

URL Adapter

PAS 23.1.1 With the URL adapter, it is possible to get and post messages from and to a web endpoint. The table below lists what you can do with the URL adapter and what protocols are supported.

Task	Protocols	Adapter Action	Description	Documentation Reference
Send a request to a server	<ul style="list-style-type: none">http, httpsldap, ldapsftp, ftps	request	Send a request to a given URL using the given parameters. The response data is stored in memory.	
		streamToFile	Send a request to a given URL using the given parameters. The response data is stored to a file. Streaming is mostly relevant in FTP context. In order to avoid consuming too much memory, the URL adapter offers the option to receive data in little chunks. This mechanism is called streaming: After receiving a small data chunk from the server, it is immediately written to a file before receiving the next one.	
		streamFromFile	Send a request to a given URL using the given parameters. The request data is read from a file. Streaming is mostly relevant in FTP context. In order to avoid consuming too much memory, the URL adapter offers the option to send data in little chunks. This mechanism is called streaming: The request data is available as a file and send in small data chunks.	

The input and output of this adapter is always of type **Blob** to support all possible data types (e.g. binary for images, text with HTML code, XML messages, etc.).

Static and Dynamic Requests

URL request can be **static** or **dynamic**.

- Static means that the whole backend information (host, path, protocol, etc.) is defined on the URL alias, and is not modified on the **request** action.
- For dynamic requests, the default information provided by the URL alias can be overridden on the **request** action before sending the request to the host. This is sometimes required, for example, when URL parameters must be calculated or are known at runtime only. Refer to [URL Adapter Reference](#) for more information on the URL adapter parameters.

If large amounts of data have to be transferred, the URL adapter offers the possibility of streaming the data from or into a file so avoiding to utilize large memory chunks. [Streaming FTP](#) explains how to do this for the ftp protocol. Applying streaming to the other protocols works analogously.

Find an overview on all attributes and parameters of the URL adapter on [URL Adapter Reference](#).

HTTP Headers

With xUML service adapter calls, the xUML Runtime adds the following outgoing HTTP headers containing correlation information to the request:

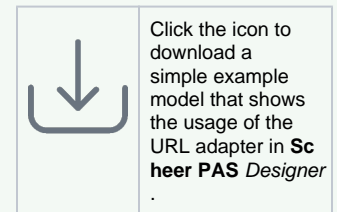
- **X-Transaction-Id** or **xTransactionId** (in JMS context)
This header identifies the transaction the call belongs to. You can set the transaction id manually with [setTransactionID](#). If not set, the Runtime will generate one.
This header will be passed through the callstack to identify all service calls that belong to a transaction.
- **X-Request-Id**
This header identifies the unique request. The Runtime generates a unique number for each adapter call.
- **X-Sender-Host** and **X-Sender-Service**
These headers contain the sender host resp. the sender service. They are set by the Runtime automatically.

Transaction id and request id will be logged to the transaction log on the adapter call (refer to [Administration Guide > Contents of the Transaction Log](#) for more details). Having this information, you can use this for error analysis or usage metrics.

On this Page:

- [Static and Dynamic Requests](#)
- [HTTP Headers](#)
- [Adding an URL Adapter Operation to a Diagram](#)
- [Configuring the URL Adapter Operation](#)

UrlAdapter_Example



Related Pages:

- [Using the URL Adapter with the HTTP Protocol](#)
- [Using the URL Adapter with the FTP Protocol and Related Protocols](#)
 - [FTPS](#)
- [Using the URL Adapter with the LDAP Protocol](#)
- [Setting cURL Options on the URL Adapter](#)
- [URL Adapter Reference](#)

Related Documentation:

- [Administration Guide](#)
 - [Contents of the Transaction Log](#)

Adding an URL Adapter Operation to a Diagram

Search Criteria

Assets

Base Types

Bridge Base

Base Components

Add Ons

FileSystem

FlatFile

HTTP

JSON

Kafka

Logger

Memory

MongoDB

REST

S3

SAP

SQL

URL

Authentication

Certificate

HeaderField

Key

Option

Proxy

SSL

URL

Utilities

FTP

FTPS

HTTP

HTTPS

LDAP

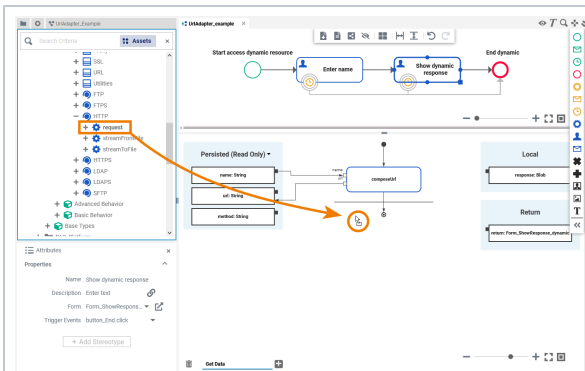
LDAPS

SFTP

Advanced Behavior

Basic Behavior

Expand the path to the URL adapter in the service panel (**Base Types /Bridge Base /Base Components/Add Ons/URL**).



You can drag out operations from the data model to any diagram:

- BPMN execution diagram
- mapping diagram
- activity diagram

The example on the left shows how to add an URL adapter operation to a BPMN execution diagram.

Configuring the URL Adapter Operation

Once an operation has been added to a diagram, it needs to be configured as an URL adapter.

Attributes

Properties

Name request

+ Add Stereotype

Symbol Type Execution Step

Select the newly added URL adapter operation and switch to the **Attributes** panel. Depending on the diagram type you can see the following information (example BPMN execution diagram):

Attribute	Description	Allowed Values / Example
Name	The name of the URL adapter operation.	request
Symbol Type	Operations added to an execution diagram are execution steps.	Execution Step

All this is predefined and cannot be changed.

Attributes

Properties

Name request

+ Add Stereotype

Symbol Type Execution Step

Click **Add Stereotype** to define the selected operation as to be an URL adapter.

Select Stereotype

☐ Flat File Adapter

☐ Kafka Producer Adapter

☐ Logger

☐ Memory Adapter

☐ MongoDB Adapter

☐ REST Adapter

☐ S3

☐ SAP IDoc Composer

☐ SAP IDoc Parser

☐ SAP IDoc Record Composer

☐ SAP IDoc Record Parser

☐ SAP RFC Adapter

☐ SAP TRFC Adapter

☐ SAP TRFC Confirm Transaction

☐ SAP TRFC Create Transaction

☐ SAP XML IDoc Composer

☐ SAP XML IDoc Parser

☐ SQL Adapter

☒ URL Adapter

Save

Cancel

Select **SQL Adapter** from the list of available adapter stereotypes.
Click **Save**.

Attributes

Properties

Name request

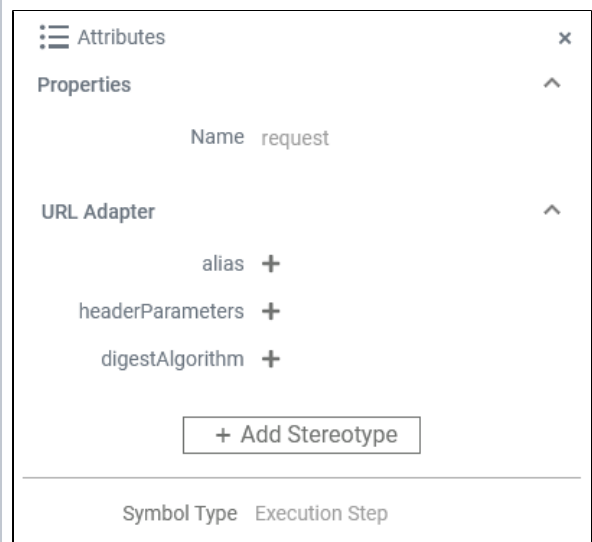
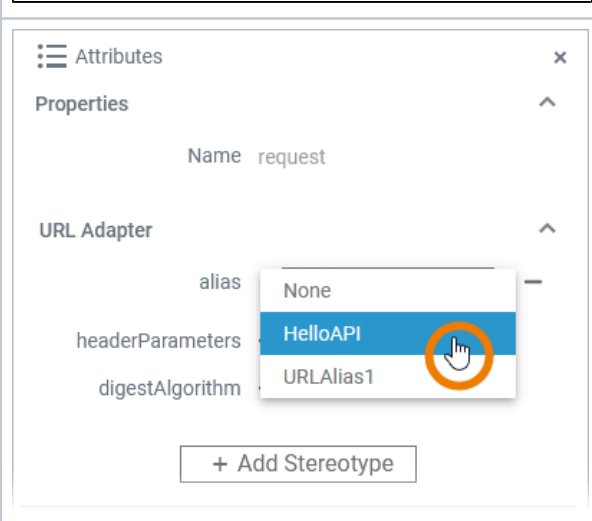
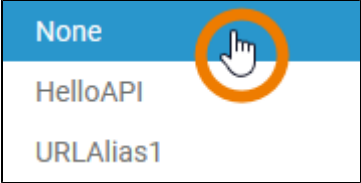
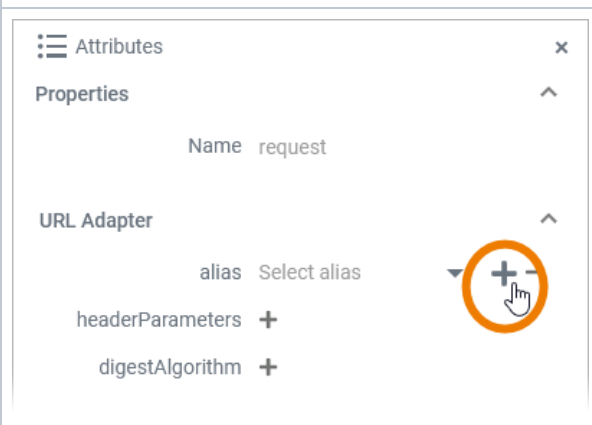
URL Adapter

+ Add Stereotype

Symbol Type Execution Step

The **Attributes** panel shows the added adapter stereotype. Now you still need to configure the adapter.

Expand the stereotype by clicking the arrow on the right.

	<p>An URL adapter is configured via its alias (see Aliases for more information on aliases).</p> <p>Additionally, you can insert</p> <ul style="list-style-type: none"> • headerParameters • digestAlgorithm <p>See further information below and on page URL Adapter Reference.</p>
	<p>You can select an existing alias from a drop-down list by clicking the text Select alias.</p> <p>If you want to remove an added alias, select None from the drop-down list:</p> 
	<p>You can also create and add a new alias by clicking the corresponding icon +.</p> <p>Refer to Aliases for more information on how to create a new alias.</p>

Add Alias

Name:

Standard ▼

Advanced ▼

Proxy ▼

SSL ▼

A dialog opens where you can name and configure the new alias. Refer to [URL Adapter Reference](#) for more information on the configuration options of this adapter.

Click **Save** to create and add the new alias.

Attributes ×

Properties ^

Name request

URL Adapter ^

alias Select alias ▼ + -

headerParameters -

digestAlgorithm +

Symbol Type Execution Step

To add **headerParameters** click on the corresponding + icon and enter additional header information for the URL adapter call in form of name and value pairs.

Refer to [URL Adapter Reference](#) for more details.

Attributes ×

Properties ^

Name request

URL Adapter ^

alias Select alias ▼ + -

headerParameters Enter text -

digestAlgorithm -

Symbol Type Execution Step

None

MD5

SHA

SHA-1

SHA-256

SHA-512

To select a **digestAlgorithm**, click icon + and select an algorithm from the list. When applied, a digest header is generated using the specified algorithm, and sent with the request.

Refer to [URL Adapter Reference](#) for detailed explanations of all options.