Oracle® Communications IP Service Activator

SDK Installation and Setup Guide

Release 7.2

E47720-01

October 2013

ORACLE®

Oracle Communications IP Service Activator SDK Installation and Setup Guide, Release 7.2

E47720-01

Copyright © 2011, 2013, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Preface v

Related Documents v
Documentation Accessibility v
Online References v
Web References vi

1 Installing the Software Development Kit

About the SDK 1-1
About the SDK Environment 1-1
Third Party Tools and Versions 1-1
Setting Environmental Variables 1-2
Sample Commands to Set Windows Third Party Environment Variables 1-3
Sample Commands to Set Solaris Third Party Environment Variables 1-3
Installing the SDK 1-4
Test Environments 1-4
Upgrading an SDK Cartridge Installed on a Service Activator Host 1-4
Uninstalling 1-5
Uninstalling the SDK 1-5
Deleting a Generated Cartridge or Configuration Policy From an SDK Host 1-5
Uninstalling an SDK-generated Cartridge or Configuration Policy 1-5

Preface

This guide explains how to install the Oracle Communications IP Service Activator Software Development Kit (SDK). It details the components and related environment for installing and running SDK tools.

This guide provides details for installing, validating, and uninstalling the SDK.

Related Documents

For details about how to use the SDK to develop cartridges and configuration policies, refer to:

- *IP Service Activator SDK Developer Overview Guide*
- *IP Service Activator SDK Base Cartridge Developer Guide*
- *IP Service Activator SDK Service Cartridge Developer Guide*
- *IP Service Activator SDK Configuration Policy Extension Developer Guide*

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Online References

Online references are included in the **ipsaSDK-devGuides-***VERSION_STRING***-sdk1.0.zip** distribution file. (See "Installing the SDK".)

IP Service Activator Schema Online Documentation

An online reference is provided containing schema documentation which outlines how to populate and read XML instance documents used by the IP Service Activator Network Processor.

It is accessed through:

*Service_Activator_home***ipsaSDK\doc\schemaDoc\index.html**

IP Service Activator Configuration Policy Online Documentation

An online reference is provided that documents the Configuration Policies that are provided with the IP Service Activator core.

It is accessed through:

*Service_Activator_home***ipsaSDK\doc\schemaDoc\configPolicyIndex.html**

where *Service_Activator_home* is the directory to which IP Service Activator is deployed.

Web References

You can obtain the XML Schema Primer from the W3C, located at:

http://www.w3.org/TR/xmlschema-0/

1

Installing the Software Development Kit

This chapter explains how to install the Oracle Communications IP Service Activator Software Development Kit (SDK), how to install the required third-party utilities that support the SDK, and how to set up required environment variables.

About the SDK

The SDK is used to develop cartridges and configuration policies. The cartridges are deployable into an IP Service Activator Network Processor installation, and the configuration policies are deployable into an IP Service Activator client installation.

Together, the developed cartridges and configuration policies extend the functionality of IP Service Activator. This can include, for example, support for additional services, and support for vendor devices not directly supported by the IP Service Activator product.

About the SDK Environment

The SDK is intended to be used on either a Windows-based or Oracle Solaris system. The documentation assumes that a Windows-based system is being used.

The SDK can run independently of IP Service Activator and Configuration Management. It does not need to be installed on a IP Service Activator or Configuration Management host. However, there may be some advantages to installing on a test IP Service Activator host, for example, to make it simpler to perform end-to-end testing.

Table 1–1 lists the placeholders used in this guide.

Table 1–1 SDK Directory Placeholders

Placeholder	Description
SDK_home	The directory in which the SDK is installed. This directory name as delivered is **ipsaSDK**.
Service_Activator_home	The directory to which IP Service Activator is deployed.

Examples giving paths are expressed with backslashes ('\') to conform to Windows. If you are using Solaris, use forward slashes ('/') in path names.

Third Party Tools and Versions

In order to use the SDK you must download and install the following third party utilities:

- Java Development Kit (JDK) version 1.6.0_03

  Download from:

  http://www.oracle.com/technetwork/java/javasebusiness/downloads/java-archive-downloads-javase6-419409.html#jdk-6u31-oth-JPR

- Ant 1.6.2

  Download the **apache-ant-1.6.2-bin.zip** file from:

  http://archive.apache.org/dist/ant/binaries/

- JUnit 3.8.1

  Download from:

  http://sourceforge.net/projects/junit/files/junit/3.8.1/

  Note: After JUnit is installed, copy the **junit.jar** file to the ant library directory.

  On Windows:

  ```
  copy path_to_junit_install_dir\junit.jar path_to_ant_install_
  dir\lib
  ```

  On Solaris:

  ```
  cp path_tojunit_install_dir/junit.jar path_to_ant_install_dir/lib
  ```

- Saxon 8.8.0.7

  Download:

  http://sourceforge.net/projects/saxon/files/Saxon-B/8.8.0.7/

  Note: Ensure that you have the Saxon 8.8.0.7 release. The name of the .jar file (**saxon8.jar**) does not change with different sub-versions of Saxon 8.

- Velocity 1.4

  Download from:

  http://archive.apache.org/dist/velocity/engine/1.4/

  If **velocity1-4.zip** is reported to be an invalid archive, download **velocity1-4.tar.gz**

Setting Environmental Variables

In order to run the SDK tools, certain environment variables must be set. If one of the third party tools can't be located because an environment variable is not set correctly, you'll receive an error when you run Ant.

Note: An existing CLASSPATH environment variable may interfere with the CLASSPATH required by the SDK. It is therefore recommended that the CLASSPATH environment variable be cleared/unset in the session where the SDK is being used.

For example, in Windows:

```
set CLASSPATH=
```

You can create a batch or script file to manually set up environment variables each time you are ready to work with the SDK, or you can adapt your system settings so that the environment variables are permanently configured.

The required environment variables include:

- PATH: must include paths to the **...\bin** directories for Java and Ant
- JAVA_HOME: home directory of the JDK
- JUNIT: directory containing JUnit binaries
- ANT_HOME: directory containing Ant binaries
- SAXON: directory containing Saxon binaries
- VELOCITY: directory containing Velocity binaries

Sample Commands to Set Windows Third Party Environment Variables

The commands to configure the required environment variables from a Windows command line session are similar to the following example.

Note: Replace *SDK_home* with the actual path to your SDK installation.

```
set CLASSPATH=
set JAVA_HOME=C:\Program Files\Java\jdk1.6.0_03
set ANT_HOME=C:\apache-ant-1.6.2
set JUNIT=C:\junit3.8.1
set SAXON=C:\saxonb-8.8.0.7
set VELOCITY=C:\velocity-1.4
set IPSA_SDK_HOME=SDK_home
set PATH=%JAVA_HOME%\bin;%ANT_HOME%\bin;%IPSA_SDK_HOME%\bin;%PATH%
```

Sample Commands to Set Solaris Third Party Environment Variables

The commands to configure the required environment variables on Solaris are similar to the following example.

Note: Replace *SDK_home* with the actual path to your SDK installation.

```
unset CLASSPATH
export JAVA_HOME='/usr/jdk/1.6.0_03'
export ANT_HOME='/opt/apache-ant-1.6.2'
export JUNIT='/opt/junit3.8.1'
export SAXON='/opt/saxonb-8.8.0.7'
export VELOCITY='/opt/velocity-1.4'
```

```
export IPSA_SDK_HOME='SDK_home'
export PATH=$JAVA_HOME/bin:$ANT_HOME/bin:$IPSA_SDK_HOME/bin:$PATH
```

Installing the SDK

You can unzip the SDK files to any directory. If you have some IP Service Activator components already installed, it is acceptable, but not necessary to unzip the SDK files to the IP Service Activator directory (which is referred to in the documentation as *Service_Activator_home*).

The files will be copied to a new directory named **ipsaSDK** under the directory you select. For example, if you unzip the files to *Service_Activator_home*, the files will be placed under *Service_Activator_home***ipsaSDK** and *Service_Activator_home***ipsaSDK** will be referred to as SDKHome in the SDK documentation.

To install the SDK:

1. Install third party utilities as described in "Third Party Tools and Versions".
2. Ensure the SDK environment variables have been set up as described in "Setting Environmental Variables".
3. Download the IP Service Activator SDK from the Oracle software delivery Web site:

   http://edelivery.oracle.com

   The files to be installed are **ipsaSDK-***VERSION_STRING***-sdk1.0.zip** and **ipsaSDK-devGuides-***VERSION_STRING***-sdk1.0.zip**.
4. Unzip the SDK files to any directory.

Note: The distribution file **ipsaSDK-devGuides-***VERSION_STRING***-sdk1.0.zip** includes online references. See "Online References" for more information.

Test Environments

Basic tests of cartridges and configuration policies created with the SDK can be performed without an IP Service Activator or Configuration Management system.

To perform end-to-end tests of installed cartridges and configuration policies that you have developed, it is recommended that you have a Solaris-based IP Service Activator installation dedicated to the purpose. This system should have access to some actual devices.

It is not recommended to perform testing on a live IP Service Activator installation.

Upgrading an SDK Cartridge Installed on a Service Activator Host

The procedure for upgrading an SDK-based custom cartridge is impacted by the nature of the cartridge customization and its implementation.

If SDK-based custom cartridges and service extensions have been written to comply with the upgrade interfaces in the SDK developer guides (configuration policy and service cartridge), then the **dbUpgrade** and **npUpgrade** scripts will automatically upgrade SDK-based custom cartridges and service extensions.

It is strongly recommended that SDK developers adhere to these interfaces so that their upgrade is in line with the upgrade procedure given in this chapter.

To upgrade an SDK cartridge installed on a Service Activator host:

1. Run npSnapshot on the old system.
2. Uninstall any old custom components, preserving required configuration files.
3. Uninstall the old system, preserving required configuration files.
4. Install the new system.
5. Install new custom components.
6. Integrate required configuration files from the old system.
7. Run **dbupgrade**.
8. Run **npUpgrade**.

Uninstalling

This section describes:

- How to uninstall the complete SDK
- How to delete individual SDK components (i.e. a base cartridge, service cartridge, or configuration policy) from an SDK host
- How to uninstall SDK-generated cartridges or configuration policies deployed into a IP Service Activator installation

Uninstalling the SDK

Remove all sub-directories and files in the *SDK_home* directory.

Deleting a Generated Cartridge or Configuration Policy From an SDK Host

Cartridges and configuration policies reside in the following directories:

SDK_home/samples/baseCartridge/*sdk_global_cartridgeName*
SDK_home/samples/serviceCartridges/*sdk_global_cartridgeName*
SDK_home/samples/configPolicies/*sdk_global_configPolicyName*

Remove the specific cartridge or configuration policy directory you want to delete, including all of its files and sub-directories.

WARNING: Generated cartridge or configuration policy files may have been customized after they were generated. Permanently deleting these files will result in the permanent loss of the customizations to the cartridge source files. Execute 'ant clean' in a cartridge directory, if instead of deleting the source files, your intent is to delete the files resulting from compilation of the cartridge.

Uninstalling an SDK-generated Cartridge or Configuration Policy

Each SDK cartridge or configuration policy contains a manifest file that lists all of the files packaged into its zip file. When a cartridge or configuration policy is installed on a IP Service Activator host, this manifest file is place into the *Service_Activator_home*/**cartridgeUninstall** directory.

Note: For this topic, it is assumed that the cartridge or configuration policy has been deployed into a Solaris environment, so the instructions are in Solaris format instead of Windows format.

To uninstall an SDK cartridge or configuration policy from an IP Service Activator host:

1. Log in to a command shell on the IP Service Activator host where the SDK cartridge or configuration policy is installed.
2. Change directory to *Service_Activator_home*/**cartridgeUninstall**.

   Note the manifest file name of the SDK cartridge or configuration policy that you want to remove. The file has one of the following formats:

   - *sdk_global_cartridgeName*-baseCartridge-*versionString*.manifest
   - *sdk_global_configPolicyName*-configPolicy-*versionString*.manifest
3. Change directory to *Service_Activator_home*.
4. Remove the cartridge or configuration policy by entering one of the following commands:

   ```
   bin/uninstallCartridge.sh [-v] [-k] cartridgeUninstall/sdk_global_
   cartridgeName-baseCartridge-versionString.manifest

   bin/uninstallCartridge.sh [-v] [-k] cartridgeUninstall/sdk_global_
   cartridgeName-serviceCartridge-versionString.manifest

   bin/uninstallCartridge.sh [-v] [-k] cartridgeUninstall/sdk_global_
   configPolicyName-configPolicy-versionString.manifest
   ```

   where the -v option provides extra output details about the uninstall operation and the -k option leaves intact any empty directories.

   The script removes any files and directories used by this cartridge or configuration policy. Any directories that are shared with other cartridges or configuration policies are removed only if they are empty. With the -k option, empty directories are not removed.

   The script also removes the cartridge or configuration policy manifest file.
5. Restart the network processor after the cartridge or configuration policy is uninstalled.

To remove the HTML for an SDK-generated configuration policy from an IP Service Activator client host:

1. Delete the HTML file in the *Service_Activator_home*/**Config** directory that is named after the configuration policy.
2. Remove the reference to the HTML file in the **ConfigurationPolicy.cfg** file. This consists of removing the object and policytype elements that refer to the configuration policy being removed. For example:

   ```
   <GenericRule>
   ...
   <object>
   <policytypeReference ref="banner"/>
   </object>
   ...
   ```

```
<policytype name="banner">
<policyurl>file:///C:/ProgramFiles/Oracle Communications/IP Service
Activator/Config/banner.html
</policyurl>
</policytype>
...
</GenericRule>
```

Both of these files are in the *Service_Activator_home*/**Config** directory.