

# HP Operations Orchestration Software

Software Version: 9.00.05

## *Microsoft System Center Configuration Manager Integration Guide*

Document Release Date: April 2011

Software Release Date: April 2011



## Legal Notices

### Warranty

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

The information contained herein is subject to change without notice.

### Restricted Rights Legend

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

### Copyright Notices

© Copyright 2011 Hewlett-Packard Development Company, L.P.

### Trademark Notices

For information on open-source and third-party software acknowledgements, see in the documentation set for this release, Open-Source and Third-Party Software Acknowledgements (3rdPartyOpenNotices.pdf).

# On the Web: Finding OO support and documentation

There are two Web sites where you can find support and documentation, including updates to OO Help systems, guides, and tutorials:

- The OO Support site
- HP Live Network

## Support

Documentation enhancements are a continual project at Hewlett-Packard Software. You can obtain or update the HP OO documentation set and tutorials at any time from the HP Software Product Manuals Web site. You will need an HP Passport to log in to the Web site.

### To obtain HP OO documentation and tutorials

1. Go to the HP Software Product Manuals Web site (<http://support.openview.hp.com/selfsolve/manuals>).
2. Log in with your HP Passport user name and password.

OR

If you do not have an HP Passport, click **New users – please register** to create an HP Passport, then return to this page and log in.

If you need help getting an HP Passport, see your HP OO contact.

3. In the **Product** list box, scroll down to and select **Operations Orchestration**.
4. In the **Product Version** list, click the version of the manuals that you're interested in.
5. In the **Operating System** list, click the relevant operating system.
6. Click the **Search** button.
7. In the **Results** list, click the link for the file that you want.

## HP Live Network

For support information, including patches, troubleshooting aids, support contract management, product manuals and more, visit the following site: <https://www.www2.hp.com/>.

This is the **HP Live Network** Web page. To sign in:

1. Click **Login**.
2. On the **HP Passport sign-in** page, enter your HP Passport user ID and password and then click **Sign-in**.
3. If you do not already have an HP Passport account, do the following:
  - a. On the **HP Passport sign-in** page, click **New user registration**.
  - b. On the **HP Passport new user registration** page, *enter the required information and then click **Continue***.
  - c. On the confirmation page that opens, check your information and then click **Register**.
  - d. On the **Terms of Service** page, read the Terms of use and legal restrictions, select the **Agree** button, and then click **Submit**.
4. On the **HP Live Network** page, click **Operations Orchestration Community**.

**The Operations Orchestration Community** page contains links to announcements, discussions, downloads, documentation, help, and support.

**Note:** Contact your OO contact if you have any difficulties with this process.

## In OO: How to find Help, PDFs, and tutorials

The HP Operations Orchestration Software (HP OO) documentation set is made up of:

- **Help for Central**  
Central Help provides information to the following:
  - Finding and running flows
  - For HP OO administrators, configuring the functioning of HP OO
  - Generating and viewing the information available from the outcomes of flow runsThe Central Help system is also available as a PDF document in the HP OO home directory, in \Central\docs.
- **Help for Studio**  
Studio Help instructs flow authors at varying levels of programming ability.  
The Studio Help system is also available as a PDF document in the HP OO home directory, in \Studio\docs directory.
- **Animated tutorials for Central and Studio**  
HP OO tutorials can each be completed in less than half an hour and provide basic instruction on the following:
  - In Central, finding, running, and viewing information from flows
  - In Studio, modifying flowsThe tutorials are available in the Central and Studio subdirectories of the HP OO home directory.
- **Self-documentation for HP OO operations, flows, and Accelerator Packs**  
Self-documentation is available in the descriptions of the operations and steps that are included in the flows.

# Table of Contents

Warranty .....	ii
Restricted Rights Legend .....	ii
Trademark Notices .....	ii
On the Web: Finding OO support and documentation.....	iii
Support .....	iii
HP Live Network .....	iii
In OO: How to find Help, PDFs, and tutorials.....	iv
Overview of Microsoft System Center Configuration Manager integration.....	1
Use cases and scenarios .....	1
Installation and configuration instructions.....	2
Versions .....	2
Architecture .....	3
Microsoft SCCM Integration operation infrastructure .....	3
Common inputs in the integration .....	4
For the site server system .....	4
For the client system .....	5
Operation specifics.....	5
Client.....	5
Get Client Advertisement Status .....	5
Request Machine Policy .....	6
Trigger Hardware Inventory Schedule.....	6
Trigger Software Inventory Schedule .....	6

Collections .....	7
Add Resource to Static Collection .....	7
Create Dynamic Collection .....	7
Create Static Collection .....	8
Delete Collection .....	9
Get Collection Names and Ids .....	10
Get Resources in Collection .....	10
Is Resource in Collection .....	11
Remove Resource from Static Collection .....	11
Request Refresh Collection .....	12
Update Collection Schedule .....	12
Desired Configuration .....	13
Assign Baseline to Collection .....	13
Delete Baseline Assignment .....	13
Get Baseline Ids .....	14
OS Deployment .....	15
Advertise Task Sequence to Collection .....	15
Import New System .....	15
Samples .....	16
Prepare for OS Deployment .....	16
Prepare for Software Distribution .....	17
Site Management .....	17
Check Component Status .....	17
Check Site Status .....	18
Software Distribution .....	18
Assign Distribution Point to Package .....	18
Create Advertisement .....	19
Create Package .....	20
Create Program .....	20
Delete Advertisement .....	21
Delete Distribution Point from Package .....	22
Delete Package .....	22
Delete Program .....	23
Update Advertisement .....	23
Update Package .....	24
Update Program .....	24
Software Update .....	25
Add Updates to Deployment Package .....	25
Create Deployment .....	26
Create Deployment Package .....	26
Delete Deployment .....	27
Delete Deployment Package .....	28
Get Content Id by Software Update CI Id .....	28
Get Software Update CI Ids .....	28

Update Deployment .....	29
Update Deployment Package .....	30
<b>Troubleshooting .....</b>	<b>30</b>
General troubleshooting procedures and tools .....	31
Error messages .....	33
<b>Security .....</b>	<b>35</b>
<b>Tools .....</b>	<b>35</b>

# Overview of Microsoft System Center Configuration Manager integration

With the Microsoft System Center Configuration Manager (SCCM) integration, administrators can build HP Operations Orchestration (OO) flows that are integrated with Microsoft SCCM.

This document explains how this integration has been implemented, and how the integration's OO operations communicate between OO and SCCM.

## Use cases and scenarios

The following are the major use cases for the SCCM integration, and the operations and flows that you can use to implement them.

1. Manage clients
  - Get Client Advertisement Status
  - Request Machine Policy
  - Trigger Hardware Inventory Schedule
  - Trigger Software Inventory Schedule
2. Manage collections
  - Add Resource to Static Collection
  - Create Dynamic Collection
  - Create Static Collection
  - Delete Collection
  - Get Collection Names and Ids
  - Get Resources in Collection
  - Is Resource in Collection
  - Remove Resource from Static Collection
  - Request Refresh Collection
  - Update Collection Schedule
3. Manage desired configurations
  - Assign Baseline to Collection
  - Delete Baseline Assignment
  - Get Baseline Ids
4. Manage OS deployment
  - Advertise Task Sequence to Collection
  - Import New System
5. Manage sites
  - Check Component Status
  - Check Site Status
6. Manage software distribution
  - Assign Distribution Point to Package
  - Create Advertisement
  - Create Package
  - Create Program

- Delete Advertisement
  - Delete Distribution Point from Package
  - Delete Package
  - Delete Program
  - Update Advertisement
  - Update Package
  - Update Program
7. Manage software update
- Add Updates to Deployment Package
  - Create Deployment
  - Create Deployment Package
  - Delete Deployment
  - Delete Deployment Package
  - Get Content Id By Software Update CI Id
  - Get Software Update CI Ids
  - Update Deployment
  - Update Deployment Package

## Installation and configuration instructions

The Microsoft SCCM integration does not require any special integration installation and configuration. The only requirement is that the system that has the RSJRAS service running on it can access the SCCM site, SMS provider, and SCCM client SMS providers.

The default SCCM site server SMS provider name space is:

`\\<siteServer>\root\sms\site_<siteCode>`

The default SCCM client system SMS provider name space is:

`\\<clientSystem>\root\ccm`

In the OO SCCM integration, the namespace is handled so that you do not need to specify it. You can however, use the namespace in wbemtest or CIM studio to check access from the RAS server.

You need a username and password to log on to the SCCM site server or SCCM client system. The username should be in *domain/username* format.

## Versions

Operations Orchestration Version	Microsoft System Center Configuration Manager Version
9.00.05	SCCM 2007, SMS 2003

# Architecture

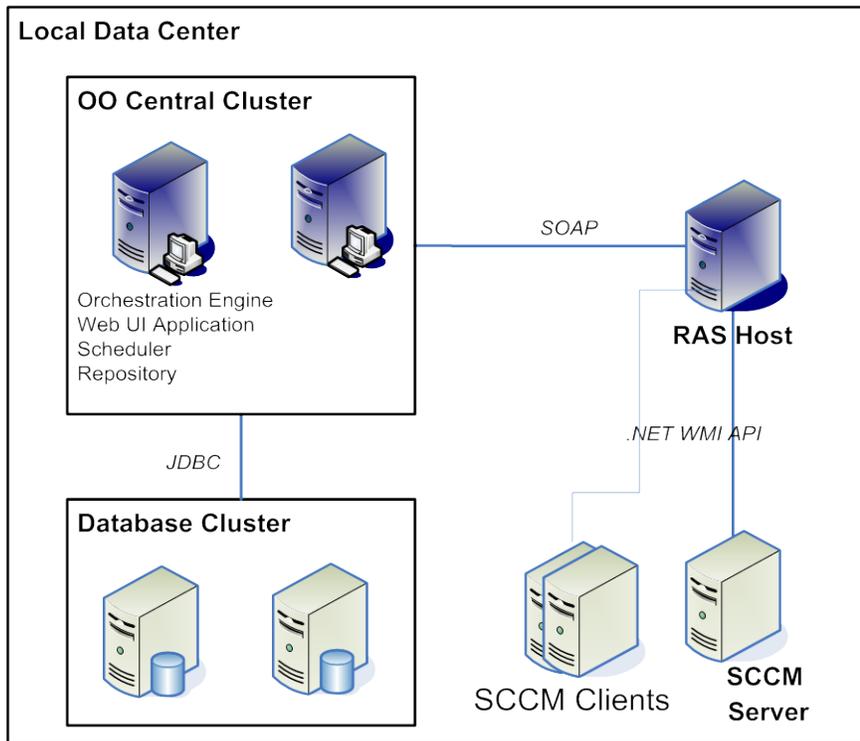


Figure 1 – SCCM OO integration architecture

## Microsoft SCCM Integration operation infrastructure

The Microsoft SCCM integration includes the following operations in the Studio Library, in the Library/Integrations/Microsoft/System Center Configuration Manager/ folder.



Figure 2 – Microsoft SCCM integration operation infrastructure

## Common inputs in the integration

OO flows and operations use inputs to specify how they obtain the data that they need and when the data is obtained. The following inputs are used consistently throughout the Microsoft SCCM integration's site server system and client system operations.

### For the site server system

#### host

The name of the SCCM site server to which you want to connect. The default value is **localhost**.

#### username

The username with which to connect to the SCM site server. The username format is *domain\username* (for example, **roo\oadmin**). The user should have rights to access WMI on the site server. If you specify **localhost** as the **host** input value, you do not have to specify a username.

### password

The password associated with the **username** input value. If you specify **localhost** as the **host** input value, you do not have to specify a password.

### siteCode

The three character alphanumeric code of the SCCM site (for example, **ROS**).

## For the client system

### host

The name of the SCCM client host. The default value is **localhost**.

### username

The username with which to connect to the SCM client host. The username format is *domain\username* (for example, **roo\oadmin**). The user should have rights to access WMI on the site server. If you specify **localhost** as the **host** input value, you do not have to specify a username.

### password

The password associated with the **username** input value. If you specify **localhost** as the **host** input value, you do not have to specify a password.

## Operation specifics

This section describes the Microsoft SCCM integration's operations, including any operation-specific inputs. The operations are grouped by their basic functionality:

- Client
- Collections
- Desired Configuration
- OS Deployment
- Samples
- Site Management
- Software Distribution
- Software Update

## Client

### Get Client Advertisement Status

The **Get Client Advertisement Status** operation gets the advertisement status of a client system in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### advertisementId

The advertisement ID of a task sequence advertisement or software distribution advertisement. You can find the advertisement ID in **Computer Management/Software Distribution/Advertisements** in SCCM.

**resourceId**

The resource ID of the client system. You can find the resource ID in the resource's **Properties** in SCCM.

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

**retLastState**

The last state of the advertisement when it runs on the system.

**retLastStatusTime**

The time of the last state of the advertisement when it ran on the system. The return time can be empty depending on the value of **retLastState**. If it is not empty, it is in the format *yyyyMMddHHmmss.mmmmmmsUUU* (for example, **20100601113000.000000+\*\*\***).

**retLastAcceptanceState**

The state that indicates whether the advertisement ran on the system. The valid values are **1** (accept) and **0** for (no status).

**retLastAcceptanceStatusTime**

The time of the last acceptance state. The return time can be empty depending on the value of **retLastAcceptanceState**. If it is not empty, it is in the format *yyyyMMddHHmmss.mmmmmmsUUU* (for example, **20100601113000.000000+\*\*\***).

## Request Machine Policy

The **Request Machine Policy** operation requests a client system of SCCM to execute the **RequestMachinePolicy** method of the **SMS\_Client**.

All of the operation's inputs are described in [Common inputs in the integration](#).

The operation returns the following:

**returnCode**

Indicates whether the request is successfully sent to the client system.

## Trigger Hardware Inventory Schedule

The **Trigger Hardware Inventory Schedule** operation requests a client system of the SCCM to execute the **TriggerSchedule** method of the **SMS\_Client** on **Hardware Inventory**. The action starts the hardware inventory scan on the client system.

All of the operation's inputs are described in [Common inputs in the integration](#).

The operation returns the following:

**returnCode**

Indicates whether the request is successfully sent to the client system.

## Trigger Software Inventory Schedule

The **Trigger Software Inventory Schedule** operation requests a client system of SCCM to execute the **TriggerSchedule** method of **SMS\_Client** on **Software Inventory**. The action starts the software inventory scan on the client system.

All of the operation's inputs are described in [Common inputs in the integration](#).

The operation returns the following:

**returnCode**

Indicates whether the request is successfully sent to the client system.

## Collections

### Add Resource to Static Collection

The **Add Resource to Static Collection** operation adds a resource to a static collection in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**collectionName**

The name of the collection to which to add the resource. The collection has to be a static collection (for example, **My Test Systems**). The **collectionName** and **collectionName** inputs cannot both be empty at the same time.

**collectionId**

The collection ID of the static collection to which to add the resource in SCCM (for example, **ROS00030**). You can find the value of the **collectionId** input in the collection folder's properties in SCCM. If you specify a value for the **collectionId** input, the value of the **collectionName** input is ignored. The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

**resourceName**

The name of the resource to add to the collection. The valid values are a system name, a user name, a user group name, and an unknown system name. An example is **ROS-SANDRA**. The resource must have already been discovered by SCCM.

**resourceType**

The resource type for the resource to add. The valid values are **System**, **User**, **User Group**, and **Unknown System**. If you do not specify a value or specify an incorrect value, the default is **System**. You can find the resource type in **SCCM - Resource Type** under **Configuration/Selection Lists**.

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

**retResourceId**

The ID of the resource in SCCM that has been added to the collection.

**retCollectionId**

The collection ID in SCCM to which the resource was added.

### Create Dynamic Collection

The **Create Dynamic Collection** operation creates a dynamic collection in SCCM. If you specify valid values for the **parentCollectionId** or **parentCollectionName** inputs, the collection is created under the parent collection. Otherwise, the collection is created under **Collections** in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**collectionName**

The name of the dynamic collection to create. The collection name should be unique in SCCM.

**ownedByThisSite**

Specifies whether the collection owned by this site is specified in the **siteCode** input. The valid values are **True** and **False**. The default value is **True**.

**comment**

A comment for the collection created by this operation.

**parentCollectionName**

The name of the parent collection under which the new collection is created. If you do not specify values for the **parentCollectionName** and **parentCollectionId** inputs, the operation uses the root collection **Collections** in SCCM.

**parentCollectionId**

The ID of the parent collection under which the new collection is created (for example, **ROS00030**). You can find the value of the collection ID in the collection folder's properties in SCCM. If you specify a value for the **parentCollectionId** input, the value of the **parentCollectionName** input is ignored. If you do not specify values for both the **parentCollectionName** and **parentCollectionId** inputs, the operation uses the root collection **Collections** in SCCM.

**ruleName**

The name for the query for this dynamic collection. If you do not specify a value for this input, the default name **newQuery** is used.

**query**

The query for this dynamic collection. It must be a valid WMI query. If you do not specify a value for this input, the operation uses the default query **SELECT \* FROM SMS\_R\_System**.

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

**retCollectionId**

The ID of the collection created in SCCM.

## Create Static Collection

The **Create Static Collection** operation creates a static collection in SCCM. If you not specify valid values for the **parentCollectionId** or **parentCollectionName** inputs, the collection is created under the parent collection. Otherwise, the collection is created under **Collections** in SCCM. If you specify valid resource names in the **resourceNames** input, the resources are added to the collection.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**collectionName**

The name of the static collection to create. The collection name should be unique in SCCM.

**ownedByThisSite**

Specifies whether the collection owned by this site is specified in the **siteCode** input. The valid values are **True** and **False**. The default is **True**.

**comment**

A comment for the collection created by this operation.

### resourceNames

The list of resource names to add to the collection, delimited by the value you specify in the **delimiter** input. The list can contain system names, user names, user group names, or unknown system names (for example, **ROS-SANDRA,ROS-TINA**). The resources in the list must have already been discovered by SCCM.

### resourceType

The type of the resources listed in the **resourceNames** input value. The valid resource types are **System**, **User**, **User Group**, and **Unknown System**. If you do not specify a value for this input or you specify the wrong type, the input value defaults to **System**. Check **SCCM - Resource Type** under **Configuration/Selection Lists**.

### parentCollectionName

The parent collection name under which the new collection is created. If you do not specify values for the **parentCollectionName** and **parentCollectionId** inputs, the operation uses the root collection **Collections** in SCCM.

### parentCollectionId

The parent collection ID under which the new collection is created (for example, **ROS00030**). You can find the value of the collection ID in collection folder's properties in SCCM. If you specify a value for **parentCollectionId**, the value of the **parentCollectionName** input is ignored. If you do not specify values for the **parentCollectionName** and **parentCollectionId** inputs, the operation uses the root collection **Collections** in SCCM.

### delimiter

The delimiter for the **resourceNames** input. The default is a comma (,).

The operation returns the following:

### returnResult

A message that indicates the status of the operation.

### retCollectionId

The ID of the collection created in SCCM.

## Delete Collection

The **Delete Collection** operation removes a collection in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### collectionName

The name of the collection to delete in SCCM (for example, **My Test Systems**). The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

### collectionId

The ID for the collection to delete in SCCM (for example, **ROS00030**). You can find the value of **collectionId** in the collection folder's properties in SCCM. If you specify a value for the **collectionId** input, the value of the **collectionName** input is ignored. The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

The operation returns the following:

### returnResult

A message that indicates the status of the operation.

### retCollectionId

The deleted collection ID in SCCM.

## Get Collection Names and Ids

The **Get Collection Names and Ids** operation gets collections names and IDs in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### collectionName

The name of the collection to search for in SCCM (for example, **My Test Systems**). If you do not specify a value for this input, the operation returns all of the collection names and IDs in SCCM.

### exactMatch

Specifies whether you want to search on the exact value of the **collectionName** input. The valid values are **True** and **False**. If you specify a value of **False**, it indicates that you want to search on a partial value of the **collectionName** input.

The operation returns the following:

### returnResult

The pipe (|) delimited list of the collection names retrieved by the operation (for example, **My Test Systems|My Dev System**).

### retCollectionNamesIds

The pipe (|) delimited list of the pairs of names and IDs of the collections retrieved by the operation in the format *name=id* (for example, **My Test Systems=ROS0003|My Dev System=ROS0004**).

### retCollectionNames

The pipe (|) delimited list of collection names retrieved by the operation (for example, **My Test Systems|My Dev System**).

## Get Resources in Collection

The **Get Resources in Collection** operation retrieves all of the resource names in a collection in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### collectionName

The name of the collection from which to get the resources in SCCM (for example, **My Test Systems**). The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

### collectionId

The ID for the collection from which to get the resources in SCCM (for example, **ROS00030**). You can find the value of **collectionId** in the collection folder's properties in SCCM. If you specify a value for the **collectionId** input, the value of the **collectionName** input is ignored. The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

The operation returns the following:

### returnResult

The pipe (|) delimited list of resource names retrieved by the operation (for example, **ROS-TINA|ROS-SANDRA**).

### retCollectionId

The collection ID in SCCM to which the resources belong.

## retResources

The pipe (|) delimited list of resource names retrieved by the operation (for example, **ROS-TINA|ROS-SANDRA**).

## Is Resource in Collection

The **Is Resource in Collection** operation checks whether a resource is in a collection in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### collectionName

The name of the collection in which to check for the resource in SCCM (for example, **My Test Systems**). The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

### collectionId

The ID for the collection with which to check the resource in SCCM (for example, **ROS00030**). You can find the value of **collectionId** in the collection folder's properties in SCCM. If you specify a value for the **collectionId** input, the value of the **collectionName** input is ignored. The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

### resourceName

The resource name to check in the collection. The value of this input can be a system name, a user name, a user group name, or an unknown system name (for example, **ROS-SANDRA**).

### resourceType

The type of the resource to check. The valid values are **System**, **User**, **User Group**, and **Unknown System**. If you do not specify a value for this input or specify an incorrect type, it defaults to **System**. Check **SCCM - Resource Type** under **Configuration/Selection Lists** for more information.

The operation returns the following:

### returnResult

A message that indicates the status of the operation.

## Remove Resource from Static Collection

The **Remove Resource from Static Collection** operation removes a resource from a static collection in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### collectionName

The name of the collection from which to remove the resource. The collection must be a static collection (for example, **My Test Systems**). The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

### collectionId

The collection ID for the static collection from which to remove the resource in SCCM (for example, **ROS00030**). You can find the **collectionId** value in collection folder's properties in SCCM. If you do not specify a value for the **collectionId** input, the value of the **collectionName** input is ignored. The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

### resourceName

The name of the resource to remove from the collection. The value of this input can be a system name, a user name, a user group name, or an unknown system name (for example,

**ROS-SANDRA**). The value of the **resourceName** input should have been already discovered by SCCM.

#### **resourceType**

The type for the resource to remove. The valid values are **System**, **User**, **User Group**, and **Unknown System**. If you do not specify a value for this input or specify the wrong type, the default is **System**. Check **SCCM - Resource Type** under **Configuration/Selection Lists**.

The operation returns the following:

#### **returnResult**

A message that indicates the status of the operation.

#### **retResourceId**

The resource ID in SCCM that has been removed from the collection.

### **Request Refresh Collection**

The **Request Refresh Collection** operation sends a request to refresh a collection in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

#### **collectionName**

The name of the collection to refresh in SCCM (for example, **My Test Systems**). The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

#### **collectionId**

The ID for the collection in SCCM (for example, **ROS00030**). You can find the **collectionId** value in the collection folder's properties in SCCM. If you specify a value for the **collectionId** input, the **collectionName** value is ignored. The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

The operation returns the following:

#### **returnResult**

A message that indicates the status of the operation.

### **Update Collection Schedule**

The **Update Collection Schedule** operation updates the refresh schedule of a collection in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

#### **collectionName**

The name of the collection to update in SCCM (for example, **My Test Systems**). The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

#### **collectionId**

The ID for the collection in SCCM (for example, **ROS00030**). You can find the **collectionId** value in the collection folder's properties in SCCM. If you specify a value for the **collectionId** input, the **collectionName** value is ignored. The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

#### **schedule**

The refresh schedule for the collection (for example, **Every 30 Minutes** or **Every 1 Day**). Check **SCCM - Schedule** under **Configuration/Selection Lists** for more information. If you do not specify a value for this input, it defaults to **Every 1 Day**.

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

## Desired Configuration

### Assign Baseline to Collection

The **Assign Baseline to Collection** operation assigns a configuration baseline to a collection by creating a baseline assignment in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**baselineId**

The unique CI ID of a configuration item (**SMS\_ConfigurationItem**) with baseline type (**CIType\_ID = 2**). Even though SCCM allows duplicate baseline names when you create a baseline, the best practice is to create a configuration baseline with a unique name in SCCM. This allows you to use the **Get Baseline Ids** operation to find one baseline CI ID to match to the baseline name.

**collectionName**

The name of the collection to which to assign the baseline in SCCM (for example, **My Test Systems**). The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

**collectionId**

The ID of the collection to which to assign the baseline in SCCM (for example, **ROS00030**). You can find the value of **collectionId** in the collection folder's properties in SCCM. If you specify a value for the **collectionId** input, the **collectionName** input value is ignored. The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

**evalSchedule**

The evaluation schedule to run the baseline assignment (for example, **Every 30 Minutes** or **Every 1 Day**). Check **SCCM - Schedule** under **Configuration/Selection Lists** for more information. If you do not specify a value for this input, it defaults to **Every 1 Day**.

**description**

The description for the baseline assignment.

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

**retAssignmentId**

The assignment ID after the baseline is assigned to the collection. You can find the baseline assignment by checking the **Assignments** tab in **Properties** of the configuration baseline. You can also find this ID in **WMI SMS\_BaselineAssignment**.

### Delete Baseline Assignment

The **Delete Baseline Assignment** operation deletes a baseline assignment based on assignment ID or baseline ID or name and collection ID or name in SCCM.

If you specify the assignment ID, the operation uses it. If you do not specify the assignment ID, the operation may retrieve multiple baseline assignments and the operation deletes the multiple baseline assignments.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

#### **assignmentId**

The assignment ID for a configuration baseline after the baseline is assigned to a collection. You can find this ID in **WMI SMS\_BaselineAssignment**. If you do not specify a value for this input, the operation uses the **baselineId** or **baselineName** and **collectionId** or **collectionName** to retrieve the assignment ID(s).

#### **baselineName**

The name of the configuration baseline. This name can be a duplicate name in SCCM. The best practice is to create a configuration baseline with a unique name in SCCM. If you do not specify a value for the **assignmentId** input, the **baselineId** and **baselineName** inputs cannot both be empty at the same time.

#### **baselineId**

The unique CI ID of a configuration item (**SMS\_ConfigurationItem**) with baseline type (**CIType\_ID = 2**). Even though SCCM allows duplicate baseline names, the best practice is to create a configuration baseline with a unique name in SCCM. Then you can use the **Get Baseline Ids** operation to find one baseline CI ID to match to the baseline name. If you do not specify a value for the **assignmentId** input, the **baselineId** and **baselineName** inputs cannot both be empty at the same time.

#### **collectionName**

The name of the collection to assign the baseline to in SCCM (for example, **My Test Systems**). If you do not specify a value for the **assignmentId** input, the **collectionId** and **collectionName** inputs cannot both be empty at the same time.

#### **collectionId**

The ID for the collection to which to assign the baseline in SCCM (for example, **ROS00030**). You can find the value for the **collectionId** in the collection folder's properties in SCCM. If you do not specify a value for the **collectionId** input, the **collectionName** is ignored. If you do not specify a value for the **assignmentId** input, the **collectionId** and **collectionName** inputs cannot both be empty at the same time.

The operation returns the following:

#### **returnResult**

A message that indicates the status of the operation.

#### **retAssignmentId**

The assignment ID of the deleted baseline assignment if you specified a value for the **assignmentId** input. Otherwise, this is a comma-delimited list of assignment IDs if multiple assignments are found based on the baseline and collection information.

## **Get Baseline Ids**

The **Get Baseline Ids** operation gets baseline CI IDs based on a baseline name in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

#### **baselineName**

The name of a configuration baseline. This name can be a duplicate name in SCCM although the best practice is to create a configuration baseline with a unique name in SCCM.

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

**retBaselineIds**

A comma-delimited list of baseline CI IDs.

## OS Deployment

### Advertise Task Sequence to Collection

The **Advertise Task Sequence to Collection** operation advertises a task sequence to a collection in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**advertisementName**

A unique name for the task sequence advertisement (for example, **Install XP OS Task Sequence Advertisement**). After the operation creates the advertisement, you can find it under **Computer Management/Software Distribution/Advertisements**. The task sequence advertisement should have **[Task Sequence]** in the **Program** column.

**taskSequenceName**

A unique task sequence name. You can find the task sequence name under **Computer Management/Operating System Deployment/Task Sequences** in SCCM.

**collectionName**

The name of the collection to which to advertise the task sequence in SCCM (for example, **My Test Systems**). The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

**collectionId**

The ID for the collection to which to advertise the task sequence in SCCM (for example, **ROS00030**). You can find the value for **collectionId** in the collection folder's properties in SCCM. If you specify a value for the **collectionId** input, the value of the **collectionName** input is ignored. The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

**comment**

A comment for the task sequence advertisement.

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

**retAdvertisementId**

The advertisement ID provided after the task sequence is advertised to the collection. You can find the advertisement by checking the **Advertisements** tab in **Properties** of the collection. You can also find this ID in **Computer Management/Software Distribution/Advertisements** in SCCM.

### Import New System

The **Import New System** operation imports a system to **All Systems** in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**systemName**

A unique name for the system to import.

**smBiosGuid**

The SMBIOS GUID of the system to import (for example, "421440F7-2DF1-6453-0C4C-5A4B04442B17").

**macAddress**

The MAC address of the system to import (for example, "00:50:56:94:00:31").

**overwriteExistingRecord**

Specifies whether to overwrite the existing system. The valid values are **True** and **False**. The default value is **False**. If you specify a value of **True** for the **overwriteExistingRecord** input, and the value of the **systemName** input is different from the existing system name in SCCM but the values of the **smBoiGuid** and **macAddress** inputs are the same as they are in the existing system, the existing system is overwritten. Otherwise, the system stays the way it is.

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

**retResourceId**

The resource ID in SCCM for the imported system.

**retSystemExists**

Indicates if the imported system already exists in SCCM.

## Samples

### Prepare for OS Deployment

The **Prepare for OS Deployment** sample flow shows you how to prepare for OS deployment in SCCM. It creates a static collection under **Collections** based on the value of the **collectionName** input, and updates the collection refresh schedule to be every 1 day. It also creates a task sequence advertisement called **Sample Advertisement for Task Sequence** and assigns it to the collection created. Then, it imports a system into SCCM. After the flow imports the system, it adds the system into the created collection and makes sure that it is there.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**collectionName**

The name of the collection to create in SCCM (for example, **My Test Systems**).

**taskSequenceName**

The unique task sequence name. You can find the task sequence name under **Computer Management/Operating System Deployment/Task Sequences** in SCCM.

**systemName**

The unique name for the system to import.

**smBiosGuid**

The SMBIOS GUID of the system to import (for example, "421440F7-2DF1-6453-0C4C-5A4B04442B17").

## macAddress

The MAC address of the system to import (for example, "00:50:56:94:00:31").

The flow has three possible responses:

- **success** - The sample flow ran successfully.
- **failure** - The sample flow failed to run.
- **timeout** - The flow timed out.

## Prepare for Software Distribution

The **Prepare for Software Distribution** sample flow shows you how to prepare for software distribution in SCCM. It creates a dynamic collection based on the values of the **collectionName** input and the **query** input. It updates the collection refresh schedule to be every 12 hours. It also creates a software distribution package with the specified package source. Once the package is available, it creates a program in the package with a specific command line. It assumes the program duration is about 10 minutes. Then it assigns a server distribution point to the package. Finally, it creates an advertisement of the package and assigns it to the collection.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### collectionName

The name of the collection to create in SCCM (for example, **My Test Systems**).

### query

The query for this dynamic collection. Make sure it is a valid WMI query (for example, **SELECT \* FROM SMS\_R\_System WHERE Name LIKE '%BROOKE%'**).

### packageName

The unique name of the software distribution package.

### packageSourcePath

The source path of the software distribution package. This must be a network share path (for example, **\\ros-jolie\packages**).

### programName

The unique name for the program to create in SCCM.

### commandLine

The command line for running the program (for example, **SSHSecureShellClient.exe**).

### distributionPoint

The server distribution point to assign to the package (for example, **ros-jolie**).

### advertisementName

A unique name for the advertisement to create (for example, **Test Advertisement**).

## Site Management

### Check Component Status

The **Check Component Status** operation checks the status of a component in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**componentName**

The name of the component whose status you want to check (for example, **SMS\_COMPONENT\_MONITOR**). You can find the component name under **System Status/Site Status/<site name>/Component Status** in SCCM.

**tallyInterval**

The interval for the status of the component (for example, "**Since 12:00 AM**" or "**Since Monday**"). Check **SCCM - Tally Interval** under **Configuration/Selection Lists** for more information. If you do not specify a value for this input, it defaults to "**Since 12:00 AM**".

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

**retComponentStatus**

The status of the component. The valid values are **0** (healthy), **1** (warning conditions), and **2** (error conditions).

## Check Site Status

The **Check Site Status** operation checks the status of the specified site in SCCM.

All of the operation's inputs are described in [Common inputs in the integration](#).

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

**retSiteStatus**

The status of the site. The valid values are **0** (healthy), **1** (warning conditions), and **2** (error conditions).

## Software Distribution

### Assign Distribution Point to Package

The **Assign Distribution Point to Package** operation assigns a distribution point to a software distribution package in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**packageName**

The unique name of the software distribution package to assign. You can find the package name under **Computer Management/Software Distribution/Packages** in SCCM. The **packageName** and **packageId** inputs cannot both be empty at the same time.

**packageId**

The ID of the software distribution package to assign. You can find the package ID under **Computer Management/Software Distribution/Packages** in SCCM. If you specify a value for the **packageId** input, the **packageName** input value is ignored. The **packageName** and **packageId** inputs cannot both be empty at the same time.

**distributionPoint**

The distribution point to assign to the package (for example, **ROS-JOLIE** or **ROS-JOLIE\SMSPXEIMAGES\$**). If you do not specify a value for this input, all distribution points specified by the value of the **dpResourceType** input are added.

**dpResourceType**

The type of distribution point to assign. The valid values are **Windows NT Server** and **Windows NT Share**. The default value is **Windows NT Server**.

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

## Create Advertisement

The **Create Advertisement** operation creates an advertisement on a software distribution package in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**advertisementName**

A unique name for the advertisement to create (for example, **SSH Share Package Advertisement**).

**packageName**

The unique name of the software distribution package. You can find the package name under **Computer Management/Software Distribution/Packages**. The **packageName** and **packageId** inputs cannot both be empty at the same time.

**packageId**

The ID of the software distribution package. You can find the package ID under **Computer Management/Software Distribution/Packages** in SCCM. If you specify a value for the **packageId** input, the value of the **packageName** input is ignored. The **packageName** and **packageId** inputs cannot both be empty at the same time.

**collectionName**

The name of the collection to advertise the software package to in SCCM (for example, **My Test Systems**). The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

**collectionId**

The ID for the collection to advertise the software package to in SCCM (for example, **ROS0003**). You can find the collection ID in the collection folder's properties in SCCM. If you specify a value for the **collectionId** input, the **collectionName** is ignored. The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

**programName**

The name of the program in the software package. You can find the program name under **Computer Management/Software Distribution/Packages/<Package Name>/Programs** in SCCM.

**comment**

A comment for the advertisement.

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

**retAdvertisementId**

The ID for the created advertisement for the package. You can find the advertisement by checking the **Advertisements** tab in **Properties** of the collection. You can also find this ID in **Computer Management/Software Distribution/Advertisements** in SCCM.

## Create Package

The **Create Package** operation creates a software distribution package in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**packageName**

A unique name for the software distribution package. After the operation creates the package, you can find the package name under **Computer Management/Software Distribution/Packages**.

**packageSourceFlag**

Specifies whether the package has a source path. The valid values are **2** (it has source path) and **1** (it has no source path). The default is **1**.

**packageSourcePath**

The source path of the software distribution package to create. You must specify a value for this input if you specify a value of **2** for the **packageSourceFlag** input. The value must be a network share path (for example, `\\ros-jolie\packages`).

**packageManufacturer**

The manufacturer of the software distribution package (for example, **HP**).

**packageVersion**

The version of the software distribution package (for example, **1.0**).

**packageLanguage**

The language of the software distribution package (for example, **En**).

**description**

A description of the software distribution package.

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

**retPackageId**

The ID for the created software distribution package. You can find the package ID under **Computer Management/Software Distribution/Packages** in SCCM.

## Create Program

The **Create Program** operation creates a program in a software distribution package in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**programName**

The unique program name for the program to create in SCCM.

**packageName**

The unique name of the software distribution package. You can find the package name under **Computer Management/Software Distribution/Packages** in SCCM. The **packageName** and **packageId** inputs cannot both be empty at the same time.

**packageId**

The ID of the software distribution package. You can find the package ID under **Computer Management/Software Distribution/Packages** in SCCM. If you specify a value for the **packageId** input, the value of the **packageName** input is ignored. The **packageName** and **packageId** inputs cannot both be empty at the same time.

**commandLine**

The command line for running the program (for example, **SSHSecureShellClient.exe**).

**comment**

A comment for the program.

**duration**

The number of minutes for the program to run.

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

**retPackageId**

The package ID where the created program resides. You can find the package ID under **Computer Management/Software Distribution/Packages** in SCCM.

**retProgramName**

The name of the created program.

## Delete Advertisement

The **Delete Advertisement** operation deletes an advertisement in SCCM. The advertisement can be a software distribution advertisement or a task sequence advertisement.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**advertisementId**

The ID of the advertisement to delete. You can find the advertisement ID in **Computer Management/Software Distribution/Advertisements** in SCCM. If you specify a value for the **advertisementId** input, the value of the **advertisementName** input is ignored. The **advertisementId** and **advertisementName** inputs cannot both be empty at the same time.

**advertisementName**

The name of the advertisement to delete. You can find the advertisement name in **Computer Management/Software Distribution/Advertisements** in SCCM. The **advertisementId** and **advertisementName** inputs cannot both be empty at the same time. Since SCCM allows duplicate advertisement names, if use the **advertisementName** input, multiple advertisements may be deleted.

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

## **retAdvertisementId**

If you specify a value for the **advertisementId** input, this is the ID of the deleted advertisement. Otherwise, this is a comma-delimited list of advertisement IDs.

## **Delete Distribution Point from Package**

The **Delete Distribution Point from Package** operation deletes distribution points from a software distribution package in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### **packageName**

The unique name of the software distribution package. You can find the package name under **Computer Management/Software Distribution/Packages** in SCCM. The **packageName** and **packageId** inputs cannot both be empty at the same time.

### **packageId**

The ID of the software distribution package. You can find the package ID under **Computer Management/Software Distribution/Packages** in SCCM. If you specify a value for the **packageId** input, the value of the **packageName** input is ignored. The **packageName** and **packageId** inputs cannot both be empty at the same time.

### **distributionPoint**

The distribution point to delete from the package (for example, **ROS-JOLIE** or **ROS-JOLIE\SMSPXEIMAGES\$**). If you specify a value for the **distributionPoint** input, that distribution point is deleted from the package. Otherwise, all of the distribution points are deleted.

The operation returns the following:

### **returnResult**

A message that indicates the status of the operation.

## **Delete Package**

The **Delete Package** operation deletes a software distribution package in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### **packageName**

The name of the software distribution package to delete. You can find the package name under **Computer Management/Software Distribution/Packages** in SCCM. The **packageName** and **packageId** inputs cannot both be empty at the same time.

### **packageId**

The ID of the software distribution package to delete. You can find the package ID under **Computer Management/Software Distribution/Packages** in SCCM. If you specify a value for the **packageId** input, the value of the **packageName** input is ignored. The **packageName** and **packageId** inputs cannot both be empty at the same time.

The operation returns the following:

### **returnResult**

A message that indicates the status of the operation.

### **retPackageId**

The package ID for the deleted software distribution package.

## Delete Program

The **Delete Program** operation deletes a program from a software distribution package in SCCM. All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### programName

The name of the program to delete.

### packageName

The name of the software distribution package to which the program belongs in SCCM. You can find the package name under **Computer Management/Software Distribution/Packages** in SCCM. The **packageName** and **packageId** inputs cannot both be empty at the same time.

### packageId

The ID of the software distribution package to which the program belongs in SCCM. You can find the package ID under **Computer Management/Software Distribution/Packages** in SCCM. If you specify a value for the **packageId** input, the value of the **packageName** input is ignored. The **packageName** and **packageId** inputs cannot both be empty at the same time.

The operation returns the following:

### returnResult

A message that indicates the status of the operation.

## Update Advertisement

The **Update Advertisement** operation updates advertisements in SCCM. It can only update simple data types and non-key properties. For the **datetime** type in **WMI SMS\_Advertisement**, the input string must be in the format of "`yyyyMMddHHmmss.000000+000`" (for example, "`20100601113000.000000+000`").

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### advertisementId

The ID of the advertisement to update. You can find the advertisement ID in **Computer Management/Software Distribution/Advertisements** in SCCM. If you specify a value for the **advertisementId** input, the value of the **advertisementName** input is ignored. The **advertisementId** and **advertisementName** inputs cannot both be empty at the same time.

### advertisementName

The name of the advertisement to update. You can find the advertisement name in **Computer Management/Software Distribution/Advertisements** in SCCM. The **advertisementId** and **advertisementName** inputs cannot both be empty at the same time.

### attrNameValuePair

A delimited list of the property name and value pairs of **WMI SMS\_Advertisement**. Key name and value pairs are ignored. If an item in the list is not in the format `Name=Value pair`, it is ignored.

### delimiter

The delimiter for the **attrNameValuePair** input. The default is comma (,).

The operation returns the following:

### returnResult

A message that indicates the status of the operation.

## retAdvertisementId

If you specify a value for the **advertisementId** input, this is the ID of the updated advertisement. Otherwise, it is a comma-delimited list of advertisement IDs.

## Update Package

The **Update Package** operation updates a software distribution package in SCCM. It can only update simple data types and non-key properties. For the **datetime** type in **WMI SMS\_Package**, the input string must be in the format "*yyyyMMddHHmmss.000000+000*" (for example, "**20100601113000.000000+000**").

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### packageName

The name of the software distribution package to update. You can find the package name under **Computer Management/Software Distribution/Packages** in SCCM. The **packageName** and **packageId** inputs cannot both be empty at the same time.

### packageId

The ID of the software distribution package to update. You can find the package ID under **Computer Management/Software Distribution/Packages** in SCCM. If you specify a value for the **packageId** input, the value of the **packageName** input is ignored. The **packageName** and **packageId** inputs cannot both be empty at the same time.

### attrNameValuePair

A delimited list of the property name and value pairs of **WMI SMS\_Package**. Key name and value pairs are ignored. If a value is not in the format *Name=Value pair*, it is ignored.

### delimiter

The delimiter for the **attrNameValuePair** input. The default is a comma (,).

The operation returns the following:

### returnResult

A message that indicates the status of the operation.

### retPackageId

The ID of the updated package.

## Update Program

The **Update Program** operation updates a program in a software distribution package in SCCM. It can only update simple data types and non-key properties. For the **datetime** type in **WMI SMS\_Program**, the input string must be in the format "*yyyyMMddHHmmss.000000+000*" (for example, "**20100601113000.000000+000**").

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### programName

The name of program to update. You can find the program name under the **Programs** folder in each software distribution package in SCCM.

### packageName

The name of the software distribution package to which the program belongs. You can find the package name under **Computer Management/Software Distribution/Packages** in SCCM. The **packageName** and **packageId** inputs cannot both be empty at the same time.

### **packageId**

The ID of the software distribution package to which the program belongs. You can find the package ID under **Computer Management/Software Distribution/Packages** in SCCM. If you specify a value for the **packageId** input, the value of the **packageName** input is ignored. The **packageName** and **packageId** inputs cannot both be empty at the same time.

### **attrNameValuePair**

A delimited list of the property name and value pairs of the **WMI SMS\_Program**. Key name and value pairs are ignored. If a value is not in the format *Name=Value pair*, it is ignored.

### **delimiter**

The delimiter for the **attrNameValuePair** input. The default is comma (,).

The operation returns the following:

### **returnResult**

A message that indicates the status of the operation.

## **Software Update**

### **Add Updates to Deployment Package**

The **Add Updates to Deployment Package** operation assigns software updates to a software update deployment package in SCCM. It assigns the content IDs and content source paths. The updated contents must be already downloaded to the network shares on the SCCM site.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### **packageName**

The name of the software update deployment package. You can find the package name under **Computer Management/Software Updates/Deployment Packages** in SCCM. If you specify a value for the **packageId** input, the value of the **packageName** input is ignored. The **packageName** and **packageId** inputs cannot both be empty at the same time.

### **packageId**

The ID of the software update deployment package. You can find the package ID under **Computer Management/Software Updates/Deployment Packages** in SCCM. If you specify a value for the **packageId** input, the value of the **packageName** input is ignored. The **packageName** and **packageId** inputs cannot both be empty at the same time.

### **contentIds**

A delimited list of software update content IDs. You can use the **Get Content Id By Software Update CI Id** operation to get the content IDs. The list of content IDs should match the list in the **contentSourcePaths** input. The contents must already be downloaded to the content source paths.

### **contentSourcePaths**

A delimited list of software update content source paths. The paths must be in network share format (for example, `\\server\share`, `\\server2\share`). The list in the **contentIds** input should match the list in **contentSourcePaths**.

### **delimiter**

The delimiter for the **contentIds** and **contentSourcePaths** input values. The default value is a comma (,).

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

## Create Deployment

The **Create Deployment** operation creates a software update assignment to deploy software updates in SCCM. It specifies which collection to deploy and which CI IDs should be included.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**deploymentName**

The name of this software update deployment. After the operation creates the deployment, you can find it under **Computer Management/Software Updates/Deployment Management** in SCCM.

**assignedCIs**

A delimited list of software update CI IDs. You can use the **Get Software Update CI Ids** operation to search the CI IDs. The CI IDs can also be found in **WMI SMS\_SoftwareUpdate**.

**collectionName**

The name of the collection to which the software update CIs are assigned in SCCM (for example, **My Test Systems**). The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

**collectionId**

The ID of the collection to which the software update CIs are assigned in SCCM (for example, **ROS00030**). You can find the collection ID in the collection folder's properties in SCCM. If you specify a value for the **collectionId** input, the value of the **collectionName** input is ignored. The **collectionId** and **collectionName** inputs cannot both be empty at the same time.

**description**

The description of the software update deployment package.

**delimiter**

The delimiter for the values in the **assignedCIs** input. The default value is a comma (,).

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

**retDeploymentId**

The assignment (deployment) ID for the created software update deployment on the collection. You can find the deployment ID in **Computer Management/Software Updates/Deployment Management** in SCCM. You can find the assignment ID (deployment ID) in **SMS\_UpdatesAssignment**.

**retCollectionId**

The ID of the collection to which the software updates are deployed. You can find the collection ID in **Computer Management/Collections** in the collection's properties.

## Create Deployment Package

The **Create Deployment Package** operation creates a software update deployment package in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**packageName**

The unique name of the software update deployment package. After the operation creates the package, you can find the package name under **Computer Management/Software Updates/Deployment Packages** in SCCM.

**packageSourcePath**

The source path of the software update deployment package. It must be a network share path (for example, `\\ros-jolie\UpdatePackages`).

**packageManufacturer**

The manufacturer of the software update deployment package (for example, **HP**).

**packageVersion**

The version of the software update deployment package (for example, **1.0**).

**packageLanguage**

The language of the software update deployment package (for example, **En**).

**description**

The description of the software update deployment package.

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

**retPackageId**

The ID for the created software update deployment package. You can find the package ID under **Computer Management/Software Updates/Deployment Packages** in SCCM.

## Delete Deployment

The **Delete Deployment** operation deletes the software update assignment (deployment) in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**deploymentName**

The name of the software update deployment or assignment to delete. You can find the name under **Computer Management/Software Updates/Deployment Management** in SCCM. If you specify a value for the **deploymentId** input, the value of the **deploymentName** input is ignored. The **deploymentName** and **deploymentId** inputs cannot both be empty at the same time.

**deploymentId**

The deployment ID of the software updates assignment to delete. You can find it under **Computer Management/Software Updates/Deployment Management** in SCCM. You can find the assignment ID in **SMS\_UpdatesAssignment**. If you specify a value for the **deploymentId** input, the value of the **deploymentName** input is ignored. The **deploymentName** and **deploymentId** inputs cannot both be empty at the same time.

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

## retDeploymentId

The deployment ID for the deleted software update assignment (deployment).

## Delete Deployment Package

The **Delete Deployment Package** operation deletes a software update deployment package in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### packageName

The name of the software update deployment package to delete. You can find the package name under **Computer Management/Software Updates/Deployment Packages** in SCCM. If you specify a value for the **packageId** input, the value of the **packageName** input is ignored. The **packageName** and **packageId** inputs cannot both be empty at the same time.

### packageId

The ID of the software update deployment package to delete. You can find the package ID under **Computer Management/Software Updates/Deployment Packages** in SCCM. If you specify a value for the **packageId** input, the value of the **packageName** input is ignored. The **packageName** and **packageId** inputs cannot both be empty at the same time.

The operation returns the following:

### returnResult

A message that indicates the status of the operation.

### retPackageId

The ID for the deleted software update deployment package.

## Get Content Id by Software Update CI Id

The **Get Content Id by Software Update CI Id** operation gets a software update content ID based on the software update CI ID in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### assignedCI

The software update CI ID. You can use the **Get Software Update CI Ids** operation to search for the CI ID. You can also find the CI ID in **WMI SMS\_SoftwareUpdate**.

The operation returns the following:

### returnResult

A message that indicates the status of the operation.

### retContentId

The software update content ID. The content ID can be passed to the **Add Updates to Deployment Package** operation. You can find the content ID in **WMI SMS\_CIToContent**.

## Get Software Update CI Ids

The **Get Software Update CI Ids** operation gets software update CI IDs based on a software update name in SCCM.

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### **softwareUpdateName**

The name of the software update in which to search in SCCM (for example, "**Update for Windows Server 2008 R2 x64 Edition (KB976662)**"). You can specify an exact or partial name for the input value. You can find the update name or title of a software update in the **Software Updates/Update Repository** after you have successfully synchronized the **Update Repository**.

### **bulletinId**

The bulletin ID of the software update (for example, **MS10-054**). You can find the bulletin ID of a software update in the **Software Updates/Update Repository** after you have successfully synchronized the **Update Repository**.

### **articleId**

The article ID of the software update (for example, **982214**). You can find the article ID in the **Software Updates/Update Repository** after you have successfully synchronized the **Update Repository**.

### **exactMatch**

Specifies whether you want to search on an exact **softwareUpdateName** input value match or on a partial match. The valid values are **True** and **False**.

The operation returns the following:

### **returnResult**

A message that indicates the status of the operation.

### **retCIIds**

The comma-delimited CI IDs for the software updates.

## **Update Deployment**

The **Update Deployment** operation updates a software update assignment (deployment) in SCCM. It can only update simple data types and non-key properties. For the **datetime** type in **WMI SMS\_SoftwareUpdateAssignment**, the input string must be in the format "**yyyyMMddHHmmss.000000+000**" (for example, "**20100601113000.000000+000**").

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

### **deploymentName**

The name of the software update deployment or assignment to update. You can find it under **Computer Management/Software Updates/Deployment Management** in SCCM. If you specify a value for the **deploymentId** input, the value of the **deploymentName** input is ignored. The **deploymentName** and **deploymentId** inputs cannot both be empty at the same time.

### **deploymentId**

The deployment ID of the software update assignment to update. You can find the deployment ID in the **Computer Management/Software Updates/Deployment Management** in SCCM. If you specify a value for the **deploymentId** input, the value of the **deploymentName** input is ignored. The **deploymentName** and **deploymentId** inputs cannot both be empty at the same time.

### **attrNameValuePair**

A delimited list of the property name and value pairs of **WMI SMS\_UpdatesAssignment**. Key name and value pairs are ignored. If an input value is not in the *Name=Value pair* format, it is ignored.

**delimiter**

The delimiter for the **attrNameValuePair** input values. The default value is a comma (,).

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

**retDeploymentId**

The deployment ID for the updated software update assignment.

## Update Deployment Package

The **Update Deployment Package** operation updates a software update deployment package in SCCM. It can only update simple data types and non-key properties. For the **datetime** type in **WMI SMS\_SoftwareUpdatePackage**, the input string must be in the format `"yyyymmddHHmmss.000000+000"` (for example, `"20100601113000.000000+000"`).

All of the operation's inputs except the following are described in [Common inputs in the integration](#).

**packageName**

The name of the software update deployment package to update. You can find the package name under **Computer Management/Software Updates/Deployment Packages** in SCCM. The **packageName** and **packageId** inputs cannot both be empty at the same time.

**packageId**

The ID of the software update deployment package to update. You can find the package ID under **Computer Management/Software Updates/Deployment Packages** in SCCM. If you specify a value for the **packageId** input, the value of the **packageName** input is ignored. The **packageName** and **packageId** inputs cannot both be empty at the same time.

**attrNameValuePair**

A delimited list of the property name and value pairs of **WMI SMS\_SoftwareUpdatePackage**. Key name and value pairs are ignored. If a value it is not in the format *Name=Value pair*, it is ignored.

**delimiter**

The delimiter for the values in the **attrNameValuePair** input. The default value is a comma (,).

The operation returns the following:

**returnResult**

A message that indicates the status of the operation.

**retPackageId**

The ID of the updated package.

## Troubleshooting

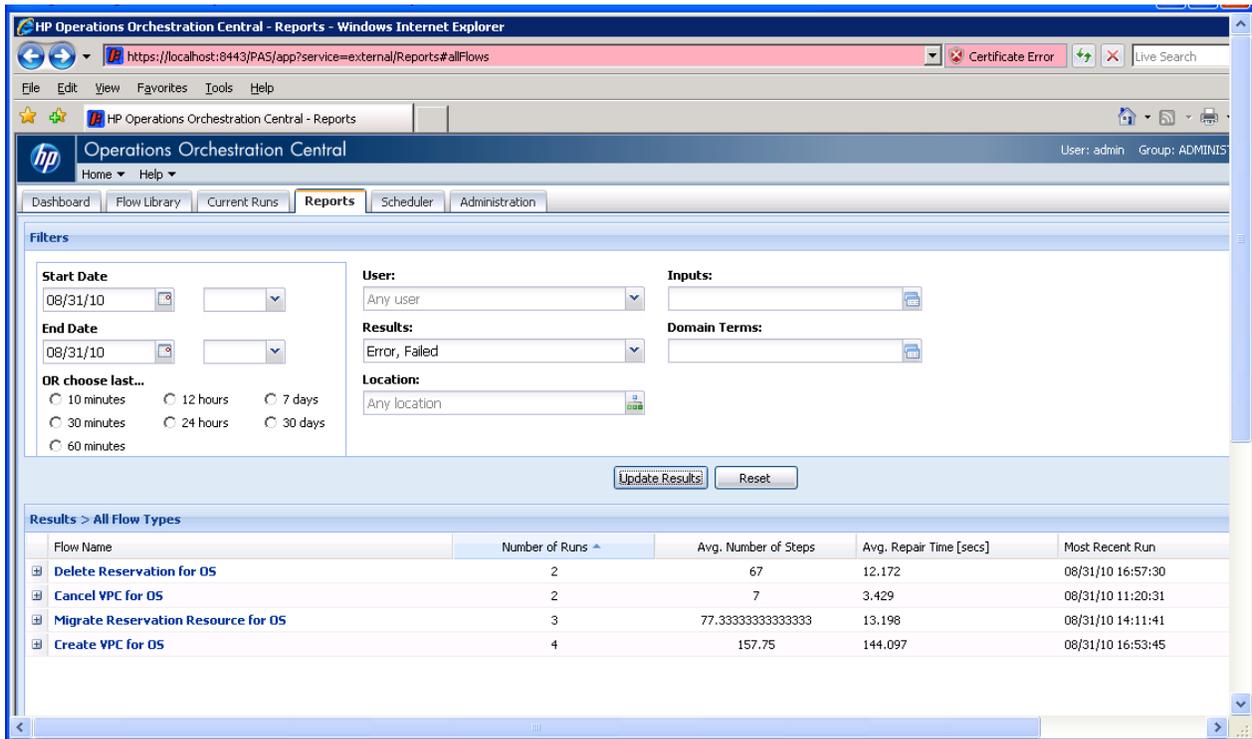
This section provides troubleshooting procedures and tools you can use to solve problems you may encounter while using this integration. It also includes a list of the error messages you may receive while using the integration and offers descriptions and possible fixes for the errors.

## General troubleshooting procedures and tools

This section describes the troubleshooting procedures and tools you can use to fix problems you may experience while using this integration.

### If a flow fails when run in OO Central

1. In Operations Orchestration Central, click the **Reports** tab and select **Error** and **Failed** in the **Results** list, or select nothing.



The screenshot shows the HP Operations Orchestration Central interface in a Windows Internet Explorer browser. The browser address bar shows the URL: <https://localhost:8443/PAS/app?service=external/Reports#allFlows>. The page title is "HP Operations Orchestration Central - Reports". The navigation menu includes Dashboard, Flow Library, Current Runs, Reports (selected), Scheduler, and Administration. The user is logged in as "admin" with the group "ADMINIS".

The "Filters" section contains the following fields:

- Start Date:** 08/31/10
- End Date:** 08/31/10
- OR choose last...:** Radio buttons for 10 minutes, 30 minutes, 60 minutes, 12 hours, 24 hours, 7 days, and 30 days.
- User:** Any user
- Results:** Error, Failed
- Location:** Any location
- Inputs:** (Empty text field)
- Domain Terms:** (Empty text field)

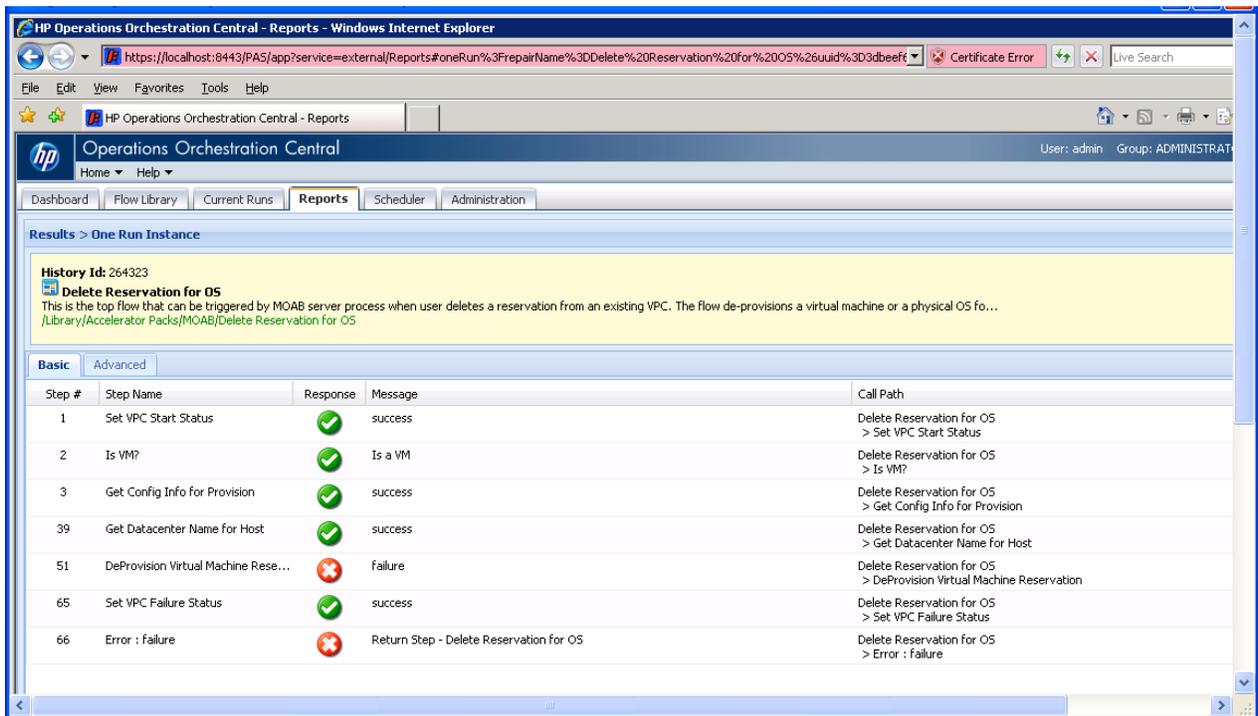
Buttons for "Update Results" and "Reset" are located below the filters.

The "Results > All Flow Types" table displays the following data:

Flow Name	Number of Runs	Avg. Number of Steps	Avg. Repair Time [secs]	Most Recent Run
Delete Reservation for OS	2	67	12.172	08/31/10 16:57:30
Cancel VPC for OS	2	7	3.429	08/31/10 11:20:31
Migrate Reservation Resource for OS	3	77.33333333333333	13.198	08/31/10 14:11:41
Create VPC for OS	4	157.75	144.097	08/31/10 16:53:45

Figure 3 - OO Central Reports tab

2. Find the failed flow you want to examine and click it. For example, in the above figure, you could click **Delete Reservation for OS** and then click the **History ID** of the job based on time.



**Figure 4 - OO Central History ID**

3. Locate the failure step.  
For example, in the above figure note that step number 51 failed.
4. To research the failure, click the **Advanced** tab.

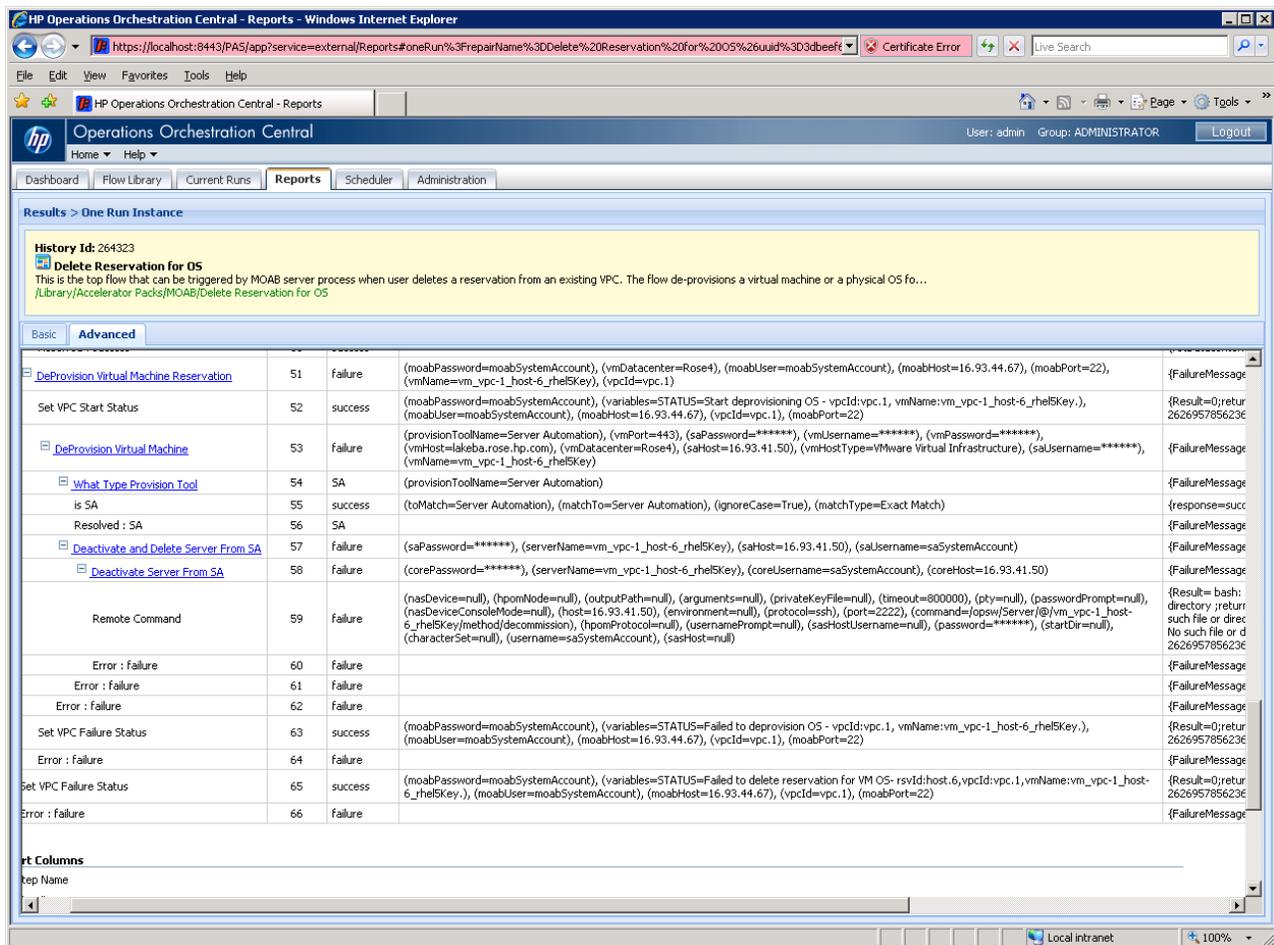


Figure 5 - OO Central History ID Advanced tab

5. In the **Report** column, select the step name, step number, response, bound inputs, and result.
6. Find step 51 and find which step actually failed.
7. Examine the error message from the result.

If you run the flow in OO Studio using the **Debug** feature, you can use the **Step Result Inspector** to check the errors and exceptions in detail.

Similar information can also be found in the OO Central log Central\_wrapper.log located in the OO home folder in /Central/logs/. For information about the RAS service, check the log wrapper.log located in the OO home folder in /RAS/Java/Default/webapp/logs/.

## Error messages

This section lists the error messages you may receive while using this integration. Each error message includes possible causes and fixes for the error.

### The RPC server is unavailable. (Exception from HRESULT: 0x800706BA);

The SMS provider is not available. Check that the input for the host is correct and that the server is running.

### Access is denied. (Exception from HRESULT: 0x80070005 (E\_ACCESSDENIED));

This message indicates username or password might be wrong. Or the user doesn't have access to the SMS provider. You can use the wbemtest to check the user access.

## Invalid namespace

The site code might not be correct. Make sure you have correct site code.

## Generic failure

You may get this message for one of the following reasons:

- If you try to remove a resource that was dynamically added by query in a dynamic collection.
- When you import a new system into SCCM, if the value of either of the **macAddress** or **smBiosGuid** inputs is in the wrong format. Make sure that the value of the **macAddress** input is in the format of the following example: "00:50:56:94:00:31", and that the value of the **smBiosGuid** input is in the format of the following example: "421440F7-2DF1-6453-0C4C-5A4B04442B17".
- If you enter an invalid CI ID when you create a deployment.
- If you enter an invalid content ID when you add updates to a deployment package.

## Resource (<resourceName>) not found in SCCM

This error occurs when a specified resource is not in SCCM or the resource type is not correct and you try to add a resource to a collection, get resources in a collection, create a static collection with resource names, or remove a resource from a collection. Make sure that the resource has already been discovered by SCCM.

## Object reference not set to an instance of an object

This error occurs for one of the following reasons:

- You update a collection schedule or assign a baseline to a collection, if you enter a schedule that is not defined in the **SCCM – Schedule** selection list. Make sure that you enter a proper schedule string.
- You check a component status, if you enter an incorrect component name or tally interval string. Make sure that you enter the correct component name and proper tally interval. You can find the component names under **System Status/Site Status/<site name>/Component Status** in SCCM. You can find the tally interval in the **SCCM – Tally Interval** selection list.

## Failed to create distribution point for package (ROS00026) in SCCM

This error can happen when you assign a distribution point to a package and the distribution point is not valid or the distribution point resource type is wrong for the distribution point. Make sure that the distribution point is valid with a proper distribution point resource type. The valid values for distribution point types are **Window NT Server** and **Window NT Share**.

## Software distribution advertisement did not show up

This error occurs when you create an advertisement with the wrong program name. The operation is successful, but the advertisement does not show up in **Computer Management/Software Distribution/Advertisements** in SCCM. Make sure that the program name is a correct name in the package.

## Not found

This error occurs when you enter an incorrect name when you update an advertisement, package, program, deployment, and deployment package. Check the SMS class for the correct name.

## **Input string was not in a correct format**

### **Type mismatch**

### **String was not recognized as a valid Boolean**

These errors can occur if you enter an incorrect value which when you update an advertisement, package, program, deployment, and deployment package. Check the SMS class for the correct value format.

### **Add updates to deployment package, but updates did not show up in package source path**

Make sure that the update contents in the content source path are available on the network and that the content source path is a valid network path. Also make sure that the deployment package's package source path is a valid network path. The deployment package source path is the target path to which the source content is copied.

## **Security**

The SCCM site SMS provider and the SCCM client SMS provider are accessed via the .NET WMI API. The username and password are the domain username and password that you use to log on to the SCCM site SMS provider or client SMS provider.

## **Tools**

Following are OO tools that you can use with the SCCM integration:

- **RSFlowInvoke.exe** and **JRSFlowInvoke.jar**

RSFlowInvoke (RSFlowInvoke.exe or the Java version, JRSFlowInvoke.jar) is a command-line utility that allows you to start a flow without using Central (although the Central service must be running). RSFlowInvoke is useful when you want to start a flow from an external system, such as a monitoring application that can use a command line to start a flow.

- **Web Services Wizard (wswizard.exe)**

When you run the Web Services Wizard, you provide it with the WSDL for a given Web service. The WSDL string you provide as a pointer can be a file's location and name or a URL. The Web Services Wizard displays a list of the methods in the API of the Web service that you specify. When you run the wizard, pick the methods you want to use, and with one click for each method you have selected, the wizard creates an HP OO operation that can execute the method. This allows you to use the Web Services Wizard to create operations from your monitoring tool's API.

These tools are available in the Operations Orchestration home directory under the /Studio/tools/ folder.