Additional Connectivity Options

from the book

SAP® Cloud Platform Integration: The Comprehensive Guide

by Bilay, Gutsche, Krimmel, and Stiehl

"Your map for any SAP Cloud Platform Integration project."

SAP PRESS

Additional Connectivity Options

While studying this book, you've made yourself familiar with various connectivity options. However, if you back to check out Chapter 1, Section 1.3.3 (Table 1.1), you'll notice that we have, by far, not covered all the adapters provided by SAP Cloud Platform Integration. Covering each and every adapter, each in the detail it deserved, would have filled another book! In addition to that, not every adapter would have fit into the flow of our chapters, and some of them would have required complex system setups to create a demo scenario.

That is why we've provided this additional chapter where we briefly touch on some key adapters that you should know as an integration expert. In the Additional Adapters section, for each adapter, we provide a brief introduction and refer to additional information material publicly available on the web.

Note

In many integration flows described within this book, we used a SOAP sender channel (see, for example, Chapter 4, Section 4.1.3). However, we haven't yet touched on the SOAP receiver adapter. In the Invoking a SOAP-Based Web Service section, finally, we'll round up this chapter with a simple demo scenario that involves an outbound SOAP connection. As SOAP is one of the key protocols used in the context of SAP Cloud Platform Integration, we didn't want to leave you without covering this connectivity option (on the outbound side).

Additional Adapters

This section provides a brief summary of several adapters that have not been covered in the book. For each adapter, we refer to additional information sources that you can check out in order to get more details.

FTP Adapter

The file transfer protocol (FTP) adapter allows you to send messages to an FTP server or to receive messages from an FTP server. The communication channel uses transport protocol FTPS which is an extension to the well-known file transfer protocol.

For more information, check out the documentation of SAP Cloud Platform at http://s-prs.co/5077200.

LDAP Receiver Adapter

The lightweight directory access protocol (LDAP) receiver adapter allows you to connect SAP Cloud Integration to a service that supports. LDAP allows you to access and maintain distributed directory information services. Such services map the names of network resources (for example, printers, users, and devices) to their respective network addresses. One example of such a service is the MS Active Directory.

For more information, read the following blog available in SAP Community (http://s-prs.co/5077201).

RFC Receiver Adapter

You can use the remote function call (RFC) receiver adapter to connect SAP Cloud Integration to an SAP system based on the ABAP application server through the RFC. As prerequisite to use this connectivity option, you need to connect the SAP system with SAP Cloud Integration through the cloud connector. At the time when writing this book, the RFC adapter supports synchronous communication.

For more information, read the following blog available in SAP Community (http://s-prs.co/5077202).

SFTP Sender and Receiver Adapter

This adapter allows you to access files on a remote system through the secure shell file transfer (SFTP) protocol. Like the mail sender adapter, the SFTP sender adapter is a polling adapter that sends a request to the sender system (in this case, an SFTP server) to read data from it. The SFTP receiver adapter allows you to write files to the SFTP server.

For more information, read the following blog available in SAP Community (http://s-prs.co/5077203). The blog at http://s-prs.co/5077204 explains how to use SAP Cloud Integration together with the cloud connector to connect to an on-premise SFTP server.

SOAP Receiver Adapter

The SOAP sender adapter has been used in many scenarios throughout this book. Using a SOAP client like Postman, you can easily set up a SOAP client to simulate a SOAP sender. Simulating a SOAP request through Postman is also

a simple way to start message processing. However, we haven't yet covered the SOAP receiver adapter (which is also one of the key adapters of SAP Cloud Integration). For an easy-to-set-up integration flow using this connectivity option, see Invoking a SOAP-Based Web Service.

XI Sender and Receiver Adapter

The exchange infrastructure (XI) adapter allows you to connect SAP Cloud Integration to an on-premise backend system over the XI 3.0 protocol. XI is the name of the first version of SAP Process Integration. Technically speaking, the XI adapter supports the connectivity of SAP Cloud Integration with a local integration engine configured in the on-premise system.

For more information, read the following blogs available in SAP Community: *Cloud Integration – Configuring Scenario Using the XI Sender Adapter* (http://s-prs.co/5077205) and *Cloud Integration – Configuring Scenario Using the XI Receiver Adapter* (http://s-prs.co/5077206).

Invoking a SOAP-Based Web Service

Throughout this book, you've worked extensively with the SOAP (SOAP 1.x) sender adapter. SOAP is an important protocol often used in integration projects. Therefore, we would like to show you in this appendix how you can easily configure an integration flow with a SOAP receiver adapter (which we have not covered in the book).

What we will do is to invoke a public web service (at http://www.dneonline.com/calculator.asmx) though the SOAP protocol. This web service provides functions to calculate integers. We have tested it at the time when writing this book. Note that, however, public web services can change. Therefore, we cannot guarantee that the integration flow works always as expected.

For your convenience, we provide the WSDL file for the web service on the book's website (to be updated for the second edition). The file name is: *calculator.wsdl*.

The following simple integration flow (Figure 1) allows you to add two integers (provided with an inbound SOAP request). Optionally, you can send the result to your email inbox. Here, you can use the same mail receiver adapter configuration like, in previous scenarios in this book.

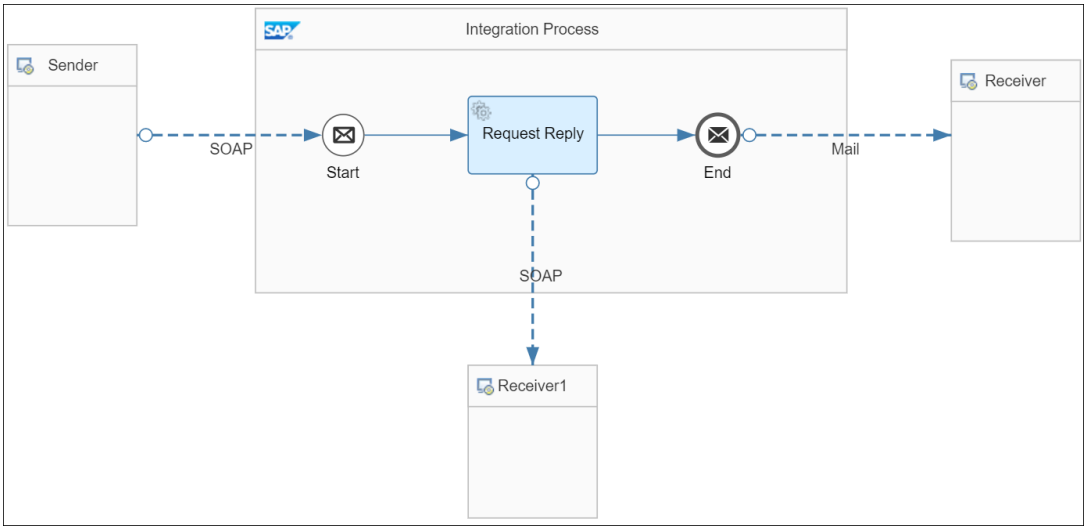

Figure 1 Target Integration Flow

For the configuration of the SOAP sender adapter, we propose to proceed as described in Chapter 4, Section 4.1.3. When creating the channel between the **Sender** pool and the **Start Message** event, choose **SOAP • SOAP 1.x**. We propose to configure the adapter settings (**Connection** tab) like shown in Figure 2.

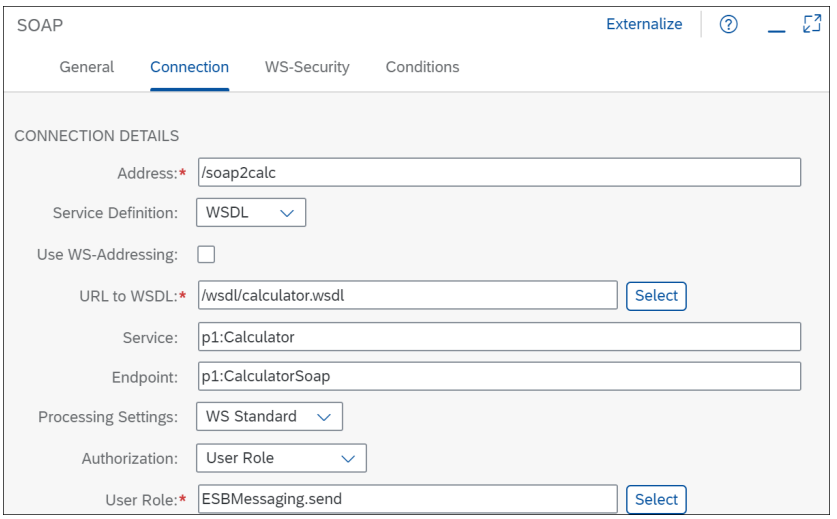

Figure 2 SOAP 1.x Sender Adapter Connection Settings

Here, take the following steps:

1. As **Address**, enter `/soap2calc`.
2. As **Service Definition** select **WSDL**.

3. Keep the option **Use WS-Addressing** disabled.
4. Next to **URL to WSDL** choose **Select** and browse for the WSDL file *calculator.wsdl* (downloaded before from the book website and uploaded as integration flow resource).
5. When selecting the WSDL file, you are asked to select a service operation. Select the Operation **Add**. Based on your choice, the entries for **Service** and **Endpoint** are automatically filled. For further information on the namespace prefix (p1), refer to Chapter 5, Section 5.2.3.
6. As **Processing Settings**, select **WS Standard**.
7. Keep the default settings for **Authorization** and **User Role**.

For more information on the SOAP 1.x sender adapter, see Chapter 4, Section 4.1.3.

For the channel between the **Request Reply** step and the **Receiver1** pool, choose **SOAP • SOAP 1.x**. Specify the following settings (in the **Connection** tab) as shown in Figure 3.

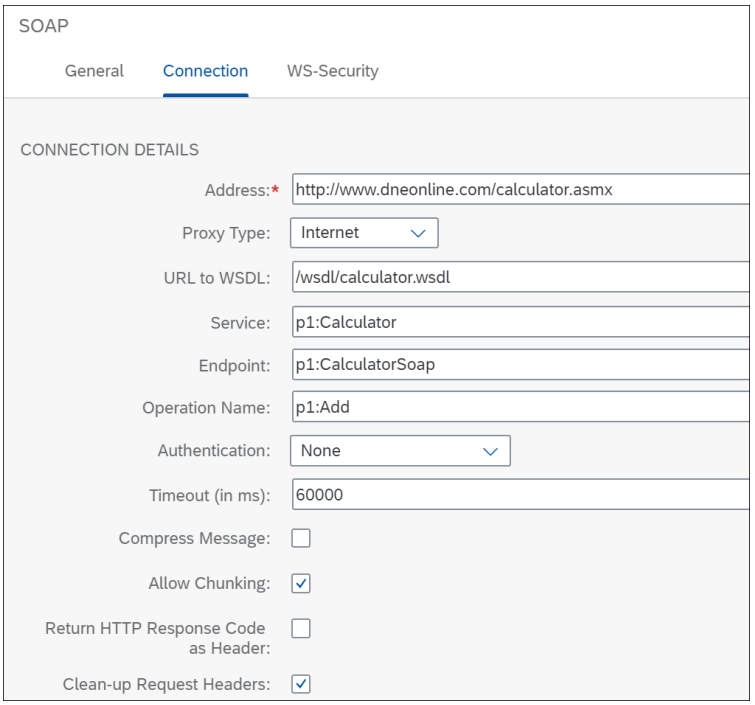

Figure 3 SOAP 1.x Receiver Adapter Settings

Take the following steps:

1. In the **Address** field, enter `http://www.dneonline.com/calculator.asmx`.
2. This is the same URL which you can find when opening the WSDL file *calculator.wsdl* and searching for the tag `<soap:address>`. The attribute named `location` points to the address we are interested in.
3. As **Proxy Type**, keep the setting **Internet** (as we are invoking a service over the Internet).
4. Click **Select** next to **URL to WSDL**. Select the operation (**Add** in our case).
5. The entries for **Service** and **Endpoint** are automatically filled like shown in Figure 3.
6. For **Authentication** choose **None**.

There are additional SOAP receiver adapter settings that you can keep (default values). We briefly explain them:

- **Timeout**: Specify the time the client should wait for a response before the connection is being interrupted.
- You can keep the **Compress Message** setting as it is (disabled) and the **Allow Chunking** enabled.
- You can keep the option **Return HTTP Response Code as Header** disabled.
- **Clean-up Request Headers**: Certain adapter might add headers to the message (depending on the adapter type). With this setting, you have the option to remove those headers before passing over the message to the next processing step (after the request reply step, see Figure 1). With this setting, you can make sure that another SOAP request after the request reply step is not executed with the same headers.

Finally, add a mail receiver channel and then deploy the integration flow.

In the SOAP client (e.g. Postman), enter the endpoint address for your integration flow (which you find in the **Monitor** application under **Manage Integration Content** when selecting the deployed integration flow [in the **Endpoints** tab]).

In the SOAP client, click the request message and enter two integers you like to add, as seen in Listing 1.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/
envelope/" xmlns:tem="http://tempuri.org/">
   <soapenv:Header/>
   <soapenv:Body>
      <tem:Add>
         <tem:intA>1</tem:intA>
         <tem:intB>2</tem:intB>
      </tem:Add>
   </soapenv:Body>
</soapenv:Envelope>
```

Listing 1 Request Message to Add Integers 1 and 2

When invoking the integration flow, as a result you should receive an email with the result of the calculation.