webMethods SOPERA Adapter Installation and User's Guide

Version 8.0

July 2012

WEBMETHODS

This document applies to webMethods SOPERA Adapter 8.0 and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Document ID: ADAPTER-SEA-IUG-80-20210331

Table of Contents

About this Guide

This guide describes how to install, configure, and use webMethods SOPERA Adapter. It contains information for administrators who manage a webMethods system, and for application developers who create services to communicate with the SOPERA infrastructure.

To use this guide effectively, you should:

- Understand the basic concepts of SOPERA Adapter and the SOPERA infrastructure.
- Be familiar with the set up and operation of webMethods Integration Server.
- Have a general idea about how to perform basic tasks with Software AG Designer.
- Be familiar with the set up and operation of My webMethods Server.
- Know how to create flow services and/or Java services.

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
Narrowfont	Identifies service names and locations in the format *folder.subfolder.service*, APIs, Java classes, methods, properties.
Italic	Identifies: Variables for which you must supply values specific to your own situation or environment. New terms the first time they occur in the text. References to other documentation sources.
`Monospace font`	Identifies: Text you must type in. Messages displayed by the system. Program code.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.
|	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the | symbol.
[ ]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [ ] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

Online Information and Support

Software AG Documentation Website

You can find documentation on the Software AG Documentation website at http://documentation.softwareag.com.

Software AG Empower Product Support Website

If you do not yet have an account for Empower, send an email to empower@softwareag.com with your name, company, and company email address and request an account.

Once you have an account, you can open Support Incidents online via the eService section of Empower at https://empower.softwareag.com/.

You can find product information on the Software AG Empower Product Support website at https://empower.softwareag.com.

To submit feature/enhancement requests, get information about product availability, and download products, go to Products.

To get information about fixes and to read early warnings, technical papers, and knowledge base articles, go to the Knowledge Center.

If you have any questions, you can find a local or toll-free number for your country in our Global Support Contact Directory at https://empower.softwareag.com/public_directory.aspx and give us a call.

Software AG TECHcommunity

You can find documentation and other technical information on the Software AG TECHcommunity website at http://techcommunity.softwareag.com. You can:

- Access product documentation, if you have TECHcommunity credentials. If you do not, you will need to register and specify "Documentation" as an area of interest.
- Access articles, code samples, demos, and tutorials.
- Use the online discussion forums, moderated by Software AG professionals, to ask questions, discuss best practices, and learn how other customers are using Software AG technology.
- Link to external websites that discuss open standards and web technology.

Data Protection

Software AG products provide functionality with respect to processing of personal data according to the EU General Data Protection Regulation (GDPR). Where applicable, appropriate steps are documented in the respective administration documentation.

1 Overview of webMethods SOPERA Adapter

About the Adapter

webMethods SOPERA Adapter is a webMethods Integration Server adapter that allows you to exchange data with the SOPERA infrastructure. The adapter provides seamless and real-time communications to and from the SOPERA ESB.

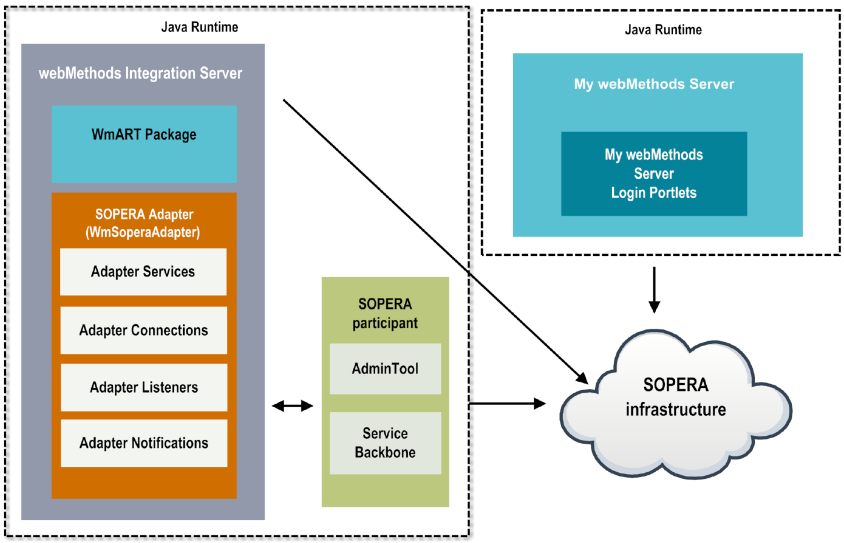
Using webMethods SOPERA Adapter, Integration Server clients can invoke SOPERA services and allow the SOPERA infrastructure to call the Integration Server services. For example, you can use SOPERA Adapter in a flow that receives data from the SOPERA infrastructure and then sends the processed data back to the SOPERA infrastructure.

SOPERA Adapter supports the request-response, one-way, notification, and request-callback SOPERA communication styles. For more information about SOPERA communication styles, see "Communication Styles Support" on page 15.

Architecture and Components

webMethods SOPERA Adapter provides a set of user interfaces, services, and templates that enable you to create integrations with the SOPERA infrastructure. The adapter is provided as a single package that must be installed on Integration Server. For detailed installation instructions, see "Installing webMethods SOPERA Adapter " on page 21.

The following diagram shows at a high level how the adapter components connect to the SOPERA infrastructure:



- webMethods Integration Server

  webMethods SOPERA Adapter is installed and runs on webMethods Integration Server.

- WmART Package

  The WmART package provides a common framework for webMethods version 6.0 and later adapters to use the Integration Server's functionality, making Integration Server the run-time environment for SOPERA Adapter. The WmART package is installed with Integration Server

and provides logging, transaction management, and error handling for the adapter and its connections, services, and notifications.

- SOPERA Adapter

  webMethods SOPERA Adapter is delivered as a single package called WmSoperaAdapter . The adapter installation includes templates from which all adapter connections, adapter services, adapter listeners, and adapter notifications can be created.

- SOPERA Participant

  A SOPERA Participant is a running business application that consumes or provides services. SOPERA Adapter enables the communication between the SOPERA infrastructure and the webMethods product suite by processing requests from SOPERA participants and sending requests to those participants. SOPERA Adapter uses the following components to interact with the SOPERA infrastructure:

  - AdminTool

    The SOPERA AdminTool is the utility used to accomplish the main administration tasks required for SOPERA Participants, such as looking up and registering services. You must specify the path to the SOPERA AdminTool in the SOPERA Adapter configuration file after you install the adapter.

  - Service Backbone

    The SOPERA Service Backbone (SBB) library interfaces with the SOPERA infrastructure. To access the SOPERA infrastructure, you must specify the path to the SOPERA Participant SBB library when you configure SOPERA Adapter connections.

- SOPERA Infrastructure

  The SOPERA infrastructure consists of SOPERA Technical Service Providers, the SOPERA Service Registry, a directory server, and a messaging system. webMethods SOPERA Adapter communicates with the SOPERA infrastructure to invoke SOPERA services or to handle requests from the SOPERA infrastructure to execute services on Integration Server.

- Java Runtime

  The execution environment for Integration Server and the SOPERA Participant interacting with webMethods SOPERA Adapter.

- My webMethods Server is a run-time container for functions made available by webMethods applications. The user interface in which you perform these functions is called My webMethods. You can access Integration Server using the My webMethods user interface.

- My webMethods Server Login Portlets

  The My webMethods Server Login Portlet allows logging on to My webMethods Server as a SOPERA user.

Adapter Package Management

webMethods SOPERA Adapter is provided as a package called WmSoperaAdapter that you manage like any package on Integration Server.

There are several considerations regarding how you set up and effectively manage your packages on Integration Server, such as those described in the following list:

- Configure user-defined packages for your adapter connections and adapter services. For more information, see "Managing the Adapter Package" on page 32.
- Understand how package dependencies work so you make the best decisions regarding how you manage your adapter services and adapter notifications. For more information, see "Package Dependency Requirements and Guidelines" on page 32.
- Control which development groups have access to which adapter services and adapter notifications. For more information, see "Controlling Group Access" on page 36.
- Enable and disable packages. For more information, see "Enabling and Disabling Packages" on page 33.
- Load, reload, and unload packages. For more information, see "Loading, Reloading, and Unloading Packages" on page 34.

Adapter Connections

An adapter connection enables Integration Server to connect to the SOPERA infrastructure at run time. You must configure an adapter connection before you can create adapter services or notifications. You create one or more connections at design time to use in integrations. The number of connections you create depends on your integration needs. You configure connections using Integration Server Administrator. You must have webMethods administrator privileges to access the administrative screens of the adapter. For instructions on configuring, viewing, editing, enabling, and disabling SOPERA Adapter connections, see "Adapter Connections" on page 37. For information about setting user privileges, see the *webMethods Integration Server Administrator's Guide* for your release. For a list of tasks that you must do before you can create your connections, see "Before Configuring or Managing Adapter Connections" on page 38.

Connection Pools

Integration Server includes a connection management service that dynamically manages connections and connection pools based on configuration settings that you specify for the connection. By default, connection pooling is enabled for all adapter connections.

Run-Time Behavior of Connection Pools

In SOPERA Adapter no physical connections are created or managed, but each connection pool creates a new SBB instance and uses that instance for all operations involving interaction with the SOPERA infrastructure.

Each SBB instance created is identified uniquely with a common AppID, but a different InstanceID that maps to the connection pool name.

It is recommended to restrict the number of connection pools to preserve possibly scarce resources such as memory space or network connections.

For information about configuring connections, see "Adapter Connections" on page 37.

Built-in Services for Connections

Integration Server provides built-in services that enable you to programmatically control connections. You can use the services to enable and disable a connection, and to return usage statistics and the current state (Enabled or Disabled) and error status for a connection. These services are located in the WmART package, in the pub.art.connection folder.

For complete information about the services in the WmART folder, see the *webMethods Integration Server Built-In Services Reference* for your release.

Changing the Connection Associated with an Adapter Service at Design Time

Integration Server provides a built-in service that you can use at design time to change the connection associated with an adapter service. This built-in service is named pub.art.service:setAdapterServiceNodeConnection. Using this service, you can change the specific connection associated with an adapter service at design time so that you do not need to recreate adapter services.

Note:
This built-in service can be run at design time only; do not use it within an Integration Server flow or Java service. You must run this service directly from Designer by selecting the service in Designer and running it.

For details about the setAdapterServiceNodeConnection service, see the *webMethods Integration Server Built-In Services Reference* for your release.

Adapter Services

Adapter services enable you to connect to the adapter's resource (that is the SOPERA infrastructure) and initiate an operation on the resource from Integration Server. Every invocation of a SOPERA service by other Integration Server services or Integration Server clients goes through an adapter service.

You call adapter services from flow services or Java services to interact with the SOPERA infrastructure. Integration Server uses adapter connections that you defined earlier to execute the adapter services.

Adapter services are based on templates provided with SOPERA Adapter. Each template represents a specific technique for doing work on a resource. An adapter service template contains all the

code necessary for interacting with the resource but without the data specifications. You provide these specifications when you configure a new adapter service.

Configuring a new service from an adapter service template is straightforward. Using Software AG Designer, you assign the service a default adapter connection.

After you select the connection for the adapter service, you select the adapter service template and supply the data specifications, including the mapping of data types from the adapter to or from your adapter resource. You use Software AG Designer to configure the adapter service. Some familiarity with using Software AG Designer is required. For more information, see the *webMethods Service Development Help*.

SOPERA Adapter provides the following adapter service templates:

Adapter Service Template	Description
SOPERA Generic Consumer	Invokes a generic SOPERA service.
SOPERA Notification Provider	Invokes a SOPERA notification service.
SOPERA Callback Sender Provider	Sends a callback response.

Adapter Listeners and Listener Notifications

When the interaction is initiated by the adapter resource (for example, SOPERA calling an Integration Server service), SOPERA Adapter uses adapter listeners and notifications.

By creating an adapter notification, an Integration Server service is registered as a provider in the SOPERA Service Registry or a handler is created to consume messages from the SOPERA infrastructure. When a notification is enabled, the provider is active and SOPERA Adapter can handle any requests by SOPERA participants.

You use Integration Server Administrator to configure and manage adapter notifications. For detailed instructions on how to configure listener notifications, see "Listener Notifications" on page 68.

SOPERA Adapter provides the following adapter notification templates that you configure using Software AG Designer:

Adapter Notification Template	Description
SOPERA Generic Provider	Publishes an IS service as a SOPERA service.
SOPERA Notification Consumer	Consumes SOPERA notification.
SOPERA Callback Consumer	Notifies for a callback response.
SOPERA Request-callback Provider	Provides a handler for Request-callback.

For detailed instructions on how to configure adapter notifications using Designer, see "Adapter Notifications" on page 61.

Communication Styles Support

SOPERA Adapter enables the communication between the SOPERA infrastructure and the webMethods product suite by processing requests from SOPERA participants and sending requests to those participants. SOPERA participants are business applications that consume services (service consumers) or provide services (service providers). You can use one of the following communication styles of SOPERA services as part of calling the SOPERA services from your adapter services:

- **Request-response.** The service consumer sends a request message to the service provider. The provider processes the request and returns a response message to the consumer. The request-response operation can be:
  - **Blocking.** The service blocks the service invocation until a response message is sent to the consumer.
  - **Non-blocking.** The service invocation returns before the response has been received.
- **One-way.** The service consumer sends a request message to the service provider, but does not expect any response.
- **Notification.** The service provider sends a message to multiple service consumers transparently. In effect this communication style implements a publish-subscribe communication. The service provider "publishes" a message that is received by any consumer that has subscribed to it.

  Note:*SOPERA notification* should not be confused with an *adapter notification*. An adapter notification is an element on Integration Server that receives data from the backend system. SOPERA notification is a communication style for a publish-subscribe type of message exchange.
- **Request-callback.** The request-callback communication style is similar to the request-response style in that the service consumer sends a message to the provider and then the provider sends back a message. The difference is that the sending of the request and callback messages is implemented as independent one-way operations that are correlated by the SOPERA library. With this style, the callback can be received at any future point of time after the request has been sent and might actually go to a different physical system than that from which the request originated. In a similar way, the provider does not have to send the callback immediately and might delegate the sending of the callback to a different component.

Creating Service Consumers

SOPERA Adapter provides adapter service and notification templates that you use to create service consumers for all supported communication styles. The following table lists the adapter templates that you use to create a service consumer for each communication style:

Communication Style	Adapter Service/Notification Template Name	Template Type
Request-response	SOPERA Generic Consumer	Adapter service

Communication Style	Adapter Service/Notification Template Name	Template Type
One-way	SOPERA Generic Consumer	Adapter service
Notification	SOPERA Notification Consumer	Adapter notification
Request-callback	■ SOPERA Generic Consumer	Adapter service
	■ SOPERA Callback Consumer	Adapter notification
	Note: Used only when the **Send request only** field in the SOPERA Generic Consumer template is selected.	

With the request-callback communication style, you create a service consumer in one of the following ways:

- Configure the SOPERA Generic Consumer adapter service by selecting a specific SOPERA service and the request-callback operation as the operation that the adapter service will provide. This is similar to a non-blocking request-response style.
- Create a Request-callback service consumer in two steps:
  - Create the SOPERA Generic Consumer adapter service that submits the request.
  - Create the SOPERA Generic Consumer adapter notification that receives the callback.

For more information about creating service consumers using the adapter templates, see "Adapter Services" on page 45 and "Adapter Notifications" on page 61.

Creating Service Providers

SOPERA Adapter supports creating service providers for the supported communication styles in the following ways:

- Use an existing IS service and SOPERA Adapter generates internally the service description (SDX) from the signature of that service.
- Select an existing SDX and create an empty IS service with a signature matching the signature of the existing SDX. This approach is known as the SDX-first approach.

With the request-callback communication style you create adapter service providers in the following ways:

- Create an adapter notification that calls an IS service and returns the output of that service as the callback.
- Create an adapter service provider in two steps:

- Create an adapter notification that receives the request and publishes a document.
- Create an adapter service that is invoked separately to return the callback.

SOPERA Adapter provides adapter service and notification templates that you use to create adapter service providers for all supported communication styles. The following table lists the adapter templates that you use to create an adapter service provider for each communication style:

Communication Style	Adapter Service/Notification Template Name	Template Type
Request-response	SOPERA Generic Provider	Adapter notification
One-way	SOPERA Generic Provider	Adapter notification
Notification	SOPERA Notification Provider	Adapter service
Request-callback	■ SOPERA Generic Provider or ■ SOPERA Request-callback provider ■ SOPERA Callback Sender Provider	Adapter notification Adapter service Adapter notification

For more information about creating service providers using the adapter templates, see "Adapter Services" on page 45 and "Adapter Notifications" on page 61.

SOPERA User Authentication

SOPERA Adapter authenticates a SOPERA user against the SOPERA infrastructure using JAAS login modules for:

- Logging on to Integration Server Administrator
- Logging on to My webMethods Server
- Sending outgoing SOPERA calls
- Receiving incoming SOPERA calls

To enable authentication for Integration Server Administrator, incoming, and outgoing SOPERA calls, you must configure the Integration Server Login Module (IS Login Module).

To enable authentication for My webMethods Server, you need to configure the My webMethods Server Login Module (MWS Login Module).

For more information about JAAS login modules, see the *webMethods Integration Server Administrator's Guide* for your release.

For both the IS Login Module and the MWS Login Module, the functionality depends on central user management being properly configured beforehand. For information about how to configure

these login modules, see "Configure the IS Login Module" on page 25 and "Configure the MWS Login Module" on page 27.

After you configure the IS Login Module and the MWS Login Module, they authenticate SOPERA users against the SOPERA infrastructure and place them in a group called SOPUsers. You assign access control lists (ACLs) to the SOPUsers group as required to access the functionality that you want to use.

For outgoing SOPERA calls, the adapter propagates the session information containing the SOPERA SAML token to the SOPERA infrastructure. In this way if the provider or consumer policy has been configured for authentication and authorization, the infrastructure validates the token before the service is invoked. The SOPERA call is performed by the same user that invokes the adapter service on Integration Server.

For incoming SOPERA calls, authentication happens on two levels. First, the infrastructure authenticates and authorizes the user based on the SAML token that has been sent if the provider and consumer have the required security configuration. Then the user is authenticated in Integration Server when incoming call authentication has been configured. The adapter validates the SAML token it receives and creates a session for the respective user on Integration Server. The IS service configured as a provider is invoked with the user credentials.

For more information on SOPERA authentication and authorization, see the SOPERA documentation.

Using Version Control Systems to Manage Adapter Elements

The adapter supports the Version Control System (VCS) Integration feature provided by Designer. When you enable the feature in Integration Server, you can check adapter packages or elements into and out of your version control system from Designer. For more information about the VCS Integration feature, see the *Configuring the VCS Integration Feature*.

Beginning with Integration Server 8.2 SP3, the adapter supports the local service development feature in Designer. This feature extends the functionality of the VCS Integration feature to check package elements and their supporting files into and out of a VCS directly from Designer. For more information about local service development and how it compares to the VCS Integration feature, see the *webMethods Service Development Help*.

Optimize Infrastructure Data Collector Support for the Adapter

Optimize Infrastructure Data Collector monitors the system and operational data associated with webMethods run-time components such as Integration Servers, Broker Servers, Brokers, and adapters, and reports the status of these components on Optimize for Infrastructure or other external tools. When you start monitoring an Integration Server, Infrastructure Data Collector automatically starts monitoring all ART-based adapters that are installed on Integration Server. For information about monitored key performance indicators (KPIs) collected for the monitored adapter components, see the *webMethods Optimize User's Guide*.

Viewing the Adapter's Update Level

You can view the list of updates that have been applied to the adapter. The list of updates appears in the **Updates** field on the adapter's About page in Integration Server Administrator.

Controlling Pagination

When using the adapter on Integration Server 9.0 and later, you can control the number of items that are displayed on the adapter Connections screen. By default, 10 items are displayed per page. Click **Next** and **Previous** to move through the pages, or click a page number to go directly to a page. To change the number of items displayed per page, set the `watt.art.page.size` property and specify a different number of items.

To set the number of items per page

1. From Integration Server Administrator, click **Settings > Extended**.

2. Click **Edit Extended Settings**. In the Extended Settings editor, add or update the `watt.art.page.size` property to specify the preferred number of items to display per page. For example, to display 50 items per page, specify: watt.art.page.size=50

3. Click **Save Changes**. The property appears in the Extended Settings list.

   For more information about working with extended configuration settings, see the *webMethods Integration Server Administrator's Guide* for your release.

2 Installing webMethods SOPERA Adapter

Overview

This chapter explains how to use the Software AG Installer wizard to install, upgrade, and uninstall webMethods SOPERA Adapter 8.0. For complete information about other installation methods or installing other webMethods products, see the *Installing webMethods Products On Premises* for your release.

This *webMethods SOPERA Adapter Installation and User's Guide* refers to the main installation directory as *Software AG_directory* .

Requirements

For a list of the operating systems and products supported by SOPERA Adapter, see *webMethods Adapters System Requirements* , available in the webMethods area of the Software AG Documentation Website.

Installing SOPERA Adapter

The instructions in this section are meant to be used with the more complete instructions in the *Installing webMethods Products On Premises* for your release.

Important:
You must install the SOPERA ESB infrastructure before you install SOPERA Adapter.

To install SOPERA Adapter 8.0

1. Download the Software AG Installer from the Empower Product Support website.

2. If you are installing the adapter on an existing Integration Server, shut down Integration Server.

3. Start the Software AG Installer wizard.

   Choose the webMethods release that includes the Integration Server on which to install the adapter. For example, if you want to install the adapter on Integration Server 9.5, choose the 9.5 release.

   If you are installing on an existing Integration Server, specify the webMethods installation directory that contains the host Integration Server. If you are installing both the host Integration Server and the adapter, specify the installation directory to use. The Installer will install the adapter in the *Integration Server_directory* \packages directory.

4. In the product selection list, select **Adapters > webMethods SOPERA Adapter 8.0**.

5. To install documentation for webMethods SOPERA Adapter, on the Documentation panel in Installer, select **Adapter Readmes and Documentation**. Alternatively, you can download the adapter documentation at a later time from the Software AG Documentation Website (http://documentation.softwareag.com/).

6. After the Installer completes the adapter installation, close the Installer.

7. Copy the following files from the SOPERA *SBB_directory*\lib directory into the *Integration Server_directory* \packages\ WmSoperaAdapter \code\jars directory:

   jax-qname-namespace.jar

   sbb-factory-impl.jar

   sbb-papi.jar

8. Copy the following files from the SOPERA infrastructure into the *Integration Server_directory* \packages\ WmSoperaAdapter \code\jars\static directory:

   login.jar

   sbb-se-core.jar

9. Copy the following files from the SOPERA *AdminTool_directory*\lib directory into the *Integration Server_directory* \packages\ WmSoperaAdapter \code\jars directory:

   jaxmeapi-0.5.1.jar

   jaxmexs-0.5.1.jar

   jaxme2-0.5.1.jar

   jaxme2-rt-0.5.1.jar

   privilegemanagement-jaxb.jar

   privilegemanagement-proxy.jar

   sbb-admin-tool.jar

   sbb-configrepos-scopepath.jar

   sbb-instance-proxy-plugin.jar

   sbb-proxy-util.jar

   sbb-toolsuite-service-utilities.jar

   sbb-toolsuite-common.jar

   userAccountManagement-proxy.jar

   userAccountManagement-jaxb.jar

10. Complete the installation as described in "Completing the Installation" on page 23.

11. Start the host Integration Server.

Completing the Installation

To complete the installation, you must perform the following tasks before start up.

Set SOPERA Specific Properties in the SOPERA Adapter Configuration File

To set the properties in the sopera.config file

1. Navigate to the *Integration Server_directory* \packages\ WmSoperaAdapter \config directory.
2. Open the sopera.config file in a text editor.
3. Specify the properties as described in the following table:

Property	Description/Action
org.sopware.sbb.home	Required. Set to the home directory of the SOPERA SBB Library installation on the local system. For example, `C:\SOPERA\ServiceBackbone`
org.sopware.admintool.home	Set to the home directory of the SOPERA Admin Tool location on the local system. This parameter is required. For example, `C:\SOPERA\AdminTool`
org.sopware.transport.http.enable	Optional. Set the HTTP inbound transport for the SOPERA SBB Library. Valid values are: ■ `true` Enables HTTP. This is the default value. ■ `false` Disables HTTP.
org.sopware.transport.http.port	Optional. Set the HTTP inbound port for the SBB Library running on the Java VM. If you do not specify a value, the adapter will throw an error when you attempt to enable the adapter notification.
org.sopware.transport.https.enable	Optional. Set the HTTPS inbound transport for the SOPERA SBB Library. Valid values are: ■ `true` Enables HTTPS. ■ `false` Disables HTTPS. This is the default value.
org.sopware.transport.https.port	Optional. Set the HTTPS inbound port for the SOPERA SBB Library running on the Java VM.
org.sopware.transport.https.keystore	Optional. Set the file name of the keystore used by the SOPERA SBB Library for enabling the HTTPS transport.

Property	Description/Action
org.sopware.transport.https.password	Optional. The password for accessing the keystore used by the SBB Library for enabling the HTTPS transport.
org.sopware.transport.https.clientauth	Optional. Set client authentication for the HTTP or HTTPS inbound transport. Valid values are: ■ `true` Enables client authentication. ■ `false` Disables client authentication. This is the default value.

Note:
You can remove or comment the optional properties in the sopera.config file. For details about these system properties, see the SOPERA configuration reference guide.

4. Save and close the file.

Configure the IS Login Module

You use the IS Login Module to authenticate a SOPERA user against a SOPERA infrastructure.

The IS Login Module is a JAAS login module that is automatically installed as part of the SOPERA Adapter installation. By default, the IS Login Module is not enabled. For more information about SOPERA user authentication, see " SOPERA User Authentication" on page 17.

To configure the IS Login Module

1. Configure central user management as described in the *webMethods Integration Server Administrator's Guide* for your release.

   Central user management is required to store and manage information about SOPERA users.

2. Navigate to the *Integration Server_directory* \config directory.

3. Open the is_jaas.cnf file in a text editor.

4. Edit the file as follows:

   a. Add the following code in the IS_Transport section to include the WmLoginModuleIS login module. Set the authenticationTSPEndpoint parameter to point to your token service:

```
IS_Transport
{
        com.softwareag.security.login.webmethods.WmLoginModuleIS
sufficient
   create_user_principal=true
```

```
          auth-type=auth-id-password
          authenticationTSPEndpoint=

"http://SOPERAInfrastructureServer:18080/TokenService/services/Trust"
          soapRequestTemplateFile="config/sts-request.template";

        com.wm.app.b2b.server.auth.jaas.X509LoginModule requisite;
        com.wm.app.b2b.server.auth.jaas.BasicLoginModule requisite;
};
```

where *SOPERAInfrastructure Server* is the hostname of your actual SOPERA infrastructure server, for example: `"http://localhost:18080/TokenService/services/Trust"`

b. Configure incoming SOPERA call authentication. SOPERA Adapter verifies the authentication information it receives from an incoming SOPERA call and creates an Integration Server session for it. This configuration is required only when SOPERA Adapter invokes an Integration Server service that does not have the anonymous ACL for execution. To enable incoming call authentication, add the following section:

Note:
The values for *verificationKeystore* and *verificationKeystorePwd* parameters will depend on your environment.

```
sopware.auth-token-checking {
    org.sopware.security.login.webmethods.WMLoginModule REQUIRED
    auth-type=auth-token-checking
    expiryClearance="525600"
    verificationKeystore="/Users/root/keystore.jks"
    verificationKeystorePwd="atleast8"
    verificationCertAlias="client"
    verifyTrustChain="false"
    cacheCleanupInterval="5"
    samlAttributeForPrincipalName="pxpUid";
};
```

5. Save and close the is_jaas.cnf file.

6. Copy the sts-request.template file from the *Integration Server_directory* \packages\ WmSoperaAdapter \config directory into the *Integration Server_directory* \config directory.

7. Restart Integration Server for the changes to take effect.

8. Log on to Integration Server Administrator as Administrator and go to **Security > Access Control Lists**.

9. Add the group of SOPERA users named `SOPUsers` to the ACL that you require.

   For example, if you want to log on as Administrator in Integration Server Administrator with a SOPERA user, you can add the `SOPUsers` group to the **Allowed** ACL list as described in the *webMethods Integration Server Administrator's Guide* for your release.

Configure the MWS Login Module

When you use My webMethods Server and you want to log on to My webMethods Server using a SOPERA user, you must configure the MWS Login Module. The My webMethods Server portlets that you must deploy are packaged with SOPERA Adapter and are located in the *Integration Server_directory* \packages\WmSoperaAdapter\portlets directory.

Note:
Configuring the MWS Login Module is required *only* when you want to use My webMethods interfaces to access Integration Server. If you are using only Integration Server Administrator, configure only the IS Login Module as described on "Configure the IS Login Module" on page 25.

To configure the MWS Login Module

1. Configure central user management as described in the *webMethods Integration Server Administrator's Guide* for your release.

   Central user management is required to store and manage information about SOPERA users.

2. Copy the is_jaas.cnf file from the *Integration Server_directory* \config directory into the *My webMethods Server_directory* \default\config directory, and then rename the file to sopera_login.properties.

   Important:
   If you want to use only MWS Login Module and you have not configured IS Login Module, you must edit the is_jaas.cnf file as described in step 4a in "Configure the IS Login Module" on page 25.

3. Copy the sts-request.template file from the *Integration Server_directory* \packages\ WmSoperaAdapter \config directory into the *My webMethods Server_directory* \default\config directory.

4. Install both portlets, wm_sopera_auth.pdp and wm_sopera_login.pdp, using My webMethods:

   a. Log on toMy webMethods Server as system administrator.

   b. Navigate to **Administration > Configuration > Install Administration**.

   c. Click **Install New Component**.

   d. Specify the *Integration Server_directory* \packages\ WmSoperaAdapter \portlets directory as the location for the portlets.

   e. Click **Next**.

   f. Click **Install**.

If the portlets are installed successfully, you will get a confirmation message verifying that the install succeeded. If the installation fails, check your log files to troubleshoot the installation failure. For more information about how to install portlets, see the *Administering My webMethods Server* for your release.

5. Copy the login.jar file and the login-webmethods.jar file from the *Integration Server_directory* \packages\ WmSoperaAdapter \code\jars\static directory into the following location in the wm_sopera_auth portlet installation: *My webMethods Server_directory* \server\default\system\wm_sopera_auth\lib

6. Create a custom login page as follows. With this custom login page, you can log on as both a SOPERA user and a normal My webMethods Server user. For detailed instructions for each of these steps, see the *Administering My webMethods Server* for your release.

   a. Log on to My webMethods Server as system administrator.

   b. Go to the **Public Folders** and add a new page from the **Tools** menu.

   c. Add the SOPERA login portlet from **Portlets > System** to the new page.

   d. Ensure that the authentication scheme for your custom login page and the SOPERA login portlet is `Anonymous`.

   e. Create a new login rule that invokes the new custom login page. Ensure that this rule is the first to be executed.

Upgrading from SOPERA Adapter 7.1

To upgrade and migrate from SOPERA Adapter 7.1 to SOPERA Adapter 8.0

1. Back up your existing SOPERA Adapter 7.1 installation and all custom packages.

2. In Integration Server Administrator, go to **Packages > Management** and locate the WmSoperaAdapter package.

3. To uninstall SOPERA Adapter 7.1, select **Safe Delete**. The existing installation is deleted and preserved for reference.

4. Install SOPERA Adapter 8.0 as described in "Installing SOPERA Adapter " on page 22.

5. Complete the installation as described in "Completing the Installation" on page 23.

Uninstalling SOPERA Adapter

The instructions in this section are meant to be used with the uninstallation instructions in the *Installing webMethods Products On Premises* for your release.

To uninstall SOPERA Adapter 8.0

1. Shut down the host Integration Server. You do not need to shut down any other webMethods products or applications that are running on your machine.

2. Start the Software AG Uninstaller, selecting the webMethods installation directory that contains the host Integration Server. In the product selection list, select **Adapters > webMethods SOPERA Adapter 8.0**.

3. After the Uninstaller completes, restart the host Integration Server.

4. The Uninstaller removes all SOPERA Adapter 8.0-related files that were installed. However, it does not delete files created after you installed the adapter (for example, jar files copied into *Integration Server_directory* \packages\ WmSoperaAdapter \ code\jars\static\ or configuration files), nor does it delete the adapter directory structure. You can go to the *Integration Server_directory* \packages directory and delete the WmSoperaAdapter directory.

5. The Uninstaller does not delete any user-defined SOPERA Adapter 8.0 components such as connections, adapter services, or adapter notifications. Because these components will not work without the adapter, delete them manually, either at the file system level or using Designer. For instructions, see the *webMethods Service Development Help*.

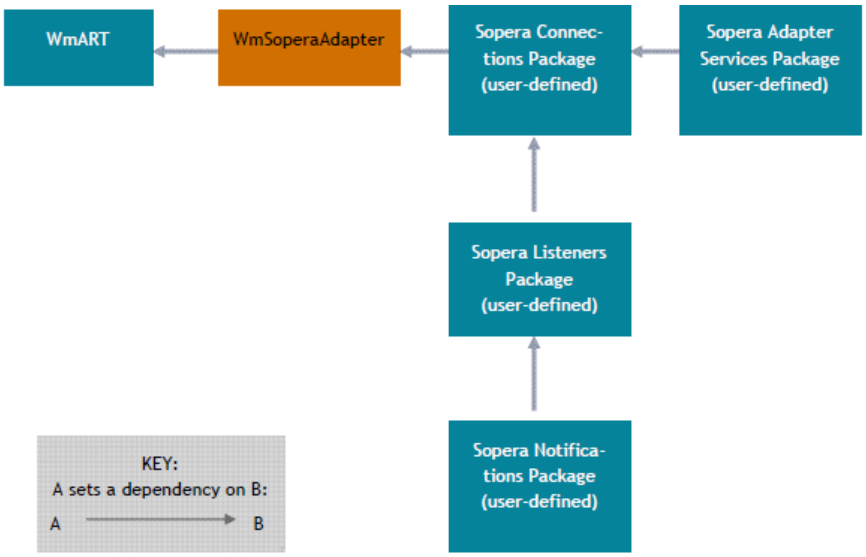
3 Package Management

Managing the Adapter Package

SOPERA Adapter is provided as a package called WmSoperaAdapter . You manage the WmSoperaAdapter package as you would manage any package on Integration Server.

When you create connections, adapter services, and adapter notifications, define them in user-defined packages rather than in the WmSoperaAdapter package. Doing so will allow you to manage the package more easily.

As you create user-defined packages in which to store connections, adapter services, and adapter notifications, use the package management functionality provided in Designer and set the user-defined packages to have a dependency on the WmSoperaAdapter package. That way, when the WmSoperaAdapter package loads or reloads, the user-defined packages load automatically. See the following diagram:



Package management tasks include:

- Setting package dependencies
- Enabling and disabling packages
- Loading, reloading, and unloading packages
- Importing and exporting packages
- Controlling group access

Package Dependency Requirements and Guidelines

This section contains a list of dependency requirements and guidelines for user-defined packages. For instructions for setting package dependencies, see the *webMethods Service Development Help*.

- By default, the WmSoperaAdapter package has a dependency on the WmART package. Do not change this dependency.

- A user-defined package must have a dependency on its associated adapter package, WmSoperaAdapter .
- Package dependencies ensure that at startup Integration Server automatically loads or reloads all packages in the proper order: the WmART package first, the adapter package next, and the user-defined package(s) last. The WmART package is automatically installed when you install Integration Server. You should not need to manually reload the WmART package.
- If the connections and adapter services of an adapter are defined in different packages, then:
  - A package that contains the connection(s) must have a dependency on the adapter package, WmSoperaAdapter .
  - Packages that contain adapter services must have a dependency on their associated connection package.
- Keep connections for different adapters in separate packages so that you do not create interdependencies between adapters. If a package contains connections for two different adapters, and you reload one of the adapter packages, the connections for both adapters will reload automatically.
- Integration Server will not allow you to enable a package if it has a dependency on another package that is disabled. That is, before you can enable your package, you must enable all packages on which your package depends. For information about enabling packages, see "Enabling and Disabling Packages" on page 33.
- Integration Server will allow you to disable a package even if another package that is enabled has a dependency on it. Therefore, you must manually disable any user-defined packages that have a dependency on the adapter package before you disable the adapter package.
- You can name connections, adapter services, and notifications the same name provided that they are in different folders and packages.

Enabling and Disabling Packages

All packages are automatically enabled by default. When you want to temporarily prohibit access to the elements in a package, disable the package. When you disable a package, the server unloads all of its elements from memory. Disabling a package prevents Integration Server from loading that package at startup. A disabled package will remain disabled until you explicitly enable it using Integration Server Administrator.

Enabling a Package

To enable a package

1. In Integration Server Administrator: **Packages > Management**.
2. Click **No** in the **Enabled** column. The server displays **Yes** in the **Enabled** column.

Note:
Enabling an adapter package will not cause its associated user-defined package(s) to be reloaded. For information about reloading packages, see "Loading, Reloading, and Unloading Packages" on page 34.

Important:
Before you manually enable a user-defined package, you must first enable its associated adapter package (WmSoperaAdapter). Similarly, if your adapter has multiple user-defined packages, and you want to disable some of them, disable the adapter package first. Otherwise, errors will be issued when you try to access the remaining enabled user-defined packages.

Disabling a Package

To disable a package

1. In Integration Server Administrator: **Packages > Management**.

2. Click **Yes** in the **Enabled** column for the package that you want to disable. The server issues a prompt to verify that you want to disable the package. Click **OK** to disable the package. When the package is disabled, the server displays **No** in the **Enabled** column.

A disabled adapter will:

- Remain disabled until you explicitly enable it using Integration Server Administrator.
- Not be listed in Designer.

Loading, Reloading, and Unloading Packages

As described in "Package Dependency Requirements and Guidelines" on page 32, if user-defined packages are properly configured with a dependency on the adapter package, at startup Integration Server automatically loads or reloads all packages in the proper order: the WmART package first, the adapter package next, and the node package(s) last. You should not need to manually reload the WmART package.

Reloading Packages Manually

Reloading a user-defined package will not cause its associated adapter package to be reloaded. You can reload adapter packages and user-defined packages from either Integration Server Administrator (by clicking the Reload icon on the Management window) or from Designer (by right-clicking the package and selecting the **Reload Package** option from the menu).

Unloading Packages

At shutdown, Integration Server unloads packages in the reverse order in which it loaded them: it unloads the node package(s) first, the adapter package next, and the WmART package last (assuming the dependencies are correct).

Importing and Exporting Packages

You import and export packages using Designer. Exporting allows you to export the package to a .zip file and save it to your hard drive. The .zip file can then be imported for use by another package.

Important:
Do not rename packages you export; the rename function is comparable to moving a package, and when you import the renamed package, you lose any triggers, connections, and notifications associated with this package.

For details about importing and exporting packages, see the *webMethods Service Development Help*.

Setting Package Dependencies

You set package dependencies if a given package needs services in another package to load before it can load. For example, any packages you create for SOPERA Adapter services should identify the adapter's package (WmSoperaAdapter) as a package dependency because they require services in WmSoperaAdapter to load first. Use the following guidelines:

- Set package dependencies from the adapter service package to the package containing the connection if you configure a connection in one package and the adapter services in another package. That is, the package that contains the connection should load before the adapter service package.

  When you set this package dependency, it ensures that if someone disables the connection package and then re-enables it, the adapter services will reload correctly.

- If both the connection and adapter services are in the same package, set this package to have a dependency on the WmSoperaAdapter package.

- In general, packages containing connections should have a dependency set to the adapter package itself. That is, the adapter service package should depend on the adapter connection package, which should depend on the adapter package. Similarly, if the adapter services are in the same package as the connections, the only dependency that you need to set is between the adapter connection package and the adapter package.

For more information about setting package dependencies, see the *webMethods Service Development Help*.

Controlling Group Access

To control which development group has access to which adapter services, use access control lists (ACLs). You can use ACLs to prevent one development group from inadvertently updating the work of another group, or to allow or deny access to services that are restricted to one group but not to others.

For general information about assigning and managing ACLs, see the *webMethods Service Development Help*.

4 Adapter Connections

Before Configuring or Managing Adapter Connections

To prepare to configure or manage an adapter connection

1. Install webMethods Integration Server and SOPERA Adapter on the same machine. For details, see "Installing webMethods SOPERA Adapter " on page 21.

2. Make sure you have webMethods administrator privileges so that you can access the SOPERA Adapter's administrative screens. For information about setting user privileges, see the *webMethods Integration Server Administrator's Guide* for your release.

3. Start Integration Server and Integration Server Administrator, if they are not already running.

4. Using Integration Server Administrator, make sure the SOPERA package is enabled. See "Enabling and Disabling Packages" on page 33 for instructions.

5. Using Software AG Designer, create a user-defined package to contain connections, if you have not already done so. See "Managing the Adapter Package" on page 32 for instructions.

Configuring Adapter Connections

SOPERA Adapter requires a connection to the SOPERA infrastructure whenever services from the SOPERA infrastructure are to be invoked. That is, whenever SOPERA Adapter acts as a client for a SOPERA infrastructure. A connection is also required when an Integration Server service is to be invoked through the adapter by any of the SOPERA participants.

When you configure SOPERA Adapter connections, you specify information that Integration Server uses to connect to a SOPERA participant. In most cases, you will need to configure at least two adapter connections: one for the adapter listener and one for all adapter services.

To configure an adapter connection

1. In Integration Server Administrator, select **Adapters > SOPERA Adapter**.

2. On the Connections screen, click **Configure New Connection**.

3. On the Connection Type screen, click **SOPERA Connection** as the connection type.

4. Complete the following fields in the SOPERA Adapter section:

Field	Description/Action
Package	The package in which to create the connection. You must create the package using Designer before you can specify it using this parameter.

Field	Description/Action
	For general information about creating packages, see the *webMethods Service Development Help*. **Note:** Configure the connection in a user-defined package rather than in the adapter's package. See "Package Management" on page 31 for other important considerations when creating packages for SOPERA Adapter.
Folder Name	The folder in which to create the connection.
Connection Name	The name of the new connection.

5. In the **Connection Properties** section, complete the following fields using the values from your configured SOPERA infrastructure:

Field	Description/Action
SOPERA Location	The location of the SOPERA infrastructure. The default value is `generic_location`.
SOPERA User	The SOPERA default user name for this adapter connection. For example, `SOPAdministrator`.
SOPERA Password	The password for the SOPERA default user name. **Note:** You must retype the password in the **Retype SOPERA Password** field.

6. In the **Connection Management Properties** section, complete the following fields to configure connection pool sizes:

Field	Description/Action
Enable Connection Pooling	Enables the connection to use connection pooling. Default: `true` For more information about connection pooling, see "Connection Pools" on page 12.
Minimum Pool Size	The minimum number of connection objects that remain in the connection pool at all times. When the adapter creates the pool, it creates this number of connections. Default: `1`

Field	Description/Action
Maximum Pool Size	The maximum number of connection objects that can exist in the connection pool. When the connection pool has reached its maximum number of connections, the adapter will reuse any inactive connections in the pool or, if all connections are active, it will wait for a connection to become available. Default: `10`
Pool Increment Size	If connection pooling is enabled, this field specifies the number of connections by which the pool will be incremented if connections are needed, up to the maximum pool size. Default: `1`
Block Timeout (msec)	If connection pooling is enabled, this field specifies the number of milliseconds that Integration Server will wait to obtain a connection with the SOPERA infrastructure before it times out and returns an error. Default: `1000`
Expire Timeout (msec)	If connection pooling is enabled, this field specifies the number of milliseconds that an inactive connection can remain in the pool before it is closed and removed from the pool. For example, to specify 10 seconds, specify `10000`. Enter `0` to specify no timeout. Default: `1000` **Note:** The adapter will never violate the **Minimum Connections** parameter. These connections remain in the pool regardless of how long they are inactive.
Startup Retry Count	The number of times that the system should attempt to initialize the connection pool at startup if the initial attempt fails. Default: `0`
Startup Backoff Timeout (sec)	The minimum number of connection objects that remain in the connection pool at all times. When the adapter creates the pool, it creates this number of connections. Default: `10`

7. Click **Save Connection**.

The connection that you configured appears on the adapter's Connections screen and in the Designer Package Navigator view.

By default, when you configure a connection, it is not enabled. For more information about enabling connections, see "Enabling Adapter Connections" on page 41.

Enabling Adapter Connections

A connection must be enabled before you can configure any adapter service using the connection, or before an adapter service can use the connection at run time. You enable adapter connections using Integration Server Administrator.

Note:
When you reload a package that contains enabled connections, the connections will automatically be enabled when the package reloads. If the package contains connections that are disabled, they will remain disabled when the package reloads.

To enable a connection

1. In Integration Server Administrator, select **Adapters > SOPERA Adapter**.

2. On the Connections screen, click **No** in the **Enabled** column for the connection you want to enable.

   Integration Server Administrator enables the adapter connection and displays **Yes** in the **Enabled** column.

Viewing Adapter Connections from Integration Server Administrator

You can view adapter connections and each connection's parameters from Integration Server Administrator or Designer.

To view the adapter connections using Integration Server Administrator

1. Make sure the connection is enabled. See "Enabling Adapter Connections" on page 41 for details.

2. In Integration Server Administrator, select **Adapters > SOPERA Adapter**.

   You can sort and filter the list of connections that appears on the Connections screen.

   - To sort information on the Connections screen, click the **Up** and **Down** arrows.
   - To filter the list of connections:

     1. On the Connections screen, click **Filter Connections**.

     2. Type the criterion by which you want to filter into the **Filter criteria** box. Filtering is based on the node name, not the connection alias. To locate all connections containing specific alphanumeric characters, use asterisks (*) as wildcards. For example, if you want to display all connections containing the string "abc", type `*abc*` in the **Filter criteria** box.

3. Click **Submit**. The Connections screen displays the connections that match the filter criteria.

4. To re-display all connections, click **Show All Connections**.

- The Connections screen appears, listing all the current connections. You can control the number of connections that are displayed on this screen. For more information, see "Controlling Pagination" on page 19.
- On the Connections screen, click the **View** icon for the connection you want to see.

  The View Connection screen displays the parameters for the connection. For descriptions of the connection parameters, see the table of parameters in "Configuring Adapter Connections" on page 38.

3. Click **Return to SOPERA Adapter Connections**to return to the Connections screen.

Viewing Adapter Connections in Designer

To view the parameters for a connection using Designer

1. Start Designer if it is not already running.

2. From the Designer Package Navigator view, open the package and folder in which the connection is located.

3. Double-click the connection you want to view.

4. The parameters for the connection appear on the **Connection Information** tab. For descriptions of the connection parameters, see "Configuring Adapter Connections" on page 38.

Disabling Adapter Connections

SOPERA Adapter connections must be disabled before you can edit or delete them. You disable adapter connections using Integration Server Administrator.

To disable a connection

1. In Integration Server Administrator, select **Adapters > SOPERA Adapter**.

2. On the Connections screen, click **Yes** in the **Enabled** column for the connection you want to disable.

   The adapter connection becomes disabled and you see a **No** in the **Enabled** column.

Editing Adapter Connections

If a connection parameter changes, or if you want to redefine parameters that a connection uses when connecting to a SOPERA infrastructure, you can update a connection's parameters using Integration Server Administrator.

To edit a connection

1. In Integration Server Administrator, select **Adapters > SOPERA Adapter**.

2. Make sure that the connection is disabled before editing it. See "Disabling Adapter Connections" on page 42 for instructions.

3. On the Connections screen, click the **Edit** icon for the connection you want to edit.

   The Edit Connection screen displays the current parameters for the connection. Update the connection's parameters by typing or selecting the values you want to specify.

   For descriptions of the connection parameters, see "Configuring Adapter Connections" on page 38.

4. Click **Save Changes** to save the connection and return to the Connections screen.

Copying Adapter Connections

You can copy an existing SOPERA Adapter connection to configure a new connection with the same or similar connection properties without having to retype all of the properties for the connection. You copy adapter connections using Integration Server Administrator.

To copy a connection

1. In Integration Server Administrator, select **Adapters > SOPERA Adapter**.

2. On the Connections screen, click the **Copy** icon for the connection you want to copy.

   The Copy Connection screen displays the current parameters for the connection you want to copy. Name the new connection, specify a package name and folder name, and edit any connection parameters as needed by typing or selecting the values you want to specify.

   Note:
   When you copy a connection, the new connection does not save the password of the original connection. You must enter and then retype the password before you can save the new connection.

   For descriptions of the connection parameters, see "Configuring Adapter Connections" on page 38.

3. Click **Save Connection** to save the connection and return to the Connections screen.

Deleting Adapter Connections

If you no longer want to use a particular SOPERA Adapter connection, you can delete it by following the instructions in this section. You delete adapter connections using Integration Server Administrator.

If you delete a connection, the adapter services or notifications that are defined to use the connection will no longer work. However, you can change which connection an adapter service uses. Therefore, if you delete a connection, you can assign a different connection to an adapter service and re-use the service. To do this, you use the built-in functionsetAdapterServiceNodeConnection. For more information, see "Changing the Connection Associated with an Adapter Service at Design Time" on page 13.

To delete a connection

1. In Integration Server Administrator, select **Adapters > SOPERA Adapter**.

2. Make sure that the connection is disabled before deleting. To disable the connection, click **Yes** in the **Enabled** column and click **OK** to confirm. The **Enabled** column now shows **No** (disabled) for the connection.

3. On the Connections screen, click the **Delete** icon for the connection you want to delete.

   Integration Server deletes the adapter connection.

5 Adapter Services

Before Configuring or Managing Adapter Services

To prepare to configure or manage an adapter service

1. Start your Integration Server and Integration Server Administrator, if they are not already running.

2. Make sure you have Integration Server administrator privileges so that you can access SOPERA Adapter administrative screens. For information about setting user privileges, see the *webMethods Integration Server Administrator's Guide* for your release.

3. Using Integration Server Administrator, make sure the WmSoperaAdapter package is enabled. For instructions, see "Enabling a Package" on page 33.

4. Using Integration Server Administrator, configure an adapter connection to use with the adapter service. For instructions, see "Configuring Adapter Connections" on page 38.

5. Start Designer if it is not already running.

6. Using Designer, create a package to contain adapter services, if you have not already done so.

   When you configure adapter services, define them in user-defined packages instead of in the WmSoperaAdapter package. For more information about managing packages for the adapter, see "Package Management" on page 31.

Using Adapter Services

The following table lists the task required to use adapter services.

Task	Use this tool
1. Configure an adapter connection. For instructions, see "Configuring Adapter Connections" on page 38.	Integration Server Administrator
2. Select the appropriate adapter service template and configure the adapter service. For instructions, see "Configuring Adapter Services" on page 47.	Designer
3. If you plan to use an Integration Server flow or Java service to invoke the adapter service, design the flow or Java service to use this adapter service. For information about flow and Java services, see the *webMethods Service Development Help*.	Designer
4. Manage the adapter service.	Integration Server Administrator and Designer

Configuring Adapter Services

Every invocation of a SOPERA service by other Integration Server services or Integration Server clients goes through an adapter service. The SOPERA Adapter provides the following adapter service templates:

- SOPERA Generic Consumer
- SOPERA Notification Provider
- SOPERA Callback Sender Provider

You configure adapter services in Designer.

SOPERA Generic Consumer

The SOPERA Generic Consumer adapter services perform the actual invocation of a SOPERA service. In general, you can use the SOPERA Adapter service consumer template to configure two kinds of service invocation adapter services:

- Adapter service tailored to a specific SOPERA service and operation.
- Adapter service that invokes any SOPERA service and operation.

Configuring a Service for a Specific SOPERA Service and Operation

This configuration is accomplished by selecting a specific SOPERA service and one of its operations that the adapter service will provide. In this case, one adapter service is required for every SOPERA service operation. In most cases, it is better to use one adapter service per SOPERA service. The SOPERA registry ensures that a valid SOPERA service with a valid operation is used. Additionally, these adapter services are more easily reused in flow services and business processes.

For this configuration, use the SOPERA Generic Consumer adapter service template to configure the parameters for the new service as described in the following procedure.

To configure an adapter service that invokes a specific SOPERA service

1. Start Designer.
2. Right-click the package in which the service should be contained and select **New > Adapter Service**.
3. Select the parent namespace and type a name for the adapter service.
4. Click **Next**.
5. Select **SOPERA Adapter** as the adapter type and click **Next**.

6. Select the appropriate **Adapter Connection Name** and click **Next**.

7. Select the SOPERA Generic Consumer adapter template and click **Finish**.

   The adapter service editor for the invocation service appears.

8. In the service editor, select the **Adapter Settings** tab at any time to confirm adapter service properties such as **Adapter Name**, **Adapter Connection Name**, and **Adapter Service Template**, as necessary.

9. In the service editor, select the **SOPERA Generic Consumer** tab and specify values in the fields, based on the type of consumer service you want to configure:

   - For request-response consumer services, see "Configuring Request-Response Consumer Services" on page 48.
   - For one-way consumer services, see "Configuring One-Way Consumer Services" on page 50.
   - For request-callback consumer services, see "Configuring Request-Callback Consumer Services" on page 51.

10. From the **File** menu, click **Save**.

Using the XML schema of the SOPERA operations retrieved from the registry, the SOPERA Adapter automatically creates Integration Server input and output document types adhering to the operation's schema.

When the adapter service is invoked, the data is automatically transformed into XML and passed to the SOPERA service. When the service returns the result, the XML is then transformed back into structured data elements that business processes, services, and clients can easily use inside Integration Server.

Configuring Request-Response Consumer Services

When you want to create a request-response consumer, you configure a request-response consumer service using the SOPERA Generic Consumer template. For the steps to configure the service, see "Configuring a Service for a Specific SOPERA Service and Operation" on page 47. The following table describes the required fields and the values you provide for a request-response consumer service:

Field	Description
Service name	The name and the path to the SOPERA service to invoke. For example, `{http://services.sopware.org/ExampleURI}ExampleService`
Operation name	The name of the operation the adapter service must perform. For example. Select `RequestResponseOperation`
Callback Operation Name	Leave empty.

Field	Description
Send request only	Do not select.
Interaction style	The way in which the adapter service interacts with the SOPERA infrastructure. When you select the request-response operation in the **Operation name** field, the value in this field changes automatically to `REQUEST_RESPONSE`.
Input document type	The fully qualified document type name for the input IS document. For example, `doc.myservice.input:inDoc` **Important:** Specify a unique name for the input document type. If you specify an existing document type name, the IS document type will not be generated correctly.
Output document type	The fully qualified document type name for the output IS document. For example, `doc.myservice.output:inDoc` **Important:** Specify a unique name for the output document type. If you specify an existing document type name, the IS document type will not be generated correctly.
Policy	The conditions under which the adapter service invokes a SOPERA service. SOPERA Adapter retrieves policies from the SOPERA Service Registry based on the location configured in the Connection Settings. For information about configuring the adapter connection settings, see "Configuring Adapter Connections" on page 38.
Blocking	Determines the invocation style of the adapter service: ■ When selected, the adapter service is configured as blocking and it waits for the actual service result. By default, blocking is selected. ■ When not selected, the adapter service is configured as non-blocking and returns a response message handler. This message handler can be passed to the pub.wmsopera.service:getNonBlockingServiceResponse utility service, using a flow service that gets the actual result. This allows a complex service to start as soon as all of its input parameters are available and to only block when the output of the service is needed.
Return the actual message object	Determines how the service provider returns the response message to the service consumer. ■ When not selected, the response message is transformed into Integration Server data structures. This is the default. ■ When selected, the response message returns as an actual message object. This is useful when a response message contains attachments or is very

Field	Description
	large, because an actual message object enables you to retrieve the attachments or the message content in a stream-based manner. **Note:** If the response contains attachments, then the response is returned as an object whether you select this option or not. Ensure that the adapter services are configured appropriately.

Configuring One-Way Consumer Services

When you want to create a one-way consumer, you configure a one-way consumer service using the SOPERA Generic Consumer template. For the steps to configure the service, see "Configuring a Service for a Specific SOPERA Service and Operation" on page 47. The following table describes the required fields and the values you provide for a one-way consumer service:

Field	Description
Service name	The name and the path to the SOPERA service to invoke. For example, `{http://services.sopware.org/ExampleURI}ExampleService`
Operation name	The name of the operation the adapter service must perform. Select `OneWayOperation`.
Callback Operation Name	Leave empty.
Send request only	Do not select.
Interaction style	The way in which the adapter service interacts with the SOPERA infrastructure. When you select the one-way operation in the **Operation name** field, the value in this field changes automatically to `ONEWAY`.
Input document type	The fully qualified document type name for the input IS document. For example, `doc.myservice.input:inDoc` **Important:** Specify a unique name for the input document type. If you specify an existing document type name, the IS document type will not be generated correctly.
Output document type	When the **Interaction style** is `ONEWAY`, no output document type is required or generated.

Field	Description
Policy	The conditions under which the adapter service invokes a SOPERA service. SOPERA Adapter retrieves policies from the SOPERA Service Registry based on the location configured in the Connection Settings. For information about configuring the adapter connection settings, see "Configuring Adapter Connections" on page 38.
Blocking	Do not select.
Return the actual message object	Do not select.

Configuring Request-Callback Consumer Services

When you want to create a request-callback consumer, you configure a request-callback consumer service using the SOPERA Generic Consumer template. For the steps to configure the service, see"Configuring a Service for a Specific SOPERA Service and Operation" on page 47. However, there are two ways two create a request-callback consumer as described in "Creating Service Consumers" on page 15. Based on the method you use, you provide different values in the **Operation name**, **Send request only**, and **Output document type** template fields. The values in the remaining fields are the same for both methods:

Field	Description
Service name	The name and the path to the SOPERA service to invoke. For example, `{http://services.sopware.org/ExampleURI}ExampleService`
Operation name	When you want to create an adapter consumer service that uses one of the request-callback operations, select the respective request-callback operation.
Callback Operation name	■ The name of the callback operation that corresponds to the request-callback operation selected in the **Operation name** field. ■ When the **Send request only** checkbox is selected, leave this field empty.
Send request only	■ When you want to create an adapter service that only submits a request and has a listener for receiving the callback, select this box. ■ When you want to receive the response as part of the consumer service, do not select this box.

Field	Description
Interaction style	When you select one of the request-callback operations in the **Operation name** field, the value in this field changes automatically to `REQUEST_CALLBACK`.
Input document type	The fully qualified document type name for the input IS document. For example, `doc.myservice.input:inDoc`
Output document type	■ When you want to create a consumer service that uses one of the request-callback operations, provide the fully qualified document type name for the output IS document. For example, `doc.myservice.output:inDoc`. In this case, the signature of the adapter service will contain only one output parameter: *responseMessageHandler*. You can use the *responseMessageHandler* parameter as input to the `pub.wmsopera.service:getNonBlockingServiceResponse` service that will wait and block for the actual result. The consumer service and the `getNonBlockingServiceResponse` service function together in much the same way as for a non-blocking request-response style. For more information about the `getNonBlockingServiceResponse` service, see pub:wmsopera.service:getNonBlockingServiceResponse. ■ When you want to create an adapter service that only submits a request, leave this field empty. In this case, the consumer service does not have any output parameters. You must configure an adapter notification to receive the callback. For information about how to configure a consumer adapter notification, see " SOPERA Callback Consumer" on page 73.
Policy	The conditions under which the adapter service invokes a SOPERA service. SOPERA Adapter retrieves policies from the SOPERA Service Registry based on the location configured in the Connection Settings. For information about configuring the adapter connection settings, see "Configuring Adapter Connections" on page 38.
Blocking	Do not select.
Return the actual message object:	Do not select.

Configuring a Service for Any SOPERA Service and Operation

In this configuration the SOPERA service and operation are parameters of the adapter service. The benefits of this approach are increased flexibility (run-time selection of SOPERA service is possible) and less configuration overhead (only one adapter service can be used for multiple SOPERA service invocation).

To configure a generic invocation adapter service, use the same procedure as described in the "Configuring a Service for a Specific SOPERA Service and Operation" on page 47 section, but leave the fields on the **SOPERA Generic Consumer** tab blank. Specify configuration values as input values for the adapter service parameters that you can view on the **Input/Output** tab of the adapter service template:

Input Parameters

serviceName	The name of the SOPERA service.
operationName	The name of the operation to perform for the adapter service.
policyId	The policy identifier for the SOPERA service.
requestMessageString	The request that the adapter service sends as an XML String.

Output Parameters

responseMessageString	The response message as an XML String, when the **Return the actual message object** field on the **SOPERA Generic Consumer** tab is set to `false`.
responseMessageObject	The response message as an Object, when the **Return the actual message object** field on the **SOPERA Generic Consumer** tab is set to `true`. When the response contains attachments, the response always returns as a `responseMessageObject`.
responseMessageHandler	The message handler passed to the pub.wmsopera.service:getNonBlockingServiceResponse utility service when the communication style is request-response non-blocking.

SOPERA Notification Provider

You use the SOPERA Notification Provider service template to create an adapter service that functions as a service provider when using the SOPERA notification communication style. For more information about the SOPERA notification style and service providers, see "Communication Styles Support" on page 15 and "Creating Service Providers" on page 16.

To create a SOPERA notification provider adapter service

1. Start Designer.

2. Right-click the package in which the service should be contained and select **New > Adapter Service**.

3. Select the parent namespace and type a name for the adapter service.

4. Click **Next**.

5. Select **SOPERA Adapter** as the adapter type and click **Next**.

6. Select the appropriate **Adapter Connection Name** and click **Next**.

7. Select the SOPERA Notification Provider adapter template and click **Finish**.

   The adapter service editor for the notification service appears.

8. In the service editor, select the **Adapter Settings** tab at any time to confirm adapter service properties such as **Adapter Name**, **Adapter Connection Name**, and **Adapter Service Template**, as necessary.

9. In the service editor, select the **SOPERA Notification Provider** tab and specify values in the following fields based on the method for creating a provider that you want to use:

   a. Using an existing IS document type:

Field	Description
Service name	The name and the path to the new SOPERA notification service. For example, `{http://services.sopware.org/ExampleURI}ExampleService`
Operation name	The name of the operation for the SOPERA service. For example, `notify`
Input document type	The fully qualified document type name for an existing IS document. For example, `doc.myservice.output:notificationDoc`
Use existing SOPERA service	Do not select.

   Based on the IS document type that you specify, SOPERA Adapter creates the notification operation in the service registry.

   b. Using the SDX-first approach:

Field	Description
Service name	The name of the SOPERA service. For example, `{http://services.sopware.org/ExampleURI}ExampleService`

Field	Description
Operation name	The name of one of the notification operations for the SOPERA service. For example, `notifyForArrival`
Input document type	The fully qualified document type name for the IS document that will be generated. For example, `doc.myservice.input:notifyDoc`
Use existing SOPERA service:	Select this box.

SOPERA Adapter creates an IS document type that matches the signature of the notification operation in the existing SOPERA service.

10. On the JMS Provider Settings tab, provide values for the following fields:

Field	Description
JMS Provider	The name of the JMS MOM Provider. Select `ActiveMQ`.
JMS Communication Style	The JMS communication style. Select `topic`.
JNDI URL	Location of the JNDI. For example, `tcp://dewm:61616`

11. From the **File** menu, click **Save**.

SOPERA Callback Sender Provider

The SOPERA Callback Sender Provider template is used in one of the methods for creating a request-callback provider. With this method you first create an adapter notification that receives the request and publishes a document. For information about how to create the callback provider adapter notification, see " SOPERA Generic Provider" on page 69. Then using the SOPERA Callback Sender Provider template, you create an adapter service that sends a response to the request.

The adapter notification publishes a document that contains the *callContextKey* parameter which the adapter service uses to correlate the callback with the request. You can also store the *callContextKey* in a database during a long running operation, wait for input from an external system, retrieve the *callContextKey* from the database, and invoke the consumer callback operation.

To create a Callback Sender Provider adapter service

1. Start Designer.

2. Right-click the package in which the service should be contained and select **New > Adapter Service**.

3. Select the parent namespace and type a name for the adapter service.
4. Click **Next**.
5. Select **SOPERA Adapter** as the adapter type and click **Next**.
6. Select the appropriate **Adapter Connection Name** and click **Next**.
7. Select the SOPERA Callback Sender Provider adapter template and click **Finish**.

   The adapter service editor for the callback service appears.
8. In the service editor, select the **Adapter Settings** tab at any time to confirm adapter service properties such as **Adapter Name**, **Adapter Connection Name**, and **Adapter Service Template**, as necessary.
9. In the service editor, select the **SOPERA Callback Sender Provider** tab and specify the following values:

Field	Description
Service name	The name and path to an existing SOPERA service. For example, `{http://services.sopware.org/ExampleURI}UpperCallback`
Operation name	The name of the SOPERA service operation. For example, `getUpper`.
Callback Operation name	The name of the callback operation corresponding to the one specified in the **Operation name** field. For example, `getUpperResponse`.
Interaction style	When you select one of the request-callback operations in the **Operation name** field, the value in this field changes automatically to `REQUEST_CALLBACK`.
Output document type	The fully qualified document type name for the IS Document that will be generated. This document is the response to the callback operation. For example, `doc.myservice.output:callbackDoc`

10. From the **File** menu, click **Save**.

Testing Adapter Services

You use Designer to test adapter services.

For more information about testing and debugging services, see the *webMethods Service Development Help*.

To test adapter services

1. In Designer, expand the package and folder that contain the service you want to test.
2. Double-click the service you want to test.

   Designer displays the configured service in the service template's adapter service editor.
3. Select **Run > Run As > Run Service**.
4. For every service input field, you will be prompted to enter an input value. Enter a value for each input field and then click **OK**.
5. Click the **Service Result** tab to view the output from this service.

Viewing Adapter Services

You use Designer to view adapter services.

To view adapter services

1. In Designer, expand the package and folder that contain the service you want to view.
2. Double-click the service you want to view.

   Designer displays the configured service in the service template's adapter service editor.

Editing Adapter Services

You use Designer to edit adapter services.

To edit an adapter service

1. In Designer, browse to and open the adapter service that you want to edit.
2. Double-click the service that you want to edit.

   Designer displays the adapter service in the service template's Adapter Service Editor.
3. Do one of the following:
   - If you have the VCS Integration feature enabled, right-click the service and select **Check Out**.
   - If you do not have the VCS Integration feature enabled, right-click the service and select **Lock for Edit**.

- If you are using the local service development feature, from the **Team** menu in Designer, select the appropriate option to check out the service. The options available in the **Team** menu depend on the VCS client that you use.

4. Modify the values for the adapter service's parameters as needed. For detailed descriptions of the service's parameters, see the section on configuring a service for the specific type of service you want to edit.

5. After you complete your modifications, save the service and do one of the following:

   - If you have the VCS Integration feature enabled, right-click the service and select **Check In**. Enter a check-in comment and click **OK**.
   - If you do not have the VCS Integration feature enabled, right-click the service and select **Unlock**.
   - If you are using the local service development feature, from the **Team** menu in Designer, select the appropriate option to check in the service. The options available in the **Team** menu depend on the VCS client that you use.

6. Save the service.

Deleting Adapter Services

You use Designer to delete adapter services.

To delete adapter services

1. In Designer, expand the package and folder that contain the service you want to delete.

2. Right-click the service you want to delete and then select **Delete**.

Automatic Data Validation and Reloading Adapter Values

You can enable SOPERA Adapter to reload and validate user-defined data for adapter services at design time in Designer.

Disabling Data Validation

The automatic data validation for adapter services is enabled by default in Designer. When the option is selected, Designer always validates values for adapter services for all webMethods WmART-based adapters installed on Integration Server.

To disable automatic data validation for all adapter services

1. Start Designer.

2. Select **Window > Preferences > Software AG > Service Development > Adapter Service/Notification Editor**.

3. Disable the **Automatic data validation** option.

4. Click **OK**.

For more information about the adapter service/notification editor and other Designer menu options and toolbar icons, see *webMethods Service Development Help*.

Reloading Adapter Values

The option to always reload values for adapter services is disabled by default in Designer. If you enable that option, Designer will always reload values for all webMethods WmART-based adapters installed on Integration Server.

To reload the adapter values for all adapter services

1. Start Designer.

2. Select **Window > Preferences > Software AG > Service Development > Adapter Service/Notification Editor**.

3. Enable the **Automatic polling of adapter metadata** option.

4. Click **OK**.

For more information about the adapter service/notification editor and other Designer menu options and toolbar icons, see the *webMethods Service Development Help*.

6 Adapter Notifications

Listeners

This section describes how to create, modify, and delete listeners.

Before You Configure New Listeners

To prepare to configure a new listener

1. Make sure that you have webMethods administrator privileges so that you can access webMethods SOPERA Adapter's administrative screens. For information about setting user privileges, see the *webMethods Integration Server Administrator's Guide* for your release.

2. Start Integration Server and Integration Server Administrator, if they are not already running.

3. Using Integration Server Administrator, make sure that the WmSoperaAdapter package is enabled. To verify the status of the WmSoperaAdapter package, see "Enabling and Disabling Packages" on page 33.

4. Using Designer, create a user-defined package to contain the listener, if you have not already done so. For more information about managing packages, see "Package Management" on page 31 for details.

Configuring Listeners

SOPERA Adapter requires a listener to listen for inbound requests from the SOPERA infrastructure. Use the following procedure to create a listener on SOPERA Adapter to respond to requests issued by the SOPERA infrastructure.

To create a listener on SOPERA Adapter

1. In Integration Server Administrator, select **Adapters > SOPERA Adapter**.

2. In the **SOPERA Adapter** menu, click **Listeners**.

3. Select **Configure new listener** and then select **SOPERA invocation listener** from the list of available listener types.

4. Complete the following fields on the Configure Listener Type screen:

Field	Description/Action
Package	The package in which to create the listener. You must create the package using Designer before you can specify it using this parameter. For general

Field	Description/Action
	information about creating packages, see the *webMethods Service Development Help*. **Note:** Configure the listener in a user-defined package rather than in the adapter's package. See "Package Management" on page 31 for other important considerations when creating packages for SOPERA Adapter.
Folder Name	The folder in which to create the listener.
Listener Name	The name of the new listener.
Connection Name	The name of the connection to associate with the new listener. From the list of connections, select an appropriate connection that you created on the Configure Connections Type screen.
Retry Limit	The number of times the adapter tries to reconnect if the adapter fails to connect, or loses connection with the SOPERA infrastructure. Default: 5
Retry Backoff Timeout	The number of seconds that elapse between each of the retries specified in the retry limit. Default: 10

5. Click **Save Listener**.

The adapter listener is created. The listener name is prefixed by the folder name and is separated by a colon. For example: If the folder name is "Folder1" and the listener name is "Listener1", then the listener name in the Listeners screen will be "Folder1:Listener1".

Enabling Listeners

After you have configured notifications, you must enable the listener so that the associated notifications will communicate appropriately with the listener at run time. You enable the listeners using Integration Server Administrator.

The **Status** column indicates the readiness of the listener. If the status is **Succeeded**, the listener is ready to be enabled. If the status is **Failed**, an error occurred during startup. If an error occurs during startup, the state will not change to **Enabled** when refreshing the page. Errors at this stage typically indicate a problem with either the listener configuration or the network. Review the listener settings and check the network.

Note:

When you reload a package that contains enabled listeners, the listeners will be enabled automatically when the package reloads. If the package contains disabled listeners, they will remain disabled when the package reloads.

To enable a listener

1. In Integration Server Administrator, select **Adapters > SOPERA Adapter**.
2. In the **SOPERA Adapter** menu, click **Listeners**.
3. On the Listeners screen, select **Enabled** from the list in the **State** field. Integration Server Administrator enables the listener.
4. The state changes to **Pending enabled**. After refreshing the Listeners page, you should see the state changed to **Enabled**.

   After a listener is enabled, a connection exists between SOPERA Adapter and the SOPERA infrastructure.

   Tip:
   The **Enable all suspended** link helps you change the state quickly for multiple listeners.

Viewing Listeners

You can view listeners and each listener's parameters from Integration Server Administrator. You can also view the notification order of a listener.

To view listeners

1. In Integration Server Administrator, select **Adapters > SOPERA Adapter**.
2. In the **SOPERA Adapter** menu, click **Listeners**. The Listeners screen appears, listing all of the current listeners.
3. To view a listener's parameters:

   a. On the Listeners screen, click the **View** icon for the listener that you want to see.

   b. The View Listener screen displays the parameters for the listener.

4. Click **View Notification Order** to view the notification order of a listener.

   The View Listener screen displays the order of the notifications for the listener. To change the notification order for the listener, refer to the procedure in "Editing the Notification Order of Listeners" on page 65 for detailed instructions.

5. Click **Return to SOPERA Adapter Listeners** to return to the Listeners screen.

Editing Listeners

You use Integration Server Administrator to edit the listener in the following situations:

- If you need to select a newly configured connection, or if you need to change any listener properties you can update the listener parameters.
- If you need to change the order of the notifications that are associated with the listener.

To edit a listener

1. In Integration Server Administrator, select **Adapters > SOPERA Adapter**.
2. In the **SOPERA Adapter** menu, click **Listeners**.
3. On the Listeners screen, make sure that the listener is disabled before editing. To disable the listener, see "Disabling Listeners" on page 67.
4. On the Listeners screen, click the **Edit** icon for the listener that you want to edit.

   The Edit Listener screen displays the current parameters for the listener. Update the listener's parameters by typing or selecting the values you want to specify.

   For descriptions of the listener parameters, see "Configuring Listeners" on page 62.
5. Click **Save Changes** to save the listener and return to the Listeners screen.

Editing the Notification Order of Listeners

To edit the notification order of a listener

1. In Integration Server Administrator, select **Adapters > SOPERA Adapter**.
2. In the **SOPERA Adapter** menu, click **Listeners**.
3. On the Listeners screen, make sure that the listener is disabled before editing. To disable the listener, see "Disabling Listeners" on page 67 for details.
4. On the Listeners screen, click the **Edit** icon for the listener that you want to edit.
5. On the Edit Listener screen, click **Edit Notification Order**.

6. On the Edit Notification Order screen, use the **Up** and **Down** buttons to determine the processing order in which SOPERA Adapter invokes the notifications.

   Note:
   For better processing results, arrange your notifications from ascending to descending order, starting with the most detailed notifications to the least detailed notifications.

7. Click **Save Changes**.

8. Click **Return to Edit Listeners** to return to the Edit Listener screen.

Copying Listeners

You can copy an existing listener to create a new listener with the same or similar properties without having to type or specify all properties for the listener. You copy adapter listeners using Integration Server Administrator.

To copy a listener

1. In Integration Server Administrator, select **Adapters > SOPERA Adapter**.

2. In the **SOPERA Adapter** menu, click **Listeners**.

3. On the Listeners screen, click the **Copy** icon for the listener that you want to copy.

   The Copy Listener screen displays the current parameters for the listener that you want to copy. Name the new listener and edit any listener parameters as needed by typing or selecting the values you want to specify.

   For descriptions of the listener parameters, see "Configuring Listeners" on page 62.

4. Click **Save Changes** to save the listener and return to the Listeners screen.

Deleting Listeners

If you no longer want to use a listener, use the following instructions to delete the listener. You use Integration Server Administrator to delete listeners.

Important:
Ensure that there are no notifications attached to the listener that you delete. You cannot change which listener a notification uses after the notification is configured. However, you can change the parameters for an existing listener. For instructions, see "Editing Listeners" on page 65.

To delete a listener

1. In Integration Server Administrator, select **Adapters > SOPERA Adapter**.

2. In the **SOPERA Adapter** menu, click **Listeners**.

3. On the Listeners screen, make sure that the listener is disabled before deleting it. To disable the listener, see "Disabling Listeners" on page 67 for details.

4. On the Listeners screen, click the **Delete** icon for the listener you want to delete.

   Integration Server deletes the listener.

Suspending Listeners

You can suspend listeners for an indefinite period of time. Suspended listeners cannot be edited or deleted.

Important:
Suspending listeners for SOPERA Adapter has the same effect as disabling them. For more information about disabling listeners, see "Disabling Listeners" on page 67.

To suspend listeners

1. In Integration Server Administrator, select **Adapters > SOPERA Adapter**.

2. In the **SOPERA Adapter** menu, click **Listeners**.

3. Click **Suspend all enabled** to change the state to suspended for multiple listeners.

   When you suspend a listener, the action might not take effect right away. You might have to wait as long as the time specified in the Timeout parameter for the listener. If one or more messages appear on the queue within that time interval, the adapter may receive and process the first message.

Disabling Listeners

Listeners must be disabled before you can edit or delete them. You disable listeners using Integration Server Administrator.

To disable a listener

1. In Integration Server Administrator, select **Adapters > SOPERA Adapter**.

2. In the **SOPERA Adapter** menu, click **Listeners**.

3. On the Listeners screen, select **Disabled** from the list in the **State** field. Integration Server Administrator disables the listener.

When you disable a listener, the action might not take effect right away. You might have to wait as long as the time specified in the Timeout parameter for the listener. If one or more messages appear on the queue within that time interval, the adapter may receive and process the first message.

Listener Notifications

This section describes how to create, modify, and delete listeners notifications.

To prepare to configure a new listener notification

1. Make sure that you have webMethods administrator privileges so that you can access SOPERA Adapter's administrative screens. For information about setting user privileges, see the *webMethods Integration Server Administrator's Guide* for your release.

2. Start Integration Server and Integration Server Administrator, if they are not already running.

3. Using Integration Server Administrator, make sure that the WmSoperaAdapter package is enabled. To verify the status of the WmSoperaAdapter package, see "Enabling and Disabling Packages" on page 33.

4. Configure a listener using Integration Server Administrator. For more information on how to configure a new listener, see "Configuring Listeners" on page 62.

5. Using Designer, create a user-defined package to contain the listener, if you have not already done so. For more information about managing packages, see "Package Management" on page 31 for details.

Dependencies for Listener Notifications

The following table lists other objects you must configure or tasks you must complete to use listener notifications:

Task	Use this tool
1. Configure an adapter connection. For details, see "Configuring Adapter Connections" on page 38.	Integration Server Administrator
2. Configure an adapter listener. For details, see "Configuring Listeners" on page 62.	Integration Server Administrator
3. Configure the notification. For instructions on how to configure notifications, see "Configuring Listener Notifications" on page 69.	Designer
4. Enable the notifications. For instructions on how to enable notifications, see "Enabling Listener Notifications" on page 76.	Integration Server Administrator

Configuring Listener Notifications

You configure listener notifications using Designer. SOPERA Adapter provides the following adapter notification templates that you use to configure listener notifications:

- SOPERA Generic Provider
- SOPERA Notification Consumer
- SOPERA Callback Consumer
- SOPERA Request-Callback Provider

SOPERA Generic Provider

The following procedure describes the steps to configure a SOPERA Generic Provider listener notification using the SOPERA Generic Provider template. You use this type of listener notification to create:

- Request-response service providers
- One-way service providers
- Request-callback service providers that consist of a listener notification that calls an IS service and returns the output of that service as the callback.

To configure a Generic Provider listener notification

1. Start Designer.
2. Right-click the package in which the notification should be contained and select **New > Adapter Notification**.
3. Select the parent namespace and type a name for the adapter notification.
4. Click **Next**.
5. Select **SOPERA Adapter** as the adapter type and click **Next**.
6. Select the SOPERA Generic Provider notification template and click **Next**.
7. Select the appropriate **Notification Listener Name** and click **Next**.
8. Click **Finish**.

   Designer creates a listener notification and a publishable document type, and the editor for the adapter notification appears.

9. In the editor for adapter notifications, select the **Adapter Settings** tab to confirm notification properties such as the **Adapter Name**, **Adapter Listener Name**, and **Adapter Notification Template** as necessary.

10. In the editor for adapter notifications, select the **SOPERA Generic Provider** tab and specify the following fields:

    a. To create a one-way or request-response service provider:

Field	Description
SOPERA Service	The name of the SOPERA service under which the specified Integration Server service will be registered. For example: `{http://service.sopware.org/ExampleURI}ExampleService`
SOPERA Operation Name	The name of the operation in the specified SOPERA service. For example: `NotificationFS`
Callback operation name	Leave empty. Used only in the request-callback communication style.
Integration Server Service	■ To use the SDX-first approach, specify the qualified name of the IS service that will be generated. **Note:** In this case SOPERA Adapter creates both the IS service and its corresponding IS document types. ■ To use the IS document-first approach, specify the qualified name of the Integration Server service that will be exposed through the SOPERA infrastructure. For example: `test.string:concat` **Note:** The input and output parameters of the specified Integration Server service must be document references. The input parameter is required, but the output parameter is optional. If no output parameter is found, the adapter creates a `ONEWAY` operation.
Use Existing SOPERA Service	■ To use the SDX-first approach, select this check box to use an existing SOPERA service specification. The SOPERA service and operation will be loaded from the SOPERA Service Registry. You can then select the service and operation you require in the **SOPERA Service Name** and **SOPERA Operation Name** fields. **Important:**

Field	Description
	When this option is selected, the Integration Server service will overwrite the current provider definition stored in the registry. ■ Do not select when you want to generate the SDX from an existing IS document type. In this case, you must specify a name for the SOPERA service and its operation in the **SOPERA Service Name** and **SOPERA Operation Name** fields respectively.
Is Req-callback operation	Do not select. Used only in the request-callback communication style.
Use HTTPS	Select this check box if you want to use HTTPS as the underlying transport for this provider. The HTTP and HTTPS ports are configured in the SOPERA Adapter configuration file. See "Set SOPERA Specific Properties in the SOPERA Adapter Configuration File" on page 24 for details.
Use Plain XML	When selected, the service description will not contain any of the data structure given in the signature. **Note:** Selecting this option works for services with one *requestMessage* input parameter and one *responseMessage* output parameter, and is recommended when conversion to IData is not necessary or should be avoided for performance reasons.

b. To create a request-callback service provider, you specify the same values as for request-response and one-way service providers, except for the following fields:

Field	Description
Callback Operation name	Specify a callback operation that corresponds to the operation specified in the **Operation name** field. For example, if the value in the **Operation name** field is `inOut`, the value in this field could be `inOutResponse`.
Is Req-callback operation	Select this check box.

11. From the **File** menu, select **Save**.

SOPERA Notification Consumer

You use the SOPERA Notification Consumer template to create service consumers when using the SOPERA Notification communication style.

Note:SOPERA Adapter supports creating only one SOPERA Notification Consumer for a given SOPERA service using the same adapter connection.

To create a SOPERA Notification Consumer listener notification

1. Start Designer.
2. Right-click the package in which the notification should be contained and select **New > Adapter Notification**.
3. Select the parent namespace and type a name for the adapter notification.
4. Click **Next**.
5. Select **SOPERA Adapter** as the adapter type and click **Next**.
6. Select the SOPERA Notification Consumer notification template and click **Next**.
7. Select the appropriate **Notification Listener Name** and click **Next**.
8. Click **Finish**.

   Designer creates a listener notification and a publishable document type, and the editor for the adapter notification appears.
9. In the editor for adapter notifications, select the **Adapter Settings** tab to confirm notification properties such as the **Adapter Name**, **Adapter Listener Name**, and **Adapter Notification Template** as necessary.
10. In the editor for adapter notifications, select the **SOPERA Notification Consumer** tab and specify the following fields:

Field	Description
SOPERA Service	The name of the SOPERA service under which the specified Integration Server service will be registered. For example: `{http://service.sopware.org/ExampleURI}ExampleService`
SOPERA service operation	The name of the operation in the specified SOPERA service. For example: `notify`
Policy	The conditions under which the adapter service invokes a SOPERA service. Policies are retrieved from the SOPERA Service Registry based on the location configured in the Connection Settings. For information

Field	Description
	about configuring the adapter connection settings, see "Configuring Adapter Connections" on page 38.
Integration Server Service	■ The qualified name of the IS service that will be generated. The generated IS service has only an input parameter. ■ Leave empty, when the **Publish IS Document** check box is selected.
Publish IS document	■ Select this check box when you want the service consumer to publish an IS document. ■ Do not select when you want the service consumer to invoke an IS service.
Input document type	■ When **Publish IS document** is selected, provide a name for the IS document type that will be generated. SOPERA Adapter generates the specified IS document type and adds a reference to the IS document in the publish document for the listener notification. ■ When **Publish IS document** is unselected, leave this field empty.
Use plain XML	When selected, the response from the provider is returned as XML instead of an IS document type.

11. From the **File** menu, select **Save**.

Note:
When the template is saved, the existing flow service will be deleted and an empty flow service will be created to match the signature of the service.

SOPERA Callback Consumer

You use the SOPERA Callback Consumer template to receive notification for the callback of a request-callback service. For information about creating a request only adapter service, see "Configuring Request-Callback Consumer Services" on page 51.

To create a SOPERA Callback Consumer listener notification

1. Start Designer.
2. Right-click the package in which the notification should be contained and select **New > Adapter Notification**.
3. Select the parent namespace and type a name for the adapter notification.
4. Click **Next**.

5. Select **SOPERA Adapter** as the adapter type and click **Next**.

6. Select the SOPERA Callback Consumer notification template and click **Next**.

7. Select the appropriate **Notification Listener Name** and click **Next**.

8. Click **Finish**.

   Designer creates a listener notification and a publishable document type, and the editor for the adapter notification appears.

9. In the editor for adapter notifications, select the **Adapter Settings** tab to confirm notification properties such as the **Adapter Name**, **Adapter Listener Name**, and **Adapter Notification Template** as necessary.

10. In the editor for adapter notifications, select the **SOPERA Callback Consumer** tab and specify the following fields:

Field	Description
SOPERA Service	The name of the SOPERA service. For example: `{http://service.sopware.org/ExampleURI}ExampleService`
SOPERA service operation	The name of the operation in the specified SOPERA service. For example: `inOut`
SOPERA service callback operation	Changes automatically to a callback operation that corresponds to the operation specified in the **Operation name** field. For example, if the value in the **Operation name** field is `inOut`, the value in this field could be `inOutResponse`.
Output document type	The fully qualified document type name for the output IS document. For example, `doc.myservice.output:outDoc`. SOPERA Adapter generates the output IS document type and adds a reference to the IS document in the listener notification publish document.
Policy	The conditions under which the adapter service invokes a SOPERA service. Policies are retrieved from the SOPERA Service Registry based on the location configured in the Connection Settings. For information about configuring the adapter connection settings, see "Configuring Adapter Connections" on page 38.
Use plain XML	When selected, the service description will not contain any of the data structure given in the signature.

11. From the **File** menu, select **Save**.

SOPERA Request-Callback Provider

You use the SOPERA Request-callback Provider template in one of the methods for creating a request-callback provider. With this method you first create an adapter notification that receives the request and publishes a document. Then using the SOPERA Callback Sender Provider template, you create an adapter service that sends a response to the request as described in "SOPERA Callback Sender Provider" on page 55. The following procedure describes how to create a request-callback provider listener notification.

To create a SOPERA request-callback provider listener notification

1. Start Designer.
2. Right-click the package in which the notification should be contained and select **New > Adapter Notification**.
3. Select the parent namespace and type a name for the adapter notification.
4. Click **Next**.
5. Select **SOPERA Adapter** as the adapter type and click **Next**.
6. Select the SOPERA Request-callback Provider notification template and click **Next**.
7. Select the appropriate **Notification Listener Name** and click **Next**.
8. Click **Finish**.

   Designer creates a listener notification and a publishable document type, and the editor for the adapter notification appears.
9. In the editor for adapter notifications, select the **Adapter Settings** tab to confirm notification properties such as the **Adapter Name**, **Adapter Listener Name**, and **Adapter Notification Template** as necessary.
10. In the editor for adapter notifications, select the **SOPERA Request-callback Provider** tab and specify the following fields:

Note:
This type of listener notification supports only the SDX-first approach. As a result, the services and their operations are populated on loading the listener notification.

Field	Description
SOPERA Service	The name of the SOPERA service. For example: `{http://service.sopware.org/ExampleURI}ExampleService`
SOPERA service operation	The name of the operation in the specified SOPERA service. For example: `getUpper`

Field	Description
Input document type	The fully qualified document type name for the input IS document. For example, `doc.myservice.input:inDoc`. **Important:** Specify a unique name for the input document type. If you specify an existing document type name, the IS document type will not be generated correctly.
Use HTTPS	Select this check box if you want to offer Integration Server services via HTTPS. The HTTP and HTTPS ports are configured in the SOPERA Adapter configuration file. See "Set SOPERA Specific Properties in the SOPERA Adapter Configuration File" on page 24 for details.

11. From the **File** menu, select **Save**.

Enabling Listener Notifications

After you configure a listener notification, you need to enable it using Integration Server Administrator.

To enable a listener notification

1. In Integration Server Administrator, select **Adapters > SOPERA Adapter**.
2. In the **SOPERA Adapter** menu, select **Listener Notifications**.
3. On the Listener Notifications screen, click **No** in the **Enabled** column for the listener notification you want to enable.

   The Integration Server Administrator enables the listener notification and displays a **Yes** in the **Enabled** column.

Testing Listener Notifications

You can test listener notifications to ensure that you have configured them correctly.

To test listener notifications

1. Configure a listener using Integration Server Administrator. For instructions to configure a listener, see "Configuring Listeners" on page 62.
2. Configure a listener notification using Designer. For instructions to configure a notification, see "Configuring Listener Notifications" on page 69.

3. Enable the listener notification using Integration Server Administrator. For instructions to enable a listener notification, see "Enabling Listener Notifications" on page 76.

4. Enable the listener using Integration Server Administrator. For instructions to enable a listener, see "Enabling Listeners" on page 63.

5. On your SOPERA infrastructure, invoke the service registered when creating the listener notification.

Viewing Listener Notifications

You use Integration Server Administrator or Designer to view listener notifications.

Viewing Listener Notifications Using Integration Server Administrator

To view listener notifications using Integration Server Administrator

1. In Integration Server Administrator, select **Adapters > SOPERA Adapter**.

2. In the **SOPERA Adapter** menu, select **Listener Notifications**.

   The Listener Notifications screen appears, listing all the listener notifications.

Viewing Listener Notifications Using Designer

To view a listener notification using Designer

1. In Designer, expand the package and folder that contain the listener notification you want to view.

2. Click the listener notification that you want to view.

   Designer displays the configured listener notification in the adapter's adapter notification editor.

Editing Listener Notifications

You use Designer to edit listener notifications. When editing the listener notification, you can also edit the publishable document type associated with the asynchronous listener notifications.

Listener notifications must be disabled before you can edit or delete them.

To edit a listener notification

1. In Designer, expand the package and folder that contain the listener notification you want to edit.

2. Select the listener notification you want to edit.

   The adapter's adapter notification editor displays details about the configured listener notification.

3. Do one of the following:

   - If you have the VCS Integration feature enabled, right-click the notification and select **Check Out**.
   - If you do not have the VCS Integration feature enabled, right-click the notification and select **Lock for Edit**.
   - If you are using the local service development feature, from the **Team** menu in Designer, select the appropriate option to check out the notification. The options available in the **Team** menu depend on the VCS client that you use.

4. Modify the values for the listener notification's parameters as needed. For detailed descriptions of the listener notification's parameters, see "Configuring Listener Notifications" on page 69.

   Note:
   Because listener notifications inherently depend on listeners, you cannot change a listener for a listener notification after you configure it.

5. After you complete your modifications, save the notification and do one of the following:

   - If you have the VCS Integration feature enabled, right-click the notification and select **Check In**. Enter a check-in comment and click **OK**.
   - If you do not have the VCS Integration feature enabled, right-click the notification and select **Unlock**.
   - If you are using the local service development feature, from the **Team** menu in Designer, select the appropriate option to check in the notification. The options available in the **Team** menu depend on the VCS client that you use.

Deleting Listener Notifications

If you no longer want to use a particular SOPERA Adapter listener notification, you can delete it by following the instructions in this section. You delete listener notifications using Designer. Listener notifications must be disabled before you can edit or delete them.

To delete a listener notification

1. In Designer, expand the package and folder that contains the listener notification you want to delete.

2. Right-click the listener notification and click **Delete**.

Disabling Listener Notifications

You disable listener notifications using Integration Server Administrator.

To disable a listener notification

1. In Integration Server Administrator, select **Adapters > SOPERA Adapter**.
2. In the **SOPERA Adapter** menu, select **Listener Notifications**.

   The Listener Notifications screen appears, listing all the listener notifications.
3. On the Listener Notifications screen, click **Yes** in the **Enabled** column for the listener notification you want to disable.

   The listener notification becomes disabled and **No** displays in the **Enabled** column

7 Integration Scenarios

Calling SOPERA Services from Integration Server

Integration Server is registered as one of the participants of the SOPERA Service Backbone (SBB) and it can invoke any service in the infrastructure that is provided by other SBB participants. SOPERA Adapter creates an adapter service for Request-response, One-way, and Request-callback services, which can be used in any Integration Server flows. SOPERA Adapter creates a notification for the SOPERA notification style, which can be configured to perform any action on Integration Server.

Calling Integration Server Services from the SOPERA Infrastructure

You call Integration Server services from SOPERA when you want to:

- Offer custom Integration Server services (implemented, for example, as flow or Java services)
- Offer Integration Server adapter functionality
- Implement existing SOPERA service specification
- Offer built-in Integration Server services

> **Note:**
> Offering built-in Integration Server services is not recommended because built-in Integration Server services are rather low-level, technical services.

Generating the Certificate for Secure Transport and Incoming Call Authentication

When you want to expose any IS service using the HTTPS protocol or authenticate any incoming calls, a keystore file should be used.

1. To export the certificate from Higgins, use this command:

```
> keytool -exportcert -keystore ./keystore.jks -alias client -file
mycert.certPassword: atleast8 (default)
```

2. To import the certificate into the client keystore, use this command:

```
> keytool -importcert -alias client -file mycert.cert -keystore keystore.jksPassword:
 atleast8 (default)
```

8 Adapter Logging

Adapter Logging Levels

SOPERA Adapter uses Integration Server logging mechanism to log messages. You can configure and view Integration Server logs to monitor and troubleshoot SOPERA Adapter. For detailed information about logging in Integration Server, including instructions for configuring and viewing the different kinds of logs supported by the server, see the *webMethods Audit Logging Guide* for your release.

Viewing the Logging Information for the Adapter

To access the adapter's logging information

1. From Integration Server Administrator screen, select **Settings > Logging**.

   The Logging Settings screen appears. The **Loggers** section has **Adapters** included in the **Facility** section.

2. Expand the **Adapters** tree to see a list of all installed adapters with their code number and adapter description, along with the logging level.

Configuring Adapter Logging Levels

Beginning with Integration Server 7.1, you can configure different logging levels for SOPERA Adapter.

To change logging settings for the adapter

1. From Integration Server Administrator screen, select **Settings > Logging**.

   The Logging Settings screen appears.

2. From the Logging Settings screen, select **Edit Logging Settings**.

3. Select the required **Level of logging** for SOPERA Adapter.

4. Click **Save Changes**.

For complete information about specifying the amount and type of information to include in the log, see the *webMethods Audit Logging Guide* for your release.

SOPERA Message Logging

You can also configure how the SBB Library generates logging output. The SOPERA Adapter SBB Library uses the standard Java logging mechanism provided by the java.util.logging package.

To enable logging for the Service Backbone, you must set the java.util.logging.config.file JVM system property to point to a properties file that specifies the logging configuration for the SBB library. For example:

```
-Djava.util.logging.config.file=mylogging.properties
```

If the java.util.logging.config.file system property is not defined, the Log Manager reads its initial configuration from a properties file located in the *$<JAVA_DIR>*/lib/logging.properties JRE directory, where *$<JAVA_DIR>* is the JAVA directory used by the Integration Server.

A sample configuration file can be located in your local SOPERA ServiceBackbone installation at:

SOPWare_directory/ServiceBackbone/work/components/org.sopware.SOPEngine/install/conf/logging.properties

For more information about how to configure Java logging, see the SOPERA installation and operations guide and the Oracle documentation.

Adapter Message Logging

Integration Server maintains several types of logs; however, SOPERA Adapter only logs messages to the audit, error, and server logs. Because SOPERA Adapter works in conjunction with the WmART package, the adapter's messages and exceptions typically appear within log messages for the WmART package. The following table describes the type of messages SOPERA Adapter logs into each type of Integration Server log:

Log	Description
Audit	You can monitor individual adapter services using the audit log as you would audit any service in the Integration Server. The audit properties for an adapter service are available in the SOPERA Adapter service template on the **Audit** tab.
Error	SOPERA Adapter automatically posts fatal-level and error-level log messages to the error log. These log messages appear as adapter run-time messages.
Server	SOPERA Adapter posts messages to the server log, depending on how the server log is configured. Fatal-level through debug-level log messages appear as adapter run-time log messages. Trace-level log messages appear as SOPERA Adapter log messages.

The SOPERA Adapter's log messages appear in the following format, ADA.0510.*nnnnc*, where:

- ADA is the facility code that indicates that the message is from an adapter.
- 0510 is the SOPERA Adapter major code, which indicates that the error is generated by SOPERA Adapter.
- *nnnn* represents the error's minor code. For detailed descriptions of the SOPERA Adapter's minor codes, see " SOPERA Adapter Error Codes" on page 86.
- *c* represents the message's severity level (optional).

To monitor SOPERA Adapter's log messages in the Server log, ensure that your server log's logging settings are configured to monitor the following facilities:

- 0113 Adapter Runtime (Managed Object)
- 0114 Adapter Runtime
- 0115 Adapter Runtime (Listener)
- 0116 Adapter Runtime (Notification)
- 0117 Adapter Runtime (Adapter Service)
- 0118 Adapter Runtime (Connection)
- 0121 Adapter Runtime (SCC Transaction Manager)
- 0126 Adapter Runtime (SCC Connection Manager)

SOPERA Adapter Exception Handling

If a SOPERA Adapter object (for example, a connection or service) encounters an error with the SOPERA Infrastructure, it will throw an adapter error coupled with the SOPERA error, exactly as it was thrown by the SOPERA infrastructure. For example, if an adapter service fails on the SOPERA infrastructure at run time because the adapter failed to invoke a SOPERA service, you will receive an adapter error. This error indicates that the adapter service failed, and the adapter error will contain the specific error generated on the SOPERA infrastructure indicating why the service failed.

SOPERA Adapter Error Codes

The following section lists the SOPERA Adapter's error codes and provides information on the error message, reason, and possible action for each error.

Error Code	Description
5502	**Unable to create the node** document_name **under package** package_name**. Ensure that you have sufficient permission.**
	Explanation: Error. Occurs when the adapter fails to create folders for document generation.
	Action: Make sure that you have webMethods administrator privileges and sufficient credentials. For information about setting user privileges, see the *webMethods Integration Server Administrator's Guide*.
5555	**Exception interfacing the Service Backbone.**
	Explanation: Error. Generic error that occurs during the handling of the SOPERA Service Backbone library.

Error Code	Description
	Action: Check the SOPERA infrastructure error in the logs to see the cause of the exception and take appropriate action. If the problem persists, contact Software AG Global Support.
5556	**Failed to release message handler for service** IS_service_name**.**
	Explanation: Error. Occurs when the adapter fails to release the message handler for the specified Integration Server service.
	Action: Check the SOPERA infrastructure error in the logs to see the cause of the exception and take appropriate action. If the problem persists, contact Software AG Global Support.
5557	**Failed to create service skeleton for service** IS_service_name.
	Explanation: Error. Occurs when the adapter fails to create the specified Integration Server service as a SOPERA service.
	Action: Check the SOPERA infrastructure error in the logs to see the cause of the exception and take appropriate action. If the problem persists, contact Software AG Global Support.
5558	**Unable to retrieve XML string from SOPERA incoming message.**
	Explanation: Error. Occurs when receiving an XML message from the SOPERA infrastructure.
	Action: Verify the input message to the adapter service or notification.
5559	**Unable to invoke IS service to satisfy SOPERA request. It might be that the service does not adhere to the predefined signature (one input parameter** incomingMessage **and one output parameter** outgoing message**).**
	Explanation: Error. Occurs when the adapter fails to invoke an Integration Server service in response to a request from the SOPERA infrastructure.
	Action: Check the Integration Server error logs for the Integration Server service invoked and verify the implementation of the Integration Server service.
5561	**Unable to produce outgoing message for SOPERA.**
	Explanation: Error. Occurs when the adapter fails to create a response message from the Integration Server service invoked by the SOPERA infrastructure.
	Action: Check the SOPERA infrastructure error in the logs to see the cause of the exception and take appropriate action. If the problem persists, contact Software AG Global Support.
5562	**Unable to send service response via SOPERA.**
	Explanation:

Error Code	Description
	Explanation: Error. Occurs when the adapter fails to send an Integration Server service response message to SOPERA.
	Action: Check the SOPERA infrastructure error in the logs to see the cause of the exception and take appropriate action. If the problem persists, contact Software AG Global Support.
5563	**SBB error while invoking SOPERA service** SOPERA_service_name **operation** operation_name **with request message** request_message_sent**.**
	Explanation: Error. Generic error that occurs when invoking an adapter service.
	Action: Verify the values of the input parameters for the service. Check the SOPERA infrastructure error in the logs to see the cause of the exception and take appropriate action. If the problem persists, contact Software AG Global Support.
5564	**Error identifying how many attachments the response message contains.**
	Explanation: Error. Occurs when attempting to get the attachment count in the response message in a RequestResponse Call or the pub.wmsopera.service:getNonBlockingServiceResponse service.
	Action: Check the SOPERA infrastructure error in the logs to see the cause of the exception and take appropriate action. If the problem persists, contact Software AG Global Support.
5567	**No message handler delivered.**
	Explanation: Error. No responseMessageHandler passed to the pub.wmsopera.service:getNonBlockingServiceResponse service.
	Action: Ensure that the message handler object is passed as input to the pub.wmsopera.service:getNonBlockingServiceResponse service.
5568	**Error generating IS document** document_name **in package** package_name**. Errors:** error_text
	Explanation: **Explanation: Error. Occurs when the adapter fails to generate a document from a SOPERA service. The value of *error_text* is null when no document is generated.**
	Action: Ensure that the correct folder name is appended to both the Input document type and the Output document type.
5569	**Invalid value of document type. Either folder name or document type name is missing in** document_type_value
	Explanation: Error. Occurs when either the folder name or document type name is missing from the value of the timeout parameter.

Error Code	Description
	Action: Ensure that the document type value has folder name and document name separated by ':'.
5570	**Unable to read and transform SOPERA request message into readable format for the Integration Server service.**
	Explanation: Error. Occurs when the adapter fails to read or transform a SOPERA request message into an Integration Server service.
	Action: Check the SOPERA infrastructure error in the logs to see the cause of the exception and take appropriate action. If the problem persists, contact Software AG Global Support.
5571	**Unable to instantiate Admin Facade at** Admin_Facade_location**.**
	Explanation: Error. Occurs when there is an error during the AdminFacade creation when enabling an adapter connection or starting Integration Server.
	Action: Verify the connection parameters in the SOPERA Adapter sopera.config file, located in the *Integration Server_directory* \packages\ WmSoperaAdapter \config directory. For information about the SOPERA Adapter configuration file, see "Completing the Installation" on page 23.
5572	**Unable to instantiate ServiceBackbone.**
	Explanation: Error. Occurs when there is an error during the ServiceBackbone creation when enabling an adapter connection or starting Integration Server.
	Action: Verify the connection parameters in the SOPERA Adapter sopera.config file, located in the *Integration Server_directory* \packages\ WmSoperaAdapter \config directory. For information about the SOPERA Adapter configuration file, see "Completing the Installation" on page 23.
5601	**Unable to create/register the one-way service** IS_service_name **in the SOPERA registry.**
	Explanation: Error. Occurs when there is an issue during the creation or registration of an Integration Server service as a one-way service in the SOPERA service registry.
	Action: Check the AdminFacade error in the logs to see the cause of the exception and take appropriate action. If the problem persists, contact Software AG Global Support.
5602	**Unable to create/register the request-response service** IS_service_name **in the SOPERA registry.**
	Explanation: Error. Occurs when there is an issue during the creation or registration of an Integration Server service as a request-response service in the SOPERA service registry.

Error Code	Description
	Action: Check the AdminFacade error in the logs to see the cause of the exception and take appropriate action. If the problem persists, contact Software AG Global Support.
5603	**Transport not enabled or port not configured.**
	Explanation: Error. Occurs when the transport port property is not configured properly in the SOPERA Adapter configuration file.
	Action: Configure the transport port property as required. For information about the SOPERA Adapter configuration file, see "Completing the Installation" on page 23.
5604	**Unable to create/register provider for service:** IS_service_name
	Explanation: Error. Occurs for any error occur during the registration of a new SOPERA service provider for the specified Integration Server service.
	Action: Check the AdminFacade error in the logs to see the cause of the exception and take appropriate action. If the problem persists, contact Software AG Global Support.
5605	**Required parameters not provided for the notification.**
	Explanation: Error. Validation error for a notification in which required message parameters are missing.
	Action: Check all required parameters in the notification and set them as required. For information about configuring adapter notifications, see "Configuring Listener Notifications" on page 69.
5606	**Invalid SOPERA service name** service_name **found. The service name should be of the format {<namespace_uri>}<service_name>.**
	Explanation: Error. Occurs when an incorrect SOPERA service name is specified while configuring the notification to invoke an Integration Server service via SOPERA.
	Action: Check if the service name is specified in the following format and correct it if it is not: `{<namespace_uri}<service_name>` Example: {http://myservice.com/example} TestService
5607	**Error while generating IS service/document** service/document_name**. Error** error_text
	Explanation: Error. Occurs while generating an empty Integration Server service or document.

Error Code	Description
	Action: Ensure that the IS service name or IS document name is a valid Integration Server name and no other Integration Server service or document of the same name exists.
5608	**IS document does not exist** document_name**. Enter an existing one.**
	Explanation: Error. Occurs when you attempt to create a SOPERA notification provider with a non-existing IS document.
	Action: Ensure that the IS document exists onIntegration Server.
5573	**Error while retrieving the SOPERA service** service_name **from the registry.**
	Explanation: Error. Occurs when attempting to retrieve the specified adapter service name instance from the registry name.
	Action: Ensure that the service name is correct and the service is registered. See the AdminFacade logs for the exact reason for the error.
5574	**Error** while processing input parameters while invoking SOPERA service *service_name*
	Explanation: Error. Occurs when the adapter is unable to convert the input message into XML or the request message is missing.
	Action: Recreate the input document type for the specified service.
5575	**Fault message received - error encountered in service provider while invoking SOPERA service**.
	Explanation: Error. The exception occurs during the invocation of the SOPERA provider service.
	Action: Check the error logs at the SOPERA provider end.
5576	**SBB Error encountered when receiving non-blocking response from provider.**
	Explanation: Error. Occurs when the adapter is unable to get non-blocking response from the pub.wmsopera.service:getNonBlockingServiceResponse service.
	Action: Check the SOPERA infrastructure error in the logs to see the cause of the exception and take appropriate action. If the problem persists, contact Software AG Global Support.
5577	**Invalid value of timeout parameter.**
	Explanation: Error. Occurs when the value of the timeout parameter is not numeric.
	Action: Verify the value of the timeout parameter passed in the pub.wmsopera.service:getNonBlockingServiceResponse service.
5578	**Callback timeout. Increase timeout value.**

Error Code	Description
	Explanation: Error. Occurs when attempting to get a response for a Nonblocking request.
	Action: Increase the timeout value in the input parameter and retry. Also, ensure that the provider has not thrown any error.
5579	**Configure local endpoint for service** service_name **and operation** operation_name **and then restart all adapter connections.**
	Explanation: Error. Local endpoint not found for the specified SOPERA callback operation in the SOPERA infrastructure.
	Action: Configure the local endpoint for the SOPERA callback service and the service operation. Restart all adapter connections.
5580	**Authentication error invoking operation** operation_name **on service** service_name **with subject** user_credentials
	Explanation: Error. A SOPERA outgoing call failed because of invalid user credentials.
	Action: Ensure that the user who invoked the adapter service on Integration Server has access to the specified SOPERA service on the SOPERA Infrastructure.

9 Adapter Built-in Services

Overview

This appendix describes the built-in services provided with the WmSoperaAdapter package of webMethods SOPERA Adapter.

WmSoperaAdapter Package

The WmSoperaAdapter package contains the public services used to manage the SOPERA services.

pub.wmsopera.admin.adminService

Contains the package startup and shutdown services.

pub.wmsopera.service

Contains the utility services to manage the SOPERA services.

pub:wmsopera.service:getNonBlockingServiceResponse

Processes the returned request message handler, when the SOPERA service consumer adapter service template is configured as a non-blocking request-response service.

Input Parameters

timeout	**String** Optional. The timeout, in milliseconds, specifies how long to wait for a response message. Valid values are: ■ A negative value means wait forever. ■ `0` means return immediately regardless of whether the message has arrived. This is the default value.
messageAsObject	**String** Optional. Valid values are: ■ `true` The response is returned as an Object. ■ `false` The response is returned as a String. This is the default value.
responseMessageHandler	**Object** The message handler returned from the SOPERA service consumer adapter service template. Typically, you should use a flow service to map the *responseMessageHandler* output value returned from the adapter service to this parameter.

Output Parameters

responseMessageObject	**Object** The response message as an Object, when the *messageAsObject* input parameter is set to `true`.
responseMessageString	**String** The response message as a String, when the *messsageAsObject* input parameter is set to `false`.