSMEX V1.10 Patch 1

After the Service Manager Process Designer (PD) Content Pack 9.30.3 is installed, perform the following tasks to upgrade SMSSMEX from v1.10 to v1.10 patch 1:

## Task 1: Upgrading SMSSMEX middleware on Tomcat

Do the following:

- 1 Stop SMSSMEX v1.10.  
See *Starting/Stopping SMSSMEX* on page 26 to stop SMSSMEX v1.10.
- 2 Update the `ovictex.properties` configuration file by adding the following codes to `<SMSSMEX1.10_installDir>\config\ovictex.properties`.  

```
# default category  
ovhd.incident.default.category = incident
```
- 3 Deploy `ovictex.jar` by copying `<SMSSMEX1.10p1 Release Package>\jars\ovictex.jar` to `<SMSSMEX1.10_installDir>\tomcat\webapps\ovictex\WEB-INF\lib`.
- 4 Start SMSSMEX v1.10 with patch applied.  
See *Starting/Stopping SMSSMEX* on page 26 to start SMSSMEX v1.10 with patch applied.

## Task 2: Creating a Service Manager user for web service

See *Creating a Service Manager User for Web Service* on page 63 for more information.

## Task 3: Importing the PD core unload file

See *Core Unload* on page 64 for more information.

## Task 4: Updating the IncidentManagement expression

See *Expose Custom Fields in Web Service Interface IncidentManagement* on page 65 for more information.

## Task 5: Customizing incident workflow

See *Customize Incident Workflow* on page 66 for more information.

## Task 6: Handling SAP configuration items

See *SAP Configuration Item Handling* on page 72 for more information.



# 6 Customizing HP Service Manager in Non-PD Environment

This chapter describes the customization required for HP Service Manager for the integration if the Service Manager Process Designer (PD) Content Pack is not installed.

## Creating a Service Manager User for Web Service

Incident Exchange uses one Service Manager user to connect to Service Manager web services. The user and the user role should be dedicated for the integration. The user requires the following permissions:

The screenshot shows the 'User Role: OVICTEX' configuration window. The 'User Role' field is set to 'OVICTEX' and the 'Description' is 'Automated Incident Exchange user role'. Below this, there are tabs for 'Profiles', 'Startup', and 'Data Access'. Under the 'Profiles' tab, several profile fields are visible: 'Service Profile', 'Incident Profile', 'Problem Profile', 'Configuration Profile', 'Contract Profile', 'SLA Profile', 'Change Profiles', and 'Request Profiles'. All these fields are currently set to 'SYSADMIN'.

Do the following:

- 1 Log in to Service Manager with a System Administrator account.
- 2 Select **System Administration** → **Ongoing Maintenance** → **User Roles**.
- 3 Search for **system administrator** on Service Manager 9.x and above.
  - 🚩 In case your database is configured to case sensitive, try to use all lowercase search keyword instead of all UPPERCASE one, or vice versa.
- 4 Enter **OVICTEX** as the User Role.
- 5 Change Description to **Automated Incident Exchange user role**.
- 6 Click **Add**.
- 7 Select **System Administration** → **Ongoing Maintenance** → **User Quick Add Utility**.
- 8 Enter **ovictex**, **INCIDENT EXCHANGE**, **Incident**, **Exchange**, **ovictex@hp.com**.
- 9 Click **Next**.

- 10 For **User to clone** select **falcon**.
- 11 Click **Finish**.
- 12 Click **Save**.
- 13 Go to **System Administration** → **Ongoing Maintenance** → **Operators**, enter **ovictex** in the Login Name field, then click **Search**.
- 14 Change User Role to **OVICTEX**.
- 15 In the Security tab:
  - a Enter the operator password for Password.
  - b Uncheck Expire Password.
  - c Check Never Expire Password.
- 16 Click **Save**.

## Importing Customizations Using Unload

This section describes how to configure Service Manager using unload. Additional customization of Service Manager is later required for the integration.

### Core Unload

Unloads are used to transfer customizations from one Service Manager installation to another Service Manager installation. The Incident Exchange provides core unloads at the following paths:

- <SMSSMEX1.10p1 Release Package>\unloads\SM9.2\core\_sm9.2.unl for Service Manager 9.2x
- <SMSSMEX1.10p1 Release Package>\unloads\SM9.3\core\_sm9.3.unl for Service Manager 9.3x

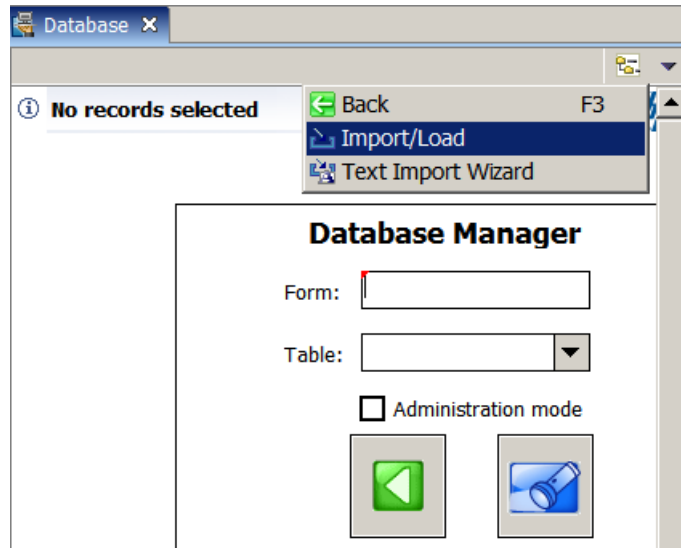
This unload contains new Service Manager records that are unique to Incident Exchange and do not override any existing Service Manager records.

To import the unload do the following:

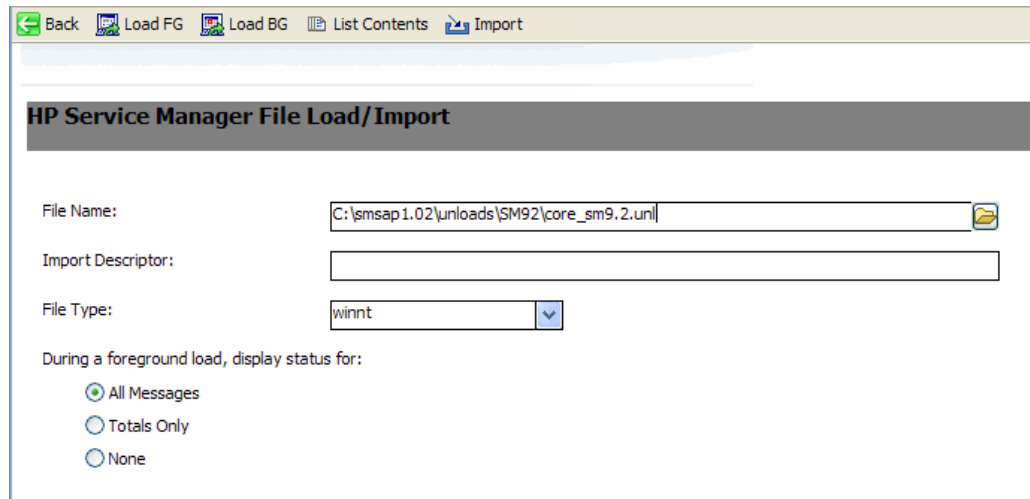
- 1 In the Service Manager client select **Tailoring** → **Database Manager**.



- 2 Select **Import/Load** from the menu.



- 3 Select <SMSSMEX1.10p1 Release Package>\unloads\SM9.2\core\_sm9.2.unl for Service Manager 9.2x, or select <SMSSMEX1.10p1 Release Package>\unloads\SM9.3\core\_sm9.3.unl for Service Manager 9.3x.
- 4 Click **Load FG** to start the import.



## Demo Unload

The demo unload has all the customizations (for an uncustomized, default installation of Service Manager) required to set up a working Incident Exchange for a customer demonstration or evaluation.



Do not import the demo unload into existing development or production systems. The demo unload requires an uncustomized, default installation of Service Manager. The demo unload overrides many standard Service Manager records, and can not be removed or undone.

To import the unload do the following:

- 1 In the Service Manager client select **Tailoring** → **Database Manager**.

- 2 Select **Import/Load** from the menu.
- 3 Browse to the unload at
  - <SMSSMEX1.10p1 Release Package>\unloads\SM9.2\demo\_sm9.2.unl for Service Manager 9.2x
  - <SMSSMEX1.10p1 Release Package>\unloads\SM9.3\demo\_sm9.3.unl for Service Manager 9.3x
- 4 Press **Load FG** to start the import.

## Customizing Demo Unload Manually

This section describes how to manually customize your integration with the demo unload.

### Adding Integration Name to the Info Table

For every SMIS integration, a field with the integration name should be created in the Info table.

- 1 Log in to Service Manager with a System Administrator account.
- 2 Select **System Definition** → **Tables** → **info** → **Tab Fields and Keys**.
- 3 Create the following field in SMIS structure.

Field Name	Data type
SMSAP	Logical

- 4 Click **Save**.

### Incident Custom Fields and Web Service Exposure

Incident Exchange accesses Service Manager Incidents via the `probsummary` table. The factory-default exposure `IncidentManagement.wsdl` (service name `IncidentManagement` and object name `Incident`) is used, allowing Incident Exchange to function with other clients. Incident Exchange requires exposure of additional fields in the web service and creation of new fields.

- 1 Select **System Definition** → **Tables** → **probsummary** → **Tab Fields and Keys**.
- 2 Create the following additional fields in table `probsummary`.

**Table 3 Incident custom fields and web service**

Field name	Caption	Data type
custom.text.01	CustomText01	Character
custom.text.02	CustomText02	Character
...	...	...
custom.text.10	CustomText10	Character

**Table 3 Incident custom fields and web service (cont'd)**

sap.sid	Sap SID	Character
sap.client	Sap Client	Character
sap.installationnumber	Sap installation number	Character
hidden.meta.data	Hidden meta data for Incident Exchange	Character
is.incident.exchange	Flag for affiliation with Incident Exchange	Logical
exthd	External helpdesk for Incident Exchange	Character
sap.incident.type	Sap Incident type	Character

- 3 The history is written to an additional field in `probsummary` table:
  - Add an Array `exchange.history`.
  - Add structures that are also named `exchange.history`.
  - Add structure fields `date.stamp` of type `Date/time` and `history.update` of type `Text`.
- 4 Incident Exchange is triggered asynchronously, requiring a handshaking mechanism to avoid triggering an action multiple times. To implement the mechanism, add a Boolean field named `is.ictex.action.blocked`. The fields are updated through Event Services and do not need to be exposed.

Field name	Type	Caption
exchange.history exchange.history date.stamp history.update	Array Structure Date/time Character	Log of Incident Exchange actions and events.
is.ictex.action.blocked	Logical	Flag that indicates if Incident Exchange is performing an exchange action.

## Expose Custom Fields in Web Service Interface IncidentManagement

To expose custom fields, do the following:

- 1 Select **Tailoring** → **Web Services** → **WSDL Configuration** on Service Manager 9.31 and earlier, or select **Tailoring** → **Web Services** → **Web Service Configuration** on Service Manager 9.32 and later.
- 2 Enter **IncidentManagement** in the `Service Name` field.

- Click **Search**.

**External Access Definition**

Service Name: IncidentManagement

Name: probsummary      Object Name: Incident

Allowed Actions    Expressions    Fields

Allowed Actions	Action Na...	Action Type
add	Create	
close	Close	
reopensave	Reopen	
resolvesave	Resolve	
save	Update	

- Select the **Fields** tab.

**External Access Definition**

Service Name: IncidentManagement

Name: probsummary      Object Name: Incident

Allowed Actions    Expressions    Fields

Field	Caption	Type
action	IncidentDescription	
agreement.id	SLAAgreementID	
alert.status	AlertStatus	
assignee.name	AssigneeName	
assignment	PrimaryAssignmentGr...	
brief.description	BriefDescription	
category	Category	
close.time	ClosedTime	DateTimeType
closed.by	ClosedBy	
company	Company	
contact.name	Contact	
explanation	Solution	
first.name	ContactFirstName	
fix.type	ResolutionFixType	
folder		
initial.impact	InitialImpact	
last.name	ContactLastName	
location	Location	
logical.name	ConfigurationItem	

- Add the custom fields in table probsummary that are exposed in web service interface IncidentManagement. If the field type is Character, then its not required to select the web service interface type for the field. The type of the field will be StringType.

**Table 4 Custom fields in IncidentManagement**

Field	Caption	Type
custom.text.01	CustomText01	
custom.text.02	CustomText02	
...	...	
custom.text.10	CustomText10	
sap.sid	SapSid	
sap.client	SapClient	

**Table 4 Custom fields in IncidentManagement (cont'd)**

sap.installationnumber	SapInstallationNumber	
hidden.meta.data	HiddenMetaData	
is.incident.exchange	IsIncidentExchange	BooleanType
exthd	Exthd	
priority.code	PriorityCode	
planned.end	PlannedEnd	DateTimeType

6 Change Expressions from

update.action in \$L.file=update.action in \$L.file.save  
to ([INCIDENTMANAGEMENT] in code\_sm9.txt)

**if (not null(number in \$L.file)) then (update.action in \$L.file=update.action in \$L.file.save)**

Then, add the following expression:

```
if (hidden.meta.data in $L.file="Closed") then (problem.status in $L.file="Closed";status in $L.file="closed";if null(resolution.code in $L.file) then (resolution.code in $L.file="Automatically Closed";resolution in $L.file=insert(resolution in $L.file, 1, 1, "Closed by SMSAP integration.")))
```

**External Access Definition**

Service Name:   Released  
 Name:   Deprecated  
 Object Name:

Allowed Actions Expressions Fields

```
cleanup($apm.activity);cleanup($pmc.actions);if same(update.action in $L.file, update.action in $L.file.save) then ($L.need.to.update=true)
if journal.pm in $G.pm.global.environment then (journal.pm.order in $G.pm.global.environment=nullsub(journal.pm.order in $G.pm.global.environment, 1);$pmc.details=nullsub(action in $L.file.save, {})+{"**** Past Updates ****"}+nullsub(update.action in $L.file.save, {});
$pmc.actions=nullsub(update.action in $L.file) else ($pmc.details=nullsub(action in $L.file.save, {});$pmc.actions=nullsub(update.action in $L.file, {}))
if ($pmc.details={}) then ($pmc.details={});if ($pmc.actions={}) then ($pmc.actions={"no update provided"})
if (not null(number in $L.file)) then (update.action in $L.file=update.action in $L.file.save)
if (status in $L.file.save~="closed" and status in $L.file.save~="resolved") then ($apm.activity="external update")
if ($L.need.to.update=true) then ($pmc.actions=NULL)
if (hidden.meta.data in $L.file="Closed") then (problem.status in $L.file="Closed";status in $L.file="closed";if null(resolution.code in $L.file) then (resolution.code in $L.file="Automatically Closed";resolution in $L.file=insert(resolution in $L.file, 1, 1, "Closed by SMSAP integration.")))
```

## Contacts Web Service Exposure

To expose the contacts web service, do the following:

- 1 Select **Tailoring** → **Web Services** → **WSDL Configuration** on Service Manager 9.31 and earlier, or select **Tailoring** → **Web Services** → **Web Service Configuration** on Service Manager 9.32 and later.
- 2 Input **contacts** in the Name field.

- 3 Click **Search**.
- 4 Select the **Fields** tab.
- 5 Add the following fields:

Field	Caption	Type
fax.phone	Fax	
operator.id	OperatorID	

## Journal Separator Line Format

New entries are added to the Journal at the top. When an Incident is exchanged with SAP Solution Manager, only updates are exchanged to avoid duplication of journal entries.

Incident Number:  Ticket Status:

Incident Title:

Incident Details
  Sap Solution Mana...
  Activities
  Contact
  CIs and Services
  Attachment
  Hist

Action / Resolution
  Site Visit
  Journal Updates
  Historic Activities

```

08/19/08 22:44:00 US/Pacific (ovictex):
External Helpdesk : SAP Solution Manager provided solution
test fff
08/19/08 22:38:02 US/Pacific (falcon):
The solution for this Incident as proposed by SAP SolutionManager has been rejected.
08/19/08 20:25:44 US/Pacific (ovictex):
Additional information received from External Helpdesk : SAP Solution Manager
test eee
  
```

The Incident Exchange separator string separates blocks in the Journal, allowing easy identification of new blocks that must be sent. The string is configured in the `ovictex.properties` file (property `sd.incident.informationlog.entry.separator`). The configured value must match the string in the customized Service Manager.

For this customization, all processes in the Document Engine related to Incident updates must be updated with the separator between Journal entries starting with the configured string (default is “----”). In the default Service Manager installation, the following processes are affected:

- `im.save`
- `im.close`
- `im.resolve`
- `im.reopen`
- `im.first`

The Incident Exchange core unload provides an additional process `im.exchange.incident` that already contains the modification. However, this must also be modified if the separator string deviates from the default.

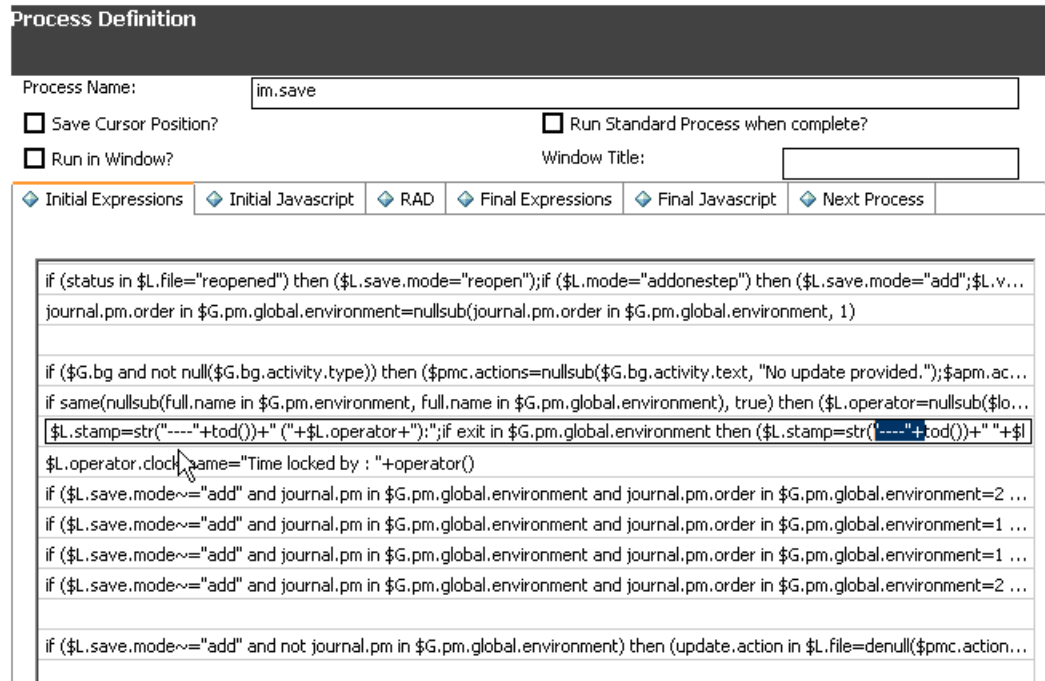
- 1 Select **Tailoring** → **Document Engine** → **Processes**.
- 2 Search for all Processes starting with `im..`

- 3 In Initial Expressions, look for modifications of the Journal timestamp separator (variable `$L.stamp`), and add the configured separator string to the beginning as shown in the following

```
$L.stamp=str("----"+tod())+" ("+$L.operator+):"
```

And the final entire line (**[IM.JOURNAL]** in `code_sm9.txt`) is as follows:

```
$L.stamp=str("----"+tod())+" ("+$L.operator+):";if exit in
$G.pm.global.environment then ($L.stamp=str("----"+tod())+"
"+$lo.time.zone+" ("+$L.operator+):")
```



## Template

When a new Incident is sent from SAP Solution Manager to Service Manager, Incident Exchange creates a new Incident with data for exchanged fields. The Incident management process inside Service Manager may require additional mandatory fields (such as `category`, `subcategory`, and `product type`) that must be filled out in order to submit the Incident. Values for these fields must be provided when the Incident is opened by Incident Exchange. In an uncusomized Service Manager, the Process `im.first` is invoked when an Incident is submitted.

- 1 Select **Tailoring** → **Document Engine** → **Processes**.
- 2 Search for `im.first`.
- 3 Add the following expressions to the Initial Expressions tab of `im.first`.

*On Service Manager 9.x ([IM.FIRST\_INIT] in `code_sm9.txt`):*

```
if (is.incident.exchange in $L.file=true) then (subcategory in
$L.file="access";product.type in $L.file="authorization
error";problem.type in $L.file="incident";assignment in
$L.file="Application";if null(severity in $L.file) then (severity in
```

```

$L.file="4");initial.impact in $L.file="1";site.category in
$L.file="B";action in $L.file={"default description"};affected.item in
$L.file="MyDevices"
$L.comment="siehe JS"
if same(nullsub(full.name in $G.pm.environment, full.name in
$G.pm.global.environment), true) then ($L.operator=nullsub($lo.ufname,
nullsub(operator(), "NULL"))) else ($L.operator=nullsub(operator(),
"NULL"))
$L.stamp=str("----"+tod())+" ("+$L.operator+");if exit in
$G.pm.global.environment then ($L.stamp=str("----"+tod())+"
"+$lo.time.zone+" ("+$L.operator+");)
if (is.incident.exchange in $L.file=true) then (update.action in
$L.file={$L.stamp}+denull(update.action in $L.file))

```

- 4 In addition, if you want to set different default values for incident from different SAP client, you can update the code:

```

if (is.incident.exchange in $L.file=true) then (subcategory in
$L.file="access";product.type in $L.file="authorization
error";problem.type in $L.file="incident";assignment in
$L.file="Application";if null(severity in $L.file) then (severity in
$L.file="4");initial.impact in $L.file="1";site.category in
$L.file="B";action in $L.file={"default description"};affected.item in
$L.file="MyDevices"
to multiple line of
if (is.incident.exchange in $L.file=true and exthd in $L.file="exthd1")
then (category in $L.file="telecoms";subcategory in $L.file="fixed
infrastructure";product.type in $L.file="fixed
infrastructure";problem.type in $L.file="not specified";assignment in
$L.file="AUTO";severity in $L.file="1";initial.impact in
$L.file="1";site.category in $L.file="B";action in $L.file={"default
description"})

```

where exthd1 refers to your SAP client exthd1.

- The Logical field `is.incident.exchange` in `probsummary` is set by Incident Exchange and indicates if the Incident is opened by the Incident Exchange (or some other way). If multiple external helpdesks are connected to Service Manager via the Incident Exchange, the text field `exthd` can be compared with the configured external helpdesk IDs in order to set different values, depending on where the Incident originated from.

## Incident Process

To control the visibility of SAP Solution manager according to SMSAP instance's status, the `im.view.init` and the `im.open.setup` incident processes needs to be customized:

- 1 Log in to Service Manager with a System Administrator account.
- 2 Select **Tailoring** → **Document Engine** → **Processes**.
- 3 Enter **im.view.init** in the Process Name field and click **Search**.
- 4 Add the following expression to the end of the Initial Expressions tab:

```
$SMSAP=nullsub(SMSAP in $G.system.info, "false")
```



- 5 Add the following scripts to the Initial Javascript tab (**[IM.VIEW\_INIT]** in `code_sm9.txt`):

```
var configItem =
lib.smis_ConfigurationManager.getEnabledConfigItem("SMSAP");
if (configItem != null) {
    var solMans = configItem.getParametersByCategory("SolutionManager");

    var values = [];
    var names = [];

    for (var id in solMans) {
        values.push(id);
        names.push(solMans[id]);
    }
    system.vars.$G_solMans_values=values;
    system.vars.$G_solMans_names=names;
}
```

- 6 Click **Save** and **OK**.
- 7 Enter **im.open.setup** to the name field and click **Search**.
- 8 Add the following expression to the Initial Expressions tab:  
\$SMSAP=false
- 9 Click **Save** and **OK**.

## Incident Form

Incident Exchange must be integrated into the incident management workflow. The operator working on the incident must be able to control and trigger Incident Exchange. If more than one external helpdesk is connected to Service Manager, then the target system must be selected.

### Status and Hidden Metadata

The hidden metadata field stores the current Incident Exchange state and Service Manager role (`Requester` or `Provider`). This field determines which actions are currently valid for the Incident. The field is updated by Incident Exchange. Customizations can read but must not write this field.

The Incident exchange state model must be integrated into the Incident workflow. Updates to the hidden metadata field by Incident Exchange can change the Incident status. For example, when a solution has been proposed by SAP Solution Manager, the assigned Service Manager operator must be notified that a new solution is now available for processing. This can be done by inspecting the hidden metadata field and putting the Incident into a special queue if the status has been changed to **SolutionProvided**.

### Exchange History

Incident Exchange keeps a log of all exchange actions and failures. Information from the log and hidden metadata field can be used to explain to the operator what kind of problem occurred.

The probsummary table contains the Array field exchange.history that contains a Structure of date.stamp and history.update. The table can be shown on the Incident Form as a table with two columns. The table can be placed anywhere. In the following example it is placed in a subform in a separate notebook tab named “SAP Solution Manager”. Additional elements (such as a combo box for selection of external helpdesks) can be placed on the notebook tab.

- 1 Open all Incident Forms that are parts of the Incident workflow.

```
IM.open.incident
IM.update.incident
IM.close.incident
```

- 2 Embed the created subform on the Incident form in a new Notebook tab (or in a new section on Service Manager 9.2x and above).

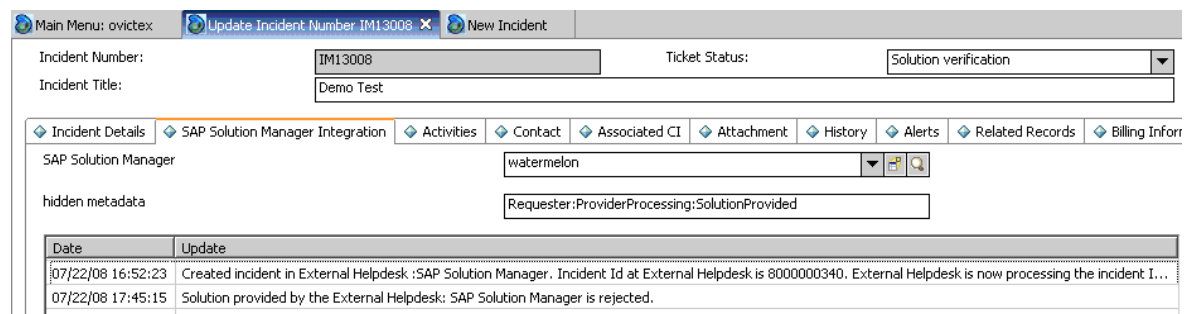
- 3 Add a Notebook tab or section to the following forms:

```
IM.open.incident
IM.update.incident
IM.close.incident
```

Property	Value
Caption	Sap Solution Manager
Visible condition	[\$SMSAP]=true

- 4 add a subform control into the SAP Solution Manager tab or section.

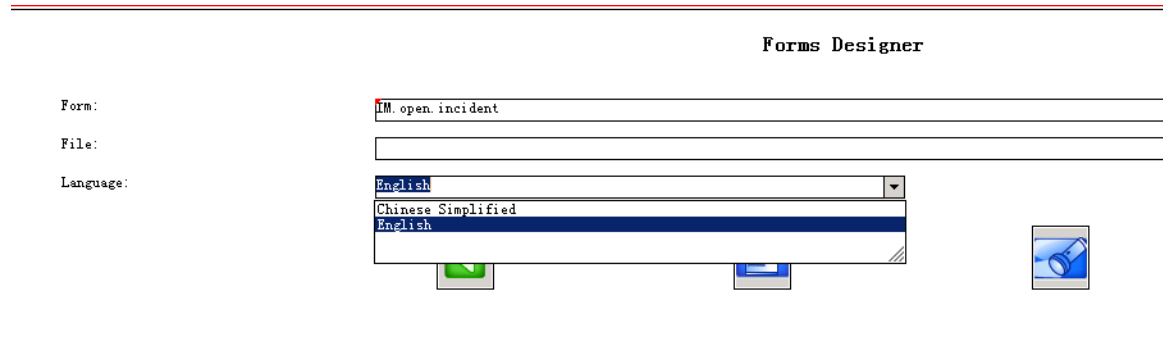
Property	Value
X	1
Y	0
Width	151
Height	28
Format	hp.sap.solution.sub





If multiple Language packs are applied to HP Service Manager, do the following to update the incident related forms for other languages.

- 1 Copy the `hp.sap.solution.sub` form from English to other languages and perform translation.
- 2 Repeat aforementioned [step 3](#) and [step 4](#) for `IM.open.incident`, `IM.update.incident` and `IM.close incident` for other languages.



## Trigger Buttons

The Incident Exchange web service is triggered by an HTTP request from Service Manager. This request is submitted by a JavaScript function in the Script Library. The trigger transmits the

- Incident ID
- Action that triggered the status change
- ID of the external helpdesk

Before triggering an Incident exchange, the Incident must be saved and the `exthd` field set. An `im.exchange.incident` process performs the save operation and invokes the JavaScript trigger function.

A straightforward way is to allow operators to trigger Incident Exchange actions via additional buttons on the Incident form. Instead of pressing **New**, **Save** or **Close**, the operator selects to **Send Incident**, **Add Info** or **Refuse Solution**, and so on. The implementation must follow the Incident Exchange state diagram. Display options that enable or disable the trigger buttons must inspect the value of the hidden metadata to decide which trigger actions are currently available.

An action cannot be triggered multiple times, since the action request is sent asynchronously to the Incident Exchange. The exchange state of the Incident will only be updated (via Event In) during processing with the external helpdesk. Event In can only update the Incident that is not locked. This typically means that the operator has to abandon or refresh the Incident after invoking the Incident Exchange. An exception is the `Addinfo` action, which does not change the exchange state, but only synchronizes updates with the external helpdesk (and can thus be invoked multiple times without updating the Incident in Service Manager.) To block the action buttons after a button has been pressed (and trigger invoked) until the updated incident has been reloaded (including the updated exchange status modified via Event In), evaluate the field `is.ictex.action.blocked`. This field (initially NULL) is set by the trigger process (`im.exchange.incident`) and cleared via `EventIn`.

To set up buttons for the Incident Exchange:

- 1 Select **Tailoring** → **Document Engine** → **States** and search for `im.view`.



The display options are all created after the core unload is imported.

- 2 Connect the newly created Display Options with the provided Process `im.exchange.incident`.

**State Definition**

State:

Display Screen:

Initialization Process:

Format:

Input Condition(view state only):

Display Action	Process Name	Condition	Save First
ess.related	ess.related	\$G.ess	
kmsearch	kmmappedsearch	true	
kmauthor	kmmappedcreation	true	
usesolution	kmdocument.usesolution	not null(\$kmmapsource)	
kmviewsource	kmview.knowledgesource	true	
view.biz.services	view.biz.services	true	
fill	context.pre.fill	true	
massclose	incidents.massclose	true	
runcontext	run.context.wizard	true	
processincident	im.exchange.incident	not null(exthd in \$L.file)	
closeincident	im.exchange.incident	true	
verifyincidentsolution	im.exchange.incident	true	
rejectincidentsolution	im.exchange.incident	true	
addinfo	im.exchange.incident	true	
acceptincidentproces...	im.exchange.incident	true	

Display Action	Process Name	Condition
processincident	im.exchange.incident	not null(exthd in \$L.file)
closeincident	im.exchange.incident	true
verifyincidentsolution	im.exchange.incident	true
rejectincidentsolution	im.exchange.incident	true
addinfo	im.exchange.incident	true
acceptincidentprocessing	im.exchange.incident	true

To disable the duplicate button for exchange incidents:

- 1 Select **Tailoring** → **Tailoring Tools** → **Display Options**.
- 2 Enter `apm.edit.problem_clone` in the Unique ID field and click **Search**.
- 3 Add `and hidden.meta.data in $L.file=NULL` to the end of the Condition field.

## Selection of External Helpdesk System

If Service Manager is connection to multiple SAP Solution Manager helpdesks, then the helpdesk must be selected before initiating Incident Exchange. This could be implemented with new trigger buttons (“Send to SolMan1”, “Send to SolMan2”) or a Combo box on the Incident form. The helpdesk could be automatically selected based on the assigned operator or workgroup (or whatever the Incident workflow requires). If the connection is fixed between one SAP Solution Manager system and Service Manager, then the value can be hardcoded. In any case, the `exthd` field must be set before the Incident is exchanged.

## SAP Configuration Item handling

### Overview

From an SAP perspective, a Configuration Item (CI) is identified by three attributes:

- Installation number
- SID
- Client

Incident Exchange can send the SAP CI information that is attached to an Incident to SAP Solution Manager, and associate an SAP CI with an Incident based on the CI information provided by SAP Solution Manager. In Service Manager, SAP CI's may be modeled and set up in any way, as long as the three identifying attributes are present.

### Implementation

Incident Exchange stores SAP CI information in three fields in the `probsummary` table

- `sap.sid`
- `sap.client`
- `sap.installationnumber`

The Service Manager customization implements the bi-directional synchronization between the `Incident` fields and the Service Manager CIs, allowing the Incident Exchange to be adapted to any existing SAP CI configuration.

### Example Implementation via New Device Type and `fill.fc`

The following describes an example implementation of a new device type `SAPInstance` (created via **Configuration Management** → **Administration** → **Add New Device Type**). This device type needs fields for SAP SID, client and installation number. A new table `SAPInstance` should be created in advance with the following fields:

**Table 5** New `SAPInstance` table fields

Field name	Type	Caption	Other Properties	
SID	Character	SID	Not null	Unique
client	Character	client	Not null	
installation.number	Character	installation.number	Not null	
logical.name	Character	logical.name		Unique

Do the following:

- 1 Generate a new join Def named joinsapinstance in **System Definition** → **Tables** → **joindefs** → **Forms** → **joindefs.g** → **Database Manager**.

Join Table Name:

Common Name:

File Name	Site
device	
SAPInstance	

- 2 Generate a new erddef in **System Definition** → **Tables** → **erddef** → **Forms** → **erddef.g** → **Database Manager**.

erddef x

First Filename:

Second Filename:

Relationship type:

Cascade Deletes?

Casual Relationship?

Distributed Definition?

Field Names from First Filename	Field Names from Second Filename
logical.name	logical.name

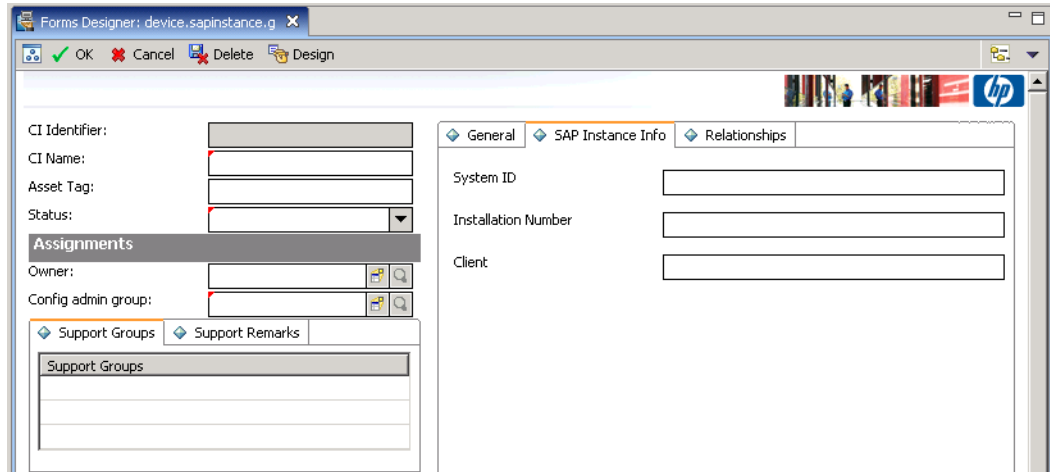
- 3 Create a form view for the device type you want to create in Service Manager.

To create form `device.sapinstance.g` for SAP Solution Manager device type, you can copy an existing form of device. For example,

— *On Service Manager 9.x*

Go to **Tailoring** → **Form Designer**, enter `configurationItem` in the Form field, then click **Search** to open the form in Forms Designer view, and click **Copy/Rename** in the pop-up menu to copy the form, and rename the newly copied form as `device.sapinstance.g`.

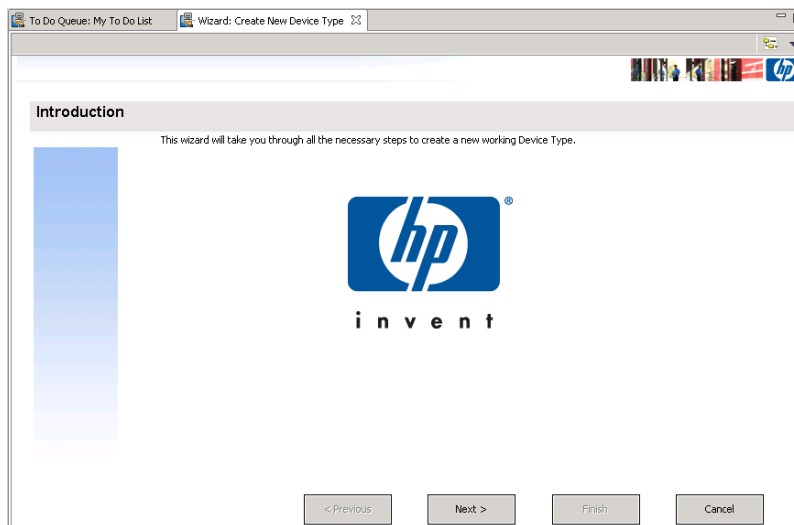
Then add a new tab **SAP Instance Info** in the form, and remove other tabs except for General and Relationships from the form:



The SAP Instance Info tab form should include at least three fields: System ID, Installation Number, and Client.

Control Component	Property	Value
System ID	input	SID
Installation Number	input	installation.number
Client	input	client

- 4 Add a new device type in Service Manager.
  - a Go to **Configuration Management** → **Administration** → **Add New Device Type**. The Introduction page of Wizard: Create New Device Type appears.



- b Click **Next**. Follow the wizard to add a new device type.

**Enter Device Name and Type**

The new device type will be displayed to the user by the Device Type Name and referenced by the system by the Device Type. The Device Type will also be used to create the attribute file and will be included in the building of the join definition name.

Device Type Name:

Device Type:

- c Click **Next**. Enter **device.sapinstance.g** in the View Form field, and then click **Next**.

**Enter the form names**

Please enter the names of the forms to use for the Configuration Item records of this Device Type. If the form does not exist you will be given the option to create one based off of the ICM.device.g and device.example forms.

View Form:

Print Form:

Bulk Update Form:

**Modify the forms associated with the new Device Type**

You may now go to any of the forms which you created in the previous step. These are indicated by the magnifying glass to the right. Any forms which you did not create may either be created later or you may go back and let the wizard create them for you.

View Form (gui):

View Form (text):

Print Form (text):

Bulk Update Form (gui):

- d Click **Next**. Enter **SAPInstance** in Attribute File field.

**Please enter the Attribute filename for this Device Type.**

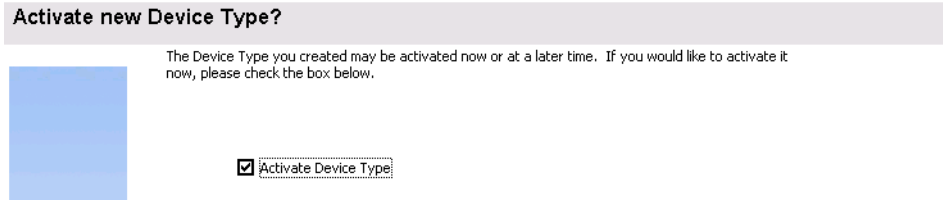
If this Device Type is going to use an attribute file to supplement the standard device information it must be entered here. If you do not wish to use an attribute file you may clear out the field and hit 'Next' to continue.

Attribute File:



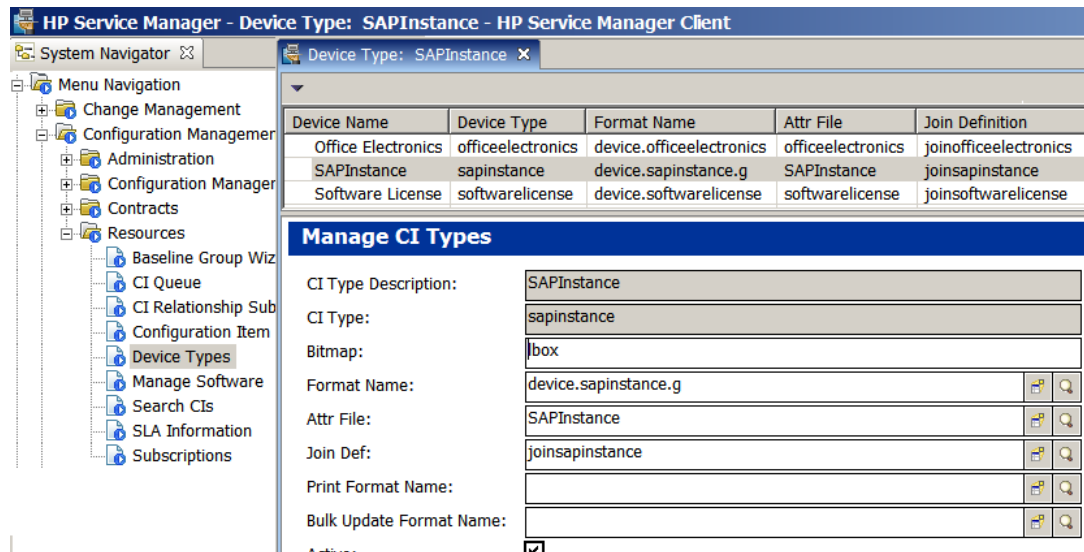


- h Click **Next**. Check the checkbox to activate the device type.

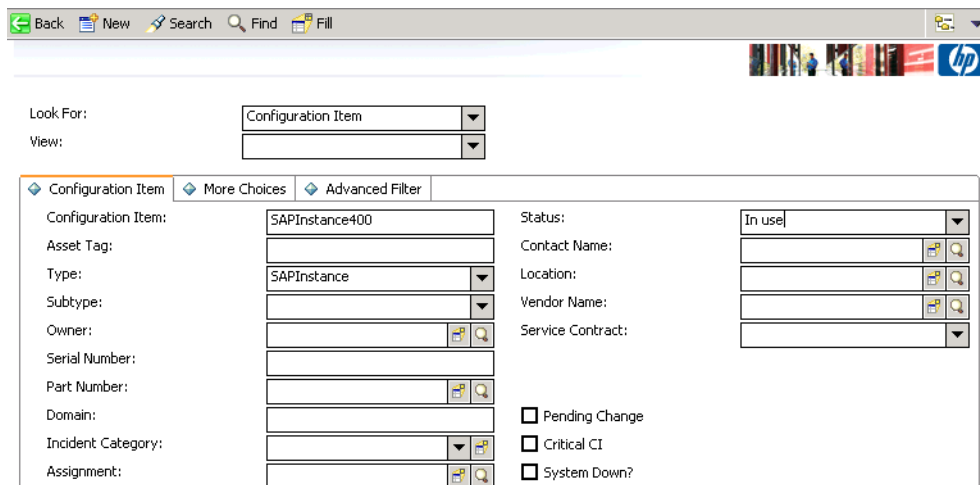


- i Click **Next**. The device type is created.

- 5 The newly generated device type is in **Configuration Management** → **Resources** → **Device Types**.



- 6 Log out and log in to the Windows client again.
- 7 Add a new SAP device item in Service Manager.
  - a Go to **Configuration Management** → **Resources** → **CI Queue** → **New**. Select **SAPInstance** in the Type dropdown list, and provide values for fields of your choices, such as what is shown below.



- b Click **New**. Provide values for the fields needed, and then click **Add** to add the SAP device item in Service Manager.

- c Add relationship for the newly added SAP device item.

Go to **Configuration Management** → **Resources** → **CI Queue** → **New**. Enter the newly added SAP device item name in Configuration Item field, for example, **SAPInstance200**, and then click **Search**.

Select **Relationships** tab. The Relationships tab page appears.

The screenshot shows a software window titled "Configuration Item: SAPInstance200". The "Relationships" tab is selected. The interface is divided into two main sections: "Assignments" and "Classification".

**Assignments:**

- CI Identifier: CI10870
- CI Name: SAPInstance200
- Asset Tag: (empty)
- Status: Planned/On order
- Owner: (empty)
- Config admin group: SAP Support (Asia)
- Support Groups: (empty table)
- Part Number: (empty)

**Classification:**

- CI Type: sapinstance
- CI Subtype: (empty)
- Environment: (empty)
- Security classification: (empty)
- SOX classification: (empty)
- Export control classification: (empty)
- IT service continuity plan enabled
- Critical CI
- Priority: (empty)
- Default Impact: (empty)
- Outstanding Incidents: 0
- Outstanding Problems: 0
- Outstanding Known Errors: 0
- Outstanding Changes: 0
- User Base: (empty)
- System Down
- Pending Change
- Allow Subscribe

Click **Add Upstream Relationship**. The Configuration Item Relationship page appears. Provide values as necessary, and then click **OK**. The relationship is added.

The screenshot shows a software window titled "New Configuration Item Relationship". The "Configuration Item Relationship" section is active.

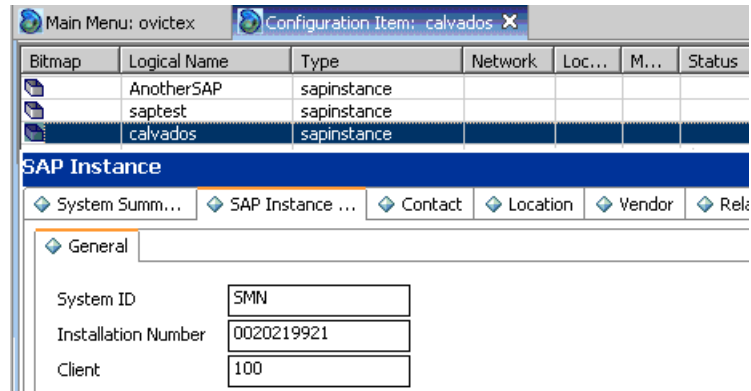
**Configuration Item Relationship:**

- Upstream CI: MyDevices
- Relationship Name: SAPInstance200
- Relationship Type:  Logical,  Physical
- Relationship Subtype: Contains
- Downstream CIs: SAPInstance200

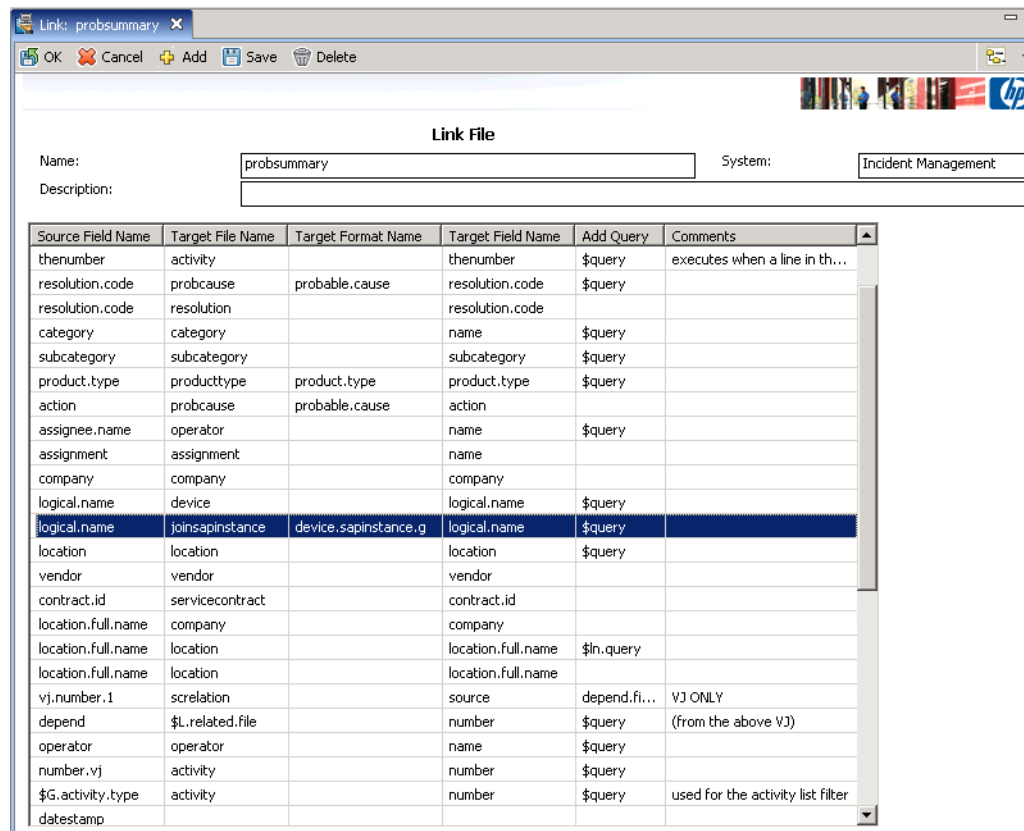
**Outage Dependency:**

- Outage Dependency
- This Configuration Item will be considered down if
- or more of the supporting configuration items are down

- 8 A custom form allows entry of the three attributes (and any other SAP-specific CI attributes) at **Configuration Management** → **Resources** → **CI Queue** → **New**. Select **SAPInstance** in the Type dropdown list.



- 9 The connection between the new device type `SAPInstance` and the fields in the `probsummary` table is made via a link definition. Select **Tailoring** → **Tailoring Tools** → **Links** and search for **probsummary**.
- 10 The existing CI lookup (links between `logical.name` and device have to be modified. Insert an additional link line that links `logical.name` with `joinsapinstance`.



- 11 The three attributes that define the SAP CI and the CI primary key must be linked.

In Expressions tab, add the following two lines  
 ([**LOGICAL.NAME\_JOINSAPINSTANCE\_EXPR**] in `code_sm9.txt`):

```

$fill.recurse=false
if (not null(logical.name in $File)) then
($query="logical.name#\ "+logical.name in $File+"\ ") else
($query="SID=\ "+sap.sid in $File+"\ "+ " and
installation.number=\ "+sap.installationnumber in $File+"\ "+ " and
client=\ "+sap.client in $File+"\ ")

```

Field Name	Value
Field(From/Source)	logical.name
File(To/Target)	joinsapinstance
Format(To/Target)	device.sapinstance.g
Field(To/Target)	logical.name
Query	\$query

Source Field(Fill To/Post From)	Target Field(Fill From/Post To)
sap.sid	SID
sap.installationnumber	installation.number
sap.client	client
sap.incident.type	device.type
logical.name	logical.name

On Service Manager 9.x:

Source Field (Fill...)	Target Field (Fill From/Post To)
sap.sid	SID
sap.installationn...	installation.number
sap.client	client
sap.incident.type	device.type
logical.name	logical.name

12 Click the line with device as Target File Name:

Source Field Name	Target File Name	Target Format Na...	Target Field Name	Add Query
number	schedule		pm.number	
agreement.id	sla		agreement.id	
number.vj	activity		number	
thenumber	activity		thenumber	\$query
cause.code	probcause	probable.cause	cause.code	\$query
resolution.code	probcause	probable.cause	resolution.code	\$query
resolution.code	resolution		resolution.code	
category	category		name	\$query
subcategory	subcategory		subcategory	\$query
product.type	producttype	product.type	product.type	\$query
problem.type	problemtype	problem.type	problem.type	\$query
contact.name	contacts		contact.name	\$query
logical.name	device		logical.name	\$query
location	location		location	\$query

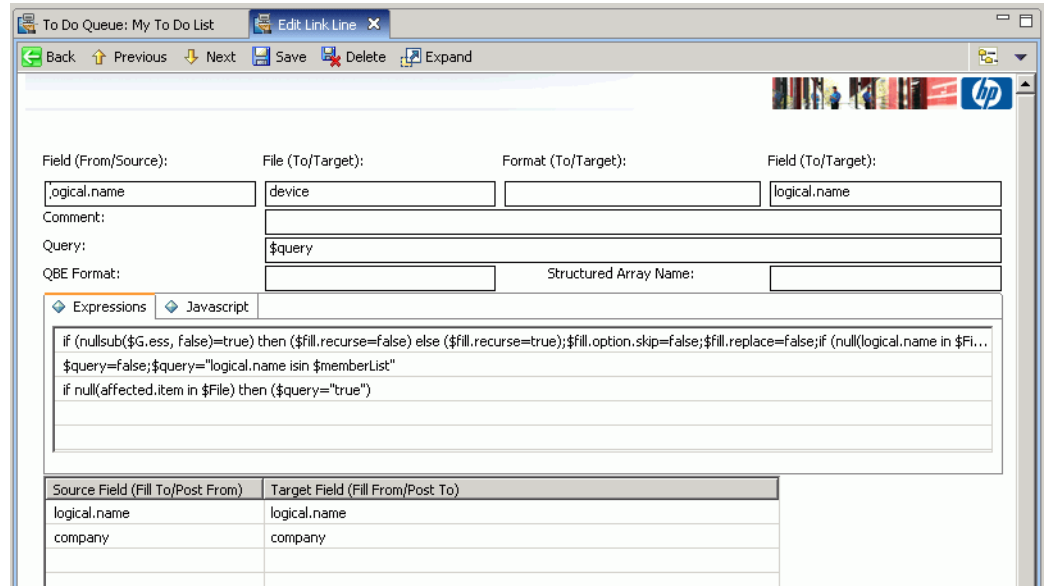
13 Insert the following expression on Service Manager 9.x (**[LOGICAL.NAME\_DEVICE\_EXPR]** in code\_sm9.txt):

```

if (nullsub($G.ess, false)=true) then ($fill.recurse=false) else
($fill.recurse=true);$fill.option.skip=false;$fill.replace=false;if
(null(logical.name in $File) and not null(sap.sid in $File)) then
($fill.skip=true)

```

On Service Manager 9.x:



- 14 In the Post Expressions tab, add the following expression:

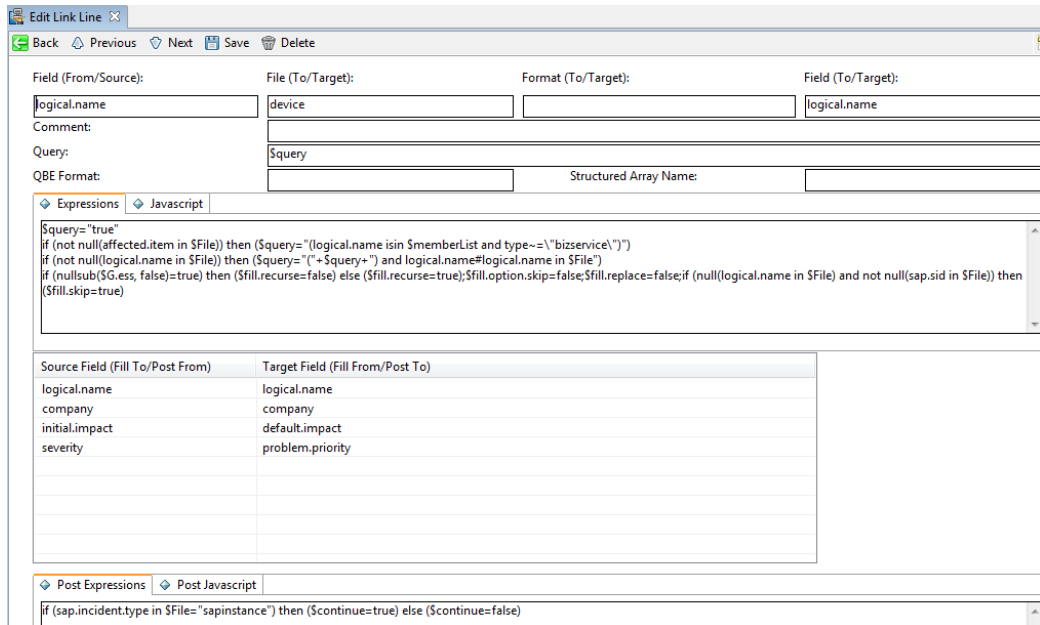
On Service Manager 9.x (**LOGICAL.NAME\_DEVICE\_POST** in code\_sm9.txt):

```

if (sap.incident.type in $File="sapinstance") then ($continue=true) else
($continue=false)

```

On Service Manager 9.x:





- 15 Click the line with `location` as Target File Name:

**Link File**

Name:  System:

Description:

Source Field Name	Target File Name	Target Format ...
cause.code	probcause	probable.cause
resolution.code	probcause	probable.cause
resolution.code	resolution	
category	category	
subcategory	subcategory	
product.type	producttype	product.type
problem.type	problemtype	problem.type
contact.name	contacts	
logical.name	device	
location	location	
logical.name	joinsapinstance	
type	devtype	
serial.no.	device	
model	model	

- 16 In the Expressions tab, replace the first line with the following expression (`[LOCATION_LOCATION_EXPR]` in `code_sm9.txt`):

```
if ($continue=true) then ($fill.recurse=true) else
($fill.recurse=false);$fill.replace=false
```

- 17 In the Post Expressions tab add the following expression:

**cleanup(\$continue)**

Field (From/Source):	rget):	Format (To/Target):	Field (To/Target)
<input type="text" value="location"/>	<input type="text" value="location"/>	<input type="text"/>	<input type="text" value="location"/>
Comment:	<input type="text"/>		
Query:	<input type="text" value="\$query"/>		
QBE Format:	<input type="text"/>	Structured Array Name:	<input type="text"/>

◆ Expressions ◆ Javascript

```
if ($continue=true) then ($fill.recurse=true) else ($fill.recurse=false);$fill.replace=false
if (not null(location in $File)) then ($query="location=\""+location in $File+"\"") else ($query=true)
$fill.recurse.msg=scmsg(5, "fill")
```

Source Field (Fill To/Post From)	Target Field (Fill From/Post To)
location	location
site.category	site.category

◆ Post Expressions ◆ Post Javascript

```
cleanup($continue)
```

- 18 To automatically write the CI attributes into `probsummary` when a CI of type `SAPInstance` is attached to an Incident, the `fill.fc` application is invoked, select **Tailoring** → **Format Control** and search for `probsummary` → **Subroutines**, then right click and select **Show Expanded Form** from the pop-up menu.

Format Control: probsummary

Mass Unload

Name	File Name	System
probsummary		PHD/PM
probsummary.qbe		miscellaneous

OK Back Previous Next Add Save Delete

Forms Queries Calculations JavaScript Validations Subroutines Add Options Privileges

### Format Control Maintenance - Subroutines

Name: probsummary View: short

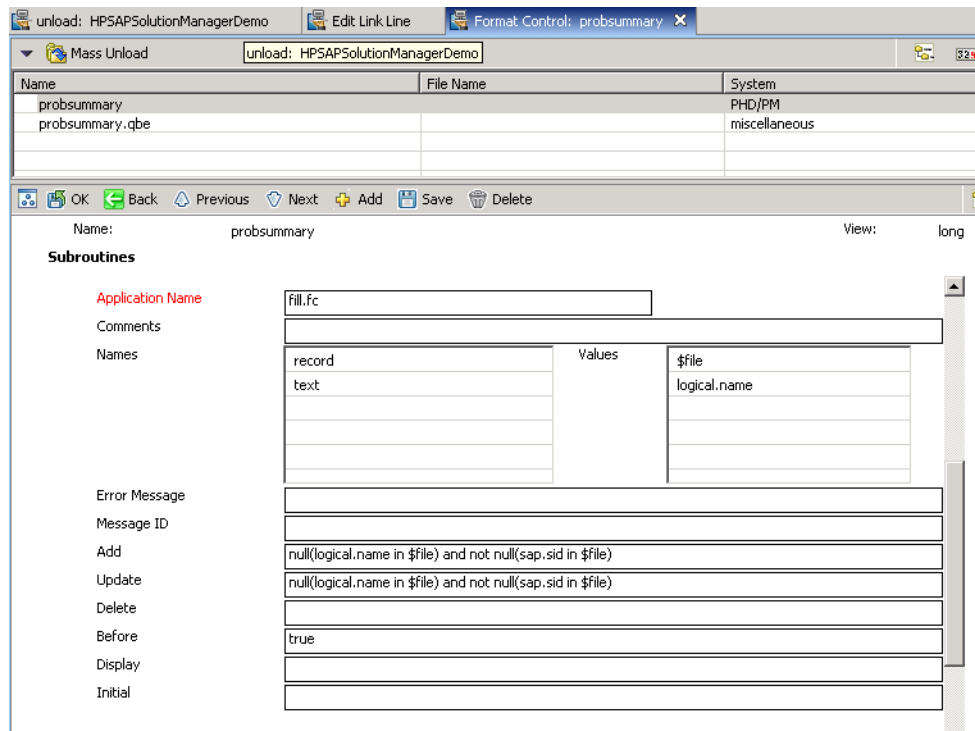
**Subroutines**

Add	Upd	Del	Dis	Initial	Before	Application
Msg ID	Error Message					
	Names			Values		
false					false	us.link
	cannot fill contact information					
	record				\$file	
type in \$!	type in \$!				true	fill.fc
	record				\$file	

- OK F2
- Back F3
- Previous F11
- Next F10
- Add F1
- Save F4
- Delete F5
- Export/Unload
- Insert Line
- Delete Line
- Show Expanded Form
- Forms
- Queries
- Calculations
- Validations
- Additional Options

19 Add a `fill.fc` application, and fill the fields as shown in the screenshot below:

Field	Value
Application Name	fill.fc
Name	record
Value	\$file
Name	text
Value	logical.name
Add ( <b>[FILL.FC]</b> in <code>code_sm9.txt</code> )	<code>null(logical.name in \$file) and not null(sap.sid in \$file)</code>
Update ( <b>[FILL.FC]</b> in <code>code_sm9.txt</code> )	<code>null(logical.name in \$file) and not null(sap.sid in \$file)</code>
Before	true



When an operator assigns a CI of type `sapinstance` to an Incident, the three attributes `SAP SID`, `client` and `installation number` are read from the SAP CI and put into the corresponding Incident fields (which are then exchanged with SAP Solution Manager). Similarly, when an Incident is submitted from SAP Solution Manager, the Incident created in Service Manager contains values in the fields `sap.sid`, `sap.client` and `sap.installationnumber`. These values are used to search for a corresponding CI of type `sapinstance`. If the CI exists, it is automatically attached to the Incident.

## Implementation Alternatives and Enhancements

The above implementation assumes a simple CI setup. SAP CI's may be modeled in more complex ways.

For example, the three attributes `SAP SID`, `client` and `installation number` can be distributed over multiple CI's. A "parent" CI represents the entire SAP system, containing the attributes `SAP SID` and `installation number`, combined with "child" CI's that represent individual clients and contain the `SAP client` attribute. This model allows identification of problems affecting the entire SAP system or just a particular client.

The customization within Service Manager needs to be adapted to a particular SAP CI configuration. Incident Exchange directly interacts only with the `Incident` fields in the `probsummary` table. The synchronization with CI's inside Service Manager is the responsibility of the Service Manager customization.

## Configuring WSDL Mapping

Configure the `IncidentManagement` WSDL Mapping table in WSDL Configuration of Service Manager as follow:

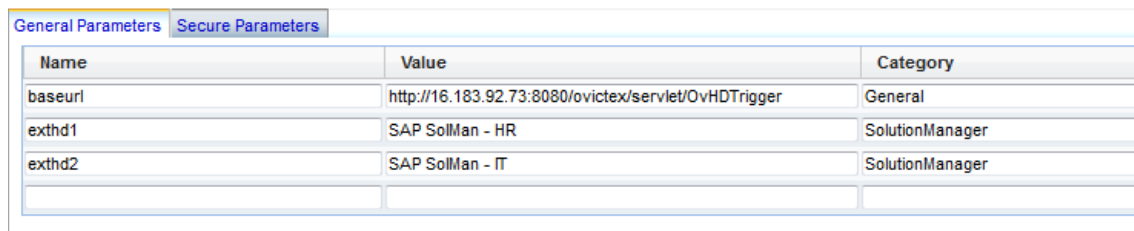
Field name	Caption
<code>action</code>	<code>IncidentDescription</code>
<code>assignee.name</code>	<code>AssigneeName</code>
<code>brief.description</code>	<code>BriefDescription</code>
<code>initial.impact</code>	<code>InitialImpact</code>
<code>assignment</code>	<code>PrimaryAssignmentGroup</code>
<code>product.type</code>	<code>ProductType</code>
<code>resolution</code>	<code>Resolution</code>
<code>subcategory</code>	<code>SubCategory</code>
<code>severity</code>	<code>Urgency</code>

## Adding Instance in SMIS and Configuring Parameters

The SMSAP instance in SMIS integration is used to enable and disable the SMSAP integration. By configuring the SMSAP instance, you can also specify the integration parameters such as SMSSMEX accessing URL and SAP clients. After the SMSAP instance is enabled, the customization to incident is visible and you can select a SAP Solution Manager client for incident information exchange. If the SMSAP instance is disabled or removed, the incident form customization for incident exchange with SAP Solution Manager will be invisible to the end users. Note that the incident information exchange from SAP SolMan to HP Service Manager will not be affected by SMSAP SMIS configuration. For more information about SMIS, refer to *Service Manager Web Help > Integrations > Integration Manager*.

The SM-SAP integration operates only when the SMSAP instance in SMIS is enabled. Do the following to add a new instance in SMIS and configure the parameters:

- 1 Log in to Service Manager with a System Administrator account.
  - 2 Select **Tailoring** → **Integration Manager**. Integration Manager opens.
  - 3 Click **Add** to open the wizard.
  - 4 Select **SMSAP**, click **Next**.
  - 5 Click **Next** to configure the parameters.
    - a Configure the baseurl for connection to SMSSMEX.  
Replace <host> and <port> according to the middleware installation.
    - b Configure SAP Solution Manager clients.  
Add general parameters for each of your SAP Solution Manager clients configured in SMSSMEX `ovictex.properties`. For example, `exthd.instances.id.1`, `exthd.instances.id.2` and so on.
- ▶ Make sure to set your SAP Solution Manager client instance ID in the Name field, and **SolutionManager** in the Category field of each record. Value of each SAP Solution Manager client instance ID is the reference in HP Service Manager.



Name	Value	Category
baseurl	http://16.183.92.73:8080/ovictex/servlet/OvHDTrigger	General
exthd1	SAP SolMan - HR	SolutionManager
exthd2	SAP SolMan - IT	SolutionManager

- 6 Click **Next** twice to the end of the wizard and click **Finish** to save the configurations.
- 7 Select the SMSAP instance and click **Enable** to enable the SM-SAP integration.



# 7 Customizing HP Service Manager in PD Environment

This chapter describes the customization required for HP Service Manager for the integration if the Service Manager Process Designer (PD) Content Pack 9.30.3 is installed.

## Creating a Service Manager User for Web Service

Incident Exchange uses one Service Manager user to connect to Service Manager web services. The user and the user security role should be dedicated for the integration. The user requires the following permissions:

The screenshot shows the 'Rights' configuration window in HP Service Manager. The window title is 'Rights' and it has a menu bar with 'Cancel', 'Save & Exit', 'Save', and 'More'. Below the menu bar is a 'Select a section...' dropdown. The main content area is titled 'Rights' and contains several sections:

- Role:** incident manager
- Area:** Incident
- Permissions:**
  - View
  - New
  - Update:** Always
  - Delete/Close:** Always
- Assignment Groups:** (Empty table)
- Allowed Statuses:** (Empty table)
- Allowed Categories:** (Empty table)
- Expert/Admin/Modify Template:**
  - Expert
  - Admin
  - Modify Template

Do the following:

- 1 Log in to Service Manager with a System Administrator account.
- 2 Select **System Administration** → **Ongoing Maintenance** → **User Quick Add Utility**.
- 3 Type **ovictex**, **INCIDENT EXCHANGE**, **Incident**, **Exchange**, **ovictex@hp.com**.
- 4 Click **Next**.
- 5 For **User to clone** select **Incident.Manager**.
- 6 Click **Finish**.
- 7 Click **OK**.
- 8 Click **System Administration** → **Ongoing Maintenance** → **Operators**, type **ovictex** in the Login Name field, then click **Search**.

- 9 In the Application Profiles tab, change the value in the User Role field to incident manager.
  - ▶ The Security Role is automatically changed to 'incident manager'.
- 10 In the Security tab:
  - a Enter the operator password for Password.
  - b Uncheck Expire Password.
  - c Check Unlimited Sessions.
- 11 In the Startup tab, add SOAP API to Execute Capabilities.
- 12 Click **Save**.

## Importing Customizations Using Unload

This section describes how to configure Service Manager using unload. Additional customization of Service Manager is later required for the integration.

### Core Unload

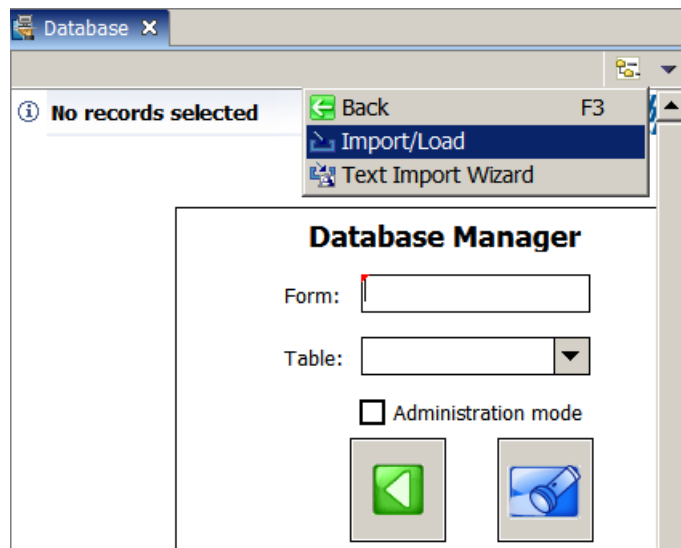
Unloads are used to transfer customizations from one Service Manager installation to another Service Manager installation. The Incident Exchange provides core unloads at the following path:

```
<SMSSMEX1.10p1 Release Package>\unloads\SM9.3\core_sm9.3_pd.unl
```

This unload contains new Service Manager records that are unique to Incident Exchange and do not override any existing Service Manager records.

To import the unload, do the following:

- 1 In the Service Manager client select **Tailoring** → **Database Manager**.
- 2 Select **Import/Load** from the menu.





- 3 Select <SMSSMEX1.10p1 Release Package>\unloads\SM9.3\core\_sm9.3\_pd.unl, and then click **Load FG** to start the import.

## Demo Unload

The demo unload has all the customizations (for an uncustomized, default installation of Service Manager) required to set up a working Incident Exchange for a customer demonstration or evaluation.



Do not import the demo unload into existing development or production systems. The demo unload requires an uncustomized, default installation of Service Manager. The demo unload overrides many standard Service Manager records, and can not be removed or undone.

To import the unload do the following:

- 1 In the Service Manager client, click **Tailoring** → **Database Manager**.

- 2 Select **Import/Load** from the menu.

- 3 Browse to the unload at the following path:

<SMSSMEX1.10p1 Release Package>\unloads\SM9.2\demo\_sm9.3\_pd.unl

- 4 Press **Load FG** to start the import.

## Customizing Demo Unload Manually

This section describes how to manually customize your integration with the demo unload.

### Adding Integration Name to the Info Table

See *Adding Integration Name to the Info Table* on page 34.

### Incident Custom Fields and Web Service Exposure

See *Incident Custom Fields and Web Service Exposure* on page 65.

### Expose Custom Fields in Web Service Interface IncidentManagement

To expose custom fields, do the following:

- 1 Refer to step 1 to step 5 in *Expose Custom Fields in Web Service Interface IncidentManagement* on page 35.

- 2 Click the Expressions tab and add the following expressions:

```
if (is.incident.exchange in $L.file=true and null(number in
$L.file)) then
($L.operator=operator();$L.stamp=str("----"+tod())+"
"+"$L.operator+");if exit in $G.pm.global.environment then
($L.stamp=str("----"+tod())+" "+"$lo.time.zone+"
"+"$L.operator+");update.action in $L.file=insert(update.action
in $L.file, 1, 1, $L.stamp)
```

```

if null(brief.description in $L.file) then (brief.description in
$L.file="SAP checker incident")

if (is.incident.exchange in $L.file=true and null(number in
$L.file)) then (severity in $L.file="4";initial.impact in
$L.file="4";action in $L.file={"default
description"};affected.item in $L.file="MyDevices";assignment in
$L.file="Application")

if (is.incident.exchange in $L.file=true) then
($L.void=jscall("HPSAPTrigger.setCIBySAPInfo", $L.file))

if (hidden.meta.data in $L.file="Closed") then (problem.status in
$L.file="Closed";current.phase in
$L.file="Closure";resolution.code in $L.file="Automatically
Closed")

before:

if (status in $L.file.save~="closed" and status in
$L.file.save~="resolved") then ($apm.activity="external update")

if ($L.need.to.update=true) then ($pmc.actions=NULL)

```

Refer to the [\[INCIDENTMANAGEMENT for PD Content Pack\]](#) section in `code_sm9.txt` for the complete expressions.

## Contacts Web Service Exposure

See *Contacts Web Service Exposure* on page 37.

## Customize Incident Workflow

To customize the incident workflow, do the following:

- 1 Copy the existing incidents workflow to the Incident for SAP workflow.
  - a Select **Tailoring** → **Process Designer** → **Copy Existing Workflow**. The Wizard: Clone a Workflow page opens.
  - b Select Incident and then input values as follow:

Field	Value
New workflow name	Incident for SAP
Copy rule sets	Yes
Rule set prefix	SAP

## Clone a Workflow



Please specify the new workflow name, as well as the prefix for new rule sets if they are to be copied as well.

New workflow name: \* Incident for SAP

Copy rule sets?

Rule set prefix: \* SAP

- c Click **OK**.
- 2 Associate the Incident for SAP workflow with Incident category.
  - a Select **Incident Management** → **Configuration** → **Incident Categories**. The Search Incident Category Records page opens.
  - b Type `incident` in the Name field and click **Search**.
  - c Change the value in the Workflow field to Incident for SAP.

### Incident Category

Name:

Active:

Description:

Workflow: \* Incident for SAP

- d Click **Save**.
- 3 Add actions in different phases.
  - a Select **Tailoring** → **Process Designer** → **Workflows**. The Workflows page opens.
  - b Select Incident for SAP in the list, and then click **Open**. The Workflow: Incident for SAP page opens.
  - c Select a phase.
  - d In the actions tab, click **Add** to add the actions as follows:

Id	Action	Location	Optio...	Action Condition	Action when complete	Requires lock
<input type="checkbox"/>	Add Info	Tray	62	( nullsub(SMSAP in SG.system.info, fal...	Save	false
<input type="checkbox"/>	Send Back	Tray	61	( nullsub(SMSAP in SG.system.info, fal...	Save	false
<input type="checkbox"/>	Send Incident	Tray	60	( nullsub(SMSAP in SG.system.info, fal...	Save	false

<b>Phase</b>	<b>ID</b>	<b>Action</b>	<b>Location</b>	<b>Action Condition</b>	<b>Action when complete</b>	<b>Require lock</b>
Investigation	Send Incident	(SAP) Action for sending to SAP	tray	nullsub(SMSAP in \$G.system.info, false) and nullsub(\$G.ess, false)=false and not nullsub(is.ictex.action.blocked in \$L.file, false) and (hidden.meta.data in \$L.file=NULL or hidden.meta.data in \$L.file="Requester:StartExchange:Error")	save	false
Investigation, Recovery, Review	Send Back	(SAP) Action for sending back	tray	nullsub(SMSAP in \$G.system.info, false) and nullsub(\$G.ess, false)=false and not nullsub(is.ictex.action.blocked in \$L.file, false) and (hidden.meta.data in \$L.file="Provider:ProviderProcessing" or hidden.meta.data in \$L.file="Provider:ProviderProcessing:Error" or hidden.meta.data in \$L.file="Requester:RequesterProcessing" or hidden.meta.data in \$L.file="Requester:RequesterProcessing:Error")	save	false
Investigation, Recovery, Review	Add Info	(SAP) Action for adding info	tray	nullsub(SMSAP in \$G.system.info, false) and nullsub(\$G.ess, false)=false and not nullsub(is.ictex.action.blocked in \$L.file, false) and hidden.meta.data in \$L.file~=NULL and hidden.meta.data in \$L.file~="Requester:StartExchange:Error" and hidden.meta.data in \$L.file~="Closed"	save	false

Recovery	Refuse Solution	(SAP) Action for rejecting solution	tray	nullsub(SMSAP in \$G.system.info, false) and nullsub(\$G.ess, false)=false and not nullsub(is.ictex.action.blocked in \$L.file, false) and (hidden.meta.data in \$L.file="Requester:SolutionProvided" or hidden.meta.data in \$L.file="Requester:SolutionProvided:Error")	save	false
Recovery	Send Solution	(SAP) Action for sending solution	tray	nullsub(SMSAP in \$G.system.info, false) and nullsub(\$G.ess, false)=false and not nullsub(is.ictex.action.blocked in \$L.file, false) and (hidden.meta.data in \$L.file="Provider:ProviderProcessing" or hidden.meta.data in \$L.file="Provider:ProviderProcessing:Error" )	save	false
Review	Close SAP Incident	(SAP) Action for closing incident	tray	nullsub(SMSAP in \$G.system.info, false) and nullsub(\$G.ess, false)=false and not nullsub(is.ictex.action.blocked in \$L.file, false) and (hidden.meta.data in \$L.file)#"Requester" and hidden.meta.data in \$L.file~="Requester:StartExchange:Error"	save	false

- e Click **Save**.
- 4 Add or update the transitions between phases.
  - a Select **Tailoring** → **Process Designer** → **Workflows**. The Workflows page opens.
  - b Select Incident for SAP in the list, and then click **Open**. The Workflow: Incident for SAP page opens.
  - c Add transition between phases as follows:

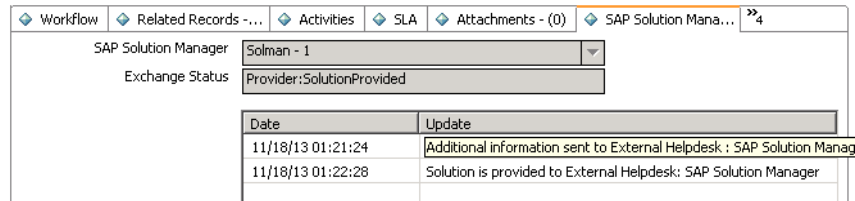
From phase	To phase	Add/update	Transition type	Condition	Rule sets
Recovery	Closure	Add	Automatic	( hidden.meta.data in \$L.file="Closed" )	SMSAP.im.set.status.closed
Review	Closure	Add	Automatic	( hidden.meta.data in \$L.file="Closed" )	SMSAP.im.set.status.closed
Review	Recovery	Add	Automatic	( hidden.meta.data in \$L.file="Provider:ProviderProcessing" )	SMSAP.im.back.to.Recovery
Recovery	Closure	update	Manual	Add to the end: and hidden.meta.data in \$L.file=NULL	No change
Review	Closure	update	Manual	Add to the end: and Hidden Meta Data in CurrentRecord is NULL	No change

- d Click **Save**.
- 5 Add a SAP Solution Manager tab for the following formats:

```

im.incident.categorization
im.incident.investigation
im.incident.recovery
im.incident.review
im.incident.closure

```



- a Click **Tailoring** → **Forms Designer**.
- b Type `im.incident.categorization` and click **Search**.
- c Add a new notebook tab with the following configurations:

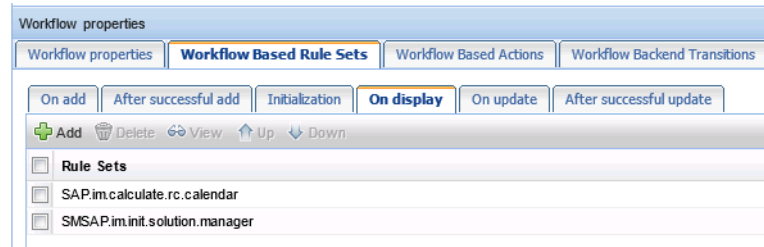
Property	Value
Name	sapsolutionmanager
Visible	True
Visible condition	[\$SMSAP]=true
Caption	Sap Solution Manager

- d Add a subform control to the SAP Solution Manager tab.

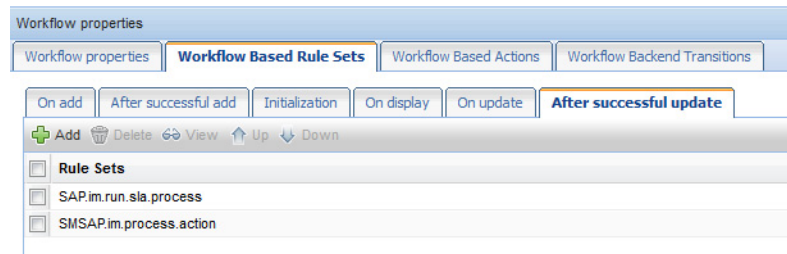
Property	Value
Format	hp.sap.solution.sub

- e Click **Save**.
  - f Search for `im.incident.investigation`, `im.incident.recovery`, `im.incident.review`, and `im.incident.closure` respectively and repeat [step c](#) to [step e](#).
- 6 Control the visibility of SAP Solution Manager and init solution managers, and then execute the SAP Solution Manager integration related Actions.
    - a Select **Tailoring** → **Process Designer** → **Workflows**. The Workflows page opens.
    - b Select Incident for SAP in the list, and then click **Open**. The Workflow: Incident for SAP page opens.
    - c Click Workflow properties, and then click the Workflow Based Rule Sets tab.
    - d Click the On display tab.

- e Click **Add**, and then select `SMSAP.im.init.solution.manager` in the list.

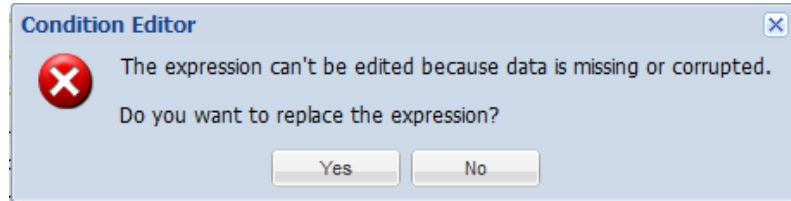


- f Click the After successful update tab.
- g Click **Add**, and then select `SMSAP.im.process.action` in the list.

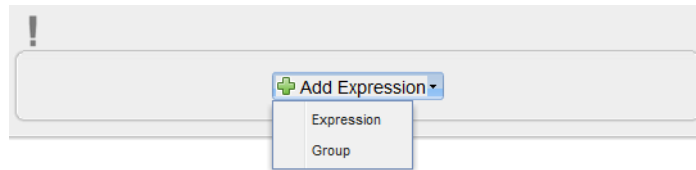


- h Click the On add tab.
  - i Click **Add**, and then select `SMSAP.IM.SET.SAPINFO` in the list.
  - j Click the On update tab, and then repeat the previous step.
  - k Click **OK** and **Save**.
- 7 To disable the copy record action for the SM-SAP integration:
- a Select **Tailoring** → **Process Designer** → **Workflows**. The Workflows page opens.
  - b Select Incident for SAP in the list, and then click **Open**. The Workflow: Incident for SAP page opens.
  - c Select a phase such as Categorization, and then click the Action tab.
  - d Append the following script to the end of the condition of the "copy record" action.  
and hidden.meta.data in \$L.filed~=NULL
  - e Repeat [step c](#) and [step d](#) for the other phases, including Investigation, Recovery and Review.

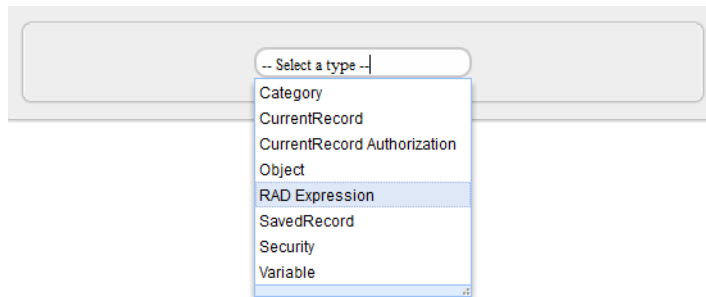
- ▶ If you encounter the following error when opening the "copy record" condition, click **Yes** and follow the steps below to replace the old expression.



- 1 Click **Add Expression** and then select **Expression** in the drop-down list:

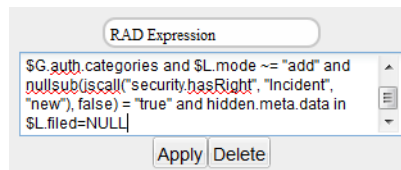


- 2 Select **RAD Expression**.



- 3 Copy and paste the following expression to the input field:

```
$G.ess=false and category in $L.file isin $G.auth.categories and $L.mode ~= "add" and nullsub(jscall("security.hasRight", "Incident", "new"), false) = "true" and hidden.meta.data in $L.file=NULL
```



- 4 Click **Apply**.

## SAP Configuration Item Handling

### Overview

See *Overview* on page 45.



## Implementation

See *Implementation* on page 45.

### Example Implementation via New Device Type and fill.fc

This section describes an example implementation of a new device type `SAPInstance` (created via **Configuration Management** → **Administration** → **Add New Device Type**). This device type needs fields for SAP SID, client and installation number. A new table `SAPInstance` should be created in advance with the following fields:

**Table 6** New `SAPInstance` table fields

Field name	Type	Caption	Other Properties	
SID	Character	SID	Not null	Unique
client	Character	client	Not null	
installation.number	Character	installation.number	Not null	
logical.name	Character	logical.name		Unique

Do the following:

- 1 Refer to *step 1* to *step 10* as described in *Example Implementation via New Device Type and fill.fc* on page 45.
- 2 The three attributes that define the SAP CI and the CI primary key must be linked.

In the Expressions tab, add the following scripts:

(**[LOGICAL.NAME\_JOINSAPINSTANCE\_EXPR]** in `code_sm9.txt`):

```
$fill.recurse=false
```

```
$query="true"
```

```
if (not (null(affected.item in $File))) then ($query="(logical.name isin  
$memberList and type~=\"bizservice\")")
```

if (not (null(logical.name in \$File))) then (\$query="("+ \$query+") and logical.name=logical.name in \$File")

Field Name	Value
Field(From/Source)	logical.name
File(To/Target)	joinsapinstance
Format(To/Target)	device.sapinstance.g
Field(To/Target)	logical.name
Query	\$query

Source Field(Fill To/Post From)	Target Field(Fill From/Post To)
sap.sid	SID
sap.installationnumber	installation.number
sap.client	client
sap.incident.type	device.type
logical.name	logical.name

Field (From/Source): logical.name      File (To/Target): joinsapinstance      Format (To/Target): device.sapinstance.g      Field (To/Target): logical.name

Comment: \_\_\_\_\_

Query: \$query

QBE Format: \_\_\_\_\_      Structured Array Name: \_\_\_\_\_

Expressions      Javascript

```
$fill.recurse=false
$query="true"
if (not (null(affected.item in $File))) then ($query="(logical.name isin $memberList and type~="bizservice")")
if (not (null(logical.name in $File))) then ($query="("+ $query+") and logical.name=logical.name in $File")
```

Source Field (Fill To/Pos...)	Target Field (Fill From/Post To)	Sort Field(To/Target):
sap.sid	SID	_____
sap.installationnumber	installation.number	
sap.client	client	
sap.incident.type	device.type	
logical.name	logical.name	

Sort Order(To/Target): \_\_\_\_\_

Post Expressions      Post Javascript

\_\_\_\_\_

- Refer to step 12 to step 17 as described in *Example Implementation via New Device Type and fill.fc* on page 45.

## Implementation Alternatives and Enhancements

See *Implementation Alternatives and Enhancements* on page 60.

# Configuring WSDL Mapping

Configure the IncidentManagement WSDL Mapping table in WSDL Configuration of Service Manager as follow:

Field name	Caption
action	IncidentDescription
assignee.name	AssigneeName
brief.description	BriefDescription
initial.impact	InitialImpact
assignment	PrimaryAssignmentGroup
product.type	ProductType
resolution	Resolution
subcategory	SubCategory
severity	Urgency

## Adding Instance in SMIS and Configuring Parameters

The SMSAP instance in SMIS integration is used to enable and disable the SMSAP integration. By configuring the SMSAP instance, you can also specify the integration parameters such as SMSSMEX accessing URL and SAP clients. After the SMSAP instance is enabled, the customization to incident is visible and you can select a SAP Solution Manager client for incident information exchange. If the SMSAP instance is disabled or removed, the incident form customization for incident exchange with SAP Solution Manager will be invisible to the end users. Note that the incident information exchange from SAP SolMan to HP Service Manager will not be affected by SMSAP SMIS configuration. For more information about SMIS, refer to *Service Manager Web Help > Integrations > Integration Manager*.

The SM-SAP integration operates only when the SMSAP instance in SMIS is enabled. Do the following to add a new instance in SMIS and configure the parameters:

- 1 Log in to Service Manager with a System Administrator account.
- 2 Select **Tailoring** → **Integration Manager**. Integration Manager opens.
- 3 Click **Add** to open the wizard.
- 4 Select **SMSAP**, click **Next**.
- 5 Click **Next** to configure the parameters.

- a Configure the baseurl for connection to SMSSMEX.

Replace <host> and <port> according to the middleware installation.

- b Configure SAP Solution Manager clients.

Add general parameters for each of your SAP Solution Manager clients configured in SMSSMEX `ovictex.properties`. For example, `exthd.instances.id.1`, `exthd.instances.id.2` and so on.



Make sure to set your SAP Solution Manager client instance ID in the Name field, and **SolutionManager** in the Category field of each record. Value of each SAP Solution Manager client instance ID is the reference in HP Service Manager.

Name	Value	Category
baseurl	http://16.183.92.73:8080/ovictex/servlet/OvHDTrigger	General
exthd1	SAP SolMan - HR	SolutionManager
exthd2	SAP SolMan - IT	SolutionManager

- 6 Click **Next** twice to the end of the wizard and click **Finish** to save the configurations.
- 7 Select the SMSAP instance and click **Enable** to enable the SM-SAP integration.

# 8 Configuring SAP Solution Manager

This chapter describes how to configure the SAP Solution Manager.

## Prerequisites

The prerequisites are:

- SAP Solution Manager 7.0 SP 12 (or higher) or SAP Solution Manager 7.1
- SAP Solution Manager SP12 if copying of business transaction SLFN for customization in a customer name space (for example ZLFN) is required
- SAP Solution Manager SP12. Required to copy a business transaction into a customer name space for customization (for example, to copy business transaction SLFN into customer name space ZLFN)
- Configured SAP Solution Manager Service Desk

Configured SAP Solution Manager Service Desk SSL encryption between SAP Solution Manager and Apache Tomcat requires:

- Sapcryptolib 5.5.5C or higher
- SSL Server and SSL Client PSE
- SSL Server and SSL Client certificates trusted against a CA

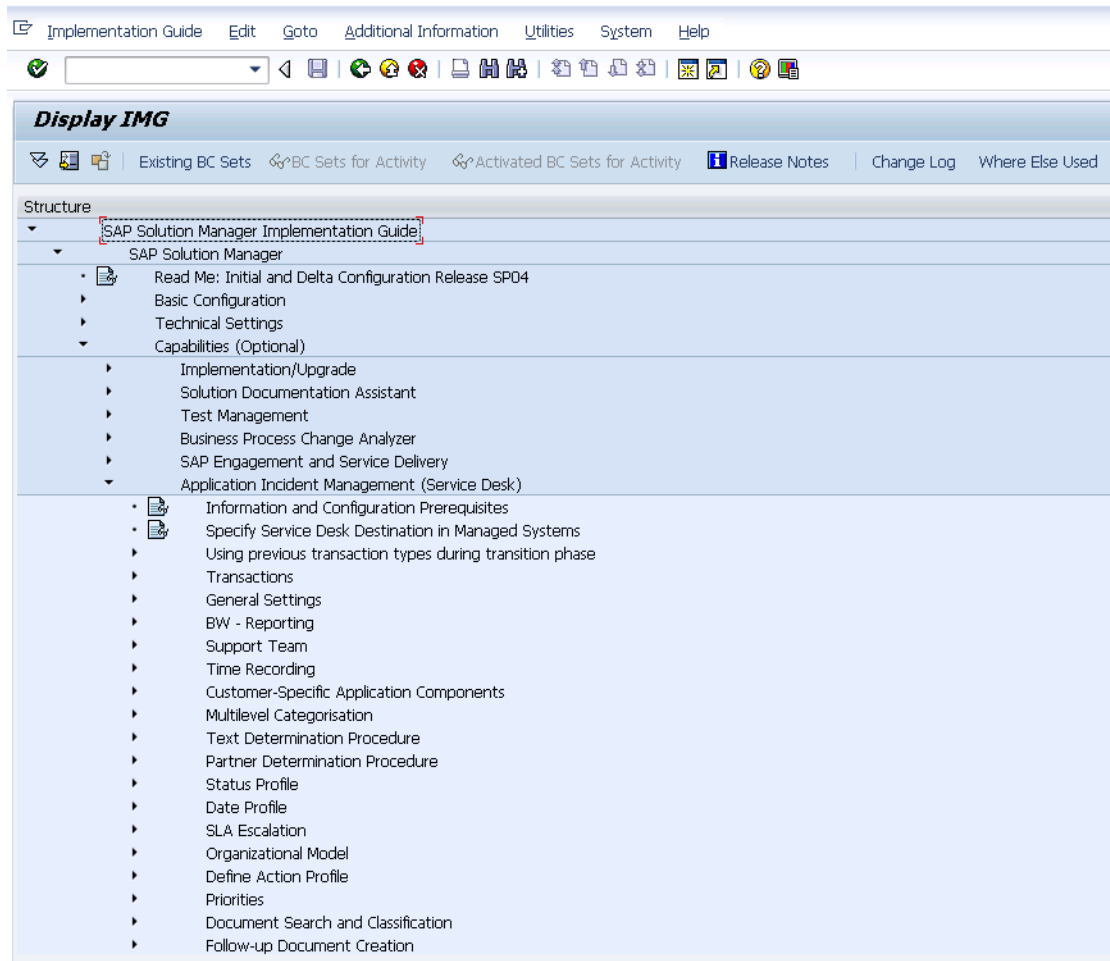
Integration with HP Service Manager requires implementation of the latest SAP notes (SAP application area SV-SMG-SUP-IFA) for the SP level stack of SAP Solution Manager. The following diagram shows the search results of SAP notes in the SAP Support Portal.

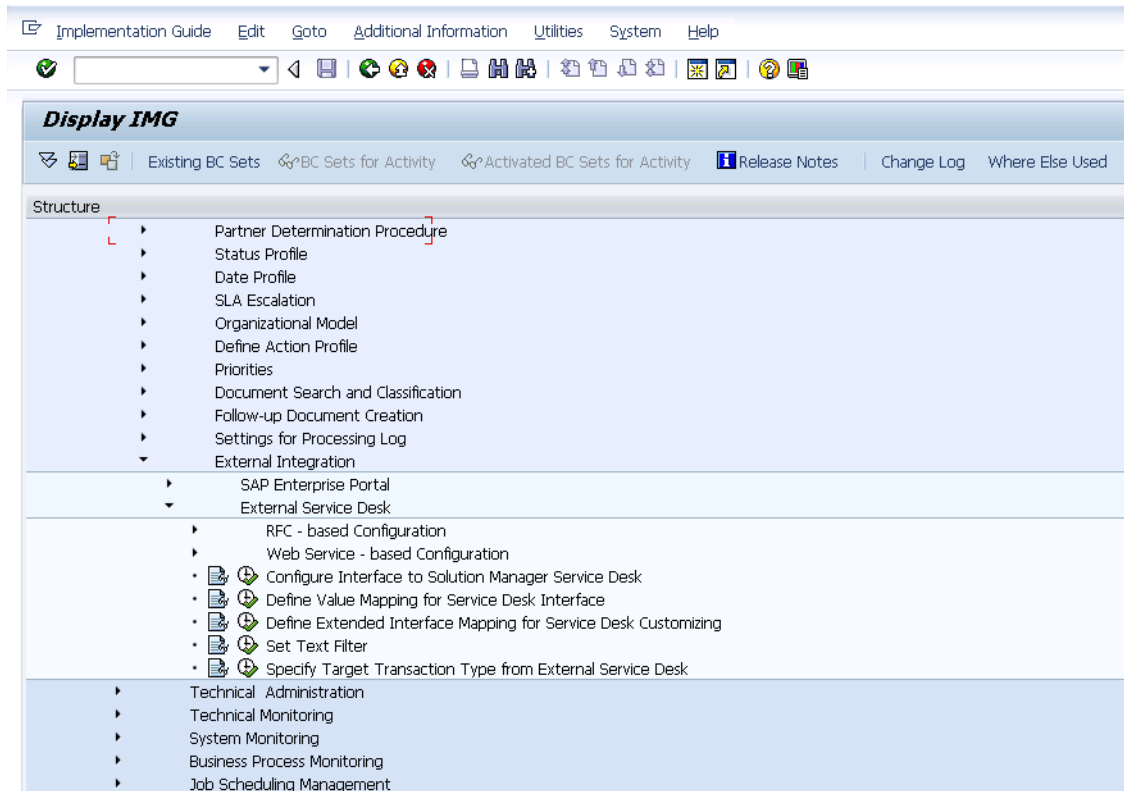
The screenshot shows the SAP Support Portal interface. The search results for SAP notes are displayed in a table. The table has columns for Ranking, Application Area, Number, Short text, and Last Changed On. The search results show 21 SAP Notes found, with the first 9 results listed below.

Ranking	Application Area	Number	Short text	Last Changed On	
1.	1.000	SV-SMG-SUP-IFA	1123416	No status change for proposed solution	11.12.2007
2.	1.000	SV-SMG-SUP-IFA	926682	Error handling in partner interface	10.12.2007
3.	1.000	SV-SMG-SUP-IFA	1091156	Dates are only transferred with time zone GMT	11.09.2007
4.	1.000	SV-SMG-SUP-IFA	1089075	No status change for message transfer	05.09.2007
5.	1.000	SV-SMG-SUP-IFA	1088339	System transfers texts of documents incorrectly	05.09.2007
6.	1.000	SV-SMG-SUP-IFA	1078629	User data is not transferred for texts and attachments	13.08.2007
7.	1.000	SV-SMG-SUP-IFA	1078622	Notes are not transferred correctly	08.08.2007
8.	1.000	SV-SMG-SUP-IFA	1054007	Problems with iBase components and assigned notes	11.05.2007
9.	1.000	SV-SMG-SUP-IFA	1050675	Problems with customer-defined actions	02.05.2007

# Configuring SAP Solution Manager External Service Desk Interface

SAP provides the Implementation Guide “External Service Desk” for configuring the external help interface. The Implementation Guide is located in SAP transaction /nspro under path \SAP Solution Manager ImplementationGuide\SAP Solution Manager\Capabilities (Optional)\Application Incident Management (Service Desk)\External Integration\ External Service Desk. The following diagrams show the Implementation Guide for configuring the connection to the external Service Desk.



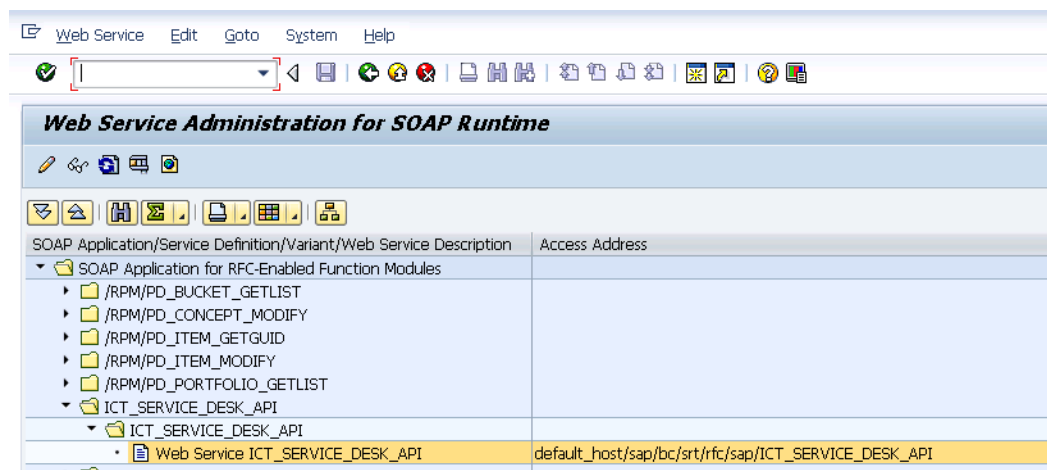


Click the leftmost text sign to view configuration steps. Click the clock sign to enter the corresponding transaction and edit the configuration.

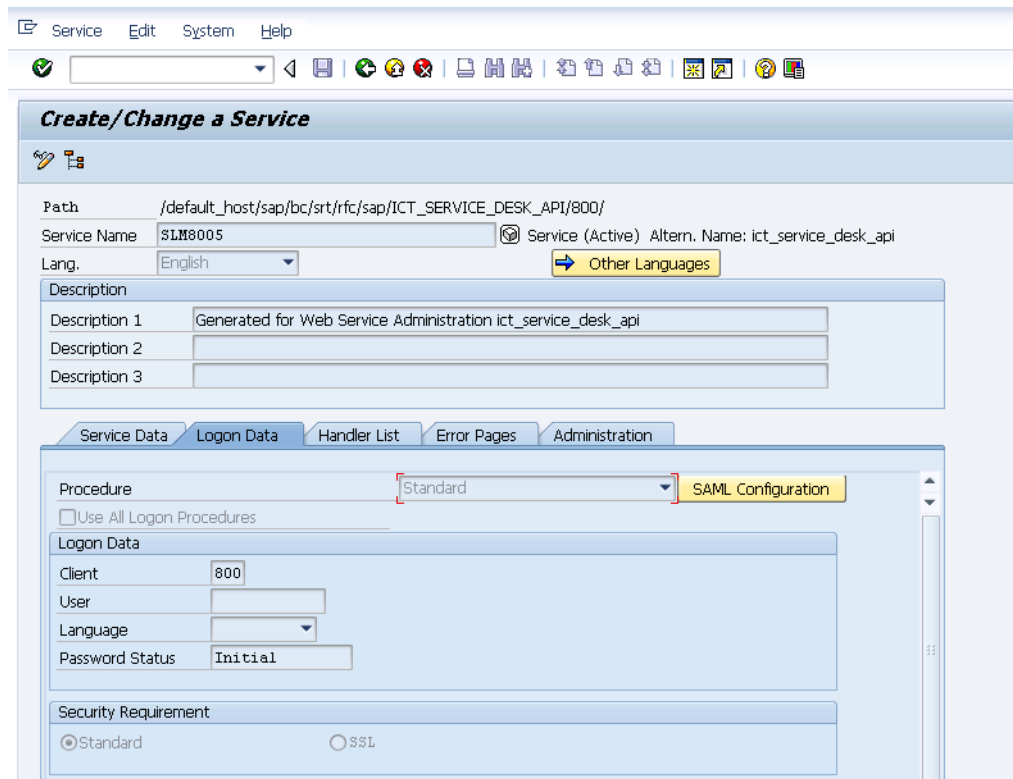
## Release Web Service

The Incident Exchange Web Service is deactivated by default. It is required to release the WEB-Service in the Internet Communication Manager Service tree.

- 1 The transaction to release a Web Service is **/nwsconfig**. After the Web Service is released the WS is in SAP transaction **/nwsadmin**.



- 2 To determine the logon procedure of the Web Service for incoming requests, go to SAP transaction **/nsicf**.
- 3 Enter **ICT\_SERVICE\_DESK\_API** as the service name.
- 4 Click **Execute** to execute the search.
- 5 Double-click the Service to edit or navigate to the path **/default\_host/sap/bc/srt/rfc/sap/** and select **ICT\_SERVICE\_DESK\_API**.
- 6 In the Logon tab of Create/Change a Service dialog , select **Standard**.



➤ The security section of this manual contains additional information for setting up SSL communications. Adding a user is not required. The Incident Exchange Web Service will use the user and password that is configured in the properties file for HTTP Basic authentication. This user must exist as an SAP user. It is not recommended to use a dialog user for this purpose.

## Assign Roles to the Communication User

Configure an SAP user with permission to manage incidents in SAP Solution Manager Service Desk. Follow the instruction in the Implementation Guide and add the roles **SAP\_SUPPDESK\_PROCESS** and **SAP\_SUPPDESK\_INTERFACE** to the user. Exchanging a business partner with a default configuration interface requires the additional role **SAP\_CRM\_BUSINESS\_PARTNER**.

To configure a user:

- 1 Select transaction **/nsu01**.
- 2 Input the name of the user.



3 Click **Display** . The user configuration transaction appears.

- ▶ • A person who is assigned to an incident in HP Service Manager but does not exist in Solution Manager will be created as a Business Partner when the incident is forwarded to Solution Manager. Without the business partner role `SAP_CRM_BUSINESS_PARTNER` the incident can not be created or updated in Solution Manager and the error code 99 appears.
- A communication user is recommended, but not necessary.

Sending support messages to SAP AGS requires assigning an SAP Support Portal contact to Solution Manager users who will communicate with the SAP Support Portal via RFC connections. The contact maintained corresponds to the S-user in the SAP Support Portal without “S”. See SAP Note 834534 and the SAP Solution Manager configuration guide for details of Solution Manager roles and authorizations.

## Create HTTP Connection

Define the endpoint of the SMSSMEX Web-Service for communication between SAP Solution Manager and Apache Tomcat.

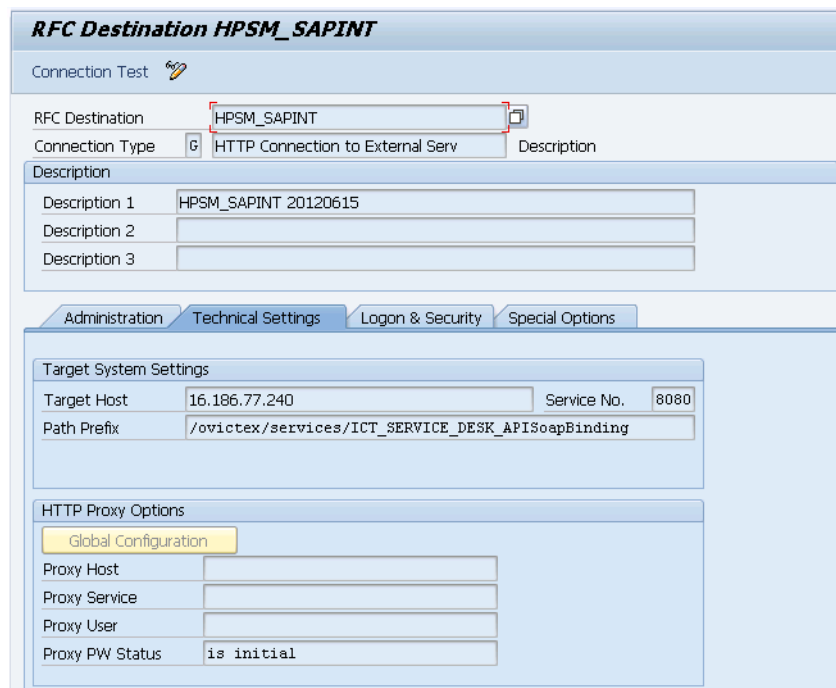
- 1 Select transaction `/nsm59`.
- 2 Create an RFC destination of type **G** (HTTP connection to external server).
- 3 Go to the tab **Technical settings** and specify the endpoint of the SMSSMEX Web-Service. The default is:

Target Host: `<host>`


Service No: `<port>`

Path Prefix: `/ovictex/services/ICT_SERVICE_DESK_APISoapBinding`

- 4 Add the endpoint in the RFC destination. Your network configuration may require specification of a proxy. The following example shows the RFC destination for host `itsamqavm130`.



**RFC Destination HPSM\_SAPINT**

Connection Test 

RFC Destination

Connection Type **G** HTTP Connection to External Serv Description

Description

Description 1

Description 2

Description 3

Administration Technical Settings Logon & Security Special Options

Target System Settings

Target Host  Service No.

Path Prefix

HTTP Proxy Options

Global Configuration

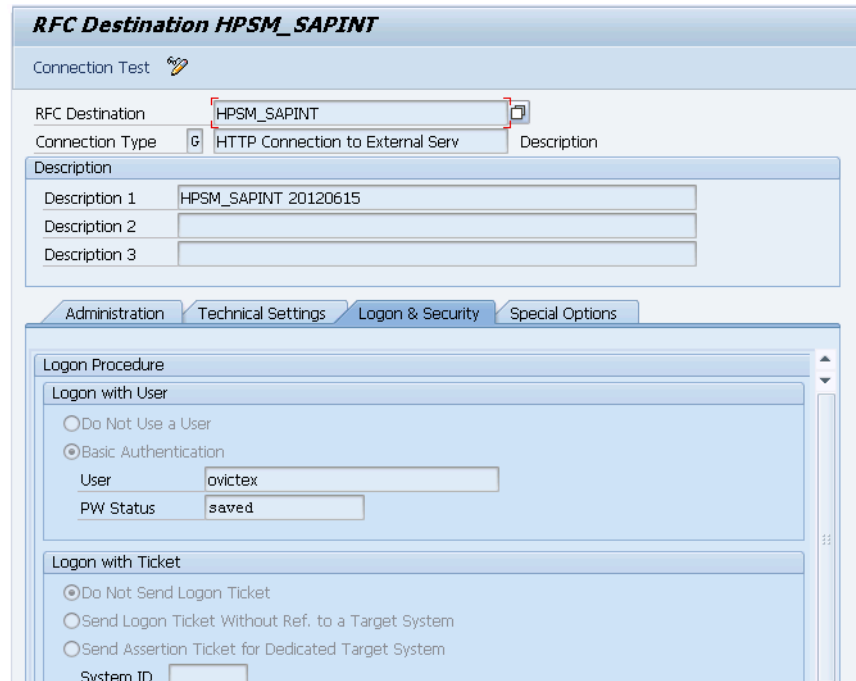
Proxy Host

Proxy Service

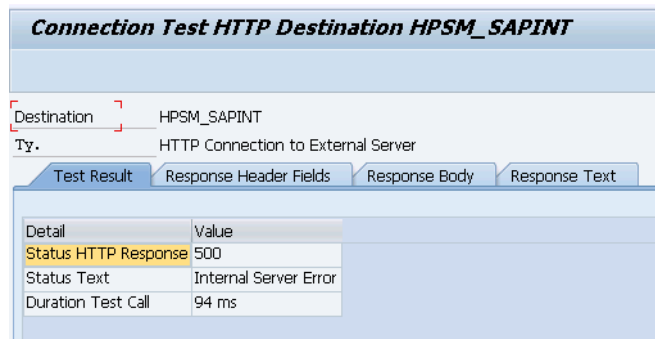
Proxy User

Proxy PW Status

- 5 In the Logon & Security tab define the security settings for outgoing requests. Select **Basic Authentication** for HTTP basic authentication. Add the user and password specified in **ovictex.properties** for HTTP basic authentication. The more secure SSL communication configuration is described in the security chapter of the manual. You can also select **No Logon** which is the default selection for “Logon&Security”.



The following diagram shows the SMSSMEX Web service returning error 500. This result indicates the connection between SAP and SMSSMEX is established.

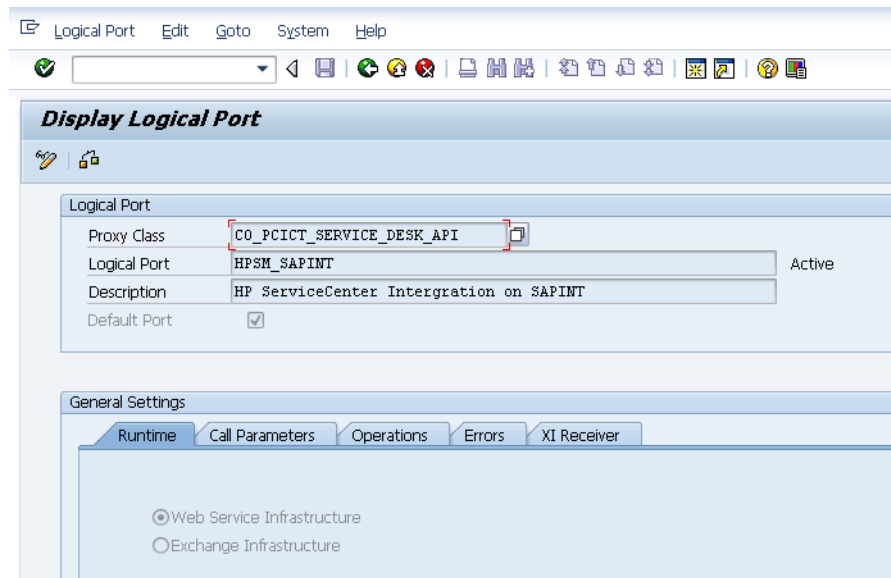


## Create a Logical Port

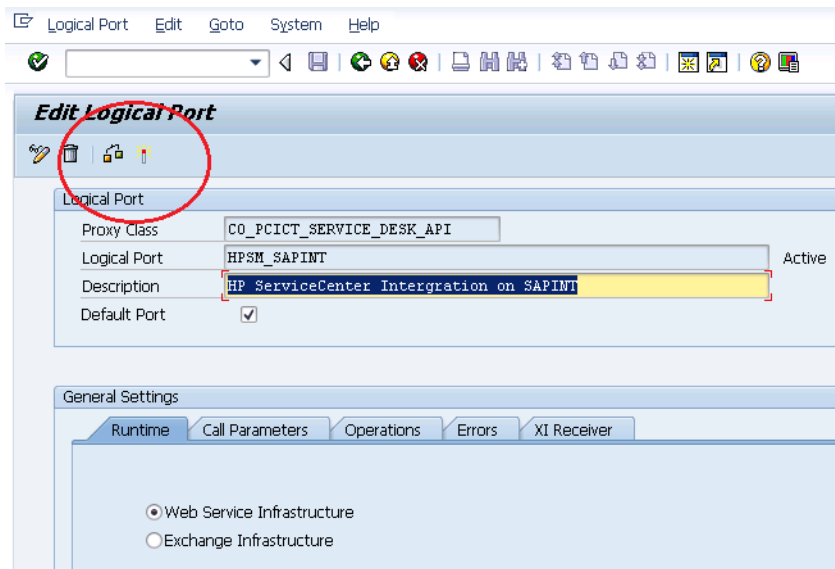
The logical port is the container that encapsulates the outgoing requests. Define the logical port as specified in the Implementation Guide instructions.

- 1 Go to transaction **/nlpconfig**.
- 2 Select **CO\_PCICT\_SERVICE\_DESK\_API** as the Proxy Class name.

- 3 In Call Parameters tab add the HTTP destination configured in the previous chapter.



- The port must be activated. Click **Active** to activate the logical port.



## Configure Interface to SAP Solution Manager Service Desk

This activity configures the interface between the SAP Solution Manager service desk and the HP Service Manager. Follow the instructions in the Implementation Guide. The configuration requires that Apache Tomcat and the web service are configured and running. In this implementation step the SMSSMEX web service must deliver a unique Service Desk ID. If the Service Desk ID is changed, then the configuration must be repeated.

Configure SAP Solution Manager Service Desk Interface							
External Service Desk	Logical Port	RFC Destination	Active/Inac.	External Service Desk Type	Get Reporter	Service Desk ID	Keep in sync
HP Service Manager	HPSM_SAPINT		Active	Standard	New BP if none with same E-mail address	4601A679504F479EA6E07C3FF28A16C7	<input type="checkbox"/>

Use the **Check** button to verify the configuration. Any error message will be displayed in the output window. Use transaction `/nictconf` to jump to configuration transactions.

- ▶ Do not select the **Keep in sync** checkbox when configuring the interface to SAP Solution Manager service desk.
- ▶ If the check fails, try **Generate Default Mapping** → **Overwrite Old Values** and then run the check again. After configuration, click **Save** to save the configured interface.

## Define Value Mapping for the Service Desk Interface

This IMG activity configures the value mapping between SAP Solution Manager Service Desk and SMSSMEX for ingoing and outgoing requests. Changing the default value mapping of the SAP Solution Manager is not required. If changes are necessary, use the field mapping file of the SMSSMEX configuration file. To change the default Mapping of the SAP Solution Manager, consult the instructions in the implementation guide.

## Define Extended Interface Mapping for Service Desk

If SAP Solution Manager Service Desk is highly customized (not using standard SAP objects) then it might be necessary to change the interface mapping. The IMG activity instructions provide more information.

## Get SAP Solution Manager Service Port

Go to SAP transaction `/nsmicm`. Select **Goto** → **Services**.

### ICM Monitor - Service Display

Active Services							
No.	Log	Service Name/Port	Host Name	Keep Alive	Proc.Timeo	Actv	External Bind
<input type="checkbox"/>	1	HTTP 8003	gomorrah.deu.hp.com	30	60	✓	
<input type="checkbox"/>	2	HTTPS 8001	gomorrah.deu.hp.com	30	60	✓	

This transaction shows the host and the port required for access to the SAP Solution Manager Service Desk web service. Specify in `ovictex.properties` the host/port as the endpoint entry.

# Solution Manager Tracing

SolutionManager is able to trace incoming and outgoing web-service XML messages. The messages can be downloaded and used for failure analysis.

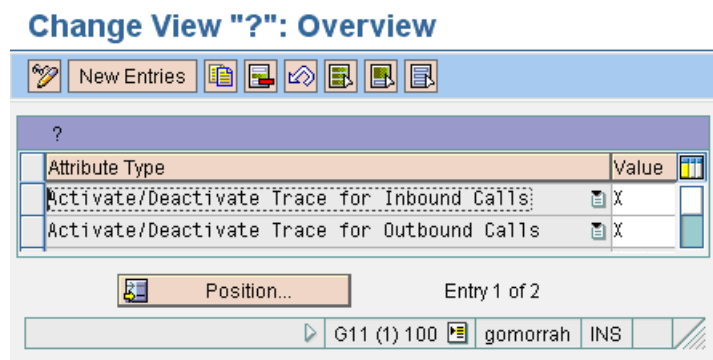
## Enable tracing

To enable tracing, implement a SolutionManager Implementation Guide in transaction / **nspro**.

```
spro -> SAP Reference IMG ->
      SAP SolutionManager Implementation Guide ->
      SAP SolutionManager ->
      Configuration ->
      Scenario-Specific Settings ->
      Service Desk ->
      Connecting an External Service Desk ->
      Define Extended Interface Mapping for Service Desk Customizing
```

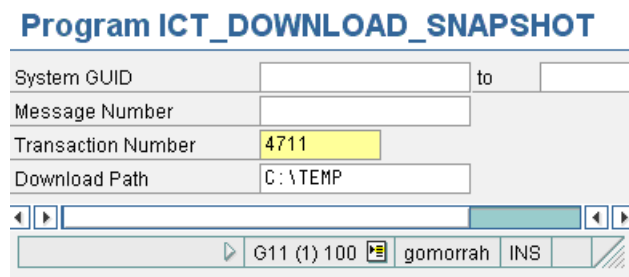
Add new entries to enable tracing for incoming and outgoing calls:

- Activate/Deactivate Trace for Inbound Calls = **X**
- Activate/Deactivate Trace for Outbound Calls = **X**



## Download Trace File

To download the trace file, run **ict\_download\_snapshot** in transaction /**nse38**. Enter the SolutionManager incident id in the field Transaction Number and run the program (**F8**). The trace file will be downloaded to the local computer (for example, incident 4711 traces will be downloaded to C:/TEMP).





# 9 Configuring Security

This chapter describe the required security configuration settings.

## Security between SAP Solution Manager and Tomcat

This section describes the security configuration between SAP Solution Manager and Tomcat.

### Configure SAP Solution Manager for SSL

This section describes how to configure SAP Solution Manager for SSL.

#### Checking SAP SSL Configuration

SAP WEB AS does not support or allow self-signed certificates for communication between Solution Manager and the SMSSMEX Web Service. All certificates must be trusted against a CA.

Before configuring SSL for the External Help Desk interface, check if the WEB AS that hosts the SAP Solution Manager is configured for using SSL.

ICM (Internet Communication Manager) HTTPS service is required for SSL communication. Check if SSL communication is possible in SAP transaction `/nsmicm` (select menu entry **GOTO** and select **Services** or press **SHIFT+F1**).

If SSL communication is possible then an active HTTPS service that is listening to a port is visible. In the example below, the HTTPS port is 8001. This port must be configured in the SMSSMEX web service properties file.

#### ICM Monitor - Service Display

Active Services						
No.	Log	Service Name/Port	Host Name	Keep Alive	Proc.Timeo	Actv External Bind
<input type="checkbox"/>	1	HTTP 8003	gomorrah.deu.hp.com	30	60	✓
<input type="checkbox"/>	2	HTTPS 8001	gomorrah.deu.hp.com	30	60	✓

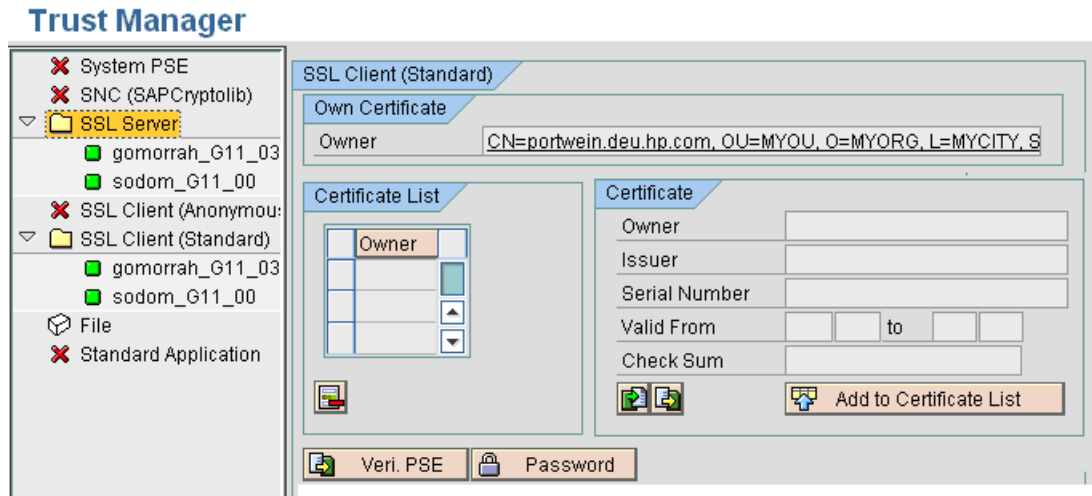
If an HTTPS service in the ICM monitor is not visible, then check the SSL Server configuration in Trust Manager. Start the Trust Manager with SAP transaction `/nstrust`.

If the the PSE entries SSL Server and SSL Client (Standard) are not shown in the Trust Manager status section, then install and configure the SAP `sapcryptolib` library.

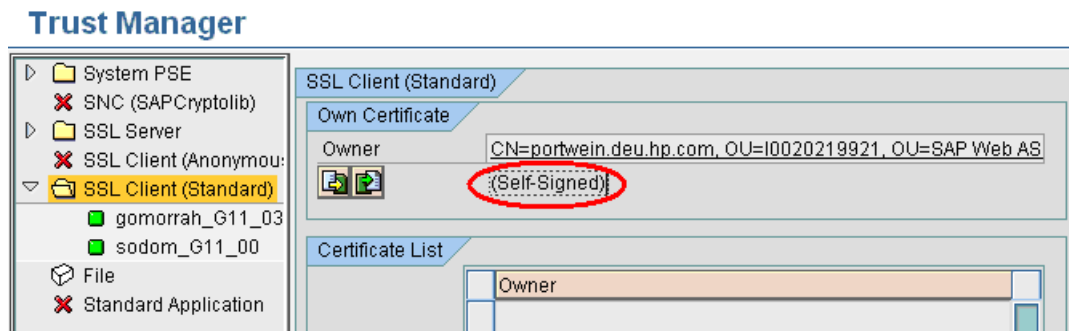


Installing and configuring `sapcryptolib` requires a restart of the SAP WEB AS instance. The installation instructions are in the SAP online help. For more information, see [Appendix B, Installing and Configuring SAPCRYPLIB](#).

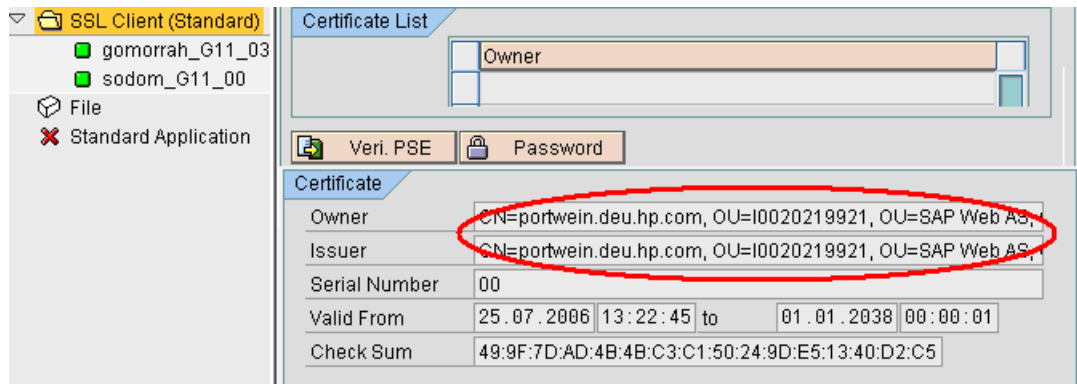
The following diagram shows the Trust Manager with the created PSE “SSL Server” and “SSL Client (Standard)”. The red X in front of the other PSE's indicates that the PSE's have not been created. The PSE “SSL Server” and “SSL Client (Standard)” must be created.



In the next diagram the certificate of the PSE “SSL Client (Standard)” is “Self Signed”. Self-signed certificates are not supported for communication with Apache Tomcat (the certificate must be signed against a CA). If the certificate is signed the 'Self signed' certificate text will disappear.



Check the certificate by double-clicking the Owner attribute. The certificate details are shown in the Certificate section. If the Owner and Issuer have the same DN the certificate is self-signed.

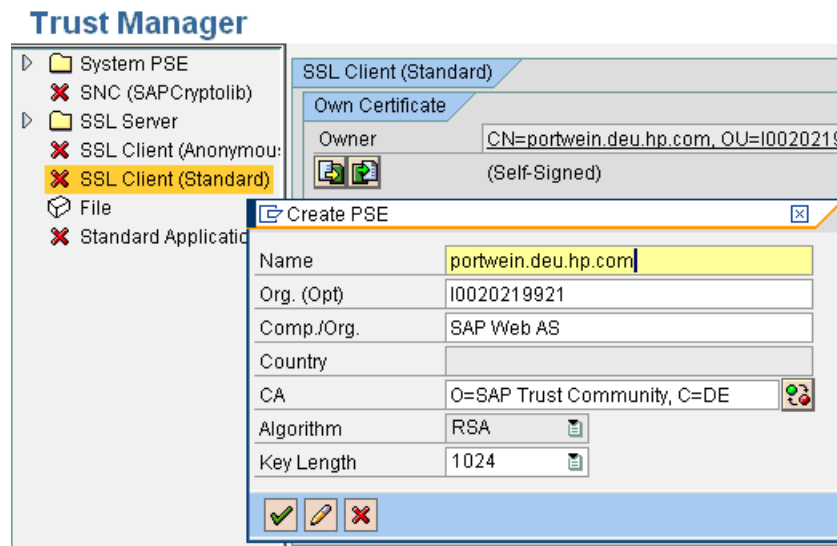




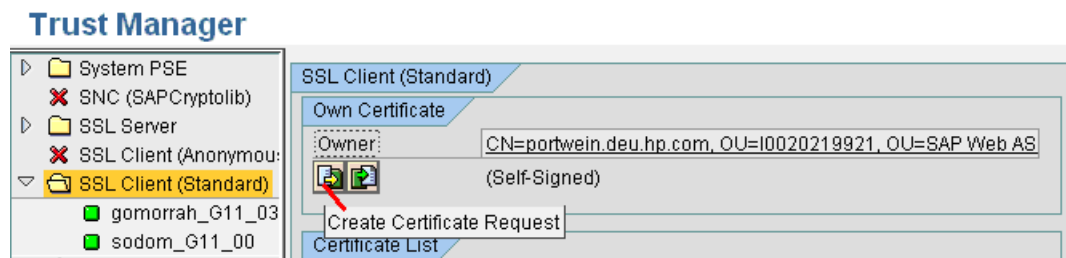
## Creating a Client PSE in Trust Manager

To create a client PSE in Trust Manager, do the following:

- 1 Start the Trust Manager.
- 2 Select the **PSE SSL Client (Standard)** in the status section of the Trust Manager.
- 3 Click **Create**.



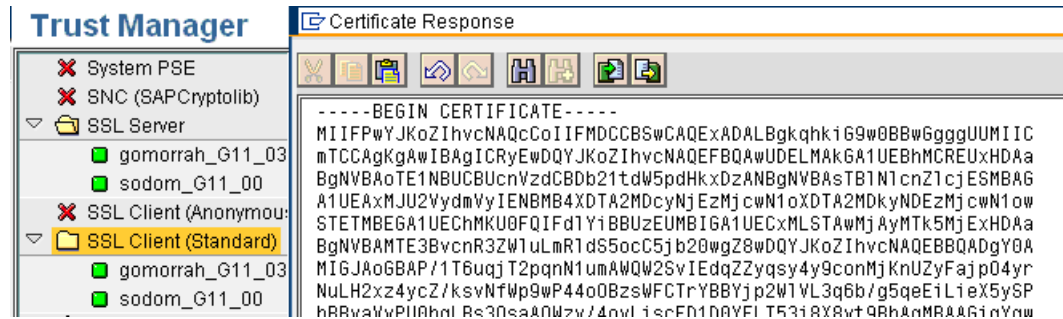
- 4 For the CN (Name) enter the fully qualified hostname of the SAP WEB AS system. All other entries must not be changed. The key length should be 1024.
- 5 Save the settings.
- 6 Double click **SSL Client (Standard)** in the status section. The Own certificate in the Own Certificate section is shown.
- 7 Click **Create Certificate Request**.



- 8 The Certification Request is shown. Copy the request to the Clipboard.



- Copy the Certificate response to your client PSE.



The certificate is now trusted against a CA. The required steps are different for other CA's. Contact the Trust Center for details. A certificate for the SSL Server PSE is also required.

## Setting Up an Outgoing Connection in SAP Solution Manager

The outgoing connection from SAP Solution Manager to HP Apache Tomcat must be configured in SAP transaction `/nsm59`. Add a new or change an existing HTTP RFC destination with type G.


- In SAP transaction `/nsm59` configure the HTTPS port of the Tomcat Server. A redirect from the HTTP port to the HTTPS port of Apache Tomcat will not work with the SAP WEB AS. The HTTPS port is defined in the `server.xml` configuration file of the Tomcat Server.
- The SSL configuration of Apache Tomcat is switched off by default. Enable the configuration.
- In the settings for the SSL HTTP connector, set the Tomcat default port for SSL communication to **8443**.
- The diagram below shows the example configuration of the RFC Destination (in the Target Host field enter the server name (case sensitive) instead of the IP address).

### RFC Destination HPSC\_SMCINT-HELENHTTPS

Connection Test	
RFC Destination	HPSC_SMCINT-HELENHTTPS
Connection Type	G HTTP Connection to External Serv
Description	
Description 1	HP SERVICE CENTER integration to solution manager
Description 2	
Administration   <b>Technical Settings</b>   Logon & Security   Special Options	
Target System Settings	
Target Host	helen2006.asiapacific.hpqcorp.net
Service No.	8443
Path Prefix	/ovictex/services/ICT_SERVICE_DESK_APISoapBinding
HTTP Proxy Options	
Global Configuration	
Proxy Host	
Proxy Service	
Proxy User	
Proxy PW Status	is initial

- 5 In the Logon&Security tab of the RFC configuration define the logon procedure and the security protocol. Basic authorization with SSL communication and certificates is not supported by Apache Tomcat. Set the Logon Procedure to **No Logon**.
- 6 In the security protocol status enable SSL and select a PSE from the certification list. SAP provides PSE “ANONYM SSL Client” and “DEFAULT SSL Client (Standard)”.

**RFC Destination HPSC\_SMCINT-HELENHTTPS**

Connection Test 

RFC Destination: HPSC\_SMCINT-HELENHTTPS

Connection Type: G HTTP Connection to External Serv Description

Description

Description 1: HP SERVICE CENTER integration to solution manager

Description 2:

Administration Technical Settings **Logon & Security** Special Options

Security Options

Logon Procedure

No Logon

Basic Authentication

Send SAP Logon Ticket

Status of Secure Protocol

SSL  Inactive  Active

SSL Client Certificate: DEFAULT SSL Client (Standard) Cert. List

Authorization for Destination:

Logon

User:

PW Status: is initial

- 7 Check with the SAP Basis Administrator what client PSE should be used. In most cases this will be the PSE “SAP Client (Standard)”.
- 8 After assigning a client Certificate to the RFC destination, save the settings. The RFC destination is configured for using SSL with Apache Tomcat. A connection test will fail if the Server certificate in Apache Tomcat is not trusted against a CA.
- 9 Create a logical port (see [Create a Logical Port](#) on page 82).
- 10 Configure the interface between the SAP Solution Manager Service Desk and the HP Service Manager for the SSL outgoing connection (see [Configure Interface to SAP Solution Manager Service Desk](#) on page 84).

## Set up an Incoming Connection in SAP Solution Manager

Configure the incoming connection in the ICF Service tree in SAP transaction `/nsicf`.

- 1 In SAP transaction `/nsicf` enter `ICT_SERVICE_DESK_API` as service name.
- 2 Execute the search of the service.
- 3 Double-click the Service to edit (or navigate to `/default_host/sap/bc/srt/rfc/sap/` and select `ICT_SERVICE_DESK_API`).

- 4 Open the **Create/Change a Service** dialog.
- 5 In the Logon tab select **Required with client Certificates (SSL)**.
- 6 Save the settings. Service **ICT\_SERVICE\_DESK\_API** is configured for SSL connection only. In this procedure the lowest possible security level is specified. If “Required with Logon Data” is configured, then connecting via SSL and the client certificate is allowed.

## Create/Change a Service

The screenshot shows the SAP 'Create/Change a Service' dialog. The 'Logon Data' tab is selected. The 'Procedure' dropdown menu is open, showing options: Standard, Alternative Logon Procedure, Required with Logon Data, and Required with Client Certificate (SSL) (highlighted). Other fields include: Path (/default\_host/sap/bc/srt/rfc/sap/), Service Name (SMD20025), Lang. (English), Description 1 (Web Service ICT\_SERVICE\_DESK\_API), Client (100), User, Password, Language, and Password Status (Initial).

► For SSL communication, ensure that the ICM uses HTTPS.

Define the user mapping to the DN of the Certificate. The different ways of mapping are described in the SAP online help. Defining a user mapping to a DN is described below.

- 7 In SAP transaction **/nse16** open the view **VUSREXTID** (enter **VUSREXTID** in the table Name field).
- 8 Select the Work Area **DN of Certificate X.500**.
- 9 In the user mapping dialog, as an external ID add the DN of the client certificate of Apache Tomcat (see *Create Keystore and Truststore* on page 94). Specify the exact DN of the certificate. For example:  
CN=helen2006.asiapacific.hpqcorp.net, OU=TEST, O=GDCC, L=SH, SP=CN, C=CN
- 10 For **Seq. No.** enter **000, 001...** (for internal use only).
- 11 Assign the SAP user for the Web Service. This user has all required permissions for managing incidents in SAP Solution Manager.

## New Entries: Details of Added Entries

The screenshot shows the 'New Entries: Details of Added Entries' dialog. The 'External ID type' is set to 'DN' and the 'External ID' is 'CN=tcwm112.deu.hp.com, OU=DEPP, O=MYORG, L=MYCITY, SP=MYST'. Other fields include: Seq. No. (000), User (SERVICE\_DESK), and Min. date.

## Set up SSL between SAP and SMSSMEX

This section describes how to setup SSL between SAP and SMSSMEX.

### Create Keystore and Truststore

SMSSMEX requires

- Two separate stores that contain the certificates used to authenticate and encrypt communication.
- The following certificates
  - Signed certificate with the long hostname of the SMSSMEX server in the CN section (for example **CN="server.hp.com"**). This certificate must be mapped to an SAP user in SAP Solution Manager.
  - Certificate of the root CA used to sign the certificate of the SAP Solution Manager.
  - Certificate of the root CA used to sign the certificate of the SMSSMEX certificate.

The keystore must contain the following certificates:

- Root CA certificate used to sign the SMSSMEX certificate
- SMSSMEX certificate

The truststore must contain the root certificate used to sign the certificate of the SAP Solution Manager.

Any tool can be used to create and manage the key- and truststores. The following examples use the Java JDK tool `keytool` to create and import a signed certificate.

- 1 Create a self-signed certificate. The keypass and the storepass must be identical.

```
keytool -genkey -alias <alias> -keyalg RSA -keystore <keystorefile>
-storepass <password> -keypass <password> -dname "CN=<serverhost>,
OU=<MYOU>, O=<MYORG>, L=<MYCITY>, ST=<MYSTATE>, C=<MY>"
```

For example:

```
keytool -genkey -alias ovictex -keyalg RSA -keystore "C:\Program
Files\HP\SMSSMEX\config\certs\ovictex.keystore" -storepass ovictex
-keypass ovictex -dname "CN=helen2006.asiapacific.hpqcorp.net, OU=TEST,
O=GDCC, L=SH, ST=CN, C=CN"
```

- 2 Create a certificate request:

```
keytool -certreq -keystore <keystorefile> -alias <alias> -storepass
<password>
```

For example:

```
keytool -certreq -keystore "C:\Program
Files\HP\SMSSMEX\config\certs\ovictex.keystore" -alias ovictex -storepass
ovictex
```

- 3 Use the resulting certificate request to acquire a signed certificate from SAP Web (<https://websmp102.sap-ag.de/SSLTest>) with chain PKCS#7. Copy the signed response **<filename>.p7b** (for example, **sap\_rp.p7b**).
- 4 Download the root certificate file for the following web site:  
**<https://tcs.mysap.com/invoke/tc/getCert?SAPServerCA.der>**.
- 5 Import the root certificate from the Certificate Authority (CA) into the keystore.

```
keytool -import -v -alias <alias2> -keystore <keystorefile> -storepass
<password> -file <rootcertificatefile>
```

For example:

```
keytool -import -v -alias saproot -keystore "C:\Program
Files\HP\SMSSMEX\config\certs\ovictex.keystore" -storepass ovictex -file
"C:\Program Files\HP\SMSSMEX\config\certs\getCert.cer"
```

- 6 Import the answer from the Certificate Authority into the keystore. Use the same keystore file and alias the request was created from.

```
keytool -import -v -alias <alias> -keystore <keystorefile> -storepass
<password> -file <certificatefile>
```

For example:

```
keytool -import -v -alias ovictex -keystore "C:\Program
Files\HP\SMSSMEX\config\certs\ovictex.keystore" -storepass ovictex -file
"C:\Program Files\HP\SMSSMEX\config\certs\sap.p7b"
```

To import the certificates into the truststore, use the same command as in the step above, but instead of **<keystorefile>** use the filename of the truststore (if it does not exist, it will be created automatically). For example:

```
keytool -import -v -alias saproot -keystore "C:\Program
Files\HP\SMSSMEX\config\certs\ovictex.truststore" -storepass ovictex -file
"C:\Program Files\HP\SMSSMEX\config\certs\getCert.cer"
```

## Configure Tomcat SSL Use

To enable SSL with Tomcat, configure a new connector in the `server.xml` configuration file. The standard `server.xml` contains a connector definition that has been commented out. The following attributes are required:

```
port=<port>
scheme="https"
secure="true"
clientAuth="false"
sslProtocol = "TLS"
keystoreFile=<keystorefile>
keystorePass=<keystorepass>
truststoreFile=<truststorefile>
truststorePass=<truststorepass>
```

For example:

```
<Connector port="8443"
maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
enableLookups="false" disableUploadTimeout="true"
acceptCount="100" debug="0" scheme="https" secure="true"
clientAuth="false" sslProtocol="TLS"
keystoreFile="C:/Program Files/HP/SMSSMEX/config/certs/ovictex.keystore"
keystorePass="password"
truststoreFile="C:/Program Files/HP/SMSSMEX/config/certs/ovictex.truststore"
truststorePass="password"
/>
```

## Configure Property Files

- 1 Modify `exthd.properties`.

- a In the SAP configuration files in property `exthd.webservice.endpoint` specify the new port (default is 8443) and use **https://** as the protocol. For example:

```
exthd.webservice.endpoint = https://watermelon.chn.hp.com:8001/sap/bc/srt/rfc/sap/ICT_SERVICE_DESK_API?sap-client=300
```

- b Set the `exthd.webservice.authentication.scheme` to **HTTPS**.

- 2 Add the following configuration entries in `ovictex.properties`:

```
<saphostname>.keystore=C:/Program Files/HP/SMSSMEX/certs/ovictex.keystore
<saphostname>.keystore.password=~X1~H+7JA0rcX/R6k05diPxV0w==
<saphostname>.truststore=C:/Program Files/HP/SMSSMEX/certs/ovictex.truststore
<saphostname>.truststore.password=~X1~H+7JA0rcX/R6k05diPxV0w==
```

For example:

```
watermelon.chn.hp.com.keystore= C:/Program Files/HP/SMSSMEX/certs/ovictex.keystore
watermelon.chn.hp.com.keystore.password=~X1~H+7JA0rcX/R6k05diPxV0w==
watermelon.chn.hp.com.truststore= C:/Program Files/HP/SMSSMEX/certs/ovictex.truststore
watermelon.chn.hp.com.truststore.password=~X1~H+7JA0rcX/R6k05diPxV0w==
```



# Security Between HP Service Manager and SMSSMEX

This section describes how to configure security between HP Service Manager and SMSSMEX.

## Configure HP Service Manager for SSL

This section describes how to configure HP Service Manager for SSL.

The prerequisite is OpenSSL version 9.7 or higher. For more information about downloading and installing OpenSSL, see <http://www.openssl.org>.

Service Manager provides in the /RUN directory an OpenSSL executable file that can be used to generate and sign most certificates. Implementing the trusted sign-on requires file `openssl.conf` in addition to the executable (the file is available after installing OpenSSL).



- The \*.pem files must be different in at least one section when being prompted for distinguished name information. For Windows clients, this difference is the common name. If the Web Tier or Windows client is on the same system as the server then an additional differentiating entry, such as organization, must be made.
- When prompted, always use the fully qualified name (**computer.domain.com**) as the first/last name.

## Generate a Private/Public Key Pair for Root Certificate Authority

- 1 Generate an RSA private key.

```
openssl genrsa -des3 -out cakey.pem 2048
```

- 2 Create a self-signed root certificate for the Certificate Authority (CA).

```
openssl req -new -key cakey.pem -x509 -days 1095 -out mycacert.pem -config openssl.conf
```



To make a unique .pem file, give a unique Organization Name (for example org1). When asked for a Common Name, enter the fully qualified name of the Service Manager Server host.

- 3 Import the self-signed root certificate into a trust key store.

```
keytool -import -keystore <trustkeystore> -trustcacerts -alias <alias> -file <certificate>
```

For example:

```
keytool -import -keystore cacerts -trustcacerts -alias scca -file mycacert.pem
```

## Generate a Private/Public Key Pair for Service Manager Server

- 1 Generate a private/public key pair.

```
keytool -genkey -alias <alias> -keystore <keystorefile>
```

For example:

```
keytool -genkey -alias scserver -keystore scserver.keystore
```



When asked for organization name, enter a unique name (for example `org2`).  
When asked for first and last name, enter the fully qualified name of the Service Manager Server host.

**2** Generate the request file.

```
keytool -certreq -alias <alias> -keystore <keystorefile> -file  
<requestfile>
```

For example:

```
keytool -certreq -alias scserver -keystore scserver.keystore -file  
scservercert_req.crs
```

**3** Self-sign the request.

```
openssl x509 -req -days <validdays> -in <requestfile> -CA  
<certificatefile> -CAkey <keystorefile> -CAcreateserial -out  
<certificatefile>
```

For example:

```
openssl x509 -req -days 1095 -in scservercert_req.crs -CA mycacert.pem  
-CAkey cakey.pem -CAcreateserial -out scservercert.pem
```

**4** Import the root CA certificate into the server keystore.

```
keytool -import -trustcacerts -alias <alias> -keystore <keystorefile>  
-file <certificatefile>
```

For example:

```
keytool -import -trustcacerts -alias scca -keystore scserver.keystore  
-file mycacert.pem
```

**5** Import the signed certificate into the keystore.

```
keytool -import -trustcacerts -alias <alias> -keystore <keystorefile> -  
file <certificatefile>
```

For example:

```
keytool -import -trustcacerts -alias scserver -keystore scserver.keystore  
-file scservercert.pem
```

## Generate the Client Keystore for Service Manager Client

**1** Generate the private/public key pair (with the first and last name and the case-sensitive fully qualified name of the machine).

```
keytool -genkey -alias <alias> -keystore <keystorefile>
```

For example:

```
keytool -genkey -alias scclient -keystore scclient.keystore
```



When asked for organization name, input a unique one (for example `org3`). When asked for the first and last name, enter the fully qualified name of the Service Manager client host.

**2** Generate the request file.

```
keytool -certreq -alias <alias> -keystore <keystorefile> -file
<requestfile>
```

For example:

```
keytool -certreq -alias scclient -keystore scclient.keystore -file
scclientcert_req.crs
```

**3 Self-sign the request.**

```
openssl x509 -req -days <validdays> -in <requestfile> -CA
<certificatefile> -CAkey <keystorefile> -CAcreateserial -out
<certificatefile>
```

For example:

```
openssl x509 -req -days 365 -in scclientcert_req.crs -CA mycacert.pem
-CAkey cakey.pem -CAcreateserial -out scclientcert.pem
```

**4 Import the root CA certificate into the client keystore.**

```
keytool -import -trustcacerts -alias <alias> -keystore <keystorefile>
-file <certificatefile>
```

For example:

```
keytool -import -trustcacerts -alias scca -keystore scclient.keystore
-file mycacert.pem
```

**5 Import the self-signed certificate into the client keystore.**

```
keytool -import -trustcacerts -alias <alias> -keystore <keystorefile>
-file <certificatefile>
```

For example:

```
keytool -import -trustcacerts -alias scclient -keystore scclient.keystore
-file scclientcert.pem
```

## Generate the Client Keystore for SMSSMEX

**1 Generate the private/public key pair.**

```
keytool -genkey -alias <alias> -keystore <keystorefile>
```

For example:

```
keytool -genkey -alias ovictex -keystore ovictex.keystore
```



When asked for organization name, enter a unique name (for example org4). When asked for the first and last name, enter the fully qualified name of the incident exchange middleware host.

**2 Generate the request file.**

```
keytool -certreq -alias <alias> -keystore <keystorefile> -file
<requestfile>
```

For example:

```
keytool -certreq -alias ovictex -keystore ovictex.keystore -file
ovictexcert_req.crs
```

**3 Self-sign the request.**

```
openssl x509 -req -days <validdays> -in <requestfile> -CA
<certificatefile> -CAkey <keystorefile> -CAcreateserial -out
<certificatefile>
```

For example:

```
openssl x509 -req -days 365 -in ovictexcert_req.crs -CA mycacert.pem
-CAkey cakey.pem -CAcreateserial -out ovictexcert.pem
```

**4 Import the root CA certificate into the client keystore.**

```
keytool -import -trustcacerts -alias <alias> -keystore <keystorefile>
-file <certificatefile>
```

For example:

```
keytool -import -trustcacerts -alias scca -keystore ovictex.keystore -file
mycacert.pem
```

**5 Import the self-signed certificate into the client keystore.**

```
keytool -import -trustcacerts -alias <alias> -keystore <keystorefile>
-file <certificatefile>
```

For example:

```
keytool -import -trustcacerts -alias ovictex -keystore ovictex.keystore
-file ovictexcert.pem
```

## Generate the Trust-List Keystore for Service Manager Server

**1 Export the certificate file.**

```
keytool -export -alias <alias> -keystore <keystorefile> -file
<certificatefile>
```

For example:

```
keytool -export -alias scclient -keystore scclient.keystore -file
scclientpubkey.crt
keytool -export -alias ovictex -keystore ovictex.keystore -file
ovictexpubkey.crt
```

**2 Import the certificate file.**

```
keytool -import -alias <alias> -file <certificatefile> -keystore <jksfile>
```

For example:

```
keytool -import -alias scclient -file scclientpubkey.crt -keystore
trustedclients.jks
keytool -import -alias ovictex -file ovictexpubkey.crt -keystore
trustedclients.jks
```

## SSL Configuration in Service Manager Server

**1 Import the root certificate of the SAP Certificate Authority into the trust key store.**

```
keytool -import -keystore <trustkeystore> -trustcacerts -alias <alias>
-file <certificate>
```

For example:

```
keytool -import -keystore cacerts -trustcacerts -alias sapca -file
sapca.cert
```

2 Copy the generated files `cacerts`, `scserver.keystore`, and `trustedclients.jks` into `<Service Manager installation path>\Server\RUN`.

3 Add the following entries to `sm.ini`:

```
#
# SSL configuration
#
ssl:1
ssl_reqClientAuth:1

#
# Certificates
#
truststoreFile:cacerts
truststorePass:password
keystoreFile:scserver.keystore
keystorePass:password
ssl_trustedClientsJKS:trustedclients.jks
ssl_trustedClientsPwd:password
```

4 Open the Service Manager Client.

5 Go to **Tailoring** → **Script Library**.

6 Search for name **HPSAPTrigger**.

7 Change the following javascript code.

```
var url = "http://<smssmex full host name>:<port>/ovictex/servlet/OvHDTrigger?parameters="
+ encodeURIComponent(action) + ";" + encodeURIComponent(incidentId) + ";"
+ encodeURIComponent(extHdId);

to

var url = "https://<smssmex full host name>:<ssl port>/ovictex/servlet/OvHDTrigger?parameters="
+ encodeURIComponent(action) + ";" + encodeURIComponent(incidentId) + ";"
+ encodeURIComponent(extHdId);
```

## SSL Configuration in Service Manager Client

To configure SSL in Service Manager client, do the following:

- 1 Open the Service Manager Client.
- 2 From the menu select **Window** → **Preferences...** to open the Preferences dialog.
- 3 Expand the HP Service Manager node in the left menu tree. Select **Security** to open the client security dialog.
- 4 Click **Browse...**
- 5 Specify the CA certificates file and Client keystore file.
- 6 Input the password of the client keystore in the Client keystore password field.
- 7 Click **OK** to save the Security configuration.
- 8 Restart Service Manager Client to enable the newly configured Security information.
- 9 In the Connections dialog, the value of field Server host name must be the fully qualified name of the Service Manager server.

- 10 In the Advanced tab, make sure that **Use SSL Encryption** is checked.

## Service Manager Web Client SSL Configuration

To configure SSL in Service Manager web client, do the following:

- 1 Copy the trust keystore and client keystore files to the WEB-INF folder of the Service Manager Web Application Server.
- 2 Open the Web configuration file `web.xml` in a text editor.
- 3 Modify the following configuration entry.

```
<init-param>
  <param-name>serverHost</param-name>
  <param-value>servername.domainname.com</param-value>
</init-param>
```

For example:

```
<init-param>
  <param-name>serverHost</param-name>
  <param-value>SMCI02.chn.hp.com</param-value>
</init-param>
<init-param>
  <param-name>serverPort</param-name>
  <param-value>serverPort</param-value>
</init-param>
```

For example:

```
<init-param>
  <param-name>serverPort</param-name>
  <param-value>13080</param-value>
</init-param>
<init-param>
  <param-name>ssl</param-name>
  <param-value>>true</param-value>
</init-param>
<init-param>
  <param-name>cacerts</param-name>
  <param-value>trustKeystore</param-value>
</init-param>
```

For example:

```
<init-param>
  <param-name>cacerts</param-name>
  <param-value>/WEB-INF/cacerts</param-value>
</init-param>
<init-param>
  <param-name>keystore</param-name>
  <param-value>clientKeystore</param-value>
</init-param>
```

For example:

```
<init-param>
  <param-name>keystore</param-name>
  <param-value>/WEB-INF/scclient.keystore</param-value>
</init-param>
```

```

<init-param>
  <param-name>keystorePassword</param-name>
  <param-value>clientKeystorePassword</param-value>
</init-param>

```

For example:

```

<init-param>
  <param-name>keystorePassword</param-name>
  <param-value>sm9client</param-value>
</init-param>

```

- 4 Open `WEB-INF/classes/application-context.xml` in a text editor. Change
 

```

/**=httpSessionContextIntegrationFilter,anonymousProcessingFilter

```

 to
 

```

/
**=httpSessionContextIntegrationFilter,preAuthenticationFilter,anonymousP
rocessingFilter

```

## Configure SMSSMEX for SSL Communication with Service Manager

To configure SMSSMEX for SSL communications with Service Manager, do the following:

- 1 Import the root CA into the trust keystore.

```

keytool -import -trustcacerts -alias <alias> -keystore <keystorefile>
-file <certificatefile>

```

For example:

```

keytool -import -trustcacerts -alias scca -keystore ovictex.truststore
-file mycacert.pem

```

- 2 Configure `ovictex.properties`.

- a Set `sc.webservice.endpoint`.

```

sc.webservice.endpoint = http://<smhostname>:<port>/sc62server/PWS

```

- b Add the following configuration entries in `ovictex.properties`.

```

<smhostname>.keystore=<ovictex keystore file>
<smhostname>.keystore.password=<keystore password>
<smhostname>.truststore=<ovictex truststore file>
<smhostname>.truststore.password=<truststore password>

```

For example:

```

sc.webservice.endpoint = http://SMCI02.chn.hp.com:13080/sc62server/PWS
.....
SMCI02.chn.hp.com.keystore=C:/Program Files/HP/SMSSMEX/config/certs/
ovictex.keystore
SMCI02.chn.hp.com.keystore.password=~X1~eD+6cy60MNxdK9tcCQVBww==
SMCI02.chn.hp.com.truststore=C:/Program Files/HP/SMSSMEX/config/certs/
ovictex.truststore

```

SMCI02.chn.hp.com.truststore.password=-X1~eD+6cy6OMNxdK9tcCQVBww==



The `keystore.password` and `truststore.password` should use `<SMSSMEX_installDir>/bin/encryptPasswords.bat` to encrypt. For usage of `encryptPasswords.bat`, refer to *Tools* on page 130.



# 10 Licensing

This chapter describes licensing.

## License Types

The following license types are available:

- InstantOn license provides full access to all features for 60 days.
- Permanent license is node-locked (restricted to a range of IP addresses).

## Autopass License Management

Autopass License Management is a tool for license management of HP products.

### 1 Start Autopass License Management.

- a For Windows (Autopass is by default installed under C:\Program Files\Common Files\Hewlett-Packard\HPOvLIC):

```
"<OvLIC_Install_Path>\demo\hpovliccli.bat" -gui  
"%SMSSMEX_HOME%\bin\SMSSMEX_pdf.txt"
```

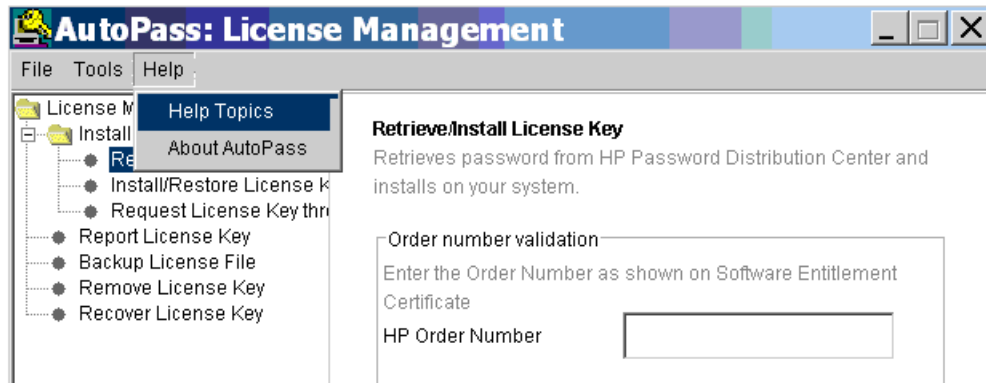
- b For HP-UX:

```
/bin/sh/opt/OV/HPOvLIC/demo/hpovliccli.sh -gui /opt/HP/SMSSMEX/bin/  
SMSSMEX_pdf.txt
```

- c For Linux:

```
/opt/OV/HPOvLIC/demo/hpovliccli.sh -gui /opt/HP/SMSSMEX/bin/  
SMSSMEX_pdf.txt
```

- 2 To install or remove the license, refer to the Autopass help (click **Help** on the toolbar of Autopass License Management and click **Help Topics** from the menu).



# 11 Status Page

The HP Incident Management Service provides a comprehensive overview of the status of the incident exchange systems and services and provides extensive information for troubleshooting. The URL of the status page is

`http://<hostname>:<port>/ovictex/servlet/OvHDTrigger?status`

The following is an example status page.

HP Integrated Incident Management Service

<b>Product info</b>	<b>ServiceDesk Host:</b> http://hefehell.9999/sc62server/ws	<b>Database Host:</b> alfacon2
<b>Version:</b> 02.03.002	<b>ServiceDesk Version:</b> servicecenter6.2	<b>DB Status:</b> <b>Alive</b>
<b>Created:</b> 2007-06-05 10:39:29	<b>GUID:</b> ALFACON2_BORYS123456789	<b>Is Proxy Mode:</b> FALSE
<b>License:</b> Unlimited license	<b>ServiceDesk Status:</b> <b>Alive</b>	<b>Attachment Mode:</b> LOCAL

External Helpdesk Instance Name	Status	GUID	URL
exthdl	<b>Alive</b>	AA3DD5EE16387E4AB0AA5AED62D66A14	http://portwein.84/sap/bc/srt/rfc/sap/ICT_SERVICE_DESK_API?sap-client=200

Page generated at: 2007-09-28 13:48:26

When a service becomes unavailable, the status changes from Alive to Dead.

HP Integrated Incident Management Service

<b>Product info</b>	<b>ServiceDesk Host:</b> http://hefehell.9999/sc62server/ws	<b>Database Host:</b> alfacon2
<b>Version:</b> 02.03.002	<b>ServiceDesk Version:</b> servicecenter6.2	<b>DB Status:</b> <b>Dead</b>
<b>Created:</b> 2007-06-05 10:39:29	<b>GUID:</b> ALFACON2_BORYS123456789	<b>Is Proxy Mode:</b> FALSE
<b>License:</b> Error during license check	<b>ServiceDesk Status:</b> <b>Alive</b>	<b>Attachment Mode:</b> LOCAL

External Helpdesk Instance Name	Status	GUID	URL
exthdl	<b>Alive</b>	AA3DD5EE16387E4AB0AA5AED62D66A14	http://portwein.84/sap/bc/srt/rfc/sap/ICT_SERVICE_DESK_API?sap-client=200

Page generated at: 2007-10-08 16:54:49

HP Integrated Incident Management Service

<b>Product info</b>	<b>ServiceDesk Host:</b> http://hefehell.9999/sc62server/ws	<b>Database Host:</b> alfacon2
<b>Version:</b> 02.03.002	<b>ServiceDesk Version:</b> servicecenter6.2	<b>DB Status:</b> <b>Dead</b>
<b>Created:</b> 2007-06-05 10:39:29	<b>GUID:</b> ALFACON2_BORYS123456789	<b>Is Proxy Mode:</b> FALSE
<b>License:</b> Error during license check	<b>ServiceDesk Status:</b> <b>Alive</b>	<b>Attachment Mode:</b> LOCAL

External Helpdesk Instance Name	Status	GUID	URL
exthdl	<b>Dead</b>	AA3DD5EE16387E4AB0AA5AED62D66A14	http://portwein.84/sap/bc/srt/rfc/sap/ICT_SERVICE_DESK_API?sap-client=200

Page generated at: 2007-10-08 16:56:43



# 12 Troubleshooting

This chapter describes how to troubleshoot common problems. The checker tool (see [Verifying Configuration](#) on page 25) is a good aid for troubleshooting.

## checker.bat and encryptPasswords.bat Fail

### Problem

The exception “Class not found” appears in the console when running **checker.bat** or **encryptPasswords.bat**.

### Cause

The library files that checker requires were not extracted to the required Tomcat.

### Solution

- 1 Run **setup startup**. Tomcat extracts `ovictex.war` and a copies the required jar files.
- 2 Restart Tomcat.

## Incident not Sent to SAP AGS

### Problem

Incident is not sent to SAP AGS when using a newly configured priority in Solution Manager.

### Cause

Incidents that have set new priorities in Solution Manager can not be sent to SAP AGS (only default priorities can be sent).

### Solution

`fieldMapping.xml` maps to default priorities.

## java.lang.OutOfMemoryError

### Cause

Too many incidents with big attachments are exchanged simultaneously.

### Solution

Increase the Java Virtual Machine heap size in `catalina.bat` (Tomcat).

```
set JAVA_OPTS=-Xms512m -Xmx1024m
```

## Record in EventIn is not Executed

### Problem

The record in table EvenIn is not executed. After Service Manager sends the incident to Solution Manager, the process is finished, but the following problems occur:

- Integration buttons for the incident are not shown correctly.
- Field `hidden.meta.info` is not updated.

### Cause

The Event In process threads are not started when the Service Manager server starts, so in the Input Events window (**Tailoring** → **Event Services** → **Input Events**) the input events are not handled (as shown in the following diagram).

The screenshot shows the SAP Event Services Input Queue interface. At the top, there is a table with columns: Type, Checkpoint, Event Time, User ID, and evfields. The table contains several rows of event data. Below the table is a toolbar with buttons for OK, Cancel, Previous, Next, Add, Save, Delete, Find, and Fill. Below the toolbar is the 'Event Services Input Queue' header, followed by a form for event details. The form includes fields for Event Code, Status, System Sequence, Time Stamps (First Expiration, Time Processed), User Information (User Name, Password, User Sequence), Incident Information (Network Name, Cause Code, Incident ID), Filter Information (Count, Next Expiration), System Option, and Field Separation Character. At the bottom, there is a text area containing the event details for the selected row.

Type	Checkpoint	Event Time	User ID	evfields
hpsapepmu	0EF1AAD279...		ovictex	IM16010^Requester:ProviderProcessing^800000318^false^Created incident in External Helpdesk :SAP Solution Manager. Incident Id at E...
hpsapepmu	11F21BD25B7...		ovictex	IM16003^Provider:ProviderProcessing^^true^Failed to process action addinfo as the incident is locked by the external helpdesk. Request ...
hpsapepmu	14498DE25B6...		ovictex	IM16002^Provider:ProviderProcessing^^true^Additional information sent to External Helpdesk : SAP Solution Manager
hpsapepmu	27D136E279A...		ovictex	IM16013^Requester:ProviderProcessing^800000321^false^Created incident in External Helpdesk :SAP Solution Manager. Incident Id at E...
hpsapepmu	520599D463D...		ovictex	IM16008^Requester:ProviderProcessing^800000314^false^Created incident in External Helpdesk :SAP Solution Manager. Incident Id at E...
hpsapepmu	52E73F925B7...		ovictex	IM16003^Provider:ProviderProcessing^^true^Failed to process action addinfo as the incident is locked by the external helpdesk. Request ...
hpsapepmu	58832C625B6...		ovictex	IM16001^Requester:ProviderProcessing^800000306^false^Created incident in External Helpdesk :SAP Solution Manager. Incident Id at E...
hpsapepmu	58CD60B2639...		ovictex	IM16006^Provider:SolutionProvided^^false^Solution is provided to External Helpdesk: SAP Solution Manager
hpsapepmu	638432A2799...		ovictex	IM16012^Requester:ProviderProcessing^800000320^false^Created incident in External Helpdesk :SAP Solution Manager. Incident Id at E...

**Event Services Input Queue**

**Event Code**

hpsapepmu

Status

System Sequence:

520599D463D8753

**Time Stamps**

First Expiration:

Time Processed:

**User Information**

User Name

ovictex

Password

User Sequence

**Incident Information**

Network Name

Cause Code

Incident ID

**Filter Information**

Count

Next Expiration:

System Option

Field Separation Character

External Information String

IM16008^Requester:ProviderProcessing^800000314^false^Created incident in External Helpdesk :SAP Solution Manager. Incident Id at External Helpdesk is 800000314. External Helpdesk is now processing

## Solution

To handle the input events, start the event in process threads.

### 1 Go to **System Status**

TOTAL USERS: 1 - use Refresh Display to refresh statistics

Command	User N...	PID	Device ID	Login Time	Idl...	TID	Session ID
	ovictex	3208	Soap-Windows...	08/07/23 17:...	00:...	4416	3271
	KMUpdate	3056	SYSTEM	08/07/21 11:...	00:...	4512	48
	sync	3056	SYSTEM	08/07/21 11:...	00:...	4564	47
	alert	3056	SYSTEM	08/07/21 11:...	00:...	4588	46
	ocm	3056	SYSTEM	08/07/21 11:...	00:...	4584	45
	contract	3056	SYSTEM	08/07/21 11:...	00:...	4580	44
	availability	3056	SYSTEM	08/07/21 11:...	00:...	4508	43
	event	3056	SYSTEM	08/07/21 11:...	00:...	3404	42
	linker	3056	SYSTEM	08/07/21 11:...	00:...	4568	40
	lister	3056	SYSTEM	08/07/21 11:...	00:...	4524	39
	marquee	3056	SYSTEM	08/07/21 11:...	00:...	2812	37
	agent	3056	SYSTEM	08/07/21 11:...	00:...	3300	36
	sla	3056	SYSTEM	08/07/21 11:...	00:...	4420	35
	change	3056	SYSTEM	08/07/21 11:...	00:...	4424	34
	problem	3056	SYSTEM	08/07/21 11:...	00:...	4432	33
	report	3056	SYSTEM	08/07/21 11:...	00:...	4428	32
	spool	3056	SYSTEM	08/07/21 11:...	00:...	4336	31
	system....	3056	SYSTEM	08/07/21 11:...	2 0...	-1	30
	Thread...	3208	SYSTEM	08/07/21 11:...	2 0...	-1	29

### 2 Click **Start Scheduler**.

Name	Description
agent	query/chart agent
alert.processor	Standard Alert processor
availability.startup	availability processor
change.startup	ChM alert/notification processor
contract	contract background agent
event.startup	Event Services processor
gie.startup	Generic Input Event Services processor
inactive.startup	dismiss inactive users
KMUpdate	Checks for update records and sends them to the indexer
linker.startup	Problem/Incident Sync Task
lister.startup	Global List Builder Routine
marquee	marquee agent
ocm.startup	OCM processor
printer.startup	print scheduler
problem	IM alert and message processor
report.startup	report processor
scauto.startup	SCAUTO startup
scemail.startup	SCEMAIL startup
SLA	SLA background agent
startup	system startup default
Sync	

### 3 Start `event.startup` and dependent process threads.

## Incident Update or Process Action Fails

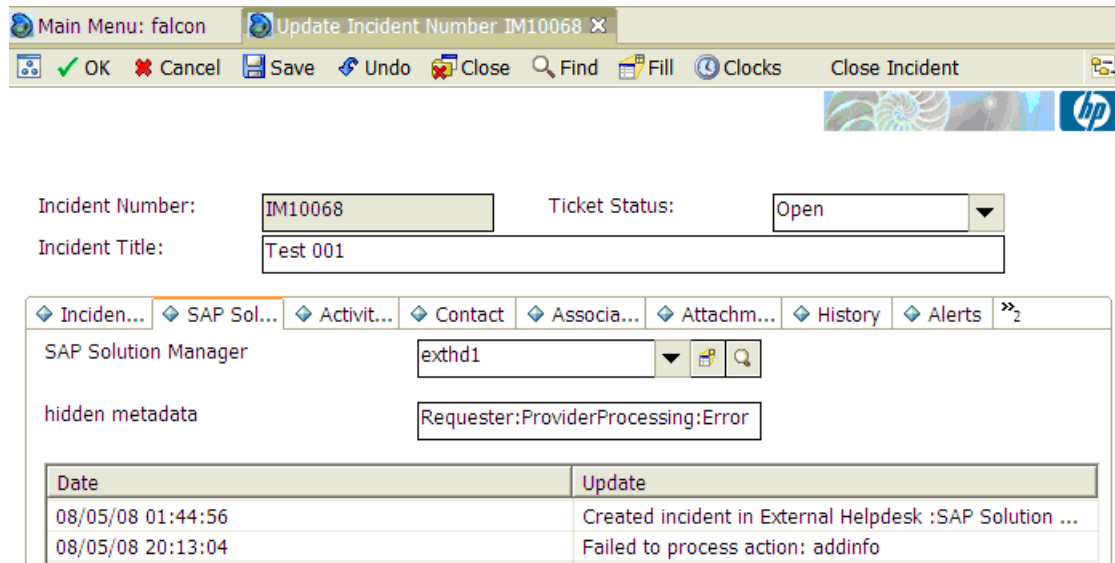
### Problem

Some incidents are not exchanged between Service Manager and SAP Solution Manager. The log file or console message of SMSSMEX displays WARN or FATAL level information as described below.

#### 1 Service Manager sends an incident to SAP Solution Manager:

```
WARN com.hp.ov.ictex - Failed to process action addinfo as the incident
is locked by the external helpdesk. Request will be sent again later.
```

The following alert may appear in Service Manager:



Incident Number:  Ticket Status:

Incident Title:

Date	Update
08/05/08 01:44:56	Created incident in External Helpdesk :SAP Solution ...
08/05/08 20:13:04	Failed to process action: addinfo

#### 2 SAP Solution Manager updates the incident to Service Manager.

```
DEBUG com.hp.ov.ictex - Failed to update incident. id:IM10068
DEBUG com.hp.ov.ictex - Response code = 3. Probably an Incident: IM10068
is locked.
FATAL com.hp.ov.ictex - Saving of incident failed. Received Message from
ServiceCenter: Resource Unavailable
null
FATAL com.hp.ov.ictex - An error ocured while processing incident ID
IM10068. Message: Resource Unavailable
null
DEBUG com.hp.ov.ictex - An error ocured while processing incident ID
IM10068. Message: Resource Unavailable
null
com.hp.ov.ictex.ovhdaccess.OvHDException: Resource Unavailable
null
at com.hp.ov.ictex.ovhdaccess.servicecenter.Incident.save(Unknown Source)
at
com.hp.ov.ictex.exthdrequesthandler.OvictexServer.updateIncident(Unknown
Source)
...
```



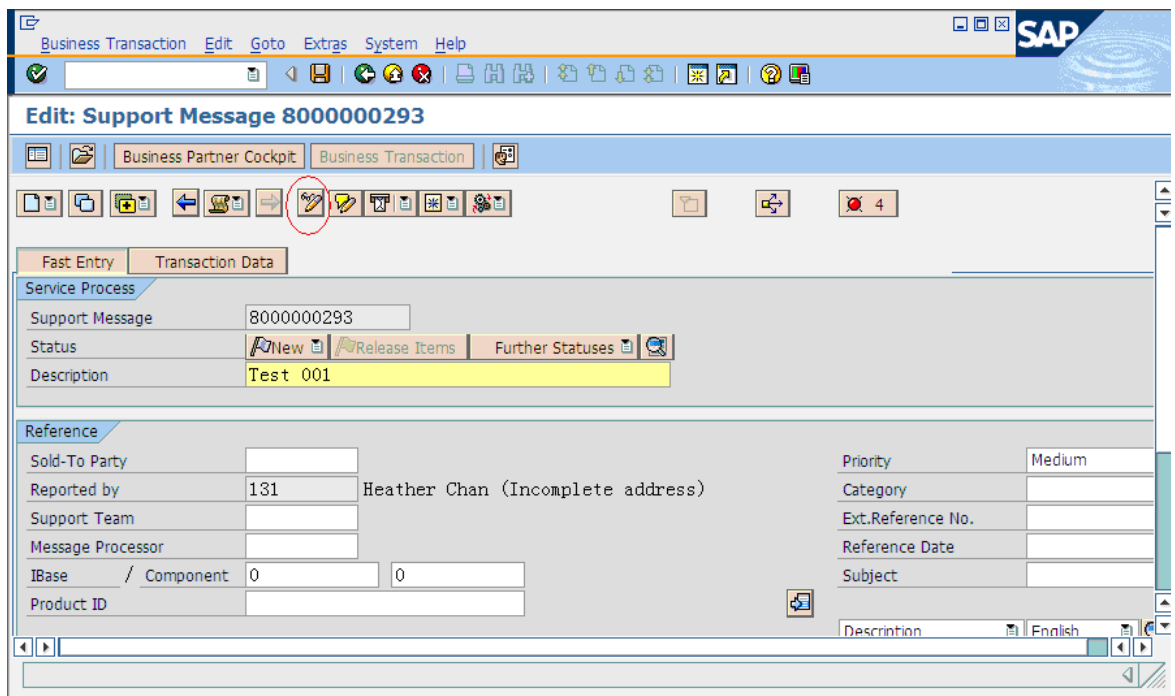
## Cause

The incident in HP Service Manager or SAP Solution Manager is locked:

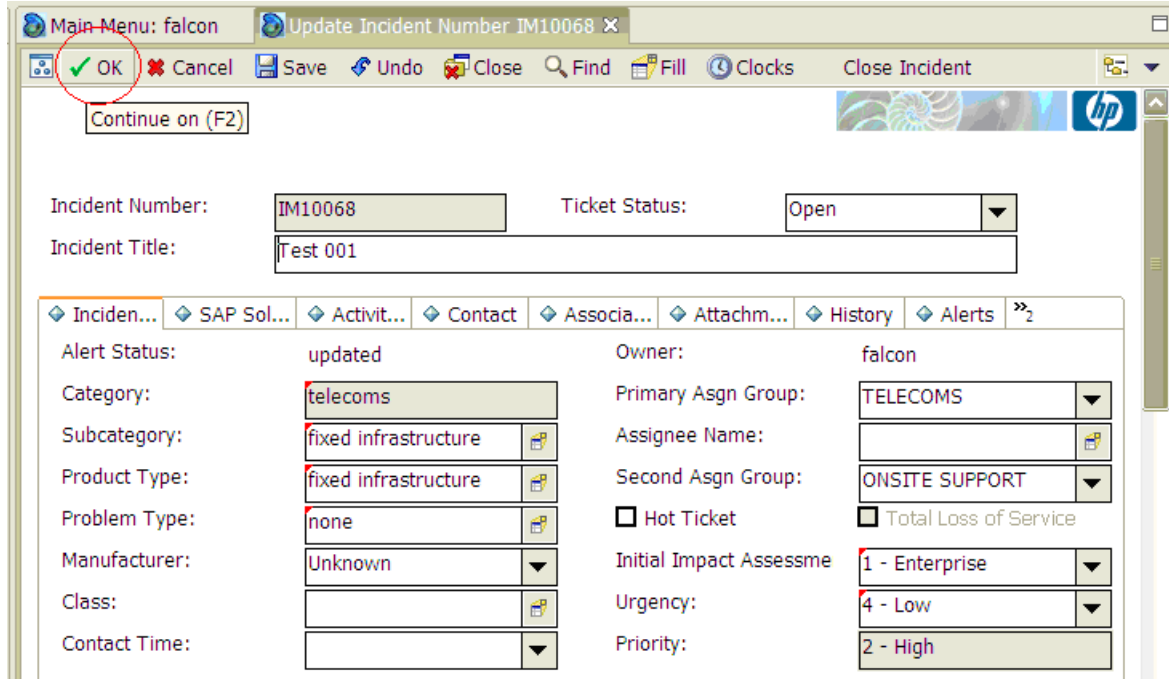
- 1 In SAP Solution Manager, if the user does not click the button **Display/Change Trans.** to release an incident write lock, HP Service Manager can not update or send a message to SAP.
- 2 In HP Service Manager, if the user does not click **OK** to release an incident write lock in time, the incident maintains the “Updating” status and no message from SAP can be accepted (until the status changes).

## Solution

In SAP Solution Manager always click **Display/Change Trans.** after finishing or updating an activity.



In HP Service Manager click **OK** after finishing or updating an activity.



## Information is not Updated in SAP Solution Manager

### Problem

An open support message is not changed after synchronization from Service Manager to SAP.

### Cause

SAP solution manager does not refresh the support message automatically.

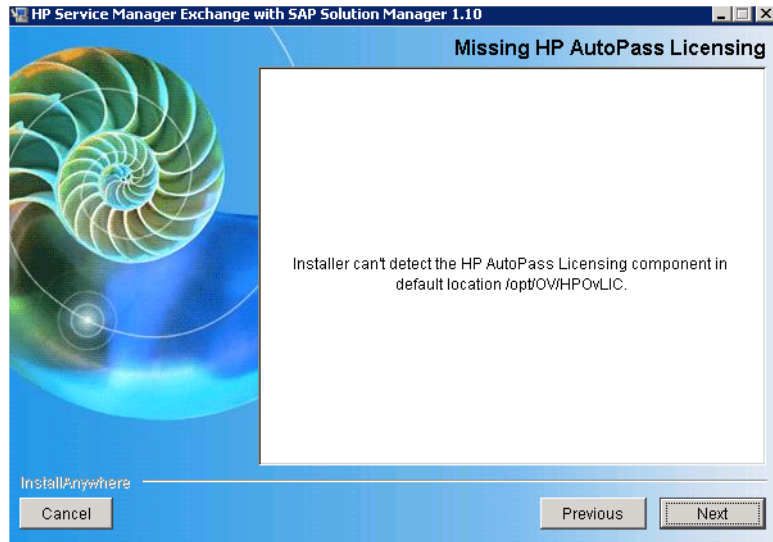
### Solution

In SAP GUI, exit from the current transaction and execute transaction **crmd\_order**.

## Error Occurs When Installing SMSSMEX on Windows 2008 R2 (64-bit) Server

### Problem

The “Installer can’t detect the HP AutoPass Licensing component in default location /opt/OV/HPOvLIC” error occurs when trying to install SMSSMEX on a Windows 2008 R2 (64-bit) server.



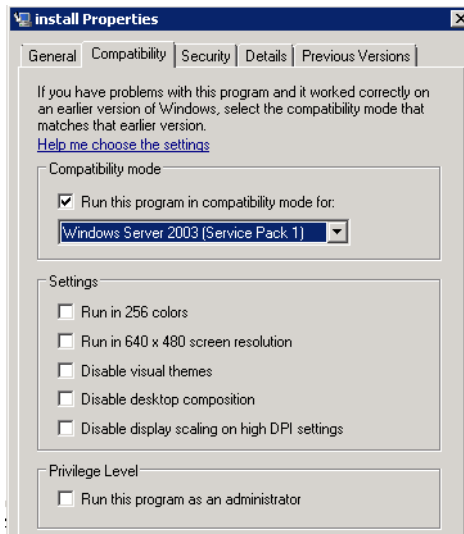
### Cause

SMSSMEX does not support Windows 2008 R2 (64-bit) server.

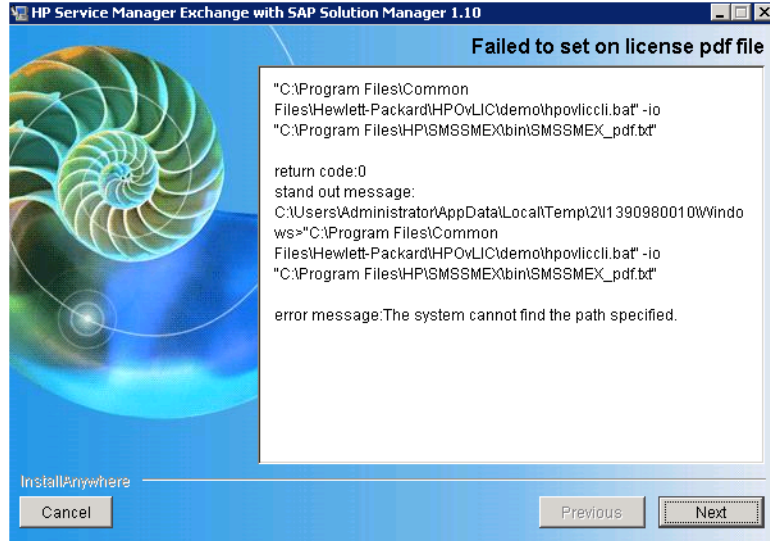
### Solution

Use the compatibility mode of Windows 2003 to install SMSSMEX.

- 1 Right-click on the installer and select **Properties**. Change to use the compatibility mode to run the installer. Refer to the following screenshot as an example:



- 2 Run the installer. If the following error occur:



Ignore the failure, and then click **Cancel** to quit the installation.

- 3 Copy C:\Program Files (x86)\Common Files\Hewlett-Packard\HPOvLIC to C:\Program Files\Common Files\Hewlett-Packard\HPOvLIC.
- 4 Run the installer again. This time the installation will succeed.

# 13 Upgrading SMSSMEX

## Upgrading SMSSMEX from V1.00 to V1.01

To upgrade SMSSMEX from v1.00 to v1.01, perform the following steps:

- Open maps **hp sap problem update**, and update **hidden.meta.data** Post-Map instruction.
  - a Go to **Tailoring** → **Event Services** → **Maps** on Service Manager, or **Utilities** → **Event Services** → **Administration** → **Maps** on Service Center, the Event Map page appears.
  - b Enter **hp sap problem update** in Map Name field, and click **Search**.

Map Name	Seq	Pos	File Name	Field Name	Query
hp sap problem update	1	1	probsummary	number	
hp sap problem update	1	2	probsummary	hidden.meta.data	
hp sap problem update	1	3	probsummary	update.action	
hp sap problem update	1	4	probsummary	is.ictex.action.blocked	
hp sap problem update	1	5	probsummary	\$exchange.history	

Event Map

Map Name:  Type:  Fixed or Variable:

Sequence:  Position:  Length:

Basics Expressions

File Name:

Query:

Field Name:  Nullsub:

Data Type:  Translate:

Array Information

Element Type:

Element Separator:  Element Separator (structure):

Element Length:

- c Select the entry with Field Name as `hidden.meta.data`, and go to **Expressions** tab. Insert the following line (`[HIDDEN.META.DATA_UPGRADE]` in `code_sm9.txt`) in Post-Map Instructions before the `cleanup ($isIncidentExchangeFlag) ; cleanup ($hmd)` line:

```
if ($hmd="" or index(":Error", $hmd)>0) then (is.ictex.action.blocked
in $axces.target=false);if ($hmd="Closed") then (problem.status in
$axces.target="Closed"; status in $axces.target="closed")
```

The screenshot displays the HP Service Manager interface. The main window, titled "Mass Unload", shows a table with the following data:

Map Name	Seq	Pos	File Name	Field Name	Query
hp sap problem update	1	1	probsummary	number	
hp sap problem update	1	2	probsummary	hidden.meta.data	
hp sap problem update	1	3	probsummary	update.action	
hp sap problem update	1	4	probsummary	is.ictex.action.blocked	
hp sap problem update	1	5	probsummary	\$exchange.history	

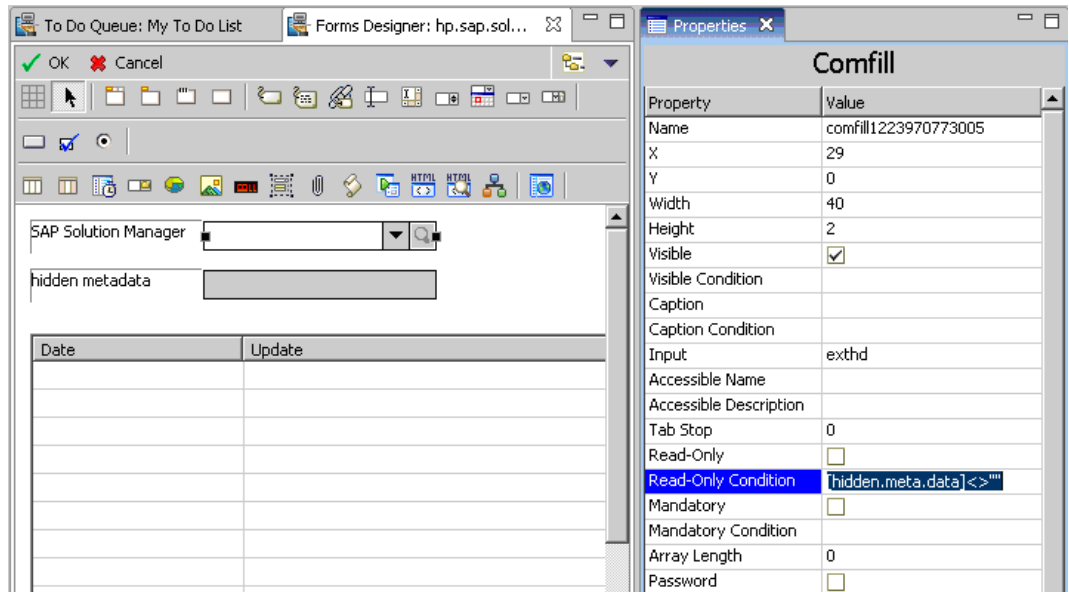
Below the table, the "Map Name" is set to "hp sap problem update" and the "Position" is set to "2". The "Expressions" tab is active, showing the following code in the "Post-Map Instructions" field:

```
$isIncidentExchangeFlag=not same("Requester:StartExchange:Error", $hmd)
is.incident.exchange in $axces.target=$isIncidentExchangeFlag
if ($hmd="Closed") then (problem.status in $axces.target="Closed");if ($hmd=
if ($hmd="" or index(":Error", $hmd)>0) then (is.ictex.action.blocked in $axces
cleanup($isIncidentExchangeFlag);cleanup($hmd)
```

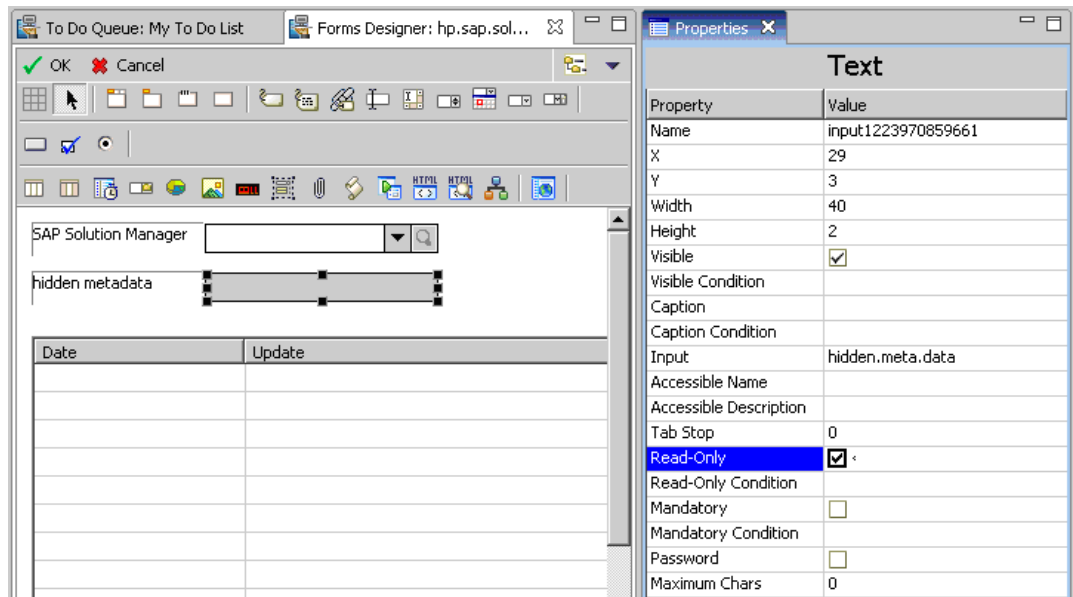
An "HP Service Manager - Edit Popup Window" is also visible, showing the same code snippet as above.

- Modify hp.sap.solution.sub form.
  - a Go to **Tailoring** → **Forms Designer** on Service Manager, or **Toolkit** → **Forms Designer** on Service Center. The Forms Designer page appears.
  - b Type **hp.sap.solution.sub** in the Form field, and then click **Search**.
  - c The Design window appears. Click **Design** to enable design mode for the hp.sap.solution.sub form.

- d Select **comfill** control from the dropdown list for **SAP Solution Manager** field, then in the Properties pane displayed, enter **[hidden.meta.data] <> ""** as the value for Read-Only Condition property, and uncheck **Fill Button Visible** to hide the fill button.



- e Set **hidden metadata** field to read-only mode by checking Read-Only property in the Properties pane.



- Add `hp.sap.solution.sub` sub-form into the following forms:  
`IM.default.open.g`  
`IM.default.update.g`
  - Refer to step 1 to step 4 in [Exchange History](#) for detailed instructions on Service Manager.
- Add a link record for field `exthd` of `probsummary` table:
- Add the following expression.

- a Go to **External Access** page.
  - On Service Manager 7.x, go to **Tailoring** → **Tailoring Tools** → **External Access**
  - On Service Center 6.2, go to **Utilities** → **Tools** → **Web Services** → **External Access**
- b Enter **IncidentManagement** in the Service Name field, click **Search**, and then add the following expression in Expressions.
 

```
if (hidden.meta.data in $L.file="Closed") then (problem.status in $L.file="Closed")
```

## Upgrading SMSSMEX from V1.01 to V1.02

The SAP System Landscape Directory Registration is a new feature in SMSSMEX v1.02. However, this feature is optional. If you do not deploy the SAP System Landscape Directory, the functionality of SMSSMEX v1.02 will not be affected.

For detailed SAP System Landscape Directory registration instructions, refer to [Appendix E, SAP System Landscape Directory Registration](#).

## Upgrading SMSSMEX from V1.02 to V1.10

To upgrade SMSSMEX from v1.02 to v1.10, perform the following tasks:

### Task 1: Updating the probsummary table

Do the following:

- 1 Select **System Definition** → **Tables** → **probsummary** → **Tab Fields and Keys**.
- 2 Create the following additional field in table probsummary.

Field name	Caption	Data type
sap.incident.type	Sap Incident type	Character

### Task 2: Setting default closure code and resolution when closing incidents

When closing incidents from SAP Solution Manager, change the incidents' status and set default closure code and resolution if empty. Do the following:

- 1 Click **Tailoring** → **Web Services** → **WSDL configuration**.
- 2 Enter **IncidentManagement** in the Service Name field and click **Search**.
- 3 In the Expression tab, replace the last code line with the following:
 

```
if (hidden.meta.data in $L.file="Closed") then (problem.status in $L.file="Closed";status in $L.file="closed";if null(resolution.code in $L.file) then (resolution.code in $L.file="Automatically Closed");resolution in $L.file=insert(resolution in $L.file, 1, 1, "Closed by SMSAP integration."))
```
- 4 In the Fields tab, update the caption for severity to Urgency.



- 5 Click **Save**.

### Task 3: Configuring WSDL Mapping

See *Configuring WSDL Mapping* on page 60.

### Task 4: Setting default closure code and resolution when updating from event service

Do the following:

- 1 Click **Tailoring** → **Event Service** → **Maps**.
- 2 Enter **hp sap problem update** in the Map Name field. Enter **2** in the Position field and click **Search**.
- 3 In the Expressions tab, locate the code line starting with `if ($hmd="Closed")`. Replace this line with the following:

```
if ($hmd="Closed") then (problem.status in $axces.target="Closed";status
in $axces.target="closed";if null(resolution.code in $axces.target) then
(resolution.code in $axces.target="Automatically Closed");resolution in
$axces.target=insert(resolution in $axces.target, 1, 1, "Closed by SMSAP
integration."))
```
- 4 Click **Save**.

### Task 5: Updating the exchange process

Do the following:

- 1 Click **Tailoring** → **Document Engine** → **Processes**.
- 2 Enter **im.exchange.incident** in the Process Name field and click **Search**.
- 3 In the Initial Expressions tab, append the following scripts to the end of the codes:

```
if ($L.action="processincident") then (update.action in
$L.file=insert(update.action in $L.file, 1, 1, scmsg(3, "SMSAP", {number
in $L.file}));update.action in $L.file=insert(update.action in $L.file, 1,
1, $L.stamp))

if ($L.action="acceptincidentprocessing") then (update.action in
$L.file=insert(update.action in $L.file, 1, 1, scmsg(7,
"SMSAP"));update.action in $L.file=insert(update.action in $L.file, 1, 1,
$L.stamp))

if ($L.action="rejectincidentsolution") then (update.action in
$L.file=insert(update.action in $L.file, 1, 1, scmsg(8,
"SMSAP"));update.action in $L.file=insert(update.action in $L.file, 1, 1,
$L.stamp))
```
- 4 Click **Save**.

### Task 6: Updating the HPSAPTrigger script

Do the following:

- 1 Click **Tailoring** → **Script Library**.
- 2 Enter **HPSAPTrigger** in the Name field and click **Search**.
- 3 Replace the content with the **[HPSAPTrigger]** section in `code_sm9.txt`.
- 4 Click **Save**.

### Task 7: Updating link line in probsummary

Do the following:

- 1 Click **Tailoring** → **Tailoring Tools** → **Links**.
- 2 Enter **probsummary** in the Name field and click **Search**.
- 3 Locate the line with `device` as Target File Name. Click **More** → **Select Line**.
- 4 Change the scripts in the Post Expressions to the following:

```
if (sap.incident.type in $File="sapinstance") then ($continue=true) else
($continue=false)
```
- 5 Click **Save**.
- 6 Locate the line with `joinsapinstance` as Target File Name. Click **More** → **Select Line**.
- 7 In Source Field (Fill To/Post From), change **type** to **sap.incident.type**.
- 8 Click **Save**.

### Task 8: Updating format control - probsummary

Do the following:

- 1 select **Tailoring** → **Format Control** and search for **probsummary** → **Subroutines**, then right-click and select **Show Expanded Form** from the pop-up menu.
- 2 Update the Add field and the Update field in the `fill.fc` application as shown below:

Field	Value
Add ( <b>[FILL.FC]</b> in <code>code_sm9.txt</code> )	<code>null(logical.name in \$file) and not null(sap.sid in \$file)</code>
Update ( <b>[FILL.FC]</b> in <code>code_sm9.txt</code> )	<code>null(logical.name in \$file) and not null(sap.sid in \$file)</code>

### Task 9: Creating system messages

Do the following:

- 1 Enter **scmsg** in the command field and click **Execute Command**. The Search Message Records page opens.
- 2 Enter each of the following messages, and click **Add**:

Language Code	Class	Message Number	Severity	Text
en	SMSAP	1	1	SAP Solution Manager has received Incident %S from Service Manager. This incident is "%S".
en	SMSAP	2	1	Select a SAP Solution Manager before sending the incident.
en	SMSAP	3	1	Incident ID at external helpdesk is %S
en	SMSAP	4	1	Incident %S's sending is failed: %S

Language Code	Class	Message Number	Severity	Text
en	SMSAP	5	1	%S: (trigger #%S: Incident %S;%S;%S
en	SMSAP	6	1	---Automatically send to SAP:
en	SMSAP	7	1	Send Back from External Service Desk.
en	SMSAP	8	1	Refuse Solution from External Service Desk.

#### Task 10: Adding SMSAP SMIS template

Before managing the SM-SAP integration in SMIS, do the following to create a template for the SM-SAP integration:

- 1 Click **Tailoring** → **Script Library**.
- 2 Enter **SMSAPManager** in the Name field and then click **Add**.
- 3 Copy and paste the following scripts for SMSAPManager, and then click **Compile** to verify the syntax.

```
//SMSSM1.10 SM9.3
var Class = lib.smis_Prototype.getClass();

var SMSAPManagerClass = Class.create(lib.smis_Manager.getClass(), {

    isScheduleBased: function() {
        return false;
    }
});

function getClass() {return SMSAPManagerClass;}
```

- 4 Click **Save** and **OK**.
- 5 Click **Tailoring** → **Database Manager**.
- 6 Enter **SMISRegistry** in the Table field and then click **Search**.
- 7 Enter the following values to the fields in the General tab:

Field	Value
Name	SMSAP
Version	1.10
Manager Class Name	SMSAPManager
SM Adapter	smis_DummyAdapter
Endpoint Adapter	smis_DummyAdapter

Field	Value
Instance Count	1
Category	UI-based
Description	Service Manager integrate with SAP Solution Manager

- 8 Add the following parameter in the Parameter tab:

Name	Value	Category
baseurl	http://<host>:<port>/ovictex/servlet/OvHDTrigger	General

- 9 Click **Add** to save the template.

#### Task 11: Adding the SMSAP field in info table

For every SMIS integration, a field with the integration name should be created in info table.

- 1 Select **System Definition** → **Tables** → **info** → **Tab Fields and Keys**.
- 2 Create a field in SMIS structure. The Field Name is **SMSAP** and the Data Type is **Logical**.
- 3 Click **Save**.

#### Task 12: Customizing incident processes

To control the visibility of SAP Solution Manager according to the SMSAP instance's status, do the following to customize the incident processes for `im.view.init` and `im.open.setup`:

- 1 Click **Tailoring** → **Document Engine** → **Processes**.
- 2 Enter `im.view.init` and click **Search**.
- 3 In the Initial Expressions tab, append the following scripts to the end of the codes:

```
$SMSAP=nullsub(SMSAP in $G.system.info, "false")
```

- 4 In the Initial Javascript tab, copy and paste the following scripts:

```
var configItem =
lib.smis_ConfigurationManager.getEnabledConfigItem("SMSAP");
if (configItem != null) {
    var solMans =
configItem.getParametersByCategory("SolutionManager");

    var values = [];
    var names = [];

    for (var id in solMans) {
        values.push(id);
```

```

        names.push(solMans[id]);
    }

    system.vars.$G_solMans_values=values;
    system.vars.$G_solMans_names=names;
}

```

- 5 Click **Save** and **OK**.
- 6 Enter **im.open.setup** and click **Search**.
- 7 Add the following code to the Initial Expressions tab:

```
$SMSAP=false
```

- 8 Click **Save** and **OK**.

### Task 13: Updating display options

Check SMIS enabled/disabled status for all exchange buttons. Do the following to add a statement to the condition of each display option:

- 1 Click **Tailoring** → **Tailoring Tools** → **Display Options**.
- 2 Enter **processincident** and click **Search**.
- 3 In the Condition field, append the following script to the end of the codes:
 

```
and nullsub(SMSAP in $G.system.info, "false")
```
- 4 Click **Save**.
- 5 Repeat step 3 and step 4 for other display options, which include **closeincident**, **verifyincidentsolution**, **rejectincidentsolution**, **addinfo** and **acceptincidentprocessing**.

### Task 14: Adding condition in formats

Do the following to add SMIS status condition of each format:

- 1 Click **Tailoring** → **Forms Designer**.
- 2 Enter **IM.open.incident** in the Form field and click **Search**.
- 3 Click **Design**.
- 4 Select the SAP Solution Manager section, and enter **[\$SMSAP]=true** in the Visible Condition field of the Properties tab.
- 5 Click **OK**.
- 6 Click **OK**.
- 7 Repeat step 2 to step 6 for other display formats, which include **IM.update.incident** and **IM.close.incident**.

### Task 15: Updating the SAP Solution Manager subform

The SAP Solution Manager clients are configured in SMIS. Do the following to update the SAP Solution Manager subform:

- 1 Click **Tailoring** → **Forms Designer**.
- 2 Enter **hp.sap.solution.sub** in the Form field and click **Search**.

- 3 Click **Design**.
- 4 Click **Comfill**.
- 5 In the Property tab, clear the checkbox for the Third Button Visible field.
- 6 Remove the value in the Value List Condition field and the Display List Condition field.
- 7 Type **\$G.solMans.values** in the Value List field.
- 8 Type **\$G.solMans.names** in the Value List field.
- 9 Click the **Hidden Metadata** label and update the caption to **Exchange Status**.
- 10 Click **OK**.
- 11 Click **OK**.

#### Task 16: Disabling the duplicate button for exchange incidents

Do the following to disable the duplicate button for exchange incidents:

- 1 Click **Tailoring** → **Tailoring Tools** → **Display Options**.
- 2 Enter **apm.edit.problem\_clone** in the Unique ID field and click **Search**.
- 3 In the Condition field, append the following script to the end of the codes:  
and hidden.meta.data in \$L.filed=NULL
- 4 Click **Save** and **OK**.

#### Task 17: Renaming labels in Service Manager

Do the following to rename Reject Solution to Refuse Solution:

- 1 Click **Tailoring** → **Tailoring Tools** → **Display Options**.
- 2 Enter **apm.edit.problem\_rejectincidentsolution** in the Unique ID field and click **Search**.
- 3 In the Default Label field, rename **Reject Solution** to **Refuse Solution**.

Do the following to rename Close Incident to Close SAP Incident:

- 1 Click **Tailoring** → **Tailoring Tools** → **Display Options**.
- 2 Enter **apm.edit.problem\_closeincident** in the Unique ID field and click **Search**.
- 3 In the Default Label field, rename **Close Incident** to **Close SAP Incident**.

#### Task 18: Renaming icons

Do the following:

- 1 Rename `tclose_i.gif` to `tclose_s.gif`.
- 2 Rename `treject.gif` to `trefuse.gif`.

Refer to [Appendix D, Deploying Button Icons](#) for more information about where the buttons are stored.

#### Task 19: Checking if SAP Solution Manager is selected when clicking Send Incident

Do the following:

- 1 Click **Tailoring** → **Tailoring Tools** → **Display Options**.
- 2 Enter **apm.edit.problem\_processincident** in the Unique ID field and click **Search**.

- 3 In the Pre JavaScript tab, add the following script:
 

```
if (system.vars.$L_file.exthd==null || system.vars.$L_file.exthd=="")
system.functions.msg(system.functions.scmmsg( 2, "SMSAP" ),3);
```
- 4 Click **Save**.
- 5 Click **Tailoring** → **Document Engine** → **Processes**.
- 6 Enter **im.view** and click **Search**.
- 7 In the condition of processincident Display Action, add the following script:
 

```
not null(exthd in $L.file)
```
- 8 Click **Save**.

#### Task 20: Upgrading SMSSMEX middleware on Tomcat

Do the following:

- 1 Stop SMSSMEX V1.02.  
See *Starting/Stopping SMSSMEX* on page 26 to stop SMSSMEX V1.02.
- 2 Backup the configuration files.
  - a Backup all files in the <SMSSMEX\_installDir>\config folder.
  - b Backup the <SMSSMEX\_installDir>\tomcat\conf\server.xml file.
  - c Backup other configuration files which have been customized.
- 3 Uninstall SMSSMEX V1.02.  
See *Uninstall SMSSMEX* on page 18 to uninstall SMSSMEX V1.02.
- 4 Install SMSSMEX V1.10.  
See *Install SMSSMEX* on page 17 to install SMSSMEX V1.10.
- 5 Configure SMSSMEX V1.10.
  - a See *Configuring Tomcat* on page 19 to configure Tomcat.
  - b See *Configuring ovictex.properties* on page 23 to configure ovictex.properties.
  - c See *Configuring File ovictexInternal.properties* on page 24 to configure ovictexInternal.properties.
  - d See *External Helpdesks* on page 24 to configure external Helpdesks.
  - e See *Configuring FieldMapping.xml* on page 25 to configure FieldMapping.xml.

Or you can copy parameter values from the backup configuration files to configure SMSSMEX V1.10. Do not just copy and replace ovictex.properties and FieldMapping.xml because these files have been updated in SMSSMEX V1.10.
- 6 Start SMSSMEX V1.10.  
See *Starting/Stopping SMSSMEX* on page 26 to start SMSSMEX V1.10.

#### Task 21: Upgrading SMSSMEX middleware on Weblogic

Do the following:

- 1 Stop SMSSMEX V1.02 and the WebLogic server.
- 2 Backup the configuration files.

- a Backup all files in the <SMSSMEX\_installDir>\config folder.
  - b Backup other configuration files which have been customized.
- 3 Uninstall SMSSMEX V1.02.
- a See *Installing SMSSMEX* on page 27 to uninstall SMSSMEX V1.02.
  - b Remove `ovictex.war` and other unzipped files from the % SMSSMEX\_HOME%/war folder.
- 4 Install SMSSMEX V1.10.
- See *Installing SMSSMEX* on page 27 to install SMSSMEX V1.10.
- 5 Configure SMSSMEX V1.10.
- a See *Configuring ovictex.properties* on page 27 to configure `ovictex.properties`.
  - b See *Configuring File ovictexInternal.properties* on page 27 to configure `ovictexInternal.properties`.
  - c See *External Helpdesks* on page 27 to configure external Helpdesks.
  - d See *Configuring FieldMapping.xml* on page 27 to configure `FieldMapping.xml`.
- Or you can copy parameter values from the backup configuration files to configure SMSSMEX V1.10. Do not just copy and replace `ovictex.properties` and `FieldMapping.xml` because these files have been updated in SMSSMEX V1.10.
- 6 Deploy the `ovictex.war` file on WebLogic.
- See *Deploying on WebLogic* on page 28 to deploy the `ovictex.war` file on WebLogic.
- 7 Start SMSSMEX V1.10 and the WebLogic server.

Task 22: *Adding instance in SMIS and configuring parameters*

See *Adding Integration Name to the Info Table* on page 34 for more information.

## Upgrading SMSSMEX from V1.10 to V1.10 Patch 1

See *5 Applying SMSSMEX V1.10 Patch 1* on page 29



# A Incident Exchange Details

## Database Tables

The database tables required to operate the exchange service are created with the SQL scripts `create_tables_oracle.sql` or `create_tables_sqlserver.sql`.

**Table 7 Database tables required for exchange service**

Table	Field	Description
systemguid	systemguid	Unique system web service GUID
tasklist	ovhdid	ID of incident that triggered the action
	action	Action for the incident (state transitions of status diagram). Can be ProcessIncident, AddInfo, AcceptIncidentProcessing, RejectIncidentSolution, VerifyIncidentSolution, or CloseIncident.
	startTimestamp	Creation timestamp of entry.
	enqueueTimestamp	Timestamp for ordering of tasks. Initial value is startTimestamp.
	earliestReadyTimestamp	Timestamp that specifies the earliest time when this entry can be processed. Empty means immediately. Task becomes ready only after this time.
	state	Task state. Can be 1=READY or 2=INPROCESS (task is processed already).
	tries	Number of attempts to complete this task.
	guid	GUID of the task to delete the correct entry in database.
	sapid	Name of external help desk instance that incident is exchanged with.

**Table 7 Database tables required for exchange service (cont'd)**

runtimedat a	incidentguid	GUID of exchanged incident
	ovhdincidentid	ID of incident in helpdesk managed by web service
	exthdincidentid	ID of incident in the external helpdesk
	requesterguid	System GUID of requester helpdesk for that incident
	providerguid	System GUID of provider helpdesk for that incident
	metadata	Incident state in statement diagram and role the ServiceDesk has for this incident (Requester or Provider). Stored in the same format used for the Hidden_Meta_Data field in ServiceDesk (such as Requester:RequesterProcessing, Provider:SolutionProvided).
	infologid	Reference to multiple entries in runtimedata_infolog.
	attachmentid	Reference to multiple entries in runtimedata_attachments.
runtimedat a_infolog	lastchange	Timestamp of last change of entry.
	infologid	Key referenced from runtimedata.
	infologblock	Number of infolog block sent already.
runtimedat a_attachm ents	attachmentid	Key referenced from runtimedata.
	filename	Filename of an attachment for incident.
	attachmentguid	GUID for attachment (also known by external helpdesk) to delete attachment.

## Tools

There are several configuration tools in the installation \bin directory. Tool scripts are available for Windows (.bat) and Unix (.sh):

- `encryptPasswords` encrypts the passwords in the configuration file. All properties ending with `.password` must be configured with this tool. Use `-global` or `<instance key>` as a parameter.
  - `global`  
Encrypt a password in the global properties file (`ovictex.properties`). For example:  
`encryptPasswords.bat -global`
  - `<instance key>`  
Encrypt a password in the configuration file of a specific instance. For example:  
`encryptPasswords.bat exthd`
- `setup` is the setup script for Tomcat start/stop. For Tomcat

- Start: Setup startup
- Shutdown: Setup shutdown
- Start with debug mode: Setup debug startup
- checker checks the configuration in `ovictex.properties` and Service Manager configuration (see [Verifying Configuration](#) on page 25 for more information) .

## Field Mapping Configuration

Incident exchange web service exchanges incident data as XML documents between Service Manager and the external HelpDesk SAP Solution Manager. Incident exchange transforms the incident data in Service Manager to an XML message for SAP Solution Manager, and transforms Solution Manager data to an XML message for Service Manager. The transformation maps the field name in Service Manager to XML elements in Solution Manager while taking into account the following:

- Field names in Service Manager are usually different from the message element name.
- Service Manager field data type can differ from the message element data type.
- Not all message elements have corresponding data fields in Service Manager. Such fields are usually combined into a single log field called `Journal`.
- Some fields also require value mapping. For example, the possible values for the `Priority` field in Service Manager are **1 - Critical, 2 - High, 3 - Average, 4 - Low**. The Solution Manager Priority can be **5, 4, 3, 2, 1**. These values must be specified in the `FieldValueMapping` configuration.
- Service Manager can assign customized fields to an Incident. These fields can be mapped to message elements.

A declarative field mapping file defines the mapping outlined above and

- Enables the exchange of incident data between two helpdesks with reduced code size (the same code can handle any number of fields)
- Improves flexibility (mapping can be changed without changing code)
- Improves extensibility and customizability (a deployment-specific mapping can be added without changing code)
- Used to map incident data with an external helpdesk other than Solution Manager

### Types of Mapping

The mapping file supports field mapping and field value mapping. Field mapping is simple (XML message element is a single value) or composite (multiple values such as an array).

### Structure of FieldMapping XML file

The field mapping configuration is related to the `ICT_SERVICE_DESK_API` WSDL scheme defined by SAP Solution Manager. The mapping consists of field mapping and value mapping.

Field mapping includes:

- `IctHead`

- IctIncidentAttachment
- IctIncidentSapNotes
- IctIncidentSolutions
- IctIncidentUrls
- IctIncidentStatement
- IctIncidentAdditionalInfo

The following is a mapping file example:

```
<FieldMapping ExtHDFField="IctHead/AgentId" >
  <OutOvHDFField>AssigneeName</OutOvHDFField>
  <OutDataType>Person</OutDataType>
  <InOvHDFField>AssigneeName</InOvHDFField>
  <InDataType>Person</InDataType>
</FieldMapping>
```

In the above example:

- Element IctHead/AgentId of SAP Solution Manager (sub-element AgentId of top level element IctHead) maps to the field AssigneeName exposed by the Service Manager IncidentManagement Web Service.
- Data types for the IN and OUT exchange modes are specified in the InDataType and OutDataType tags.
- Person type indicates that the Exchange must convert incoming data (to/from the Service Manager) to/from an internal Person type that corresponds with the IctIncidentPerson type of the SAP SolutionManager web service.
- InDataType and OutDataType tags declare types on the Service Manager side.

## Composite Field Mapping

Composite field mapping maps a message element to a OvHD field depending upon the value of a sub-element (key) of the element ExtHDKeyField (OvHD and ExtHD are old terms; in this document, OvHD correspond to HP Service Manager and ExtHD correspond to SAP Solution Manager). A different value for the key defines mapping to a different Service Manager field. The following is a composite field mapping example.

```
<CompositeFieldMapping ExtHDFField="IctIncidentStatement"
  ExtHDKeyField="IctIncidentStatement/TextType">
<!-- For exchanging information log -->
<FieldMapping ExtHDFField="IctIncidentStatement/Text" >
  <InDataType>InformationLog</InDataType>
  <OutDataType>InformationLog</OutDataType>
  <KeyFieldOutVal>SU99</KeyFieldOutVal>
  <KeyFieldInVal>SU99</KeyFieldInVal>
</FieldMapping>
<!-- for exchanging Solution Provided -->
<FieldMapping ExtHDFField="IctIncidentStatement/Text" >
  <InOvHDFField>Resolution</InOvHDFField>
  <OutOvHDFField>Resolution</OutOvHDFField>
  <KeyFieldOutVal>SU99</KeyFieldOutVal>
  <KeyFieldInVal>SU01</KeyFieldInVal>
</FieldMapping>
<!-- for exchanging CustomText01 (as example) -->
```

```

<FieldMapping ExtHDFfield="IctIncidentStatement/Text" >
  <InOvHDFfield>CustomText01</InOvHDFfield>
  <OutOvHDFfield>CustomText01</OutOvHDFfield>
  <KeyFieldOutVal>SU99</KeyFieldOutVal>
  <KeyFieldInVal>SU77</KeyFieldInVal>
</FieldMapping>
<!-- For sending custom fields from OVHD to external HD create an entry as
the example below. Replace the place holder strings as per your
configuration -->
<!--
  <FieldMapping ExtHDFfield="IctIncidentStatement/Text" >
    <OutOvHDFfield>USER_VISIBLE_FIELDNAME_FOR_THAT_CUSTOM_FIELD
    </OutOvHDFfield>
    <KeyFieldOutVal>TEXT_TYPE_AS_DEFINED_BY_USER_FOR_THIS_FIELD
    </KeyFieldOutVal>
  </FieldMapping>
-->
</CompositeFieldMapping>

```

Element `IctIncidentStatement/Text` is mapped to the information log if the key element `IctIncidentStatement/TextType` is **SU99** or to Resolution field if the key element is **SU01** (for an incoming message).

This is used when a message has multiple occurrences of the same element that have different sub-element values. The sub-element is referred to as the key field. In the example above the `IctIncidentStatement/TextType` element is the key field. For a composite field mapping, every instance of `FieldMapping` has a unique `KeyFieldInVal`.

## Field Value Mapping

Field value mapping maps the values of a message element to the corresponding value of an OvHD field. The following is an example.

```

<FieldValueMapping Id="IctHead/Priority">
  <ValueMapping OvHDValue="4" ExtHDValue="5"/>
  <ValueMapping OvHDValue="4" ExtHDValue="4"/>
  <ValueMapping OvHDValue="3" ExtHDValue="3"/>
  <ValueMapping OvHDValue="2" ExtHDValue="2"/>
  <ValueMapping OvHDValue="1" ExtHDValue="1"/>
</FieldValueMapping>

```



Since both helpdesks priority lists can be configured, check the actual values in the field value mapping.

## Field Mapping Schema

The RelaxNG Compact Schema of the mapping file is shown below.

```

default namespace =
  "http://schemas.hp.com/openview/incidentExchange/mapping"
start =
  element IncidentExchMapping {
    attribute targetNamespace { xsd:anyURI },
    element FieldMappings {
      (FieldMapping

```

```

    | element CompositeFieldMapping {
        attribute ExtHDField { string },
        attribute ExtHDKeyField { string },
        FieldMapping+
    })+
} &
element ValueMappings {
    element FieldValueMapping {
        attribute Id { string },
        element ValueMapping {
            attribute ExtHDValue { string },
            attribute OvHDValue { string }
        }+
    }
}
}
}
FieldMapping =
    element FieldMapping {
        ## field accessor in XML document using XPath like notation. Example:
        ## ExtHDField="IctHead/AgentId"
        attribute ExtHDField { string },
        attribute ValueMappingId { string }?,
        (element InOvHDField { string } &
        (element DefaultOutOvHDField { string }
        | element OutOvHDField { string }))? &
        element InDataType { "InformationLog" | "Priority" | "Date" |
        "Attachment" | "OvCISearchKey" }? &
        element OutDataType { "Person" | "Priority" | "Date" | "Attachment" |
        "OvCISearchKey" }? &
        element KeyFieldOutVal { string }? &
        element KeyFieldInVal { string }? )
    }
}

```

The schema elements are described in the following table.

**Table 8 Schema element functionality**

Schema element	Function
IncidentExchMapping	Top-level element of the mapping schema.
FieldMappings	Container element for all FieldMapping and CompositeFieldMapping elements.
ValueMappings	Container element for FieldValueMapping elements.
FieldMapping	Maps a message element to an OvHD field and includes type information for storing to and loading from OvHD. Optionally contains a reference to a FieldValueMapping element through attribute ValueMappingId. The value of this attribute must match the value of attribute Id in a FieldValueMapping element. When this reference is present, the information in the FieldValueMapping must be used to map field value.

**Table 8 Schema element functionality (cont'd)**

<b>Schema element</b>	<b>Function</b>
CompositeFieldMapping	Maps a message element to a OvHD field depending upon the value of a sub-element (key) of the element. A different key value defines mapping to a different OvHD field. The keyFieldInVal must be unique for each individual field mapping within a composite field mapping.
ValueMapping	Maps the OvHD value of a message element to an ExtHD value.
ExtHDField	Field accessor in XML message document in XPATH like expression that identifies a specific field of exchanged incident information.
InOvHDField	Indicates the OvHD field name where information received from the external helpdesk is written for a specific ExtHDField.
OutOvHDField	Indicates the OvHD field name whose value is sent to the external helpdesk for a specific ExtHDField.
DefaultOutOvHDField	If this element appears in a field mapping then the value of this element is taken as the default value sent to the external helpdesk for a specific ExtHDField. For example, if a mapping DefaultOutOvHDField is specified as DefaultUserId and OutDataType is specified as Person, the default user ID will be sent to the external helpdesk for a specific ExtHDField.
InDataType	Datatype for storing the field value to OvHD.
OutDataType	Datatype for loading the field value from OvHD.
InDataType and OutDataType	Specifies the method to call for reading/writing information from/to the incident using the OvHDAccess layer. InDataType and OutDataType are optional elements. If not specified, then the field types are assumed to be String. Otherwise the following data types can be specified: <ul style="list-style-type: none"> <li>• Priority: Priority of an incident</li> <li>• Date: A date field</li> <li>• Attachment: Refers to an attachment</li> <li>• Person: Indicates that the information is a person detail.</li> </ul>
OvCISearchKey	Indicates the information is used as a search key for CI in OvHD.
InformationLog	Applicable only for InDataType. Indicates the information should be appended to the Information Log.
KeyFieldInVal	Value stored in OvHD for the element used as the key field.
KeyFieldOutVal	Value sent to ExtHD for the element used as the key field.

## Default Field Mapping File and Customization

### Prerequisites

SMSSMEX operates with Service Manager based on the extended IncidentManagement Web Service and supports only the fields listed below (exposed in the Service Manager IncidentManagement WS).

**Table 9 SMSSMEX supported fields**

<b>Field</b>	<b>Type</b>	<b>Field</b>	<b>Type</b>
IncidentID	Text	Subcategory	Text
Category	Text	SLAAgreementID	Decimal
OpenTime	Datetime	PlannedEnd	Datetime
OpenedBy	Text	SiteCategory	Text
PriorityCode	Text	ProductType	Text
Severity	Text	ProblemType	Text
UpdatedTime	Datetime	ResolutionFixType	Text
PrimaryAssignment Group	Text	UserPriority	Text
ClosedTime	Datetime	Solution	Text
ClosedBy	Text	InitialImpact	Text
ClosureCode	Text	CustomText01	Text
ConfigurationItem	Text	CustomText02	Text
Location	Text	CustomText03	Text
IncidentDescription		CustomText04	Text
Resolution	Resolution	CustomText05	Text
AssigneeName	Text(OperatorID)	CustomText06	Text
Contact	Text(ContactID)	CustomText07	Text
JournalUpdates		CustomText08	Text
AlertStatus	Text	CustomText09	Text
ContactLastName	Text	CustomText10	Text
ContactFirstName	Text	SapSid	Text
Company	Text	SapClient	Text
BriefDescription	Text	SapInstallationNumber	Text



**Table 9 SMSSMEX supported fields (cont'd)**

TicketOwner	Text	HiddenMetaData	Text
UpdatedBy	Text	IsIncidentExchange	Boolean
IMTicketStatus	Text	attachments	Attachments

## Adding Fields to fieldMapping.xml

The default field mapping file (provided with the incident exchange web service) does not include all fields from the web service and can be extended. Any additional field mapping can be included in section `IctIncidentStatement`. The following is an example:

```
<FieldMapping ExtHDFfield="IctIncidentStatement/Text">
  <OutOvHDFfield>SC_WS_FIELDNAME</OutOvHDFfield>
  <KeyFieldOutVal>SOLMAN_FIELD_TYPE </KeyFieldOutVal>
</FieldMapping>
<FieldMapping ExtHDFfield="IctIncidentStatement/Text">
  <OutOvHDFfield>CustomText09</OutOvHDFfield>
  <KeyFieldOutVal>SU99 </KeyFieldOutVal>
</FieldMapping>
```

In the above example, the custom field defined in Service Manager is sent to the external HD, so `KeyFieldOutVal` is defined at the external helpdesk. No `InOvHDFfield` or `KeyFieldInVal` is specified since the example only sends to the external helpdesk.

IN/OUT data exchange requires definition of IN and OUT:

```
<FieldMapping ExtHDFfield="IctIncidentStatement/Text">
  <OutOvHDFfield>SC_WS_FIELDNAME1</OutOvHDFfield>
  <InOvHDFfield>SC_WS_FIELDNAME2</InOvHDFfield>
  <KeyFieldOutVal> SOLMAN_FIELD_TYPE1 </KeyFieldOutVal>
  <KeyFieldInVal> SOLMAN_FIELD_TYPE2 </KeyFieldInVal>
</FieldMapping>
```

In this example if the values of `SC_WS_FIELDNAME1` and `C_WS_FIELDNAME2` are the same, then the `OvHD` field is overwritten when information is sent from external helpdesk (1:1 field synchronization). For example:

```
<FieldMapping ExtHDFfield="IctIncidentStatement/Text">
  <OutOvHDFfield>CustomText09</OutOvHDFfield>
  <InOvHDFfield>CustomText09</InOvHDFfield>
  <KeyFieldOutVal> SU01</KeyFieldOutVal>
  <KeyFieldInVal>SU01</KeyFieldInVal>
</FieldMapping>
```

In the following example, `CustomText08` updates field `ZZ08` in SAP Solution Manager, but `ZZ08` updates `CustomText09` in Service Manager (does not overwrite `CustomText08`).

```
<FieldMapping ExtHDFfield="IctIncidentStatement/Text">
  <OutOvHDFfield>CustomText08</OutOvHDFfield>
  <InOvHDFfield>CustomText09</InOvHDFfield>
  <KeyFieldOutVal> ZZ08</KeyFieldOutVal>
  <KeyFieldInVal>ZZ08</KeyFieldInVal>
</FieldMapping>
```

## Additional Information

Section `IctIncidentAdditionalInfo` defines synchronization of CIs between SAP Solution Manager and Service Manager and defines the method for sending SAP Attributes from SAP Solution Manager.



The first part of the mapping describes CI mapping handling and must not be changed.

```
<FieldMapping ExtHDField="IctIncidentAdditionalInfo/AddInfoValue" >
  <OutDataType>OvCISearchKey</OutDataType>
  <InDataType>OvCISearchKey</InDataType>
  <KeyFieldOutVal>SAPSystemID</KeyFieldOutVal>
  <KeyFieldInVal>SAPSystemID</KeyFieldInVal>
</FieldMapping>
<FieldMapping ExtHDField="IctIncidentAdditionalInfo/AddInfoValue" >
  <OutDataType>OvCISearchKey</OutDataType>
  <InDataType>OvCISearchKey</InDataType>
  <KeyFieldOutVal>SAPSystemClient</KeyFieldOutVal>
  <KeyFieldInVal>SAPSystemClient</KeyFieldInVal>
</FieldMapping>
<FieldMapping ExtHDField="IctIncidentAdditionalInfo/AddInfoValue" >
  <OutDataType>OvCISearchKey</OutDataType>
  <InDataType>OvCISearchKey</InDataType>
  <KeyFieldOutVal>SAPInstNo</KeyFieldOutVal>
  <KeyFieldInVal>SAPInstNo</KeyFieldInVal>
</FieldMapping>
```

The following two attributes are used only when Solution Manager forwards an Incident to SAP Solution Manager.

```
<FieldMapping ExtHDField="IctIncidentAdditionalInfo/AddInfoValue" >
  <InDataType>InformationLog</InDataType>
  <KeyFieldInVal>SAPIncidentID</KeyFieldInVal>
</FieldMapping>
<FieldMapping ExtHDField="IctIncidentAdditionalInfo/AddInfoValue"
  ValueMappingId="IctIncidentAdditionalInfo/AddInfoValue
  /SAPIncidentStatus" >
  <InDataType>InformationLog</InDataType>
  <KeyFieldInVal>SAPIncidentStatus</KeyFieldInVal>
</FieldMapping>
```

The only attributes that do not have read-only status in the SAP Solution Manager are CI attributes, allowing IN-mode mapping (from SAP Solution Manager to Service Manager). The following table defines the available attributes:

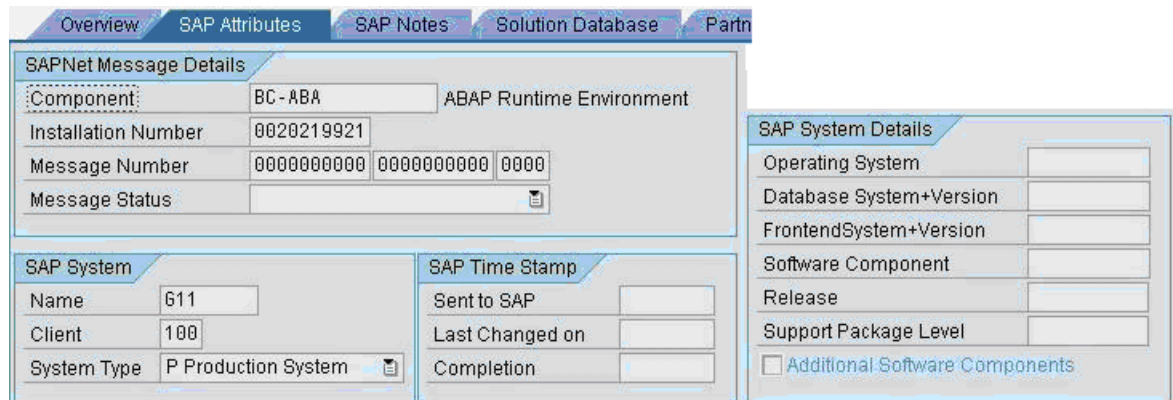
**Table 10 Attribute Types of the SAP Solution Manager**

AttributeType	Description
SAPComponent	SAP Component (e.g. SV-SMG-SUP)
SAPSystemID	SAP System ID
SAPSystemClient	Client of SAP System
SAPCategory	Category of the Incident
SAPSystemType	SAP System Type

**Table 10 Attribute Types of the SAP Solution Manager**

SAPInstNo	SAP Installation Number
SAPSubject	Subject of the Incident
SAPOperatingSystem	Operating System of SAP System
SAPDatabase	Database of SAP System
SAPFrontend	Frontendsystem and Version
SAPSoftwareComponent	Software Component
SAPSoftwareComponentRelease	Software Component Release
SAPSoftwareComponentPatch	Software Component Patch
SAPIncidentID	ID of the Incident at SAP (when forwarded to SAP)
SAPIncidentStatus	Status of the Incident at SAP (when forwarded to SAP)

In the SAP GUI most attributes are in the SAP Attributes tab.



The following example writes all incoming additional values of type SAPDatabase to the Journal in Service Manager:

```
<FieldMapping ExtHDFfield="IctIncidentAdditionalInfo/AddInfoValue" >
  <InDataType>InformationLog</InDataType>
  <KeyFieldInVal>SAPDatabase</KeyFieldInVal>
</FieldMapping>
```

The following example updates field CustomText03.

```
<FieldMapping ExtHDFfield="IctIncidentAdditionalInfo/AddInfoValue" >
  <InOvHDFfield>CustomText03</InOvHDFfield>
  <KeyFieldInVal>SAPDatabase</KeyFieldInVal>
</FieldMapping>
```

## Changeable Mappings

The following mappings can be modified.

**Table 11 Changeable mappings**

Mapping	Description
IctHead/AgentID IctHead/ReporterID IctIncidentAttachment/PersonId IctIncidentStatement/PersonId	OutOvHDField/InOvHDField field name can be modified if the replacement field contains the ID of a Contact joined with a contact table that is exposed via ConfigurationManagement Web Service (defined in the default configuration). The AssigneeName field contains the operator name of Service Manager instead of the contacts name.
IctHead/ShortDescription	OutOvHDField/InOvHDField can be modified with any text field from Service Manager.
IctHead/RequestedEnd	Can be modified with any datetime field in the Service Manager.



**Required Mappings:** The following mappings are required and must not be changed.

- IctHead/Priority (the value mapping for this field mapping can be changed)
- IctIncidentSapNotes/item
- IctIncidentSolutions/item
- IctIncidentUrls/item
- IctIncidentAdditionalInfo/AddInfoValue (first 3 mappings)

## Person Synchronization Details

### SAP Solution Manager to Service Manager

Persons sent from SAP Service Manager can be mapped to person fields in Service Manager. When Person details are received, the corresponding contact record is found in Service Manager by querying the Configuration Management Web Service. The resolved contact ID must be set in the mapped field. The exchange web service describes persons with the following fields:

- Sex
- First name
- Last name
- Telephone
- Mobile phone
- Fax
- Email

Fields that are used to find persons in Service Manager:

- Email
- First name
- Last name

Persons are searched by all three fields. If no matching person is found in Service Manager or duplicates are found, then a notification is added to the Journal. For example, an empty email causes the following message in response to Journal updates:

```
Warning! Contact can not be found. Firstname,Lastname,Email fields should
not be empty. Invalid contact: FirstName: "Nicholas" LastName: "Brown"
Phone number: "(770) 954-4588" Fax number: "(770) 954-4590" ...
```

SMSSMEX does not create Persons or Contacts. An operator-type lookup is enforced only for the AssigneeName field.

Mapping from Service Manager to SAP Solution Manager is performed in the same way. The ID of the Person field in the Service Manager is used to make an additional call to Configuration Management WS to get all details about the Person. The collected data is forwarded to the Solution Manager. In SAP Solution Manager the ID of the Person is checked. If the ID is

- Known: Solution Manager assigns an existing record to the Incident.
- Not known: Solution Manager tries to resolve a Person via the email field. If this is not possible, a new Person is created.

## SMSSMEX Version

To find out the version of the SMSSMEX service in Tomcat, do one of the following:

- Open `<SMSSMEX_installDir>\tomcat\webapps\ovictex\WEB-INF\lib\ovictex.war` with a zip tool. The war file MANIFEST.MF file contains the version information.
- Go to the Status page.



## B Installing and Configuring SAPCRYLIB

To install SAPCRYLIB (see <https://service.sap.com/sap/support/notes/510007>) do the following:

- 1 Download SAPCRYLIB from the website “SAP Download Area - SAP Cryptographic Software” at [https://websmp101.sap-ag.de/~form/handler?\\_APP=00200682500000000917&\\_EVENT=DISPLAY](https://websmp101.sap-ag.de/~form/handler?_APP=00200682500000000917&_EVENT=DISPLAY).

- 2 Use `sapcar.exe` to extract the SAR file:

```
sapcar -xvf sarfile_name
```

- 3 Copy the extra files to `\usr\sap\[Instance folder]\DVEBMGS00\exe`.
- 4 In transaction `/nrz10` in the Profile field, select the profile with `prof.type` of “Instance profile”.
- 5 Select **Extended maintenance** in Edit Profile.
- 6 Click **Change**.
- 7 Add the following parameters:

```
ssf/name          = SAPSECULIB
ssf/ssfapi_lib    = $(DIR_EXECUTABLE)\sapcrypto.dll
```

- 8 Restart the system.
- 9 Go to transaction `/nsmicm`.
- 10 Select the menu entry **GOTO** and select **Services** or press **SHIFT+F1**.
- 11 If the HTTPS port is not listed, then configure the profile. Add or change the following parameter:

```
icm/server_port_2 PROT=HTTPS,PORT=[SSL Port]
```

- 12 In transaction `/nsmicm` select from the **Administration** → **ICM** → **Restart** → **Yes** to restart ICM.





# C Logging

The following describes the location of log files.

- Windows: If you start SMSSMEX from
  - **setup -startup**  
    %SMSSMEX\_HOME%/logs/smssmex.log.<date>
  - Tomcat  
    %SMSSMEX\_HOME%/tomcat/logs/smssmex.log.<date>
- Unix: If you start SMSSMEX from
  - **setup.sh -startup**  
    %SMSSMEX\_HOME%/logs/smssmex.log.<date>
  - Tomcat  
    %SMSSMEX\_HOME%/tomcat/logs/smssmex.log.<date>



## D Deploying Button Icons

SMSSMEX enhances the functionality of Service Manager by adding some buttons in incident form to trigger message exchange related actions. The icons for the buttons are provided additionally in the release package (under `<SMSSMEX1.10p1 Release Package>\icons` folder). You can deploy them to the Service Manager Client manually.

Service Manager has two client applications: Windows Client and Web Client. For each of the clients, the icons should be deployed separately.

### Windows Client

- SC 6.2.x

Copy button icons from `<SMSSMEX1.10p1 Release Package>\icons` folder to `<Client_Home>\plugins\com.peregrine.eclipse.user_6.2.x.x\icons\obj16`.

For example,

```
C:\Program Files\Peregrine Systems\ServiceCenter
6.2\Client\plugins\com.peregrine.eclipse.user_6.2.7.0\icons\obj16.
```

For more information, see page 110 of *ServiceCenter 6.2 Installation Guide*.

- Service Manager 9.2x / 9.3x

Copy button icons from `<SMSSMEX1.10p1 Release Package>\icons` folder to `<Client_Home>\plugins\com.hp.ov.sm.client.eclipse.user_9.xx\src\resources\icons\obj16`

For more information, see *Service Manager 9.20 Installation Guide*.

### Web Client

Copy the button icons from `<SMSSMEX1.10p1 Release Package>\icons` folder to the following locations:

- On Service Center 6.2x, `<WebApps_Root>\sc\images\obj16` directory. For example, `C:\apache-tomcat-5.0.28\webapps\sc\images\obj16`.
- On Service Manager 9.2x and 9.3x, `<WebApps_Root>\webtier-9.20\images\obj16` directory.



# E SAP System Landscape Directory Registration

System Landscape Directory is the central information repository for your system landscape (Software Catalogue). It contains information about all installable and installed components in a system landscape. This section describes how to register this integration into System Landscape Directory.

## Prerequisites

Service Landscape Directory is running.

## Registering System Landscape Directory

- 1 Browse to the *<SMSSMEX1.10p1 Release Package>* and copy the SLDReg folder to your computer.
- 2 Open the SLDReg folder. Modify the `HPSMISystem.properties` file according to the parameter descriptions in the file. For example, update the `ComputerName` variable to the host name which is running SMSSMEX.

*ComputerName = <your computer name>*

- 3 Run the following command to compile XML file:

```
java -cp SLDReg.jar com.hp.sm.sld.XMLGenerator
```

After execution, `HPSMI.xml` is generated.

- 4 Run the following command to register System Landscape Directory:

```
java -cp SLDReg.jar com.hp.sm.sld.Register <SLD_HOST> <SLD_HTTPPORT>  
<UserName> <Password>
```

In this command:

- `<SLD_HOST>` is the host name of the Service Landscape Directory server.
- `<SLD_HTTPPORT>` is the http port of the service landscape directory service.
- `<UserName>` is the name that you use to log in to the server.
- `<Password>` is the password that you use to log in to the server.

