Prime Fulfillment XDE SDK

This appendix gives a brief overview of the Cisco Prime Fulfillment XDE SDK (XDE SDK) and explains how to use it. The XDE SDK makes it possible to extend the Prime Fulfillment feature set to customize it to meet your needs.

This appendix contains the following sections:

Introduction

The Prime Fulfillment XDE SDK is an Eclipse-based plug-in that enables you to extend and test the existing provisioning functionality supplied in Prime Fulfillment.

The XDE SDK provides a complete integrated development environment for

- building and viewing the provisioning logic
- building and viewing the device templates
- graphically debugging provisioning logic
- running and defining regression tests

For more information about the XDE SDK, go to

http://developer.cisco.com/web/xde/

System Recommendations

General system requirements for running the XDE SDK are synonymous with those for Eclipse. See eclipse.org.

Specific requirements:

- Operation system support:
  - Windows
  - Linux
  - OS X (Mac)
- Memory requirements: 4 GB
- Disc space: 1 GB
- Java: Use the latest version of JRE or JDK

  Java information for Eclipse: http://www.eclipse.org/downloads/moreinfo/jre.php

Installing and Setting Up the Prime Fulfillment XDE SDK

To make the Prime Fulfillment XDE SDK work as intended, the following steps are recommended in the installation process:

Step 1   Set up your environment.

Before installing Eclipse, you need to make sure that your environment will support Eclipse and the XDE package. This includes:

- Checking the system recommendations: System Recommendations, page D-1.
- Ensuring that a supported version of Java is installed For this, use the latest version of JRE or JDK.

Step 2   Install the latest Eclipse version.

Go to eclipse.org/downloads/ and look for Eclipse for RCP and RAP developers. This page also contains a link to the Eclipse installation documentation.

Once you have Java and Eclipse installed, it is recommended that you ensure adequate memory is allocated to Eclipse (see System Recommendations).

You can do this by editing the **eclipse.ini** file

<eclipse-install>/eclipse/eclipse.ini

and changing the last two lines to:

```
-Xms512m
-Xmx1024m
```

Step 3   Install the XDE SDK Eclipse plug-in.

Go to http://wikicentral.cisco.com/display/PROJECT/Installing+the+XDE+Plugin.

Overview of Eclipse plug-ins:

Go to http://wikicentral.cisco.com/display/NMTGA/Eclipse+plugins.

Step 4   Go to the Prime server:

<isc-install>/resources/xde/packages/std

and copy the provisioning logic files onto the server on which you are running Eclipse/XDE.

Step 5   Import the provisioning logic files into the XDE SDK.

For information about how to import packages into the Eclipse SDK, go to

http://www.eclipse.org/downloads/moreinfo/jre.php

Process Overview

The Prime Fulfillment functionality has four main parts:

1. **Query phase:** The devices involved in the fulfillment operation are queried for their current configuration. This determines what configuration is already provisioned on the network.

    XDE parses the device profiles and configurations into a set of XML files that can be processed by the provisioning logic.

2. **Logic phase:** The provisioning logic runs within XDE. XDE processes the existing device configurations and requested service design to create device configuration.

    The required device configuration templates are run to generate the actual device configuration.

3. **Provisioning phase:** XDE creates the minimal required set of commands to provision the service and pushes the configuration to the devices.

4. **Audit:** The configuration pushed to the devices are audited to make sure it has been applied correctly.

Creating or Modifying Provisioning Logic

It is important to understand the difference between the two basic ways in which the provisioning logic can be changed. It will help you make informed choices, which can reduce risk of introducing errors in the logic.

The provisioning logic can be modified in the following two ways:

*   Changing the existing provisioning logic

*   Using an existing extension point (recommended)

Note   Cisco Prime Fulfillment 6.2 does not support changing the existing provisioning logic.

Note   It's important to note that direct editing of the provisioning logic may cause Prime provisioning to fail. This may also put the system into a state where upgrades can no longer be obtains for the product. If in doubt, please contact Cisco.

Changing Existing Provisioning Logic

For information on how to edit workflows and use the XDE SDK, go to http://developer.cisco.com/web/xde/resources

Using Extension Points

Extension points are the recommended way to extend the provisioning system. Extension points allow the system to be extending without risking the existing functionality, and allowing new updates to be seamlessly installed.

For information about how to use extension points in the XDE SDK, go to http://developer.cisco.com/web/xde/resources

XDE SDK Functional Tests

The Prime Fulfillment XDE SDK includes a range of standardized regression tests that can be used to test whether the new extension point logic has broken the existing provisioning logic.

New tests can also be written to exercise any new provisioning logic and to validate that it works correctly.

With the XDE Eclipse plug-in installed, navigate to the **ProvTest** package, which contains all the of the functional tests for provisioning. Use this to test what you have developed to make sure that no existing functionality has been broken. It currently contains some 1800 tests and additional tests can be added.

For information about how to run XDE SDK functional tests in the XDE SDK, go to http://developer.cisco.com/web/xde/resources

Support Information and Help

For further information, FAQs, and help, go to http://developer.cisco.com/web/xde/.