Hands-on Lab:
SOAP to REST Transformation

Objectives

In this hands-on lab, you will import a SOAP API then use the REST transformation to enhance the API by adding a REST resource.

Steps

1. Open Windows Services UI and check that the following services are started. If they are not running, start the service.

   a) **Software AG Integration Server 10.5 (default)**

   b) **Software AG Internal Integration Server 10.5 (default)**

   c) **Software AG API Gateway Data Store 10.5**

> *Note*: Since we are not going to interface with any other Software AG components, only these are needed to be started.

2. You can monitor the progress of the Integration Server startup sequence in **Baretail** by opening the following logfiles in **Baretail:**

   a) **C:\SoftwareAG\IntegrationServer\instances\default\logs\server.log**

   b) **C:\SoftwareAGInternal\IntegrationServer\instances\default\logs\server.log**

3. Open File Explorer and copy **Numbers.zip** from:

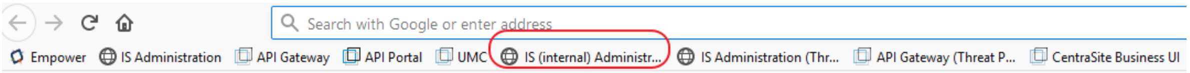
   C:\Training\E456C03-75E\Resources\Package

   to folder:

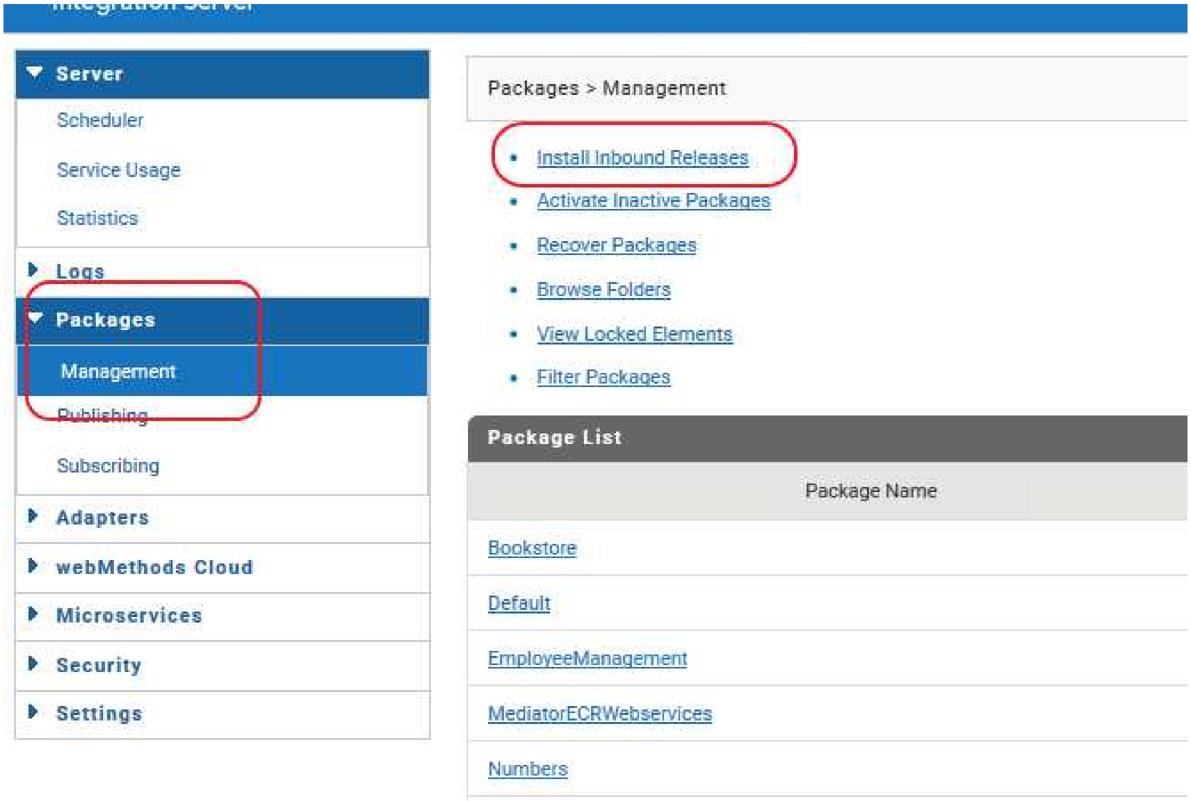
   C:\SoftwareAGInternal\IntegrationServer\instances\default\replicate\inbound

4. Login to **Internal Integration Server Administration** by opening **Firefox** and clicking the **IS (internal) Administration** link (just below the URL field). Login as Administrator (**Administrator** / **manage**).

> *Note*: The Integration Server takes a few minutes to start, therefore you may have to refresh the Administration page until you are asked to login:



5. Select **Packages -> Management -> Install Inbound Releases**



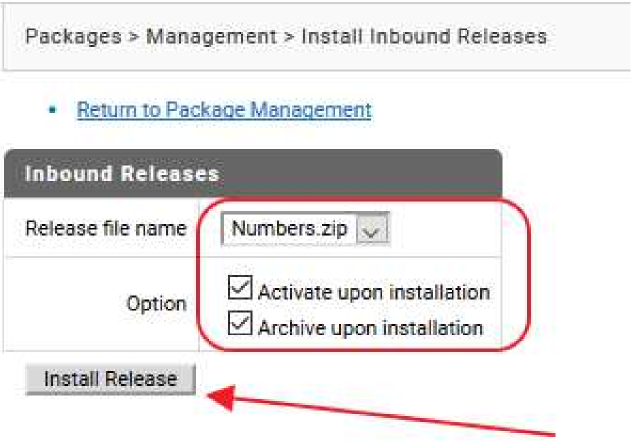
Packages > Management

- Install Inbound Releases
- Activate Inactive Packages
- Recover Packages
- Browse Folders
- View Locked Elements
- Filter Packages

Package List

Package Name
Bookstore
Default
EmployeeManagement
MediatorECRWebservices
Numbers

6. Verify the following:



Packages > Management > Install Inbound Releases

- Return to Package Management

Inbound Releases

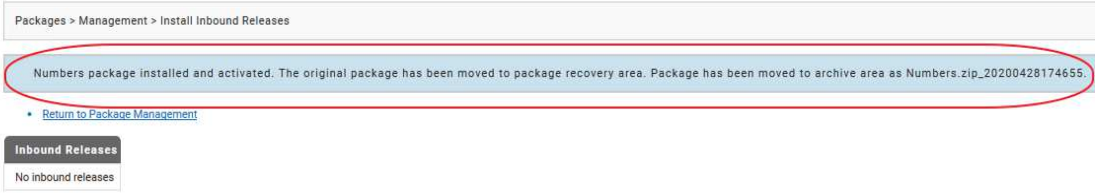
Release file name	Numbers.zip
Option	☑ Activate upon installation ☑ Archive upon installation

Release File name: **Numbers.zip**
Options:

- **Activate upon installation** - <checked>
- **Archive upon installation** - <checked>

Click **Install Release**.

7. You will receive a message:



Packages > Management > Install Inbound Releases

Numbers package installed and activated. The original package has been moved to package recovery area. Package has been moved to archive area as Numbers.zip_20200428174655.

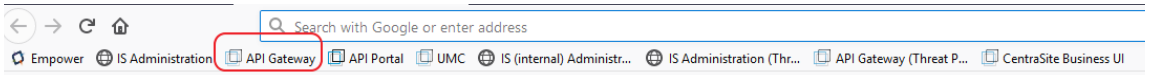
- Return to Package Management

Inbound Releases
No inbound releases

> *Note:* The Numbers.zip_ file name will be different.

You may now close the Internal Integration Server tab in Firefox.

8. Login to API Gateway by clicking the API Gateway link (just below the URL field). Login as a API Gateway Administrator **Administrator** | **manage**.



   a) Select **User Management** from the user drop-down.

   b) If user Sumala already exists, continue with step 9.
   Only in case user Sumala does not exist yet, click **+ Add user** and provide the following properties:

      i. Login ID: **Sumala**

      ii. First name: **Sumala**

      iii. Last name: **Sumus**

      iv. Password: **manage**

      v. Email addresses: **Sumala@company.com** Click **+ Add**.

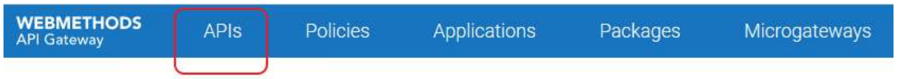
   c) Click **Continue to associate groups**. Add Sumala to group **API-Gateway-Administrators**. Click **Save.**

   d) Logout from API Gateway.

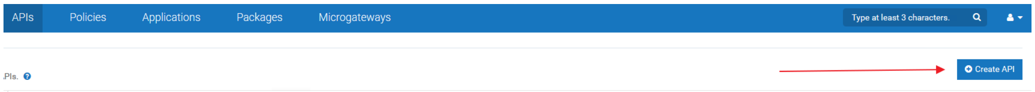
9. Login to API Gateway as user **Sumala** |**manage.**

10. Create the **Divide** SOAP API:

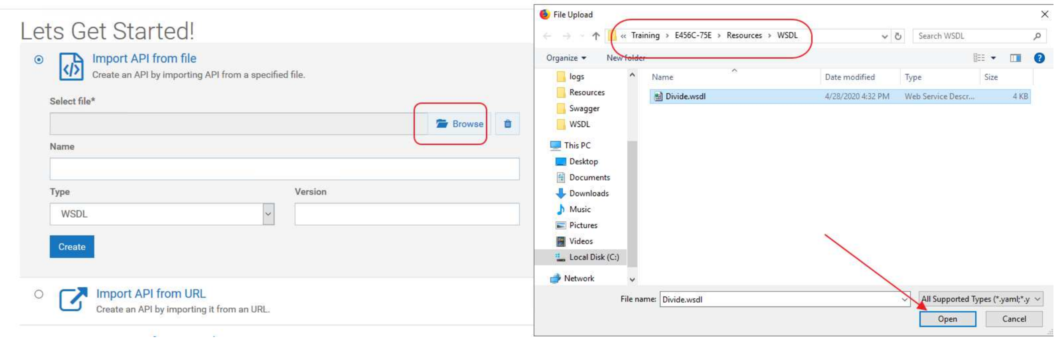
   a) Select APIs



   b) Then select the **+ Create AP**" button located in the upper right.



c) Select the **Browse** button and navigate to **Divide.wsdl** located in: **C:\Training\E456C03-75E\Resources\WSDL**



Name	Date modified	Type	Size
Divide.wsdl	4/28/2020 4:32 PM	Web Service Descr...	4 KB

File name: Divide.wsdl

All Supported Types (*.yaml;*.y

d) Enter the following information:

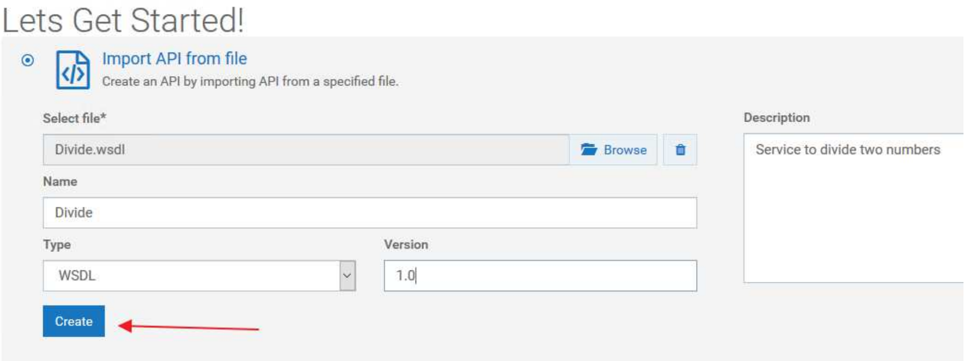
i. Name: **Divide**

ii. Type: **WSDL**

iii. Version: **1.0**

iv. Description: **Service to divide two numbers.**

e) Select the "Create" button



Select file*: Divide.wsdl

Name: Divide

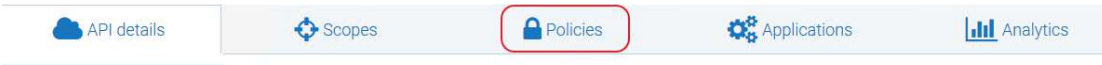
Type: WSDL

Version: 1.0

Description: Service to divide two numbers

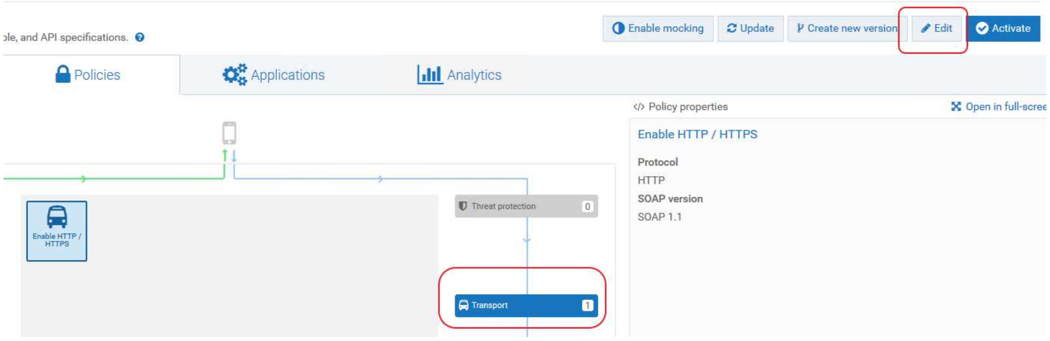
The Divide API is now created.

11. Select the **Policies** tab (located to the right of Scopes).



12. Adjust the SOAP version from SOAP 1.1 to **SOAP 1.2**:

a) Select the **Transport** policy group, then select the **Edit** button.



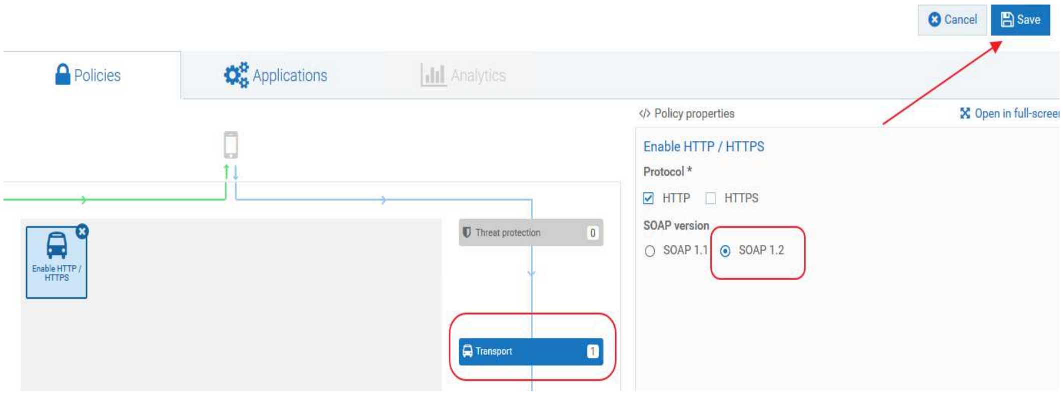
Policy properties

Enable HTTP / HTTPS

Protocol
HTTP

SOAP version
SOAP 1.1

b) Select **SOAP 1.2**, then select **Save.**



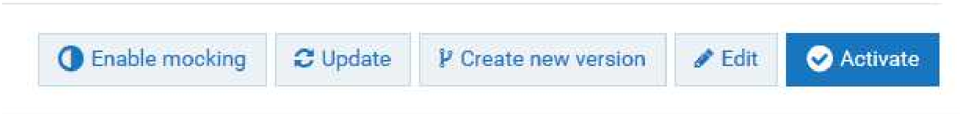
Policy properties

Enable HTTP / HTTPS

Protocol *
☑ HTTP ☐ HTTPS

SOAP version
○ SOAP 1.1 ● SOAP 1.2

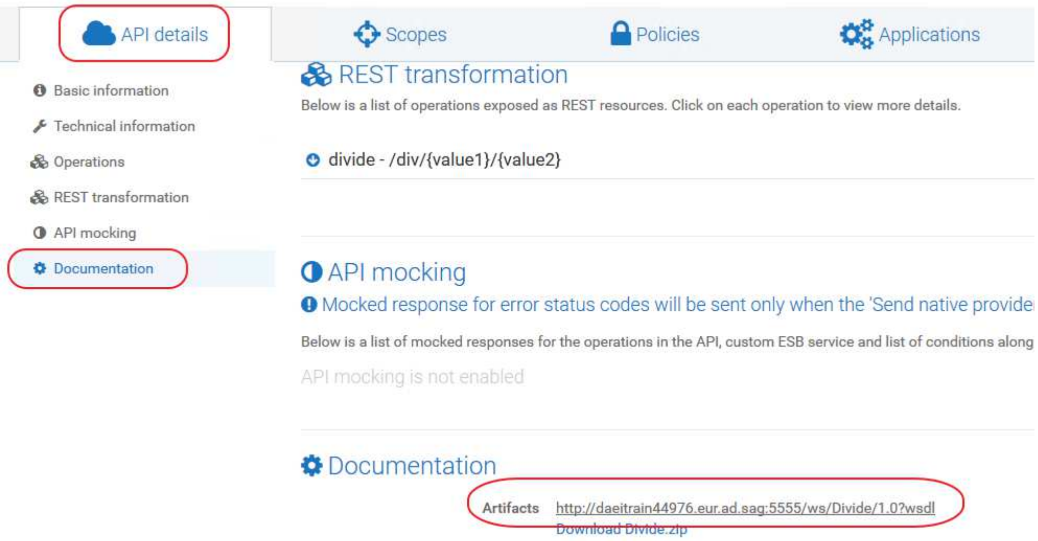
13. Activate the API be selecting the **Activate** button. Confirm with **Yes**.



The API is now successfully activated.

14. Import the API into SOAP UI and test to ensure the Divide API is working:

a) Select the **API details** tab, then **Documentation**. The WSDL URL is displayed next to Artifacts. Copy the URL into the clipboard which we'll use to import into SOAP UI.



REST transformation

Below is a list of operations exposed as REST resources. Click on each operation to view more details.

divide - /div/{value1}/{value2}

API mocking

Mocked response for error status codes will be sent only when the 'Send native provide

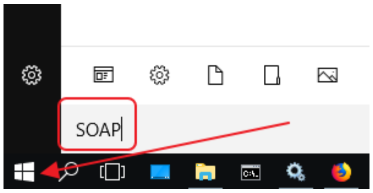
Below is a list of mocked responses for the operations in the API, custom ESB service and list of conditions along

API mocking is not enabled

Documentation

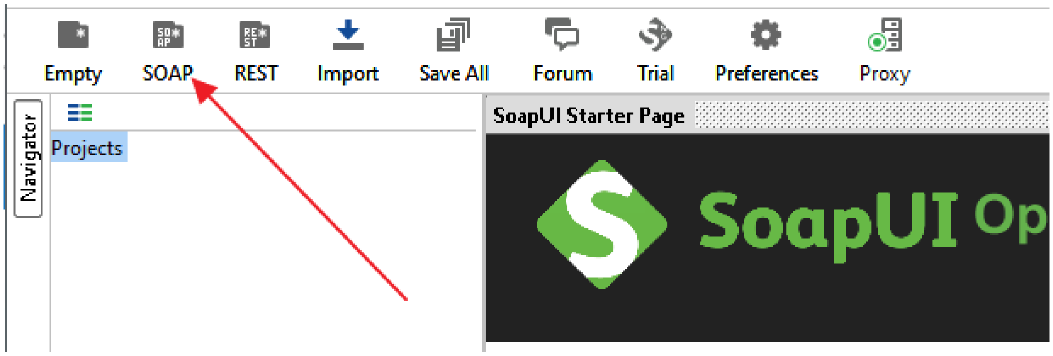
Artifacts http://daeitrain44976.eur.ad.sag:5555/ws/Divide/1.0?wsdl
Download Divide.zip

b) Click on the Windows Start icon and type **SOAP.**



The SoapUI-5.3.0 program will display. Click on **SoapUI-5.3.0.**

c) In **SOAP UI**, click on the **SOAP** icon located at the top



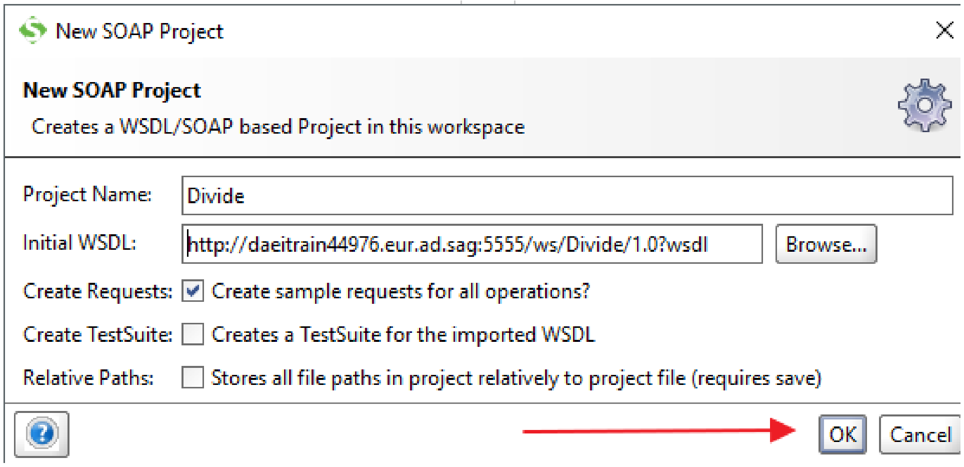
The SOAP Wizard will display.

d) In the wizard provide:

   i. Project Name: **Divide**

   ii. Initial WSDL: <Paste in the URL copied from Divide API in API Gateway>

   Select **OK**.



New SOAP Project

New SOAP Project

Creates a WSDL/SOAP based Project in this workspace

Project Name: Divide

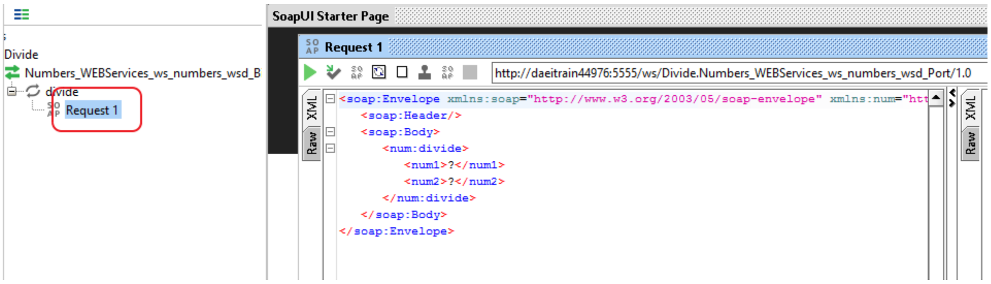
Initial WSDL: http://daeitrain44976.eur.ad.sag:5555/ws/Divide/1.0?wsdl

Create Requests: ☑ Create sample requests for all operations?

Create TestSuite: □ Creates a TestSuite for the imported WSDL

Relative Paths: □ Stores all file paths in project relatively to project file (requires save)

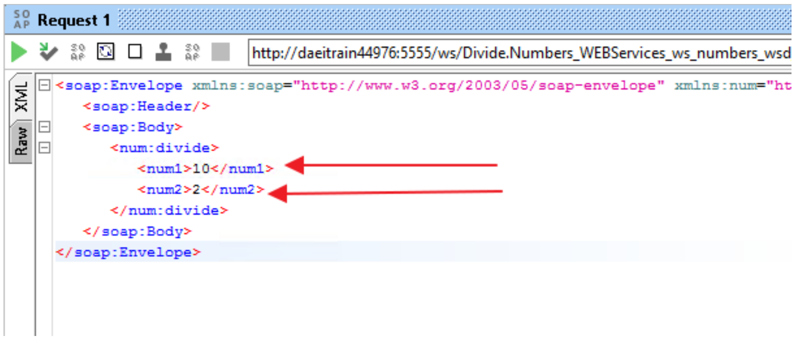
e) Navigate to **Request 1** and double-click. The Request 1 body will display.



http://daeitrain44976:5555/ws/Divide.Numbers_WEBServices_ws_numbers_wsd_Port/1.0

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope" xmlns:num="htt
   <soap:Header/>
   <soap:Body>
      <num:divide>
         <num1>?</num1>
         <num2>?</num2>
      </num:divide>
   </soap:Body>
</soap:Envelope>
```

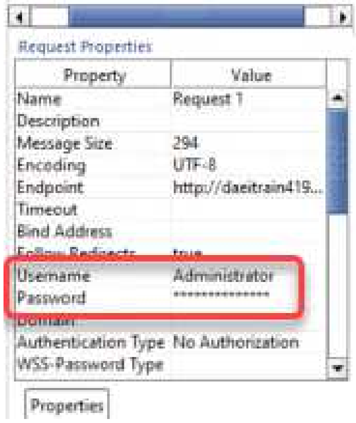
f) We will now test the API. Replace the two question marks (?) between the num1 and num2 tags with **10** and **2** respectively. Will now tell the API to Divide 10 by 2 (10/2).



http://daeitrain44976:5555/ws/Divide.Numbers_WEBServices_ws_numbers_wsd

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope" xmlns:num="ht
   <soap:Header/>
   <soap:Body>
      <num:divide>
         <num1>10</num1>
         <num2>2</num2>
      </num:divide>
   </soap:Body>
</soap:Envelope>
```

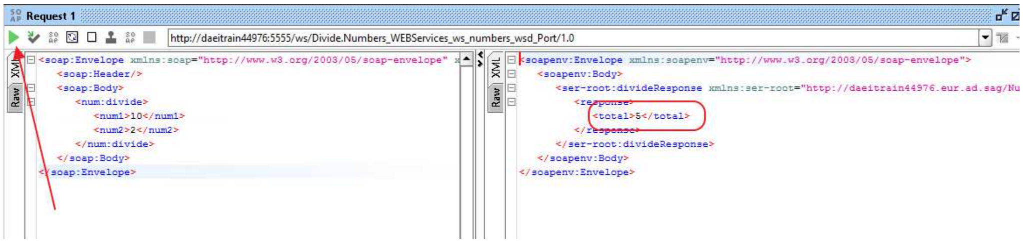
g) Because the native Divide SOAP service requires for basic authentication at the internal is, provide **Administrator** | **manage** as Username | Password in the Request Properties.



Request Properties

Property	Value
Name	Request 1
Description	
Message Size	294
Encoding	UTF-8
Endpoint	http://daeitrain419...
Timeout	
Bind Address	
Follow Redirects	true
Username	Administrator
Password	***************
Domain	
Authentication Type	No Authorization
WSS-Password Type	

h) Click on the Send button (green triangle) – results are displayed in the right pane between the <total> tags.



http://daeitrain44976:5555/ws/Divide.Numbers_WEBServices_ws_numbers_wsd_Port/1.0

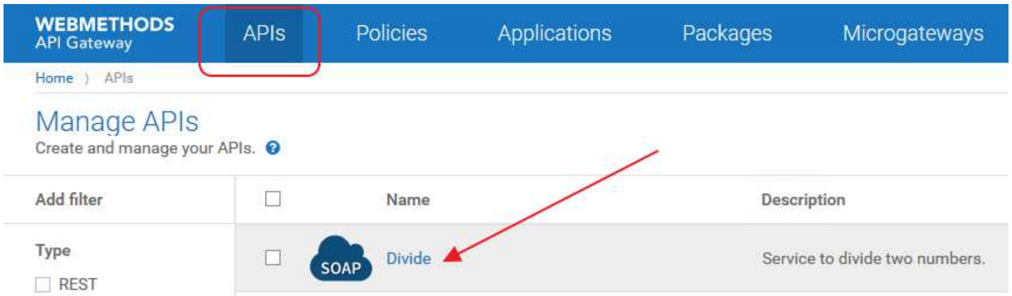
```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope" x
   <soap:Header/>
   <soap:Body>
      <num:divide>
         <num1>10</num1>
         <num2>2</num2>
      </num:divide>
   </soap:Body>
</soap:Envelope>
```

```
<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
   <soapenv:Body>
      <ser-root:divideResponse xmlns:ser-root="http://daeitrain44976.eur.ad.sag/Nu
         <response>
            <total>5</total>
         </response>
      </ser-root:divideResponse>
   </soapenv:Body>
</soapenv:Envelope>
```

i) Verify the answer is **<total>5</total>**. The SOAP service is working as designed.

15. Enhance the Divide SOAP service by transforming the SOAP to REST:

a) Login to API Gateway as Sumala (**Sumala** | **manage**), then click on the **APIs** tab. Select the Divide API.

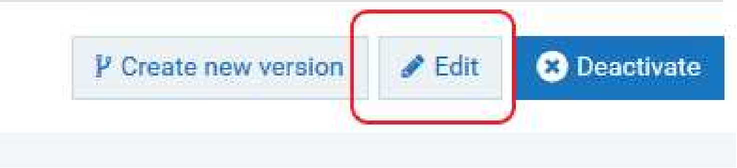


Manage APIs

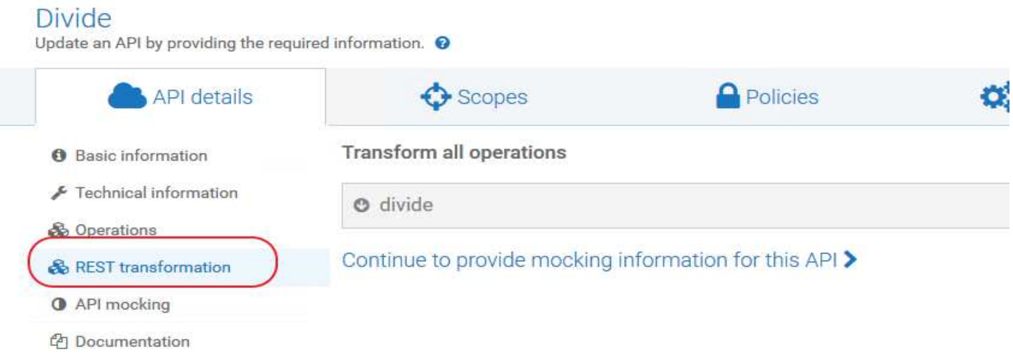
Create and manage your APIs.

□	Name	Description
□	SOAP Divide	Service to divide two numbers.

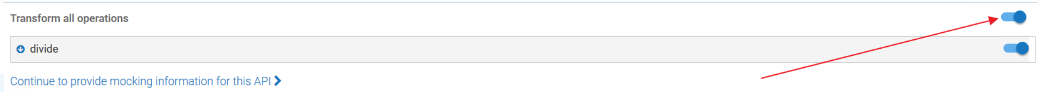
b) Select **Edit**. Select **Yes** to "Are you sure…"



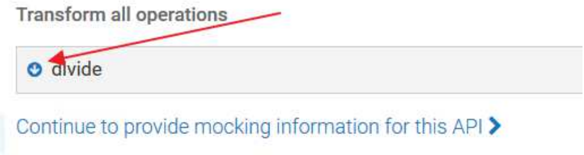
c) Click on the **REST transformation** link.



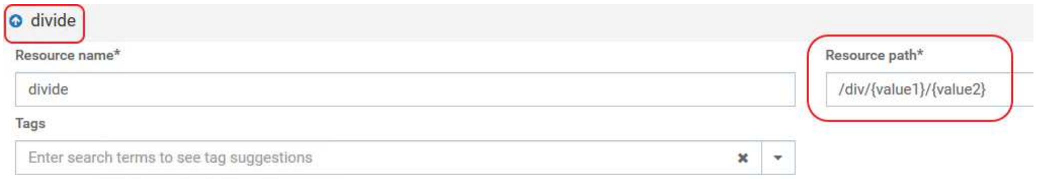
d) Enable the **Transform all operations** by clicking on the slider. This will enable the divide resource.



e) Select the **divide** operation.



f) Change the Resource path to: **/div/{value1}/{value2}**



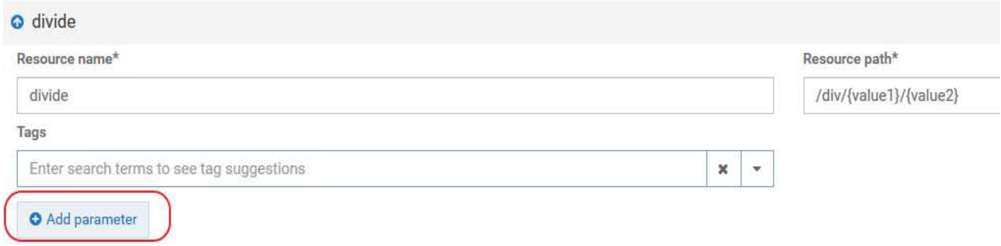
Resource name*: divide

Resource path*: /div/{value1}/{value2}

Tags: Enter search terms to see tag suggestions

Note: value1 and value2 are now variables when consuming the REST call. Will now need to provide mappings for value1 and value2 to num1 and num2 respectively.

g) Click on **Add parameter** to provide the maps of value1 to num1:



divide

Resource name*: divide

Resource path*: /div/{value1}/{value2}

Tags: Enter search terms to see tag suggestions

Add the parameters:

i. Name*: **value1**

ii. Description: **The first value**

iii. Type: **Path**

iv. Data type: **String**

v. XPath: //**num1**

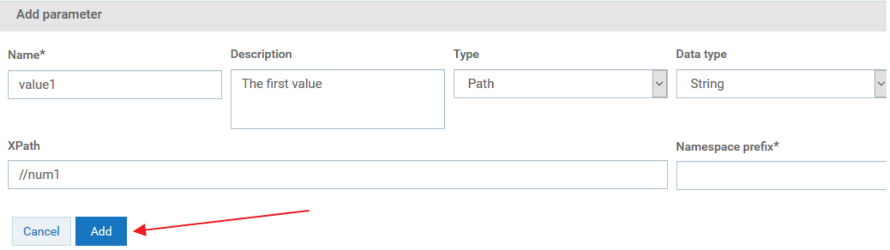
vi. Required: <*leave unselected*>

vii. Repeat: <*leave unselected*>

viii. Namespace prefix*: <*leave blank*>

ix. Namespace URI*: <*leave blank*>

Select the **Add** button.



Add parameter

Name*: value1

Description: The first value

Type: Path

Data type: String

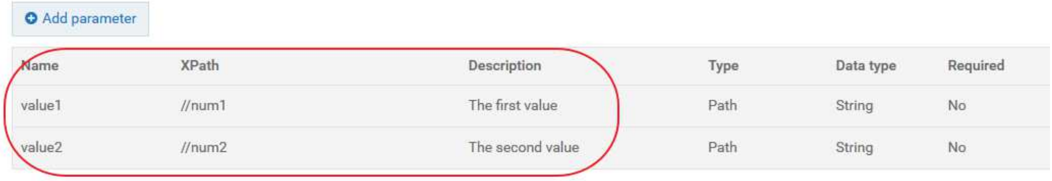
XPath: //num1

Namespace prefix*:

h) Add another parameter and provide the mapping for value2 to num2:

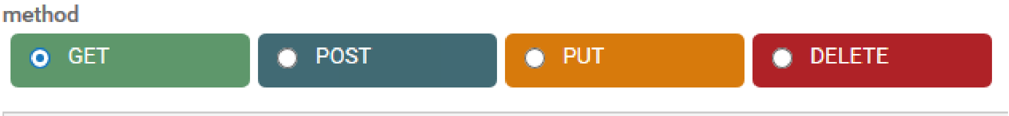
   i. Name*: **value2**
   ii. Description: **The second value**
   iii. Type: **Path**
   iv. Data type: **String**
   v. XPath: //**num2**
   vi. Required: <*leave unselected*>
   vii. Repeat: <*leave unselected*>
   viii. Namespace prefix*: <*leave blank*>
   ix. Namespace URI*: <*leave bank*>

i) Select **Add parameter** – the parameters will look like:



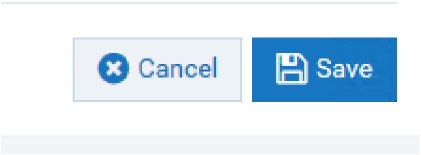
Name	XPath	Description	Type	Data type	Required
value1	//num1	The first value	Path	String	No
value2	//num2	The second value	Path	String	No

j) Select **GET** for the method.

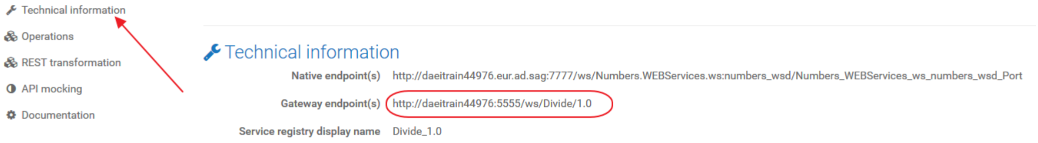

method

◉ GET ○ POST ○ PUT ○ DELETE

k) Save the changes by selecting the **Save** button.



l) Navigate to **Technical information** and copy the Gateway endpoint to the clipboard.



Technical information

Native endpoint(s) http://daeitrain44976.eur.ad.sag:7777/ws/Numbers.WEBServices.ws:numbers_wsd/Numbers_WEBServices_ws_numbers_wsd_Port

Gateway endpoint(s) http://daeitrain44976:5555/ws/Divide/1.0

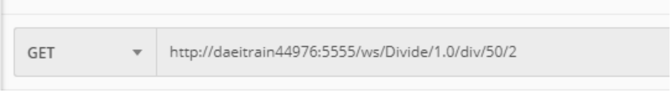
Service registry display name Divide_1.0

16. Test the REST call by using Postman:

a) Open Postman by selecting the shortcut at the bottom of the screen.

b) Paste the Gateway endpoint from the clipboard into the URL, then append:

/div/50/2

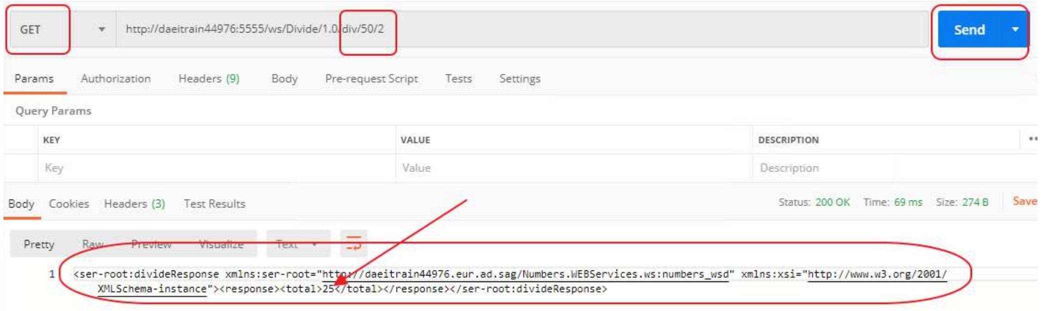


http://daeitrain44976:5555/ws/Divide/1.0/div/50/2

Note: This will send 50 and value1 and 2 as value2 to the service. (50 divided by 2).

c) In the **Request Properties** pane set Username to **Administrator** and Password to **manage**.

d) Ensure the **GET** method is selected. Click on **Send**.



http://daeitrain44976:5555/ws/Divide/1.0/div/50/2

Status: 200 OK Time: 69 ms Size: 274 B

```
<ser-root:divideResponse xmlns:ser-root="http://daeitrain44976.eur.ad.sag/Numbers.WEBServices.ws:numbers_wsd" xmlns:xsi="http://www.w3.org/2001/
  XMLSchema-instance"><response><total>25</total></response></ser-root:divideResponse>
```

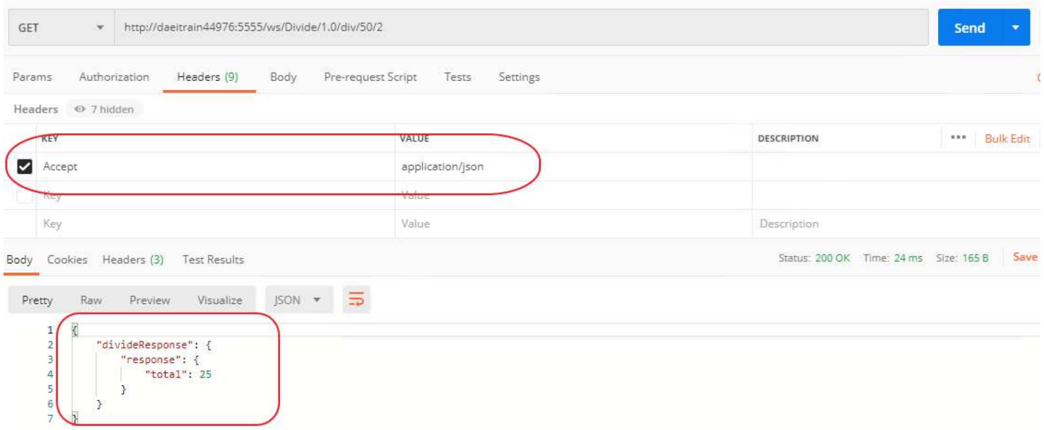
Note how the results are in XML which was the same as when SOAP-UI was used.

e) To have a different result type (such as json) add an **Accept** key/value pair to the header. In Postman, click on the **Headers** tab to add the KEY/VALUE:

i. KEY: **Accept**

ii. VALUE: **application/json**

Click on **Send.**



http://daeitrain44976:5555/ws/Divide/1.0/div/50/2

	KEY	VALUE	DESCRIPTION
☑	Accept	application/json	

Status: 200 OK Time: 24 ms Size: 165 B

```
{
    "divideResponse": {
        "response": {
            "total": 25
        }
    }
}
```

The results are now in json format.