

URL

All URL adapter parameters, that are specified on the **URL alias**, can also be supplied dynamically via action script. Find below a list of all relevant parameters of the URL adapter and their description.

With the URL adapter, it is possible to get and put messages via file transfer protocols: **FTP**, **FTPS** and **SFTP**. These protocols often get confused and mixed up due to the similarity in function and names. Please refer to [Using the URL Adapter with the FTP Protocol and Related Protocols](#) for a distinction and more information on the usage with the URL adapter.

Stereotype Attributes

URL Alias

Attribute	Description	Allowed Values / Examples	
Name	Specify a name for the alias.	any string	
Standard			
protocol	Transport protocol.	one of: ldap, ldaps, file, ftp, ftps, sftp, gopher, http (default), https, telnet	
port	Machine port number the service is binding to. This port number can be given at service level only.	a valid port number (default: 80)	
path	HTTP path for the SOAP request.		
host	Host name. Default is "localhost" if no value is specified.	a valid hostname (default: localhost)	
method	HTTP method of the URL adapter call. Default is "POST" if no value is specified.	one of: get, post (default), read, write, put, list, patch, head, options, delete	
Advanced			
user	Username/password.		
followRedirects	The maximum number of redirects to follow.	any positive integer or 0	
ignoreHttpErrors	If true, HTTP error codes > 300 will not cause an exception in the model. This implies, that the response body is accessible even if HTTP errors occur. The default value is false.	true	Ignore HTTP error codes > 300.
		false	Do not ignore HTTP errors (default).
options	Native cURL options as listed on Setting cURL Options on the URL Adapter . Use one of the following syntax rules: <ul style="list-style-type: none">values separated by ' , ' in one linevalues separated by ' ' in one linelist of tagged values	valid cURL options , e.g. CURLOPT_TIMEOUT=20	
requestHttpHeaderRoles	In the context of HTTP based adapters (URL , REST), enable automatic header generation for the listed headers. These definitions overwrite the default behavior, and X-Transaction-Id , X-Request-Id , X-Sender-Host and/or X-Sender-Service will be substituted by this definition. requestHttpHeaderRoles can hold a list of definitions in format <http header name>:<role>, that will automatically be generated for each adapter call on this alias. <role> can be one of the listed allowed values (one list entry per line). Refer to HTTP Header Support > Overwriting the Standard HTTP Headers for more details on header roles.	client_host	Provide the client host in a header <http header name> instead of X-Sender-Host .
		client_service	Provide the client service in a header <http header name> instead of X-Sender-Service .
		correlation_id	Provide the correlation ID in a header <http header name> instead of X-Request-Id .
		transaction_id	Provide the transaction ID in a header <http header name> instead of X-Transaction-Id .

On this Page:

- [Stereotype Attributes](#)
 - [URL Alias](#)
 - [URL Adapter](#)
- [URL Adapter Operations](#)
 - [Incoming Parameters](#)
 - [Outgoing Parameters](#)
- [URL Utility Functions](#)
- [URL Adapter Parameter Types](#)
 - [Authentication](#)
 - [Certificate](#)
 - [Entry](#)
 - [HeaderField](#)
 - [Key](#)
 - [Option](#)
 - [Proxy](#)
 - [SSL](#)
 - [URL](#)


UrlAdapter_Example



Click the icon to download a simple example model that shows the usage of the URL adapter in **Sc heer PAS Designer**.

Related Pages:

- [Using the URL Adapter with the FTP Protocol and Related Protocols](#)




		passthrough	Pass a present header <http header name> to the called service.
		passthrough=<request header name>	Pass an incoming header <request header name> to the called service under the name of <http header name>. This is equivalent to renaming a header.
digestAlgorithm	Generates a HTTP digest header using the specified algorithm. When applied, a digest header is generated using the specified algorithm, and sent with the request. The generated header conforms with RFC3230 and RFC5843. <div> Only one value is supported (no multi-value header).</div>	None	No header generated (default).
		MD5	Generate header using MD5 algorithm.
		SHA	Generate header using SHA algorithm.
		SHA-1	Generate header using SHA-1 algorithm.
		SHA-256	Generate header using SHA-256 algorithm.
		SHA-512	Generate header using SHA-512 algorithm.
Proxy			
proxyURL	URL of the proxy server.		
proxyUser	Proxy user.		
proxyType	Type of the proxy	one of: HTTP, SOCKS5	
SSL			
sslCAInfo	Name of the file containing additional certificates for the connection verification (e.g. additional root CAs).		
sslCertificateFile	Name of the file that contains the client certificate.		
sslPrivateKeyFile	Name of the file that contains the private key.		
sslPrivateKeyPassword	Password for the private key.		
sslCertificateType	Type of the certificate.	one of: PEM (default), DER	
sslVerifyHost	Whether to verify the host information from the SSL connection.	On	Verification on.
		Off	Verification off (default).
		Existence	Limit verification to the mere existence of the host.
sslVerifyPeer	Whether to verify the peer information from the SSL connection.	On	Verification on (default).
		Off	Verification off.
sslPrivateKeyType	Type of the private key.	one of: PEM (default), DER	

URL Adapter

The Runtime uses **Transfer-encoding: chunked** for POST requests. You can override this behavior by setting

- Transfer-Encoding: <empty string>
- Content-Length: <length of the body>"

Attribute	Type	Description	Allowed Values
-----------	------	-------------	----------------

alias		Specify the URL alias the adapter should use to establish the connection.	any valid URL alias	
headerParameters	Array of HeaderField	<p>Supply additional header information for the URL adapter call in form of name and value pairs (see type HeaderField below).</p> <div>  The xUML Runtime assigns a correlation ID to each adapter call. This ID is stored in header field X-Bridge-CorrelationID. Adapter calls can be identified by this ID. Also, it is logged to the transaction log. </div> <div>  If you set this header manually, the new header will be send to the target URL. Nevertheless, the correlation ID logged to the transaction log will still be the one generated by the Runtime! </div>		
digestAlgorithm		<p>Generates a HTTP digest header using the specified algorithm. When applied, a digest header is generated using the specified algorithm, and sent with the request. The generated header conforms with RFC3230 and RFC5843.</p> <div>  Only one value is supported (no multi-value header). </div>	None	No header generated (default).
			MD5	Generate header using MD5 algorithm.
			SHA	Generate header using SHA algorithm.
			SHA-1	Generate header using SHA-1 algorithm.
			SHA-256	Generate header using SHA-256 algorithm.
			SHA-512	Generate header using SHA-512 algorithm.

URL Adapter Operations

In general, three adapter operations can be used: **request**, **streamFromFile** and **streamToFile**. Exceptions are **ldap** and **ldaps** where **streamToFile** is not available. Depending on the protocol used, the parameters of the operation differ slightly.

Incoming Parameters

Attribute	Type	Direction	Restrictions		Description
			to listed protocol only	to listed method only	
authentication	Authentication	in	ldap, ftp, ftps, sftp, http, https		This parameter provides an object of type Authentication containing the user and the password.
commands	Array of String	in	ftp, ftps		Use this parameter to specify ftp commands. All commands are executed after executing the ftp method. For a list of available ftp commands see RFC 959 .
content	Blob	in		post, put, write	Use this parameter to supply the content to be used by the listed method.

followRedirects	Integer	in	http, https		Specifies the maximum number of redirects to follow.
headerParameters	Array of HeaderField	in	http, https		Use this parameter to supply additional header information for the URL adapter call in the form of name and value pairs.
ignoreHTTPErrors	Boolean	in	http, https		If true, HTTP error codes > 300 will not cause an exception in the model. This implies, that the response body is accessible even if HTTP errors occur. The default value is false.
method	String	in			This parameter provides the method of the URL adapter call.
options	Array of Option	in	ldap, ftp, ftps, sftp, http, https		Use this parameter to specify cURL options for the URL adapter call.
proxy	Proxy	in	ldap, ftp, ftps, sftp, http, https		Use this parameter to supply necessary proxy information.
ssl	SSL	in	https, ftps		Use this parameter to supply SSL information.
streamSource	String	in	ftp, ftps, sftp, http, https	post, put	If you want to use file streaming with streaming method toFile , supply the streaming target here. Then, response is not used.
streamTarget	String	in	ftp, ftps, sftp, http, https	get, post, put, list	If you want to use file streaming with streaming method toFile , supply the streaming target here. Then, response is not used.
url	String	in			This parameter contains the URL that will be accessed by the adapter. The URL contains protocol, port and path.

Outgoing Parameters

Attribute	Type	Direction	Restrictions		Description
			to listed protocol only	to listed method only	
httpHeaderMap	Map of Entry	out			Header information as a map. The map contains arrays of header value strings whereas the header name is the key of the map. <ul style="list-style-type: none"> Header names are lowercase and treated case insensitive. Multiple headers with the same name are treated as arrays. Refer to HTTP Header Support for more information on the standard xUML HTTP headers.
httpStatus	Integer	out	http, https		Contains the HTTP status code of the response.
response	Blob	out		get, post, put, list, read	Contains the response content in relation to the used method.

URL Utility Functions

Function	Parameter	Direction	Return Value	Description	Example
parseUrl()	urlString of type String	in	object of type URL	Parses an URL given in a String variable into a data structure. For details see parseUrl() Function .	<pre>create aUrlObject; set aUrlObject = aUrlString. parseUrl();</pre>

composeUrl()	urlStructure of type URL	in	String	Builds a String containing a URL from an object of type URL . For details see composeUrl() Function .	<code>aUrlString = composeUrl(aUrlObject);</code>
---------------------	------------------------------------------	----	--------	-------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------

URL Adapter Parameter Types

Authentication

Name	Type	Description
password	String	Password needed for authentication.
username	String	Username needed for authentication.

Certificate

Name	Type	CURL Option	Description	Allowed Values / Examples
file	String	CURLOPT_SSLCERT	Name of the file containing the client certificate.	
type	String	CURLOPT_SSLCERTTYPE	Type of the certificate.	PEM , DER

Entry

Name	Type	Description
key	String	The header name is the key of the map.
value	Any	Value of the header field.

HeaderField

Name	Type	Description
name	String	Name of the header field.
value	String	Value of the header field.

Key

Name	Type	CURL Option	Description	Allowed Values / Examples
file	String	CURLOPT_SSLKEY	Name of the file containing the private key.	
type	String	CURLOPT_SSLKEYTYPE	Type of the key.	PEM, DER
password	String	CURLOPT_KEYPASSWD	Password for the private key.	

Option

Name	Type	Description	Allowed Values / Examples
name	String	Native cURL option.	CURLOPT_CONNECTTIMEOUT
value	String	Value of the cURL option.	10

Proxy

Name	Type	CURL Option	Description	Allowed Values / Examples
url	String	CURLOPT_PROXY	URL of the proxy server.	
type	String	CURLOPT_PROXYTYPE	Type of the proxy.	HTTP, SOCKS5
authentication	Authentication	CURLOPT_PROXYUSERPWD	Proxy user.	

SSL

Name	Type	CURL Option	Description	Allowed Values / Examples	
verifyPeer	String	CURLOPT_SSL_VERIFY_PEER	Whether to verify the peer information from the SSL connection.	On	Verification on (default).
				Off	Verification off.
verifyHost	String	CURLOPT_SSL_VERIFY_HOST	Whether to verify the host information from the SSL connection.	On	Verification on (default).
				Off	Verification off.
				Existence	Limit verification to the mere existence of the host.
caInfo	String	CURLOPT_CAINFO	Name of the file containing additional certificates for the connection verification (e.g. additional root CAs).		
certificate	Certificate		Client certificate.		
key	Key		Private key of client certificate.		

URL

Name	Type	Description	Allowed Values / Examples
scheme	String	URL scheme.	<code>https:</code>
user	String	User.	
password	String	Password.	
host	String	Name of the host.	<code>acme.com</code>
port	Integer	Port number.	<code>8080</code>
path	String	Path.	<code>resources/logo.png</code>
query	String	Query parameters.	<code>?download=true</code>
fragment	String	URL fragment.	<code>#anAnchor</code>