

# HP Operations Orchestration

Software Version: CP16 (9.x)

Windows and Linux operating systems

## OpenStack Integration Guide

Document Release Date: November 2014

Software Release Date: November 2014



## Legal Notices

### Warranty

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

The information contained herein is subject to change without notice.

### Restricted Rights Legend

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

### Copyright Notice

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

### Trademark Notices

Adobe™ is a trademark of Adobe Systems Incorporated.

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

UNIX® is a registered trademark of The Open Group.

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

## Documentation Updates

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

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to: <http://h20229.www2.hp.com/passport-registration.html>

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

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

## Support

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:

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

To find more information about access levels, go to:

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

**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

Introduction .....	5
Overview .....	5
About OpenStack Integration .....	5
Supported Versions .....	7
Downloading OO Releases and Documents on HP Live Network .....	7
Installing and Configuring the Integration .....	8
Getting Started .....	9
OpenStack Use Cases .....	9
OpenStack Architecture .....	14
Audience .....	14
Location of OpenStack Integration Operations and Flows in OO Studio .....	15
Creating Custom OpenStack Integration Operations .....	16
Using Studio to Create OO Operations .....	16
General Troubleshooting Procedures and Tools .....	17
REST Client .....	17
Network Sniffers .....	17
Logs .....	17
Error Messages .....	17
Security .....	19
Authentication .....	19
Identity Service .....	19
Tenants and Scoped vs. Unscoped Tokens .....	19
Getting and Using Auth Tokens .....	20
Legacy Auth Tokens .....	20
Privacy and Other HTTP Concerns .....	21
OO Tools You Can Use with the OM Incident – OO Integration .....	22
OpenStack Terminology .....	22

# Introduction

This chapter includes:

Overview .....	5
About OpenStack Integration .....	5
Supported Versions .....	7
Downloading OO Releases and Documents on HP Live Network .....	7
Installing and Configuring the Integration .....	8

## Overview

With the HP Operations Manager (HPOM) Incident Web Service integration, administrators can build HP Operations Orchestration (OO) flows that are integrated with HPOM through the Incident Web Service interface.

This document explains how this integration has been implemented and how the integration's OO operations communicate between OO and the HPOM Incident Web Service interface.

The HP Operations Manager Incident Web Service interface exposes its messages as IT Infrastructure Library (ITIL) incidents and uses terminology (such as Close and Reopen) that is compliant with the Distributed Management Task Force (DMTF) Web Services for Management (WS-Management) standard. This terminology is different, yet similar, to HPOM native terminology.

## About OpenStack Integration

This integration enables administrators to create HP Operations Orchestration (OO) flows that are integrated with OpenStack. The OpenStack integration uses the REST API (via OO's HTTP Client) to integrate with OO.

OpenStack is a global collaboration of developers and cloud computing technologists producing a global open source cloud computing platform for public and private clouds. The project aims to deliver solutions for all types of clouds by being simple to implement, massively scalable, and feature rich. The technology consists of a series of interrelated projects delivering various components for a cloud infrastructure solution.

Founded by Rackspace Hosting and NASA, OpenStack has grown to be a global software community of developers collaborating on a standard and massively scalable open source cloud operating system. Rackspace contributed their "Cloud Files" platform (code) to power the Object Storage part of the OpenStack, while NASA contributed their "Nebula" platform (code) to power the Compute part. OpenStack enables organizations to create and offer cloud computing services running on standard hardware.

There are six main OpenStack components that OO integrates with:

- Compute Infrastructure (Nova)
- Storage Infrastructure (Swift)
- Imaging Service (Glance)
- Identity (Keystone)
- Cinder (Volume)
- Neutron (Network)

The OpenStack Integration allows you to easily integrate and interact from HP Operations Orchestration to HP Cloud Services.

The HP Cloud offers you freedom to choose, experiment and grow. From a variety of languages to third-party services, you choose how you develop in the cloud. Our public cloud is developer friendly and backed by exceptional customer support. That's the advantage of HP. Focus on your business, and let HP Cloud Services handle your infrastructure and platform needs. For more information on HP Cloud Services, visit the official [www.hpcloud.com](http://www.hpcloud.com) website.

## Supported Versions

Operations Orchestration Version	OpenStack
OO Content Pack 16	Diablo 2011.3 Essex 2012.1 Grizzly 2013.1 Havana 2013.2 IceHouse 2014.1 Juno 2014.2 Helion Public Cloud 13.5 Helion Openstack 1.0 or versions compatible with the APIs listed below: <a href="#">OpenStack Compute API v2 Reference</a> <a href="#">OpenStack Image Service API v1 Reference</a> <a href="#">OpenStack Object Storage API 1.0 Reference</a> <a href="#">OpenStack Identity Service API 2.0 Reference</a> <a href="#">OpenStack Block Storage API 2.0 Reference</a> <a href="#">OpenStack Networking API v2.0 Reference</a>

## Downloading OO Releases and Documents on HP Live Network

HP Live Network provides an Operations Orchestration Community page where you can find and download supported releases of OO and associated documents.

**Note:** The Community page requires that you register for an HP Passport and sign-in.

To register for an HP Passport ID:

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

Or

Click the **New users - please register** link on the HP Passport login page

To download OO releases and documents:

1. Go to the HPLN site: <https://hpln.hp.com/>. Page 1 of HP Live Network page opens.
2. Go to page 2 and click the **Content** link under **Operations Orchestration**.



## Operations Orchestration

[Announcements](#) | [Forum](#) | [Content](#)

Optimize operational cost and  
Improve service quality by  
enabling end-to-end IT  
Process Automation

3. From the **Content Catalog** tab, select **Operations Orchestration Content Packs**.
4. Select the Downloads link.
5. Click **Downloads > HP Operations Orchestration 10.00**.
6. Search for the required HP Operations Orchestration Content Pack in which the integration can be found (oo10-hp-solutions-cp).

## Installing and Configuring the Integration

There is no special installation required for the OpenStack integration for Windows other than a working OpenStack environment.

The OpenStack environment has a number of different installation options:

- Red Hat Enterprise Linux, CentOS, and Fedora
- OpenStack Installation Guide for Ubuntu 12.04 (LTS)

For more information, see OpenStack Integration Guide Operations Manager available on <http://support.openview.hp.com/selfsolve/manuals>.



# Getting Started

This chapter includes:

OpenStack Use Cases .....	9
OpenStack Architecture .....	14
Audience .....	14
Location of OpenStack Integration Operations and Flows in OO Studio .....	15
Creating Custom OpenStack Integration Operations .....	16

## OpenStack Use Cases

The following use cases are developed using OO flows only and can be modified without changing the Java code.

1. OpenStack Compute Integration – Block Storage-Snapshots (OpenStack/Compute/ Block Storage/Snapshots)
  - Create Snapshot
  - Delete Snapshot
  - Get Snapshot Details
  - Get Snapshots List
2. OpenStack Compute Integration – Block Storage-Volumes (OpenStack/Compute/Block Storage/Volumes)
  - Attach Volume to Server
  - Create Volume
  - Create Volume from Snapshot
  - Delete Volume
  - Detach Volume from Server
  - Get Volume Details

- Get Volumes Attached to Server
- Get Volumes List
- 3. OpenStack Compute Integration - Flavors (OpenStack/Compute/Flavors)
  - Get Flavor Details
  - Get Flavors Detail List
  - Get Flavors List
- 4. OpenStack Compute Integration - Images (OpenStack/Compute/Images)
  - Delete Image
  - Get Image Details
  - Get Images Detail List
  - Get Images List
- 5. OpenStack HP Cloud extension - Key Pairs (OpenStack/Compute/Key Pairs)
  - Create Key Pair
  - Delete Key Pair
  - Get Key Pairs List
- 6. OpenStack Compute Integration - Servers Metadata (OpenStack/Compute/Server Metadata)
  - Delete Image Metadata Item
  - Delete Server Metadata Item
  - Get Image Metadata Item
  - Get Image Metadata List
  - Set Image Metadata
  - Set Image Metadata Item
  - Set Server Metadata
  - Set Server Metadata Item

- Update Image Metadata
  - Update Server Metadata
7. OpenStack HP Cloud extension - Security Groups (OpenStack/Compute/Security Groups)
- Create Security Group
  - Create Security Group Rule
  - Delete Security Group
  - Delete Security Group Rule
  - Get Security Group Details
  - Get Security Group List
8. OpenStack Compute Integration - Server Actions (OpenStack/Compute/Server Actions)
- Change Server Admin Password
  - Confirm Resize Server
  - Create Image
  - Reboot Server Hard
  - Reboot Server Soft
  - Rebuild Server
  - Resize Server
  - Revert Resize Server
9. OpenStack HP Cloud extension - Floating IPs (OpenStack/Networking/Floating IPs)
- Add Floating IP to Server
  - Create Floating IP
  - Delete Floating IP
  - Get Addresses List
  - Get Addresses List by Network
  - Get Floating IP Address Details

- Get Floating IP Address List
  - Remove Floating IP from Server
10. OpenStack Compute Integration - Servers (OpenStack/Compute)
- Create Server
  - Delete Server
  - Get Server Details
  - Get Servers Detail List
  - Get Servers List
  - Update Server Name
  - Update Server IPv4 Access Address
  - Update Server IPv6 Access Address
11. OpenStack Identity Service Integration (OpenStack/Identity Service)
- Get Tenant List
12. OpenStack Image Service Integration (Openstack/ Image Service)
- Add a Member to an Image
  - Add a Member to an Image with Can Share
  - Download VM Image
  - Get Image Memberships
  - Get Public VM Images Detail List
  - Get Public VM Images List
  - Get Shared Images
  - Get VM Image Detailed Metadata
  - Remove a Member from an Image
  - Replace a Membership List for an Image

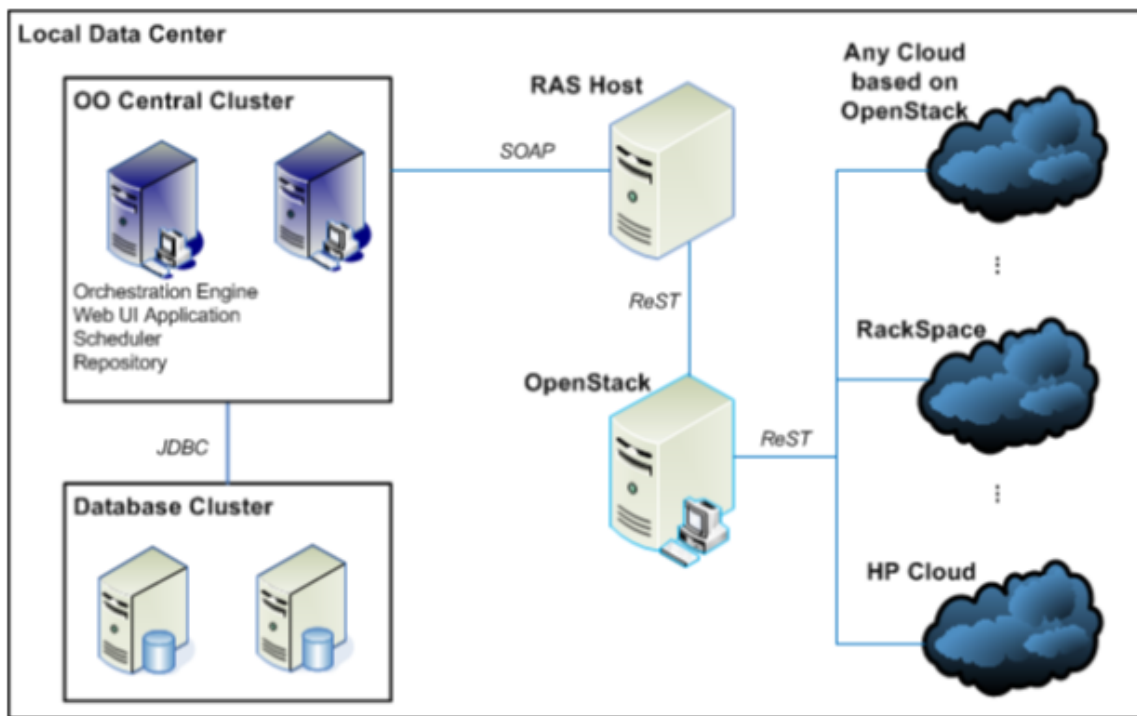
- Update VM Image
  - Upload VM Image
13. OpenStack Object Storage Integration - Storage Container Services (OpenStack/Object Storage/Storage Container)
- Create Container
  - Delete Container
  - Get Container List
  - Get Container Metadata
  - Get Container Object List
14. OpenStack Object Storage Integration - Storage Object Services (OpenStack/Object Storage/Storage Object)
- Copy Storage Object
  - Create or Update Storage Object
  - Delete Storage Object
  - Get Storage Object Metadata
  - Retrieve Storage Object
  - Update Storage Object Metadata
15. Utility flows (OpenStack/Utilities)
- Add to Name Value Lists
  - Select HTTP Client Error
16. OpenStack
- Get Auth Token
  - Get Unscoped Auth Token
  - OpenStack REST Delete
  - OpenStack REST Get
  - OpenStack REST Get to File

- OpenStack REST Head
- OpenStack REST Post
- OpenStack REST Put

17. OpenStack/Compute

- OpenStack Dashboard

## OpenStack Architecture



## Audience

This guide is intended for system administrators who establish and maintain the implementation of integration between OpenStack and HP OO. This guide assumes that you have administrative access to both systems.

# Location of OpenStack Integration Operations and Flows in OO Studio

The OpenStack integration includes the following operations and flows after deployment of oo-cloud Content Pack in OO Studio: **Library/Integrations/OpenStack**.



## Creating Custom OpenStack Integration Operations

If you need to perform tasks that are not currently supported by the OpenStack integration's operations, you can use OO Studio to create new operations. This requires that you have knowledge of the OpenStack API.

### Using Studio to Create OO Operations

The OpenStack Integration is based on flows that use the HTTP Client to perform operations, and additional flows can be created using the existing OpenStack flows as examples. See to the *Application Program Interface (API) Guide* for information on how to develop REST-based flows using HP Operation Orchestration, located at <https://hpln.hp.com/>. For instructions on using the HP Operations Orchestration Studio, see the *Studio Authoring Guide* in the documentation set for the current OO release.



## General Troubleshooting Procedures and Tools

In general, troubleshooting problems with the OpenStack integration will involve referencing the OpenStack API Guide for the operation you are having trouble with, and possibly using a standalone REST client, as described below, to validate the data that OO is returning.

### REST Client

A standalone Rest Client tool, the Advanced Rest Client available in Google Chrome can be used for an integration of this type. This allows you to perform HTTP requests to verify the APIs you are working with and examine the response status, code headers, and body.

All of the operations in the OpenStack integration should contain a step which uses one of the OpenStack REST <HTTP request> operations, where <HTTP request> can be, GET, POST, DELETE, and so on. The URL to use in a standalone REST client is obtained by combining the endpoint and uri inputs of the OpenStack REST <HTTP request> operation. The request body is obtained from the string input if necessary. There can also be request headers associated with the operation, and these can be seen in the headerNamesList and headerValuesList of the OpenStack REST <HTTP request> step.

Once all this information has been transposed into the REST client, you send the request, and validate that the data returned is the same as what OO is seeing. The standalone REST client also makes it easier to modify inputs, and test the results.

### Network Sniffers

For advanced troubleshooting, you can use a network sniffer, such as Wireshark, to examine exactly what is being sent to, and received from, the OpenStack server.

### Logs

If you encounter problems, the available API logs for the different OpenStack components to help you solve those issues. For more information about the different OpenStack components, refer to the Administration Guide for the problematic OpenStack component.

### Error Messages

1. All the flows in the OpenStack integration return an http response code with a brief description of the error, in the event that an error occurs. For example:

```
This server could not verify that you are authorized to access the document
you requested.
Either you supplied the wrong credentials (e.g., bad password), or your
browser does not understand
how to supply the credentials required.
```

This message was returned by the Get Server Details operation, due to an authentication failure.

This means that there is a problem with the authToken supplied to the operation.

2. If authentication succeeds, the error message returned should be in JSON format, but still has the code and brief description. For example:

```
{"itemNotFound": {"message": "The resource could not be found.", "code":
404}}
```

This error message was also returned from the Get Server Details operation, this time due to an invalid server ID being supplied

3. Some of the flows in the Openstack integration require the supply of either JSON objects or JSON arrays.

- valid JSON array format : [{"addr": "192.168.0.3", "version": 4 }]

- valid JSON object format : { "My Server Name" : "Apache1", "min\_ram" : "2" }

If an invalid format of JSON is used, an error similar to the one below will occur:

```
Expecting to find valid JSON in request body. The server could not comply
with the request since
it is either malformed or otherwise incorrect. The client is assumed to be
in error.
```

4. Some flows require a certain state for the Openstack object that is used. The following is an example of running the flow Start Server when the server specified is already running:

```
Instance <instance_uuid> in vm_state active. Cannot start while the instance
is in this state.
```

# Security

## Authentication

OpenStack uses the concept of an authorization token to control access to its API. You make an initial request to the OpenStack system with your login credentials, which returns a token that is valid for a selected period of time set by the system administrator. This token is then used for all subsequent API calls.

With the exception of the two operations used to get an auth token, every OpenStack operation has a required `authToken` input. This means that every flow which uses the OpenStack operations, starts with a step to retrieve an auth token, which should remain valid for the duration of the flow.

For more information on the authentication process, see the examples below

## Identity Service

The current method used to manage access to an OpenStack environment is using the OpenStack Identity Service (Keystone). Two operations are provided to interact with the identity server:

- **Get Auth Token** : requires a username, password and tenant
- **Get Unscoped Auth Token** : requires only a username and password

## Tenants and Scoped vs. Unscoped Tokens

With the introduction of the identity service to OpenStack, comes the concept of a tenant. A tenant is basically a group of resources, such as compute and object storage. Each user login can be granted access to any number of tenants, and multiple logins can be granted access to the same tenant.

This makes it possible for a group of developers to each have their own login, and share a common tenant, or a single manager who has control over a number of tenants. The only difference between a scoped and unscoped auth token is that a scoped auth token is tied to a tenant, and an unscoped token is not. What this means is that every OpenStack operation which manages a resource requires a scoped auth token. The only use for an unscoped token is to retrieve a list of tenants which a particular user has access to, which is done using the operation **OpenStack/Identity Service/Get Tenant List**.

For more information on when you would need an unscoped auth token, and the differences between scoped and unscoped tenants, refer to the following document: <http://api.openstack.org/api-ref-identity.html>.

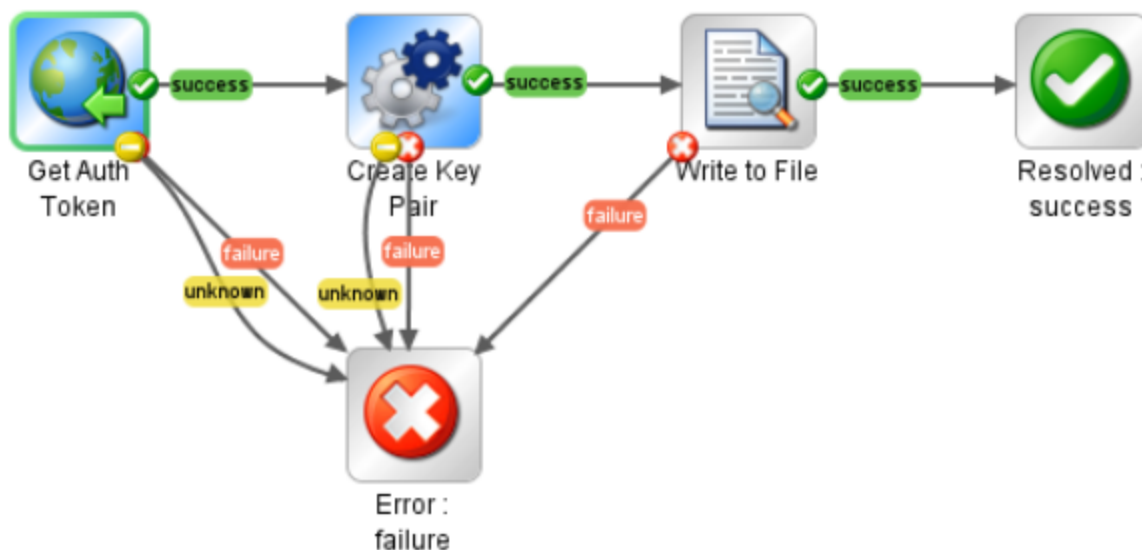
## Getting and Using Auth Tokens

To get an auth token, you must use one of the two operations mentioned previously (Get Auth Token and Get Unscoped Auth Token). In almost all cases, you will need to use **Get Auth Token**.

The following sample flows demonstrate the use of Get Auth Token:

- OpenStack/Compute/Samples/Create Keypair and Save Private Key
- OpenStack/Compute/Samples/Create Web Server Rule Group
- OpenStack/ObjectStorage/Samples/Delete Container with Contents
- OpenStack/ObjectStorage/Samples/Push Filesystem to the Cloud

The following flow shows a sample of **Create Keypair** and **Save Private Key**.



For an example of how to use an unscoped token, reference the following sample:

**OpenStack/Compute/Samples/OpenStack Dashboard Multitenant**

## Legacy Auth Tokens

Prior to the use of the OpenStack identity service, an auth token was obtained by an http GET request to the nova-api server using request headers that contain an access key and a secret key. Although this integration is not tested against older versions of OpenStack, the API should be backwards compatible, and it should be possible to create a flow which requests an auth token and generates the API endpoints required by the operations being used.

For more information, consult the API guide for the OpenStack version you are using.

## Privacy and Other HTTP Concerns

As the OpenStack integration communicates with the OpenStack server using the http protocol, it inherits all of the strengths and weaknesses of the http protocol.

As such, the following should be considered:

- Any endpoint used in this integration which does not use the https protocol results in data being sent as clear text.
- Any authentication method supported by the http client, such as basic auth, NTLM, kerberos, SSL client certificate authentication, and so on, can be added on top of the OpenStack auth token authentication mentioned above. This may result in multiple sets of credentials being required.

## OO Tools You Can Use with the OM Incident – OO Integration

Following are OO tools that you can use with the OM Incident 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 folder in **/Studio/tools/**.

## OpenStack Terminology

The following terms are used in the OpenStack Integration Guide.

**Cinder.** Project name for the Networking Service, which provides persistent block-level storage devices for use with OpenStack compute instances. The Cinder system manages the creation, attaching and detaching of the block devices to servers. Block storage volumes are fully integrated into OpenStack Compute and the Dashboard allowing for cloud users to manage their own storage needs. Snapshot management provides functionality for backing up data stored on block storage volumes. Snapshots can be restored or used to create a new block storage volume.

**Compute.** OpenStack Compute is a compute service that provides server capacity in the cloud. Compute Servers come in different flavors of memory, disk space, and CPU, and can be provisioned in minutes. Interactions with Compute Servers can occur programmatically via the OpenStack Compute API or the Dashboard.

**Flavor.** Flavor is an available hardware configuration for a server. Each flavor has a unique combination of disk space, memory capacity and priority for CPU time.

**Floating IP.** A representation of an external IP address mapped to a OpenStack Networking port attached to an internal network.

**Glance.** Project name for the Image Service software, which is the main image repository piece of OpenStack, it is the place where you will be uploading your images as well as the place from which they will be consumed by the rest of the OpenStack system.

**Image.** Images are your templates for creating new VMs. The project under OpenStack that stores the available images is called Glance.

**Keypairs.** These are simple ssh keys and are your credentials for accessing any running instances. Keypairs are added and managed using the Keypairs section of the user dashboard.

**Keystone.** Project name for the Identity service software, which offers an integrated identity management system for OpenStack. Initially using token-based authentication, but eventually supporting plug-in modules for identity storage (LDAP, DB, file, PAM, Active Directory, etc...), protocols (SAML, OAUTH, OpenID, and so forth...)

**MySQL.** Datastore that stores build-time and run-time state for a cloud infrastructure.

**Neutron.** Project name for the Networking Service (formerly Quantum[14]) is a system for managing networks and IP addresses. Neutron provides networking models which include flat networks or VLANs for separation of servers and traffic. It manages IP addresses, allowing for dedicated static IPs or DHCP. Floating IPs allow traffic to be dynamically rerouted to any compute resources.

**Nova.** Project name for the Compute service that provisions and manages large networks of virtual machines, creating a redundant and scalable cloud computing platform.

**Rabbit MQ.** Provides robust messaging for applications. It is completely open source and based on open standard protocols.

**Router.** A logical entity for forwarding packets across internal subnets and NATting them on external networks through an appropriate external gateway.

**Security Groups.** Security groups at this time exist mostly as tags for the servers and can be consumed via the meta-data API via a simple curl command. Security groups can be specified as part of the "personality" of an instance.

**Server.** A server is a virtual machine instance in the compute system. Flavor and image are requisite elements when creating a server.

**Swift.** Project name for the Object Storage software that creates redundant, scalable object storage using clusters of standardized servers to store petabytes of accessible data. Swift is used as an inexpensive bulk storage system for programmatic object storage.

**Swift Proxy Node.** The Swift node that runs Proxy services and accepts incoming API requests.

**Swift Ring.** The Swift Ring is a set of mappings of OpenStack Object Storage data to physical devices.

**Swift Storage Node.** The node that runs Account, Container, and Object services.

**Snapshot.** A snapshot is a point in time copy of the data contained in a volume.

**Volume.** A volume is a detachable block storage device. It can be attached to one instance at a time.

