Rename job

Rename job is a default Switch app that provides various mechanisms for renaming the incoming job and/or the files inside a job folder.

Keywords

Keywords can be used with the search function above the Elements pane.

The keywords for the **Rename job** element are:

- remove
- add
- replace
- filename
- name
- extension
- prefix
- suffix
- regular expression
- regexp

Connections

Rename job allows only a single outgoing connection.

Properties

Property	Description
Element type	The flow element type: Rename job. This property is useful to identify renamed flow elements. It cannot be changed.
Name	The name of the flow element displayed in the canvas
Description	A description of the flow element displayed in the canvas. This description is also shown in the tooltip that appears when moving your cursor over the flow element
Affect job name for	Determines for which jobs the rename actions are applied to the name of the job (individual file or job folder):

Property	Description
	• All jobs
	• Individual files only (= files not being part of a job folder!)
	• Job folders only
	• No jobs
Remember original name	If set to yes, the original filename of the incoming job is stored in private data in the internal job ticket for the job so that it can be restored with a subsequent rename flow element; see *Restoring the original filename* in the *Switch Reference Guide*.
	Only the name of the job is remembered; not the name of any files inside a job folder
Private data key	The first portion of the private data key used to store the original filename; see *Restoring the original filename* in the *Switch Reference Guide*.
Files inside job folder	Determines what to do with files inside a job folder:
	• Leave alone
	• Rename files
	• Rename files and folders
	• Flatten hierarchy (discarding folders)
	• Flatten hierarchy and rename files
Nested folder levels	Specifies the number of nested subfolder levels to act on (zero means "leave alone"; use a large number to act on all levels)
Resolve collisions	Determines how to resolve situations where two files on the same level end up with the same name (after all other actions specified in this tool have been performed):
	• Add suffix after filename proper (= filename without extension)
	• Add suffix after complete filename
	• Modify name without changing length
Pattern	A pattern for the suffix or replacement string used to resolve collisions; "count" is replaced by a number with an appropriate number of digits (that is, a single digit unless there are many files colliding to the same name)
Action 1	Determines the first rename action taken:
	• None

Property	Description
	• Add prefix
	• Add suffix
	• Remove segment
	• Keep segment
	• Replace
	• Search and replace
	• Reduce character set
	• Remove extension
	• Add extension
	• Add leading zeros
	• Replace extension
Act on	Determines which portion of the filename is affected:
	• Filename proper (= filename without extension)
	• Complete filename (select this if the file has no extension)
Prefix	The prefix to add at the start of the filename
Suffix	The suffix to add at the end of the filename
Start index	The one-based start index of the filename segment to remove/keep
End index	The one-based end index of the filename segment to remove/keep; zero means until the end of the filename
Search for	The search string to be replaced, or a regular expression that matches the string to be replaced. **Note:** The values specified using the 'Single-line text with variables' and 'Script expression' editors are considered regular expressions. As a consequence, special characters like () [] {}) are considered part of the regular expression. If you want to search for strings that contain those brackets, we recommend using the "Inline value" or "Define a regular expression" editor. For example, to replace *(1)* you could enter the inline value *(1)* as search string, or define a regular expression: \(1\). Single-line text with variables and script expression will find *1* instead of *(1)* (unless you're using a workaround that escapes the special characters in variables e.g. through a script element).

Property	Description
Replace by	The replacement string, or a regular expression that can use the groups captured by the search regexp
Repeat	Determines the repeat behavior of the search/replace: • Once: only the first match is replaced • Consecutive: repeated as many times as needed to reach the end of the filename (however the replaced portion is NOT searched again) • Recursive: consecutive is repeated until the filename no longer changes (with a max. of 99 iterations)
Allowed character set	Determines the allowed character set: • Local8bit: the current ANSI code page on Windows (Latin1 on Western systems); UTF-8 (i.e. anything) on Mac OS X • Portable ASCII: printable 7-bit ASCII less those characters that aren't allowed in filenames on some systems
Replacement character	Determines the replacement for disallowed characters: • Underscore • Space • Remove
New extension	The new filename extension
Add leading zeros	Adds leading zeros to the numbers in the filename based on the subordinate properties: • *Position of the number*: Define a regular expression to indicate where in the filename proper leading zeros should be added. Only if the regular expression matches the filename, leading zeros will be added. • *Maximum number of digits*: Define the maximum number of digits for the number that needs to get leading zeros. For example, the filename 'File_1.pdf' with *Maximum number of digits* set to "4", will become 'File_0001.pdf' (3 leading zeros are added to get a 4-digit number). **Note:** • Only the filename proper (= filename without extension) is considered for this action. • If the maximum number of digits is lower than the digits in the number present in the filename, no leading zeros are added. For example, the filename '12345_abc_11.pdf' with position of number "\d+"

Property	Description
	and maximum number of digits is 4, the filename will become '12345_abc_0011.pdf'. So no leading zeros are added to the first number, two leading zeros are added to the second number.
Action 2 ... Action 5	The subsequent rename actions taken; these properties have the same subordinate properties as the first action property

Restoring the original filename

If the "Remember original name" property is set to yes, the tool stores the original name of the incoming job in the internal job ticket for the job using private data keys as follows:

Private data key	Stored value
<key>	The original filename including extension
<key>.Name	
<key>.NameProper	The original filename without extension
<key>.Extension	The original filename extension

For example, if the private data key property is left to its default value, the original filename without extension can be retrieved later on with the following variable:

[Job.PrivateData:Key="Original.NameProper"]

The "Remember original name" option works even if all rename actions are set to "none"; this can be useful to capture the job name before it gets changed by some other tool or application.