HP Operations Orchestration

For Windows and Linux

HP OO Software Version 10.01

Concepts Guide

Document Release Date: August 2013

Software Release Date: August 2013

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

Trademark Notices

Adobe™ is a trademark of Adobe Systems Incorporated.

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

AMD and the AMD Arrow symbol are trademarks of Advanced Micro Devices, Inc.

Google™ and Google Maps™ are trademarks of Google Inc.

Intel®, Itanium®, Pentium®, and Intel® Xeon® are trademarks of Intel Corporation in the U.S. and other countries.

Java is a registered trademark of Oracle and/or its affiliates.

Microsoft®, Windows®, Windows NT®, Windows® XP, and Windows Vista® are U.S. registered trademarks of Microsoft Corporation.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates.

UNIX® is a registered trademark of The Open Group.

Documentation Updates

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

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

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

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:

http://www.hp.com/go/hpsoftwaresupport

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

HP Software 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

Contents

HP OO Concepts

The *HP OO 10.00 Concepts Guide* introduces you to the basic components and ideas behind HP Operations Orchestration.

What is HP Operations Orchestration?

HP Operations Orchestration (**HP OO**) is the industry-leading solution for IT process automation and runbook automation.

HP OO is a system for creating and using actions in structured sequences (called flows) which maintain, troubleshoot, repair, and provision your Information Technology (IT) resources by:

- Checking the health of, diagnosing, and repairing, networks, servers, services, software applications, and individual workstations.
- Checking client, server, and virtual machines for needed software and updates, and, if needed, performing the necessary installations, updates, and distributions.
- Performing repetitive tasks, such as checking status on internal or external web site pages.

Why HP OO?

Why Orchestration?

In many companies, the following issues can result in poor service quality, delayed time-to-market, and high operating costs:

- Incidents – floods of alerts, unnecessary escalations
- Change and releases – too many manual errors, lack of audit trails
- Process management – need for processes for complex tasks, for example, disaster recovery
- Virtualization – inconsistent management of physical and virtual assets

Orchestration enables automation:

- Element automation – automation of either networks, servers, or storage, spanning tasks from provisioning and change management to compliance enforcement and reporting
- Runbook automation – automation of common and repeatable IT processes across all infrastructure tiers, IT groups, and systems
- Integrated automation of applications, servers, networks, storage, and common processes across the data center

- Automation of the business service, with continuous control of each phase of the service life cycle, across the data center and client end points, from automated operations to monitoring and ticketing

Key Benefits of HP OO

The key benefits of HP OO include:

- Reduced operational cost with automation of common tasks and processes
- Improved service quality with accelerated incident resolution
- Improved audit compliance through documentation generation and reporting
- Integration with current IT environment to ensure minimal impact on procedures and tools

Benefits for Flow Authors

Easy-to-use

HP OO Studio offers an intuitive drag-and-wire capability to design, create, share, and customize flows. The drag-and-wire visual interface enables rapid time-to-value. A visual flow debugger makes it easy to debug flows.

Out-of-the-box Content

HP OO offers out-of-the-box content to manage operating systems, databases, app/web servers and networking platforms. You can utilize out-of-the-box integrations with common HP and third-party systems management tools, such as ticketing, monitoring and event consoles, virtualization, CMDB, and data center automation.

Standalone Studio

HP OO Studio is a standalone tool that doesn’t require a connection to Central. All of its repository operations are available offline. If a source control interaction is required, you decide when the interaction occurs. In this manner, remote teams can use various standalone Studios, and it is even possible to author outside of the office network.

Standard Source Control Integration

HP OO Studio integrates with standard source control software. Even the out-of-the box solution is based on a common source control software (SVN). This means that the common capabilities of source control software are available for Studio, so you can connect and use your organization's source control software. This also means that the automation code can reside with other source code and follow the same life cycle (automation as code).

Multi-Authors and Multi-Geographies

HP OO Studio works offline and leverages standard source control software to share work between multiple and distributed authors.

Annotation-Based Content

HP OO Studio includes ‘@Action’ annotations that can be added directly to your own code. This means that your code can be leveraged to be OO content and still be tested in the context of your development framework.

Fine-Grained HP Content

HP OO content has been organized into a set of about 15 content packs. Each content pack provides flows and operations for a functional domain.You have control over which content packs to download and which to deploy. You can use only what you really need and ignore others.

Fine-Grained Customer Content

In HP OO Studio, your content can be separated into projects and managed separately for each author and group. This gives you complete flexibility in defining the flows that are grouped together and the workspace of each author. Different authors get a focused development environment with the flows that are relevant to them, and do not affect other authors' flows.

Benefits for Administrators

UI Experience

The Central web UI has been designed to reflect common OO use cases. Functionalities for the same roles are grouped into the same areas. The permissions assigned to the logged-on user define the parts of the UI that the user sees. By assigning these permissions to roles, the Administrator can formulate a dedicated UI experience for each group of users.

Live Scale Out

In HP OO, you can add a component while the system is live, without restarting the other components. Simply add a new instance and point it to the database. You can scale out RASes via a grouping mechanism, without the need for a load balancer. You can add a RAS instance while the system is live, without needing to change flow.

High Performance

HP OO has a powerful execution mechanism, which enables high performance. HP OO Central can be scaled to reach up to 6500 steps/sec (or operations/second).

Highly Parallelized Runs

The HP OO execution mechanism is based on an asynchronous execution mechanism, which enables large bursts of executions. Central supports launching 100 flows/sec and there is no limit to the number of parallel executions.

Automatic Content Distribution

In HP OO, content binaries are automatically distributed to the appropriate RASes. When there is a change in their version, the system automatically distributes them to the relevant RAS upon their first use.

Isolated Content Dependencies

HP OO lets you use different versions of third-party libraries for every content pack. You can use your own required third party versions and these are not affected by HP changing the out-of-the-box version.

Simple RAS Management

HP OO uses a group mechanism that separates the logical notion of the target RAS (which is what the author is aware of) and the physical notion of the target RAS (which is what the op admin is aware of). For every environment, the op admin can map between the logical and physical notions. Therefore, promotion of content between environments does not require content changes, and the addition of a RAS doesn’t require going back to the author.

Benefits for Integrators

Complete REST API

HP OO provides a complete REST API for the entire set of Central functionalities. Every functionality that is exposed in Central is implemented over a public REST API, enabling you to utilize the same set of APIs to integrate your systems with OO.

Live Event Stream

HP OO exposes the execution events in a live stream with RSS and ATOM feed formats. You can listen to the feed and react to execution events immediately.

Embeddable

HP OO is provided not just as a standalone application but also as a WAR file. You can embed it in your application servers and save the need of a separate service and management for that OO.

Benefits for End Users

Simplified Execution Gateway

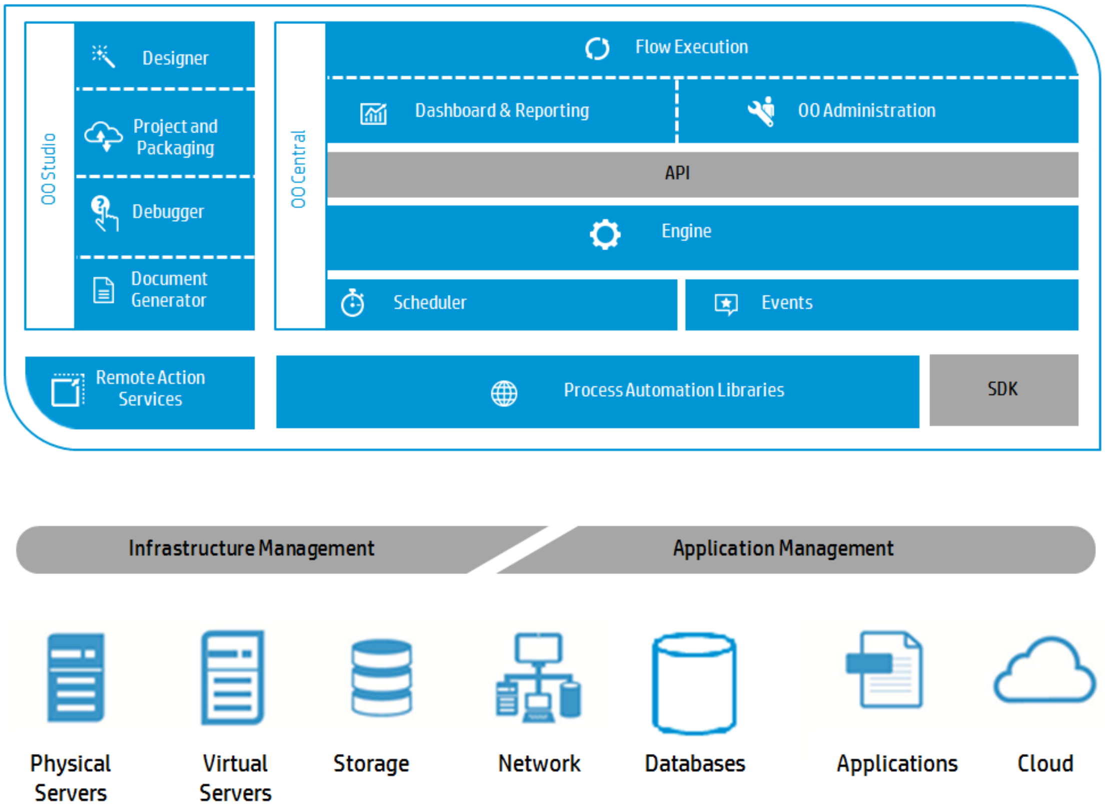
HP OO Central has a refined permission model, which lets you set end user permissions accurately, to allow end users to use Central and be exposed just to the information they need, while limiting their ability to harm the system.

In addition, HP OO Central provides interactive execution capabilities and UI-embedding capabilities, which enable end users to interact with the system in a much more usable fashion.

Mobility

HP OO Central is designed to be mobile-compliant, and can be accessed from various tablets.

Functional Architecture



HP Operations Orchestration 10.00 is composed of four main functional components:

- OO Studio
- OO Central
- OO Remote Action Service (RAS)
- OO Content

Together, the components of HP OO enable you to manage various services and devices across the organization and across their lifecycle.

HP OO Studio

HP OO Studio is a desktop-based application that is used by flow authors to create the HP OO flows. Studio enables the author to design flows, debug them, and package them. It provides automation via code capabilities, such as integration to Source Control Management software, project separation, and multi-authoring.

Designer

HP OO Studio provides a drag-and-wire graphical designer to formulate flows out of various operations and sub-flows.

Debugger

HP OO Studio provides a debugger to test the designed flows. The debugger reflects the behavior of the flow in the Central environment.

Project and Packaging

OO Studio lets you break down your set of authored content into granular projects. Each project groups together contents of a similar nature, for example, according to functionality, development owner, geographic location, and release time lines.

In addition, HP OO Studio lets you package each project to a content pack. The content pack is the read-only run time artifact for the project sources. The content pack is the artifact that is deployed to HP OO Central.

Document Generator

HP OO Studio lets you generate documents for every flow or group of workflows. These auto-generated documents include information about the flow, including its graphical presentation.

HP OO Central

HP OO Central is the run time environment of HP OO. It is used for running flows, monitoring the various runs, and generating reports. It has a web-based UI and a set of APIs, which are accessed by the administrators, end users, and integrators.

HP OO Central is available as a WAR file to be included within an application server or as a standalone installation.

Flow Execution

HP OO Central provides execution capabilities for the deployed flows. The execution is done from a web-based UI and from RESTful APIs. The execution capabilities include browsing the flow library, launching an execution, and tracking that execution until its end.

Dashboard & Reporting

HP OO Central provides reports about the various executions. These include running flows, finished flows, successful and failed flows, and so on.

In addition, the rich set of RESTful APIs let you to gather execution information and to compound a more advanced set of dashboard and reports.

OO Administration

OO Central provides administrators with the means to administrate HP OO from system and operational perspectives. It covers areas such as setting permissions, defining system components, configuring LDAP, and more.

API

HP OO Central provides a complete set of RESTful APIs for every capability. In fact, every capability of the web UI is implemented on top of public RESTful API, enabling you to implement your own web UI and to combine HP OO capabilities within your web applications.

Engine

The engine works behind the scenes of Central. This is the back end component that processes and manages the entire flow execution.The engine manages the step execution, the persistency, and the manual interaction with the users.

Scheduler

HP OO Central includes an out-of-the-box scheduler. This enables the HP OO administrator to define various recurrence patterns for flow executions, to track, and to manage them.

Events

Through the execution of a flow, HP OO Central exposes a live feed (Atom and RSS) for the execution events. Clients can monitor that feed and track the execution in a live manner.

HP OO Remote Action Service (RAS)

The HP OO RAS enables execution in remote data centers and networks. The HP OO RAS interacts with HP OO Central and polls it for operations to execute. Since the communication is from the RAS to Central, you need to open ports for inbound communication only in Central. Moreover, to achieve high availability of RASes, you simply add another RAS and point it to Central.

In addition, the RASes support a grouping mechanism, which enables you to correlate between a step in the flow and the type of the RASes that can execute this step. Therefore, the binding between the flow steps and the RASes is dynamic.

HP OO Content

HP OO provides a rich set of out-of-the-box operations and flows that enable you to author complex flows, orchestrating various services. The HP OO content is delivered as a set of granular content packs that you can download, deploy, and manage individually. These are the **Process Automation Libraries**.

In addition, HP OO provides wizards for generating additional content over other services such as Web Service Wizard.

HP OO provides Java and .NET SDK, to enable developing custom content and operations.

By using the HP OO content, you can build a rich set of process automation libraries.

HP OO Personas

HP OO personas represent the people responsible for performing particular tasks in the HP OO workflow. Icons are used in the HP OO documentation to identify the persona who performs a task.

Please note that when we suggest a persona for a task, this is a recommendation only, as practices may vary across different workplaces. For example, some workplaces may have an integrator, who is responsible for promoting all content packs, while we have assigned this task to the Op Admin. In other workplaces, a flow author may also be an action developer.

Major Personas

The following major personas play a key role in the HP OO processes.

Flow Author

	The Flow Author creates and debugs flows in HP OO Studio. The Flow Author leverages the HP OO out-of-the-box content, the content developed by Action Developers, and the utilities provided by SMEs.

Operational Administrator (Op Admin)

	The Op Admin is responsible for the day to day operation of HP OO. This includes configuration, maintenance, content pack promotion, and setting permissions for HP OO users. The Op Admin troubleshoots any failures in the production environment and either solves the problem or routes it to the System Administrator, to HP Support, or to the Flow Author (depending on the issue).

System Administrator (Sys Admin)

	The System Administrator is responsible for the HP OO hardware and software. The Sys Admin installs and patches HP OO (Central and RASes), and is responsible for the correct functioning of the application from the system perspective, dealing with things such as CPU, memory, and OS environment.

End User

	The End User triggers and monitors flows. The End User can access entitled HP OO flows directly through Central or indirectly through an embedded web UI in another application.

Minor Personas

The following minor personas play a supportive role in the HP OO processes. It is possible to use HP OO successfully without requiring the services of an action developer or SME. However, these personas can enrich the content of your flows and operations.

Action Developer

</>	The Action Developer is an expert in code development, who creates actions that can be used in flows. The Action Developer creates these actions mainly in Java and occasionally in .NET.

Subject Matter Expert (SME)

	The SME may be an expert in DBA, Linux, or a specific application. The SME creates focused automations (through scripts, executable files, and so on) for day-to-day operations in their area of expertise. These focused automations can be leveraged into the HP OO flow authoring.

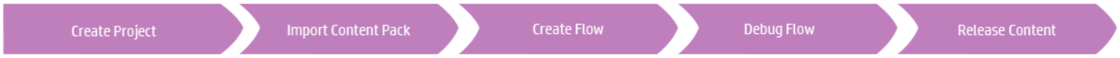
HP OO Processes

The major processes in HP OO are:

- Authoring flows
- Promoting content
- Running and monitoring flows

Authoring Flows

The authoring process is performed by the Flow Author.



1. **Create a new project** - The Flow Author creates a project to contain the flows, operations, folders, and configuration items for a business purpose.

2. **Import a content pack** - The Flow Author imports any content packs that are needed, in order to copy the relevant content into the project. If required, the Flow Author gets content from the Action Developer and SME.

   > **Note:** The first two steps do not have to be performed in this order. It is possible to import a content pack before creating the project.

3. **Create a flow** - The Flow Author puts together the operations, inputs, transitions, responses, and return steps that make up the flow.

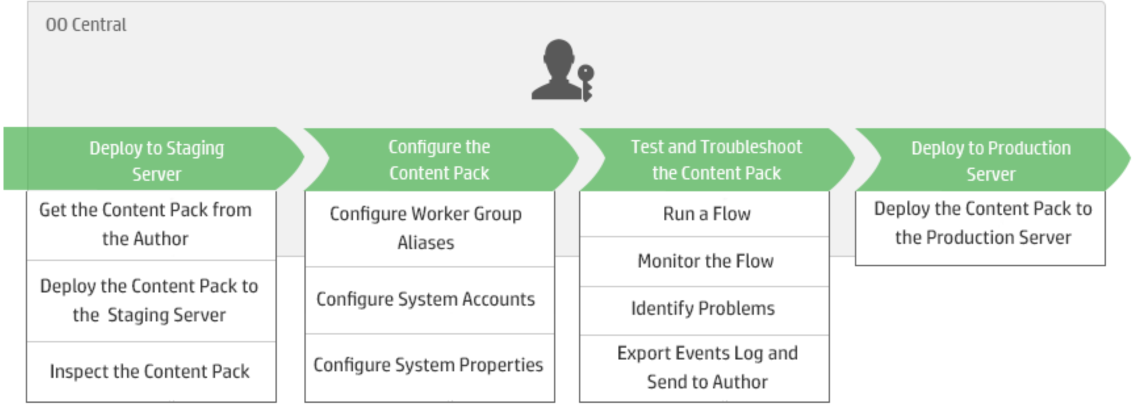
4. **Run and debug the flow** - The Flow Author validates the flow in the Debugger.

5. **Release the content, packaged into a content pack** - The Flow Author packages the project into a content pack, containing the flows, operations, actions, and configuration items, in order to deploy it to HP OO Central.

For more information, see the *HP OO Studio Authoring Guide*.

Promoting Content

The promotion process is performed by the Op Admin .


OO Central
Deploy to Staging Server
Get the Content Pack from the Author
Deploy the Content Pack to the Staging Server
Inspect the Content Pack
Configure the Content Pack
Configure Worker Group Aliases
Configure System Accounts
Configure System Properties
Test and Troubleshoot the Content Pack
Run a Flow
Monitor the Flow
Identify Problems
Export Events Log and Send to Author
Deploy to Production Server
Deploy the Content Pack to the Production Server

Step 1: Get the Content Pack From the Flow Author

1. The Flow Author creates a content pack in HP OO Studio.
2. The Flow Author saves the content pack to the file system.
3. The Flow Author shares the content pack with the Op Admin.

Step 2: Deploy to the Staging Server

1. The Op Admin receives the content pack from the flow author.
2. The Op Admin deploys the content pack on the Staging server.

Step 3: Configure the Content in the Content Pack

In this step, the Op Admin adjusts the content of the content pack to the environment.

1. If the content pack includes system accounts, the Op Admin assigns values to the system accounts in the content pack.
2. The Op Admin maps the aliases of the worker groups to actual worker groups.
3. The Op Admin sets the entitlements on the flows.

Step 4: Test and Troubleshoot the Content Pack

1. The Op Admin runs a flow from **Content Management** > **Flow Library**, and checks

whether it runs properly.

2. If a flow run fails, the Op Admin can drill down into a run to identify the problems.

3. If the flow contains problems that need to be fixed by the flow author, the Op Admin can save the flow events log, send the log to the author to review, and ask the author to fix the flow.

Step 5: Deploy to the Production Server

1. If required, the Op Admin adjust the configuration of the content pack, for the Production server.

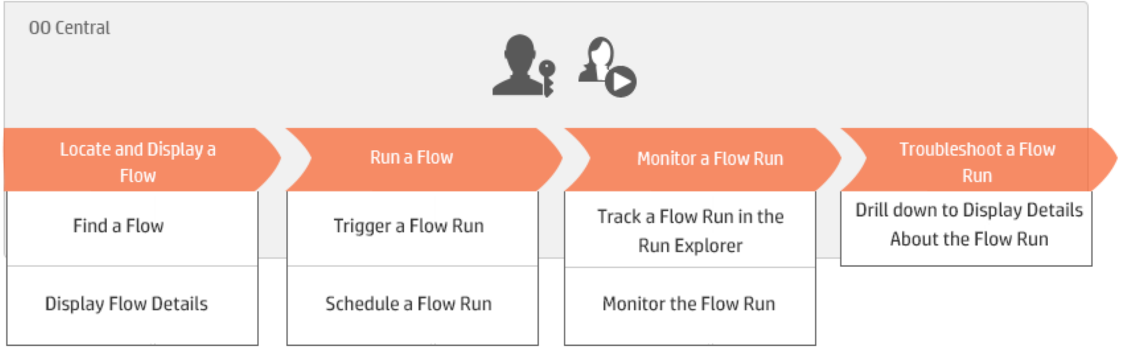
2. Finally, the Op Admin deploys the content pack to the Production server.

The deployment process between the Staging server and the Production server is similar to that between the Development and Staging servers.

For more information, see the *HP OO Central User Guide*.

Running and Monitoring Flows

The process of running and monitoring flows is performed by the Op Admin or the end user .



Step 1: Find the flow that you want to run

The Op Admin/End User browses to the flow, and views the flow's metadata, permissions. In addition you can view a visual map of the flow.

Step 2: Run the flow

The Op Admin/End User runs the flow.

Alternatively, the Op Admin/End User schedules the flow to run later.

Step 3: Monitor the flow run

The Op Admin/End User tracks the flow run in the Run Explorer.

If required, the Op Admin/End User performs actions on flow runs, such as pausing, resuming, and canceling flow runs.

Step 4: Troubleshoot the flow run

If required, the Op Admin investigates any problems with the run. The Op Admin either solves the problem or routes it to the relevant person.

For more information, see the *HP OO Central User Guide*.

HP OO Entities

HP OO Studio

HP OO Studio is a standalone authoring program used for creating, modifying, and testing flows. For more information, see "Functional Architecture" on page 10.

HP OO Central

HP OO Central is a server that is in charge of running flows. For more information, see "Functional Architecture" on page 10.

HP OO Content

HP OO contains an out-of-box content library with over 4,000 flows and operations, accelerator packs, and integrations. For more information, see "Functional Architecture" on page 10.

RAS

A RAS is a remote action server, containing a worker and a remote protocol for connecting with Central. For more information, see "Components" on page 25 and "Functional Architecture" on page 10.

Worker

Workers are the software components that are responsible for executing flows. A worker connects to Central to obtain tasks (flow execution messages) to process. For more information, see "RASes and Workers" in "Components" on page 25.

Worker Group

A worker group is a logical collection of workers. A worker may belong to more than one group simultaneously. You can define a worker group in Central. For more information, see "Worker Groups and Group Aliases" on page 30.

Group Alias

Group aliases let you separate between assigning an operation to a worker during authoring time and in the run time environment. In Studio, the author can define an operation to execute on a group alias rather than a worker group. In Central, the Op Admin maps the group alias to the actual worker group. For more information, see "Worker Groups and Group Aliases" on page 30.

Flow

A flow is the composition of steps and transitions together forming a set of actions that are linked by decision-making logic in order to automate tasks. For example, health checks, troubleshooting, or any other repetitive IT support tasks.

Flows are created in Studio and run in Central.

Subflow

A subflow is a flow that is used as a step within another flow. The flow that contains the subflow step is known as the parent flow. It is possible (and recommended) to debug the parent flow and subflow separately in HP OO Studio.

Flow Run

A flow run is a single execution of a flow in Central. Flow runs collect data, enabling you to analyze the performance of your IT system.

Content Pack

A content pack is a file containing operations, flows, actions (Java-based or .Net based), localization data, and configuration items. Content packs are deployed to the Central server and stored in the database.

A content pack can be created in Studio by an author, or it can be provided by HP or a third party.

HP provides a wide range of content packs with prepackaged flows, out-of-the-box integrations, and orchestration of multiple systems. The HP content packs are available from HPLN.

Role Permission

A permission is a predefined ability to perform a task. Central includes a set of permissions that can be assigned to roles. For example, the **Schedule** permission grants the ability to view and create flow run schedules.

Role

A role is a collection of permissions. For example, the **Flow Administrator** role may be assigned the **View Schedules** permission and the **Manage Schedules** permission.

User

Roles are assigned to users, to define the actions they have permission to perform in Central. For example, the user Joe Smith may be assigned the **Flow Administrator** role.

It is possible to configure different kinds of users:

- **LDAP users** log on to Central using their LDAP user name and password.
- **Internal users** log on to Central using the user name and password that was set up in Central.
- **LWSSO** users can enter the HP OO Central application directly, bypassing the HP OO Central logon screen, if they have logged onto another HP product web client that has LWSSO enabled.

When an internal user and an LDAP user with the same role are logged in, there is no difference between them.

Content Permission

Content permission is permission to view or run individual flows or the flows in a particular folder. A role, for example, **Administrator**, may be entitled to view and run all the flows in the system, while another role might be entitled to run certain flows only, and just have view permission for others.

All the users who have been assigned a specified role will be able to access the flows according to the content permissions assigned to their role.

Studio Project

A project is a directory in HP OO Studio, containing the flows and operations that a flow author is working on. In order to create a flow, a flow author must be working within a project.

Every project contains two folders:

- **Library** - holds all the flows and operations in that project.
- **Configuration** - holds other HP OO objects (filters, scriptlets, system properties, and so on) that you can use to process operation results, create reports, and facilitate the running of flows

When flow authors complete a project, they package it into a content pack for promotion to the Central server.

Action

An action is a software component invoked by the HP OO run time in order to achieve a certain task. An action can be developed using the Java or .NET programing language. Actions are packaged in the form of .JAR or .DLL files, and can be used as the basis for operations.

In addition to the actions developed by HP, an action can be developed at your own site or provided by a third party vendor aiming to extend the capabilities of HP OO.

Operation

An operation holds the inputs, outputs, responses, and other properties that are required to perform a certain action. Operations are available for the flow author to use as steps inside a flow. A rich set of out-of-the-box operations is available with HP OO Studio.

For example, one operation checks a web page to see whether it contains specific text, and another operation copies a file.

Step

Steps are the building blocks of a flow. A flow author creates a step by dragging an operation onto the authoring pane.

A step is an instance of the operation, and it inherits the inputs, flow variables, and other properties of the operation. A step can be modified without affecting the original operation.

Response

Responses are the possible outcomes of an operation. For example, a **Read Web Page** operation may have three possible responses:

- The web page can’t be found (failure)
- The page is there and the desired text is present (success)
- The page is there but the text isn’t present (partial success - needs another action)

Transition

A transition is the connection from an operation's response to one of the possible next steps.

Input

Inputs give the operation the data that it needs to act upon. For example, an operation to check a web page needs to know which page to check and what text to look for.

Inputs can be:

- Set to a specific value
- Obtained from information gathered by another step
- Entered by the person running the flow, at the start of the flow

Output

Outputs are data produced by operations—for example, success code, output string, error string, or failure message.

Primary Output

Primary output is the output used to populate the step's primary result. The primary output supplies a value to an input whose assignment is **Previous Step’s Result**.

Result

Results are output produced by a step in a flow. For each step, you decide which of the operation's outputs you want to retain.

Outputs are not automatically retained in the flow. If they were, this could affect performance, by slowing down the flow with unnecessary data.

Raw Result

The raw result is *all* of the operation’s return code, data output, and error strings.

Variable

Variables make data available in a flow run. There are two kinds of variable:

- **Global variables** are name/value pairs that come from system properties and selection lists, and are provided to a flow run at its start.
- **Flow variables** are available only for the flow within which they are defined. You can assign a value to a flow variable from:
  - A step's result - for example, a step with an operation to count hits can be set to store the result in a flow variable
  - An input value - for example, a step where a user needs to enter an IP address can be set to store the address as a flow variable
  - A scriptlet - for example, a scriptlet that evaluates data that is returned from a step can be set to store the data in a flow variable

Context

The context is a container that holds various values that can be exchanged with a step at various points. There are two kinds of context:

- **Local context** exists for the duration of the step.
- **Global context** exists for the duration of the flow.

You can pass values to and from the local or global context.

Output Fields

In a subflow (a flow within a flow), output fields are used to store a step's results so that the result data is accessible to operations, transitions, and prompts in the parent flow.

Scriptlet

Scriptlets (written in JavaScript) are optional parts of an operation that you can use to manipulate data from either the operation’s inputs or results for use in other parts of the operation or flow.

Soft Copied Operation

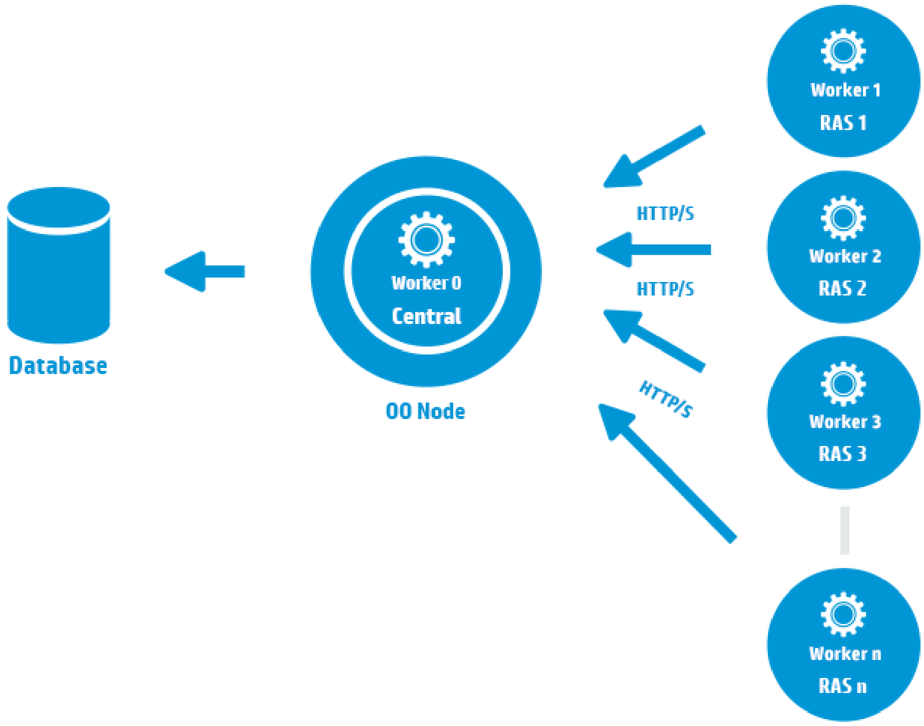
When you copy an operation that is linked to an action plugin jar file, the copied operation continues to reference the original operation. If the action plugin jar file is upgraded—for example if the name of the JAR or the class is changed— when you update the original operation to call the new version, the copied operations are all updated automatically. This is known as a **soft copy**.

Hard Copied Operation

In a **hard copy** of an operation, the copy is directly linked to the action plugin in the same way that the original operation was. When the action plugin jar file is updated—for example if the name of the JAR or the class is changed—this needs to be updated in all the hard copied operations.

System Architecture

HP Operations Orchestration 10.00 has modernized the technology stack and the main product design pattern.



- **Central** is the focal point for user interaction.
- **RAS and Worker** – The component that executes a flow is a “worker”. To scale out and withstand more concurrent flow executions and a higher throughput, you can add more RASes or install more Centrals. Each RAS contains a worker and a remote protocol for communicating with Central.
- The flow step executions (tasks) reside in a queue, and are executed in a first-come-first-served order. Central manages queues and distributes the tasks to the workers.

For more information on each of the components, see "Components" below.

Components

This section describes how the components of the architecture work.

Central

Central is a web-based application used to:

- Deploy content onto the Production server, from Studio or from a previous Central environment
- Trigger or schedule flows
- Monitor flow execution progress
- View execution history
- Extract and analyze data resulting from flow runs
- Perform administration tasks:
  - System configuration – set system-wide configuration. For example, setting the periodical time interval when workers connect to Central for their configuration changes, and configuring LDAP authentication.
  - Configuration of content packs – system accounts, users, and so on.
  - Manage worker groups – see "Worker Groups and Group Aliases" on page 30
  - Manage worker properties – see "Worker" below

You can cluster OO by adding more HP OO Central nodes with a load balancer in front of them. There is no need for external clustering software, a clustered operating system, or a shared file system. See "Typical HP OO Deployments " on page 31 for more details.

Since the Central server connects to the database, this affects the deployment topology—it can only be deployed in a network environment that allows access to the database.

RASes and Workers

Workers are responsible for executing flows. The worker pulls tasks (step executions) from Central and processes actions and calculates the next step to execute before returning the results to the Central. Workers reside on Central and on each RAS.

In HP OO 10.00, each Central node includes a single worker by default. You can add more RASes, or install additional Central nodes if required, to scale out or to create a cluster. Having multiple RASes enables you to withstand more concurrent flow executions and reach a higher throughput.

A RAS contains a single worker and a remote protocol for communicating with Central.

Unlike a RAS in the previous version of HP Operations Orchestration, a worker in HP OO 10.00 process all types of steps, and not only remote actions.

The communication direction of a RAS in HP OO 10.00 is reversed from that of a RAS in previous versions. Previously, the Central server connected to the RAS server. In this version, the worker initiates the connection to the Central server, working like a client.

A RAS connects to the Central server over HTTP(S) to get its configuration. This is done periodically. The information in this configuration includes the group that a specific worker belongs to. A worker may belong to many groups simultaneously. A worker group is an enhancement over the “Override RAS” feature in the previous version. See "Worker Groups and Group Aliases" on page 30 for more details.

A RAS connects to Central to obtain tasks (flow execution messages) to process. Note that even though, conceptually, Central “sends” a flow to a worker in a RAS to be processed, in actuality, the worker initiates the request and “pulls” tasks to process. This makes the RAS much more firewall-friendly.

There are two types of processing that a worker performs:

- Action execution – this is similar to the IAction execution in a RAS in previous versions—running the actual action of a step. For example, running a ping operation.

  > **Note:** Built-in operations (or plugins as they were previously called) and scriptlets are now also executed in the worker. Built-in operations were converted to actions and are part of the base content pack of HP OO 10.00.

- Flow execution processing – in previous versions this processing was done in the Central server. In this version, the processing was moved to the worker. Flow step execution processing now includes all the pre-action and post action tasks and calculating further navigation in the flow as well.

All workers can perform both action execution and flow execution processing. However, sometimes, usually when the RAS containing the worker is behind a firewall, it is preferable in terms of performance to process only the action execution on the worker on the other side of the firewall, and everything else in other workers. This is possible by assigning workers to groups.

Modernized Platform

The new platform is designed to reduce the total cost of ownership. It supports scaling out rather than up, and offers improved enterprise readiness.

Inherent Scaling by Design

The execution of flows and management of the environment has been separated into two autonomous subsystems, each of which can function without the other.

This results in:

- Greater scalability – you can scale out by adding components online with zero downtime
- Support for advanced installation topologies, such as multiple reverse proxies
- The possibility of placing workers behind a firewall
- More effective parallel/multi-instance executions

Simplified Topology and System Requirements

Each component has a single high level function:

- Central – the front end and orchestration (task distribution)
- RAS containing workers – execution of flows

This simple topology means that:

- Everything is stateless
- Automatic load balancing in workers – no load balancing or reverse proxy software is required.
- No clustering software is required (such as Terracotta, Windows Cluster, and so on). See "Typical HP OO Deployments " on page 31.
- Offline authoring – Studio does not require any external components to function
- No shared file system
- Simplified APIs – RESTful APIs only
- Zero downtime when deploying new content – you can use the new content immediately without having to restart.

Enterprise Readiness

- Security improvements – the workers initiate the connection to the Central server. This means that workers can be placed on a RAS behind a firewall without opening a hole in it, using standard HTTPS to communicate. For more information about workers, see "Components" on page 25.
- High availability improvements – fewer components are required, and they can be deployed online with zero downtime. There is no single point of failure by design. See "Typical HP OO Deployments " on page 31.
- Improved runtime topology management – instead of deploying a load balancer, it is possible to create worker groups, when you need more than one worker in an environment. We've also separated the authoring time group definition from the runtime definition with aliases. For more information, see "Worker Groups and Group Aliases" on page 30.

Comparing Architecture between HP OO 9.x and HP OO 10.00

While there are some functions in common between Central in HP OO 10.00 and Central in HP OO 9.x, and between the HP OO 10.00 worker and the HP OO 9.x RAS, there are also some differences. The table identifies the components that are responsible for each function.

Function	**9.x component**	**10.00 component**
Source control	Central	Integration with external source control management systems
Deploy content pack/flow	Shared repository	Central
Trigger flow execution	Central	Central
Execute flow	Central	Worker
Execute built-in operation	Central	Worker
Execute IAction/@Action	RAS	Worker
Execute script/rule/filter	Central	Worker
Schedule flows	Central	Central
Reports	Central	Central
Monitor execution	Central	Central
Administration	Central	Central
Orchestrate flow execution	Central	Central
Interim products (paused executions, parallel branches results, multi-instance results)	In memory	DB
Location of execution logs	Distributed in RASes	Central (monitoring feed)

Worker Groups and Group Aliases

Worker Groups

Many deployments can benefit from having more than a single worker in a specific environment. For example, this could be helpful if you are managing a remote data center in which you need workers to be able to withstand the action execution load, or simply for high availability of the workers in that data center. In previous versions, a load balancer would have been required to balance the load between two RASes, which Central would know as a single logical RAS.

In HP Operations Orchestration 10.00, we have addressed the issue of workers having high availability and scalability, and there is no longer a need for a load balancer as part of the topology between Central and the workers. Instead, we've introduced the notion of a "*worker group*". A group is a logical collection of workers and a worker may belong to more than one group simultaneously. Having a worker group helps prevent bottlenecks, because a step can run on any of the workers in the group.

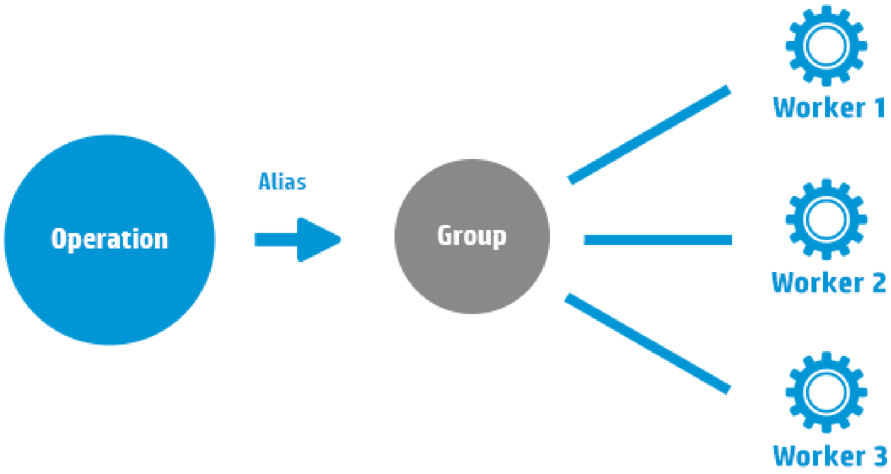
Groups can be created and workers assigned to groups in Central. For more information, see the *HP OO Central User Guide*.

Group Aliases

In previous versions, the author of a flow was exposed to the runtime topology when a RAS was defined for a specific step in the flow. In this situation, you could not change the host name of the runtime RAS without changing it in all the flows that used it, or it had to be "overridden" at runtime.

In HP Operations Orchestration 10.00, in addition to worker groups, there is another level of flexibility in the assignment of an operation to a worker (or workers). Group aliases let you separate between assigning an operation to a worker during authoring time and in the runtime environment. At authoring time, an operation can be defined to execute on a group alias rather than a group. The alias is then mapped to a group in the runtime environment at the Central level. As a fallback, if the group alias is identical to the group name, it is mapped automatically to that group. Optionally, at triggering time, it is possible to override the group alias and map the operation to a different worker group.

Using this functionality, the HP OO administrator can let the author use an alias, for example, **WorkerAlias1**. In Central, the administrator maps the alias to the actual worker group. When the content pack is imported to Central, there is no need for the administrator to dive into the flows and modify the worker assignment manually.

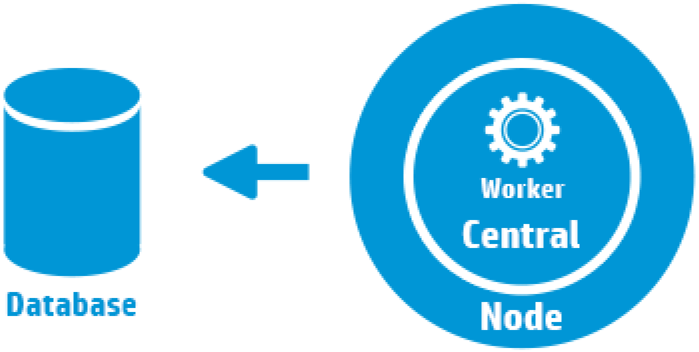


Typical HP OO Deployments

This section provides an overview of some of the options for deployment of HP OO. For more detailed information about deployments, see the *HP OO Installation Guide*.

Simple Deployment

In the following schematic view, you can see that in a typical deployment, all components are deployed in a single web application within the internal Tomcat server that is bundled in the HP Operations Orchestration installation.

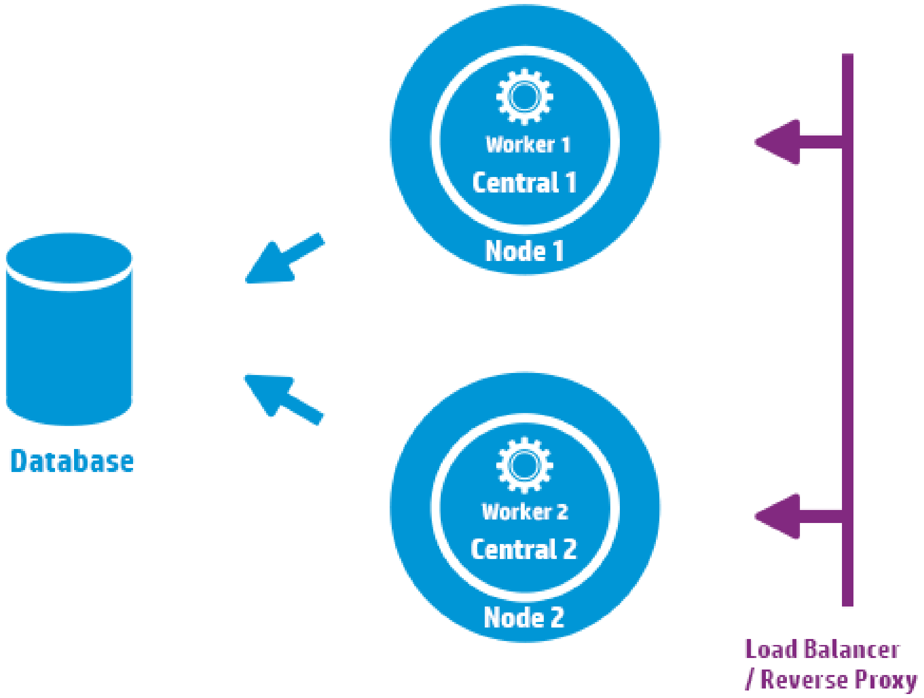


Obviously, this deployment does not provide high availability, because all components are located on the same machine.

Note: This topology also works as an embedded version of HP OO. The web application can be hosted in any application server that supports the servlet 2.5 standard.

Simple Cluster

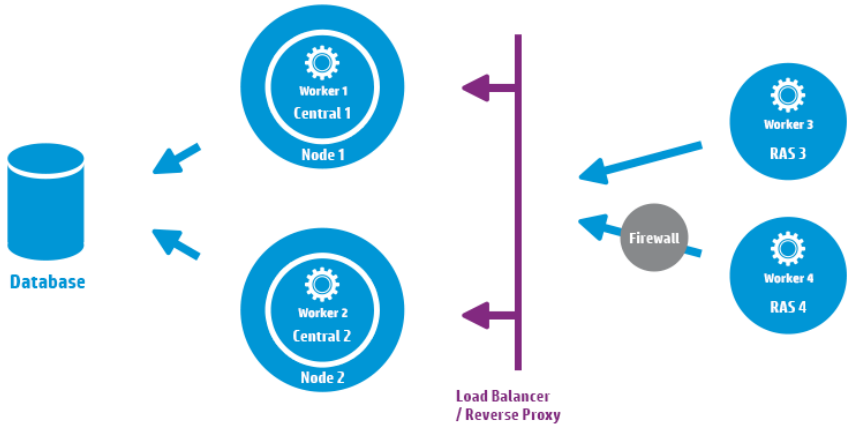
In the following schematic view, you can see a simple cluster comprised of two installations of HP Operations Orchestration pointing to the same database schema.



Unlike in previous versions, there is no need for external clustering software, nor is there a requirement for a shared file system. The purpose of the load balancer/reverse proxy is to expose a single URL to the clients of the system; however, the load balancer/reverse proxy is optional and not imperative to the cluster. Exposing a single URL can also be done with DNS load balancing instead.

Deploying a RAS Behind a Firewall

In the following schematic view, you can see a more complex environment. The main feature demonstrated here is the RAS behind a firewall—RAS 4.



Note that in this topology, there is also a standalone RAS—RAS 3, containing worker 3. RAS 3 is a RAS just like RAS 1 and RAS 2. It is simply installed as a standalone process.

One way of scaling HP Operations Orchestration 10.00 is by adding more RASes. Adding workers increases the throughput capabilities of processing flows in the system.

The above topology is more flexible, because it incorporates both the high availability features for Central and the execution subsystem (all workers) and also allows for RASes to reside behind the firewall.

Op Admin can enable or disable the workers after the installation, as needed.