API SOAP Request or Response Activity

Introduction

An API SOAP request activity interacts with an API connection to receive data from a Jitterbit Custom API that has been set up through the Harmony API Manager, and is to be used as a source in an operation.

An API SOAP response activity interacts with an API connection to return data to a Jitterbit Custom API, and is to be used as a target in an operation.

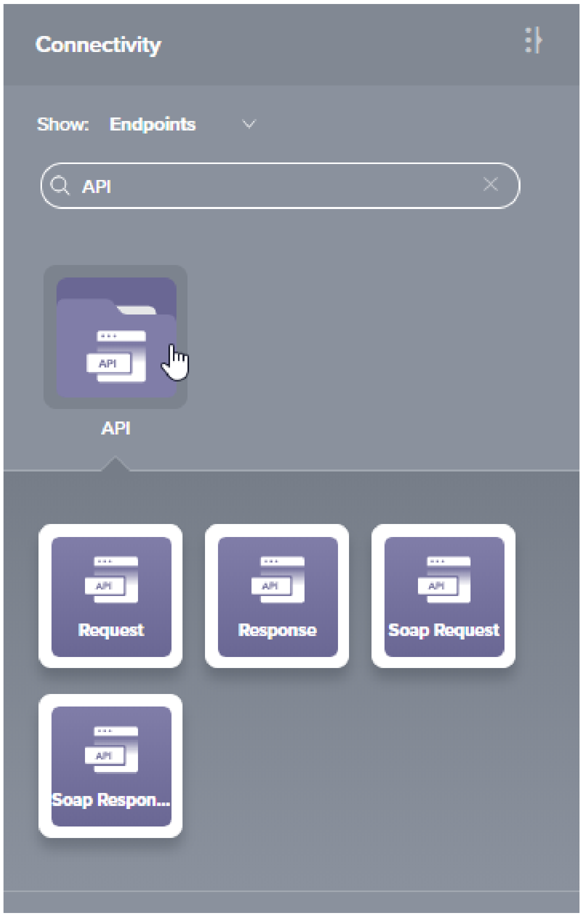
The API SOAP request or response activities can be configured using a WSDL schema only. If instead you want to interact with an API connection using a JSON, XML, CSV, or XSD schema, use a non-SOAP API request or response activity. Or if you want to interact with a SOAP connection without using an API, use a SOAP endpoint.

Using the preconfigured API connection, you can configure any number of API SOAP activities associated with a Jitterbit Custom API.

Creating an API SOAP Activity

From the design canvas, open the **Connectivity** tab of the design component palette:

Within the **Endpoints** filter, click the preconfigured API connection block to display activities that are available to be used with an API connection:



To create an activity that can be configured, the activity must first be added to an operation on the design canvas. To add an activity to an operation, drag the activity block from the palette to the operation.

For more information about the parts of an operation and adding activities to operations, see Operation Creation and Configuration.

On This Page

Related Articles

- API Connection
- API Request or Response Activity
- Component Dependencies, Deletion, and Removal
- Component Deployment
- Component Groups
- Component Palette
- Design Canvas
- Operation Creation and Configuration
- Operation Deployment and Execution
- Operation Validity
- Plugins Added to an Activity
- Project Pane
- Transformation Creation and Configuration

Related Topics

- API
- API Manager
- Cloud Studio
- Connectors
- Operations
- Scripts
- Transformation Mapping
- Transformations
- Variables

Last updated: Jan 17, 2020

Accessing Menu Actions

After an API SOAP activity has been added to an operation, menu actions for that activity are accessible from the project pane in both the **Workflows** and **Components** tabs, and from the design canvas:

- **Project Pane:** In the **Workflows** or **Components** tab of the project pane, hover over an activity name and click the actions menu icon ••• to open the actions menu.
- **Design Canvas:** Within the operation, click an existing activity block to open the actions menu.

Each of these menu actions is available:

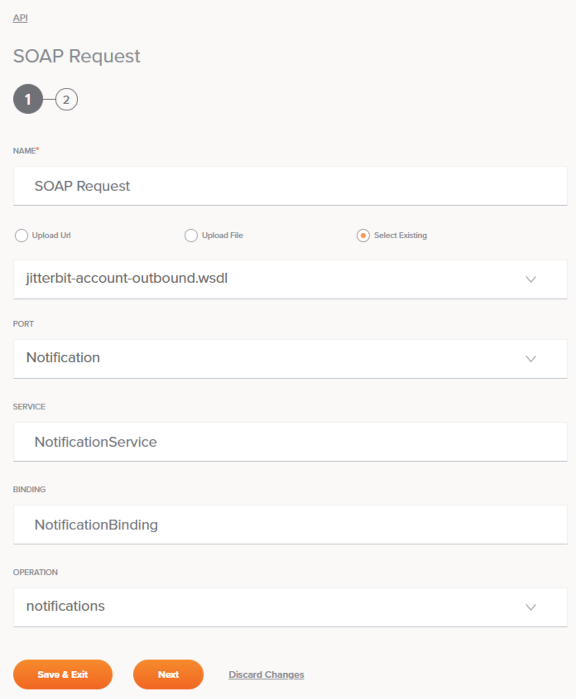
- **View/Edit:** This opens the activity configuration screen for you to configure the activity. For details, see Configuring an API SOAP Activity later on this page.
- **Delete:** This is used to permanently delete the activity (see Component Dependencies, Deletion, and Removal).
- **Rename:** This positions the cursor on the activity name in the project pane for you to make edits.
- **View Dependencies:** This changes the view in the project pane to display any other parts of the project that the activity is dependent on (see Component Dependencies, Deletion, and Removal).
- **Remove:** Available only from the actions menu on the design canvas, this removes the activity as a step in the operation without deleting it from the project. When you remove an activity that is adjacent to a transformation, if schemas are provided within that activity, they will no longer be referenced by the transformation. Removed components can be accessed or permanently deleted from the project pane (see Component Dependencies, Deletion, and Removal).
- **Deploy:** This deploys the activity and any components it is dependent on (see Component Deployment).
- **Configurable Deploy:** This opens the deployment screen, where you can select project components to deploy (see Component Deployment).
- **Add to group:** This opens a prompt to create a new custom group or to add the component to an existing group. Custom groups are an organizational tool to help organize a project (see Component Groups).
- **Duplicate:** This creates a copy of the activity as a new, unreferenced component. Upon creating the component copy, the cursor is positioned on the component name within the project pane for you to rename the component.

Configuring an API SOAP Activity

Follow these steps to configure an API SOAP request or response activity:

- Step 1 – Enter Information and Provide File Schema
- Step 2 – Review Data Schema

Step 1 – Enter Basic Information and Provide WSDL Schema



API

SOAP Request

1 2

NAME*
SOAP Request

○ Upload Url ○ Upload File ● Select Existing

jitterbit-account-outbound.wsdl

PORT
Notification

SERVICE
NotificationService

BINDING
NotificationBinding

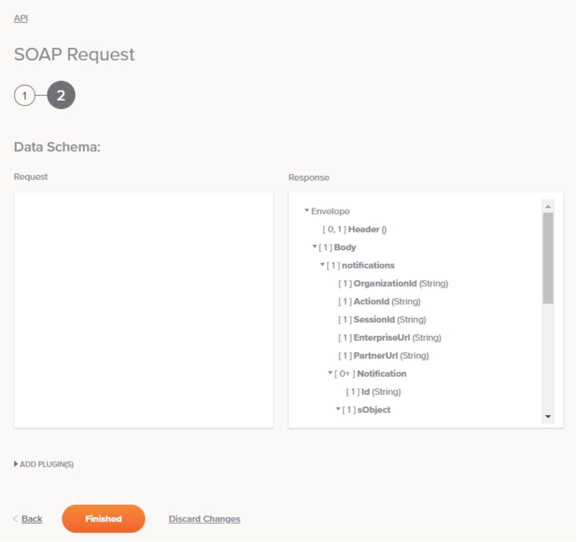
OPERATION
notifications

- **Name:** Enter a name to use to identify the API SOAP activity. The name must be unique for each API SOAP request or response activity and must not contain forward slashes (`/`) or colons (`:`).
- **Upload URL**, **Upload File**, or **Select Existing:** Use the radio button to select the source of the WSDL (Web Services Description Language file) to upload or reuse. Files up to 5 MB in size can be uploaded. At this time, using a WSDL containing cyclical references is not supported.
    - **Upload URL:** Enter the URL into the text box, then click the **Upload** button.
    - **Upload File:** Use the **Browse** button to the right to browse to a WSDL, then click the **Upload** button.

    > **NOTE:** Any `schemaLocation` must be resolved to a local file using a relative reference. This usually means that instead of supplying a tuple such as this:
    >
    > ```
    > xsi:schemaLocation='http://schemas.xmlsoap.org/wsdl
    > /mime/ http://ws-i.org/profiles/basic/1.1/wsdlmime-
    > 2004-08-24.xsd'
    > ```
    >
    > Instead, you would supply this:
    >
    > ```
    > xsi:schemaLocation='http://schemas.xmlsoap.org/wsdl
    > /mime/ wsdlmime-2004-08-24.xsd'
    > ```
    >
    > In the above example, the XSD file `wsdlmime-2004-08-24.xsd` is located either in the same directory or a path relative to the WSDL being loaded.

    - **Select Existing:** Use the dropdown to select from an existing WSDL that has previously been used in the current project.
- **Port:** Use the dropdown to select the appropriate port.
- **Service** & **Binding:** By default, these fields will autopopulate based on the provided WSDL and port and cannot be edited.
- **Operation:** Use the dropdown to select the method from the provided WSDL that you want to execute with this activity.
- **Save & Exit:** If enabled, click to save the configuration for this step and close the activity configuration.
- **Next:** Click to temporarily store the configuration for this step and continue to the next step. The configuration will not be saved until you click the **Finished** button on the last step.
- **Discard Changes:** After making changes, click **Discard Changes** to close the configuration without saving changes made to any step. A message will ask you to confirm that you want to discard changes.

Step 2 – Review Data Schema



```
▾ Envelope
    [ 0, 1 ] Header ()
  ▾ [ 1 ] Body
    ▾ [ 1 ] notifications
        [ 1 ] OrganizationId (String)
        [ 1 ] ActionId (String)
        [ 1 ] SessionId (String)
        [ 1 ] EnterpriseUrl (String)
        [ 1 ] PartnerUrl (String)
      ▾ [ 0+ ] Notification
          [ 1 ] Id (String)
        ▾ [ 1 ] sObject
```

- **Data Schema:** The request and/or response data schemas are displayed. If the operation uses a transformation, the data schemas will be displayed again later during the transformation mapping process, where you can map to target fields using source objects, scripts, variables, custom values, and more.
- **Add Plugin(s):** Plugins are Jitterbit- or user-provided applications that extend Harmony's native capabilities. To apply a plugin to the activity, click to expand this section and select the checkbox next to the plugin to be used. For additional instructions on using plugins, including details on setting any required variables used by the plugin, see Plugins Added to an Activity.
- **Back:** Click to temporarily store the configuration for this step and return to the previous step.
- **Finished:** Click to save the configuration for all steps and close the activity configuration.
- **Discard Changes:** After making changes, click **Discard Changes** to close the configuration without saving changes made to any step. A message will ask you to confirm that you want to discard changes.

Next Steps

After configuring an API SOAP request or response activity, you can use it within an operation as described below. Once the operation is set up, make sure to include it in the configuration of the Jitterbit Custom API to expose the operation or set of operations as a consumable REST endpoint.

Completing the Operation

After configuring an API SOAP request or response activity, complete the configuration of the operation by adding and configuring other activities, transformations, or scripts as operation steps (see Operation Creation and Configuration). You can also configure an operation's operation settings, which include the ability to chain operations together that are in the same or different workflows (see Operation Settings).

API SOAP request activities can be used as a source with these operation patterns:

- *Archive Pattern*
- *Transformation Pattern*
- *Two-Target Archive Pattern*
- *Two-Transformation Pattern*

API SOAP response activities can be used as a target with these operation patterns:

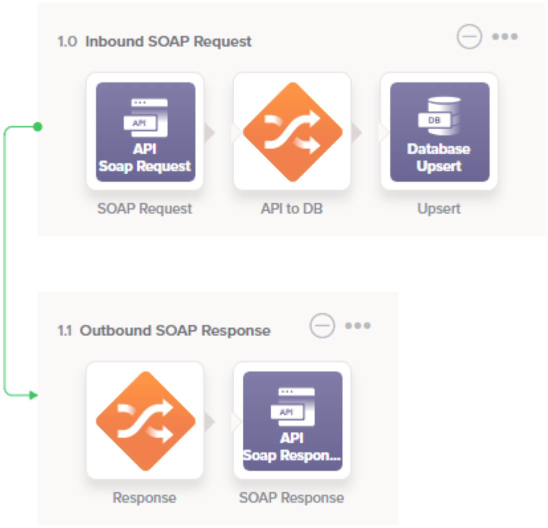
- *Archive Pattern*
- *Script Pattern*
- *Transformation Pattern*
- *Two-Target Archive Pattern* (as the second target only)
- *Two-Transformation Pattern* (as the second target only)

Other patterns are not valid using API SOAP activities. See the validation patterns on the Operation Validity page.

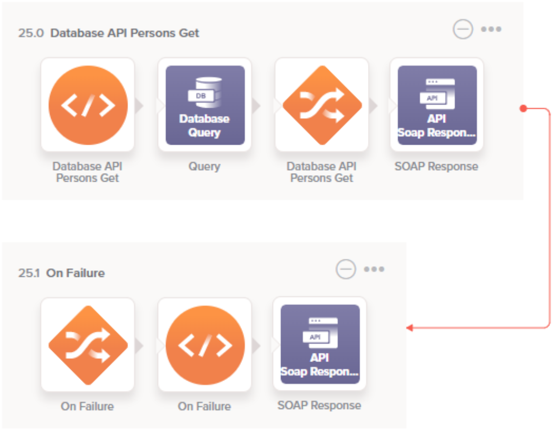
When used in an operation chain, no more than one API or API SOAP request activity can be present. In addition, the API or API SOAP request activity must be the source of the first operation. That is, no other operation may be calling this operation from a script or an "on success" or "on failure" operation action.

⚠ **CAUTION:** Operations that begin with an API request cannot be executed manually using the operation **Deploy and Run** option because they retrieve the source data externally.

This example depicts how API SOAP request and response activities might be used within an operation chain:



Another setup uses the same API SOAP response activity twice within an operation chain, shown below, which can be used for handling different error responses. For example, this setup provides the ability to override HTTP error code responses for Custom APIs using the Jitterbit variable `jitterbit.api.response.status_code`, set in a Jitterbit Script:



Other examples using API or API SOAP activities as sources or targets in an operation include Capturing Data Changes with a Harmony API or HTTP Endpoint and Configuring Outbound Messages with Harmony API (These patterns use Design Studio as an example, but the same concepts can be applied in Cloud Studio).

When ready, deploy and run the operation (see Operation Deployment and Execution) and validate behavior by checking the operation logs (see Operation Logs).

Configuring the Jitterbit Custom API

After the operation setup is complete, make sure to complete the configuration of the Jitterbit Custom API.

Jitterbit Custom APIs are configured within the Harmony API Manager, accessible through the Jitterbit Harmony Portal for eligible subscribers. To configure a Jitterbit Custom API, refer to documentation under API Manager.

Note that after you have configured a Jitterbit Custom API to call a Cloud Studio operation, you will not be able to delete the operation without first changing the API's configuration so it no longer calls the operation.