File System

This page explains the **Filesystem Adapter** in Bridge context. If you were looking for the same information regarding the PAS Designer, refer to Filesystem Adapter in the Designer guide.

On this Page:

- Tagged Values
  - <<DirectoryAlias>>
  - <<FileAlias>>
- Parameters File Handling
  - Action "read"
  - Action "readLine"
  - Action "write"
  - Action "append"
  - Action "status"
  - Action "rename"
  - Action "remove"
  - Action "Copy"
  - Action "close"
- Parameters Directory Handling
  - Action "readDirectory"
  - Action "createDirectory"
  - Action "status"
  - Action "remove"
  - Action "removeRecursively"

Related Pages:

- File System Adapter
- Charset Definitions
- Importing File Resources
- Catching Errors

Tagged Values

<<DirectoryAlias>>

Tagged Value	Description	Allowed Values
directory	Specify a path to a directory. This directory will be treated as a base directory. It will be either used directly (for a File System Adapter operating directly on directories), or as a base directory for a path segment given by a dynamic file **name** parameter. It is not possible to jump out of the base directory by `../../aDirectory` tricks. The base directory can only be changed in the service settings on the Bridge.	Example: `./tmp/testDirectory`
fileEncoding	Specify the default encoding be used with the directory. This default encoding can be overwritten on adapter calls, using a dynamic **encoding** parameter, e.g. on File System Adapter calls reading lines, or with the Flat File Adapter.	Refer to Charset Definitions for possible values. default: `iso-8859-1`

<<FileAlias>>

Tagged Value	Description	Allowed Values
file	Specify name and path of the file you want to access.	Example: `.tmp/testDirectory/testFile.txt`
encoding	Specify the encoding of the file.	Refer to Charset Definitions for possible values. default: `iso-8859-1`
resource	Instead of a **file**, you can select a resource from the list of imported resources. The File System Adapter will then use the resource instead of **file**. The resource can be replaced changing the resource path in the **File System Adapter** settings of the xUML service. If **resource** is specified neither **file** nor **encoding** can be overridden dynamically. See Importing File Resources for more information on importing resources and xUML Service Settings for more information on changing the settings of a service.	

Parameters File Handling

Action "read"

Name	Type	Direction	Description

name	String	in	Fully qualified file name of the file you want to read, including the path. When using the Windows style with backward slashes "\" you have to be aware that you have escape this character. The escape character is also the "\". To avoid this, use forward slashes with Windows as well.
position	Integer	in	Position (in Bytes) from which you want to start reading the file.
length	Integer	in/out	Length (in Bytes) of the snippet of the file you want to read. The actual length of the read data is given back (e.g. on file end).
data	Blob	out	Content that has been read.

Action "readLine"

Name	Type	Direction	Description
name	String	in	Fully qualified file name of the file you want to read, including the path. When using the Windows style with backward slashes "\" you have to be aware that you have escape this character. The escape character is also the "\". To avoid this, use forward slashes with Windows as well.
encoding	String	in	Encoding of the file. Default encoding is **iso-8859-1**. For a list of possible encodings see Charset Definitions.
line	String	out	File content of the line that has been read. On file end, **line** is NULL.
length	Integer	out	Length (in Bytes) of the line that has been read. On file end, **length** is 0.

Action "write"

Name	Type	Direction	Description
name	String	in	Fully qualified file name of the file you want to write, including the path. When using the Windows style with backward slashes "\" you have to be aware that you have escape this character. The escape character is also the "\". To avoid this, use forward slashes with Windows as well.
position	Integer	in	Position (in Bytes) from which you want to start writing the file.
data	Blob		Content you want to write to the file.
length	Integer	in	Length (in Bytes) of the **data** you want to write, if you want to only write part of the Blob.

Action "append"

To action **append**, two modes can be applied:

- append
  If mode is empty or set to **append**, the content stored in **data** will be appended to the file specified in **name**. If the file does not exist, it will be created.
- overwrite
  If mode is set to **overwrite**, the file specified in **name** will be overwritten with the content stored in **data**. If the file does not exist, it will be created.
  overwrite will only be applied to the first append. All subsequent appends (e.g. in a loop) automatically get mode **append**.

Name	Type	Direction	Description

name	String	in	Fully qualified file name of the file you want to write, including the path. When using the Windows style with backward slashes "\" you have to be aware that you have escape this character. The escape character is also the "\". To avoid this, use forward slashes with Windows as well.
data	Blob	in	Content you want to write to the file.
length	Integer	in	Length (in Bytes) of the **data** you want to append, if you want to only write part of the Blob.

Action "status"

Name	Type	Direction	Description
name	String	in	Fully qualified file name of the file/directory you want to read, including the path. When using the Windows style with backward slashes "\" you have to be aware that you have escape this character. The escape character is also the "\". To avoid this, use forward slashes with Windows as well.
type	String	out	Type of accessed element (**File** or **Directory**).
size	Integer	out	Size of the file in Bytes. **size** is meaningless for directories, as it is *not* the combined size of all contained files and subfolders, but rather some constant defined by the operating system (e.g. 0 or 4096).
accessTime	DateTime	out	Timestamp when the file/directory was accessed for the last time.
creationTime	DateTime	out	Timestamp when the file/directory was created.
modificationTime	DateTime	out	Timestamp when the file/directory was modified for the last time.

Action "rename"

Name	Type	Direction	Description
name	String	in	Fully qualified file name of the file you want to rename, including the path.
newName	String	in	New name you want to rename the file to. We recommend to provide a fully qualified file name (including the path).

When using the Windows style with backward slashes "\" you have to be aware that you have escape this character. The escape character is also the "\".
To avoid this, use forward slashes with Windows as well.

Action "remove"

Name	Type	Direction	Description
name	String	in	Fully qualified file name of the file you want to remove, including the path. When using the Windows style with backward slashes "\" you have to be aware that you have escape this character. The escape character is also the "\". To avoid this, use forward slashes with Windows as well.

Action "Copy"

Name	Type	Direction	Description
source	String	in	Fully qualified file name of the file you want to copy, including the path.
target	String	in	Fully qualified file name of the target file, including the path.

> When using the Windows style with backward slashes "\" you have to be aware that you have escape this character. The escape character is also the "\".
> To avoid this, use forward slashes with Windows as well.

Action "close"

Name	Type	Direction	Description
name	String	in	Fully qualified file name of the file you want to close, including the path. When using the Windows style with backward slashes "\" you have to be aware that you have escape this character. The escape character is also the "\". To avoid this, use forward slashes with Windows as well.

Parameters Directory Handling

Action "readDirectory"

Name	Type	Direction	Description
name	String	in	Fully qualified path to the directory you want to read. When using the Windows style with backward slashes "\" you have to be aware that you have escape this character. The escape character is also the "\". To avoid this, use forward slashes with Windows as well.
entries	Array of String	out	**readDirectory** returns an array of Strings containing the directory names. The array elements will not contain the directory prefix (path).

Action "createDirectory"

Name	Type	Direction	Description
name	String	in	Fully qualified file name of the directory you want to create, including the path. When using the Windows style with backward slashes "\" you have to be aware that you have escape this character. The escape character is also the "\". To avoid this, use forward slashes with Windows as well.

Action "status"

Name	Type	Direction	Description

name	String	in	Fully qualified file name of the file/directory you want to read, including the path. When using the Windows style with backward slashes "\" you have to be aware that you have escape this character. The escape character is also the "\". To avoid this, use forward slashes with Windows as well.
type	String	out	Type of accessed element (**File** or **Directory**).
size	Integer	out	Size of the file in Bytes. **size** is meaningless for directories, as it is *not* the combined size of all contained files and subfolders, but rather some constant defined by the operating system (e.g. 0 or 4096).
accessTime	DateTime	out	Timestamp when the file/directory was accessed for the last time.
creationTime	DateTime	out	Timestamp when the file/directory was created.
modificationTime	DateTime	out	Timestamp when the file/directory was modified for the last time.

Action "remove"

You can only remove an empty directory with action **remove**. In case the directory contains files or subfolders, use **removeRecursively** (see below).

Name	Type	Direction	Description
name	String	in	Fully qualified file name of the directory you want to remove, including the path. When using the Windows style with backward slashes "\" you have to be aware that you have escape this character. The escape character is also the "\". To avoid this, use forward slashes with Windows as well.

Action "removeRecursively"

Name	Type	Direction	Description
name	String	in	Fully qualified file name of the top node of the directory tree you want to remove, including the path. When using the Windows style with backward slashes "\" you have to be aware that you have escape this character. The escape character is also the "\". To avoid this, use forward slashes with Windows as well.