

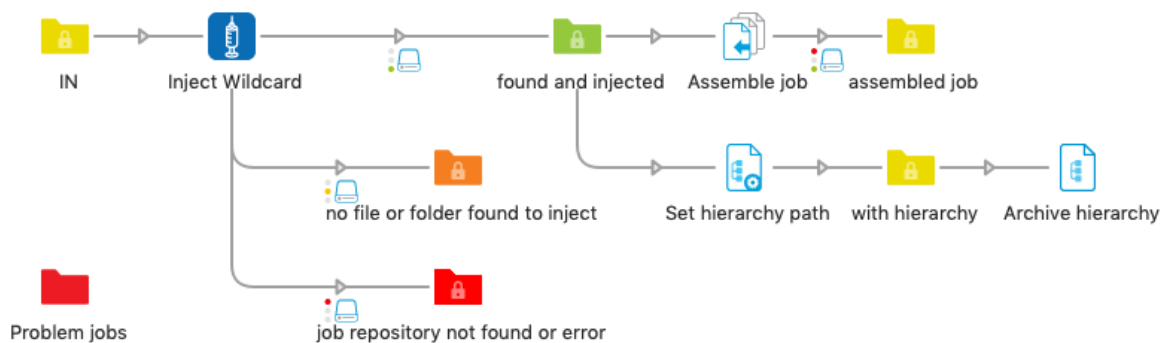
Inject Wildcard

Description

This app allows you to inject one or more job(s) into the flow when triggered by an incoming job. The injected job(s) is/are injected from a repository of jobs, with optional recursive search, optional parent filtering, and optional time filter.

The difference with the standard Inject Job element is that you can use the ? and * wildcards (1 or many characters) in the name of the file(s) or folder(s) to be injected, in addition to Switch variables. You can also use regular expressions.

This is the perfect solution when you don't know the complete name of the job(s) to be injected or want to inject more than one job.



Job assembling after injection

Inject Wildcard stores the following private data in the internal job ticket for each injected job file or folder in order to easily assemble the job further in the flow:

- <key>.JobID
- <key>.JobName
- <key>.NumFiles
- <key>.FilePath

The default value of the private data key is "Ungroup", so that if you use the *Assemble job* element with the *Ungrouped job* scheme, all injected file(s) and or folder(s) are assembled automatically, even with creation of the hierarchy if you increase the *Subfolder levels* property of the *Assemble job* element.

There is also another way to attach the hierarchy info to the injected file(s) and/or folder(s): use the *Set hierarchy path* element, and set the *hierarchy Path segment 1* property to the "Ungroup.FilePath" job private data: [Job.PrivateData:Key="Ungroup.FilePath"]

See the example flow illustrated above and available on the app description page.

Connections

Inject Wildcard requires at least one input and three output connections.

If the job(s) to be injected was / were found, the injected job(s) will be sent to the “success” output connection(s).

If no file or folder to be injected was found, the incoming job will be sent to the “warning” output connection(s), and a warning message will be logged.

If the job repository is not available or another error occurs, the incoming job will be sent to the “error” output connection(s), and an error message will be logged with more info.

Flow elements properties

- Job repository
 - folder containing the files and folders to be injected
- Subfolder levels: Which subfolder level(s) should be scanned?
 - All, Root level only or Custom
 - If Custom: From (top level), considering root level = 1
To (bottom level), must be greater than or equal to From level
- Inject
 - Files only / Folders only / Files and folders
- Injected jobs name filter
 - String to be found in the job(s) name to be injected.
Use the ? and * wildcards (1 or many characters) in combination with Switch variables, or use a regular expression (start ^ and end \$ can be useful)
- Name should (ignored with regular expressions)
 - begin with filter / end with filter / contain filter / match filter
- Case sensitive
 - Yes / No
- Hierarchy parents name filter - *default “None”, no parent filtering*
 - String to be found in the parent folders name of the jobs to be injected.
Use the ? and * wildcards (1 or many characters) in combination with Switch variables, or use a regular expression (start ^ and end \$ can be useful)
- Name should (ignored with regular expressions)
 - begin with filter / end with filter / contain filter / match filter
- Case sensitive
 - Yes / No
- Parent folder(s) level: Which parent level(s) should be scanned?
 - Any, One up or Custom
 - If Custom: From (closest level up), considering one up = 1
To (further level up), must be greater than or equal to From level
- Time filter
 - None / Newest n job(s) / Oldest n job(s) / Before date / After date / In last n days
 - Amount of jobs / Date / Amount of days
 - Compare using: Modification date / Creation date

- Delete after injection
 - Delete original files or folders from the repository after injection
- Incoming job
 - Send to output / Discard (delete)
- Private data key
 - The first portion of the private data key used to store the information for assembling the job later on

Example 1

Property	Value
Name	Inject Wildcard
Description	
Job repository	/Users/Shared/Repository
Subfolder search levels	Root level only
Inject	Files and folders
Injected jobs name filter	1234??
Name should (ignored with regex)	begin with filter
Case sensitive	No
Hierarchy parents name filter	None
Time filter	In last n days
Amount of days	7
Compare using	Modification date
Delete after injection	No
Incoming job	Discard
Private data key	Ungroup

Example 2

Property	Value
Name	Inject Wildcard
Description	
Job repository	/Users/Shared/Repository
Subfolder search levels	All
Inject	Files only
Injected jobs name filter	123456*pdf
Name should (ignored with regex)	match filter
Case sensitive	No
Hierarchy parents name filter	123
Name should (ignored with regex)	contain filter
Case sensitive	No
Parent folder(s) level	One up
Time filter	None
Delete after injection	No
Incoming job	Discard
Private data key	Ungroup

Example 1

This will inject all files and folders (including content), modified in the last 7 days, whose name begins with "1234" followed by two extra characters, only located on the root level of the job repository.

Example 2

This will inject all pdf files whose name begins with "123456", from root level and all subfolders, whose direct parent folder contains "123".

Example 3

If you want to inject all pdf or indd files whose name begins with "1234" and two extra digits, use this regular expression as injected jobs name filter: `^1234\d{2}.*(indd|pdf)`

Compatibility

Switch 2020 Spring and higher, Windows & Mac

What's new

v2

- Regular expressions
- Case sensitivity
- Job name should begin with string / end with string / contain string / match string
- Inject files and/or folders

v3

- Recursive search in the job repository
- Optional parent hierarchy filtering
- And, by the way, this is the first app in the Appstore written in Node.js, supported by the new scripting engine in Switch 2020!

v4

- Job repository check. Job sent to the error connection if the job repository is unavailable
- Bug fix with very deep hierarchy and special characters

v5

- Bug fix with very large folders and hierarchies

v6

- Time filter

It is still possible to install version 2 if you don't use Switch 2020 yet:

After app installation, go to My Enfocus account and under "My Appstore", "My apps", select the app and click on "Change version".

This url will lead you to the right page: <https://www.enfocus.com/en/appstore/licenses>