

# HP ITSM Enterprise Suite

Software Version: 2015

## Release Notes

Document Release Date: May 2015  
Software Release Date: May 2015



## Legal Notices

## Warranty

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

The information contained herein is subject to change without notice.

## Restricted Rights Legend

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

## Copyright Notice

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

## Trademark Notices

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

AMD is a trademark of Advanced Micro Devices, Inc.

Intel and Xeon are trademarks of Intel Corporation in the U.S. and other countries.

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

## Documentation Updates

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

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

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

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to: <https://hpp12.passport.hp.com/hppcf/createuser.do>

Or click the **the Register** link at the top of the HP Software Support page.

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

## Support

Visit the HP Software Support Online web site at: <https://softwaresupport.hp.com>

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

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

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

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

<https://hpp12.passport.hp.com/hppcf/createuser.do>

To find more information about access levels, go to:

<https://softwaresupport.hp.com/web/softwaresupport/access-levels>

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



# Contents

Warranty .....	3
Restricted Rights Legend .....	4
Introduction .....	22
How to use this document .....	22
Components in the ITSM Enterprise Suite .....	24
System topology .....	25
Sizing requirements .....	30
Suggested administrator resources .....	33
Compability matrix .....	33
Installation guidance .....	34
Using Deployment Manager .....	34
Platform limitations .....	35
Create a Deployment Manager environment .....	35
Install Service Manager .....	36
Install and integrate Knowledge Management .....	38
Integrate Knowledge Management and the Service Portal .....	39
Install and integrate Smart Analytics .....	39
Install and connect UCMDB .....	40
Install Asset Manager .....	42
Installing Operations Manager i .....	43
Install Service Health Reporter .....	43
Preinstallation Tasks and Checklist .....	49
Install Business Service Management .....	76
Install Executive Scorecard .....	76
Enable the IT Business Analytics data sources and content packs .....	76
Integration guidance .....	78
Chapter 2: Integrate UCMDB and Service Manager using the enhanced adapter .....	79

Introduction .....	79
Who Should Read this Guide .....	79
Purpose of the Integration .....	80
Supported Use Cases .....	80
Enabling ITIL Processes .....	81
Managing Planned Changes .....	81
Managing Unplanned Changes .....	82
Retrieving Service Manager Ticket Information .....	82
Retrieving Actual State of UCMDB CIs .....	82
Accessing UCMDB CIs from Service Manager .....	83
Core Features .....	83
Push .....	83
Federation .....	84
Population .....	84
How CI information is Synchronized Between UCMDB and Service Manager .....	84
CI Information Usage .....	85
High-Level Components of the Integration .....	86
Relationships Between Integration Components .....	86
What Information Is Stored in UCMDB .....	87
What Information Is Stored in Service Manager .....	87
Integration Setup .....	87
Integration Requirements .....	88
How to Migrate Your Integration .....	89
Upgrade Service Manager to Version 9.40 .....	90
Upgrade UCMDB to Version 10.20 .....	90
Enable the RESTful APIs for Custom CI Types in Service Manager .....	90
Reconfigure an Integration Point Using the Service Manager Enhanced Generic Adapter in UCMDB .....	93
Update the Configurations for Custom CI Types in UCMDB .....	94
Task 1. Convert the mapping scripts from XSLT to XML and Groovy. ....	94
Task 2. Update the configuration files. ....	99
Task 3. Enable Push, Population and Federation for CI types. ....	101
Integration Setup Overview .....	101
HP Service Manager Setup .....	101
How to Create an Integration User Account .....	102
How to Add the UCMDB Connection Information .....	103

HP Universal CMDB Setup .....	104
How to Create an Integration Point in UCMDB .....	104
Centralized CI Management .....	107
Visual Mapping Tool .....	108
Populating UCMDB with Service Manager CI Data .....	109
How to Define Population Jobs in UCMDB .....	109
View Service Manager CI Data in UCMDB .....	112
How to Schedule CI Population Jobs .....	112
Pushing UCMDB CI Data to Service Manager .....	113
How to Define Data Push Jobs in UCMDB .....	113
How to Schedule Data Push Jobs .....	117
How to View UCMDB CI Data in Service Manager .....	118
How to View the Change History of the Primary CI of a Problem Record .....	119
Federating Service Manager Ticket Data to UCMDB .....	120
Federation Queries .....	120
Examples of Using Federation .....	120
Example 1: Federate All SM Incident Tickets .....	121
Example 2: Federate SM Incident Records that Affect a UCMDB Business Service CI .....	125
Example 3: Federate Incident, Change, and Problem Record Data from Service Manager for UCMDB CIs .....	133
Example 4: Retrieve Service Manager Records Related to a UCMDB CI .....	136
Multi-Tenancy (Multi-Company) Setup .....	138
Multi-Tenancy (Multi-Company) Support .....	139
Implementing Multi-Tenancy in the UCMDB-SM Integration .....	139
Mandanten SM Security Layer .....	140
What Multi-Tenant Information is Stored in UCMDB .....	140
What Multi-Tenant Information is Stored in Service Manager .....	140
Unique Logical Names .....	141
Synchronization of Company Records .....	141
UCMDB Customer ID .....	143
UCMDB User ID and Password .....	143
Company Code .....	143
CI Reconciliation Rules .....	144
Company Information Pushed to CI and CI Relationship Records .....	144
Company Information Replicated to Incident Records .....	144
Schedule Records .....	144

Tenant-Specific Discovery Event Manager (DEM) Rules .....	145
Multi-Tenancy Use Cases .....	145
Multi-Tenancy Requirements .....	146
Setting up the Multi-Tenancy Integration in UCMDB .....	147
How to Install a Separate Data Flow Probe for Each Tenant .....	147
How to Start Tenant-Specific Data Flow Probes .....	149
How to Configure IP Ranges for Tenant-Specific Data Flow Probes .....	149
Setting up the Multi-Tenancy Integration in Service Manager .....	150
How to Start the Schedule Process .....	151
How to Configure the Service Manager System Information Record .....	152
How to Add Tenant-Specific UCMDB User ID and Password Values .....	153
How to Add UCMDB Customer ID values to Existing Companies .....	154
How to Synchronize Existing Companies from Service Manager to UCMDB .....	154
How to View Whether Company Information Is in UCMDB .....	155
How to Resynchronize an Existing Company with UCMDB .....	156
How to Inactivate a Synchronized Company .....	157
How to Reactivate an Inactive Company .....	157
How to Add Tenant-Specific DEM Rules .....	158
Standards and Best Practices .....	158
UCMDB-SM Configuration Best Practices .....	158
CI Name Mapping Considerations .....	159
Bi-Directional Data Synchronization Recommendations .....	160
Push Scheduling Recommendations .....	162
Push in Clustered Environments .....	163
Dedicated Web Services .....	163
Step-by-Step Cluster Configuration Process .....	163
How to Configure Web Clients .....	164
How to Configure the Debugnode .....	164
Connecting to Multiple SM Processes .....	165
Initial Load Configurations .....	165
Push Performance in a Single-Threaded Environment .....	166
Implementing Multi-Threading .....	167
Push Performance in Multi-Threaded Environments .....	168
Push Performance in Multiple SM Processes Environments .....	168
How to Set up SM DEM Rules for Initial Loads .....	169
How to Configure Differential or Delta Load DEM Rules .....	170

Fault Detection and Recovery for Push .....	171
How to Enable Lightweight Single Sign-On (LW-SSO) Configuration .....	172
Frequently Asked Questions .....	172
When Is a New CI Created in Service Manager? .....	173
Can I Analyze the Reason for a CI Deletion in SM? .....	174
How Do I Monitor Relationship Changes Between UCMDB and SM? .....	174
What Kinds of Relationships are Pushed from UCMDB to SM? .....	174
What is a Root CI Node? .....	175
What Is a Root Relationship? .....	175
What is the “Virtual-Compound” Relationship Type Used in a UCMDB-SM Integration Query? .....	175
When Do I Need the Population Feature? .....	176
Can I Populate Physically Deleted CIs from SM to UCMDB? .....	176
How Do I Keep the Outage Dependency Setting of a CI Relationship in SM? .....	176
How Do I Create an XML Configuration File? .....	179
How Do I Use the Load Fields Button to Add Multiple Managed Fields? .....	180
What Is the Purpose of the <container> Element in the Population Configuration File (smPopConf.xml)? .....	180
Can I Populate Sub-Item Deletions? .....	181
What Happens if a Population Job Failed or Completed? .....	181
Tailoring the Integration .....	182
Integration Architecture .....	182
Integration Class Model .....	183
Integration Queries .....	183
Queries for Push .....	183
Queries for Actual State .....	186
Queries for Federation .....	186
Queries for Population .....	187
Query Requirements .....	188
Service Manager Web Services .....	188
Managed Fields .....	189
Service Manager Reconciliation Rules .....	193
Performance Implications .....	194
Dependence on DEM Rules .....	194
Service Manager Discovery Event Manager Rules .....	195
Change the Conditions Under Which a DEM Rule Runs .....	195
Change the Action the DEM Rule Takes .....	195

Create Custom JavaScript to Open Change or Incident Records .....	196
Default values to create a new CI .....	196
Default values to create a new change .....	196
Default values to create a new incident .....	197
Integration Tailoring Options .....	197
How to Update the Integration Adapter Configuration File (sm.properties) .....	198
How to Add DEM Reconciliation Rules .....	203
How to Add Discovery Event Manager Rules .....	205
DEM Rules .....	206
Action if matching record does not exist .....	206
Action if record exists but unexpected data discovered .....	207
Action if record is to be deleted .....	207
Duplication Rules .....	208
CI Attributes Displayed in Change and Incident Records .....	209
Searching for Change and Incident Records Opened by the Integration .....	210
How to Add a CI Attribute to the Integration for Data Push .....	210
How to Add the CI Attribute to the UCMDB Class Model .....	211
How to Add a CI Attribute to the Query Layout .....	212
How to Add a Web Service Field for the Service Manager CI Type .....	214
Add a Simple Attribute to the SM CI Type .....	215
Add an Array of Structure or Structure to the CI Type .....	217
How to Map the CI Attribute to a Service Manager Web Service Field .....	222
How to Add a CI Type to the Integration for Data Push .....	224
How to Add the CI Type to the UCMDB Class Model .....	225
How to Create a Query to Synchronize the CI Type .....	228
How to Add the CI Type's Attributes to the Query Layout .....	232
How to Add the CI Type to Service Manager .....	234
How to Map the CI Type's Attributes to Web Service Fields .....	237
How to Add a CI Relationship Type to the Integration for Data Push .....	241
How to Create a Query to Push a Relationship Type .....	242
How to Map a Relationship Type Query to the Service Manager Web Service Object .....	245
How to Create an XML Configuration File for a Relationship Type .....	246
How to Add a Custom Query to an Integration Job .....	248

How to Add a CI Type, Attribute or Relationship Type to the Integration for Population .....	249
How to Enable or Disable UCMDB ID Pushback for a CI Type .....	249
How to Add an Attribute of a Supported CI Type for Federation .....	251
Troubleshooting .....	257
Troubleshooting Data Push Issues .....	257
How to Check the Error Message of a Failed Push Job .....	258
How to Check the Error Messages of Failed CIs or Relationships in a Push Job .....	260
How to Check the Push Log File .....	261
Typical Push Errors and Solutions .....	262
Query not Configured in smPushConf.xml .....	262
Mapping File not Well Formed .....	264
Troubleshooting Population Issues .....	267
How to Check the Error Message of a Failed Population Job .....	267
How to Check the Population Log File .....	267
Typical Population Error Messages and Solutions .....	268
No TQL Query Configured in smPopConf.xml .....	268
Nonexistent Mapping File Name Defined for a TQL Query in smPopConf.xml .....	271
Troubleshooting Federation Issues .....	272
How to Check the Error Message of a Failed Federation Request .....	272
Typical Federation Error Messages and Solutions .....	273
Wrong Configuration for a Federation CI Type in smFedConf.xml .....	273
Mapping File for the Federation TQL Query Is not Well Formed .....	275
Using Service Manager and UCMDB .....	278
Enable an integration to HP Universal CMDB .....	279
Configuration item actual states .....	281
View the actual state of a configuration item .....	281
Reconciling configuration items between HP Service Manager and HP Universal CMDB .....	282
Create a DEM reconciliation rule .....	283
Multi-tenant (multi-company) support .....	284
HP Universal CMDB Configuration Manager .....	285
HP Universal CMDB Browser .....	285
Discovery Event Manager .....	286
Discovery Event Manager managed fields .....	286
Add a managed field in Discovery Event Manager .....	287
View, modify, or delete a managed field in Discovery Event Manager .....	288

Discovery Event Manager rules .....	289
Discovery Event Manager rule options .....	289
Add a rule in Discovery Event Manager .....	291
View or modify rules in Discovery Event Manager .....	292
Delete a set of rules in Discovery Event Manager .....	293
Add a configuration item in Discovery Event Manager .....	293
View, modify, or delete a configuration item in Discovery Event Manager .....	294
Customize changes in Discovery Event Manager .....	295
Customize incidents in Discovery Event Manager .....	295
Integrate uCMDB and Asset Manager .....	296
HP Asset Manager Integration with the AM Generic Adapter .....	296
Overview .....	297
Supported Versions .....	297
Architecture .....	297
How to Integrate UCMDB and Asset Manager .....	298
HP Asset Manager Setup .....	299
Validate Pre-Loaded Data in Asset Manager .....	299
Create an Account with Administrative Rights .....	299
Update Asset Manager Schema .....	300
Activate Workflows for Population .....	301
Prepare Asset Manager for Parallel Push .....	301
Create Asset Manager API Zip Package .....	302
HP UCMDB Setup .....	303
Deploy Asset Manager API Zip Package .....	303
Install a Database Client .....	303
Create an Integration Point in UCMDB .....	304
Out-of-Box Integration Jobs .....	306
Asset Manager Population Jobs .....	308
Asset Manager Push Jobs .....	310
Asset Manager Federation Configuration .....	317
Verify Out-of-Box Population and Push Jobs .....	319
Synchronize Data between UCMDB and Asset Manager .....	320



<b>How to View UCMDB Data in Asset Manager</b> .....	<b>322</b>
Nodes .....	322
Business Elements .....	322
<b>How to View Asset Manager Data in UCMDB</b> .....	<b>323</b>
<b>How to Federate Asset Manager Data in UCMDB</b> .....	<b>324</b>
<b>Integration Jobs Configuration</b> .....	<b>327</b>
<b>How to Schedule Data Integration Jobs</b> .....	<b>328</b>
<b>Edit Data Integration Jobs</b> .....	<b>329</b>
Standards and Concepts .....	330
Asset Manager Entity .....	330
<b>Asset Manager Entity Definition Steps</b> .....	<b>332</b>
<b>Import Tables and Entity Definition</b> .....	<b>333</b>
<b>Out-of-Box Entity Definition</b> .....	<b>335</b>
UCMDB TQL .....	336
Groovy Functions .....	337
<b>Basic Functions</b> .....	<b>338</b>
<b>AM Population Groovy</b> .....	<b>339</b>
<b>AM Push Groovy</b> .....	<b>341</b>
<b>Utility Functions</b> .....	<b>345</b>
<b>Reconciliation Functions</b> .....	<b>346</b>
Data Mapping Schema .....	346
Population and Federation .....	348
<b>Criteria for Asset Manager Records to be Populated</b> .....	<b>349</b>

Transformation for Asset Manager Records to be Populated .....	350
Reconciliation .....	354
Population Condition and Push Back Definition .....	355
Built-in attributes from AM .....	356
Federation Tags .....	357
Population Tags .....	358
Push and Reconciliation .....	358
Data Flow Architecture .....	359
Integration TQL Queries .....	360
Reconciliation Proposals .....	361
Asset Manager Rules and Flows .....	363
Mapping Attributes .....	364
Reconciliation .....	366
Target CI Validation .....	368
Reference Attribute .....	369
Attribute Reconciliation .....	370
Action on Delete .....	371
Enum Attribute .....	372
Ignored Attributes .....	373
Deletion .....	373
Population Deletion Configuration .....	374

<b>Push Deletion Configuration</b> .....	<b>376</b>
Installed Software .....	376
HP Asset Manager Push Integration .....	380
Quick Start .....	381
Overview .....	381
Supported Versions .....	383
How to Integrate UCMDB and Asset Manager .....	384
Validate Pre-Loaded Data in Asset Manager .....	384
Set Up Asset Manager .....	384
Set Up UCMDB .....	390
Push CI Data from UCMDB to Asset Manager .....	393
How to View UCMDB Data in Asset Manager .....	398
Nodes .....	398
Business Elements .....	399
How to Schedule Data Push Jobs .....	399
Installed Software .....	401
How to Tailor the Integration .....	403
Integration Architecture .....	404
Data Flow Architecture .....	404
Integration TQL Queries .....	405
Reconciliation Proposals .....	405
Asset Manager Rules and Flows .....	406
Data Mapping .....	406
Push Mapping .....	407
<b>Basic Information</b> .....	<b>408</b>
<b>Reconciliation</b> .....	<b>410</b>
<b>Target CI Validation</b> .....	<b>412</b>
<b>Reference Attribute</b> .....	<b>413</b>
<b>Attribute Reconciliation</b> .....	<b>414</b>
<b>Action on Delete</b> .....	<b>415</b>

<b>Enum Attribute</b> .....	<b>416</b>
<b>Ignored Attributes</b> .....	<b>417</b>
How to Change Adapter Settings .....	417
How to Customize an Existing Mapping .....	418
How to Add a New Mapping to the Integration .....	420
Frequently Asked Questions .....	424
Troubleshooting and Limitations .....	427
Logs .....	432
<b>HP Asset Manager Population Integration</b> .....	<b>433</b>
Overview .....	434
Supported Versions .....	434
How to Integrate Asset Manager with UCMDB .....	434
Verify UCMDB to AM Configuration .....	440
What CI data is populated from AM to UCMDB? .....	441
Population TQLs .....	441
Criteria for AM records to be propagated .....	442
What is created in UCMDB during population? .....	445
Reconciliation .....	446
What happens when changes occur in AM during data population? .....	446
Supported CI Types .....	446
Supported CI attributes and the mapping with the AM fields .....	447
CI Type: Asset .....	447
CI Type: IpAddress .....	447
CI Type: Node .....	448
CI Type: CPU .....	449
CI Type: DiskDevice .....	450
CI Type: FileSystem .....	450
CI Type: InstalledSoftware .....	450
CI Type: Interface .....	450
CI Type: LogicalVolume .....	451
CI Type: Printer .....	451
CI Type: Location .....	451
CI Type: BusinessElement .....	451
The configuration files used by the integration .....	453
Where are the configuration files located .....	453

discriminator.properties .....	453
server_virtual_distinguisher.properties .....	453
server_desktop_distinguisher.properties .....	454
fixed_values.txt .....	455
location_type_transformer.xml .....	456
condition_rules.xml .....	456
Global_id_mapping.properties .....	457
Integrate BSM and OMi .....	457
Integrate Service Manager to OMi .....	457
Incident Exchange (OMi - SM) integration .....	458
Incident Exchange (OMi - SM) integration setup .....	459
Create user accounts for the Incident Exchange (OMi - SM) integration .....	460
Configure the Service Manager server as a connected server in Operations Manager i (OMi) .....	461
Configure an event forwarding rule in Operations Manager i (OMi) .....	464
Enable incident drill-down from Operations Manager i (OMi) Event Browser .....	465
Configure SSL for the Incident Exchange (OMi - SM) integration .....	465
Configure the Instance Count in the SMOMi integration template .....	466
Add an integration instance for each Operations Manager i (OMi) server .....	467
Enable LW-SSO for the Incident Exchange (OMi - SM) integration .....	473
Configure automatic closure for OMi incidents .....	474
Change the default assignment group for OMi incidents .....	477
Synchronization of incident changes back to Operations Manager i (OMi) .....	478
Working with the Incident Exchange (OMi - SM) integration .....	479
View related OMi event details from an incident .....	479
Mark an incident for automatic closure .....	480
Operations Manager i - Service Manager Integration .....	481
Operations Manager i - Service Manager Integration Overview .....	482
Downtime Exchange Between Operations Manager i and Service Manager .....	484
Integration Overview .....	484
Step 1: Send OMi Downtime Events to SM .....	485
Step 2: Integrate SM Downtimes with OMi .....	487
Incident Exchange Between Service Manager and Operations Manager i .....	490
Step 1: Configure the SM Server as a Connected Server .....	490
Step 2: Configure an Event Forwarding Rule .....	494
Step 3: Configure a URL Launch of the Event Browser from SM .....	496

Step 4: Configure a URL Launch of SM from the Event Browser .....	497
Step 5: Configure the SM Server .....	498
Step 6: Mapping and Customization .....	499
Step 7: Test the Connection .....	500
Step 8: Synchronize Attributes .....	501
Tips for Customizing Groovy Scripts .....	502
<b>View Changes and Incidents in Service Health Using Standalone HP Universal CMDB ....</b>	<b>506</b>
Prerequisite .....	507
Step 1: Load the .unl File to Provide External Access to Service Manager .....	507
Step 2: Configure the Service Desk Adapter Time Zone .....	508
Step 3: Configure UCMDB to Generate Global IDs .....	510
Step 4 (for SM 9.2x only): Add a Domain .....	510
Step 5: Configure SM Adapter in UCMDB .....	511
Step 6: Configure the SM-UCMDB Integration: Create an Integration Point .....	511
Step 7: Configure the SM-UCMDB Integration: Set Up Data Push Jobs .....	513
Step 8: Configure the SM-UCMDB Integration: Run Data Push Jobs .....	513
Step 9: Configure the SM-UCMDB Integration: Add UCMDB Connection Information to SM .....	514
Step 10: Configure the OMi-UCMDB Integration: Deploy CMS_to_RTSM_Sync.zip on UCMDB .....	514
Step 11: Configure the OMi-UCMDB Integration: Create an Integration Point on OMi .....	515
Step 12: Configure the OMi-UCMDB Integration: Create an Integration Point on the CMS .....	517
Step 13 (Optional): Add CI Types to the Service Health Changes and Incidents Component .....	520
Step 14 (Optional): Map Siebel Application CITs .....	520
Troubleshooting .....	520
<b>View Changes and Incidents in Service Health Using RTSM .....</b>	<b>522</b>
Prerequisite .....	522
Step 1: Configure the Service Desk Adapter Time Zone .....	522
Step 2: Create an Integration User Account in Service Manager .....	524
Step 3: Add the OMi Connection Information in SM .....	525
Step 4: Create an Integration Point in OMi .....	525
Step 5: Create New Jobs to Synchronize Between OMi and SM .....	527
Step 6: Run the Job .....	527
Step 7: Test the Configuration .....	528

Step 8 (Optional): Add CI Types to the Service Health Changes and Incidents Component .....	530
Troubleshooting .....	530
How to Customize the Changes and Incidents Component .....	531
Naming Constraints for New Request for Change TQLs .....	532
Naming Constraints for New Incident TQLs .....	533
Generate Incidents in SM When an OMi Alert is Triggered .....	534
Integrate BSM and SM .....	534
BSM - Service Manager Integration Overview .....	535
Downtime Exchange Between BSM and HP Service Manager .....	537
Integration Overview .....	537
Prerequisites .....	538
Step 1: Send BSM Downtime Events to Service Manager .....	539
Step 2: Integrate Service Manager Downtimes With BSM .....	542
Configuring HP Service Manager to Send Downtimes .....	542
Integrating SM RFC Downtimes with RTSM/uCMDB .....	544
Push CIT ScheduledDowntime to CIT BSMDowntime by BSMDowntimeAdapter .....	546
Incident Exchange between HP Operations Manager i and HP Service Manager .....	549
Step 1: Configure the HP Service Manager Server as a Connected Server .....	549
Step 2: Configure an Event Forwarding Rule .....	553
Step 3: Configure URL Launch of Event Browser from HP Service Manager .....	555
Step 4: Configure URL Launch of HP Service Manager from the Event Browser .....	556
Step 5: Configure HP Service Manager Server .....	557
Step 6: Mapping and Customization .....	558
Step 7: Test the Connection .....	559
Step 8: Synchronize Attributes .....	560
Tips for Customizing Groovy Scripts .....	561
View Changes and Incidents in Service Health Using Standalone HP Universal CMDB .....	565
Prerequisites .....	566
Step 1: Load .unl Files to Provide External Access to Service Manager .....	566
Step 2: Configure the Service Desk Adapter Time Zone .....	568
Step 3: Verify that the UCMDB is the Global ID Generator .....	569
Step 4 (for SM 9.20 and earlier only): Add a Domain .....	570
Step 5: Configure SM Adapter in UCMDB .....	570
Step 6: Configure the SM-UCMDB Integration: Create an Integration Point .....	571
Step 7: Configure the SM-UCMDB Integration: Set Up Data Push Jobs .....	572

Step 8: Configure the SM-UCMDB Integration: Run Data Push Jobs .....	573
Step 9: Configure the SM-UCMDB Integration: Add UCMDB Connection Information to SM .....	573
Step 10: Configure the BSM-UCMDB Integration: Deploy CMS_to_RTSM_Sync.zip on UCMDB .....	574
Step 11: Configure the BSM-UCMDB Integration: Create an Integration Point on BSM .....	574
Step 12: Configure the BSM-UCMDB Integration: Create an Integration Point on the CMS .....	577
Step 13 (Optional): Add CI Types to the Service Health Changes and Incidents Component .....	579
Step 14 (Optional): Map Siebel Application CITs .....	579
Result .....	580
Troubleshooting .....	580
View Changes and Incidents in Service Health Using RTSM .....	581
Prerequisite .....	581
Step 1: Configure the Service Desk Adapter Time Zone .....	581
Step 2: Create an Integration User Account in Service Manager .....	583
Step 3: Add the BSM Connection Information in Service Manager .....	584
Step 4: Create an Integration Point in BSM .....	584
Step 5: Create New Jobs to Synchronize Between BSM and Service Manager .....	587
Step 6: Run the Job .....	587
Step 7: Test the Configuration .....	587
Step 8 (Optional): Add CI Types to the Service Health Changes and Incidents Component .....	590
Troubleshooting .....	590
How to Customize the Changes and Incidents Component .....	591
Naming Constraints for New Request for Change TQLs .....	592
Naming Constraints for New Incident TQLs .....	592
Generate Incidents in Service Manager When a BSM Alert is Triggered .....	594
CI Status Alerts .....	594
SLA Alerts .....	594
EUM Alerts .....	595
View Incident Data in BSM, and Manage SLAs Based on Service Manager .....	596
Overview: Understanding the Integration with EMS .....	596
Prerequisites .....	600
Step 1: Enable Access to HP Service Manager From Within Service Health .....	601
Step 2: Define HP Service Manager Tables for External Access to the Clocks .....	601



Step 3: Correct the Clocks WSDL .....	602
Step 4: Add the Type Field to the logical.name Link Line .....	603
Step 5: Create a Corresponding HP Service Manager User .....	604
Step 6: Configure the HP Service Manager Monitor in SiteScope .....	604
Step 7: Specify the HP Service Manager Web Tier URL in the Infrastructure Settings ...	606
Step 8: Customize the HP Service Manager EMS Integration Adapter and Check the Assignment – Optional .....	606
Step 9: Specify the State and Severity of Open Incidents to Be Displayed – Optional ...	607
Step 10: Include HP Service Manager CIs in Service Level Management Agreements ...	608
Results .....	608
Integrate Service Health Reporter to BSM .....	609
<b>Next steps .....</b>	<b>610</b>
<b>License information .....</b>	<b>611</b>
Required licenses .....	611
<b>ITSM Enterprise Suite file list .....</b>	<b>613</b>
<b>Glossary .....</b>	<b>618</b>
<b>Index .....</b>	<b>619</b>
<b>Send Documentation Feedback .....</b>	<b>620</b>

## Introduction

The HP ITSM Enterprise Suite provides an extensive service management solution that includes incident management, problem management, change management, request fulfillment, event management, and knowledge management, as well as service level management, service catalog management, service portfolio management and service asset & configuration management. In addition to core process functionality, the suite includes extensive monitoring and reporting capabilities that support proactive management activities and availability management.

The flexible design of the ITSM Enterprise Suite enables modules to be used independently whilst still being completely relational. This allows customers to build on their service management capabilities within timescale and budget constraints.

## How to use this document

The service management solution system described in this document is a very large and complex one. The Release Notes contains many complex and large products, each of which are complex systems in their own right. Each of these products have their own documentation, installation and integration steps. Due to large volume of the documentation, it is not within the scope of this document to re-create all of that information. Instead, the purpose of this document is to provide an overall example scheme of what a complete ITSM solution might look like. It will define a system topology including many of the products (but not all) contained within the ITSM Enterprise Suite, and how to deploy those products in such a way as to provide a complete service management solution.

As such, this document will provide guidance and installation steps that describes one possible service management solution. The intent is to show an example that can be tailored for your organization's needs. In many cases, this document will refer you to the associated product documentation. For example, this document will not describe all possible steps to install Operations Manager i or IT Business Analytics. This document will refer you to the product documentation for the installation and deployment of these products. Further, due to the use of Deployment Manager, it will not describe how to install Service Manager, instead relying on Deployment Manager.

## Links to all referenced documentation

### **Service Manager:**

- *HP Service Manager Help Center*

**Asset Manager:**

*HP Asset Manager 9.50 Release Notes*

<https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM01446907>

**Universal Configuration Management Database:**

*HP Universal CMDB 10.20 Deployment Guide*

<https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM01364377>

*HP Universal CMDB 10.20 Support Matrix*

<https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM01364276>

*HP Universal CMDB 10.20 Discovery and Integrations Content - HP Integrations*

<https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM01367254>

HP Universal CMDB 10.20 All PDFs

<https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM01502033>

**Operations Manager i:**

*HP Operations Manager i 10.01 Installation and Upgrade Guide*

<https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM01223598>

*HP Operations Manager i Integrations Guide*

<https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM01223606>

**Business Service Management:**

*Business Service Management 9.25 Installation Guide*

<https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM01134334>

*Business Service Management 9.25 Integration: Service Manager Guide*

<https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM01357692>

*Business Service Management 9.25 System Requirements and Support Matrices*

<https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM01134344>

**Service Health Reporter:**

*Service Health Reporter 9.40 Interactive Installation Guide*

<https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM01273124>

*Service Health Reporter 9.40 Support Matrix*

<https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM01273123>

*Service Health Reporter 9.40 Integration Guide*

<https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM01403734>

**IT Business Analytics (formerly Executive Scorecard):**

*IT Business Analytics 9.50 Installation and Configuration Guide*

<https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM01275262>

*IT Business Analytics 9.50 Content Reference Guides*

<https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM01010240>

*IT Business Analytics 9.50 Support Matrix*

<https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM01010277>

## Components in the ITSM Enterprise Suite

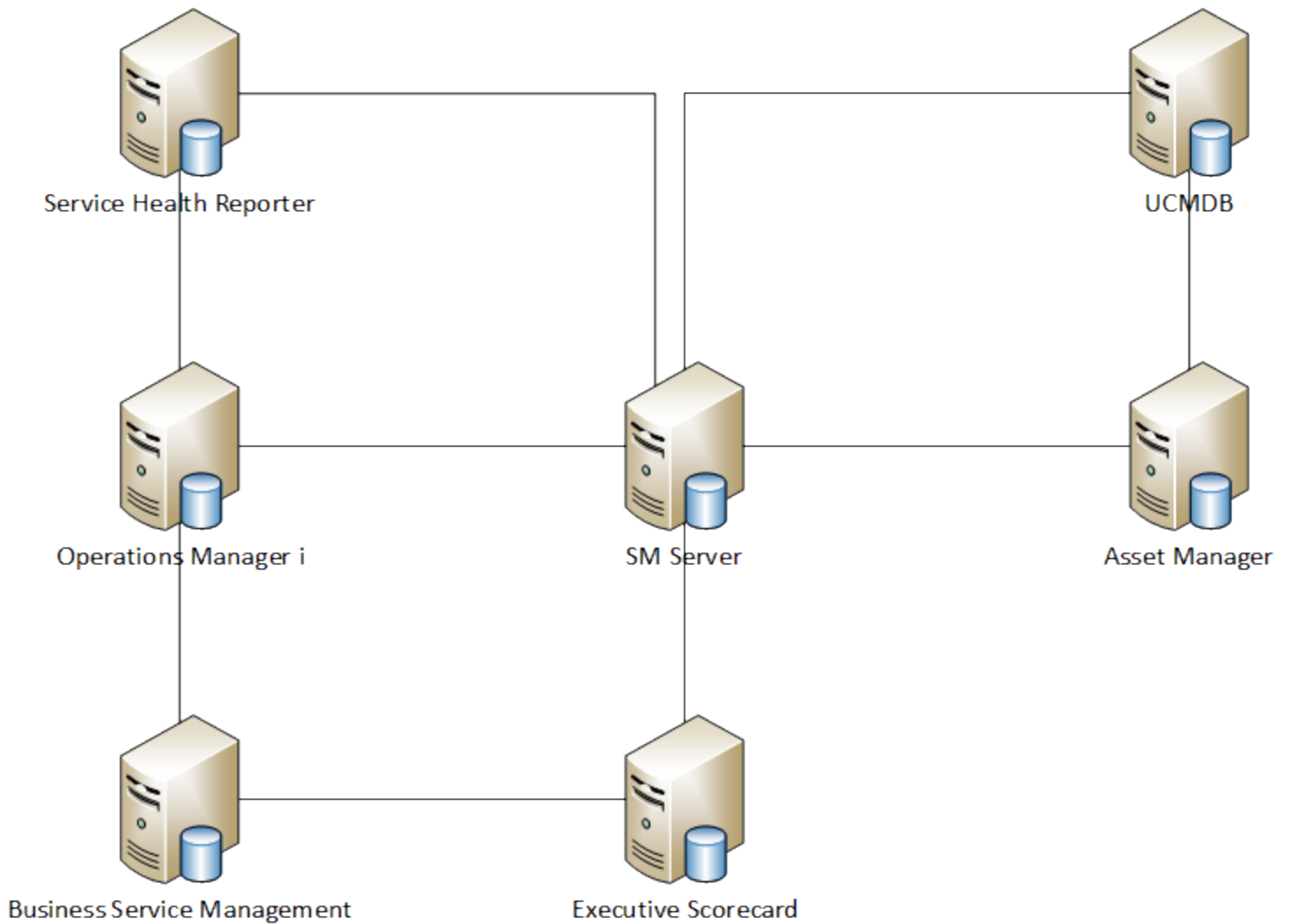
The ITSM Enterprise Suite provides a complete service management solution. It is comprised of the following HP products:

- Service Manager Enterprise Suite
  - Service Manager 9.40, including HP SM Smart Analytics, Knowledge Management the Service Portal.
  - Universal Configuration Management Database 10.20
- HP Asset Manager Enterprise Suite
  - Asset Manager 9.50
- HP IT Business Analytics (Formerly IT Executive ScoreCard)
  - IT Business Analytics 9.50
- HP Operations Bridge Suite Premium Edition
  - Operations Manager i 10.01
  - Service Health Reporter 9.40
  - Business Service Management 9.25
  - Also includes:
    - Operations Agent 11.14
    - SiteScope 11.30

## System topology

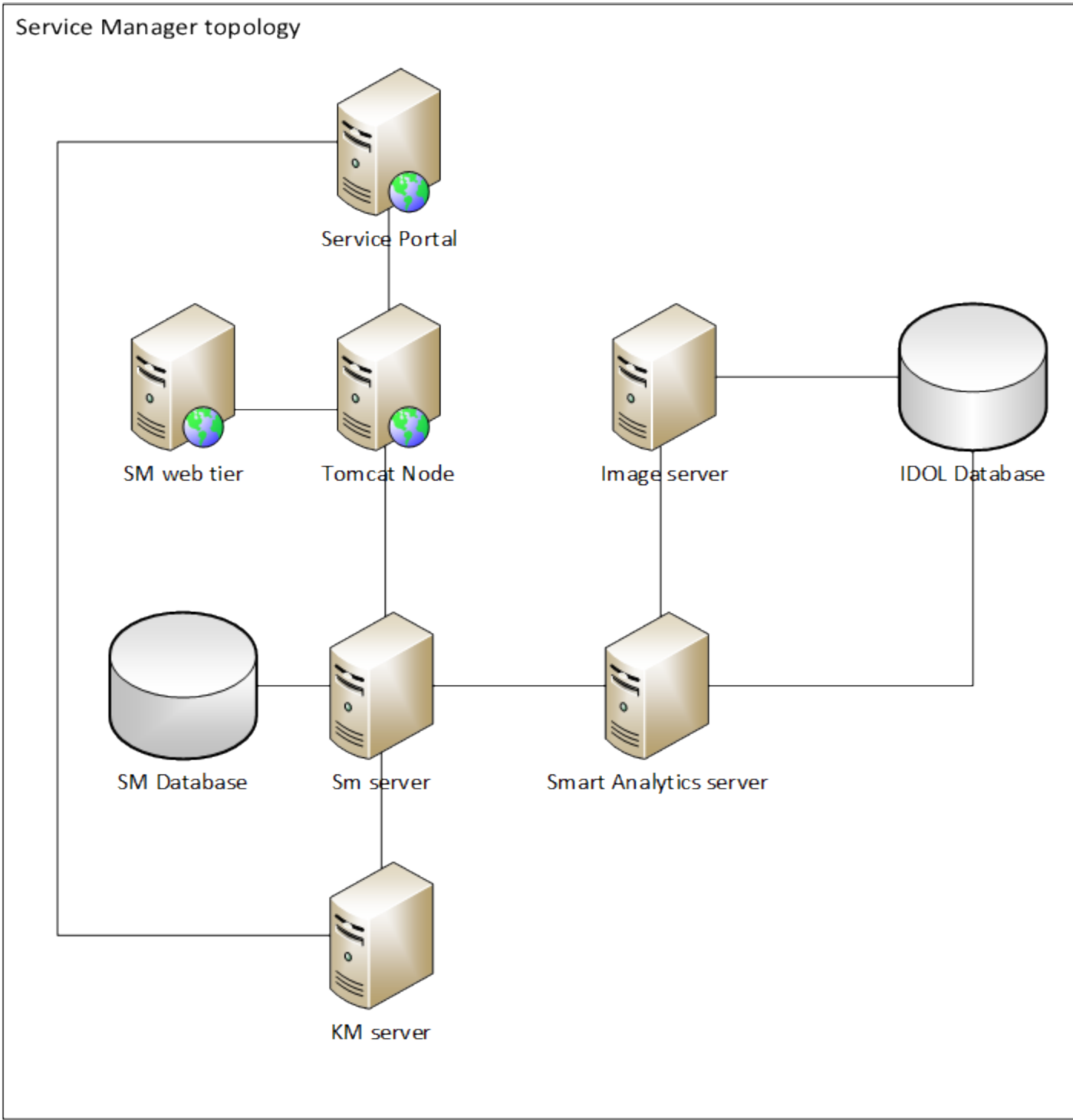
For simplicity, in defining the system topology, we require that each of the products within the ITSM Enterprise Suite requires its own server. By doing this, we simplify the deployment model, presenting as clear and concise a system topology as possible, and limiting our exposure to hardware failure. In our deployment model, we used virtual machines (VMs) due to their flexibility and configurability. However, it is expected that your organization will adapt the model provided in this document for your own needs.

### **An overview of the system topology:**

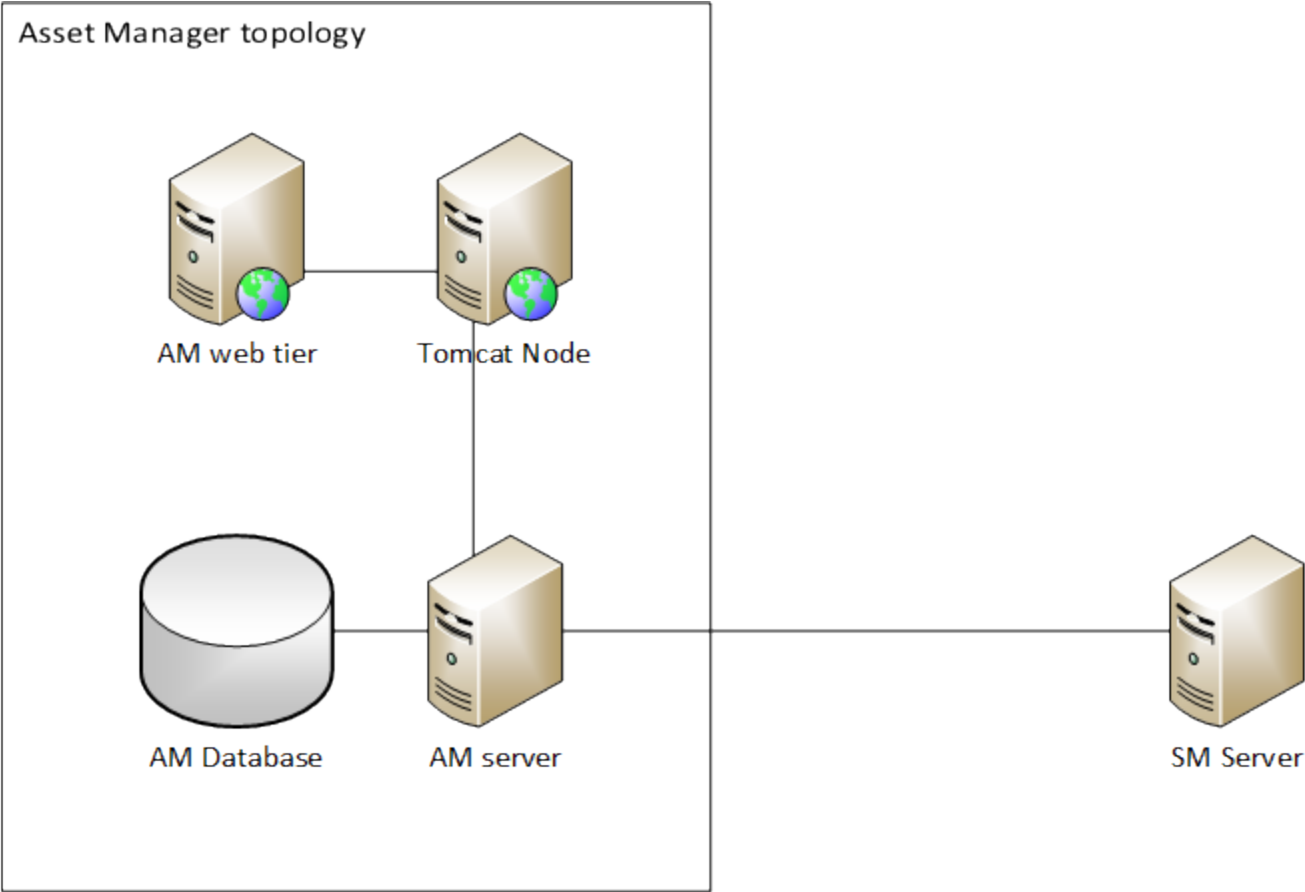


As indicated in the diagram, each component of the ITSM Enterprise Suite may itself contain other components. For example, Service Manager alone includes the Service Manager server, the database, the Service Manager web tier (including Apache and tomcat nodes), the Service Portal, the knowledge management server, and Smart Analytics, which itself may include the IDOL server and an image server. In fact, every node in the preceding diagram, contains at least its own application server and separate database. These more detailed topologies are shown in the following diagrams.

### **Service Manager**

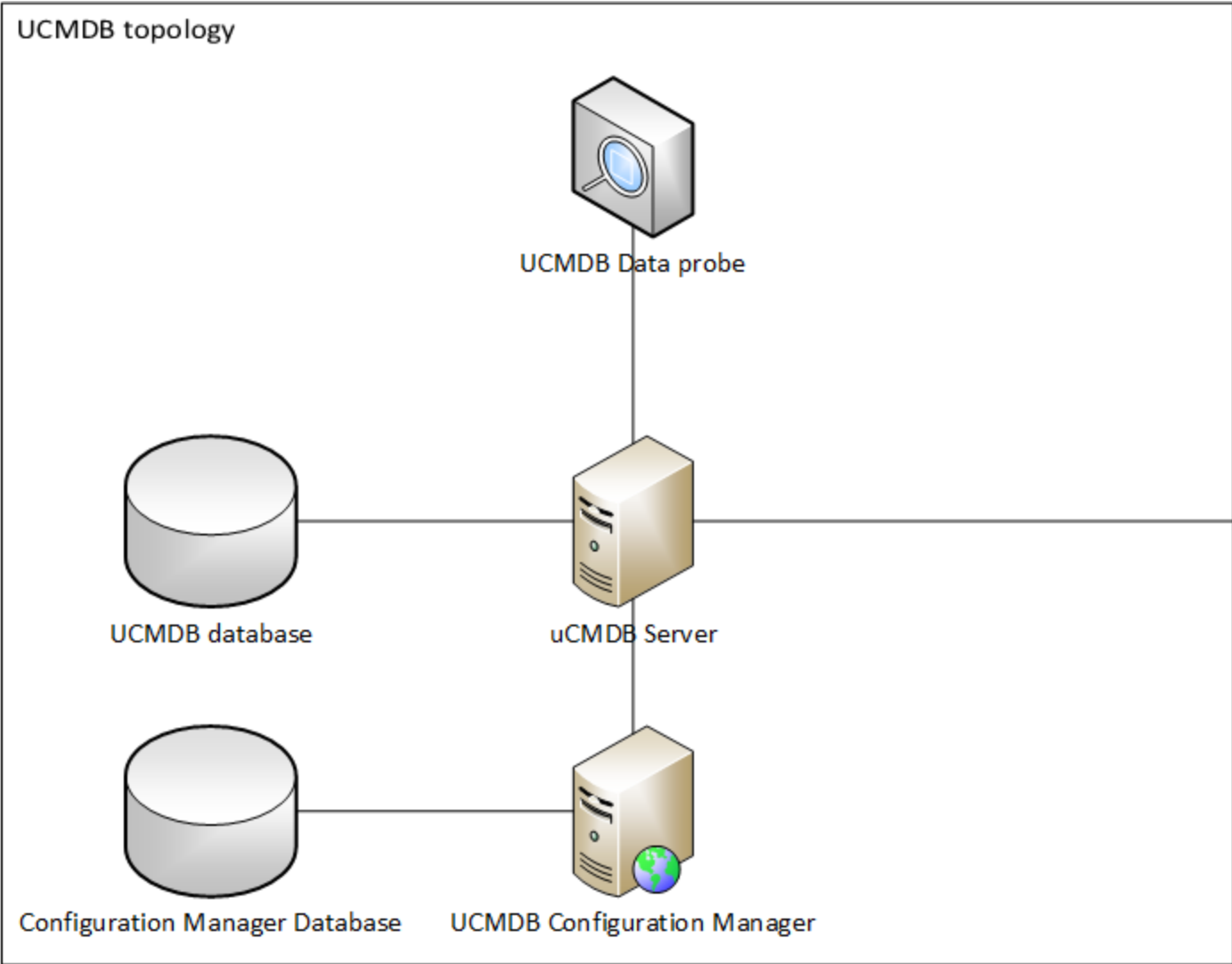


**Asset Manager**

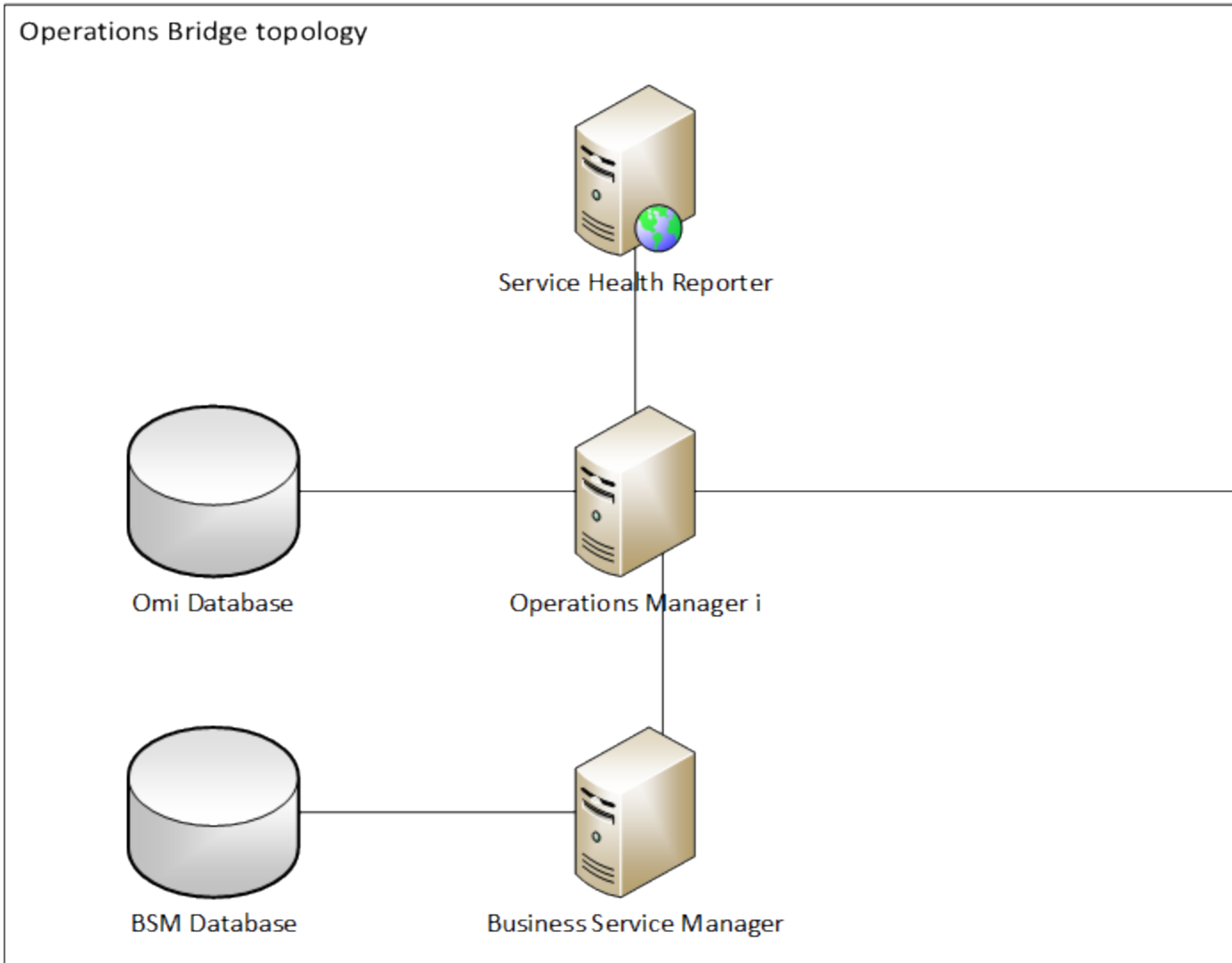


**Universal Configuration Management Database**





**Operations Bridge topology**



## Sizing requirements

The following table indicates basic sizing requirements for each product. For simplicity, we standardize core speed to the greatest common denominator in this case, 2.4 GHz. In general, the higher the CPU speed the better.

<b>Product</b>	<b>Component</b>	<b>Minimum Cores (@ 2.4 GHz)</b>	<b>RAM</b>	<b>Hard Disk</b>	<b>Operating System/Manufacturer (64 bit)</b>	<b>Notes</b>
Service Manager	Server	8	48 GB	120GB	Windows Server 2012 Red Hat Linux 6.5	
	RDBMS	6	16 GB	4-6 x148 GB RAID	MS SQL Server/Oracle	
	Smart Analytics	4	16 GB	100 GB	Windows Server 2012 Red Hat Linux 6.5	
	Web tier	8	16 GB	70 GB	Tomcat 7.0	
	Knowledge Management	4	8 GB	120 GB		
	Service Portal	2	3 GB	32 GB	Flash	Java Heap Size >= 1024 MB

Product	Component	Minimum Cores (@ 2.4 GHz)	RAM	Hard Disk	Operating System/Manufacturer (64 bit)	Notes
Universal Configuration Management Database	Universal CMDB	8	16 GB		Windows Server 2012 Red Hat Linux 6.5	<ul style="list-style-type: none"> <li>■ The virtual memory for Windows should be at least 1.5 times the size of the physical memory.</li> <li>■ The Linux swap file size should be equal in size to the physical memory.</li> </ul>
	Universal CMDB Configuration Manager	8	16 GB		Windows Server 2012 Red Hat Linux 6.5	
Asset Manager	Server & Web tier	2	16 GB	4-6 x148 GB RAID	Windows Server 2012 Red Hat Linux 6.5	
IT Business Analytics	SAP BusinessObjects Enterprise Server	8	16 GB	80 GB	Windows Server 2012 Red Hat Linux 6.5	

Product	Component	Minimum Cores (@ 2.4 GHz)	RAM	Hard Disk	Operating System/Manufacturer (64 bit)	Notes
	Data Warehouse Server	8	16 GB	120 GB	Windows Server 2012 Red Hat Linux 6.5	
	Executive Scorecard Server	8	16 GB	80 GB	Windows Server 2012 Red Hat Linux 6.5	
	SQL Server	24	48 GB	1 TB	MS SQL Server 2012	
Operations Bridge	Operations Manager i	4	16 GB	250 GB	Windows Server 2012 Red Hat Linux 6.5	
	Service Health Reporter (server)	16	32 GB	500 Mb	Windows Server 2012 Red Hat Linux 6.5	
	Service Health Reporter (Sybase IQ)	16	32 GB	4.5 TB	Windows Server 2008 Red Hat Linux 6.5	
	Service Health Reporter (Collectors)	4	8 GB	300 GB		Maximum 10,000 nodes.
	Business Service Management	8	24 GB	250 GB	Windows Server 2012 Red Hat Linux 6.5	

## Suggested administrator resources

## Compability matrix

Delete this text and replace it with your own content.

## Installation guidance

In the following sections, we describe the simplest and most expedient method to install the Hewlett Packard Enterprise ITSM Enterprise Suite. The content related to specific products is generally sourced from the documentation for those products. However, for the purposes of this comprehensive suite, we have made a number of arbitrary design decisions.

As a result, we leverage Deployment Manager wherever possible. IN cases where Deployment Manager is not used, we present only the single most common option for installing a specific product. For example, while the IT Business Analytics installation documentation describes two, three, and four server options to install IT Business Analytics and it's associated components, we present only the recommended four server production environment installation instructions. Similar design choices have been made for all other products. At the beginning of each major section, a few paragraphs are devoted to explaining which choices were made for the subsequent installation method.

## Using Deployment Manager

You can use HP Deployment Manager to facilitate the installation and integration of Service Manager, Asset Manager, UCMDB and the other components of the ITSM Enterprise Suite. Deployment manager is a software package that can download, install, configure, and integrate several Hewlett Packard software products. It was primarily designed to facilitate installation of Service Manager (SM) and its core integrations. In the ITSM Enterprise Suite, we use it to install Service Manager (including Knowledge Management, Smart Analytics, and the Service Portal), Asset Manager (AM), Universal Configuration Management Database (uCMDB) and configure the integrations between SM and uCMDB, Knowledge Management, the Service Portal, and Smart Analytics. By leveraging Deployment Manager, we can quickly and easily create and install the base ITSM system. After which, we will install and deploy the other components of the suite and integrate those components into the overall system architecture.

**Note:** Deployment Manager cannot handle the integration between Service Manager and Asset Manager at this time.

Deployment Manager includes a number of embedded scripts that will download and install the appropriate packages. You can create an environment for the ITSM Enterprise Suite consisting of multiple servers. After creating the environment, you can select various packages and components, assign those packages to the various servers, specify the various integrations, and then simply execute the script. Deployment Manager will then automatically download all necessary packages, install all

software to the various servers, configure and deploy any appropriate web tiers, and integrate the various software packages.

For example, suppose you wanted to install Service Manager and uCMDB together on two separate servers. For the Service Manager installation, you want to create a database, install knowledge management, and install the service portal. For the uCMDB installation, you need to install the database, install uCMDB, including the uCMDB data probe and the uCMDB Configuration Manager. To do this in Deployment Manager, you create a new environment, add the two servers to the environment. Then, select the “Install Service Manager” wizard, specify the appropriate server in the script, select the “Install uCMDB” wizard, and then specify the second server in the script. When you execute the scripts, deployment manager will automatically download the packages, install the base Service Manager and uCMDB on to the appropriate servers, including the databases, any Apache server and Tomcat application servers, and deploy any appropriate web applications. After the installation and deployment of all components are complete, Deployment Manager will consider both the SM and uCMDB server side integrations. Additionally, if you later on decided to add Asset Manager into your environment, you would simply add another server to your pre-existing Deployment Manager environment, and execute the “Install Asset Manager” script.

For the purposes of this document, we will not discuss how to install or configure Deployment Manager. To do that, refer to the Deployment Manager documentation.

## Platform limitations

The use of Deployment Manager imposes certain limitations on the deployment of the ITSM Enterprise Suite. These are detailed in the following list:

- Deployment Manager only contains scripts for Microsoft Windows and Linux platforms. Therefore, if your organization is using HP UX or AIX you must manually install, configure, and integrate all products deployed by Deployment Manager. To do this, refer to the installation and configuration documentation for each individual product.
- If you plan on using multi-tenancy in the Universal Configuration Management Database, you cannot use Deployment Manager. Therefore, you must use the installation and configuration documentation for the Universal CMDB to do this.

## Create a Deployment Manager environment

To create a new environment, follow these steps:

1. On the **Environments** tab, click the **+** icon on the left-hand pane to display the Environment Details dialog box.
2. Enter the name of your new environment. The name of the environment should be descriptive (such as "San Diego Development," "San Diego Production," or "San Diego UAT").
3. Select or clear the **Visible to All** option to determine whether the environment is visible to you only or to all ITSM Deployment Manager users, and then click **Save**.
4. Click **Add Server** button to define the servers within the environment.
5. Enter an arbitrary name that will identify the server easily, such as "SM Web Tier 1" or "SM RTE Load Balancer."
6. Enter the IP address of the server. Currently, ITSM Deployment Manager does not support hostnames or Fully Qualified Domain Names.
7. Enter the username and password of the server so that ITSM Deployment Manager can successfully open a PowerShell session, and then click **Save**.

**Note:** You may not need to add username and password details if ITSM Deployment Manager and the target servers are located in the same Windows domain.

The servers are now recognized by ITSM Deployment Manager. A green label with an "Online" status appears for each server that you have defined. If any servers display a red label and an "Offline" status, check that the server is powered on and that the specified IP address is correct. If any errors occur during the process of adding a server, the errors are displayed in the server's detailed information box

## Install Service Manager

To install Service Manager, follow these steps:

1. Log in to your instance of Deployment Manager, and then navigate to **Environments**.
2. Select your environment.
3. Click the **HP Service Manager - Installation** wizard.
4. Specify the following values as appropriate for your environment and licenses. The values specified



are examples based on the default licensing for the ITSM Enterprise Suite.

- SM version you want to install: SM 9.40
- Total number of users that will run concurrently on Service Manager Web? **100**
- Total number of users that will run concurrently on Service Manager Mobility? **0**
- Total number of users that will run concurrently on Service Request Catalog? **100**
- Environment Mode: **Deploy on a single server**

**Note:** For larger environments, you may wish to deploy select **Deploy on multiple servers**. You will have the opportunity to install the Service Manager components in various locations on the several screens.

5. Click **Next**.
6. Select which server you want to host the Service Manager server.
7. Specify the drive on which you want to install the Service Manager server.
8. Click **Next**.
9. Select which server you want to host the Service Manager web tier.
10. Specify the drive on which you want to install the Service Manager web tier.
11. Click **Next**.
12. Select the database you will use ( **SQL Server** or **Oracle**).
13. Select which server you want to host the Service Manager database.
14. **Optional:** Specify an alternative name for the Windows Data Source Name (ODBC).
15. **Optional:** Specify an alternative name for the database.
16. Specify a user name for the database.
17. Specify a password for the database.
18. Click **Next**.

19. **Optional:** Click to install the Service Manager Help Center.

20. Click **Install Later**.

To execute this package immediately and install Service Manager server, web tier, databases, and the Service Portal, follow these steps:

1. Navigate to the **Packages** tab, select your environment from the list, and then click on the **Install Service Manager** package.
2. Review the individual tasks.
3. When you are ready to install Service Manager, click **Execute Package**.

## Install and integrate Knowledge Management

Deployment Manager automatically install and integrate the Knowledge Management component to Service Manager and to the Service Portal. To install Knowledge Management, follow these steps:

1. Log in to your instance of Deployment Manager, and then navigate to **Environments**.
2. Select your environment and then click on **More Wizards**.
3. Click **HP Service Manager - Knowledge Management**.
4. Specify the following values for the Knowledge Management server:
  - Which server should the master type of the SM Knowledge Management be installed on? **<any server>**
  - KM version you want to install? **SM 9.40**
5. Click **Next**.
6. Select the SM Load balancer server.
7. Add a check to the other Service Manager servers.
8. Specify the User Name and Password.

**Note:** If you are not using the default user Name and Password of Falcon/, you may need to

specify an appropriate user name and password in Service Manager first.

#### 9. Click **Install Later**

To execute this package immediately and install Knowledge Management, follow these steps:

1. Navigate to the **Packages** tab, select your environment from the list, and then click on the **Install HP Knowledge Management on...** package.
2. Review the individual tasks.
3. When you are ready to install Knowledge Management, click **Execute Package**.

## Integrate Knowledge Management and the Service Portal

To integrate Knowledge Management with the previously installed Service Portal (which was installed when you installed Service Manager), follow these steps:

1. Navigate to **Packages**.
2. Click the **+** icon to add a new package.
3. Specify a name for the package, select the environment on which you installed Service Manager and Knowledge Management, and then click save.
4. Click **Execute Package**.

## Install and integrate Smart Analytics

To install Service Manager Smart Analytics, follow these steps:

1. Log in to your instance of Deployment Manager, and then navigate to **Environments**.
2. Select your environment and then click on **More Wizards**.
3. Click **HP SM Smart Analytics Installation**.
4. Specify the following values as appropriate for your environment:

- Select version to install: **IDOL 10.7 (SM 940)**
  - Select the installation type: **Standalone**.
5. Check all boxes under, select the components you wish to install:

**IDOL Server**

**Image Server.**

6. Click **Next**.
7. Specify which server in your environment will host the IDOL server.
8. Specify which server in your environment will host the Image server.
9. Specify the server on which you installed Service Manager.
10. Click **Install Later**.

To execute this package immediately and install SM Smart Analytics, follow these steps:

1. Navigate to the **Packages** tab, select your environment from the list, and then click on the **Install SM Smart Analytics on...** package.
2. Review the individual tasks.
3. When you are ready to install Knowledge Management, click **Execute Package**.

## Install and connect UCMDB

**Note:**

- Deployment Manager does not fully integrate Service Manager and the Universal Configuration Management Database. Push and population operations must still be performed manually. For information on how to do this, see the ["Using Service Manager and UCMDB" on page 278](#) and ["Integrate UCMDB and Service Manager using the enhanced adapter" on page 79](#).
- If you plan to use multi-tenancy for the Universal Configuration Management Database, you must configure multi-tenancy at the time of installation. In this case, you cannot use

Deployment Manager. Therefore, you should use the installation procedures described in the Universal Configuration Management Database Installation Guide.

To install the Universal Configuration Management Database, follow these steps:

1. Log in to your instance of Deployment Manager, and then navigate to **Environments**.
2. Select your environment and then click on **More Wizards**.
3. Click **HP UCMDB - Installation and Integration Wizard**.
4. Specify the following values for the Knowledge Management server:
  - Which server should the master type of the SM Knowledge Management be installed on? **<any server>**
  - Select the database type: **MSSQL** or **Oracle**
  - Choose an existing database server. **<any server>**
  - Database User Name: **<any name>**
  - Database Password: **<any password>**
  - Select SM server you want to integrate: **Select your SM server**
  - Specify the User Name and Password.

**Note:** If you are not using the default user Name and Password of Falcon/, you may need to specify an appropriate user name and password in Service Manager first.

- Select UCMDB Version: **[10.20]**
5. Click **Install Later**.

To execute this package immediately and install Universal Configuration Management Database, follow these steps:

1. Navigate to the **Packages** tab, select your environment from the list, and then click on the **Install UCMDB and Integrate with SM on...** package.
2. Review the individual tasks.

3. When you are ready to install Universal Configuration Management Database, click **Execute Package**.

## Install Asset Manager

**Note:** Deployment Manager can install Asset Manager, but cannot automate any integrations involving Asset Manager. Therefore, to integrate Asset Manager to Service Manager and Universal Configuration Management Database, you must do so manually. Refer to the following sections for more details.

To install the Asset Manager, follow these steps:

1. Log in to your instance of Deployment Manager, and then navigate to **Environments**.
2. Click **HP Asset Manager Installation Wizard**.
3. Specify the following values for the Asset Manager server:
  - AM version you want to install: **AM 9.50**
  - Select which server you want to install Asset Manager: **<any server>**
  - Select the database type: **MSSQL** or **Oracle**
  - Choose an existing database server. **<any server>**
  - Database User Name: **<any name>**
  - Database Password: **<any password>**
  - Choose a language: **Any language**
4. Click **Install Later**.

To execute this package immediately and install Asset Manager, follow these steps:

1. Navigate to the **Packages** tab, select your environment from the list, and then click on the Asset Manager package.
2. Review the individual tasks.
3. When you are ready to install Asset Manager, click **Execute Package**.

## Installing Operations Manager i

**Note:** In general, you should follow the product documentation for each product to install and configure the product, and then read and implement the associated integration guides.

To install Operations Manager i, follow the steps in the *HP Operations Manager i10.01 Installation and Upgrade Guide*. The *HP Operations Manager i10.01 Installation and Upgrade Guide* is an interactive installation document, which allows you to select various configuration items. For the HP ITSM Enterprise Suite example use case, which includes 250 to 900 nodes and is a smaller deployment, we use the following options for the installation:

- **Enterprise Deployment**
- **Install and configure OMi**
- **Single Server Environment**
- **PostgreSQL (embedded)**
- **Windows or Linux**

You may additionally choose to view instructions for additional options, such as Load balancing, hardening, and so on. When you click **View**, you will see a set of instructions for you to install Operations Manager i for your specific configuration.

**Note:** Verify the hardware and software requirements according to the documentation prior to beginning the installation.

## Install Service Health Reporter

### Installation Prerequisites

These prerequisites apply to the system where you want to install HP Service Health Reporter and also the remote systems where you want to install the SHR data collector.

#### Hardware Requirements

For a list of hardware requirements, see the *HP Service Health Reporter Support Matrix*.

### Disk Space Requirements

Ensure that you have required space as follows in /opt and /tmp directories:

- To download and merge the SHR media files, allocate at least 30 GB in the /tmp directory on each system.
- To install SHR components, allocate at least 20 GB in the /opt directory on each system.
- To download and merge the SHR remote collector media files, allocate at least 15 GB in the /tmp directory on each system.
- To install SHR remote collector, allocate at least 10 GB in the /opt directory on each system.
  - Do not start the installation directly from the mount point location.
  - Do not download and merge the TAR files directly from the mount point location.
- If additional external storage space is required to be added, ensure that no other applications are installed in the /opt directory.

### Software Requirements

For the complete list of software requirements, see the *Requirements* section in the *HP Service Health Reporter Support Matrix*.

### Operating System Requirements

For the complete list of operating system requirements, see the *Requirements* section in the *HP Service Health Reporter Support Matrix*.

Before you install SHR, you must update your operating system, apply all patches, establish network connectivity, and disable the anti-virus software.

Ensure that the following libraries are available on each system where you plan to install SHR components.

Red Hat Enterprise Linux 6.x	Red Hat Enterprise Linux 5.5
The list indicates the minimum required versions of required libraries. You can install a higher version of each library, if available.	



<ul style="list-style-type: none"><li>• compat-libstdc++-33-3.2.3-69.i686</li><li>• compat-libstdc++-33-3.2.3-69.x86_64</li><li>• libXext-1.1-3.i686</li><li>• libXext-1.1-3.x86_64</li><li>• libXext-devel-1.1-3.i686</li><li>• libXext-devel-1.1-3.x86_64</li><li>• libstdc++-4.4.4-13.x86_64</li><li>• libstdc++-4.4.4-13.i686</li><li>• libXtst-1.0.99.2-3.i686</li><li>• libXtst-1.0.99.2-3.x86_64</li><li>• libXau-1.0.5-1.i686</li><li>• libXau-1.0.5-1.x86_64</li><li>• libXdmp-1.0.3-1.i686</li><li>• libxcb-1.5-1.x86_64</li><li>• libxcb-1.5-1.i686</li><li>• libXrender-0.9.5-1.i686</li><li>• libXrender-0.9.5-1.x86_64</li><li>• glibc-2.12-1.7.x86_64</li><li>• glibc-2.12-1.7.i686</li><li>• libgcc-4.4.1-13.i686</li><li>• libgcc-4.4.4-13.x86_64</li><li>• libX11-1.3-2.i686</li><li>• libX11-1.3-2.x86_64</li><li>• libXi-1.3-3.x86_64</li></ul>	<ul style="list-style-type: none"><li>• compat-libstdc++-33-3.2.3-61.x86_64</li><li>• compat-libstdc++-33-3.2.3-61.i386</li><li>• libXext-1.0.1-2.1.x86_64</li><li>• libXext-1.0.1-2.1.i386</li><li>• libXext-devel-1.0.1-2.1.x86_64</li><li>• libXext-devel-1.0.1-2.1.i386</li><li>• libstdc++-4.1.2-48.x86_64</li><li>• libstdc++-4.1.2-48.i386</li><li>• libXtst-1.0.1-3.1.x86_64</li><li>• libXtst-1.0.1-3.1.i386</li><li>• libXau-1.0.1-3.1.x86_64</li><li>• libXau-1.0.1-3.1.i386</li><li>• libXdmp-1.0.1-2.1.i386</li><li>• libXrender-0.9.1-3.1.x86_64</li><li>• libXrender-0.9.1-3.1.i386</li><li>• glibc-2.5-47.i686</li><li>• glibc-2.5-47.x86_64</li><li>• libgcc-4.1.2-48.i386</li><li>• libgcc-4.1.2-48.x86_64</li><li>• libX11-1.0.3-11.x86_64</li><li>• libX11-1.0.3-11.i386</li><li>• libXi-1.0.1-3.1.x86_64</li><li>• libXi-1.0.1-3.1.i386</li><li>• alsa-lib-1.0.17-1.x86_64</li></ul>
---	---

<ul style="list-style-type: none"><li>• libXi-1.3-3.i686</li><li>• alsa-lib-1.0.22-3.i686</li><li>• alsa-lib-1.0.22-3.x86_64</li><li>• nss-softokn-freebl-3.12.7-1.1.i686</li><li>• ncurses-libs-5.7-3.20090208.i686</li><li>• ncurses-libs-5.7-3.20090208.x86_64</li><li>• redhat-lsb.i686</li><li>• redhat-lsb.x86_64</li></ul>	<ul style="list-style-type: none"><li>• alsa-lib-1.0.17-1.i386</li><li>• glibc-2.5-47.i686</li><li>• glibc-2.5-47.x86_64</li><li>• redhat-lsb.i686</li><li>• redhat-lsb.x86_64</li></ul>
---	--

Ensure that the swap space is twice the size of the RAM. To allocate sufficient swap space, perform the following steps:

1. Log on to the system as root.

The root user must be the owner of the /opt and /var directories.

2. To set up the swap space by creating a new swap file, run the following commands :

- `dd if=/dev/zero of=<swapfile_full_path> bs=1M count=<swap_size_in_MB>`
- `mkswap <swapfile_full_path>`
- `swapon <swapfile_full_path>`

In this instance, *<swapfile\_full\_path>* is the name of the new swap file (including full path to the file) and *<swap\_size\_in\_MB>* is the space (in MB) that you want to allocate.

For example, to allocate swap space by creating a new /extraswap file:

```
dd if=/dev/zero of=/extraswap bs=1M count=16384
```

```
mkswap /extraswap
```

```
swapon /extraswap
```

3. For the change to remain in effect even after a system restart, add the following line in the

*/etc/fstab* file:

```
<swapfile_full_path>swap swap defaults 0 0
```

In this instance, *<swapfile\_full\_path>* is the name of the newly created swap file in the previous step.

For example:

```
/extraswap swap swap defaults 0 0
```

- Restart the system.

### Port Availability

SHR uses a number of default ports for different services. Ensure that the following ports are free before installing SHR components.

Service	Port Number	Protocol	Inbound	Outbound	Description
HP PMDB Platform DB Logger	21408	TCP	Yes	Yes	The HP PMDB Platform DB Logger service persists logs in the database through this port.
HP PMDB Platform Collection	21409	TCP	Yes	Yes	The JMX management port for the Collection service. The IM service monitors collection using this interface.
HP PMDB Platform IM	21410	TCP	Yes	No	The JMX management port for the IM service.
HP PMDB Platform Timer	No port	NA	NA	NA	The Timer service for SHR.
HP PMDB Platform Administrator	21411	TCP	Yes	No	The SHR web application server port, which hosts the Administration web application. The Report cross-launch functionality depends on this service.
HP Software Communication Broker	383	TCP	Yes	Yes	SHR uses this port to communicate with collectors installed on remote servers.

<b>Service</b>	<b>Port Number</b>	<b>Protocol</b>	<b>Inbound</b>	<b>Outbound</b>	<b>Description</b>
Administration Console web server	21416	TCP	Yes	Yes	The JMX management port for the SHR administration web server.
HP PMDB Platform Collection	21418	HTTP	Yes	No	The connection port to the HTTP server for the SiteScope generic data integration.
HP PMDB Platform Collection	21419	HTTPS	Yes	No	The connection port to the HTTP server for the SiteScope generic data integration.
HP PMDB Platform Collection	8080	HTTP	No	Yes	The connection port to collect data from the SiteScope Data Acquisition API.
HP PMDB Platform Sybase Service	21424	TCP	Yes	Yes	Port for the Sybase IQ server.
Sybase IQ Agent 15.4	21423	TCP	Yes	No	Port for the Sybase IQ Agent.
HP-SHR-Postgre - PostgreSQL Server 9.0	21425	TCP	Yes	Yes	Port for the PostgreSQL service.
Apache Tomcat	8080	TCP	Yes	No	This is the SAP BusinessObjects Application Service port. The SAP BusinessObjects Central Management Console and the SAP BusinessObjects InfoView web applications are hosted on this port.
SAP BOBJ Central Management Server	6400	TCP	Yes	Yes	This is the port for the SAP BusinessObjects Central Management Server, which is mainly used for SAP BusinessObjects authentication purposes.
Server Intelligence Agent (HOML01GEATON)	6410	TCP	Yes	Yes	Port for the SAP BusinessObjects Server Intelligence Agent, which manages all SAP BusinessObjects-related tasks.

Service	Port Number	Protocol	Inbound	Outbound	Description
BOE120SQLAW	2638	TCP	Yes	Yes	Port for the SAP BusinessObjects repository database.
RTSM	21212	TCP	No	Yes	This is the port that is configured in the Administration Console for the RTSM data source. Using this port, SHR connects to RTSM.
HPOM	Any	TCP	No	Yes	This is the port that is configured in the Administration Console for the HPOM database. SHR uses this port to connect to the HPOM database.
HP Operations Agent	383	TCP	No	Yes	SHR uses this port to connect to the HP Operations agent.
HP BSM Profile database	Any	TCP	No	Yes	This is the port that is configured in the Administration Console for the Profile database.  SHR uses this port to connect to the Profile database and the OMi database.

### Web Browser Requirements

To view the SHR Administration Console in Internet Explorer or Mozilla Firefox, you must enable the ActiveX and the JavaScript controls. Follow the Help menu of the web browser for assistance with enabling them.

For a list of supported Web Browsers, see *"Web Browsers and Plug-ins"* in the *HP Service Health Reporter Support Matrix*.

## Preinstallation Tasks and Checklist

After ensuring that the installation prerequisites are fulfilled, you must perform a series of tasks to prepare the server for the SHR installation.

### Task 1: Disable Anti-Virus

Anti-virus applications can hinder the installation of SHR. Temporarily disable any anti-virus software that might be running.

Re-enable the anti-virus software after the installation is complete.

## Task 2: Configure Firewall

If you use firewall software, ensure that the firewall allows traffic through the required ports (see *Installation Prerequisites > Port Availability*) on the SHR system.

To disable the firewall, perform the following steps:

1. Log on as root and run the following commands:

**Note:** The root user must be the owner of the /opt and /var directories.

```
chkconfig iptables off
```

```
chkconfig ip6tables off
```

```
/etc/init.d/iptables stop
```

```
/etc/init.d/ip6tables stop
```

## Task 3: Prepare the Linux System

On the Linux system, you must perform a set of additional steps.

### Task 3.1: Disable SELinux

To disable SELinux, in the /etc/sysconfig/selinux file, set the parameter SELINUX = disabled.

### Task 3.2: Configure the Kernel Parameters (only if you use Red Hat Enterprise Linux 6.x)

To configure the Kernel parameters, follow these steps:

1. Open the file /etc/sysctl.conf file.
2. Set the values of the parameters as given below:

**Note:** If higher values are specified for these parameters already, do not make any modifications.

- kernel.msgmnb = 65536
- kernel.msgmax = 65536
- kernel.shmmax = 68719476736
- kernel.shmall = 4294967296

- kernel.sem = 250 1024000 250 4096
- vm.max\_map\_count = 1000000

### *Task 3.3: Configure the Hostname*

Log on to the SHR system, and configure the hostname in the `/etc/hosts` file.

If you configure a hostname, it should be added after these two lines as they appear by default.

```
127.0.0.1 localhost.localdomain localhost
```

```
192.168.0.1 server1.example.com server1
```

The naming convention for the hostname is: <IP address> <FQDN of SHR host system> <Short name of SHR host system>

### *Task 3.4: Configure the `limits.conf` File*

Open the `/etc/security/limits.conf` file and increase the number of open files by setting the following values:

```
* soft nofile 65535
```

```
* hard nofile 65535
```

### *Task 3.5: Configure the `90-nproc.conf` File (only if you use Red Hat Enterprise Linux 6.x)*

Open the `/etc/security/limits.d/90-nproc.conf` file and comment out the following line (by adding a `#` character in the beginning):

```
##*soft nproc 1024
```

Restart the Linux system for all the changes to take effect.

## **Task 4: Verify the Fully Qualified Domain Name (FQDN) of the System**

Before performing the SHR installation, you must verify that DNS lookup returns the accurate FQDN of the system. If the entry for the DNS lookup is different from the host name of the system, you may experience difficulties in logging on to the SHR Administration Console. This can occur because during the SAP BusinessObjects installation, the host name of the system is used for creating the servers/services and registering them.

To verify the FQDN of the host system, follow these steps:

1. Open the command line interface and type the following command to check the hostname of the system:

```
hostname -f
```

Note down the hostname of the system.

2. Type the following command to view the IP address of the system:

```
ifconfig
```

3. Type the following command to verify the FQDN of the displayed IP address:

```
nslookup<IP_address>
```

where, <IP address> is the IP address of the system.

Ensure that the name displayed after running the `nslookup` command matches the name displayed after running the `HOSTNAME` command. If the names do not match, you must change the hostname of the system.

#### **Task 5: Assemble the media**

On the HP software download web site, the SHR installation media for Linux is distributed as a collection of the following three files:

```
HPSHR_940_Lin64.part1
```

```
HPSHR_940_Lin64.part2
```

```
HPSHR_940_Lin64.part3
```

Before you start installing SHR, you must download all three files, and then combine them into a single `.tar` file.

#### **To create the SHR installation media, follow these steps:**

Download the SHR media files into a temporary directory on the system where you want to install SHR components.

1. To create a new directory for installing SHR, run the following command:

```
mkdir <directory name>
```

For example: `mkdir /tmp/HPSHR_9.4-parts`

2. To go to the directory that you created in the previous step, run the following command:

```
cd <temp location>
```



For example: `cd /tmp/HPSHR_9.4-parts`

3. Download the `.tar` file parts into the newly created temporary directory.
4. To merge the contents into a single `.tar` file, run the following command:

```
cat HPSHR_940_Lin64.part? > /tmp/HPSHR_9.4-parts/HPSHR940.tar
```

The SHR 9.40 media is now available as a single `.tar` file in the following location:

```
/tmp/HPSHR_9.4-parts/HPSHR940.tar
```

### Additional Considerations

- Always install SHR as root.

The root user must be the owner of the `/opt` and `/var` directories.

- Ensure that system time does not change during the course of the installation. Make sure the system does not automatically transition to the daylight saving time during installation.
- Do not install SHR from a network share. Installation of SHR over the network is not supported.

**Note:** The SHR installer does not support forced reinstallation. In the event of a unsuccessful installation, you must manually remove all the files that were placed by the installer and start the installation process again.

## Typical Installation: Install on a Single System

Install HP Service Health Reporter Server, Sybase IQ Server, and SAP BusinessObjects Server on a single system.

### Installing from the Command Line Console

1. Go to the media root.

Media root is the directory where the contents of the SHR media (the `.tar` file) are extracted.

2. At the command prompt, type the following command:

```
./HP-SHR_9.40_setup.bin -i console
```

3. Press **Enter** to start the installation. The Choose Locale section appears.

**Note:** At any point in time during installation, you can type `back` to go to the previous page and type `quit` to cancel the installation.

4. Choose the locale in which you want to install SHR, and press **Enter**. The installer shows the introductory information in the console.
5. Press **Enter**.
6. The installer shows the license agreement details. Type **Y** to accept the agreement, and then press **Enter**. The installer shows different installation options.

**Note:** Review the screen prompts carefully before pressing **Enter** each time. Pressing the Enter button continuously might take you through the next steps with the default selections.

7. Type **1** for **Typical HP Service Health Reporter Installation** to install SHR, Sybase IQ, and SAP BusinessObjects. Press **Enter**.

The installer performs necessary prerequisite checks and shows the result of the check in the console.

8. If the prerequisite check fails or shows warning messages, ensure that all the prerequisites are met and start the installation again.

If the prerequisite check displays any missing libraries, check the list of missing libraries from the location `/tmp/SHR-Missing-Patches.txt` and install them. Start the SHR installation again.

9. If the prerequisite check is successful, press **Enter**. The installer shows preinstallation summary in the console.
10. Press **Enter** to start the installation.
11. After successful installation, run the following command:

```
hostname -f
```

Check if the hostname is displayed.

If the installation fails, click **Rollback** and wait till the product gets rolled back. Run the rollback utility as follows:

1. In the command line console, go to the rollback utility path.

You will find the Rollback utility file in the location from where the installation setup files were extracted for installation.

2. Run the following command:

```
sh rollback-utility.sh
```

**Note:** During SHR installation in Linux, SAP BusinessObjects client tools are also installed but not supported on Linux. If SHR is installed on a Linux server, you must install the SAP BusinessObjects client tools on a Windows operating system for developing or customizing application content. For more information, see *Developing Content in Linux using CDE* section of *HP Service Health Reporter Content Development Guide*.

To perform the post-installation configuration tasks, see the *HP Service Health Reporter Configuration Guide*.

## Post-Installation Task for Sybase IQ

On a system with the Simplified Chinese or Japanese locale, manually delete the following files after installation:

- \$PMDB\_HOME/Sybase/IQ-15\_4/res/dblgzh\_iq12\_eucgb.res
- \$PMDB\_HOME/Sybase/IQ-15\_4/res/dblgzh\_iq12\_cp936.res
- \$PMDB\_HOME/Sybase/IQ-15\_4/res/dblgja\_iq12\_eucjis.res
- \$PMDB\_HOME/Sybase/IQ-15\_4/res/dblgja\_iq12\_sjis.res

## Validating SHR Installation

Perform the following to verify the success of installation on Linux operating system:

1. Log on as root.
2. Run the following command:

```
chkconfig --list
```

The command output lists the SHR services. Run the following commands for each of the services to ensure that they are running satisfactorily :

- `service HP_PMDB_Platform_Administrator status`
- `service HP_PMDB_Platform_Collection status`
- `service HP_PMDB_Platform_DB_Logger status`
- `service HP_PMDB_Platform_IM status`
- `service HP_PMDB_Platform_PostgreSQL status`
- `service HP_PMDB_Platform_Sybase status`
- `service HP_PMDB_Platform_IA status`
- `service TrendTimer status`

To check the status of the SAP BusinessObjects services, run the following commands at the command line console:

- a. `su - SHRBOADMIN`
- b. `cd /opt/HP/BSM/BO/bobje`
- c. `sh ccm.sh -display`

The command output shows the status of SAP BusinessObjects services. All services must be enabled and running.

**Note:** If you have installed SHR on RHEL 6.6, after configuring SHR, ensure that you perform *Manual Restart of Tomcat Services* steps in *HP Service Health Reporter Configuration Guide*.

## Installing the SHR Data Collector on a Remote System (Optional)

In the typical installation mode of SHR, the data collector is installed on the same system where SHR is installed. But, you can also install the data collector on a separate server. Also, you can install collectors on multiple systems as necessary.

**To install a collector on a remote system running on Windows, perform the following steps:**

1. All software requirements mentioned in the Prerequisites section must be met on the system where you want to install the data collector.
2. In the system where you have installed SHR, browse to the SHR install directory `%PMDB_HOME%`, and locate the following file:

HP-SHR-09.40-RemotePoller\_9.40\_setup.exe

You can also find this EXE file in the `packages\HP-SHR-09.40-RemoteCollector` folder on the SHR media.

3. Copy the file to the system where you want to install the collector.
4. Log on to the system where you want to install the collector as administrator.
5. Ensure that the remote system and the SHR system are in the same time zone.
6. Ensure that the system is registered in the Domain Name System (DNS).

Alternatively, ensure that:

- The hosts file on the SHR system includes a entry of the collector system.
- The hosts file on the collector system includes a entry of the SHR system.

The hosts file is located at `C:\Windows\System32\drivers\etc`

7. Browse to the folder where you copied the `HP-SHR-09.40-RemotePoller_9.40_setup.exe` file and run it.
8. The License Agreement page appears. Review the license agreement, select **I accept...**, and then click **Next**.
9. Review the folders in which the data collector would install. To change the installation folders, use the adjoining Browse buttons. Click **Next**.

**Note:** Do not enter spaces or special characters other than the - (hyphen) in the non-default folder name. The installation path must be less than 20 characters.

The installer performs checks for installation prerequisites and shows the result of the check on the Install Check page.

10. On the Product Requirements page, if the checks are successful, click **Next**.
11. The Pre-Installation Summary page appears. Review the summary, and click **Install**.
12. After the installation is complete, click **Done**.

**To install a collector on a remote system running on Linux, perform the following steps:**

1. All software requirements mentioned in Prerequisites must be met on the system where you want to install the data collector.
2. In the system where you have installed SHR, browse to the SHR install directory `$PMDB_HOME` and locate the following file:

```
HP-SHR-09.40-RemoteCollector.tar.gz
```

3. Transfer the file to the system where you want to install the collector.
4. Log on to the system where you want to install the collector as root. The root user must be the owner of the `/opt` and `/var` directories.
5. Ensure that the remote system and the SHR system are in the same time zone.
6. Ensure that the system is registered in the Domain Name System (DNS).

Alternatively, ensure that:

- The `hosts` file on the SHR system includes a entry of the collector system.
- The `hosts` file on the collector system includes a entry of the SHR system.

The `hosts` file is located at `/etc/hosts`

7. Extract the contents of the `HP-SHR-09.40-RemoteCollector.tar.gz` file into a local directory by running the following command:

```
tar -xvf HP-SHR-09.40-RemoteCollector.tar.gz
```

The contents of the `HP-SHR-09.40-RemoteCollector.tar` file are extracted from the archive.

**Installing from the Command Line Console**

1. Run the following command in the command line console.

```
./HP-SHR-RemotePoller_9.40_setup.bin -i console
```

2. Press **Enter** to start the installation.

**Tip:** At any point in time during installation, you can type back to go to the previous page and type quit to cancel the installation.

3. Choose the locale in which you want to install SHR, and press **Enter**.
4. The installer shows the introductory information in the console. Press **Enter**.
5. Review the license agreement details. Type **Y** to accept the agreement and press **Enter**.

The installer performs checks for installation prerequisites and shows the result of the check on the Install Check page.

6. The installer shows preinstallation summary in the console. Press **Enter** to start the installation.

**Note:** The collector is enabled to collect data from data sources only after you configure the collectors through SHR Administration Console.

## Next Steps

### have Task: Start the Sybase IQ Database

On the Linux system, run the following commands:

1. `cd /etc/init.d`
2. `service HP_PMDB_Platform_Sybase status`

If the command output shows that the HP\_PMDB\_Platform\_Administrator service is stopped, run the following command:

```
service HP_PMDB_Platform_Sybase start
```

### Task: Configure SHR for Multiple Profile Database Support

**Caution:** Perform this task only if you want to configure RTSM as the topology source for SHR.

You can skip this task and proceed to *Task 1: Launching the Administration Console* section in the *HP Service Health Reporter Configuration Guide* if you want to configure HPOM or VMware vCenter as the topology source.

SHR supports the configuration of and data collection from multiple Profile databases that are deployed in your HP BSM/OMi environment.

However, to ensure that SHR identifies and displays all the existing Profile databases in the Administration Console, follow these steps:

1. Log on to the HP BSM/OMi host system through remote access.

**Note:** If your HP BSM/OMi setup is distributed, you can access through the gateway server as well as the data processing server. HP recommends that you use the gateway server.

2. Browse to the %topaz\_home%\Conf folder.
3. Copy the following files from the %topaz\_home%\Conf folder of the HP BSM/OMi host system to the %PMDB\_HOME%\config folder on the SHR system:
  - encryption.properties
  - seed.properties

**Note:** If you are configuring the Management/Profile database under Oracle RAC, you also need to copy the file `bsm-tnsnames.ora` to the %PMDB\_HOME%\config folder on the SHR system.

After copying the files, you need to start the HP PMDB Platform Administrator service. Perform the following steps:

1. On the SHR system, click **Start > Run**. The Run dialog box appears.
2. In the Open field, type `services.msc`. The Services window appears.
3. On the right pane, right-click **HP\_PMDB\_Platform\_Administrator**, and then click **Restart**.
4. Close the Services window.

Type the following command at the command prompt:

```
service HP_PMDB_Platform_Administrator restart
```

After installing SHR, you must perform configuration steps to configure SHR to use data sources. For more information, see *HP Service Health Reporter Configuration Guide*.

**Caution:** Take a backup of the SHR database so that you can restore it later. If you fail to take a



data back up, you risk losing it permanently. For more information, see the "*Database Backup and Recovery*" section in the *HP Service Health Reporter Configuration Guide*.

## Change the Default SAP BusinessObjects Database Password

SHR is installed with a default SAP BusinessObjects database password. Perform this task to change the default SAP BusinessObjects database password.

After installing SHR, follow these steps to modify the default password of the database embedded with SAP BusinessObjects:

1. Go to <BusinessObjects installed drive>:\Program Files (x86)\Business Objects\SQLAnywhere12\bin
2. Open the dbisqlc application.
3. Type the credentials as follows:

User name = DBA.

Password = pmdb\_admin.

Server name = BOE120SQLAW\_<shrshorthostname>.

Database name = BOE120.

4. Run the following command in the sql window:

```
ALTER USER <shrshorthostname> IDENTIFIED BY <new password>;
```

```
ALTER USER DBA IDENTIFIED BY <new password>;
```

5. Repeat the same steps for database BOE120\_AUDIT.

After updating the password, follow these steps:

1. Go to CCM from **Start > All Programs > BusinessObjects XI 3.1 > BusinessObjects Enterprise > Central Configuration Manager**
2. Stop the Server Intelligence Agent (SIA); right-click and select **Properties**.

3. Go to the **Configuration** tab, you may see a error pop up click **OK** to proceed, and then click **Specify** of BOE120.
4. Click on the **Update Data Source Settings** .
5. Click **OK**.
6. Select **SQL Anywhere (ODBC)** and click **OK**.
7. The Select Datasource window appears. Select **Machine Data Source** tab.
8. Double-click DSN **BOE120** and provide the password that was changed and click **OK**.
9. Repeat from step 3 for BOE120\_AUDIT database by selecting the **BOE120\_AUDIT**.
10. Start the SIA.

To check if the password is modified, log in to the database using the new password.

SHR is installed with a default SAP BusinessObjects database password. Perform this task to change the default SAP BusinessObjects database password.

After installing SHR, follow these steps to modify the default password of the database embedded with SAP BusinessObjects:

1. Log in to the system as root.
2. Run the following command to switch to SHRBOADMIN user:

```
su -SHRBOADMIN
```

3. Go to /opt/HP/BSM/BO/bobje/SQLAW/Bin/.

4. Run the following command:

```
source /opt/HP/BSM/BO/bobje/setup/env.sh
```

5. Run the following command:

```
./dbisqlc
```

The credentials window appears. Click **Cancel**.

6. Select **Command**, and then click **Connect....** The Connect credentials window appears.

7. Type the following details:

- a. USER ID = DBA.
- b. Password = pmdb\_admin.
- c. Database Name = <shrshorthostname>BOE120.
- d. Server = <shrshorthostname>BOE120\_SHR.

8. Run the following command in the sql window:

```
ALTER USER <shrshorthostname> IDENTIFIED BY <new password>;
```

```
ALTER USER DBA IDENTIFIED BY <new password>;
```

9. To modify the default password for BOE120\_AUDIT, click **Command**, and then click **Connect....**

10. Type the following details:

- a. USER ID = DBA.
- b. Password = pmdb\_admin.
- c. Database Name = <shrshorthostname>BOE120\_AUDIT.
- d. Server = <shrshorthostname>BOE120\_AUDIT\_SHR.

11. Run the following command in the sql window:

```
ALTER USER <shrshorthostname> IDENTIFIED BY <new password>;
```

```
ALTER USER DBA IDENTIFIED BY <new password>;
```

After updating the password, follow these steps:

1. Go to /opt/HP/BSM/BO/bobje.
2. Run the following command:  

```
./cmsdbsetup.sh
```
3. Provide the SIA name as *PRD\_SHR* and click on Enter.
4. The prompt shows stop SIA. Type *yes*, and then click on Enter.

5. The prompt asks you to update. Type *update*, and then click on Enter.
6. Type *yes* and click Enter.
7. Type *SQL Anywhere*.
8. Type *2*, and then click on Enter.
9. Type DSN as <shrshorthostname>BOE120, and then click on Enter.
10. Type user as SHR, and then click on Enter.
11. Type the new password, which you changed recently, and then click on Enter.
12. Repeat the above steps with the DSN as <shrshorthostname>BOE120\_AUDIT DSN.
13. Run the following commands:

```
./stopservers
```

```
./startservers
```

To check if the password is modified, log in to the database using the new password.

## Troubleshooting

### **Symptom: Installation Failure caused by SAP BusinessObjects Error**

**Description:** While running the HP Software installer, the installation fails and the following error message is displayed:

*SAP BusinessObjects is installed on the system. Please uninstall it before installing HP SH Reporter.*

**Resolution:** If you have any component of SHR (such as SAP BusinessObjects or Sybase IQ) preinstalled or not cleanly uninstalled from a previous uninstall on your system, the SHR installation will fail because the installer tries to install the components that are bundled with the product.

To resolve this problem, you must clean up the existing components from the system and rerun the installer. For a virtual system, consider reimaging the VM, if feasible.

### **Symptom: Installation Failure caused by SAP BusinessObjects Error**

**Description:** While running the HP Software installer, the installation fails and the following error message is displayed:

*SAP BusinessObjects is installed on the system. Please uninstall it before installing HP SH Reporter.*

**Resolution:** If you have any component of SHR (such as SAP BusinessObjects or Sybase IQ) preinstalled or not cleanly uninstalled from a previous uninstall on your system, the SHR installation will fail because the installer tries to install the components that are bundled with the product.

To resolve this problem, you must clean up the existing components from the system and rerun the installer. For a virtual system, consider reimaging the VM, if feasible.

**Symptom: Installation failure due to missing libraries**

**Description:** While installing SHR, if there any missing libraries the installation pre-check will fail.

**Resolution:** To resolve this problem, perform the following steps:

1. From the file `/tmp/SHR-Missing-Patches.txt` get the list of missing libraries.
2. Install the missing libraries.
3. Re-initiate SHRinstallation.

For more information, see the "Installation Prerequisites" section in this document.

**Symptom: Installation failure due to missing libraries**

**Description:** While installing SHR, if there any missing libraries the installation pre-check will fail.

**Resolution:** To resolve this problem, perform the following steps:

1. From the file `/tmp/SHR-Missing-Patches.txt` get the list of missing libraries.
2. Install the missing libraries.
3. Re-initiate SHRinstallation.

For more information, see the "Installation Prerequisites" section in this document.

**Symptom: Unable to Bring up SHR Services after Successful Installation**

**Description:** If SHR is installed on a virtual machine that is not restarted after the installation, the environment variables set by the installer will not be available to the user resulting in SHR services not coming up in spite of multiple retry.

**Resolution:** After installing SHR, ensure that you restart the virtual machine.

**Symptom: Unable to Bring up SHR Services after Successful Installation**

**Description:** If SHR is installed on a virtual machine that is not restarted after the installation, the environment variables set by the installer will not be available to the user resulting in SHR services not coming up in spite of multiple retry.

**Resolution:** After installing SHR, ensure that you restart the virtual machine.

### **Symptom: Remote Sybase IQ Database Creation Fails**

In the HP Service Health Reporter Configuration Wizard, while trying to create the Sybase database file on a remote system, the post-install fails and the following error message is displayed:

*<time stamp>,018 ERROR, com.hp.bto.bsmr.dao.helper.CreateSybaseIQDatabase.executeSQL, Could not connect to the database.*

*<time stamp>,049 ERROR, com.hp.bto.bsmr.dao.helper.CreateSybaseIQDatabase.executeSQL, Specified database not found*

**Resolution 1:** This error occurs if the database file location specified in the HP Service Health Reporter Configuration Wizard includes one or more spaces in the file path. To resolve this problem, make sure that the specified database file location exists on the remote system. In addition, make sure that the path provided in the Post-Install wizard does not contain any spaces.

**Resolution 2:** This error can occur when adequate disk space is not available on the drive. The installer does not warn in case of a remote database. Increasing the disk space should resolve the issue.

### **Symptom: Sybase IQ Hangs**

**Description:** SHR servers that have four or less CPUs, Sybase IQ hangs because of low `iqgovern` parameter value that is computed automatically.

#### **Resolution:**

Windows: Add "`-iqgovern 50`" parameter to the `%PMDB_HOME%\config\pmdbConfig.cfg` file and restart the Sybase IQ database.

Linux: Add "`-iqgovern 50`" parameter to the `$PMDB_HOME/config/pmdbConfig.cfg` file and restart the Sybase IQ database.

### **Symptom: Sybase IQ Hangs**

**Description:** SHR servers that have four or less CPUs, Sybase IQ hangs because of low `iqgovern` parameter value that is computed automatically.

#### **Resolution:**

Windows: Add "`-iqgovern 50`" parameter to the `%PMDB_HOME%\config\pmdbConfig.cfg` file and restart the Sybase IQ database.

Linux: Add "`-iqgovern 50`" parameter to the `$PMDB_HOME/config/pmdbConfig.cfg` file and restart the Sybase IQ database.

### **Symptom: SHR Fails to Create the Sybase Schema**

**Description:** If SHR fails to create the Sybase schema after you complete the post-installation configuration tasks, an error message appears in the database log files. The Sybase database log files–

<hostname>.0001.srvlog and <hostname>.0001.stderr—are present in the /opt/HP/BSM/Sybase/IQ-16\_0/logfiles directory on Linux.

The following error message appears in the Sybase database log files:

```
"utility_db" (utility_db) stopped
```

Only the Sybase database log files show the error message; no error messages appear in the Administration console.

**Resolution:** Restart the Sybase service by running the following command:

```
service HP_PMDB_Platform_Sybase stop
```

```
service HP_PMDB_Platform_Sybase start
```

**Symptom: SHR Fails to Create the Sybase Schema**

**Description:** If SHR fails to create the Sybase schema after you complete the post-installation configuration tasks, an error message appears in the database log files. The Sybase database log files—<hostname>.0001.srvlog and <hostname>.0001.stderr—are present in the /opt/HP/BSM/Sybase/IQ-16\_0/logfiles directory on Linux.

The following error message appears in the Sybase database log files:

```
"utility_db" (utility_db) stopped
```

Only the Sybase database log files show the error message; no error messages appear in the Administration console.

**Resolution:** Restart the Sybase service by running the following command:

```
service HP_PMDB_Platform_Sybase stop
```

```
service HP_PMDB_Platform_Sybase start
```

**Symptom: After Installation, User is Unable to Perform Post-Install Steps**

**Description:** After installation, when user clicks Next, the subsequent page does not load despite enabling Java scripts to run.

**Resolution:** This occurs when the system date on which SHR is installed is much older than that of the ESX (in case of a VM). In such a scenario, the Tomcat server does not allow any requests from the client. Hence, it is always advisable to update the system date to current and install.

Perform the following steps:

1. Change system date to current.
2. Apply the permanent license.

When the system date is changed by more than three months, the license expires.

3. Restart Admin service, Tomcat server, and SAP BusinessObjects servers.
4. Log on and perform the post configuration again.

**Symptom: After Installation, User is Unable to Perform Post-Install Steps**

**Description:** After installation, when user clicks Next, the subsequent page does not load despite enabling Java scripts to run.

**Resolution:** This occurs when the system date on which SHR is installed is much older than that of the ESX (in case of a VM). In such a scenario, the Tomcat server does not allow any requests from the client. Hence, it is always advisable to update the system date to current and install.

Perform the following steps:

1. Change system date to current.
2. Apply the permanent license.

When the system date is changed by more than three months, the license expires.

3. Restart Admin service, Tomcat server, and SAP BusinessObjects servers.
4. Log on and perform the post configuration again.

**Symptom: SHR Uninstall Fails**

**Description:** Uninstalling SHR might not have completely uninstalled Sybase IQ Server.

**Resolution:** Uninstall Sybase IQ Server Suite 15.4 (64-bit) manually and restart your system.

**Symptom: After Uninstalling SHR, Reinstall Fails**

**Description:** After uninstalling SHR on a Windows system, when a reinstall is performed, the installer fails to launch and displays a Scripting Host not Found error.

**Resolution:** This error is encountered when the Path environment variable in Windows is corrupted. Add the %systemroot%\System32 string to the Path environment variable by performing the following steps:



1. Right-click My Computer, and then click Properties.
2. Click the Advanced tab.
3. Click Environment Variables.
4. In the System Variable group, select Path.
5. Click Edit and add the string %systemroot%\System32 if missing.

**Symptom: After Uninstalling SHR, Reinstall Fails**

**Description:** After uninstalling SHR on a Windows system, when a reinstall is performed, the installer fails to launch and displays a Scripting Host not Found error.

**Resolution:** This error is encountered when the Path environment variable in Windows is corrupted. Add the %systemroot%\System32 string to the Path environment variable by performing the following steps:

1. Right-click My Computer, and then click Properties.
2. Click the Advanced tab.
3. Click Environment Variables.
4. In the System Variable group, select Path.
5. Click Edit and add the string %systemroot%\System32 if missing.

**Symptom: After interrupted installation, unable to continue reinstall with the installed components**

**Description:** This issue may occur when you accidentally quit the SHR installation wizard and later continue to reinstall with the existing components.

**Resolution:** Perform the following steps to resolve this problem:

1. Start the installation wizard and review the Pre-Install Summary.
2. Select the **Force repair of already installed component packages** and click **Install**.
3. If the reinstall fails then, click **Rollback** in the pop-up message. The installed components will be removed.
4. Now perform a new installation.

**Symptom: After interrupted installation, unable to continue reinstall with the installed components**

**Description:** This issue may occur when you accidentally quit the SHR installation wizard and later continue to reinstall with the existing components.

**Resolution:** Perform the following steps to resolve this problem:

1. Start the installation wizard and review the Pre-Install Summary.
2. Select the **Force repair of already installed component packages** and click **Install**.
3. If the reinstall fails then, click **Rollback** in the pop-up message. The installed components will be removed.
4. Now perform a new installation.

**Symptom: Data Collection Failure across all Configured Nodes**

**Description:** Data collection in SHR fails with an Address already in use error logged in the `topologycollector.log` file.

**Resolution:** This error occurs when the number of TCP/IP ports used exceeds the default value of 5000. To resolve this problem, you must make changes in the Windows Registry. Follow these steps:

1. Click **Start > Run**. The Run dialog box appears.
2. In the Open box, type `regedit`. The Registry Editor window appears.
3. On the left pane, expand `HKEY_LOCAL_MACHINE`, expand `SYSTEM`, expand `CurrentControlSet`, expand `Services`, expand `Tcpip`, and then click **Parameters**.
4. On the right pane, right-click anywhere, point to New, and then click **DWORD Value** to add a new entry. Add the following entries:
  - `MaxUserPort` = 65535 (decimal)
  - `MaxFreeTcbs` = 65535 (decimal)
  - `MaxHashTableSize` = 65535 (decimal)
  - `TcpTimedWaitDelay` = 30 (decimal)

Restart the system after making changes in the Registry Editor.

**Symptom: Data Collection Failure across all Configured Nodes**

**Description:** Data collection in SHR fails with an Address already in use error logged in the `topologycollector.log` file.

**Resolution:** This error occurs when the number of TCP/IP ports used exceeds the default value of 5000. To resolve this problem, you must make changes in the Windows Registry. Follow these steps:

1. Click **Start > Run**. The Run dialog box appears.
2. In the Open box, type `regedit`. The Registry Editor window appears.
3. On the left pane, expand *HKEY\_LOCAL\_MACHINE*, expand *SYSTEM*, expand *CurrentControlSet*, expand *Services*, expand *Tcpip*, and then click **Parameters**.
4. On the right pane, right-click anywhere, point to New, and then click **DWORD Value** to add a new entry. Add the following entries:
  - `MaxUserPort` = 65535 (decimal)
  - `MaxFreeTcbs` = 65535 (decimal)
  - `MaxHashTableSize` = 65535 (decimal)
  - `TcpTimedWaitDelay` = 30 (decimal)

Restart the system after making changes in the Registry Editor.

**Symptom:** After uninstall a collector and reinstall it on a system, SHR fails to communicate with the collector.

**Description:** If you uninstall a collector and reinstall it on a system, SHR fails to communicate with the collector and error messages appear when you try to configure the collector in the Administration Console.

You can occasionally experience this issue even after installing the collector for the first time.

**Resolution:** To resolve this, manually import the certificate from the SHR system to the collector system by following these steps:

1. Log on to the collector system.
2. Run the following command:  

```
ovcoreid
```

Note down the ID displayed in the console.
3. Log on to the SHR system.
4. Run the following command:

```
ovcm -issue -file <file> -name<node name>-coreid<core_ID>
```

In this instance, <core\_ID> is the ID that you noted down in step 2.

The command prompts for a password. If you do not want to use a password, press Enter without typing anything.

In this instance, <file> is the name of the certificate file that you want to manually import to the collector system; you must specify the file name with complete path to the directory where you want to store the file. <node name> is the FQDN of the collector system.

5. Transfer the certificate file to the collector system.
6. Log on to the collector system.
7. Run the following command:

```
ovcert -importcert -file<file>
```

**Symptom: After uninstall a collector and reinstall it on a system, SHR fails to communicate with the collector.**

**Description:** If you uninstall a collector and reinstall it on a system, SHR fails to communicate with the collector and error messages appear when you try to configure the collector in the Administration Console.

You can occasionally experience this issue even after installing the collector for the first time.

**Resolution:** To resolve this, manually import the certificate from the SHR system to the collector system by following these steps:

1. Log on to the collector system.
2. Run the following command:

```
ovcoreid
```

Note down the ID displayed in the console.

3. Log on to the SHR system.
4. Run the following command:

```
ovcm -issue -file <file> -name<node name>-coreid<core_ID>
```

In this instance, *<core\_ID>* is the ID that you noted down in step 2.

The command prompts for a password. If you do not want to use a password, press Enter without typing anything.

In this instance, *<file>* is the name of the certificate file that you want to manually import to the collector system; you must specify the file name with complete path to the directory where you want to store the file. *<node name>* is the FQDN of the collector system.

5. Transfer the certificate file to the collector system.
6. Log on to the collector system.
7. Run the following command:

```
ovcert -importcert -file<file>
```

**Symptom: Installation fails for Management database package while installing as Domain user**

**Description:** SHR installation fails with domain user during HPPmdbPostgreSQL package installation with the following error in the install log.

*C:/HP-SHR/Postgres/data ... initdb: could not change permissions of directory "C:/HP-SHR/Postgres/data": Permission denied in %temp%\install-postgresql.log (or) %temp%\bitrock\_installer.log*

**Resolution:** Uninstall SHR and create a local user that is a member of the Local Administrators group with administrator rights and install SHR again.

**Symptom: Installation fails for Management database package while installing as Domain user**

**Description:** SHR installation fails with domain user during HPPmdbPostgreSQL package installation with the following error in the install log.

*C:/HP-SHR/Postgres/data ... initdb: could not change permissions of directory "C:/HP-SHR/Postgres/data": Permission denied in %temp%\install-postgresql.log (or) %temp%\bitrock\_installer.log*

**Resolution:** Uninstall SHR and create a local user that is a member of the Local Administrators group with administrator rights and install SHR again.

**Symptom: Installer fails to display that installation is complete.**

**Description:** This issue may appear while performing SHR installation, upgrade or installing Remote Collectors. The installer progress bar shows that the installation is in process but the **Done** button is enabled. This is because the installer is not refreshed.

**Resolution:** Click **Done** to complete the process and check the install log files as follows to see if all the components are installed.

- Windows: %temp%/../HP-SHR\_9.40\_HPOvInstaller.txt
- Linux: /tmp/HP-SHR\_9.40\_HPOvInstaller.txt

**Symptom: Installer fails to display that installation is complete.**

**Description:** This issue may appear while performing SHR installation, upgrade or installing Remote Collectors. The installer progress bar shows that the installation is in process but the **Done** button is enabled. This is because the installer is not refreshed.

**Resolution:** Click **Done** to complete the process and check the install log files as follows to see if all the components are installed.

- Windows: %temp%/../HP-SHR\_9.40\_HPOvInstaller.txt
- Linux: /tmp/HP-SHR\_9.40\_HPOvInstaller.txt

**Symptom: Installer fails to display that installation is complete.**

**Description:** This issue may appear while performing SHR installation, upgrade or installing Remote Collectors. The installer progress bar shows that the installation is in process but the **Done** button is enabled. This is because the installer is not refreshed.

**Resolution:** Click **Done** to complete the process and check the install log files as follows to see if all the components are installed.

- Windows: %temp%/../HP-SHR\_9.40\_HPOvInstaller.txt
- Linux: /tmp/HP-SHR\_9.40\_HPOvInstaller.txt

**Symptom: Sybase IQ database removal failed after uninstall.**

**Description:** This issue occurs while performing uninstall of SHR. The Sybase IQ database is not removed properly, you will see a pop up as follows:

*"Initialize action for HP Service Health Reporter 9.40 (Removing Sybase IQ schema) was not successful."*

**Resolution:** You can move ahead with the uninstall and run the rollback utility. For the steps, see *"Post Uninstalling SHR"* section in this guide.

**Symptom: Uninstall is not clean.**

**Description:** This issue occurs while performing uninstall of SHR, you will see a pop up saying there is a failure as follows:

*"Initialize action for package HPPmdbCollector 9.40.000 (HP PMDB Collector) (Performing Collection housekeeping) was not successful."*

**Resolution:** You can ignore the pop up and move ahead with the uninstall and run the rollback utility. For the steps, see *"Post Uninstalling SHR"* section in this guide.

**Symptom: Failure in upgrade command.**

**Description:** This issue may occur after upgrade of SHR, you will see a message saying there is a failure in the upgrade command.

**Resolution:** From the command line console run the following script:

```
%Ovinstallldir%\nonOV\perl\a\bin\perl %PMDB_HOME%\upgrade\940\applyPatch.pl %PMDB_HOME%\..\ "%ovinstallldir%" %PMDB_HOME%\upgrade\940
```

```
/opt/OV/nonOV/perl/a/bin/perl /opt/HP/BSM/PMDB/upgrade/940/applyPatch.pl  
/opt/HP/BSM/ /opt/OV/ /opt/HP/BSM/PMDB/upgrade/940/
```

**Symptom: Failure in service precheck while upgrade.**

**Description:** While upgrade, if all the SHR services are not stopped this issue may occur.

**Resolution:** From the PMDB\_HOME\temp folder check the UpgradeServiceCheck.log file to find the cause of failure.

**Symptom: After upgrade, few links in the Administration Console may fail to work.**

**Description:** After upgrade, the CMC and InfoView links may fail to work SHR Administration Console. This issue may occur if the SAP BusinessObjects services are not running.

**Resolution:** Perform the following steps to stop the SAP BusinessObjects services:

1. Log on to the SHR system
2. Open the Services window
3. Start the Business Objects Webserver Service

1. Log on to the system as root.
2. Run the following command to start the webserver:

```
sh /opt/HP/BSM/BO/bobje/tomcatstartup.sh
```

**Symptom: Installation with username having special character "&" requires system startup.**

**Description:** While installing SHR with username having special character & then the system requests for startup.

**Resolution:** Click **Continue** and proceed with your installation.

**Symptom: Installation with username having special character "&" requires system startup.**

**Description:** While installing SHR with username having special character & then the system requests for startup.

**Resolution:** Click **Continue** and proceed with your installation.

**Symptom: After Uninstall, reinstall hangs on RHEL 5.5.**

**Description:** After uninstall, when you reinstall SHR on Linux RHEL 5.5, the system hangs while installing BusinessObjects.

**Resolution:** Ensure that you perform SHR installation on a new Linux RHEL 5.5 system.

**Symptom: YUM check warning after SHR installation**

**Description:** After installing SHR and meeting all the pre-requisites, the following message appears with a list of missing libraries:

*Found 42 pre-existing rpmdb problem(s), 'yum check' output follows:*

**Resolution:** If you get a list of missing libraries while performing the YUM check, you can ignore these libraries as they are not mandatory for SHR. This does not affect the functionality of SHR.

## Install Business Service Management

**Note:** In general, you should follow the product documentation for each product to install and configure the product, and then read and implement the associated integration guides.

## Install Executive Scorecard

**Note:** In general, you should follow the product documentation for each product to install and configure the product, and then read and implement the associated integration guides.

## Enable the IT Business Analytics data sources and content packs

After you have completed the installation of all products, you should activate the IT Business Analytics data sources and content packs for each component you wish to report on. Because IT Business Analytics reads data directly from the components RDBMS, you will need to know the connection information for each database, such as the IP address, user name and password.

**Note:** The following sections are excerpted from the *IT Business Analytics Administration Guide* and



the *IT Business Analytics Content Development Guide* respectively.

## Integration guidance

The ITSM Enterprise Suite includes many possible integrations between its myriad components. Those integrations and a brief description of their function are listed in the following table:

<b>Products</b>	<b>Description</b>	<b>Guide</b>
Service Manager- Universal Configuration Management Database		
Universal Configuration Management Database - Asset Manager		
Universal Configuration Management Database - Business Service Management		
Universal Configuration Management Database - Operations Manager i		
Operations Manager i - Business Service Management		
Operations Manager i - Service Health Reporter		
Service Health Reporter - Service Manager		
IT Business Analytics - Service Manager		
IT Business Analytics - Business Service Management		
IT Business Analytics - Asset Manager		

## Chapter 2: Integrate UCMDB and Service Manager using the enhanced adapter

**Note:** The following sections are excerpted from the integrations section of the Universal Configuration Management Database Help Center. However, to ensure you have the latest information, you should reference the latest published version of the document in question.

This chapter includes:

Introduction .....	79
Integration Setup .....	87
Multi-Tenancy (Multi-Company) Setup .....	138
Standards and Best Practices .....	158

### Introduction

This chapter provides an overview of the HP Universal CMDB (UCMDB) - HP Service Manager (SM) integration (also referred to as the Universal CMDB (UCMDB) integration or UCMDB-SM integration throughout this document).

This chapter includes:

- ["Who Should Read this Guide" below](#)
- ["Purpose of the Integration" on the next page](#)
- ["How CI information is Synchronized Between UCMDB and Service Manager" on page 84](#)

### Who Should Read this Guide

This guide is intended for a system implementer or system administrator who will be establishing and maintaining a connection between the HP Universal CMDB (UCMDB) and Service Manager (SM) systems. This guide assumes that you have administrative access to both systems. The procedures in this guide may duplicate information that is available in your UCMDB and Service Manager help systems, but is provided here for your convenience.

**Note:** This document provides instructions on setting up an integration between the two products by using the Service Manager Enhanced Generic Adapter, which is available only in UCMDB 10.20 or later. If you do not want to use the enhanced generic adapter, refer to the previous UCMDB-SM integration documentation that is based on your specific adapter.

## Purpose of the Integration

An integration between HP Universal CMDB (UCMDB) and HP Service Manager enables you to share information about the actual state of a configuration item (CI) between your UCMDB system and a Service Manager system. CIs commonly include IT services, hardware and software. Any organization that wants to implement the best practices Configuration Management and Change Management ITIL processes can use this integration to verify that CIs actually have the attribute values the organization has agreed to support.

You can use this integration to automate the creation of Service Manager change or incident records to update or rollback CIs that have unexpected attribute values. Service Manager allows you to programmatically define what actions you want to take whenever a CI's actual state does not match the expected state as defined in the CI record.

The integration offers several different ways for users to view CI actual state information:

- By default, the integration automatically updates the managed fields of Service Manager CI records as part of the regular UCMDB synchronization schedule. You can choose the option to configure the integration to automatically create change or incident records instead.
- A Service Manager user can view the current actual state of a CI by looking at the Actual State section in the CI record. When you open the Actual State section, Service Manager makes a web services request to UCMDB and displays all CI attributes the request returns. Service Manager only makes the web service call when you open this section.
- A Service Manager user can use the **View in UCMDB** option to log in to the UCMDB system and view the current CI attributes from UCMDB. The Service Manager user must have a valid UCMDB user name and password to log in to the UCMDB system.

## Supported Use Cases

This section describes use cases that are supported by the UCMDB-SM integration. The supported use cases provide the core business processes that are enabled by the UCMDB-SM integration.

There are four main business use cases supported by the UCMDB-SM integration. They are as follows:

- **Planned Change:** A change created in SM through the formal SM change process.
- **Unplanned Change:** A change or incident that occurred in SM and does not conform to the formal SM change process.
- **Retrieving SM Ticket Information:** The ability to view SM ticket information in UCMDB.
- **Actual State:** The ability to view the UCMDB CI information in SM.

All of the use cases provide important functionalities that enable the user to perform ITIL (IT Infrastructure Library) processes. The ITIL processes refer to a set of best practices that define and outline how organizations should manage their IT.

## Enabling ITIL Processes

By activating CI push from UCMDB to SM the user facilitates ITIL processes such as Incident, Problem and Change Management in SM.

SM utilizes the data pushed from UCMDB in the following modules:

- **Incident Management:** the Service Desk operator (SD Agent) selects the “Service” and the “Affected CI” for the specific Incident record.
- **Problem Management:** the SD agent selects the “Service”, “Affected CI(s)”, and the “Primary CI” for the specific Problem record.
- **Change Management:** the SD agent selects the “Service” and the “Affected CI(s)” for the specific Change record.

In each of the previously mentioned ITIL processes, SM utilizes CI information for Service, Affected CIs and Primary CIs that all originate in UCMDB.

## Managing Planned Changes

The purpose of the “Planned Change” use case is to provide IT organizations a formal process by which changes to the IT infrastructure are introduced after thorough review and analysis. This is performed according to the “Change Management” process defined in ITIL.

A “Planned Change” is initiated by the SM user through the formal “Change Management” process module in SM. This is followed by the actual change implementation.

The actual changes are discovered by a discovery tool such as HP DDMA, and then updated in UCMDB and the relevant modifications are pushed to SM. Once the user has validated the change, the user closes the relevant planned change in SM.

## Managing Unplanned Changes

The purpose of the “Unplanned Change” use case is to provide IT organizations a formal process by which all changes that occur to the IT infrastructure are both logged and conventionalized through the organizations formal approval process.

An “Unplanned Change” is a change that is recognized by a Discovery tool such as DDMA. The change is first updated and visible in UCMDB and then the data is pushed to SM. SM recognizes the change and as a result an “Incident” or “Change” record is generated.

These Changes are seen also in the SM “Pending Changes” section in the Configuration Item form, once approved they are moved to the SM “Historic Changes” section.

## Retrieving Service Manager Ticket Information

Retrieving SM ticket information from within UCMDB provides all HP Software applications users with access to this information by using UCMDB's federation capabilities and supporting APIs. These applications include Business Service Management (BSM), Asset Manager (AM), Operations Orchestration (OO), etc.

SM ticket data is accessed from within UCMDB using UCMDB federation capabilities. SM ticket data includes Incident, Problem and Change records as well as a key set of their attributes.

UCMDB enables users to create reports/views that combine the federated ticket data from SM with CI information from UCMDB.

## Retrieving Actual State of UCMDB CIs

The purpose of “Actual State” is to enable SM users insight into CIs' current state as detected by “Discovery Tools” and populated in UCMDB. This state provides up-to-date information that may vary from the information displayed in SM both in content and in scope.

The “Actual State” of the CI is displayed in SM in order to enable the user to validate the current state of the CI that resides in UCMDB or in another data repository.

SM users retrieve the Actual State of CIs from UCMDB or additional data sources by viewing the CI's Actual State section in the SM Configuration Item form.

## Accessing UCMDB CIs from Service Manager

SM users can open the UCMDB User Interface in the context of a specific CI, by clicking the **View in UCMDB** button in the SM CI record. When the user clicks the **View in UCMDB** button, a UCMDB login screen is displayed; After the user enters a UCMDB username and password, UCMDB displays a topological view of the specific CI together with all related CIs that are linked to it.

**Tip:** You can configure Lightweight Single Sign-On (LW-SSO) for the integration, so that Service Manager web client users can bypass the UCMDB login screen after clicking the **View in UCMDB** button. For more information, see ["How to Enable Lightweight Single Sign-On \(LW-SSO\) Configuration" on page 172.](#)

If the UCMDB Browser URL is specified in the SM System Information Record, this button is replaced by the **View in UCMDB Browser** button. When you click the **View in UCMDB Browser** button, a UCMDB Browser login screen is displayed; After you enter a UCMDB Browser username and password, the CI is displayed in the UCMDB Browser user interface.

## Core Features

This section explains the rudimentary concepts behind the Federation, Push, and Population features as they pertain to the integration.

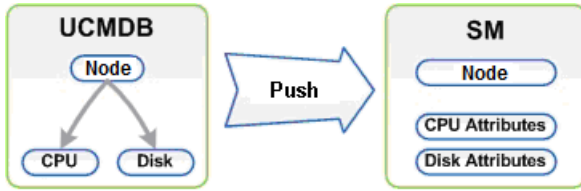
This section includes:

- ["Push" below](#)
- ["Federation" on the next page](#)
- [" Population" on the next page](#)

### Push

UCMDB can automatically discover most types of CIs available in Service Manager. This integration enables you to push these types of CIs from UCMDB to Service Manager.

The following figure shows how data is pushed from UCMDB to Service Manager (SM). The data is physically pushed (copied) from UCMDB to SM. Once the data is physically located in SM, the data is utilized by the SM user that consumes this information in various SM processes.



**Note: CI Type and Attribute Push**

Only information that is physically present in UCMDB can be pushed to SM.

Federation

With the federation feature, UCMDB pulls various ticket information (for example, Incident, Problem, and Change ticket information) from SM. This enables users to see Ticket information in UCMDB as Ticket CIs that are connected to the relevant Nodes.

When data is federated (reflected or mirrored) from SM to UCMDB, the data is not physically present in UCMDB; instead it is passed over to UCMDB through Web Services.

Population

You can also use this integration to populate those types of CIs that UCMDB cannot automatically discover or CIs that have been created in Service Manager before you have a UCMDB system deployed. For more information, see ["When Do I Need the Population Feature?" on page 176.](#)

Population is the reverse of Push. The following figure shows how data is populated from SM to UCMDB. One SM CI record with multiple attributes is transferred to UCMDB as multiple CI records.



## How CI information is Synchronized Between UCMDB and Service Manager

This section explains how CI information is transferred between the UCMDB and Service Manager systems.



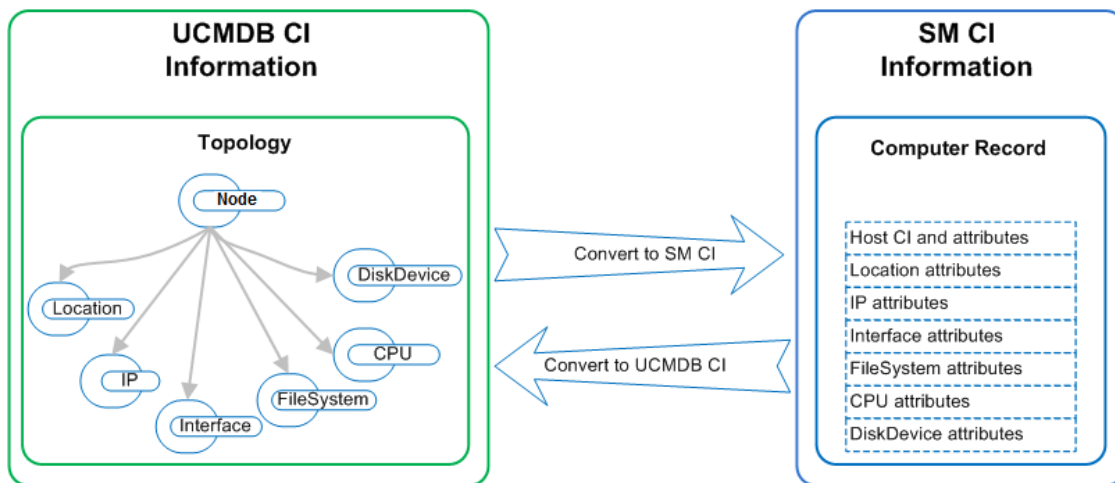
This section includes:

- ["CI Information Usage" below](#)
- ["High-Level Components of the Integration" on the next page](#)
- ["Relationships Between Integration Components" on the next page](#)
- ["What Information Is Stored in UCMDB" on page 87](#)
- ["What Information Is Stored in Service Manager" on page 87](#)

## CI Information Usage

When referring to the concept of CI information it is important to make the distinction between a UCMDB CI and a Service Manager (SM) CI. The UCMDB model represents a topology that contains a number of CI types and relationships.

The UCMDB topology can be represented in Service Manager as a single entity. Multiple CIs from UCMDB and their attributes are merged into a single record in SM and the relevant UCMDB attributes are mapped to their appropriate counterparts in the SM record.



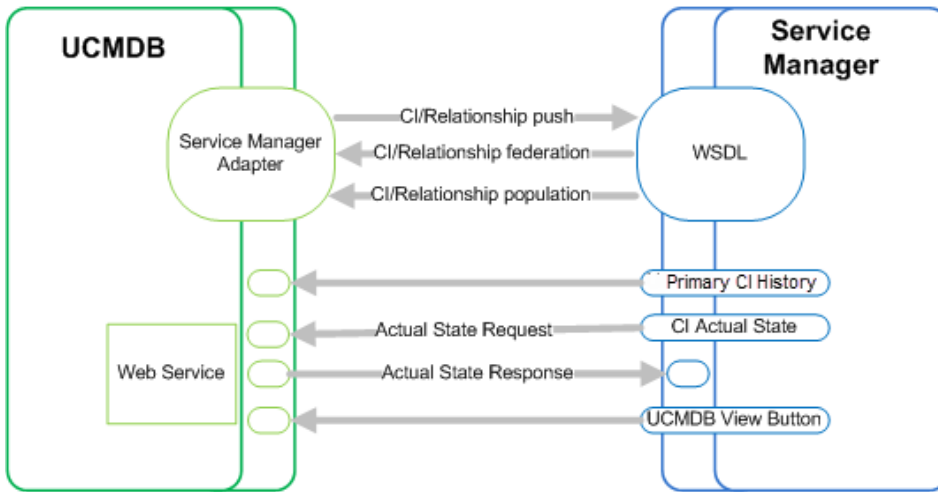
The above figure shows the correlation between the UCMDB topological model and its representation of the Computer Instance together with its parallel representation in SM. The SM computer CI contains all of the UCMDB information that is passed through the integration.

In the push flow, in the UCMDB topological view several CIs such as Node, IP, Interface, Location, File System, CPU, Disk Device and their Relationships are converted into a single SM computer record with the IP, MAC Address and Location, File System, CPU and Disk Device attributes.

In the population flow, the conversion is reversed.

## High-Level Components of the Integration

The following diagram shows the high-level components of the integration, and illustrates the interactions between UCMDB and Service Manager.

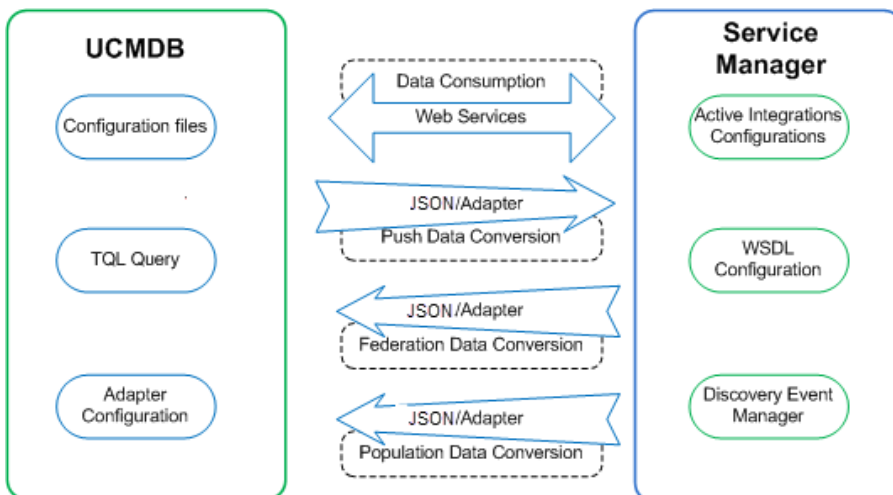


## Relationships Between Integration Components

The following figure illustrates the relationships between the Service Manager Adapter components in UCMDB and the associated components in Service Manager.

The Service Manager Adapter includes configuration files, which are used to map UCMDB entities to their counterparts in Service Manager during data push, as well as map Service Manager CIs to UCMDB entities during population.

The configuration files utilize UCMDB queries that define a superset of data relevant for the integration.



## What Information Is Stored in UCMDB

Your UCMDB system stores the actual state of CIs and CI relationships as CI attributes. Typically, UCMDB uses one or more integrations and discovery mechanisms (feeders) to automatically detect CI attribute values. The UCMDB-SM integration only uses a subset of the CI attributes available in a UCMDB system.

For more information, see ["Tailoring the Integration" on page 182](#).

## What Information Is Stored in Service Manager

Your Service Manager system stores the managed or expected state of CIs and CI relationships as attribute values in a CI record. To be part of the integration, a CI attribute in your UCMDB system must map to a managed field in the Service Manager CI record. You can add, remove, or update the managed fields that are part of the integration by tailoring the Service Manager web services that manage the integration.

Service Manager runs according to a set of rules that define what actions you want the system to take whenever a CI's actual state does not match the expected state as defined in the CI record. You define these rules from the Discovery Event Manager (DEM) in Service Manager where you can do the following:

- Automatically update a CI record to match the attribute values listed in the actual state. (This is the default behavior.)
- Automatically create a change record to review the differences between the actual state and the managed state.
- Automatically create an incident record to review the differences between the actual state and the managed state.

## Integration Setup

Before implementing the integration in your production environment, you can set up the integration in a test environment using the out-of-the-box integration configurations. This chapter describes the basic integration setup tasks without any tailoring or multi-tenancy configurations. It covers the following topics:

- ["Integration Requirements" on the next page](#)
- ["How to Migrate Your Integration" on page 89](#)

- ["Integration Setup Overview" on page 101](#)
- ["HP Service Manager Setup" on page 101](#)
- ["HP Universal CMDB Setup" on page 104](#)
- ["Populating UCMDB with Service Manager CI Data" on page 109](#)
- ["Pushing UCMDB CI Data to Service Manager" on page 113](#)
- ["Federating Service Manager Ticket Data to UCMDB" on page 120](#)

**Tip:** Before you proceed to implementing the integration in your production environment, you can refer to the following chapters for further information:

- ["Multi-Tenancy \(Multi-Company\) Setup" on page 138](#), which describes how you set up the integration in multi-tenancy mode.
- ["Standards and Best Practices" on page 158](#), which describes best practices for implementing the integration and also provides Frequently-Asked-Questions information.
- ["Tailoring the Integration" on page 182](#), which describes how you can tailor the integration to better suit your business needs.
- ["Troubleshooting" on page 257](#), which provides information on troubleshooting data push and population issues.

## Integration Requirements

The supported product versions of this integration are listed in the following table.

### Supported product versions

Service Manager	UCMDB
9.40	10.20 or later

**Note:** This document describes the integration based on the Service Manager Enhanced Generic Adapter, which has been introduced since UCMDB 10.20. If you are using a previous version of UCMDB, you can use the XSLT-based adapter (Service Manager 9.x adapter) instead.

You must set up the following required components to establish an integration between UCMDB and Service Manager.

- HP Universal CMDB installation  
Add a UCMDB Probe for the population feature if you do not already have one.
- HP Service Manager installation  
Add the UCMDB URL to the System Information Record. See ["How to Add the UCMDB Connection Information" on page 103](#).
- Network connection between the HP Universal CMDB and HP Service Manager systems.

For instructions on installing and configuring your systems, see the UCMDB and Service Manager documentation.

## How to Migrate Your Integration

Starting with version 10.20, UCMDB has introduced an adapter named Service Manager Enhanced Generic Adapter, which is based on the enhanced UCMDB integration framework. This enhanced adapter can work with Service Manager (SM) to additionally provide the following major features that ServiceManagerAdapter9-x does not support:

- A Visual Mapping tool: a tool that provides a graphic user interface for easy field mapping between UCMDB and SM, without the need to work with XSLT mapping files as you do with the previous adapters. This tool still provides an XML editor, which allows you to directly edit the mapping file code lines.
- Centralized CI type customization. For more information, see ["Centralized CI Management" on page 107](#).

For more information about the Visual Mapping tool and the Generic Adapter framework, refer to the *Universal CMDB Help Center*.

To use the Service Manager Enhanced Generic Adapter, existing customers need to migrate their product systems and integration configurations. The migration process is as follows:

1. ["Upgrade Service Manager to Version 9.40" on the next page](#)
2. ["Upgrade UCMDB to Version 10.20" on the next page](#)
3. ["Enable the RESTful APIs for Custom CI Types in Service Manager" on the next page](#)

4. ["Reconfigure an Integration Point Using the Service Manager Enhanced Generic Adapter in UCMDB" on page 93](#)
5. ["Update the Configurations for Custom CI Types in UCMDB" on page 94](#)

## Upgrade Service Manager to Version 9.40

If you want to use the Service Manager Enhanced Generic Adapter (ServiceManagerEnhancedAdapter9-x) for the UCMDB-SM integration, you must upgrade Service Manager to version 9.40, which introduced the following features that are required for the adapter to work:

- Restful APIs for Push, Population, and Federation
- Restful APIs for Device Type retrieval
- Restful APIs for Device Type synchronization

For information about how to upgrade Service Manager to version 9.40, see the *Service Manager 9.40 Installation and Upgrade Documentation Center*.

## Upgrade UCMDB to Version 10.20

The Service Manager Enhanced Generic Adapter (ServiceManagerEnhancedAdapter9-x) also relies on many new features introduced in UCMDB 10.20 to work.

For information about how to upgrade your UCMDB system to version 10.20, see the *Universal CMDB 10.20 Deployment Guide*.

**Note:** If you have upgraded one or both of your product systems but still want to stay with an old adapter, refer to the previous UCMDB-SM integration documentation that is based on your specific adapter.

## Enable the RESTful APIs for Custom CI Types in Service Manager

If your existing UCMDB-SM integration environment uses any custom CI types, you need to update the External Access Definition of each custom CI type in Service Manager to enable its RESTful API. The Service Manager Enhanced Generic Adapter requires the ucmdbIntegration RESTful APIs to work.

The following table describes the parameters that you need to update in each External Access Definition.

Parameter	Description	Example Value
RESTful Enabled	Enables the RESTful API (must be set to true to enable the RESTful API)	true
Resource Collection Name	Specifies a unique name for the resource collection. The following convention is recommended: <WS Object Name>+'s'	ucmdbATMs
Resource Name	Specifies a name for the resource. The following convention is recommended: <WS Object Name>	ucmdbATM
Unique Keys	Unique keys of the relevant device table	logical.name
Max Records Returned in Query	Maximum number of returned records in one query	1000
Resource Collection Actions - POST	Action to invoke for a POST request on the resource collection	Create
Resource Actions - POST	Action to invoke for a POST request on the resource	Create
Resource Actions - PUT	Action to invoke for a PUT request on the resource	Update
Resource Actions - DELETE	Action to invoke for a DELETE request on the resource	Delete

To access these parameters, follow these steps:

1. Open the External Access Definition for the custom CI type.
  - a. Click **Tailoring > Web Services > Web Service Configuration** to open the External Access Definition form.
  - b. In the Service Name field, type `ucmdbIntegration`.
  - c. In the Name field, select the table for the custom CI type.
  - d. In the Object Name field, type the relevant web service object name.
  - e. Click **Search**.
2. Click the **Restful** tab, and update the parameters.

As an example, the following figure shows the external access definition of a custom CI type named ATM Machine.

☑ OK    ✕ Cancel    + Add    📄 Save    ✕ Delete    🔍 Find    📄 Fill    |    More ▾

---

### External Access Definition

---

Service Name: \* ucmdbIntegration

Name: \* joinATM

Object Name: ucmdbATM

Allowed Actions    Expressions    Fields    **RESTful**

RESTful Enabled     Attachment Enabled

Resource Collection Name: \* ucmdbATMs

Resource Name: \* ucmdbATM

Unique Keys: \* logical.name

Max Records Returned in Query: 1000

Query Authorization:

Resource Collection Actions:

POST: Create

Resource Actions:

POST: Create

PUT: Update

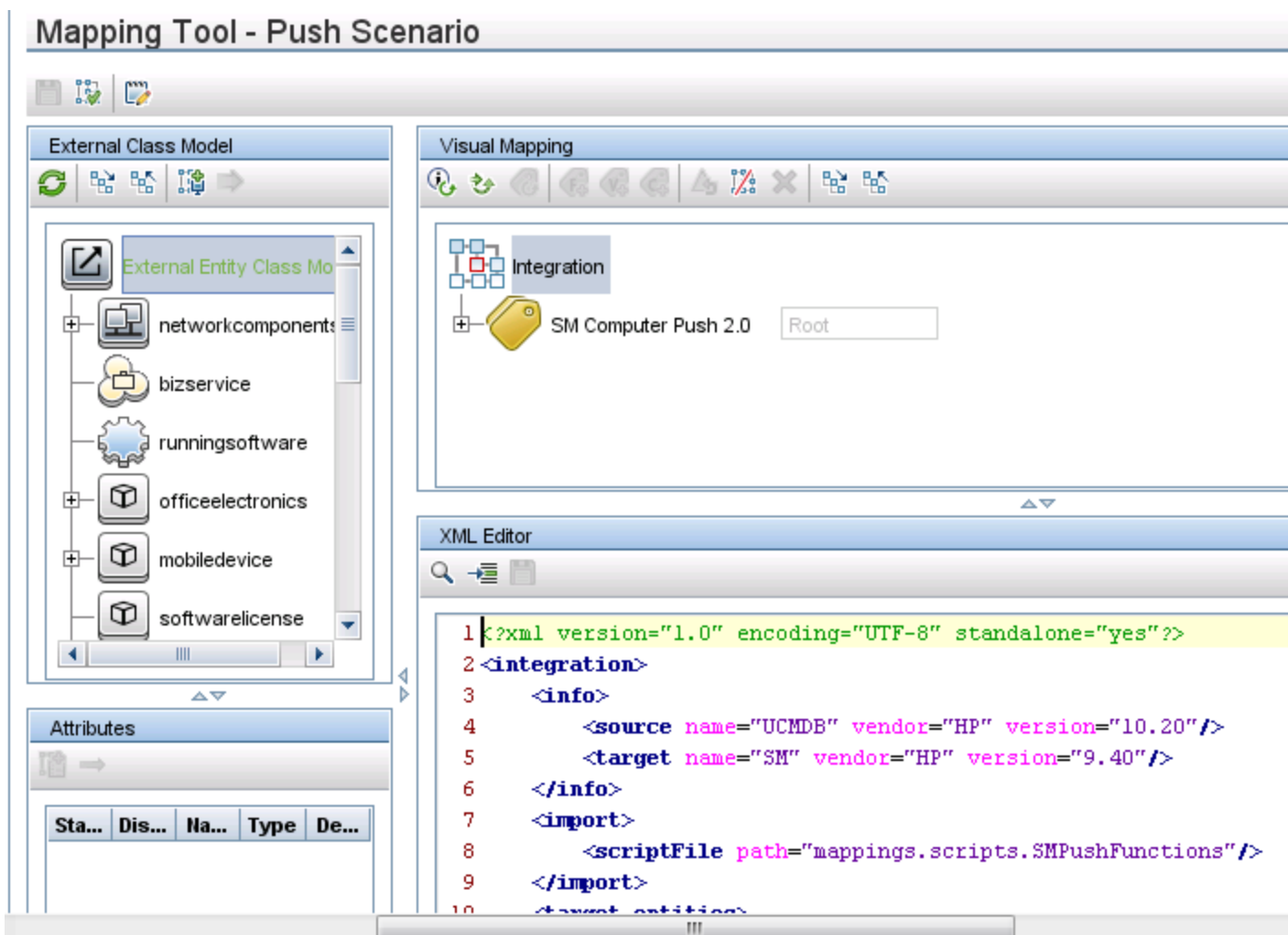
DELETE: Delete



## Reconfigure an Integration Point Using the Service Manager Enhanced Generic Adapter in UCMDB

To use the Visual Mapping tool, you need to enable the Service Manager Enhanced Generic Adapter (ServiceManagerEnhancedAdapater 9.x) on the UCMDB side. To do so, set up an integration point to use this adapter, and activate the integration point. For details, see ["How to Create an Integration Point in UCMDB" on page 104.](#)

Once this adapter is successfully enabled, you should be able to open one of the out-of-box mapping files (such as "SM Computer Push 2.0.xml") in the Visual Mapping tool interface without any problems. The following figure shows the Visual Mapping tool interface in which the "SM Computer Push 2.0.xml" file is open.



For details on how to use the Visual Mapping tool, refer to the *Universal CMDB 9.40 Data Flow Management Guide*.

## Update the Configurations for Custom CI Types in UCMDB

The configurations for all out-of-the-box CI types in UCMDB 10.20 are already based on the Service Manager Enhanced Generic Adapter. However, you still need to perform the following tasks to manually update the configurations for your custom CI types in UCMDB.

### Task 1. Convert the mapping scripts from XSLT to XML and Groovy.

The old Service Manager adapters use XSLT scripts for field mapping between UCMDB and SM. Most of the mapping scripts were developed for the four typical mapping scenarios described in the following table.

Scenario	Notes
One to One field mapping	<ul style="list-style-type: none"> <li>One field in one side is mapped to one field in the other side</li> <li>The value is synchronized between UCMDB and SM directly (without value conversion)</li> </ul>
One to Many field mapping	<ul style="list-style-type: none"> <li>One field in one side is mapped to many fields in the other side</li> <li>The value of the one field is calculated based on the values of many fields</li> </ul>
Many to Many field mapping	<ul style="list-style-type: none"> <li>Multiple fields are defined as a child CI type of the relevant CI type</li> <li>A single record has multiple child instances</li> </ul>
Value Conversion	<ul style="list-style-type: none"> <li>The value of one field is converted to another value based on a specific algorithm</li> <li>Value conversion can happen on all kinds of field mappings (one to one, one to many, and many to many)</li> </ul>

The Service Manager Enhanced Generic Adapter uses XML and Groovy mapping scripts. Old mapping scripts for the out-of-the-box CI types have been converted to XML and Groovy by default. However, you still need to convert your existing custom scripts to XML and Groovy.

To convert a custom mapping script, you need to identify its mapping scenario (One to One, One to Many, Many to Many, or Value Conversion) first; then you can convert the script by referring to the following sample scripts for your specific scenario.

**Tip:** In all mapping scenarios, you can use the Visual Mapping tool to generate an XML script skeleton. Compared with the use of an old XSLT adapter, this is much easier as you can drag and drop the mapped fields to generate the script instead of writing every code line from scratch. For details about how to use XML and Groovy scripts in the UCMDB integration framework, see the *HP Universal CMDB 10.20 Developer Reference Guide*.

**Tip:** The Service Manager Enhanced Generic Adapter has four out-of-the-box Groovy scripts, which you can use as a reference:

- `SMUtils.groovy`: defines common methods used for push, population, and federation.
- `SMPushFunctions.groovy`: defines the methods used in the out-of-the-box push mapping scripts.
- `SMPopulateFunctions.groovy`: defines the methods used in the out-of-the-box population mapping scripts.
- `SMFederationFunctions.groovy`: defines the methods used in the out-of-the-box federation mapping scripts.

Additionally, the `SMFederationConverter.groovy` script defines value conversions for federation.

Field mapping must use the correct data type for each attribute. It is convenient to determine the data type for an attribute by using the Visual Mapping tool. The following figure illustrates the data type for the `CIIdentifier` attribute. You can either find the data type from the External Class Model Attributes pane or simply drag and drop the attribute from that pane to the Visual Mapping area to automatically get the correct data type.

## Mapping Tool - Push Scenario

The screenshot displays the Mapping Tool interface for a Push Scenario. It is divided into four main panels:

- External Class Model:** A tree view showing various class categories like mobiledevice, softwarelicense, mainframe, example, computer, cigroup, switch, and telecom. The 'computer' class is selected.
- Visual Mapping:** A diagram showing the mapping of the 'computer' class to its attributes. The 'CIIdentifier' attribute is highlighted, and its mapping is shown as 'Root['display\_label']'.
- Attributes:** A table listing the attributes of the selected class. The 'CIIdentifier' attribute is circled in red.
- XML Editor:** A text editor showing the XML output. The mapping for 'CIIdentifier' is highlighted in yellow, showing the following code:
 

```
1 <?xml version="1.0" encoding="UTF-8" sta
2 <target_mapping datatype="STRING" name=
3
```

### One to One Field Mapping

Sample Script in XSLT:

```
<Type>computer</Type>
```

Or

```
<UCMDBId><xsl:value-of select="@id"/></UCMDBId>
```

Or

```
<xsl:for-each select="@display_label">
  <CIIdentifier><xsl:value-of select="."/></CIIdentifier>
</xsl:for-each>
```

Sample Script in XML (Groovy is not required in this scenario):

```
<target_mapping datatype="STRING" name="CIIdentifier" value="Root['display_label']
"/>
```

Where: The **datatype** must be the correct data type defined in Service Manager for the attribute. You can find the data type from the Visual Mapping tool interface.

### One to Many Field Mapping

Sample Script in XSLT:

```
<xsl:variable name="prefix" select="'Value&gt;'" />
<xsl:variable name="suffix" select="'&lt;/Value'" />
<Subtype>
  <xsl:choose>
    <xsl:when test="contains(@node_role,concat($prefix,'desktop',$suffix))"
">Desktop</xsl:when>
    <xsl:when test="@os_family">
      <xsl:value-of select="@os_family" />
    </xsl:when>
    <xsl:otherwise>Server</xsl:otherwise>
  </xsl:choose>
</Subtype>
```

Sample Script in XML:

```
<target_mapping datatype="STRING" name="Subtype" value="SMPushFunctions.getSubType
('computer',Root['node_role'],Root['os_family'])"/>
```

Sample Script in Groovy:

You must develop a groovy function to implement the mapping logic. Usually the relevant fields in the counterpart system will be used as the parameters of the Groovy function.

### Many to Many Field Mapping

Sample Script in XSLT:

```
<xsl:for-each select="cpus">
  <cpu>
    <xsl:for-each select="cpu">
      <cpu>
        <CpuID><xsl:value-of select="@cpu_id"/></CpuID>
        <CpuName><xsl:value-of select="@name"/></CpuName>
        <CpuClockSpeed>
          <xsl:value-of select="@cpu_clock_speed"/>
        </CpuClockSpeed>
      </cpu>
    </xsl:for-each>
  </cpu>
</xsl:for-each>
```

```

    </cpu>
</xsl:for-each>

```

#### Sample Script in XML:

```

<for-each-source-entity count-index="i" source-entities="Root.Cpu">
  <target_entity name="cpu">
    <target_mapping datatype="STRING" name="CpuID" value="Root.Cpu[i]
['cpu_id']"/>
    <target_mapping datatype="STRING" name="CpuName" value="Root.Cpu[i]
['name']"/>
    <target_mapping datatype="STRING" name="CpuClockSpeed" value="Root.Cpu
[i]['cpu_clock_speed']"/>
  </target_entity>
</for-each-source-entity>

```

Where: *<source-entities>* (**Root Cpu**) comes from the local TQL query structure, while *<target\_entity name>* (**cpu**) comes from the *<cpu>* tag in the XSLT file.

**Note:** Groovy scripts are not required for this scenario.

#### Value Conversion

##### Sample Script in XSLT:

```

<xsl:if test="@customer_id">
  <xsl:variable name="ucmdbCustomerId" select="@customer_id"/>
  <xsl:variable name="tenantMappingEntry" select="document('SM_MT_
mapping.xml')/list['TenantMapping']/entry[@ucmdb=$ucmdbCustomerId"/>
  <xsl:choose>
    <xsl:when test="$tenantMappingEntry">
      <CustomerId><xsl:value-of
select="$tenantMappingEntry/@sm"/></CustomerId>
    </xsl:when>
    <xsl:otherwise>
      <CustomerId><xsl:value-of select="@customer_id"/></CustomerId>
    </xsl:otherwise>
  </xsl:choose>
</xsl:if>

```

Or

```

<xsl:if test="contains(@node_role,concat($prefix,'virtualized_system',$suffix))">
  <IsVisualization>true</IsVisualization>
</xsl:if>

```

##### Sample Script in XML:

```
<target_mapping datatype="STRING" name="CustomerId"
  value="SMPushFunctions.getCustomerId(CustomerInformation)"/>
```

Or

```
<target_mapping datatype="BOOLEAN" name="IsVisualization"
  value="SMPushFunctions.isVisualization(Root['node_role'])"/>
```

**Note:** Usually the relevant field in the counterpart system will be used as one of the parameters of the Groovy function.

Sample Scripts in Groovy:

```
/**
 * Priority Mapping
 * [uCMDB value : SM value ]
 */
private static final def PriorityMapping = [
  "1_critical":"1",
  "2_high":"2",
  "3_average":"3",
  "4_low":"4"];

/**

 * Convert the Priority value from uCMDB to SM
 *
 * @param priority uCMDB Priority value
 * @return SM Priority value
 */
public static String convertPriority(String priority, DataAdapterLogger log)
{
  return convertEnumValue(priority,"Priority");
}
```

Task 2. Update the configuration files.

Update the files as described in the following table.

File	Description
icon.properties	<p>This configuration file defines the UCMDB icons that will be displayed in the mapping tool for Service Manager Device Types. If you have defined any custom device types in SM, you can update this configuration file to assign them appropriate icons.</p> <p>For more information, see the <i>Universal CMDB Data Flow Management Guide</i>.</p>
sm.properties	<p>This is the integration adapter configuration file. If you have modified the out-of-the-box settings in your old sm.properties file (for example, the number of concurrent threads), you need to update this configuration file accordingly for those options that are still valid in the Service Manager Enhanced Generic Adapter.</p> <p>For more information, see <a href="#">"How to Update the Integration Adapter Configuration File (sm.properties)" on page 198</a>.</p>
smFedConf.xml	<p>This configuration file is used for federation and is introduced by the Service Manager Enhanced Generic Adapter. If you have any custom logic for the federation feature, such as custom fields for federation CIs, you need to update this configuration file.</p> <p>For more information, see <a href="#">"Troubleshooting Federation Issues" on page 272</a>.</p>
smPopConf.xml	<p>This configuration file is used for population and replaces the old smPopConfFile.xml configuration file. If you have any custom logic for the population feature, such as custom query conditions, you need to update this configuration file.</p> <p>For more information, see the following topics:</p> <p><a href="#">"What Is the Purpose of the &lt;container&gt; Element in the Population Configuration File (smPopConf.xml)?" on page 180</a></p> <p><a href="#">"No TQL Query Configured in smPopConf.xml " on page 268</a></p>
smPushConf.xml	<p>This configuration file is used for relationship data push and replaces the old smSyncConfFile.xml configuration file.</p> <p>For more information about this file, see the following topics:</p> <p><a href="#">"How to Map a Relationship Type Query to the Service Manager Web Service Object" on page 245</a></p> <p><a href="#">"Query not Configured in smPushConf.xml" on page 262</a></p>



### Task 3. Enable Push, Population and Federation for CI types.

Once you have completed the tasks described above, you must edit your integration point to add Push and Population jobs for the CI types, as well as enable Federation for supported CI types. For details, see the following topics:

["How to Create an Integration Point in UCMDB" on page 104](#)

["How to Define Data Push Jobs in UCMDB" on page 113](#)

["How to Define Population Jobs in UCMDB" on page 109](#)

["How to Add an Attribute of a Supported CI Type for Federation" on page 251](#)

## Integration Setup Overview

The integration requires setup on both the UCMDB and Service Manager systems.

This task includes the following steps:

1. Set up the Service Manager system.  
See ["HP Service Manager Setup" below](#).
2. Set up the UCMDB system.  
See ["HP Universal CMDB Setup" on page 104](#).
3. Run the UCMDB population jobs to synchronize CIs to UCMDB.  
See ["Populating UCMDB with Service Manager CI Data" on page 109](#).
4. Run the UCMDB data push jobs to transfer CIs to Service Manager.  
See ["Populating UCMDB with Service Manager CI Data" on page 109](#).

## HP Service Manager Setup

You must complete the following tasks from your Service Manager system to support the integration.

1. Create a dedicated integration user account in Service Manager.  
See ["How to Create an Integration User Account" on the next page](#).
2. Add the UCMDB connection information to the system information record.  
See ["How to Add the UCMDB Connection Information" on page 103](#).

## How to Create an Integration User Account

This integration requires an administrator user account for UCMDB to connect to Service Manager. The user account must already exist in both UCMDB and Service Manager.

**Note:** The integration user account must have the **RESTful API** capability word in Service Manager, if the integration uses the Service Manager Enhanced Generic Adapter, which requires the SM ucmdbIntegration RESTful APIs to work.

To create a dedicated integration user account in Service Manager:

1. Log in to Service Manager as a system administrator.
2. Type `contacts` in the Service Manager command line, and press ENTER.
3. Create a new contact record for the integration user account.
  - a. In the Contact Name field, type a name. For example, `UCMDB`.
  - b. Click **Add**, and then **OK**.
4. Type `operator` in the Service Manager command line, and press ENTER.
5. In the Login Name field, type the username of an existing system administrator account, and click **Search**.

The system administrator account is displayed.
6. Create a new user account based on the existing one.
  - a. Change the Login Name to the integration account name that you want (for example, `ucmdb`).
  - b. Type a Full Name. For example, `UCMDB`.
  - c. In the Contact ID field, click the **Fill** icon and select the contact record that you have just created.
  - d. Click **Add**.
  - e. Select the **Security** tab, and change the password.
  - f. Select the **Startup** tab, and add the **RESTful API** capability word for the operator.
  - g. Click **OK**.

The integration user account is created. Later you will need to add this user account (username/password) in UCMDB, and then specify this user account in the Credentials ID field when creating an integration point in UCMDB. See ["How to Create an Integration Point in UCMDB" on the next page](#).

## How to Add the UCMDB Connection Information

SM requires the UCMDB connection information to obtain CI attribute information from the UCMDB system, and display it in the Actual State section in the Service Manager configuration item form.

**Caution:** If you do not specify the correct connection information, an error, instead of UCMDB CI information, will be displayed in the Actual State section.

To add UCMDB connection information in Service Manager:

1. Log in to Service Manager as a system administrator.
2. Click **System Administration > Base System Configuration > Miscellaneous > System Information Record**.
3. Click the **Active Integrations** tab.
4. Select the **HP Universal CMDB** option.

5. In the **UCMDB webservice URL** field, type the URL to the HP Universal CMDB web service API. The URL has the following format:

`http://<UCMDB server name>:<port>/axis2/services/ucmdbSMService`

Replace *<UCMDB server name>* with the host name of your UCMDB server, and replace *<port>* with the communications port that your UCMDB server uses.

6. In the **UserId** and **Password** fields, type the user credentials that are required to manage CIs on the UCMDB system. For example, the out-of-the-box administrator credentials are **admin/admin**.

7. Optionally, if you want to enable an integration to UCMDB Browser, in the **UCMDB Browser URL** field, type your UCMDB Browser URL in the following format:

`http://<UCMDB browser server name>:<port>/ucmdb-browser`

For example: `http://myucmdbbrowserserver:8081/ucmdb-browser`

**Note:** If you specify the UCMDB Browser URL here, the **View in UCMDB Browser** button will replace the **View in UCMDB** button in CI records synchronized from UCMDB; only when you leave this field empty, the **View in UCMDB** button will appear.

8. Click **Save**. Service Manager displays the message: Information record updated.
9. Log out of the Service Manager system.
10. Log back into the Service Manager system with an administrator account.

The Actual State section and the **View in UCMDB Browser** or **View in UCMDB** button will be available in CI records pushed from UCMDB.

## HP Universal CMDB Setup

You must complete the following task from your UCMDB system to support the integration:


Create an integration point between UCMDB and Service Manager. See "[How to Create an Integration Point in UCMDB](#)" below.


### How to Create an Integration Point in UCMDB

A default UCMDB installation already includes the ServiceManagerEnhancedAdapter9-x package. To use the integration package, you must create an integration point that lists the connection properties for the integration.

**Caution:** For data population, this integration supports the use of only one probe for your Service Manager system. In other words, you cannot run population jobs on different probes by setting up multiple integration points with different probes for your Service Manager system. Only one probe is allowed for one Service Manager system.

To create an integration point, follow these steps:

1. Log in to UCMDB as an administrator.
2. Add the integration user account that you created in Service Manager.
  - a. Click **Administration > Users and Roles**.
  - b. Click the **Add New User** icon .

- c. For User Name and Password, type the user name and password that you created in Service Manager. See ["How to Create an Integration User Account" on page 102.](#)
  - d. Click **Next**, and then select **Admin** from the Role List .
  - e. Click **Finish**. The integration user account is added.
3. Navigate to **Data Flow Management > Integration Studio**. UCMDB displays a list of existing integration points.
  4. Click the **New Integration Point** icon . UCMDB displays a New Integration Point properties window.
  5. Complete the integration and adapter property fields as described in the following table.

Field name	Is required?	Description
Integration Name	Yes	Type the name (unique key) of the integration point. For example, <b>sm_integration</b> .
Integration Description	No	Type a description for the integration point.
Adapter	Yes	Select <b>HP Software Products &gt; Service Manager &gt; ServiceManagerEnhancedAdapter9.x</b> .
Is Integration Activated	Yes	Enable this option to indicate the integration point is active.
Hostname/IP	Yes	Type the hostname or IP address of the Service Manager server. For example, localhost.
Port	Yes	Type the communications port of the Service Manager server. For example, 13080.

, continued

Field name	Is required?	Description
URL Override	No	<p>This field value (if any) supersedes the Hostname/IP and Port settings described above.</p> <p>Use this field If you want UCMDB to connect to Service Manager in any combinations of the following ways:</p> <ul style="list-style-type: none"> <li>■ Connect to Service Manager over HTTPS or over both HTTP and HTTPS</li> <li>■ Connect to multiple Service Manager server nodes (horizontally scaled environment)</li> <li>■ Connect to one single Service Manager server node through multiple ports (vertically scaled environment)</li> </ul> <p>For more information, see <a href="#">"Push in Clustered Environments" on page 163</a>.</p> <p>Type one or more Service Manager web services URLs (separated by a semicolon) in this field.</p> <p>The following are two example values of this field (each URL must use this format: <code>http(s)://&lt;hostname&gt;:&lt;port&gt;/SM/9/rest</code>):</p> <ul style="list-style-type: none"> <li>■ <code>https://localhost:13443/SM/9/rest</code></li> <li>■ <code>http://localhost:13080/SM/9/rest;</code> <code>https://localhost:13443/SM/9/rest;</code> <code>http://smfpe04:13080/SM/9/rest</code></li> </ul>
Credentials ID	Yes	<p>Click <b>Generic Protocol</b>, click the <b>Add</b> icon to add the integration user account that you created, and then select it. This account must exist in both Service Manager and UCMDB. See <a href="#">"How to Create an Integration User Account" on page 102</a>.</p>
Data Flow Probe	Yes	<p>Select the name of the Data Flow Probe used to run population jobs. You should have already added the data flow probe for the integration after installing UCMDB. See <a href="#">"Integration Requirements" on page 88</a>.</p>

6. Click **Test Connection** to make sure that a successful connection has been established.

7. Click **OK**.

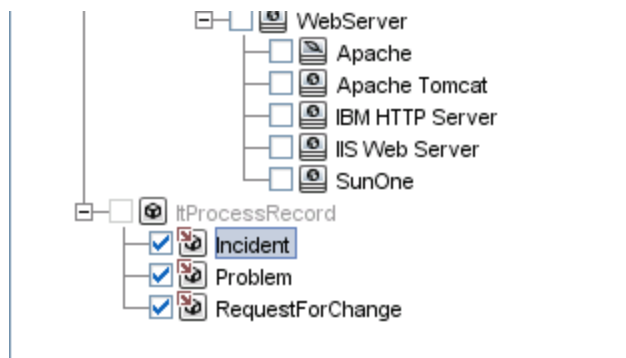
The integration point is created and its details are displayed.

8. Click the **Federation** tab, and complete the following configuration.

- a. In Supported and Selected CI Types, select the following CI types as needed from

**ItProcessRecord:**

- i. Incident
- ii. Problem
- iii. Request for Change



- b. For each CI type you selected (Incident, Problem, or RequestForChange), select **Retrieve CIs of selected CI Type** for **CI Type Retrieval Mode**.

9. Click the **Population** and **Data Push** tabs to view the default integration job details.

**Note:** UCMDB creates several default population and data push jobs when creating an integration point. If needed, you can create a new job for the integration point. For information about creating integration jobs, see ["How to Define Data Push Jobs in UCMDB" on page 113](#) and ["How to Define Population Jobs in UCMDB" on page 109](#).

10. Click the **Save Integration Point** icon .

## Centralized CI Management

The Service Manager Enhanced Generic Adapter allows centralized CI type management. Centralized CI management enables you to manage CIs from UCMDB without the need to do much tailoring work on the SM side. You can then synchronize CIs from UCMDB to SM through data push.

## Managing Out-Of-the-Box CI Types

SM and UCMDB provide out-of-the-box CI types, as well as mapping files for them. To modify an out-of-the-box CI type (for example, adding new attributes according to your business needs), you only need to open the Visual Mapping tool, manually add new attributes to this CI type, and then apply the update without leaving UCMDB. The new attribute will be automatically added to the relevant web service object on the SM side.

## Managing New Custom CI Types

When you create a new CI type in UCMDB, all relevant SM objects (DBDICT, JoinDef, Web Service API, DEM Rule, and so on) for the new CI type are automatically created on the SM side, except the format (the new CI type will use **configurationItem** as the default format).

To add a new CI type according to your business needs, you only need to add a new CI type and create a TQL query in UCMDB, and then create a matching SM CI type by using the Visual Mapping tool without the need to leave UCMDB.

**Note:** The population feature enables you to synchronize CIs from SM to UCMDB. For information about the circumstances under which you need to use the population feature, see ["When Do I Need the Population Feature?"](#) on page 176.

## Visual Mapping Tool

The Service Manager Enhanced Generic Adapter comes with a Visual Mapping tool, which provides a graphic user interface for you to configure value and field mappings for complex data push or population configurations.

The following table describes the panes of this graphic user interface.

Pane	Description
Local Query	Displays a hierarchical tree structure of the local TQL query in the CMDB.
Local Query Attributes	This is the Attributes pane for the Local Query pane. It displays attributes of a local integration TQL query.
Visual Mapping	Allows you to establish mapping for items you select from the Local Query pane and the External Class Model pane through a drag and drop.
XML Editor	Allows to you edit XML mapping files in a text editor.



Pane	Description
External Class Model	Displays a hierarchical tree structure of the external class model. For the UCMDB-SM integration, this pane displays supported CI types from the SM side.
External Class Model Attributes	The Attributes pane for the External Class Model pane displays attributes of an external class model. For the UCMDB-SM integration, this pane displays attributes from the SM side.

## Populating UCMDB with Service Manager CI Data

In addition to pushing CI data from UCMDB to Service Manager, this integration also supports the population of CI data (including CIs and CI relationships) from Service Manager to UCMDB. The integration can then update the list of CIs in UCMDB if new CIs or new attribute values are found in Service Manager. The population of data from Service Manager to UCMDB is defined in the Integration Studio in UCMDB. You can manually run the population jobs, however HP recommends that you schedule these jobs to keep your CIs and CI attributes up to date.

This task includes the following steps:

1. Define CI/CI Relationship population jobs in UCMDB.  
See ["How to Define Population Jobs in UCMDB"](#) below.
2. View the transferred CI/CI Relationship data in UCMDB.  
See ["View Service Manager CI Data in UCMDB"](#) on page 112.
3. Schedule CI population jobs to keep CIs and CI attributes up to date.  
See ["How to Schedule CI Population Jobs"](#) on page 112.

## How to Define Population Jobs in UCMDB

A CI or relationship population job copies certain types of CIs or relationships from Service Manager to UCMDB.

To define a CI or CI relationship population job, follow these steps:

1. Log in to UCMDB as an administrator.
2. Navigate to **Data Flow Management > Integration Studio**. UCMDB displays a list of existing integration points.
3. Open the integration point that you created for Service Manager.

4. Click the **Population** tab, and add a new job as follows.



**Note:** UCMDB creates several default population and data push jobs when creating an integration point. The following table lists the default population jobs and their queries. If needed, you can create, update or remove queries for each job.



**Queries for CI / CI relationship population**

Integration Job	Queries for CI / CI relationship population
SM Configuration Item Population job	<p>Out-of-the-box, the following queries are available for this job, which populates UCMDB with CI records from Service Manager:</p> <ul style="list-style-type: none"> <li>■ <b>SM Business Service Population 2.0:</b> Populates UCMDB with CIs of the bizservice type.</li> <li>■ <b>SM RunningSoftware Population 2.0:</b> Populates UCMDB with CIs of the RunningSoftware type.</li> <li>■ <b>SM Computer Population:</b> Populates UCMDB with CIs of the computer type.</li> <li>■ <b>SM CLIP Down Time Population 2.0:</b> Populates UCMDB with CIs of the planned outage type.</li> </ul>

**Queries for CI / CI relationship population, continued**

Integration Job	Queries for CI / CI relationship population
SM Relations Population job	<p>Out-of-the-box, the following queries are defined for this job, which populates UCMDB with CI Relationship records from Service Manager:</p> <ul style="list-style-type: none"> <li>■ <b>SM Biz To Biz With Containment 2.0:</b> Populates UCMDB with CI relationships in which a bizservice CI contains another.</li> <li>■ <b>SM Biz To Biz With Usage 2.0:</b> Populates UCMDB with CI relationships in which a bizservice CI uses another.</li> <li>■ <b>SM Biz To Computer With Containment 2.0:</b> Populates UCMDB with CI relationships in which a bizservice CI contains a computer CI.</li> <li>■ <b>SM Biz To Computer With Usage 2.0:</b> Populates UCMDB with CI relationships in which a bizservice CI uses a computer CI.</li> <li>■ <b>SM Computer To Computer With Connects 2.0:</b> Populates UCMDB with CI relationships in which a computer CI connects to another.</li> <li>■ <b>SM Biz To Software With Containment 2.0:</b> Populates UCMDB with CI relationships in which a bizservice CI contains a software CI.</li> <li>■ <b>SM Biz To Software With Usage 2.0:</b> Populates UCMDB with CI relationships in which a bizservice CI uses a software CI.</li> <li>■ <b>SM Computer Composition Software 2.0:</b> Populates UCMDB with CI relationships in which a computer CI connects to a software CI.</li> <li>■ <b>SM CI Connection Down Time CI 2.0:</b> Populates UCMDB with CI relationships in which a CI connects to a down time CI.</li> </ul>

- a. Click the **New Integration Job** icon .
- b. Type a Name for the integration job. For example, **CI\_Population\_Job1**.
- c. Click the **Add Query** icon  to add existing queries to the job (see the table above).
- d. Select the **Allow Integration Job to delete removed data** check box for the query.
- e. Click **OK** to save the job.

5. Run the job manually to see if the integration job works properly.
  - a. To populate UCMDB with all relevant data for the job, click the  icon.
  - b. To populate UCMDB with only CI data changes since the job last ran, click the  icon.
6. Wait for the job to complete, and click the **Refresh** icon multiple times as needed until the job is completed.

**Note:** When the job is completed, the job status becomes one of the following: Succeeded, Passed with failures, or Failed.

7. Click the **Statistics** tab to view the results. If the job failed, click the **Query Status** tab and **Job Errors** tab for more information. For details, see ["Troubleshooting Population Issues" on page 267](#).
8. Click **OK**.

If the job is completed successfully, you can view the transferred CI data in UCMDB and schedule the job so that it can run automatically.

## View Service Manager CI Data in UCMDB

After a population job is successfully completed, you can search for the Service Manager CI records in UCMDB, and verify that their attributes are correctly populated.

The Service Manager **CI Identifier** field is populated to the **Name** field on the Configuration Item Properties pane in UCMDB.

**Note:** To see the entire attribute mappings of a CI type, you can open its population XML file (for example, **SM Business Service Population.xml**), where the UCMDB attribute field names and the mapped Service Manager web service field caption names are defined. For more information, see ["Tailoring the Integration" on page 182](#).

## How to Schedule CI Population Jobs

You can schedule CI population jobs to match the discovery/maintenance schedule of your Service Manager feeders. For example, if your Service Manager feeders send CI data updates on a daily schedule, then the population jobs must also run on a daily schedule. By using a matching schedule you can ensure that your UCMDB system always has the most current CI data.

This task includes the following steps:

1. Log in to UCMDB as an administrator.
2. Navigate to **Data Flow Management > Integration Studio**. UCMDB displays a list of integration points.
3. Open the integration point for SM.
4. Click the **Population** tab, and select a population job from the list.
5. Click the **Edit Integration Job** icon.
6. Select the **Scheduler enabled** option.
7. Select the scheduling options that you want to use. For example, select Repeat every: **Day** and Ends: **Never**.
8. Select a Time Zone.
9. Click **OK**.

## Pushing UCMDB CI Data to Service Manager

The integration requires a one-time transfer of CIs from UCMDB to Service Manager to populate the Service Manager system with CIs. The integration will then update the list of CIs in Service Manager when UCMDB discovers new CIs or new attribute values. The integration accomplishes the push of CI data using data push jobs in the UCMDB system. HP recommends that you schedule these jobs to keep your CIs and CI attributes up to date.


This task includes the following steps:

1. Define CI/CI Relationship data push jobs.  
See ["How to Define Data Push Jobs in UCMDB"](#) below.
2. View the CI/CI Relationship data pushed from UCMDB.  
See ["How to View UCMDB CI Data in Service Manager"](#) on page 118.
3. Schedule data push jobs to keep CI/CI Relationship data up to date.  
See ["How to Schedule Data Push Jobs"](#) on page 117.

## How to Define Data Push Jobs in UCMDB

Data push jobs copy CI or CI Relationship records from your UCMDB system to your Service Manager system.

To define a CI or CI relationship push job, follow these steps:

1. Log in to UCMDB as an administrator.
2. Navigate to **Data Flow Management > Integration Studio**. UCMDB displays a list of existing integration points.
3. Select the Integration Point that you created for Service Manager. For example, **sm\_integration**.
4. Click the **Data Push** tab.
5. Follow these steps to add a new data push job:
  - a. Click the **New Integration Job** icon .

**Note:** UCMDB creates a default data push job when creating an integration point. The following table lists the default data push job and its queries. If needed, you can create, update, or remove queries for the push job. To access these out-of-the-box queries for push, go to **Modeling > Modeling Studio > Resources**, select **Queries** for Resource Type, and then navigate to **Root > Integration > Push**. For information about tailoring data push queries, see ["How to Create a Query to Synchronize the CI Type" on page 228](#).

**Queries for CI / CI Relationship Push**


Integration job	Queries
Integration job	Queries

**Queries for CI / CI Relationship Push , continued**



<b>Integration job</b>	<b>Queries</b>
SM Push job	<p data-bbox="451 405 1338 470">Out-of-the-box, the following queries are available for this job, which pushes CI records from UCMDB to Service Manager:</p> <ul style="list-style-type: none"> <li data-bbox="451 510 1159 541">■ <b>SM Mainframe Push 2.0:</b> pushes CIs of the mainframe type.</li> <li data-bbox="451 577 1317 642">■ <b>SM Business Element Push 2.0:</b> pushes CIs of the business application, and infrastructure service types</li> <li data-bbox="451 678 1370 709">■ <b>SM Network Component Push 2.0:</b> pushes CIs of the network component type.</li> <li data-bbox="451 745 1313 777">■ <b>SM Running Software Push 2.0:</b> pushes CIs of the running software type.</li> <li data-bbox="451 812 1286 844">■ <b>SM Business Service Push 2.0:</b> pushes CIs of the business service type.</li> <li data-bbox="451 879 1128 911">■ <b>SM Computer Push 2.0:</b> pushes CIs of the computer type.</li> <li data-bbox="451 947 1089 978">■ <b>SM Storage Push 2.0:</b> pushes CIs of the storage type.</li> <li data-bbox="451 1014 1057 1045">■ <b>SM Switch Push 2.0:</b> pushes CIs of the switch type.</li> <li data-bbox="451 1081 1156 1113">■ <b>SM Net Printer Push 2.0:</b> pushes CIs of the net printer type.</li> <li data-bbox="451 1148 1065 1180">■ <b>SM Cluster Push 2.0:</b> pushes CIs of the cluster type.</li> <li data-bbox="451 1215 1224 1247">■ <b>SM Mobile Device Push 2.0:</b> pushes CIs of the mobile device type.</li> <li data-bbox="451 1283 1190 1314">■ <b>SM Local Printer Push 2.0:</b> pushes CIs of the local printer type.</li> </ul> <p data-bbox="451 1356 1338 1421">Out-of-the-box, the following queries are available for this job, which pushes CI Relationship records from UCMDB to Service Manager:</p> <ul style="list-style-type: none"> <li data-bbox="451 1461 1338 1526">■ <b>SM Layer2 Topology Relations Push 2.0:</b> pushes compound CI relationships between nodes.</li> <li data-bbox="451 1562 1295 1627">■ <b>SM Business Service Relations Push 2.0:</b> pushes CI relationships whose upstream CI type is business service.</li> <li data-bbox="451 1663 1370 1728">■ <b>SM CRG Relations Push 2.0:</b> pushes CI relationships whose upstream CI type is cluster.</li> <li data-bbox="451 1764 1370 1795">■ <b>SM Node Relations Push 2.0:</b> pushes direct CI relationships whose upstream CI</li> </ul>

**Queries for CI / CI Relationship Push , continued**

Integration job	Queries
	type is node.

- b. In the **Name** field, type a unique name for the job. For example, **CI\_Push\_Job1**.
  - c. Click the **Add Query** icon  to add existing queries to the job.
  - d. Select the **Allow Integration Job to delete removed data** option for each query.
  - e. Click **OK** to save the job.
6. Run the job manually to verify that the integration job works properly.

**Caution:** If you have a huge amount of CI data in your UCMDB system, and this is your first time to push CI /CI Relationship data to Service Manager, it is recommended to select the “Add the record” option instead of “Open a change” or “Open an incident” for “Action if matching record does not exist” in each Discovery Event Manager Rules definition. Failure to do so may cause unnecessary performance problems. For details, see ["How to Add Discovery Event Manager Rules" on page 205](#).

- a. To push all relevant data for the job, click the  icon.
- b. To push only data that was changed since the job last ran, click the  icon.

**Tip:** You can stop a running push job by pressing the **Stops the selected job** icon .

7. Wait for the job to complete, and click the **Refresh** icon multiple times as needed until the job is completed.

**Note:** When the job is completed, the job status becomes one of the following depending on the results: Completed successfully, Completed, or Failed.

8. Click the **Statistics** tab to view the results; if any errors occur click the **Query Status** tab and **Job Errors** tab for more information. For details, see ["Troubleshooting Data Push Issues" on page 257](#).
9. Click **OK**.




If the job is completed successfully, you can view the UCMDB CI data in Service Manager, and schedule the job so that it can run automatically.

## How to Schedule Data Push Jobs

It is a best practice to schedule the data push jobs to match the discovery schedule of your Service Manager feeders. For example, if your Service Manager feeders send CI data updates on a daily schedule, the data push jobs must also run on a daily schedule. By using a matching schedule you can ensure that your Service Manager system always has the most current CI data.

UCMDB allows you to schedule updates directly from a data push job. This task includes the following steps:

1. Log in to UCMDB as an administrator.
2. Navigate to **Data Flow Management > Integration Studio**. UCMDB displays a list of integration points.
3. Select the integration point that you created for Service Manager. For example, **SM Integration**.
4. Click the **Data Push** tab.
5. Select a push job. For example, **SM Configuration Item Push Job 2.0**.
6. Click the **Edit Integration Job** icon .

**Tip:** UCMDB allows you to define two different schedules for two types of data push: Delta Sync, and Full Sync. For recommendations on push scheduling, see "[Push Scheduling Recommendations](#)" on page 162.

7. Define a schedule for Delta Sync.
  - a. Click the **Delta Synchronization** tab.
  - b. Select the **Scheduler enabled** option.

- c. Select the scheduling options that you want to use.

**Scheduler Definition**

Delta Synchronization | Full Synchronization


Scheduler enabled

Repeat: Once  
Interval  
Day of Month  
Weekly  
Monthly  
Yearly  
Cron

Starts: 1/11/15 20:14

Ends:  Never  Until 1/11/15

Repeat on:  Sun  Mon  Tue  Wed  Thu  Fri  Sat

Time Zone: Data Flow Probe Time Zone  Server Time: 1/11/15 8:16 PM

- 8. Click the **Full Synchronization** tab, and select the scheduling options that you want to use.
- 9. Click **OK** to save the data push job.
- 10. Repeat [step 6](#) to [step 9](#) for the rest of data push jobs of the integration point.
- 11. Save the integration point.

## How to View UCMDB CI Data in Service Manager

After a push job is successfully completed, you can search for and verify the pushed CI/CI relationship data in Service Manager.

CI records pushed from UCMDB contains a **View in UCMDB** or **View in UCMDB Browser** button, which enables you to access UCMDB or UCMDB Browser to view the CI information.

**Note:**

- If you specified the UCMDB Browser URL in the System Information Record in SM, the **View in UCMDB Browser** button is displayed; otherwise the **View in UCMDB** button is displayed.
- The UCMDB Browser is a lightweight UI designed for simple access to UCMDB configuration information. This is a tool for searching, locating and consuming configuration related data. It is an optional add-on to UCMDB. For more information, refer to the UCMDB Browser documentation.

To view UCMDB CI data in Service Manager:

1. Log in to Service Manager as a system administrator.
2. Navigate to **Configuration Management > Search CIs**.
3. Open a CI record pushed from UCMDB.
4. If the **View in UCMDB** button is available, view the CI record in UCMDB.
  - a. Click the **View in UCMDB** button. The UCMDB login screen opens.
  - b. Type a UCMDB username and password to log in.

The CI record opens in UCMDB. You can view its properties.

**Note:** You can enable Lightweight Single Sign-On (LW-SSO) for the integration so that Service Manager web client users can bypass the UCMDB login screen. For details, see ["How to Enable Lightweight Single Sign-On \(LW-SSO\) Configuration" on page 172](#).

5. If the **View in UCMDB Browser** button is available, view the CI record in the UCMDB Browser.
  - a. Click the **View in UCMDB Browser** button. The UCMDB Browser login screen opens.
  - b. Type a UCMDB Browser username and password to log in. The CI record opens in the UCMDB Browser. You can view its properties and other information.
6. Open the **Actual State** section.

Service Manager sends a web services request to UCMDB and displays all CI attributes that the request returns.

**Note:** The web services request uses the UCMDB webservice URL and account (for example, admin/admin) defined in the System Information Record in Service Manager. See ["How to Add the UCMDB Connection Information" on page 103](#).

## How to View the Change History of the Primary CI of a Problem Record

When integrated with UCMDB Browser, Service Manager displays a **Primary CI History in UCMDB** section in a problem record whose primary CI is synchronized from UCMDB. You can view the CI changes on that primary CI for root cause investigation.

For information about how to enable an integration with UCMDB Browser, see ["How to Add the UCMDB Connection Information" on page 103](#).

To view the primary CI change history, follow these steps:

1. Log in to Service Manager.
2. Navigate to **Problem Management**, and perform a search to open a problem record whose Primary CI is synchronized from UCMDB.
3. Click the **Primary CI History in UCMDB** tab.

## Federating Service Manager Ticket Data to UCMDB

Federation does not physically copy data from Service Manager (SM) to UCMDB; it only retrieves SM data for displaying in UCMDB. Out-of-the-box, the UCMDB-SM integration supports federation for the Incident, Problem, and RequestForChange external CI types in UCMDB. If you have enabled these CI types for federation when creating your integration point, in UCMDB you can retrieve Incident, Problem, and Change record data from SM.

Out-of-the-box, each of the CI types (Incident, Problem, or Change) supports federation of a subset of their attributes in Service Manager; however, the integration can federate more attributes if tailored as such.

This section includes:

- ["Federation Queries" below](#)
- ["Examples of Using Federation" below](#)
- ["How to Add an Attribute of a Supported CI Type for Federation" on page 251](#)

### Federation Queries

Federation uses queries to determine what data to retrieve from Service Manager. To retrieve specific ticket data from Service Manager, you need to create a TQL query first.

The ServiceManagerEnhancedGenericAdapter9-x provides a bunch of sample federation queries, which you can use as a reference. They are available from the following path in UCMDB: **Modeling > Modeling Studio > Resources > Integration > Service Manager > Federation**.

### Examples of Using Federation

You can use the federation feature in many different ways. The following are only examples of using the feature.

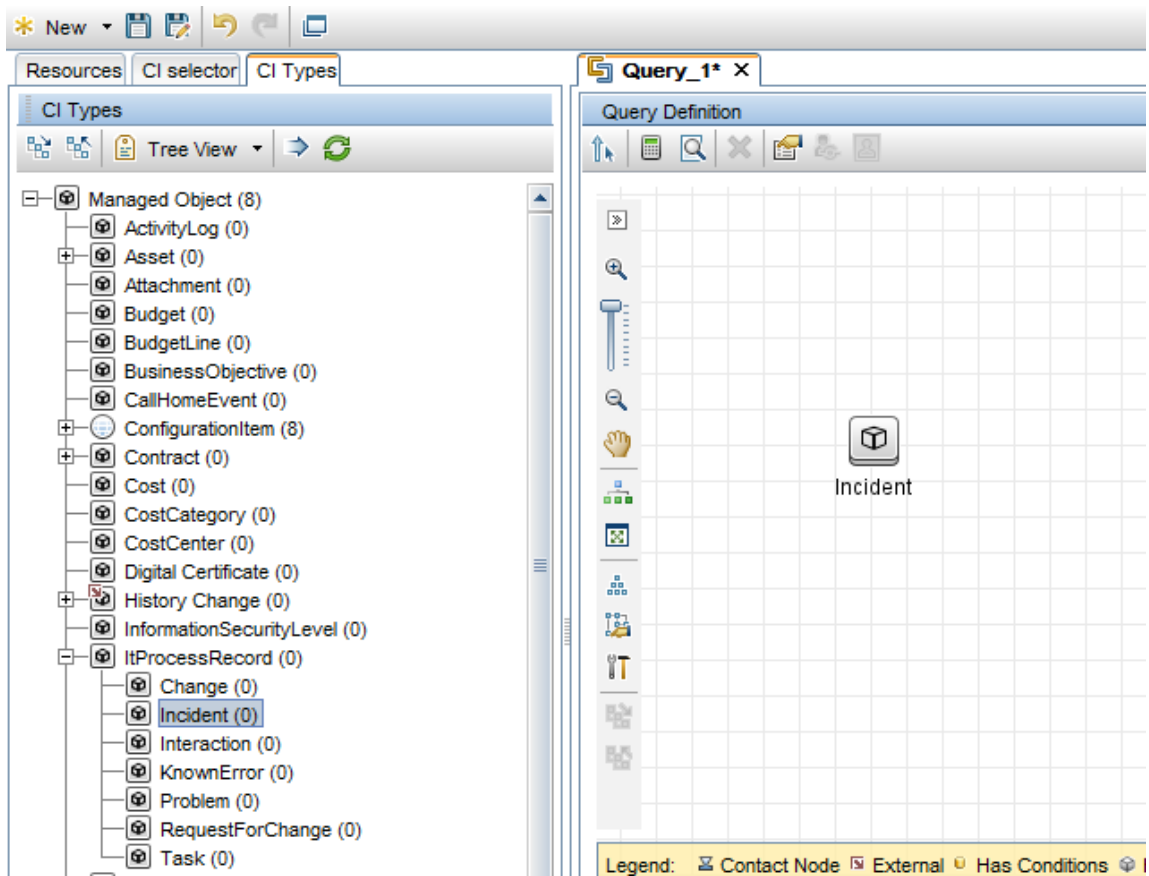
This section describes four examples:

- ["Example 1: Federate All SM Incident Tickets" below](#)
- ["Example 2: Federate SM Incident Records that Affect a UCMDB Business Service CI" on page 125](#)
- ["Example 3: Federate Incident, Change, and Problem Record Data from Service Manager for UCMDB CIs" on page 133](#)
- ["Example 4: Retrieve Service Manager Records Related to a UCMDB CI" on page 136](#)

### Example 1: Federate All SM Incident Tickets

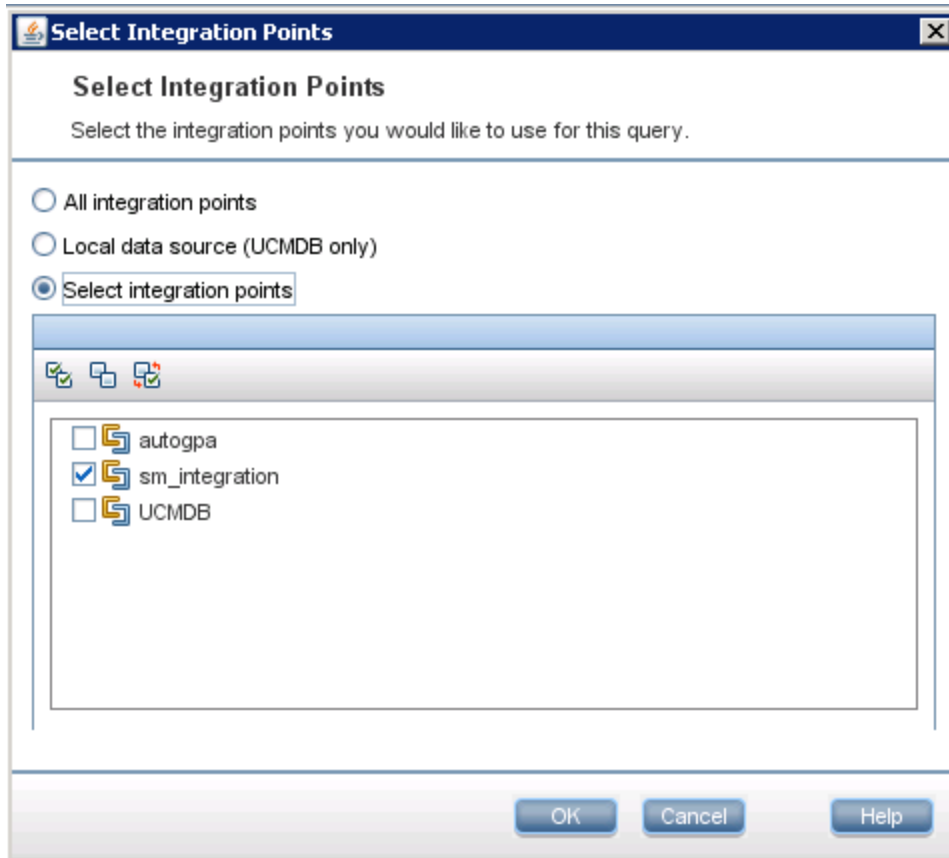
This example illustrates how you retrieve information of all Incident records that exist in Service Manager.



1. Log in to UCMDB as an administrator.
2. Navigate to **Modeling > Modeling Studio > Resources**.
3. For Resource Type, select **Queries** from the list.
4. Click **New > Query**.
5. On the **CI Types** tab, navigate to **ItProcessRecord > Incident**, and drag it to the query pane on the right side.

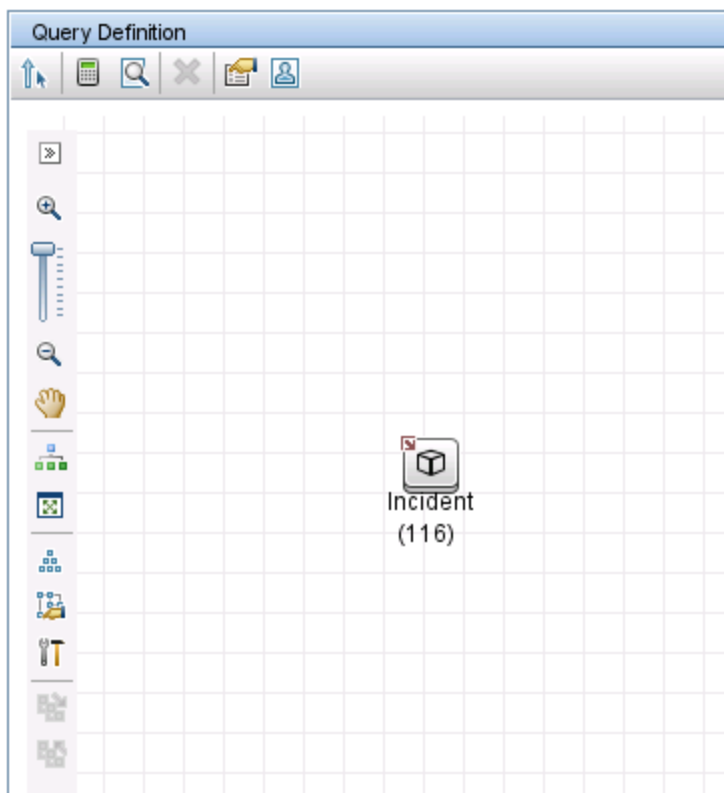


6. Specify Service Manager as the data source for the Incident query node.
  - a. Select the Incident query node, click the **Data Sources** tab on the lower right pane, and then click **Edit**.
  - b. Select the **Select integration points** option, and then select your integration point name (for

example, **sm\_integration**). Click **OK**.



7. Click the **Save** icon , and then type a query name and select a location in which to save the query (for example, select the **Root** folder).
8. Select the Incident query node, and then click the **Calculate Query Result Count** icon . UCMDB returns the query result count. For example, the following figure shows that there are a total of 116 Incident records in Service Manager.



9. Right-click the Incident query node, and select **Show Element Instances**. UCMDB displays a list of all Incident records that exist in Service Manager.



CI Instances


Here you can see all of the CI instances found for the selected query node in the table

Show CI instances of: Incident (116)

Display Label	ReferenceNumber	Create Time	ClosedTime	Category	Priority	Incident!
IM10001	IM10001	Mon Dec 30 2013 03:32 PM IST	Fri Aug 15 2014 03:38 PM IST	failure	3_average	Closed
IM10002	IM10002	Mon Dec 30 2013 03:37 PM IST			3_average	Categorize
IM10003	IM10003	Wed Aug 13 2014 07:33 PM IST			2_high	Categorize
IM10004	IM10004	Thu Aug 14 2014 08:38 PM IST	Fri Aug 15 2014 09:07 PM IST	failure	2_high	Closed
IM10005	IM10005	Wed Jan 15 2014 05:51 PM IST		failure	2_high	Work_In_Pr
IM10006	IM10006	Mon Sep 15 2014 08:55 PM IST	Sat Sep 27 2014 05:56 PM IST	failure	2_high	Closed
IM10007	IM10007	Mon Aug 11 2014 05:46 PM IST	Mon Aug 11 2014 06:05 PM IST	hardware	2_high	Closed
IM10008	IM10008	Thu Sep 25 2014 08:18 PM IST		failure	3_average	Categorize
IM10009	IM10009	Thu Sep 25 2014 08:24 PM IST		failure	3_average	Assign
IM10010	IM10010	Wed Sep 10 2014 06:02 PM IST			3_average	Categorize
IM10012	IM10012	Thu Sep 25 2014 08:09 PM IST			3_average	Categorize
IM10013	IM10013	Thu Oct 30 2014 07:30 PM IST		failure	2_high	Categorize
IM10014	IM10014	Thu Oct 30 2014 07:31 PM IST		failure	1_critical	Categorize
IM10015	IM10015	Thu Oct 30 2014 07:32 PM IST		failure	3_average	Categorize
IM10017	IM10017	Thu Oct 30 2014 08:04 PM IST		access	2_high	Categorize
IM10018	IM10018	Thu Oct 30 2014 08:05 PM IST		access	3_average	Categorize
IM10019	IM10019	Thu Oct 30 2014 08:06 PM IST		access	3_average	Categorize
IM10020	IM10020	Thu Oct 30 2014 08:06 PM IST		performance	3_average	Categorize
IM10030	IM10030	Sun Nov 9 2014 10:48 PM IST		access	3_average	Resolved
IM10031	IM10031	Sun Nov 9 2014 10:52 PM IST		failure	3_average	Pending_Cu

Total rows: 116 Rows per page: 50 1 of 3

OK Cancel Help

10. Select an Incident record from the list, and click the **Properties** icon  to view its details.

Configuration Item Properties

Name: IM ID: ckd%0Apriority%3DSTRING%3D2\_high%0Areference\_number%3DSTRING%3DIM10005%0Aurgency%3DSTRING%3D1\_critical%0A CI Type: In

Quick filter: Type here to filter properties

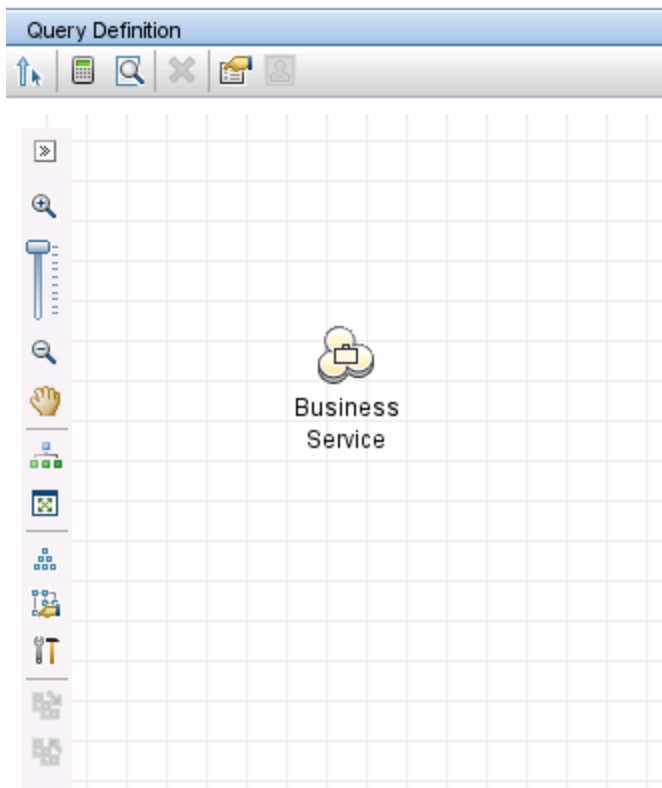
Actual Deletion Period	40
Category	failure
ClosedTime	
Create Time	Wed Jan 15 2014 05:51 PM IST
Deletion Candidate Period	20
Details	[Virus scanner blocks e-mail attachments]
Display Label	IM10005
ImpactScope	user
IncidentStatus	Work_In_Progress
LastModifiedTime	Mon Jan 5 2015 10:52 PM IST
Name	E-mail attachments being blocked
Priority	2_high
ReferenceNumber	IM10005
Urgency	1_critical

### Example 2: Federate SM Incident Records that Affect a UCMDB Business Service CI

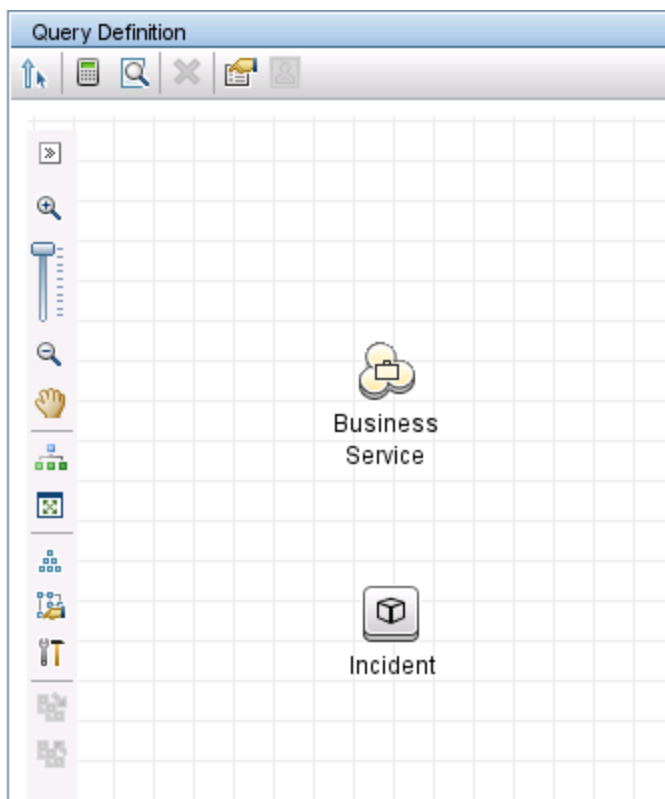
The following example illustrates how to federate a list of Service Manager Incident records whose Affected Service or Affected CI field contains a UCMDB Business Service CI.


In this example, IM10005 in SM has an affected service named **bizservice1**, which was pushed from UCMDB.

1. Log in to UCMDB as an administrator.
2. Navigate to **Modeling > Modeling Studio > Resources**.
3. For Resource Type, select **Queries** from the list.
4. Click **New > Query**.
5. On the **CI Type** tab, navigate to **ConfigurationItem > BusinessElement > Service > BusinessService**, and drag it to the query pane on the right side.

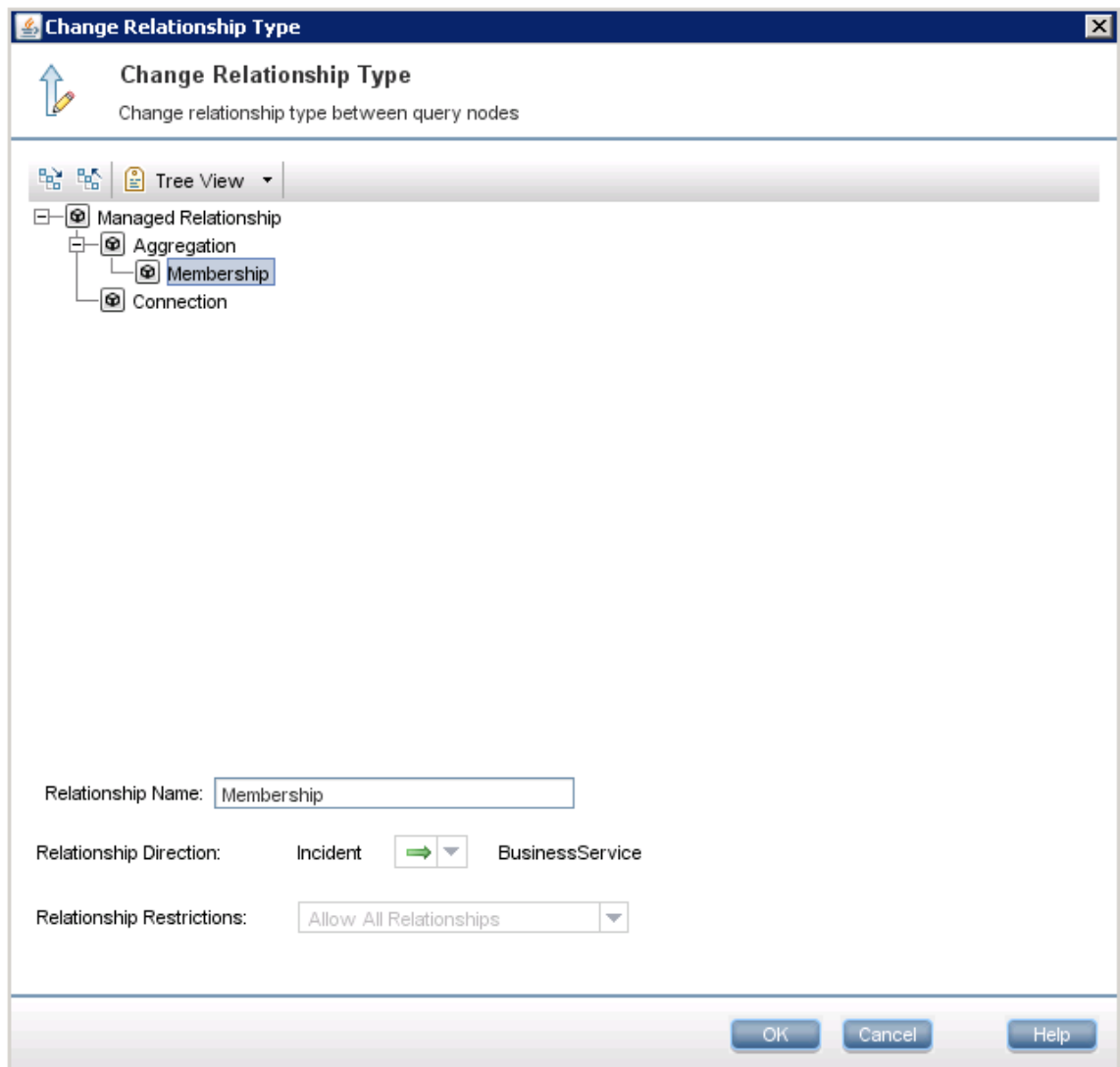


6. Navigate to **ItProcessRecord > Incident**, and drag the icon to the query pane.

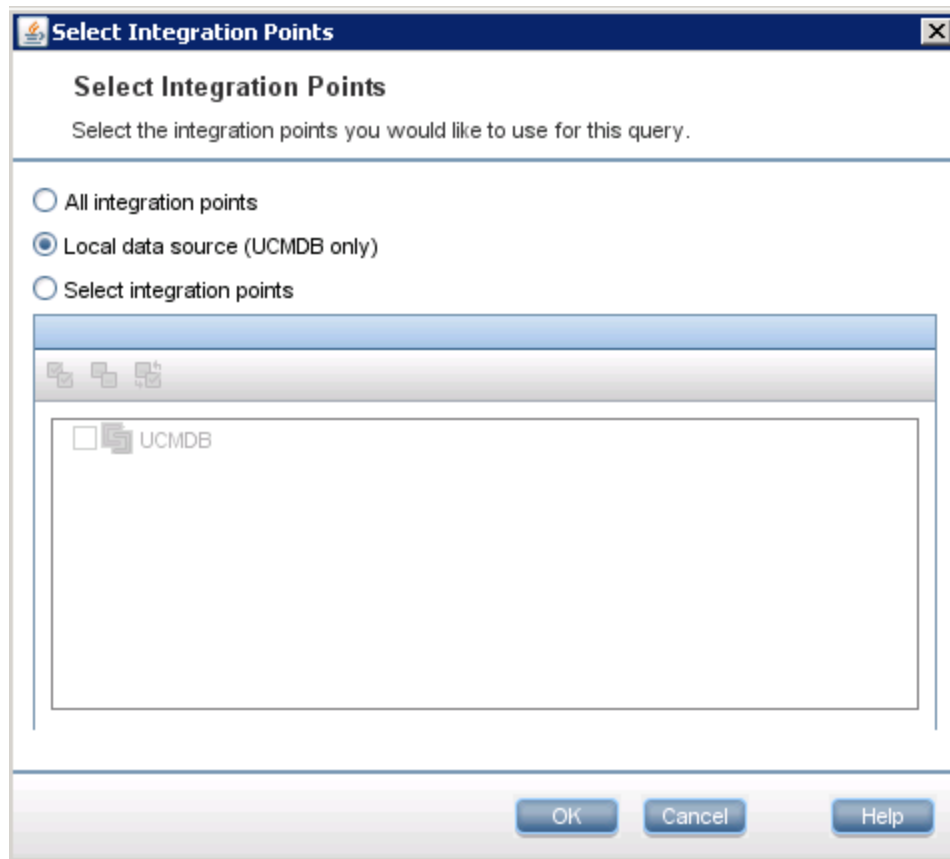


7. Click the **Create Relationship** icon .
8. Select the Incident query node, and drag the arrow from this node to the BusinessService node to create a regular relationship between the nodes.
  - a. Select **Regular Relationship**, and click **OK**.
  - b. Select **Membership**, and optionally enter a relationship name (for example, **Membership**). Click

**OK.**

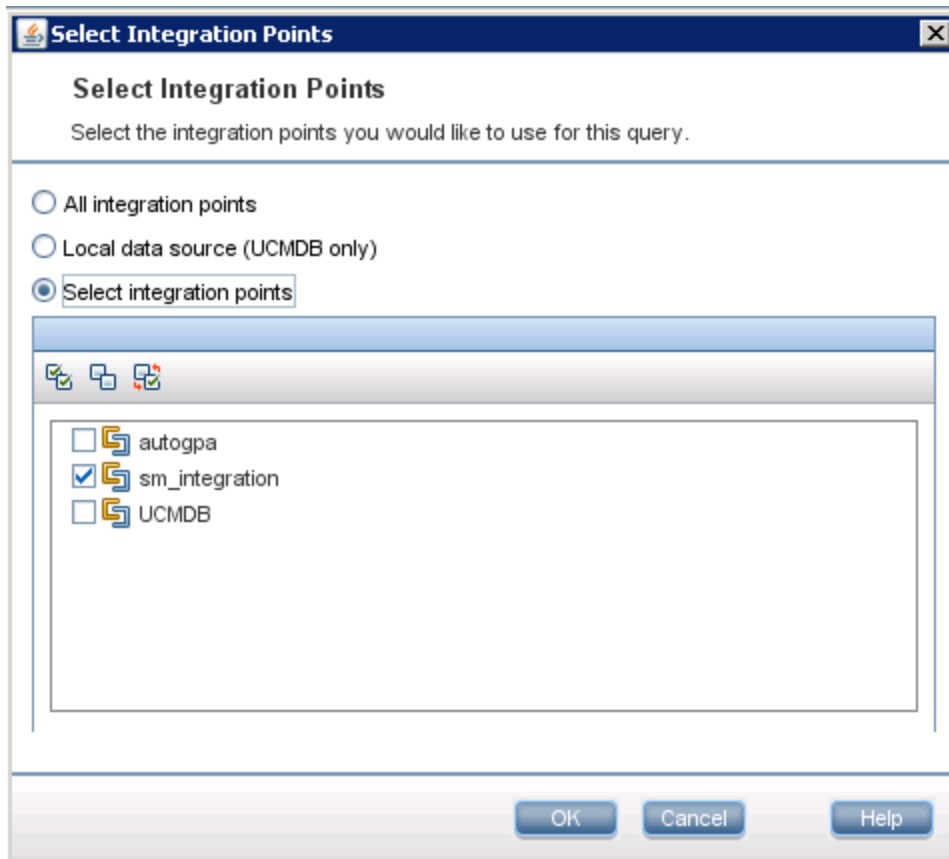



9. Specify UCMDB as the data source for the BusinessService query node. To do this, follow these steps:
  - a. Select the **BusinessService** query node.
  - b. On the lower right pane, click the **Data Sources** tab and then click **Edit**.
  - c. Make sure that the **Local data source (UCMDB only)** option is selected.

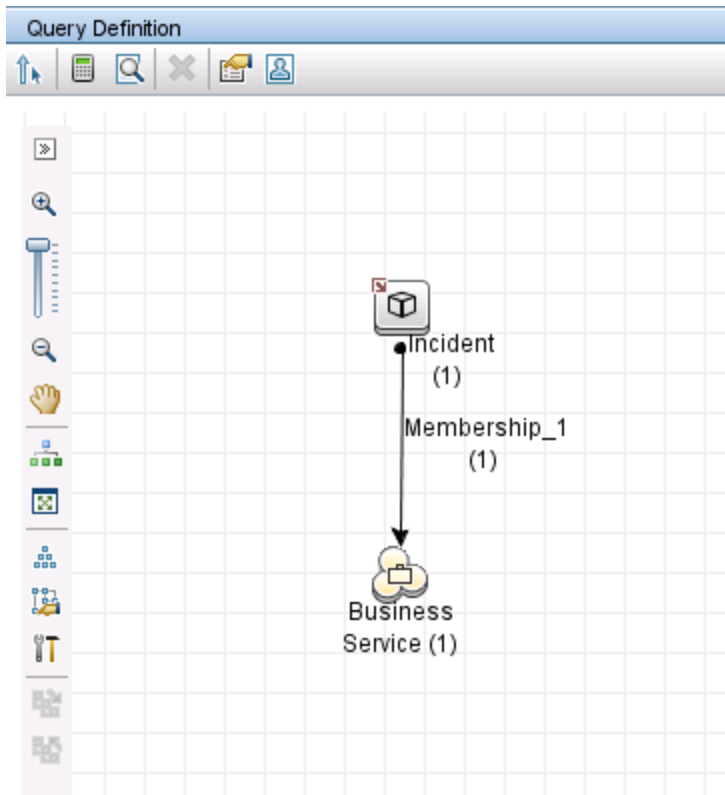


d. Click **OK**.

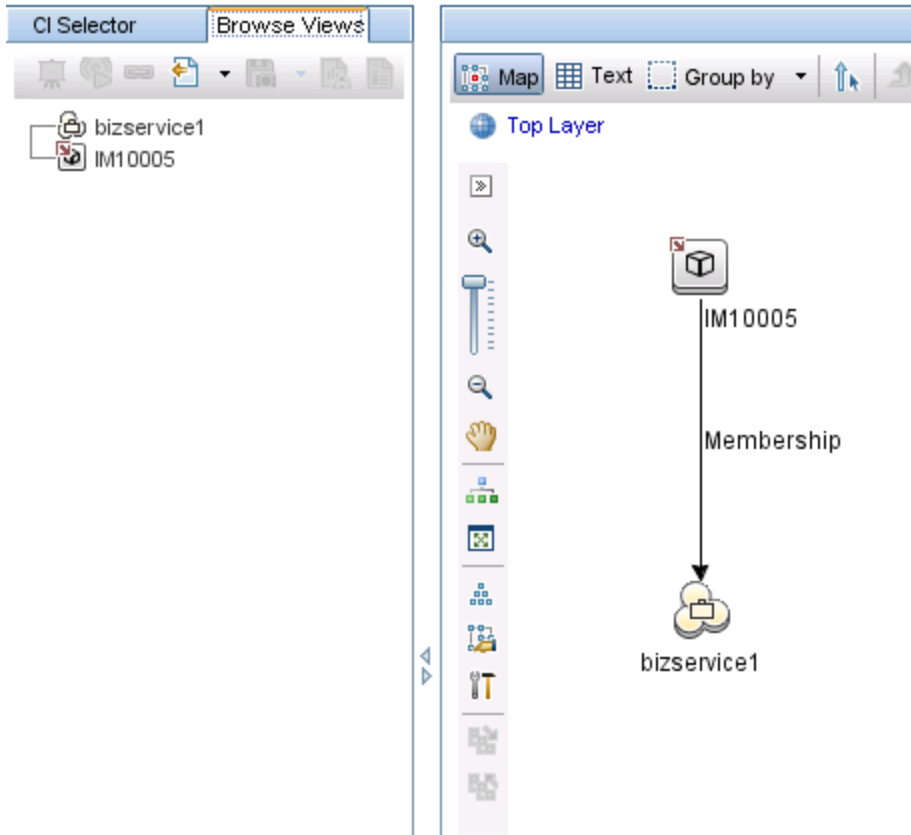
10. Repeat the steps above to specify your integration point as the data source for the Incident query node (for example, **sm\_integration**).




11. Click the **Calculate Query Result Count** icon . The number of SM Incidents and the number of their affected UCMDB Business Service CIs are displayed.



12. Click the **Preview** icon to view the query results .



13. Select each SM Incident record from either the CI Selector pane or the query pane, and click the **Properties** icon to view its details .

**Configuration Item Properties**

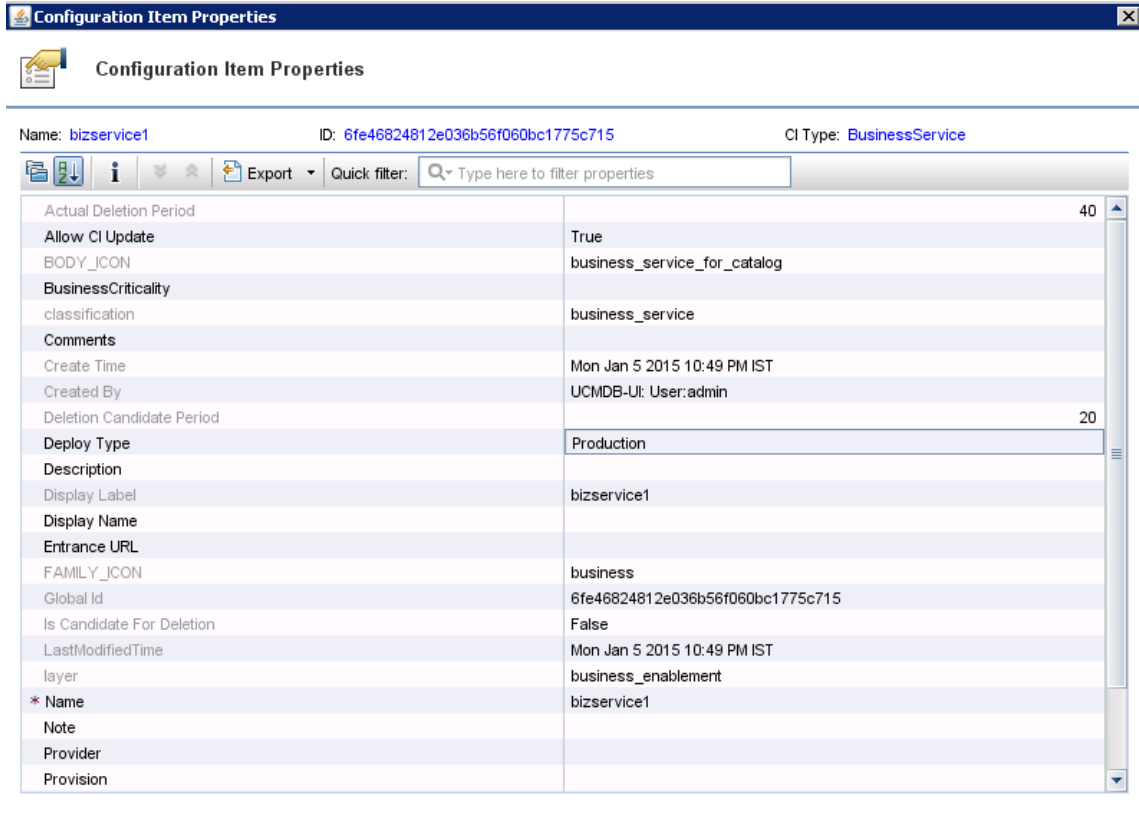
Name: IM ID: cked%0Apriority%3DSTRING%3D2\_high%0Areference\_number%3DSTRING%3DIM10005%0Aurgency%3DSTRING%3D1\_critical%0A CI Type: In

Export Quick filter:

Actual Deletion Period	40
Category	failure
ClosedTime	
Create Time	Wed Jan 15 2014 05:51 PM IST
Deletion Candidate Period	20
Details	
Display Label	IM10005
ImpactScope	user
IncidentStatus	Work_In_Progress
LastModifiedTime	Mon Jan 5 2015 10:52 PM IST
Name	E-mail attachments being blocked
Priority	2_high
ReferenceNumber	IM10005
Urgency	1_critical




14. Select each UCMDB CI record from the CI Selector pane, and click the **Properties** icon to view its details.

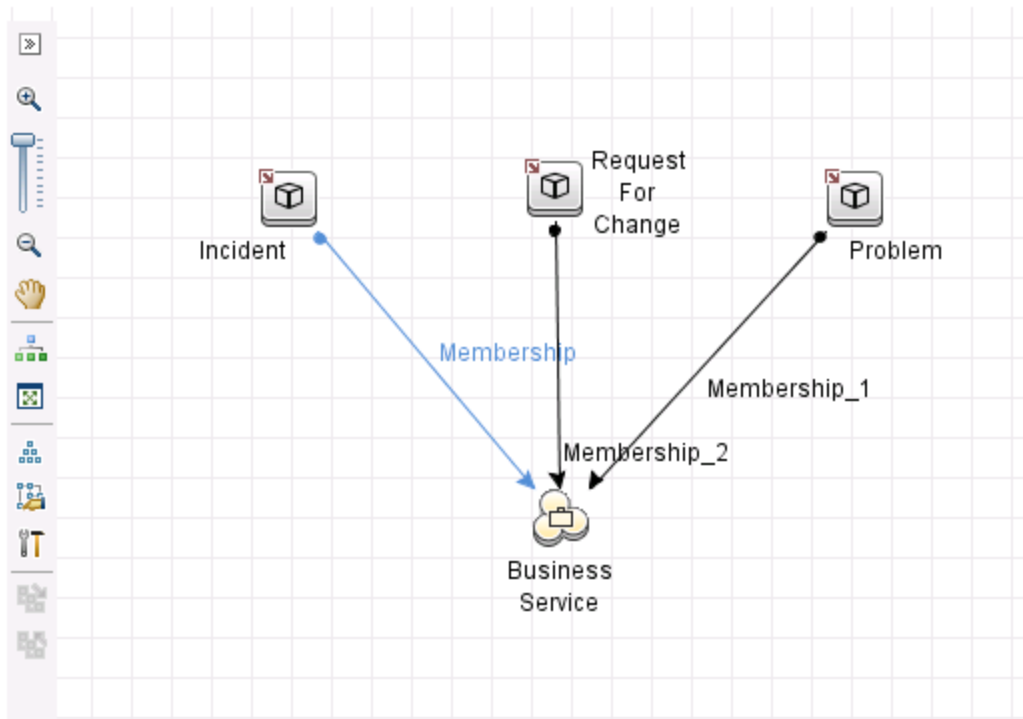


### Example 3: Federate Incident, Change, and Problem Record Data from Service Manager for UCMDB CIs

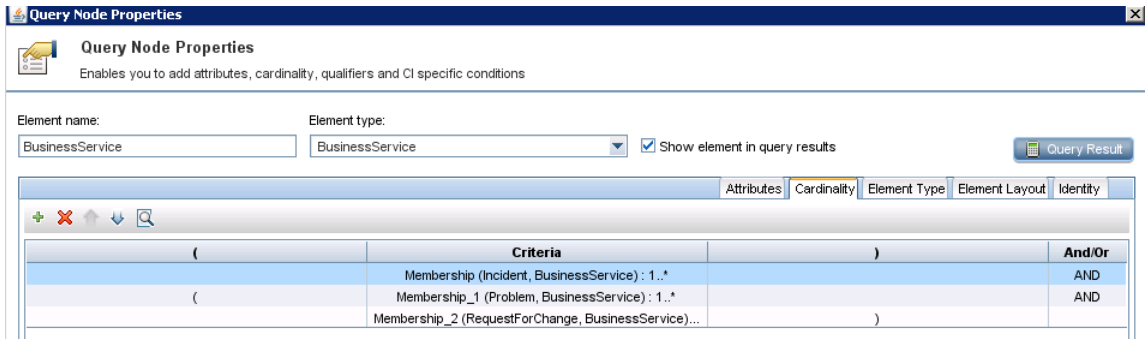
The following example illustrates how to retrieve information from the Incident, Change and Problem records in SM that affect a UCMDB Business Service CI.

1. Log in to UCMDB as an administrator.
2. Navigate to **Modeling > Modeling Studio > Resources**.
3. For Resource Type, select **Queries** from the list.
4. Click **New > Query**.
5. On the CI Type tab, go to **ConfigurationItem > BusinessElement > Service > BusinessService**, and drag it to the query pane on the right side.

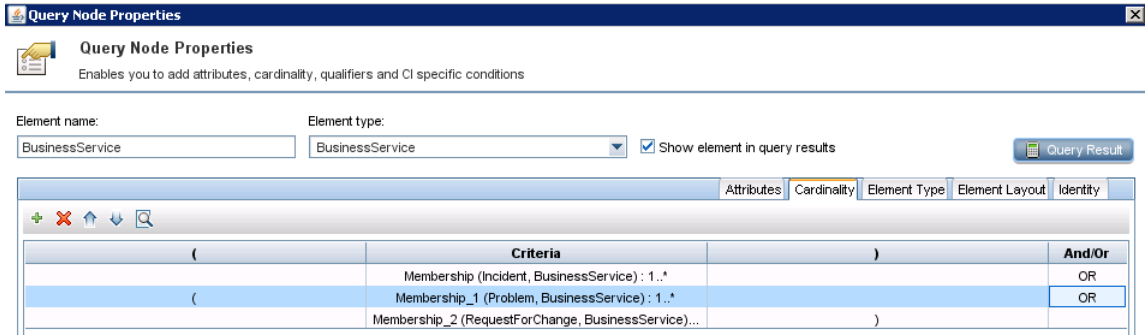
6. Go to **ItProcessRecord**, and drag **Incident**, **Problem**, and **RequestForChange** to the query pane.
7. Click the **Create Relationship** icon  to create regular relationships between the **BusinessService** node and the other nodes, as shown in the following figure.




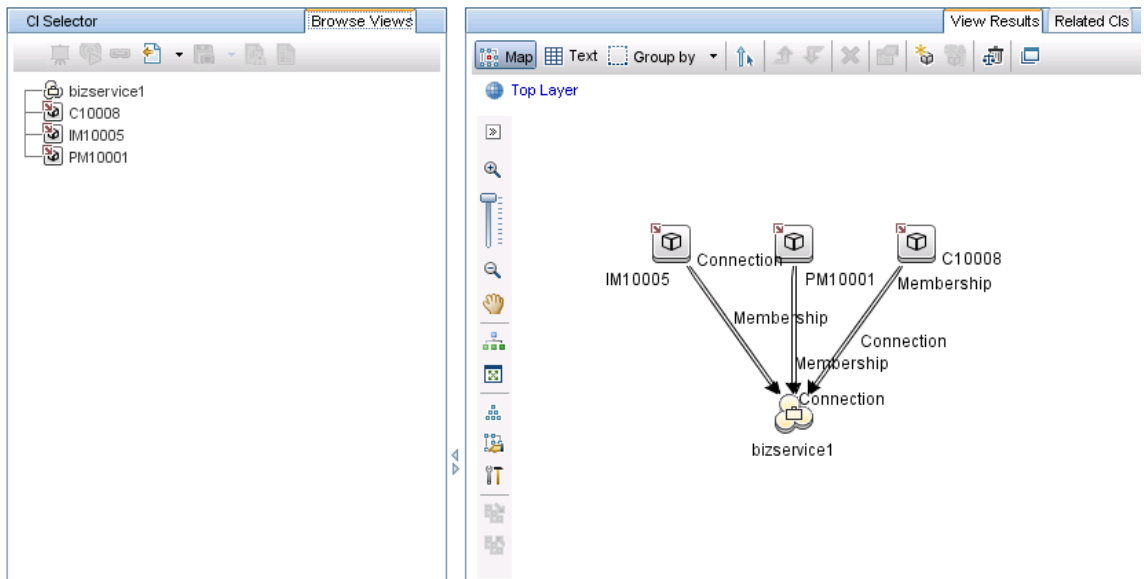
8. Select each node and click the **Data Sources** tab to specify a data source for each node.
  - a. For the **BusinessService** node, specify **UCMDB** as the data source.
  - b. For the **Incident**, **Problem**, and **RequestForChange** nodes, specify your integration point as the data source (**sm\_integration** in this example).
9. Save the query.
10. Optionally, edit the **BusinessService** node properties as needed.
  - a. Select the **BusinessService** node, and click **Edit** on the lower right pane.
  - b. Click the **Cardinality** tab. The default Cardinality setting is displayed.




- c. If you wish, change either or both of the **AND** operators to **OR**. This will change the filter criteria and therefore the query results.



11. Click the **Preview** icon  to view the query results.

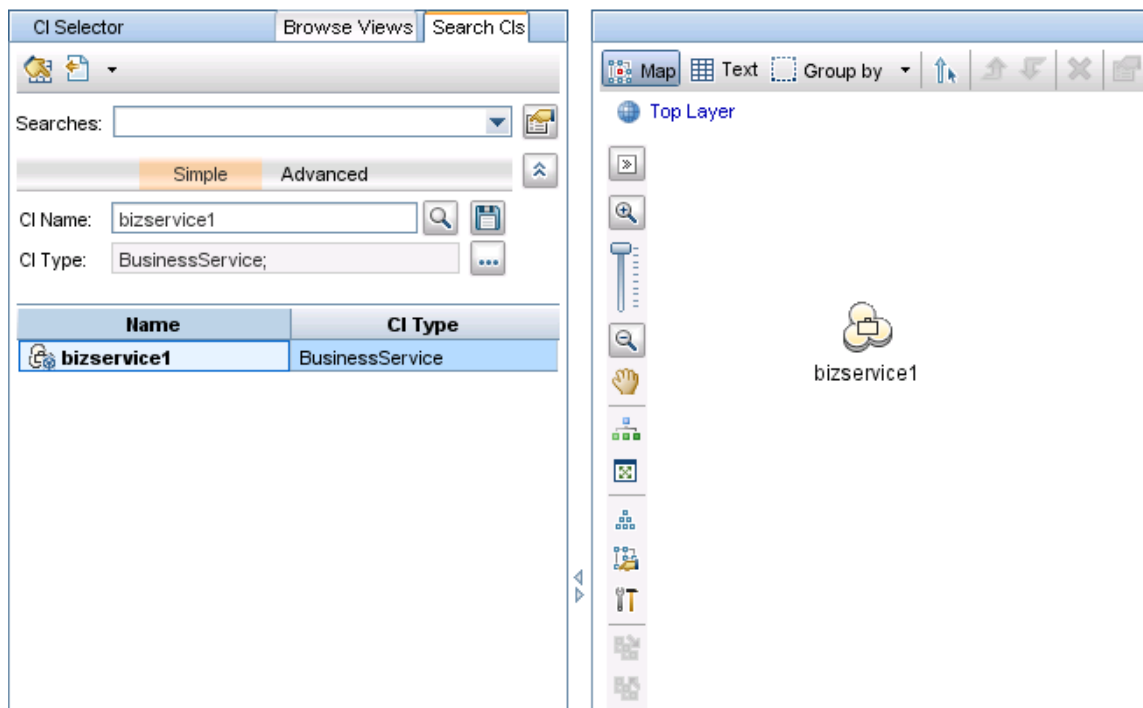




12. Select each SM Incident record from either the CI Selector pane or the query pane, and click the **Properties** icon  to view its details.
13. Select each UCMDB CI record from either the CI Selector pane or the query pane, and on the **Related CIs** tab click **Show Related CIs** to view its related CIs in both SM and UCMDB.

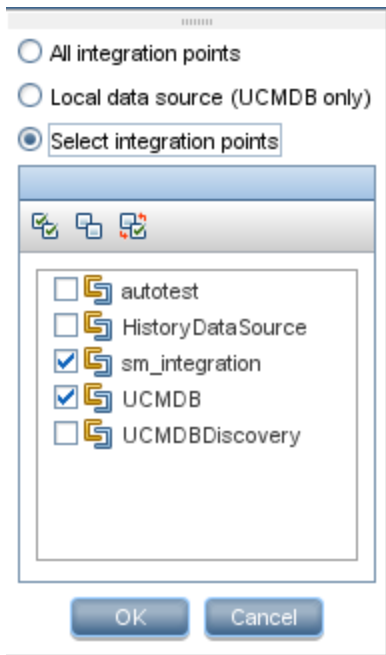
#### Example 4: Retrieve Service Manager Records Related to a UCMDB CI

The following example illustrates how to retrieve Service Manager records that are related to a UCMDB CI, by using the Get Related CIs functionality.

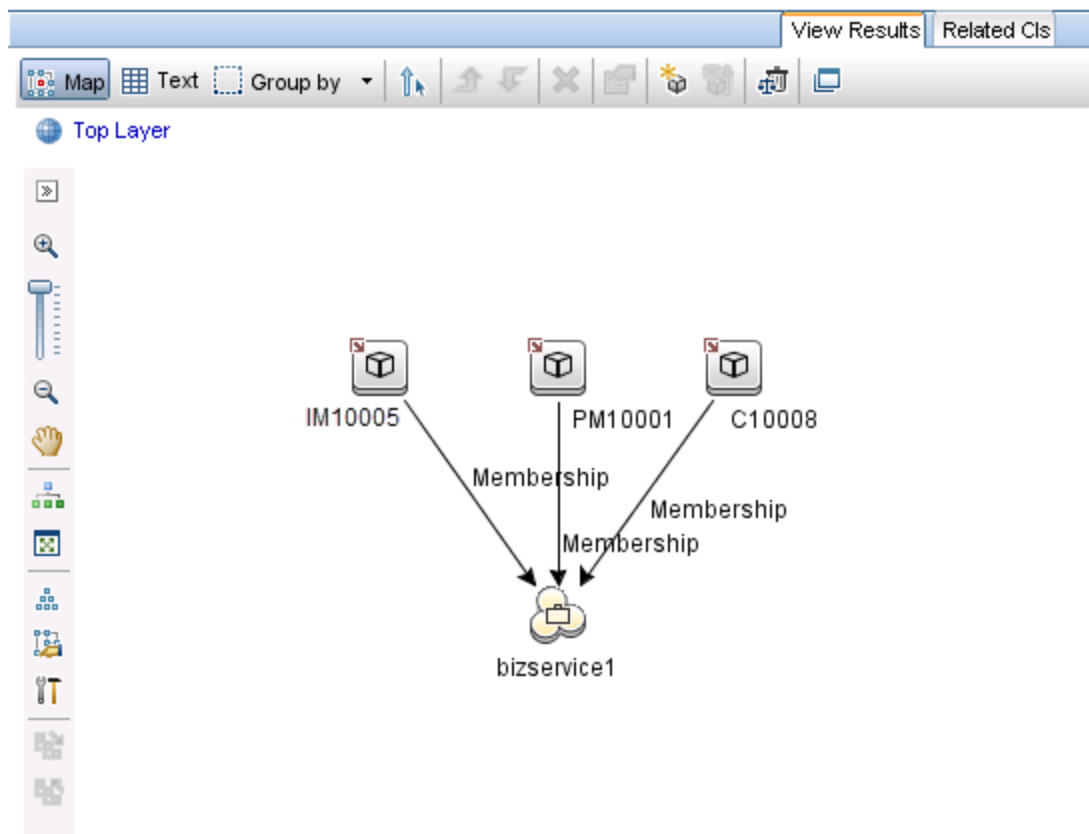
1. Log in to UCMDB as an administrator.
2. Navigate to **Modeling > IT Universe Manager**.
3. On the **Search CIs** tab, search for a UCMDB CI that has related records in Service Manager. For example, enter `bizservice1` in the CI Name field, click **Search**, and then double-click the CI to open it.




4. If the Get Related CIs pane is not displayed, click the **Show Get Related CIs pane** icon .
5. Click the **Select target Integration Points for related CIs** icon .
6. Select the **Select integration points** option, and then select both **UCMDB** and your integration point. Click **OK**.



7. Click **Show Related CIs**. The CI's related SM records and UCMDB CIs are displayed.



8. Select each SM record from the query pane, and click the **Properties** icon  to view its details.

## Multi-Tenancy (Multi-Company) Setup

**Note:** You cannot use the multi-tenancy configuration in the Universal CMDB, if you did not configure Universal CMDB for multi-tenancy at installtion.

The UCMDB-SM Integration supports a multi-tenancy configuration in which both the Service Manager and UCMDB systems track Configuration Items (CIs) and Configuration Item Relationships (CIRs) by company ID. In a multi-tenancy configuration, you can tailor the integration so that each tenant only sees and works with the CIs and CIRs that match their company ID. Multi-tenancy is intended for managed service providers (MSPs) who wish to offer Configuration Management as a service to multiple tenants.

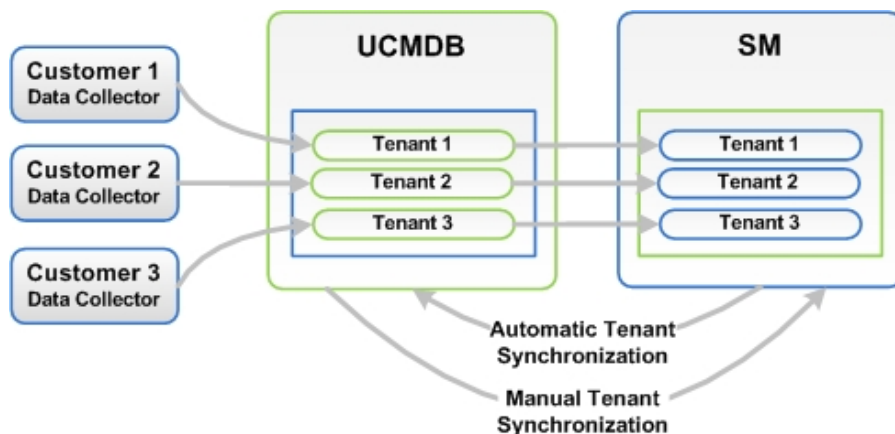
This chapter includes:

- ["Multi-Tenancy \(Multi-Company\) Support" below](#)
- ["Multi-Tenancy Requirements" on page 146](#)
- ["Setting up the Multi-Tenancy Integration in UCMDB" on page 147](#)
- ["Setting up the Multi-Tenancy Integration in Service Manager" on page 150](#)

## Multi-Tenancy (Multi-Company) Support

Multi-tenancy is when a single instance of software runs on a server, serving multiple client organizations (also referred to as tenants). Multi-tenancy contrasts with a multi-instance architecture where separate software instances or hardware systems are set up for different client organizations.

When implementing a multi-tenant architecture, a software application is designed to virtually partition its data and configuration so that each client organization works with a customized virtual application instance. The following figure illustrates an example multi-tenant integration deployment.



Every tenant configured in UCMDB works with the relevant tenant in SM. If UCMDB did not configure tenants, the tenant configuration must be activated in order to transfer the configuration from SM to UCMDB automatically. This function is performed once only by the system administrator.

In the event that UCMDB tenant configuration already exists and the SM configuration does not exist, SM tenants must be manually configured according to the UCMDB configuration.

## Implementing Multi-Tenancy in the UCMDB-SM Integration

SM stores the company records that describe each tenant in the multi-tenant configuration. The Service Manager system is the definitive source for company records and pushes all new company IDs to the UCMDB system creating the equivalent entity in UCMDB.

SM tracks the company ID of each CI and relationship in a multi-tenant configuration. CI records inherit the company ID of the UCMDB feeder that discovered them. Relationship records inherit the company ID of the parent CI in the relationship.

## Mandanten SM Security Layer

Mandanten is an SM software layer that is used to filter the customer ID from the CI information. SM uses the Mandanten to ensure that operators only see CI and relationship records where the CI's company ID matches the operator's company ID. If the view is restricted with Mandanten, then Service Manager also restricts the view to all other related records such as change requests and incidents.

## What Multi-Tenant Information is Stored in UCMDB

Your UCMDB system stores a company ID attribute for each CI and CIR. The company ID determines what adapter and synchronization schedule your UCMDB system uses to update CI data. Each CI and relationship record can only have one company ID. The UCMDB system obtains a company ID from the Service Manager system.

If more than one tenant (company) shares the same CI, each tenant has their own unique CI record describing the CI. In effect, the UCMDB system creates multiple CI records to track one managed asset. Each tenant's CI record is unique to that tenant and lists the company's unique company ID.

## What Multi-Tenant Information is Stored in Service Manager

Your Service Manager stores the company records that describe each tenant in the multi-tenant configuration. The Service Manager system is the definitive source of company IDs and pushes new and updated information to your UCMDB system.

Service Manager tracks the company ID of each CI and relationship in a multi-tenant configuration. CI records inherit the company ID of the UCMDB feeder that discovered them. Relationship records inherit the company ID of the parent CI in the relationship.

In a best practices implementation, Service Manager uses Mandanten to ensure that operators only see CI and relationship records where the CI's company ID matches the operator's company ID. If you restrict the view with Mandanten, then Service Manager also restricts the view to all other related records such as change requests and incidents.



## Unique Logical Names

Service Manager requires that all CIs have unique logical names. If the logical name generation process produces a duplicate logical name value, Service Manager appends an underscore and a number to the end of logical name to make it unique. For example, if two CIs would have the logical name **mytesthost**, then the second CI will instead have the name **mytesthost\_1**. A second duplicate CI would have the name **mytesthost\_2**.

## Synchronization of Company Records

If your system meets all the conditions for multi-tenancy support, Service Manager creates a schedule record to push the company ID of the company record to your UCMDB system. Service Manager uses the following rules to determine whether to push the company ID to your UCMDB system.

### Conditions where Service Manager synchronizes company ID with UCMDB

Conditions	Tenant information synchronized?	Schedule record created and action taken in UCMDB
<ul style="list-style-type: none"> <li>• UCMDB-SM integration <b>enabled</b></li> <li>• Multi-company mode <b>enabled</b> in Service Manager</li> <li>• You create a new company record in Service Manager</li> </ul>	Yes	Synch Company with UCMDB - <UCMDB Company ID> <ul style="list-style-type: none"> <li>• Add new company ID</li> </ul>
<ul style="list-style-type: none"> <li>• UCMDB-SM integration <b>enabled</b></li> <li>• You update an existing company record that has not been synchronized with UCMDB</li> <li>• Multi-company mode <b>enabled</b> in Service Manager</li> </ul>	Yes	Synch Company with UCMDB - <UCMDB Company ID> <ul style="list-style-type: none"> <li>• Add new company ID</li> </ul>
<ul style="list-style-type: none"> <li>• UCMDB-SM integration <b>enabled</b></li> <li>• Multi-company mode <b>enabled</b> in Service Manager</li> <li>• You disable the option to show a company in multi-company lists on a company synchronized with UCMDB</li> </ul>	Yes	Inactivate Company with UCMDB - <UCMDB Company ID> <ul style="list-style-type: none"> <li>• Inactivate existing company ID</li> </ul>

**Conditions where Service Manager synchronizes company ID with UCMDB, continued**

Conditions	Tenant information synchronized?	Schedule record created and action taken in UCMDB
<ul style="list-style-type: none"> <li>• UCMDB-SM integration <b>enabled</b></li> <li>• Multi-company mode <b>enabled</b> in Service Manager</li> <li>• You select the option to resynchronize with UCMDB on an existing company record</li> </ul>	Yes	Synch Company with UCMDB - <UCMDB Company ID> <ul style="list-style-type: none"> <li>• Add new company ID</li> </ul>
<ul style="list-style-type: none"> <li>• UCMDB-SM integration <b>enabled</b></li> <li>• Multi-company mode <b>enabled</b> in Service Manager</li> <li>• You enable the option to show a company in multi-company lists for an inactivated company</li> </ul>	Yes	Synch Company with UCMDB - <UCMDB Company ID> <ul style="list-style-type: none"> <li>• Reactivate company ID</li> </ul>
<ul style="list-style-type: none"> <li>• UCMDB-SM integration <b>disabled</b></li> <li>• Multi-company mode <b>enabled</b> in Service Manager</li> <li>• You update an existing company record that has already been synchronized with UCMDB</li> </ul>	No	None
<ul style="list-style-type: none"> <li>• UCMDB-SM integration <b>disabled</b></li> <li>• Multi-company mode <b>enabled</b> in Service Manager</li> <li>• You create a new company record in Service Manager</li> </ul>	No	None
<ul style="list-style-type: none"> <li>• UCMDB-SM integration <b>enabled</b></li> <li>• Multi-company mode <b>enabled</b> in Service Manager</li> <li>• You disable the option to show a company in multi-company lists on a company not synchronized with UCMDB</li> </ul>	No	None
<ul style="list-style-type: none"> <li>• UCMDB-SM integration <b>enabled</b></li> <li>• Multi-company mode <b>disabled</b> in Service Manager</li> <li>• You create a new company record in Service Manager</li> </ul>	No	None

## UCMDB Customer ID

When you enable the multi-tenancy integration, Service Manager displays a new field in each company record called UCMDB Customer ID. In order to synchronize a company record with UCMDB, you must first provide a value for this field. After you provide a UCMDB Customer ID value this field becomes read-only. You cannot change a company's UCMDB Customer ID after you set it.

This field only accepts numeric data up to ten characters long. Service Manager requires the field value to be a unique positive whole number. You cannot enter duplicate values or use decimals, negative numbers, or zero.

Your UCMDB system automatically uses the UCMDB customer ID of 1 when running in single tenant mode. You can reuse this default value in your multi-tenant implementation by assigning a Service Manager company to have this UCMDB customer ID value. Out-of-the-box, no Service Manager company has the UCMDB customer ID of 1.

## UCMDB User ID and Password

When you enable the multi-tenancy integration, Service Manager displays two new fields in each company record called UCMDB UserId and UCMDB Password. These fields allow you to specify the connection information you want Service Manager to use when requesting information for the Actual State section. Any user name and password you enter in these fields must be valid for your UCMDB system.

The user name and password you provide in the Company Information record takes precedence over the user name and password you provide in the System Information record. This allows managed service providers to control access to the UCMDB system on a tenant-by-tenant basis. If you do not provide a company-specific UCMDB user name and password, Service Manager uses the credentials you provided in the System Information record.

## Company Code

The multi-tenancy integration requires that each company record has a unique Company Code (company field) value. Since Company Code is a required field, your existing company records should already have Company Code values. However you should ensure that each company record has a unique Company Code value.

When using Service Manager Enhanced Generic Adapter for the UCMDB-SM integration, you must define your company codes in the `ServiceManagerEnhancedAdapter9-x/mappings/scripts/SMUtils.groovy` file using the following format (where a comma is used to separate mapping entries):

```
[ "<UCMDB Company 1 Code>":"<SM Company 1 Code>", "<UCMDB Company 2 Code>":"<SM Company 2 Code>" ]
```

For example: ["1":"hp", "2":"sap"]

**Caution:** You should not change the Company Code value after you have enabled the multi-tenancy integration because this will cause your Service Manager data to become out of synch.

## CI Reconciliation Rules

When multi-tenancy is enabled, Service Manager only reconciles the CIs whose company ID matches the company ID in the data push job. For example, when pushing CIs from company 2, the reconciliation rules only apply to the Service Manager CI records that have the company code corresponding to company number 2.

## Company Information Pushed to CI and CI Relationship Records

When you enable the multi-tenancy integration, Service Manager inserts the SM Company Code value in CI and relationship records during data push. Service Manager uses the UCMDB Customer ID to look up the matching SM Company Code value.

## Company Information Replicated to Incident Records

When you enable the multi-tenancy integration and select the option to create incidents when UCMDB discovers new, updated, or deleted CIs, Service Manager inserts the SM Company Code value in the incident record during replication. Service Manager uses the UCMDB Customer ID to look up the matching SM Company Code value.

## Schedule Records

Service Manager uses the **problem** schedule process to manage the synchronization of company IDs to your UCMDB system. You can manually enable the **problem** schedule process from the System Status form.

When the synchronization criteria are met as described in Table ["Conditions where Service Manager synchronizes company ID with UCMDB" on page 141](#), Service Manager creates a "Synch Company with UCMDB - <UCMDB Company ID>" schedule record (for example, "Synch Company with UCMDB - 1234567890"). If you inactivate a company, Service Manager creates an "Inactivate Company with UCMDB - <UCMDB Company ID>" schedule record (for example, "Inactivate Company with UCMDB - 1234567890"). The **problem** schedule process processes the new schedule record on the next background process iteration.

If your Service Manager system cannot connect to your UCMDB system for some reason, it will reschedule the company synchronization at the next scheduled interval (the out-of-the-box interval is 5 minutes). The problem schedule process updates the schedule record with the status rescheduled. If the Service Manager system receives any other error message while connecting to the UCMDB system, it updates the schedule record with the status “application failed due to error - check msglog for possible messages.”

## Tenant-Specific Discovery Event Manager (DEM) Rules

You can implement the condition field function in order to create SM DEM rules that are specific to a particular tenant in a multi-tenancy UCMDB-SM integration.

Tenant rules vary according to SM tenant configuration requirements, for each record information type pushed from UCMDB to SM different tenants can configure different DEM tenant rules.

Each tenant can have its own set of unique requirements and therefore may implement different processes through the integration.

One tenant may require the addition of CIs directly to SM while another tenant may require opening changes for each CI.

The following table shows a sample set of DEM rules that illustrate how to accomplish this.

### Tenant-specific DEM rules

DEM rule ID	Action on new CI	Condition
ucmdbNode_advantage	Add CI	company in \$L.file="advantage"
ucmdbNode_hp	Create change	company in \$L.file="HP"

#### **Note: DEM Rules**

When creating DEM rules make sure to create separate DEM rules for each tenant.

## Multi-Tenancy Use Cases

The following table describes the necessary actions to perform in various deployment situations to address multi-tenancy issues.

**Multi-tenancy use cases**

<b>Deployment Integration Type</b>	<b>Description</b>
UCMDB with multi-tenancy rules  SM without multi-tenancy rules	When implementing a UCMDB-SM deployment that has existing multi-tenancy rules in UCMDB and does not have multi-tenancy rules configured in SM, the user creates multi-tenancy rules in SM manually and according to the rules in UCMDB.
SM with multi-tenancy rules  UCMDB without multi-tenancy rules	When implementing a UCMDB-SM deployment that has existing multi-tenancy rules in configured SM and does not have multi-tenancy rules configured in UCMDB, the user creates multi-tenancy rules in UCMDB manually as well as according to the rules previously configured in SM.
UCMDB without multi-tenancy rules  SM without multi-tenancy rules	When implementing a UCMDB-SM deployment that does not have multi-tenancy rules configured in UCMDB or in SM, the user configures the rules in SM.  During the configuration process using the SM multi-tenancy wizard the user can create corresponding tenancy configuration in UCMDB. By creating corresponding tenancy configurations in SM the user also creates a corresponding tenant in UCMDB.

## Multi-Tenancy Requirements

Your system must meet the following requirements in order for the integration to support multi-tenancy.

- HP Universal CMDB version 8.02 or later
- HP Service Manager version 9.20 or later

- Integration enabled between UCMDB and Service Manager
- Multi-company mode enabled on the Service Manager system
- Problem schedule process running on the Service Manager system

For additional information about the multi-tenancy integration, you can visit the [HP Software Support Online web site](#) or refer to the Service Manager help.

## Setting up the Multi-Tenancy Integration in UCMDB

You need to perform the following tasks in UCMDB to set up the multi-tenancy integration.

1. Install a separate data flow probe for each tenant the integration will support.  
See "[How to Install a Separate Data Flow Probe for Each Tenant](#)" below.
2. Start tenant-specific data flow probes.  
See "[How to Start Tenant-Specific Data Flow Probes](#)" on page 149.
3. Configure IP address ranges for tenant-specific data flow probes.  
See "[How to Configure IP Ranges for Tenant-Specific Data Flow Probes](#)" on page 149.

**Caution:** Multi-tenancy is not supported for population when the integration uses the Service Manager Enhanced Generic Adapter.

## How to Install a Separate Data Flow Probe for Each Tenant

If you plan to support a multi-tenant configuration, you must install a separate data probe for each tenant. Out-of-the-box, the UCMDB installer only installs one data flow probe and service.

To install additional data flow probes and start them from your operating system command prompt:

1. Log in to the host of your UCMDB system as an administrator.
2. Insert the HP Universal CMDB Setup Windows DVD into the system disc drive.
3. Start the Data Flow Probe installer (HPUCMDB\_DataFlowProbe\_x.xx.exe).
4. Follow the on-screen instructions to complete the wizard, but use the following values for each data flow probe you wish to install.

- a. Type a unique path for each installation folder.
- b. Use the same UCMDB application server address for each data flow probe.
- c. Type a valid data flow probe address.
- d. Type a unique name for each data flow probe identifier.
- e. Create a unique customer Data Flow Probe domain for each probe (Clear the Use Default CMDB Domain option).
- f. Use the same probe gateway and probe manager settings for each probe (for example, use combined or separate processes).

See the *HP Universal CMDB Deployment Guide* for complete installation instructions.

- 5. Repeat [step 3](#) and [step 4](#) for each data flow probe you wish to install.
- 6. Open the probe's `DiscoveryProbe.properties` file in a text editor. By default, this file is located in the following folder:

`<UCMDB installation folder>\<data flow probe installation folder>\conf`

For example, `C:\hp\UCMDB\DataFlowProbe\conf`.

**Note:** The `<data flow probe installation folder>` must be unique for each tenant.

- 7. Edit the following properties in the configuration file.

**Discovery Probe properties set for each tenant**

Property	Value
serverName	Verify the name of the UCMDB server
customerId	Type the customer ID for the tenant this data flow probe supports
appilog.collectors.probe.name	Verify the probe name is unique such as server + tenant ID
appilog.collectors.domain	Verify the domain name of the data flow probe
appilog.collectors.local.ip	Verify the data flow probe gateway name
appilog.collectors.probe.ip	Verify the data flow probe manager name



**Discovery Probe properties set for each tenant , continued**

Property	Value
appilog.collectors.rmi.port	Type a unique port for each probe
appilog.collectors.rmi.gw.port	Type a unique port for each probe
appilog.collectors.probe.html.port	Type a unique port for each probe
appilog.collectors.local.html.port	Type a unique port for each probe
appilog.collectors.ProbeUseSpecific RMIPortFrom	Type a unique port for each probe or type 0 to have the system automatically select it
appilog.collectors.bigBrother.port	Type a unique port for each probe

8. Save the configuration file.
9. Repeat [step 6](#) to [step 8](#) for the data flow probe of each tenant.

## How to Start Tenant-Specific Data Flow Probes

To start tenant-specific data flow probes:

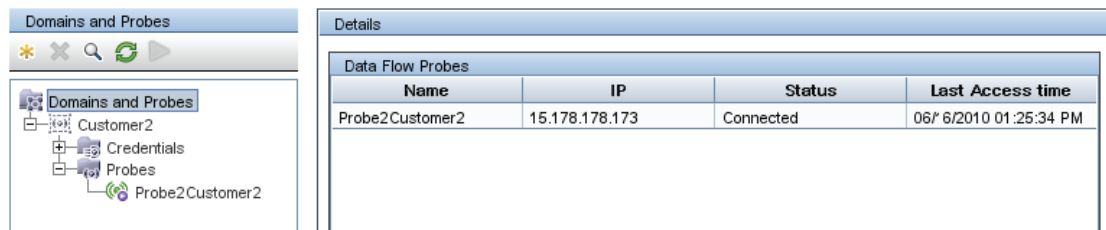
1. Open the OS command prompt and navigate to the probe's bin folder. For example, C:\hp\UCMDB\DataFlowProbe1\bin.
2. Type `gateway console`.
3. Repeat step 1 to step 2 for each data flow probe you want to start.


## How to Configure IP Ranges for Tenant-Specific Data Flow Probes

To configure IP ranges for tenant-specific data flow probes:

1. Log in to UCMDB as an administrator using the company ID of the tenant whose data flow probe you want to configure.
2. Navigate to **Data Flow Management > Data Flow Probe Setup**.
3. Expand the data flow probe domain containing the probe you want to start. For example, **Customer2**.

- Expand the Probe node and select the data flow probe you want to start. For example, **Probe2Customer2**.



- Click the **Add IP range** icon .
- Type an IP range you want the Data Flow Probe to scan. Optionally, add any IP ranges you want to exclude.
- Click **OK** to save the IP range.
- Repeat step 1 to step 7 for each data flow probe you want to configure.

## Setting up the Multi-Tenancy Integration in Service Manager

You need to perform the following tasks in Service Manager to set up the multi-tenancy integration.

Multi-tenancy support is an optional feature of the integration intended for Managed Service Providers (MSPs) who want to offer Configuration Management as a service to their tenants. In a multi-tenancy configuration, each CI and CIR record has a corresponding company ID. Out-of-the-box, Service Manager allows all operators to view CI data regardless of the company ID. If you wish to restrict access to CI data by company ID, you must enable Mandanten and use the company ID field as a restricting query. See the Service Manager help for more information about multi-company mode and Mandanten.

You must complete the following tasks from your Service Manager system to enable multi-tenancy support for the integration.

- Start the process schedule.  
See ["How to Start the Schedule Process "](#) on the next page.
- Configure the Service Manager System Information Record.  
See ["How to Configure the Service Manager System Information Record"](#) on page 152.
- Add tenant-specific UCMDB ID and password values to company records (optional).  
See ["How to Add Tenant-Specific UCMDB User ID and Password Values"](#) on page 153.

4. Add UCMDB Customer ID values to existing company records.  
See ["How to Add UCMDB Customer ID values to Existing Companies" on page 154.](#)
5. Synchronize existing company records with UCMDB.  
See ["How to Synchronize Existing Companies from Service Manager to UCMDB" on page 154.](#)
6. Verify that Service Manager synchronized company records with UCMDB (optional).  
See ["How to View Whether Company Information Is in UCMDB" on page 155.](#)
7. Resynchronize existing company records with UCMDB (as needed).  
See ["How to Resynchronize an Existing Company with UCMDB" on page 156.](#)
8. Inactivate company records you do not want to be part of the integration (as needed).  
See ["How to Inactivate a Synchronized Company" on page 157.](#)
9. Reactivate inactive company records you want to be part of the integration (as needed).  
See ["How to Reactivate an Inactive Company" on page 157](#)
10. Add tenant-specific DEM rules.  
See ["How to Add Tenant-Specific DEM Rules" on page 158.](#)

## How to Start the Schedule Process

This integration needs the **problem** schedule process in order to synchronize company records from Service Manager to UCMDB. Make sure that the process is started before you synchronize company records.

To start the **problem** schedule process:

1. Log in to Service Manager as a system administrator.
2. From the System Navigator, click **System Status** to display a list of currently started schedules.
3. Click the **Refresh Display** button to refresh the list.
4. If the **problem** schedule process is not in the list, perform the following steps:
  - a. Click the **Start Scheduler** button.
  - b. Double-click the **problem** schedule process.

A message is displayed, indicating that the **problem** schedule process is started.

## How to Configure the Service Manager System Information Record

To enable the integration to support multi-tenancy, you must provide additional information in the Service Manager System Information Record.

**Caution:** In order to enable multi-tenancy support, you must use HP Universal CMDB version 8.02 or greater. Earlier versions of HP Universal CMDB will produce an error message if you attempt to run them in multi-tenancy mode.

To provide additional information in the Service Manager System Information Record:

1. Log in to Service Manager as a system administrator.
2. Navigate to **System Administration > Base System Configuration > Miscellaneous > System Information Record**.
3. Click the **General** tab.
4. Enable the **Run in Multi-Company Mode** option.
5. Click the **Active Integrations** tab.
6. Select the **HP Universal CMDB** option.  
The form displays the UCMDB web service URL field.
7. In the UCMDB web service URL field, type the URL to the CI synchronization web service API. The URL has the following format:  
`http://<UCMDB server name>:<port>/axis2/services/ucmdbSMService`  
  
Replace *<UCMDB server name>* with the host name of your UCMDB server, and replace *<port>* with the communications port your UCMDB server uses.
8. In **UserId** and **Password**, type the user credentials required to manage CIs on the UCMDB system. For example, the out-of-the-box administrator credentials are **admin/admin**.
9. In the **Multi-tenant web service URL** field, type the URL to the company ID synchronization web service API. The URL has the following format:  
`http://<UCMDB server name>:<port>/axis2/services/UcldbManagementService`  
  
Replace *<UCMDB server name>* with the host name of your UCMDB server, and replace *<port>* with the communications port your UCMDB server uses.

10. Type the user name and password required to synchronize company IDs on the UCMDB system. For example, the out-of-the-box system administrator credentials for UCMDB are **sysadmin/sysadmin**.
11. Click **Save**. Service Manager displays the message: Information record updated.
12. Log out of the Service Manager system, and log in again with an administrator account.
13. Click **System Status > Display Options > All Tasks**.
14. Type **x** in the Command field next to the **problem** schedule process and click **Execute Commands**. Wait a few minutes for the **problem** schedule process to close.
15. Click **Start Scheduler**.
16. Double-click the **problem** schedule process. The system now supports multi-tenancy for UCMDB.

## How to Add Tenant-Specific UCMDB User ID and Password Values

You can provide a tenant-specific UCMDB user name and password for Service Manager to use when requesting information for the Actual State section. If you provide no credentials, Service Manager uses the credentials in the System Information Record for all tenants.

**Note:** Any credentials you provide in the company record take precedence over credentials you provide in the System Information Record. The UCMDB UserId and UCMDB Password fields are available only when you have enabled the multi-tenancy integration.

To add tenant-specific UCMDB user name and password:

1. Log in to Service Manager as a system administrator.
2. Navigate to **System Administration > Base System Configuration > Companies**.
3. Type the search criteria you want to use to find company records. For example, leave the search form blank to search all company records.
4. Click **Search**.
5. Type the user name you want this company to use to connect to UCMDB in the UCMDB UserId field.
6. Type the password for the UCMDB user name in the UCMDB Password field.

7. Click **Save**.
8. Repeat [step 3](#) through [step 7](#) for each company for which you want to provide credentials.

## How to Add UCMDB Customer ID values to Existing Companies

You can use the following steps to add a UCMDB Customer ID value to your existing Service Manager company records.

1. Log in to Service Manager as a system administrator.
2. Navigate to **System Administration > Base System Configuration > Companies**.
3. Type the search criteria you want to use to find company records. For example, leave the search form blank to search all company records.
4. Click **Search**.
5. Type a numeric value in the UCMDB Customer ID field for this company.
6. Click **Save**.
7. Service Manager prompts to confirm that you want to synchronize the record with UCMDB. Click **Yes** if you want to synchronize the company now, or click **No** if you want to synchronize the company later.
8. Click **Next** to go to the next company in the record list.
9. Repeat [step 5](#) through [step 8](#) for each company in the record list.

## How to Synchronize Existing Companies from Service Manager to UCMDB

Your Service Manager system may already contain company records that you want to use with the multi-tenancy integration.

If you update any field in a company record that has not yet been synchronized to UCMDB, Service Manager prompts whether you want to synchronize the company to UCMDB.

**Note:** Service Manager will not prompt you to synchronize the company record if you have disabled the option to show the company in multi-company lists, or if there is a pending schedule record associated with the company. See ["How to Inactivate a Synchronized Company" on page 157](#) for

more information.

This task includes the following steps:

1. Log in to Service Manager as a system administrator.
2. Navigate to **System Administration > Base System Configuration > Companies**.
3. Type the search criteria you want to use to find company records. For example, leave the search form blank to search all company records.
4. Click **Search**.
5. Select a company record to update.
6. Update the company record.
7. Click **Save**. Service Manager prompts you to confirm that you want to synchronize the record with UCMDB.

**Note:** Service Manager saves the company record regardless of your synchronization choice.

## How to View Whether Company Information Is in UCMDB

When you enable the multi-tenancy integration, Service Manager displays a read-only field in each company record that indicates whether or not the UCMDB Customer ID has been synchronized with your UCMDB system.

**Note:** The UCMDB Customer ID field is visible only when you enable the multi-tenant UCMDB integration.

To view whether company information is in UCMDB:

1. Log in to Service Manager as a system administrator.
2. Navigate to **System Administration > Base System Configuration > Companies**.
3. Type the search criteria you want to use to find company records. For example, leave the search form blank to search all company records.
4. Click **Search**.

5. Review the status of the **Synched with UCMDB** field.  
If the check box is checked, then Service Manager has already synchronized the company ID with your UCMDB system. If the check box is unchecked, then Service Manager has yet to add this company to your UCMDB system.

**Note:** For more information about synchronization failures, see ["Schedule Records" on page 144.](#)

## How to Resynchronize an Existing Company with UCMDB

Service Manager provides you a means to resynchronize company records with your UCMDB system in case you lose UCMDB data for some reason. For example, you might intentionally remove UCMDB data during integration testing, or you might need to recover data after a disaster. You can force Service Manager to synchronize companies with your UCMDB system with the Re-synch with UCMDB option.

To resynchronize an existing company with UCMDB:

1. Log in to Service Manager as a system administrator.
2. Navigate to **System Administration > Base System Configuration > Companies.**
3. Type the search criteria you want to use to find company records. For example, leave the search form blank to search all company records.
4. Click **Search.**
5. Select a company record to synchronize.
6. Click the **Re-synch** button next to the **Synched with UCMDB** check box.

**Note:** The **Re-synch** button is available only from company records that have already been synchronized with UCMDB and have the **Synched with UCMDB** check box checked. If your UCMDB system already has a company with this ID value, it will ignore the resynchronization request. Service Manager will also ignore a resynchronization request if there is an existing schedule record to resynchronize the company with UCMDB. In this case, it displays the message "A schedule record has already been added to re-synch this company with UCMDB."



## How to Inactivate a Synchronized Company

After you have synchronized a company record with UCMDB you can no longer delete the record. Instead, you can inactivate a company record, which causes the UCMDB system to cease all further CI updates for the company. Any existing CI data for the company remains in the UCMDB system associated with the inactive UCMDB Customer ID, but both the company and any associated CIs will no longer be visible from the UCMDB system.

To inactivate a synchronized company:

1. Log in to Service Manager as a system administrator.
2. Navigate to **System Administration > Base System Configuration > Companies**.
3. Type the search criteria you want to use to find company records. For example, leave the search form blank to search all company records.
4. Click **Search**.
5. Select a company record to inactivate.
6. Select **No** from **Show Company in Multi-Company Lists**.
7. Click **Save**.
8. If this company was previously synchronized with UCMDB, Service Manager prompts you to confirm the inactivation.
9. Click **Yes** to confirm the inactivation or **No** to cancel your changes.

## How to Reactivate an Inactive Company

You can reactivate any inactive companies on your Service Manager system to include them in the multi-tenancy integration. You must also synchronize the company with UCMDB for UCMDB to process any CI updates for this company.

To reactivate an inactive company:

1. Log in to Service Manager as a system administrator.
2. Navigate to **System Administration > Base System Configuration > Companies**.

3. Type the search criteria you want to use to find company records. For example, leave the search form blank to search all company records.
4. Click **Search**.
5. Select a company record to reactivate.
6. Select **Yes** from **Show Company in Multi-Company Lists**.
7. Click **Save**. Service Manager prompts you to reactivate the company with UCMDB.
8. Click **Yes**. Service Manager creates a schedule record to reactivate the company.

## How to Add Tenant-Specific DEM Rules

You can use the condition field to create DEM rules that are specific to a particular tenant in a multi-tenancy UCMDB-SM integration. For example, one tenant may want to add CIs directly to Service Manager while another tenant may want to open changes for each CI. The following sample DEM rules illustrate how to accomplish this.

### Tenant-specific DEM rules

DEM rule Id	Action on new CI	Condition
ucmdbNode_advantage	Add CI	company in \$L.file="advantage"
ucmdbNode_hp	Create change	company in \$L.file="HP"

**Tip:** It is a best practice to create a separate DEM rule for each tenant.

## Standards and Best Practices

This chapter includes:

- ["UCMDB-SM Configuration Best Practices" below](#)
- ["Frequently Asked Questions" on page 172](#)

## UCMDB-SM Configuration Best Practices

This section provides best practices and recommendations for successfully implementing this integration in various environments. This section provides you with valuable understandings and

techniques that will enhance the UCMDB-SM integration as well as solve common problems by providing solutions and workarounds to these issues. These practices and recommendations may vary slightly according to each implementation, as the specific system requirements and settings alter per system environment.

This section includes:

- ["CI Name Mapping Considerations" below](#)
- ["Bi-Directional Data Synchronization Recommendations" on the next page](#)
- ["Push Scheduling Recommendations" on page 162](#)
- ["Push in Clustered Environments" on page 163](#)
- ["Initial Load Configurations" on page 165](#)
- ["How to Configure Differential or Delta Load DEM Rules" on page 170](#)
- ["Fault Detection and Recovery for Push" on page 171](#)
- ["How to Enable Lightweight Single Sign-On \(LW-SSO\) Configuration" on page 172](#)

## CI Name Mapping Considerations

UCMDB allows duplicate CI names while Service Manager requires unique logical names. Before pushing UCMDB CIs, you need to define a correct CI name mapping for them. For example, many UCMDB CIs (such as CIs of the Running Software, CRG, Switch, or Router type) have the same display label.

To prevent duplicate CI names from occurring in Service Manager when pushing UCMDB CIs, the following mappings are provided out-of-the-box:

### **CRG mapping**

Out-of-the-box, UCMDB CRG records are mapped to Service Manager as follows:

- If a Cluster exists for a CRG, the CRG is mapped to this CI logical name: <Cluster display label>\_<CRG display label>;
- If the CRG does not have a Cluster, but has several IP addresses, the CRG is mapped to the following (where the IP addresses are sorted alphabetically):  
<IpAddress1>\_...\_<IpAddressN>.<authoritativeDnsName>\_<CRG display label> (when IpAddress.authoritativeDnsName exists)

<IpAddress1>\_...\_<IpAddressN>\_<CRG display label> (when IpAddress.authoritativeDnsName does not exist)

- If neither a Cluster nor an IP address exists for the CRG, it is mapped directly to <CRG display label>.

### Running Software mapping

Running Software CIs are prefixed with their root container node display label when mapped to a Service Manager CI: <Node display label>\_<Running Software display label>.

### Switch and Router mapping

Switch or Router type CI records in UCMDB are prefixed with their MAC addresses when mapped to a Service Manager CI: <MACAddress1>\_...\_<MACAddressN>\_<Switch or Router display label>, where the MAC addresses are sorted alphabetically.

## Bi-Directional Data Synchronization Recommendations

The UCMDB-SM integration supports bi-directional data synchronization between UCMDB and Service Manager (SM). HP recommends that you follow the following best practices to avoid unnecessary problems due to improper use of the data push and population features:

- For CIs/CI Relationships that UCMDB can automatically discover, use UCMDB as the data source. Do not make changes to them in Service Manager; instead, always let UCMDB discover their changes and push the changes to SM.
- For CIs/CI Relationships that UCMDB cannot automatically discover, use SM as the data source. Do not make changes to them in UCMDB; instead, always make changes to them in SM and populate the changes to UCMDB.
- For CIs/CI Relationships that are already created in SM and UCMDB can automatically discover, run a one-time population to synchronize them to UCMDB, and then use UCMDB as their data source.

**Caution:** Problems like the following may occur if you do not follow these best practices:

#### Problem 1:

[Population Adapter] After CIs/CI Relationships are pushed from UCMDB, if you directly make changes in SM to these records without ever populating them back to UCMDB first, the changes cannot be populated to UCMDB.

#### Workaround:

Changing these UCMDB records in SM is not recommended; however if you need to do so you can do the following to solve this issue: After the records are pushed to SM, populate them back to UCMDB first before making any changes to them in SM. This way the changes can then be populated to UCMDB.

**Problem 2:**

[Population Adapter] After a Composition relationship between a Node CI (node 1) and Running Software CI is pushed to SM, if you change the upstream CI of the relationship from node 1 to node 2 and then run a change population to populate this change, the Running Software CI will be removed in UCMDB.

**Workaround:**

It is recommended that you remove the running software in UCMDB and create a new one instead of directly replacing the container of the running software in SM. If you cannot avoid doing so, do the following:

After you change the upstream CI of the relationship from node 1 to node 2, do not directly run the change population. Follow these steps to avoid this issue:

1. Update the Running Software CI in SM (or simply save it to mark it as updated).
2. Run a Running Software CI change population. This will create node 2 (if it does not already exist in UCMDB) and a new Composition relationship between node 2 and this Running Software CI.
3. Run a delta population to synchronize the relationship change to UCMDB. The relationship between node 1 and the Running Software CI will be removed, and the new relationship created in step 2 will remain.

If you have run a delta population after changing the upstream CI of the relationship from node 1 to node 2, and as a result the Running Software CI has been removed in UCMDB, follow these steps to solve this issue:

1. Update the Running Software CI in SM (or simply save it to mark it as updated).
2. Run a Running Software CI change population. This will create the Running Software CI, node 2 (if it does not already exist in UCMDB) and a new Composition relationship between node 2 and this Running Software CI.

## Push Scheduling Recommendations

Push jobs are run using two main methods, the first of which is manually executing the push job and the second is scheduling the push job.

All push jobs can potentially produce a strain on the UCMDB and SM systems; HP recommends that you adhere to the following guidelines.

### **Scheduler time frames**

It is important for you to understand the function of the Scheduler “time frame” concept. Running push jobs creates an increase in system activity and may affect application responsiveness. In order to enable users to effectively interact with applications HP recommends the following guidelines:

In order to reduce system strain, schedule the UCMDB to SM push to run at non-peak usage hours, preferably when system usage is at a minimum.

### **Scheduler frequency**

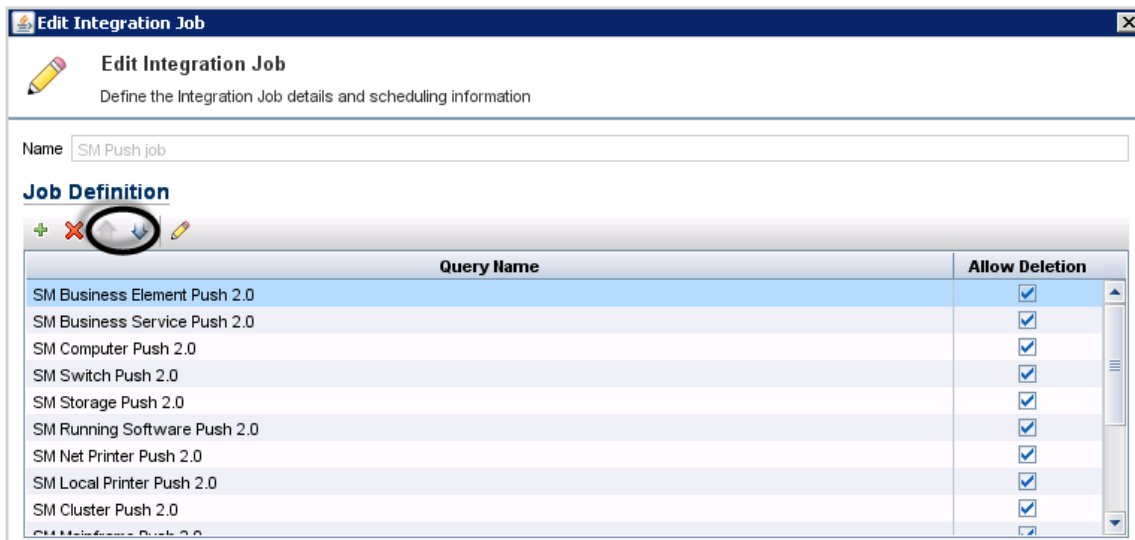
It is important to be aware of the business requirements when configuring the schedule frequency. The scheduler frequency depends on infrastructure environment changes that must be synchronized between UCMDB and SM.

Define the scheduling frequency based on the business requirements for consuming up-to-date CI information. Most implementations require a daily update. When scheduling small IT systems that are prone to frequent changes, the scheduling frequency may need to be increased.

### **Push Job dependencies**

UCMDB Push Jobs do not support dependencies between each other. Each “Push Job” is considered a separate task and users cannot define job dependencies. For example, that one job is dependent on another or upon completion before the next job is run.

It is important that both CI queries and their dependent Relationship queries exist in the same Job in order to avoid relationships not being pushed to Service Manager. You can change the position of each query in the list to define an appropriate execution sequence. See the following figure for an example.



## Push in Clustered Environments

A clustered SM environment is comprised of multiple servlets running in parallel with a load balancer that dispatches user requests to any available servlet. You must configure the UCMDB-SM integration to point to a specific servlet and not to the SM loadBalancer. In order to perform this, you must first create a dedicated web service listener.

This section includes:

- ["Dedicated Web Services" below](#)
- ["Step-by-Step Cluster Configuration Process" below](#)
- ["Connecting to Multiple SM Processes" on page 165](#)

### Dedicated Web Services

A Service Manager system configured for vertical or horizontal scaling uses a load balancer to redirect client connection requests to an available SM process. However, most Web Services clients cannot handle a redirect request and will fail if they use the SM load balancer as the endpoint URL.

HP recommends creating one or more SM processes dedicated to Web Services requests. The user must configure the relevant external web service clients to connect directly to the dedicated Service Manager processes.

### Step-by-Step Cluster Configuration Process

This section describes the steps to configure a cluster environment for the integration.

## How to Configure Web Clients

To configure the relevant external web clients:

1. Stop the Service Manager service.
2. Open the `sm.cfg` file, and create a dedicated SM process to listen for Web Services requests using the `-debugnode` parameter.

The following entries create a dedicated process listening on ports 13085 and 13445.

```
01 sm -httpPort:13080 -loadbalancer
02 sm -httpPort:13081 -httpsPort:13443
03 sm -httpPort:13083 -httpsPort:13444
04 sm -httpPort:13085 -httpsPort:13445 -debugnode
```

### Explanation

The code excerpt illustrates the various settings for each of the SM process listeners (web services) that enable SM clients to connect to the SM service.

Line 01 defines the load balancer port (13080).

Lines 02 and 03 define the SM ports to which non-dedicated SM clients are redirected by the SM load balancer.

Line 04 defines the debugnode port that is utilized by the dedicated SM clients.

### Note: Debugnode parameter

The debugnode parameter tells the SM load balancer not to forward any client connection requests to this Service Manager process. Only clients that directly connect to the process can access this port.

## How to Configure the Debugnode

To configure the debugnode:



1. Start the SM service.
2. Configure any external web service clients to connect directly to the SM processes running in debugnode. When performing an integration by using UCMDB, the Service Manager Adapter for SM should be configured to connect to the debugnode port.  
For example, for normal connections set the endpoint URL to:

`http://<fully qualified host name>:13085/SM/9/rest/<Service Name>`

and for SSL-encrypted connections set the URL to:

`https://<fully qualified host name>:13445/SM/9/rest/<Service Name>`

These clients may include UCMDB (for push purposes), Connect-It, and additional applications.

## Connecting to Multiple SM Processes

If you want to improve performance, you can connect to multiple Service Manager processes. The integration supports both Service Manager vertical and horizontal load balancer environments.

You can create more than one SM process that is dedicated to Web Services requests, and configure the **URL Override** field of the integration point with the dedicated SM processes. This field value (if any) overrides the Hostname/IP and Port settings.

The following is an example value of this field, which connects two SM processes:

`http://<fully qualified host name1>:13080/SM/9/rest;http://<fully qualified host name2>:13082/SM/9/rest`

## Initial Load Configurations

Before the configuration process can begin, you must first assess the amount of CI and relationship data to be transferred from UCMDB to SM and ascertain the iteration process that is required based on the volume.

You must first assess whether or not all of the data can be pushed in a single iteration. This is ascertained by the amount of data that is included in the push queries and the amount of time you have to push this data.

This section includes:

- ["Push Performance in a Single-Threaded Environment" below](#)
- ["Implementing Multi-Threading" on the next page](#)
- ["Push Performance in Multi-Threaded Environments" on page 168](#)
- ["Push Performance in Multiple SM Processes Environments" on page 168](#)
- ["How to Set up SM DEM Rules for Initial Loads" on page 169](#)

**Note:** The performance data presented in this document is based on tests that were performed at HP and is provided for reference only. The integration performance may significantly differ in your environment depending on your hardware configuration.

### Push Performance in a Single-Threaded Environment

The push of 22,500 UCMDB root CIs (roots in queries), relationships, or both in a single-threaded environment takes about an hour and is performed in a linear fashion. For more information, see the following table.

**Performance Data in a Single-Threaded Environment**

Number of root CIs/CI Relationships Pushed per Hour	Multi-Threading Settings in sm.properties
22,500	number.of.concurrent.sending.threads=1  min.objects.for.concurrent.sending=50  number.of.chunks.per.thread=3  recommended.min.cis.per.chunk=50

To view or edit the sm.properties file in UCMDB, navigate to **Data Flow Management > Adapter Management > ServiceManagerEnhancedAdapter9-x > Configuration Files > sm.properties.**

**Number of Root CIs and Relationships/22,500**

The push time (in hours) in any given environment is calculated as follows:

If the push of a single planned query has the potential of breaching the permitted time frame, the data must be divided into several queries. Each query must be pushed individually.

This query division is performed by creating several queries, each with different node conditions that enable data filtering. Once all queries are pushed for the first time, the Initial Load process is complete.

**Note: Applying node conditions**

When applying node conditions to the various SM Sync Queries, you must make sure that all of the information is included in the queries, so that all relevant data is copied to SM.

## Implementing Multi-Threading

In order to improve performance, the Service Manager adapter utilizes multiple threads to push CI and Relationship data to SM. The following section explains these settings and how to configure them for maximum performance.

The multi-threading configuration is defined in the `sm.properties` file on the UCMDB server. To view or edit the file in UCMDB, navigate to **Data Flow Management > Adapter Management > ServiceManagerEnhancedAdapter9-x > Configuration Files > sm.properties**.

The following are example multi-threading definitions in the `sm.properties` file:

```
01 number.of.concurrent.sending.threads=6
02 min.objects.for.concurrent.sending=50
03 number.of.chunks.per.thread=3
04 recommended.min.cis.per.chunk=50
```

**Explanation**

The code excerpt illustrates the relevant multi-threading settings on the UCMDB server.

- Line 01 defines the number of parallel threads UCMDB will open to SM for CI push. Setting this parameter to 1 disables multi-threading, while a values of 2 or higher enables multi-threading.
- Line 02 defines the minimum number of SM objects needed to use multiple threads as opposed to a single thread.
- Line 03 defines the number of chunks per thread. This number multiplied by the number of threads gives you the total number of CI data chunks.
- Line 04 defines the recommended minimum number of CIs per CI data chunk.

The total number of chunks = `number.of.chunks.per.thread * number.of.concurrent.sending.threads`

The integration implements a queue mechanism as follows:

The data passed from UCMDB to SM is divided into equal chunks, and these chunks are placed in a queue.

Each available thread pulls the next chunk from the queue until all threads are available. Once this process has completed, the push is complete.

The mechanism is designed to minimize the idle time of each thread. As each thread processes its chunk in parallel, some threads may finish before others, and it is inefficient for them to wait for each other.

**Caution: Defining too many threads**

It is ineffective to over-increase the number of threads as this causes the SM server to overload. In enterprise environments where the SM server processing the push data is very robust the number of threads can be increased to 10 and in some cases even 20; however, you must take into account that increasing the number of threads raises CPU usage on the SM server during push, which may reduce application performance.

**Push Performance in Multi-Threaded Environments**

The push of 60,000 UCMDB root CIs (roots in queries) and/or Relationships in an out-of-the-box multi-threaded environment takes about an hour and is performed in a linear fashion. See the following table.

**Performance Data in an Out-Of-The-Box Multi-Threaded Environment**

Number of Root CIs/Relationships Pushed per Hour	Multi-Threading Settings in sm.properties (Default)
60,000	number.of.concurrent.sending.threads=6 min.objects.for.concurrent.sending=50 number.of.chunks.per.thread=3 recommended.min.cis.per.chunk=50

The push time (in hours) in any given environment is calculated as follows:

**Number of Root CIs and Relationships/60,000**

**Push Performance in Multiple SM Processes Environments**

The Push of 160,000 UCMDB root CIs (roots in queries), relationships, or both in a multi-threaded environment with multiple SM processes takes about an hour and is performed in a linear fashion. For more information, see the following table.

**Performance Data in a Multiple SM Processes Environment**

<b>Number of root CIs/Relationships pushed per hour</b>	<b>SM processes</b>	<b>Multi-threading settings in sm.properties (default)</b>
160,000	2 server hosts, with each host running 4 processes	number.of.concurrent.sending.threads=60 min.objects.for.concurrent.sending=50 number.of.chunks.per.thread=3 recommended.min.cis.per.chunk=50 Data Push Chunk Size = 4000 (in Integration Settings in UCMDB)

For more information about defining multiple SM processes for the integration, see ["How to Create an Integration Point in UCMDB" on page 104](#).

The push time (in hours) in any given environment is calculated as follows:

**Number of Root CIs and Relationships/160,000****How to Set up SM DEM Rules for Initial Loads**

SM Discovered Event Manager Rules (DEM Rules) enable the user to define the appropriate action to take for each event type that is reported to SM.

Each CI and relationship record pushed from UCMDB to SM is analyzed against the existing SM records and open Change requests. SM rules define the appropriate action to be taken for each type of CI data update that is sent to SM.

To view or update the SM Discovered Event Manager Rules:

1. Log in to Service Manager as a system administrator.
2. Navigate to **Tailoring > Web Services > Discovered Event Manager Rules**.
3. Press **ENTER** or click the **Search** button.  
A list of all the Discovered Event Manager Rules is displayed. Each rule is usually linked to a CI Type or a subset of CIs of the same type.
4. Click the individual CI Discovered Event Manager Rule to view its details.

To set up DEM Rules for initial loads:

**Tip:** When performing an Initial Load, HP recommends setting the SM Discovered Event Manager

Rules to add newly reported CIs as described below. This minimizes the “noise” of an Initial Load, which could potentially create tens of thousands of Changes or Incidents.

For each of the Discovered Event Manager Rules, perform the following steps:

1. Select the relevant Discovered Event Manager Rule.
2. Go to the **Action if matching record does not exist** section, select the **Add the record** option.
3. In the **Action if record does not exist but unexpected data discovered** section, select the **Log Results and Update Record** option.
4. In the **Action if record is to be deleted** section, select the **Delete Record** option.
5. Save the Discovered Event Manager Rule record.

## How to Configure Differential or Delta Load DEM Rules

**Tip:** Once the “Initial Load” or “Data Load” of the CI data is completed, HP recommends applying Differential or Delta Load settings. These settings apply to all data loaded from UCMDB to SM.

These loads send only updates regarding modifications discovered in the IT infrastructure from UCMDB to SM.

To set up the SM DEM Rules for Differential or Delta Loads:

1. Log in to Service Manager as a system administrator.
2. Navigate to **Tailoring > Web Services > Discovered Event Manager Rules**.
3. Press Enter or click the **Search** button.  
A list of all the Discovered Event Manager Rules in SM is displayed.
4. For each of the Discovered Event Manager Rules, perform the following steps:
  - a. Select the relevant Discovered Event Manager Rule.
  - b. In the **Action if matching record does not exist** section, select the appropriate action required for each newly detected CI. If uncertain, select the **Add the record** option.
  - c. In the **Action if record does exist but unexpected data discovered** section, select the appropriate action for each CI that was modified, resulting in an unexpected or incorrect result. The recommended best practice is to select the **Open a Change** option.

- d. In the **Action if record is to be deleted** section, select the appropriate action required for each CI that was removed. The recommended best practice is to select the **Delete Record** option for CI Relationships, and select the **Update record to the selected status** option for CIs.
- e. Save the Discovered Event Manager Rule record.

## Fault Detection and Recovery for Push

Universal CMDB has provided a fault detection and recovery mechanism since version 9.05: individual CI failures no longer cause the entire push to fail, and you can review all failed CIs in the Universal CMDB studio and then re-push them.

### Duplicate logical.name issue

A typical fault you may encounter is the duplicate logical name issue, which is caused by the use of different unique key fields in Universal CMDB and Service Manager: CI logical.name in Service Manager is unique, and it usually maps to CI display label in Universal CMDB (which is not unique). HP recommends that you follow the following guidelines (listed from the highest to lowest priority) to resolve this issue:

- Make sure that each display label field value in UCMDB is unique;
- If uncertain of the above, in the adapter mapping configuration file avoid direct mapping between Universal CMDB display label and SM logical name;
- Map SM logical name to another Universal CMDB field that is unique;
- Add a prefix or suffix to UCMDB display label values;

**Note:** Out-of-the-box, SM logical name of Running Software is mapped with a prefix of DNS name:

```
<target_mapping datatype="STRING" name="CIIdentifier"
  value="SMPushFunctions.getCIIdentifier(Root['display_label'],Root.Node*.
  getAt('display_label'))"/>

public static String getCIIdentifier(String name, def arr){
    if( fIsEmpty(arr) ){ return name;}else{
        return covertArray2String(arr)+name;
    }
    return name;
}
```

- If you cannot do any of the above, you can use the UCMDB Fault Detection and Recovery mechanism together with the “Duplication Rule” setting of DEM rules as described in the following.

To set up DEM Rules for duplicate logical names:

1. Log in to Service Manager as a system administrator.
2. Navigate to **Tailoring > Web Services > Discovered Event Manager Rules > Duplication Rule** tab.
3. For each of the Discovered Event Manager Rules, perform the following steps:
  - a. Go to the **Action if logical name is duplicated** section, and select the **Return Error** option.
  - b. Save the Discovered Event Manager Rule record.

**Note:** After you run a push job, CIs with a duplicate logical name are reported as failed CIs with a duplicate name exception. You can review the failed CIs in the Universal CMDB studio, fix the errors by either changing the data in Universal CMDB or in the adapter mapping configuration file (XML and Groovy), and then re-push the failed CIs.

## How to Enable Lightweight Single Sign-On (LW-SSO) Configuration

You can enable LW-SSO for the integration so that users can directly view UCMDB CI records from the Service Manager web client by clicking the **View in UCMDB** button, without providing a UCMDB username and password.

**Note:** LW-SSO is not supported for the Service Manager Windows client.

To enable LW-SSO for the integration:

1. For each Service Manager user account that needs LW-SSO, create a user account in UCMDB with the same username. The passwords in the two systems can be different.
2. Enable LW-SSO in the Service Manager Web tier. For details, see the *Configure LW-SSO in the Service Manager Web tier* topic in the Service Manager help.
3. Enable LW-SSO in UCMDB. For details, see the *HP Universal CMDB Deployment Guide*.

## Frequently Asked Questions

This section provides answers to frequently asked questions about the UCMDB-SM integration.

This section includes:



- ["When Is a New CI Created in Service Manager?"](#) below
- ["Can I Analyze the Reason for a CI Deletion in SM?"](#) on the next page
- ["How Do I Monitor Relationship Changes Between UCMDB and SM?"](#) on the next page
- ["What Kinds of Relationships are Pushed from UCMDB to SM?"](#) on the next page
- ["What is a Root CI Node?"](#) on page 175
- ["What Is a Root Relationship?"](#) on page 175
- ["What is the "Virtual-Compound" Relationship Type Used in a UCMDB-SM Integration Query?"](#) on page 175
- ["When Do I Need the Population Feature?"](#) on page 176
- ["Can I Populate Physically Deleted CIs from SM to UCMDB?"](#) on page 176
- ["How Do I Keep the Outage Dependency Setting of a CI Relationship in SM?"](#) on page 176
- ["How Do I Create an XML Configuration File?"](#) on page 179
- ["How Do I Use the Load Fields Button to Add Multiple Managed Fields?"](#) on page 180
- ["What Is the Purpose of the <container> Element in the Population Configuration File \(smPopConf.xml\)?"](#) on page 180
- ["Can I Populate Sub-Item Deletions?"](#) on page 181
- ["What Happens if a Population Job Failed or Completed?"](#) on page 181

## When Is a New CI Created in Service Manager?

CIs are created in SM under the following circumstances:

- A CI is manually added to SM through the Configuration Management module.
- UCMDB reports a newly discovered CI according to the following:
  - When a new CI is reported and the Discovered Event Manager Rules are set to **Add the Record**.
  - When a new CI is reported, the Discovered Event Manager Rules are set to **Open an Incident** and the Incident has been closed.

- When a new CI is reported, the Discovered Event Manager Rules are set to **Open a Change** and the Change has been verified.

## Can I Analyze the Reason for a CI Deletion in SM?

No.

SM opens a change request on the deleted CI and includes the following information:

“Delete event for CI “CI Name” triggered by discovery”.

### **Workaround**

An SM change request does not contain a description of the reason for deletion, however it is possible to extract specific information about CI deletions from the UCMDB History Database. UCMDB data provides information about the user or the discovery pattern that initiated the CI deletion.

## How Do I Monitor Relationship Changes Between UCMDB and SM?

To understand the relationship change in SM, a distinction must be made between the various types of Relationship Changes:

- The second endpoint of the relationship has Changed, so instead of CI X being linked to CI Y through a relationship, now CI X is related to CI Z.
- An attribute of the relationship has changed.

The first type of Relationship change is supported by the UCMDB-SM integration, therefore, such “Relationship Changes” can either invoke CI relationship updates, or perform the creation of Incidents or Changes, which are then reviewed and monitored.

The second is also supported, but it is not covered out-of-the-box; you can configure the Universal CMDB query to expose such attributes of the relationship, and configure the Service Manager WSDL to expose the mapped field, and then configure the adapter mapping configuration in the XML and Groovy. However such a Relationship Attribute Change cannot perform the creation of Incidents or Changes, and only supports invoking CI relationship updates directly.

## What Kinds of Relationships are Pushed from UCMDB to SM?

Any kinds of relationships are pushed from UCMDB to SM under the following conditions:

- The relationship appears in a Push Query located in the **Service Manager > Push** folder in the UCMDB Query Manager.
- The relationship is named **Root** in the Push Query.
- The relationship is mapped to an appropriate target in SM in the UCMDB configuration files (XML and Groovy files).

The out-of-the-box relationships that are pushed from UCMDB to SM are relationships between two CIs such as:

- Between Business Services and Applications;
- Between Business Service and Host;
- Between an Application and a Network Component; or
- Between Host, Network Components and Printers.

## What is a Root CI Node?

A Root node is a TQL query node that represents the CI type that is created through push to SM from the TQL query structure. The rest of the TQL query structure contains information that can be incorporated within the Root CI type and is used to enrich the record in SM with additional information and or attributes.

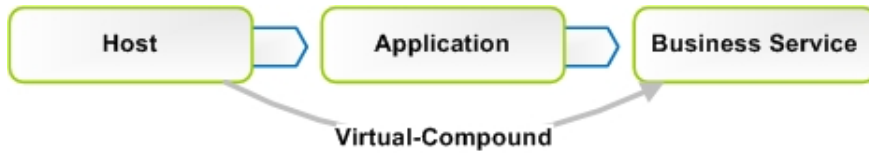
## What Is a Root Relationship?

A Root relationship is a relationship within a query and created in SM through push. It represents a relationship between two Root CIs. Only the relationships marked with Root are pushed to SM.

## What is the “Virtual-Compound” Relationship Type Used in a UCMDB-SM Integration Query?

When more than two UCMDB CI entities are connected in series, the “Virtual-Compound” represents the relationship between the first and last entities. This is a virtual relationship, as no physical representation exists.

The “Virtual-Compound” relationship type is a relationship that links two CI type entities that have a logical relationship. See the following figure.



### Explanation

The illustration shows an example of a Virtual-Compound relationship. The relationship in SM is created directly between the Host and the Business Service.

## When Do I Need the Population Feature?

You need the population feature under any of the following circumstances:

- You have done modeling in SM, especially when you are in the planning and design phases, and you want your models to be reflected in UCMDB;
- You want to implement the UCMDB-SM integration, however you have already invested in your SM CMDB and do not want to lose that investment;
- You want to continue to maintain some parts of the SM CMDB while maturing your UCMDB/Discovery implementation.

## Can I Populate Physically Deleted CIs from SM to UCMDB?

No.

Physical deletions of CIs are allowed in SM, but SM cannot get such “deletion changes” and the population feature will not synchronize such changes to UCMDB.

Physical deletions of CIs can be considered as exceptions, which occur only after you create CIs by mistake. Normally, you delete a CI by setting its status to something like **Retired/Consumed**. In case such CIs have been populated to UCMDB, it is your responsibility to remove them manually from UCMDB.

## How Do I Keep the Outage Dependency Setting of a CI Relationship in SM?

Out-of-the-box, CI relationships that are pushed from UCMDB to SM do not have outage dependency information by default. If you need such information, you can set the DEM rule of the CI Relationship WSDL as follows:

1. Log in to Service Manager as a system administrator.
2. Navigate to **Tailoring > Web Services > Discovered Event Manager Rules.**
3. Open the ucmdbRelationship record.
4. On the Rules tab, select **Add the record, and set dependency as true.**

Id:

Table Name:

Condition:

---

Rules
  Managed Fields
  Incident Customization
  Change Customization

Action if matching record does not exist

- Add the record
- Add the record, and set dependency as true
- Open a Change
- Open an Incident

This will set the Outage Dependency of each CI Relationship to true, and set the number of dependent downstream CIs to 1 (because UCMDB supports only one-to-one relationships).

If you want to set outage dependency only for some relationships (for example, if you want to configure outage dependency for relationships that start from Business Service), you can configure the adapter configuration file (XML) and WSDL definition; you can also configure outage dependency per relationship type (UCMDB query).

1. In the WSDL definition, expose fields **outage.dependency** and **outage.threshold**.

External Access Definition

Service Name:

Name:

Object Name:

---

Allowed Actions
  Expressions
  Fields
  RESTful

Field	Caption	Type
relationship.name	RelationshipName	
logical.name	ParentCI	
related.cis	ChildCIs	
relationship.type	RelationshipType	
relationship.subtype	RelationshipSubtype	
outage.dependency	OutageDependency	
outage.threshold	OutageThreshold	

2. In the XML file, set the exposed outage fields. For example, if you want to set the outage dependency to `true` and the threshold to `1` for Business Service relationships, you simply need to change the XML mapping file **SM Business Service Relations Push 2.0.xml**. In the XML mapping file, use the following `OutageDependency` and `OutageThreshold` settings:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
  <integration>
    <info>
      <source name="UCMDB" vendor="HP" version="10.20"/>
      <target name="SM" vendor="HP" version="9.40"/>
    </info>
    <import>
      <scriptId path="mappings.scripts.SMPushFunctions"/>
    </import>
    <target_entities>
      <source_instance root-element-name="Root_directly" query-name="EA_SM
Business Service Relations Push">
        <target_entity name="Relationship">
          <target_mapping datatype="STRING" name="RelationshipType"
value="SMPushFunctions.getDisplayName(Root_directly['element_
type'],ClassModel)"/>
          <target_mapping datatype="STRING" name="ParentCI"
value="SMPushFunctions.getEndId(OutputCI.getExternalId()).getEnd1Id()"/>
          <target_mapping datatype="STRING_LIST" name="ChildCIs"
value="[SMPushFunctions.getEndId(OutputCI.getExternalId()).getEnd2Id()]">
            <target_mapping datatype="BOOLEAN" name="OutageDependency"
value="true"/>
            <target_mapping datatype="NUMBER" name="OutageThreshold"
```

```


value="1"/>
        </target_entity>
    </source_instance>
</target_entities>
</integration>

```

## How Do I Create an XML Configuration File?

You create an XML configuration file in Adapter Management. You can copy the content of an existing XML configuration file to the new file and then make necessary edits.

To create an XML configuration file, follow these steps:

1. Log in to UCMDB as an administrator.
2. Navigate to **Data Flow Management > Adapter Management. ServiceManagerEnhancedAdapter9-x > Configuration Files.**
3. Click the **Create new resource** icon .
4. Select **New Configuration File.**
5. Enter a name for the file. The file name should use this format: *<AdapterID>/mappings/<Synch Type>/<filename>*. For example: *ServiceManagerEnhancedAdapter9-x/mappings/push/SM Computer Push.xml*.
6. Click **OK.**

UCMDB creates the new XML configuration file in the Configuration Files folder of the adapter.

7. Copy the content of an existing XML configuration file to the new file.
8. Make necessary edits to the new file.

### **Caution: Invalid XML**

When removing XML elements from an XML file, keep in mind that the remaining elements must constitute a valid XML file, which will be used to translate the UCMDB Query Definition.

## How Do I Use the Load Fields Button to Add Multiple Managed Fields?

Service Manager stores a list of managed fields in the **ucmdbIntegration** web service, which consists of a number of web services objects. You can add more managed fields to DEM Rules so that Service Manager can monitor changes in more CI attributes in UCMDB and trigger the actions defined in relevant DEM Rules.

You can manually add managed fields that are exposed in associated WSDL definitions to DEM Rules; however, you can use the **Load Fields** button to automatically (and therefore correctly) add managed fields to DEM Rules.

1. Click the **Managed Fields** tab of the DEM Rule.
2. Click the **Load Fields** button.
3. If the table (in the Table Name field) of the DEM rule record has only one WSDL definition associated to it, all fields exposed in the WSDL definition are immediately added to the Managed Fields list.  
A message is displayed: <XX> new fields loaded.
4. If the table has more than one WSDL definition associated to it, the Managed Fields Importing wizard opens, and a list of WSDL definitions (ucmdbIntegration web service objects) is displayed.
  - a. Select one or more objects, and click **Next**. All new fields that can be added from the selected web service objects are displayed.
  - b. If you want to add all of the fields, click **Finish**; if you want to ignore some of them, change their Action value from **Add to Ignore**, and then click **Finish**.  
A message is displayed: <XX> new fields loaded.
5. Save the DEM Rule record.

## What Is the Purpose of the <container> Element in the Population Configuration File (smPopConf.xml)?

Out-of-the-box, the smPopConf.xml file has a container element.

```
<tql name="SM RunningSoftware Population 2.0" citype="running_software">
  <request type="Retrieve" dataType="ci"
    resourceCollectionName="ucmdbRunningSoftwares"
    resourceName="ucmdbRunningSoftware"
    basicQueryCondition="type='runningsoftware'"
    fullQueryCondition="istatus~='Retired/Consumed'">
```



```

    changedCreationQueryCondition="created.by.date>='{fromDate}' and
    istatus~='Retired/Consumed';"
    changedUpdateQueryCondition="created.by.date<='{fromDate}' and
    devicemodtime>='{fromDate}' and istatus~='Retired/Consumed';"
    changedDeletionQueryCondition="devicemodtime>='{fromDate}' and
    istatus='Retired/Consumed';"/>
    <container tql="SM Computer Population 2.0"
    keyFields="CIIdentifier"
    linkTql="SM Computer Composition Software 2.0"
    linkRetrieveCondition="downstreamci.logical.name='$$$' and
    upstreamci.type='computer' and
    downstreamci.type='runningsoftware' and
    relationship.subtype='Composition';"
    linkRetrieveConditionKey="CIIdentifier"
    linkValueFields="upstreamci.logical.name"/>
</tql>

```

- In UCMDB, RunningSoftware CIs must exist together with a Root Container (Node); however, Service Manager allows RunningSoftware CIs without a Node.
- The integration adapter synchronizes CIs and relationships separately; when populating a RunningSoftware CI, the integration does not check to see if a relationship exists between the CI and a Node.

When you use the <container> element, the integration populates RunningSoftware CIs together with a container.

## Can I Populate Sub-Item Deletions?

Yes.

Service Manager and UCMDB store CI information in different data structures, and therefore one SM CI may be synchronized to UCMDB as several CIs. For example, during population, an SM computer CI record is synchronized to a Node CI in UCMDB, and the computer CI's attributes to CIs such as IP, Interface, Location, etc (which are referred to as sub-items of the Node CI). In this case, the Node CI is the root CI.

The integration allows you to populate sub-item deletions to UCMDB. For example, if you delete the IP Address attribute value of a computer, the corresponding IP CI record in UCMDB will be deleted too.

## What Happens if a Population Job Failed or Completed?

### **When a population job failed**

The failure prevents the remaining population tasks from running. The next job run will start from the last Success time. If pagination occurs (that is, the tasks are divided into multiple pages), the tasks will run again and again within the first page from the last "Success" time (once the end of the first page is reached, no new tasks will be executed).

#### **When a population job completed**

When the status of a job is "completed" (but not "completed successfully"), warnings occurred. A warning does not prevent the remaining population tasks from running. The next job run will run all tasks again starting from the last Success time. If pagination occurs (the tasks are divided into multiple pages), the tasks on all pages will be re-run (including those that successfully completed last time).

## Tailoring the Integration

You can tailor the UCMDB-SM integration to meet your business needs by adding or removing managed CI types, attributes, and relationship types. This chapter describes the integration architecture and tailoring options for data push, population, and federation.

This section includes:

- ["Integration Architecture" below](#)
- ["Integration Tailoring Options" on page 197](#)

## Integration Architecture

Before you tailor the integration, you should understand how the following components of the out-of-the-box integration work.

- ["Integration Class Model" on the next page](#)
- ["Integration Queries" on the next page](#)
- ["Service Manager Web Services" on page 188](#)
- ["Service Manager Reconciliation Rules" on page 193](#)
- ["Service Manager Discovery Event Manager Rules" on page 195](#)

## Integration Class Model

UCMDB 9.x or later no longer uses a private class model of CI types to manage integration CIs, as was required in prior versions. Instead, the integration uses the standard UCMDB managed objects and maps them to Service Manager CI types and attributes with queries and transformation files.

## Integration Queries

This section describes out-of-the-box queries used for data push, Actual State, and population.

- ["Queries for Push" below](#)
- ["Queries for Actual State" on page 186](#)
- ["Queries for Population" on page 187](#)
- ["Query Requirements" on page 188](#)

### Queries for Push

For the push feature, the integration uses a collection of queries to gather CI attribute information from Universal CMDB and send it to the Service Manager system.

To access the out-of-the-box data push queries, navigate to **Modeling > Modeling Studio**, select **Queries** for Resource Type, and then navigate to the **Root > Integration > Service Manager > Push** folder.

If you want to change what CI types, Relationship types or attributes are part of the integration, you must also edit the integration queries to support your updated CI types, CI Relationship types and attributes.

Query name	Description
SM Local Printer Push 2.0	This query gathers CI attributes from printer CIs. It also gathers related CI attributes from the Node CI type.
SM Net Printer Push 2.0	This query gathers CI attributes from Node CI types whose NodeRole contains "printer." It also gathers related CI attributes from the following CI types through containers and links: IPAddress, Interface, CPU, FileSystem, DiskDevice, and Location.

Query name	Description
SM Mainframe Push 2.0	<p>This query gathers CI attributes from the following Node CI types: Mainframe Logical Partition, and Mainframe CPC.</p> <p>It also gathers related CI attributes from the following CI types through containers and links: IPAddress, Interface, CPU, FileSystem, DiskDevice, and Location.</p>
SM Mobile Device Push 2.0	<p>This query gathers CI attributes from Node CI types whose NodeRole contains “pda_handheld.”</p> <p>It also gathers related CI attributes from the following CI types through containers and links: IPAddress, Interface, CPU, FileSystem, DiskDevice, and Location.</p>
SM Network Component Push 2.0	<p>This query gathers CI attributes from Node CI types whose NodeRole contains router, adsl_modem, appletalk_gateway, bandwidth_manager, cable_model, csu_dsu, ethernet, fddi, firewall, hub, kvm_switch, load_balancer, multicast_enabled_router, nat_router, token_ring, undefined_network_component, voice_gateway, voice_switch, or vpn_gateway.</p> <p>It also gathers related CI attributes from the following CI types through containers and links: IPAddress, Interface, CPU, FileSystem, DiskDevice, and Location.</p>
SM Cluster Push 2.0	<p>This query gathers CI attributes from the following Node CI type: ClusterResourceGroup.</p> <p>It also gathers related CI attributes from the following CI types through containers and links: IPAddress, Interface, CPU, FileSystem, DiskDevice, Location, and Cluster.</p>
SM Computer Push 2.0	<p>This query gathers CI attributes from the node CI type with NodeRole containing “desktop”, “server”, “virtualized_system” or not set. It also gathers related CI attributes from the following CI types through containers and links: IPAddress, Interface, CPU, FileSystem, DiskDevice, and Location.</p>
SM Storage Push 2.0	<p>This query gathers CI attributes from the Node CI type whose NodeRole contains san_switch, san_gateway, san_router, or whose Display Name is Storage Array.</p> <p>It also gathers related CI attributes from the following CI types through containers and links: IPAddress, Interface, CPU, FileSystem, DiskDevice, and Location.</p>
SM Running Software Push 2.0	<p>This query gathers CI attributes from Running Software CIs.</p>

Query name	Description
SM Switch Push 2.0	<p>This query gathers CI attributes from the Node CI type whose NodeRole contains atm_switch, frame_relay_switch, or lan_switch.</p> <p>It also gathers related CI attributes from the following CI types through containers and links: IPAddress, Interface, CPU, FileSystem, DiskDevice, and Location.</p>
SM Service Element Push 2.0	<p>This query gathers CI attributes from the Business Element CI type.</p>
SM Business Service Push 2.0	<p>This query gathers CI attributes from the Business Service CI type.</p>
SM Layer2 Topology Relations Push 2.0	<p>This query gathers relationships between the following components: Two or more nodes. The query includes compound relationships because the relationships can extend through a group.</p>
SM Business Service Relations Push 2.0	<p>This query gathers relationships between the following components:</p> <ul style="list-style-type: none"> <li>• Business Service and Running Software CIs</li> <li>• Business Service and Node CIs</li> <li>• Two or more Business Services</li> </ul> <p>The query includes compound relationships because the relationships can extend through a group.</p>
SM CRG Relations Push 2.0	<p>This query gathers relationships between the following components: Node and Cluster Resource Group CIs.</p> <p>The query includes compound relationships because the relationships can extend through a group.</p>
SM Node Relations Push 2.0	<p>This query gathers relationships between the following components:</p> <ul style="list-style-type: none"> <li>• Node and Printer CIs</li> <li>• Node and RunningSoftware CIs</li> </ul> <p>The root class of the relationship is composition.</p>

**Note:** The Service Manager Enhanced Generic Adapter was introduced in UCMDB version 10.20. This adapter can coexist with an old XSLT adapter. In order to distinguish its queries from those of an XSLT adapter, all queries of the enhanced adapter are suffixed with **2.0**.

## Queries for Actual State

Out-of-the-box, the queries in the following table are used for retrieving CI information from UCMDB to the Actual State section of the Service Manager Configuration Item (CI) form. Service Manager retrieves CI Actual State information by calling a UCMDB web service that retrieves CI data according to these queries.

The queries are located in the **Integration > SM Query** folder in the UCMDB Modeling Studio.

Query name	Description
localPrinterExtendedData	This query gathers real-time extended information from Printer CIs in UCMDB.
applicationExtendedData	This query gathers real-time extended information from RunningSoftware CIs in UCMDB.
businessServiceExtendedData	This query gathers real-time extended information from business service CIs in UCMDB.
hostExtendedData	This query gathers real-time extended information (such as Asset, Party, Location, LogicalVolume, WindowsService, Printer, InstalledSoftware, FileSystem, IPAddress, Interface, DiskDevice, and Cpu) from the node CI type in UCMDB.

## Queries for Federation

Due to a technical limitation of the Generic Adapter framework, queries that describe the relationships between UCMDB CI types and the federated CI types (Incident, Problem, and RFC for Service Manager) are required. There are approximately 300 queries that are defined for federation in the **Integration > Service Manager > Federation** folder in the Modeling Studio, so that you do not have to develop them by yourself for the out-of-the-box CI Types in UCMDB. Usually, you do not need to modify these queries; instead, you only need to define new queries for your own custom CI types in UCMDB.

For information on how to define queries for federation, see the *Achieving Data Federation Using the Generic Adapter* section in the *HP Universal CMDB Developer Reference Guide*.

**Note:** This technical limitation was introduced in UCMDB 10.20, and might be fixed in a future

release.

## Queries for Population

For CI/CI Relationship population, the integration uses the following queries to save CI/CI Relationship attribute information to UCMDB.

Query Name	Description
SM Business Service Population 2.0	Defines the CI store structure of business service CIs.
SM Business Application Population 2.0	Defines the CI store structure of Application Service CIs.
SM Infrastructure Service Population 2.0	Defines the CI store structure of Infrastructure Service CIs.
SM Running Software Population 2.0	Defines the CI store structure of running software CIs.
SM Computer Population 2.0	Defines the CI store structure of computer CIs.
SM CLIP Down Time Population 2.0	Defines the CI store structure of ScheduledDowntime CIs.
SM Biz Containment Biz 2.0	This query defines the CI store structure of CI relationships in which a bizservice CI contains another.
SM Biz Usage Biz 2.0	Defines the CI store structure of CI relationships in which a bizservice CI uses another.
SM Biz Containment Computer 2.0	This query defines the CI store structure of CI relationships in which a bizservice CI contains a computer CI.
SM Biz Containment Software 2.0	Defines the CI store structure of CI relationships in which a bizservice CI contains a RunningSoftware CI.
SM Biz Usage Computer 2.0	Defines the CI store structure of CI relationships in which a bizservice CI uses a computer CI.
SM Biz Usage Software 2.0	Defines the CI store structure of CI relationships in which a bizservice CI uses a RunningSoftware CI.

Query Name	Description
SM Computer Connects Computer 2.0	Defines the CI store structure of CI relationships in which a computer CI connects to another.
SM Computer Composition Software 2.0	Defines the CI store structure of CI relationships in which a RunningSoftware CI is contained within a computer CI and the RunningSoftware CI cannot exist without the container.
SM CI Connection Down Time CI 2.0	Defines the CI store structure of CI relationships in which a ScheduledDowntime CI connects to an affected CI.

## Query Requirements

The integration requires that any custom queries you create meet certain formatting conditions. Any queries that you want to include in the integration must meet these conditions:

- To query CIs, a query must contain one CI type labeled Root. The Root node is the main CI that UCMDB synchronizes. All other CIs are contained CIs of the Root CI.
- To query relationships, a query must contain one or more relationships labeled Root.
- A query must contain only the Root CI and CIs that are directly connected to it. The Root CI is always the top node of the query hierarchy.
- A query layout cannot have cycles.
- If a query synchronizing relationships has cardinality, it must be cardinality 1...\*. Additional cardinality entries must have an OR condition between them.
- If you want the integration to only synchronize specific CIs, you must configure the condition on the query to filter such CIs.

## Service Manager Web Services

Service Manager uses web services messages to get and receive CI information from your UCMDB system. Out-of-the-box, UCMDB sends more CI attribute information than the Service Manager system actually manages. Service Manager users can view all of the CI attribute information the UCMDB system sends from the Actual State section of the CI record.

Service Manager publishes several web services for use by the UCMDB-SM integration. The UCMDB system uses the web services to map UCMDB CI types and CI attributes to web services objects that the



Service Manager system recognizes. You can add UCMDB CI types or CI attributes that you want Service Manager to manage by using the Visual Mapping tool, and the tool will automatically update one or more of these web services to define web service objects.

## Managed Fields

**Note:** Managed fields are used only for the data push feature.

A Service Manager managed field is a field where the system compares the CI attribute value in the incoming UCMDB web services message to the value in a Service Manager CI record. If the values in the web services message do not match those in the CI record, Service Manager runs a Discovery Event Manager (DEM) rule to determine what action to take. The DEM rule determines which of the fields that are published as web services objects are fields managed by the integration. Only value changes in managed fields trigger the DEM rule.

The **ucmdbIntegration** web service consists of a set of web services objects, each of which defines a list of web service fields. Out-of-the-box, the integration uses only part of them (see the [Mappings Between Service Manager Web service Objects, Tables, and DEM Rules](#) table), some of them (along with their relevant DEM Rules) have been deprecated (see the [Deprecated ucmdbIntegration Web Service Objects for Data Push](#) table), and some are used for population or federation (see the [ucmdbIntegration Web Service Objects Used for Population or Federation](#) table).

### Mappings between Service Manager Web Service Objects, Tables, and DEM Rules

This web service object	Publishes fields from this table	And uses this DEM rule ID
Relationship	cirelationship	ucmdbRelationship
ucmdbRunningSoftware	device	ucmdbRunningSoftware
ucmdbBusinessService	joinbizservice	ucmdbBusinessService
ucmdbNode	joinnode	ucmdbNode

### Deprecated ucmdbIntegration Web Service Objects for Data Push

This web service object	Publishes fields from this table	Recommended replacement (object)
ucmdbApplication	device	ucmdbRunningSoftware
ucmdbComputer	ucmdbComputer	ucmdbNode
UcmdbDevice	device	ucmdbRunningSoftware
ucmdbNetwork	joinnetworkcomponents	ucmdbNode
ucmdbPrinter	joinofficeelectronics	ucmdbNode

**ucmdbIntegration Web Service Objects Used for Population or Federation**

This web service object	Publishes fields from this table	And is used for	Requires a DEM Rule?
cirelationship1to1	cirelationship1to1	Population	No
ucmdbIDPushBack	device	Population	No
UcldbChange	cm3r	Federation	No
UcldbChangeTask	cm3t	Federation	No
UcldbIncident	probsummary	Federation	No
UcldbProblem	rootcause	Federation	No

The following sections list the fields published as web services objects used for data push (see the [Mappings between Service Manager web service objects, tables, and DEM rules](#) table) and indicate whether or not they are managed fields in an out-of-the-box Service Manager system. You can use this reference to determine if you need to publish a field as a web service object, and also if you need to create a DEM rule for the object.

**Object Name: Relationship**

Service Manager publishes the following fields from the cirelationship table:

**Web Service and Managed Fields of the Relationship Object**

Field Published as Web Service Object	Caption Used in Web Service Messages	Is the Field a Managed Field?
relationship.name	RelationshipName	
logical.name	ParentCI	
related.cis	ChildCIs	Yes
relationship.subtype	RelationshipSubtype	

**Object Name: ucldbRunningSoftware**

Service Manager publishes the following fields from the device table:

**Web Service and Managed Fields of the ucldbRunningSoftware Object**

Field Published as web service object	Caption Used in Web Service Messages	Is the Field a Managed Field?
ucldb.id	UCMDBid	

**Web Service and Managed Fields of the ucmdbRunningSoftware Object, continued**

Field Published as web service object	Caption Used in Web Service Messages	Is the Field a Managed Field?
ci.name	ApplicationName	Yes
type	Type	
subtype	Subtype	
company	CompanyId	
logical.name	CIIdentifier	Yes
product.version	ProductVersion	
vendor	Vendor	
version	Version	
id <sup>1</sup>	CIName	

**Object Name: ucmdbBusinessService**

Service Manager publishes the following fields from the joinbizservice table:

**Web Service and Managed Fields of the ucmdbBusinessService Object**

Field Published as Web Service Object	Caption Used in Web Service Messages	Is the Field a Managed Field?
ucmdb.id	UCMDBId	
ci.name	ServiceName	Yes
type	Type	
subtype	Subtype	
company	CustomerId	
logical.name	CIIdentifier	Yes
vendor	ServiceProvider	
id <sup>2</sup>	CIName	

<sup>1</sup>This attribute is used only for the population feature.

<sup>2</sup>This attribute is used only for the population feature.

**Object Name: ucmdbNode**

Service Manager publishes the following fields from the joinnode table.

**Web Service and Managed Fields of the ucmdbNode Object**

<b>Field Published as Web Service Object</b>	<b>Caption Used in Web Service Messages</b>	<b>Is the Field a Managed Field?</b>
ucmdb.id	UCMDBId	
type	Type	
subtype	Subtype	
company	CustomerId	
logical.name	CIIdentifier	Yes
default.gateway	DefaultGateway	Yes
network.name	DNSName	Yes
building	Building	Yes
room	Room	Yes
floor	Floor	Yes
location	Location	
addIIPAddr[addIIPAddress]	AddIIPAddress	Yes
addIIPAddr[addSubnet]	AddSubnet	Yes
addMacAddress	AddMacAddress	Yes
bios.id	BIOSId	Yes
operating.system	OS	Yes
os.version	OSVersion	Yes
physical.mem.total	PhysicalMemory	Yes
serial.no.	SerialNo	
vendor	Vendor	
cpu[cpu.id]	CpuID	
cpu[cpu.name]	CpuName	

**Web Service and Managed Fields of the ucmdbNode Object, continued**

Field Published as Web Service Object	Caption Used in Web Service Messages	Is the Field a Managed Field?
cpu[cpu.clock.speed]	CpuClockSpeed	
file.system[mount.point]	MountPoint	
file.system[disk.type]	DiskType	
file.system[file.system.type]	FilesystemType	
file.system[disk.size]	DiskSize	
asset.tag	AssetTag	
machine.name	HostName	Yes
disk.device[model.name]	ModelName	
disk.device[disk.vendor]	DiskVendor	
disk.device[disk.name]	DiskName	
id <sup>1</sup>	CIName	
isVisualization	IsVisualization	
istatus	AssetStatus	

## Service Manager Reconciliation Rules

Service Manager reconciliation rules allow the integration to identify CI records in your Service Manager system that match CIs in your UCMDB system. Service Manager attempts to reconcile CI records with every push of CI attributes from your UCMDB system. The integration uses the following workflow to match UCMDB CIs with Service Manager CIs.

1. The UCMDB system sends a web service message containing the latest CI attribute data to Service Manager.
2. Service Manager scans the web service message for the CI ucmdb.id value.

**Note:** Out-of-the-box, Service Manager does not display the ucmdb.id field on CI record forms

<sup>1</sup>This attribute is used only for the population feature.

to prevent users from changing the value. If you want to add this value to your forms, you can find the `ucmdb.id` field defined in the **device** table. HP recommends that you make this a read-only field.

3. Service Manager searches for an existing CI record that has the same `ucmdb.id` value.
4. If Service Manager finds a CI with a matching `ucmdb.id` value, no reconciliation is needed. Service Manager compares the UCMDB CI attributes to the Service Manager managed fields and runs the appropriate Discovery Event Manager (DEM) rules as needed.
5. If Service Manager cannot find a CI with a matching `ucmdb.id` value, it runs the reconciliation rules.
6. Service Manager searches for an existing CI record with the same reconciliation field values.
7. If Service Manager finds a CI with a matching reconciliation field value, it updates the CI record with the `ucmdb.id` value of the matching UCMDB CI. Service Manager compares the UCMDB CI attributes to the Service Manager managed fields and runs the appropriate DEM rule as needed.
8. If Service Manager cannot find a CI with a matching reconciliation field value, it runs the DEM rule for "Action if matching record does not exist." Out-of-the-box, the DEM rule has Service Manager create a new CI record. Service Manager creates the CI record using the `ucmdb.id` value of the incoming UCMDB CI.

## Performance Implications

Because Service Manager attempts to reconcile CIs with every push, the number of reconciliation fields you have will affect the integration's performance. The more reconciliation rules you have, the more searches Service Manager must perform to match CIs. To improve the performance of reconciliation searches, you should choose reconciliation fields that are unique keys of the underlying Service Manager table. For example, if you want to reconcile CI records in the **device** table, use the `logical.name` field as a reconciliation field because it is a unique key. For information about how to create a reconciliation rule, see ["How to Add DEM Reconciliation Rules" on page 203](#).

## Dependence on DEM Rules

Service Manager uses the **Action if matching record does not exist** DEM rule whenever it cannot reconcile CIs. You must review the DEM settings and decide if they meet your business standards prior to the initial push of CIs from UCMDB to Service Manager. For example, you can have Service Manager create a change request for every CI in the initial CI push by selecting the **Open a Change** option.

## Service Manager Discovery Event Manager Rules

You only need to create Discovery Event Manager (DEM) rules if you want to accomplish any of the following custom actions:

- ["Change the Conditions Under Which a DEM Rule Runs" below](#)
- ["Change the Action the DEM Rule Takes" below](#)
- ["Create Custom JavaScript to Open Change or Incident Records" on the next page](#)

### Change the Conditions Under Which a DEM Rule Runs

Service Manager will run a DEM rule only if the condition field evaluates to true. Out-of-the-box, no DEM rule has a condition statement that restricts when the rule runs, and all the integration DEM rules will always run by default.

You can update a DEM rule's condition statements if you want to restrict when Service Manager runs your DEM rules. For example, adding the following condition to the ucmdbNode DEM rule restricts the rule to desktop CIs.

```
subtype in $L.file="Desktop"
```

You can also use the condition field to create multiple DEM rules that apply to the same table name. For example, the following DEM rules both apply to the joinnode table.

#### DEM rules using different conditions to affect the same table

DEM rule Id	Table Name	Condition
ucmdbNode	joinnode	subtype in \$L.file!="Desktop"
ucmdbDesktop	joinnode	subtype in \$L.file="Desktop"

Typically, you will only need to add conditions if your business processes require the integration to take different actions with certain CI types or SLAs.

### Change the Action the DEM Rule Takes

Out-of-the-box, the integration DEM rules take the following actions:

- Add a CI record when the UCMDB data does not match an existing Service Manager CI record
- Open a Change or log results and update a CI record when the UCMDB CI attribute data does not match the CI attribute data in the Service Manager CI record
- Delete a CI record when the UCMDB data specifies that the CI has been deleted

You can change the integration DEM rules to meet your business processes. For example, you could use the ucldbNode DEM rule to open a change when the integration finds a non-desktop CI with unexpected data, and use the ucldbDesktop DEM rule to log results and update the record when the integration finds a desktop CI with unexpected data.

**Caution:** If you want to use the Change Management verification and Change Management validation features of the integration, your DEM rules must use the **Open a Change** option for the “Action if record exists but unexpected data discovered” event.

### Create Custom JavaScript to Open Change or Incident Records

Service Manager uses the **discoveryEvent** JavaScript to create CI names and to set the values of required fields when opening change or incident records. Out-of-the-box, the script uses the following default values.

## Default values to create a new CI

You can update the createCIName and populateNewCI functions to set the following CI values.

#### Default values used to create a new CI

CI attribute	Default value defined in discoveryEvent
record.logical_name	System generated ID number
record.assignment	AUTO
record.istatus	Installed
record.os_name	Value in record.operating_system

## Default values to create a new change

You can update the populateChange function to set the following change values.



**Default values used to create a new change**

CI attribute	Default value defined in discoveryEvent
change.category	Unplanned Change
change.reason	Value in reason
change.initial_impact	3
change.severity	3
change.coordinator	Change.Coordinator
change.requested_by	discovery
change.status	initial

## Default values to create a new incident

You can update the populateIncident function to set the following incident values.

**Default values used to create a new incident**

CI attribute	Default value defined in discoveryEvent
incident.category	incident
incident.subcategory	hardware
incident.product_type	missing or stolen
incident.assignment	Hardware
incident.initial_impact	3
incident.severity	3
incident.logical.name	Value of id
incident.site_categry	C
incident.contact_name	ANALYST, INCIDENT
incident.affected_item	MyDevices

## Integration Tailoring Options

The integration offers the following tailoring options:

- ["How to Update the Integration Adapter Configuration File \(sm.properties\)" below](#)
- ["How to Add DEM Reconciliation Rules" on page 203](#)
- ["How to Add Discovery Event Manager Rules" on page 205](#)
- ["How to Add a CI Attribute to the Integration for Data Push" on page 210](#)
- ["How to Add a CI Type to the Integration for Data Push" on page 224](#)
- ["How to Add a CI Relationship Type to the Integration for Data Push" on page 241](#)
- ["How to Add a Custom Query to an Integration Job" on page 248](#)
- ["How to Add a CI Type, Attribute or Relationship Type to the Integration for Population" on page 249](#)
- ["How to Enable or Disable UCMDB ID Pushback for a CI Type " on page 249](#)
- ["How to Add an Attribute of a Supported CI Type for Federation" on page 251](#)

## How to Update the Integration Adapter Configuration File (sm.properties)

The integration uses a properties file (sm.properties) as a configuration file of the adapter. Out-of-the-box, this file has been set up based on best practices, so usually you can keep the default parameter values. Optionally, you can update the parameter values to better suit your needs.

To update the sm.properties file:

1. Log in to UCMDB as an administrator.
2. Navigate to **Data Flow Management > Adapter Management > ServiceManagerEnhancedAdapter9-x > Configuration Files.**
3. Click the properties configuration file: sm.properties.
4. Update the parameter values as needed. For a list of the parameters, see the following table.

#if true, will use globalId instead of ucmbld in the SM integration.

use.global.id=true

# Type 0, the feature will be disabled

```
# Type 1, the enum type will expand to "{value}"  
  
# Type 2, the enum type will expand to "{index}-{value}"  
  
type.of.expand.enum=2  
  
#if false, will use the type directly instead of label of the type.  
  
use.type.label=true  
  
#parameter for populate data from service manager  
  
pop.pagination.switch=on  
  
pop.pagination.recordcount=1000  
  
pop.createci.key=sm_id  
  
# the property used as ucmdb id key for population in xslt files, it is treated as "global_id" by  
# adapter if is.global.id is true, otherwise it is treated as CMDB ID.  
  
pop.ucmdb.id.key=ucmdb_id  
  
#SM web service for push back uCMDB ID  
  
ucmdbid.pushback.request=UpdateucmdbIDPushBackRequest  
  
ucmdbid.pushback.xslt=ucmdbid_pushback.xslt  
  
#whether checking the soap connections to SM instances before running the job  
  
check.sm.connections=false  
  
#the external web service interface type. restful or soap  
  
connector.type=restful
```

5.

6.

7.

**Parameters in the sm.properties file**

Parameter	Default value	Comment
timeout.minutes	10	The integration connection timeout value (in minutes)
number.of.concurrent.sending.threads	6	<p>The number of concurrent threads used for the data push feature</p> <ul style="list-style-type: none"> <li>■ 1: Disabled</li> <li>■ 2 or higher: Enabled</li> </ul> <p><b>Note:</b> If you are connecting to multiple Service Manager instances to improve the CI data push performance (see the URL Override configuration in "<a href="#">How to Create an Integration Point in UCMDB</a>" on <a href="#">page 104</a>), you are recommended to increase this value for optimized performance. For example, set it to 12 if you are connecting to two Service Manager instances.</p>
min.objects.for.concurrent.sending	50	<p>The minimum number of Service Manager objects that is required to use concurrent sending instead of single thread sending</p> <p><b>Note:</b> It is used for the push feature.</p>
number.of.chunks.per.thread	3	<p>The number of chunks per thread used for the push feature</p> <p>Total of number of chunks = number.of.chunks.per.thread * number.of.concurrent.sending.threads</p>

8.

**Parameters in the sm.properties file, continued**

Parameter	Default value	Comment
recommended.min.cis.per.chunk	50	<p>The minimum allowed number of CIs per CI data chunk. This value is used to mitigate the risk of making your data chunks too small.</p> <p>For example, suppose your system uses the following settings:</p> <ul style="list-style-type: none"> <li>■ number.of.chunks.per.thread=3</li> <li>■ number.of.concurrent.sending.threads=3</li> <li>■ recommended.min.cis.per.chunk=50</li> </ul> <p>When pushing 1000 CIs, the number of CIs in each chunk is calculated as <math>1000/3/3</math>. The calculated value is greater than the recommended.min.cis.per.chunk value, so the calculated value is used. However, when pushing 270 CIs, the calculated value is less than the recommended.min.cis.per.chunk value, so the calculated value is used.</p>
max.running.hours.for.multi.threaded	20	<p>The maximum number of hours before a push request times out in a multi-threaded environment.</p>
number.of.cis.per.request	1000	<p>The maximum number of objects retrieved from Service Manager by ID</p> <div style="background-color: #f0f0f0; padding: 5px; border: 1px solid #ccc;"> <p><b>Caution:</b> It is used for the population and federation features. Do not set it to a number greater than 1000 in case the request has a 64K limit.</p> </div>

**Parameters in the sm.properties file, continued**

Parameter	Default value	Comment
federation.request.pagination.switch	on	
federation.request.max.size	2000	size of query conditions for federation
federation.isin.max.count	500	size of isin{} size
type.of.expand.enum	2	<p>It configures the value mapping rule for the UCMDB enum type</p> <ul style="list-style-type: none"> <li>■ 0: The feature will be disabled</li> <li>■ 1: The enum type will expand to “{value}”</li> <li>■ 2: The enum type will expand to “{index}-{value}”</li> </ul> <p><b>Note:</b> It is used for the push feature.</p>
op.pagination.switch	on	<p>It indicates if pagination (client driven) is enabled</p> <ul style="list-style-type: none"> <li>■ on: Enabled.</li> <li>■ off: Disabled.</li> </ul> <p><b>Note:</b> It is used for the population feature.</p>
pop.pagination.recordcount	1000	<p>The maximum number of records displayed on each page when pagination is enabled.</p> <p><b>Note:</b> It is used for the population feature.</p>

**Parameters in the sm.properties file, continued**

Parameter	Default value	Comment
pop.createci.key	sm_id	<p>The UCMDB field of a CI record that stores the Service Manager CI ID.</p> <p><b>Note:</b> It is used for UCMDB ID Push Back by the population feature.</p>
ucmdbid.pushback.request	UpdateucmdbID PushBackRequest	<p>The web service request for pushing the UCMDB ID back to Service Manager.</p> <p><b>Note:</b> It is used for the population feature.</p>
check.sm.connections	false	<p>It indicates whether to check the SOAP connections to Service Manager instances before running a job.</p> <p>You can enable it under any of the following circumstances:</p> <ul style="list-style-type: none"> <li>■ Your Service Manager is running in High Availability mode (with load balancing), and you want to connect UCMDB to multiple Service Manager instances.</li> <li>■ You want UCMDB to not run a job when no integration connections are available, rather than run the job and then report a failure.</li> </ul>

## How to Add DEM Reconciliation Rules

It is possible that your Service Manager system already contains CI records that match CIs in your UCMDB system. Rather than add duplicate CI records to your Service Manager system, you can configure Service Manager to reconcile CI records between the two systems based on the values of specific fields.

Service Manager always attempts to reconcile CI records based on the unique key field of the Service Manager table and the **ucmdb.id** field. You can specify additional fields to reconcile on from the DEM

Reconciliation Rules form. If Service Manager finds a matching value in any one of these fields, it updates the Service Manager CI record with the attributes from the incoming UCMDB record.

When multi-tenancy is enabled, Service Manager only reconciles the CIs whose company ID matches the company ID in the data push job. For example, when pushing CIs from company 2, the reconciliation rules only apply to the Service Manager CI records that have the company code corresponding to company number 2.

In order to specify reconciliation fields, you will need to be familiar with the table and field names in both your Service Manager and UCMDB systems. If you want to reconcile on a particular attribute from the UCMDB system, you should verify that there is a corresponding Service Manager managed field for the attribute. Without such a mapping, Service Manager will not know to search for matching values in the CI record.

**Note:** Not all UCMDB attributes have a corresponding field in Service Manager. You may need to tailor your Service Manager system to add a matching field if one does not already exist.

### Using join tables for reconciliation

When setting reconciliation rules, if the device type you are reconciling has a joindef definition (as defined in the **devtype** table), use the join table name instead of the **device** table. For example, if you want to reconcile computer CIs, use the **joincomputer** table instead of the **device** table.

### Sequence of reconciliation

A reconciliation rule specifies what Service Manager table and field you want to search for matching CI values. It also specifies the sequence in which you want Service Manager to process reconciliation rules. By default, Service Manager processes rules in alphabetical order by field name. For example, Service Manager will reconcile CIs against the **asset.tag** field before reconciling CIs on the **ci.name** field.

To change the order in which Service Manager reconciles CIs, you can add a numeric value to the sequence field. For example, the following reconciliation rules ensure that Service Manager processes CIs by the **ci.name** field prior to reconciling them against the **asset.tag** field.

### Sample reconciliation rules ordered by sequence

Table Name	Field Name	Sequence
joincomputer	ci.name	1
joincomputer	asset.tag	2

A Discovery Event Manager (DEM) reconciliation rule allows you to specify which Service Manager fields you want to use to determine if an existing CI record matches a CI in a UCMDB system. An administrator typically specifies reconciliation rules prior to starting UCMDB data push jobs so that Service Manager will not create duplicate CI records.



To create a DEM reconciliation rule:

1. Log in to Service Manager as a system administrator.
2. Navigate to **Tailoring > Web Services > DEM Reconciliation Rules**. Service Manager displays the DEM Reconcile Record form.
3. In **Table Name**, type the name of the Service Manager table containing the field you want to reconcile on.
4. In **Field Name**, type the name of the Service Manager field containing the values you want to reconcile on.
5. In **Sequence**, type a number to specify what order you want Service Manager to run this rule.

**Note:** If you do not specify a sequence value, Service Manager will process field names alphabetically.

6. Click **New**. Service Manager creates the reconciliation rule.

## How to Add Discovery Event Manager Rules

Service Manager uses Discovery Event Manager (DEM) to define the actions the system should perform when the actual state of an incoming configuration item (CI) record differs from the managed state of a CI record in HP Service Manager. The DEM rules allow you to define whether the Service Manager system adds, updates, or deletes CI records based on incoming UCMDB data.

For CI records only, the DEM rules also allow you to define how Service Manager should handle duplicate logical names.

To access DEM rules in Service Manager, navigate to **Tailoring > Web Services > Discovered Event Manager Rules**, and then click **Search** to view existing rules or click **New** to create new rules.

This section includes:

- ["DEM Rules" on the next page](#)
- ["Duplication Rules" on page 208](#)
- ["CI Attributes Displayed in Change and Incident Records" on page 209](#)
- ["Searching for Change and Incident Records Opened by the Integration" on page 210](#)

## DEM Rules

Service Manager offers the following rules options:

# Action if matching record does not exist

This is the action you want Service Manager to take if it cannot find a matching CI record.

- **Add the record:**(Default) Service Manager will add a CI record when it cannot find a matching record. See ["How to Add DEM Reconciliation Rules" on page 203](#) to define what fields Service Manager uses to match CI records.
- **Add the record, and set dependency as true:** This option is available only for synchronization of CI relationship data. Service Manager adds the CI relationship record and enables outage dependency for the record by doing the following:
  - Checks the Outage Dependency check box;
  - Sets the number of dependent downstream CIs to 1. This is because UCMDB only supports one-to-one CI relationships.

### Configuration Item Relationship

Upstream CI:	E-mail / Webmail (Europe)
Relationship Name:	E-mail / Webmail (Europe) Service
Relationship Type:	Contains
Downstream CIs:	adv-eur-server-mail Microsoft Outlook     

### Outage Dependency

Outage Dependency

This Configuration Item will be considered down if

or more of the supporting configuration items are down

- **Open an Incident:**Service Manager opens an incident for someone to review the new CI record. The incident enables someone to investigate whether the new CI record is compliant with your business practices.
- **Open a Change:**Service Manager opens an unplanned change for someone to review the new CI record. The change allows you to investigate whether the new CI record is compliant with your business practices. If the CI record is compliant, the change can be approved. If the CI record is not compliant, then the change can be denied and the CI record removed. The change record lists both the current and proposed attribute values.

## Action if record exists but unexpected data discovered

This is the action you want Service Manager to perform if it does not find a matching CI attribute value.

- **Open a Change:** (Default) Service Manager opens an unplanned change to review the actual state of the CI record. The change allows someone to investigate whether the new attribute value is compliant with your business practices. If the value is compliant, the change can be approved. If the value is not compliant, then the change can be denied and the CI attribute value reverted to its managed state.
- **Log Results and update record:**Service Manager logs the results of the actual state of the CI record, and then updates the CI record.
- **Open an Incident:**Service Manager opens an Incident to investigate the actual state of a CI record and determines what actions must be performed or initiated to bring the record into compliance with Service Manager.

## Action if record is to be deleted

This is the action you want Service Manager to perform if an external event specifies that the record needs to be deleted.

- **Delete record:** (Default for CI Relationship records) This option is available for synchronization of both CI and CI Relationship records. Service Manager automatically deletes the CI/CI Relationship record.

- **Open an Incident:** This option is available only for synchronization of CI Relationship records. Service Manager opens an incident to investigate the deleted record and determines which actions must be performed or initiated to bring the record into compliance with Service Manager.
- **Open a Change:** This option is available only for synchronization of CI Relationship records. Service Manager opens an unplanned change to review the deleted record. The change allows someone to investigate whether the deleted record is compliant with your business practices. If the record is compliant, the change can be approved. If the record is not compliant, then the change can be denied and the record added back to the system.
- **Update record to the selected status:** (Default for CI records) This option is available only for synchronization of CI records. Service Manager updates the status of the CI record to a value selected from the drop-down list (for example, **Retired/Consumed**), rather than delete the record permanently.

**Note:** Values available from the drop-down list are defined in the **ICM Status** global list.

- **Open a Change to update record to the selected status:** This option is available only for synchronization of CI records. Service Manager opens an unplanned change to update the CI record's status to a value selected from the drop-down list (for example, **Retired/Consumed**). The change allows someone to investigate whether the requested status change is compliant with your business practices. Once the change has been approved and closed, Service Manager automatically changes the CI record to the selected status. If the change has been denied, Service Manager makes no changes to the CI record.
- **Open an Incident to update record to the selected status:** This option is available only for synchronization of CI records. Service Manager opens an incident to update the record's status to a value selected from the drop-down list (for example, **Retired/Consumed**). Once the incident has been closed, Service Manager automatically updates the CI record to the selected status.

## Duplication Rules

UCMDB may create two completely separate yet legit CI records that happen to have the same "name." The UCMDB name field is mapped to the logical.name field (which must be unique) in Service Manager. Pushing the two CI records to Service Manager would cause a duplicate logical name problem. You have several ways to avoid this problem. See the following table.

**Solutions to the duplicate logical name problem**

Product side	Solution
UCMDB	<p>Change the names directly in UCMDB or change the UCMDB reconciliation rule to make sure the names are not the same.</p> <p>This is highly recommended.</p> <p>In the integration adapter mapping configuration (xslt) file, avoid mapping the UCMDB name field to the SM logical name field directly in either of these ways:</p> <ul style="list-style-type: none"> <li>• Map another UCMDB unique attribute to the SM logical.name field, and map the UCMDB name field to another SM field;</li> <li>• Add a prefix to the name. The following are examples.                             <ul style="list-style-type: none"> <li>■ UCMDB switches or routers are simply named as “Router” or “Switch” and identified by their underlying MACs. You can configure their “SM logical name” to be “&lt;MAC&gt; + &lt;name&gt;”.</li> <li>■ UCMDB databases often have the same name (due to the implementation of clusters and Oracle RACs). You can configure their “SM logic name” to be “&lt;full DNS name&gt; + &lt;name&gt;”.</li> </ul> </li> </ul>
Service Manager	Use the duplication rule options in DEM Rules in Service Manager.

Service Manager offers the following duplication rule options on the Duplication Rule tab in each DEM rule with a Table Name other than “cirelationship”:

- **Action if logical name is duplicated (CI with different uCMDB ID):** This is the action you want Service Manager to perform if the logical name is already used by another CI record when a CI record is added or updated.
  - **Rename to <name>\_[RENAMED]\_1/2/3:** (Default) Service Manager changes the logical name by adding a suffix.
  - **Return Error:** Service Manager returns a duplicate key error to UCMDB.

**CI Attributes Displayed in Change and Incident Records**

Service Manager displays a Change Details section on the corresponding change or a CMDB Changes section on the corresponding incident when you configure DEM to open either change records or incident records when it discovers CI attribute changes through the UCMDB-SM integration. Service Manager only displays a tab for CI attributes when the UCMDB-SM integration is enabled and you have

defined a rule in the Discovery Event Manager to create a change or incident record when a CI is added, updated, or deleted.

Both the Change Details and CMDB Changes sections display the current CI attribute values alongside the actual attribute values discovered by UCMDB. You can use this information to approve or deny a change or escalate an incident to the proper assignment group.

## Searching for Change and Incident Records Opened by the Integration

You can use the following search criteria to find change and incident records opened by the UCMDB-SM integration.

### Search options available for change and incident records

Record type	Search option available
Change	Search for records with the category <b>unplanned change</b> .
Incident	Search for records using the <b>Generated by UCMDB Integration</b> option.

## How to Add a CI Attribute to the Integration for Data Push

You can use the following steps to add a CI attribute to the integration.

- Does the CI attribute already exist in the UCMDB class model?  
Yes. Go to [Step 3](#).  
  
No. Go to [Step 2](#).
- Add the CI attribute to the UCMDB class model.  
See ["How to Add the CI Attribute to the UCMDB Class Model" on the next page](#).
- Add the CI attribute to the query layout.  
See ["How to Add a CI Attribute to the Query Layout" on page 212](#).
- Add a web service field for the Service Manager CI type.  
See ["How to Add a Web Service Field for the Service Manager CI Type" on page 214](#).
- Map the CI attribute to the SM web service field.  
  
See ["How to Map the CI Attribute to a Service Manager Web Service Field" on page 222](#).

## How to Add the CI Attribute to the UCMDB Class Model

The integration only uses a subset of the CI attributes available from your UCMDB system. Out-of-the-box, the integration consists of CI attributes that are typically managed from a Service Manager system such as host name and host DNS name. Before creating a new UCMDB CI attribute, you should determine if there are any existing CI attributes in your UCMDB system that provide the data you want. In most cases, there is an existing attribute tracking the data that you want to add to the integration. For example, the Node CI type contains many attributes that you can add to the integration.

Display Name	Name	Type	Description	Default Value	Visible	Editable	Key	Comparable
ack_cleared_time	ack_cleared_time	long				✓		
ack_id	ack_id	string				✓		
Actual Delete Time	root_actualeleletime	date	When will t...					
Actual Deletion Period	root_actualeletionp...	integer	What is the...	40	✓	✓		
Admin-State	adminstate...	adminstate...	Admin-State	Managed				
Allow CI Update	data_allow_auto_dis...	boolean		true	✓	✓		
BiosAssetTag	bios_asset_tag	string	Asset tag ...		✓	✓		
BiosDate	bios_date	date	The BIOS/F...		✓	✓		
BiosSerialNumber	bios_serial_number	string	A manufac...		✓	✓		
BiosSource	bios_source	string	Shows the...		✓	✓		
BiosUuid	bios_uuid	string	A System ...		✓	✓		
BiosVersion	bios_version	string	Shows the...		✓	✓		
BODY_JCON	BODY_JCON	string		host		✓		
Calculated ID	calculated_id	bytes	Calculated ID			✓		
CalculatedLocation	calculated_location	string			✓	✓		
Candidate For Deleti...	root_candidatefordel...	date	When will t...					
Change-Corr-State	data_changecorrstate	changestat...	Change-St...	No-Change				

The following steps illustrate how to add a new CI attribute to an existing CI type. This is not the expected typical scenario. Typically, you would add an existing CI attribute to the integration, which means you do not need to perform these steps.

**Note:** The integration does not require any special steps to add a CI attribute to the UCMDB class model. You can use the standard CI attribute creation procedures to add a CI attribute. For more information on CI attribute creation, see the UCMDB Help Center.

As an example, the following steps will add an attribute named **comments** to the Business Service CI type.

To add a CI attribute to the UCMDB class model:

1. Log in to UCMDB as an administrator.
2. Navigate to **Modeling > CI Type Manager**.

3. Select the CI type to which you want to add a new CI attribute from the CI Types navigation tree. For example, **ConfigurationItem > BusinessElement > BusinessService**.

4. Click the **Attributes** tab.

5. Click the **Add** icon. 

The Add Attribute window opens.

6. In Attribute Name, type the unique name you want to use for the new CI attribute. For example, `comments`.

**Caution:** The name cannot include any of the following characters: ` / \ [ ] : | < > + = ; , ? \*.

7. In Display Name, type the name that you want UCMDB to display in the interface. For example, `Comments`.


8. (Optional) In Description, type a description of the new CI attribute.

9. In Attribute Type, select **Primitive** or **Enumeration/List** depending on the data type of the attribute. For example, select **Primitive** and select **string**.

10. In Value Size, type the maximum character length that the attribute can have. For example, `300`.

11. Select the **Enable default value** option, and enter a default value. Or, do not select this option to leave the default value blank.

12. Click **OK** to save the attribute.

13. Click the **Save** icon  to save attribute changes to the CI type.

Now, the attribute has been added to the CI type. Next, you need to add the attribute to a query that synchronizes the CI type. See ["How to Add a CI Attribute to the Query Layout"](#) below.

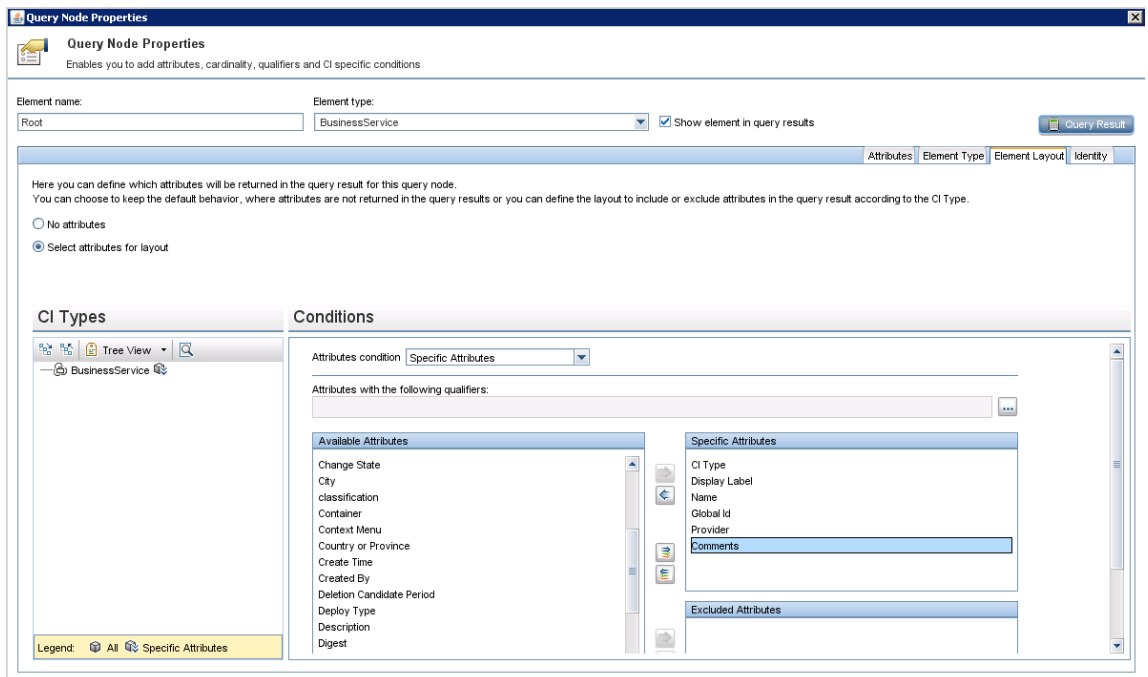
## How to Add a CI Attribute to the Query Layout


To add an existing CI attribute to the integration, you must add this attribute to the layout setting from the query that synchronizes the CI type. You must know which CI type contains the CI attributes that you want to add to the integration.

To add a CI attribute to the query layout:



1. Log in to UCMDB as an administrator.
2. Navigate to **Modeling > Modeling Studio**.
3. For Resource Type, select **Queries**.
4. From the Queries navigation tree, click **Integration > Service Manager**.
5. Select the query that manages the CI type whose attributes you want to add to the integration. For example, **Push > SM Computer Push 2.0**. UCMDB displays the TQL layout of the query.
6. Select the node from the query layout that contains the CI attribute that you want to add to the integration. For example, **Root**.
7. Right-click the node and select **Query Node Properties**. The Query Node Properties window opens.
8. Select the CI type (**Database** in this example), and click the **Element Layout** tab.
9. Select the CI attribute that you want to include in the integration from the **Available Attributes** list, and click the **Add** icon to add it to **Specific Attributes** list. For example, **Comments**.



10. Click **OK** to save the node properties.
11. Click the **Save** icon  to save the query.

Now, the CI attribute has been added to the relevant query layout. Next, you need to add a Service Manager web service field that will be mapped to this UCMDB CI attribute.

### How to Add a Web Service Field for the Service Manager CI Type

The integration uses only a subset of the CI attributes available from your Service Manager (SM) system. CI attributes must be mapped between UCMDB and SM. Before creating a new SM CI attribute that will be mapped to the UCMDB CI attribute you added previously, you must determine if there are any existing CI attributes in your Service Manager system that provide the data you want. In most cases, there is an existing attribute tracking the data that you want to add to the integration. For example, the Computer CI type contains many attributes that you can add to the integration.

Join Table Name:

Common Name:

Table Name	Field Names	Field Captions
node	node,addIPAddr	Add I P Addr
node	node,addMacAddress	Addl Mac Address
node	node,bios.id	Bios Id
node	node,cpu	Cpu
node	node,disk.device	Disk Device
node	node,file.system	File System
node	node,logical.name	Logical Name
node	node,machine.name	Machine Name
node	node,os.manufacturer	Os Manufacturer
node	node,os.version	Os Version
node	node,physical.mem.total	Physical Mem Total
device	device,ac.category	Ac Category
device	device,addl	Addl
device	device,admin.id	Admin Id
device	device,admin.password	Admin Password
device	device,admin.urlport	Admin Urlport
device	device,allow.subscription	Allow Subscription
device	device,asset.tag	Asset Tag
device	device,assignment	Assignment
device	device,auditDate	Audit Date
device	device,auditDiscrepancy	Audit Discrepancy
device	device,auditPolicy	Audit Policy
device	device,auditStatus	Audit Status
device	device,auditedBy	Audited by

If you decide to use an existing attribute instead of creating a new one, you can directly modify the web service definition in Service Manager to expose that attribute, and then map this SM attribute to the UCMDB attribute by using the UCMDB Visual Mapping tool (see ["How to Map the CI Attribute to a Service Manager Web Service Field" on page 222](#)).

If you decide to add a new attribute, follow the steps below. The steps vary depending on the data type of the attribute: simple (string, for example), or complex (array of structure or structure).

## Add a Simple Attribute to the SM CI Type

As an example, the following steps illustrate how to add a simple attribute (**comments**) to an existing CI type (Business Service).

1. Log in to UCMDB as a system administrator.
2. Click **Data Flow Management > Adapter Management**, and then select **ServiceManagerEnhancedAdapter 9-x** to open the corresponding XML mapping file (**SM Business Service 2.0.xml** in this example) with the Visual Mapping tool editor.
3. On the External Class Model pane, select the CI type (**bizservice**).

The screenshot displays the Visual Mapping tool editor for the 'SM Business Service Push 2.0.xml' file. The 'External Class Model' pane on the left shows a tree view with 'bizservice' selected. The 'Visual Mapping' pane in the center shows a tree view of the 'bizservice' CI type with several fields: UCMBId, CustomerId, Type, Subtype, ServiceProvider, and ServiceName. Each field has a corresponding mapping configuration, including a root element name and an 'Ignore on Null' checkbox. The 'XML Editor' pane at the bottom shows the XML structure with target mappings for each field. The 'Attributes' panes on the left and right show tables of attributes for the selected CI type, including fields like 'Identifier', 'UCMBId', 'CustomerId', 'Name', 'Type', 'Subtype', and 'Provider'.

4. On the left-side Attributes pane (which displays the fields that are retrieved from the Service Manager web service object for the Business Service CI type), click the **Add New Attribute to Selected External Node** icon.
5. Type a name and display label for the new attribute, and click **OK**.

**Caution:** If you enter a name that duplicates or is part of an existing field name in the attribute table of the Service Manager CI type or in the **device** table, an error will occur

indicating the attribute could not be created.

6. Click OK again to confirm the attribute creation.

The new attribute (**mycomments**) appears in the Service Manager attribute list.

Status	Display Name	Name	Type	Descrip...
*	logical.name	CIidentif...	STRING	
*	ucmdb.id	UCMDBId	STRING	
	ci.name	Service...	STRING	
	company	Custom...	STRING	
	id	CIName	STRING	
	mycomments	MyCom...	STRING	
	subtype	Subtype	STRING	
	type	Type	STRING	
	vendor	Service...	STRING	

At the same time, UCMDB invokes the Service Manager Web Service API to add the new attribute, expose it in the Web Service API and set it as a managed field. You can open the web service object in Service Manager to verify the new attribute (**Tailoring > Web Services > Web Service Configuration**).

External Access Definition

Service Name:

Name:

Object Name:

◆ Allowed Actions   ◆ Expressions   ◆ **Fields**   ◆ RESTful

Field	Caption
ucmdb.id	UCMDBId
ci.name	ServiceName
type	Type
subtype	Subtype
company	CustomerId
logical.name	CIIdentifier
vendor	ServiceProvider
id	CIName
mycomments	MyComments

7. Save the configuration file.

## Add an Array of Structure or Structure to the CI Type

Service Manager uses an Array of Structure or Structure to store complex attributes, for example, the ports of a Computer CI. You can also create such kind of complex attributes through the Visual Mapping tool in UCMDB.

The following steps illustrate how to add an attribute named **comport** to the Computer CI type.

1. Log in to UCMDB as a system administrator.
2. Open the corresponding XML mapping file (**SM Computer Push 2.0.xml** in this example), with the Visual Mapping tool editor.

3. Select the CI type (**computer**), and click the **Add New CI Type to External Class Model** icon.
4. In the child CI type creation dialog that appears, enter a name and description for the attribute, select **Many to One** for **Relation with Parent** if the attribute data type is array of structure or select **One to One** if the data type is structure.

**Add new child node**

You must define a new node's properties for an external class model.

**General**

\* Name:

Description:

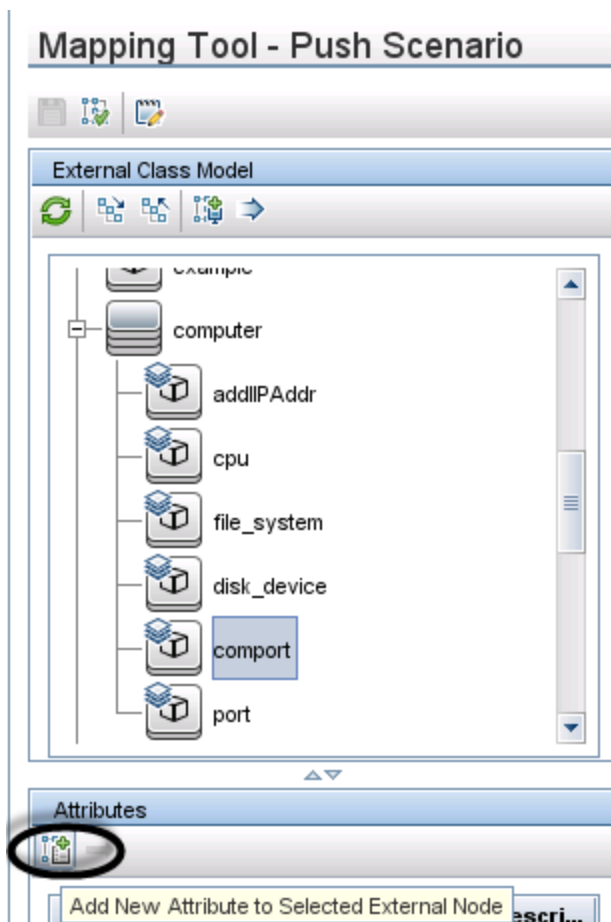
Parent Node:

**Metadata**

Relation with Parent

OK Cancel

5. Click **OK**. The attribute is created on the Service Manager side.
6. Select the new attribute in the External Class Model pane, and click the **Add New Attribute to Selected External Node** icon.



7. Add a field (**httpPort**) to the array of structure, and click **OK**. Click **OK** again to confirm the attribute creation.

**Add new attribute**

You must define a new attribute's properties for an external node.

### General

Name:

Display Label:

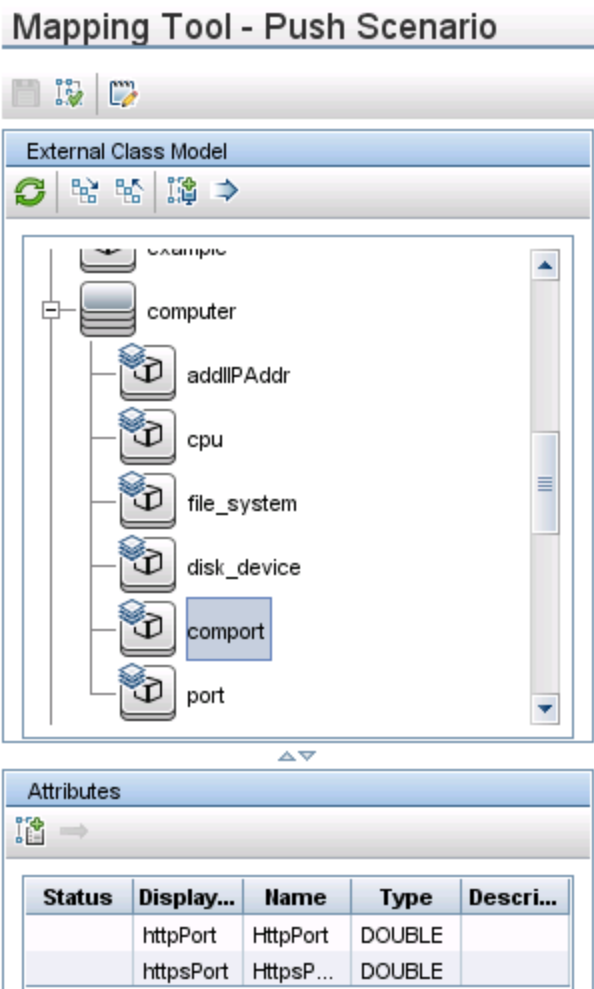
Description:

Type:

OK Cancel

8. Repeat the previous two steps to add more fields to the array of structure as needed.





The attribute is also automatically added to the corresponding web service object in Service Manager.

## External Access Definition

Service Name:

Name:

Object Name:

Field	Caption
cpu[cpu.name]	CpuName
cpu[cpu.clock.speed]	CpuClockSpeed
file.system[mount.point]	MountPoint
file.system[disk.type]	DiskType
file.system[file.system.type]	FilesystemType
file.system[disk.size]	DiskSize
asset.tag	AssetTag
machine.name	HostName
id	CIName
disk.device[model.name]	ModelName
disk.device[disk.vendor]	DiskVendor
disk.device[disk.name]	DiskName
isVisualization	IsVisualization
istatus	AssetStatus
comport[httpPort]	HttpPort
comport[httpsPort]	HttpsPort

9. Save the configuration file.

Next, you need to map this Service Manager CI type attribute to the one you added previously in UCMDB.

### How to Map the CI Attribute to a Service Manager Web Service Field

The integration uses an adapter to transform UCMDB CI attributes to web services objects recognized by Service Manager. The adapter in turn specifies what XML transformation files the integration should use to convert UCMDB queries into properly formatted Service Manager web services messages.

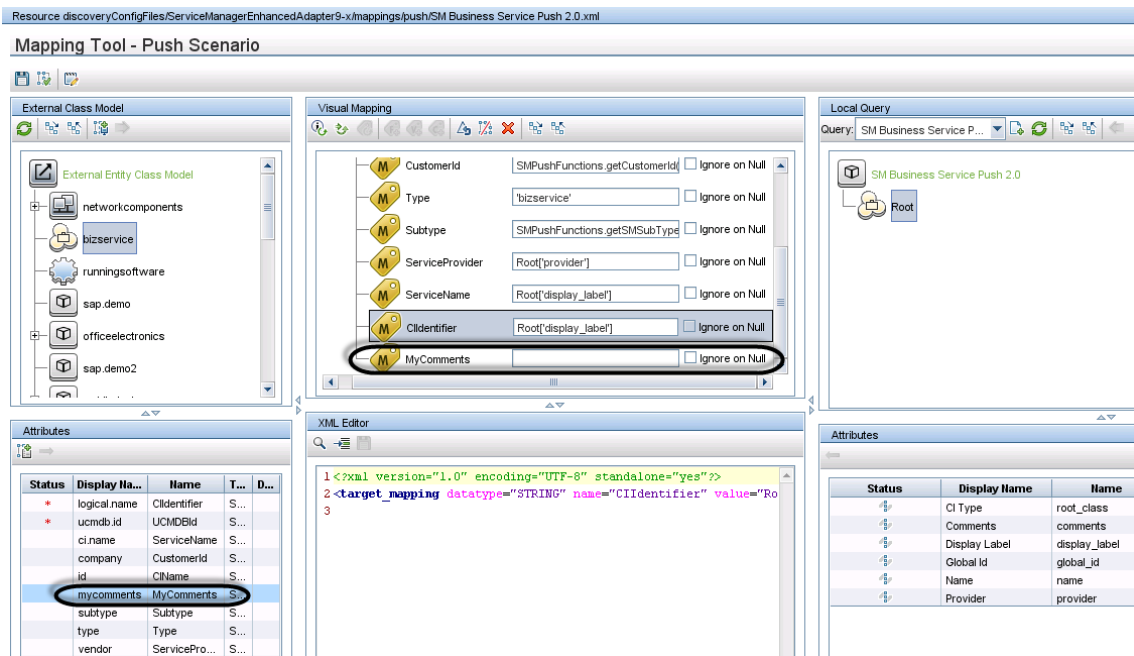
Out-of-the-box, each integration query has a corresponding XML configuration file that maps to a particular CI type in UCMDB. In addition, each attribute for which you enabled calculation requires its own entry in the XML configuration file. Without an XML transformation entry, Service Manager cannot receive any CI attribute updates from your UCMDB system.

If you want to add a new attribute to the integration, you must edit the XML configuration file for the parent CI type and add an entry for the CI attribute. For information about which CI types each query manages, see ["Queries for Push" on page 183](#).

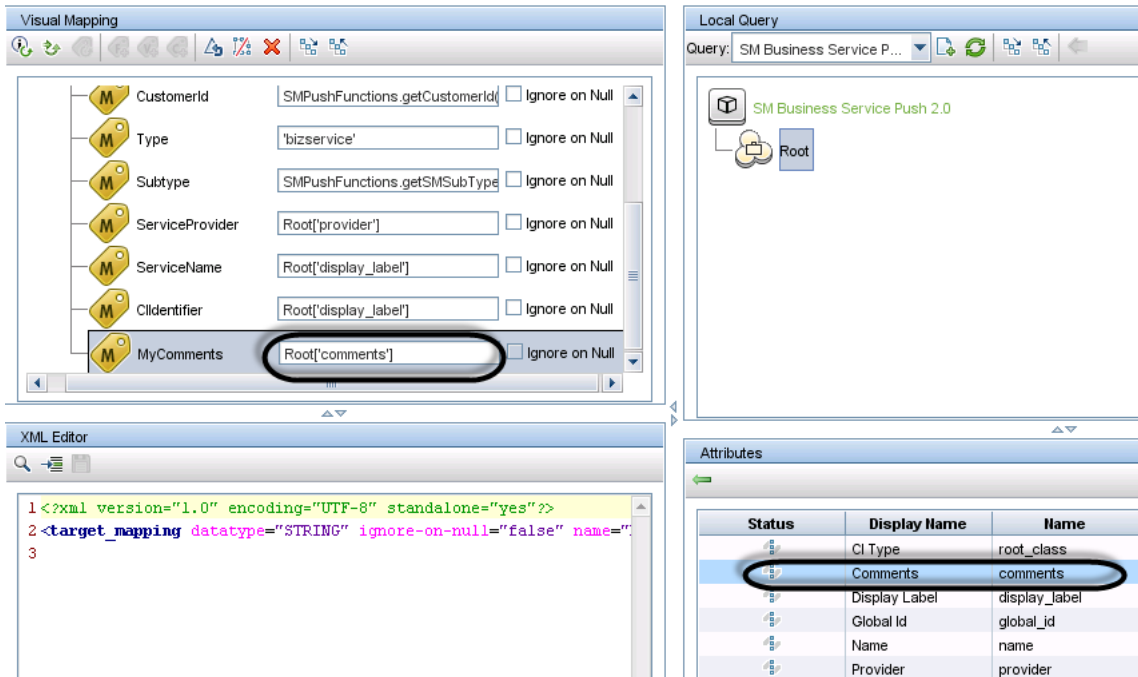
The following steps illustrate how to map UCMDB CI attribute **Comments** (which you added previously) to the **MyComments** Service Manager attribute (which you created in the previous step).

To map a UCMDB CI attribute to an SM CI attribute:

1. Log in to UCMDB as an administrator.
2. Navigate to **Data Flow Management > Adapter Management > ServiceManagerEnhancedAdapter9-x > Configuration Files**.
3. Double-click the XML configuration file that manages the parent CI type of your CI attribute. For example, open **SM Business Service Push 2.0.xml** to add an attribute to the **SM Business Service Push 2.0** query.
4. Select the relevant CI type in the External Class Model pane (**bizservice** in this example), and drag and drop the attribute (**mycomments**) from the left-side Attributes pane into the Visual Mapping pane.



5. Select the relevant Node in the Local Query panel (**Root** in this example), and drag and drop the attribute (**comments** in this case) into the mapping entry that you created in the previous step.



6. Save the XML configuration file. The **Comments** attribute in UCMDB is now mapped to the **MyComments** attribute in Service Manager.

**Note:** When you create or edit and then save a configuration file in Adapter Management, UCMDB automatically restarts the adapter with the new configuration file.

## How to Add a CI Type to the Integration for Data Push

You can use the following steps to add a CI type to the integration.

1. Does the CI type already exist in the UCMDB class model?  
 Yes. Go to [Step 3](#).  
 No. Go to [Step 2](#).
2. Add the CI type to the UCMDB class model.  
 See ["How to Add the CI Type to the UCMDB Class Model"](#) on the next page.
3. Create a query to synchronize the CI type.  
 See ["How to Create a Query to Synchronize the CI Type"](#) on page 228.

4. Add the CI type's attributes to the query layout.  
See ["How to Add the CI Type's Attributes to the Query Layout"](#) on page 232.
5. Add the CI type to Service ManagerService Manager.  
See ["How to Add the CI Type to Service Manager"](#) on page 234.
6. Map the CI type's attributes to web service fields.  
See ["How to Map the CI Type's Attributes to Web Service Fields"](#) on page 237.
7. Add custom queries to integration data push jobs.  
See ["How to Add a Custom Query to an Integration Job"](#) on page 248.


## How to Add the CI Type to the UCMDB Class Model

Before creating a new UCMDB CI type, you should determine if there are any existing CI types in your UCMDB system that provide the CI attributes you want. In most cases, you can create links to one or more existing CI types to create a new logical CI type for use by the integration.

The following steps illustrate how to create a new CI type called SM RDBMS based on an existing CI type called database.

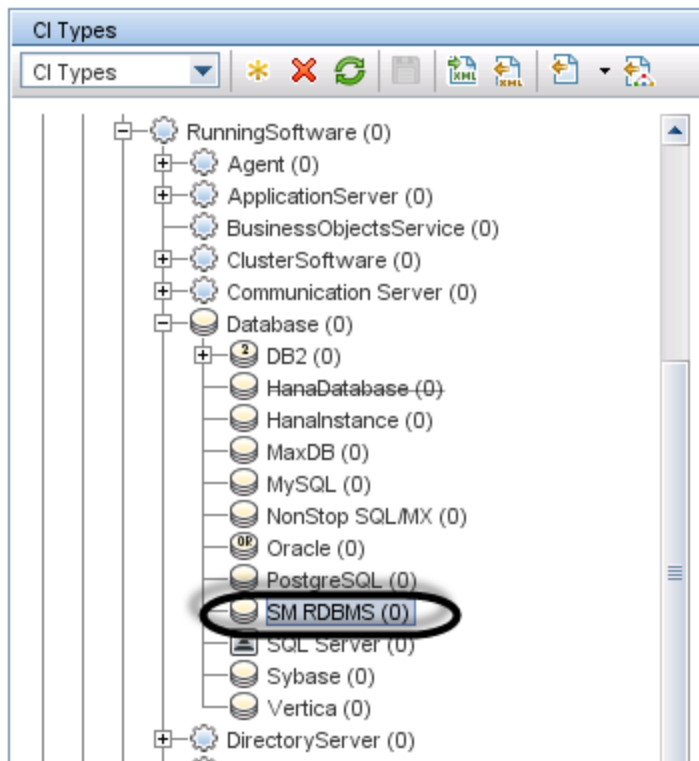
**Note:** The integration does not require any special steps to add a CI type to the UCMDB class model. You can use the standard CI type creation procedures to add a CI type. For more information on CI type creation, see the UCMDB Help Center.

To add a CI type to the UCMDB class model:

1. Log in to UCMDB as an administrator.
2. Navigate to **Modeling > CI Type Manager**.
3. Select the base CI type you want to use for your new CI type from the CI Types navigation tree:  
**Managed Object > ConfigurationItem > Infrastructure Element > Running Software > Database**.
4. Click the **New** icon .  
The Create Configuration Item Type window opens.
5. In Name, type the unique name you want to use for the new CI type. For example, `sm_rdbms`.

**Caution:** The name cannot include any of the following characters: ` / \ [ ] : | < > + = ; , ? \*.

6. In **Display Name**, type the name you want UCMDB to display in the interface. For example, `SM RDBMS`.
7. In **Description**, type a description of the new CI type. This is an optional field. For example, `Hosts running relational databases`.
8. In **Base CI Type**, verify that the proper base CI type is selected. Your new CI type will inherit the attributes of the base CI type you select here. For example, **Database**.
9. Click **Next**. The wizard displays a list of CI attributes from the base CI type.
10. Add, edit, or remove CI attributes as needed for the new CI type. For example, accept the default attributes inherited from **Database**.
11. Click **Next**. The wizard displays a list of qualifiers from the base CI type.
12. Add or remove qualifiers as needed for the new CI type. For example, accept the default qualifiers.
13. Click **Next**. The wizard displays a list of icons associated with the CI type.
14. Select the icons associated with this CI type. For example, accept the default abstract class icon.
15. Click **Next** to add any menu item properties or label definitions as needed. For example, accept the default settings from the base CI type.
16. Click **Finish** to create the CI type.

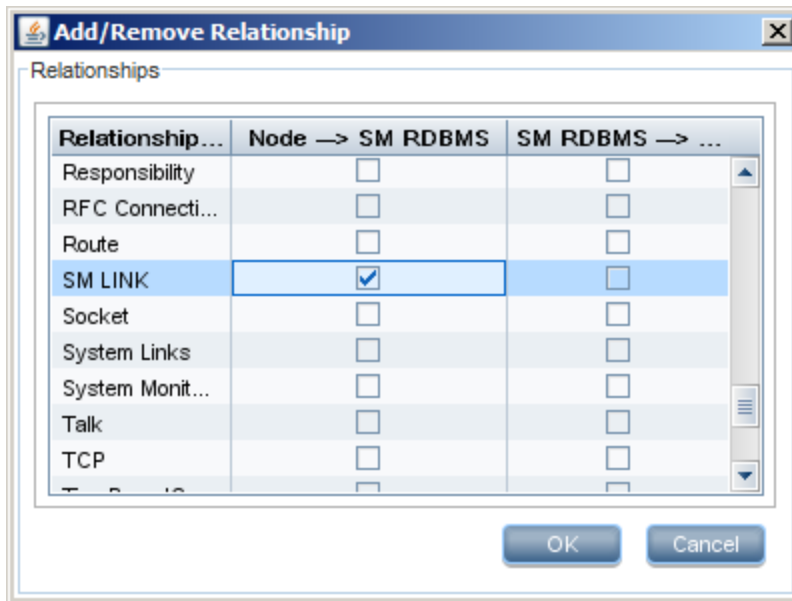


17. Select your new CI type from the tree. For example, **SM RDBMS**.
18. Browse to an existing CI type you want to link to, and control-click it to add it to your selection. For example, **Node**.

**Note:** Choose an existing CI type that has the attributes that you want to be part of your new logical CI type.

19. Right-click one of the selected CI types, and click **Add/Remove Relationship**. The Relationships window opens.
20. Create an SM Link relationship from the existing CI type to the new CI type. For example, from **Node** to **SM RDBMS**.

**Note:** You need to create a new SM Link relationship if it does not exist.



21. Click **OK** to create the relationship.
22. Click the **Save** icon to save the CI type.

### How to Create a Query to Synchronize the CI Type

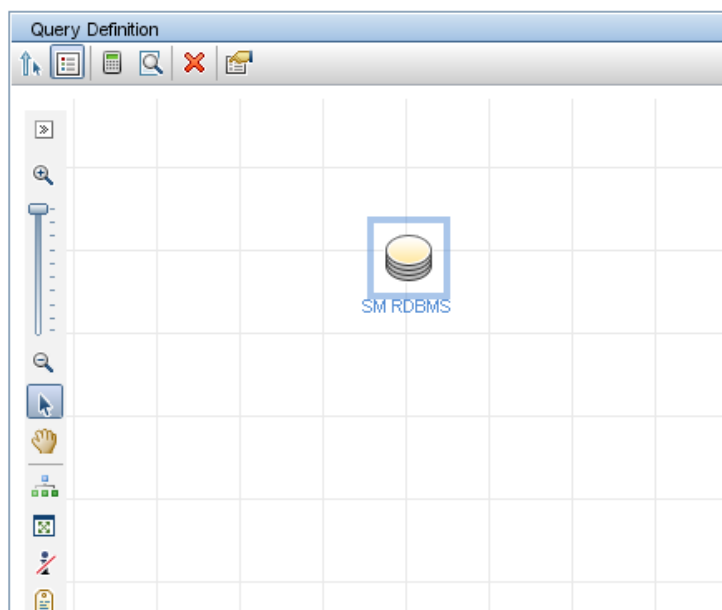
The integration uses queries to gather CI attribute values and pass them to your Service Manager system. You must create a query for any CI type you add to the integration. Any query you create must conform to the ["Query Requirements" on page 188](#).

The following steps illustrate how to create a new query named **rdbmsData** for the **SM RDBMS** CI type described in previous sections.

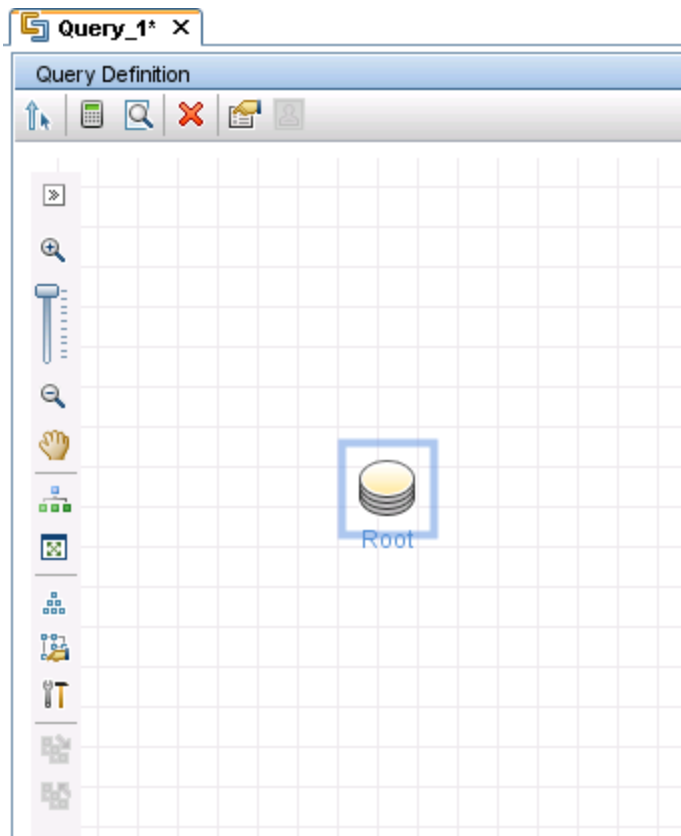
1. Log in to UCMDB as an administrator.
2. Navigate to **Modeling > Modeling Studio**.
3. From the Queries navigation tree, click **Integration > Service Manager**.
4. Right-click **Service Manager**, and select the **New > Query**.  
The Query Definition window opens.
5. Find the CI type that will be the root node of your query from the CI Type Selector. This CI type is typically the one that provides the most attributes for the CI. For example, **Managed Object > ConfigurationItem > InfrastructureElement > RunningSoftware > Database > SM RDBMS**.



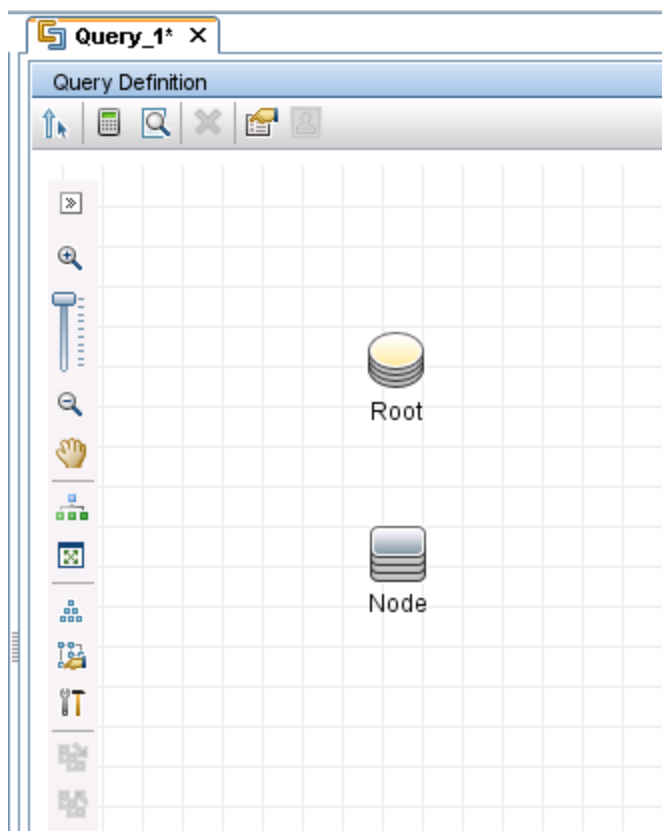
6. Drag the root CI type from the CI Type Selector and drop it into the empty Editing pane. UCMDB displays the icon of the CI type.



7. Right-click the CI type icon, and select **Query Node Properties**. The Node Properties window opens.
8. Change the Element Name to **Root**.
9. Click **OK** to save the node properties.

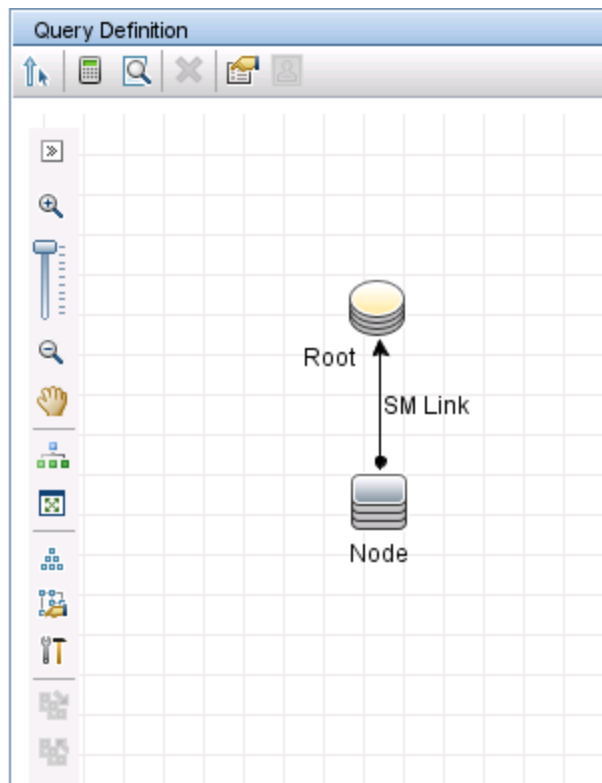



10. Find any additional CI types you want to add to the query from the CI Type Selector. These CI types typically provide additional CI attributes. For example, **Managed Object > ConfigurationItem > Infrastructure Element > Node**.
11. Drag the additional CI type from the CI Type Selector and drop it into the Query Definition pane. UCMDB displays the icon of the additional CI type.



12. Create relationships between the Root CI type and the additional CI types as needed. For example, create an SM Link between **Root** and **Node**.
  - a. Select **Root** and control-click the additional CI type. For example, **Node**.
  - b. Right-click one of the selected items, and click **Add Relationship**. The Add Relationship window opens.
  - c. Type a Relationship Name. For example SM Link.
  - d. Select the relationship direction.

- e. Click **OK** to add the relationship.



13. Repeat step 10 to step 12 for each additional CI type you want to add to the query. For example, SM RDBMS does not need any additional CI types.
14. Click the **Save** icon  to save the query.
  - a. In **Query Name**, type the unique name you want to use for the new query. For example, rdbmsData.
  - b. In the folder tree, select the folder in which you want to save the query. For example, **Root > Integration > Service Manager > Push**.
  - c. Click **OK**. UCMDB adds your new query to the query list.

### How to Add the CI Type's Attributes to the Query Layout

To add a CI attribute to the integration, you must enable the calculation layout setting from the query that synchronizes the CI type. Because you must enable calculation for each attribute you want to add to the integration, you should be familiar with the integration CI types and their attributes.

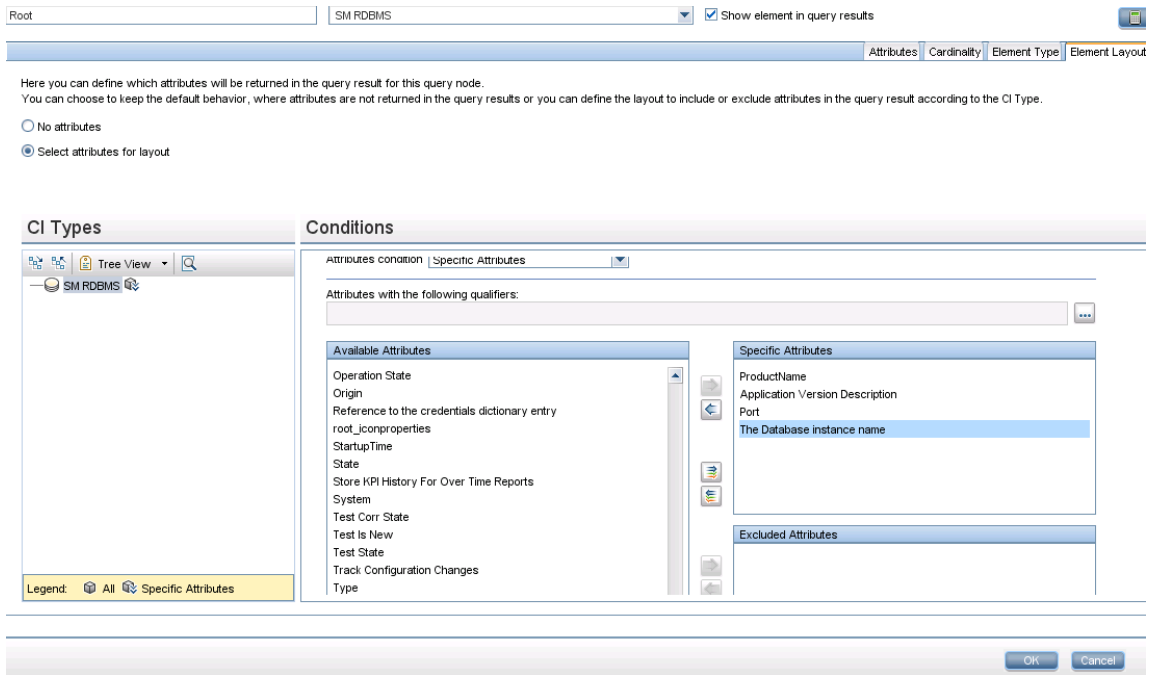
The following steps illustrate how to enable calculation for attributes of the **SM RDBMS** CI type described in previous sections.


To add a CI type's attributes to the query layout:

1. Log in to UCMDB as an administrator.
2. Navigate to **Modeling > Modeling Studio**.
3. From the Queries navigation tree, click **Integration > Service Manager**.
4. Double-click the query that manages the CI type whose attributes you want to add to the integration. For example, **rdbmsData**. UCMDB displays the TQL layout of the query.
5. Right-click the **Root** node from the query layout, and then select **Query Node Properties**. The Query Node Properties window opens.

**Caution:** Your integration query must contain a node called **Root**. See ["Query Requirements" on page 188](#) for more information.

6. Click the **Element Layout** tab, and select the **Select attributes for layout** option.
7. Select **Specific Attributes** from the **Attributes condition** list, and from the Available Attributes list select the attributes you want and move them to the Specific Attributes list. For example, add the **Product Name, Application Version Description, The Database Instance Name, and Port** attributes.



8. Click **OK** to save the query node properties.
9. Select any additional node that contains the attributes you want to add to the integration. For example, **Node**.
10. Repeat [step 4](#) through [step8](#) for each additional node.
11. Click **OK** to save the query node properties.
12. Repeat step 9 to step 13 for each additional node that contains the attributes you want to add to the integration.
13. Click the **Save** icon to save the query .

### How to Add the CI Type to Service Manager

Before creating a new Service Manager CI type, you should determine if there are any existing CI types in your Service Manager system that provide the CI attributes you want. In most cases, you can reuse the existing CI types for the integration.

You can add a new CI type to Service Manager by using the Visual Mapping tool in UCMDB, which means you do not have to log out of UCMDB and then log in to Service Manager.

The following steps illustrate how to create a new CI type called **RDBMS**.

**Note:** This example is provided only as an illustration of the steps. The best practice is to reuse the existing Service Manager CI type **RunningSoftware** to map with UCMDB CI type **SM RDBMS**.

1. Log in to UCMDB as an administrator.
2. Open an existing XML mapping file (**SM Computer Push 2.0.xml**, for example) with the Visual Mapping tool editor.
3. Select the Root node in the External Class Model panel, and click the **Add New CI Type to External Class Model** icon.
4. Enter parameter values as follows.

**Name:** rdbms

**Description:** CI type for RDBMS

**table:** smrdbms

**subtype:** Oracle, SQL Server



5. Click **OK**.
6. Click **OK** again to confirm the CI type creation. The CI type is automatically created in Service Manager, as shown in the following figure.



## Manage CI Types

CI Type Description:	Rdbms
CI Type:	rdbms
Bitmap:	lbox
Format Name:	configurationItem
Attr File:	smrdbms
Join Def:	joinsmrdbms
Print Format Name:	
Bulk Update Format Name:	
Active:	<input checked="" type="checkbox"/>

Sub Types
Oracle
SQL Server

### What happens on the Service Manager (SM) side?

All relevant SM objects (DBDICT, JoinDef, Web Service API, DEM Rule, and so on) for the new CI type are automatically created on the SM side, except the format. The new CI type will use **configurationItem** as the default format.

If you want to use custom formats in SM for the new CI type, you have to create them manually. You can create the formats in Forms Designer based on existing view forms and bulk update forms. To access Forms Designer in Service Manager, type `fd` in the command line or navigate to **Tailoring > Forms Designer**. For more information about creating forms in Service Manager, see the Service Manager online help and the *Tailoring Best Practices Guide*.


### How to Map the CI Type's Attributes to Web Service Fields

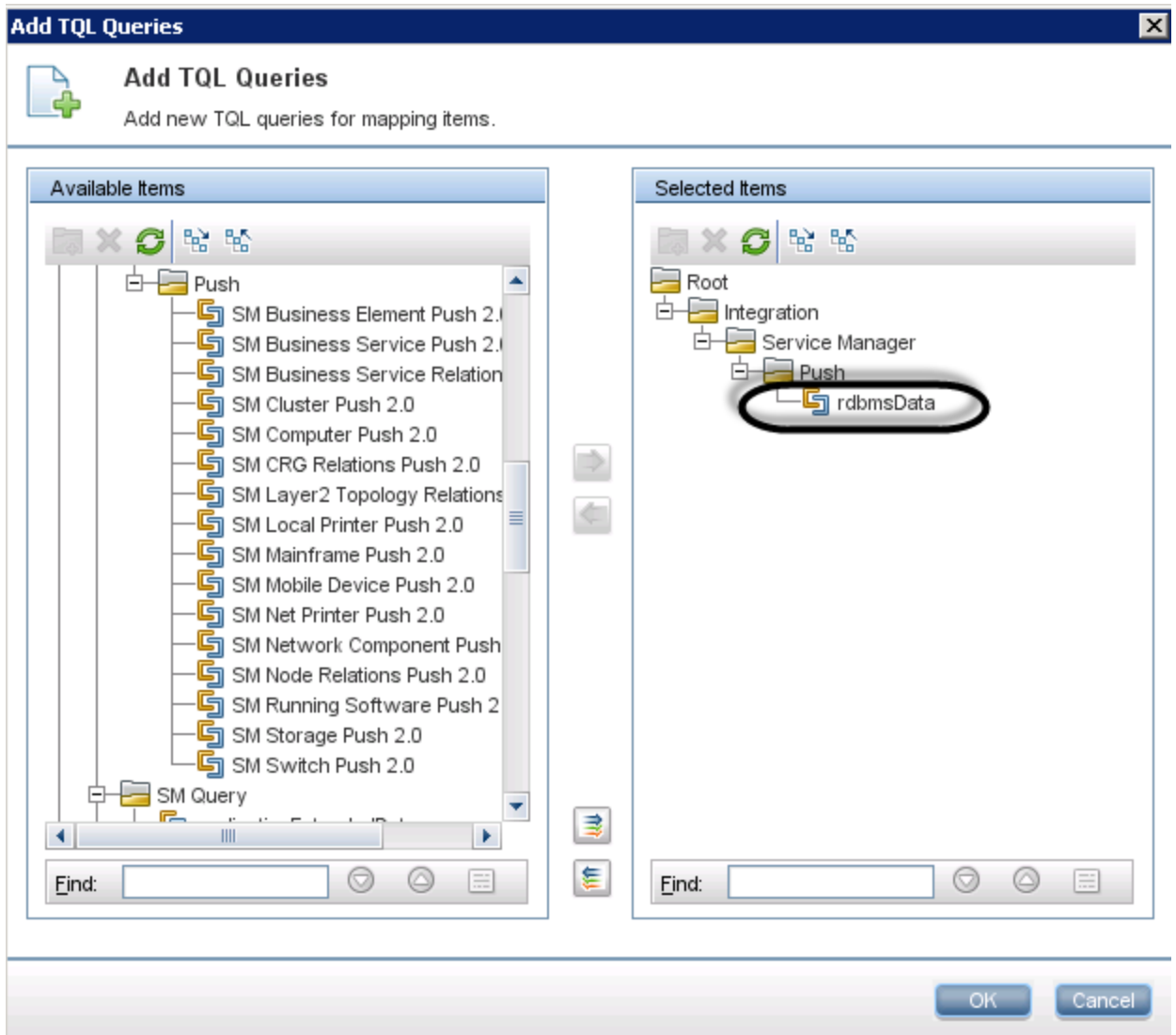
The integration uses the Service Manager Adapter to transform UCMDB CI attributes to web services objects recognized by Service Manager. The Service Manager Adapter uses XML configuration files to convert UCMDB queries into a properly formatted Service Manager web services messages. Out-of-the-box, each integration query has a corresponding XML configuration file. In addition, each attribute you enable for synchronization from advanced layout settings requires its own entry in the XML configuration file.

If you want to add a CI type to the integration, you must create a matching XML configuration file that defines how the Service Manager Adapter transforms each CI type into a Service Manager web service object. See ["Integration Queries" on page 183](#) for information about the CI types each query manages.

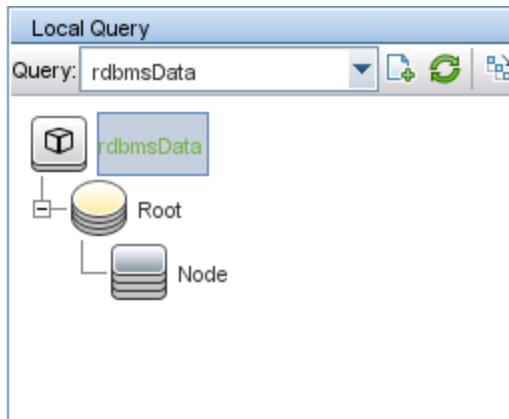
The following steps illustrate how to create an XML configuration file for the `rdbmsData` query described in previous sections.

To map a CI type's attributes to web service fields:

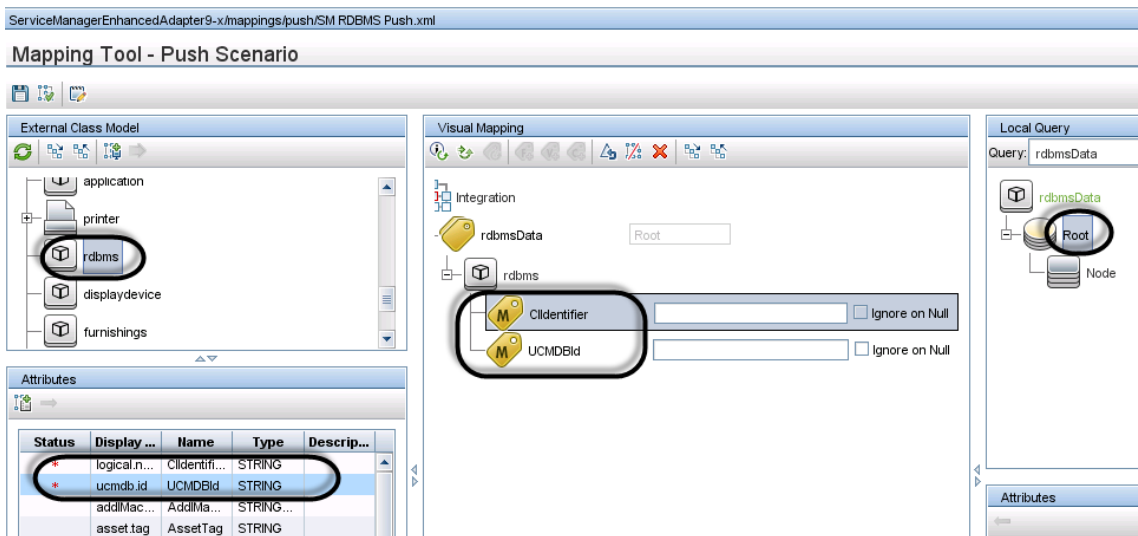
1. Log in to UCMDB with an administrator account.
2. Navigate to **Data Flow Management > Adapter Management > ServiceManagerEnhancedAdapter9-x**, and select the adapter.
3. Click the **Create New Resource** icon .
4. Select **New Configuration File**.
5. Enter the full file name: `<AdapterID>/mappings/push/<filename>`. For example, `ServiceManagerEnhancedAdapter9-x/mappings/push/SM RDBMS Push.xml`.
6. Click **OK**. The mapping file is created.
7. Double-click the new mapping file to open it with the Visual Mapping tool editor.
8. In the Local Query pane, click the **Add TQL Queries** icon to add the `rdbmsData` query.



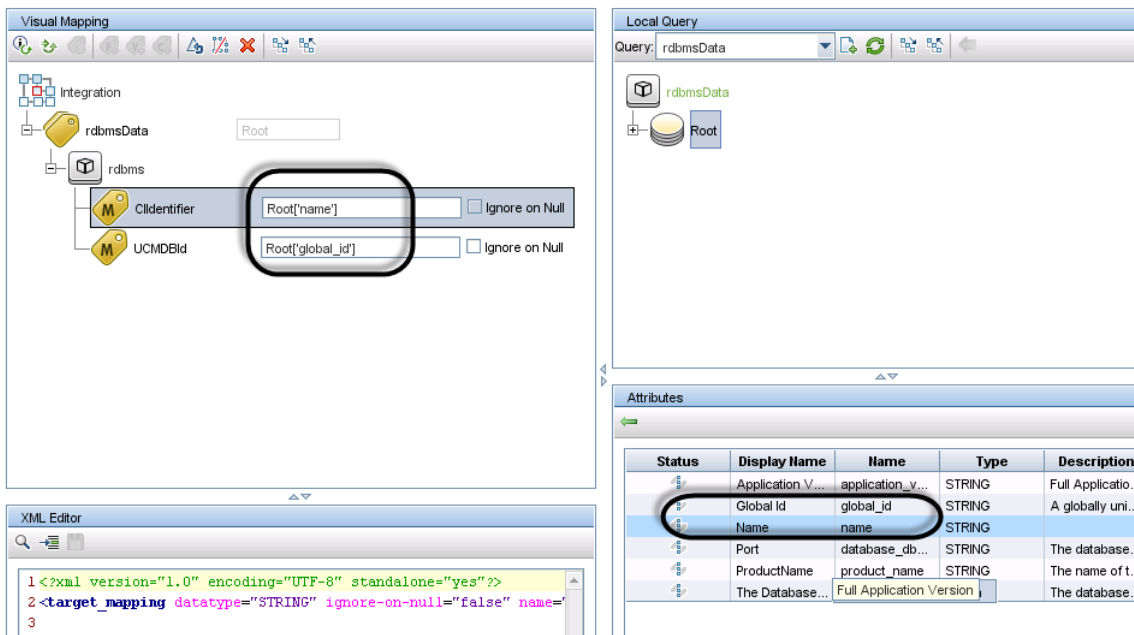
9. Click **OK**.



10. Drag and drop the node “Root” from the Local Query pane into the mapping area.
11. Drag and drop the node “RDBMS” from the External Class Model pane into the mapping area.
12. Drag and drop SM attributes you want into the mapping area.



13. Drag and drop the relevant UCMDB attributes into the mapping area.



14. Save the new mapping file.

**Note:** When you create or edit and then save a configuration file in Adapter Management, UCMDB automatically restarts the adapter with the new configuration file.

### Using Groovy scripts in XML Configuration Files

The Service Manager Enhanced Generic Adapter uses XML and Groovy mapping scripts for four field mapping scenarios: One to One, One to Many, Many to Many, and Value Conversion. Groovy scripts have to be used for complex scenarios. For more information, see ["Update the Configurations for Custom CI Types in UCMDB" on page 94.](#)

### How to Add a CI Relationship Type to the Integration for Data Push

Once you have added a new CI type to the integration and have created relationships between it and other CI types in UCMDB, for each of these relationship types you need to perform the following tasks so that UCMDB can push each type of relationships to Service Manager.

As an example, the following steps illustrate how you add a relationship type named **Ownership** (between the **Cost** and **CostCategory** CI types) to the integration for data push.

**Note:** These steps assume that you have already added the **Cost** and **CostCategory** CI types to the

integration.

1. Create a query to push the relationship type.  
See ["How to Create a Query to Push a Relationship Type" below](#).
2. Map the relationship query to a Service Manager web service object.  
See ["How to Map a Relationship Type Query to the Service Manager Web Service Object" on page 245](#).
3. Create an XML configuration file for the new relationship type.  
See ["How to Create an XML Configuration File for a Relationship Type" on page 246](#).

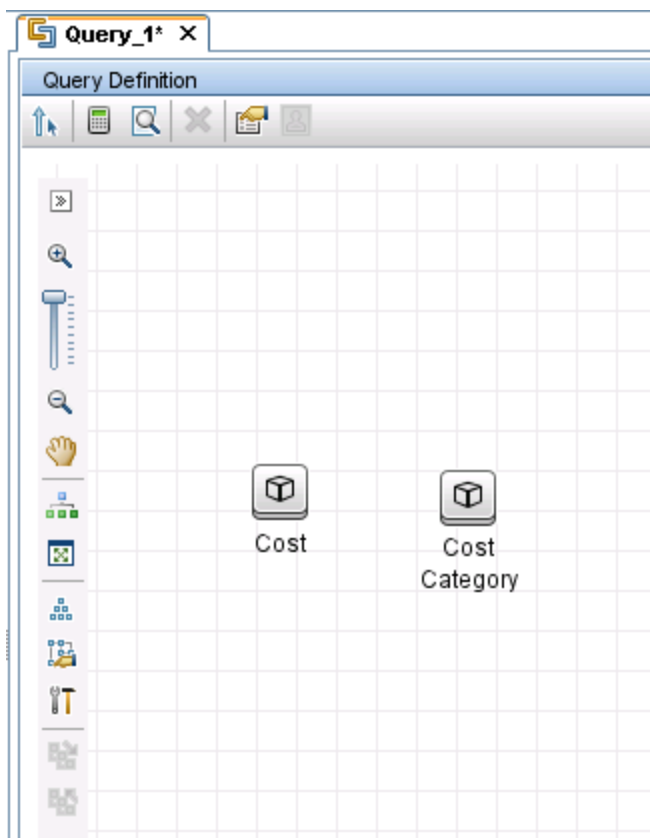
## How to Create a Query to Push a Relationship Type

Once you have created a relationship between two CI types, you must create a query to push the relationship to Service Manager.

**Note:** Any query you create must conform to the ["Query Requirements" on page 188](#).

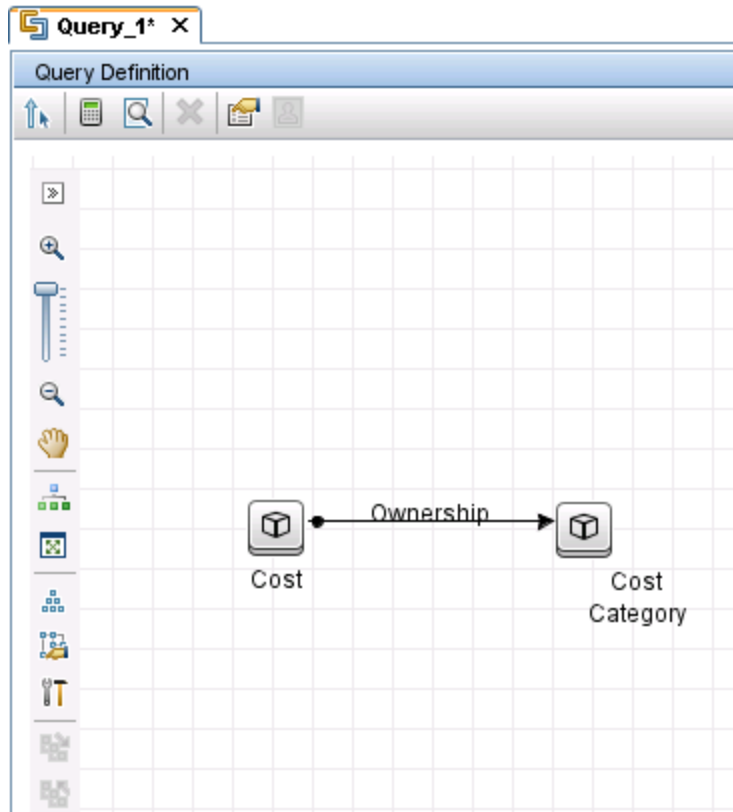
To create a new query called **Cost CostCategory Ownership Relations** for Ownership relationships between the Cost and CostCategory CI types:

1. Log in to UCMDB as an administrator.
2. Navigate to **Modeling > Modeling Studio**.
3. Click **New > Query**. The Query Definition pane is displayed.
4. From the CI Type Selector, drag the Cost and CostCategory CI types to the query pane.



5. Create an Ownership relationship from Cost to CostCategory.
  - a. Click the **Create Relationship** icon.
  - b. Select the **Cost** node, and drag the arrow from it to the CostCategory node.
  - c. Select **Regular Relationship**, and click **OK**.
  - d. Select **Connection > Ownership**, enter `Ownership` for Relationship Name, and click **OK**. An

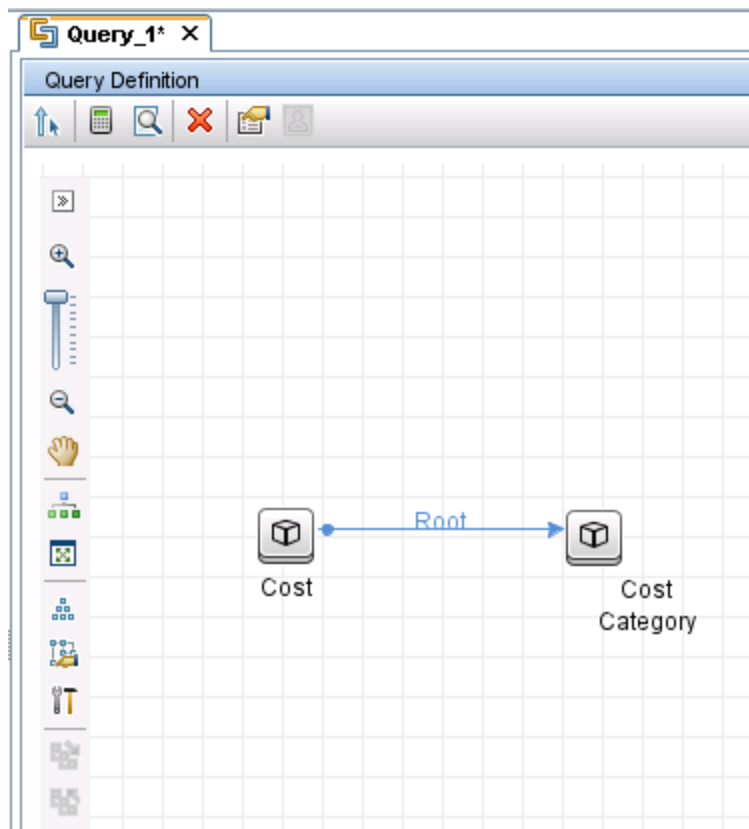
Ownership relationship is created between the CI types.



- 6. Right-click the relationship arrow, and select **Relationship Properties**.
- 7. Change the element name from **Ownership** to **Root** (or a name starting with “Root\_”), and then click **OK**.

The screenshot shows the 'Relationship Properties' dialog box. It has a title bar with the text 'Relationship Properties'. Below the title bar is a small icon of a hand pointing to a document and the text 'Relationship Properties' followed by 'Enables you to add attributes, cardinality, qualifiers and CI specific conditions'. There are two input fields: 'Element name:' with the text 'Root' entered, and 'Element type:' with a dropdown menu showing 'Ownership'. To the right of these fields is a checked checkbox labeled 'Show element in query results'.





8. Click the **Save** icon, and save the query as described in the following.
  - a. Enter a query name. For example, `Cost CostCategory Ownership Relations Push`.
  - b. Select the **Integration > Service Manager > Push** folder.
  - c. Click **OK**.

The query is now created. You are ready to add this query to the push configuration file (`smPushConf.xml`).

### How to Map a Relationship Type Query to the Service Manager Web Service Object

Once you have created a query for a relationship type, you need to map the query to the Relationship web service object in SM as described in the following steps.

1. Navigate to **Data Flow Management > Adapter Management > ServiceManagerEnhancedAdapter9-x > Configuration Files**.
2. Click the `smPushConf.xml` file.

3. Add a mapping entry by copying an existing one for relationship push. For example, add the following mapping entry.

```
<tql name="SM Layer2 Topology Relations Push 2.0"
  resourceCollectionName="Relationships" resourceName="Relationship" />
```

4. Change the TQL query name to the name of the query you created for the relationship type. For example, `Cost CostCategory Ownership Relations Push`.

```
<tql name="Cost CostCategory Ownership Relations Push"
  resourceCollectionName="Relationships" resourceName="Relationship" />
```

5. Click **Save** to save the configuration file.

## How to Create an XML Configuration File for a Relationship Type

To create an XML configuration file for a relationship type:

1. Log in to UCMDB with an administrator account.
2. Navigate to **Data Flow Management > ServiceManagerEnhancedAdapter9-x**, and select the adapter.
3. Click the **Create New Resource** icon .
4. Select **New Configuration File**.
5. Enter the full file name: `<AdapterID>/mappings/push/<filename>`. For example, `ServiceManagerEnhancedAdapter9-x/mappings/push/SM Cost CostCategory Ownership.xml`.
6. Click **OK** to save to the new file. The new file appears in the list of configuration files.
7. Open the new file in the XML Editor of the Visual Mapping interface.
8. Copy the content from an existing mapping file to overwrite the content of the new file in the XML Editor pane, and update query name to the name of the query you created (`Cost CostCategory Ownership Relations Push`).

```
<?xml version="1.0" encoding="UTF-8"?>
  <integration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="../mappings_schema.xsd">
    <info>
      <source name="UCMDB"      version="10.20" vendor="HP"/>
```

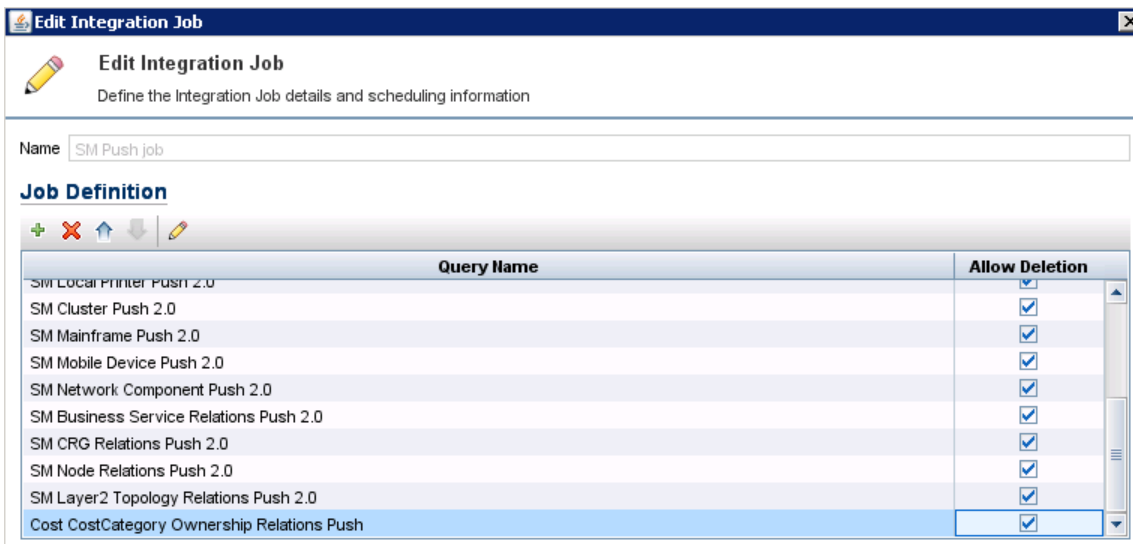
```

        <target name="SM"          version="9.40"    vendor="HP"/>
    </info>
    <import>
        <scriptFile path="mappings.scripts.SMPushFunctions"/>
    </import>
    <!--
        Push Cost to Cost Category Relations.
    -->
    <target_entities>
        <source_instance query-name="Cost CostCategory Ownership Relations
    Push" root-element-name="Root">
            <target_entity name="Relationship">
                <target_mapping name="RelationshipType" datatype="STRING"
    value="SMPushFunctions.getDisplayName(Root['element_type'],ClassModel)"/>
                <target_mapping name="ParentCI" datatype="STRING"
    value="SMPushFunctions.getEndId(OutputCI.getExternalId().getEnd1Id())"/>
                <target_mapping name="ChildCIs" datatype="STRING_LIST"
    value="[SMPushFunctions.getEndId(OutputCI.getExternalId().getEnd2Id())]"/>
            </target_entity>
        </source_instance>
    </target_entities>
</integration>

```

9. Click the **Save** icon to save the new configuration file.

Now, you have added the new relationship type to the integration. Next, you need to add the new relationship query to a data push job, as shown in the following figure. For detailed steps, see ["How to Add a Custom Query to an Integration Job" on the next page.](#)



**Caution:** Before you run the relationship push job, make sure you have already added the relevant CI types (**Cost** and **CostCategory** in this example) to the integration and pushed the relevant CIs to Service Manager.

## How to Add a Custom Query to an Integration Job



In order for the integration to synchronize your custom CI types, attributes, and relationships between your Service Manager and UCMDB systems, you must add your custom queries to a data push or population job.

The following steps illustrate how to add an custom query named **rdbmsData**, which you created previously (see ["How to Create a Query to Synchronize the CI Type" on page 228](#)), to a data push job. The steps for adding a population query to a population job are similar.

To add a custom query to a data push or population job:

1. Log in to UCMDB as an administrator.
2. Navigate to **Data Flow Management > Integration Studio**.
3. Click the name of your Service Manager integration point. For example, **sm\_integration**.
4. Click the **Data Push** tab.

**Note:** To add a query to a population job, click the **Population** tab instead.

5. Click the name of the integration job. For example, **SM Configuration Item Push job**.
6. Click the **Edit** icon .
7. Click the **Add** icon .
8. Click **Integration > Service Manager > Push > rdbmsData**.

**Note:** For population, navigate to **Integration > Service Manager > Population**, and click the population query instead.

9. Click **OK** to add the custom query.

10. Enable the **Allow Deletion** option for the query to allow the integration job to delete removed data.
11. Click **OK** to close the Update Job Definition window.

## How to Add a CI Type, Attribute or Relationship Type to the Integration for Population

All out-of-the-box mapping files for Population can be found by navigating to **Data Flow Management > Adapter Management > ServiceManagerEnhancedAdapter9-x > ServiceManagerEnhancedAdapter9-x/mappings/population/<Name of Mapping File>**.

With the Visual Mapping tool, you can add a CI Type, CI attribute, or relationship to the integration for Population by following similar steps to those for data push. Additionally, you also need to add your population queries to a population job. Refer to the following steps for data push:

["How to Add a CI Attribute to the Integration for Data Push" on page 210](#)

["How to Add a CI Type to the Integration for Data Push" on page 224](#)

["How to Add a CI Relationship Type to the Integration for Data Push" on page 241](#)

["How to Add a Custom Query to an Integration Job" on the previous page](#)

## How to Enable or Disable UCMDB ID Pushback for a CI Type

When a new CI is created in UCMDB, a UCMDB ID value is assigned to the CI. When CIs are synchronized from SM to UCMDB through population, the UCMDB ID Pushback feature can push their UCMDB ID values back to SM. The UCMDB ID field is then used as a flag to indicate if a CI has already been synchronized to UCMDB through population.

There are also cases when you do not want to enable the UCMDB ID pushback feature. For example, the `scheduled_downtime` CI type in UCMDB does not physically exist in SM. Instead, the integration retrieves Scheduled Downtime information from several entities in SM and then synchronizes the information to `scheduled_downtime` CIs in UCMDB. For this reason, the UCMDB ID Pushback feature is disabled by default for `scheduled_downtime` CIs; otherwise a pushback error will occur during population.

To enable UCMDB ID pushback for a CI type, follow these steps:

1. Add the following entries to your population configuration file for the CI type (for example, **My Business Service Population.xml**).

```
<target_mapping name="sm_id" datatype="STRING" value="bizservice['CIName']"/>
<target_mapping name="global_id" datatype="STRING" value="bizservice['UCMDBId']
"/>
```

**Note:** In this example, **bizservice** is the CI Type display name defined in SM and also the external class model name displayed in the UCMDB Visual Mapping interface, as shown in the following figures.

### Manage CI Types

CI Type Description:	Business Service
CI Type:	bizservice
Bitmap:	_box
Format Name:	configurationItem
Attr File:	bizservice
Join Def:	joinbizservice
Print Format Name:	
Bulk Update Format Name:	device.businessservice.bulkupdate
Active:	<input checked="" type="checkbox"/>

Sub Types
Business Service
Application Service
Infrastructure Service

### Mapping Tool - Push Scenario

The screenshot shows two panels. The left panel, 'External Class Model', displays a tree structure with nodes: 'External Entity Class Model', 'networkcomponents', 'bizservice' (highlighted with a red circle), and 'runningsoftware'. The right panel, 'Visual Mapping', shows a diagram with 'Integration' at the top, 'SM Business Service ...' as the root (highlighted with a red circle), and 'bizservice' (highlighted with a red circle) connected to the root.

2. Save the XML configuration file.
3. Open the smPopConf.xml file, and make sure that the UCMDB ID Pushback setting for the CI type is either not present or is set to true:

```

<pushback>
  <type name="business_service" enable="true"/>
</pushback>

```

Where: <type name> is the name of the CI type in UCMDB.

To disable UCMDB ID Pushback feature for a CI type, follow these steps:

1. Open the smPopConf.xml file.
2. In the <pushback> section, insert one line for the CI type:

```

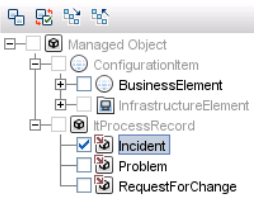
<config>
  <pushback>
    <type name="scheduled_downtime" enable="false"/>
    <type name="business_service" enable="false"/>
  </pushback>
  <mapping>
    <tql name="SM Business Service Population 2.0">
      ...
    </tql>
    ...
  </mapping>
</config>

```

## How to Add an Attribute of a Supported CI Type for Federation

Out-of-the-box, the integration supports federation for three external CI types in UCMDB: Incident, Problem, and RequestForChange. For each of the supported CI types, there is a list of attributes in UCMDB that you can map to Service Manager web service objects for federation. The following figure shows the out-of-the-box UCMDB CI attributes available for the Incident CI type.

### Supported and Selected CI Types



- Managed Object
  - ConfigurationItem
    - BusinessElement
    - InfrastructureElement
  - #ProcessRecord
    - Incident**
    - Problem
    - RequestForChange

### CI Type Retrieval Mode

Retrieve CIs of selected CI Type  
 Retrieve selected attributes

Selected attributes will be retrieved from the Integration. The CIs must already exist in the UCMDB.

#### Select Attributes

- ActiveProcess
- Actual Deletion Period
- Allow CI Update
- Category
- ClosedTime
- CompletionCode
- Consumer Tenants
- Container
- Create Time
- Created By
- Deletion Candidate Period
- Description
- Display Label
- Enable Aging
- Escalated
- ExternalProcessReference
- Global Id
- ImpactScope
- IncidentStatus
- IncidentType
- Is Candidate For Deletion
- Last Access Time
- LastModifiedTime
- Name
- Note
- Origin
- OutageEndTime

For example, to add an SM Incident attribute for federation, you need to expose the field in the SM UcmdbIncident web service object and then map it to an appropriate UCMDB attribute (if one does not already exist, you need to create it in UCMDB first).

The following figure shows the fields that are exposed in the UcmdbIncident web service object in Service Manager.

External Access Definition

Service Name:

Name:

Object Name:

Allowed Actions  
  Expressions  
  Fields  
  RESTful

Field	Caption	Type
logical.name	ConfigurationItem	
number	IncidentID	
brief.description	BriefDescription	
subcategory	Category	
severity	Urgency	
open.time	OpenTime	DateTimeType
update.time	UpdatedTime	DateTimeType
close.time	ClosedTime	DateTimeType
problem.status	IMTicketStatus	
affected.item	Service	
priority.code	PriorityCode	
initial.impact	ImpactScope	



You can expose more fields so that more Incident attributes can be federated to UCMDB. As an example, the following describes how to add the “action” field in the Service Manager **probsummary** (Incident) file for federation, by mapping it to a new UCMDB attribute named **details**.

**Note:** On the Incident form in Service Manager, the “action” field is labeled “Description,” which is used to describe the incident record in more detail. See the following figure.

The screenshot shows the 'Incident - IM10001' form. The 'Title' field is empty, and the 'Description' field contains the text: 'A description of the incident in more detail'. A yellow highlight is placed over the 'Description' field, and a tooltip is visible over it. The tooltip contains the following text: 'Cannot add attachment to OneNote Shared directory.', 'Cannot add attachment to OneNote Shared directory.', 'A description of the incident in more detail', 'file=probsummary', 'form=im.incident.closure', and 'field=action'.

To add an attribute of a supported CI type for federation:

**1. Add the SM attribute to its web service object.**

The following example describes how to expose the SM “action” field of Incident in the UcmdbIncident web service object.

- a. Log in to Service Manager as a system administrator.
- b. Navigate to **Tailoring > Web Services > Web Service Configuration**.
- c. Enter the following field values, and then click **Search**.
  - **Service Name:** ucmdbIntegration
  - **Name:** probsummary

The UcmdbIncident web service object is displayed.

- d. On the **Fields** tab, add the following row:
  - **Field:** action
  - **Caption:** Description

**External Access Definition**

Service Name:

Name:

Object Name:

Allowed Actions
  Expressions
  Fields
  RESTful

Field	Caption	Type
logical.name	ConfigurationItem	
number	IncidentID	
brief.description	BriefDescription	
subcategory	Category	
severity	Urgency	
open.time	OpenTime	DateTimeType
update.time	UpdateTime	DateTimeType
close.time	ClosedTime	DateTimeType
problem.status	IMTicketStatus	
affected.item	Service	
priority.code	PriorityCode	
initial.impact	ImpactScope	
action	Description	

e. Save the web service object.

2. **Map the SM attribute to a UCMDB attribute.**

The following example describes how to map the SM “action” attribute to a new UCMDB attribute named **details**.

- a. Log in to UCMDB as an administrator.
- b. Navigate to **Modeling > CI Type Manager**.
- c. Navigate to **ItProcessRecord > Incident**, and open its properties pane.
- d. Click the **Add** icon to add a new attribute named **details** to the Incident CI type.
  - name: details
  - Display Name: Details
  - Description: Incident details
  - Attribute Type: select **Primitive > String**
  - Value Size: 500

**Add Attribute**

Details | Advanced | UCMDB Browser Qualifiers

Attribute Name:

Display Name:

Scope:

Description:

Attribute Type:

Primitive  Enumeration/List

Value Size:

Default value policy:  Enable default value

Default Value:

OK Cancel

- e. Save the Incident CI type record.
- f. Navigate to **Data Flow Management > Adapter Management > ServiceManagerEnhancedAdapter9-x > Configuration Files.**
- g. Edit the relevant federation mapping file (**ServiceManagerEnhancedAdapter9-x/mappings/federation/SM Incident 2.0.xml**) to add a mapping entry for the new attribute.

```
<target_entities>
  <source_instance query-name="SM Incident 2.0" root-element-
name="ucmdbIncident">
    <target_entity name="incident">
      <target_mapping name="reference_number"
datatype="STRING" value="ucmdbIncident['IncidentID']"/>
      <target_mapping name="name"
datatype="STRING" value="ucmdbIncident['BriefDescription']"/>
      <target_mapping name="priority"
datatype="STRING" value="SMFederationFunctions.getEnumValue
(ucmdbIncident['PriorityCode'], 'Priority')"/>
    </target_entity>
  </source_instance>
</target_entities>
```

```

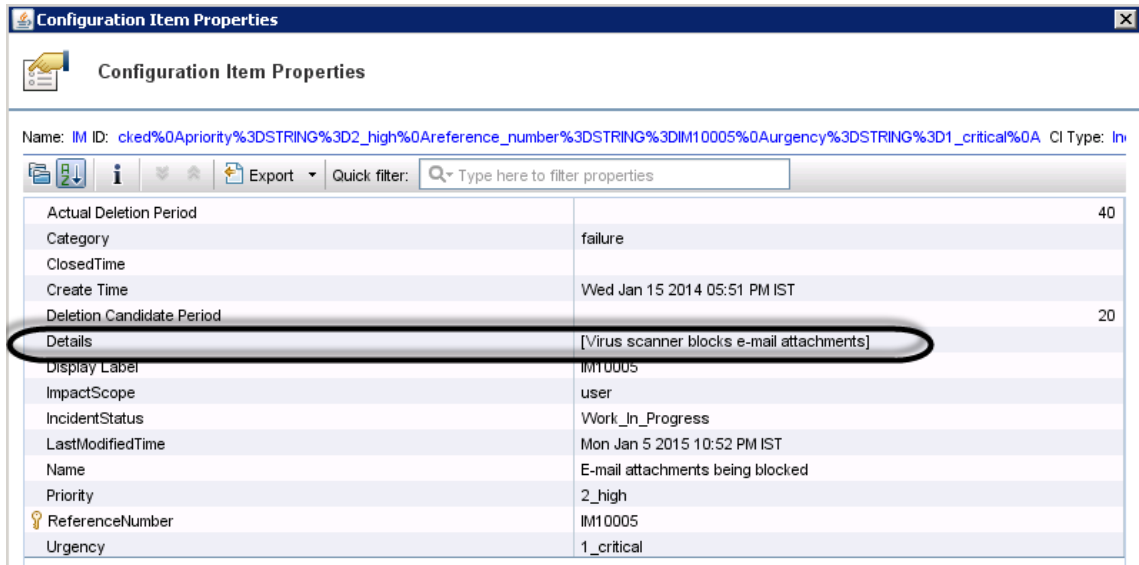
        <target_mapping name="incident_status"
datatype="STRING"
value="SMFederationFunctions.firstLetterToLowerAndReplaceSpaceWithUnders
core(ucmdbIncident['IMTicketStatus'])"/>
        <target_mapping name="category"
datatype="STRING"
value="SMFederationFunctions.replaceSpaceWithUnderscore(ucmdbIncident
['Category'])"/>
        <target_mapping name="closed_time"
datatype="DATE"    value="SMFederationFunctions.convertDate
(ucmdbIncident['ClosedTime'])"/>
        <target_mapping name="create_time"
datatype="DATE"    value="SMFederationFunctions.convertDate
(ucmdbIncident['OpenTime'])"/>
        <target_mapping name="last_modified_time"
datatype="DATE"    value="SMFederationFunctions.convertDate
(ucmdbIncident['UpdatedTime'])"/>
        <target_mapping name="impact_scope"
datatype="STRING"  value="SMFederationFunctions.getEnumValue
(ucmdbIncident['ImpactScope'],'ImpactScope')"/>
        <target_mapping name="urgency"
datatype="STRING"  value="SMFederationFunctions.getEnumValue
(ucmdbIncident['Urgency'],'Urgency')"/>
        <target_mapping name="details" datatype="STRING"
value="ucmdbIncident['Description']"/>
    </target_entity>
</source_instance>
</target_entities>

```

h. Click **Save** to save the file.

Now the Description (field name: action) attribute of SM Incident has been added to the integration for federation. You can run an Incident federation query in the UCMDB Modeling Studio to see if the SM Description data is properly federated. For information about how to run a federation query, see ["Examples of Using Federation" on page 120](#).

The following figure shows an example where the Description of an SM incident record has been federated to UCMDB as **Details**.



## Troubleshooting

When data push and population errors occur, you can check the error messages and the integration log files to identify the root causes and fix the errors. This chapter describes the general troubleshooting steps, as well as typical errors and solutions.

This section includes:

- ["Troubleshooting Data Push Issues" below](#)
- ["Troubleshooting Population Issues" on page 267](#)

## Troubleshooting Data Push Issues

When data push errors or problems occur, you can check the error messages and the log file to figure out the root causes and then fix the errors.

This integration uses the following error codes for data push.

SM Error Code	Description	UCMDB Error Code
-1	Server Application Error from SM	900008
-4	401 Authorization error from SM	900001
3	Data Locked Error from SM	900003

SM Error Code	Description	UCMDB Error Code
9	Restful configuration not found	900009
51	Changed By Another Process Error from SM	900004
70	Invalid Action Error from SM	900005
71	Validation Check Failed error from SM	900002
881	Invalid Data Error from SM	900007
882	Cannot find CI in SM for relationship operation	900010
	Failed to create message in SM	900013
	Communication with SM Failed	900014
	Other Error from SM	900099

When a data push job has failed, the job status becomes **Failed**. Troubleshoot the failed job as follows:

- Check the error messages of the failed job in the Universal CMDB studio.  
See ["How to Check the Error Message of a Failed Push Job"](#) below.
- Check the log file for more details.  
See ["How to Check the Push Log File"](#) on page 261.

When a data push job was completed with failures of partial records, the job status becomes **Completed**. Troubleshoot the failed records as follows:

- Check the error messages of failed CIs in the Universal CMDB studio.  
See ["How to Check the Error Messages of Failed CIs or Relationships in a Push Job"](#) on page 260.
- Check the log file for more details.  
See ["How to Check the Push Log File"](#) on page 261.

## How to Check the Error Message of a Failed Push Job

To check the error message of a failed job:

1. Log in to UCMDB as an administrator.
2. Navigate to **Data Flow Management > Integration Studio**.
3. Select the integration point for this integration from Integration Point list.

4. Click the **Data Push** tab.
5. Select the job from Integration Jobs.
6. Click the **Job Errors** sub-tab, and double-click the Severity of a message from the list.  
A popup window displays the detailed error message of this failed job. The following is an excerpt of a sample error message indicating that no mapping configuration was found for the push TQL query.

```

java.lang.RuntimeException: No mapping is found for TQL: "SM Business Service
  Push 2.0", or Cannot retrieve the mapping from SM side by QueryNodeName
  [bizservice", please configure in smPushConf.xml or SM configuration
    at
  com.mercury.topaz.fcmdb.adapters.serviceDeskAdapter.push.SmGenericPusher.pus
  h(SmGenericPusher.java:119)
    ... 35 more
  --- End of probe-side exception ---

    at
  com.hp.ucmdb.discovery.probe.agents.probemgr.adhocoTasks.AdHocProbeRequestOpe
  ration.convertThrowableToStringSafeException
  (AdHocProbeRequestOperation.java:86)
    at
  com.hp.ucmdb.discovery.probe.agents.probemgr.adhocoTasks.AdHocProbeRequestOpe
  ration.performAction(AdHocProbeRequestOperation.java:77)
    at
  com.hp.ucmdb.discovery.probe.agents.probemgr.taskdispatcher.AdHocTaskDispatc
  her.dispatchTask(AdHocTaskDispatcher.java:70)
    at sun.reflect.GeneratedMethodAccessor59.invoke(Unknown Source)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke
  (DelegatingMethodAccessorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:601)
    at com.sun.jmx.mbeanserver.StandardMBeanIntrospector.invokeM2
  (StandardMBeanIntrospector.java:111)
    at com.sun.jmx.mbeanserver.StandardMBeanIntrospector.invokeM2
  (StandardMBeanIntrospector.java:45)
    at com.sun.jmx.mbeanserver.MBeanIntrospector.invokeM
  (MBeanIntrospector.java:235)
    at com.sun.jmx.mbeanserver.PerInterface.invoke(PerInterface.java:138)
    at com.sun.jmx.mbeanserver.MBeanSupport.invoke(MBeanSupport.java:252)
    at javax.management.StandardMBean.invoke(StandardMBean.java:405)
    at com.sun.jmx.interceptor.DefaultMBeanServerInterceptor.invoke
  (DefaultMBeanServerInterceptor.java:819)
    at com.sun.jmx.mbeanserver.JmxMBeanServer.invoke(JmxMBeanServer.java:792)
    at javax.management.MBeanServerInvocationHandler.invoke
  (MBeanServerInvocationHandler.java:305)
    at org.springframework.jmx.access.MBeanClientInterceptor.doInvoke

```

```
(MBeanClientInterceptor.java:405)
    at org.springframework.jmx.access.MBeanClientInterceptor.invoke
(MBeanClientInterceptor.java:353)
    at org.springframework.aop.framework.ReflectiveMethodInvocation.proceed
(ReflectiveMethodInvocation.java:172)
    at org.springframework.aop.framework.JdkDynamicAopProxy.invoke
(JdkDynamicAopProxy.java:202)
    at com.sun.proxy.$Proxy57.dispatchTask(Unknown Source)
    at
com.hp.ucmdb.discovery.probe.agents.probegw.managementtasks.adhocktasks.Adhoc
Thread.run(AdhocThread.java:54)
... 3 more
```

## How to Check the Error Messages of Failed CIs or Relationships in a Push Job

When a data push job is completed with failures of partial records, in the Universal CMDB studio, you can check the detail error message for each failed record.

To check the error messages of failed records in a data push job:

1. Log in to UCMDB as an administrator.
2. Navigate to **Data Flow Management > Integration Studio**.
3. Select the integration point for this integration from Integration Point list.
4. Click the **Data Push** tab.
5. Select the job from Integration Jobs.
6. Click the **Query Status** sub-tab.
7. Double-click a query with failures. The Error Message and CI Count for each failed CI Type are displayed.

The screenshot shows a web interface with tabs for 'Statistics', 'Query Status', and 'Job Errors'. The 'Job Errors' tab is active, displaying a table with the following data:

Error Message	CI Type	CI Count
Validation Check Fails Error from SM	BusinessApplication	3
Validation Check Fails Error from SM	BusinessService	1
Validation Check Fails Error from SM	InfrastructureService	2

8. Double-click an error message. A list of failed records is displayed.



9. Double-click a failed record.

The detailed error message of the record is displayed.

## How to Check the Push Log File

You need to set the Development adapter log level to DEBUG or TRACE, so that you can check the push result tree nodes of UCMDB, and then send messages to and receive messages from Service Manager. Alternatively, you can set the log level to TRACE so as to check the conversion period based on a mapping file.

**Note:** To troubleshoot push, population, and federation issues, you need to set the Development adapter log level to DEBUG or TRACE in the **fcmdb.properties** and **fcmdb.push.properties** files, which are located in the `\conf\log` folder of your Data Flow Probe or Integration Service installation. Additionally, you can view push, population and federation log information in the **fcmdb.push.all.log** file, which is located in the `\runtime\log` folder of your Data Flow Probe or Integration Service installation. Your Data Flow Probe or Integration Service may be installed on the same host as your UCMDB Server or on a separate host.

To set the Development adapter log level to DEBUG or TRACE:

1. Log in as an administrator to the host on which your UCMDB Data Flow Probe or Integration Service is installed.
2. Navigate to the `<UCMDB installation folder>\DataFlowProbe\conf\log` folder or the `<UCMDB installation folder>\UCMDBServer\integrations\conf\log`. For example:

`C:\hp\UCMDB\DataFlowProbe\conf\log\`

Or `C:\hp\UCMDB\UCMDBServer\integrations\conf\log`

3. Open the **fcmdb.properties** file in a text editor, and change the log level to `DEBUG` or `TRACE`. For example:

```
#loglevel can be any of TRACE DEBUG INFO WARN ERROR FATAL
loglevel=DEBUG
def.file.max.size=5000KB
def.files.backup.count=10
msg.layout=%d %-5p [%t] - %m%n
```

4. Open the **fcmdb.push.properties** files in a text editor, and change the log4j log level to `DEBUG` or

TRACE. For example:

```
### UCMDB log4j Properties
log.file.path=log/${log.folder.path.output}
#loglevel can be any of TRACE DEBUG INFO WARN ERROR FATAL
loglevel=DEBUG
def.file.max.size=5000KB
def.files.backup.count=10
msg.layout=%d %-5p [%t] - %m%n
```

5. Save the files.

#### To check the push log file:

1. Log in as an administrator to the host on which your UCMDB Data Flow Probe or Integration Service is installed.
2. Navigate to the *<UCMDB installation folder>\DataFlowProbe\runtime\log* folder or the *<UCMDB installation folder\UCMDBServer\integrations\runtime\log* folder.
3. Open the **fcmdb.push.all.log** file in a text editor.

## Typical Push Errors and Solutions

This section describes typical error messages that may occur during data push, as well as their solutions.

This section includes:

- ["Query not Configured in smPushConf.xml" below](#)
- ["Mapping File not Well Formed" on page 264](#)

### Query not Configured in smPushConf.xml

#### Sample Configuration

The TQL queries used for pushing relationships must be configured in the smPushConf.xml file.

```
<config>
  <mapping>
    <!--
      Wildcard is supported for TQL names while you configure the mapping now,
      you can add the '*' at the end of the TQL name while configuring the
      mapping.
```

We use the wildcard for mapping in OOTB, so each time you do save as to a TQL, you could still push it to SM without manual changing the mapping. For e.g., if you have saved the <TQL\_name> query to <TQL\_name>\_1, and <TQL\_name>\_2, the TQL name is specified as <TQL\_name>\* in this configuration file, and the integration will automatically use this mapping entry on all of the three TQLs. However, using the exact TQL name to configure the mapping is still supported.

```
-->
    <tql name="SM Business Service Relations Push 2.0"
resourceCollectionName="Relationships" resourceName="Relationship"/>
    <tql name="SM CRG Relations Push 2.0"
resourceCollectionName="Relationships" resourceName="Relationship"/>
    <tql name="SM Node Relations Push 2.0"
resourceCollectionName="Relationships" resourceName="Relationship"/>
    <tql name="SM Layer2 Topology Relations Push 2.0"
resourceCollectionName="Relationships" resourceName="Relationship"/>
  </mapping>
</config>
```

## Error Message

The push job fails with a “Failed” status. From both the log file and the detail error message of the failed job in the Universal CMDB studio (see ["How to Check the Error Message of a Failed Push Job" on page 258](#)), you receive an error like the following:

```
java.lang.RuntimeException: No mapping is found for TQL: "SM Business Service Push
2.0", or Cannot retrieve the mapping from SM side by QueryNodeName [bizservice",
please configure in smPushConf.xml or SM configuration
    at
com.mercury.topaz.fcmdb.adapters.serviceDeskAdapter.push.SmGenericPusher.push
(SmGenericPusher.java:119)
    ... 35 more
--- End of probe-side exception ---

    at
com.hp.ucmdb.discovery.probe.agents.probemgr.adhoctasks.AdHocProbeRequestOperati
on.convertThrowableToStringSafeException(AdHocProbeRequestOperation.java:86)
    at
com.hp.ucmdb.discovery.probe.agents.probemgr.adhoctasks.AdHocProbeRequestOperati
on.performAction(AdHocProbeRequestOperation.java:77)
    at
com.hp.ucmdb.discovery.probe.agents.probemgr.taskdispatcher.AdHocTaskDispatcher.
dispatchTask(AdHocTaskDispatcher.java:70)
    at sun.reflect.GeneratedMethodAccessor59.invoke(Unknown Source)
```

```

    at sun.reflect.DelegatingMethodAccessorImpl.invoke
(DelegatingMethodAccessorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:601)
    at com.sun.jmx.mbeanserver.StandardMBeanIntrospector.invokeM2
(StandardMBeanIntrospector.java:111)
    at com.sun.jmx.mbeanserver.StandardMBeanIntrospector.invokeM2
(StandardMBeanIntrospector.java:45)
    at com.sun.jmx.mbeanserver.MBeanIntrospector.invokeM
(MBeanIntrospector.java:235)
    at com.sun.jmx.mbeanserver.PerInterface.invoke(PerInterface.java:138)
    at com.sun.jmx.mbeanserver.MBeanSupport.invoke(MBeanSupport.java:252)
    at javax.management.StandardMBean.invoke(StandardMBean.java:405)
    at com.sun.jmx.interceptor.DefaultMBeanServerInterceptor.invoke
(DefaultMBeanServerInterceptor.java:819)
    at com.sun.jmx.mbeanserver.JmxMBeanServer.invoke(JmxMBeanServer.java:792)
    at javax.management.MBeanServerInvocationHandler.invoke
(MBeanServerInvocationHandler.java:305)
    at org.springframework.jmx.access.MBeanClientInterceptor.doInvoke
(MBeanClientInterceptor.java:405)
    at org.springframework.jmx.access.MBeanClientInterceptor.invoke
(MBeanClientInterceptor.java:353)
    at org.springframework.aop.framework.ReflectiveMethodInvocation.proceed
(ReflectiveMethodInvocation.java:172)
    at org.springframework.aop.framework.JdkDynamicAopProxy.invoke
(JdkDynamicAopProxy.java:202)
    at com.sun.proxy.$Proxy57.dispatchTask(Unknown Source)
    at
com.hp.ucmdb.discovery.probe.agents.probegw.managementtasks.adhocktasks.AdhocThre
ad.run(AdhocThread.java:54)
    ... 3 more

```

## Solution

Search for text string No mapping is found for TQL to find the name of the query that is not yet configured, and then add a mapping entry for the query in the `smPushConf.xml` file.

## Mapping File not Well Formed

### Sample Configuration

The "target\_entity name" should be **bizservice** (which is the CI type display name defined in Service Manager, and also the external class model name displayed in the UCMDB Visual Mapping tool interface); however you have configured a wrong name **businessservice**.

```

<?xml version="1.0" encoding="UTF-8"?>
  <integration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="../mappings_schema.xsd">

```

```

<info>
  <source name="UCMDB"    version="10.20" vendor="HP"/>
  <target name="SM"      version="9.40"  vendor="HP"/>
</info>
<import>
  <scriptFile path="mappings.scripts.SMPushFunctions"/>
</import>
<!--
  Push uCMDB CIT to SM Business Service.
-->
<target_entities>
  <source_instance query-name="SM Business Service Push 2.0" root-element-
name="Root" >
    <target_entity name="businessservice">
      <target_mapping name="UCMDBId" datatype="STRING" value="Root
['global_id']"/>
      <target_mapping name="CustomerId" datatype="STRING"
value="SMPushFunctions.getCustomerId(CustomerInformation)"/>
      <target_mapping name="Type" datatype="STRING"
value="'bizservice'"/>
      <target_mapping name="Subtype" datatype="STRING"
value="SMPushFunctions.getSMSubType(Root['element_
type'],ClassModel,'BizService')"/>
      <target_mapping name="ServiceProvider" datatype="STRING"
value="Root['provider']"/>
      <target_mapping name="ServiceName" datatype="STRING" value="Root
['display_label']"/>
      <target_mapping name="CIIdentifier" datatype="STRING"
value="Root['display_label']"/>
    </target_entity>
  </source_instance>
</target_entities>
</integration>

```

## Error Message

The data push job fails with a “Failed” status. From both the log file and the detailed error message of the failed job in the Universal CMDB studio (see [How to Check the Error Message of a Failed Push Job on page 258](#)), you receive an error like the following:

```

java.lang.RuntimeException: No mapping is found for TQL: "SM Business Service Push
2.0", or Cannot retrieve the mapping from SM side by QueryNodeName
[businessservice", please configure in smPushConf.xml or SM configuration
at
com.mercury.topaz.fcmdb.adapters.serviceDeskAdapter.push.SmGenericPusher.push
(SmGenericPusher.java:119)
... 35 more
--- End of probe-side exception ---

```

```

    at
com.hp.ucmdb.discovery.probe.agents.probemgr.adhocktasks.AdHocProbeRequestOperati
on.convertThrowableToStringSafeException(AdHocProbeRequestOperation.java:86)
    at
com.hp.ucmdb.discovery.probe.agents.probemgr.adhocktasks.AdHocProbeRequestOperati
on.performAction(AdHocProbeRequestOperation.java:77)
    at
com.hp.ucmdb.discovery.probe.agents.probemgr.taskdispatcher.AdHocTaskDispatcher.
dispatchTask(AdHocTaskDispatcher.java:70)
    at sun.reflect.GeneratedMethodAccessor59.invoke(Unknown Source)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke
(DelegatingMethodAccessorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:601)
    at com.sun.jmx.mbeanserver.StandardMBeanIntrospector.invokeM2
(StandardMBeanIntrospector.java:111)
    at com.sun.jmx.mbeanserver.StandardMBeanIntrospector.invokeM2
(StandardMBeanIntrospector.java:45)
    at com.sun.jmx.mbeanserver.MBeanIntrospector.invokeM
(MBeanIntrospector.java:235)
    at com.sun.jmx.mbeanserver.PerInterface.invoke(PerInterface.java:138)
    at com.sun.jmx.mbeanserver.MBeanSupport.invoke(MBeanSupport.java:252)
    at javax.management.StandardMBean.invoke(StandardMBean.java:405)
    at com.sun.jmx.interceptor.DefaultMBeanServerInterceptor.invoke
(DefaultMBeanServerInterceptor.java:819)
    at com.sun.jmx.mbeanserver.JmxMBeanServer.invoke(JmxMBeanServer.java:792)
    at javax.management.MBeanServerInvocationHandler.invoke
(MBeanServerInvocationHandler.java:305)
    at org.springframework.jmx.access.MBeanClientInterceptor.doInvoke
(MBeanClientInterceptor.java:405)
    at org.springframework.jmx.access.MBeanClientInterceptor.invoke
(MBeanClientInterceptor.java:353)
    at org.springframework.aop.framework.ReflectiveMethodInvocation.proceed
(ReflectiveMethodInvocation.java:172)
    at org.springframework.aop.framework.JdkDynamicAopProxy.invoke
(JdkDynamicAopProxy.java:202)
    at com.sun.proxy.$Proxy57.dispatchTask(Unknown Source)
    at
com.hp.ucmdb.discovery.probe.agents.probegw.managementtasks.adhocktasks.AdhocThre
ad.run(AdhocThread.java:54)
    ... 3 more

```

## Solution

Search for the relevant CI type in Service Manager to find the correct display name, and then modify the mapping file with the correct name.

**Tip:** You can easily fix such kind of validation issues by using the UCMDB Visual Mapping tool to generate the mapping file.

## Troubleshooting Population Issues

When population errors or problems occur, you can check the error messages and the population log file to identify the root causes and then solve the problems.

When a population job has failed, the job status becomes **Failed**. Troubleshoot the failed job as follows:

- Check the error message of the failed job in the Universal CMDB studio.  
See ["How to Check the Error Message of a Failed Population Job"](#) below and ["Typical Population Error Messages and Solutions"](#) on the next page.
- Check the log file for more details.  
See ["How to Check the Population Log File"](#) below.

### How to Check the Error Message of a Failed Population Job

While a population job fails, you can check the detailed error messages in the Universal CMDB studio.

To check the error message of a failed population job:

1. Log in to UCMDB as an administrator.
2. Navigate to **Data Flow Management > Integration Studio**.
3. Select the integration point for this integration.
4. Click the **Population** tab.
5. Select the failed job from Integration Jobs, and click the **Job Errors** subtab.
6. Double-click an error message from the list.  
A pop-up window opens to display the error details.

### How to Check the Population Log File

You can set the Development adapter log level to DEBUG so that you can check the population result tree nodes of UCMDB, and send messages to and receive messages from Service Manager.

Alternatively, you can set the log level to TRACE so as to check the conversion period based on a mapping file.

**Note:** To troubleshoot push, population, and federation issues, you need to set the Development adapter log level to DEBUG or TRACE in the **fcmdb.properties** and **fcmdb.push.properties** files, which are located in the \conf\log folder of your Data Flow Probe or Integration Service installation. Additionally, you can view push, population and federation log information in the **fcmdb.push.all.log** file, which is located in the \runtime\log folder of your Data Flow Probe or Integration Service installation. Your Data Flow Probe or Integration Service may be installed on the same host as your UCMDB Server or on a separate host.

To set the Development adapter log level and view the population log file, follow the steps described in ["How to Check the Push Log File" on page 261](#).

## Typical Population Error Messages and Solutions

The following describes typical error messages that may occur during population, and their solutions.

This section includes:

- ["No TQL Query Configured in smPopConf.xml " below](#)
- ["Nonexistent Mapping File Name Defined for a TQL Query in smPopConf.xml" on page 271](#)

### No TQL Query Configured in smPopConf.xml

#### Error Message

If you have not yet added a TQL query to your job, you cannot select this query from the list while you create or update your job.

If you have already added this TQL query to your job, but forgot to add the configuration for this query in smPopConf.xml, you will get a "Failed" status while you run this population job. In addition, in the Universal CMDB studio, you will get an error message like the following (see ["How to Check the Error Message of a Failed Population Job" on the previous page](#)).

```
running population. Destination ID: sm, Failed during query: SM Business
Application Population 2.0, all queries: [SM Business Application Population
2.0, SM Business Service Population 2.0], finished queries: [].
ERROR: com.mercury.topaz.cmdb.shared.base.CmdbException: [ErrorCode [802]
General Integration Error{sm}]
appilog.framework.shared.manage.impl.MamResponseException: [ErrorCode [802]
General Integration Error{sm}]
CMDB Operation Internal Error: class
com.mercury.topaz.cmdb.shared.fcmdb.dataAccess.exception.AdapterAccessGeneralExc
eption : Unsupported Query [SM Business Application Population 2.0], only
```



```

population TQL is supported : operation Data Access Adapter Query: Retrieve
Changed Data
    at
com.mercury.topaz.cmdb.shared.manage.operation.impl.AbstractCommonOperation.execute(AbstractCommonOperation.java:160)
    at
com.hp.ucmdb.dataAccess.manager.DataAccessAdapterManagerProbeImpl.executeOperation(DataAccessAdapterManagerProbeImpl.java:50)
    at
com.hp.ucmdb.discovery.probe.agents.probemgr.adapters.DataAccessAdaptersFacade.invokeOperation(DataAccessAdaptersFacade.java:406)
    at
com.hp.ucmdb.discovery.probe.services.dynamic.core.AdapterService.runChangesOnPopulateChangesAdapter(AdapterService.java:1262)
    at
com.hp.ucmdb.discovery.probe.services.dynamic.core.AdapterService.runQueriesOnPopulateChangesAdapter(AdapterService.java:1102)
    at com.hp.ucmdb.discovery.probe.services.dynamic.core.AdapterService.runQueries(AdapterService.java:354)
    at
com.hp.ucmdb.discovery.probe.services.dynamic.core.AdapterService.runDiscovery(AdapterService.java:198)
    at com.hp.ucmdb.discovery.probe.services.dynamic.core.AdapterService.discover(AdapterService.java:149)
    at
com.hp.ucmdb.discovery.probe.agents.probemgr.taskexecuter.JobExecuter.launchTask(JobExecuter.java:1188)
    at
com.hp.ucmdb.discovery.probe.agents.probemgr.taskexecuter.JobExecuter$JobExecuterWorker.launch(JobExecuter.java:945)

```

**In the fcmdb.push.all.log file, you can find more details like the following:**

```

2014-11-20 10:40:50,949 [JobExecuterWorker-0:DS_sm_SM BS pop] ERROR - sm >> Fail to
Create or Return PopulationConnectorOutput
com.hp.ucmdb.federationspi.exception.DataAccessGeneralException: Unsupported
Query [SM Business Application Population 2.0], only population TQL is supported
    at com.hp.ucmdb.adapter.smpush.ServiceManagerGenericAdapter.populate
(ServiceManagerGenericAdapter.java:337)
    at com.hp.ucmdb.adapters.GenericAdapter.getChanges(GenericAdapter.java:881)
    at
com.hp.ucmdb.dataAccess.operations.DataAccessAdapterQueryRetrieveChanges.getChangesResult(DataAccessAdapterQueryRetrieveChanges.java:50)
    at
com.hp.ucmdb.dataAccess.operations.DataAccessAdapterQueryRetrieveChanges.doDataAccessQueryExecute(DataAccessAdapterQueryRetrieveChanges.java:38)
    at
com.hp.ucmdb.dataAccess.operations.AbstractDataAccessLifeCycleAdapterQuery.doLifeCycleExecute(AbstractDataAccessLifeCycleAdapterQuery.java:34)

```

```
at
com.hp.ucmdb.dataAccess.operations.AbstractDataAccessLifeCycleAdapterOperation.doDataAccessExecute(AbstractDataAccessLifeCycleAdapterOperation.java:57)
at
com.hp.ucmdb.dataAccess.operations.AbstractDataAccessAdapterOperation.dataAccessExecute(AbstractDataAccessAdapterOperation.java:59)
at
com.hp.ucmdb.dataAccess.operations.AbstractDataAccessAdapterOperation.doExecute(AbstractDataAccessAdapterOperation.java:37)
at
com.mercury.topaz.cmdb.shared.manage.operation.impl.AbstractFrameworkOperation.commonExecute(AbstractFrameworkOperation.java:17)
at
com.mercury.topaz.cmdb.shared.manage.operation.impl.AbstractCommonOperation$OperationExecuteFlowTrackingCommand.execute(AbstractCommonOperation.java:87)
at
com.mercury.topaz.cmdb.shared.manage.operation.impl.AbstractCommonOperation$OperationExecuteFlowTrackingCommand.execute(AbstractCommonOperation.java:60)
at com.mercury.topaz.cmdb.shared.manage.flowmanagement.api.FlowManager.execute(FlowManager.java:227)
at
com.mercury.topaz.cmdb.shared.manage.operation.flow.OperationInFlowDefaultExecutor.execute(OperationInFlowDefaultExecutor.java:23)
at
com.mercury.topaz.cmdb.shared.manage.operation.impl.AbstractCommonOperation.execute(AbstractCommonOperation.java:158)
at
com.hp.ucmdb.dataAccess.manager.DataAccessAdapterManagerProbeImpl.executeOperation(DataAccessAdapterManagerProbeImpl.java:50)
at
com.hp.ucmdb.discovery.probe.agents.probemgr.adapters.DataAccessAdaptersFacade.invokeOperation(DataAccessAdaptersFacade.java:406)
at
com.hp.ucmdb.discovery.probe.services.dynamic.core.AdapterService.runChangesOnPopulateChangesAdapter(AdapterService.java:1262)
at
com.hp.ucmdb.discovery.probe.services.dynamic.core.AdapterService.runQueriesOnPopulateChangesAdapter(AdapterService.java:1102)
at com.hp.ucmdb.discovery.probe.services.dynamic.core.AdapterService.runQueries(AdapterService.java:354)
at
com.hp.ucmdb.discovery.probe.services.dynamic.core.AdapterService.runDiscovery(AdapterService.java:198)
at com.hp.ucmdb.discovery.probe.services.dynamic.core.AdapterService.discover(AdapterService.java:149)
at
com.hp.ucmdb.discovery.probe.agents.probemgr.taskexecuter.JobExecuter.launchTask(JobExecuter.java:1188)
at
```

```

com.hp.ucmdb.discovery.probe.agents.probemgr.taskexecuter.JobExecuter$JobExecute
rWorker.launch(JobExecuter.java:945)
    at
com.hp.ucmdb.discovery.probe.agents.probemgr.taskexecuter.JobExecuter$JobExecute
rWorker.executeTask(JobExecuter.java:867)
    at
com.hp.ucmdb.discovery.probe.agents.probemgr.taskexecuter.JobExecuter$JobExecute
rWorker.run(JobExecuter.java:728)

```

## Solution

Search for text string “Unsupported Query by this adapter” to find out the TQL query name that has not yet been configured, and then configure it in the smPopConf.xml file.

## Nonexistent Mapping File Name Defined for a TQL Query in smPopConf.xml

### Error Message

You will get a “Failed” status while you run the population job. In addition, from the Universal CMDB studio, you will get an error message like the following:

```

running population. Destination ID: sm, Failed during query: SM Business Service
Population 2.0, all queries: [SM Business Application Population 2.0, SM
Business Service Population 2.0], finished queries: [SM Business Application
Population 2.0].
ERROR: com.mercury.topaz.cmdb.shared.base.CmdbException: [ErrorCode [802]
General Integration Error{sm}]
appilog.framework.shared.manage.impl.MamResponseException: [ErrorCode [802]
General Integration Error{sm}]
CMDB Operation Internal Error: class
com.mercury.topaz.cmdb.shared.fcldb.dataAccess.exception.AdapterAccessGeneralExc
eption : Query Definition [SM Business Service Population 2.0] has no matching
mapping. Make sure the mapping exist exists under folder 'mapping' and contains
the exact query name and root name. Available mappings:
[QueryRoot{queryName='SM Business Application Population 2.0',
rootName='bizservice'}] : operation Data Access Adapter Query: Retrieve Changed
Data
    at
com.mercury.topaz.cmdb.shared.manage.operation.impl.AbstractCommonOperation.exec
ute(AbstractCommonOperation.java:160)
    at
com.hp.ucmdb.dataAccess.manager.DataAccessAdapterManagerProbeImpl.executeOperati
on(DataAccessAdapterManagerProbeImpl.java:50)
    at
com.hp.ucmdb.discovery.probe.agents.probemgr.adapters.DataAccessAdaptersFacade.i
nvokeOperation(DataAccessAdaptersFacade.java:406)
    at

```

```

com.hp.ucmdb.discovery.probe.services.dynamic.core.AdapterService.runChangesOnPopulateChangesAdapter(AdapterService.java:1262)
    at
com.hp.ucmdb.discovery.probe.services.dynamic.core.AdapterService.runQueriesOnPopulateChangesAdapter(AdapterService.java:1102)
    at com.hp.ucmdb.discovery.probe.services.dynamic.core.AdapterService.runQueries(AdapterService.java:354)
    at
com.hp.ucmdb.discovery.probe.services.dynamic.core.AdapterService.runDiscovery(AdapterService.java:198)
    at com.hp.ucmdb.discovery.probe.services.dynamic.core.AdapterService.discover(AdapterService.java:149)
    at
com.hp.ucmdb.discovery.probe.agents.probemgr.taskexecuter.JobExecuter.launchTask(JobExecuter.java:1188)
    at
com.hp.ucmdb.discovery.probe.agents.probemgr.taskexecuter.JobExecuter$JobExecuterWorker.launch(JobExecuter.java:945)
    at
com.hp.ucmdb.discovery.probe.agents.probemgr.taskexecuter.JobExecuter$JobExecuterWorker.executeTask(JobExecuter.java:867)
    at
com.hp.ucmdb.discovery.probe.agents.probemgr.taskexecuter.JobExecuter$JobExecuterWorker.run(JobExecuter.java:728)

```

## Solution

Search for text “has no matching mapping” to find out the missing mapping file name, and then add the mapping file to the adapter package.

## Troubleshooting Federation Issues

When a federation error occurs, you can check the error message in the federation log file to identify the root cause and then solve the problem.

For more information, see the following:

["How to Check the Error Message of a Failed Federation Request" below](#)

["Typical Federation Error Messages and Solutions" on the next page](#)

### How to Check the Error Message of a Failed Federation Request

When a federation query fails, a pop-up window similar to the following appears in the UCMDb studio.



**Note:** To troubleshoot push, population, and federation issues, you need to set the Development adapter log level to DEBUG or TRACE in the **fcmdb.properties** and **fcmdb.push.properties** files, which are located in the \conf\log folder of your Data Flow Probe or Integration Service installation. Additionally, you can view push, population and federation log information in the **fcmdb.push.all.log** file, which is located in the \runtime\log folder of your Data Flow Probe or Integration Service installation. Your Data Flow Probe or Integration Service may be installed on the same host as your UCMDB Server or on a separate host.

To set the Development adapter log level and view the detailed error message of a failed federation query, follow the steps described in ["How to Check the Push Log File" on page 261](#).

## Typical Federation Error Messages and Solutions

The following sections describe typical errors that may occur during federation and their solutions.

["Wrong Configuration for a Federation CI Type in smFedConf.xml" below](#)

["Mapping File for the Federation TQL Query Is not Well Formed" on page 275](#)

### Wrong Configuration for a Federation CI Type in smFedConf.xml

#### Error Message

If you have not correctly configured a CI type in the smFedConf.xml file, UCMDB cannot federate CIs of this CI type. When you run a related federation query, the UCMDB studio will pop up an error window; additionally, you can find an error message that resembles the following in the federation log file:

```
2014-11-20 13:22:52,055 [AdHoc:AD_HOC_TASK_PATTERN_ID-51-1416460969254] ERROR - sm
  >> Failed to retrieve or parsing Root/parent CI message from SM side, exit from
  federation!
  java.lang.NullPointerException
        at
  com.mercury.topaz.fcmdb.adapters.serviceDeskAdapter.federation.SmGenericFederato
  r.processRootCiQueryByPage(SmGenericFederator.java:461)
        at
```

```
com.mercury.topaz.fcmdb.adapters.serviceDeskAdapter.federation.SmGenericFederato
r.generateRootRtnResult(SmGenericFederator.java:162)
    at
com.mercury.topaz.fcmdb.adapters.serviceDeskAdapter.federation.SmGenericFederato
r.generateFederationRtnResult(SmGenericFederator.java:131)
    at
com.mercury.topaz.fcmdb.adapters.serviceDeskAdapter.federation.SmGenericFederato
r.generateFederationOutput(SmGenericFederator.java:120)
    at
com.mercury.topaz.fcmdb.adapters.serviceDeskAdapter.federation.SmGenericFederato
r.getNextResultChunk(SmGenericFederator.java:578)
    at com.hp.ucmdb.adapter.smpush.ServiceManagerGenericAdapter.populate
(ServiceManagerGenericAdapter.java:363)
    at com.hp.ucmdb.adapters.GenericAdapter.getDataResult(GenericAdapter.java:740)
    at
com.hp.ucmdb.discovery.probe.processor.FederationTopologyGetDataResultProbeReque
stProcessor.processRequest
(FederationTopologyGetDataResultProbeRequestProcessor.java:40)
    at
com.hp.ucmdb.discovery.probe.processor.FederationTopologyGetDataResultProbeReque
stProcessor.processRequest
(FederationTopologyGetDataResultProbeRequestProcessor.java:26)
    at com.hp.ucmdb.discovery.probe.processor.AbstractProbeProcessor.process
(AbstractProbeProcessor.java:56)
    at com.hp.ucmdb.discovery.probe.processor.AbstractProbeProcessor.process
(AbstractProbeProcessor.java:19)
    at
com.hp.ucmdb.discovery.probe.agents.probemgr.adhocktasks.AdHocProbeRequestOperati
on.performAction(AdHocProbeRequestOperation.java:63)
    at
com.hp.ucmdb.discovery.probe.agents.probemgr.taskdispatcher.AdHocTaskDispatcher.
dispatchTask(AdHocTaskDispatcher.java:70)
    at sun.reflect.GeneratedMethodAccessor75.invoke(Unknown Source)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke
(DelegatingMethodAccessorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:601)
    at com.sun.jmx.mbeanserver.StandardMBeanIntrospector.invokeM2
(StandardMBeanIntrospector.java:111)
    at com.sun.jmx.mbeanserver.StandardMBeanIntrospector.invokeM2
(StandardMBeanIntrospector.java:45)
    at com.sun.jmx.mbeanserver.MBeanIntrospector.invokeM
(MBeanIntrospector.java:235)
    at com.sun.jmx.mbeanserver.PerInterface.invoke(PerInterface.java:138)
    at com.sun.jmx.mbeanserver.MBeanSupport.invoke(MBeanSupport.java:252)
    at javax.management.StandardMBean.invoke(StandardMBean.java:405)
    at com.sun.jmx.interceptor.DefaultMBeanServerInterceptor.invoke
(DefaultMBeanServerInterceptor.java:819)
    at com.sun.jmx.mbeanserver.JmxMBeanServer.invoke(JmxMBeanServer.java:792)
    at javax.management.MBeanServerInvocationHandler.invoke
```

```

(MBeanServerInvocationHandler.java:305)
    at org.springframework.jmx.access.MBeanClientInterceptor.doInvoke
(MBeanClientInterceptor.java:405)
    at org.springframework.jmx.access.MBeanClientInterceptor.invoke
(MBeanClientInterceptor.java:353)
    at org.springframework.aop.framework.ReflectiveMethodInvocation.proceed
(ReflectiveMethodInvocation.java:172)
    at org.springframework.aop.framework.JdkDynamicAopProxy.invoke
(JdkDynamicAopProxy.java:202)
    at com.sun.proxy.$Proxy51.dispatchTask(Unknown Source)
    at
com.hp.ucmdb.discovery.probe.agents.probegw.managementtasks.adhocktasks.AdhocThre
ad.run(AdhocThread.java:54)
    at org.eclipse.jetty.util.thread.QueuedThreadPool.runJob
(QueuedThreadPool.java:599)
    at org.eclipse.jetty.util.thread.QueuedThreadPool$3.run
(QueuedThreadPool.java:534)
    at java.lang.Thread.run(Thread.java:722)

```

## Solution

Open the smFedConf.xml file (which is located in the Configuration Files folder of the ServiceManagerEnhancedAdapter9-x adapter), and make sure the configuration for the CI type is correct.

## Mapping File for the Federation TQL Query Is not Well Formed

### Error Message

There are cases where the mapping file for a federation TQL query is not well formed. For example, target\_mapping\_name "priority" in the following example federation mapping file specifies a wrong Groovy function name (SMFederationFunctions.getEnumValue1):

```

<?xml version="1.0" encoding="UTF-8"?>
  <integration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="../mappings_schema.xsd">
    <info>
      <source name="SM"          version="9.40"   vendor="HP"/>
      <target name="UCMDB"      version="10.20"  vendor="HP"/>
    </info>
    <import>
      <scriptFile path="mappings_scripts.SMFederationFunctions"/>
    </import>
    <target_entities>
      <source_instance query-name="SM Incident 2.0" root-element-
name="ucmdbIncident">

```

```

        <target_entity name="incident">
            <target_mapping name="reference_number"      datatype="STRING"
value="ucmdbIncident['IncidentID']"/>
            <target_mapping name="name"                  datatype="STRING"
value="ucmdbIncident['BriefDescription']"/>
            <target_mapping name="priority"              datatype="STRING"
value="SMFederationFunctions.getEnumValue1(ucmdbIncident
['PriorityCode'],'Priority')"/>
            <target_mapping name="incident_status"      datatype="STRING"
value="SMFederationFunctions.firstLetterToLowerAndReplaceSpaceWithUnderscore
(ucmdbIncident['IMTicketStatus'])/>
            <target_mapping name="category"              datatype="STRING"
value="SMFederationFunctions.replaceSpaceWithUnderscore(ucmdbIncident
['Category'])/>
            <target_mapping name="closed_time"           datatype="DATE"
value="SMFederationFunctions.convertDate(ucmdbIncident['ClosedTime'])/>
            <target_mapping name="create_time"           datatype="DATE"
value="SMFederationFunctions.convertDate(ucmdbIncident['OpenTime'])/>
            <target_mapping name="last_modified_time"    datatype="DATE"
value="SMFederationFunctions.convertDate(ucmdbIncident['UpdatedTime'])/>
            <target_mapping name="impact_scope"          datatype="STRING"
value="SMFederationFunctions.getEnumValue(ucmdbIncident
['ImpactScope'],'ImpactScope')"/>
            <target_mapping name="urgency"              datatype="STRING"
value="SMFederationFunctions.getEnumValue(ucmdbIncident
['Urgency'],'Urgency')"/>
        </target_entity>
    </source_instance>
</target_entities>
</integration>

```

When you try to federate incidents, no records are retrieved from Service Manager. If you check the `fcmdb.push.all.log` file, you can find a detailed error message that resembles the following for the federation query:

```

2014-11-20 14:09:58,670 [AdHoc:AD_HOC_TASK_PATTERN_ID-66-1416463796366] ERROR - >>
Failed executing value [SMFederationFunctions.getEnumValue1(ucmdbIncident
['PriorityCode'],'Priority')] of mapping <target_mapping name="priority">
<target_ci_type name="incident"> , Root cmdbId [null]
groovy.lang.MissingMethodException: No signature of method: static
mappings.scripts.SMFederationFunctions.getEnumValue1() is applicable for
argument types: (java.lang.String, java.lang.String) values: [1, Priority]
Possible solutions: getEnumValue(java.lang.String, java.lang.String)
    at groovy.lang.MetaClassImpl.invokeStaticMissingMethod(MetaClassImpl.java:1359)
    at groovy.lang.MetaClassImpl.invokeStaticMethod(MetaClassImpl.java:1345)
    at org.codehaus.groovy.runtime.callsite.StaticMetaClassSite.call
(StaticMetaClassSite.java:50)
    at org.codehaus.groovy.runtime.callsite.CallSiteArray.defaultCall
(CallSiteArray.java:42)

```



```
    at org.codehaus.groovy.runtime.callsite.AbstractCallSite.call
(AbstractCallSite.java:108)
    at org.codehaus.groovy.runtime.callsite.AbstractCallSite.call
(AbstractCallSite.java:120)
    at Mapping_4a8761e7adc009e8a7d268c2f1349ab4.run(Mapping_
4a8761e7adc009e8a7d268c2f1349ab4.groovy:1)
    at
com.hp.ucmdb.adapters.instance.mapping.AbstractResultTreeNodeMapper.calculatePro
perties(AbstractResultTreeNodeMapper.java:231)
    at
com.hp.ucmdb.adapters.instance.mapping.AbstractResultTreeNodeMapper.processTarge
tEntity(AbstractResultTreeNodeMapper.java:301)
    at com.hp.ucmdb.adapters.instance.mapping.RtnToRtnMapper.processTargetEntities
(RtnToRtnMapper.java:214)
    at
com.hp.ucmdb.adapters.instance.mapping.RtnToRtnMapper.processSourceInstanceWrapp
er(RtnToRtnMapper.java:101)
    at com.hp.ucmdb.adapters.instance.mapping.RtnToRtnMapper.buildResultTreeNode
(RtnToRtnMapper.java:76)
    at
com.hp.ucmdb.adapters.GenericAdapter$PopulationFederationChunkProcessor.invoke
(GenericAdapter.java:1213)
    at com.hp.ucmdb.adapters.GenericAdapter.getDataResult(GenericAdapter.java:743)
    at
com.hp.ucmdb.discovery.probe.processor.FederationTopologyGetDataResultProbeReque
stProcessor.processRequest
(FederationTopologyGetDataResultProbeRequestProcessor.java:40)
    at
com.hp.ucmdb.discovery.probe.processor.FederationTopologyGetDataResultProbeReque
stProcessor.processRequest
(FederationTopologyGetDataResultProbeRequestProcessor.java:26)
    at com.hp.ucmdb.discovery.probe.processor.AbstractProbeProcessor.process
(AbstractProbeProcessor.java:56)
    at com.hp.ucmdb.discovery.probe.processor.AbstractProbeProcessor.process
(AbstractProbeProcessor.java:19)
    at
com.hp.ucmdb.discovery.probe.agents.probemgr.adhoctasks.AdHocProbeRequestOperati
on.performAction(AdHocProbeRequestOperation.java:63)
    at
com.hp.ucmdb.discovery.probe.agents.probemgr.taskdispatcher.AdHocTaskDispatcher.
dispatchTask(AdHocTaskDispatcher.java:70)
    at sun.reflect.GeneratedMethodAccessor75.invoke(Unknown Source)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke
(DelegatingMethodAccessorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:601)
    at com.sun.jmx.mbeanserver.StandardMBeanIntrospector.invokeM2
(StandardMBeanIntrospector.java:111)
    at com.sun.jmx.mbeanserver.StandardMBeanIntrospector.invokeM2
(StandardMBeanIntrospector.java:45)
```

```

    at com.sun.jmx.mbeanserver.MBeanIntrospector.invokeM
(MBeanIntrospector.java:235)
    at com.sun.jmx.mbeanserver.PerInterface.invoke(PerInterface.java:138)
    at com.sun.jmx.mbeanserver.MBeanSupport.invoke(MBeanSupport.java:252)
    at javax.management.StandardMBean.invoke(StandardMBean.java:405)
    at com.sun.jmx.interceptor.DefaultMBeanServerInterceptor.invoke
(DefaultMBeanServerInterceptor.java:819)
    at com.sun.jmx.mbeanserver.JmxMBeanServer.invoke(JmxMBeanServer.java:792)
    at javax.management.MBeanServerInvocationHandler.invoke
(MBeanServerInvocationHandler.java:305)
    at org.springframework.jmx.access.MBeanClientInterceptor.doInvoke
(MBeanClientInterceptor.java:405)
    at org.springframework.jmx.access.MBeanClientInterceptor.invoke
(MBeanClientInterceptor.java:353)
    at org.springframework.aop.framework.ReflectiveMethodInvocation.proceed
(ReflectiveMethodInvocation.java:172)
    at org.springframework.aop.framework.JdkDynamicAopProxy.invoke
(JdkDynamicAopProxy.java:202)
    at com.sun.proxy.$Proxy51.dispatchTask(Unknown Source)
    at
com.hp.ucmdb.discovery.probe.agents.probegw.managementtasks.adhocktasks.AdhocThre
ad.run(AdhocThread.java:54)
    at org.eclipse.jetty.util.thread.QueuedThreadPool.runJob
(QueuedThreadPool.java:599)
    at org.eclipse.jetty.util.thread.QueuedThreadPool$3.run
(QueuedThreadPool.java:534)
    at java.lang.Thread.run(Thread.java:722)

```

## Solution

Search for word "Exception" in the fcmdb.push.all.log file to find the wrong method name, and then correct it in the corresponding federation mapping file.

## Using Service Manager and UCMDB

The connection between Service Manager and Universal Configuration Management Database is performed by Deployment Manager during installation of Universal Configuration Management Database.

**Note:** The following sections are excerpted from the integrations section of the Service Manager Help Center. However, to ensure you have the latest information, you should reference the latest published version of the document in question.

## Enable an integration to HP Universal CMDB

**Applies to User Roles:**

System Administrator

You must have the **SysAdmin** capability word to use this procedure.

You can configure Service Manager to display the actual state of CIs in the HP Universal CMDB server by defining an active integration in the System Information Record. After you define this active integration, Service Manager displays the Actual State tab in Configuration Management forms, and displays the **View in UCMDB** or **View in UCMDB Browser** button in CI records synchronized from UCMDB.

**Note:** The UCMDB Browser is an optional add-on to UCMDB. For more information, see "[HP Universal CMDB Browser](#)" on page 285.

To enable an integration to UCMDB in Service Manager:

1. Click **System Administration > Base System Configuration > Miscellaneous > System Information Record**.
2. Click the **Active Integrations** tab.
3. Select the **HP Universal CMDB** option.
4. Complete the following fields.

Field	Description
UCMDB webservice URL	<p>This is the URL to the HP Universal CMDB web service API. The URL has the following format:</p> <p><code>http://&lt;UCMDB server name&gt;:&lt;port&gt;/axis2/services/ucmdbSMService</code></p> <p>Replace <i>&lt;UCMDB server name&gt;</i> with the host name of your UCMDB server, and replace <i>&lt;port&gt;</i> with the communications port your UCMDB server uses.</p>
UserId Password	<p>This is the UCMDB account (user name/password) used to synchronize CI information with Service Manager. For example, admin/admin.</p>

Field	Description
Multi-tenant UCMDB webservice URL	<p>This is the URL that Service Manager uses to synchronize company records with UCMDB when running in multi-company mode. This URL should use the following format:</p> <p>http://&lt;ucmdb server name&gt;:&lt;port&gt;/axis2/services/UcmdbManagementService.</p>
UserId Password	<p>This is the UCMDB account (user name/password) that has the privilege to add/delete company records.</p>
UCMDB Browser URL	<p>This is the UCMDB Browser URL. It is in the following format:</p> <p>http://&lt;UCMDB browser server name&gt;:&lt;port&gt;/ucmdb-browser</p> <p>For example: <b>http://myucmdbbrowserserver:8081/ucmdb-browser</b></p> <p>The UCMDB Browser has two themes. By default, it uses the dark color theme; if you want to use the light color theme, use this format for the UCMDB Browser URL:</p> <p>http://&lt;UCMDB browser server name&gt;:&lt;port&gt;/ucmdb-browser/?theme=LIGHT</p> <div style="background-color: #f0f0f0; padding: 10px; margin-top: 10px;"> <p><b>Note:</b> The <b>UCMDB Browser URL</b> field is required to enable an integration to the UCMDB Browser. If you specify a value for this field, the <b>View in UCMDB Browser</b> button will replace the <b>View in UCMDB</b> button in CI records synchronized from UCMDB; only when you leave this field empty, the <b>View in UCMDB</b> button will display.</p> </div>

5. Click **Save**.

Service Manager displays the message:

Information record updated.

**Note:** To enable the integration, you still need to set up an integration point in UCMDB. For details, see the *HP Universal CMDB Integration Guide*.

## Configuration item actual states

By default, HP Service Manager only stores and displays the expected or managed state of CIs. The information Service Manager displays in the Configuration Management form is essentially the definitive list of attributes that the CI should have. However, the actual state of the CI may differ from the expected state.

To view the actual state of the CI, you must first create an integration to an HP Universal CMDB server. The HP Universal CMDB server periodically discovers the actual state of CIs and records the actual state in the Configuration Management database. Service Manager accesses the actual state information by using a Web services connection. Service Manager sends the CI ID to the HP Universal CMDB server and receives a full list of the attributes for that CI. Service Manager displays the CI attributes in the Actual State section of the Configuration Management form.

If a Service Manager CI does not have a matching CI in the HP Universal CMDB server, then Service Manager does not display the Actual State section. For example, you may track office furnishing CIs in Service Manager that cannot be discovered and tracked in the HP Universal CMDB.

## View the actual state of a configuration item

### **Applies to User Roles:**

Configuration Manager

Configuration Administrator

Configuration Auditor

This procedure requires that your HP Service Manager system has an active integration to an HP Universal CMDB server.

To view the actual state of a configuration item, follow these steps:

1. Click **Configuration Management > Search CI**.
2. Use search or advanced search to find one or more records.
3. Select the CI with the actual state you want to see.
4. From the CI detail form, open the **Actual State** section. Service Manager displays the actual state of the CI.

**Note:** The **Actual State** section is only visible if your system has an active integration to an HP Universal CMDB server and the CI has a matching entry in the Configuration Management database. Service Manager does not display the **Actual State** section if there is no matching CI in the HP Universal CMDB server.

## Reconciling configuration items between HP Service Manager and HP Universal CMDB

Since both HP Service Manager and HP Universal CMDB contain information about configuration items (CIs), it is possible that your Service Manager system contains CIs that match CIs in the HP Universal CMDB system. Rather than add duplicate CI records, you can configure Service Manager to reconcile CI records between the two systems based on the values of specific fields.

Service Manager always attempts to reconcile CI records based on the unique key field of the Service Manager table and the ucldb.id field. You can specify additional fields to reconcile on from the DEM Reconciliation Rules form. If Service Manager finds a matching value in any one of these fields, it updates the Service Manager CI record with the attributes from the incoming HP Universal CMDB record.

In order to specify reconciliation fields, you will need to be familiar with the table and field names in both your Service Manager and HP Universal CMDB systems. If you want to reconcile on an attribute in the HP Universal CMDB system, you must ensure that there is a corresponding Web Services Definition for the attribute in the Service Manager system. Without such a mapping, Service Manager will not know to search for matching values in a Service Manager table and field name.

For example, suppose you want to reconcile CIs on the asset.tag field in your Service Manager system. You could update the ucldbIntegration web service definition with the following field information.

Service Name	Object Name	Field	Caption
ucldbIntegration	ucldbComputer	asset.tag	AssetTag

You would then need to add an attribute and mapping for the asset.tag field in the HP Universal CMDB system. You would also probably want to add the new attribute to the regular discovery schedule so that the HP Universal CMDB system picks up any updates to this attribute.

After you create a web services definition for the HP Universal CMDB attribute, you must then define a Discovery Event Manager (DEM) reconciliation rule. The reconciliation rule specifies what Service Manager table and field you want to search for matching CI values. In addition, you can specify the sequence you want Service Manager to process reconciliation rules. By default, Service Manager

processes rules in alphabetical order by field name. Thus, Service Manager will reconcile CIs against the asset.tag field before reconciling CIs on the ci.name field.

To change the order in which Service Manager reconciles CIs, you can add a numeric value to the sequence field. For example, the following reconciliation rules ensure that Service Manager processes CIs by the ci.name field prior to reconciling them against the asset.tag field.

Table Name	Field Name	Sequence
device	ci.name	1
device	asset.tag	2

## Create a DEM reconciliation rule

### Applies to User Roles:

System Administrator

A Discovery Event Manager (DEM) reconciliation rule allows you to specify what Service Manager fields you want to use to determine if an existing CI record matches a CI in a HP Universal CMDB system. An administrator typically specifies reconciliation rules prior to starting the integration to the HP Universal CMDB system so that Service Manager will not create duplicate CI records.

To create a DEM reconciliation rule:

1. Click **Tailoring > Web Services > DEM Reconciliation Rules**.

Service Manager displays the DEM Reconcile Record form.

2. In **Table Name**, type the name of the Service Manager table containing the field you want to reconcile on.
3. In **Field Name**, type the name of the Service Manager field containing the values you want to reconcile on.
4. In **Sequence**, type a number to specify what order you want Service Manager to run this rule.

**Note:** If you do not specify a sequence value, Service Manager will process field names alphabetically.

5. Click **New**.

Service Manager creates the reconciliation rule.

## Multi-tenant (multi-company) support

The HP Universal CMDB (UCMDB) to HP Service Manager Integration supports a multitenancy configuration in which both the Service Manager and UCMDB systems track Configuration Items (CIs) and Configuration Item Relationships (CIRs) by company ID. In a multi-tenancy configuration, you can tailor the integration so that each tenant only sees and works with the CIs and CIRs that match their company ID. Multi-tenancy is intended for managed service providers (MSPs) who wish to offer Configuration Management as a service to multiple tenants.

### What multi-tenant information is stored in UCMDB?

Your UCMDB system stores a company ID attribute for each CI and CIR. The company ID determines what adapter and synchronization schedule your UCMDB system uses to update CI data. Each CI and relationship record can only have one company ID. The UCMDB system obtains a company ID from the Service Manager system.

If more than one tenant (company) shares the same CI, each tenant has their own unique CI record describing the CI. In effect, the UCMDB system creates multiple CI records to track one managed asset. Each tenant's CI record is unique to that tenant and lists the company's unique company ID.

### What multi-tenant information is stored in Service Manager?

Your Service Manager stores the company records that describe each tenant in the multi-tenant configuration. The Service Manager system is the definitive source of company IDs and pushes new and updated information to your UCMDB system.

Service Manager tracks the company ID of each CI and relationship in a multi-tenant configuration. CI records inherit the company ID of the UCMDB feeder that discovered them. Relationship records inherit the company ID of the parent CI in the relationship.

In a best practices implementation, Service Manager uses Mandanten to ensure that operators only see CI and relationship records where the CI's company ID matches the operator's company ID. If you restrict the view with Mandanten, then Service Manager also restricts the view to all other related records such as change requests and incidents.

For more information, refer to the *HP Universal CMDB to HP Service Manager Integration Guide* available from the HP Software Product Manuals site.



## HP Universal CMDB Configuration Manager

HP Universal CMDB Configuration Manager (Configuration Manager) provides the tools to help the system manager better control the CMS (Configuration Management System) data. It focuses primarily on analyzing and controlling the data in the CMS, as the ITIL directs. Configuration Manager provides an environment for controlling the CMS infrastructure, which encompasses many data sources and serves a variety of products and applications.

The Configuration Manager to Service Manager (CM-SM) integration provides a policy-based change control solution that can handle configuration changes effectively. This solution is different from the UCMDB to Service Manager (UCMDB-SM) integration solution, which provides attribute modeling change control. The UCMDB-SM integration solution provides the most control but requires manual attribute data entry and can generate a large amount of unplanned changes to be reviewed and accepted or backed out. For customers that do not need change planning at this level of detail, the CM-SM integration solution is recommended.

Using Configuration Manager, the UCMDB authorized state is pushed to Service Manager, and Service Manager benefits from working with the same data that resides in the CMS and can be easily consumed by other parties. The data provided to Service Manager is already controlled and of high quality, and Service Manager therefore no longer holds a distinct state of its CIs. As the state of a configuration item is fully managed by the CMS (including actual, authorized, and historical states), the configuration controls implemented in the UCMDB-SM integration in which Service Manager must analyze which CI changes require an RFC are not used. Service Manager now consumes both actual and authorized CI states from UCMDB.

For more information about the CM-SM integration solution, refer to the following Universal CMDB Configuration Manager documents:

- *HP Universal CMDB Configuration Manager User Guide*, which is available from the HP Software Manuals Site
- *Technical white paper: Handle configuration changes effectively*  
(<http://h20195.www2.hp.com/V2/GetPDF.aspx/4AA0-7486ENW.pdf>)

## HP Universal CMDB Browser

UCMDB Browser is a lightweight web-based client to access UCMDB data. UCMDB Browser provides a simple and intuitive search for Configuration Items (CIs) in UCMDB and displays important data in the context of the selected CI. It is an ideal tool for providing quick access to specific CI information. For

each CI, relevant data is presented and gathered into information widgets (for example, Properties, Environment, and Impact widgets). Data is presented by default in a Preview mode, with the option to view more comprehensive data in an Expanded mode.

You enable an integration to UCMDB Browser when enabling an integration to UCMDB. For details, see ["Enable an integration to HP Universal CMDB" on page 279](#).

When integrated with UCMDB Browser, Service Manager provides the following two features:

- A **View in UCMDB Browser** button is displayed in each CI record that is synchronized from UCMDB. When a user clicks this button in the CI, a UCMDB Browser login screen is displayed; after the user logs in with a UCMDB Browser account (username/password), the UCMDB Browser opens in the context of this CI.
- A **Primary CI History in UCMDB** section is displayed in a problem record whose primary CI is synchronized from UCMDB. Users can view the CI changes on that primary CI for root cause investigation.

**Note:** For accessibility support of the embedded UCMDB widget, refer to the UCMDB Browser documentation.

## Discovery Event Manager

The Discovery Event Manager tool provides you with information about configuration items (CIs) in use by your organization. Discovery Event Manager collects data from associated Web services, such as the HP Universal Configuration Management Database (UCMDB) for enterprise IT organizations. UCMDB captures, documents, and stores information about CIs, service dependencies, and relationships that support business services. The Discovery Event Manager tool takes the information captured by UCMDB and compares the actual state of each incoming CI record (both existing and new) to the managed field state of the CI record in HP Service Manager.

If the actual state of an incoming UCMDB CI record differs from the managed field state of the CI record in Service Manager, the Discovery Event Manager tool works with Configuration Management and Change Management to perform any required changes to the incoming CI record according to the rules you have set in the Discovery Event Manager tool.

## Discovery Event Manager managed fields

Managed fields are key fields in the HP Service Manager configuration item (CI) record types that the Discovery Event Manager tool uses to validate the incoming CI records from Web services. By default,

the fields of the incoming CI records should match the key fields in the Service Manager CI records.

If Discovery Event Manager discovers any discrepancies between the actual state of the incoming CI record fields and the Service Manager managed fields, these discrepancies are handled by Change Management by default. Rules define what to do with a CI record to make it compliant with the CI record types in Service Manager.

## Add a managed field in Discovery Event Manager

### **Applies to User Roles:**

System Administrator

Managed fields are key fields in HP Service Manager configuration item (CI) record types. They are important to helping the Discovery Event Manager tool to discover differences between the data in the incoming CI records from Web services versus what Service Manager is expecting to receive in those records. If the incoming records do not match the managed key fields in Service Manager, the rules set in the Discovery Event Manager tool determine what will happen to those incoming CI records.

For example: If the `joinbizservice` table has three fields in the Managed Fields tab in Discovery Event Manager and some of the expected information is different or missing from any of those fields, the Discovery Event Manager tool will deal with that incoming CI record that is not compliant with the Service Manager managed fields for that CI record type.

If you determine that a necessary field is not included in the existing discovery process, you can add a managed field to the applicable table using the settings on the **Managed Fields** tab.

**Note:** If a managed field does not exist in an incoming CI record, the Discovery Event Manager tool is not able to find discrepancies based on that field.

**Tips:** The following steps describe how to manually add a managed field. If you want to automatically add multiple or all fields from a table as managed fields, you can click the **Load Fields** button on the **Managed Fields** tab.

To add a managed field:

1. Click **Tailoring > Web Services > Discovered Event Manager Rules**. The Discovery Event Manager rules form is displayed.
2. Click **Search** to retrieve a list of CI ID types.
3. Select the ID type for the fields you want to view, and then select the **Managed Fields** tab. You can see which fields are being managed for this CI type.

4. To select a new field you want managed, do the following:
  - a. Select the drop-down arrow in the next blank **Field Name** field. You will see the list of fields related to this CI type.
  - b. Select the field you want to add to the **Managed Fields** tab for this CI type.
  - c. For a field that is part of an array of structures, choose a **Structure** and specify the field's **Index** value. For example, the CI type computer contains array elements, such as ports, printers, and scanners. Choose an array element within the structure, and then choose the corresponding index number.
  - d. Click **Save**.
5. Repeat this process to add a managed field for another ID type.
6. Click **OK**.

## View, modify, or delete a managed field in Discovery Event Manager

### Applies to User Roles:

System Administrator

Managed fields are important to ensuring that the Discovery Event Manager tool can process incoming configuration item (CI) data from Web services. When the inventory of your organization changes, you can add or update the CI types based on your new inventory requirements. You can also review the managed fields for existing CI types to determine if the necessary fields exist and add, modify, or delete managed fields.

**Warning:** If you are deleting a field, make sure you select the field you want to delete. If you delete the wrong field, you must add the field back into the Managed Fields tab.

To view, modify, or delete a managed field:

1. Click **Tailoring > Web Services > Discovered Event Manager Rules**. The Discovery Event Manager form is displayed.
2. Click **Search** to retrieve a list of CI ID types.
3. Select the ID type for the fields you want to view, and then select the **Managed Fields** tab. A list of managed fields for the selected CI type is displayed.

4. Select the field you want to edit or delete.
  - Make any necessary changes.
  - Click **Delete** to delete a field that is no longer valid.
  - Click **Save**.
5. Click **OK**.

## Discovery Event Manager rules

Discovery Event Manager processing is governed by rules that define what actions Discovery Event Manager will perform when the actual state of an incoming configuration item (CI) record differs from the managed state of a CI record in HP Service Manager.

During initial configuration of Discovery Event Manager, you start with a basic set of rules for Change Management. You can refine those rules as necessary. Each rule identifies a series of checks that Discovery Event Manager processes whenever a CI record is received from Web services. Depending on the rule, Discovery Event Manager may add or delete the record, open a change, log the information, or update the record's status.

Service Manager opens a change or incident based on the settings defined in the `populateChange` or `populateIncident` function in the `discoveryEvent` ScriptLibrary record. You can override the default settings by writing a custom JavaScript on the **Change Customization** or **Incident Customization** tab.

## Discovery Event Manager rule options

Rules help to automate the process of managing incoming configuration item (CI) records. When you have rules set up, you can manage whether a record is added, updated, or deleted to the CI records in HP Service Manager. The Discovery Event Manager tool checks the incoming CI records and determines what to do when the actual state of the CI records does not match the managed state of the CI records in HP Service Manager. When Discovery Event Manager finds the rule that applies to an incoming CI record type, the server checks the rule, and then updates the record according to the rules you have set up.

**Example:** If a user's machine has 4 GB of RAM added and the Discovery Event Manager tool discovers that the actual state of the CI record does not match the HP Service Manager managed state for that CI record type, the Discovery Event Manager tool opens an unplanned change. This then gives the Change Manager the opportunity to review the CI record and determine what tasks to complete.

Actions are required in the following cases:

- Records that do not exist.
- Records that contain unexpected data.
- Records are marked for deletion.

The following options are available when configuring rules to meet your business needs:

- **Action if matching record does not exist:** If a CI record does not exist in Service Manager.
  - **Add the record:** (Default) When the information received from Web services through Discovery Event Manager does not bring up a matching record in Service Manager, add the record.
  - **Add the record, and set dependency as true:** This option is available only for synchronization of CI relationship data. Service Manager will add the CI relationship record and enable outage dependency for the record by checking its **Outage Dependency** check box and setting its number of dependent downstream CIs to 1.
  - **Open an Incident:** Open an incident to investigate a new CI record that currently does not exist in Service Manager to determine if it is compliant with Service Manager.
  - **Open a Change:** Open an unplanned change to review the new CI record, because it currently does not exist in Service Manager. This change gives the Service Manager Administrator an opportunity to deny the Change Management request and back it out by using the change management process or to accept the actual state of the new CI record and assign tasks accordingly.
- **Action if record exists but unexpected data discovered:** Changes to the information in an existing CI record raise a flag for the Discovery Event Manager tool. Some of the information is not part of the managed state in Service Manager. The unexpected data in the CI record must be logged or reviewed.
  - **Open a Change:** (Default) Open an unplanned change to review the actual state of the CI record. This change gives the Service Manager Administrator an opportunity to deny the Change Management request and back it out by using the change management process, or to accept the actual state of the existing CI record and assign tasks accordingly.
  - **Log Results and update record:** Log the results of the actual state of the CI record, and then update the record.
  - **Open an Incident:** Open an Incident to investigate the actual state of a CI record and determine what actions must be performed or initiated to bring the record into compliance with Service Manager.

- **Action if record is to be deleted:** If an external event specifies that the record needs to be deleted.
  - **Delete record:** (Default for CI relationship records) This option is available for synchronization of both CI and CI relationship records. Service Manager automatically deletes the record.
  - **Open an Incident:** This option is available only for synchronization of CI relationship records. Service Manager opens an incident to investigate the deleted record and determines which actions must be performed or initiated to bring the record into compliance with Service Manager.
  - **Open a Change:** This option is available only for synchronization of CI relationship records. Service Manager opens an unplanned change to review the deleted record. The change allows someone to investigate whether the deleted record is compliant with your business practices. If the record is compliant, the change can be approved. If the record is not compliant, then the change can be denied and the record added back to the system.
  - **Update record to the selected status:** (Default) This option is available only for synchronization of CI records. Service Manager updates the status of the CI record to a value selected from the drop-down list (for example, **Retired/Consumed**), instead of deleting the record permanently.
  - **Open an Incident to update record to the selected status:** This option is available only for synchronization of CI records. Service Manager opens an incident to update the record's status to a value selected from the drop-down list (for example, **Retired/Consumed**). Once the incident has been closed, Service Manager automatically updates the CI record to the selected status.
  - **Open a Change to update record to the selected status:** This option is available only for synchronization of CI records. Service Manager opens an unplanned change to update the CI record's status to a value selected from the drop-down list (for example, **Retired/Consumed**). The change allows someone to investigate whether the requested status change is compliant with your business practices. Once the change has been approved and closed, Service Manager automatically changes the CI record to the selected status. If the change has been denied, Service Manager makes no changes to the CI record.

## Add a rule in Discovery Event Manager

### Applies to User Roles:

System Administrator

Rules are the core of Discovery Event Manager processing. Based on your organization's requirements, you can refine the basic set of rules that are configured with the Discovery Event Manager tool by adding rules.

To add a rule:

1. Click **Tailoring** > **Web Services** > **Discovered Event Manager Rules**. The Discovery Event Manager form opens.
2. Click **New**. The new rule form opens.
3. Enter the new rule name.
4. Select a table to be associated with the rule from the **Table Name** list, and then click **Next**.
5. Enter the condition for the rule. The rule is added to the records table.
6. Click **Save**.
7. Click **OK**.

## View or modify rules in Discovery Event Manager

### **Applies to User Roles:**

System Administrator

Rules help you to automate the change control process, so that incoming configuration item (CI) records can be updated to comply with the CI record fields in HP Service Manager. As you reevaluate your organization's requirements, you may view the rules that are set up and make changes as you see fit.

To view or modify existing rules:

1. Click **Tailoring** > **Web Services** > **Discovered Event Manager Rules**. The Discovery Event Manager form opens.
2. Click **Search** to retrieve a list of CI ID types.
3. Select the ID type for the rules you want to view.
4. Select the **Rules** tab. The existing rules settings for the selected CI type are displayed.
5. If you want to edit the rule, do the following:
  - Make the necessary changes. For example, if you choose to select a different action step for records that do not exist, make your change.
  - Click **Save**.



6. Click the **Duplication Rule** tab. This tab specifies whether Service Manager renames the incoming CI record or returns an error if the record has a duplicated logical name. By default, Service Manager renames the record by appending a suffix to the original CI name.
7. Click **OK**.

## Delete a set of rules in Discovery Event Manager

### Applies to User Roles:

System Administrator

Rules help to automate the change control process, so that incoming configuration item (CI) records can be updated to comply with the CI record fields in HP Service Manager. As you reevaluate your organization's requirements, you may realize that the existing rules settings for a CI type are no longer valid. You can delete the existing rules settings to replace them with a new set of rules.

**Warning:** Make sure you are deleting the rules for the CI ID type you want deleted. If you delete the wrong set of rules by mistake, you will have to add the CI ID type and set up the rules for each action that needs to be taken.

To delete a set of rules:

1. Click **Tailoring > Web Services > Discovered Event Manager Rules**. The Discovery Event Manager form opens.
2. Click **Search** to retrieve a list of CI ID types.
3. Select the CI ID type, and then select the **Rules** tab. Existing rules settings for the selected CI type are displayed.
4. After you determine that you want to delete the rules for this CI type, click **Delete**.
5. Click **OK**.

## Add a configuration item in Discovery Event Manager

### Applies to User Roles:

System Administrator

As inventory changes within your organization, you will need to add new configuration item (CI) types, or possibly update or delete others, to keep your inventory records up-to-date for the Discovery Event Manager tool. You can manage the CI types in the CI record type table.

To add a configuration item in Discovery Event Manager:

1. Click **Tailoring > Web Services > Discovered Event Manager Rules**. The Discovery Event Manager form opens.
2. Click **New**.
3. Enter the name of the new CI record type.
4. Select a table from the table list, and then click **Next**.
5. Enter the Condition for the CI record type. The CI record type is added to the records table.

- The condition must ensure that only one rule is applied when the web service request is processed.
- An empty condition evaluates to true by default.

6. Click **Save**.
7. Click **OK**.

## View, modify, or delete a configuration item in Discovery Event Manager

### Applies to User Roles:

System Administrator

As inventory changes within your organization, you will need to add, update, or delete configuration item (CI) types to keep your inventory records up-to-date for the Discovery Event Manager tool. You can manage the CI types in the CI record type table.

To view, modify, or delete a configuration item in Discovery Event Manager:

**Warning:** If you are deleting a CI record type, make sure you select the CI record type you want to delete. If you delete the wrong record type, you must add the record back into the CI Record ID table.

1. Click **Tailoring > Web Services > Discovered Event Manager Rules**. The Discovery Event Manager form opens.
2. Click **Search** to retrieve a list of CI ID types.

3. Select the ID type you want to view, and then click **Previous** or **Next** to scroll through the list of records.
4. Select the **Rules** tab to view the rules that are set for the selected CI type.
5. Select the record you want to change and make any necessary changes.
6. Select the ID type you want to delete, and then click **Delete**.
7. Click **Save**.
8. Click **OK**.

## Customize changes in Discovery Event Manager

### **Applies to User Roles:**

System Administrator

If your best practice is to have an incoming configuration item (CI) record that is not compliant with the managed state of that CI record in HP Service Manager go through the change management process, you can customize how you want the Discovery Event Manager tool to process those incoming CI records.

To customize the way changes are handled within Discovery Event Manager:

1. Click **Tailoring > Web Services > Discovered Event Manager Rules**. The Discovery Event Manager form opens.
2. Click **Search** to retrieve a list of Configuration Item (CI) ID types.
3. Select the ID type for the customization you want to make, and then select the **Change Customization** tab.
4. Enter the customized script you want to override the default values set by the `discoveryEventScriptLibrary` record (the change record may be referenced as "change").
5. Click **Save**.
6. Click **OK**.

## Customize incidents in Discovery Event Manager

### **Applies to User Roles:**

## System Administrator

If your best practice is to have an incident logged against an incoming configuration item (CI) record that is not compliant with the managed state of a CI record in HP Service Manager, you can customize the rules you set for automatically opening an incident record.

To customize the JavaScript for opening an incident record:

1. Click **Tailoring > Web Services > Discovered Event Manager Rules**. The Discovery Event Manager form opens.
2. Click **Search** to retrieve a list of Configuration Item (CI) ID types.
3. Select the ID type for the customization you want to make to incidents, and then select the **Incident Customization** tab.
4. Enter the customized script you have created to override the default values set by the discoveryEventScriptLibrary record (the incident record may be referenced as "incident").
5. Click **Save**.
6. Click **OK**.

## Integrate uCMDB and Asset Manager

**Note:** The following sections are excerpted from the integrations section of the Universal Configuration Management Database Help Center. However, to ensure you have the latest information, you should reference the latest published version of the document in question.

There are three primary integrations that you may want to consider between Universal Configuration Management Database and Asset Manager:

## HP Asset Manager Integration with the AM Generic Adapter

This chapter includes:

This is a mini TOC level 2

## Overview

Integration between HP Universal CMDB (UCMDB) and HP Asset Manager enables you to share information between UCMDB and Asset Manager. Common use cases include pulling asset from Asset Manager and pushing inventory CIs like hardware, installed software, and business services from UCMDB to Asset Manager.

You can use the integration to automate the creation and update of Asset and Portfolio information in Asset Manager through data push. This ensures that Asset Manager is kept up-to-date with real, accurate, and discovered data in your environment. On the other hand, it automates the creation and update of Node and Asset information in UCMDB by populating asset data from Asset Manager to UCMDB. It takes advantage of the built-in procurement and retirement processes of Asset Manager to enable UCMDB to manage IT assets which are not in operation and are undiscoverable.

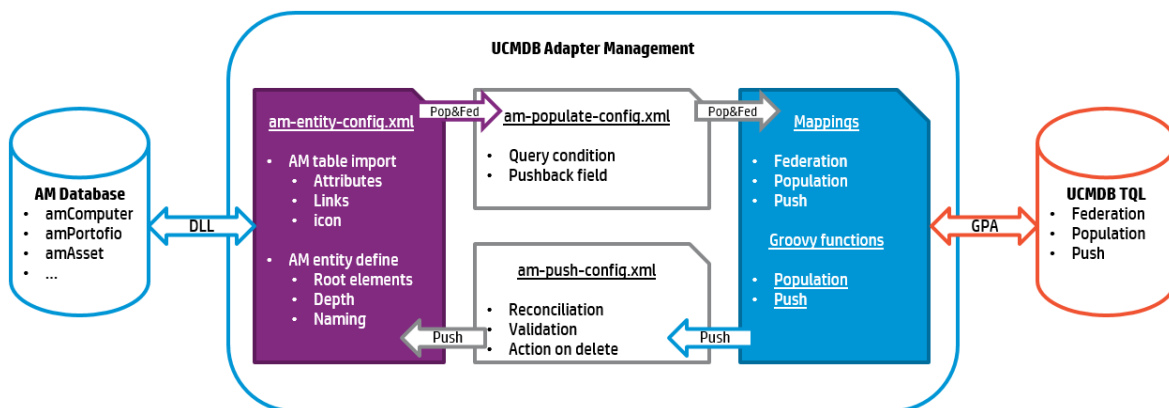
**Note:** This Integration replaces the Connect-It Scenarios used for synchronizing hardware and software information from DDMI 9.3x (and earlier versions) to Asset Manager. Also, this integration replaces the Connect-It Scenarios used for synchronizing Business Services and Business Applications from UCMDB to Asset Manager.

When referring to the concept of data information, it is important to distinguish between a UCMDB CI (Configuration Item) and an Asset Manager Asset. Both are defined in a different Data Model, and there must be a conversion before transferring CIs in UCMDB to Assets in Asset Manager and vice versa.

## Supported Versions

This integration supports HP Asset Manager 9.41 and later versions.

## Architecture



Asset Manager Generic Adapter is a new version of the AM adapter for exchanging data between Asset Manager and UCMDB. It bundles 3 core features of the AM and UCMDB integration in a single adapter: population, federation, and push. The AM Generic Adapter is built on top of the Generic Adapter (GA), which is a unified framework for external systems to integrate with UCMDB. The GA framework provides a graphic user interface for integration developers and administrators to ease the creation and customization of data mapping files. In the graphic mapping UI, you can easily map attributes and relationships between AM Entity and UCMDB CI.

AM Entity is a new layer introduced in the AM Generic Adapter. For more information, see "[Asset Manager Entity](#)" on page 330.

### **Population and Federation**

The AM Generic Adapter retrieves data from the AM database through the AM native APIs and creates AM entity objects with regards to the AM entity structure defined in `am-entity-config.xml`. Conditions specified in the file `am-populate-config.xml` filter the data retrieved from AM for a given entity.

The AM entity objects are converted to UCMDB CI structure according to the mapping script defined in mapping files. The Generic Adapter framework transmits the mapped CIs to the UCMDB Data In engine for processing.

### **Push**

UCMDB stores its information using CIs. The integration chooses which data to be pulled from UCMDB by defining integration TQL queries. Each TQL query defines a superset of data relevant for the integration.

The UCMDB Push Engine:

- Retrieves the required data from UCMDB, using the given TQL query.
- Filters the data to include only the data that has changed since the last execution of this synchronization.
- Splits the data into multiple chunks without breaking consistency.
- Sends the information to the Probe/Adapter.

The Generic Adapter framework allows easy mapping of the data from the UCMDB data model into the Asset Manager Data Model. It also allows transfer of this converted data into the AM Generic Adapter.

The AM Generic Adapter connects to the AM database through the AM native APIs to reconcile, push, and handle the complex logic needed to synchronize data into Asset Manager.

## **How to Integrate UCMDB and Asset Manager**

This section describes how to integrate UCMDB and Asset Manager.

## HP Asset Manager Setup

To set up Asset Manager for the integration, following the steps described in this section.

### Validate Pre-Loaded Data in Asset Manager

For the integration to succeed, some basic data is required to exist in the Asset Manager database.

This data may either be imported during the database creation (using the Asset Manager Application Designer), or may be added later. For more information, see the Asset Manager Documentation – Administration.

For hardware synchronization the required data is:

- Shared Data
- UNSPSC Product Classification
- Portfolio – Line-of-business data
- Virtualization – Line-of-business data
- Business services management – Line-of-business data

For software synchronization the required data is:

- Software Asset Management – Line-of-business data

### Create an Account with Administrative Rights

For the integration, any user with administrative rights will suffice. Asset Manager OOTB installations include an administrator account.

The details of the default Administrator user are:

- User: Admin
- Password: <empty>

The following example shows how to create a new user (named Integration-Admin) with administration rights, specifically for the integration.

1. Log on to Asset Manager as the Administrator.
2. Go to **Organization Management > User actions > Add a user.**
  - a. In **ID #**, type: Integration-Admin.
  - b. In **Name**, type: Integration-Admin.
  - c. In **First**, type: Integration-Admin.
  - d. Click **Next**.
  - e. Click **Next**.
  - f. Click **Finish**.
3. Go to **Organization Management > Organization > Employees.**
4. Select the newly created user and click the **Profile** tab.
5. In **User name**, type: Integration-Admin.
6. In **Password**, type: <A password you would like to use>.
7. In the **Password Administration** pane, ensure that **Never Expires** is selected.
8. In the **Profile** pane, ensure **Administration rights** is selected.
9. Click **Modify**.

### Update Asset Manager Schema

By default, the Asset Manager database schema includes column lengths that may be significantly shorter than their counterparts in the UCMDb database schema. For attributes used for reconciliation, this may be critical and may cause creation of multiple records.

To fix this issue, we recommend that you change the Asset Manager Column Sizes to the values shown in the following table of Asset Manager attributes.

Table	Name	Default Max Length	New Value
<b>amAsset</b>	<b>SerialNo</b>	128	250
<b>amModel</b>	<b>Name</b>	128	250
<b>amSoftInstall</b>	<b>Field1</b>	26	255



Table	Name	Default Max Length	New Value
amBrand	Name	128	250
amEmplDept	UserName	128	200
amEmplDept	UserDomain	128	200

**Note:** The new values in the table are only a suggestion, and you may need to change them according to actual data per customer use case.

**Note:** For DB2, the default table space page size of 4K may be too small in some cases. We recommend that you use 8K or higher larger.

## Activate Workflows for Population

In Asset Manager, the amComputer is an overflow table to the amPortfolio table. Part of its attributes are stored in the amPortfolio table. For instance, seAssignment. Some other attributes are distributed to other tables, such as amAsset. By default, if you make changes to any other tables than amComputer, its Last Modified Time does not change. As a result, when you run a delta sync on a job to populate Node, the changes will not be captured. To solve this problem, you need to activate two workflows in Asset Manager. To do this, follow these steps.

1. Log on to the Asset Manager client.
2. On the Tools menu, point to **Workflow**, click **Workflow schemes**.
3. Locate the **Update dtRecCreation** (SQL name: sysCoreUpCrTime) workflow and the **Update last modify time** (SQL name: sysCoreUpMdifTime) workflow.
4. On the **General** tab, empty the **End field of the Validity** pane.
5. Save the changes.

## Prepare Asset Manager for Parallel Push

Enabling Parallel Push significantly improves the performance of the push. However, some advanced preparations are necessary. Different actions are needed for different database types, as shown in the following table.

Database	Action
DB2	Mandatory: Follow <b>Eliminating locks and deadlocks</b> in the Asset Manager Tuning Guide.
Oracle	Optional: Follow <b>Eliminating locks and deadlocks</b> in the Asset Manager Tuning Guide.
SQL Server	Mandatory: Follow the steps below.

### Mandatory steps for SQL Server

1. Alter the SQLServer Schema isolation level. To do this, execute the following command on the database, replacing <AMSchema> with the real schema name.

```
ALTER DATABASE <AMSchema> SET READ_COMMITTED_SNAPSHOT ON
ALTER DATABASE <AMSchema> SET ALLOW_SNAPSHOT_ISOLATION ON
GO
```

**Note:** If the execution takes too long, you may need to disconnect all connections to the database. One possible way is to restart the database service, then execute the command. If you restart SQL Server and an Integration Point has already been created in UCMDB, you should restart the UCMDB Probe as well to avoid the issue of dead connections.

2. Alter Asset Manager database options:
  - a. Open the Asset Manager client and connect to the appropriate database schema.
  - b. Navigate on the top menus to **Administration > Database options**.
  - c. For the option **Sql Server specifics'Isolation command before starting a write transaction**, change the current value to **set transaction isolation level snapshot**.
3. Follow **Eliminating locks and deadlocks** in the Asset Manager Tuning Guide.

### Create Asset Manager API Zip Package

In order for the adapter to connect to the appropriate Asset Manager version, you must supply the Data Flow Probe/Integration Service with the appropriate Asset Manager API DLLs and Jars. To do this, follow these steps.

1. Copy the files below:

- <Asset Manager Installation folder>\x64\\*.dll
- <Asset Manager Installation folder>\websvc\lib\\*.jar

2. Create a package named AMGenericAdapterAPI\_<AM Version Number>.zip. For example, for version 9.41 the package name is AMGenericAdapterAPI\_9.41.zip.

3. Paste the copied files to:

AMGenericAdapterAPI\_<AM Version Number>.zip\discoveryResources\AMGenericAdapter\amVersion\<AM Version Number>

For example, for version 9.41, the path is:

AMGenericAdapterAPI\_9.41.zip\discoveryResources\AMGenericAdapter\amVersion\9.41

## HP UCMDB Setup

To set up UCMDB for the integration, follow the steps detailed in this section.

### Deploy Asset Manager API Zip Package

1. In the UCMDB client, go to **Administration > Package Manager**.
2. Deploy the AMGenericAdapterAPI\_<AM Version Number>.zip package you created.

### Install a Database Client


You must install the database client software according to the type of database the Asset Manager schema is installed on, as detailed in the following table.


Database	Client Software
SQL Server	None required.

Database	Client Software
DB2	<ol style="list-style-type: none"> <li>1. Download and Install "IBM Data Server Client" 64 bit for windows on your Data Flow Probe/Integration Service computer. This may be downloaded from:  <a href="http://www-01.ibm.com/support/docview.wss?rs=4020&amp;uid=swg21385217">http://www-01.ibm.com/support/docview.wss?rs=4020&amp;uid=swg21385217</a></li> <li>2. Create a connection to the DB2 database of Asset Manager.  You may create this using the DB2 Control Center.  <b>Note:</b> Remember the Database Alias you define in the connection, because you need it when creating the integration point.</li> <li>3. Copy the <b>db2cli64.dll</b> file from the DB2 client bin directory (By default: C:\Program Files\IBM\SQLLIB\BIN) to the <b>&lt;Data Flow Probe/Integration Service&gt;\lib</b> folder.</li> <li>4. Restart the Data Flow Probe/Integration Service.</li> </ol>
Oracle	<p><b>Oracle client windows 64 bit</b></p> <p>For example: Oracle Database 11g Release 2 Client (11.2.0.1.0) for Microsoft Windows (x64).</p> <ol style="list-style-type: none"> <li>1. Download the client installation from:  <a href="http://www.oracle.com/technetwork/database/enterprise-edition/downloads/index.html">http://www.oracle.com/technetwork/database/enterprise-edition/downloads/index.html</a></li> <li>2. Install the client, in Administrator mode, on the Data Flow Probe/Integration Service computer.</li> <li>3. Copy <b>oci.dll</b> from the <b>&lt;Oracle Client installation directory&gt;</b> into:  <b>&lt;Probe/Integration Service installation directory&gt;\lib.</b></li> <li>4. Restart the Data Flow Probe/Integration Service.</li> </ol>

### Create an Integration Point in UCMDB

1. Log on to UCMDB as an administrator.
2. Go to **Data Flow Management > Integration Studio**. UCMDB displays a list of existing integration points.

3. Click the  button. The New Integration Point dialog box is displayed.
4. Complete the Integration and Adapter Properties fields as shown in the following table:

Field	Required	Description
Integration Name	Yes	Type the name (unique key) of the integration point.
Integration Description	No	Type a description of the current integration point.
Adapter	Yes	Select <b>HP Software Products &gt; Asset Manager &gt; Asset Manager Generic Adapter</b>
Is Integration Activated	Yes	Select this option to indicate the integration point is active.
Hostname/IP	Yes	Type the hostname or IP Address of the Asset Manager database.
DB Type	Yes	Select the database type your Asset Manager schema is located on.
DB Port	Yes	Type the communication port of the Asset Manager Data Base.
DB Name/SID	Yes	<ul style="list-style-type: none"> <li>■ <b>DB2:</b> type in the name of the Database Alias you defined in the database connection.</li> <li>■ <b>Oracle:</b> type the service name.</li> <li>■ <b>SQL Server:</b> type the name of the schema.</li> </ul>
DB Owner Name	No	Enter the owner of the AM database schema.
Credentials ID	Yes	<p>Select <b>Asset Manager Protocol</b>. To create a new protocol, click the  button. Under <b>Asset Manager Protocol</b> complete:</p> <ul style="list-style-type: none"> <li>■ <b>Asset Manager User Name:</b> An AM administrator's user name.</li> <li>■ <b>Asset Manager Password:</b> An AM administrator's password.</li> <li>■ <b>DB User Name:</b> The AM database user's name.</li> <li>■ <b>DB Password:</b> The AM database user's password.</li> </ul>

Field	Required	Description
AM Version	Yes	Select the version of Asset Manager this integration point is to connect to.
AM Population Push Back ID	No	Select <b>True</b> to allow the adapter to push the Global Id of UCMDB CI back to the corresponding AM entity.
AM Other Connection Options	No	Options can be used in the amdb.ini file. See Asset Manager Installation Guide, Chapter .ini, .cfg, and .res files > Amdb.ini File Entries.
Enable Parallel Push	No	Select to allow parallel (multi-threaded) data push to Asset Manager. This improves performance. Note that you must configure SQL Server & DB2 to support parallel push. See "Prepare Asset Manager for Parallel Push".
Data Flow Probe	Yes	Select the name of the Data Flow Probe/Integration service used to execute the synchronization from.
Additional Probes	No	Select additional probes to use when pushing to AM in order to increase redundancy.

5. Click **Test Connection** to make sure there is a valid connection.
6. Click **OK**.

The integration point is created and its detailed are displayed.

UCMDB creates a default data push job when creating the integration point. If needed you may create or edit the existing job. For more information, see "Work with Data Push Jobs" in the *HP Universal CMDB Data Flow Management Guide*.

## Out-of-Box Integration Jobs

AM Generic Adapter is shipped with out-of-box jobs for population and push between Asset Manager and UCMDB. Integration job is driven by UCMDB TQLs.

To access these TQLs, follow these steps.

1. Log on to UCMDB as an administrator.
2. Go to **Modeling > Modeling Studio > Resources**, and then select **Queries** from the **Resource Type** drop-down list.
3. Expand **Root/Integration/AMGenericAdapter**. There are 3 sub folders: federation, population and push.
  - All queries under the federation folder federate CI data from AM to UCMDB.
  - All queries under the population folder manage CI data synchronization from AM to UCMDB.
  - All queries under the push folder manage CI data synchronization from UCMDB to AM.

## Asset Manager Population Jobs

In the following table, all TQLs which are included in the default population job are listed. For each TQL query, it describes the source AM entity type and the target CI types converted by the out-of-box mapping XMLs. In the description column, it contains a summary of the TQL query, the out-of-box mapping XML to convert the data, and the other data populations it depends on, which need to be run beforehand.

TQL Query	AM Entity	UCMDB CI Type	Description
AM Location Population 2.0	Location	Location <ul style="list-style-type: none"> <li>Location</li> </ul>	Populates location.  <b>Mapping XML:</b> AM Location Population.xml
AM Node Population 2.0	ITEquipment	Node <ul style="list-style-type: none"> <li>Location</li> <li>Asset</li> </ul>	Populates Node with Asset and Location.  <b>Mapping XML:</b> AM Node Population.xml
AM Interface Population 2.0	NetworkCard	Interface <ul style="list-style-type: none"> <li>Node</li> <li>IpAddress</li> </ul>	Populates Interface with IpAddress and Node relations.  <b>Depends On:</b> AM Node Population 2.0  <b>Mapping XML:</b> AM Interface Population.xml
AM BusinessElement Population 2.0	Asset	BusinessElement <ul style="list-style-type: none"> <li>Asset</li> </ul>	Populate Business Element with Asset.  <b>Mapping XML:</b> AM BusinessElement Population.xml



TQL Query	AM Entity	UCMDB CI Type	Description
AM BusinessElement Relations Population 2.0	Client_ Resource_ Relationship	BusinessElement <ul style="list-style-type: none"> <li>• BusinessElement</li> <li>• Node</li> </ul>	Populate Business Element relationship with Business Element and Node. Depends On: <ul style="list-style-type: none"> <li>• AM Node Population 2.0</li> <li>• AM BusinessElement Population 2.0</li> </ul> <b>Mapping XML:</b> AM BusinessElement Relations Population.xml
AM Printer Population 2.0	Portfolio_ Printer	Printer <ul style="list-style-type: none"> <li>• Node</li> </ul>	Populates Printer with the relation to Node. <b>Depends On:</b> AM Node Population 2.0 <b>Mapping XML:</b> AM Printer Population.xml

## Asset Manager Push Jobs

The below table lists all TQLs for data push contained in the out-of-box AM Generic Adapter. Some of them are included in the default push job. For each TQL query, it describes the source CI type and the target AM Entity type converted by the out-of-box mapping XMLs. In the description column, it contains a summary of the TQL query, the out of the box mapping XML to convert the data, and the other data pushes it depends on, which need to be run beforehand.

TQL Query	AM Entity	UCMDB CI Type	Description
AM Node Push 2.0	ITEquipment <ul style="list-style-type: none"> <li>PhysicalDrives</li> <li>NetworkCards</li> <li>LogicalDrives</li> <li>ExtensionCards AddOn (Printer, Monitor)</li> </ul>	Node <ul style="list-style-type: none"> <li>Asset</li> <li>CPU</li> <li>FileSystem</li> <li>LogicalVolume</li> <li>HardwareBoard</li> <li>DisplayMonitor</li> <li>DiskDevice</li> <li>InventoryScanner</li> <li>IpAddress</li> <li>Virtual Host Resource</li> <li>PhysicalPort</li> <li>VMWare Host Resource</li> <li>Printer</li> <li>Interface</li> </ul>	Pushes nodes (Computers, Network Devices, etc.). Also pushes IPs, Interfaces, Disk Devices, Physical Ports, Hardware Boards, Display Monitors, CPUs, Printers, Inventory Scanners, File Systems, and Assets.  Minimal attributes for pushing a Node: <ul style="list-style-type: none"> <li>Serial Number</li> <li>Vendor or Discovered Vendor</li> <li>Model or Discovered Model</li> <li>Node Role</li> </ul> <div style="background-color: #f0f0f0; padding: 5px; margin-top: 10px;"> <p><b>Note:</b> These are required values, and depend on the capability of the data source to report them.</p> </div> <p><b>Mapping XML:</b>                      AM Node Push.xml</p>

TQL Query	AM Entity	UCMDB CI Type	Description
AM Business Element Push 2.0	Asset	BusinessElement	<p>Pushes Business Applications, Business Services and Business Infrastructure CIs.</p> <p><b>Mapping XML:</b></p> <p>AM BusinessElement Push.xml</p>
AM Business Element Relations Push 2.0	Client_Resource_Relationship	BusinessElement <ul style="list-style-type: none"> <li>• BusinessElement</li> <li>• Node</li> </ul>	<p>Pushes relationships between Business Elements (pushed by AM Business Element Push Query) to other business elements or to nodes.</p> <p><b>Depends On:</b></p> <ul style="list-style-type: none"> <li>• AM Business Element Push 2.0</li> <li>• AM Node Push 2.0</li> </ul> <div style="background-color: #f0f0f0; padding: 5px; margin: 10px 0;"> <p><b>Note:</b> If the Asset Manager Business Element is not related to computers, AM Business Element Relations Push does not depend on AM Computer Push.</p> </div> <p><b>Mapping XML:</b></p> <p>AM BE Relations Push.xml</p>

TQL Query	AM Entity	UCMDB CI Type	Description
AM Host Server And Running VM Relations Push 2.0	Client_Resource_Relationship	Node <ul style="list-style-type: none"> <li>• Node</li> </ul>	Pushes relations between Host and Guest (Virtualized) operating systems.  <b>Depends On:</b> AM Node Push 2.0  <b>Mapping XML:</b> AM Host VM Relations Push.xml
AM Host Server And Running Solaris VM Relations Push 2.0	Client_Resource_Relationship	Unix <ul style="list-style-type: none"> <li>• Unix</li> </ul>	Pushes relations between Solaris Host and Guest (Virtualized) operating systems.  <b>Depends On:</b> AM Node Push 2.0  <b>Mapping XML:</b> AM Host VM Solaris Relations Push.xml
AM Host Server And Running LPAR VM Relations Push 2.0	Client_Resource_Relationship	Unix <ul style="list-style-type: none"> <li>• Unix</li> </ul>	Pushes relations between Host and Guest (virtualized) systems of LPAR type.  <b>Depends On:</b> AM Node Push 2.0  <b>Mapping XML:</b> AM Host VM Lpar Relations Push.xml

TQL Query	AM Entity	UCMDB CI Type	Description
AM Computer Relations Push 2.0	Client_Resource_Relationship	Node <ul style="list-style-type: none"> <li>• Node</li> </ul>	Pushes relations between Computers to any node (Computers, Network Devices, etc.).  <b>Depends On:</b> AM Node Push 2.0  <b>Mapping XML:</b> AM Computer Relations Push.xml
AM Net Device Relations Push 2.0	Client_Resource_Relationship	Node <ul style="list-style-type: none"> <li>• Node</li> </ul>	Pushes relations between Network Devices and Network Devices.  <b>Depends On:</b> AM Node Push 2.0  <b>Mapping XML:</b> AM NetDevice Relations Push.xml
AM Installed Software Push 2.0	SoftwareInstallation	InstalledSoftware <ul style="list-style-type: none"> <li>• Node</li> <li>• InventoryScanner</li> <li>• UserSoftwareUtilization</li> </ul>	Pushes Installed Software and User_Software_Utilization CIs. These Installed Software will be normalized in AM after push.  <b>Depends On:</b> AM Node Push 2.0  <b>Mapping XML:</b> AM InstalledSoftware Normalize Push.xml

TQL Query	AM Entity	UCMDB CI Type	Description
AM Installed Software Normalized Push 2.0	SoftwareInstallation	InstalledSoftware <ul style="list-style-type: none"> <li>• Node</li> <li>• InventoryScanner</li> <li>• UserSoftwareUtilization</li> </ul>	Pushes Installed Software and User_Software_Utilization CIs. These Installed Software should already be recognized by normalization technology in UCMDB, for example, BDNA. They will not be normalized in AM after push. <p><b>Depends On:</b></p> AM Node Push 2.0 <p><b>Mapping XML:</b></p> AM InstalledSoftware Non-normalize Push.xml
AM Software Hypervisor Push 2.0	SoftwareInstallation	Hypervisor <ul style="list-style-type: none"> <li>• Node</li> </ul>	Pushes Hypervisor Installed Software. <p><b>Depends On:</b></p> AM Node Push 2.0 <p><b>Mapping XML:</b></p> AM Hypervisor Push.xml
AM Software Solaris Push 2.0	SoftwareInstallation	Unix <ul style="list-style-type: none"> <li>• Unix</li> </ul>	Pushes Solaris Installed Software. <p><b>Depends On:</b></p> AM Node Push 2.0 <p><b>Mapping XML:</b></p> AM Solaris Zone Push.xml

TQL Query	AM Entity	UCMDB CI Type	Description
AM Oracle LMS Push 2.0	MonitoredApplication	Oracle <ul style="list-style-type: none"> <li>Node</li> <li>InstalledSoftware</li> <li>AuditDocuement</li> </ul>	<p>Pushes the Oracle Running Software and its Oracle LMS data.</p> <p><b>Depends On:</b> AM Node Push 2.0</p> <p><b>Mapping XML:</b> AM Oracle LMS Push.xml</p>
AM Cluster Push 2.0	Cluster	Cluster <ul style="list-style-type: none"> <li>Asset</li> </ul>	<p>Pushes Cluster CIs. It is not included in the default push job.</p> <p><b>Mapping XML:</b> AM Cluster Push.xml</p>
AM Cluster Node Relations Push 2.0	Cluster_Component_Relationship	Cluster <ul style="list-style-type: none"> <li>Node</li> </ul>	<p>Pushes cluster relations between Computer Elements and Cluster.</p> <p><b>Depends On:</b> AM Node Push 2.0 AM Cluster Push 2.0</p> <p><b>Mapping XML:</b> AM Cluster Node Relations Push.xml</p>
AM Cluster Runningsoftware Push 2.0	MonitoredApplication	Runningsoftware <ul style="list-style-type: none"> <li>Node</li> <li>Cluster</li> </ul>	<p>Pushes Cluster Runningsoftware CIs. By default, it is not included in the push job.</p> <p><b>Depends On:</b></p> <ul style="list-style-type: none"> <li>AM Node Push 2.0</li> <li>AM Cluster Push 2.0</li> </ul> <p><b>Mapping XML:</b> AM Cluster RunningSoftware Push.xml</p>

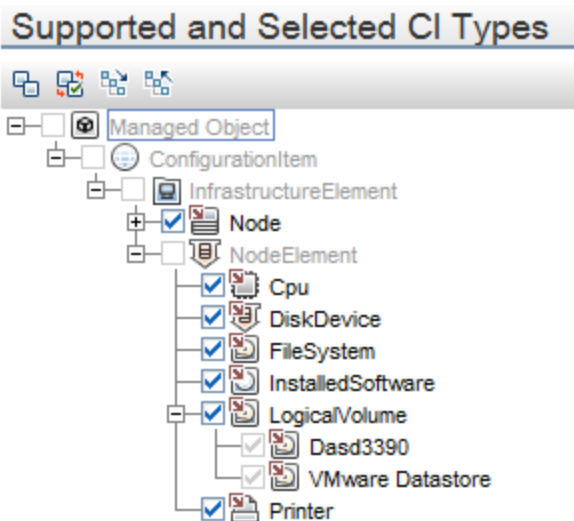
TQL Query	AM Entity	UCMDB CI Type	Description
AM Cluster Runningsoftware Relations Push 2.0	Cluster_Component_Relationship	Cluster <ul style="list-style-type: none"> <li>• Runningsoftware</li> </ul>	<p>Pushes cluster RunningSoftware relations between Computer Elements and Cluster. By default, it is not included in the push job.</p> <p><b>Depends On:</b></p> <p>AM Cluster Push 2.0</p> <p>AM Cluster Runningsoftware Push 2.0</p> <p><b>Mapping XML:</b></p> <p>AM Cluster RunningSoftware Relations Push.xml</p>



# Asset Manager Federation Configuration

By default, the following supported and selected CI types can be federated from Asset Manager:

- Node
- CPU
- DiskDevice
- FileSystem
- InstalledSoftware
- LogicalVolume
- Printer



The following table lists all TQLs for the data federation that are contained in the out-of-box AM Generic Adapter. For each TQL query, it describes the source AM Entity type and the target CI type converted by the out-of-box mapping XMLs.

TQL Query	AM Entity	UCMDB CI Type	Mapping XML
AM Federation cpu_12 2.0	ITEquipment	CPU	AM Cpu Federation.xml

TQL Query	AM Entity	UCMDB CI Type	Mapping XML
AM Federation cpu_node 2.0	ITEquipment	CPU <ul style="list-style-type: none"> <li>• Node</li> </ul>	AM Cpu Node Relations Federation.xml
AM Federation disk_device_12 2.0	PhysicalDrive	DiskDevice	AM DiskDevice Federtion.xml
AM Federation disk_device_node 2.0	PhysicalDrive	DiskDevice <ul style="list-style-type: none"> <li>• Node</li> </ul>	AM DiskDevice Node Relations Federtion.xml
AM Federation file_system_12 2.0	LogicalDrive	FileSystem	AM FileSystem Federtion.xml
AM Federation file_system_node 2.0	LogicalDrive	FileSystem <ul style="list-style-type: none"> <li>• Node</li> </ul>	AM FileSystem Node Relations Federtion.xml
AM Federation installed_software_12 2.0	SoftwareInstallation	InstalledSoftware	AM InstalledSoftware Federtion.xml
AM Federation installed_software_node 2.0	SoftwareInstallation	InstalledSoftware <ul style="list-style-type: none"> <li>• Node</li> </ul>	AM InstalledSoftware Node Relations Federtion.xml
AM Federation logical_volume_12 2.0	LogicalDrive	LogicalVolume	AM LogicalVolume Federtion.xml
AM Federation logical_volume_node 2.0	LogicalDrive	LogicalVolume <ul style="list-style-type: none"> <li>• Node</li> </ul>	AM LogicalVolume Node Relations Federtion.xml
AM Federation printer_12 2.0	Portfolio_Printer	Printer	AM Printer Federtion.xml
AM Federation printer_node 2.0	Portfolio_Printer	Printer <ul style="list-style-type: none"> <li>• Node</li> </ul>	AM Printer Node Relations Federtion.xml

## Verify Out-of-Box Population and Push Jobs



This section describes how to verify out-of-box population and push jobs.

# Synchronize Data between UCMDB and Asset Manager

The AM Generic Adapter provides two types of jobs:




- Data push jobs copy CI or CI relationship records from your UCMDB system to your Asset Manager system.
- Data population jobs copy Asset or Asset relationship records from your Asset Manager system to your UCMDB system.

To run a data push/population job, follow these steps:

1. Log on to UCMDB as an administrator.
2. Go to **Data Flow Management > Integration Studio**. UCMDB displays a list of existing integration points.
3. Select the integration point you created for Asset Manager.
4. Add a new data push/population job as follows:
  - a. Click the  button on the right panel.
  - b. In the Name field, type a unique name for the job.
  - c. Click the  button to add existing TQL queries to the job.
  - d. For push jobs, select or unselect the **Allow Deletion** option for each query. (The setting determines if this TQL query is allowed to delete data from Asset Manager, though the actual action on delete is defined by CI type in the mapping xml.)

For population jobs, select the **Allow Integration Job to Delete Removed Data** according to the requirement. (The setting determines if the population job is allowed to delete CIs from UCMDB.)

- e. Click **OK**.
- f. Save the integration point.

5. Manually run the job to see if the integration job works properly:
  - a. To push/populate all the relevant data for the job, click the  button.
  - b. To push/populate only the changes in the data since the job last executed, click the  button.
6. Wait for the job to complete, click the  button multiple times as needed until the job is completed.
7. When the job is completed, the job status becomes one of the following depending on the results:
  - Succeeded
  - Completed
  - Failed
8. Click the **Statistics** tab to view the results. If any errors occur, click the **Query Status** tab and **Job Errors** tab for more information. For more information about errors, see "[Troubleshooting and Limitations](#)".

**Note:** For details about these tabs and managing the integration, see **Integration Jobs Pane** in the *HP Universal CMDB Data Flow Management Guide* .

If the job completes successfully, you can view the UCMDB CI data in Asset Manager.

## How to View UCMDB Data in Asset Manager

After a push job is successfully completed, you can search for and verify that the pushed CI/relationship data is in Asset Manager.

### Nodes

This includes computers, network devices, and so on.

To view the nodes:

1. Log on to Asset Manager as an administrator.
2. Go to **Portfolio Management > Asset Configurations > IT equipment > Computers and virtual machines**.
3. In the opened dialog box, for **IP name**, type the name of the computer you are searching for.
4. You may use '<name prefix>%' for easier searching. For example, searching IP name for **mycomp%** returns computer mycomp1, mycomp2, and so on.
5. Browse the computer for the different data.

### Business Elements

This includes Business Applications, Business Services and Business Infrastructures.

1. Log on to Asset Manager as an administrator.
2. Go to **Asset Lifecycle > IT Services and virtualization > Business services > Business services**.
3. Browse the different Services.

For more information about viewing data in Asset Manager, see the Asset Manager Documentation.

## How to View Asset Manager Data in UCMDB

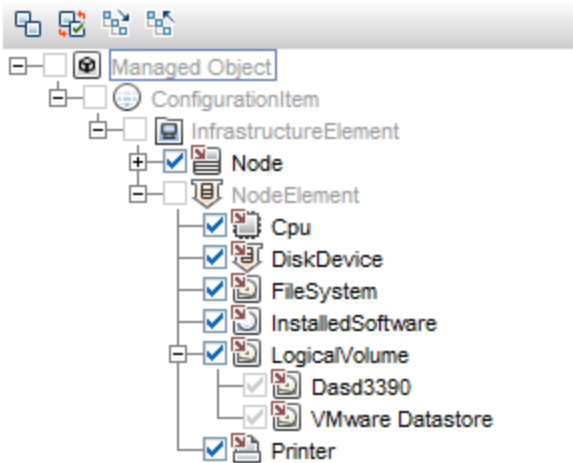
After a population job is successfully completed, you can search for and verify that the populated Asset data is in UCMDB. To do this, follow these steps.

1. Log on to UCMDB.
2. Go to **Modeling > IT Universe Manager**.
3. Click the **Search CIs** tab and search for the name of the Asset populated from Asset Manager.

# How to Federate Asset Manager Data in UCMDB

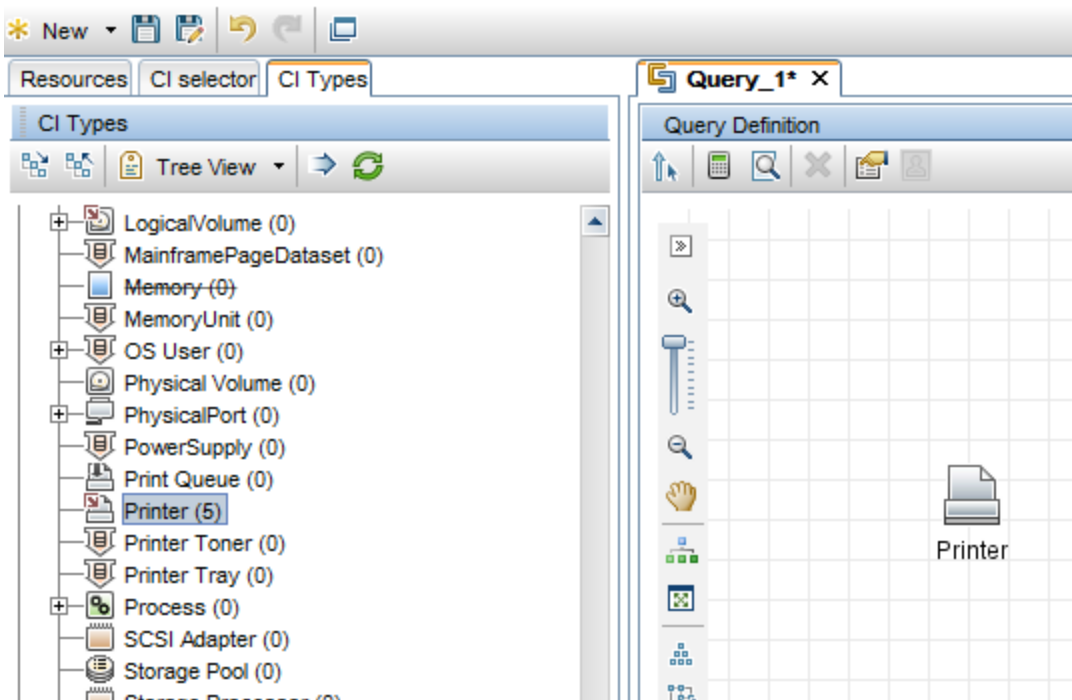
After Integration point of AM Generic Adapter is created and activated, you will find the supported CI types in the Federation page.

## Supported and Selected CI Types

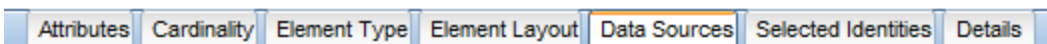


To federate these supported CI in UCMDB, you can create a query in **UCMDB client > Modeling > Modeling Studio > Resources**. Then, add a supported CI to federate data from AM. For example: Printer.

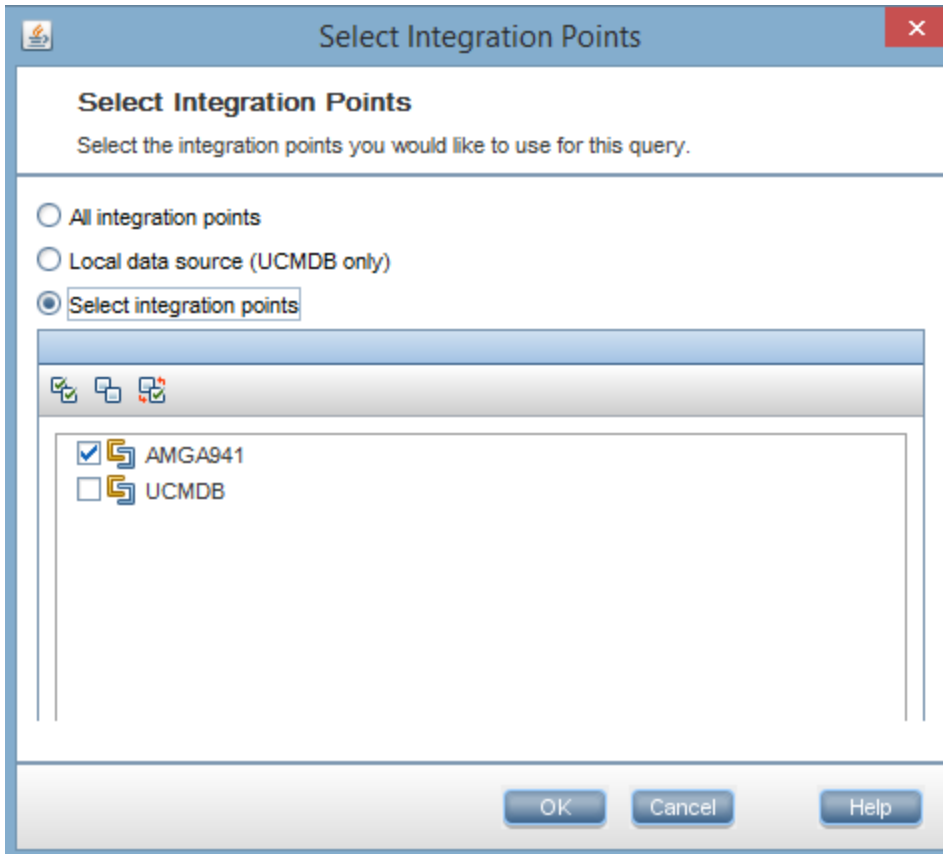




Click the CI icon, and then find Data Sources tab from the following area.



Edit Data Sources, select integration points from the following options, and then select integration points, for example, AMGA941.



Verify Calculate Query Result Count or Preview, you can federate those printers from AM server.




# Integration Jobs Configuration

This section describes the configurations to be made to the integration jobs.

## How to Schedule Data Integration Jobs

UCMDB allows you to schedule job executions directly from integration jobs. To do this, follow these steps:



1. Log on to UCMDB as an administrator.
2. Go to **Data Flow Management > Integration Studio**. UCMDB displays a list of existing integration points.
3. Select the integration point you created for the UCMDB - AM integration.
4. Select the push job.
5. Click the  button.

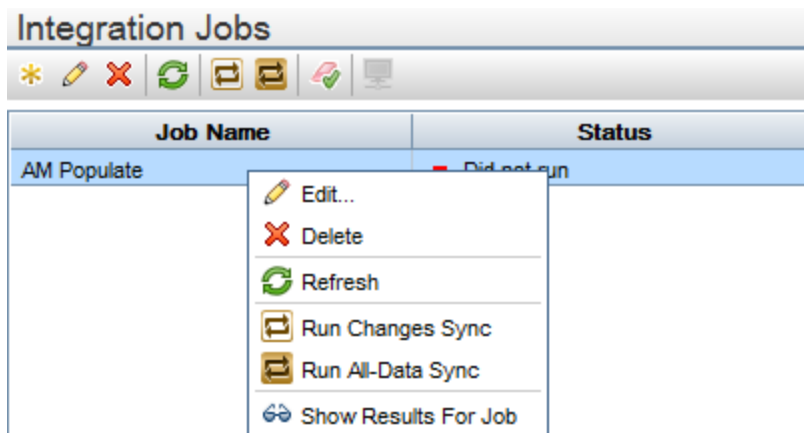
**Note:** UCMDB allows you to define two different schedules for two types of data push: **Changes Synchronization** and **All Data Synchronization**. We recommend that you use the Changes Sync schedule to only synchronize changes and avoid synchronizing the entire set of data each time.

6. Define a schedule for Changes Sync.
  - a. Click on the **Changes Synchronization** tab.
  - b. Select the **Scheduler enabled** option.
  - c. Select the scheduling options you want to use.
7. Click the **All Data Synchronization** tab and select the scheduling options you want to use.
8. Click **OK**.
9. Save the integration point.

## Edit Data Integration Jobs

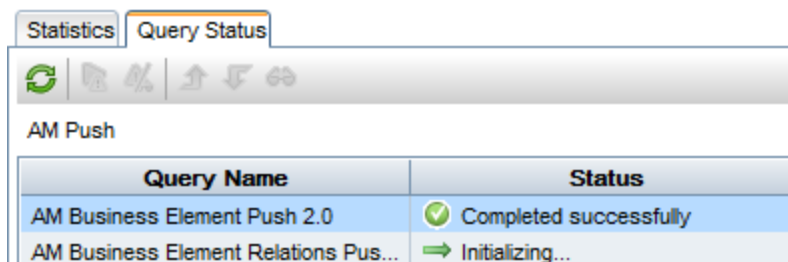
When you create an AM Generic Adapter integration point, a default job is created for data population and data push respectively. You can perform the following actions in the jobs.

- Add or remove TQLs.
- Clear cache  (available only for population jobs)
- Show Results For Job  (available only for population jobs)



### Query Status

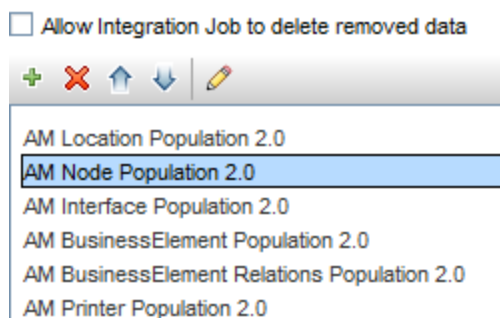
- Push selected failed data
- Suppress selected failures/warnings
- Up One Level/Down One Level
- View details



### Job Definition

- Allow Integration Job to delete removed data
- Add/Delete
- Move Query Up/Move Query Down
- Edit Query Resources

### Job Definition



## Standards and Concepts

This section describes standards and concepts.

### Asset Manager Entity

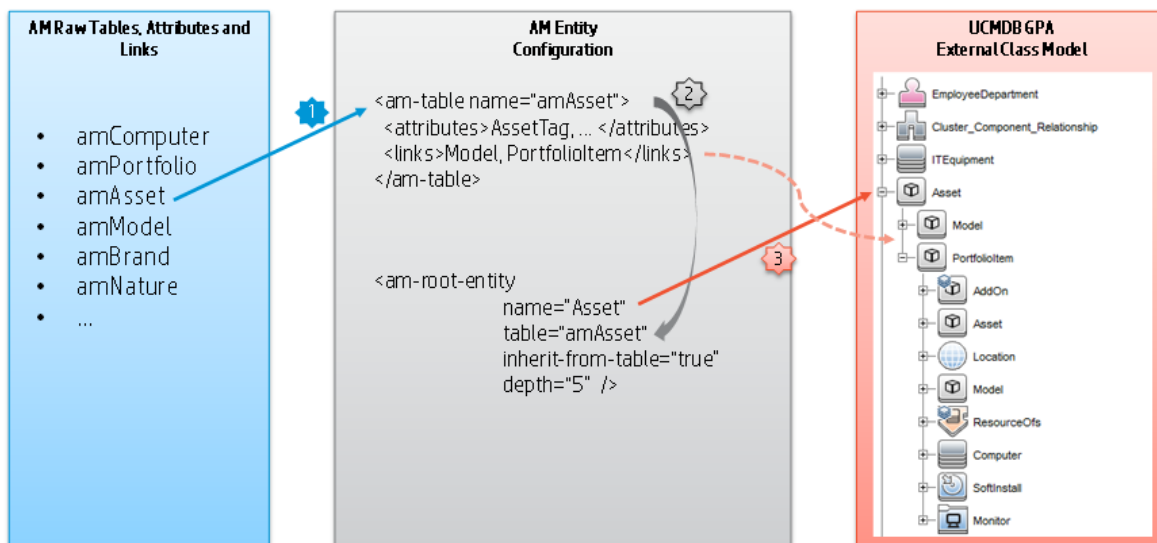
The AM Generic Adapter introduces a new concept named AM Entity. The AM Entity is a type of object that represents a collection of attributes abstracted from an AM table. All entities defined in the AM Generic Adapter make up the AM Class Model which represents the subset of AM objects that can be used in the population, push, as well as the graphic mapping UI. With the introduction of the AM Class Model (i.e. Entity) as an additional layer on top of AM database API, population and push mappings do not need to directly specify exact AM database tables or views. Extending AM Class Model is as simple as defining new entities by editing the am-entity-config.xml file. The entity name can be customized to a value that best fits your preference.

The entity is used mainly in the following places.

- It is displayed in the external class model pane on the mapping UI.
- It is referenced in mapping scripts.

- It is used in am-populate-config.xml for configuring initial AQL (AM Advanced Query Language) conditions to produce data from AM (used by population and federation).
- It is used in am-push-config.xml for reconciliation rule definition (used by push).

## Asset Manager Entity Definition Steps



The AM entity definition steps are as follows.

1. Import AM tables.
2. Create a new AM entity based on an imported AM table.
3. Define the entity's attributes and links to be used in the AM Generic Adapter.

The definition of the entities is saved in the am-entity-config.xml file. To access the file, go to **Data Flow Management > Adapter Management > Packages > AMGenericAdapter > Configuration Files**.

The am-entity-config.xml file consists of two sections.

- The **Import Tables** section contains the declaration of AM tables imported to the AM Generic Adapter, as well as what attributes and links in a table can be used in the adapter.
- The **Entity Definition** section contains the definition of AM entities which are used as the root level entity for data mapping. An entity is based on a table imported in the **Import Table** section. Additionally, an entity can specify a subset of attributes and links in the imported table to limit its own attributes and links.



## Import Tables and Entity Definition

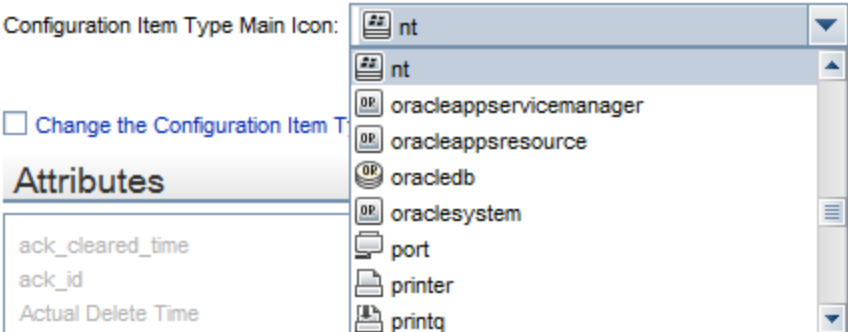
The following tables describe the schema of the am-entity-config.xml file.

### Import Tables

Name	Type	Description
am-tables	Tag	
name	Attribute	The SQL name of the AM database table.
attributes	Tag	Specifies attributes (SQL name) in the table that can be used in the adapter. Multiple attributes are separated by comma. Use * to import all attributes.
links	Tag	Specifies links (SQL name) in the table that can be used in the adapter. Multiple links are separated by comma. Use * to import all links.  <b>Note:</b> The table to which the link references must also be imported.

### Entity Definition

Name	Type	Description
am-entities	Tag	
am-root-entity	Tag	
name	Attribute	AM entity name. The entity name is used to reference an entity in the data mapping, AM Class Model, configuration for push, and population. We strongly recommend that you give it a meaningful name.
table	Attribute	The SQL name of the AM database table where the entity's attributes and links come from. The table must be imported in the <b>Import Tables</b> section.
inherit-from-table	Attribute	<ul style="list-style-type: none"><li>• True: merge attributes and links defined in the imported table to the entity.</li><li>• False: only attributes and links defined in the entity can be used. The default value is False.</li></ul>

Name	Type	Description
depth	Attribute	Defines the depth of level links in the entity can be extended. The default depth is 4, which means the deepest remote link you can use is entity.linkA.linkB.linkC.linkD.
icon-type	Attribute	<p>The icon to be displayed in the graphic mapping UI to represent the entity. It refers to the values from <b>UCMDB &gt; Modeling &gt; CI Type Manager &gt; Icon</b>.</p> 
attributes	Tag	Specifies the attributes (SQL name) in the table that can be used in the adapter. Multiple attributes are separated by comma. Use * to import all attributes.
links	Tag	Specifies the links (SQL name) in the table that can be used in the adapter. Multiple links are separated by comma. Use * to import all links. Note: The table to which the link references must be imported.

## Out-of-Box Entity Definition

The following table covers all of the out-of-box entities defined in the AM Generic Adapter. For the list of tables imported to the adapter, see the am-entity-config.xml file.

AM Entity Name	AM Table	Comment
ITEquipment	amComputer	<p>You can customize entity names by suffixing a sub CI type to define a more exact AM Entity. For example,</p> <ul style="list-style-type: none"> <li>ITEquipment_Windows</li> <li>ITEquipment_Unix</li> </ul> <p>If you want to query an AM table with different AQL conditions, you need to separate entities.</p>
Location	amLocation	
NetworkCard	amNetworkCard	
PhysicalDrive	amPhysicalDrive	
SoftwareInstallation	amSoftInstall	
Asset	amAsset	
LogicalDrive	amLogicalDrive	
Cluster	amCluster	
ClusterComponent	amClusterComponent	
ExtensionCard	amExtensionCard	
Monitor	amMonitor	
MonitoredApplication	amMonitoredApp	
EmployeeDepartment	amEmplDept	

AM Entity Name	AM Table	Comment
Portfolio	amPortfolio	<p>In practice, you can customize the entity name by AM natures or models to define more exact AM entities. For example,</p> <ul style="list-style-type: none"> <li>• Portfolio_NetworkHardware</li> <li>• Portfolio_StorageConsumer</li> <li>• Portfolio_StorageAggregation</li> <li>• Portfolio_StorageProvider</li> <li>• Portfolio_VirtualMachine</li> </ul>
Cluster_Component_Relationship	amClusterComponent	
Client_Resource_Relationship	amClientResource	
Portfolio_Printer	amPortfolio	<p>This entity is for local printers that only saved in amPortfolio table but not in the amComputer table. It is used in the data population. Its query condition is:</p> <pre>Model.Nature.Code = 'PRN' AND lParentId &lt;&gt; 0</pre>

## UCMDB TQL

AM Generic Adapter communicates with UCMDB through TQL queries.

- For push operations, a TQL query produces CIs which are the source to be translated to Asset Manager.
- For population, a TQL query defines CI types to be pulled from Asset Manager.
- For federation, a TQL query is used to configure the Data Sources of a CI node. For example, you can specify the Integration Points of a CPU node to an AM Generic Adapter integration point.
- For graphic mapping UI, a TQL query displays the CI relation tree.

A TQL query used for the data push job must contain a root query node.

In the OOTB TQLs for data push, some query nodes are set with criteria to query valid CIs to be translated to Asset Manager. For example, "Not NodeRole is null" is one of the conditions specified in the Node CI of the AM Node Push TQL.

Any attribute used in the mapping flow must be marked in the selected layout of the query node. Each TQL query may only have one mapping. We recommend that you only add attributes which are used in the data mapping to the selected layout. Add unused attributes to the selected layout may have impact on the performance.

You can customize TQLs, for example, add a condition to a query node or add a new query node .

### **TQL Naming Convention**

All OOTB TQLs follow the naming convention below. We recommend that you use the same naming convention when creating new TQLs for the AM Generic Adapter.

AM <CI Type> <Push | Population> <version #>

For example:

- AM Node Population 2.0
- AM Business Element Push 2.0

### **Groovy Functions**

Groovy is an agile and dynamic language, natively supported by the Java Virtual Machine. It allows simple scripting capabilities, while maintaining all the strengths and capabilities of Java. It can execute simple String manipulation, and use 3rd party libraries. For more information, see <http://groovy.codehaus.org/>.

## Basic Functions

There are two basic function groovy files:

- `AMPopulate.groovy`
- `AMPush.groovy`

Basic functions in these two files are for implementing data conversion and transformation in mapping scripts, and they will simplify mapping scripts, although mapping scripts also support basic groovy syntax.

## AM Population Groovy

There are some functions in population groovy implement translation and discrimination logic. The following list shows all the functions in `AMGenericAdapter/mappings/scripts/AMPopulate.groovy`.

- `convertIpAddressProperty`
- `getAssignment`
- `getCRState`
- `getHostDataCl`
- `getKpiComparisonOperator`
- `getLocationType`
- `getIpAddressProperty`
- `getIpAddressValue`
- `getNodeRole`
- `getNodeType`
- `getServiceClass`
- `isAcceptableNode`
- `isDebugEnabled`
- `isDeletedNode`
- `isDesktop`
- `isValidLocationParent`
- `isValidIp`
- `isValidNode`

- **isVirtual**
- **setLog**



## AM Push Groovy

There are some functions in push groovy implement business logic conversion, especially for Asset Manager SAM calculation. The following list shows all the functions in AMGenericAdapter/mappings/scripts/AMPush.groovy.

- calculateComputerType
- convertCsvBytesToString
- flsSAIValid
- fEDDGetMACAddressEx
- fEDDGetACComputerBrand
- fEDDGetACWorkGroup
- fEDDGetComputerManufacturer
- fEDDGetDomainNameEx
- fEDDGetSCLogicalName
- fEDDITruncate
- getAMPrimaryID
- getAMPrimaryID
- getAssetTagCRSystem
- getBrandName
- getCardID
- getCardName
- getCardVendorName
- getClusterModelName
- getClusterTechnology

- getCpuInternal
- getCpuType
- getCsvFile
- getDdEdition
- getDNSSuffix
- getDomainName
- getEnd1ExternalId
- getEnd2ExternalId
- getEnumValueByKey
- getExternalId
- getFinalModelId
- getFirstIpV4Address
- getFirstIpV6Address
- getFolder
- getLMSOptionStatus
- getLMSPacksStatus
- getInterfaceDescription
- getLogicalCpuCount
- getIpAddress
- getIpAddress
- getIpV4SubnetMask
- getIpV6SubnetMask
- getIsInventModelExist

- `getIsInventModelResolved`
- `getModelIdByBarCode`
- `getModelName`
- `getModelParentCode`
- `getMonitorCode`
- `getMonitorModelName`
- `getNatureCode`
- `getNatureCodeFromType`
- `getNatureNameFromType`
- `getNormalizedCPU`
- `getNormalizedModelName`
- `getOracleComponentList`
- `getOracleEditionShortName`
- `getOracleFolder`
- `getParentBarCodeFromType`
- `getParentName`
- `getPhysicalAddress`
- `getPrimaryIpAddress`
- `getSafeAMPrimaryID`
- `getSoftwareBarCode`
- `getUserName`
- `getValueFromCsv`
- `getValueFromStringCsv`

- `getVMType`
- `hasFinalModelId`
- `isDHCPEnabled`
- `isExtensionCardValid`
- `isLparVpar`
- `isSolarisZone`
- `nicelyPrintExternalId`
- `setLog`
- `shouldMapMonitor`
- `sortIpList`
- `validateAndRetrieveSingleUCCaseAsset`

## Utility Functions

Some general logic and functions are the same in the utility groovy file. `AMPopulate.groovy`, `AMPush.groovy`, and `AMReconcil.groovy` are inherited from this groovy. So in push or population mapping, even the utility groovy is not imported, all of the following functions in `AMGenericAdapter/mappings/scripts/AMUtils.groovy` can be used normally.

- `boolToInt`
- `compareDate`
- `containsIgnoreCase`
- `Extractvalue`
- `fCleanValueWithDefault`
- `fisContainOne`
- `fIsContainList`
- `fIsEmpty`
- `fIsEmptyOrZero`
- `isMatchVersion`
- `isNull`
- `leftPart`
- `rightPart`
- `toBoolean`
- `toSmart`
- `toStringList`
- `trimRight`
- `uCase`

## Reconciliation Functions

Reconciliation groovy file provides advanced features about AM reconciliation proposal, insert or update script, and so on. The following list shows all the functions in `AMGenericAdapter/mappings/scripts/AMReconcil.groovy`.

- `createReconcProposalAdvance`
- `getCurrDate`
- `getCurrDateLong`
- `getIndexByName`
- `getModelFromInstalledSoftwareCI`
- `getShortHostName`
- `getSqlTextConst`
- `getValueByIndex`
- `getValueByName`
- `isDateAfter`
- `isResultEmpty`
- `setReconcProposallInvalidate`
- `setReconcProposalObsolete`
- `stringToDom`
- `updateMemorySize`

### Data Mapping Schema

AM Generic adapter is built on top of the Generic Adapter framework. Its data mapping schema follows the standard schema in the framework. For more information, see [HP Universal CMDB Developer Reference Guide](#).

The AM Generic Adapter uses some of the elements specifically to fit the AM-UCMDB integration requirement.

### **Automatic Creation of Relation CI in Data Population**

When populating two CIs that are defined in TQL, their relation CIs are created in UCMDB automatically according to the TQL query definition. The explicit mapping for the relation between the two CIs is not required. For example, in the AM Interface Population XML, it does not define the data mapping for the relation between Interface and Node. However, the relation will be created automatically when you run the AM Interface Population job.

If you want to create the relation CI with non-default values, you need to define a relation CI mapping in the mapping XML file.

### **Define Reconciliation Rule for Target AM Entity in Push**

The name attribute of the target\_entity tag represents:

- The entity name defined in the am-entity-config.xml file when defining the target\_entity for the root CI.
- The link's SQL name in the AM Class Model when defining the target\_entity for non-root CIs.

For example, in the AM Node Push.xml, the name of the target\_entity for the root CI is set to ITEquipment, which is an entity type based on the amComputer table. The name of the target\_entity for the Root.File\_System CI is set to LogicalDrives, which is the SQL name of the link to the amLogicalDrive table in the ITEquipment entity.

The type attribute of the target\_entity tag represents an alias to be used in the am-push-config.xml file to define the reconciliation rules for the data mapping. For more information about how to define reconciliation rules, see ["Reconciliation" on page 366](#).

It is an optional attribute. When it is not given a value, it takes the value of the name attribute as its value.

For example, in the AM Node Push.xml, the type of the target\_entity for the Root.Display\_Monitor CI is set to Monitor-amPortfolio. In the am-push-config.xml file, it uses the type value Monitor-amPortfolio to define the reconciliation rule for the mapping.

```
<am-mapping ci-type="Monitor-amPortfolio" name="AddOn" primary-  
key="lPortfolioItemId" operation-type="update_else_insert" parallel-push-  
allowed="true" merge-allowed="true" to-version="9.4*">  
  <reconciliation>  
    <reconciliation-keys>  
      <reconciliation-key>Folder</reconciliation-key>  
      <reconciliation-key>lParentId</reconciliation-key>  
    </reconciliation-keys>
```

```
    </reconciliation>  
...  
</am-mapping>
```

### **Specify Sub CI Type in Population Data Mapping**

In the population data mapping, the type attribute of the target\_entity is used to specify the sub type of the target CI.

For example, in the AM Node Population.xml, the AM ITEquipment entity is mapped to the Node CI (the Node CI is set to Root in the TQL query). When the population job is running, the exact CI type the ITEquipment entity is converted and is determined by the value of the type attribute. In the out-of-box mapping, it calls a groovy function to set the value of the type attribute.

### **Determine the Deletion of AM Entities in Population**

In the population data mapping, the is-deleted attribute of the target\_entity tag is used to determine if the target UCMDB CI corresponding to the AM entity needs to be deleted.

### **Produce Log**

You can use the Logger object to produce log in the before-mapping and after-mapping tag.

For example,

```
<before-mapping>Logger.debug('before')</before-mapping>  
<after-mapping>Logger.debug('after')</after-mapping>
```

## Population and Federation

AM Generic Adapter is able to retrieve data from AM database via AM DLL API. This section describes the detailed criteria before query AM Entity.



## Criteria for Asset Manager Records to be Populated

AM Entity	AQL
ITEquipment	TcplpHostName <> " AND (TcplpHostName IS NOT NULL)
NetworkCard	Computer.TcplpHostName <> " AND (Computer.TcplpHostName IS NOT NULL) AND (TcplpAddress IS NOT NULL) AND (PhysAddress IS NOT NULL) AND (TcplpAddress <> ") AND (PhysAddress <> ")
SoftwareInstallation	IParentPortfolioId <> 0 AND ParentPortfolio.CMDBid IS NOT NULL AND ParentPortfolio.CMDBid <> " AND ParentPortfolio.Computer.TcplpHostName <> " AND (ParentPortfolio.Computer.TcplpHostName IS NOT NULL)
Portfolio_Printer	Model.Nature.Code = 'PRN' AND IParentId <> 0
LogicalDrive	Computer.TcplpHostName <> " AND (Computer.TcplpHostName IS NOT NULL)
PhysicalDrive	Computer.TcplpHostName <> " AND (Computer.TcplpHostName IS NOT NULL)
Asset	IAstId <> 0 AND PortfolioItem.Model.Nature.bSystem = 1
Client_Resource_Relationship	CRSystem.IAstId <> 0 AND CRSystem.Model.Nature.bSystem = 1

**Note:** If you want to populate the same AM entity with different query condition, you should duplicate AM entity, rename it to a new entity, and then add an AQL for this new entity.

# Transformation for Asset Manager Records to be Populated

## IT equipment

The default population mapping will populate an IT equipment when its status is In use or In stock.

According to the IT equipment type, data will be populated to UCMDB as different CI types:

CI types in UCMDB	IT Equipment Type in AM
host_node	<ul style="list-style-type: none"> <li>• Computer</li> <li>• Desktop computers</li> <li>• Computer servers</li> <li>• Laptop</li> <li>• Virtual Machine</li> <li>• Smart phone</li> </ul>
nt	<ul style="list-style-type: none"> <li>• Windows computer</li> <li>• Windows desktop computer</li> <li>• VMware VirtualCenter</li> </ul>
unix	<ul style="list-style-type: none"> <li>• Unix server computer</li> <li>• Unix desktop computer</li> <li>• Solaris Zone server</li> </ul>
vmware_esx_server	VMware ESX Server
mainframe	<ul style="list-style-type: none"> <li>• Mainframe</li> <li>• Mainframe CPC</li> </ul>
lpar	Mainframe Logical Partition

When a CI is populated to UCMDB, the Host is Virtual attribute of the CI is defined according to the value of IT Equipment type for the CI in AM.

Host is Virtual	IT Equipment Type in AM
Yes	<ul style="list-style-type: none"> <li>• Virtual Machine</li> <li>• Mainframe Logical Partition</li> </ul>
No	<ul style="list-style-type: none"> <li>• Windows computer</li> <li>• Windows desktop computer</li> <li>• VMware VirtuaCenter</li> <li>• VMware ESX server</li> <li>• Unix server computer</li> <li>• Unix desktop computer</li> <li>• Solaris Zone Server</li> <li>• Desktop computers</li> <li>• Computer servers</li> <li>• Laptop</li> <li>• Mainframe</li> <li>• ATM switch</li> <li>• Firewall</li> <li>• Router</li> <li>• Switch</li> <li>• Network printer</li> <li>• Smart phone</li> <li>• Mainframe CPC</li> </ul>

When a CI is populated to UCMDB, the Host is Desktop attribute of the CI is defined according to the value of IT Equipment for the CI in AM.

Host is Desktop	IT Equipment Type in AM
Yes	<ul style="list-style-type: none"> <li>• Windows desktop computer</li> <li>• Unix desktop computer</li> <li>• Desktop computers</li> </ul>
No	<ul style="list-style-type: none"> <li>• Windows computer</li> <li>• VMware VirtuaCenter</li> <li>• VMware ESX server</li> <li>• Unix server computer</li> <li>• Solaris Zone Server</li> <li>• Computer servers</li> <li>• Laptop</li> <li>• Mainframe</li> <li>• ATM switch</li> <li>• Firewall</li> <li>• Router</li> <li>• Switch</li> <li>• Network printer</li> <li>• Smart phone</li> <li>• Mainframe CPC</li> </ul>

**Business Element**

In AM, only those business service assets with status (amAsset.Status) Built, Catalogued, Chartered, Designed , and Requested will be populated to UCMDB. If the status is retired, it will be removed from UCMDB.

Relations between AM business service asset and CI types are as follows.

Transformation for Asset Manager Records to be Populated

Nature code	CI Type
BIZSVC	business_service
BIZAPP	business_application
INFRASVC	infrastructure_service

In AM Generic Adapter, all the transformers and discriminators are implemented in groovy functions. For more information, see ["Groovy Functions" on page 337](#).

## Reconciliation

For each CI Type, the data reconciliation is governed by the reconciliation rule set in UCMDB.

You can check the reconciliation rule for each CI Type on the Details tab of the CI Type. The field name is Identification.

## Population Condition and Push Back Definition

Name	Type	Description
am-populations	Tag	
am-population	Tag	
entity-name	Attribute	AM Entity name
push-back-field	Tag	The AM database field to hold the global id pushed from UCMDB CI back to AM. It supports AQL syntax, for example:  PortfolioItem.CMDBId
delta-sync-date-field	Tag	It is used to define a time condition field for AQL in changes population. By default, it is the field 'dtLastModif' that is from the AM entity which is used as the 'root-element' in the population mapping.
query-condition	Tag	The condition of the query to retrieve the entity data from AM. It supports AM AQL. If the condition includes special characters: >, <, ' surround with CDATA.  <query-condition><![CDATA[ lAstId <> 0 AND PortfolioItem.Model.Nature.bSystem = 1 ]]></query-condition>

## Built-in attributes from AM

- **AM\_PUSHBACK\_ID**

This attribute is created by population adapter automatically from AM. It is used to identify which AM record is written with push back global id.

- **LAST\_SYNC\_TIME**

This attribute is created by population adapter automatically. It keeps the last job changes sync time in order to be used in population mapping.



## Federation Tags

The configuration must be provided if you want to use federation.

It defines all valid federation TQL names.

Name	Type	Description
supported-federation-queries	Tag	
tql-query	Tag	
name	Attribute	TQL name

## Population Tags

The configuration is used to define all valid population TQL names.

Name	Type	Description
supported-population-queries	Tag	
enable	Tag	<ul style="list-style-type: none"><li>• True: only displays valid population TQL names for you, to select required population TQLs for a population job.</li><li>• False: displays all TQL names from UCMDB for you, to select required population TQLs for a population job.</li></ul>
tql-query	Tag	
name	Attribute	TQL name

### Push and Reconciliation

This section describes push and reconciliation operations.

## Data Flow Architecture

The data flow architecture is as follows.

1. The Push Engine executes the TQL query.
2. For a differential flow, the data is compared to the last synchronized data, and only the changes are forwarded.
3. Data is converted into Composite CIs (instances of data according to the TQL Root elements).
4. Data is then pushed to the Generic Adapter.
5. The Generic Adapter loads the correct mapping for the specific TQL query.
6. All **dynamic\_mappings** are executed and saved to maps, to allow usage in the next mapping stage.

For more information, see "Developing Enhanced Generic Push Adapters" in the *HP Universal CMDB Developer Reference Guide*.

7. Data is mapped from the UCMDB data Model into the AM Data model according to the mapping XML.
8. Data is sent to the AM Connector.
9. AM Connector orders all the data in a set of dependency trees, starting with the records that do not depend on any other record.
10. AM Connector attempts to merge any duplicate records
11. AM Connector starts reconciling and pushing any record without any dependencies, or a record whose dependencies have already been reconciled/pushed to Asset Manager.
  - a. AM Connector first tries to reconcile with existing records.
  - b. If it finds a match, it attempts to update that record.
  - c. If it does not find a match, it attempts to create a new record.
12. AM Connector deletes any records that are required to be deleted in AM, as permitted by action-on-delete.

## Integration TQL Queries

A TQL query used for the integration must contain a root query node.

Any attribute using in the mapping flow of the data push must be marked in the selected layout of the query node.

Each TQL query for data push job may only have one mapping.

For more information, see **Data Flow Management > Integration > Integration Studio > Integration Jobs Pane**.

## Reconciliation Proposals

When pushing data to Asset Manager, there is an option to create a reconciliation proposal (RP). A reconciliation proposal should be created if there is a change in a specific attribute that may need AM Operator validation or action to support AM business processes.

The OOTB configuration creates a reconciliation proposal record when the memory size of the pushed computer has decreased compared to the AM computer.

### How to use Reconciliation Proposals

In the OOTB configuration **IMemorySizeMb** is marked for attribute-reconciliation. The update script calls the **updateMemorySize** function. This function verifies if the memory size of the computer was decreased. It initializes all the parameters that are passed to the function **validateReconcUpdateAdvance**. Calls **validateReconcUpdateAdvance** and return its returned value.

**validateReconcUpdateAdvance** is a function that returns the value that should be set to the attribute according to the Reconciliation Proposal status. The following table describes its parameters:

Parameter	Description
<b>AMApiWrapper</b>	The wrapper that is used to communicate with the AM.
<b>newVal</b>	The value of the attribute in the pushed data.
<b>oldVal</b>	The value of the attribute that is retrieved from AM.
<b>recordId</b>	The primary key of the table that the attribute belongs to.
<b>strCode</b>	The prefix of the <b>code</b> field in the reconciliation proposal.
<b>strName</b>	The name of the reconciliation proposal.
<b>path</b>	The name of the attribute.
<b>recordTable</b>	The table that the attribute belongs to.

**validateReconcUpdateAdvance** returns the value that should be set for the attribute, according to the Reconciliation Proposal status.

In order to create a reconciliation proposal flow on a different attribute, the following steps must be completed:

1. Add the **<attribute-reconciliation>** tag for this attribute.
2. The update-script should call a new function that initializes the parameters passed to **validateReconcUpdateAdvance**, and returns the value returned from **validateReconcUpdateAdvance**.

**Note:** We recommend that you use the **updateMemorySize** function as a reference.

## Asset Manager Rules and Flows

Asset Manager has its own set of rules and flows that are enforced by the Asset Manager API. Some customizations may need to later these rules and flows as well. For more information, see the Asset Manager documentation.

## Mapping Attributes

Attributes in the <am-mapping-config> tag are as follows

Attribute	Description
<b>ci-type</b>	<p>It is used in am-push-config.xml to recognize the reconciliation rule. It must be the same as the 'type' attribute in AM mapping XML files.</p> <p>For example:</p> <ul style="list-style-type: none"> <li>• Mapping file AM Node Push.xml: &lt;target_entity name="AddOn" type=""Printer-amPortfolio"&gt;</li> <li>• Reconciliation file am-push-config.xml: &lt;am-mapping ci-type="Printer-amPortfolio" name="AddOn" ...&gt;</li> </ul>
<b>name</b>	AM entity name or sub link name.
<b>primary-key</b>	The primary key column in the AM database schema.
<b>operation-type</b>	<p>It defines what operations may be done with the record.</p> <p><b>insert</b> - Only allows creation of new records; if it already exists, an exception is thrown.</p> <p><b>update</b> - Only allows updates of an existing record; if it does not exist, an exception is thrown.</p> <p><b>update_else_insert</b> - If the record exists, it is updated. Otherwise, the record is created.</p> <p><b>reference-only</b> - The record is only used for being referenced by other records (and is not updated). An exception is thrown if the record does not exist.</p> <p><b>ignore</b> - The record is unaffected by operations.</p> <p><b>insert_else_reference</b> - If the record does not exist, it is created. Otherwise it is only used as a reference and is not be updated; see reference-only.</p> <p><b>optional_reference</b> - If reconciliation fails, it will not fail dependent CIs. Instead, 0 value is returned as ID.</p>
<b>parallel-push-allowed</b>	If enabled with the enabled-parallel-push configuration of the integration point, will attempt to push to the entity with multiple threads in order to increase performance.



Attribute	Description
<b>merge-allowed</b>	If enabled and this entity is an exact duplicate of another entity in the chunk, it merges both entities into one and fixes any relevant references.
<b>errorcode-override</b>	If used together with the adapter specific errors, allows the printing of a customized error message to the UI if the push or reconciliation of this entity fails.
<b>from-version</b>	Use this mapping only from (and including) this version. The version is taken from the integration point configuration.
<b>to-version</b>	Use this mapping only up to (and including) this version. The version is taken from the integration point configuration.

## Reconciliation

Reconciliation defined for each mapping may include more than one set of reconciliation rules. When attempting to reconcile the record with existing ones in the AM database, the AM Connector tries each of the reconciliation sets until it finds a matching record. Priority is defined by the order of reconciliation rules. If no record in the AM database matches this record, an insert operation is performed if the operation type permits it.

Name	Type	Description
<b>reconciliation</b>	Tag	Parent XML tag for all reconciliation configuration.
<b>reconciliation-keys</b>	Tag	Represents a single reconciliation rule that may be made of one or more attributes. All attributes inside the rule must match in order for the reconciliation of this rule to be successful.
<b>reconciliation-key</b>	Tag	Represents a single attribute used for reconciliation as part of the reconciliation-keys rule.
<b>Ignore-case</b>	Attribute	Part of the reconciliation-key tag. Specifies that this attribute comparison ignores case.
<b>reconciliation-advanced</b>	Tag	Allows definition of the reconciliation rule by manually defining the WHERE clause of the AQL (Asset Query Language). Uses GString (Groovy String) to generate the replacement String. Any variable or property defined in this record or its parent during the mapping stage (in the Push Adapter) may be used as a variable in the GString. (See <a href="http://groovy.codehaus.org/Strings+and+GString">http://groovy.codehaus.org/Strings+and+GString</a> for more information).  <b>Note:</b> AMPushAdvancesReconciliationException may be thrown inside this tag to skip to the next rule.
<b>follow-parent</b>	Tag	Used to define overflow tables. See the AM documentation for more information on overflow tables. When using follow-parent, no other reconciliation may be used as this target CI has a 1:1 connection with its parent, and it uses the parent reconciliation to push data to AM.
<b>am-prefix</b>	Attribute	Part of the follow-parent tag. Defines the name that the parent target CI uses to reference to this table. (To find out the correct value, navigate to <b>AM Application Designer &gt; Edit Links</b> ).

### Example:

```
<reconciliation>
  <reconciliation-advanced>Portfolio.CMDBId = '${if(globalId==null) { throw new
com.hp.ucmdb.adapters.ampush.exception.
AMPushAdvancesReconciliationException
('Not enough reconciliation data') }else{ return globalId}}'
  </reconciliation-advanced>
  <reconciliation-keys>
    <reconciliation-key ignore-case="true">AssetTag</reconciliation-key>
  </reconciliation-keys><reconciliation-keys>
    <reconciliation-key>TcpIpHostName</reconciliation-key>
    <reconciliation-key>Workgroup</reconciliation-key>
  </reconciliation-keys>
</reconciliation>
```

## Target CI Validation

This tag allows the definition of a validation rule that is executed on specific attribute values: the new one held in memory, and the old one stored in the AM database.

The following table shows the attributes of the **<target-ci-validation>** tag:

Name	Description
<b>attribute-name</b>	The attribute that you want to use for validation.
<b>validation-script</b>	A Groovy based script that returns <b>true</b> if this record is to be pushed, and <b>false</b> if it is not to be pushed. The script may access any external Groovy code in the path in order to run the evaluation. <ul style="list-style-type: none"><li>• <b>vNewVal</b> - Attribute value of the record in memory.</li><li>• <b>vOldVal</b> - Attribute value of the record in the AM database.</li></ul>
<b>failed-validation-error-code</b>	This optional attribute holds the error code that appears if there is a validation failure. The arguments that can be referenced in the error message are: <ul style="list-style-type: none"><li>• {0} - The validated attribute name.</li><li>• {1} - The property value in UCMDB.</li><li>• {2} - The property value in Asset Manager.</li><li>• {3} - The validation script.</li><li>• {4} - The additional message from the <b>additional-failure-message</b> attribute, or 'null' if there is no additional message.</li></ul>
<b>additional-failure-message</b>	This optional attribute holds an additional error message that can be referenced by the error message in the <b>properties.error</b> file. See <b>failed-validation-error-code</b> , above.

### Example:

```
<target-ci-validation attribute-name="dtLastScan" validation-script="mappings.scripts.AMReconciliationAdvanced.isDateAfter(vNewVal,vOldVal)"/>
```

## Reference Attribute

A reference attribute defines a column that references another record from a different or same table. This record is not pushed, or reconciled against existing AM database records, until this reference is resolved. Resolved references are replaced by a reference ID that represents the primary ID of the referenced record.

The following table shows the attributes of the **<reference-attribute>** tag:

Name	Description
<b>ci-name</b>	The CI-type of the referenced record.
<b>datatype</b>	The value type of the record.
<b>name</b>	The column in the current record that is to be populated by the reference ID.
<b>reference-direction</b>	According to the tree created by the Push Adapter, the value specifies if the referenced record is a parent or child of the current record.

### Example:

```
<reference-attribute ci-name="SW_amModel" datatype="STRING"
name="lModelId"reference-direction="child"/>
```

## Attribute Reconciliation

This tag allows the AM connector to decide what to do with an attribute value according to the existing value in the AM database.

The following table shows the attributes of the **<attribute-reconciliation>** tag:

Name	Description
<b>attribute-name</b>	The attribute to be reconciled.
<b>update-script</b>	The script to execute in case of an update operation on the record. The returned value by the groovy script will be push to AM as the value of this attribute. <ul style="list-style-type: none"><li>• <b>vNewVal</b> - Attribute value of the record in memory.</li><li>• <b>vOldVal</b> - Attribute value of the record in the AM database.</li></ul>
<b>Insert-script</b>	The script to execute in case of an insert operation on the record. The value returned by the Groovy script is pushed to AM as the value of this attribute. <b>vNewVal</b> - Attribute value of the record in memory.

### Example:

```
<attribute-reconciliation attribute-name="AssetTag" update-script="mappings.scripts.AMPushFunctions.fIsEmpty(vOldVal) ? vNewVal : vOldVal"/>
```

## Action on Delete

This tag allows customization of the behavior on receipt of a delete notification for a record.

**Note:** No deletion occurs if the **Allow Delete** option in the job definition is disabled.

The following actions are possible

- **<ignore>** - Do nothing.
- **<delete-ci>** - Delete this record from the AM database.
- **<set-attribute-value>** - Change the value of one or more attributes in the AM database.

**Example:**

```
<action-on-delete>  
  <set-attribute-value name="bMissing" datatype="BOOLEAN" value="1"/>  
</action-on-delete>
```

## Enum Attribute

This tag allows a specific enum attribute to be pushed in a serial mode, when the adapter is configured to push data in parallel mode.

**Note:** This option exists to prevent duplicate key exceptions occurring when several threads push the same enum value.

The following table shows the attributes of the **<enum-attribute>** tag:

Name	Description
<b>attribute-name</b>	The enum attribute name.
<b>itemized-name</b>	The itemized list format (amOS) of the enum.



## Ignored Attributes

This tag allows specific attributes to be ignored and not pushed to the AM database. This capability is commonly used with the **from-version** and **to-version** attributes or tags, to ignore certain attributes for specific versions of Asset Manager.

The following table shows the attributes of the **<ignored-attributes>** tag:

Name	Description
<b>from-version</b>	Ignore this attribute only from (and including) this version. The version is taken from the integration point configuration.
<b>to-version</b>	Ignore this attribute only up to (and including) this version. The version is taken from the integration point configuration.

### Example:

```
<ignored-attributes>  
  <ignored-attribute>lSeq</ignored-attribute>  
</ignored-attributes>
```

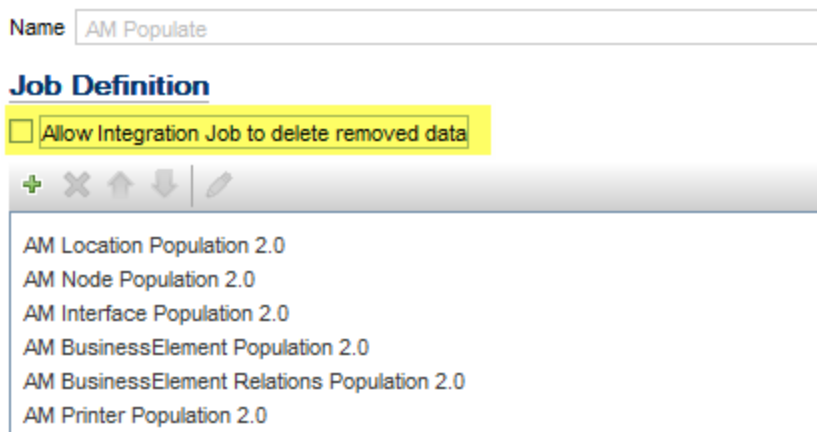
## Deletion

To allow the AM Generic Adapter to delete CIs in UCMDB on data population or delete assets in AM on data push, additional configurations are required. The following sections describes how to configure the AM Generic Adapter to allow data deletion in the AM-UCMDB integration.

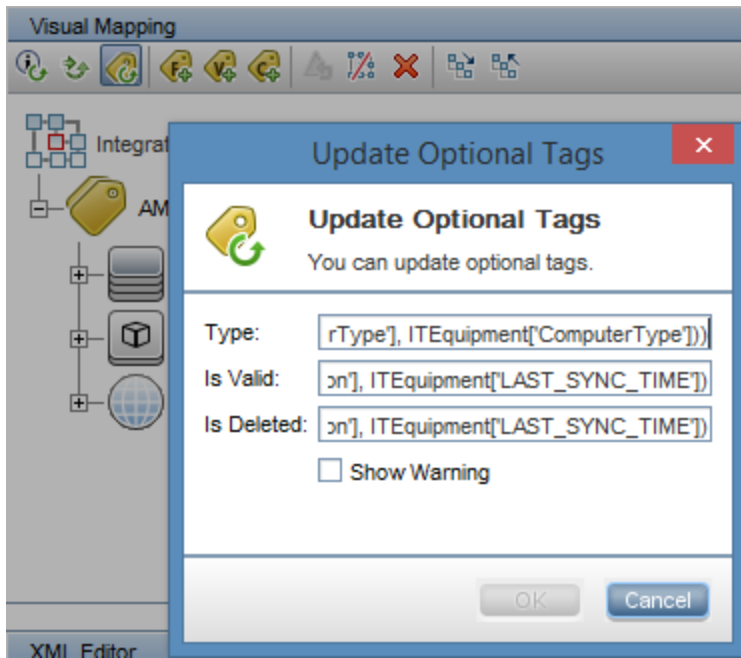
## Population Deletion Configuration

When a computer is retired in Asset Manager, the corresponding CI in UCMDB should be deleted as well. By default, it is not allowed to delete CIs from UCMDB through the AM population jobs. You need to complete the following three steps to allow population jobs to delete CIs from UCMDB.

1. In population job definition, enable Allow Integration Job to delete removed data to allow mapping script to send delete CI action to the UCMDB server.



2. In population mapping script, use the `is-deleted` attribute on target-entity to determine what CIs should be deleted. You may also configure it from the graphic mapping UI.



3. Choose the Deletion Method in the Adapter Setting.

Enable Automatic Deletion      Only on Success ▼

Automatic Deletion	
CI Type	Deletion Method
Asset	Auto Delete
BusinessElement	Auto Delete
Interface	Auto Delete
IpAddress	Auto Delete
Link	Auto Delete
Node	Auto Delete
Object	Auto Delete

## Push Deletion Configuration

When a CI is deleted from UCMDB, the corresponding asset record in AM should be deleted as well. By default, it is not allowed to delete assets from AM through the AM push jobs. You need to complete the following steps to allow push jobs to delete assets from AM.

1. In the push job definition, enable Allow Deletion for TQL queries which are allowed to delete asset records in AM.



2. Define the Action on Delete for the target AM entity type in the am-push-config.xml file. For more information, see ["Action on Delete" on page 371](#).

### Installed Software

The Integration supports the following different flows for pushing Installed Software to Asset Manager. You may switch between these flows.

**Note:** The flows below show a simplified high-level flow of the different Installed Software synchronization behavior. The actual behavior may be more complex in some cases, mainly for performance improvement. See also ["Switching between Installed Software Flows" on page 378](#).

### Normalized Installed Software

This flow uses an InventoryModel to catalog each exact Software Version. Therefore, if the AM Operator decides to map a certain Software version to a different model, he only has to do it once to the

Inventory Model, and does not have to process all the Installed Software in AM. This flow allows using either the SAI Version ID, or the attributes name, version, and vendor, to correctly reconcile the Installed Software, and uses the information to automatically create Models according to major versions as needed.

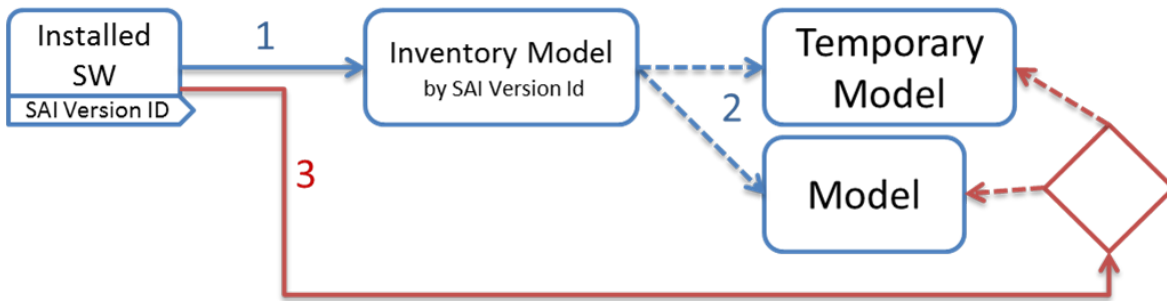


1. Each Installed Software is first mapped to an Inventory Model. If one does not exist, it creates one. The mapping is done according to the SAI Version ID which is an inventory ID from the Universal Discovery Scanner, or by using the Installed Software's name, version, and vendor.
2. It then sees if the InventoryModel has a final mapping to a Model. If it is a new InventoryModel, or the InventoryModel has no final mapping to a Model, it attempts to search for one with the same name and version. If one is found, it connects the InventoryModel to it; otherwise it creates a new Model.
3. It then connects the Installed Software to the Model as well.

**Note:** Normalized Installed Software is the default flow.

### Normalized Installed Software – No Model Creation

This flow uses an InventoryModel to catalog each exact Software Version. Therefore, if the AM Operator decides to map certain Software version to a different model, he only has to do it once to the InventoryModel, and does not have to process all the Installed Software in AM. This flow does not automatically create a Model. Instead, the Model must be connected to the InventoryModel by a different flow, or manually by an Asset Manager Operator.



1. Each Installed Software is first mapped to an Inventory Model. If one does not exist, it creates one. The mapping is done according to the SAI Version ID, which is an inventory ID from the Universal Discovery Scanner, or by using the Installed Software attributes: name, version, and vendor.
2. It then sees if the InventoryModel has a final mapping to a Model. If not, it chooses the temporary model (an Unknown Software Model).
3. It then connects the Installed Software to the Model found in the step 2.
4. Later, an Asset Manager Operator manually connects each Inventory Model to a final Model, as he wishes.

### Non-Normalized Installed Software

This flow pushes Installed Software and Models only. (It does not map or use the Inventory Models in any way).



Each Installed Software is mapped to a matching Model which is created if not found.

### Switching between Installed Software Flows

There are two OOTB TQLs for the Installed Software push in the AM Generic Adapter.

- **AM Installed Software Push 2.0** is the TQL that contains the Installed Software CIs which need to be normalized in Asset Manager. Running this TQL job triggers the installed software flow that is normalized.

- **AM Installed Software Normalized Push 2.0** is the TQL contains the Installed Software CIs which are already normalized in UCMDB and do not need to be normalized in Asset Manager. Running this TQL job triggers the installed software flow that is not normalized.

# HP Asset Manager Push Integration

This chapter includes:

This is a mini TOC level 2



## Quick Start

**Note:** This section is only for **advanced users** who want to start using Asset Manager Push Integration quickly, without reading the full documentation. It therefore provides the minimum information required before you run your first integration.

Before starting the integration for the first time, you must complete the following:

- ["Validate Pre-Loaded Data in Asset Manager" on page 384](#)
- ["Update Asset Manager Schema" on page 386](#)
- ["Create AMPushAdapterAPI Package with Required AM API Files" on page 390](#)
- ["Install a Database Client" on page 390](#)

## Overview

Integration between HP Universal CMDB (UCMDB) and HP Asset Manager enables you to share information from UCMDB with Asset Manager. Common use cases include Hardware, Installed Software and Business Services.

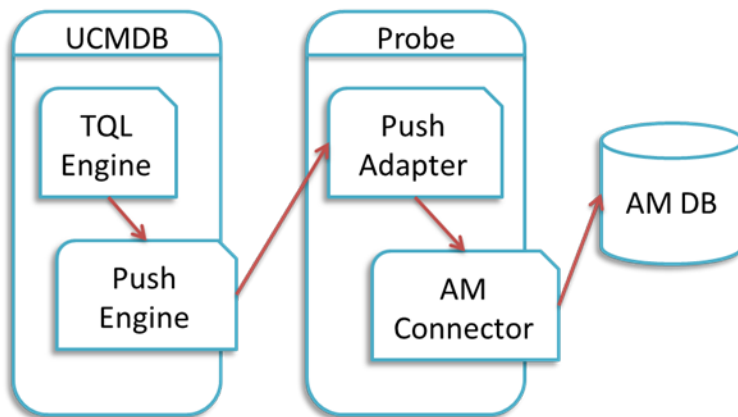
You can use the integration to automate the creation and update of Asset and Portfolio information in Asset Manager. This ensures Asset Manager is kept up to date with real, accurate, discovered data in your environment.

**Note:** This Integration replaces the Connect-It Scenarios used for syncing Hardware and Software information from DDMI 9.3x and below to Asset Manager. Also, this integration replaces the Connect-It Scenarios used to sync Business Services and Business Applications from UCMDB to Asset Manager.

### How Data is Synchronized Between UCMDB and Asset Manager

When referring to the concept of data information, it is important to distinguish between a UCMDB **CI** (Configuration Item) and an Asset Manager **Asset**. Both are defined in a different Data Model, and there must be a conversion before transferring CIs in UCMDB to Assets in Asset Manager.

The following graphic shows the high-level components of the integration:



**Note:** The Push Adapter and AM Connector are executed in the Data Flow Probe/Integration Service process.

UCMDB stores its information using CIs. The integration chooses which data to pull from UCMDB by defining integration TQL queries. Each TQL query defines a superset of data relevant for the integration.

#### The **UCMDB Push Engine:**

- Retrieves the required data from UCMDB, using the given TQL query.
- Filters the data to include only the data that has changed since the last execution of this synchronization.
- Splits the data into multiple chunks without breaking consistency.
- Sends the information to the Probe/Adapter

The **Push Adapter** is a generic framework for easily configuring push adapters, using only XML and Groovy<sup>1</sup>. It allows easy mapping of the data from the UCMDB data model into the Asset Manager Data model, and the transfer of this converted data into the AM Connector. For more information, see **Developing Push Adapters** in the *HP Universal CMDB Developer Reference Guide*.

The **AM Connector** is a component that connects to the Push Adapter, built specifically to reconcile, push, and handle the complex logic needed to synchronize data into Asset Manager.

<sup>1</sup>Groovy is an agile and dynamic language, natively supported by the Java Virtual Machine. It allows simple scripting capabilities, while maintaining all the strengths and capabilities of Java. It can execute simple String manipulation, and use 3rd party libraries. For more information, see <http://groovy.codehaus.org/>

## Supported Versions

This integration supports HP Asset Manager 9.30 and later versions.

## How to Integrate UCMDB and Asset Manager

To set up integration between UCMDB and Asset Manager, you must complete the following steps:

- ["Validate Pre-Loaded Data in Asset Manager" below](#)
- ["Set Up Asset Manager" below](#)
- ["Set Up UCMDB" on page 390](#)
- ["Push CI Data from UCMDB to Asset Manager" on page 393](#)

### Validate Pre-Loaded Data in Asset Manager

For the integration to succeed, it requires there to be some basic data already in the Asset Manager database.

This data may either be imported during the database creation (using the Asset Manager Application Designer), or may be added later. For more information, see the **Asset Manager Documentation – Administration**.

For **hardware synchronization** the required data is:

- Shared Data
- UNSPSC Product Classification
- Portfolio – Line-of-business data
- Virtualization – Line-of-business data
- Business services management – Line-of-business data

For **software synchronization** the required data is:

- Software Asset Management – Line-of-business data

### Set Up Asset Manager

To set up Asset Manager you must complete the following steps:

## Create an Account with Administrative Rights

For the integration, any user with administrative rights will suffice. Asset Manager OOTB installations include an administrator account.

The details of the default Administrator user are:

- **User:** Admin
- **Password:** <empty>

The following example shows how to create a new user (named Integration-Admin) with administration rights, specifically for the integration.

1. Log on to Asset Manager as an administrator.
2. Go to **Organization Management > User actions > Add a user.**
  - a. In **ID #**, type: **Integration-Admin.**
  - b. In **Name**, type: **Integration-Admin.**
  - c. In **First**, type: **Integration-Admin.**
  - d. Click **Next.**
  - e. Click **Next.**
  - f. Click **Finish.**
3. Go to **Organization Management > Organization > Employees.**
4. Select the newly created User and go to the **Profile** tab.
5. In **User name**, type: **Integration-Admin.**
6. In **Password**, type: **<A password you would like to use>.**
7. In the **Password Administration** pane, ensure **Never Expires** is selected.
8. In the **Profile** pane, ensure **Administration rights** is selected.
9. Click **Modify.**

## Update Asset Manager Schema

The default Asset Manager database schema includes column lengths that may be significantly shorter than their counterparts in the UCMDB database schema. For attributes used for reconciliation, this may be critical and may cause creation of multiple records.

To fix this issue, you are recommended to change the **Asset Manager Column Sizes** to the values shown in the following table of Asset Manager Attributes.

Table	Name	Default Max Length	New Value	New Value as of AM 9.31 <sup>[*]</sup>
amComputer	TcplpHostName	40	255	255
amComputer	WorkGroup	40	250	250
amPortfolio	Folder	128	250	250
amAsset	SerialNo	36	250	250
amModel	Name	80	250	250
amCompany	Name	30	100	100
amSoftInstall	Folder	128	255	255
amSoftInstall	Field1	26	255	255
amSoftInstall	TechnicalInfo	128	255	255
amBrand	Name	64	250	250
amEmplDept	UserName	100	200	200
amEmplDept	UserDomain	100	200	200
amBrand	FullName			500
amComputer	FullName			500
amModel	FullName			500

[\*] Also available in Asset Manager versions 5.22 and 9.3 with an appropriate hotfix for column length limit.

### Note:

- The new values in the table are only a suggestion, and you may need to change them according to actual data per customer use case.

- DB2 default table space page size of 4K may be too small in some cases; using 8K or higher is recommended.

## Prepare Asset Manager for Parallel Push

Enabling Parallel Push significantly increases the performance of the push. However, some advance preparation is necessary. Different actions are needed for different database types, as shown in the following table:

Database	Action
<b>DB2</b>	<i>Mandatory:</i> follow <b>Eliminating locks and deadlocks</b> in the <i>Asset Manager Tuning Guide</i> . <b>Limitation:</b> DB2 parallel push is not supported in UCMDB 10.01.
<b>Oracle</b>	<i>Optional:</i> follow <b>Eliminating locks and deadlocks</b> in the <i>Asset Manager Tuning Guide</i> .



Database	Action
<b>SQL Server</b>	<p>The following are all mandatory steps:</p> <ol style="list-style-type: none"> <li>1. Alter the <b>SQLServer Schema</b> isolation level: <p>Execute the following command on the database, replacing <b>&lt;AMSchema&gt;</b> with the real schema name:</p> <pre style="background-color: #f0f0f0; padding: 10px;">ALTER DATABASE &lt;AMSchema&gt; SET READ_COMMITTED_SNAPSHOT ON ALTER DATABASE &lt;AMSchema&gt; SET ALLOW_SNAPSHOT_ISOLATION ON GO</pre> <p><b>Note:</b> If the execution takes too long, you may need to disconnect all connections to the database. One possible way is to restart the database service, then execute the command. If you restart the SQL Server, and an Integration Point has already been created in UCMDB, you should restart the UCMDB Probe as well to avoid the issue of dead connections.</p> </li> <li>2. Alter Asset Manager database options: <ol style="list-style-type: none"> <li>a. Open the Asset Manager Client and connect to the appropriate database schema</li> <li>b. Navigate on the top menus to <b>Administration &gt; Database options</b></li> <li>c. For option <b>'Sql Server specifics'</b> <b>'Isolation command before starting a write transaction'</b> change the current value to <b>set transaction isolation level snapshot</b></li> </ol> </li> <li>3. Create a table for each of the following counters, to do this, follow the instructions in the Asset Manager <b>Tuning Guide</b>, Chapter <b>Eliminating locks and deadlocks</b>. <ul style="list-style-type: none"> <li>■ amAsset_AssetTag</li> <li>■ amBrand_BarCode</li> <li>■ amModel_BarCode</li> <li>■ amModel_ModelRef</li> <li>■ amAssignment_Code</li> <li>■ amEmplDept_BarCode</li> <li>■ amComputer_Group</li> <li>■ amComputer_Domain</li> <li>■ amComputer_Vm</li> <li>■ amComputer_MD</li> <li>■ amComputer_Name</li> <li>■ amSoftInstall_Code</li> <li>■ amMonitor_Serial</li> </ul> </li> </ol>

## Set Up UCMDB

To set up UCMDB you must complete the following steps:

### Create AMPushAdapterAPI Package with Required AM API Files

**Note:** If you want to create several integrations to **different versions of Asset Manager**, follow this procedure for each version of Asset Manager you want to integrate to. Otherwise, see "[Set Up UCMDB](#)" above.

In order for the adapter to connect to the appropriate Asset Manager version, you must supply the Data Flow Probe/Integration Service with the appropriate Asset Manager API DLLs and Jars, as follows:

1. Copy the files below:
  - **<Asset Manager Installation folder>\x64\\*.dll**
  - **<Asset Manager Installation folder>\websvc\lib\\*.jar**
2. Create a package called **AMPushAdapterAPI\_<AM Version Number>.zip**. For example, for version 9.3 the package is **AMPushAdapterAPI\_9.3.zip**.
3. Paste the copied files to:

**<AMPushAdapterAPI\_{AM Version Number}.zip>\discoveryResources\AMPushAdapter\amVersion\<AM Version Number>**

For example, for version 9.3 the path is:

**AMPushAdapterAPI\_9.3.zip\discoveryResources\AMPushAdapter\amVersion\9.3**

4. Add the subfolder **AMPushAdapter/amVersion/<AM Version Number>/\*. \*** to the **additionalClasspath** property in the **globalSettings.xml** file.
5. Deploy the **AMPushAdapterAPI\_<AM Version Number>.zip** package.


### Install a Database Client


You must install database client software according to the type of database the Asset Manager schema is installed on, as detailed in the following table:

Database	Client Software
DB2	<ol style="list-style-type: none"> <li>Download and Install "IBM Data Server Client" 64 bit for windows on your Data Flow Probe/Integration Service computer. This may be downloaded from:  <a href="http://www-01.ibm.com/support/docview.wss?rs=4020&amp;uid=swg21385217">http://www-01.ibm.com/support/docview.wss?rs=4020&amp;uid=swg21385217</a></li> <li>Create a connection to the DB2 database of Asset Manager.  You may create this using the DB2 Control Center.  <b>Note:</b> Remember the Database Alias you define in the connection, because you need it when creating the integration point.</li> <li>Copy the <b>db2cli64.dll</b> file from the DB2 client bin directory (By default: C:\Program Files\IBM\SQLLIB\BIN) to the <b>&lt;Data Flow Probe/Integration Service&gt;\lib</b> folder.</li> <li>Restart the Data Flow Probe/Integration Service.</li> </ol>
Oracle	<p><b>Oracle client windows 64 bit</b></p> <p>For example: Oracle Database 11g Release 2 Client (11.2.0.1.0) for Microsoft Windows (x64).</p> <ol style="list-style-type: none"> <li>Download the client installation from:  <a href="http://www.oracle.com/technetwork/database/enterprise-edition/downloads/index.html">http://www.oracle.com/technetwork/database/enterprise-edition/downloads/index.html</a></li> <li>Install the client, in Administrator mode, on the Data Flow Probe/Integration Service computer.</li> <li>Copy <b>oci.dll</b> from the <b>&lt;Oracle Client installation directory&gt;</b> into:  <b>&lt;Probe/Integration Service installation directory&gt;\lib.</b></li> <li>Restart the Data Flow Probe/Integration Service.</li> </ol>
SQL Server	None required.

### Create an Integration Point in UCMDB

- Log on to UCMDB as an administrator.

2. Go to **Data Flow Management > Integration Studio**. UCMDB displays a list of existing integration points.
3. Click the  button. The New Integration Point dialog box is displayed.
4. Complete the Integration and Adapter Properties fields as shown in the following table:

Field	Required	Description
Integration Name	Yes	Type the name (unique key) of the integration point.
Integration Description	No	Type a description of the current integration point.
Adapter	Yes	Select <b>HP Software Products &gt; Asset Manager &gt; Asset Manager Push Adapter</b>
Is Integration Activated	Yes	Select this option to indicate the integration point is active.
Hostname/IP	Yes	Type the hostname or IP Address of the Asset Manager database.
DB Type	Yes	Select the database type your Asset Manager schema is located on.
DB Port	Yes	Type the communication port of the Asset Manager Data Base.
DB Name/SID	Yes	<ul style="list-style-type: none"> <li>■ <b>DB2:</b> type in the name of the Database Alias you defined in the database connection.</li> <li>■ <b>Oracle:</b> type the service name.</li> <li>■ <b>SQL Server:</b> type the name of the schema.</li> </ul>
Credentials ID	Yes	Select <b>Asset Manager Protocol</b> . To create a new protocol, click the  button. Under <b>Asset Manager Protocol</b> complete: <ul style="list-style-type: none"> <li>■ <b>Asset Manager User Name:</b> An AM administrator's user name.</li> <li>■ <b>Asset Manager Password:</b> An AM administrator's password.</li> <li>■ <b>DB User Name:</b> The AM database user's name.</li> <li>■ <b>DB Password:</b> The AM database user's password.</li> </ul>

Field	Required	Description
AM Version	Yes	Select the version of Asset Manager this integration point is to connect to.
Enable Parallel Push	Yes	Select to allow parallel (multi-threaded) data push to Asset Manager. This improves performance. Note that you must configure SQL Server & DB2 to support parallel push. See " <a href="#">Prepare Asset Manager for Parallel Push</a> ".
Data Flow Probe	Yes	Select the name of the Data Flow Probe/Integration service used to execute the synchronization from.
Additional Probes	No	Select additional probes to use when pushing to AM in order to increase redundancy.
Default owner name	No	Not required for this integration point.  <b>Note:</b> This field only appears in a Multi-Tenant Enabled UCMDB.

5. Click **Test Connection** to make sure there is a valid connection.
6. Click **OK**.



The integration point is created and its details are displayed.

UCMDB creates a default data push job when creating the integration point. If needed you may create or edit the existing job. For more information, see "Work with Data Push Jobs" in the *HP Universal CMDB Data Flow Management Guide*.

## Push CI Data from UCMDB to Asset Manager

Data push jobs copy CI or CI relationship records from your UCMDB system to your Asset Manager system. To run a data push job, complete the following steps:

1. Log on to UCMDB as an administrator.
2. Go to **Data Flow Management > Integration Studio**. UCMDB displays a list of existing integration points.
3. Select the integration point you created for Asset Manager.
4. Add a new data push job as follows:

- a. Click the  button on the right panel.
- b. In the Name field, type a unique name for the job.
- c. Click the  button to add existing TQL queries to the job.

UCMDB Creates a default data push job when creating an integration point. The following table lists the Topology Query Language (TQL) queries in the default data push job. If required, you may create, update, or remove TQL queries for the push job. You may also need to update the mapping. See ["How to Customize an Existing Mapping" on page 418](#). To access these OOTB TQL queries for push, go to **Modeling > Modeling Studio > Resources**, select **Queries for Resource Type** and then go to **Root > Integration > AM Push**.

**Note:** If you want to use cluster with Asset Manager, you must install the latest version of the SAM Best Practice package in Asset Manager.

TQL Query	Description
AM Business Element Push	<p>Pushes Business Applications, Business Services and Business Infrastructure CIs.</p> <p>Mapping XML: pushMappingAMBusinessElement.xml</p>
AM Business Element Relations Push	<p>Pushes relationships between Business Elements (pushed by AM Business Element Push Query) to other business elements or to nodes.</p> <p>Business Elements must have been pushed before this TQL query synchronization in the 'AM Business Element Push' TQL query synchronization.</p> <p>Nodes must have been pushed before this TQL query synchronization in the 'AM Computer Push' TQL query synchronization.</p> <p>Mapping XML: pushMappingBeRelations.xml</p>

TQL Query	Description
<p>AM Computer Push</p>	<p>Pushes nodes (Computers, Network Devices, etc.). Also pushes IPs, Interfaces, Disk Devices, Physical Ports, Hardware Boards, Display Monitors, CPUs, Printers, Inventory Scanners, File Systems, and Assets.</p> <p>Minimal attributes for pushing a Node:</p> <ul style="list-style-type: none"> <li>○ Serial Number</li> <li>○ Vendor or Discovered Vendor</li> <li>○ Model or Discovered Model</li> <li>○ Node Role</li> </ul> <div style="border: 1px solid #ccc; background-color: #f0f0f0; padding: 5px; margin: 10px 0;"> <p><b>Note:</b> These are required values, and depend on the capability of the data source to report them.</p> </div> <p>Mapping XML: pushMappingAMCcomputer.xml</p>
<p>AM Computer Relations Push</p>	<p>Pushes relationships between Computers to any node (Computers, Network Devices, etc.).</p> <p>Nodes must have been pushed before this TQL query synchronization in the 'AM Computer Push' TQL query synchronization.</p> <p>Mapping XML: pushMappingAMComputerRelations.xml</p>
<p>AM Host Server And Running LPAR VM Relations Push</p>	<p>Pushes relationships between Host and Guest (virtualized) systems of LPAR type. Nodes must have been pushed before this TQL query synchronization in the 'AM Computer Push' TQL query synchronization.</p> <p>Mapping XML: pushMappingHostToVMLpar.xml</p>
<p>AM Host Server And Running Solaris VM Relations Push</p>	<p>Pushes relationships between Solaris Host and Guest (Virtualized) operating systems.</p> <p>Nodes must have been pushed before this TQL query synchronization in the 'AM Computer Push' TQL query synchronization.</p> <p>Mapping XML: pushMappingHostToVMSolaris.xml</p>




TQL Query	Description
AM Host Server And Running VM Relations Push	<p>Pushes relationships between Host and Guest (Virtualized) operating systems.</p> <p>Nodes must have been pushed before this TQL query synchronization in the 'AM Computer Push' TQL query synchronization.</p> <p>Mapping XML: pushMappingHostToVM.xml</p>
AM Installed Software Sync	<p>Pushes Installed Software and User_Software_Utilization CIs.</p> <p>Nodes must have been pushed before this TQL query synchronization in the 'AM Computer Push' TQL query synchronization.</p> <p>Mapping XML: pushMappingNormalizedSW.xml</p> <p>(Possible alternate mapping: pushMappingSWNonNorm.xml See "<a href="#">Installed Software</a>" on page 401.)</p>
AM Net Device Relations Push	<p>Pushes relationships between Network Devices to Network Devices.</p> <p>Nodes must have been pushed before this TQL query synchronization in the 'AM Computer Push' TQL query synchronization.</p> <p>Mapping XML: pushMappingAMNetDeviceConnections.xml</p>
AM Oracle LMS Push	<p>Pushes the Oracle Running Software and its Oracle LMS data.</p> <p>Nodes must have been pushed before this TQL query synchronization in the 'AM Computer Push' TQL query synchronization.</p> <p>Mapping XML: pushMappingOracleLMS.xml</p>
AM Software Sync Hypervisor	<p>Pushes Hypervisor Installed Software.</p> <p>Nodes must have been pushed before this TQL query synchronization in the 'AM Computer Push' TQL query synchronization.</p> <p>Mapping XML: pushMappingHypervisor.xml</p>



TQL Query	Description
AM Cluster Push	<p>Pushes Cluster CIs</p> <p>Nodes must have been pushed before this TQL query synchronization in the 'AM Computer Push' TQL query synchronization.</p> <p>Mapping XML: pushMappingCluster.xml</p>
AM Cluster Node Relations Push	<p>Pushes cluster relations between Computer Elements (pushed by AM Computer Push) and Cluster.</p> <p>Cluster Elements must have been pushed before this TQL query synchronization in the 'AM Cluster Push' TQL query synchronization.</p> <p>Nodes must have been pushed before this TQL query synchronization in the 'AM Computer Push' TQL query synchronization.</p> <p>Mapping XML: pushMappingClusterNodeRelations.xml</p>
AM Cluster Runningsoftware Push	<p>Pushes Cluster Runningsoftware CIs</p> <p>Nodes must have been pushed before this TQL query synchronization in the 'AM Cluster Push' TQL query synchronization.</p> <p>Mapping XML: pushMappingClusterRunningsoftware.xml</p>
AM Cluster Runningsoftware Relations Push	<p>Pushes cluster runningsoftware relations between Computer Elements (pushed by AM Computer Push) and Cluster.</p> <p>Cluster Runningsoftware Elements must have been pushed before this TQL query synchronization in the 'AM Cluster Runningsoftware Push' TQL query synchronization.</p> <p>Nodes must have been pushed before this TQL query synchronization in the 'AM Computer Push' TQL query synchronization.</p> <p>Mapping XML: pushMappingClusterRunningsoftwareRelations.xml</p>

- d. Select or unselect the **Allow Deletion** option for each query. (The setting determines if this TQL query is allowed to delete data from Asset Manager, though the actual action on delete is defined by CI type in the mapping xml.)

**Note:** For scheduling configuration, see ["How to Schedule Data Push Jobs" on page 399.](#)

- e. Click **OK**.
  - f. Save the integration point.
5. Run the job manually to see if the integration job works properly:
    - a. To push all the relevant data for the job, click the  button.
    - b. To push only the changes in the data since the job last executed, click the  button.
  6. Wait for the job to complete; click the  button multiple times as needed until the job is completed.
  7. When the job is completed, the job status becomes one of the following depending on the results:
    - Succeeded
    - Completed
    - Failed
  8. Click the **Statistics** tab to view the results; if any errors occur, click the **Query Status** tab and **Job Errors** tab for more information. For more information about errors, see "[Troubleshooting and Limitations](#)".

**Note:** For details about these tabs and managing the integration, see **Integration Jobs Pane** in the *HP Universal CMDB Data Flow Management Guide*.

If the job completes successfully, you can view the UCMDB CI data in Asset Manager.

## How to View UCMDB Data in Asset Manager

After a push job is successfully completed, you can search for and verify that the pushed CI/relationship data is in Asset Manager.

### Nodes

This includes computers, network devices, etc.

To view:

1. Log on to Asset Manager as an administrator.
2. Go to **Portfolio Management > Asset Configurations > IT equipment > Computers and virtual machines**.
3. In the opened dialog box, for **IP name**, type the name of the computer you are searching for.
4. You may use '<name prefix>%' for easier searching. For example, searching IP name for **mycomp%** returns computer mycomp1, mycomp2, etc.
5. Browse the computer for the different data.

## Business Elements


This includes Business Applications, Business Services and Business Infrastructures.

1. Log on to Asset Manager as an administrator.
2. Go to **Asset Lifecycle > IT Services and virtualization > Business services > Business services**.
3. Browse the different Services.

For more information about viewing data in Asset Manager, see the Asset Manager Documentation.

## How to Schedule Data Push Jobs

UCMDB allows you to schedule job executions directly from a data push job.

1. Log on to UCMDB as an administrator.
2. Go to **Data Flow Management > Integration Studio**. UCMDB displays a list of existing integration points.
3. Select the integration point you created for the UCMDB - AM integration.
4. Select the push job.
5. Click the  button.

**Note:** UCMDB allows you to define two different schedules for two types of data push:

**Changes Synchronization** and **All Data Synchronization**. It is recommended to use the Changes Sync schedule to only synchronize changes and avoid synchronizing the entire set of data each time.

6. Define a schedule for Changes Sync.
  - a. Click on the **Changes Synchronization** tab.
  - b. Select the **Scheduler enabled** option.
  - c. Select the scheduling options you want to use.
7. Click the **All Data Synchronization** tab and select the scheduling options you want to use.
8. Click **OK**.
9. Save the integration point.

## Installed Software

The Integration supports the following different flows for pushing Installed Software to Asset Manager. You may switch between these flows.

**Note:** The flows below show a simplified high-level flow of the different Installed Software synchronization behavior. The actual behavior may be more complex in some cases, mainly for performance improvement. See also "[Switching between Installed Software Flows](#)" on page 403.

### Normalized Installed Software

This flow uses an InventoryModel to catalog each exact Software Version. Therefore, if the AM Operator decides to map a certain Software version to a different model, he only has to do it once to the Inventory Model, and does not have to process all the Installed Software in AM. This flow allows using either the SAI Version ID, or the attributes name, version, and vendor, to correctly reconcile the Installed Software, and uses the information to automatically create Models according to major versions as needed.

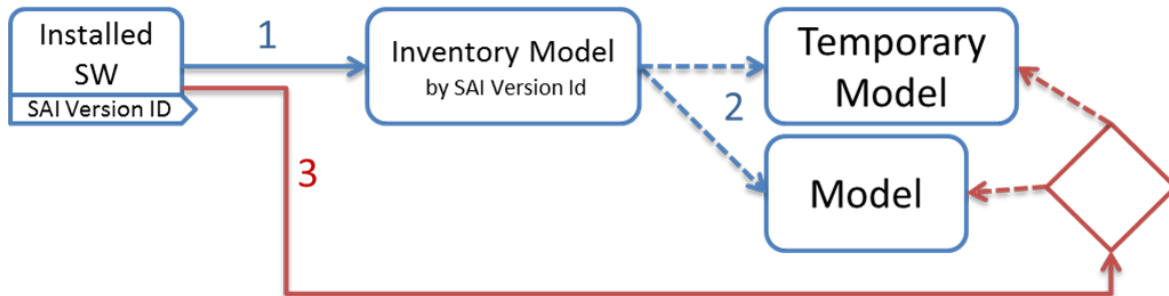


1. Each Installed Software is first mapped to an Inventory Model. If one does not exist, it creates one. The mapping is done according to the SAI Version ID which is an inventory ID from the Universal Discovery Scanner, or by using the Installed Software's name, version, and vendor.
2. It then sees if the InventoryModel has a final mapping to a Model. If it is a new InventoryModel, or the InventoryModel has no final mapping to a Model, it attempts to search for one with the same name and version. If one is found, it connects the InventoryModel to it; otherwise it creates a new Model.
3. It then connects the Installed Software to the Model as well.

**Note:** Normalized Installed Software is the default flow.

### Normalized Installed Software – No Model Creation

This flow uses an InventoryModel to catalog each exact Software Version. Therefore, if the AM Operator decides to map certain Software version to a different model, he only has to do it once to the InventoryModel, and does not have to process all the Installed Software in AM. This flow does not automatically create a Model. Instead, the Model must be connected to the InventoryModel by a different flow, or manually by an Asset Manager Operator.



1. Each Installed Software is first mapped to an Inventory Model. If one does not exist, it creates one. The mapping is done according to the SAI Version ID, which is an inventory ID from the Universal Discovery Scanner, or by using the Installed Software attributes: name, version, and vendor.
2. It then sees if the InventoryModel has a final mapping to a Model. If not, it chooses the temporary model (an Unknown Software Model).
3. It then connects the Installed Software to the Model found in the step 2.
4. Later, an Asset Manager Operator manually connects each Inventory Model to a final Model, as he wishes.

### Non-Normalized Installed Software

This flow pushes Installed Software and Models only. (It does not map or use the Inventory Models in any way).



Each Installed Software is mapped to a matching Model which is created if not found.

## Switching between Installed Software Flows

### Switching to Normalized Installed Software Flow

This is the default flow, enabled OOTB for this integration. If the flow was changed and you would like to return to this flow:

- Change the Installed Software push TQL query name (original name: AM Software Push) to **AM Installed Software Push**.

### Switching to Normalized Installed Software – No Model Creation

In Adapter Management, edit **am-push-mapping.xml**:

1. Go to **ci-type="Complete\_amModel"**
2. Change **operation-type** to **optional\_reference**

### Switching to Non-Normalized Installed Software

Change the Installed Software push TQL query name (original name: AM Installed Software Push) name to: **AM Software Non Norm**.

## How to Tailor the Integration

This section includes:

- ["Integration Architecture" on the next page](#)
- ["How to Change Adapter Settings" on page 417](#)
- ["How to Customize an Existing Mapping" on page 418](#)
- ["How to Add a New Mapping to the Integration" on page 420](#)

**Note:** For more detailed information about customizing the mapping, see "Developing Enhanced Generic Push Adapters" in the *HP Universal CMDB Developer Reference Guide*.

## Integration Architecture

This section contains details about the architecture of the integration.

- ["Data Flow Architecture" below](#)
- ["Integration TQL Queries" on the next page](#)
- ["Reconciliation Proposals" on the next page](#)
- ["Asset Manager Rules and Flows" on page 406](#)
- ["Data Mapping" on page 406](#)
- ["Push Mapping" on page 407](#)

### Data Flow Architecture

1. The Push Engine executes the TQL query.
2. For a differential flow, the data is compared to the last synchronized data, and only the changes are forwarded.
3. Data is converted into Composite CIs (instances of data according to the TQL Root elements).
4. Data is then pushed to the Push Adapter.
5. The Push Adapter loads the correct mapping for the specific TQL query.
6. All **dynamic\_mappings** are executed and saved to maps, to allow usage in the next mapping stage.

For more information, see "Developing Enhanced Generic Push Adapters" in the *HP Universal CMDB Developer Reference Guide*.

7. Data is mapped from the UCMDB data Model into the AM Data model according to the mapping XML.
8. Data is sent to the AM Connector.
9. AM Connector orders all the data in a set of dependency trees, starting with the records that do not depend on any other record.



10. AM Connector attempts to merge any duplicate records
11. AM Connector starts reconciling and pushing any record without any dependencies, or a record whose dependencies have already been reconciled/pushed to Asset Manager.
  - a. AM Connector first tries to reconcile with existing records
  - b. If it finds a match, it attempts to update that record,
  - c. If it does not find a match, it attempts to create a new record.
12. AM Connector deletes any records it is required to delete in AM, as permitted by action-on-delete.

## Integration TQL Queries

A TQL query used for the integration must contain a root query node.

Any attribute using in the mapping flow of the Push Adapter must be marked in the selected layout of the query node.

Each TQL query may only have one mapping.

For more information, see **Data Flow Management > Integration > Integration Studio > Integration Jobs Pane**.

## Reconciliation Proposals

When pushing data to Asset Manager, there is an option to create a reconciliation proposal (RP). A reconciliation proposal should be created if there is a change in a specific attribute that may need AM Operator validation or action to support AM business processes.

The OOTB configuration creates a reconciliation proposal record when the memory size of the pushed computer has decreased compared to the AM computer.

### How to use Reconciliation Proposals

In the OOTB configuration **IMemorySizeMb** is marked for attribute-reconciliation. The update script calls the **updateMemorySize** function. This function verifies if the memory size of the computer was decreased. It initializes all the parameters that are passed to the function **validateReconcUpdateAdvance**. Calls **validateReconcUpdateAdvance** and return its returned value.

**validateReconcUpdateAdvance** is a function that returns the value that should be set to the attribute according to the Reconciliation Proposal status. The following table describes its parameters:

Parameter	Description
<b>AMApiWrapper</b>	The wrapper that is used to communicate with the AM.
<b>newVal</b>	The value of the attribute in the pushed data.
<b>oldVal</b>	The value of the attribute that is retrieved from AM.
<b>recordId</b>	The primary key of the table that the attribute belongs to.
<b>strCode</b>	The prefix of the <b>code</b> field in the reconciliation proposal.
<b>strName</b>	The name of the reconciliation proposal.
<b>path</b>	The name of the attribute.
<b>recordTable</b>	The table that the attribute belongs to.

**validateReconcUpdateAdvance** returns the value that should be set for the attribute, according to the Reconciliation Proposal status.

In order to create a reconciliation proposal flow on a different attribute, the following steps must be completed:

1. Add the **<attribute-reconciliation>** tag for this attribute.
2. The update-script should call a new function that initializes the parameters passed to **validateReconcUpdateAdvance**, and returns the value returned from **validateReconcUpdateAdvance**.

**Note:** It is suggested to use the **updateMemorySize** function as a reference.

## Asset Manager Rules and Flows

Asset Manager has its own set of rules and flows that are enforced by the Asset Manager API. Some customizations may need to later these rules and flows as well. For more information, see the Asset Manager documentation.

## Data Mapping

For details, see **Developing Push Adapters** in the *HP Universal CMDB Developer Reference Guide*.

## Push Mapping

This section includes tables explaining each XML tag and the attributes available for configuration.

- ["Basic Information" on the next page](#)
- ["Reconciliation" on page 410](#)
- ["Target CI Validation" on page 412](#)
- ["Reference Attribute" on page 413](#)
- ["Attribute Reconciliation" on page 414](#)
- ["Action on Delete" on page 415](#)
- ["Ignored Attributes" on page 417](#)

## Basic Information

Attributes in the **<am-mapping>** tag.

Attribute	Description
<b>ci-type</b>	Name used in push adapter mapping to recognize this record type.
<b>primary-key</b>	The primary key column in AM database schema.
<b>operation-type</b>	Defines what operations may be done with the record: <ul style="list-style-type: none"><li>• <b>insert</b> Only allows creation of new records; if it already exists, an exception is thrown.</li><li>• <b>update</b> - Only allows updates of an existing record; if it does not exist, an exception is thrown.</li><li>• <b>update_else_insert</b> - If the record exists it is updated, otherwise the record is created.</li><li>• <b>reference-only</b> - The record is only used for referencing by other record (and is not updated). An exception is thrown if the record does not exist.</li><li>• <b>ignore</b> - The record is unaffected by operations.</li><li>• <b>insert_else_reference</b> - If the record does not exist, it is created. Otherwise it is only used as a reference and is not be updated; see <b>reference-only</b>.</li></ul>
<b>parallel-push-allowed</b>	If enabled with the <i>enabled-parallel-push</i> configuration of the integration point, will attempt to push to the entity with multiple threads in order to increase performance.
<b>merge-allowed</b>	If enabled and this entity is an exact duplicate of another entity in the chunk, it merges both entities into one and fixes any relevant references.
<b>errorcode-override</b>	If used together with the adapter specific errors, allows the printing of a customized error message to the UI if the push or reconciliation of this entity fails.
<b>target-ci-type</b>	Real name of the AM database table to push this record to. If missing , it uses ci-type instead.

<b>Attribute</b>	<b>Description</b>
<b>from-version</b>	Use this mapping only from (and including) this version. The version is taken from the integration point configuration.
<b>to-version</b>	Use this mapping only up to (and including) this version. The version is taken from the integration point configuration.

## Reconciliation

Reconciliation defined for each mapping may include more than one set of reconciliation rules. When attempting to reconcile the record with existing ones in the AM database, the AM Connector tries each of the reconciliation sets until it finds a matching record. Priority is defined by order of reconciliation rules. If no record in the AM database matches this record, an insert operation occurs if the operation type permits it.

Name	Type	Description
<b>reconciliation</b>	Tag	Parent XML tag for all reconciliation configuration.
<b>reconciliation-keys</b>	Tag	Represents a single reconciliation rule that may be made of one or more attributes. All attributes inside the rule must match in order for the reconciliation of this rule to be successful.
<b>reconciliation-key</b>	Tag	Represents a single attribute used for reconciliation as part of the reconciliation-keys rule.
<b>Ignore-case</b>	Attribute	Part of the reconciliation-key tag. Specifies that this attribute comparison ignores case.
<b>reconciliation-advanced</b>	Tag	Allows definition of the reconciliation rule by manually defining the WHERE clause of the AQL (Asset Query Language). Uses GString (Groovy String) to generate the replacement String. Any variable or property defined in this record or its parent during the mapping stage (in the Push Adapter) may be used as a variable in the GString. (See <a href="http://groovy.codehaus.org/Strings+and+GString">http://groovy.codehaus.org/Strings+and+GString</a> for more information).  <b>Note:</b> AMPushAdvancesReconciliationException may be thrown inside this tag to skip to the next rule.
<b>follow-parent</b>	Tag	Used for defining overflow tables. See the AM documentation for more information on overflow tables. When using follow-parent, no other reconciliation may be used as this target CI has a 1:1 connection with its parent, and it uses the parent reconciliation to push data to AM.
<b>am-prefix</b>	Attribute	Part of the follow-parent tag. Defines the name that the parent target CI uses to reference to this table. (To find out the correct value, navigate to <b>AM Application Designer &gt; Edit Links.</b> )

**Example:**

```
<reconciliation>
  <reconciliation-advanced>Portfolio.CMDBId = '${if(globalId==null) { throw new
com.hp.ucmdb.adapters.ampush.exception.
AMPushAdvancesReconciliationException
('Not enough reconciliation data') }else{ return globalId}}'
  </reconciliation-advanced>
  <reconciliation-keys>
    <reconciliation-key ignore-case="true">AssetTag</reconciliation-key>
  </reconciliation-keys><reconciliation-keys>
    <reconciliation-key>TcpIpHostName</reconciliation-key>
    <reconciliation-key>Workgroup</reconciliation-key>
  </reconciliation-keys>
</reconciliation>
```

## Target CI Validation

This tag allows the definition of a validation rule that is executed on specific attribute values: the new one held in memory, and the old one stored in the AM database.

The following table shows the attributes of the **<target-ci-validation>** tag:

Name	Description
<b>attribute-name</b>	The attribute that you want to use for validation.
<b>validation-script</b>	A Groovy based script that returns <b>true</b> if this record is to be pushed, and <b>false</b> if it is not to be pushed. The script may access any external Groovy code in the path in order to run the evaluation. <ul style="list-style-type: none"><li>• <b>vNewVal</b> - Attribute value of the record in memory.</li><li>• <b>vOldVal</b> - Attribute value of the record in the AM database.</li></ul>
<b>failed-validation-error-code</b>	This optional attribute holds the error code that appears if there is a validation failure. The arguments that can be referenced in the error message are: <ul style="list-style-type: none"><li>• {0} - The validated attribute name.</li><li>• {1} - The property value in UCMDB.</li><li>• {2} - The property value in Asset Manager.</li><li>• {3} - The validation script.</li><li>• {4} - The additional message from the <b>additional-failure-message</b> attribute, or 'null' if there is no additional message.</li></ul>
<b>additional-failure-message</b>	This optional attribute holds an additional error message that can be referenced by the error message in the <b>properties.error</b> file. See <b>failed-validation-error-code</b> , above.

### Example:

```
<target-ci-validation attribute-name="dtLastScan" validation-script="mappings.scripts.AMReconciliationAdvanced.isDateAfter(vNewVal,vOldVal)"/>
```



## Reference Attribute

A reference attribute defines a column that references another record from a different or same table. This record is not pushed, or reconciled against existing AM database records, until this reference is resolved. Resolved references are replaced by a reference ID that represents the primary ID of the referenced record.

The following table shows the attributes of the **<reference-attribute>** tag:

Name	Description
<b>ci-name</b>	The CI-type of the referenced record.
<b>datatype</b>	The value type of the record.
<b>name</b>	The column in the current record that is to be populated by the reference ID.
<b>reference-direction</b>	According to the tree created by the Push Adapter, the value specifies if the referenced record is a parent or child of the current record.

**Example:**

```
<reference-attribute ci-name="SW_amModel" datatype="STRING"
name="lModelId"reference-direction="child"/>
```

## Attribute Reconciliation

This tag allows the AM connector to decide what to do with an attribute value according to the existing value in the AM database.

The following table shows the attributes of the **<attribute-reconciliation>** tag:

Name	Description
<b>attribute-name</b>	The attribute to be reconciled.
<b>update-script</b>	The script to execute in case of an update operation on the record. The returned value by the groovy script will be push to AM as the value of this attribute. <ul style="list-style-type: none"><li>• <b>vNewVal</b> - Attribute value of the record in memory.</li><li>• <b>vOldVal</b> - Attribute value of the record in the AM database.</li></ul>
<b>Insert-script</b>	The script to execute in case of an insert operation on the record. The value returned by the Groovy script is pushed to AM as the value of this attribute. <b>vNewVal</b> - Attribute value of the record in memory.

### Example:

```
<attribute-reconciliation attribute-name="AssetTag" update-script="mappings.scripts.AMPushFunctions.fIsEmpty(vOldVal) ? vNewVal : vOldVal"/>
```

## Action on Delete

This tag allows customization of the behavior on receipt of a delete notification for a record.

**Note:** No deletion occurs if the **Allow Delete** option in the job definition is disabled.

The following actions are possible

- **<ignore>** - Do nothing.
- **<delete-ci>** - Delete this record from the AM database.
- **<set-attribute-value>** - Change the value of one or more attributes in the AM database.

**Example:**

```
<action-on-delete>  
  <set-attribute-value name="bMissing" datatype="BOOLEAN" value="1"/>  
</action-on-delete>
```

## Enum Attribute

This tag allows a specific enum attribute to be pushed in a serial mode, when the adapter is configured to push data in parallel mode.

**Note:** This option exists to prevent duplicate key exceptions occurring when several threads push the same enum value.

The following table shows the attributes of the **<enum-attribute>** tag:

Name	Description
<b>attribute-name</b>	The enum attribute name.
<b>itemized-name</b>	The itemized list format (amOS) of the enum.

## Ignored Attributes

This tag allows specific attributes to be ignored and not pushed to the AM database. This capability is commonly used with the **from-version** and **to-version** attributes or tags, to ignore certain attributes for specific versions of Asset Manager.

The following table shows the attributes of the **<ignored-attributes>** tag:

Name	Description
<b>from-version</b>	Ignore this attribute only from (and including) this version. The version is taken from the integration point configuration.
<b>to-version</b>	Ignore this attribute only up to (and including) this version. The version is taken from the integration point configuration.

### Example:

```
<ignored-attributes>  
  <ignored-attribute>lSeq</ignored-attribute>  
</ignored-attributes>
```

## How to Change Adapter Settings

1. Go to **Data Flow Management > Adapter Management > AMPushAdapter > Adapters**.
2. Right-click **AMPushAdapter** and click **Edit Adapter Source**.
3. Scroll down to the **<adapter-settings>** tag.
4. Edit the settings as required and click the Save button.

The following table shows the Settings relevant to the AM Push Adapter:

Setting	Default	Description
<b>replication.chunk.size</b>	4,000	Defines the requested number of CIs and Relationships sent in each chunk. It is possible to adjust this setting to try and improve performance of the server and the Probe.
<b>replication.chunk.root.limit</b>	850	Defines the maximum requested number of Roots sent in each chunk. This works with <b>replication.chunk.size</b> as a limiter on the amount of data sent in each chunk.
<b>PushConnector.class.name</b>		The name of the Java class used to load the AM Connector.
<b>parallel.thread.pool.size</b>	8	The amount of threads used in Parallel Push Mode. The more threads, the more CPU used. Increasing the pool size may lower performance.
<b>mapping.size.fuse</b>	100,000	The maximum number of records the AM Connector may to retrieve in <dynamic_mapping> .
<b>transaction.deadlock.max.retry.count</b>	3	The maximum number of retries the AM Connector attempts to push a deadlocked database transaction.

## How to Customize an Existing Mapping

This example shows you how to add the **BarCode/RFID** attribute to the integration, including the TQL query, Push Adapter Mapping and AM Mapping configuration. It allows the integration to both push the BarCode/RFID attribute to Asset Manager, as well as use it for reconciliation of a Hardware Asset.

After completing the following steps, you may run the job with the customized mapping:

- 1. Add the BarcodeOrRfidTag attribute to the AM Computer Push TQL query layout.**

In this step we add the attribute of the Asset CI to the integration TQL query so that we can use the attribute and value in the mapping.

- a. Go to **Modeling > Modeling Studio > Resources** and select the **Queries** Resource Type.
- b. Go to **Queries: Root > Integration > AM Push > AM Computer Push**.
- c. Select **Asset**, right-click and select **Query Node Properties**.
- d. Go to the **Element Layout** tab.
- e. Move the **BarcodeOrRfidTag** to the **Specific Attributes** box.
- f. Click **OK**.
- g. Save the Query.

## 2. Add the BarCode Mapping to the pushMappingAMComputer.xml push adapter mapping.

In this step we take the value from the TQL result and remodel it to the Asset Manager Data Model.

- a. Go to **Data Flow Management > Adapter Management > Packages > AMPushAdapter > Configuration Files > pushMapingAMComputer.xml**.
- b. Go to the **<target\_ci\_type name="amComputer">** XML tag.
- c. Add the variable to hold the value of the barcode:

```
<variable name="vBarCode" datatype="STRING" value="Root.Asset[0]['barcode_or_rfid_tag']/>
```

- d. Go to the **<target\_ci\_type name="amAsset">** XML tag.
- e. Below the tag, add the following XML tag:

```
<target_mapping name="BarCode" datatype="STRING" value="vBarCode"/>
```

- f. Click **OK**.

## 3. Add the BarCode to the am-push-mapping.xml configuration.

**Note:** This example is specific to Asset Manager version 9.3 and above.

- a. Go to **Data Flow Management > Adapter Management > Packages > AMPushAdapter > Configuration Files > am-push-mapping.xml**.

- b. Go to the **<am-mapping ci-type="amComputer" from-version="9.3">** XML tag.
- c. Add the following rule to the advanced reconciliation rule:

```
<reconciliation-advanced> Portfolio.Asset.BarCode = '${if(vBarCode==null)
{ throw new
com.hp.ucmdb.adapters.ampush.exception.AMPushAdvancesReconciliationExcep
tion
('Not enough reconciliation data')}else{ return vBarCode}}
'</reconciliation-advanced>
```

This rule retrieves the computer whose asset has the same barCode value as vBarCode. If the VBarCode is null, an exception is thrown stating that there is insufficient reconciliation data. The exception thrown orders the AM Connector to try the next available reconciliation rule.

- d. To avoid overwriting an existing value, add the following to **<am-mapping ci-type="amAsset">**:

```
<attribute-reconciliation attribute-name="BarCode" update-
script="mappings.scripts.AMPushFunctions.fIsEmpty(vOldVal) ? vNewVal :
vOldVal"/>
```

- e. Click **OK**.

## How to Add a New Mapping to the Integration

This example shows how to add a new TQL query, push-mapping, and am-mapping to the integration. It also shows how to push Locations from UCMDB to Asset Manager. It consists of the following steps:


### Step 1: Create a TQL Query

1. Go to **Modeling > Modeling Studio > New > Query**.
2. From the **CI Types** tab, add a **Location CIT** to the query.
3. Right-click the **Location Query Node** and select **Query Node Properties**.
4. Rename the **Element Name** to **Root**.
5. Go to the **Element Layout** tab.



6. Select **Select attributes for layout**.
7. In the **Attributes condition** drop down, select **Specific Attributes**, and add the following attributes:
  - a. **Name**
  - b. **LocationType**
  - c. **LocationBarCode**
8. Click **OK**.
9. Save the query to **Root > Integration > AM Push > AM Location Push**.

## Step 2: Create a Push-Mapping

1. Go to **Data Flow Management > Adapter Management > AMPushAdapter**.
2. Click the  button and select **New Configuration File**.
3. Type the following Name: **AMPushAdapter/mappings/hw/pushMappingAMLocation.xml**.
4. Select the **AMPushAdapter** package.
5. Click **OK**.
6. Copy the following into the newly created XML file:

```
<integration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="/am-push-adapter/src/
main/resources/schema/am-push-adapter.xsd">
  <info>
    <source name="UCMDB" versions="10.0" vendor="HP"/>
    <target name="AssetManager" versions="9.3" vendor="HP"/>
  </info>
  <import>
    <!--Allows the XMLs to use the AMPushFunctions groovy      methods-->
    <scriptFile path="mappings.scripts.AMPushFunctions"/>
  </import>
  <targetcis>
    <!--Query: Location(Root) -->
    <source_instance_type query-name="AM Location Push"      root-element-
name="Root" >
    <!-- Single_amLocation is the name we will refer to this CI      in the
```

```
am-mapping.xml-->
  <target_ci_type name="Single_amLocation">
    <!--Retrieve the LocationBarCode from the Location CI          and
place it in the amLocation record-->
    <!--'name' is the column name in Asset Manager's          table -->
    <!--'datatype' is the type of the data of this column -->
    <!--'value' the script to execute in order to retrieve          the
value of this target_mapping-->
    <target_mapping name="BarCode" datatype="STRING"          value="Root
['location_bar_code']"/>
    <target_mapping name="LocationType" datatype="STRING"
value="Root['location_type']"/>
    <target_mapping name="Name" datatype="STRING"          value="Root
['name']"/>
    <!--Marks the location as created on the fly (So we can distinguish from AM
UI created Locations)-->
    <target_mapping name="bCreatedOnTheFly"          datatype="BOOLEAN"
value="1"/>
  </target_ci_type>
</source_instance_type>
</targetcis>
</integration>
```

- 7. Click **OK**.

### Step 3: Create an AM-Mapping



- 1. Go to **Data Flow Management >Adapter Management > AMPushAdapter**.
- 2. Click **am-push-mapping.xml**.
- 3. Insert the new mapping:

```
<!-- 'ci-type' is the name of the ci as appear in the          mapping file.
'target-ci-type' the actual table name in Asset Manager
this target ci is mapped to -->
<am-mapping ci-type="Single_amLocation" primary-key="lLocaId"
target-ci-type="amLocation" operation-type="update_else_insert"
parallel-push-allowed="true" from-version="9.3">
  <reconciliation>
    <reconciliation-keys>
      <!-- The reconciliation of 'location' will be done by the
two attributes: Name, LocationType.
Those are the two identifiers of Location CIT in UCMDB-->
      <reconciliation-key>Name</reconciliation-key>
```



```
<reconciliation-key>LocationType</reconciliation-key>
</reconciliation-keys>
</reconciliation>
<!-- In case the CI has been deleted from the UCMDB,
it won't effect the amLocation in Asset Manager -->
<action-on-delete>
  <ignore/>
</action-on-delete>
</am-mapping>
```

4. Click **OK**.

#### Step 4: Create a Job with the New TQL Query

1. Go to **Data Flow Management > Integration Studio**.
2. Create an integration point with Asset Manager:
  - a. In the **Integration Jobs** tab, click the  button.
  - b. Insert a job name in the **Name** field.
  - c. Click the  button, and choose the **AM Location Push** query.
  - d. Click **OK**.

#### Step 5: Run the Job

1. Click on the job created in "[Step 4: Create a Job with the New TQL Query](#)".
2. Click the  button.
3. Wait for the job to finish. You should click the  button to see progress.
4. Make sure that the status is **Succeeded**.

#### Step 6: View the Results

1. Go to the **Asset Manager Client**.

2. Go to **Organization Management > Organization > Location**.
3. Validate that the Location pushed in the previous steps is displayed.

## Frequently Asked Questions

### What is a Root CI node?

A Root node is a TQL node that represents the CI type that is created via the push to Asset Manager from the TQL Structures. Usually the rest of the TQL structure contains information that can be incorporated within the Root CI type and is used to enrich the record in Asset Manager. The Root is the heart of the Composite CI (or Instance), and if it is deleted from UCMDB we send a delete notification to Asset Manager for the entire record.

### When is a new Asset created in Asset Manager?

In the out of the box integration we create Assets for 4 types of UCMDB CIs:

- **Nodes**
- **Business Elements**
- **Printers**
- **Display Monitors**

Whenever UCMDB sends a CI of one of these types, the integration first tries to detect if this Asset already exists in Asset Manager, using the defined reconciliation rules. If a matching Asset is found, it is updated, otherwise a new Asset is created.

### How do I control the action taken when a CI is deleted in UCMDB?

See ["Action on Delete" on page 415](#).

### I deleted a Node CI in UCMDB - Why isn't it deleted in Asset Manager?

The default Action on Delete for nodes in the integration is to do nothing.

You may either change the action to delete the Asset in Asset Manager by changing the `<action-on-delete>` xml mapping in the `am-push-mapping.xml`:

```
<am-mapping ci-type="amComputer" ...>  
  ...
```

```
...
  <action-on-delete>
    <delete-ci/>
  </action-on-delete>
</am-mapping>
```

Or you may change the action to set the Asset as Missing in Asset Manager by changing the `<action-on-delete>` xml mapping in the `am-push-mapping.xml`:

```
<am-mapping ci-type="amComputer" ...>
  ...
  ...
  <action-on-delete>
    <set-attribute-value name="Portfolio.seAssignment" datatype="INTEGER"
value="6"/>
  </action-on-delete>
</am-mapping>
```

Validate that the **Allow Delete** check box in the job configuration is selected.

### **I deleted an Installed Software CI in UCMDB - Why isn't it deleted in Asset Manager?**

The default Action on delete for Installed Software in the integration is to mark it as missing.

You may change the action to delete the Soft Installed in Asset Manager, by changing the `<action-on-delete>` xml mapping in the `am-push-mapping.xml`:

```
<am-mapping ci-type="Complete_amSoftInstall" ...>
  ...
  ...
  <action-on-delete>
    <delete-ci/>
  </action-on-delete>
</am-mapping>
```

Due to the complexity and amount of flows available for Installed Software you must also change these mappings to match:

- SW\_amSoftInstall
- Complete\_amSoftInstall\_User
- soft\_Hyper\_amSoftInstall
- SW\_amSoftInstall\_User

## Is it possible to avoid overwriting an attribute in Asset Manager?

Yes. By using the Attribute Reconciliation feature, you choose to never overwrite an existing value.

### Example:

```
<attribute-reconciliation attribute-name="AssetTag" update-script=
"mappings.scripts.AMPushFunctions.fIsEmpty(vOldVal) ? vNewVal : vOldVal"/>
```

See ["Integration Architecture" on page 404](#).

## What is the different between the mapping XMLs and the am-push-mapping.xml?

The Mapping XMLs (for example: pushMappingAMBusinessElement.xml) define the way we convert the data from the UCMDDB data model into the Asset Manager Data Model and are executed by the Push Adapter.

For more information, see **Developing Push Adapters** in the *HP Universal CMDB Developer Reference Guide*.

The **am-push-mapping.xml** is the Asset Manager Connector configuration file. It configures the way we reconcile and handle the data, before we update Asset Manager with the record.

## Should I select the 'Enable Parallel' Feature?

Asset Manager configured over an Oracle database supports parallel push out of the box.

Asset Manager configured over an SQL Server database, needs some tuning before enabling this capability. See ["Prepare Asset Manager for Parallel Push" on page 388](#).

## Why does an integration point that synchronizes only the AM Installed Software Push TQL query, keep failing?

The AM Installed Software Push TQL query contains both a query node of Installed Software and a query node of Node. The Node in this mapping is only referenced, and is mapped to Asset Manager by saving its Asset Manager ID from an earlier run. Before pushing this TQL query, the same integration point must push the Node to Asset Manager (using the AM Computer Push TQL).

## How can I push nodes without Model Name or Serial Number information, to Asset Manager?

To avoid pushing nodes not yet fully discovered, we avoid sending ones without a Model Name or Serial Number that provide us with a physical identification of the Asset. If you would like to push these nodes

as well, simply remove the appropriate condition of the node from the **AM Computer Push** TQL query.

However removing the **Node Role** condition (filtering nodes without a Node Role) from the TQL query is not recommended, as the integration will not know what type of an Asset/Portfolio to create in Asset Manager.

## Troubleshooting and Limitations

- **Limitation:** A single probe may only connect to one version of Asset Manager. (Use of multiple instances of the same version is supported). This is due to the JVM limitation of loading only one Asset Manager API per process.
- **Limitation:** The out of the box implementation does not support synchronization of mobile devices.
- **Limitation:** The Data Flow Probe or Integration Service must be installed on a Windows OS.
- **Limitation:** DB2 parallel push mode is not supported in UCMDB 10.01.
- **Problem: Missing DDLs or jars.**

When testing the connection of the integration point, an error with the following phrase appears:

### **Asset Manager DLLs and/or Jars are missing**

**Solution:** See "[Create AMPushAdapterAPI Package with Required AM API Files](#)" on page 390

- **Problem: First time synchronization has many failed CIs.**

The first synchronization in the integration creates a large number of enum values in the Asset Manager database. In some cases, when enabling the parallel push for the first synchronization, it may cause a very large number of deadlocks during the push that is more than the Adapter's auto deadlock handling mechanism can handle.

**Solution 1:** You may try to re-synchronize the failed CIs until they all pass.

**Solution 2:** Add the enum attribute that caused the duplicate key exception to **am-mapping** in the **am-push-mapping.xml**.

- **Problem: A push integration fails with the error message 'Only one connected Asset CI is allowed'**

When running a push integration of Computers to Asset Manager, one of the reconciliation attributes used is Asset Tag. If there is more than one Asset CI connected to the Node CI, it means there is more than one Asset Tag for a single Node; this is not a valid state.

**Solution:** Remove the CIT Computer from the incorrect Asset CIs. This should ensure the Node is connected to either one or no Asset CIs.

- **Problem: Some CIs in a Relations Push fail.**

The Relations flow (TQL query) assumes that you schedule (either in the same job, or in a different job) the different flows that this Relations push depends on, to run before this flow. The following table shows the dependencies:

TQL Query	Depends On
<b>AM Business Element Push</b>	
<b>AM Business Element Relations Push</b>	<b>AM Business Element Push and AM Computer Push</b>  <b>Note:</b> If the Asset Manager Business Element is not related to computers, <b>AM Business Element Relations Push</b> does not depend on <b>AM Computer Push</b> .
<b>AM Computer Push</b>	
<b>AM Computer Relations Push</b>	<b>AM Computer Push</b>
<b>AM Host Server And Running LPAR VM Relations Push</b>	<b>AM Computer Push</b>
<b>AM Host Server And Running Solaris VM Relations Push</b>	<b>AM Computer Push</b>
<b>AM Host Server And Running VM Relations Push</b>	<b>AM Computer Push</b>
<b>AM Installed Software Push</b>	<b>AM Computer Push</b>



TQL Query	Depends On
AM Net Device Relations Push	AM Computer Push
AM Oracle LMS Push	AM Computer Push
AM Software Hypervisor Push	AM Computer Push
AM Software Solaris Push	AM Computer Push

- **Problem: Some CIs in a Software Push fail.**

The software flows assume that you schedule the computer push flow (either in the same job, or in a different job) to run first. See ["Why does an integration point that synchronizes only the AM Installed Software Push TQL query, keep failing?"](#) on page 426.

- **Problem: Missing Root in a TQL Query.**

**Solution:** The integration TQL query must contain a Root Element (1 if it is a CI, 1 or more if it is a Relationship). Update your TQL query by renaming one of the query Elements to **Root**. Make sure your mapping xml is updated accordingly.

See ["What is a Root CI node?"](#) on page 424.

- **Problem: Error in Test Connection. Unable to connect to this database engine.**

The following solution assumes you are connecting to a DB2 or Oracle database.

**Solution:** Validate the following:

1. The database client is installed on the Data Flow Probe/Integration Service machine. See ["Install a Database Client"](#) on page 390.
2. The installed client is a 64 bit version.
3. The client is installed on the actual Probe selected in the integration point configuration.

- **Problem: Multiple Assets are created in Asset Manager for a single UCMDB Node.**

**Solution 1:** This can happen when the maximum length of an attribute is too short compared to the attribute's value in UCMDB. It causes the attribute value to truncate when pushed to Asset Manager. However, on a different execution, when attempting to reconcile the attribute, there will not be a match because of the truncated value in asset Manager. Therefore, increase the attribute length. See "[Update Asset Manager Schema](#)" on page 386.

**Solution 2:** Check and fix customized or changed reconciliation rules.

- **Problem: Error in test connection: Module Ssl : Unable to load dynamic library (libeay64.dll).**

This error occurs when the Windows operating system of the probe is missing the Visual C++ 2008 SP1 or later.

**Solution 1:** Download and install the Microsoft Visual C++ 2008 SP1 Redistributable Package (x64). You may download this from:

<http://www.microsoft.com/download/en/details.aspx?id=2092>.

Thereafter, reboot Windows and restart the Probe.

**Solution 2:** Run Windows update and retrieve the Visual C++ 2008 SP1 or later update. Thereafter, reboot Windows and restart the Probe.

- **Problem: Reconciliation gaps when pushing and populating Business Elements using reconciliation by Global-ID.**

**Solution:** To populate Business Elements using reconciliation by Global-ID, complete the following:

- a. Go to **Dataflow Management > Adapter Managent > Packages** and select **AMPushAdapter**.
- b. Select **Configuration Files > pushMappingAMBusinessElement.xml**.
- c. Replace the file content with the following:

```
<integration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="/am-push-adapter/src/main/resources/schema/am-
push-adapter.xsd">
<info>
  <source name="UCMDB" versions="10.0" vendor="HP"/>
  <target name="AssetManager" versions="9.3" vendor="HP"/>
</info>
<import>
  <scriptFile path="mappings.scripts.AMPushFunctions"/>
</import>
<targetcis>
```

```
<!--Query: BusinessElement(root class = Business Service)(Root) -->
<source_instance_type query-name="AM Business Element Push*" root-element-
name="Root" >
  <target_ci_type name="IS_amAsset">
    <variable name="globalId" datatype="STRING" value="Root['global_id']"/>
    <variable name="vAssetTag" datatype="STRING"
value="AMPushFunctions.uCase(Root['name'])"/>
    <target_mapping name="AssetTag" datatype="STRING" value="vAssetTag"/>
    <target_mapping name="Label" datatype="STRING" value="Root['name']"/>
    <target_ci_type name="IS_amPortfolio">
      <target_mapping name="CMDBId" datatype="STRING" value="globalId"/>
    <target_ci_type name="IS_amModel">
      <target_mapping name="Name" datatype="STRING" value="Root['name']"/>
    <target_ci_type name="IS_amNature">
      <target_mapping name="Code" datatype="STRING" value="'BIZSVC'"/>
    </target_ci_type>
    <target_ci_type name="IS_amModel-Parent">
      <target_mapping name="BarCode" datatype="STRING"
value="AMPushFunctions.getParentBarCodeFromType(Root['root_class'])"/>
    </target_ci_type>
  </target_ci_type>
</target_ci_type>
</source_instance_type>
</targetcis>
</integration>
```

- d. Click **OK**.
- e. In the same directory select the **am-push-mapping.xml** file.
- f. Delete the **am-mapping of ci-type="IS\_amAsset"** and replace with the following:

```
<am-mapping ci-type="IS_amAsset" primary-key="lAstId" operation-type="update_
else_insert" target-ci-type="amAsset" merge-allowed="true" parallel-push-
allowed="true">
  <reconciliation>
    <reconciliation-advanced>PortfolioItem.CMDBId = '${globalId}
'</reconciliation-advanced>
    <reconciliation-keys>
      <reconciliation-key ignore-case="true">AssetTag</reconciliation-key>
    </reconciliation-keys>
  </reconciliation>
  <reference-attribute ci-name="IS_amPortfolio" datatype="STRING"
name="PortfolioItem" reference-direction="child"/>
  <attribute-reconciliation attribute-name="AssetTag" update-
script="mappings.scripts.AMPushFunctions.fIsEmpty(vOldVal) ? vNewVal :
```

```
vOldVal"/>
  <action-on-delete>
    <delete-ci/>
  </action-on-delete>
</am-mapping>

<am-mapping ci-type="IS_amPortfolio" primary-key="lPortfolioItemId" target-
ci-type="amPortfolio" operation-type="update_else_insert" parallel-push-
allowed="true">
  <reconciliation>
    <follow-parent am-prefix="PortfolioItem"/>
  </reconciliation>
  <reference-attribute ci-name="IS_amModel" datatype="STRING" name="lModelId"
reference-direction="child"/>
  <action-on-delete>
    <delete-ci/>
  </action-on-delete>
</am-mapping>
```

- g. Click **OK**.

## Logs

The push adapter framework uses a different logs than the normal fcldb.adapters.\*.log files.

To change the level of the log files to debug, edit the following file:

- On the Data Flow Probe machine:  
**..\DataFlowProbe\conf\log\fcldb.push.properties**
- If using the integration service, on the UCMDB server:  
**..\UCMDB Server\Integrations\conf\log\fcldb.push.properties**

Change the log level to DEBUG:

```
loglevel=DEBUG
```

The integration generates **fcldb.push.\*** logs in the following folder:

- On the Data Flow Probe machine:  
**..\DataFlowProbe\runtime\log\**
- If using the integration service, on the UCMDB server:  
**..\UCMDB Server\Integrations\runtime\log\**

# HP Asset Manager Population Integration

This chapter includes:

## Overview

HP Asset Manager is an asset lifecycle management solution with modular components allowing an IT organization to measure and communicate the value it provides to the business it supports.

UCMDB-Asset Manager integration is implemented by the Asset Manager adapter (AMAdapter) pulling CIs and relationships from Asset Manager to UCMDB.

## Supported Versions

The Asset Manager adapter supports Asset Manager Versions 9.30 and later.

## How to Integrate Asset Manager with UCMDB

This task consists of the following steps:

### Step 1: Get the adapter content package

1. Log on to **UCMDB**.
2. Browse to **Data Flow Management > Adapter Management**.
3. Expand **AMAdapter** from the resources list.
4. Expand **External resources** and select **AMadapter/Content.zip**.
5. Export to the local server and extract all files from **Content.zip**. The AMdatakit and AMDBUpdate folders are displayed. AMdatakit is used to sync dtLastModif among amAsset, amPortfolio, and amComputer tables.

### Step 2: Activate workflows in AM

1. Log on to **Asset Manager Client**.
2. On the **Tools** menu, point to **Workflow**, click **Workflow schemes**.
3. Locate the **Update dtRecCreation** (SQL name: sysCoreUpCrTime) workflow and the **Update last modify time** (SQL name: sysCoreUpMdfifyTime) workflow.
4. On the **General** tab, empty the **End** field of the **Validity** pane.

5. Save the changes.

### Step 3: Create SQL views in the AM database

**Caution:** This batch file can only be used on Windows computers, not UNIX computers.

The batch file should be run from a computer where the client layers of the DBMS (for example, Oracle Database 10g Client R2) used for the AM database are installed.

Running this batch file alters the AM database structure.

Administrative privileges are required at the DBMS level to create the SQL views.

1. Set the ORACLE\_HOME environment variable.
2. Run the create view script.
  - a. Click **start > Run**.
  - b. Enter **cmd**.
  - c. Enter **sqlplus <schema name>/<schema password>@<SID\_Hostname of oracle server>**.
  - d. Enter **GRANT create ANY VIEW to <Username>**.
  - e. Enter **GRANT SELECT ANY TABLE to <Username>**.
  - f. Enter **GRANT create MATERIALIZED VIEW to <Username>**.
  - g. Enter **exit** to exit Oracle.
3. Browse to the AMDBupdate folder.
4. If using an Oracle server, run the command **CreateViews.bat Oracle <Oracle SID> <Username> <Password>**; for example, **CreateViews.bat Oracle SSG\_labm3amdb35 SACM4 topaz**.  
  
If using an SQL server, run the command **CreateViews.bat [MSSQL2005|MSSQL2008] <Server> <Database> <Username><Password>**.
5. Confirm all views created in the AM database.

### Step 4: Make some CI attributes visible


1. Open a browser and log on to the UCMDB Server as an administrator.
2. From the left navigation bar, select **Modeling**.
3. Select **CI Type Manager** and select the CI Type that contains the CI attribute from the navigation tree.
4. On the **Attributes** tab, double click the attribute you want to modify.

By default, the following CI attributes are not visible:

CI Type	Attribute
IpAddress	IP Address
	IP is DHCP
CPU	CPU Speed

5. Select the **Advanced** tab, and select **Visible**.

### Step 5: Create an integration point

1. Log on to **UCMDB**.
2. Browse to **Data Flow Management > Integration Studio**.
3. Click the **New Integration Point**  button.
4. In the New Integration Point dialog box, fill in the following **Integration Properties**.


Name	Recommended Value	Description
<b>Integration Name</b>	AM population	Name given to the integration point.
<b>Adapter</b>	AM Population and Federation Adapter 9.30 and later versions	Adapter to be used for the integration point.



Name	Recommended Value	Description
<b>Is Integration Activated</b>	selected	Select checkbox to create an active integration point.

5. In the **Adapter Properties**, fill in the relevant details.

Name	Recommended Value	Description
<b>Hostname/IP</b>	<user defined>	Enter the hostname or IP address of the database server. For SQL Server with instance name, enter <hostname>\<instance name>
<b>Port</b>	<user defined>	Enter the database port default port for Oracle 1521 and SQL 1433.

Name	Recommended Value	Description
<b>Credentials ID</b>	<user defined>	<p>Click the <b>Add new connection details for selected protocol type</b>  button.</p> <p>Enter the following information:</p> <ul style="list-style-type: none"> <li>■ In <b>Database Type</b>, select the database type of your Asset Manager schema: <ul style="list-style-type: none"> <li>○ Microsoft SQL Server</li> <li>○ Oracle</li> <li>○ DB2</li> </ul> </li> <li>■ In <b>Port Number</b>, type the communication port of the Asset Manager Database.</li> <li>■ In <b>Connection Timeout[msec]</b>, enter the timeout value set by the schema.</li> <li>■ In <b>User Name</b>, enter the user of the schema.</li> <li>■ In <b>Password</b>, enter the password of the schema.</li> <li>■ In <b>Instance Name/ Database Name</b>: <ul style="list-style-type: none"> <li>○ DB2: type the name of the Database Alias you defined in the database connection in <b>Database Name</b>.</li> <li>○ Oracle: type the service name in <b>Instance Name</b>.</li> <li>○ SQL Server: type the name of the schema in <b>Instance Name</b>.</li> </ul> </li> </ul>
<b>DB Name/SID</b>	<user defined>	Enter the database name. For Oracle, enter a service name instead of the SID.

Name	Recommended Value	Description
<b>DB Type</b>	<user defined>	Select on of the following the database types: <ul style="list-style-type: none"> <li>■ Oracle</li> <li>■ SQL Server</li> </ul>
<b>Push Back IDs</b>	Enabled	Identify a unique CI if the feature is disabled and the reconciliation of CIs will not work.
<b>Version</b>	<user defined>	Version of Asset Manager to access
<b>Data Flow Probe</b>	<user defined>	Select the correct Probe from the drop-down list.

6. Click **Test connection** and **OK**.

## Verify UCMDB to AM Configuration

**Note:** On first use, run the full population job. After any changes are made, only run the diff population job.

In order to verify Asset Manager to UCMDB Population flow, we create an Asset in Asset Manager, and verify it is synchronized as a configuration item in UCMDB.

1. Log on to Asset Manager.
2. Create a new virtual machine in the IT equipment module, specifying its name and IT equipment type (in our case, **Virtual Machine**).
3. In the **Assets** view, add the serial number.
4. Log on to **UCMDB**.
5. Browse to **Data Flow Management > Integration Studio**.
6. Select the **AM Population** integration point.
7. In the Integration Jobs pane, select the **Population** tab and click the **AM Population** job.
8. Click the **Full Synchronization - Runs the selected job, synchronizing all of the data** button.
9. When the Confirm synchronizing window appears, click **Yes**.
10. Click the **Refresh** button and wait until the **Succeeded** message appears in the **Status** tab. The updated synchronization status appears.
11. Browse to **Modeling > IT Universe Manager**.
12. Select the **Search CIs** tab and search for the name of the virtual machine specified in [step 2](#).
13. Confirm that a corresponding CI exists in UCMDB.

**Note:** Note its name as it will be used in the next section.

## What CI data is populated from AM to UCMDB?

This section describes the CI data that is populated from AM to UCMDB.

### Population TQLs

The types of CIs that can be propagated from AM to UCMDB are defined by the following TQLs included in the integration package.

- locationDataImport\_930

This query is used to populate location data from AM.

- hostImport\_930

This query is used to populate computer data from AM.

- networkImport\_930

This query is used to populate network data from AM.

- printerImport\_930

This query is used to populate printer data from AM.

- businessElementDataImport\_930

This query is used to populate business service assets from AM.

- businessElementRelationsImport\_930

This query is used to build relations between business service assets.

#### **To access these TQLs:**

1. Open a browser and connect to the UCMDB Server.
2. Click the **Modeling** tab.
3. Select **Modeling Studio**.
4. Click the **Resources** tab, and select **Queries** from the **Resource Type** drop-down list.

5. Expand the **Root/Integration/AM Data In** menu. The TQLs located under the path are used to manage CI data synchronization from AM to UCMDB. They have the same structure but manage different type of CI data.

## Criteria for AM records to be propagated

### Step 1: Node CIs

Node CIs in UCMDB correspond to records in the **amComputer** table in AM.

Out-of-the-box, the **amComputer** records need to satisfy the following conditions to be propagated to UCMDB as Node CIs:

- The status for the **Portfolio Item** linked to the **amComputer** record is in use or in stock (**amPortfolio:seAssignment = In use or In stock**).

This can be configured by modifying a configuration file (see "[condition\\_rules.xml](#)" on page 456).

- The **amComputer:ComputerType** field of the **amComputer** record must have a value that is present in the third column of the following table.

**Relationship between TQL, CI Types, and computer types**

**Note:** The path populates CIs of these CI Types to UCMDB under **Managed Objects/ConfigurationItem/ InfrastructureElement/Node/**

This TQL	Populates CIs of these CI Types to UCMDB	The CI Type corresponds to these computer types in AM ( <i>amComputer:ComputerType</i> )
hostDataImport_930	Computer	Computer
		Desktop computers
		Computer servers
		Laptop
		Virtual Machine
	Computer\Windows	Windows computer
		Windows desktop computer
		VMware VirtualCenter
		VMware ESX Server
	Computer\Unix	Unix server computer
		Unix desktop computer
		Solaris Zone server
	Computer\Mainframe	Mainframe Storage Array Mainframe CPC <b>Note:</b> Storage Array is not a computer type value. By default, AM will handle those records where <b>amNature.Name</b> is Storage Provider as <b>Storage Array</b> .
networkDataImport_930	Net Device\Firewall	Firewall
	Net Device Router	Router
	Net Device\Switch	Switch

This TQL	Populates CIs of these CI Types to UCMDB	The CI Type corresponds to these computer types in AM ( <i>amComputer:ComputerType</i> )
networkDataImport_930	Net Device\ATM Switch  <b>Node:</b> ADSL Modem Appletalk Gateway Bandwidth Manager Cable modem CSU_DSU Ethernet FDDI HUB KVM Switch Load Balancer Multicast Enabled Router NAT Router Token Ring Undefined Network Component VoIP Gateway VoIP Switch VPN Gateway Wireless Access Point, Frame_relay_switch San_gateway San_router San_switch Pad_handhel	ATM switch
printerDataImport_930	Net Device\Net Printer	Network printer



**Note:** In addition to the Node CIs, each of the TQLs also populates information (if exists) such as IP address, interface, and locations from AM to UCMDB.

- The mapping between CI Type and computer type (value of the **amComputer:ComputerType** field) is defined in the **discriminator.properties** file (see "[discriminator.properties](#)" on [page 453](#)).

## Step 2: Business Element

In AM, only those business service asset whose status (`amAsset.Status`) in the list of **built**, **catalogued**, **chartered**, **designed** and **requested** will be populated to UCMDB. If the status is **retired**, it will be removed from UCMDB.

Relationship between AM business service asset and CI types:

Nature code	CI Type
BIZSVC	business_service
BIZAPP	business_application
INFRASVC	infrastructure_service

## What is created in UCMDB during population?

When an **amComputer** record (with the associated information) is populated from AM to UCMDB:

- A Node CI is created in UCMDB.
  - Its CI Type depends on the value of the **amComputer:ComputerType** field.
    - Relationship between TQL, CI Types, and computer types.
- For each Networkcard record (**amComputer:NetworkCards**) attached to the AM Node CI, two CIs are created in UCMDB: **IP address** and **Interface**.

The values of the following AM **amNetworkCard** table field are used to create the IPAddress and Interface CIs:

- **amNetworkCard:TcplpAddress**, together with other field values, is used to create an IP Address CI.

- **amNetworkCard:PhysAddress**, together with other field values, is used to create an Interface CI.
- If the AM **amComputer** data has an associated location which is comprised of several layers, then one Location CI is created for each layer of location.

For example, in AM one computer has a location of **/Ariane Building/31st Floor/030 – Office/**, then when the Node CI for the computer is populated to UCMDB, 3 location CIs are created for each layer in the location, representing the building, floor and room respectively.

**Note:** The layers of location in AM should be less than the layer defined in the TQL, which by default is 3. Otherwise, you need to add elements to the corresponding TQL to support it.

## Reconciliation

For each CI Type, the data reconciliation is governed by the reconciliation rule set in UCMDB.

You can check the reconciliation rule for each CI Type on the **Details** tab of the CI Type. The field name is **Identification**.

## What happens when changes occur in AM during data population?

- If the value of the **amPortfolio:seAssignment** field has been changed to a value not present in `condition.rules.xml` (see "[condition\\_rules.xml](#)" on page 456), the corresponding CI is deleted from UCMDB with the associated IPAddress CIs (if exists).
- However, if the IT equipment record is deleted from AM, the previously populated CI remains in UCMDB.

## Supported CI Types

**To find out the CI Types supported by the AM adapter out-of-the-box:**

1. Start UCMDB Server.
2. Open a browser and connect to UCMDB Server as administrator.
3. From the left navigation bar, click the **Data Flow Management** tab.
4. Click **Integration Studio**.

5. Select the integration point created for AM.
6. Select the **Federation** tab on the right pane. The supported CI Types are shown in the **Supported and Selected CI Types** section. You can click **Expand all** in the toolbar to view all CI Types at a glance.

**Note:**

- Most CI Types supported by the integration are UDM compliant, except for **Printer** and those CI Types under Node.
- Those grayed CI Types are not supported.
- If a CI Type is supported, all its children CI Types are automatically supported.

## Supported CI attributes and the mapping with the AM fields

For each CI Type supported by the integration, below are the mappings between the UCMDB CI attribute and AM field. For each CI attribute, its compliance with BTO Data Model (UDM) 1.1 is also identified.

### CI Type: Asset

UCMDB attribute	AM field	UDM entity?
asset_tag	amPortfolio.asset_tag	Yes
description	amAsset.AssetSerialNo	Yes
id1	amPortfolio.lPortfolioItemld	No
assignment	amPortfolio.seAssignment	Yes

### CI Type: IpAddress

UCMDB attribute	AM field	UDM entity?
id1	amNetworkCard.lNetworkCardld	No
ip_address	amNetworkCard.TcpIpAddress	No
ip_address_property	View_amNetworkCard.ip_address_property	Yes

UCMDB attribute	AM field	UDM entity?
ip_address_value	View_amNetworkCard.ip_address_value	Yes
ip_dhcpdomainname	amNetworkCard.DHCPServer	No
ip_isdhcp	amNetworkCard.BDHCPEnabled	No
ip_netmask	amNetworkCard.SubnetMask	No
name	amNetworkCard.TcplpAddress	Yes

## CI Type: Node

All the children CIs of the Node CI inherit the same attribute from the Node CI. In particular these CI Types are relevant with the integration:

- Computer
- Windows
- Unix
- Mainframe
- Firewall
- Router
- Switch
- ATM Switch
- Net Printer

UCMDB attribute	AM field	UDM entity?
asset_tag	amPortfolio.AssetTag	Yes
create_time	amPortfolio.dtRecCreation	Yes
data_externalid	amComputer.lComputerId	No
data_note	amPortfolio.seAssignment	No
default_gateway_ip_address	amNetworkCard.DefaultGateway	Yes

UCMDB attribute	AM field	UDM entity?
description	amAsset.Description	Yes
discovered_os_name	amComputer.OperatingSystem	Yes
discovered_os_version	amComputer.OSServiceLevel	Yes
host_isdesktop	amComputer.ComputerType	No
host_isvirtual	amComputer.ComputerType	No
id1	amComputer.ItemId	No
last_modified_time	amComputer.dtLastModif	Yes
memory_size	amComputer.IMemorySizeMb	Yes
name	amComputer.TcplpHostName	Yes
node_model	amModel.ModelName	Yes
node_role	amComputer.ComputerType	Yes
primary_dns_name	amComputer.TcplpDomain	Yes
serial_number	amAsset.SerialNo	Yes
globalid	amPortfolio.CMDBId	Yes

## CI Type: CPU

UCMDB attribute	AM field	UDM entity?
core_number	amComputer.ICPUCoreNumber	No
cpu_id	amComputer.IComputerId	Yes
cpu_speed	amComputer.ICPUSpeedMHz	No
id1	amComputer.ItemId	No
name	amComputer.Name	Yes
cpu_clock_speed	amComputer.ICPUSpeedMHz	Yes

## CI Type: DiskDevice

UCMDB attribute	AM field	UDM entity?
Id1	amPhysicalDrive.IPhysDriveId	No
name	amPhysicalDrive.Description	Yes

## CI Type: FileSystem

UCMDB attribute	AM field	UDM entity?
disk_size	amPhysicalDrive.ITotalSizeMb	No
Id1	amPhysicalDrive.IPhysDriveId	No
name	amPhysicalDrive.Description	Yes

## CI Type: InstalledSoftware

UCMDB attribute	AM field	UDM entity?
description	Model.comment.memComment	Yes
File_system_path	amSoftInstall.Folder	Yes
id1	amSoftInstall.ISoftInstId	No
Name	amModel.Name	Yes
Vendor	amBrand.Name	Yes
Version	amSoftInstall.VersionLevel	Yes

## CI Type: Interface

UCMDB attribute	AM field	UDM entity?
id1	amNetworkCard.INetworkCardId	No
interface_description	amNetworkCard.Description	Yes
mac_address	amNetworkCard.PhysAddress	Yes

## CI Type: LogicalVolume

UCMDB attribute	AM field	UDM entity?
id1	amLogicalDrive.ILogDriveId	No
logicalvolume_free	amLogicalDrive.IFreeSpaceMb	No
logicalvolume_fstype	amLogicalDrive.Media	No
logicalvolume_size	amLogicalDrive.ITotalSizeMb	No
Name	amLogicalDrive.MountPoint	Yes

## CI Type: Printer

UCMDB attribute	AM field	UDM entity?
id1	amPortfolio.IPortfolioItemId	No
name	amPortfolio.ModelName	Yes

## CI Type: Location

UCMDB attribute	AM field	UDM entity?
id1	amLocation.ILocalId	No
location_type	amLocation.LocationType	Yes
name	amLocation.FullName	Yes
Location_bar_code	amLocation.BarCode	Yes

## CI Type: BusinessElement

UCMDB attribute	AM field	UDM entity?
Id1	amPortfolio.IPortfolioItemId	No
Name	amAsset.AssetTag	Yes

<b>UCMDB attribute</b>	<b>AM field</b>	<b>UDM entity?</b>
last_modified_time	amPortfolio.dtLastModif	Yes
create_time	amPortfolio.dtRecCreation	Yes
description	amAsset.Description	Yes
data_note	amAsset.Status	No
global_id	amPortfolio.CMDBId	Yes



## The configuration files used by the integration

This section describes the configuration files used by the integration.

### Where are the configuration files located

The configuration files that are relevant with the integration are located in the following path in UCMDB:

#### **Data Flow Management\Adapter Management\AMAdapter\Configuration Files**

##### discriminator.properties

When a CI is populated to UCMDB, its CI Type is defined according to the value of **amComputer:ComputerType** in AM.

This file defines the mapping between the values of **amComputer:ComputerType** and the UCMDB CI Type.

A summary of the mapping can be found at the following location: ["What CI data is populated from AM to UCMDB?" on page 441](#).

##### server\_virtual\_distinguisher.properties

When a CI is populated to UCMDB, the **Host is Virtual** attribute of the CI is defined according to the value of **amComputer:ComputerType** for the CI in AM.

This file defines the mapping between the values of **amComputer:ComputerType** and the **Host is Virtual** attribute.

<b>If the value of <i>amComputer:ComputerType</i> in AM is...</b>	<b>the value of <i>Host is Virtual</i> in UCMDB is...</b>
Virtual Machine	Virtual Machine

If the value of <i>amComputer:ComputerType</i> in AM is...	the value of <i>Host is Virtual</i> in UCMDB is...
<ul style="list-style-type: none"> <li>• Windows computer</li> <li>• Windows desktop computer</li> <li>• VMware Virtual Center</li> <li>• VMware ESX server</li> <li>• Unix server computer</li> <li>• Unix desktop computer</li> <li>• Solaris Zone Server</li> <li>• Desktop computers</li> <li>• Computer servers</li> <li>• Laptop</li> <li>• Mainframe</li> <li>• ATM switch</li> <li>• Firewall</li> <li>• Router</li> <li>• Switch</li> <li>• Network printer</li> </ul>	<p>No</p>

server\_desktop\_distinguisher.properties

When a CI is populated to UCMDB, the **Host is Desktop** attribute of the CI is defined according to the value of **amComputer:ComputerType** for the CI in AM.

This file defines the mapping between the values of **amComputer:ComputerType** and the **Host is Desktop** attribute.

If the value of <i>amComputer:ComputerType</i> in AM is...	the value of <i>Host is Desktop</i> in UCMDB is...
<ul style="list-style-type: none"> <li>• Windows desktop computer</li> <li>• Unix desktop computer</li> <li>• Desktop computers</li> </ul>	Yes
<ul style="list-style-type: none"> <li>• Windows computer</li> <li>• VMware Virtual Center</li> <li>• VMware ESX server</li> <li>• Unix server computer</li> <li>• Solaris Zone Server</li> <li>• Computer servers</li> <li>• Laptop</li> <li>• Mainframe</li> <li>• ATM switch</li> <li>• Firewall</li> <li>• Router</li> <li>• Switch</li> <li>• Network printer</li> </ul>	No

### fixed\_values.txt

When populating IP address data from AM to UCMDB, the *IpAddress:routing\_domain* attribute value of the IP Address CI Type is populated with DefaultDomain.

This file allows you to change the default value for the attribute.

### location\_type\_transformer.xml

This file contains all the mappings between location types in AM and UCMDB. Mapping for a location type is represented by an entry like this:

```
<value cmdb-value= "room" external -db-value="room"/>
```

The UCMDB location type is the value for the cmdb-value attribute, while the AM location type is the value for the external-db-value attribute.

All the AM location types must be mapped with those in UCMDB for the data population to be successful.

### condition\_rules.xml

Out-of-the-box, this file defines the basic rules to decide what kind of CIs can be populated from AM to UCMDB and remove CIs from UCMDB.

Out-of-the-box, only **amComputer** records whose associated **Portfolio Item** is in stock (**amPortfolio:seAssignment = In stock**) can be populated to UCMDB as Node CIs.

This file allows you to customize the assignments (**amPortfolio:seAssignment**) computers could have for them to be populated.

This configuration file uses values stored in the AM database to designate the value of the seAssignment field. Refer to the following table for the relationship between the value stored in the database and the displayed value.

Value stored in the displayed	Actual displayed text for <i>seAssignment</i>
0	In use
1	In stock
2	Retired
3	Awaiting receipt
4	Return for maintenance
5	Return to supplier
6	Missing

You can designate multiple values in this file to allow populating computers with different assignments to UCMDB.

Find the statement:

```
<expression join="AND" field="data_note" data-type="INTEGER" operator="IN"
value="0,1"/>
```

This means it will only add/update those Node CIs whose data\_note (amPortfolio.seAssignment) is 0 (In use) or 1 (In stock).

Find the statement:

```
<expression join="NOT" field="data_note" data-type="INTEGER" operator="IN"
value="0,1"/>
```

This means it will only remove those Node CIs whose data\_note is not in the list of 0 and 1.

**Note:** Keep the same value for these two expressions.

This means that computers with the assignments (**amPortfolio:seAssignment**) **In use** and **In stock** can be populated to UCMDB. If you change their assignment to other values in AM, the corresponding CIs will be deleted when you run the population job next time (see ["What happens when changes occur in AM during data population?" on page 446](#)).

Global\_id\_mapping.properties

Make sure **Push back ID** is enabled when creating the integration point. Define how to write global\_id back to AM. By default, **global\_id** is saved in the amPortfolio.CMDBId field.

## Integrate BSM and OMi

## Integrate Service Manager to OMi

To integrate OMi and Service Manager, follow the steps on pages 72-137 in the *HP Operations Manager i Integrations Guide* and the "Incident Exchange (OMi - SM) integration" portion of the Business Service Management integration section of the *HP Service Manager Help Center*.

**Note:** Chapter 24 of the *HP Operations Manager i Integrations Guide* discusses how to configure Service Health to view changes and incidents. Some of the steps described in this chapter that relate to the Service Manager- Universal Configuration Management Database integration are performed automatically by the Deployment Manager. Therefore, you should review the integration

configuration carefully before proceeding.

**Note:** The following sections are excerpted from the integrations section of the Service Manager Help Center. However, to ensure you have the latest information, you should reference the latest published version of the document in question.

## Incident Exchange (OMi - SM) integration

The Incident Exchange (OMi - SM) integration is a bidirectional integration between incident records in HP Service Manager and events in HP Business Service Management (BSM) Operations Management provided by the Operations Manager i (OMi) license.

This integration requires configurations on both product sides. For details, see ["Incident Exchange \(OMi - SM\) integration setup" on the next page](#).

**Note:** As of version 9.34, ITSM Enterprise Suite can integrate with multiple BSM OMi servers. For details, see ["Add an integration instance for each Operations Manager i \(OMi\) server" on page 467](#).

Service Manager can accept RESTful-based requests from OMi to create incidents in Service Manager, based on events information in OMi. When Service Manager accepts an incident creation request from a remote OMi server, it creates an incident record and automatically assigns it to an existing group based on certain field values. Incident Management users can view the details of the related OMi event by clicking the **View OMi Event** menu option from the incident record. See ["View related OMi event details from an incident" on page 479](#).

When an Incident Management user makes any changes to the incident record, Service Manager automatically synchronizes the changes to the corresponding event in OMi, by sending an update request to the OMi RESTful web service interface. In the event of a synchronization failure, a queuing mechanism will re-synchronize the changes. See ["Synchronization of incident changes back to Operations Manager i \(OMi\)" on page 478](#).

System administrators can configure global settings that determine whether and when incident records opened from OMi events can be automatically closed. However, Incident Management users can mark individual OMi incident records as eligible or ineligible for automatic closure. See ["Configure automatic closure for OMi incidents" on page 474](#) and ["Mark an incident for automatic closure" on page 480](#).

## Incident Exchange (OMi - SM) integration setup

The Incident Exchange (OMi - SM) integration requires the following configuration tasks be completed on the HP Service Manager and Business Service Management (BSM) systems.

1. ["Create user accounts for the Incident Exchange \(OMi - SM\) integration" on the next page.](#)

This task creates a user account on each product side for the two systems to connect to each other and to synchronize data.

2. ["Configure an event forwarding rule in Operations Manager i \(OMi\) " on page 464.](#)

This task configures a rule for the OMi server to forward events to the Service Manager server.

3. ["Configure the Service Manager server as a connected server in Operations Manager i \(OMi\)" on page 461.](#)

Perform this task in BSM OMi. If you need to integrate ITSM Enterprise Suite with more than one OMi server, perform this task on each of the OMi servers.

4. ["Enable incident drill-down from Operations Manager i \(OMi\) Event Browser" on page 465.](#)

This task configures the Service Manager web tier in the **sm:ServiceManagerAdapter** script in OMi.

5. ["Configure SSL for the Incident Exchange \(OMi - SM\) integration " on page 465.](#)

This task is needed if your OMi server requires HTTPS connections. If SSL is not configured in this case, changes on incidents that are created from OMi will not be able to be synchronized back to OMi.

6. ["Configure the Instance Count in the SMOMi integration template" on page 466.](#)

This task is needed only when you have more than one OMi server. The Instance Count setting defines the allowed number of SMOMi integration instances in Service Manager (default: 1).

7. ["Add an integration instance for each Operations Manager i \(OMi\) server" on page 467.](#)

This task creates and enables an instance of this integration in Integration Manager (SMIS). A separate integration instance is required for each OMi server.

8. ["Enable LW-SSO for the Incident Exchange \(OMi - SM\) integration" on page 473.](#)

Lightweight Single Sign-On (LW-SSO) is optional but recommended for the Incident Exchange (OMi - SM) integration. This task includes enabling LW-SSO in the Service Manager server and Web tier, as well as in BSM.

9. (Optional) ["Configure automatic closure for OMi incidents" on page 474.](#)

This task is optional. By default, automatic closure is disabled for OMi incidents. System administrators can enable automatic closure and further configure under what conditions OMi incidents can be automatically closed.

10. (Optional) ["Change the default assignment group for OMi incidents" on page 477.](#)

This task is optional. When created, OMi incidents are automatically assigned with an assignment group based on their certain field values and a predefined default assignment group. System Administrators can change the default assignment group setting ( **Application**).

## Create user accounts for the Incident Exchange (OMi - SM) integration

### **Applies to User Roles:**

System Administrator

The Incident Exchange (OMi - SM) integration is bidirectional. Synchronizing events and incidents between the HP Service Manager and BSM OMi systems requires integration accounts be set up for the two systems to access each other.

1. Create an operator record with system administration privileges in Service Manager.

This is the user account that the OMi server uses to access Service Manager and to forward events to Service Manager.

Later, when configuring the Service Manager server as a connected server in OMi, you need to specify this operator's the login name and password on the **Outgoing Connection** tab. See ["Configure the Service Manager server as a connected server in Operations Manager i \(OMi\)" on the next page.](#)

2. Create a user account with system administration privileges on each BSM OMi server. For details, see the BSM online help.

This is the user account that Service Manager uses to access the OMi server and to synchronize incident changes back to the OMi server.



Later, when configuring the Service Manager server as a connected server in OMi, you must specify this user's login name as the **Name** of the Service Manager server on the **General** tab, and specify this user's password as the **Password** on the **Incoming Connection** tab. See ["Configure the Service Manager server as a connected server in Operations Manager i \(OMi\)"](#) below.

Also, you need to specify the same user account when adding an SMOMi integration instance for each OMi server (the **username** and **Password** parameters). See ["Add an integration instance for each Operations Manager i \(OMi\) server"](#) on page 467.

## Configure the Service Manager server as a connected server in Operations Manager i (OMi)

Synchronizing changes between HP Business Service Management (BSM) Operations Management events and HP Service Manager incidents requires configuring a Connected Server within OMi to correctly identify the target Service Manager server instance.

To configure the Service Manager server as a target connected server, perform the following steps:

1. Log on to HP Business Service Management as a system administrator.
2. Navigate to the Connected Servers manager in the Operations Management user interface:  
**Admin > Operations Management > Setup > Connected Servers**
3. Click the **New** button to open the **Create New Server Connection** dialog box.
4. In the **Display Name** field, enter a name for the Service Manager server.

The **Name** field is filled automatically. If the auto-completed value is not the user name of the BSM user account you created for Service Manager to access the OMi server, change the value to the correct user name. See ["Create user accounts for the Incident Exchange \(OMi - SM\) integration"](#) on the previous page.

Make a note of the Name of the new target server. You need to provide it later on as the **username** when configuring the Service Manager server to communicate with the OMi server. See ["Add an integration instance for each Operations Manager i \(OMi\) server"](#) on page 467.

Optionally, enter a description for the new target server.

Make sure that you check the **Active** check box.

Click **Next**.

5. Select **External Event Processing** to choose the server type suitable for an external incident manager like Service Manager.

Click **Next**.

6. Enter the **Fully Qualified DNS Name** of the target Service Manager server.

Click **Next**.

7. In the **Integration Type** dialog box, you can choose between using a Groovy script adapter, or the Event Synchronization Web Service.

- a. As an HP Service Manager Groovy script adapter is provided for integrating with Service Manager, select **Call Script Adapter**.

- b. In the **Script Name** field, select **sm:ServiceManagerAdapter**.

- c. Click **Next**.

8. In the **Outgoing Connection** dialog, enter the credentials (user name, password, and port number) to connect to the Service Manager server and to forward events to that server.

- a. In the **User Name** and **Password** fields, enter the Service Manager user credentials you created for the integration. See ["Create user accounts for the Incident Exchange \(OMi - SM\) integration" on page 460](#).

- b. Repeat the password entry in the **Password (Repeat)** field.

- c. In the **Port** field, enter the communications port of the Service Manager server.

The Service Manager server configuration file (sm.ini) defines the http and https ports. Enter the http port when Service Manager is running in http mode, or enter the https port when it is running in secure http mode.

**Tip:** Clicking **Set default port** automatically populates the **Port** field with the default port (**13080** for http or **13443** for secure http). However, your actual Service Manager ports may differ from the default values.

- d. If the Service Manager server uses secure http (SSL) mode, select the **Use Secure HTTP** check

box. If it uses http mode, make sure the check box is not checked.

- e. If the **Use Secure HTTP** check box is selected, download and install a copy of the Service Manager server's SSL certificate by clicking the link **Retrieve from Server**, or **Import from File** if the certificate is available in a local file.
- f. Make sure that the **Enable Synchronize and Transfer Control** check box is checked.

When the **Enable Synchronize and Transfer Control** flag is set, an Operations Management operator is then able to transfer ownership of the event to the target connected server. If the flag is not set, then the option **Synchronize and Transfer Control** does not appear in the list of forwarding types when configuring forwarding rules.

Also, note that if the **Enable Synchronize and Transfer Control** flag is not set for any target connected server, the **Transfer Control to** option does not appear at all in the Event Browser context menu.

If a specific server is configured without the **Enable Synchronize and Transfer Control** flag set, then that server is not available in the Event Browser context menu as a server to which you can transfer ownership.

- g. Click **Test Connection**. A **Success** or **ERROR** hyperlink appears.

Click the link to get a more detailed message.

- h. Click **Next**.

- 9. If, in addition to automatically generating Service Manager incidents from OMi events, you want to also be able to drill-down into Service Manager, you need to specify the fully qualified DNS name and port of the Service Manager web application server (for example, Tomcat).

**Note:** To enable incident drill-down to Service Manager, you must have the Service Manager web tier deployed on a web application server.

In the **Event Drilldown** dialog box, configure the **Fully Qualified DNS Name** and **Port** of the web application server where the Service Manager web tier is deployed.

**Note:** If you do not specify a server in the **Event Drilldown** dialog box, it is assumed that the web tier is deployed on the same machine as the Service Manager server.

Click **Next**.

10. The next thing to do is to enable event changes to be synchronized back from Service Manager to OMi. For this you need to provide credentials for the Service Manager server to access the OMi server in the **Incoming Connection** dialog box.
  - a. Select the **Accept event changes from external processing server** check box.

**Note:** If **Enable Synchronize and Transfer Control** was previously selected, the **Accept event changes from external processing server** option is assumed, and cannot be disabled.

- b. Enter the password of the user account that you created for the Service Manager server to access the OMi server. See "[Create user accounts for the Incident Exchange \(OMi - SM\) integration](#)" on page 460.
  - c. Make sure the **User Name** (auto-generated) matches the one of the user account you created. If not, click the **General** tab, and change the **Name** field to the correct value.
  - d. Click **Finish**. The target Service Manager server appears in the list of Connected Servers.

## Configure an event forwarding rule in Operations Manager i (OMi)

Once you have configured the HP Service Manager server as a connected server in HP Business Service Management (BSM) Operations Management i (OMi), you need to configure an event forwarding rule for the OMi server to forward events to Service Manager.

1. Log on to BSM as a system administrator.
2. Navigate to **Admin > Operations Management > Event Automation**.
3. Click the **New Item** button.
4. In the **Display Name** field, enter a name for the rule. For example, `smserver1`.
5. In the **Event Filter** field, select **Critical**.
6. In the **Target Servers** field, select the Service Manager server you configured as a connected server, and then click the **Add** button.

The details of the target server display.

7. In the **Forwarding Type** field, select **Synchronize and Transfer Control**.
8. Click **OK**.

The forwarding rule displays in the **Event Forwarding Rules** section.

## Enable incident drill-down from Operations Manager i (OMi) Event Browser

Once you have configured the HP Service Manager server as a connected server in HP Business Service Management (BSM) Operations Management i (OMi), if you want to be able to drill down to Service Manager incidents from the OMi Event Browser, you need to configure the Service Manager web tier in the **sm:ServiceManagerAdapter** script in OMi.

To configure the Service Manager web tier name in the **sm:ServiceManagerAdapter** script in OMi, follow these steps:

1. Log on to BSM as a system administrator.
2. Navigate to **Admin > Operations Management > Setup > Connected Servers**.
3. Click the **Manage Scripts** icon in the toolbar.
4. Click the **sm:ServiceManagerAdapter** field, and then click the **Edit** icon.
5. On the **Script** tab, locate the following string in the script:

```
private static final String SM_WEB_TIER_NAME
```

6. Change the value of the parameter above to the name of your Service Manager web tier. For example, if your web tier file name is **sm-2015.war**, change the parameter value to:

```
private static final String SM_WEB_TIER_NAME=sm-2015
```

7. Click **OK** to save the script.

## Configure SSL for the Incident Exchange (OMi - SM) integration

### **Applies to User Roles:**

System Administrator

When Business Service Management (BSM) is configured to accept https connections only, you must configure SSL for the integration. If you do not do so, changes on an incident that is created from OMi cannot be synchronized back to BSM/OMi.

**Note:** The following steps describe how you do so by using the built-in keytool in Service Manager, and the file paths are for Windows only. Be sure to change the file paths accordingly if your Service Manager system is running on Unix.

To configure SSL for the integration, follow these steps:

1. Import the BSM root certificate to the Service Manager server trusted keystore.

The following is an example of the command line:

```
<SM Install path>\server\RUN\jre\bin\keytool -import -alias myCA -file <.pem
file of your BSM root certificate> -keystore <SM Install
path>\Server\RUN\jre\lib\security\cacerts -storepass <changeit>
```

**Note:** Where: *changeit* is the default password of the trusted keystore. Change it to your own password if you have changed it.

2. Add the following parameters to the Service Manager server configuration file (<SM install path>\Server\RUN\sm.ini):

```
truststoreFile:<SM install path>\Server\RUN\jre\lib\security\cacerts

truststorePass:<changeit>
```

3. Restart the Service Manager Server service.

## Configure the Instance Count in the SMOMi integration template

### Applies to User Roles:

System Administrator

As of version 9.34, HP Service Manager can integrate with more than one BSM OMi server. However, by default, only one OMi server is allowed. If you need to integrate Service Manager with more than one OMi server, you need to configure the Instance Count setting in the SMOMi integration template, as described below.

1. Log on to ITSM Enterprise Suite as a system administrator.
2. Type `db` in the command line, and press Enter.
3. In the **Table** field, type `SMISRegistry`, and click **Search**.

The SMIS integration template form opens.

4. Click **Search**.

A list of SMIS integration templates opens.

5. Select **SMOMi** from the list.
6. In the **Instance Count** field, change the value of 1 to the number of OMi servers that you want to integrate with ITSM Enterprise Suite. For example, if you need two OMi servers, change the value to 2.
7. Click **Save**.

## Add an integration instance for each Operations Manager i (OMi) server

### Applies to User Roles:

System Administrator

Once you have completed your configuration in HP Business Service Management Operations Management, and have changed the Instance Count setting in the SMOMi integration template (which is needed only when you have multiple OMi servers), you are ready to add and enable a separate integration instance in Service Manager for each OMi server. For example, if you have two OMi servers, you must configure two SMOMi integration instances.

### Support of multiple OMi servers

As of version 9.34, Service Manager can integrate with multiple OMi servers. This is implemented through the Instance Count setting and the **omi.mgr.id** parameter in the SMOMi integration template.

By default, the SMOMi integration template supports only one integration instance. If you have multiple OMi servers, before you proceed, make sure you have already updated the Instance Count setting in the SMOMi integration template. For details, see ["Configure the Instance Count in the SMOMi integration template" on the previous page](#).

When using the **omi.mgr.id** parameter, keep the following in mind:

- If you have only one OMi server (and hence need only one SMOMi integration instance), you must either correctly configure this parameter or clear the entire row of this parameter (both the parameter name and value) in the SMOMi integration instance, otherwise the integration will not work.
- If you have multiple OMi servers (and hence need multiple SMOMi integration instances), you must correctly configure this parameter in all SMOMi integration instances. Only those correctly configured integration instances will work. If none of the SMOMi integration instances are correctly configured, none of them will work.
- Users can view the OMi event details from an OMi incident record only when you specify the **omi.mgr.id** parameter correctly. If the value you specify in the corresponding SMOMi integration instance does not match the Universally Unique Identifier (UUID) which is automatically generated in the OMi server for the target Service Manager server and stored in the Incident record, users will not see the **View OMi Event** option from the Incident record.

To add and enable an Incident Exchange (OMi - SM) integration instance:

1. Log on to ITSM Enterprise Suite as a system administrator.
2. Click **Tailoring > Integration Manager**.
3. Click **Add**.

The Integration Template Selection wizard opens.

4. Select **SMOMi** from the Integration Template list.

**Note:** Ignore the **Import Mapping** check box, which has no effect on this integration.

5. Click **Next**.
6. Complete the integration instance information:
  - Modify the **Name** and **Version** fields to the exact values you need.
  - In the **Interval Time (s)** field, enter a value. For example: 600. If an OMi opened incident fails to be synchronized back to OMi, Service Manager will retry the failed task at the specified interval (for example, 600 seconds).



- In the **Max Retry Times** field, enter a value. For example: 10. This is the maximum allowed number of retries for each failed task.
  - (Optional) In the **SM Server** field, specify a display name for the Service Manager server host. For example: my\_Local\_SM.
  - (Optional) In the **Endpoint Server** field, specify a display name for the BSM server host. For example: my\_BSM\_1.
  - (Optional) In the **Log File Directory** field, specify a directory where log files of the integration will be stored. This must be a directory that already exists on the Service Manager server host.
  - (Optional) In the **Log Level** field, change the log level from INFO (default) to another level. For example: **WARNING**.
  - (Optional) If you want this integration instance to be automatically enabled when the Service Manager Server service is started, select **Run at system startup**.
7. Click **Next**. The Integration Instance Parameters page opens.
  8. On the **General Parameters** tab, complete the following fields as necessary:

Field	Sample Value	Description
omi.server.url	http://<servername>:opr-gateway/rest/synchronization/event	This is the URL address of the OMi server's RESTful web service. Replace <servername> with the fully qualified domain name of your OMi server.
http.conn.timeout	30	The HTTP connection timeout setting in seconds.  <b>Note:</b> The out-of-box value is 30 (seconds), and 15 (seconds) is used if this field is empty.

Field	Sample Value	Description
http.rec.timeout	30	<p>The HTTP receive timeout setting in seconds.</p> <p><b>Note:</b> The out-of-box value is 30 (seconds), and 15 (seconds) is used if this field is empty.</p>
http.send.timeout	30	<p>The HTTP send timeout setting in seconds.</p> <p><b>Note:</b> The out-of-box value is 30 (seconds), and 15 (seconds) is used if this field is empty.</p>
sm.mgr.id	55436DBE-F81E-4799-BA05-65DE9404343B	<p>The Universally Unique Identifier (UUID) automatically generated for this instance of Service Manager.</p> <p><b>Note:</b> This field is automatically completed each time when you add an SMOMi integration instance. Do not change it, otherwise the integration will not work properly.</p>

Field	Sample Value	Description
omi.reference.prefix	urn:x-hp:2009:opr:	<p>The prefix of the BDM External Process Reference field, which will be present in incoming synchronization requests from the OMi server.</p> <p><b>Note:</b> This field is automatically completed and has a fixed value. Do not change it.</p>
sm.reference.prefix	urn:x-hp:2009:sm:	<p>The prefix of the BDM External Process Reference field, which will be present in outgoing synchronization requests from Service Manager.</p> <p><b>Note:</b> This field is automatically completed and has a fixed value. Do not change it.</p>
omi.eventdetail.baseurl	http://<servername>/opr-console/opr-evt-details.jsp?eventId=	<p>The basic URL address of the event detail page in OMi. Replace &lt;servername&gt; with the fully qualified domain name of your OMi server.</p>

9. On the **General Parameters** and **Secure Parameters** tabs, enter three parameter values that you specified when configuring the Service Manager server as a connected server in BSM OMi. The following table lists the parameters, whose values you can copy from your BSM OMi server.

To copy the parameter values from BSM OMi, follow these steps:

- a. Log on to BSM as a system administrator.
- b. Navigate to **Admin > Operations Management > Setup > Connected Servers**.

- c. Locate your ITSM Enterprise Suite server configuration entry and double-click anywhere on the entry pane.
- d. On the **General** tab, copy the **ID** string at the bottom into the **omi.mgr.id** field in Service Manager.
- e. On the **Incoming Connection** tab, copy the **User Name** and **Password** to the **username** and **Password** fields in Service Manager, respectively.

Field	Sample Value	Description
omi.mgr.id (on the <b>General Parameters</b> tab)	f3832ff4- a6b9-4228- 9fed- b79105afa3e4	<p>The Universally Unique Identifier (UUID) automatically generated in OMi for the target Service Manager server.</p> <p><b>Note:</b> This parameter was introduced to support multiple OMi servers. Service Manager uses the UUID to identify from which OMi server an incident was opened. Be aware that if you delete the connected server configuration for the Service Manager server in OMi and then recreate the same configuration, OMi generates a new UUID. You need to reconfigure the integration instance by changing the old UUID to the new one.</p> <p><b>Tip:</b> If you have only one OMi server, you can simply remove this parameter (remove both the parameter name and value) from the integration instance. See "<a href="#">Support of multiple OMi servers</a>" on page 467.</p>
username omi.mgr.id (on the <b>General Parameters</b> tab)	SM_Server	This is the user name that the ITSM Enterprise Suite server uses to synchronize incident changes back to the OMi server.
Password (on the <b>Secure Parameters</b> tab)	SM_Server_ Password	This is the password that the ITSM Enterprise Suite server uses to synchronize incident changes back to the OMi server.

10. Click **Next** twice, and then click **Finish**.

**Note:** Leave the Integration Instance Mapping and Integration Instance Fields settings blank. This integration does not use these settings.

Service Manager creates the instance. You can edit, enable, disable, or delete it in Integration Manager.

11. Enable the integration instance.
12. If you have multiple OMi servers, repeat the steps above for the rest of your OMi servers.

Next, you can optionally enable Lightweight Single Sign-On (LW-SSO) in both BSM and Service Manager so that users can bypass the log-in prompts. For details, see ["Enable LW-SSO for the Incident Exchange \(OMi - SM\) integration" below](#).

## Enable LW-SSO for the Incident Exchange (OMi - SM) integration

### **Applies to User Roles:**

System Administrator

Lightweight Single Sign-On (LW-SSO) is optional but recommended for the Incident Exchange (OMi - SM) integration. You have different LW-SSO configuration choices depending on your needs. The following describes how LW-SSO can be used in the Incident Exchange (OMi - SM) integration workflow.

### When OMi creates an incident from an OMi event record

OMi creates an incident from an OMi event record by sending RESTful-based requests to Service Manager. The incident ID is then stored in the event record.

*LW-SSO is NOT needed in this process.* A dedicated Service Manager user account was specified when configuring the Service Manager integration in OMi. OMi uses this dedicated user account when calling the Service Manager RESTful Web Service to create the incident.

### When an OMi user views the incident details

The user can log in to Service Manager and view the incident details using the incident ID stored in the event record. For more information, refer to the BSM documentation.

If the user wants to view the incident details by clicking the incident link from the event record, LW-SSO can be used; otherwise a Service Manager login prompt will appear.

*LW-SSO is optional for this process.* To enable LW-SSO for this process, configure LW-SSO in both the Service Manager server and Web tier (because the server needs to trust the Web tier), as well as in BSM.

## When Service Manager synchronizes the OMi incident status back to OMi

When a user has updated the OMi incident, Service Manager calls the OMi server's RESTful Web Service to update the incident changes to the OMi event record.

*LW-SSO is NOT needed in this process.* A dedicated OMi user account was specified when the Incident Exchange (OMi - SM) integration was set up in SMIS, and Service Manager uses this user account when calling the OMi server's RESTful Web Service to synchronize the incident status back to the OMi event record.

## When a user views the event details from the OMi incident

The user clicks the **View OMi Event** option from the incident to view the event details.

*LW-SSO is optional for this process.* If you enable LW-SSO in the Service Manager Web tier and in BSM, the BSM login prompt is bypassed.

## Configure automatic closure for OMi incidents

### Applies to User Roles:

System Administrator

OMi incidents can be automatically closed after a predefined amount of time since they were last updated (or resolved if they have not been updated after being resolved).

The workflow is as follows:

1. An incident is opened from OMi.
2. If the **Schedule Condition** is met, the system creates a schedule record for the incident. The schedule record will expire at a future time based on the **Calc Expression**.
3. A user updates the incident and saves the changes.
4. The **Reset alerts if** expression on the **Alerts** tab of the **probsummary** object definition is evaluated. If it evaluates to true, the Expiration time of the schedule record is updated based on

the Calc Expression. By default, the expiration time of the schedule record is updated only when the incident has a category of **incident**.

5. When the schedule record expires, the **Alert Condition** is evaluated. If it evaluates to true, the incident is automatically closed.

To enable automatic closure for OMi incidents:

1. Configure the global settings in the Incident Management Environment record.
  - a. Click **System Administration > Ongoing Maintenance > Environment Records > Incident Management Environment**.
  - b. Change the following settings as necessary.

Field	Value
Close Incident Automatically?	This option disables or enables the automatic closure of OMi incidents at the global level. <ul style="list-style-type: none"> <li>○ If this option is not selected, no incidents will be automatically closed.</li> <li>○ If this option is selected, incidents will be automatically closed under specified conditions.</li> </ul> Default: Not selected
Closure Code	This value will be copied to the <b>Closure Code</b> field of incidents when they are automatically closed.                     Default: Automatically Closed
Solution	This description will be appended to the end of the <b>Solution</b> field of incidents when they are automatically closed.                     Default: This incident which belongs to OMi has been closed automatically.

- c. Click **Save**.
- d. Restart the Service Manager server.

**Note:** If you have made any changes to any of the configuration options in the Incident Management Environment record, the Service Manager server must be restarted for the changes to take effect.

2. Configure the alert definition that determines when an incident should be closed.

**Note:** The **alert** and **problem** processes must be running to enable the successful closure of OMi incidents.

- a. Click **Tailoring > Document Engine > Alerts**.
- b. In the Alert Name field, enter: **OMI Auto-Close**.
- c. Click **Search**. The OMI Auto-Close alert definition detail form opens.

**Caution:** These fields in the alert definition are used to control automatic closure of OMi incidents. You can change the default values of these fields. However, you must be aware of the risk that automatic closure of OMi incidents will not work properly if the **Schedule Condition** and **Alert Condition** fields are not configured correctly.

Field	Value
Schedule Condition	<p>This expression is used to determine if an incident should be scheduled for automatic closure.                      Default: <code>jscall("SMOMi.isAutoCloseAndResolved")</code>.</p> <p>An incident is scheduled for automatic closure when the following conditions are met.</p> <ul style="list-style-type: none"> <li>◦ The <b>Close Incident Automatically?</b> option in the Incident Management Environment record is selected.</li> <li>◦ In the incident record, the <b>Do not close this incident automatically</b> option is not selected.</li> <li>◦ The incident has a status of <b>Resolved</b>.</li> </ul>



Field	Value
Alert Condition	<p>This expression is evaluated when an incident is about to be automatically closed. If it evaluates to true, the incident is closed. Default: <code>jscall("SMOMi.isAutoCloseEnabled")</code>.</p> <p>An incident is closed when the following conditions are met.</p> <ul style="list-style-type: none"> <li>○ The <b>Close Incident Automatically?</b> option in the Incident Management Environment record is selected.</li> <li>○ In the incident record, the <b>Do not close this incident automatically</b> option is not selected.</li> </ul>
Calc Expression	<p>This expression is used to determine how much time will elapse before an incident is automatically closed.</p> <p>Default: <code>\$L.alert.time=update.time in \$L.file+'7 00:00:00'</code>.</p> <p>The default value means the amount of time elapsed is equal to seven days since the incident was last updated.</p>

3. Configure alert information in the **probsummary** object.  
The OMi autoclose alert definition is configured to only be used by OMi incidents. The closure time is reset each time the incident is updated. If the closure time is reached without the incident being updated then Service Manager will automatically close the incident.
  - a. Click **Tailoring > Document Engine > Objects**.
  - b. In the **File name** field, enter **probsummary** and press ENTER. The **probsummary** object definition is displayed.
  - c. Select the **Alerts** tab.

The **Reset alerts if** expression is used to reset the automatic closure time of OMi incidents.

Default: `category in $L.file="incident" and not null(1 in external.process.reference in $L.file)`.

## Change the default assignment group for OMi incidents

### Applies to User Roles:

System Administrator

HP Service Manager can accept REST based requests from BSM Operations Manager i (OMi) to create incidents based on events information in OMi. An incident opened from an OMi event is automatically assigned to an existing group based on the following field values, listed from the highest to lowest priority:

- The **Affected Service** of the incident
- The **Category** of the incident
- The **Affected CI** of the incident

However, if none of the above field values is available, the incident is assigned to a default group named **Application**. If necessary, you can change this default group setting as follows:

1. Navigate to **System Administration > Ongoing Maintenance > BDM Mapping Management**. The BDM mapping configuration search page opens.
2. Enter **incident** in the BDM Name field, select **1.1** in the **Version** field, and then click **Search**. The BDM mapping record **incident** is displayed. The Incident Exchange (OMi - SM) integration uses this BDM mapping record when creating an incident from an OMi event.
3. Select the **Field Mapping** tab, scroll down to the **assignment** field in the SM Object Field column, and click the **SM Callback** field in the same row.
4. Change **Application** in the following code to the name of another assignment group:

```
4) A default assignment group if no other criteria is met
    if( ! $result ) {
        $result = "Application";
    }
```

5. Click **Save**. The default assignment group is now changed.

## Synchronization of incident changes back to Operations Manager i (OMi)

After OMi opens an incident in HP Service Manager, Service Manager will synchronize the incident changes back to OMi.

Operations Manager i (OMi) can forward an event record to Service Manager as an incident by calling a Service Manager Web Service. The incident ID is then stored in the event record.

When a user has updated the incident opened from OMi, Service Manager calls an OMi server RESTful Web Service to update the incident changes to the OMi event record.

If Service Manager fails to synchronize the incident changes back to OMi for some reasons (for example, because of a network problem), Service Manager behaves as follows:

- Displays a warning message, indicating that the incident failed to be synchronized to OMi.
- Saves the failed task in the SMIS task queue, and retries the task to re-synchronize the changes to OMi based on an interval time and a maximum retry times configured when adding the Incident Exchange (OMi - SM) integration in SMIS. When the re-synchronization is successfully completed, the failed task is removed from the task queue.

System Administrators can monitor failed tasks, and reset their retry times or rerun expired tasks.

## Working with the Incident Exchange (OMi - SM) integration

Once the integration is set up, Service Manager Incident Management users with the right permissions can view event details from OMi incidents and mark individual OMi incidents as eligible or ineligible for automatic closure (if their system administrator has enabled automatic closure for OMi incidents).

### View related OMi event details from an incident

If all of the following conditions are met, you can view the related Operations Manager i (OMi) event details from an OMi incident:

- You are accessing the incident through the HP Service Manager standard Web client (not the employee self-service (ESS) interface).
- One or more SMOMi integration instances are set up and enabled in Integration Manager.
- You are also a Business Service Management (BSM) user who has been granted the permission **Events assigned to user** including the required actions.

To view related OMi event details from an OMi incident:

1. Log on to the Service Manager Web client.
2. From **Incident Management**, search for the incident record created from OMi.
3. Click **More** and then select **View OMi Event**.

**Note:** The **View OMi Event** option displays only when the **omi.mgr.id** parameter in the corresponding SMOMi integration instance is set correctly.

If Lightweight Single Sign-On (LW-SSO) is enabled in both the Service Manager Web client and HP Business Service Management (BSM) , the OMi event detail page opens in a new browser window, displaying the details of the related event in OMi. If LW-SSO is not enabled, a BSM login page opens, and the related OMi event detail page displays after you log on to BSM.

## Mark an incident for automatic closure

### **Applies to User Roles:**

Incident Coordinator

You can mark an OMi incident as eligible or ineligible for automatic closure after a predefined amount of time since it was last updated.

To mark an incident as ineligible or eligible for automatic closure:

1. From **Incident Management**, search for an incident record opened from OMi.
2. Select or deselect the **Do not close this incident automatically** check box to mark this incident as ineligible or eligible for automatic closure.

**Note:** By default, this check box is not selected.

## Operations Manager i - Service Manager Integration

**Note:** The following sections are excerpted from the integrations section of the Operations Manager i Help Center. However, to ensure you have the latest information, you should reference the latest published version of the document in question.

# Operations Manager i - Service Manager Integration Overview

You can integrate HP Service Manager (SM) with one or more of the OMi components, as described below. Each integration can be performed separately.

**Note:** In general, the information provided in this guide is for integrating OMi with SM 9.3x.

For instructions on integrating OMi with earlier versions of SM, see [http://support.openview.hp.com/selfsolve/document/KM1303768/binary/BSM9.12\\_SM\\_Integration\\_Interactive\\_Docs.html](http://support.openview.hp.com/selfsolve/document/KM1303768/binary/BSM9.12_SM_Integration_Interactive_Docs.html). Download and extract the zip file contents, and then open the **SM\_interactive\_document.htm** file and follow the guidelines.

The options are as follows:

- **Downtime exchange between OMi and SM.** OMi enables you to forward downtimes (also known as outages) from OMi to SM, and from SM to OMi. The downtime defined in OMi is converted to a request for change in SM, and vice versa. For details, see "[Downtime Exchange Between Operations Manager i and Service Manager](#)" on page 484.
- **Incident exchange between SM and OMi.** OMi enables you to forward events from OMi to SM. Forwarded events and subsequent event changes are synchronized back from SM to OMi. You can also drill down from OMi events to SM incidents. For details, see "[Incident Exchange Between Service Manager and Operations Manager i](#)" on page 490.
- **View planned changes and incident details in Service Health.** This integration enables you to view planned changes and incident details in the Changes and Incidents tab in the 360° View page in Service Health. For details, see "[View Changes and Incidents in Service Health Using Standalone HP Universal CMDB](#)" on page 506 and "[View Changes and Incidents in Service Health Using RTSM](#)" on page 522.
- **Submit an incident through OMi alerts.** Incidents are automatically opened in SM when a CI Status alert is triggered in OMi. For details, see "[Generate Incidents in SM When an OMi Alert is Triggered](#)" on page 534.
- The **Business Impact Report** integration is described in the *Closed Loop Incident Process (CLIP) Guide*. When deployed as part of the OMi solution, Incident Management users can launch an impact report from an incident in context with the incident's affected CI. Service Desk Agents can validate the updated status of the Business Impact to categorize and prioritize the incident accordingly. For details, see the CLIP page in the Solutions Portal at:

<http://support.openview.hp.com/sc/solutions/index.jsp#tab=tab1>

**Note:**

- **Service Manager Query Security.** If you have set up an integration from OMi to SM, there is a CI context menu that enables you to access SM from OMi Service Health. This drill-down option is not available if you have enabled Service Manager query security.
- **Troubleshooting Multiple Domains.** If OMi and SM are in different domains, and you are using Internet Explorer as your browser, you may need to add the domains to the list of allowed domains in the Privacy tab (**Internet Options > Privacy > Sites**).

# Downtime Exchange Between Operations Manager i and Service Manager

OMi enables you to forward downtimes (also known as outages) from OMi to SM, and from SM to OMi. The downtime defined in OMi is converted to an incident in SM, and vice versa.

This chapter includes the following:

- ["Integration Overview" below](#)
- ["Prerequisites" on the next page](#)
- ["Step 1: Send OMi Downtime Events to SM" on the next page](#)
- ["Step 2: Integrate SM Downtimes with OMi" on page 487](#)

## Integration Overview

The downtime integration between OMi and SM includes information exchanges in both of the following directions:

- **SM > OMi.** When you create a downtime RfC (request for change) in SM, the RfC includes the CI that is under change and a start and end date/time of the downtime. If you do not want to waste effort with false alarms in your operations center, and do not want to have these times included in service availability reports, you can set up the integration so that these RfCs are translated to downtimes in OMi.

In this scenario, you install and set up a downtime adapter on your CMDB (depending on your setup, you might be working with a uCMDB as central CMS, or with an RTSM contained in your OMi). The RfC creates a planned downtime CI in the CMDB, and the adapter sends the planned downtime CI to OMi to create a downtime.

- **OMi > SM.** When you define downtimes using OMi (for example, every Monday and Saturday from 8:30 PM-9:30 PM) to proactively support end users, the help desk should be aware of such operational downtimes. After you set up the integration, downtimes in OMi trigger events, which create corresponding incidents in SM.

In this scenario, when a downtime starts, OMi generates an event. Using the event forwarding mechanism, the event generates an incident in SM. When the downtime ends, an event is sent to close the downtime incident.



A single downtime can be defined on more than one CI. In the case of OMi > SM, a separate event is sent for each CI in the downtime.

## Prerequisites

### Supported Platforms

To set up the downtime integration, you must meet the following prerequisites:

- Service Manager 9.31 and higher.
- uCMDB (RTSM/CMS) 9.05 CUP 5 and higher with content pack 11 update 2, or uCMDB 10.01 or higher with content pack 12.
- Before deploying the adapter, verify that CP11 or higher is installed. If it is not, install the content pack.
- If the adapter is installed on the RTSM, and the adapter is working behind a reverse proxy, the DPS must have the correct certificates installed to send requests to the reverse proxy.
- If you are using CMS, make sure that the CMS integration is set up.

### Global ID Generator

To enable the downtime integration, you must have a global ID generator configured in your environment.

If you are working with RTSM, perform the following to configure the global ID generator:

1. Access the following location with your browser: `http://<DPS name>:21212/jmx-console/HtmlAdaptor?action=inspectMBean&name=UCMDB:service=Multiple CMDB Instances Services`
2. In the **setAsGlobalIdGenerator()** method, fill the **customerID** parameter with the value of **1**, and click **Invoke**.

## Step 1: Send OMi Downtime Events to SM

To enable OMi to send downtime definitions to SM, you must edit an infrastructure setting as described below. This procedure generates events in OMi. You can then use the event forwarding mechanism to generate incidents in SM when a downtime in OMi begins and ends.

1. Access the following location in OMi:

**Administration > Setup and Maintenance > Infrastructure Settings > Foundations > Downtime**

2. Change the value of the **Downtime Send Event** parameter to **true**.
3. Restart your OMi services on all Gateway Servers and Data Processing Servers.

A corresponding forwarding rule that configures forwarding downtime start and end events from OMi to SM should be configured in the Event Forwarding Rule dialog box. The forwarding rule should be based on the ETI Hint, as follows:

- ETI Hint equals ignore case “downtime:start”
- ETI Hint equals ignore case “downtime:end”

For details on how to use the event forwarding mechanism to generate incidents in SM, see the OMi Administration Guide.

Downtime events use the following formats:

- **Downtime Start**

Event field	OMi Downtime
Severity	Normal
Category	Downtime Notification
Title	Downtime for <CI Type><Affected CI Name>started at <Downtime Start Time>
Key	<OMi Downtime ID>:<Affected CI ID>:downtime-start
SubmitCloseKey	False
OutageStartTime	<Downtime Start Time>
OutageEndTime	<Downtime End Time>
CiName	<Affected CI Name>
Ciid	<Affected CI Global ID>
CiHint	GUCMDB:<Affected CI Global ID> UCMDB:<Affected CI ID>
HostHint	GUCMDB:<Related Host Global ID> UCMDB:<Related Host ID>
EtiHint	downtime:start

- **Downtime End**

Event field	OMi Downtime
Severity	Normal
Category	Downtime Notification
Title	Downtime for <CI Type><Affected CI Name> ended at < Downtime End Time>
Key	<OMi Downtime ID>:<Affected CI ID>:downtime-stop
SubmitCloseKey	true
CloseKeyPattern	<OMi Downtime ID>:<Affected CI ID>:downtime-start
EtiHint	downtime:end
LogOnly	true

## Step 2: Integrate SM Downtimes with OMi

To enable downtimes defined in SM to be sent to OMi, you must add an integration adapter to the uCMDB where downtimes are defined.

**Important:**

- Following the initial integration, a large amount of data may be communicated from SM to OMi. It is highly recommended that you perform this procedure during off-hours, to prevent negative impact on system performance.
- The integration consists of two parts: SM > CMS/RTSM, and CMS/RTSM > OMi adapter. You should configure both parts of the integration as one flow, without a significant time lag between setting up the two parts. If you set up the SM > CMS/RTSM part, and then wait a long time before setting up the CMS/RTSM > OMi adapter part, the number of downtimes communicated to OMi initially may be extremely high.

**Note:**

- The following procedure does not describe the SM > CMS/RTSM connection setup. SM should be configured to create its CIs in the CMS. This procedure connects the adapter between the

CMS/RTSM and OMi.

- The default job synch frequency is one minute.

Create a new integration point as follows:

1. Create the integration point credentials:

- a. If you have OMi, do the following on your OMi. If you use a CMS, do the following on your CMS:  
Access the Data Flow Probe Setup:

**Administration > RTSM Administration > Data Flow Management > Data Flow Probe Setup**

Alternatively, click [Data Flow Probe Setup](#).

**Note:** You do not need a probe to perform this integration. Nevertheless you create credentials using the Data Flow Probe Setup tab.

- b. Click **Add domain or probe**, and enter a name and description of your choice.
- c. Expand the submenus and select **HTTP protocol**.
- d. Click the + sign (**Add new connection details**) and enter the OMi Gateway host name, Port 80, and the OMi username and password. Leave the **Trust** fields blank. When you are done, click **OK** to save the credentials.

2. Create a new integration point:

- a. If you have OMi, do the following on your OMi. If you use a CMS, do the following on your CMS:  
Access the Integration Studio:

**Administration > RTSM Administration > Data Flow Management > Integration Studio**

Alternatively, click [Integration Studio](#).

- b. Click **New Integration Point**, enter a name and description of your choice, and select **BSMDowntimeAdapter/SM scheduled Downtime Integration into BSM**.
- c. Enter the following information for the adapter: OMi Gateway hostname and port, the integration point credentials you just created, communication protocol, and the context root

(if you have a non-default context root).

- d. Click **OK**, then click the **Save** button above the list of the integration points.
3. You can use the **Statistics** tab in the lower pane to track the number of downtimes that are created or updated. By default, the integration job runs every minute. If a job has failed, you can open the **Query Status** tab and double-click the failed job to see more details on the error.

If there is an authentication error, verify the OMi credentials entered for the integration point.

If you receive an unclear error message with error code, this generally indicates a communication problem. Check the communication with OMi. If no communication problem is found, restart the **MercuryAS** process.

A failed job will be repeated until the problem is fixed.

# Incident Exchange Between Service Manager and Operations Manager i

OMi enables you to forward events from OMi to SM. Forwarded events and subsequent event changes are synchronized back from SM to OMi. You can also drill down from OMi events to SM incidents.

**Note:** HP recommends this integration option for new integrations with SM. However, existing integrations that use other integration options are still supported.

This chapter includes the following:

- ["Step 1: Configure the SM Server as a Connected Server" below](#)
- ["Step 2: Configure an Event Forwarding Rule" on page 494](#)
- ["Step 3: Configure a URL Launch of the Event Browser from SM" on page 496](#)
- ["Step 4: Configure a URL Launch of SM from the Event Browser" on page 497](#)
- ["Step 5: Configure the SM Server" on page 498](#)
- ["Step 6: Mapping and Customization" on page 499](#)
- ["Step 7: Test the Connection" on page 500](#)
- ["Step 8: Synchronize Attributes" on page 501](#)
- ["Tips for Customizing Groovy Scripts" on page 502](#)

## Step 1: Configure the SM Server as a Connected Server

Synchronizing events and event changes between OMi and SM incidents requires configuring a connected server within OMi to correctly identify the target SM instance. The first step to achieve this is to configure HP Service Manager as a target connected server in the Connected Servers manager.

For full details about how to configure a connected server, see the OMi Administration Guide.

**Note:** Before you continue with the below procedure, set up an integration user with a user name and password in SM. This is the user name and password needed by OMi to access the SM target server.

To configure the SM server as a target connected server, perform the following steps:

1. Navigate to the Connected Servers manager:

**Administration > Setup and Maintenance > Connected Servers**

Alternatively, click [Connected Servers](#).

2. Click the New button and select **External Event Processing**. The Create New Server Connection - External Event Processing dialog box opens.
3. In the General page, in the **Display Name** field, enter a name for the target SM server. By default, the Name field is filled automatically. For example, if you enter `Service Manager 1` as the Display Name for the target SM server, `Service_Manager_1` is automatically inserted in the Name field. You can specify your own name in the Name field, if you want to change it from the one suggested automatically.

**Note:** Make a note of the name of the new target server (in this example, `Service_Manager_1`). You need to provide it later as the `username` when configuring the SM server to communicate with the server hosting OMi.

*Optional:* Enter a description for the new target server.

Make sure that you select the **Active** check box.

Click **Next** to open the Server Properties page.

4. In the Server Properties page, enter the Fully Qualified DNS Name of the SM target server.

Click **Next** to open the Integration Type page.

5. In the Integration Type page, complete the following information:
  - a. Select **Call Script Adapter** as the integration type.
  - b. From the Script Name menu, select the SM Groovy script adapter **sm:ServiceManagerAdapter**.

**Note:** In case the **Test Connection** fails and the error indicates that there could be a problem with timeout, increase the timeout value. Otherwise, you can leave the default

timeout value.

- c. Click **Next** to open the Outgoing Connection page.
6. In the Outgoing Connection page, enter the credentials (user name, password, and port number) required to access the SM target server and to forward events to that server:
  - a. In the **User Name** field, enter the user name for the integration user you set up in SM.
  - b. In the **Password** field, enter the password for the user you specified. Repeat the password entry in the **Verify Password** field.
  - c. In the **Port** field, specify the port configured on the SM side for the integration with OMi.

To find the port number to enter:

- If you are using default ports in SM, select or clear **Use Secure HTTP** as appropriate, and then click **Set default port**. The port is set automatically.

**Note:** If you do not want to use secure HTTP, make sure that the **Use secure HTTP** check box is cleared.

If the Use Secure HTTP check box is selected, download and install a copy of the target server's SSL certificate using the **Retrieve from Server** or **Import from File** link, if the certificate is available in a local file.

- If you need to find the port number, access the following file on your SM system:

```
<HP Service Manager root directory>/HP/Service Manager
<version>/Server/RUN/sm.cfg
```

In the `sm.cfg` file, check for the `sm -loadBalancer` line and add the port entry at the end of the line. The line looks similar to this:

```
sm -loadBalancer -httpPort:13080
```

Enter the appropriate value of the port used by SM in the **Port** field of the Outgoing Connection page.

- d. Select the **Enable Synchronize and Transfer Control** check box.



If the Enable Synchronize and Transfer Control check box is selected, an OMi operator can transfer ownership of the event to the target connected server using the Transfer Control option in the Event Browser context menu.

If it is not selected, the Synchronize and Transfer Control option is not available from the Event Browser context menu or from the list of forwarding types for configuring forwarding rules.

- e. Test the connection by clicking the **Test Connection** link in the upper center of the dialog box. A **Success** or **ERROR** hyperlink is displayed. Click the link to get a more detailed message.
  - f. Click **Next** to open the Event Drilldown page.
7. If you want to drill down into SM, in addition to automatically generating SM incidents from OMi events, you need to specify the fully qualified DNS name and port of the SM system into which you want to perform the incident drilldown.

**Note:** To enable incident drilldown to SM, you must install a web tier client for your SM server according to your SM server installation or configuration instructions.

In the **Event Drilldown** page, configure the server where you installed the web tier client along with the configured port used.

If you do not specify a server in the Event Drilldown page, it is assumed that the web tier client is installed on the server used for forwarding events and event changes to SM, and receiving event changes back from SM.

If nothing is configured in the Event Drilldown dialog box, and the web tier client is not installed on the SM server machine, the web browser will not be able to find the requested URL.

Select or clear the **Use Secure HTTP** check box according to your configuration. In case the SM server is configured for SSL access, click the **Retrieve from Server** link to import the certificate from the SM server to allow SSL encryption.

Click **Next** to open the Incoming Connection page.

8. To enable event changes to be synchronized back from SM to OMi, you must provide credentials for the SM server to access the server hosting OMi.
- a. In the Incoming Connection page, select the **Accept event changes from external event processing server** check box, and then enter a password that the SM server requires to

connect to the server hosting OMi.

**Note:** Make a note of this password. You need to provide it later when configuring the SM server to communicate with the server hosting OMi. This password is associated with the server name (*Service\_Manager\_1*) you configured in SM.

If **Enable Synchronize and Transfer Control** was previously selected, the **Accept event changes from external event processing server** option is assumed and cannot be disabled.

- b. Click **Finish**. The target SM server appears in the list of Connected Servers.
9. If you have SM 9.34 or higher, perform the following additional steps:
    - a. Reopen the SM connected server that you configured in the previous steps. To do so, double-click the connected server entry in the connected servers list.
    - b. Copy the ID of the connected server (displayed in the lower right corner of the General tab) and save it. You need to specify this ID as `omi.mgr.id` on the SM system.

An example of a connected server ID is as follows:

**ID: 22f42836-fd36-473e-afc9-a81290f4f73b**

## Step 2: Configure an Event Forwarding Rule

The next step is to configure an event forwarding rule that determines which events are forwarded automatically to SM.

For details about configuring filters, see the OMi Administration Guide.

To configure a forwarding rule, follow these steps:

1. Navigate to the Forwarding Rules manager:

**Administration > Event Processing > Automation > Event Forwarding**

Alternatively, click [Event Forwarding](#).

2. Click the **New Item** button to open the Create New Event Forwarding Rule dialog box.
3. Under **General**, in the **Display Name** field, enter a name for the forwarding rule, in this example

Forward Critical (Sync and Transfer Control).

*Optional.* Enter a description for the forwarding rule you are creating.

4. Under Condition, click the browse button next to the Event Filter field. The Select an Event Filter dialog box opens.

In the Select an Event Filter dialog box, do one of the following:

- Select an existing filter
- Create a new filter as follows:
  - i. Click the **New** button to open the Filter Configuration dialog box. You can choose between **New Simple Filter** or **New Advanced Filter**.
  - ii. In the **Display Name** field, enter a name for the new filter, in this example, **FilterCritical**.

Clear the check boxes for all severity levels except for the severity Critical.

Click **OK**.

- iii. You should see your new filter in the Select an Event Filter dialog box (select it, if it is not already highlighted).

Click **OK**.

5. Under Target Servers, select the target connected server you configured in "[Step 1: Configure the SM Server as a Connected Server](#)" on page 490. In this example, this is Service Manager 1.

Click the **Add** button next to the target servers selection field. You can now see the connected server's details. In the **Forwarding Type** field, select the **Synchronize and Transfer Control** forwarding type. Although other selections are technically possible, only **Synchronize and Transfer Control** is supported by SM.

Click **OK**.

6. Make sure the **Activate Rule after creation** check box is selected. A rule must be active in order for its status to be available in SM.

## Step 3: Configure a URL Launch of the Event Browser from SM

Before operators are able to perform event drill-down from SM into the OMi user interface using a URL launch of the Event Browser, the operators must be set up as valid users with appropriate permissions in OMi:

### User account requirements

- If Single Sign-On (SSO) authentication is configured, set up each user in OMi with the *same* user name that is used by the SM operator to log on to SM and to perform the URL call. (The password of each OMi user can be any string, but not empty.) After successfully logging on to SM, the OMi users can launch the OMi Event Browser without further authentication.

For details on setting up SSO, see **System Administration > Integrations > Service Manager integration methods and tools > Integration Manager > Using LW-SSO with integrations** in the SM documentation library.

- If SM is not configured to use SSO authentication, set up each user with the *same* user name that is used by the SM operator and specify a valid password. The users are required to enter their user name and password when launching the OMi Event Browser.

### Required user permissions

You must grant the permission `Events assigned to user` including the required actions to each OMi user. To do so, select:

#### Administration > Users > Users, Groups, and Roles

Alternatively, click [Users, Groups, and Roles](#).

Select a role or create a new one. In the Permissions section, go to the **Operations Console** category, select **Events** and specify the actions users can perform on **Events assigned to user**.

You can optionally grant the permission to view events not assigned to each user.

**Note:** Without valid user names, or if a user does not have the required viewing permissions, any attempt to perform a URL launch of the OMi Event Browser from SM results in an empty browser window.

## Step 4: Configure a URL Launch of SM from the Event Browser

To be able to perform a URL launch of SM from the OMi Event Browser using the web tier client, perform the following:

1. Navigate to Connected Servers:

**Administration > Setup and Maintenance > Connected Servers**

Alternatively, click [Connected Servers](#).

Click the **Manage Scripts** icon.

2. Select the **sm:ServiceManagerAdapter** script, and click the **Edit Item** button.
3. Click the **Script** tab and locate the following text in the Groovy script:

```
private static final String SM_WEB_TIER_NAME = 'webtier-9.30'
```

4. Change the value of `webtier-9.30` to the value required to access the SM web tier client.

The drill-down URL is made up like this:

```
http://<FQDN of HP Service Manager web tier server>/<web path to HP Service Manager>/<URL query parameters>
```

In this instance, `<FQDN of HP Service Manager web tier server>` is the fully qualified DNS name of the SM server where the web tier client is installed. This part of the URL is added automatically (together with `http://` or `https://`) according to the values that you provided when you configured SM as a target connected server in the Connected Servers manager. For details, see "[Step 1: Configure the SM Server as a Connected Server](#)" on page 490.

An example of a drill-down URL:

```
http://smserver.example.com/SM930/index.do?ctx=docEngine&file=probsummary&query=number%3D%22IM10216%22&queryHash=bf52f465
```

In this example, you need to replace `webtier-9.30` with `SM930`. All the other parts of the URL are configured automatically.

5. When finished editing, save the new version of the script. Note that the script can always be

reverted to its original version.

For details, see the OMi Administration Guide.

6. If you are using SM 9.34 or lower, set the value of the `querySecurity` parameter from the default value (`true`) to `false` in the SM web tier configuration file `web.xml`.

For more details, see the HP Service Manager online help:

**Guides and reference > System Configuration Parameters > Security parameters > Parameter: querysecurity**

and

**Guides and reference > System Configuration Parameters > Client parameters for Web clients > Web parameter: querySecurity**

## Step 5: Configure the SM Server

The next step describes how to configure the SM server to integrate with OMi.

**Note:** If you want to configure more than one OMi server, you need to increase the connection count on the SM side:

**System Administration > Integrations > HP Business Service Management (BSM) > Incident Exchange (OMi - SM) integration > Incident Exchange (OMi - SM) integration setup > Configure the Instance Count setting in the SM-OMi integration template.**

This functionality is only available for SM 9.34 and higher.

To configure the SM server, complete the following steps in the HP Service Manager user interface:

1. From the left hand pane, navigate to:

**Tailoring > Integration Manager**

2. Click **Add** to add a new configuration.
3. Select the **SMOMi** integration template from the Integration Template field. Click **Next**.
4. *Optional.* Change the log level to the desired value.

*Optional.* Change the description, for example, to This is for SMOMi integration.

Specify **Interval Time (s)** and **Max Retry Times**, and then click **Next**.

5. In the General Parameters tab, replace the existing entries with the following values:

Name	Value	Category
omi.server.url	http://<Omi_gateway_ FQDN>:<port>/ opr-gateway/rest/ synchronization/event/	General
username	Service_Manager_1  (This is the name of the SM target server you configured in " <a href="#">Step 1: Configure the SM Server as a Connected Server</a> " on page 490).	Header
omi.eventdetail.baseurl	http://<Omi_gateway_ FQDN>:<port>/ opr-web/opr-evt-details.jsp? eventId=	General
omi.mgr.id	The ID obtained when reentering the Connected Servers window.	General

6. In the Secure Parameters tab, set the password to the one you specified in the Incoming Connection page when configuring the target connected server in "[Step 1: Configure the SM Server as a Connected Server](#)" on page 490. In our example, this is HPqwer1\_.

Click **Next**.

7. In the Integration Instance Fields dialog box, click **Next**.
8. In the Integration Instance Mapping dialog box, click **Finish**.

**Note:** Ensure that the rule is active. To make the rule active, select the rule and click **Enable**.

## Step 6: Mapping and Customization

You can add your own custom attributes in the Groovy script selected for the SM server in the Connected Servers pane, and then map these custom attributes to the appropriate field in SM. For

details about groovy scripting, see the OMi Extensibility Guide.

You can also change how attributes are mapped from OMi to SM. The mapping is done in the BDM Mapping Manager in SM:

**System Administration > Ongoing Maintenance > BDM Mapping Management**

For details about mapping attributes, see the HP Service Manager online help:

**System Administration > Integrations > Service Manager integration methods and tools > BDM Mapping Management**

## Step 7: Test the Connection

To test the connection, send an event to the server hosting OMi that matches the filter you defined (in our example filter, the severity value is *Critical*), and then verify that the event is forwarded to SM as expected.

To test the connection, do the following:

1. On the Gateway Server system running OMi, open an Event Browser.
2. On the system running OMi, open a command prompt and change to the following directory:

```
<OMi_HOME>\opr\support
```

3. Send an event using the following command:

```
sendevent -s critical -t test111-1
```

4. Verify that the event appears in the OMi Event Browser.
5. Select the **Forwarding** tab.
6. In the External Id field, you should see a valid SM incident ID.
7. Verify that the incident appears in the Incident Details in HP Service Manager:

If the event drill-down connection is configured correctly, click the hyperlink created with the incident ID. A browser window opens, which takes you directly to the incident in the Incident Details in HP Service Manager.

If the event drill-down connection is not configured, do the following:



- a. In the Forwarding tab in the OMi Event Browser, copy or note the incident ID from the External Id field.
  - b. In the HP Service Manager user interface, navigate to:  
**Incident Management > Search Incidents**
  - c. Paste or enter the incident ID in the Incident Id field.
  - d. Click the **Search** button. This takes you to the incident in the Incident Details.
8. Close the incident in HP Service Manager.
  9. Verify that the change in the state of the incident (it is now `closed`) is synchronized back to OMi. You should not be able to see the event that was closed in SM in the active Event Browser, but it should now be in the History Browser.

## Step 8: Synchronize Attributes

Not all attributes are synchronized back from SM to OMi by default. When the SM incident is initially created from an OMi event, event attributes are mapped to the corresponding SM incident attribute. Out of the box, after the initial incident creation, whenever the incident or event subsequently changes, only a subset of the changed event and incident attributes are synchronized. The following describes how to customize the list of attributes to synchronize upon change.

### **Unidirectional Synchronization: OMi to SM**

The following attributes are transferred to SM from OMi on a one-time basis, that is, when the event was initially created, and the transfer of control of the event was configured in the Connected Servers manager.

These attributes support bidirectional synchronization, but are disabled out-of-the-box:

- Title
- Severity
- Priority
- Operator: the operator assigned to the event who forwarded the event
- Category

- Subcategory
- Related CI

For the above attributes, there is no back synchronization from SM to OMi.

### **Bidirectional Synchronization**

Attributes that support bidirectional synchronization between OMi and SM are:

- Description
- Lifecycle state (the state is only updated when the state changes to closed)
- Solution
- OMi event annotations are synchronized to SM activity log
- Contents under the Forwarding tab in the Event Details

### **Attribute Synchronization using Groovy Scripts**

If you want to change the out-of-the-box behavior regarding which attributes are updated, you can specify this in the Groovy script used on the OMi side for synchronization or incident creation. In the Groovy script, you can specify which fields are updated in SM, and which fields are updated in OMi. You can also specify custom attributes in the Groovy script.

### **Tips for Customizing Groovy Scripts**

This section provides some tips about customizing Groovy scripts. It contains a few selected examples of what you can customize. To see further items that can be modified, see the configuration section of a Groovy script.

In the configuration section of the Groovy script, you can define and modify the attributes that are to be synchronized between OMi and SM. The configuration section of the Groovy script also contains the default value mappings for lifecycle state, severity, and priority. You can also modify these, and it is possible to define the mappings for in-going and out-going requests differently.

More advanced configuration can be done in other parts of the Groovy script if required.

The beginning and the end of the configuration section of the Groovy script is marked as follows:

```
//  
// configuration section to customize the Groovy script
```

```
// BEGIN
...
...
//
// configuration section to customize the Groovy script
// END
```

**Note:** Modifications to Groovy scripts are not overwritten by patches and hotfixes. Your customized version of a script will remain after an update or a patch. If you want to use the newer version of a script, make a copy of your version, revert back to the predefined version, and then reapply your changes.

The mapping from OMi to SM is compliant to BDM 1.1 incident web service specifications. The mapping of the BDM 1.1 incident web service to SM is specified in SM in the BDM Mapping Manager. For more information about the BDM Mapping Manager, see the BDM Mapping Manager section of the HP Service Manager online help.

## Controlling Attribute Synchronization

You can control how updates to certain attributes are synchronized between OMi and SM by setting some Boolean variables to `true` or `false`.

Examples:

- `SyncAllProperties` variable. By default, it is `false`. If you set it to `true`, all properties will be synchronized in both directions. The other variables will be ignored.
- `private static final SyncTitleToSMOnUpdate = false;`

This line of the Groovy script disables the synchronization of changes to the title made in OMi to SM.

- `private static final Boolean SyncTitleToOPROnUpdate = false;`

This line of the Groovy script disables the synchronization changes to the title made in SM to OMi.

The title is a required attribute in SM, and it is set, independently of the flags above, using the title given in OMi during the creation of the incident.

## Mapping OPR Lifecycle States to BDM Lifecycle States

Individual OPR event state and SM incident status changes may be selected for synchronization. Out of

the box, only the "closed" state is synchronized in both directions. To change this behavior, add the desired states to the appropriate list, `SyncOPRStatesToSM` or `SyncSMStatusToOPR`.

#### Examples:

- ```
private static final Set SyncOPRStatesToSM = ["closed", "in_progress", "resolved"]
```
- ```
private static final Set SyncSMStatusToOPR = ["closed", "resolved"]
```

In the example, the OPR event lifecycle states `closed`, `in_progress`, and `resolved` are synchronized to the SM incident status, and SM incident statuses `closed` and `resolved` are synchronized to the OPR event state.

**Note:** The special state "\*" denotes all states, so to synchronize all OPR event states to the SM incident status property, specify the following:

```
private static final Set SyncOPRStatesToSM = ["*"]
```

Additionally, two maps are used to specify the mapping of the OPR event lifecycle state to the BDM incident status. The maps are named `MapOPR2SMStatus` and `MapSM2OPRState`. Out of the box, all possible states have a mapping.

#### Examples:

- ```
private static final Map MapOPR2SMStatus = ["open": "open", "in_progress": "work-in-progress", "resolved": "resolved", "closed": "closed"]
```
- ```
private static final Map MapSM2OPRState = ["accepted": "open", "assigned": "open", "open": "open", "reopened": "open", "pending-change": "in_progress", "pending-customer": "in_progress", "pending-other": "in_progress", "pending-vendor": "in_progress", "referred": "in_progress", "suspended": "in_progress", "work-in-progress": "in_progress", "rejected": "resolved", "replaced-problem": "resolved", "resolved": "resolved", "cancelled": "resolved", "closed": "closed"]
```

## **Syntax Errors**

If you get a syntax error when customizing your Groovy scripts, you will get an event in the event browser with a detailed description of the error. In addition, you may view the `opr-event-sync-adapter.log` log file for information about how to resolve the error. You can find this log file at the following location:

`<Gateway Server root directory>/log/opr-event-sync-adapter.log`

# View Changes and Incidents in Service Health Using Standalone HP Universal CMDB

This integration enables you to view planned changes and incident details in the Changes & Incidents tab in the 360° View page in Service Health when you are using a standalone HP Universal CMDB.

**Note:** Beginning with UCMDB version 9.05, a new SM adapter (ServiceManagerAdapter9-x) is supplied with UCMDB out of the box, in addition to the legacy adapter (ServiceManagerAdapter7-1):

- For SM versions 9.3x, use ServiceManagerAdapter9.xx.
- For SM versions 9.2x, use ServiceManagerAdapter7-1.

This chapter includes the following:

- ["Prerequisite" on the next page](#)
- ["Step 1: Load the .unl File to Provide External Access to Service Manager" on the next page](#)
- ["Step 2: Configure the Service Desk Adapter Time Zone" on page 508](#)
- ["Step 3: Configure UCMDB to Generate Global IDs" on page 510](#)
- ["Step 4 \(for SM 9.2x only\): Add a Domain" on page 510](#)
- ["Step 5: Configure SM Adapter in UCMDB" on page 511](#)
- ["Step 6: Configure the SM-UCMDB Integration: Create an Integration Point" on page 511](#)
- ["Step 7: Configure the SM-UCMDB Integration: Set Up Data Push Jobs" on page 513](#)
- ["Step 8: Configure the SM-UCMDB Integration: Run Data Push Jobs" on page 513](#)
- ["Step 9: Configure the SM-UCMDB Integration: Add UCMDB Connection Information to SM" on page 514](#)
- ["Step 10: Configure the OMi-UCMDB Integration: Deploy CMS\\_to\\_RTSM\\_Sync.zip on UCMDB" on page 514](#)
- ["Step 11: Configure the OMi-UCMDB Integration: Create an Integration Point on OMi" on page 515](#)

- ["Step 12: Configure the OMi-UCMDB Integration: Create an Integration Point on the CMS" on page 517](#)
- ["Step 13 \(Optional\): Add CI Types to the Service Health Changes and Incidents Component" on page 520](#)
- ["Step 14 \(Optional\): Map Siebel Application CITs" on page 520](#)
- ["Troubleshooting" on page 520](#)

## Prerequisite

**Trusted Sign-on and LW-SSO.** If you want SM to use the SSL-based Trusted Sign-on protocol and LW-SSO, configure it according to the instructions in the HP Service Manager online help.

## Step 1: Load the .unl File to Provide External Access to Service Manager

To enable OMi to query incidents and changes, you must apply the fix described in <http://support.openview.hp.com/selfsolve/document/KM1015767>. This is required because the .unl file expects the length of the name attribute in the EXTACCESSM1 table to be 50 in the database, but its default out-of-the-box length is 100.

Therefore, you must reduce the size of the attribute, load the .unl file, and then increase the size again:

**Note:** These steps are for the SQL Server, but you can see the KM document for the equivalent Oracle syntax.

1. Reduce the length of the name attribute in the EXTACCESSM1 table:
  - a. Database field truncation may result in data loss if data in the field exceeds the default length, so first check the size of the data in the field:

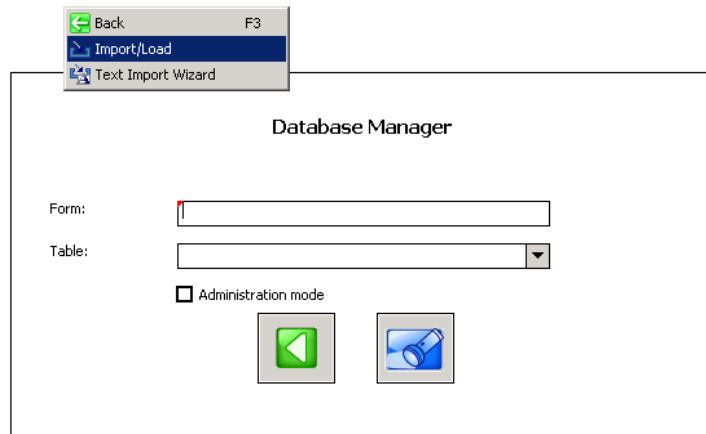
```
Select NAME, LEN(NAME) from EXTACCESSM1 order by 2 desc
```

- b. Reduce the size of the field:

```
alter table EXTACCESSM1 alter column NAME VARCHAR(50)
```

2. In SM, type **db** in the command line text widget in the menu bar at the top of the client display.

- Right-click the white background and select **Import/Load** from the context menu that appears.



- Click the folder icon at the end of the File Name box and navigate to the .unl file on the DPS:

- **Windows:** %TOPAZ\_HOME%\odb\runtime\fcmdb\CodeBase\ServiceManagerAdapter7-1\ucmdbIntegration7\_1x.unl
- **Linux:** /opt/HP/BSM/odb/runtime/fcmdb/CodeBase/ServiceManagerAdapter7-1\ucmdbIntegration7\_1x.unl

Select the file, and click **Open**.

- Click **Load FG** on the toolbar to load the file. If you receive a message saying that the file you are loading will change the keys, click **Yes**.
- Increase the size of the field back to what it was originally:

```
alter table EXTACCESSM1 alter column NAME VARCHAR(100)
```

## Step 2: Configure the Service Desk Adapter Time Zone

Configure the time zone so Incidents and Planned Changes have the correct time definitions:

- In SM, select **Navigation pane > Menu navigation > System Administration > Base System Configuration > Miscellaneous > System Information Record**.
- In the **Date Info** tab, check the Time Zone setting.



3. On the DPS, open the following file:

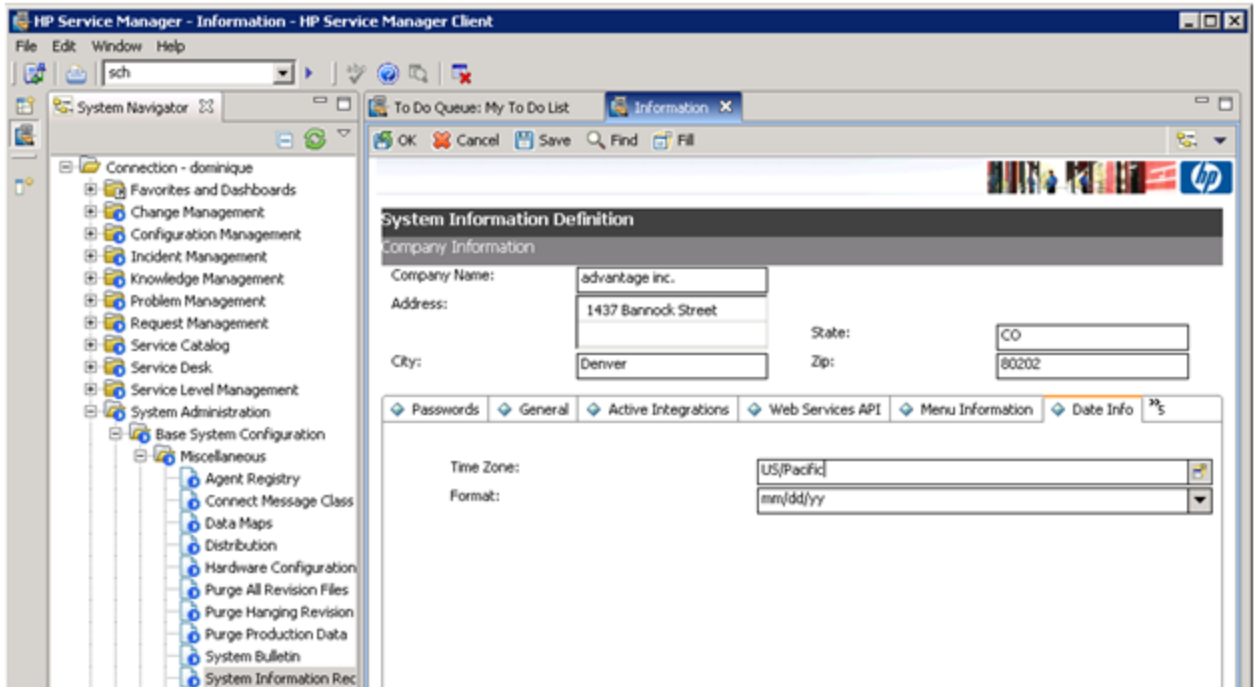
- **Windows:** %TOPAZ\_HOME%\odb\runtime\fcmdb\CodeBase\*<ServiceManagerAdapter9-x or ServiceDeskAdapter7-1>*\serviceDeskConfiguration.xml
- **Linux:** /opt/HP/BSM/odb/runtime/fcmdb/CodeBase/*<ServiceManagerAdapter9-x or ServiceDeskAdapter7-1>*/serviceDeskConfiguration.xml

4. Find the row that includes the following string:

```
<globalConnectorConfig><![CDATA[<global_configuration><date_pattern>MM/dd/yy
HH:mm:ss</date_pattern><time_zone>US/Pacific</time_zone>
```

Check the date and time format, as well as a time zone. Note that the date is case-sensitive. Change either SM or the xml file so that they both match each other's settings.

**Note:** Specify a time zone from the Java time zone list that matches the time zone used in SM (for example, America/New York).



5. Restart the corresponding server to make the change take effect. (If you changed the time zone on SM, restart the SM server; if you changed the time zone on OMi, restart the OMi server.)

## Step 3: Configure UCMDB to Generate Global IDs


1. On the UCMDB server, navigate to:

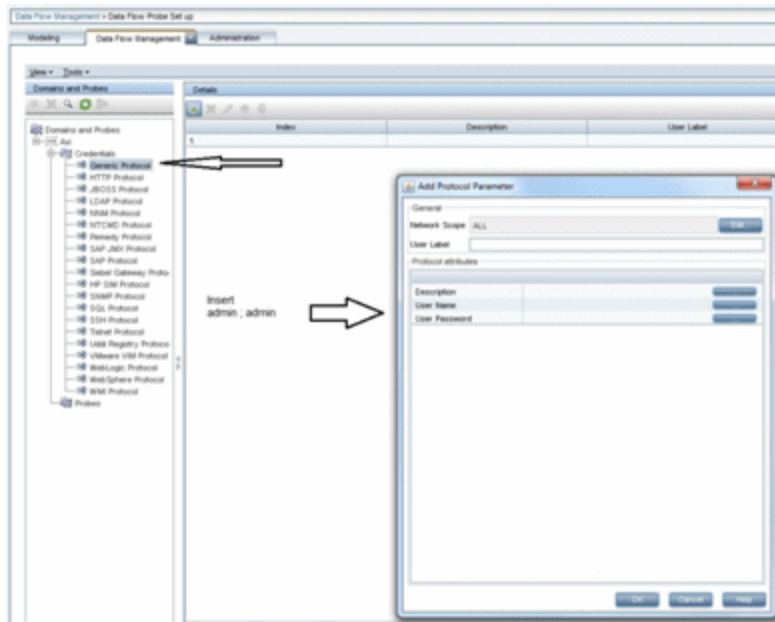
`http://<UCMDB server name>:21212/jmx-console`

2. Enter the user name and password.
3. In the UCMDB section, select **UCMDB:service=Multiple CMDB Instances Services**.
4. For **setAsGlobalIdGenerator**, click **Invoke**.
5. On the DPS, open the following file:
  - **Windows:** %TOPAZ\_HOME%\odb\runtime\fcmdb\CodeBase\*<ServiceManagerAdapter9-x or ServiceManagerAdapter7-1>*\sm.properties
  - **Linux:** /opt/HP/BSM/odb/runtime/fcmdb/CodeBase/*<ServiceManagerAdapter9-x or ServiceManagerAdapter7-1>*/sm.properties
6. Set the **use.global.id** parameter to **true**.

For SM versions 9.2x, proceed with the next step. For SM versions 9.3x, skip to "[Step 5: Configure SM Adapter in UCMDB](#)" on the next page.

## Step 4 (for SM 9.2x only): Add a Domain

1. In OMi, select **Administration > RTSM Administration > Data Flow Management > Data Flow Probe Setup**.
2. In the **Domains and Probes** pane, click .
3. In the **Add New Domain** dialog box, enter a new domain name, and then click **OK**. This creates a new domain and its protocols.
4. Within the domain you added, select **Credentials > Generic Protocol**, and then click the **Add new connection details** button in the right pane. In the **Add Protocol Parameter** dialog box that opens, insert the SM administrator credentials.



## Step 5: Configure SM Adapter in UCMDB

1. Within the UCMDB user interface, access **Data Flow Management > Adapter Management**.
2. In the resources window, select **ServiceManagerAdapter9-x** or **ServiceManagerAdapter7-1 > Configuration files**.
3. Select **ServiceManagerAdapter9-x/sm.properties** or **ServiceManagerAdapter7-1/sm.properties**.
4. In the window on the right side of the screen, modify the **use.global.id** parameter, set it to **false**, and click **OK**.

## Step 6: Configure the SM-UCMDB Integration: Create an Integration Point

1. Within the UCMDB user interface, select **Data Flow Management > Integration Studio**.
2. In the Integration Point pane, select **Create New Integration Point**. The Create New Integration Point dialog box opens. Enter the following:

Name	Recommended Value	Description
<b>Integration Name</b>	<b>SM Integration</b>	The name you give to the integration point.
<b>Adapter</b>	<b>&lt;user defined&gt;</b>	Select the appropriate adapter for the version of SM that you are using.
<b>Is Integration Activated</b>	<b>selected</b>	Select this check box to create an active integration point.
<b>Hostname/IP</b>	<b>&lt;user defined&gt;</b>	The name of the SM server.
<b>Port</b>	<b>&lt;user defined&gt;</b>	The port through which you access SM.
<b>Credentials</b>	<b>&lt;user defined&gt;</b>	<ul style="list-style-type: none"> <li>■ For SM 9.2x, select the user credentials created in <a href="#">"Step 4 (for SM 9.2x only): Add a Domain" on page 510</a>.</li> <li>■ For SM 9.3x, in the default domain select Generic Protocol, and enter the credentials of the SM administrator.</li> </ul>
<b>Probe Name</b> (for ServiceManagerAdapter9-x only)	<b>&lt;user defined&gt;</b>	If you are using ServiceManagerAdapter9-x, select the probe which reports to CMS (see <a href="#">"Prerequisite" on page 507</a> ).

**Note:** It is recommended to click the **Test Connection** button to verify that the details entered are working before continuing.

If your SM backend server (SM Tomcat server) is configured to accept SSL connections, the port setting is ignored and you must configure URL Override. For details, see the UCMDB documentation.

3. In the **Integration Point** pane, click the Integration Point you just created, and click the **Federation** tab in the right pane.
4. In the **Supported and Selected CI Types** area, verify that **Incident**, **Problem**, and **RequestForChange** are selected.

## Step 7: Configure the SM-UCMDB Integration: Set Up Data Push Jobs

Depending on your adapter version, perform the following:

### For ServiceManagerAdapter9-x:

1. Edit the **SM Push** job, and select **Scheduler Definition**.
2. For the **Repeat** field, you can select **Changes Sync/All Data Sync**.
3. Set the **Repeat Every** field to **1 Day**, and click **OK**.

### For ServiceManagerAdapter7-1:

1. Edit the **SM Topology Comparison Push** job, and select **Scheduler Definition**.
2. For the **Repeat** field, select **interval**.
3. Set the **Repeat Every** field to **1 Day**, and click **OK**.
4. Edit the **SM History-based Push** job, and select **Scheduler Definition**.
5. For the **Repeat** field, select **interval**.
6. Set the **Repeat Every** field to **1 Day**, and click **OK**.

## Step 8: Configure the SM-UCMDB Integration: Run Data Push Jobs

1. In the Integration Point pane, select the correct integration.
2. Select the **Data Push** tab. The Job Definition pane is displayed.
3. Select your job and click **Synchronize All** to run the push job.

**Note:** For ServiceManagerAdapter7-1, run this first for the **SM History-based Push** job, then repeat for the **SM Topology Comparison Push** job.

4. When the Confirm synchronizing window is displayed, click **Yes**.
5. Click the **Statistics** tab to view the progress of the synchronization.
6. Click **Refresh** to view the updated synchronization status.

## Step 9: Configure the SM-UCMDB Integration: Add UCMDB Connection Information to SM

1. Log on to your UCMDB system as an administrator. Verify that all UCMDB services are running.
2. Log on to your SM system as an administrator.
3. Select **System Administration > Base System Configuration > Miscellaneous > System Information Record**.
4. Select the **Active Integrations** tab.
5. Select the **HP Universal CMDB** option. The form displays the UCMDB Web service URL field.
6. In the UCMDB Web service URL field, enter the URL to the UCMDB Web service API. The URL has the following format:  
  
**http://<UCMDB server name>:<port>/axis2/services/ucmdbSMService**
7. In the UserId dialog box, enter your UCMDB user name and password and click **Save**.

## Step 10: Configure the OMi-UCMDB Integration: Deploy CMS\_to\_RTSM\_Sync.zip on UCMDB

1. Copy the `CMS_to_RTSM_Sync.zip` file located on the OMi-DPS machine file system under **%TOPAZ\_HOME%\odb\conf\factory\_packages** (Windows) or **/opt/HP/BSM/odb/conf/factory\_packages** (Linux) to the file system on the UCMDB machine.
2. Within the UCMDB user interface, select the **Administration** tab.
3. Select **Package Manager > Deploy Packages to server (from local disk)**.
4. Click the **Add** button and select the **CMS\_to\_RTSM\_Sync.zip** file through the file system browser.
5. Select **Deploy**.

## Step 11: Configure the OMi-UCMDB Integration: Create an Integration Point on OMi

1. Within the OMi user interface, select **Administration > RTSM Administration > Data Flow Management > Integration Studio**.
2. In the Integration Point pane, select **Create New Integration Point**. The Create New Integration Point dialog box opens. Enter the following:

	<b>Recommended</b>	
<b>Name</b>	<b>Value</b>	<b>Description</b>
<b>Integration Name</b>	<b>&lt;user defined&gt;</b>	The name you give to the integration point.
<b>Adapter</b>	<b>UCMDB 9.x</b>	Select the adapter type from the drop-down list.
<b>Is Integration Activated</b>	<b>selected</b>	Select this check box to create an active integration point.
<b>Hostname/IP</b>	<b>&lt;user defined&gt;</b>	The name of the UCMDB server, load balancer, or reverse proxy.
<b>Port</b>	<b>&lt;user defined&gt;</b>	The port through which you access UCMDB, load balancer, or reverse proxy.
<b>Credentials</b>	<b>&lt;user defined&gt;</b>	<p>If credentials appear in the Credentials column, select them.</p> <p>If no credentials appear, select <b>Generic Protocol</b> and click the <b>Add new connection details for selected protocol type</b> button.</p> <p>Enter the following information:</p> <ul style="list-style-type: none"> <li>■ <b>Description.</b> Enter <b>UCMDB</b>.</li> <li>■ <b>User Name.</b> Enter the UCMDB user name. The default value is <b>admin</b>.</li> <li>■ <b>User Password.</b> Enter and confirm a password.</li> </ul>

	<b>Recommended</b>	
<b>Name</b>	<b>Value</b>	<b>Description</b>
<b>Probe Name</b> (for ServiceManagerAdapter9-x only)	<b>&lt;user defined&gt;</b>	If you are using ServiceManagerAdapter9-x, select the probe which reports to <i>OMi</i> (see <a href="#">"Prerequisite" on page 507</a> ).

3. Click the **Add** icon on the right side of the window and add Job definitions as follows:
  - a. Name the **Job definition**.
  - b. Select the **Allow Delete** check box.
  - c. Click the **Add** icon in the Job definition window.
  - d. From the pop-up window, browse to **root - CMS sync** and select the **ActiveDirectory\_sync** job and click **OK**.
  - e. Select the **Scheduler definition** check box.
  - f. In the Repeat window, select **Cron**.
  - g. For the Cron expression, enter the following string: **\* 0/10 \* \* \* ? \***.
  - h. Adjust other settings as needed.
  - i. When finished, click **OK** and save the integration.
  - j. Repeat steps **a** to **i** and configure the following jobs:
    - **FailoverCluster\_Sync**
    - **IIS\_Sync**
    - **SOA\_Sync**
    - **BusinessAndFacilities\_Sync**
    - **ExchangeServer\_Sync**
    - **Virtualization\_Sync**
    - **Siebel\_Sync**



- **Credentials\_Sync**
  - **Basicinfrastructure\_Sync**
  - **J2EE\_Sync**
  - **SAP\_Sync**
4. Browse to UCMDB on port 21212 (for example, [http://<DPS\\_host>.<domain>:21212](http://<DPS_host>.<domain>:21212)), and select the **JMX Console**.
  5. Log on to the JMX console.
  6. From the UCMDB section, select **UCMDB:service=Multiple CMDB Instances Services**.
  7. Invoke:
    - a. **setAsGlobalIdGenerator** and verify it succeeded.
    - b. **getGlobalIdGeneratorScopes** and verify it succeeded.
  8. Within OMi, access **Administration > RTSM Administration > Data Flow Management > Integration Studio**.
  9. Select the Integration Point that you have configured.
  10. In the Job definition section, click **Synchronize All** to run the synchronization.

The Integration Point should be active and the jobs are displayed properly.

## Step 12: Configure the OMi-UCMDB Integration: Create an Integration Point on the CMS

1. Log on to UCMDB and select **Data Flow Management > Integration Studio**.
2. In the Integration Point pane, select **Create New Integration Point**. The Create New Integration Point dialog box opens. Enter the following:

	<b>Recommended</b>	
<b>Name</b>	<b>Value</b>	<b>Description</b>
<b>Integration Name</b>	<b>&lt;user defined&gt;</b>	The name you give to the integration point.
<b>Adapter</b>	<b>UCMDB 9.x</b>	Select the adapter type from the drop-down list.
<b>Is Integration Activated</b>	<b>selected</b>	Select this check box to create an active integration point.
<b>Hostname/IP</b>	<b>&lt;user defined&gt;</b>	The name of the OMi server, load balancer, or reverse proxy.
<b>Port</b>	<b>&lt;user defined&gt;</b>	The port through which you access OMi, load balancer, or reverse proxy.
<b>Credentials</b>	<b>&lt;user defined&gt;</b>	<p>If credentials appear in the Credentials column, select them.</p> <p>If no credentials appear, select <b>Generic Protocol</b> and click the <b>Add new connection details for selected protocol type</b> button.</p> <p>Enter the following information:</p> <ul style="list-style-type: none"> <li>■ <b>Description.</b> Enter <b>UCMDB</b>.</li> <li>■ <b>User Name.</b> Enter the UCMDB user name. The default value is <b>admin</b>.</li> <li>■ <b>User Password.</b> Enter and confirm a password.</li> </ul>
<b>Probe Name</b> (for ServiceManagerAdapter9-x only)	<b>&lt;user defined&gt;</b>	If you are using ServiceManagerAdapter9-x, select the probe which reports to the <i>CMS</i> (see <a href="#">"Prerequisite" on page 507</a> ).

3. Click the **Add** icon on the right side of the window and add Job definitions as follows:
  - a. Name the **Job definition**.
  - b. Select the **Allow Delete** check box.
  - c. Click the **Add** icon in the Job definition window.

- d. From the pop-up window, browse to **root - CMS sync** and select the **ActiveDirectory\_sync** job and click **OK**.
  - e. Select the **Scheduler definition** check box.
  - f. In the Repeat window, select **Cron**.
  - g. For the Cron expression, enter the following string: **\* 0/10 \* \* \* ? \***.
  - h. Adjust other settings as needed.
  - i. When finished, click **OK** and save the integration.
  - j. Repeat steps **a** to **i** and configure the following jobs:
    - **FailoverCluster\_Sync**
    - **IIS\_Sync**
    - **SOA\_Sync**
    - **BusinessAndFacilities\_Sync**
    - **ExchangeServer\_Sync**
    - **Virtualization\_Sync**
    - **Siebel\_Sync**
    - **Credentials\_Sync**
    - **Basicinfrastructure\_Sync**
    - **J2EE\_Sync**
    - **SAP\_Sync**
4. Browse to UCMDB on port 8080 (for example, <http://yourUCMDBhost.domain:8080>), and select the **JMX Console**.
  5. Log on to the JMX console.
  6. From the UCMDB section, select **UCMDB:service=Multiple CMDB Instances Services**.

7. Invoke:
  - a. **setAsGlobalIdGenerator** and verify it succeeded.
  - b. **getGlobalIdGeneratorScopes** and verify it succeeded.
8. Within UCMDB, access **Data Flow Management > Integration Studio**.
9. Select the Integration Point that you have configured.
10. In the Job definition section, click **Synchronize All** to run the synchronization.

The Integration Point should be active and the jobs are displayed properly.

## Step 13 (Optional): Add CI Types to the Service Health Changes and Incidents Component

By default, OMi Service Health displays information on incidents and requests for change for the following CI types: Business Service, Siebel Application, Business Application, and Node.

If you want to view change and incident information for other CITs, perform the procedure described in ["How to Customize the Changes and Incidents Component" on page 531](#).

## Step 14 (Optional): Map Siebel Application CITs

To create a mapping between the **Hand Held Devices** or **Display Device** CIT in SM with **Siebel Application** CITs in OMi, perform one of the following procedures:

- In SM, select **Main page > To Do > Queue: Configuration Item > New > New** and click **Device**. In the Configuration Item field, enter the exact name (case sensitive) of the OMi CI that corresponds to the **Siebel Application** CIT in OMi.
- Create a new population job that includes the **Hand Held Devices** or **Display Device** CIT. Those CITs correspond to the Siebel application CITs. For details about how to create a population job, see the *Modeling Guide*.

## Troubleshooting

If you are not seeing expected incidents in OMi, perform the following:

1. On the Data Processing Server, search the **odb\odb\Error.log** file for **Error Code 802**.
2. In this error message, locate the following string: **property [<category or incident\_status>=<attribute value>[STRING] ] is defined as attribute.**

This indicates that a certain attribute value is missing in RTSM.

3. Access **RTSM Administration > CI Type Manager**.
4. From the **CI Types** menu, select **System Type Manager**, and open **Category** or **Incident Status** (depending on the error message) for editing.
5. Click the Add button (+), and add the missing attribute value (exactly as it appears in the error message) to the list of values.

# View Changes and Incidents in Service Health Using RTSM

This integration enables you to view planned changes and incident details in the Changes & Incidents tab in the 360° View page in Service Health when you are working with RTSM. For details, see the *OMi User Guide*.

This chapter includes the following:

- ["Prerequisite" below](#)
- ["Step 1: Configure the Service Desk Adapter Time Zone" below](#)
- ["Step 2: Create an Integration User Account in Service Manager" on page 524](#)
- ["Step 3: Add the OMi Connection Information in SM" on page 525](#)
- ["Step 4: Create an Integration Point in OMi" on page 525](#)
- ["Step 5: Create New Jobs to Synchronize Between OMi and SM" on page 527](#)
- ["Step 6: Run the Job" on page 527](#)
- ["Step 7: Test the Configuration" on page 528](#)
- ["Step 8 \(Optional\): Add CI Types to the Service Health Changes and Incidents Component" on page 530](#)
- ["Troubleshooting" on page 530](#)

## Prerequisite

If you are using SM versions 9.3x, before you begin, you must install a data-flow probe with the OMi Gateway Server as its target. When you configure the integration point, you will select this probe for the integration.

## Step 1: Configure the Service Desk Adapter Time Zone

Configure the time zone so Incidents and Planned Changes have the correct time definitions:

1. In SM, select **Navigation pane > Menu navigation > System Administration > Base System Configuration > Miscellaneous > System Information Record**.

2. In the **Date Info** tab, open the following file:

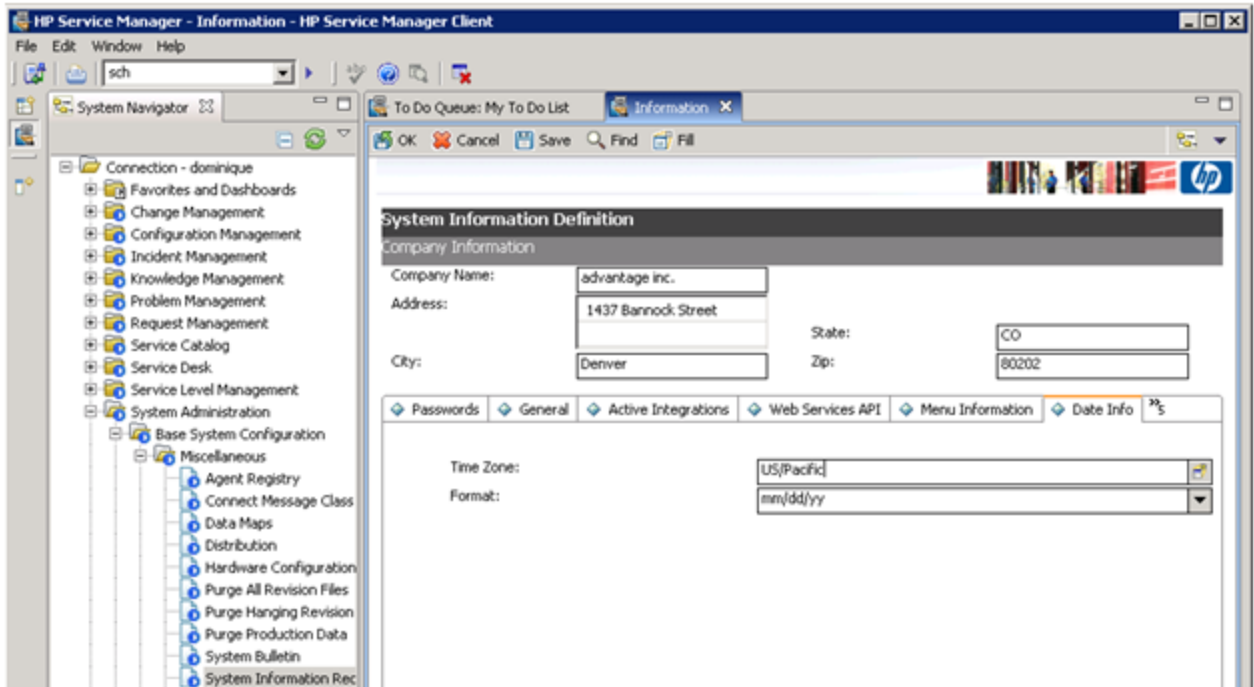
- **Windows:** %TOPAZ\_HOME%\odb\runtime\fcmdb\CodeBase\*<ServiceManagerAdapter9-x or ServiceDeskAdapter7-1>*\serviceDeskConfiguration.xml
- **Linux:** /opt/HP/BSM/odb/runtime/fcmdb/CodeBase/*<ServiceManagerAdapter9-x or ServiceDeskAdapter7-1>*/serviceDeskConfiguration.xml

3. Find the row that includes the following string:

```
<globalConnectorConfig><![CDATA[<global_configuration><date_pattern>MM/dd/yy
HH:mm:ss</date_pattern><time_zone>US/Pacific</time_zone>
```

Check the date and time format, as well as a time zone. Note that the date is case-sensitive. Change either SM or the xml file so that they both match each other's settings.

**Note:** Specify a time zone from the Java time zone list that matches the time zone used in SM (for example, America/New York).



4. Restart the corresponding server to make the change take effect. (If you changed the time zone on SM, restart the SM server; if you changed the time zone on OMi, restart the OMi server.)

## Step 2: Create an Integration User Account in Service Manager

This integration requires an administrator user account for OMi to connect to SM. This user account must already exist in both OMi and SM.

To create a dedicated integration user account in SM:

1. Log on to SM as a system administrator.
2. Type **contacts** in the SM command line, and press **ENTER**.
3. Create a new contact record for the integration user account.
  - a. In the **Full Name** field, type a full name. For example, RTSM.
  - b. In the **Contact Name** field, type a name. For example, RTSM.
  - c. Click **Add**, and then **OK**.
4. Type **operator** in the SM command line, and press **ENTER**.
5. In the **Login Name** field, type the user name of an existing system administrator account, and click **Search**.

The system administrator account displays.

6. Create a new user account based on the existing one:
  - a. Change the **Login Name** to the integration account name you want (for example, *rtsm*).
  - b. Type a **Full Name**. For example, RTSM.
  - c. In the **Contact ID** field, click the **Fill** button and select the contact record you have just created.
  - d. Click **Add**.
  - e. Select the **Security** tab, and change the password.
  - f. Click **OK**.

The integration user account is created. Later you will need to add this user account (user name/password) in RTSM, and then specify this user account in the **Credentials ID** field when creating an integration point in RTSM administration.



### Step 3: Add the OMi Connection Information in SM

The integration requires the OMi connection information to obtain CI attribute information from the OMi system, and display it in the Actual State section in the SM configuration item form.

1. Log on to SM as a system administrator.
2. Click **System Administration > Base System Configuration > Miscellaneous > System Information Record**.
3. Click the **Active Integrations** tab.
4. Select the **HP Universal CMDB** option.

The form displays the UCMDB web service URL field.

5. In the UCMDB web service URL field, type the URL to the HP Universal CMDB web service API. The URL has the following format:

**http://<UCMDB server name>:<port>/axis2/services/ucmdbSMService**

Replace <UCMDB server name> with the host name of your OMi server, and replace <port> with the communications port your OMi server uses.

6. In **UserId** and **Password**, type the user credentials required to manage CIs on the OMi system. For example, the out-of-the-box administrator credentials are **admin/admin**.
7. Click **Save**. SM displays the message: **Information record updated**.
8. Log out of the SM system.
9. Log back into the SM system with an administrator account. The **Actual State** section will be available in CI records pushed from OMi.

### Step 4: Create an Integration Point in OMi

A default RTSM installation already includes the *ServiceManagerAdapter9-x* package. To use the integration package, you must create an integration point listing the connection properties for the integration.

To create an integration point, follow these steps:

1. Create a user in OMi and set the RTSM Permissions in the Create Role page. For details about creating and configuring users, groups, and roles in OMi, see the OMi Administration Guide.
2. In OMi, select **Administration > RTSM Administration > Data Flow Management > Integration Studio**.
3. In the Integration Point pane, select **Create New Integration Point**. The Create New Integration Point dialog box opens. Enter the following:

Name	Recommended Value	Description
<b>Integration Name</b>	<b>SM Integration</b>	The name you give to the integration point.
<b>Adapter</b>	<b>&lt;user defined&gt;</b>	Select <b>HP BTO Products &gt; Service Manager &gt; Service Manager 9.xx</b> .  This adapter, which supports CI/ relationship Data Push from RTSM to Service Manager, and Population and Federation from Service Manager to RTSM.
<b>Is Integration Activated</b>	<b>selected</b>	Select this check box to create an active integration point.
<b>Hostname/IP</b>	<b>&lt;user defined&gt;</b>	The name of the SM server.
<b>Port</b>	<b>&lt;user defined&gt;</b>	The port through which you access SM.
<b>Credentials</b>	<b>&lt;user defined&gt;</b>	Click <b>Generic Protocol</b> , click the <b>Add</b> button to add the integration user account you created in " <a href="#">Step 2: Create an Integration User Account in Service Manager</a> " on page 524, and then select it. This account must exist in both Service Manager and OMi.
<b>Probe Name</b> (for ServiceManagerAdapter9-x only)	<b>&lt;user defined&gt;</b>	Select the probe that you installed for this integration.

**Note:** It is recommended to click the **Test Connection** button to verify that the details entered are working before continuing.

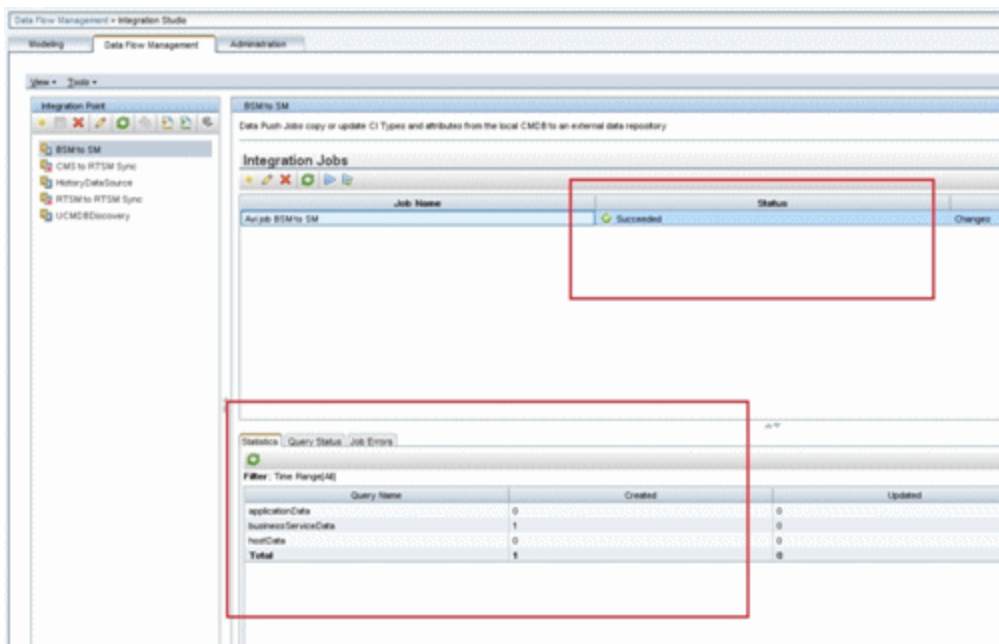
4. In the **Integration Point** pane, click the Integration Point you just created, and click the **Federation** tab in the right pane.
5. In the **Supported and Selected CI Types** area, verify that **Incident, Problem, and RequestForChange** are selected.

### Step 5: Create New Jobs to Synchronize Between OMi and SM


1. In OMi, select **Administration > RTSM Administration > Data Flow Management > Integration Studio**.
2. Click the **Data Push** tab.
3. In the New Integration Job dialog box, click the **+** icon on the left.
4. In the Available Queries dialog box, select the relevant queries for the job.

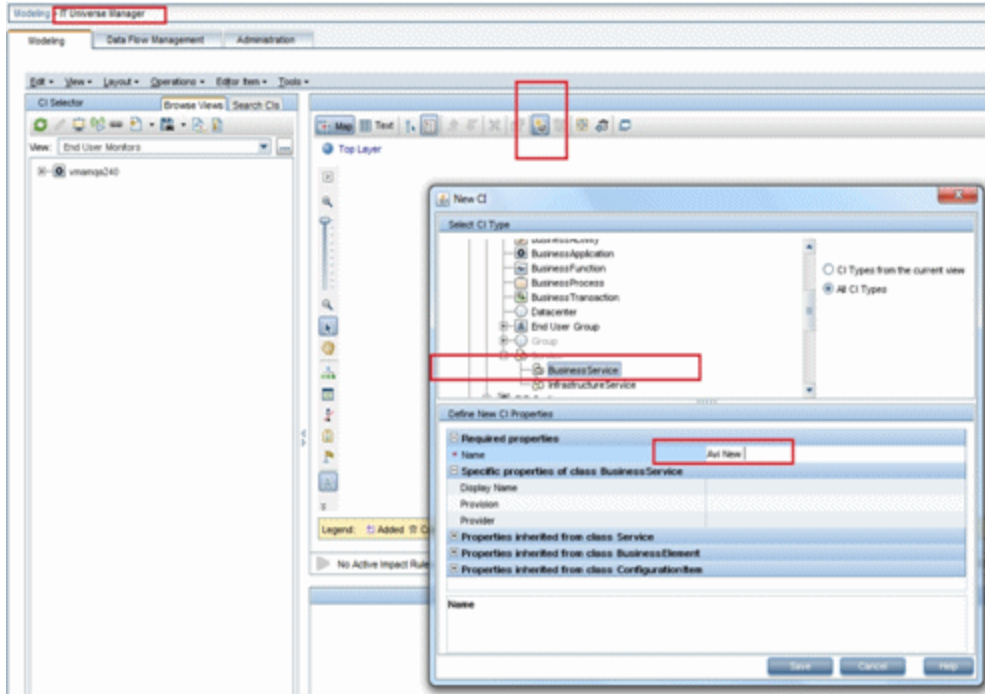
### Step 6: Run the Job

When you run the job, the CIs are synchronized between OMi and SM.



## Step 7: Test the Configuration


1. In OMi, select **Administration > RTSM Administration > Modeling > IT Universe Manager**.
2. In the **CI Selector** pane, select the relevant view, and click  in the right pane.
3. In the **New CI** dialog box that opens, create a new CI with the **BusinessService** type.



4. Create a TQL that includes only BusinessService CI Types (CITs):

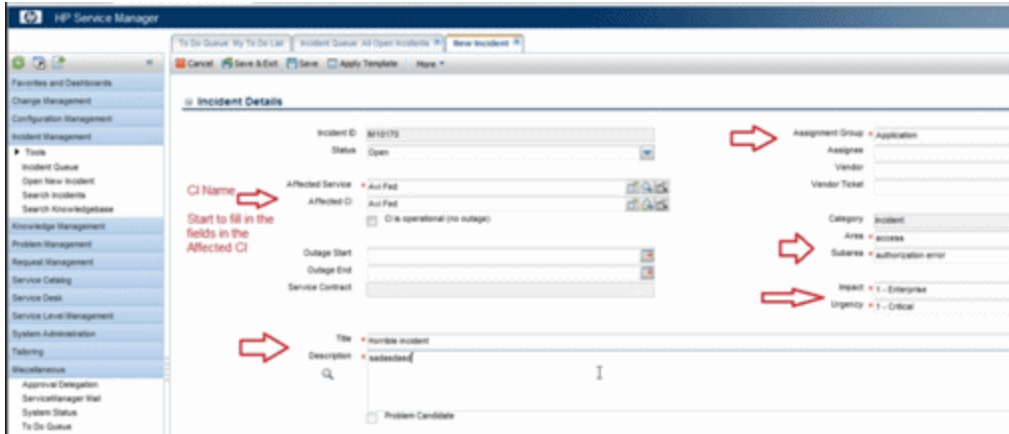
**Administration > Service Health > KPIs in Views**

Alternatively, click [KPIs in Views](#).

5. Click the **Calculate**  button. The relevant CI appears in view.
6. Click the **Data Push** tab, and run the job in order to synchronize with SM. A message that the job was successful should be issued.
7. In SM, create a new incident for the new CI that you created above:
  - a. Select **Incident Management > Open New Incident**.
  - b. **Important:** Start by entering the name of the CI you want to attach to the incident in the

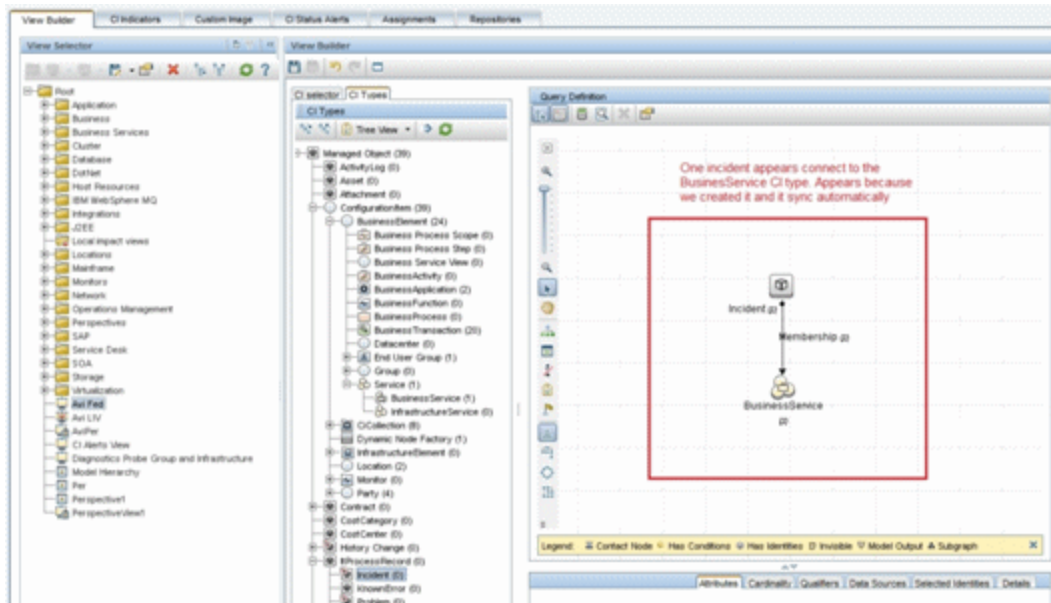
**Affected CI** field. This creates the Incident Id.

- c. Enter the CI name in the **Affected Service** field and click to search.
- d. Enter any incident detail.



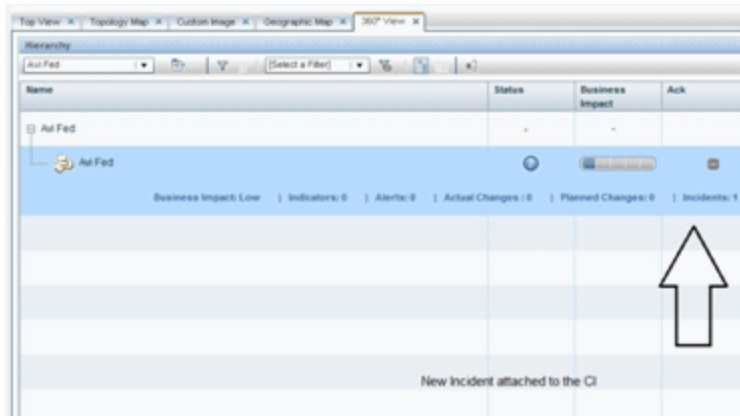
The incident is automatically attached to the CI.

- 8. In OMi, create a TQL with the CI Type you created connected to the Incident CI Type in a membership relationship link.
- 9. Click the **Calculate** button. One incident appears connected to the BusinessService CI Type because this test created it and it is synchronized automatically.



- 10. Delete the incident from the TQL and save the TQL to be a view. The TQL is only used for the test.

11. Select **Application > Service Health**, and click the **360 View** tab. Check that the new incident is attached to the CI.



## Step 8 (Optional): Add CI Types to the Service Health Changes and Incidents Component

By default, OMi Service Health displays information on incidents and requests for change for the following CI types: Business Service, Siebel Application, Business Application, and Node.

If you want to view change and incident information for other CITs, perform the procedure described in ["How to Customize the Changes and Incidents Component" on page 531](#).

## Troubleshooting

If you are not seeing expected incidents in OMi, see ["View Changes and Incidents in Service Health Using Standalone HP Universal CMDB" on page 506](#).

# How to Customize the Changes and Incidents Component

By default, incidents and requests for change are displayed for the following CI types: Business Service, Siebel Application, Business Application, and Node. If you want to view change and incident information for other CITs, perform the following procedure:

1. Open the Modeling Studio:

**Administration > RTSM Administration > Modeling > Modeling Studio**

Alternatively, click [Modeling Studio](#).

Copy one of the TQLs within the **Console** folder, and save your copy with a new name. These default TQLs perform the following:

TQL name	Description
CollectTicketsWithImpacts	Retrieves SM incidents for the selected CI, and for its child CIs which have an Impact relationship.
CollectTicketsWithoutImpacts	Retrieves SM incidents for the selected CI.
CollectRequestForChangeWithImpacts	Retrieves SM requests for change, for the selected CI, and for its child CIs which have an Impact relationship.
CollectRequestForChangeWithoutImpacts	Retrieves SM requests for change, for the selected CI.

2. Edit the new TQL as needed. You can add CITs as described in "[Naming Constraints for New Request for Change TQLs](#)" on the next page.

3. Open Infrastructure Settings:

**Administration > Setup and Maintenance > Infrastructure Settings**

Alternatively, click [Infrastructure Settings](#).

- a. Select **Applications**.
- b. Select **Service Health Application**.
- c. In the **Service Health Application - Hierarchy (360) properties** area, enter the name of the new TQL you created in the corresponding infrastructure setting.

**Note:** By default, these infrastructure settings contain the default TQL names. If you enter a TQL name that does not exist, the default value will be used instead.

After you modify the infrastructure setting, the new TQL will be used, and the Changes and Incidents component will show this information for the CITs you defined.

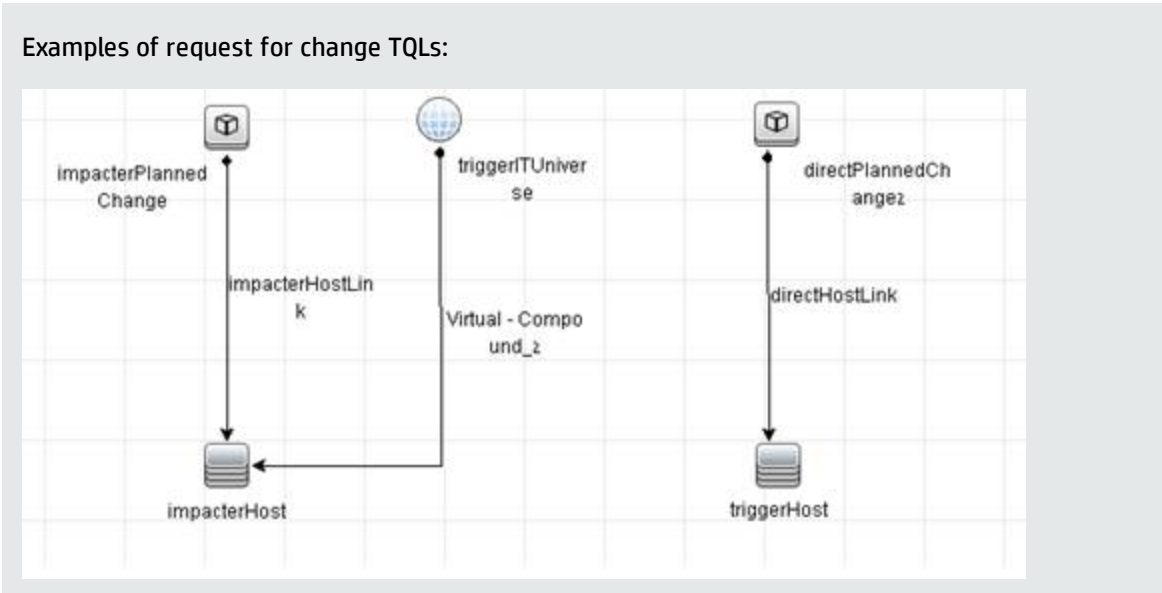
## Naming Constraints for New Request for Change TQLs

The following naming constraints should be followed in the request for change *without* impact TQL (see the TQL example below, on the right side of the image):

- The request for change CI type should start with **directPlannedChange**.
- The CI type related to the request for change should start with **trigger**.

The following naming constraints should be followed in the request for change *with* impact TQL (see the TQL example below, on the left side of the image):

- **impacterPlannedChange** represents the request for change CI type.
- The CI type related to the request for change should start with **impacter**.
- **triggerITUniverse** represents the "impacted" child CIs.





## Naming Constraints for New Incident TQLs

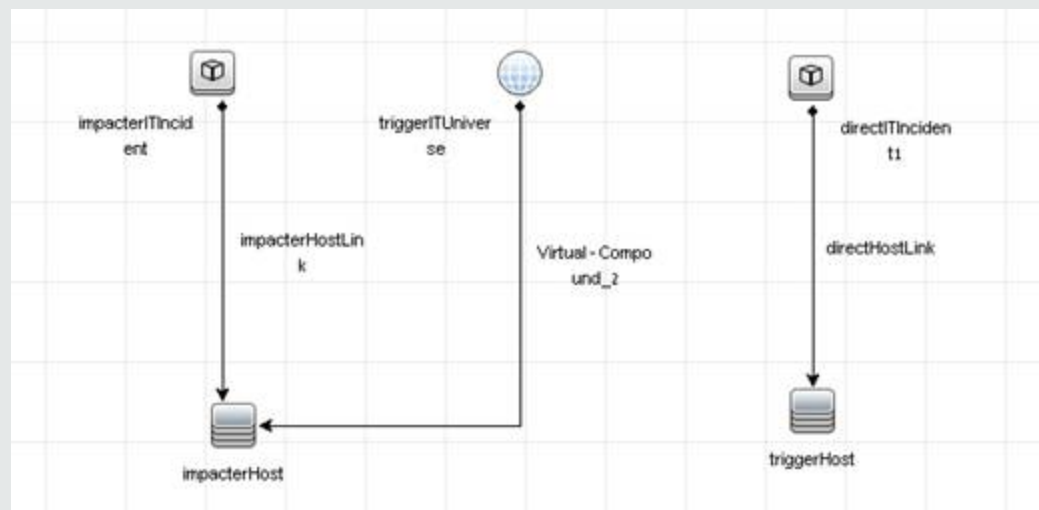
The following naming constraints should be followed in the incidents *without* impact TQL (see the TQL example below, on the right side of the image):

- The incident CI type should start with **directITIncident**.
- The CI type related to the incident should start with **trigger**.

The following naming constraints should be followed in the incidents *with* impact TQL (see the TQL example below, on the left side of the image):

- **impacterITIncident** represents the incident CI type.
- The CI type related to the incident should start with **impacter**.
- **triggerITUniverse** represents the "impacted" child CIs.

Examples of incident TQLs:



## Generate Incidents in SM When an OMi Alert is Triggered

This integration enables you to configure specific CI status alerts to automatically open a corresponding incident in SM. The alerts are mapped to the events using the Event Template.

The triggered alert forwards a corresponding event to OMi where (using the Incident exchange between the SM and OMi integration) the event is changed into an incident and sent, using the Event Forwarding Service, to SM to proactively alert the operator about a problem in the system.

By default, a CI status alert is mapped to an event using a default Event Template. You can modify the default Event Template or select a different Event Template as follows:

1. Access CI Status Alerts:

**Administration > Service Health > CI Status Alerts**

Alternatively, click [CI Status Alerts](#).

Select a view and a CI and click **New Alert** or select an existing alert and click **Edit**.

2. In the Actions page, click the **New Event Generation** link in the **Generate Events** section.
3. In the **CI Alert Template Repository** dialog box that opens, select the template you want to use to map the alert to an event and click **Select**. The template you selected is now listed in the Generate Events section.

For user interface details, see the OMi Administration Guide.

## Integrate BSM and SM

**Note:** The following sections are excerpted from the integrations section of the Business Service Management Help Center. However, to ensure you have the latest information, you should reference the latest published version of the document in question.

# BSM – Service Manager Integration Overview

You can integrate HP Service Manager with one or more of the BSM components, as described below. Each integration can be performed separately.

**Note:** In general, the following document is for integrating BSM 9.2x with Service Manager 9.31.

For instructions on integrating BSM with earlier versions of Service Manager, see [http://support.openview.hp.com/selfsolve/document/KM1303768/binary/BSM9.12\\_SM\\_Integration\\_Interactive\\_Docs.html](http://support.openview.hp.com/selfsolve/document/KM1303768/binary/BSM9.12_SM_Integration_Interactive_Docs.html). Download and extract the zip file contents; open the file **sm\_interactive\_document.htm** and follow the guidelines.

The options are as follows:

- **Downtime exchange between BSM and Service Manager.** BSM enables you to forward downtimes (also known as outages) from BSM to Service Manager, and from Service Manager to BSM. The downtime defined in BSM is converted to a request for change in Service Manager, and vice versa. For details, see "[Downtime Exchange Between BSM and HP Service Manager](#)" on page 537.
- **Incident exchange between Service Manager and Operations Manager i.** BSM enables you to forward events from Operations Management to Service Manager. Forwarded events and subsequent event changes are synchronized back from Service Manager to Operations Management. You can also drill down from Operations Manager events to Service Manager incidents. For details, see "[Incident Exchange between HP Operations Manager i and HP Service Manager](#)" on page 549.
- **View planned changes and incident details in Service Health.** This integration enables you to view planned changes and incident details in the Changes & Incidents tab in the 360° View page in Service Health. For details, see "[View Changes and Incidents in Service Health Using Standalone HP Universal CMDB](#)" on page 565 and "[View Changes and Incidents in Service Health Using RTSM](#)" on page 581.
- **Submit an incident through BSM alerts.** Incidents are automatically opened incidents in Service Manager when a CI Status alert is triggered in BSM. For details, see "[Generate Incidents in Service Manager When a BSM Alert is Triggered](#)" on page 594.
- **View the Number of Open Incidents in Service Health and create SLAs (EMS).** This integration enables you to view the Number of Open Incidents in Service Health views and reports and to manage, in Service Level Management, SLAs over Serviceability KPIs based on Service Manager

incidents (EMS option). For details, see "[View Incident Data in BSM, and Manage SLAs Based on Service Manager](#)" on page 596.

- The **Business Impact Report** integration is described in the *Closed Loop Incident Process (CLIP)* Guide. When deployed as part of the BSM solution, Incident Management users can launch an impact report from an incident in context with the incident's affected CI. Service Desk Agents can validate the updated status of the Business Impact to categorize and prioritize the incident accordingly. For details, refer to the CLIP page in the Solutions Portal:  
<http://support.openview.hp.com/sc/solutions/index.jsp#tab=tab1>.

**Note:**

- **Service Manager Query Security.** If you have set up an integration from BSM to Service Manager, there is a CI context menu that enables you to access Service Manager from BSM Service Health. This drill-down option is not available if you have enabled Service Manager query security.
- **Troubleshooting Multiple Domains.** If BSM and SM are in different domains, and you are using Internet Explorer as your browser, you may need to add the domains to the list of allowed domains in the Privacy tab (**Internet Options > Privacy > Sites**).

# Downtime Exchange Between BSM and HP Service Manager

BSM enables you to forward downtimes (also known as outages) from BSM to Service Manager, and from Service Manager to BSM. The downtime defined in BSM is converted to an incident in Service Manager, and vice versa.

This section includes the following:

- ["Integration Overview" below](#)
- ["Prerequisites" on the next page](#)
- ["Step 1: Send BSM Downtime Events to Service Manager" on page 539](#)
- ["Step 2: Integrate Service Manager Downtimes With BSM" on page 542](#)

## Integration Overview

The downtime integration between BSM and Service Manager includes information exchanges in both of the following directions:

- **Service Manager > BSM.** When you create a downtime RfC (request for change) in Service Manager, the RfC includes the CI that is under change and a start and end date/time of the downtime. If you do not want to waste effort with false alarms in your operations center, and do not want to have these times included in service availability reports, you can set up the integration so that these RfCs are translated to downtimes in BSM.

In this scenario, you install and set up a downtime adapter on your CMDB (whether you are working with a stand-alone uCMDB, or with RTSM). The RfC creates a planned downtime CI in the CMDB, and the adapter sends the planned downtime CI to BSM to create a downtime.

- **BSM > Service Manager.** When you define downtimes using BSM (for example, every Monday and Saturday from 8:30 PM-9:30 PM), in order to proactively support end users, the help desk should be aware of such operational downtimes. After you set up the integration, downtimes in BSM trigger events, which create corresponding incidents in Service Manager.

In this scenario, when a downtime starts, BSM generates an event. Using the event forwarding mechanism, the event generates an incident in Service Manager. When the downtime ends, an event is sent to close the downtime incident.

A single downtime can be defined on more than one CI. In the case of BSM > Service Manager, a separate event is sent for each CI in the downtime.

## Prerequisites

### Supported Platforms

To set up the downtime integration, you must meet the following prerequisites:

- Service Manager 9.31 and higher.
- uCMDB 9.05 CUP 5 and higher with content pack 11 update 2, or uCMDB 10.01 with content pack 12.
- Before deploying the adapter verify that CP11 is installed. If it is not, install the content pack. (This should be done whether you have upgraded to BSM 9.22, or if you installed BSM 9.22 directly.)

**Note:** To see what version of RTSM is installed, access the RTSM JMX console: **http://<BSM server>:21212/jmx-console/**. Click on **DAL services** and run **getCmdbVersion** to get the RTSM version. Click **Content Pack Services** and run **displayCurrentContentPackVersion** to get the content pack version.

- If the adapter is installed on the RTSM, and the adapter is working behind a reverse proxy, the DPS must have the correct certificates installed to send requests to the reverse proxy.
- If you have upgraded from BSM 9.1x, you need to manually redeploy the adapter. Open the Package Manager and delete the **BSMDowntimeAdapter** package. When it is deleted, redeploy the above package from the packages folder.

### Installing the Content Pack for uCMDB 9.05

The following section is only relevant if you are using uCMDB 9.05, or upgrading from BSM 9.20 (which requires the content pack to be installed). If you have not yet installed the content pack, perform the following on your BSM/uCMDB machine:

1. From the content pack installation zip file, copy the content pack zip file to the following location (depending on your environment):

**For RTSM:** <BSM data processing installation folder>\odb\content\content\_packs

**For uCMDB:** <Installation drive or folder>\HP\UCMDB\content\content\_packs

The main BSM folder in Linux is located in: /opt/HP/BSM.

2. Access the following location with your browser:

```
http://<BSM DPS or uCMDB hostname>:21212/jmx-console/HtmlAdaptor?  
action=inspectMBean&name=UCMDB:service=Content Pack Services
```

3. In the method **installContentPack()**, enter the parameters:

- a. Fill the parameter **customerID** with the value of **1**.
- b. Enter the version number found in **version.dat**, located in the content pack zip file.
- c. Invoke the method.

### Global ID Generator

To enable the downtime integration, you must have a global ID generator configured in your environment.

If you are working with RTSM, perform the following to configure the global ID generator:

1. Access the following location with your browser:

```
http://<BSM hostname>:21212/jmx-console/HtmlAdaptor?action=  
inspectMBean&name=UCMDB:service=Multiple CMDB Instances Services
```

2. In the method **setAsGlobalIdGenerator()**, assign the value **1** to the parameter **customerID**, and click **Invoke**.

If you are working with uCMDB, perform the following to configure the global ID generator:

1. Access the following location with your browser: :

```
http://<uCMDB hostname>:8080/jmx-console/HtmlAdaptor?action=  
inspectMBean&name=UCMDB:service=MultipleCMDB Instances Services.
```

2. In the method **setAsGlobalIdGenerator()**, assign the value **1** to the parameter **customerID**, and click **Invoke**.

## Step 1: Send BSM Downtime Events to Service Manager

To enable BSM to send downtime definitions to Service Manager, you must edit an infrastructure setting as described below. This procedure generates events in OMi; you can then use the event forwarding mechanism to generate incidents in Service Manager when a downtime in BSM begins and ends.

1. Access the following location in BSM: **Infrastructure Settings > Foundations > Downtime**.
2. Change the value of the parameter **Downtime Send Event** to **true**.

A corresponding forwarding rule that configures forwarding downtime start and end events from BSM to Service Manager should be configured in the Event Forwarding Rule dialog box. The forwarding rule should be based on the ETI Hint, as follows:

- ETI Hint equals ignore case “downtime: start”
- ETI Hint equals ignore case “downtime: end”

The screenshot displays the HP Business Service Management - Operations Management Administration interface. The main window is titled "Event Forwarding Rules" and shows a list of rules on the left and a "Details" pane on the right. The selected rule is "Automatically forward 'downtime started' events to T...".

**Event Forwarding Rules List:**

- Automatically forward "downtime started" events to T...
- Automatically forward "downtime stopped" events to ...
- Automatically Forward to Trouble Ticket System

**Details Pane:**

**General**

- Display Name: Automatically forward "downtime started" events to Trouble Ticket System
- Description: Automatically forward events indicating the start of a CI downtime to the Trouble Ticket System
- Active: True
- Artifact Origin:  Custom

**Condition**

- Event Filter: Downtime started

**Target Servers**

**Trouble Ticket System**

- Forwarding Type: Notify and Update
- Artifact Origin:  Custom
- Active: False
- Name: TroubleTicketSystem
- Type: Alias
- Connected Server Type: Not specified
- Connected Server Fully Qualified DNS Name: Not specified
- Description:

**sm**

- Forwarding Type: Notify and Update
- Artifact Origin:  Custom
- Active: True
- Name: sm
- Type: External Event Processing
- Fully Qualified DNS Name: [Redacted]
- Port: 13080
- Forward Dynamic Topology to this Target Server: False
- Script Name: sm:ServiceManagerAdapter
- Enable Synchronize and Transfer Control: False
- Description:



For details on how to use the event forwarding mechanism to generate incidents in Service Manager, refer to the section "Event Forwarding" in the *BSM Application Administration Guide*.

Downtime events use the following formats:

- **Downtime Start**

Event field	BSM Downtime
Severity	Normal
Category	Downtime Notification
Title	Downtime for <CI Type><Affected CI Name>started at <Downtime Start Time>
Key	<BSM Downtime ID>:<Affected CI ID>:downtime-start
SubmitCloseKey	False
OutageStartTime	<Downtime Start Time>
OutageEndTime	<Downtime End Time>
CiName	<Affected CI Name>
Ciid	<Affected CI Global ID>
CiHint	GUCMDB:<Affected CI Global ID> UCMDB:<Affected CI ID>
HostHint	GUCMDB:<Related Host Global ID> UCMDB:<Related Host ID>
EtiHint	downtime:start

- **Downtime End**

Event field	BSM Downtime
Severity	Normal
Category	Downtime Notification
Title	Downtime for <CI Type><Affected CI Name> ended at < Downtime End Time>
Key	<BSM Downtime ID>:<Affected CI ID>:downtime-stop
SubmitCloseKey	true
CloseKeyPattern	<BSM Downtime ID>:<Affected CI ID>:downtime-start
EtiHint	downtime:end

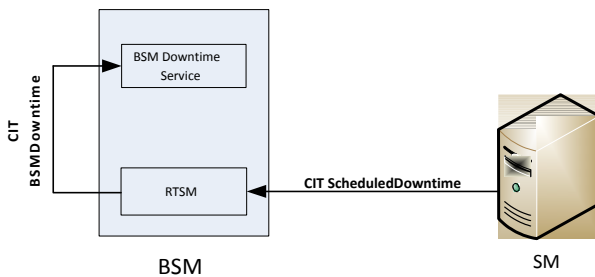
Event field	BSM Downtime
LogOnly	true

## Step 2: Integrate Service Manager Downtimes With BSM

Integrating Service Manager downtimes with BSM consists of:

- Creating an instance of the **ScheduledDowntime** CIT in RTSM/uCMDB
- Creating an instance of the **BSMDowntime** CIT in BSM

The following image shows the data flow when using RTSM:



### Important:

- Following the initial integration, a large amount of data may be communicated from Service Manager to BSM. We recommend that you perform this procedure during off-hours, to prevent negative impact on system performance.
- You should configure both parts of the integration as one flow, without a significant time lag between setting up the two parts. If you set up the Service Manager > uCMDB/RTSM part, and then wait a long time before setting up the uCMDB/RTSM > BSM adapter part, the number of downtimes communicated to BSM initially may be extremely high.

## Configuring HP Service Manager to Send Downtimes

### Note:

- The following provides the basic steps for HP Service Manager configuration. For details, see the HP Service Manager documentation.
- SMBSM\_DOWNTIME is available in HP Service Manager 9.32 and above.

- This document provides a basic description of how to configure SMBSM\_DOWNTIME integration. For details, see the HP Service Manager documentation.

1. Log in to HP Service Manager as **System.Admin**.
2. Click **Tailoring > Integration Manager > Add** to add the Service Manager Integration Suite (SMIS) configuration for SMBSM\_DOWNTIME. The Integration Template Selection page appears.
3. From the Integration Template drop-down list, select **SMBSM\_DOWNTIME**, and click **Next**. The Integration Instance Information page appears.
4. In the **Interval Time(s)** field, type the running frequency data. Set this value based on your configuration item (CI) scheduled downtime data volume for the period.
5. In the **Max Retry Times** field, type the maximum number of retries. Since you are not connecting to another system, type **0**.
6. In the **Log File Directory** field, type the full path for the log file. By default, the log name is **sm.log**.
7. Click **Next**. The Integration Instance Parameters page appears.
8. Click the **General Parameters** tab.
9. Configure the SMIS settings.
  - a. Assign a value for **WithdrawDowntime** (options are **true** or **false**). **True** means that when you make a change using **Change Phase**, if the change has a valid outage, a prompt appears enabling you to reject the outage.
  - b. In the **Category** column, assign the value **Change** for change categories and **Task** for task categories.
  - c. Assign values for the parameters in the **Change** category.

If you only want outages of one change category after your desired phase has been approved, in the **Value** column, set the phase.

If your category workflow has multiple paths with different final approval phases, use a semicolon (;) to separate the phases.
  - d. Assign a unique identifier for your Service Manager deployment to the **sm.host** parameter. This identifier represents your Service Manager server.

Do not use a colon (:) in this field.

- e. Assign a value to the **sm.reference.prefix** parameter. This value is used to populate the External Process Reference of Scheduled Downtime CI in UCMDB.

Service Manager automatically appends a colon (:) at the end of the value.

- f. Click **Next, Next, Finish**.
- g. Select the **SMIS**.
- h. Click **Enable**.
- i. Click **Yes**.

## Integrating SM RFC Downtimes with RTSM/uCMDB

To integrate SM RFC downtimes with RTSM/uCMDB, populate (synchronize) RTSM/uCMDB with the downtime CIs.

1. Log in to RTSM/uCMDB.
  - In BSM, access **Admin > RTSM Administration > Data Flow Management > Integration Studio**
  - In uCMDB, access **Administration > Data Flow Management > Integration Studio**
2. Verify that the integration point in front of the Service Manager exists and is active.

**Edit Integration Point**

**Integration Properties**

- \* Integration Name: sm
- Integration Description:
- Adapter: Service Manager 9.xx
- Is Integration Activated:

**Adapter Properties**

- \* Port: 13080
- Development Mode: False
- \* Hostname/IP:
- \* Credentials ID: genericprotocol: bsm\_admin
- Probe Name:
- URL Override:

\* Mandatory Properties

Test connection

OK Cancel

3. Click **Test connection** and verify that the connection succeeds.
4. In the **Population** tab, add the following integration jobs:
  - **DT Population** based on **CLIP Down Time Population TQL**
  - **DT Relationship** based on **CI To Down Time CI With Connection TQL**
5. Log in to the Service Manager server. Select the **Configuration Management** tab and select **Resources > Configuration Item Relationships**.
6. Add a relation between the **Upstream CI** (for example, any business service instance) and the **Downstream CI** (the affected CI), and click **Add**.
7. In the **Change Management** tab, select **Changes > Open New Change** to open a new request for change (RfC). Verify the **Service**, **Affected CI**, and **Scheduled DownTime Start/End** field are

completed.

**Note:** The **Service** and **Affected CI** values should be equal to the **Upstream/Downstream CI** values you set in the previous step.

8. Click **More > Change Phase**. Move the **RfC** phase to the **Change Approval** phase.
9. Log in to Service Manager as user **Change.Approver**. Open the **Approval In** box and approve the change.
10. Wait for **SMBSM\_DOWNTIME/DT Population/DT Relationship** to run. By default, it runs every minute.
11. Log in to RTSM/uCMDB.
  - In BSM, access **Admin > RTSM Administration > Data Flow Management > Modeling Studio**
  - In uCMDB, access **Administration > Data Flow Management > Modeling Studio**
12. In Modeling Studio, search for **ScheduledDowntime CI**. A downtime CI is created with a relationship to the affected CI.

## Push CIT ScheduledDowntime to CIT BSMDowntime by BSMDowntimeAdapter

1. If you are using uCMDB, deploy the adapter as follows:
  - a. In uCMDB, access **Administration > Package Manager**.  
 In BSM, access **Admin > RTSM Administration > Administration > Package Manager**.
  - b. Click **Deploy package to server**, and import the adapter's zip file from `<BSM DPS installation path>\odb\conf\factory_packages\BSMDowntimeAdapter.zip`.
2. Create the integration point credentials:
  - a. In uCMDB, access **Data Flow Management > Data Flow Probe Setup**.  
 In BSM, access **Admin > RTSM Administration > Data Flow Management > Data Flow Probe Setup**

**Note:** You do not need a Probe to perform this integration; nevertheless you create credentials using the Data Flow Probe Setup tab.

- b. Click **Add domain or probe**, and enter a name and description of your choice.
  - c. Expand the submenus and select **HTTP protocol**.
  - d. Click the plus sign (Add new connection details) and enter the BSM Gateway host name, Port 80, and the BSM username and password. Leave the Trust fields blank. When you are done, click **OK** to save the credentials.
3. Create a new integration point:
- a. In uCMDB, access **Data Flow Management > Integration Studio**.  
In BSM, access **Admin > RTSM Administration > Data Flow Management > Integration Studio**.
  - b. Click **New integration point**, enter a name and description of your choice, and select:
    - In BSM: **BSMDowntimeAdapter**
    - In uCMDB: **SM scheduled Downtime Integration into BSM**
  - c. Enter the following information for the adapter:
    - BSM Gateway hostname and port
    - Communication protocol
    - The integration point credentials you just created
    - Context root (if you have a non-default context root).
  - d. Click **OK**, then click the **Save** button above the list of the integration points.
4. Click the **Statistics** tab in the lower pane, to track the number of downtimes that are created or updated. By default, the integration job runs every minute. If a job failed, you can click the **Query Status** tab and double-click the failed job to view more details about the error.

## Troubleshooting

- If there is an authentication error, verify that the BSM credentials entered for the integration point are correct.
- An unclear error message with an error code generally indicates a communication problem. Check the communication with BSM. If no communication problem is found, restart the MercuryAS process.

**Note:**

- A failed job is repeated until the problem is fixed.
- Each BSM Downtime can be found in BSM Downtime Management (**Admin > Platform > Downtime Management**).



# Incident Exchange between HP Operations Manager i and HP Service Manager

BSM enables you to forward events from Operations Management to HP Service Manager 9.30, 9.21, or 9.20. Forwarded events and subsequent event changes are synchronized back from HP Service Manager to Operations Management. You can also drill down from Operations Manager events to HP Service Manager incidents.

**Note:** HP recommends this integration option for new integrations with HP Service Manager 9.30, 9.21, or 9.20. However, existing integrations that use other integration options are still supported.

This section includes the following:

- ["Step 1: Configure the HP Service Manager Server as a Connected Server" below](#)
- ["Step 2: Configure an Event Forwarding Rule" on page 553](#)
- ["Step 3: Configure URL Launch of Event Browser from HP Service Manager" on page 555](#)
- ["Step 4: Configure URL Launch of HP Service Manager from the Event Browser" on page 556](#)
- ["Step 5: Configure HP Service Manager Server" on page 557](#)
- ["Step 6: Mapping and Customization" on page 558](#)
- ["Step 7: Test the Connection" on page 559](#)
- ["Step 8: Synchronize Attributes" on page 560](#)
- ["Tips for Customizing Groovy Scripts" on page 561](#)

## Step 1: Configure the HP Service Manager Server as a Connected Server

Synchronizing events and event changes between Operations Management events and HP Service Manager incidents requires configuring a Connected Server within Operations Management to correctly identify the target HP Service Manager instance. The first step to achieve this is to configure HP Service Manager as a target connected server in the Connected Servers manager.

For full details about how to configure a connected server, see the Connecting Servers section of the Operations Management online help.

To configure the HP Service Manager server as a target connected server, perform the following steps:

1. Navigate to the Connected Servers manager in the Operations Management user interface:

**Admin > Operations Management > Setup > Connected Servers**

2. Click the New button to open the Create New Server Connection dialog box.
3. In the **Display Name** field, enter a name for the target HP Service Manager server. By default, the Name field is filled automatically. For example, if you enter `Service Manager 1` as the Display Name for the target HP Service Manager server, `Service_Manager_1` is automatically inserted in the Name field. You can specify your own name in the Name field, if you want to change it from the one suggested automatically.

**Note:** Make a note of the name of the new target server (in this example, `Service_Manager_1`). You need to provide it later on as the `username` when configuring the Service Manager server to communicate with the server hosting Operations Management.

*Optional:* Enter a description for the new target server.

Make sure that you check the **Active** checkbox.

Click **Next**.

4. Select `External Event Processing` to choose the server type suitable for an external incident manager like HP Service Manager.

Click **Next**.

5. Enter the Fully Qualified DNS Name of the HP Service Manager target server.

Click **Next**.

6. From the Integration Type dialog box:

- a. Select **Call Script Adapter** as the integration type .
- b. From the Script Name menu, select the HP Service Manager Groovy script adapter

**sm:ServiceManagerAdapter.**

- c. Click **Next**.
7. **In HP Service Manager:** Set up an Integration User with user name and password. This is the user name and password needed by Operations Management to access the HP Service Manager target server and must be specified in the next step.
  8. **In the Operations Management user interface:** In the Outgoing Connection dialog box, enter the credentials (user name, password, and port number) required to access the HP Service Manager target server and to forward events to that server:
    - a. In the **User Name** field, enter the user name for the Integration User you set up in HP Service Manager.
    - b. In the **Password** field, enter the password for the user you just specified. Repeat the password entry in the **Password (Repeat)** field.
    - c. In the **Port** field, specify the port configured on the HP Service Manager side for the integration with Operations Management.

To find the port number to enter:

- If you are using default ports in Service Manager, select/deselect **Use Secure HTTP** as appropriate, then click **Set default port**. The port is set automatically.

**Note:** If you do not want to use secure HTTP, make sure that the **Use secure HTTP** checkbox is *not* checked.

If Use Secure HTTP is selected, download and install a copy of the target server's SSL certificate using the link **Retrieve from Server** or **Import from File**, if the certificate is available in a local file.

- If you need to find the port number, access the following file on your HP Service Manager system:

```
<HP Service Manager root directory>/HP/Service Manager
<version>/Server/RUN/sm.ini
```

The `sm.ini` file contains two port entries:

httpPort - default port number 13080

httpsPort - default port number 13443

The actual values for the ports can differ from these default values depending on how they are configured.

Enter the appropriate value of the port used by Service Manager in the **Port** field of the Outgoing Connection dialog box.

d. Select **Enable Synchronize and Transfer Control**.

If Enable Synchronize and Transfer Control is selected, an Operations Management operator can transfer ownership of the event to the target connected server using the Transfer Control option in the Event Browser context menu.

If it is not selected, the option Synchronize and Transfer Control is not available from the Event Browser context menu or from the list of forwarding types for configuring forwarding rules.

e. Test the connection. A **Success** or **ERROR** hyperlink is displayed. Click the link to get a more detailed message.

f. Click **Next**.

9. If, in addition to automatically generating HP Service Manager incidents from OMi events, you want to also be able to drill down into HP Service Manager, you must specify the fully qualified DNS name and port of the HP Service Manager system where you want to perform incident drilldown.

**Note:** To enable incident drilldown to HP Service Manager, you must install a web tier client for your HP Service Manager server according to your HP Service Manager server install/configuration instructions.

In the Event Drilldown dialog box of the Connected Servers manager, configure the server where you installed the web tier client along with the configured port used.

If you do not specify a server in the Event Drilldown dialog box of the Connected Servers manager, it is assumed that the web tier client is installed on the server used for forwarding events and event changes to HP Service Manager, and receiving event changes back from HP Service Manager.

If nothing is configured in the Event Drilldown dialog box, and the web tier client is not installed on the HP Service Manager server machine, the web browser will not be able to find the requested URL.

Click **Next**.

10. To enable event changes to be synchronized back from HP Service Manager to Operations Management you must provide credentials for the HP Service Manager server to access the server hosting Operations Management.
  - a. In the Incoming Connection dialog box, select the **Accept event changes from external processing server** checkbox, and then enter a password that the HP Service Manager server requires to connect to the server hosting Operations Management.

**Note:** Make a note of this password. You need to provide it later on when configuring the HP Service Manager server to communicate with the server hosting Operations Management. This password is associated with the server name (`Service_Manager_1`) you configured in step 3.

If **Supports Synchronize & Transfer Control** was previously selected, the **Accept event changes from external processing server** option is assumed, and cannot be disabled.

- b. Click **Finish**. The target HP Service Manager server appears in the list of Connected Servers.
11. *For integrations with HP Service Manager 9.34 and later:* Complete the following steps:
  - a. Reopen the Service Manager Connected Server that you configured in the previous steps (double-click the connected server entry in the connected servers list).
  - b. Copy the ID of the connected server (Displayed in the lower right corner of the General tab) and save this ID. You need to specify this ID as the `omi.mgr.id` on Service Manager system.

An example of a Connected Server ID is as follows: **ID: 22f42836-fd36-473e-afc9-a81290f4f73b**

## Step 2: Configure an Event Forwarding Rule

The next step is to configure an event forwarding rule that determines which events are forwarded automatically to HP Service Manager.

Refer to the Operations Management online help for full details about configuring filters.

To configure a forwarding rule, carry out the following steps:

1. Navigate to the Forwarding Rules manager in the Operations Management user interface:

**Admin > Operations Management > Event Automation > Event Forwarding**

2. Click the **New** button to open the Create New Forwarding Rule dialog box.
3. In the **Display Name** field, enter a name for the forwarding rule, in this example Forward Critical (Sync and Transfer Control).

*Optional.* Enter a description for the forwarding rule you are creating.

Make sure the **Activate Rule after creation** checkbox is checked. A rule must be active in order for its status to be available in HP Service Manager.

4. Click the browse button next to the Event Filter field. The Select an Event Filter dialog box opens.

In the Select an Event Filter dialog box, do one of the following:

- Select an existing filter
- Create a new filter as follows:
  - i. Click the **New** button to open the Filter Configuration dialog box.
  - ii. In the **Filter Display Name** field, enter a name for the new filter, in this example, **FilterCritical**.  
  
Uncheck the checkboxes for all severity levels except for the severity Critical.  
  
Click **OK**.
  - iii. You should see your new filter in the Select an Event Filter dialog box (select it, if it is not already highlighted).

Click **OK**.

5. Under **Target Servers**, select the target connected server you configured in the section "[Step 1: Configure the HP Service Manager Server as a Connected Server](#)" on page 549. In this example, this is Service Manager 1.

Click the **Add** button next to the target servers selection field. You can now see the connected server's details. In the **Forwarding Type** field, select Synchronize and Transfer Control.

**Note:** Although all other forwarding types can also be configured, they are not supported. When using any other forwarding type you will lose functionality of the integration. The only supported forwarding type for Service Manager Integration is Synchronize and Transfer Control.

Click **OK**.

## Step 3: Configure URL Launch of Event Browser from HP Service Manager

Before operators are able to perform event drill-down from HP Service Manager into the Operations Management user interface using a URL launch of the Event Browser, the operators must be set up as valid users in BSM with appropriate permissions in Operations Manager i:

### User account requirements

- If Single Sign-On (SSO) authentication is configured, set up each user in BSM with the *same* user name that is used by the HP Service Manager operator to log onto HP Service Manager and to perform the URL call. (The password of each BSM user can be any string, but not empty.) After successfully logging into HP Service Manager, the BSM users can launch the Operations Management Event Browser without further authentication.

For details on setting up SSO, see *Configuring HP Service Manager to Use the SSL-based Trusted Sign-On and LW-SSO* in the Service Manager documentation library.

- If HP Service Manager is not configured to use SSO authentication, set up each user with the *same* user name that is used by the HP Service Manager operator and specify a valid password. The users are required to enter their user name and password when launching the Operations Management Event Browser.

### Required user permissions

You must grant the permission `Events assigned to user` including the required actions to each BSM user. You can optionally grant the permission to view events not assigned to each user.

**Note:** Without valid user names, or if a user does not have the required viewing permissions, any attempt to perform a URL launch of the Operations Management Event Browser from HP Service Manager results in an empty browser window.

## Step 4: Configure URL Launch of HP Service Manager from the Event Browser

To be able to perform a URL launch of HP Service Manager from the Operations Management Event Browser using the web tier client, perform the following:

1. Navigate to the **Connected Server Admin** screen, and click the **Manage Scripts** icon on the right.
2. Select the **sm:ServiceManagerAdapter** script, and click the Edit button.
3. Locate the following text in the Groovy script:

```
private static final String SM_WEB_TIER_NAME = 'webtier-9.30'
```

4. Change the value of `webtier-9.30` to the value required to access the HP Service Manager web tier client.

The full drill-down URL is made up like this:

```
http://<FQDNS of HP Service Manager web tier server>/<web path to HP Service Manager>/<URL query parameters>
```

where *<FQDNS of HP Service Manager web tier server>* is the fully qualified DNS name of the HP Service Manager server where the web tier client is installed. This part of the URL is added automatically (together with `http://`) according to the values that you provided when you configured the HP Service Manager as a target connected server in the Connected Servers manager. For details, see ["Step 1: Configure the HP Service Manager Server as a Connected Server" on page 549](#).

Here is an example of how the drill-down URL looks:

```
http://smserver.example.com/SM930/index.do?ctx=docEngine&file=probsummary&query=number%3D
```

So in this example, the you must replace `webtier-9.30` with `SM930`. All the other parts of the URL are configured automatically.

5. When finished editing, save the new version of the script. (Note that the script can always be reverted to its original version.)
6. In the HP Service Manager web tier configuration file `web.xml`, set the value of the `querySecurity`



parameter from the default value (`true`) to `false`.

For more details, see the section `Web parameter: querySecurity` in the HP Service Manager online help.

7. To integrate with Service Manager 9.34 or earlier, perform the following on the Service Manager side to fully enable the event drilldown from BSM/OMi to Service Manager:

In the **web.xml** configuration file of the Service Manager Tomcat server, search for **querySecurity** and set the value to **false**.

This will lower the security of Service Manager, but allow to drilldown from OMi directly to the related incident in Service Manager. If this is not changed, trying to drill down to the incident results in an error. You can then use the menu entries on the left pane in Service Manager to navigate to the related incident.

## Step 5: Configure HP Service Manager Server

The next step is to configure HP Service Manager server to integrate with Operations Management.

To configure the HP Service Manager server, complete the following steps in the HP Service Manager:

1. From the left hand pane of the HP Service Manager user interface, navigate to:

**Tailoring > Integration Manager**

2. Click **Add** to add a new configuration.
3. Select the **SMOMi** integration template from the Integration Template field. Click **Next**.
4. *Optional.* Change the log level to the desired value.

*Optional.* Change the description, for example, to `This is for SMOMi integration.`

Click **Next**.

5. In the General Parameters tab, replace the existing entries with the following values:

Name	Value	Category
omi.server.url	http://<BSM_gateway_FQDN>/opr-gateway/rest/9.10/synchronization/event/	General
username	Service_Manager_1  (This is the name of the HP Service Manager target server you configured previously in the section " <a href="#">Step 1: Configure the HP Service Manager Server as a Connected Server</a> " on page 549).	Header
omi.eventdetail.baseurl	http://<BSM_gateway_FQDN>/opr-console/opr-evt-details.jsp?eventId=	General

- In the Secure Parameters tab, set the password to the one you specified in the Incoming Connection dialog box when configuring the target connected server in the section "[Step 1: Configure the HP Service Manager Server as a Connected Server](#)" on page 549. In our example, this is HPqwer1\_.

Click **Next**.

- In the Integration Instance Fields dialog box, click **Next**.
- In the Integration Instance Mapping dialog box, click **Finish**.

**Note:** Ensure that the rule is active. To make the rule active, select the rule and click **Enable**.

## Step 6: Mapping and Customization

You can add your own custom attributes in a Groovy script and then map these custom attributes to HP Service Manager to the appropriate field in HP Service Manager. You can also change how attributes are mapped from Operations Management to HP Service Manager. The mapping is done in the BDM Mapping Manager in HP Service Manager:

### System Administration > Ongoing Maintenance > BDM Mapping Management

For full details about mapping attributes, see the HP Service Manager online help.

## Step 7: Test the Connection

To test the connection, send an event to the server hosting Operations Management that matches the filter you defined (in our example filter, the severity value is *Critical*), and then verify that the event is forwarded to HP Service Manager as expected.

To test the connection, do the following:

1. On the Gateway Server system running Operations Management, open an Event Browser.
2. On the system running Operations Management, open a command prompt and change to the following directory:

```
<HPBSM root directory>\opr\support
```

3. Send an event using the following command:

```
sendevent -s critical -t test111-1
```

4. Verify that the event appears in the Operations Management Event Browser.
5. Select the **Forwarding** tab.
6. In the External Id field, you should see a valid HP Service Manager incident ID.
7. Next, verify that the incident appears in the Incident Details in HP Service Manager:

If the event drill-down connection is configured correctly, click the hyperlink created with the Incident ID. A browser window opens, which takes you directly to the incident in the Incident Details in HP Service Manager.

If the event drill-down connection is not configured, do the following:

- a. In the Forwarding tab in the Operations Management Event Browser, copy or note the incident ID from the External Id field.
- b. In the HP Service Manager user interface, navigate to:  
**Incident Management** → **Search Incidents**
- c. Paste or enter the incident ID in the Incident Id field.
- d. Click the **Search** button. This takes you to the incident in the Incident Details.

8. Close the incident in HP Service Manager.
9. Verify that the change in the state of the incident (it is now closed) is synchronized back to Operations Management. You should not be able to see the event that was closed in HP Service Manager in the active Event Browser, but it should now be in the History Browser.

## Step 8: Synchronize Attributes

Not all attributes are synchronized back from Service Manager to Operations Management by default. When the Service Manager incident is initially created from an Operations Management event, all possible event attributes are mapped to the corresponding Service Manager incident attribute. Out of the box, after the initial incident creation, whenever the incident or event subsequently changes, only a subset of the changed event and incident attributes are synchronized. The following describes how to customize the list of attributes to synchronize upon change.

### **Uni-directional Synchronization: Operations Management to HP Service Manager**

The following attributes are transferred to HP Service Manager from Operations Management on a one-time basis, that is, when the event was initially created, and the transfer of control of the event was configured in the Connected Servers manager.

These attributes support bi-directional synchronization, but are disabled out-of-the-box:

- Title
- Severity
- Priority
- Operator: the operator assigned to the event who forwarded the event
- Category
- Subcategory
- Related CI

For the above attributes, there is no back synchronization from HP Service Manager to Operations Management.

## Bi-directional Synchronization

Attributes that support bi-directional synchronization between Operations Management and HP Service Manager are:

- Description
- Lifecycle state (the state is only updated when the state changes to closed)
- Solution
- Operations Management event annotations are synchronized to HP Service Manager activity log
- Contents under the Forwarding tab in the Event Details

## Attribute Synchronization using Groovy Scripts

If you want to change the out-of-the-box behavior regarding which attributes are updated, you can specify this in a Groovy script. In the Groovy script, you would specify which fields are updated in HP Service Manager, and which fields are updated in Operations Management. You can also specify custom attributes in the Groovy script.

In the Groovy script, you would specify which fields are updated in HP Service Manager, and which fields are updated in Operations Management. You can also specify custom attributes in the Groovy script.

## Tips for Customizing Groovy Scripts

This section provides some tips about customizing Groovy scripts. Below we show just a few selected examples of what you can customize. You can look at the configuration section of a Groovy script to see further items that can be modified.

In the configuration section of a Groovy script, you can define and modify the attributes that are to be synchronized between Operations Management and HP Service Manager. The configuration section of a Groovy script also contains the default value mappings for lifecycle state, severity, and priority. You can also modify these, and it is possible to define the mappings for in-going and out-going requests differently.

More advanced configuration can be done in other parts of the Groovy script if required.

The beginning and the end of the configuration section of a Groovy script is marked as follows:

```
//  
// configuration section to customize the Groovy script  
// BEGIN
```

```

...
...
//
// configuration section to customize the Groovy script
// END

```

**Note:** As of BSM 9.20, modifications to Groovy scripts are not overwritten by patches and hotfixes; your customized version of a script will remain after an update/patch. If you want to use the newer version of a script, make a copy of your version, revert back to the predefined version, and then re-apply your changes.

The mapping from Operations Management to HP Service Manager is compliant to BDM 1.1 incident web service specifications. The mapping of the BDM 1.1 incident web service to HP Service Manager is specified in HP Service Manager in the BDM Mapping Manager. For more information about the BDM Mapping Manager, see the BDM Mapping Manager section of the HP Service Manager online help.

## Controlling Attribute Synchronization

You can control how updates to certain attributes are synchronized between Operations Management and HP Service Manager by setting some Boolean variables to true or false.

Here are two examples:

- As of BSM 9.21 there is a new variable called `SyncAllProperties`. By default it is false; if you set it to true, all properties will be synchronized in both directions. The other variables will be ignored.

Here are two additional examples:

- `private static final SyncTitleToSMOnUpdate = false;`

This line of the Groovy script disables the synchronization of changes to the title made in Operations Management to HP Service Manager.

- `private static final Boolean SyncTitleToOPROnUpdate = false;`

This line of the Groovy script disables the synchronization changes to the title made in HP Service Manager to Operations Management.

The title is a required attribute in HP Service Manager, and is set, independently of the flags above, using the title given in Operations Management during the creation of the incident.

## Mapping OPR Lifecycle States to BDM Lifecycle States

Individual OPR event state and Service Manager incident status changes may be selected for synchronization. Out of the box, only the "closed" state is synchronized in both directions. To change this behavior add the desired states to the appropriate list, SyncOPRStatesToSM or SyncSMStatusToOPR .

Here are two examples:

- ```
private static final Set SyncOPRStatesToSM = ["closed", "in_progress", "resolved"]
```
- ```
private static final Set SyncSMStatusToOPR = ["closed", "resolved"]
```

In the example the OPR event lifecycle states closed, in\_progress and resolved are synchronized to the Service Manager incident status, and Service Manager incident status closed and resolved are synchronized to the OPR event state.

**Note:** The special state "\*" denotes all states, so to synchronize all OPR event states to the SM incident status property specify the following:

```
private static final Set SyncOPRStatesToSM = ["*"]
```

Additionally two maps are used to specify the mapping of OPR event lifecycle state to BDM incident status. The maps are named MapOPR2SMStatus and MapSM2OPRState. Out of the box, all possible states have a mapping.

Here is an example:

- ```
private static final Map MapOPR2SMStatus = ["open": "open", "in_progress": "work-in-progress", "resolved": "resolved", "closed": "closed"]
```
- ```
private static final Map MapSM2OPRState = ["accepted": "open", "assigned": "open", "open": "open", "reopened": "open", "pending-change": "in_progress", "pending-customer": "in_progress", "pending-other": "in_progress", "pending-vendor": "in_progress", "referred": "in_progress", "suspended": "in_progress", "work-in-progress": "in_progress", "rejected": "resolved", "replaced-problem":
```

```
"resolved",
```

```
"resolved": "resolved", "cancelled": "resolved", "closed": "closed"]
```

## **Syntax Errors**

If you get a syntax error when customizing your Groovy scripts, you will get an event in the event browser with a detailed description of the error. In addition you may view the log file `opr-event-sync-adapter.log` for information about how to resolve the error. You can find the log file here:

`<Gateway Server root directory>/log/opr-event-sync-adapter.log`



# View Changes and Incidents in Service Health Using Standalone HP Universal CMDB

This integration enables you to view planned changes and incident details in the Changes & Incidents tab in the 360° View page in Service Health, when you are using a standalone HP Universal CMDB.

This task describes how to configure the HP Service Manager - BSM federated integration in order to allow both products to share information and data.

**Note:** Beginning with UCMDB version 9.05, a new SM adapter (ServiceManagerAdapter9-x) is supplied with UCMDB out of the box, in addition to the legacy adapter (ServiceManagerAdapter7-1).

- For SM versions 9.30 and 9.31, use ServiceManagerAdapter9.xx.
- For SM versions 9.20 and earlier, use ServiceManagerAdapter7-1.

This section includes the following:

- ["Prerequisites" on the next page](#)
- ["Step 1: Load .unl Files to Provide External Access to Service Manager" on the next page](#)
- ["Step 2: Configure the Service Desk Adapter Time Zone" on page 568](#)
- ["Step 3: Verify that the UCMDB is the Global ID Generator" on page 569](#)
- ["Step 4 \(for SM 9.20 and earlier only\): Add a Domain" on page 570](#)
- ["Step 5: Configure SM Adapter in UCMDB" on page 570](#)
- ["Step 6: Configure the SM-UCMDB Integration: Create an Integration Point" on page 571](#)
- ["Step 7: Configure the SM-UCMDB Integration: Set Up Data Push Jobs" on page 572](#)
- ["Step 8: Configure the SM-UCMDB Integration: Run Data Push Jobs" on page 573](#)
- ["Step 9: Configure the SM-UCMDB Integration: Add UCMDB Connection Information to SM" on page 573](#)

- "Step 10: Configure the BSM-UCMDB Integration: Deploy CMS\_to\_RTSM\_Sync.zip on UCMDB" on page 574
- "Step 11: Configure the BSM-UCMDB Integration: Create an Integration Point on BSM" on page 574
- "Step 12: Configure the BSM-UCMDB Integration: Create an Integration Point on the CMS" on page 577
- "Step 13 (Optional): Add CI Types to the Service Health Changes and Incidents Component" on page 579
- "Step 14 (Optional): Map Siebel Application CITs" on page 579
- "Result" on page 580
- "Troubleshooting" on page 580

## Prerequisites

- **Data-Flow Probes (for SM 9.3x).** If you are using SM 9.30 or 9.31, before you begin you must install *two* data-flow probes - one with UCMDB as its target, and another with the BSM Gateway Server as its target. When you configure the integration points, you will select these probes.
- **Trusted Sign-on and LW-SSO.** If you want HP Service Manager to use the SSL-based Trusted Sign-on protocol and LW-SSO, configure it according to the instructions in the HP Service Manager online help if you have not already done so. In addition, see *Configuring HP Service Manager to Use the SSL-based Trusted Sign-On and LW-SSO* in the Service Manager documentation library.

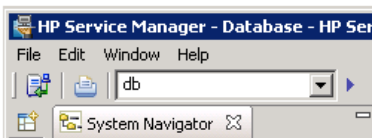
## Step 1: Load .unl Files to Provide External Access to Service Manager

This procedure enables BSM to query incidents and changes:

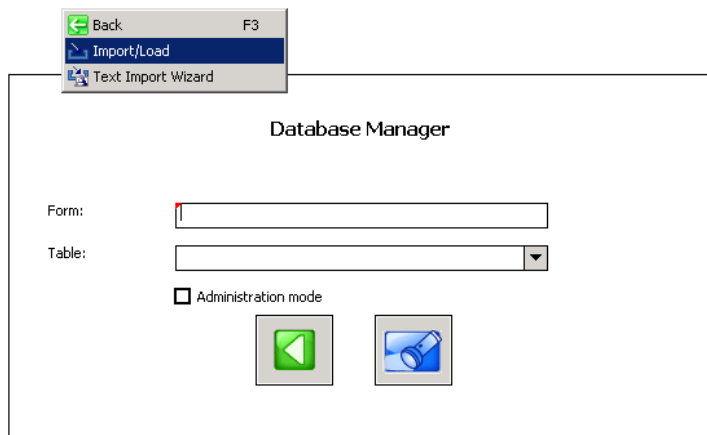
1. Copy the following files from the BSM 9.x DVD to a local directory:
  - SM\_Integration/SM\_Unloads/SM7.1/ucmdbIntegration7\_1x.unl
  - SM\_Integration/SM\_Unloads/SM7.1/BACExtAccess\_71\_v1.unl
2. Before loading these .unl files, apply the fix described in <http://support.openview.hp.com/selfsolve/document/KM1015767>. This is required because the .unl

file expects the name attribute in the EXTACCESSM1 table to be length 50 in the database, but its default out-of-the-box length is 100. You therefore need to reduce the size of the attribute, load the unl file, then increase the size again. These steps are for the SQL Server, but you can refer to the KM document for the equivalent Oracle syntax.

- a. Database field truncation may result in data loss if data in the field exceeds the default length, so first check the size of data in the field: `Select NAME, LEN(NAME) from EXTACCESSM1 order by 2 desc`
  - b. Reduce the size of the field: `alter table EXTACCESSM1 alter column NAME VARCHAR(50)`
  - c. Load the ucmdbIntegration7\_1x.unl file as described in the following steps. When you are done, you will increase the size of the field back to what it was originally.
3. In Service Manager, type **db** in the command line text widget in the menu bar at the top of the client display.



4. Right-click the white background and select **Import/Load** from the context menu that appears.



5. Click the folder icon at the end of the File Name box. and navigate to the .unl file you copied from BSM. Select the file, and click **Open**.

File Name:

Import Descriptor:

File Type:

During a foreground load, display status for:

All Messages

Totals Only

None

6. Click **Load FG** on the toolbar to load the file. If you receive a message saying "The file you are loading will change the keys...", click **Yes**.
7. Increase the size of the field back to what it was originally: `alter table EXTACCESSM1 alter column NAME VARCHAR(100)`
8. Repeat the above steps for the BACExtAccess\_71\_v1.unl file.

## Step 2: Configure the Service Desk Adapter Time Zone

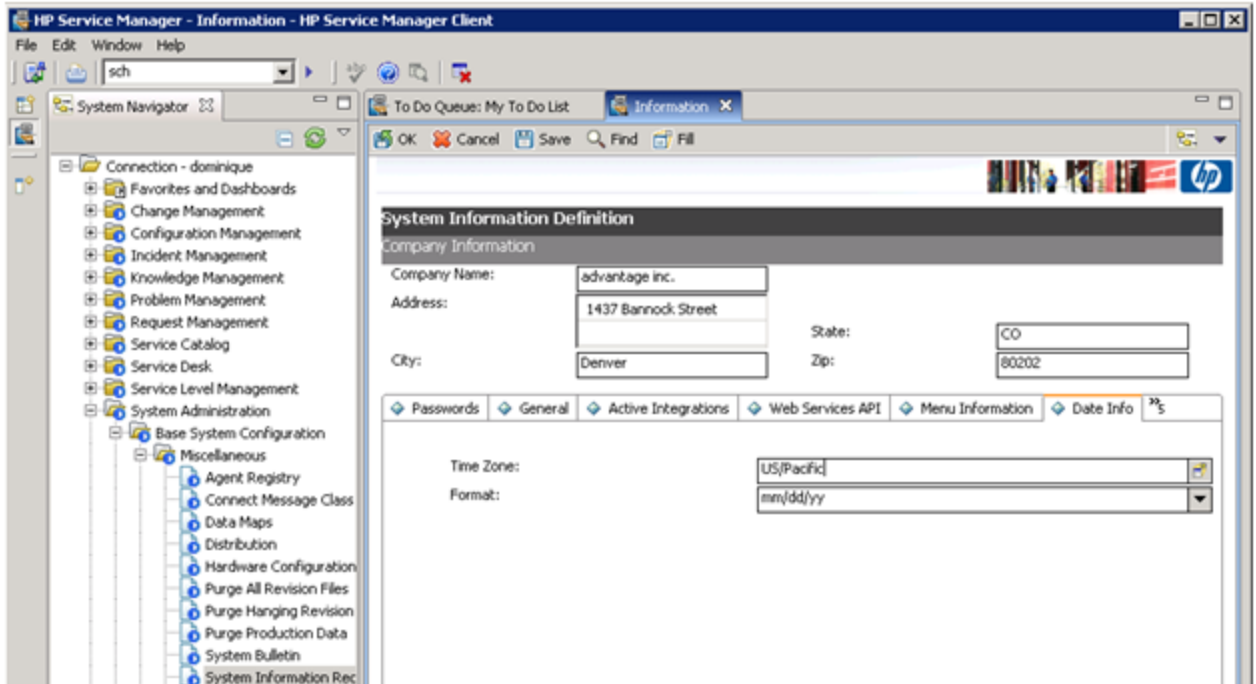
Configure the time zone so Incidents and Planned Changes have the correct time definitions:

1. In Service Manager, select **Navigation pane > Menu navigation > System Administration > Base System Configuration > Miscellaneous > System Information Record**.
2. Within the **Date Info** tab, open the <BSM DPS root directory>/odb/runtime/fcmdb/CodeBase/<ServiceManagerAdapter9-x or ServiceDeskAdapter7-1>/serviceDeskConfiguration.xml file.
3. Find the row that includes the following string:

```
<globalConnectorConfig><![CDATA[<global_configuration><date_pattern>MM/dd/yy
HH:mm:ss</date_pattern><time_zone>US/Pacific</time_zone>
```

and check the date and time format, and time zone. Note that the date is case-sensitive. Change either Service Manager or the xml file so that they both match each other's settings.

**Note:** Specify a time zone from the Java time zone list that matches the time zone used in Service Manager; for example, America/New York.




4. Restart the corresponding server to make the change take effect. (If you changed the time zone on SM, restart the Service Manager server; if you changed the time zone on BSM, restart the BSM server.)

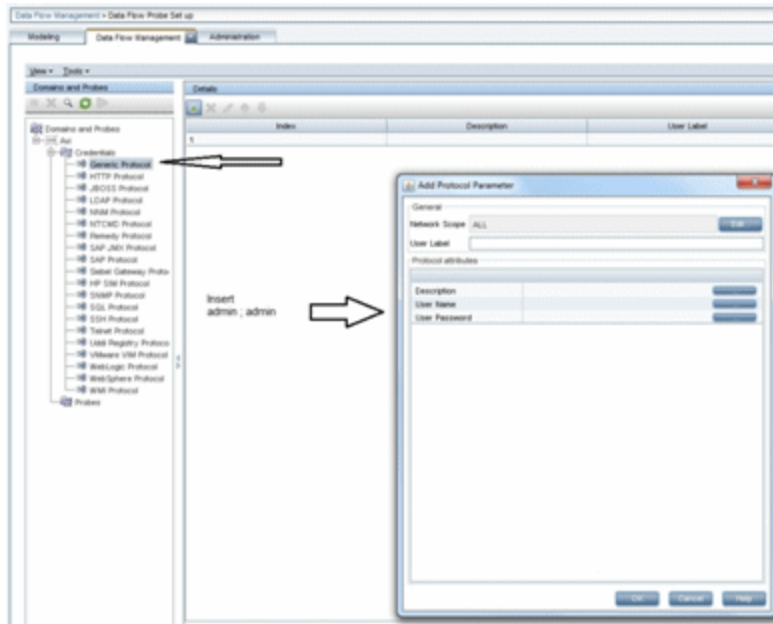
### Step 3: Verify that the UCMDB is the Global ID Generator

1. Log in to the JMX Console (<http://<UCMDB server>:8080/jmx-console/>).
2. Go to **Multiple UCMDB Instances Services**.
3. Click **getIsGlobalIdGenerator**. Verify that the call returns **true**. For more details, refer to the RTSM Best Practices guide.

For SM versions 9.20 and earlier, proceed with the next step. For SM versions 9.30 and 9.31, skip to "[Step 5: Configure SM Adapter in UCMDB](#)" on the next page.

## Step 4 (for SM 9.20 and earlier only): Add a Domain

1. In BSM, select **Admin > RTSM Administration**, click the **Data Flow Management** tab, and select **Data Flow Probe Setup**.
2. In the **Domains and Probes** pane, click .
3. In the **Add New Domain** dialog box, enter a new domain name and click **OK**. This creates a new domain and its protocols.
4. Within the domain you added, select **Credentials > Generic Protocol**, and click the **Add new connection details** button in the right pane. In the **Add Protocol Parameter** dialog box that opens, insert the SM administrator credentials.



## Step 5: Configure SM Adapter in UCMDB

1. Within the UCMDB user interface, access **Data Flow Management > Adapter Management**.
2. In the resources window, select **ServiceManagerAdapter9-x** or **ServiceManagerAdapter7-1 > Configuration files**.
3. Select **ServiceManagerAdapter9-x/sm.properties** or **ServiceManagerAdapter7-1/sm.properties**.

4. In the window on the right side of the screen, modify the **use.global.id** parameter, set it to **false**, and click **OK**.

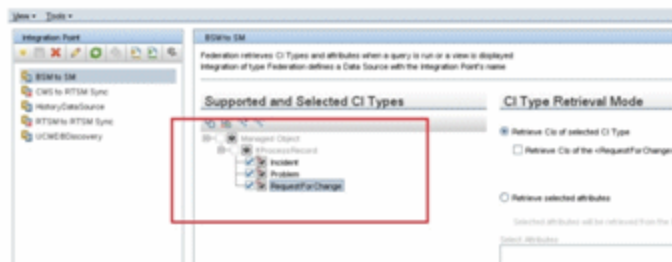
## Step 6: Configure the SM-UCMDB Integration: Create an Integration Point

1. Within the UCMDB user interface, select **Data Flow Management > Integration Studio**.
2. In the Integration Point pane, select **Create New Integration Point**. The Create New Integration Point dialog box opens. Enter the following:

Name	Recommended Value	Description
<b>Integration Name</b>	<b>SM Integration</b>	The name you give to the integration point.
<b>Adapter</b>	<b>&lt;user defined&gt;</b>	Select the appropriate adapter for the version of SM that you are using.
<b>Is Integration Activated</b>	<b>selected</b>	Select this check box to create an active integration point.
<b>Hostname/IP</b>	<b>&lt;user defined&gt;</b>	The name of the SM server.
<b>Port</b>	<b>&lt;user defined&gt;</b>	The port through which you access SM.
<b>Credentials</b>	<b>&lt;user defined&gt;</b>	<ul style="list-style-type: none"> <li>■ For SM 9.20 and earlier, select the user credentials created in <a href="#">"Step 4 (for SM 9.20 and earlier only): Add a Domain"</a> on the previous page.</li> <li>■ For SM 9.30 and 9.31, in the default domain select Generic Protocol, and enter the credentials of the SM administrator.</li> </ul>
<b>Probe Name</b> (for ServiceManagerAdapter9-x only)	<b>&lt;user defined&gt;</b>	If you are using ServiceManagerAdapter9-x, select the probe which reports to CMS (see <a href="#">"Prerequisites"</a> on page 566).

**Note:** It is recommended to click the **Test Connection** button to verify that the details entered are working before continuing.

3. In the **Integration Point** pane, click the Integration Point you just created, and click the **Federation** tab in the right pane.
4. In the **Supported and Selected CI Types** area, verify the **Incident**, **Problem**, and **Request for Change** CITs are selected.



## Step 7: Configure the SM-UCMDB Integration: Set Up Data Push Jobs

Depending on your adapter version, perform the following:

### For ServiceManagerAdapter9-x:

1. Edit the **SM Push** job, and select **Scheduler Definition**.
2. For the **Repeat** field, you can select **Changes Sync/All Data Sync**.
3. Set the **Repeat Every** field to **1 Day**, and click **OK**.

### For ServiceManagerAdapter7-1:

1. Edit the **SM Topology Comparison Push** job, and select **Scheduler Definition**.
2. For the **Repeat** field, select **interval**.
3. Set the **Repeat Every** field to **1 Day**, and click **OK**.
4. Edit the **SM History-based Push** job, and select **Scheduler Definition**.
5. For the **Repeat** field, select **interval**.
6. Set the **Repeat Every** field to **1 Day**, and click **OK**.



## Step 8: Configure the SM-UCMDB Integration: Run Data Push Jobs

1. In the Integration Point pane, select the correct integration.
2. Select the **Data Push** tab. The Job Definition pane is displayed.
3. Select your job and click **Synchronize All** to run the push job.

**Note:** For ServiceManagerAdapter7-1, run this first for the **SM History-based Push** job, then repeat for the **SM Topology Comparison Push** job.

4. When the Confirm synchronizing window is displayed, click **Yes**.
5. Click the **Statistics** tab to view the progress of the synchronization.
6. Click **Refresh** to view the updated synchronization status.

## Step 9: Configure the SM-UCMDB Integration: Add UCMDB Connection Information to SM

1. Log on to your UCMDB system as an administrator. Verify that all UCMDB services are running.
2. Log on to your SM system as an administrator.
3. Select **System Administration > Base System Configuration > Miscellaneous > System Information Record**.
4. Select the **Active Integrations** tab.
5. Select the **HP Universal CMDB** option. The form displays the UCMDB Web service URL field.
6. In the UCMDB Web service URL field, enter the URL to the UCMDB Web service API. The URL has the following format:

**http://<UCMDB server name>:<port>/axis2/services/ucmdbSMService.**

7. In the UserId dialog box, enter your UCMDB user name and password and click **Save**.

## Step 10: Configure the BSM-UCMDB Integration: Deploy CMS\_to\_RTSM\_Sync.zip on UCMDB

1. Copy the file CMS\_to\_RTSM\_Sync.zip located on the BSM-DPS machine file system under **HPBSM\odb\conf\factory\_packages** to the file system on the UCMDB machine.
2. Within the UCMDB user interface, select the **Administration** tab.
3. Select **Package Manager > Deploy Packages to server (from local disk)**.
4. Click the **Add** button and select the file **CMS\_to\_RTSM\_Sync.zip** through the file system browser.
5. Select **Deploy**.

## Step 11: Configure the BSM-UCMDB Integration: Create an Integration Point on BSM

1. Within the BSM user interface, select **RTSM Administration > Data Flow Management > Integration Studio**.
2. In the Integration Point pane, select **Create New Integration Point**. The Create New Integration Point dialog box opens. Enter the following:

	<b>Recommended</b>	
<b>Name</b>	<b>Value</b>	<b>Description</b>
<b>Integration Name</b>	<b>&lt;user defined&gt;</b>	The name you give to the integration point.
<b>Adapter</b>	<b>UCMDB 9.x</b>	Select the adapter type from the drop-down list.
<b>Is Integration Activated</b>	<b>selected</b>	Select this check box to create an active integration point.
<b>Hostname/IP</b>	<b>&lt;user defined&gt;</b>	The name of the UCMDB server, load balancer, or reverse proxy.
<b>Port</b>	<b>&lt;user defined&gt;</b>	The port through which you access UCMDB, load balancer, or reverse proxy.

Name	Recommended Value	Description
<b>Credentials</b>	<user defined>	<p>If credentials appear in the Credentials column, select them.</p> <p>If no credentials appear, select <b>Generic Protocol</b> and click the <b>Add new connection details for selected protocol type</b> button.</p> <p>Enter the following information:</p> <ul style="list-style-type: none"> <li>■ <b>Description.</b> Enter <b>UCMDB</b>.</li> <li>■ <b>User Name.</b> Enter the UCMDB user name. The default value is <b>admin</b>.</li> <li>■ <b>User Password.</b> Enter and confirm a password.</li> </ul>
<b>Probe Name</b> (for ServiceManagerAdapter9-x only)	<user defined>	<p>If you are using ServiceManagerAdapter9-x, select the probe which reports to <i>BSM</i> (see <a href="#">"Prerequisites" on page 566</a>).</p>

3. Click the **Add** icon on the right side of the window and add Job definitions as follows:
  - a. Name the **Job definition**.
  - b. Select the **Allow Delete** check box.
  - c. Click the **Add** icon in the Job definition window.
  - d. From the pop up window, browse to **root - CMS sync** and select the **ActiveDirectory\_sync** job and click **OK**.
  - e. Select the **Scheduler definition** check box.
  - f. In the Repeat window, select **Cron**.
  - g. For the Cron expression, enter the following string: **\* 0/10 \* \* \* ? \***.
  - h. Adjust other settings as needed.
  - i. When finished, click **OK** and save the integration.

- j. Repeat steps **a** to **i** and configure the following jobs:
  - **FailoverCluster\_Sync**
  - **IIS\_Sync**
  - **SOA\_Sync**
  - **BusinessAndFacilities\_Sync**
  - **ExchangeServer\_Sync**
  - **Virtualization\_Sync**
  - **Siebel\_Sync**
  - **Credentials\_Sync**
  - **Basicinfrastructure\_Sync**
  - **J2EE\_Sync**
  - **SAP\_Sync**
4. Browse to UCMDB on port 21212 (for example, [http://<DPS\\_host>.<domain>:21212](http://<DPS_host>.<domain>:21212)), and select the **JMX Console**.
5. Log on to the JMX console.
6. From the UCMDB section, select **UCMDB:service=Multiple CMDB Instances Services**.
7. Invoke:
  - a. **setAsGlobalIdGenerator** and verify it succeeded.
  - b. **getGlobalIdGeneratorScopes** and verify it succeeded.
8. Within BSM, access **RTSM Administration > Data Flow Management > Integration Studio**.
9. Select the Integration Point that you have configured.
10. In the Job definition section, click **Synchronize All** to run the synchronization.

The Integration Point should be active and the jobs are displayed properly.

## Step 12: Configure the BSM-UCMDB Integration: Create an Integration Point on the CMS

1. Log into UCMDB and select **Data Flow Management > Integration Studio**.
2. In the Integration Point pane, select **Create New Integration Point**. The Create New Integration Point dialog box opens. Enter the following:

	<b>Recommended</b>	
<b>Name</b>	<b>Value</b>	<b>Description</b>
<b>Integration Name</b>	<b>&lt;user defined&gt;</b>	The name you give to the integration point.
<b>Adapter</b>	<b>UCMDB 9.x</b>	Select the adapter type from the drop-down list.
<b>Is Integration Activated</b>	<b>selected</b>	Select this check box to create an active integration point.
<b>Hostname/IP</b>	<b>&lt;user defined&gt;</b>	The name of the BSM server, load balancer, or reverse proxy.
<b>Port</b>	<b>&lt;user defined&gt;</b>	The port through which you access BSM, load balancer, or reverse proxy.
<b>Credentials</b>	<b>&lt;user defined&gt;</b>	<p>If credentials appear in the Credentials column, select them.</p> <p>If no credentials appear, select <b>Generic Protocol</b> and click the <b>Add new connection details for selected protocol type</b> button.</p> <p>Enter the following information:</p> <ul style="list-style-type: none"> <li>■ <b>Description.</b> Enter <b>UCMDB</b>.</li> <li>■ <b>User Name.</b> Enter the UCMDB user name. The default value is <b>admin</b>.</li> <li>■ <b>User Password.</b> Enter and confirm a password.</li> </ul>

<b>Name</b>	<b>Recommended Value</b>	<b>Description</b>
<b>Probe Name</b> (for ServiceManagerAdapter9-x only)	<user defined>	If you are using ServiceManagerAdapter9-x, select the probe which reports to the <i>CMS</i> (see <a href="#">"Prerequisites" on page 566</a> ).

3. Click the **Add** icon on the right side of the window and add Job definitions as follows:
  - a. Name the **Job definition**.
  - b. Select the **Allow Delete** check box.
  - c. Click the **Add** icon in the Job definition window.
  - d. From the pop up window, browse to **root - CMS sync** and select the **ActiveDirectory\_sync** job and click **OK**.
  - e. Select the **Scheduler definition** check box.
  - f. In the Repeat window, select **Cron**.
  - g. For the Cron expression, enter the following string: **\* 0/10 \* \* \* ? \***.
  - h. Adjust other settings as needed.
  - i. When finished, click **OK** and save the integration.
  - j. Repeat steps **a** to **i** and configure the following jobs:
    - **FailoverCluster\_Sync**
    - **IIS\_Sync**
    - **SOA\_Sync**
    - **BusinessAndFacilities\_Sync**
    - **ExchangeServer\_Sync**
    - **Virtualization\_Sync**
    - **Siebel\_Sync**

- **Credentials\_Sync**
  - **Basicinfrastructure\_Sync**
  - **J2EE\_Sync**
  - **SAP\_Sync**
4. Browse to UCMDB on port 8080 (for example, <http://yourUCMDBhost.domain:8080>), and select the **JMX Console**.
  5. Log on to the JMX console.
  6. From the UCMDB section, select **UCMDB:service=Multiple CMDB Instances Services**.
  7. Invoke:
    - a. **setAsGlobalIdGenerator** and verify it succeeded.
    - b. **getGlobalIdGeneratorScopes** and verify it succeeded.
  8. Within UCMDB, access **Data Flow Management > Integration Studio**.
  9. Select the Integration Point that you have configured.
  10. In the Job definition section, click **Synchronize All** to run the synchronization.

The Integration Point should be active and the jobs are displayed properly.

## Step 13 (Optional): Add CI Types to the Service Health Changes and Incidents Component

By default, BSM Service Health displays information on incidents and requests for change for the following CI types: Business Service, Siebel Application, Business Application, and Node.

If you want to view change and incident information for other CITs, perform the procedure described in ["How to Customize the Changes and Incidents Component" on page 591](#).

## Step 14 (Optional): Map Siebel Application CITs

To create a mapping between the **Hand Held Devices** or **Display Device** CIT in Service Manager with **Siebel Application** CITs in BSM, perform one of the following procedures:

- In Service Manager, select **Main page > To Do > Queue: Configuration Item > New > New** and click **Device**. In the Configuration Item field enter the exact name (case sensitive) of the BSM CI that corresponds to the **Siebel Application** CIT in BSM.
- Create a new population job that includes the **Hand Held Devices** or **Display Device** CIT. Those CITs correspond to the Siebel application CITs. For details about how to create a population job, see "Data Push Tab" in the *Modeling Guide*.

## Result

You can now view planned changes and incident details in the Changes & Incidents tab in the 360° View page in Service Health.

Both products can now share information and data.

## Troubleshooting

If you are not seeing expected incidents in BSM, perform the following:

1. On the Data Processing Server, search the **odb\odb\Error.log** file for **Error Code 802**.
2. In this error message, locate the following string: **property [<category or incident\_status>=<attribute value>[STRING] ] is defined as attribute.**

This indicates that a certain attribute value is missing in RTSM.

3. Access **RTSM Administration > CI Type Manager**.
4. From the **CI Types** menu, select **System Type Manager**, and open **Category** or **Incident Status** (depending on the error message) for editing.
5. Click the Add button (+), and add the missing attribute value (exactly as it appears in the error message) to the list of values.



# View Changes and Incidents in Service Health Using RTSM

This integration enables you to view planned changes and incident details in the Changes & Incidents tab in the 360° View page in Service Health, when you are working with RTSM. For details, see "Changes and Incidents" in the Service Health part of the *BSM User Guide*.

This section includes the following:

- ["Prerequisite" below](#)
- ["Step 1: Configure the Service Desk Adapter Time Zone" below](#)
- ["Step 2: Create an Integration User Account in Service Manager" on page 583](#)
- ["Step 3: Add the BSM Connection Information in Service Manager" on page 584](#)
- ["Step 4: Create an Integration Point in BSM" on page 584](#)
- ["Step 5: Create New Jobs to Synchronize Between BSM and Service Manager" on page 587](#)
- ["Step 6: Run the Job" on page 587](#)
- ["Step 7: Test the Configuration" on page 587](#)
- ["Step 8 \(Optional\): Add CI Types to the Service Health Changes and Incidents Component" on page 590](#)
- ["Troubleshooting" on page 590](#)

## Prerequisite

If you are using SM versions 9.30 or 9.31, before you begin you must install a data-flow probe with the BSM Gateway Server as its target. When you configure the integration point, you will select this probe for the integration.

## Step 1: Configure the Service Desk Adapter Time Zone

Configure the time zone so Incidents and Planned Changes have the correct time definitions:

1. In Service Manager, select **Navigation pane > Menu navigation > System Administration > Base System Configuration > Miscellaneous > System Information Record**.

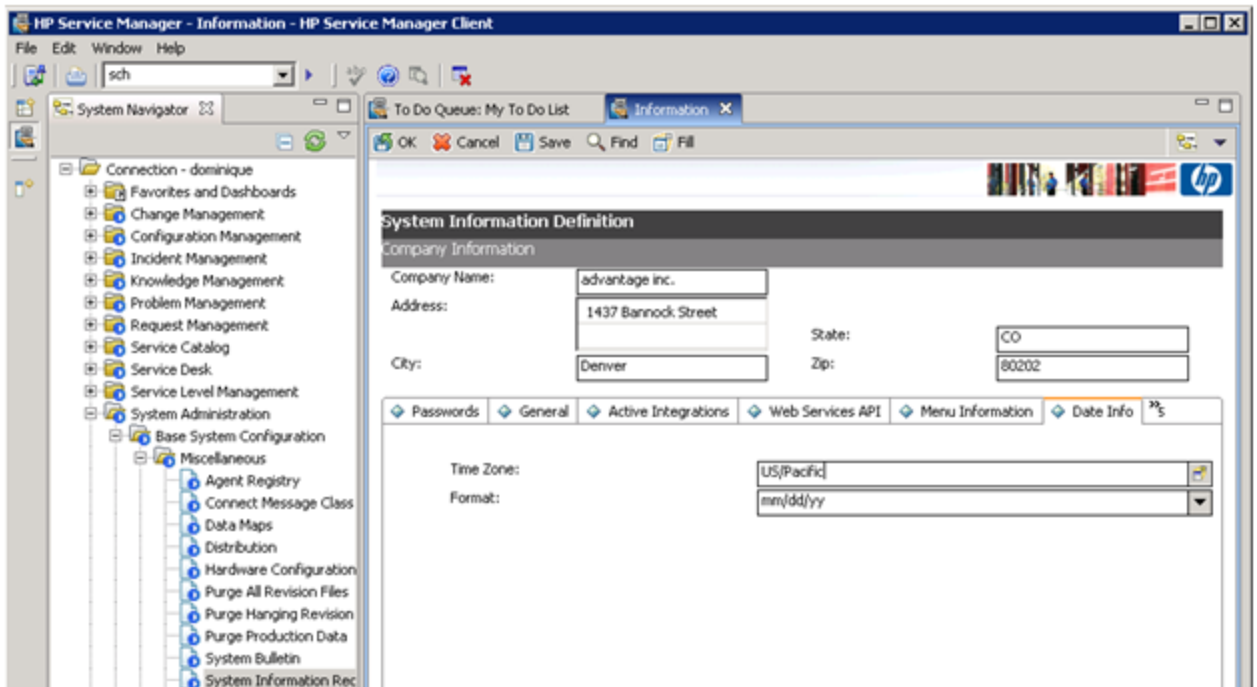
2. Within the **Date Info** tab, open the <BSM DPS root directory>/odb/runtime/fcldb/CodeBase/ServiceManagerAdapter9-x or ServiceDeskAdapter7-1/serviceDeskConfiguration.xml file.

3. Find the row that includes the following string:

```
<globalConnectorConfig><![CDATA[<global_configuration><date_pattern>MM/dd/yy
HH:mm:ss</date_pattern><time_zone>US/Pacific</time_zone>
```

and check the date and time format, and time zone. Note that the date is case-sensitive. Change either Service Manager or the xml file so that they both match each other's settings.

**Note:** Specify a time zone from the Java time zone list that matches the time zone used in Service Manager; for example, America/New York.



4. Restart the corresponding server to make the change take effect. (If you changed the time zone on SM, restart the Service Manager server; if you changed the time zone on BSM, restart the BSM server.)

## Step 2: Create an Integration User Account in Service Manager

This integration requires an administrator user account for BSM to connect to Service Manager. This user account must already exist in both BSM and Service Manager.

To create a dedicated integration user account in Service Manager:

1. Log in to Service Manager as a system administrator.
2. Type **contacts** in the Service Manager command line, and press ENTER.
3. Create a new contact record for the integration user account.
  - a. In the **Full Name** field, type a full name. For example, RTSM.
  - b. In the **Contact Name** field, type a name. For example, RTSM.
  - c. Click **Add**, and then OK.
4. Type **operator** in the Service Manager command line, and press ENTER.
5. In the **Login Name** field, type the username of an existing system administrator account, and click **Search**.

The system administrator account displays.

6. Create a new user account based on the existing one:
  - a. Change the **Login Name** to the integration account name you want (for example, rtsm).
  - b. Type a **Full Name**. For example, RTSM.
  - c. In the **Contact ID** field, click the **Fill** button and select the contact record you have just created.
  - d. Click **Add**.
  - e. Select the **Security** tab, and change the password.
  - f. Click **OK**.

The integration user account is created. Later you will need to add this user account (username/password) in RTSM, and then specify this user account in the **Credentials ID** field when creating an integration point in RTSM administration.

## Step 3: Add the BSM Connection Information in Service Manager

The integration requires the BSM connection information to obtain CI attribute information from the BSM system, and display it in the Actual State section in the Service Manager configuration item form.

1. Log in to Service Manager as a system administrator.
2. Click **System Administration > Base System Configuration > Miscellaneous > System Information Record**.
3. Click the **Active Integrations** tab.
4. Select the **HP Universal CMDB** option.

The form displays the UCMDB web service URL field.

5. In the UCMDB webservice URL field, type the URL to the HP Universal CMDB web service API. The URL has the following format: **http://<UCMDB server name>:<port>/axis2/services/ucmdbSMService**

Replace <UCMDB server name> with the host name of your BSM server, and replace <port> with the communications port your BSM server uses.

6. In **UserId** and **Password**, type the user credentials required to manage CIs on the BSM system. For example, the out-of-the-box administrator credentials are **admin/ admin**.
7. Click **Save**. Service Manager displays the message: **Information record updated**.
8. Log out of the Service Manager system.
9. Log back into the Service Manager system with an administrator account. The **Actual State** section will be available in CI records pushed from BSM.

## Step 4: Create an Integration Point in BSM

A default RTSM 9.05 installation already includes the ServiceManagerAdapter9-x package. To use the integration package, you must create an integration point listing the connection properties for the integration.

To create an integration point:

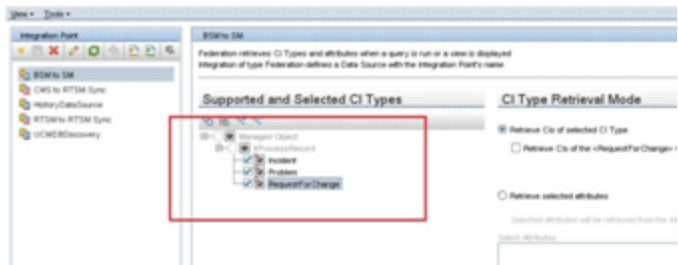
1. Access the JMX console (in case of distributed deployment) on the DPS server.
2. Navigate to **UCMDB:service=Security Services**.
3. Create a new user with the name and password that you created in SM, using the JMX **createUser**:
  - **CustomerId** = 1
  - **userName** = <userName>
  - **password** = <password>
4. Assign the user Administrator Role using the JMX **setRolesForUser** from the same section:
  - **CustomerId** = 1
  - **userName** = <userName>
  - **roles** = Admin
5. In BSM, select **Admin > RTSM Administration**, click the **Data Flow Management** tab, and select **Integration Studio**.
6. In the Integration Point pane, select **Create New Integration Point**. The Create New Integration Point dialog box opens. Enter the following:

Name	Recommended Value	Description
<b>Integration Name</b>	<b>SM Integration</b>	The name you give to the integration point.
<b>Adapter</b>	<user defined>	Select HP BTO Products > Service Manager > <b>Service Manager 9.xx</b> .  This adapter, which supports CI/ relationship Data Push from RTSM to Service Manager, and Population and Federation from Service Manager to RTSM.
<b>Is Integration Activated</b>	<b>selected</b>	Select this check box to create an active integration point.

Name	Recommended Value	Description
<b>Hostname/IP</b>	<b>&lt;user defined&gt;</b>	The name of the SM server.
<b>Port</b>	<b>&lt;user defined&gt;</b>	The port through which you access SM.
<b>Credentials</b>	<b>&lt;user defined&gt;</b>	Click <b>Generic Protocol</b> , click the <b>Add</b> button to add the integration user account you created in " <a href="#">Step 2: Create an Integration User Account in Service Manager</a> " on page 583, and then select it. This account must exist in both Service Manager and BSM.
<b>Probe Name</b> (for ServiceManagerAdapter9-x only)	<b>&lt;user defined&gt;</b>	Select the probe that you installed for this integration.

**Note:** It is recommended to click the **Test Connection** button to verify that the details entered are working before continuing.

- In the **Integration Point** pane, click the Integration Point you just created, and click the **Federation** tab in the right pane.
- In the **Supported and Selected CI Types** area, verify the **Incident, Problem, and Request for Change** CITs are selected.

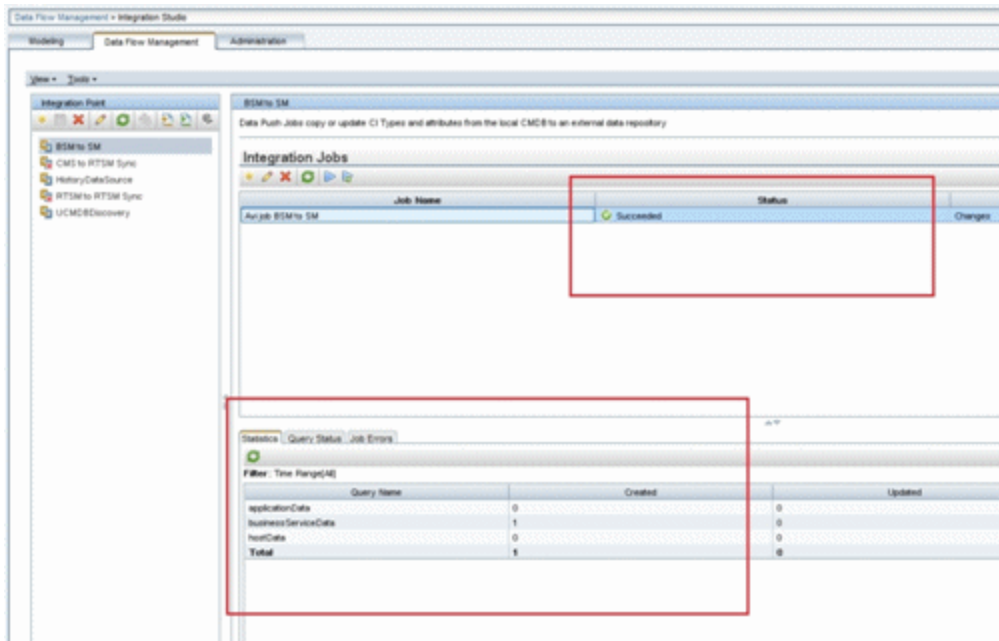


## Step 5: Create New Jobs to Synchronize Between BSM and Service Manager


1. In the same location as step 5 above, click the **Data Push** tab.
2. In the New Integration Job dialog box, click the + icon on the left.
3. In the Available Queries dialog box, select the relevant queries for the job.

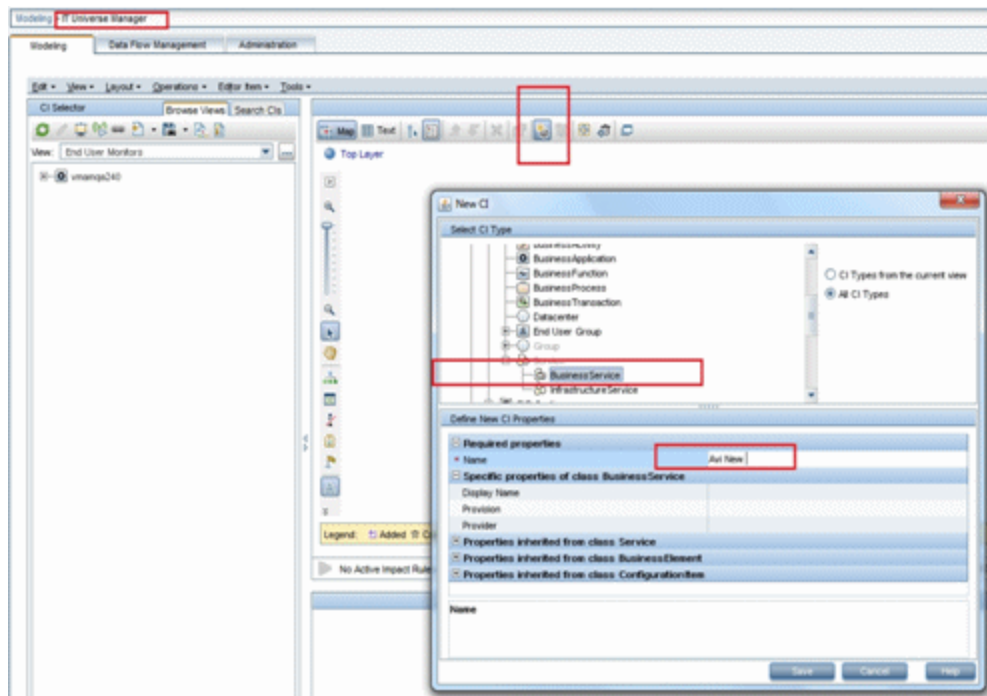
## Step 6: Run the Job

When you run the job, the CIs are synchronized between BSM and Service Manager.



## Step 7: Test the Configuration

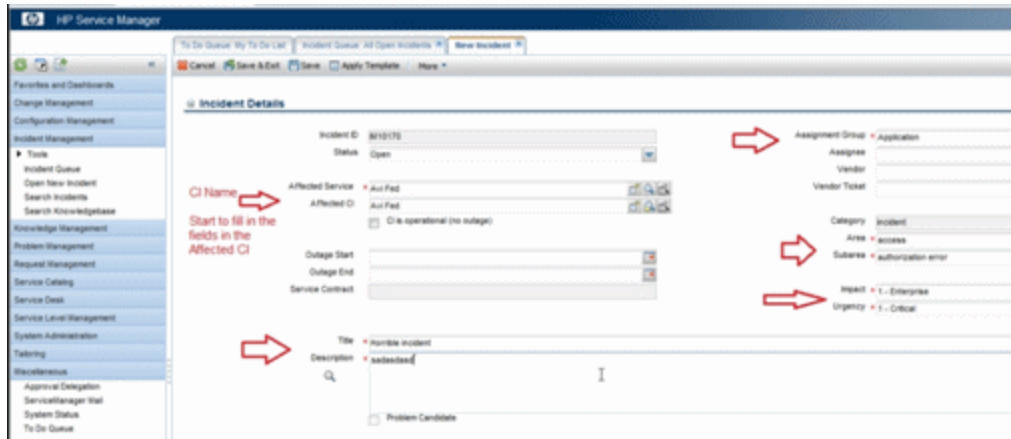
1. In BSM, select **Admin > RTSM Administration**, click the **Modeling** tab, and select **IT Universe Manager**.
2. In the **CI Selector** pane, select the relevant view, and click  in the right pane.
3. In the **New CI** dialog box that opens, create a new CI with the **BusinessService** type.



4. Create a TQL in **Admin > Service Health > View Builder** that includes only BusinessService CI Types (CITs).
5. Click the **Calculate** button. The relevant CI appears in view.
6. Click the **Data Push** tab, and run the job in order to synchronize with Service Manager. A message that the job was successful should be issued.
7. In Service Manager, create a new incident for the new CI that you created above:
  - a. Select **Incident Management > Open New Incident**.
  - b. **Important:** Start by entering the name of the CI you want to attach to the incident in the **Affected CI** field. This creates the Incident Id.
  - c. Enter the CI name in the **Affected Service** field and click to search.

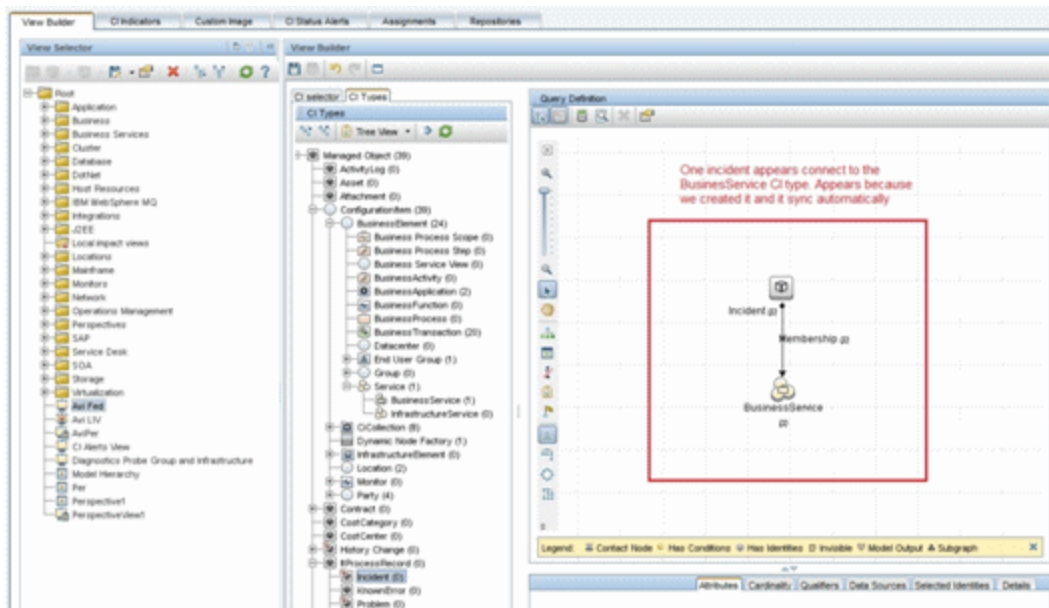


- d. Enter any incident detail.

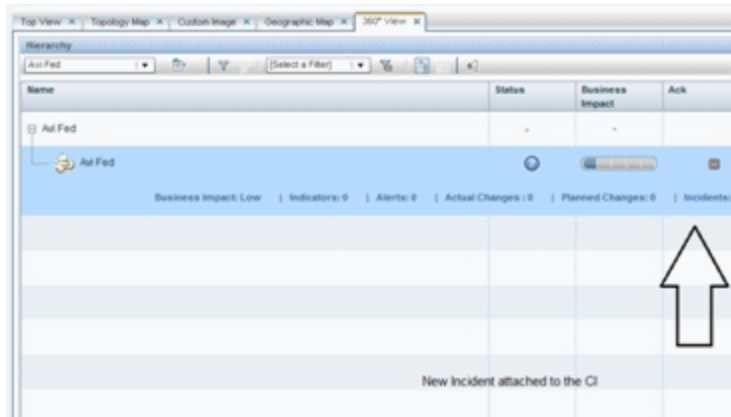


The incident is automatically attached to the CI.

8. In BSM, create a TQL with the CI Type you created connected to the Incident CI Type in a membership relationship link.
9. Click the **Calculate** button. One incident appears connected to the BusinessService CI Type because this test created it and it is synchronized automatically.



10. Delete the incident from the TQL and save the TQL to be a view. The TQL is only used for the test.
11. Select **Application > Service Health**, and click the **360 View** tab. Check that the new incident is attached to the CI.



## Step 8 (Optional): Add CI Types to the Service Health Changes and Incidents Component

By default, BSM Service Health displays information on incidents and requests for change for the following CI types: Business Service, Siebel Application, Business Application, and Node.

If you want to view change and incident information for other CITs, perform the procedure described in ["How to Customize the Changes and Incidents Component" on page 591](#).

## Troubleshooting

If you are not seeing expected incidents in BSM, see ["Troubleshooting" on page 580](#)

# How to Customize the Changes and Incidents Component

By default, incidents and requests for change are displayed for the following CI types: Business Service, Siebel Application, Business Application, and Node. If you want to view change and incident information for other CITs, perform the following procedure:

1. Within **Admin > RTSM Administration > Modeling Studio**, copy one of the TQLs within the **Console** folder, and save your copy with a new name. These default TQLs perform the following:

TQL name	Description
CollectTicketsWithImpacts	Retrieves Service Manager incidents for the selected CI, and for its child CIs which have an Impact relationship.
CollectTicketsWithoutImpacts	Retrieves Service Manager incidents for the selected CI.
CollectRequestForChangeWithImpacts	Retrieves Service Manager requests for change, for the selected CI, and for its child CIs which have an Impact relationship.
CollectRequestForChangeWithoutImpacts	Retrieves Service Manager requests for change, for the selected CI.

2. Edit the new TQL as needed. You can add CITs as described in "[Naming Constraints for New Request for Change TQLs](#)" on the next page.
3. Access **Admin > Platform > Setup and Maintenance > Infrastructure Settings**:
  - Select **Applications**.
  - Select **Service Health Application**.
  - In the **Service Health Application - Hierarchy (360)** area, enter the name of the new TQL you have create in the corresponding infrastructure setting.

Note that by default these infrastructure settings contain the default TQL names. If you enter a TQL name that does not exist, the default value will be used instead.

After you modify the infrastructure setting, the new TQL will be used, and the Changes and Incidents component will show this information for the CITs you have defined.

## Naming Constraints for New Request for Change TQLs

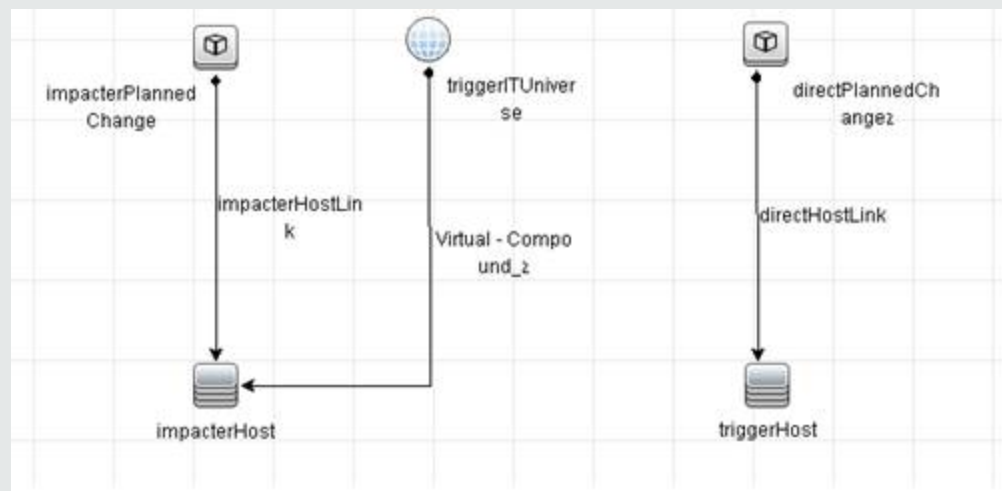
The following naming constraints should be followed in the request for change *without* impact TQL (see the TQL example below, on the right side of the image):

- The request for change CI type should start with **directPlannedChange**.
- The CI type related to the request for change should start with **trigger**.

The following naming constraints should be followed in the request for change *with* impact TQL (see the TQL example below, on the left side of the image):

- **impacterPlannedChange** represents the request for change CI type.
- The CI type related to the request for change should start with **impacter**.
- **triggerITUniverse** represents the "impacted" child CIs.

Examples of request for change TQLs:



## Naming Constraints for New Incident TQLs

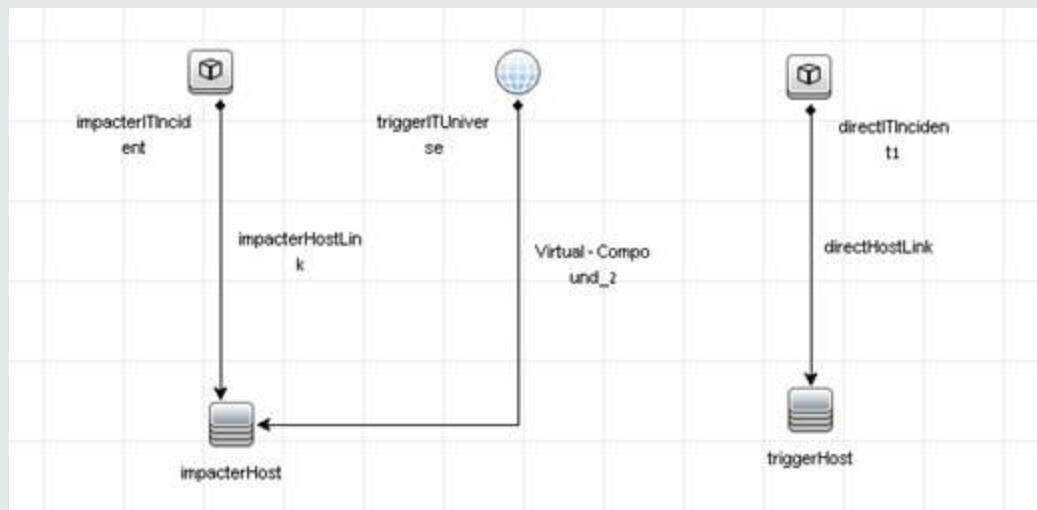
The following naming constraints should be followed in the incidents *without* impact TQL (see the TQL example below, on the right side of the image):

- The incident CI type should start with **directITIncident**.
- The CI type related to the incident should start with **trigger**.

The following naming constraints should be followed in the incidents *with* impact TQL (see the TQL example below, on the left side of the image):

- **impacterITIncident** represents the incident CI type.
- The CI type related to the incident should start with **impacter**.
- **triggerITUniverse** represents the "impacted" child CIs.

Examples of incident TQLs:



# Generate Incidents in Service Manager When a BSM Alert is Triggered

This integration enables you to configure specific CI Status alerts, SLA alerts, or EUM alerts to automatically open a corresponding incident in HP Service Manager. The alerts are mapped to the events using the Event Template.

The triggered alert forwards a corresponding event to OMi, where (using the Incident exchange between Service Manager and Operations Manager I integration) the event is changed into an incident and sent, using the Event Forwarding Service, to HP Service Manager to proactively alert the operator about a problem in the system..

To automatically forward an event when an alert is triggered, follow the steps described in this section. This section includes the following:

- ["CI Status Alerts" below](#)
- ["SLA Alerts" below](#)
- ["EUM Alerts" on the next page](#)

## CI Status Alerts

By default, a CI Status alert is mapped to an event using a default Event Template. You can modify the default Event Template or select a different Event Template as follows:

1. Select **Admin > Service Health > View Management > CI Status Alerts**, select a view and a CI and click **New Alert** or select an existing alert and click **Edit**.
2. In the Actions page, click the **New Event Generation** link in the **Generate Events** section.
3. In the **CI Alert Template Repository** dialog box that opens, select the template you want to use to map the alert to an event and click **Select**. The template you selected is now listed in the Generate Events section. For user interface details, see "CI Status Template Repository Dialog Box" in the Service Health part of the *BSM Application Administration Guide*.

## SLA Alerts

By default, an SLA alert is mapped to an event using a default Event Template. You can modify the default Event Template or select a different Event Template as follows:

1. Select **Admin > Service Level Management > SLA Alerts**, click **New Alert** or select an existing alert and click **Edit**.
2. In the Actions page, click the **New Event Generation** link in the **Generate Events** section.
3. In the **SLA Template Repository** dialog box that opens, select the template you want to use to map the alert to an event and click **Select**. The template you selected is now listed in the Generate Events section. For details, see "SLA Template Repository Dialog Box" in the Service Level Management part of the *BSM Application Administration Guide*.

## EUM Alerts

By default, an EUM alert is mapped to an event using a default Event Template. You can modify the default Event Template or select a different Event Template as follows:

1. Select **Admin > End User Management > Monitoring**, select the view and the CI in the left pane, click the **Alerts** tab, and click the **Press to create new alert** button, or select one of the alerts, and click the **Press to edit alert button**.
2. In the Actions page, select the **Generate Event** option.
3. In the Definition Details area, in the Actions section, click the first link in the **Generate events with <template name> template and <value> values Event Type Indicator**, to select or modify the default template that maps the alert to the event in the **Template Repository** dialog box. For user interface details, see "Notification Templates Dialog Box" in the End User Management part of the *BSM Application Administration Guide*.
4. Click the second link to open the Event Type Indicator dialog box, where you specify the ETI that corresponds to the alert. For user interface details, see "Event Type Indicator Dialog Box" in the End User Management part of the *BSM Application Administration Guide*.

# View Incident Data in BSM, and Manage SLAs Based on Service Manager

This integration enables you to view the Number of Open Incidents in Service Health, and manage SLAs over Serviceability KPIs based on SM incidents, using EMS configuration.

This section includes the following:

- ["Overview: Understanding the Integration with EMS" below](#)
- ["Prerequisites" on page 600](#)
- ["Step 1: Enable Access to HP Service Manager From Within Service Health" on page 601](#)
- ["Step 2: Define HP Service Manager Tables for External Access to the Clocks" on page 601](#)
- ["Step 3: Correct the Clocks WSDL" on page 602](#)
- ["Step 4: Add the Type Field to the logical.name Link Line" on page 603](#)
- ["Step 5: Create a Corresponding HP Service Manager User" on page 604](#)
- ["Step 6: Configure the HP Service Manager Monitor in SiteScope" on page 604](#)
- ["Step 7: Specify the HP Service Manager Web Tier URL in the Infrastructure Settings" on page 606](#)
- ["Step 8: Customize the HP Service Manager EMS Integration Adapter and Check the Assignment – Optional" on page 606](#)
- ["Step 9: Specify the State and Severity of Open Incidents to Be Displayed – Optional" on page 607](#)
- ["Step 10: Include HP Service Manager CIs in Service Level Management Agreements" on page 608](#)
- ["Results" on page 608](#)

## Overview: Understanding the Integration with EMS

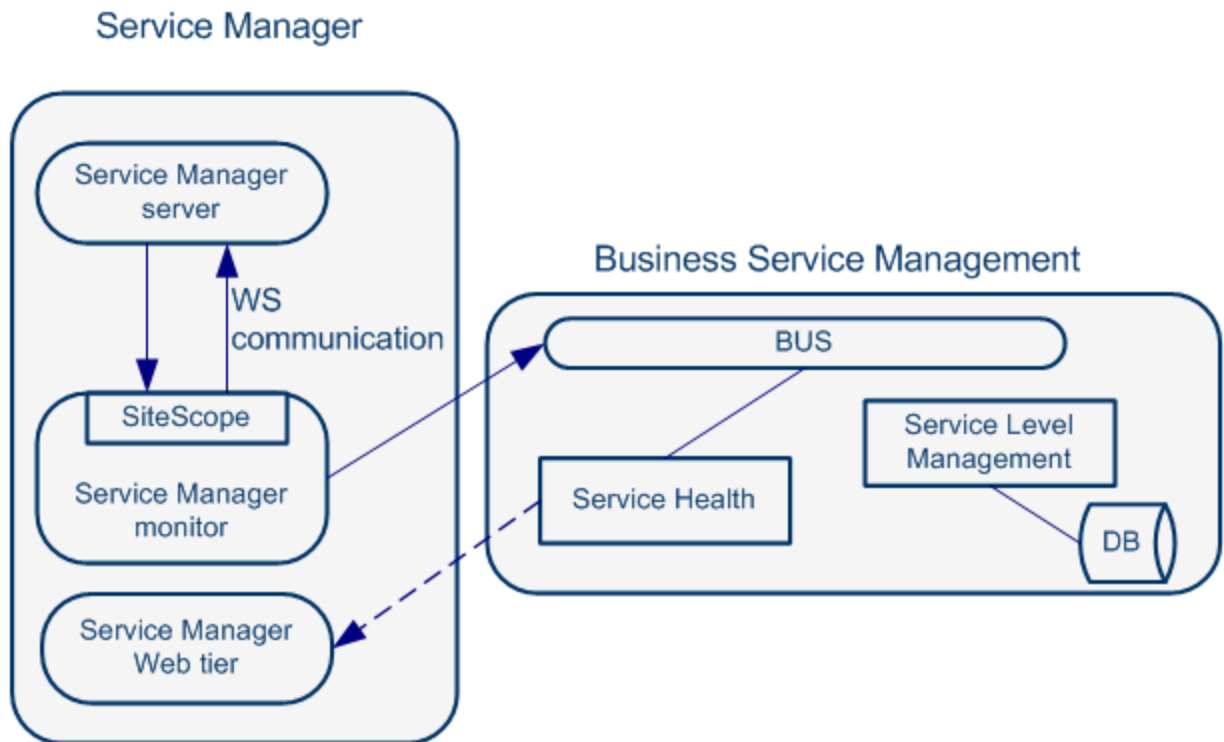
The following sections describe the capabilities provided by the integration of Business Service Management and HP Service Manager with the EMS option.

### **Architecture**

The architecture of the integration of Service Health and Service Level Management with HP Service



Manager is as follows:



You can work with one or more of the following options:

- **Number of Open Incidents KPI.** You can view the Number of Open Incidents KPI (based on data from HP Service Manager) at the business service level in the BSM Service Health views and reports. For details about the views, see "View Topology" in the Service Health part of the *BSM User Guide*. For example: the Operator/Application support can get visibility and alerts based on the Number of Open Incidents in BSM Service Health alongside operational KPIs.
- **Drill down to HP Service Manager from EMS monitor level CIs.** You can drill down from Service Health views at the EMS monitor level business service level to HP Service Manager to view the details of the related incidents. For details about the available drill downs, see "Service Health Menu Options" in the Service Health part of the *BSM User Guide*. For example: the support person can drill down to HP Service Manager to view the details on the open incidents of the selected service. Based on the number of incidents and their details, the support person can prioritize the issues that are the most important.

Name	Status	Acknowledge	Business Number Of open Incidents	Last Status Change
ServiceCenter	-	-	-	-
bsm_biz	✓	-	0	5/28/2014 12:24 PM
bsm_biz monitor	✓	-	0	5/28/2014 12:24 PM
E-mail / Webmail (Asia)	✓	-	0	5/28/2014 11:11 AM
E-mail / Webmail (Asia) monitor	✓	-	0	5/28/2014 11:11 AM
rtsm_biz_service	✗	-	1	5/28/2014 12:52 PM
rtsm_biz_service monitor	✗	-	1	5/28/2014 12:52 PM

Business Impact | Indicators: 1 | Alerts: 0 | Actual Changes : 0 | Planned Changes: 0 | Incidents: 0

The assignment of the Service Manager EMS integration enriches the relevant CIs with the appropriate KPIs, rules, and context menus that are to be assigned automatically to the CIs when the condition occurs, and the assignment is running. For details, see "EMS Integrations Application Overview" in the Integrations Administration part of the *BSM Application Administration Guide*.

### Defining SLAs

You can define SLAs based on the serviceability KPIs (MTTR, MTBF, or MTBSI KPIs) that are calculated based on incidents that come from HP Service Manager. For details, see "Agreements" in the Service Level Management part of the *BSM Application Administration Guide*.

For example: the HP Service Manager manages SLAs with operational KPIs (Availability, Performance, or other KPIs) and serviceability KPIs (MTTR, MTBF, or MTBSI KPIs) using BSM Service Level Management. The HP Service Manager can review the SLAs statuses according to the service Availability, Performance, MTTR, and MTBF side-by-side.

### Elements Created in the View by the Integration with HP Service Manager

The HP Service Manager integration creates the following elements:

Element	Service Health	Service Level Management
---------	----------------	--------------------------

<p><b>CI</b></p>	<p>EMS Monitor CIs for the monitored HP Service Manager system, based on the samples sent by the SiteScope HP Service Manager Monitor.</p> <p>Status for these CIs can be viewed in Service Health in the Business Services, Service Manager, and the Service Measurements views, and the CIs are available to add to SLAs in Service Level Management.</p> <p>Note: All HP Service Manager elements are currently mapped to Business Service CIs through EMS.</p>	
<p><b>Health Indicators</b></p>	<p>Ticketing EMS Monitor HI.</p> <p>For more information, see "Indicator Repository" in the Service Health part of the <i>BSM Application Administration Guide</i>.</p>	<p>MTBF EMS Monitor HI, MTBSI EMS Monitor HI, and MTTR EMS Monitor HI.</p> <p>For more information, see "Indicator Repository" in the Service Level Management part of the <i>BSM Application Administration Guide</i>.</p>
<p><b>KPIs</b></p>	<p>Number of Open Incidents KPI.</p> <p>For details, see "List of Service Health KPIs" in the Service Health part of the <i>BSM Application Administration Guide</i>.</p>	<p>MTTR (Mean Time to Repair, MTBF (Mean Time Between Failures, and MTBSI (Mean Time Between System Incidents KPIs.</p> <p>For details, see "List of Service Level Management KPIs" in the Service Level Management part of the <i>BSM Application Administration Guide</i>.</p>

<p><b>Rules and Tooltips</b></p>	<p>The Number of Open Incidents KPI (attached to an EMS Monitor CI) uses the Number of Open Incidents monitor rule in Service Health and the Number of Open Incidents Sentence tooltip. The rule handles the samples sent to BSM by the EMS system.</p> <p>For details on the rule, see List of Calculation Rules in Service Health" in the Service Health part of the <i>BSM Application Administration Guide</i>.</p>	<p>Each HP Service Manager KPI (attached to an EMS Monitor CI) uses its own monitor rule.</p> <p>For details on the rules, see "List of Service Level Management Business Rule Parameters" in the Service Level Management part of the <i>BSM Application Administration Guide</i>.</p>
<p><b>Context Menu</b></p>	<p>The HP SC Menu.</p> <p>For details on the context menu, see "List of Context Menus" in the Service Health part of the <i>BSM Application Administration Guide</i>.</p>	<p>N/A</p>
<p><b>Context Menu Item</b></p>	<p>The Service Manager context menu item.</p> <p>For details on the context menu, see "List of Context Menu Actions" in the Service Health part of the <i>BSM Application Administration Guide</i>.</p>	<p>N/A</p>

**Note:** Only incidents for which you select a CI in the **Affected CI** field are retrieved by EMS. The CI listed in the **Affected CI** field represents an incident-related item. The default EMS settings only support the monitoring of Business Service CITs.

EMS does not count the incidents that were open through incident exchange (OMi events to SM incidents - part of CLIPv9 solution).

## Prerequisites

The HP Service Manager server, Web tier, and Windows client components must be installed. For details, see HP Service Manager Installation guide.

**Optional.** If you want HP Service Manager to use the SSL-based Trusted Sign-on protocol, configure it according to the instructions in the HP Service Manager online help.

**Optional.** If you want HP Service Manager to use the LW-SSO, configure it according to the instructions in the HP Service Manager online help. BSM must also be configured with LW-SSO.

**Note:** Plan to put both the HP Service Manager Web tier and the webapp in the same container, so you can use the same certificate for both.

## Step 1: Enable Access to HP Service Manager From Within Service Health

Disable the query security of the HP Service Manager application to enable accessing the application, through the right-click HP Service Manager menu option in Service Health. You still have the necessary capabilities to properly secure your system without the query hash.

To enable accessing HP Service Manager from within Service Health:

1. After installing and configuring LW-SSO, edit the web.xml file. The location of the file depends on the type of Web application server the Web tier is deployed on. It is usually located in the HP Service Manager home directory under the Apache home directory. The web.xml file can be located at: **<J2EE webserver path>\webapps<webtier>\WEB-INF.**
2. In the file, locate the **<!-- Specify the Service Manager server host and port location -->** section. This section should appear after the **honorUrlPort** section.
3. Verify that the following strings exist in the section:
 

```
<init-param>
  <param-name>querysecurity</param-name>
  <param-value>>false</param-value>
</init-param>
```
4. Restart the Tomcat container using the Net stop tomcat and Net start tomcat commands.

## Step 2: Define HP Service Manager Tables for External Access to the Clocks

To enable the integration, load the appropriate .unl to provide external access to the clocks table in HP Service Manager. This step enables the display of the Number of Incidents KPI in Service Health. This can be done as follows (note that the probsummary table is accessed by default without .unl):

- In HP Service Manager, manually within HP Service Manager if the tables are used for other external internal integrations. For details, refer to the HP Service Manager documentation.

- Using the configuration file supplied with HP Business Service Management to enable external access to the clocks table:
  - a. Locate the **Clocks\_extaccess\_sm702\_10Nov08.unl** available in the **Setup\SM\_Unloads** directory on the BSM DVD or in the electronic download package, and copy it to a local directory.
  - b. Open the HP Service Manager client and connect to the server.
  - c. Select **Tailoring > Database Manager**.
  - d. In the menu on the upper right side of the Database Manager, select **Import/Load**.
  - e. Select the configuration file you copied to the local directory in the first step.
  - f. Click the **Load FG** button in the left top corner of the page.
  - g. Verify that the clocks table has the values described below. If the values do not match, edit the clocks table in HP Service Manager so that the values are the same as in the below table (for details on how to do that, see HP Service Manager documentation).

Field	Caption	Type
events[start]	start	DateTimeType
events[stop]	stop	DateTimeType
name	name	StringType
key.char	clockId	StringType
sysmodtime	sysmodtime	DateTimeType
type	type	StringType
Key.numeric	clockKey	DecimalType

### Step 3: Correct the Clocks WSDL

Correct the clocks WSDL to enable the display of the Number of Incidents KPI in Service Health.

1. In the HP Service Manager client, select **Tailoring > Web Services > Web Service Configuration**, enter **Clocks** in the **Service Name** field, and click **Search**.
2. Click the **Fields** tab.
3. Add the following entry:

Field	Caption	Type
Total	temp	StringType

**Note:** The values in the table have no meaning.

4. Click **Save** and **OK**.
5. Click **Search** again, click the **Fields** tab and clear the new entry.
6. Click **Save** and **OK**.

## Step 4: Add the Type Field to the logical.name Link Line

This step enables EMS to count incidents that were manually opened in HP Service Manager and to display of the Number of Incidents KPI in Service Health.

**Note:**

- For new customers, EMS calculates ONLY incidents that were manually opened after the tailoring process was applied. For existing customers, the previous HP Service Manager version is populating these fields and the integration works even after you upgrade to HP Service Manager to 7.10. Skip this step if you use other versions. Incidents opened by incident submission are always calculated.
- Perform this step before you configure the SiteScope HP Service Manager Monitor accessed in BSM by clicking **Admin > Integrations > EMS Integration Admin**. Only incidents that were opened after this step are displayed in BSM Service Health.

You add the Type field to the logical.name link line in the probsummary link record as follows:

In HP Service Manager, login with a System Administrator user (for example, **falcon**).

1. Select **Tailoring > Tailoring Tools > Links**.
2. Enter **probsummary** in the **Name** field and click **Search**.
3. Set the cursor on the first line that includes **logical.name** in the **Source Field Name** field (line 14).
4. Select **Select Line** in the **More** menu.

5. Make sure the following entries are present in the table:

Source Field	Target Field
logical.name	logical.name
company	company
type	type
initial.impact	default.impact
severity	problem.priority

6. Click **Save**, **Back**, and then **OK**.

## Step 5: Create a Corresponding HP Service Manager User

This step enables the display of the Number of Incidents KPI in Service Health.

1. Create a dedicated user in HP Service Manager. The user should be used solely for the purposes of the HP Business Service Management/SiteScope integration.
2. Make sure that the HP Service Manager machine and the SiteScope machine share the same time zone.
3. Make sure that the HP Service Manager machine and the SiteScope machine use the same date format (SiteScope date format): **dd/mm/yy**.
4. When configuring the monitor, use the value for the **Username** and **Password** fields that you created in HP Service Manager.

## Step 6: Configure the HP Service Manager Monitor in SiteScope

Configure the HP Service Manager monitor in SiteScope as follows:

1. Synchronize HP Service Manager and SiteScope so their time zones are the same. Match their System Time in the Windows or Unix operating system.
2. Make sure that the user you are using in SiteScope is the user you defined in ["Step 5: Create a Corresponding HP Service Manager User"](#) above.



3. Make sure you have installed the SiteScope EMS license. Note that you do not require this license if SiteScope 11.0 or later is used.
4. Configure the HP Service Manager monitors in SiteScope as follows:
  - a. Stop SiteScope.
  - b. On the SiteScope operating system go to **<SiteScope root directory>\conf\ems\peregrine\lib\<SM version>** and copy **incidentAttributesMapping.conf** to **<SiteScope root directory>\conf\ems\peregrine\**.
  - c. On the SiteScope operating system go to **<SiteScope root directory>\conf\ems\peregrine\lib\<SM version>** and copy **peregrine.jar** to **<SiteScope root directory>\WEB-INF\lib\**.
  - d. Start SiteScope
  - e. Create a new HP Service Manager monitor using the following fields:
    - **Web Service:** <protocol>://<SMhost>:<SMport>/sc62server/PWS/
    - **user name:** <user name defined in "Step 5: Create a Corresponding HP Service Manager User" on the previous page>
    - **user pass:** <password of user created in "Step 5: Create a Corresponding HP Service Manager User" on the previous page>
    - **incident management query:** <type of CI> should be the same as the **Type** field of the CI in Service Manager. For example, for the Business Service CI Type in SM, use **bizservice**.

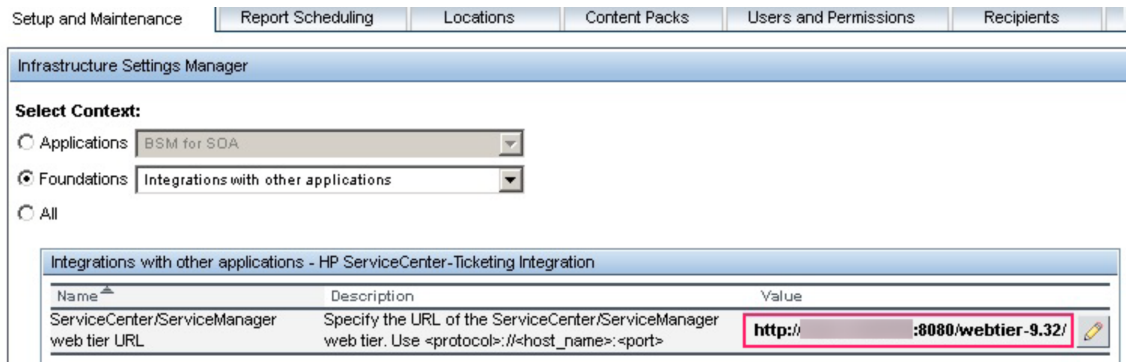
HP Service Manager Monitor Settings	
* HP ServiceManager Web Service Endpoint:	<input type="text"/>
* Username:	<input type="text" value="bsm_admin"/>
Password:	<input type="password" value="*****"/>
	<input type="checkbox"/> Synch Flag
* Date Format:	<input type="text" value="dd/MM/yy HH:mm:ss"/>
Synch Time:	<input type="text"/>
Incident Management (probsummary table) query:	<input type="text" value="(type='bizservice')"/>
* Incident Open State:	<input type="text" value="Open"/>

## Step 7: Specify the HP Service Manager Web Tier URL in the Infrastructure Settings

The HP Service Manager URL is used when drilling down from BSM to HP Service Manager using the **HP SC Menu** context menu item.

1. To specify the HP Service Manager URL, in BSM, select **Admin > Platform > Setup and Maintenance > Infrastructure Settings**, select **Foundations**, and select **Integrations with other applications**.
2. In the Integrations with other applications - HP ServiceCenter - Ticketing Integration table, enter the appropriate URL in the **ServiceCenter/Service Manager web tier URL** entry, using the following format: **<protocol>://<host\_name>:<port>/<web\_app\_name>/** where **host\_name** is the name of the HP Service Manager server, **port** is the port number of the HP Service Manager server, and **web\_app\_name** is the name of the application.

The URL of HP Service Manager is, for example, **http://fando:8080/sm7/**.



## Step 8: Customize the HP Service Manager EMS Integration Adapter and Check the Assignment – Optional

The HP Service Manager integration adapter is predefined. You can customize the configuration. Make sure that the assignment rule is running (it is running by default).

In BSM, select **Admin > Integrations > EMS Integration Admin**, select **ServiceCenter** and click **Edit**. In the Edit Integration dialog box:

1. **Configure the HP Service Manager Monitor – Optional.** The monitor is used to retrieve data from the EMS system using System Availability Management Administration. The HP Service Manager Monitor is added to a SiteScope monitor group created for this monitor and other Integration Monitor types. It is recommended that you configure Integrations Monitors only after a connection between the SiteScope and HP Business Service Management is established. For details, go to "How to Work with the HP Service Manager Integration" in *Monitor Reference* in the SiteScope documentation library.

**Note:** SiteScope cannot be deployed behind a firewall. SiteScope and the monitored system must be on the same LAN or special firewall configuration might be required.

2. **Activate the data assignment rule.** Make sure that the assignment rule is running.

When the EMS monitor sample includes open incidents in its data source, the **Number of Open Incidents** KPI (2600), the **Number of Open Incidents** rule (2600), the **HP SC Menu** context menu (hpsc), the **HP Service Manager** context menu item, and the **Number of Open Incidents** tooltip (2600) are assigned to the EMS Monitor CI.

You can use the EMS Integrations application to customize an HP Service Manager integration. The integration forwards the retrieved data captured from the HP Service Manager system by the SiteScope HP Service Manager monitor to BSM, and creates the appropriate topology that is used to display the data in Service Health. For details on the possible customizations, see "Edit Integration Dialog Box" in the Integrations Administration part of the *BSM Application Administration Guide*.

## Step 9: Specify the State and Severity of Open Incidents to Be Displayed – Optional

To modify the state and severity of the open incidents to be displayed, you can edit the parameters of the Number of Open Incidents rule parameters:

- **For the Number of Open Incidents KPIs attached to a specific EMS Monitor CI.** In BSM, select **Admin > Service Health > Assignments > KPI Assignments**, select the **ServiceCenter** view and the EMS Monitor CI, edit the **Number of Open Incidents** rule, and edit the **Initial State**, **Final State**, and **Severity** parameters.
- **Globally, for all KPIs defined with the Number of Open Incidents rule.** In BSM, select **Admin > Service Health > Repositories > Business Rules**, clone or override the **Number of Open Incidents** rule, and edit the **Initial State**, **Final State**, and **Severity** parameters.

For details on the parameters, see "List of Calculation Rules in Service Health" in the Service Health part of the *BSM Application Administration Guide*.

**Note:** The values available for the Initial State, Final State, and Severity parameters reflect the values defined in HP Service Manager. BSM severity is correlated with HP Service Manager urgency.

## Step 10: Include HP Service Manager CIs in Service Level Management Agreements

You can include HP Service Manager EMS Monitor CIs in your agreements in Service Level Management. Service Level Management contains KPIs and rules specifically configured for HP Service Manager EMS Monitor CIs. The MTTR, MTBF, and MTBSI KPIs and the MTTR, MTBF, and MTBSI rules are dedicated for this integration.

You also configure the incident initial and final state in those rules. For details, see "Service Level Management KPIs for System Incidents" in the Service Level Management part of the *BSM Application Administration Guide*, and locate "Incident State and Severity Values".

## Results

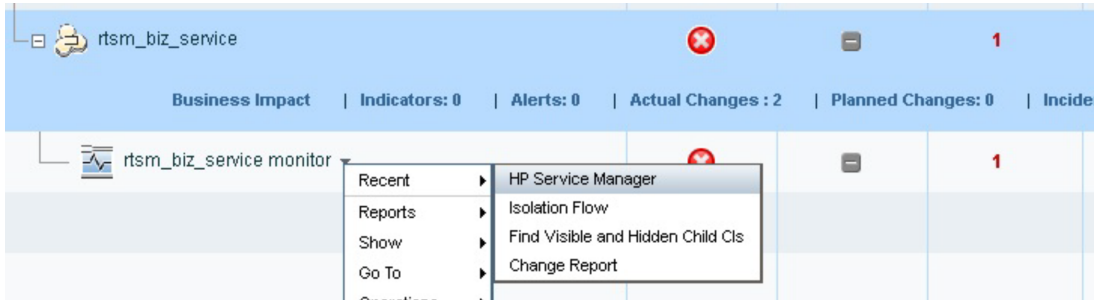
After the task is performed, HP Service Manager data is integrated into BSM. You can:

- **View HP Service Manager Data in Service Health and Service Level Management:**

SiteScope automatically creates the appropriate topology when HP Service Manager data is integrated into BSM. HP Business Service Management adds the data to the Business Services, ServiceCenter, and Service Measurements views, and you can display these views in Service Health. The Business Service and EMS Monitor CIs are added to Service Level Management.

- **Drill down to HP Service Manager from Service Health views:**

In Service Health, in the ServiceCenter, and Service Measurements views, use the **HP Service Manager** option available for **EMS Monitor** CIs under Business Service CIs, to access the relevant incident in the HP Service Manager application. For information about the HP Service Manager application, consult the HP Service Manager documentation.



## Integrate Service Health Reporter to BSM

The following sections are excerpted from the "Integrating with Other Monitoring Solutions" of the *Service Health Reporter Integration Guide*. However, to ensure you have the latest information, you should reference the latest published version of the document in question.

## Next steps

Upon completion of the previous procedures, you will have a complete and integrated system. However, it is important to note that significant work may need to be done in order to develop a production ready system for your organization. Therefore, we recommend the following guidelines to develop an effective solution for your organization.

## Service Manager

### **Delete all out of box data**

We recommend that your first step is to delete all demo data from the system. To do this, locate and then execute the "Purge OOB data" script in the Service Manager script library.

### **Define and develop services according to business needs**

Follow the ITIL service portfolio management process to define the services required by your organization. You should consider:

- Which supporting configuration items are required to run those services?
- Which services are required by your organization's people?
- What are the supporting service catalog items that should be available for users to order that are related to the service?
- What are the supporting service level agreements, operational level agreements, and underpinning contracts that will support the services?
- What are the defining service level targets that enforce the service agreements?

## License information

The HP ITSM Enterprise Suite provides a simple license package that provides an sufficient number of licenses for a complete service management solution. The information regarding the individual product numbers that comprise the ITSM Enterprise Suite is described in the following sections.

### Service Manager

The ITSM Enterprise Suite includes 100 named users for the Foundation, Help Desk Module, Request Management Module, Service Level Management Module and HP IT Change Management Suite. Additionally, each user of the SM Enterprise Suite entitles either 1000 end user self-service (ESS) licenses for the named users of the Catalog Module. Also included are named users for the Knowledge Management Module and either 1000 KM ESS user licenses for the named user. A total of one Service Manager Server is also included in the bundle. One user LTU- license each of the popular Connect-It Connectors for LDAP, E-Mail and Database comes with the suite. Finally, the ITSM Enterprise Suite includes a total of 100 Smart Analytics named users.

### Asset Manager

The ITSM Enterprise Suite SKU includes 10 HP Asset Manager Enterprise Suite Named User for the Asset Manager Portfolio, Asset Manager Contracts, Asset Manager Procurement, Asset Manager Software Asset Management, Asset Manager Financial Management modules.

### IT Business Analytics

The ITSM Enterprise Suite SKU includes 2 Named User, and 10 casual named users.

### HP Operations Bridge Suite Premium Edition

The ITSM Enterprise Suite SKU includes 10 (500 Nodes) Operations Bridge Suite licenses.

## Required licenses

The ITSM Enterprise Suite consists of the following HP products and their product numbers:

HP Product Number	Description
TBD	Service Manager Enterprise Suite Entitlement
TBD	Smart Analytics Suite Entitlement
H7P79AAE	HP Asset Manager Enterprise Suite Entitlement

HP Product Number	Description
T5707AAE	HP IT Business Analytics Suite Entitlement
TD857AAE	HP Service Health Reporter Suite Entitlement
H7T55AAE	HP Business Service Management Suite Entitlement

For reference, the default ITSM Enterprise Suite includes the following license schema:

Licenses	Number of licenses
Service Manager Enterprise Suite with Connect-It Connectors and Knowledge Management Named User	100
Smart Analytics Named User	100
HP Asset Manager Enterprise Suite Named User	10
HP IT Business Analytics Named User	2
HP IT Business Analytics Casual Named User	10
HP Operations Bridge Suite Premium Edition 250 to 950 Nodes 50 Node Pack	10



## ITSM Enterprise Suite file list

File	File Name
<b>Operations Manager i 10.01</b>	
Readme about HP OMi 10.01 installation	HP_OMi_10.01_ReadMeFirst.htm
Customer Letter for HP OMi 10.00	Customer_Letter_OMi_10.00.pdf
HP OMi 10.01 installation for Linux	HP_OMi_10.01_for_Linux.zip
HP OMi 10.01 installation for Windows	HP_OMi_10.01_for_Windows.zip
Data Flow Probe installation	HP_OMi_10.01_DataFlowProbe.zip
HP BSM Connector 10.00 installation for Windows and Linux	HP_BSM_Connector_10.00.zip
HP OMi 10.01 Virtual Appliance	HP_OMi_10.01_VirtualAppliance.ova
HP OMi 10.01 Virtual Appliance signature file	HP_OMi_10.01_VirtualAppliance.ova.sig

File	File Name
HP OMi 10.01 Virtual Appliance technical description	HP_OMi_10.01_VirtualAppliance.pdf
<b>SiteScope 11.30</b>	
Software, HP SiteScope 11.30 for Windows 64bit	T8354-15015.zip
Software, HP SiteScope 11.30 for Linux 64bit	T8354-15016.zip
<b>HP Operations Agent 11.14</b>	
HP Operations Agent v11.14	TC097-15040.iso
<b>Service Health Reporter 9.40</b>	
Reassembly Instructions SHR 9.40-Windows	HPSHR_940_Reassembly_Win.pdf
Reassembly Instructions SHR 9.40-Linux	HPSHR_940_Reassembly_Lin.pdf
HP SHR 9.40 for Windows 1 of 3	HPSHR_940_Win64.part1
HP SHR 9.40 for Windows 2 of 3	HPSHR_940_Win64.part2
HP SHR 9.40 for Windows 3 of 3	HPSHR_940_Win64.part3
HP SHR 9.40 for Linux 1 of 3	HPSHR_940_Lin64.part1

<b>File</b>	<b>File Name</b>
HP SHR 9.40 for Linux 2 of 3	HPSHR_940_Lin64.part2
HP SHR 9.40 for Linux 3 of 3	HPSHR_940_Lin64.part3
<b>Service Manager 9.40</b>	
Software, HP Service Manager 9.40 #1	T5001-15076.iso
Software, HP Service Manager 9.40 #2	T5001-15077.iso
Software, HP Service Manager 9.40 #3	T5001-15078.iso
Software, HP Service Manager 9.40 Multi- Language	T5001-15079.iso
Software, SC Auto Applications v 4.03	T4581-15004.iso
Software, Release Control v 9.2 media	T9770-15008.iso
Software, HP UCMDB/UD 10.20 Windows MLU	TF236-15009.iso

<b>File</b>	<b>File Name</b>
Software, HP UCMDB/UD 10.20 Linux MLU	TF236-15010.iso
Software, HP UCMDB/CM 10.20 Windows MLU	TF236-15011.iso
Software, HP UCMDB/CM 10.20 Linux MLU	TF236-15012.iso
Software, Connect-It 9.60 English	T4500-15030.iso
<b>Asset Manager 9.50</b>	
Software, HP AM 9.50 English	AssetManager-9.50-English.zip)
Software, SAP BusinessObjects Enterprise For AM	AssetManager-CRS-9.50.zip
Software, SAP Crystal Reports Designer For AM	AssetManager-CRD-9.50.zip
Software, Connect-It 9.60 English	T4500-15030.iso
Software, HP UCMDB/UD 10.20 Windows MLU	TF236-15009.iso
Software, HP UCMDB/UD 10.20 Linux MLU	TF236-15010.iso

File	File Name
Software, HP UCMDB/CM 10.20 Windows MLU	TF236-15011.iso
Software, HP UCMDB/CM 10.20 Linux MLU	TF236-15012.iso
<b>IT Business Analytics 9.50</b>	<p><b>Formerly IT Executive ScoreCard</b></p> <div style="background-color: #f0f0f0; padding: 10px; border: 1px solid #ccc;"> <p><b>Note:</b> IT Business Analytics 9.50 was previously known as IT Executive ScoreCard 9.50. The media for IT Business Analytics retains the previous name, IT Executive ScoreCard. Should newly mastered versions of the media be produced, the name change will be implemented then.</p> </div>
Batch file, IT Executive Scorecard 9.50 reassemble	IT_Executive_Scorecard_9.50_reassemble_TB812-15014.zip
Software, HP IT Executive Scorecard 9.50 Part 1 of 3	HP_IT_Executive_Scorecard_9.50_Part_1_of_3_TB812-15013.zip.001
Software, HP IT Executive Scorecard 9.50 Part 2 of 3	HP_IT_Executive_Scorecard_9.50_Part_2_of_3_TB812-15013.zip.002
Software, HP IT Executive Scorecard 9.50 Part 3 of 3	HP_IT_Executive_Scorecard_9.50_Part_3_of_3_TB812-15013.zip.003
Readme, HP IT Executive Scorecard 9.50	TB812-88011.pdf

# Glossary

## M

### **My Term**

My definition

# Index

# Send Documentation Feedback

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

## **Feedback on Release Notes (ITSM Enterprise Suite 2015)**

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to [ovdoc\\_itsm@hp.com](mailto:ovdoc_itsm@hp.com).

We appreciate your feedback!



