


# URL Adapter

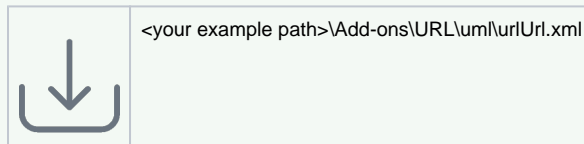
 This page explains the **URL Adapter** in Bridge context. If you were looking for the same information regarding the [PAS Designer](#), refer to [URL Adapter](#) in the Designer guide.

With the URL adapter, it is possible to get and post messages via the protocols listed below. 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.).



We differ between **static** and **dynamic** requests. Static means that the whole backend information (host, path, protocol, etc.) is defined in the component diagram and is not modified when called within an action state.

A dynamic call means that the definition is kept in the component diagram, but is modified 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.

### Example File (Builder project Add-ons/URL):



The URL adapter supports the following protocols:

Protocol	Description	Example
<a href="#">ldap</a>	Via the URL adapter, it is possible to connect an LDAP server to retrieve user information. You can also access a Microsoft Active Directory Service (ADS) if it provides an LDAP interface. An E2E example model is provided with the documentation (see example reference).	<div><div>Example File (Builder project Add-ons/URL):</div><div> <code>&lt;bridgeDocPath&gt;\Add-ons\URL\uml\urlLdap.xml</code></div></div>
<a href="#">ftp</a>		<div><div>Example File (Builder project Add-ons/URL):</div><div> <code>&lt;bridgeDocPath&gt;\Add-ons\URL\uml\urlFtp.xml</code></div></div>
<a href="#">ftps</a> <a href="#">sftp</a> <a href="#">http</a> <a href="#">https</a> <a href="#">file</a> <a href="#">gopher</a> <a href="#">telnet</a>		

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. The [Streaming FTP](#) section explains how to do this for the ftp protocol. Applying streaming to the other protocols works analogously.

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

### On this Page:

- [HTTP Headers](#)

### Related Pages:

- [Using the URL Adapter with the LDAP Protocol](#)
- [Using the URL Adapter with the FTP Protocol and Related Protocols](#)
  - [SFTP](#)
  - [FTPS](#)
- [Using the URL Adapter with the HTTP Protocol](#)
- [Using the URL Adapter with the File Protocol](#)
- [Accessing URL Resources via the URL Adapter](#)
- [Setting cURL Options on the URL Adapter](#)
- [URL Adapter Reference](#)
- [Providing the SOAP Adapter with URL Parameter](#)

# HTTP Headers

Runtime 2019.9 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. Having this information, you can use this for error analysis or usage metrics.