blueprism®

Utility – File Management VBO
User Guide

Document Revision 1.1
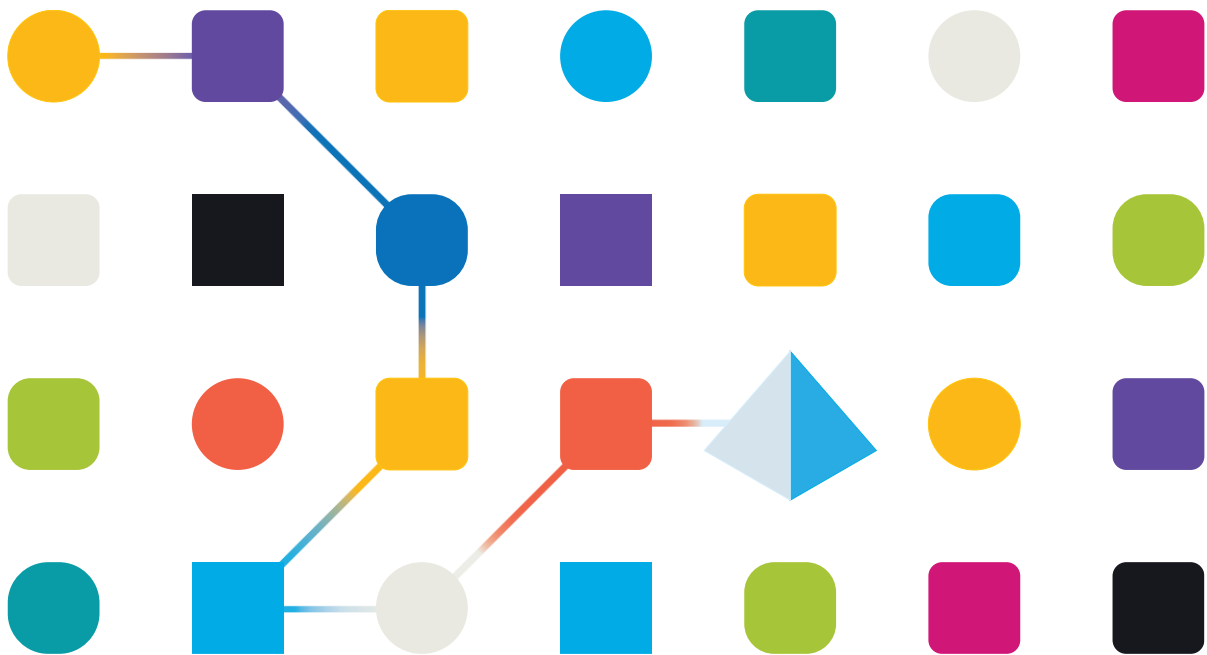
Trademarks and copyright

Contents

Introduction

This document discusses the **Utility – File Management** VBO and the actions it exposes. You can find the latest version of this VBO on the [Blue Prism Digital Exchange](#)

Prerequisites

None

Configuration

To use this asset:

- Download the asset file from the Digital Exchange to your local machine.
- Start your Blue Prism Interactive Client if it is not already running.
- Go to **File → Import → Process/Object**
- In the file dialog, navigate to the asset file you downloaded from the DX and select it.
- Complete the remaining prompts of the Import Wizard.

You are now ready to use the asset.

Using the Asset

The following actions are exposed by the VBO.

Append to Text File

Appends some text to the specified file.

Preconditions:

There must be sufficient disk space to accommodate the new data. The destination file need not exist already. If it does not exist, it will be created.

Endpoint:

The supplied text will have been appended.

Parameter	Direction	Data Type	Description
File Name	In	Text	The full path of the text file to append to
Text	In	Text	The text to append to the file
Success	Out	Flag	True if successful
Message	Out	Text	An error message if unsuccessful

Combine Path

Takes a file and folder and combines them into a full path.

Preconditions:

None.

Endpoint:

The path will have been combined.

Parameter	Direction	Data Type	Description
Folder	In	Text	The folder to combine
File Name	In	Text	The file to combine
Path	Out	Text	The combined full path

Configure

Configures this instance of the utility VBO. Calling it with no parameters will cause the configuration to be reset to default values.

Parameter	Direction	Data Type	Description
Raise exception on any failure	In	Flag	Flag indicating if failure should cause an exception to be raised rather than a return of a success output value

Copy Directory

Copies the contents of a directory to another directory.

Parameter	Direction	Data Type	Description
Source Directory	In	Text	The directory to be copied.
Destination Directory	In	Text	The location to which the directory contents should be copied.
Success	Out	Flag	Flag indicating success or failure of the action.
Message	Out	Text	If an error is encountered, this may hold additional information.

Copy File

Copies a file to the named destination.

Preconditions:

The file to be copied must exist. The full destination path must be specified. There must be sufficient free disk space to complete the operation.

Parameter	Direction	Data Type	Description
File Path	In	Text	The full path to the file to be copied
Destination Path	In	Text	The path of the directory to copy the file to
Success	Out	Flag	True if successful
Message	Out	Text	An error message if unsuccessful

Create Directory

Creates a directory under the given path.

Parameter	Direction	Data Type	Description
Directory Path	In	Text	The path in which to create the directory
Error If Already Exists	In	Flag	True to generate an error if the directory already exists
Success	Out	Flag	True if successful
Message	Out	Text	An error message if unsuccessful

Delete Directory

Deletes a directory and if Recursive is True will delete all subdirectories if present

Parameter	Direction	Data Type	Description
Directory Path	In	Text	The path of the directory to delete
Error If Not Exists	In	Flag	Set True to raise an error if the path does not exist
Recursive	In	Flag	Set True to delete all sub-directories of the path aswell
Success	Out	Flag	True if successful
Message	Out	Text	An error message if unsuccessful

Delete File

Deletes a single file from the file system.

Preconditions:

The file to be deleted must exist.

Endpoint:

The named file will have been deleted.

Parameter	Direction	Data Type	Description
File Name	In	Text	The full path to the file to delete
Success	Out	Flag	True if successful
Message	Out	Text	An error message if unsuccessful

Delete Files

Deletes all files in the specified folder, matching the supplied file extension pattern.

Preconditions:

The named folder must exist.

Endpoint:

All files within the named folder which match the supplied pattern will have been deleted.

Parameter	Direction	Data Type	Description
Folder	In	Text	The path from which to delete the files
Pattern	In	Text	The wildcard pattern of files to delete

Directory Exists

Determines whether the specified directory exsts. The full path must be specified.

Parameter	Direction	Data Type	Description
Directory Name	In	Text	The directory to check exists
Exists	Out	Flag	True if the directory exists

Download File

Downloads a file from a web URL and saves it in a folder on disk.

Preconditions:

None.

Parameter	Direction	Data Type	Description
Source URL	In	Text	The URL of the file that is to be downloaded
Destination Path	In	Text	The location of the folder where the file is to be saved

File Exists

Determines whether the specified file exists. The full path must be specified.

Preconditions:

None.

Endpoint:

The existence of the file of interest will have been determined.

Parameter	Direction	Data Type	Description
File Name	In	Text	The name of the file to check
Exists	Out	Flag	True if the file exists

Get CSV Text As Collection

Turns a comma separated values file into a collection

Parameter	Direction	Data Type	Description
CSV File Path	In	Text	The full path to the CSV file
First Line Is Header	In	Flag	Set True if you want to treat the first line as a header
CSV Values	Out	Collection	The collection populated with the values of the CSV file

Get CSV Text As Collection - Chunk

Returns a number of rows (aka a "chunk") of data from a CSV file as a Blue Prism Collection.

Parameter	Direction	Data Type	Description
CSV File Path	In	Text	The full path to the CSV file
First Line Is Header	In	Flag	Set True if you want to treat the first line as a header
Starting Row	In	Number	The row number where the reading of CSV data should begin.
Chunk Size	In	Number	The total number of rows to read in the chunk.
CSV Values	Out	Collection	The collection populated with the values of the CSV file
Next Row Number	Out	Number	The number of the row following the chunk returned in the collection.
Number of Rows in Chunk	Out	Number	The number of rows in the current chunk.

Get File Size

Gets the size of the specified file.

Preconditions:

The specified file must exist.

Endpoint:

The file size will have been determined.

Parameter	Direction	Data Type	Description
File Path	In	Text	Full path to the file to get the file size
Size	Out	Number	The size of the file in bytes

Get Files

Gets all files in the specified folder matching the supplied file type pattern string. Patterns - A comma-separated list of file patterns identifying the type(s) of files sought. Eg "*.jpg, *.png" retrieves all jpg and png files. The pattern "*.*" will return all files. It does not matter whether you leave a space between each pattern, or not. Where a comma is required in a file pattern it should be escaped out using the backslash character, e.g. a pattern of "*\,November.xlsx" will return all files that end with ",November.xlsx"

Preconditions:

None.

Endpoint:

All matching files will be returned.

Parameter	Direction	Data Type	Description
Folder	In	Text	The folder in which to look for files
Patterns CSV	In	Text	The comma separated list of wildcard patterns
Success	Out	Flag	True if successful
Message	Out	Text	A message if unsuccessful
Files	Out	Collection	The collection of files found

Get Folders

Gets a collection of all subfolders in the specified file system folder.

Preconditions:

The named folder must exist.

Endpoint:

All sub-folders will be returned, if any exist.

Parameter	Direction	Data Type	Description
Folder	In	Text	The folder in which to look for sub-folders
Success	Out	Flag	True if successful
Message	Out	Text	An error message if unsuccessful
Folders	Out	Collection	A collection of the sub-folders found within the folder

Get Special Folder Path

Gets the path of a special folder.

Parameter	Direction	Data Type	Description
Special Folder Name	In	Text	The name of the special folder. Supported values include ApplicationData, DesktopDirectory, MyDocuments, ProgramFiles. The values correspond to the .NET Environment.SpecialFolder enumeration.
Folder Path	Out	Text	The corresponding path of the requested special folder

Move File

Moves the specified file to a new location.

Preconditions:

The file to be moved must exist. There must be sufficient disk space to move the file.

Parameter	Direction	Data Type	Description
File Path	In	Text	The full path to the file to move
Destination Path	In	Text	The path to the folder to move the file to
Success	Out	Flag	True if successful
Message	Out	Text	An error message if unsuccessful

Read All Text from File

Reads the entire contents of a file as text.

Preconditions:

The file must exist.

Endpoint:

The full text of the file will be read.

Parameter	Direction	Data Type	Description
File Name	In	Text	The full path the file to read the text from
Success	Out	Flag	True if successful
Message	Out	Text	An error message if unsuccessful
Text	Out	Text	The text of the file

Read Lines From File

Reads a range of lines from a text file

Preconditions:

The text file exists, and the user has appropriate privileges to allow reading it

The Start Line argument is less than or equal to the End Line argument

Endpoint:

EOF will indicate if the end of the file was reached before End Line was reached

Parameter	Direction	Data Type	Description
File Path	In	Text	The file to read
Start Line	In	Number	The first line to read
End Line	In	Number	The last line to read
Success	Out	Flag	True is the read was successful
Message	Out	Text	The exception detail if the read was unsuccessful
Lines	Out	Collection	The lines read - the collection contains a single text column 'Line'
Line Count	Out	Number	The number of lines read
End of File	Out	Flag	True if the end of the file was reached before the 'End Line' was reached

Split Path

Splits a file path into its component parts - the filename and the folder path. The trailing backslash is omitted from the directory part.

Preconditions:

The specified path must correspond to a file which exists on disk.

Endpoint:

The path will have been separated.

Parameter	Direction	Data Type	Description
File Path	In	Text	The path to split
File Name	Out	Text	The filename
Folder	Out	Text	The folder

Sub Directory Exists

Checks if a directory contains sub directories

Parameter	Direction	Data Type	Description
Directory Path	In	Text	The path of the directory to check for sub directories
Exists	Out	Flag	True if sub directories exist
Success	Out	Flag	True if the check was successful
Message	Out	Text	An error message if unsuccessful

Write Binary Data

Writes binary data to a file. Any existing data in the file (if it exists) will be replaced.

Parameter	Direction	Data Type	Description
Binary Data	In	Binary	The data to write to the file
File Name	In	Text	The full path of the file to write to
Success	Out	Flag	True if successful
Message	Out	Text	An error message if unsuccessful

Read Binary Data

Read binary data from a file and stores it in a binary data item.

Parameter	Direction	Data Type	Description
File Name	In	Text	The full path of the file to read from.
Success	Out	Flag	True if successful
Message	Out	Text	An error message if unsuccessful
Binary Data	Out	Binary	The data read from the file

Write Text File

Writes text to a file. Any existing text in the file (if it exists) will be replaced.

Preconditions:

There must be sufficient disk space to accommodate the new data.

Endpoint:

The supplied text will have been written to the destination file.

Parameter	Direction	Data Type	Description
File Name	In	Text	The full path to the file to write the text to
Text	In	Text	The text to write to the file
Success	Out	Flag	True if successful
Message	Out	Text	An error message if unsuccessful

Write Large Text File

Writes a large text string >1MB to a file. Any existing text in the file (if it exists) will be replaced. This uses a different mechanism for saving the data and is meant for multiple MB of data.

Preconditions:

There must be sufficient disk space to accommodate the new data.

Endpoint:

The supplied text will have been written to the destination file.

Parameter	Direction	Data Type	Description
File Name	In	Text	The full path to the file to write the text to
Text	In	Text	The text to write to the file
Success	Out	Flag	True if successful
Message	Out	Text	An error message if unsuccessful

Support

This asset is provided free-of-charge by Blue Prism. Blue Prism does not provide formal support of this asset. Please direct any questions you have, related to this asset, to the Digital Exchange Community page:

https://community.blueprism.com/communities/community-home?CommunityKey=1e516cfe-4d1f-4de9-a9eb-58d15bf38c81