Using the URL Adapter with the File Protocol

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.

The usage of the URL adapter with the file protocol is deprecated. To handle files in file systems we recommend using the <<FileSystemAdapter>> instead of the <<URLAdapter>> (see File System Adapter for more information).

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

<your example path>\Add-ons\URL\uml\urlUrl.xml

On this Page:

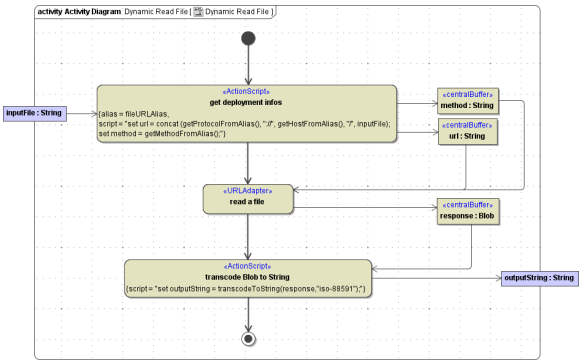
Related Pages:

- Setting cURL Options on the URL Adapter
- URL Adapter Reference

Dynamic Reading of Files

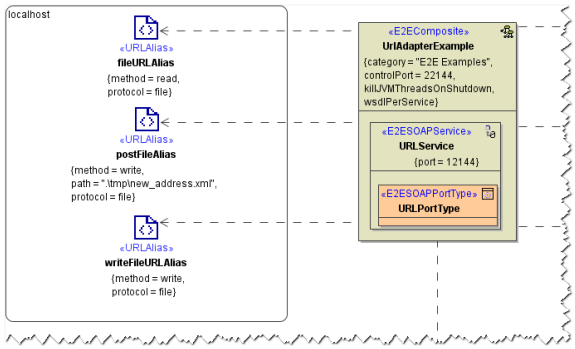
With the <<URLAdapter>> it is also possible to use dynamic parameters. The following example shows this using the file protocol. The dynamic parameter can also be used together with the HTTP protocol (GET or POST).
The activity diagram below shows how to manipulate the URL backend parameters, which are set in the component diagram.

Figure: Dynamic Read Request



The deployment macro getHostFromAlias() reads the name of the component instance; in this case localhost. The deployment macro getProtocolFromAlias() reads the tagged value protocol defined on the backend service artifact or backend interface artifact (in the present case the protocol value is file). The deployment macro `getMethodFromAlias()` reads the tagged value method defined on the backend service artifact or backend interface artifact (in the present case it is defined on the interface artifact and has the value read).

Figure: Component Diagram for Accessing a File via the URL Adapter



The example above reads a file, which is stored on the localhost (meaning the machine where this configuration instance is deployed to). The path with filename is given from outside (input parameter which is sent by the client) and is then combined with the parameter of the deployment diagram resulting in the complete URL. If the client sends as **inputFile** string **d:/mytext.txt** the resulting url would be: **file://localhost/d:/mytext.txt**

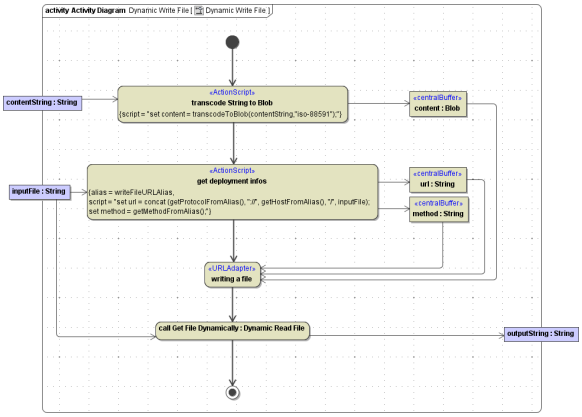
> When combining the dynamic FTP parameter **url** with the static configuration via **alias** one has to be aware that the user and password will be taken from the static component (tagged value **user** if defined) - even if it is defined in the dynamic definition of the <<URLAdapter>>. Wanting to have user and password directly in the URL (e.g. ftp://user:pw@ftpServer), the user and password must not be defined in the component diagram.

Writing Files

Writing to a file follows the same rules like reading one. An additional parameter is used called **content** of type **Blob**, which defines the content of the file, which should be written to the server. In the component diagram, you have also to define POST instead of Read.

The example below creates a file, reads the created file, and provides the client with the content of the file (which has been given as input string).

Figure: Writing a File Using the URL Adapter



Tagged Values

Find below a list of relevant tagged values, if the URL adapter is used with the file protocol.
Default values used when an option is not explicitly set are written in bold.

Tagged Value	Description	Values
protocol	Transport protocol.	file
method	File operation method.	read, write
path	File path. Tagged value **path** is optional. If it is omitted, the file will be written to/read from the Bridge data directory of the service. This implicates, that the file will be deleted, if the service gets deleted.	a valid file path, absolute or relative to the Bridge data directory.