

# SOAP Clients

- [Adding a SOAP Client](#)
- [Configuring a SOAP Client](#)

SOAP Clients are tools that check the functionality of SOAP-based services. You can add SOAP Client tools to your test scenarios and configure them to suit your testing needs. You can then create and run a job for executing the related test scenarios and with the appropriate environment provisioned. Use these tests for checking the health of specific components and for automated testing during provisioning.

The SOAP Client interface in CTP is designed to quickly enable you to create new SOAP Client tools. Use the SOAtest desktop to create more advanced SOAP Clients with more complex configurations, such as message payloads defined by custom scripts. See [SOAP Client](#).



## Creating SOAP Clients from Traffic

If you have a traffic file that captures traffic for a SOAP service, you can generate a test scenario with SOAP Clients configured to test the recorded traffic. For details, see [Creating New Test Scenarios and Test Suites](#).

## Adding a SOAP Client

1. In the left pane, select the .tst or test suite where you want the new SOAP client added.
2. Choose **Add SOAP Client** from the page-level action menu.



3. (Optional) Modify the name of the newly-created tool.
4. [Configure the tool](#) as described below and click **Save**.

The new tool will be added at the end of the selected test suite.

## Configuring a SOAP Client

1. (Optional) If you want to associate the client with requirements, tasks, or other development artifacts, click the + button in the requirements section and add details when prompted.



**SOAP Client** emdemo > CDtr.tst

**SOAP Client**

Save Cancel

Name: SOAP Client

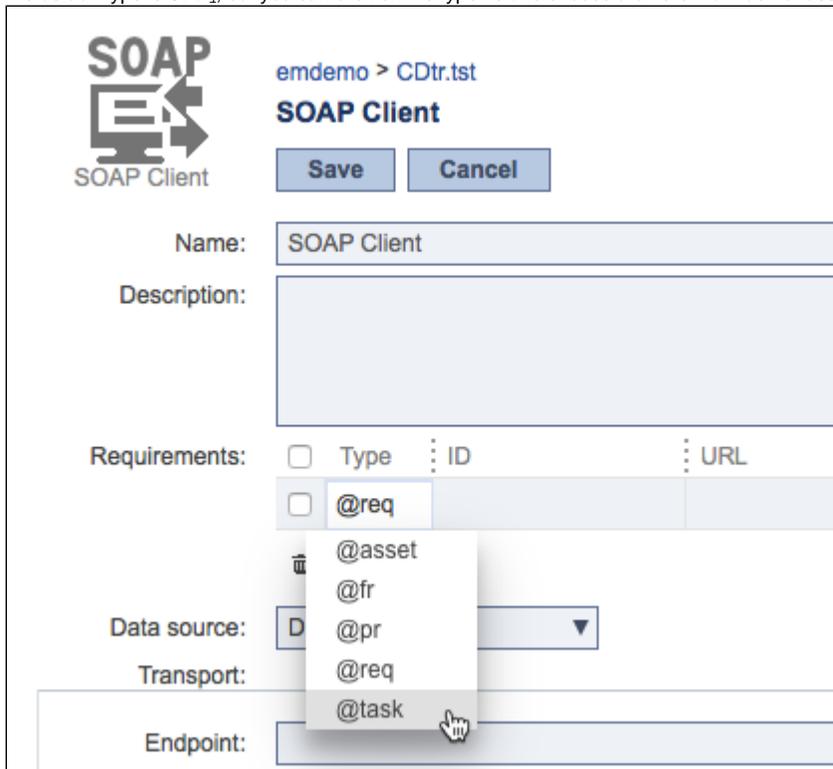
Description:

Requirements:

Type	ID	URL
+		

Data source: DOCS-DS-1

The default type is @req, but you can click on the type field to choose a different artifact for associations.



**SOAP Client** emdemo > CDtr.tst

**SOAP Client**

Save Cancel

Name: SOAP Client

Description:

Requirements:

Type	ID	URL
<input type="checkbox"/> @req		
<input type="checkbox"/> @asset		
<input type="checkbox"/> @fr		
<input type="checkbox"/> @pr		
<input type="checkbox"/> @req		
<input type="checkbox"/> @task		

Data source: D

Transport:

Endpoint:

- If the containing test suite includes multiple data sources, you can review and change which data source is used to parameterize this SOAP Client. See [Parameterizing with Data Source and Data Bank Values](#) for details.

SOAP Client

emdemo > CDtr.tst

**SOAP Client**

Save Cancel

Name: SOAP Client

Description:

Requirements:

Type	ID	URL
+		

Data source: DOCS-DS-1

Transport:

- Specify the endpoint of the service to invoke in the **Endpoint** field.
- Specify the action you want performed in the **SOAP action** field.
- (Optional) You can configure the client so that it succeeds with HTTP response codes outside the 2xx range. Specify single codes and/or code ranges as a comma-separated list in the **Valid response** field. For example, if you use "302, 500-599", a 302 code or any code in the 5xx range will be accepted. If you're using a parameterized value, be sure that the value in the data source uses this same format (e.g., "302, 500-599").
- (Optional) Specify a length of time in ms before the client timeouts if it does not receive a response.

Transport:

Endpoint:

SOAP action:

Valid response: 200-299

Timeout (ms): 30000

- (Optional) If the service requires authentication, enable the **Enable HTTP authentication** option and specify login credentials.

Enable HTTP authentication

Authentication: Basic

Username: username

Password: .....

- (Optional) Add HTTP headers if you want to override the default headers. See [HTTP Headers](#) for additional information. Choose **Literal** or **Table** mode from the HTTP headers drop-down menu to enter values in a text area or build the header using a table format.

Literal mode

HTTP Headers: Literal

Accept: application/json  
Content-type: text-json; charset=UTF-8

### Table mode

HTTP Headers: Table

<input type="checkbox"/> Header	Value
<input type="checkbox"/> Accept	application/json
<input type="checkbox"/> Content-type	text-json; charset=UTF-8

+

- (Optional) If you want to reset cookies before sending a request, enable the **Rest existing cookies before sending request** option.
- In the Request area, specify the message request that will be sent. Specify the message in the literal text editor, the JSON editor, or the XML editor (see [Editing JSON Messages](#) and [Editing XML Messages](#) for details and tips).

Input mode: XML Editor

MIME type: text/xml

A toolbar with three icons: a double-headed vertical arrow, a double-headed horizontal arrow, and a square with 'NS' inside.

```
< SOAP-ENV:Envelope />
  < SOAP-ENV:Body />
    < updateItemInCart />
      < cartId > 0 </>
      < itemId > 0 </>
      < quantity > 0 </>
```

#### Tip: Using {var\_name}

If you're familiar with SOAtest, you can use the standard `$(var_name)` notation to reference environment variables, test suite variables, and data source values that are defined for the test scenario. This applies to both URL and Payload.

When you are working with JSON, a special notation is used for parameterizing a number or boolean field within a JSON message: `$(number:<value>)` or `$(boolean:<value>)`. For example, to parameterize a number field with the column Count, you would use `$(number:Count)`.