Duplicate

V4.0 Documentation

Overview

Duplicate allows you to copy an asset one or more times, giving you full control over how your documents or datasets are reproduced and organized. You can choose to "duplicate" the incoming job or an attached dataset for output, with the option to assemble the results into one or more folders using "chunks." Folders and documents, whether grouped or standalone, can be indexed according to any convention you need.

For example, a file named Job1234.pdf duplicated 100 times can be indexed using a unique separator and padding, resulting in filenames like:
Job1234_001.pdf, Job1234_002.pdf, Job1234_003.pdf...

You can also apply more advanced logic, such as splitting folders into sets of 25, creating four folders of 25 files each.

This makes Duplicate an ideal tool for situations where documents generated in a previous process need to be reproduced for a subsequent task. Duplicate is the successor to the legacy tool **Job Repeater**.

New in Version 4.0
Version 4.0 introduces powerful content-specific duplication within folders. You can now select which contents to duplicate based on:

- **First** or **Last** items

- **Index-based selection**

- **Regular expression patterns**, with support for regex flags

This allows for precise control over folder contents, enabling you to update and push jobs forward with exactly the files you need. Whether you need to duplicate a subset of documents or apply complex selection logic, **Duplicate** 4.0 gives you the flexibility to streamline your workflow like never before.

Compatibility
Switch 2024 Fall and higher. Windows and Mac OSX

Incoming Connections
At least one incoming connection required.

Outgoing Connections
At least one outgoing Traffic Light connection required.

Flow Element Properties

➢ <u>Action on</u>: Choose to action on the Job or a specific dataset attached to the job.
  ➢ <u>Dataset</u>: *(When Dataset selected)* The name of the dataset to action on.
  ➢ <u>Duplicate</u>: (When Job selected) If a folder is the incoming job, you may duplicate files within the folder by selecting "Contents", otherwise the job itself will be duplicated. If "Contents" is selected but the incoming job is a not a directory/folder, the filter properties will be ignored.
    ➢ <u>Filter</u>: Select one of the following options below:
      First - Duplicates the first file found in the job folder.
      Index - Duplicates the file found at the index in the job folder.
      Last - Duplicates the last file found in the job folder.
      Regular Expression - Duplicates one or more files found based on the regex pattern and flags specified from the job folder.
        ➢ <u>Index</u>: *(When Index selected)* Select a numerical index for the file to be selected for duplication. Any value enter that is not a number will automatically take the last file found in the job folder.
        ➢ <u>Pattern</u>: (When Regular Expression selected). A regex pattern string is made up of literal characters and special metacharacters.
        ➢ <u>Flags</u>: (When Regular Expression selected) Select from the following:
          g - Global search, finds all matches instead of stopping after the first
          i - Case-insensitive search, matches letters regardless of case
          m - Multiline mode, ^ and $ match the start and end of each line instead of the whole string
          s - Dotall mode, makes . match newline characters as well
          u - Unicode mode, enables full Unicode matching and proper handling of surrogate pairs
          y - Sticky mode, matches starting from the current position in the string without advancing past it
➢ <u>Duplicates</u>: The number of duplicates (or "copies") to be created.
➢ <u>Assemble in job folder</u>: Choose whether to group the copied file(s) into a singular job folder on output.
  ➢ (If Yes) <u>Folder name</u>: The outgoing filename for the resulting folder created.
  ➢ (If Yes) <u>Chunk</u>: Split a single folder into one or more folder(s) by a set quantity. E.g. A folder of 50 files, "chunked" by 10 will have five folders of 10.
    ➢ (If Yes) <u>Count</u>: The "chunk" count of how many files will be set per folder created.
    ➢ (If Yes) <u>Reset file index</u>: If "yes" the file index will be reset per folder. E.g. Two folders of 10 files would have two batches of files labeled _1 to _10. If "no" the above would have the first folder _1 to _10 and the second would have _11 to _20.
    ➢ (If Yes) <u>Index folder(s)</u>: By setting this value to "yes", folders will be indexed following a specified pattern - instead of outputting folders with the exact same name per chunk.

- ➢ (If Yes) <u>Separator</u>: Specify the separator character that will be used between the name and the padded value. This can be empty if no separator is preferred.
- ➢ (If Yes) <u>Pad length</u>: When padding, enter the number of characters to pad the outgoing folder name. E.g. A length of "2" will add a character in front of numbers ranging from 1-9. Anything beyond "9" is now two characters in length and the padding will be ignored.
- ➢ (If Yes) <u>Pad string</u>: When padding, specify a prefix string/character to use to suffix the outgoing folder name.
- ➢ <u>Outgoing filename(s)</u>: The outgoing filename for the resulting file(s) created.
- ➢ <u>Index file(s)</u>: By setting this value to "yes", files will be indexed following a specified pattern - instead of outputting files with the exact same name per duplicate.
  - ➢ (If Yes) <u>Separator</u>: Specify the separator character that will be used between the name and the padded value. This can be empty if no separator is preferred.
  - ➢ (If Yes) <u>Pad length</u>: When padding, enter the number of characters to pad the outgoing filename. E.g. A length of "2" will add a character in front of numbers ranging from 1-9. Anything beyond "9" is now two characters in length and the padding will be ignored.
  - ➢ (If Yes) <u>Pad string</u>: When padding, specify a prefix string/character to use to suffix the outgoing filename.
- ➢ <u>Job type</u>: Determine whether the resulting job(s) will be a child of the incoming or a new job. When "New Job" is selected, no data or properties from the incoming job are inherited.

About Significans Automation

Significans Automation is a software integrator specializing in delivering next-generation automation to the Printing and Packaging industry.

We offer programming and expertise in custom workflow development, deployment of communication and project management systems, color management, and end-to-end business integration. While upholding software neutrality, Significans Automation advises and tailors best in class software to optimally fit the environment.

The level of sophistication that is provided increases profitability, improved quality control, and enhanced production efficiency, enabling Artificial Intelligence and Robotics, while also facilitating new revenue opportunities in e-commerce. We are driven by the conviction that customized automation is the only path forward.