

# Installing and Configuring VMware vCenter Orchestrator Plug-In SDK

vCenter Orchestrator 4.2

This document supports the version of each product listed and supports all subsequent versions until the document is replaced by a new edition. To check for more recent editions of this document, see <http://www.vmware.com/support/pubs>.

EN-000685-00

**vmware**<sup>®</sup>

You can find the most up-to-date technical documentation on the VMware Web site at:

<http://www.vmware.com/support/>

The VMware Web site also provides the latest product updates.

If you have comments about this documentation, submit your feedback to:

[docfeedback@vmware.com](mailto:docfeedback@vmware.com)

Copyright © 2011 VMware, Inc. All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. VMware products are covered by one or more patents listed at <http://www.vmware.com/go/patents>.

VMware is a registered trademark or trademark of VMware, Inc. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies.

**VMware, Inc.**  
3401 Hillview Ave.  
Palo Alto, CA 94304  
[www.vmware.com](http://www.vmware.com)

# Contents

Installing and Configuring VMware vCenter Orchestrator Plug-In SDK	5
<b>1 vCenter Orchestrator Plug-In Development Environment</b>	<b>7</b>
<b>2 Setting Up a Development Environment for Orchestrator Plug-Ins</b>	<b>9</b>
System Requirements for vCO Plug-In SDK	9
Configure the Proxy Settings of Eclipse	9
Install the vCO Plug-In SDK in Eclipse	10
Create a Project for Developing a vCO Plug-In	10
Enable Server-Side Logging for Plug-Ins	11
Install vLogBrowser	12
<b>3 Running the Sample Plug-Ins</b>	<b>13</b>
Run the Sample Solar System and Hello World Plug-Ins	13
Running the Sample Solar System SOAP and REST Plug-Ins	14
Install Tomcat 6.0 Server	14
Create a vCO Web Service Project from a Sample vCO Web Service	15
Run the Sample SOAP and REST Services on the Tomcat Server	15
Configure the Connection to the Tomcat Server	15
<b>Index</b>	<b>17</b>



# Installing and Configuring VMware vCenter Orchestrator Plug-In SDK

---

*Installing and Configuring VMware vCenter Orchestrator Plug-in SDK* provides information about installing the VMware® vCenter Orchestrator (vCO) plug-in SDK in the Eclipse Integrated Development Environment (IDE) and configuring a development environment for vCenter Orchestrator plug-ins.

## Intended Audience

This information is intended for developers who want to set up a development environment for creating plug-ins for VMware vCenter Orchestrator.



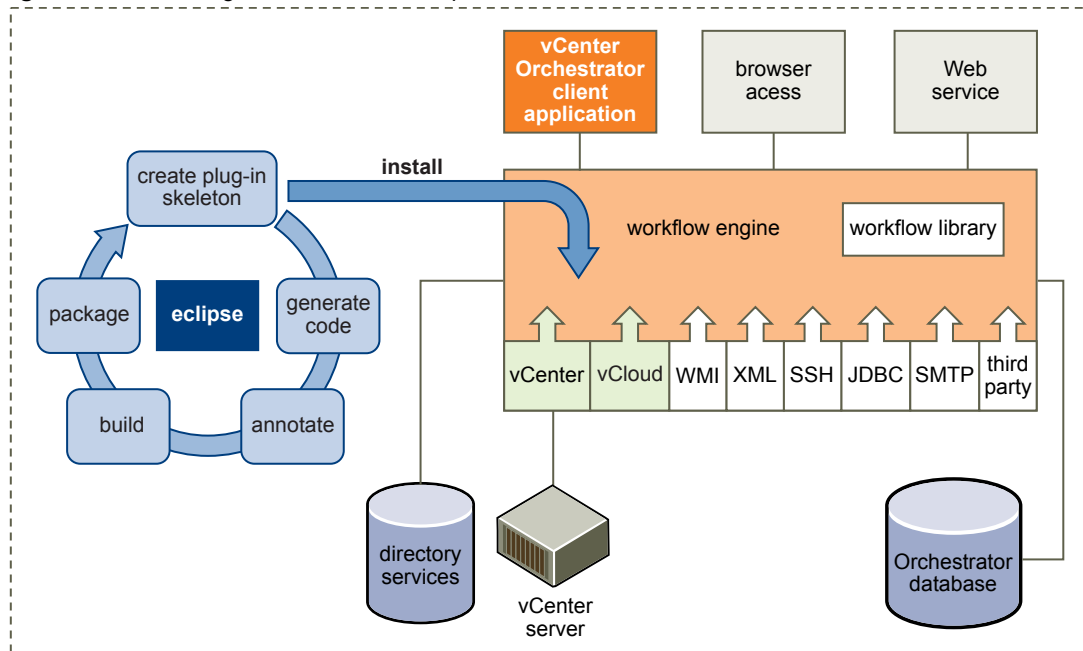
# vCenter Orchestrator Plug-In Development Environment

# 1

The vCenter Orchestrator Plug-in SDK integrates vCO plug-in examples and APIs into the Eclipse for Java Developers IDE.

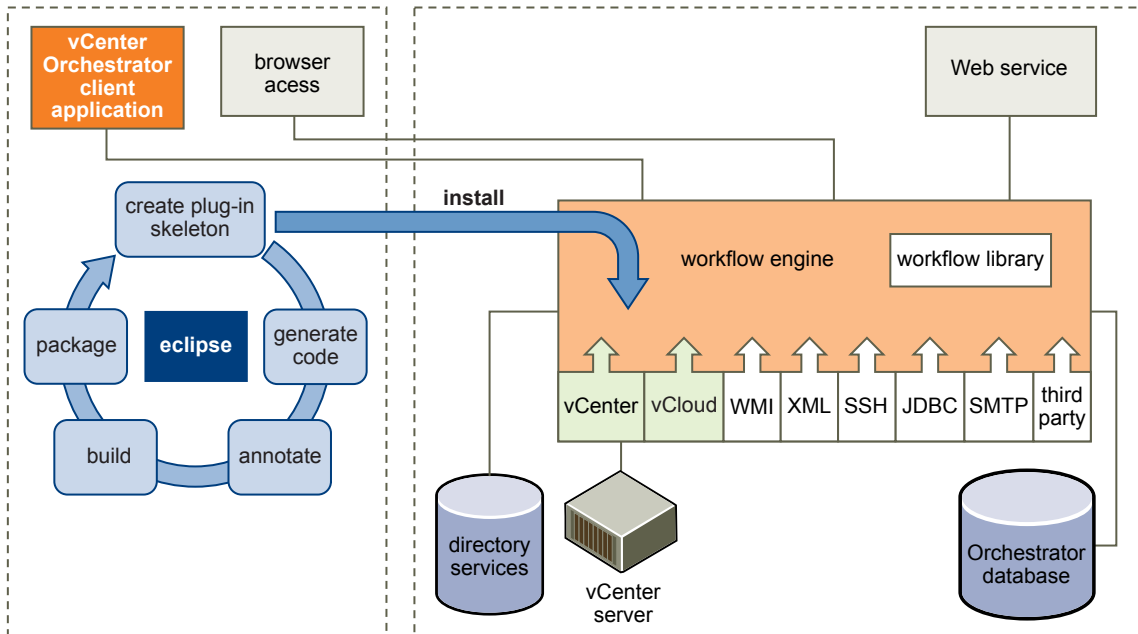
You can install the vCenter Orchestrator Plug-in SDK on the same system where vCenter Orchestrator Server runs and perform all plug-in development activities on that system.

**Figure 1-1.** vCO Plug-in SDK local development mode



You can deploy and run your vCO plug-in development environment on a different system than the Orchestrator server and perform all plug-in development activities remotely.

**Figure 1-2.** vCO Plug-in SDK remote development mode



# Setting Up a Development Environment for Orchestrator Plug-Ins

## 2

You can set up a development environment for Orchestrator plug-ins in Eclipse for Java Developers. To set up a development environment for Orchestrator plug-ins, you must install the vCenter Orchestrator Plug-in SDK in Eclipse and create projects for developing Orchestrator plug-ins. You can also install vLogBrowser to view logs for debug analysis.

This chapter includes the following topics:

- [“System Requirements for vCO Plug-In SDK,”](#) on page 9
- [“Configure the Proxy Settings of Eclipse,”](#) on page 9
- [“Install the vCO Plug-In SDK in Eclipse,”](#) on page 10
- [“Create a Project for Developing a vCO Plug-In,”](#) on page 10
- [“Enable Server-Side Logging for Plug-Ins,”](#) on page 11
- [“Install vLogBrowser,”](#) on page 12

## System Requirements for vCO Plug-In SDK

To be able to develop vCenter Orchestrator plug-ins using the vCenter Orchestrator Plug-In SDK, your system must meet certain requirements.

vCO Plug-in SDK runs on the following platforms.

- Windows Server 2003 64bit
- Eclipse Helios 3.6.x 32bit for Windows
- JDK 1.6 32bit

Ensure that the following software is installed on your system before you install vCO Plug-in SDK.

- Java JDK 1.6.0 32bit for Microsoft Windows
- Eclipse Helios 3.6.x 32bit for Microsoft Windows. You must have at least 100MB free disk space in your Eclipse workspace.

## Configure the Proxy Settings of Eclipse

If the system where you want to install and run the vCO Plug-in SDK uses a network proxy, you must configure the proxy settings for Eclipse before installing the vCO Plug-in SDK. During the install process, vCO Plug-in SDK downloads components from the Eclipse site.

### Procedure

- 1 In Eclipse, select **Window > Preferences > General > Network Connections**.

- 2 From the **Active Provider** drop-down menu, select **Manual**.
- 3 In the Proxy entries pane, select a protocol and click **Edit**.
- 4 Enter the host and port for the proxy as well as the required authentication credentials and click **OK**.
- 5 (Optional) In the Proxy bypass pane, specify hosts for which you do not want to use a proxy.
  - a Click **Add Host**.
  - b In the Proxy bypass hosts dialog box, enter the host name for which you do not want to use a proxy and click **OK**.

## Install the vCO Plug-In SDK in Eclipse

You install the vCenter Orchestrator Plug-in SDK from an Eclipse update site.

### Prerequisites

Configure the proxy settings for Eclipse if the system where you want to install the vCO Plug-in SDK is connected to a network proxy. For details, see [“Configure the Proxy Settings of Eclipse,”](#) on page 9.

### Procedure

- 1 Start Eclipse and select a workspace.  
You should use a new workspace to build Orchestrator plug-ins.
- 2 If you have more than one version of Java installed you must select JRE 6 as a default JRE.
  - a Select **Window > Preferences**.
  - b Select **Java > Installed JREs**.
  - c Select **jre6** as a default JRE and click **OK**.
- 3 In the Eclipse workspace, click **Help > Install New Software**.
- 4 Click **Add** that is located next to the **Work with** drop-down menu.
- 5 In the Add Repository dialog box, select **Archive**.
- 6 Browse to the location of the downloaded ZIP file of the vCO Plug-in SDK and click **Open**.  
The location of the SDK ZIP file appears in the **Location** text box.
- 7 Click **OK** to add the location of the vCO plug-in SDK installation.  
VMware Workbench vCenter Orchestrator Starter Kit appears in the list of software to be installed.
- 8 Select the **VMware Workbench vCenter Orchestrator Starter Kit** check box and click **Next**.
- 9 Review the items to install and click **Next**.
- 10 Review and accept the license agreements and click **Finish**.

## Create a Project for Developing a vCO Plug-In

You can create projects in Eclipse for developing vCO plug-ins. You can create projects that contain empty skeletons for a vCO plug-in or create projects from sample vCO plug-ins.

### Prerequisites

Install vCO Plug-in SDK in Eclipse.

**Procedure**

- 1 Right-click the **Package Explorer** view and select **New VMware Project/File > Development Kit Projects**.
- 2 Expand **vCenter Orchestrator Plug-in Development**.
- 3 Expand a node from the available ones.

Option	Description
<b>Create new vCenter Orchestrator Plug-in project</b>	Provides options for creating a project for developing a new vCO plug-in from an empty project or from a skeleton plug-in structure.
<b>Create vCenter Orchestrator plug-in project from samples</b>	Provides options for creating a new project for a vCO plug-in from a sample vCO plug-in. The sample vCO plug-ins are fully-functional and do not require modifications to run.

- 4 Under the expanded node, select an option for creating a project for a vCO plug-in and click **Next**.
- 5 Enter a name for your project and click **Finish**.  
To name your project, you can use lowercase alphabet, digits from 0 to 9, the underscore (\_), and the hyphen (-).

The structure of the new vCO plug-in project appears in the Package Explorer. If you have selected the **Create an empty vCenter Orchestrator plug-in project** option, the created project contains only the required libraries and the `vso.xml`. If you have selected the **Create a skeleton vCenter Orchestrator plug-in project** option, the created project contains the skeleton of a vCO plug-in where you can implement your own functionality. If you have selected one of the sample plug-ins, the created project contains a fully-functional sample vCO plug-in that you can install and run on the vCO server.

## Enable Server-Side Logging for Plug-Ins

You can enable server-side logging for every plug-in by adding the necessary configuration entries in the `log4j.xml` configuration file of the vCO server.

To enable server-side logging for plug-ins, you must add the necessary configuration entries in the `log4j.xml` file of the vCO server separately for every package of a plug-in.

**Procedure**

- 1 Open the `log4j.xml` file located at `<vcenter_orchestrator_installation_directory>\app-server\server\vmo\conf`.

For example, if vCenter Orchestrator is installed as part of vSphere, the path to the `log4j.xml` file might be `C:\Program Files\VMware\Infrastructure\Orchestrator\app-server\server\vmo\conf`.

- 2 Insert the following section for every package of the plug-in.

```
<category additivity="true" name="package name">
  <priority value="DEBUG"/>
</category>
```

## Example: Example of Server Side Logging Configuration of a vCO Plug-In

The following example configuration entries enable server-side logging for packages `com.vmware.solarsystem` and `com.vmware.orchestrator.api.sample.solarsystem`.

```
<category additivity="true" name="com.vmware.orchestrator.api.sample.solarsystem">
  <priority value="DEBUG"/>
</category>

<category additivity="true" name="com.vmware.solarsystem">
  <priority value="DEBUG"/>
</category>
```

## Install vLogBrowser

You can install vLogBrowser in Eclipse and use it to analyze log files for debugging purposes.

### Procedure

- 1 In the Eclipse workspace, click **Help > Install New Software**.
- 2 Click **Add** that is located next to the **Work with** drop-down menu.
- 3 In the Add Repository dialog box, select **Archive**.
- 4 Browse to the location where you downloaded the ZIP file of vLogBrowser and click **Open**.
- 5 Click **OK** to add the location of the vLogBrowser installation.  
VMware Workbench vLogBrowser appears in the list of software to be installed.
- 6 Select the **VMware Workbench vLogBrowser** check box and click **Next**.
- 7 Review the components to install and click **Next**.
- 8 Review and accept the license agreements and click **Finish**.

## Running the Sample Plug-Ins

---

You can create Eclipse projects for vCO plug-ins out of sample plug-ins. The projects for sample plug-ins contain fully-functional vCO plug-ins that you can install on the Orchestrator server to explore how the components of a plug-in work.

This chapter includes the following topics:

- [“Run the Sample Solar System and Hello World Plug-Ins,”](#) on page 13
- [“Running the Sample Solar System SOAP and REST Plug-Ins,”](#) on page 14

### Run the Sample Solar System and Hello World Plug-Ins

You can install the sample Solar System and Hello World plug-ins on the Orchestrator server and explore their functionality.

#### Prerequisites

- Install the vCO Plug-in SDK in Eclipse. See [“Install the vCO Plug-In SDK in Eclipse,”](#) on page 10.
- Create a project for a vCO plug-in from the sample Solar System or Hello World plug-ins. See [“Create a Project for Developing a vCO Plug-In,”](#) on page 10.

#### Procedure

- 1 In the newly created project, go to the `dist` directory, which contains the `.dar` file of the plug-in.  
For example, the `.dar` file of the sample Solar System plug-in is `o11nplugin-solarsystem.dar`.
- 2 Install the `.dar` file of the plug-in on the Orchestrator server.
  - a Log in to the Orchestrator configuration interface.
  - b Click the **Plug-ins** tab.
  - c Click the magnifying glass icon under Install new plug-in.
  - d Browse to locate the `.dar` file, and click **Open**.
  - e Click **Upload and install**.

## Running the Sample Solar System SOAP and REST Plug-Ins

Before you run the sample Solar System SOAP and REST plug-ins, you must ensure that the sample SOAP and REST services are running on a Tomcat 6.0 server that runs on the same system as the vCO Plug-in SDK.

To run the sample Solar System SOAP and REST plug-ins on the Orchestrator server, you must perform the following steps.

- 1 Install a Tomcat 6.0 server on the same system where the vCO Plug-in SDK is installed. See [“Install Tomcat 6.0 Server,”](#) on page 14.
- 2 Create a project for a sample Solar System SOAP or a REST plug-in. See [“Create a Project for Developing a vCO Plug-In,”](#) on page 10.
- 3 Create a project for a sample SOAP or REST service. See [“Create a vCO Web Service Project from a Sample vCO Web Service,”](#) on page 15.
- 4 Run the sample SOAP or REST service in the Tomcat 6.0 server. See [“Run the Sample SOAP and REST Services on the Tomcat Server,”](#) on page 15.
- 5 Install the sample Solar System SOAP or REST plug-in on the Orchestrator server. The .dar files of the sample plug-ins are located in the dist directory of the plug-in projects in Eclipse. For more information about how to install plug-ins on the Orchestrator server, see [“Run the Sample Solar System and Hello World Plug-Ins,”](#) on page 13.
- 6 Configure the connection settings for the sample SOAP or REST plug-in with the Tomcat 6.0 server. See [“Configure the Connection to the Tomcat Server,”](#) on page 15.

### Install Tomcat 6.0 Server

To run the sample Solar System SOAP and REST plug-ins, you need a Tomcat 6.0 server running where you must deploy and run the sample SOAP and REST services that the plug-ins use.

---

**NOTE** You must install the Tomcat 6.0 server on the same system as the vCO Plug-in SDK.

---

#### Procedure

- 1 In Eclipse, select **Window > Show view > Other**.
- 2 Expand the **Servers** node.
- 3 Select **Servers**, and click **OK**.  
The **Servers** view appears at the bottom of your Eclipse workspace.
- 4 Select the **Servers** view.
- 5 Right-click the empty pane in the **Server** view and select **New > Server**.
- 6 Expand the **Apache** node, select **Tomcat 6.0**, and click **Next**.
- 7 Select the **Workbench default JRE** from the **JRE** drop-down menu.  
Do not select the external JRE.
- 8 Click **Download and Install**, accept the license agreement, and click **Finish**.
- 9 Specify the folder where you want to install the Tomcat 6.0 server and click **OK**.
- 10 After the installation completes, click **Finish**.
- 11 Right-click on the newly created server and select **Start**.

- 12 (Optional) In case of port conflicts, reconfigure the ports for the Tomcat server.
  - a Select the **Servers** view.
  - b Double-click the **Tomcat 6.0 Server** node.  
A Tomcat 6.0 Server editor opens.
  - c In the Ports pane, modify the ports for the server.

## Create a vCO Web Service Project from a Sample vCO Web Service

If you want to use the SOAP and REST functionality of the sample Solar System SOAP and REST plug-ins, you need the sample SOAP and REST services.

### Prerequisites

- Install the vCO Plug-in SDK on Eclipse.
- Install Tomcat 6.0 server on the same system where the vCO Plug-in SDK is installed. For more information about installing a Tomcat server, see [“Install Tomcat 6.0 Server,”](#) on page 14.

### Procedure

- 1 Right-click the **Package Explorer** view and select **New VMware Project/File > Development Kit Projects**.
- 2 Expand **vCenter Orchestrator Plug-in Development**.
- 3 Expand **Create vCenter Orchestrator service or client project from examples**
- 4 Select a sample REST or a SOAP service and click **Next**.
- 5 Enter a name for your project and click **Finish**.

To name your project, you can use lowercase alphabet, digits from 0 to 9, the underscore (\_), and the hyphen (-).

## Run the Sample SOAP and REST Services on the Tomcat Server

You can use the sample SOAP and REST plug-ins only after you run the sample SOAP and REST services in the Tomcat 6.0 server.

### Procedure

- 1 Right-click the project of the sample SOAP or REST service, and select **Run As > Run on Server**.
- 2 Select the **Always use this server when running this project** check box.
- 3 Click **Finish**.
- 4 If a Web interface window appears in the editor panel, dismiss it.

## Configure the Connection to the Tomcat Server

To use the SOAP and REST functionality of the sample SOAP and REST plug-ins, you must configure the connection settings of the plug-ins to the Tomcat 6.0 where you deployed the sample SOAP and REST services.

### Prerequisites

- Install the sample Solar System SOAP or REST plug-ins on the vCO server.
- Ensure that the sample SOAP or REST services are running on the Tomcat 6.0 server.

## Procedure

- 1 In the vCenter Orchestrator Configuration tool, select the tab for the sample Solar System SOAP or REST plug-in.
- 2 In the corresponding text box, enter the host name or the IP address of the Tomcat server where the sample SOAP or REST services are deployed.
  - **Enter SOAP server hostname or IP** for the sample SOAP plug-in.
  - **Enter REST server hostname or IP** for the sample REST plug-in.
- 3 In the corresponding text box, enter the HTTP port where the Tomcat server runs.
  - **Enter SOAP Server Port** for the sample SOAP plug-in.
  - **Enter REST Server Port** for the sample REST plug-in.

If you have reconfigured the default HTTP port for the Tomcat server, you must enter the port value that you specified.

# Index

## C

Configure proxy settings **9**

Configure connection to Tomcat **15**

## I

install Eclipse module **10**

install sample plug-ins **10**

Install sample plug-ins **13**

Install Tomcat Server **14**

Installing vCO Plug-in SDK **5**

## P

Plug-In development environment **7**

## R

run services **15**

## S

Sample REST plug-in **14**

Sample SOAP plug-in **14**

Sample Web service **15**

server configuration **11**

system requirements **9**

## V

vCO Plug-In SDK **5, 7**

vCO sample plug-in **13, 15**

vLogBrowser **12**

