enfocus
SWITCH 13
Example Scripts

enfocus

Contents

1. Copyrights

2. About this document

This document contains links to a number of example Switch scripts, with an explanation of how they can be used in a Switch flow. You can download the scripts, and view them to learn more about scripting in Switch or use them as an example or a starting point for your own scripts. For each example script, there's also an example flow. You can use this flow in your own copy of Switch, to see how the script works.

For an overview of the Switch scripting API, refer to *the Switch Scripting documentation on the Enfocus website*.

For more information about the Script element and how to use it in Switch, refer to the *the Switch Reference Guide on the Enfocus website*.

2.1 Using an example flow

For each script in this document, you can download an example flow (under **Download link** at the bottom of the description page).

To use this flow in your copy of Switch

1. Make sure you have licensed the Switch Scripting Module, otherwise you won't be able to run the flow.

   For more information, refer to the Switch Reference Guide.

2. Download the flow (*.sflow) and save it on your computer.

3. Launch Switch.

4. In Switch, select **Flow** > **Import flow** .

5. Select the flow file you want to import.

6. Click **Open**.
   The imported flow is now visible in the Flows pane.

7. Have a look at the different elements of the flow.

   Note that the example script is called through a Script element.

8. Activate the flow and submit one or more files and check if the flow behaves as expected.

9. To make changes or to view the script, stop the flow and right-click the Script element in the canvas and select **Edit script package**.
   The script package is opened in SwitchScripter. You can now inspect (and/or change) the source code.

2.2 Viewing an example script

You can download the example scripts. They are available under **Download link** at the bottom of the description page. Note that the zip package contains both a script package and an example flow.

To view the example script

Do one of the following:

- Open the script (*.sscript) in SwitchScripter.
- Import the example flow (*.sflow) in Switch. Right-click the Script element in the canvas and select **Edit script package**.

> **Note:** Scripts that are used with a configurator (*Extract even and odd pages - Adobe Acrobat* on page 15 and *Add document info - Adobe Acrobat Professional configurator* on page 16) cannot be viewed in this way. Instead, you should locate them (see the configurator property "Script file") and open them in an editor.

3. AppleScripts

3.1 Sort jobs based on XMP metadata (AppleScript version)

Script language

AppleScript

Supported operating systems

Mac

Example flow and desciption

In this example flow, when a job comes in, the metadata is associated with the job using the XMP Pickup tool. Both the job file and the metadata are moved to the Script element, which contains the "Sort job based on XMP metadata" script. This script has defined an extra property ("XMP Sort expression") allowing Switch users to indicate the field to look for in the XMP data.

The script will read the value of this property and look for that specific field in the XMP dataset. Afterwards it will get a list of all valid connections and will try to match the XMP information with one of the connections and move the job along the matching connections. If no match was found, the job is moved along the "Error out" connection to the "Unknown" folder.

Download link

Download flow and script for Switch 12 and earlier.

Download flow and script for Switch 13.

4. Internal Scripting

4.1 Add the Job ID from an Ad Ticket to the job's hierarchy information

Script language

JavaScript

Supported operating systems

Mac, Windows

Example flow and description
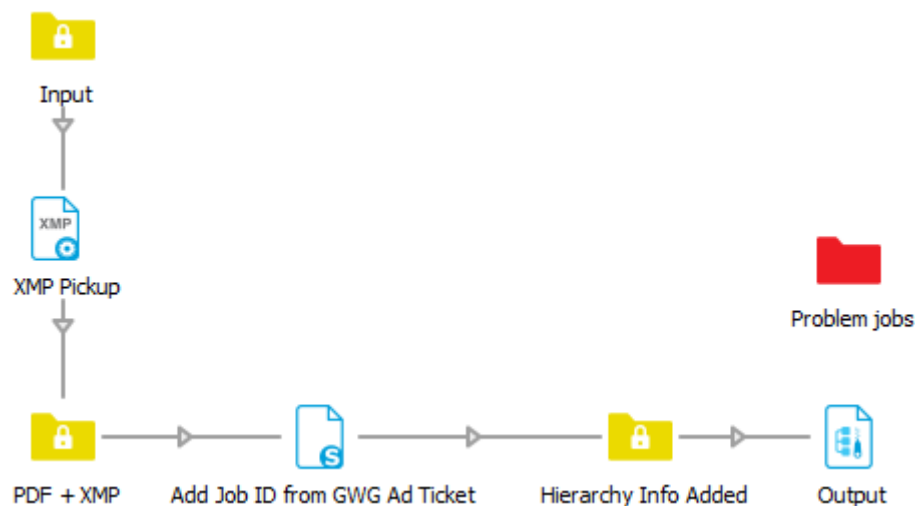
When a job is dropped in the "Input" folder, it is handed over to the XMP Pickup tool. This tool retrieves the XMP information that is stored inside the incoming job and adds it to the job ticket of that job.

Next, the Script element will run a script that accesses the XMP information in the Switch job ticket and returns the job ID from the Medibel+ or Ghent PDF Workgroup Ad Ticket. This job ID is added to the hierarchy information for that job. If no job ID was found, the level "unknown" is added to the hierarchy information.

The Archive hierarchy creates a folder structure based on the job ID placing every job in the correct subfolder. Jobs without a job ID are placed in the "Unknown" folder.

More information about the Ghent PDF Workgroup Ad Ticket is available at: *www.gwg.org*

Download link

Download flow and script for Switch 12 and earlier.

Download flow and script for Switch 13.

4.2 Label files with a Unique ID

Script language

JavaScript

Supported operating systems

Mac, Windows

Example flow and description
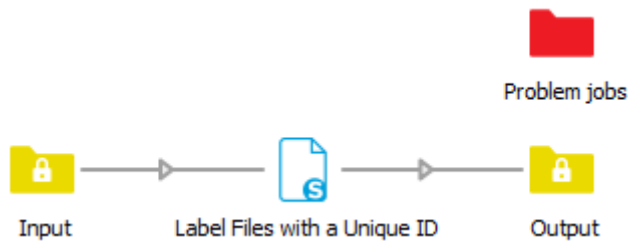
The script used in this flow shows how you can use a global counter. This results in a unique ID which is added to every job that passes through the flow. Even if you use this script element several times in the same flow or even in different flows, the counter will keep handing out unique IDs. The unique ID is also persistent, it remains unique even if Switch is stopped and restarted.

The first job processed receives the number "1" in front of its name (e.g. "1_Jobname"); this number is incremented for each subsequent job.

Download link

Download flow and script for Switch 12 and earlier.

Download flow and script for Switch 13.

4.3 Add part of the job name to a job's hierarchy information

Script language

JavaScript

Supported operating systems

Mac, Windows

Example flow and description

The script used in this flow allows you to determine which characters of the job name you want to remember and use to create an extra level in the hierarchy structure. In this example flow the property "First character to remember" is set to 1 and "Last character to remember" is set to 3.

This means that for each file, the Script element will read the first 3 characters of the job name and add these to the hierarchy information in the job ticket of that job. The three first characters of a job will be added as an extra level in the hierarchy information.

Once the hierarchy information has been adapted in the respective job ticket, the Archive hierarchy tool can be used to create the folder structure conform to what is stored in the job ticket. All jobs will be stored in the folder with the same name as the first 3 characters of the job name.

Download link

Download flow and script for Switch 12 and earlier.

Download flow and script for Switch 13.
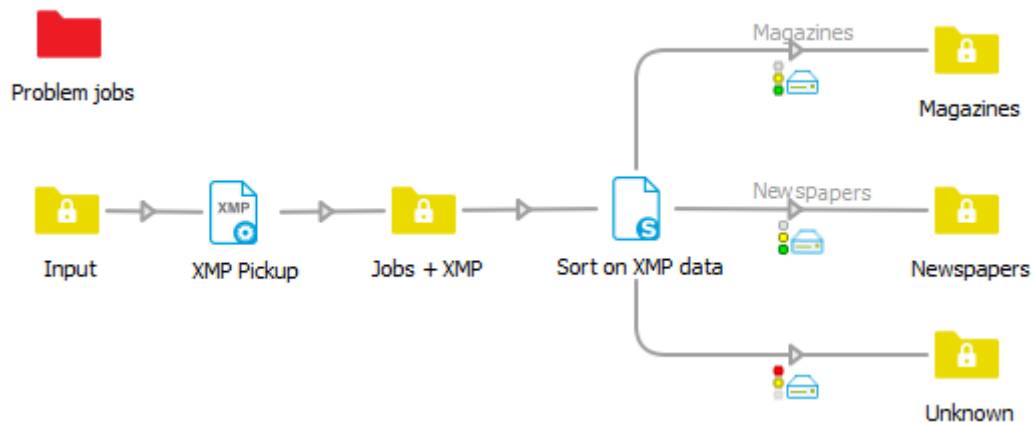
4.4 Sort jobs based on XMP metadata (JavaScript version)

Script language

JavaScript

Supported operating systems

Mac, Windows

Example flow and description

In this example flow, when a job comes in, the metadata is associated with the job using the XMP Pickup tool. Both the job file and the metadata are moved to the Script element, which contains the "Sort job based on XMP metadata" script. This script has defined an extra property ("XMP Sort expression") allowing Switch users to indicate the field to look for in the XMP data.

The script will read the value of this property and look for that specific field in the XMP dataset. Afterwards it will get a list of all valid connections and will try to match the XMP information with one of the connections and move the job along the matching connections. If no match was found, the job is moved along the "Error out" connection to the "Unknown" folder.

Download link

Download flow and script for Switch 12 and earlier.

Download flow and script for Switch 13.

4.5 Archive by date

Script language

JavaScript

Supported operating systems

Mac, Windows

Example flow and description
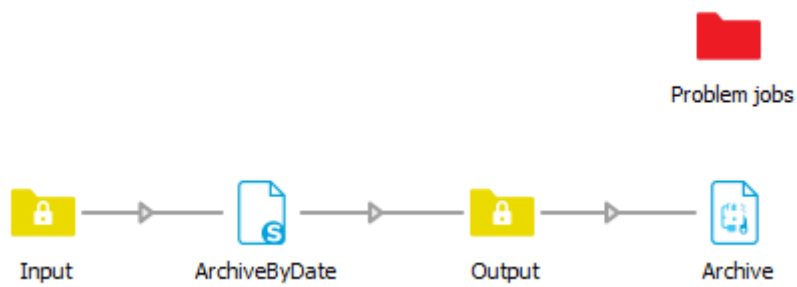
Each time a new job arrives, the script retrieves the current date from the system and stores it in a string. Depending on what you selected in the Script element property "Archive by" (Year/Month/Day), only the year, month or the complete date will be remembered.

This information is then added to the hierarchy information which enables you to use an Archive hierarchy tool to create the folder structure of your choice, i.e. grouping files in subfolders per year, month or day.

Download link

Download flow and script for Switch 12 and earlier.

Download flow and script for Switch 13.

4.6 Filter on PitStop XML Report
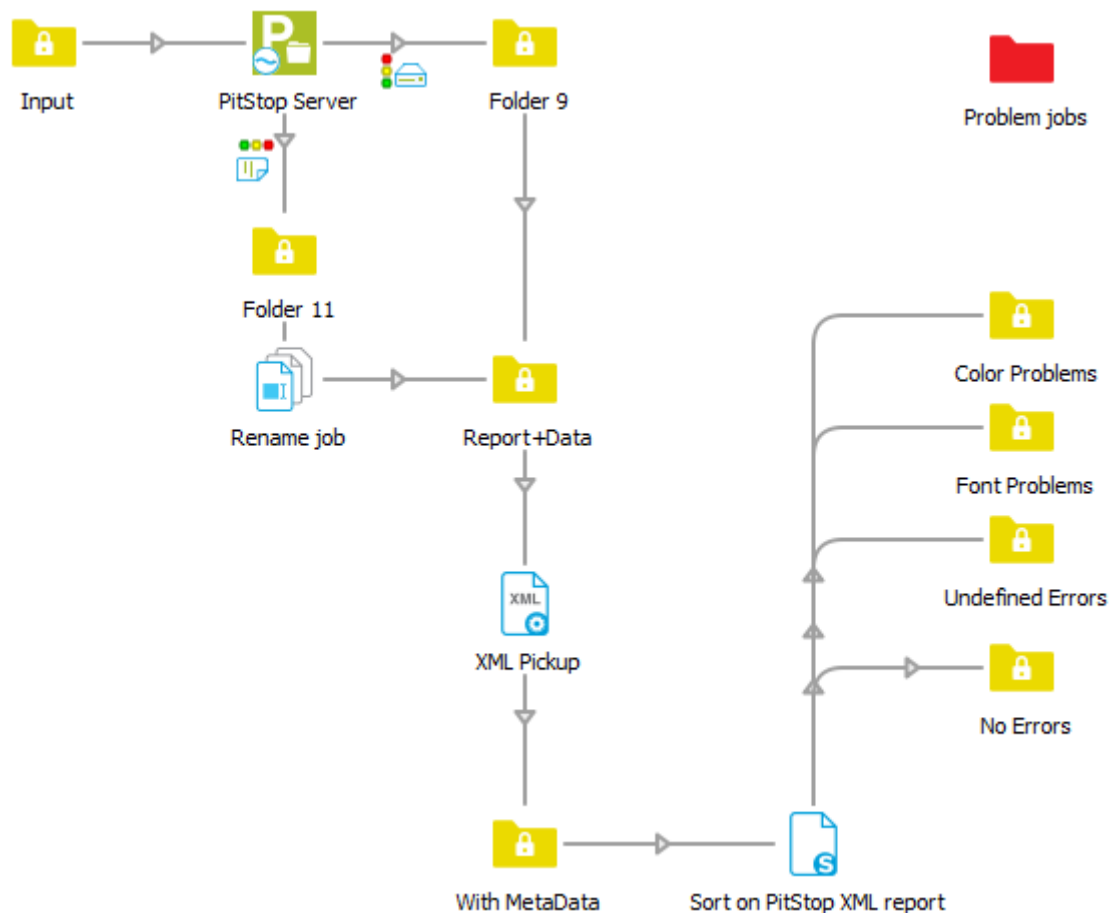
Script language

JavaScript

Supported operating systems

Mac, Windows

Example flow and description

Before you can use this script in a flow, you need to make sure that the preflight information is added to the file's job ticket. The following 2 steps are needed in order to accomplish this:

- The "Report type" property of the PitStop Server configurator in Switch needs to be set to XML.
- The XML Pickup tool should be used to add the preflight information to the job ticket for each preflighted PDF. You can do this by making sure that both the original PDF and the Preflight Report (XML) are dropped into the same folder and have the same name. (See example flow)

When you select the "Filter on PitStop XML Report" script in a flow, you will see the following additional properties:

- **Admit errors**: defines if the connection allows PDF files with preflight errors to go through.
- **Admit warnings**: defines if the connection allows PDF files with preflight warnings to go through.
- **Admit fixes**: defines if the connection allows PDF files to go through if certain problems have been fixed during the preflight process.
- **Use regexp**: defines if a regular expression should be used for filtering the preflight messages. If not, any error, warning or fix is allowed.
- **Use message regexp**: allows you to define a regular expression that is used to match one of the messages in the XML Preflight Report. If no messages match the regular expression, the PDF file will not be moved along the connection.

Each time a new file arrives, the script will try to match any of the preflight messages with any of the outgoing connections using these properties. If a connection is found that matches, the

PDF is moved along that specific connection. Once a matching connection is found, the script will continue with the next file.

Download link

Download flow and script for Switch 12 and earlier.

Download flow and script for Switch 13.
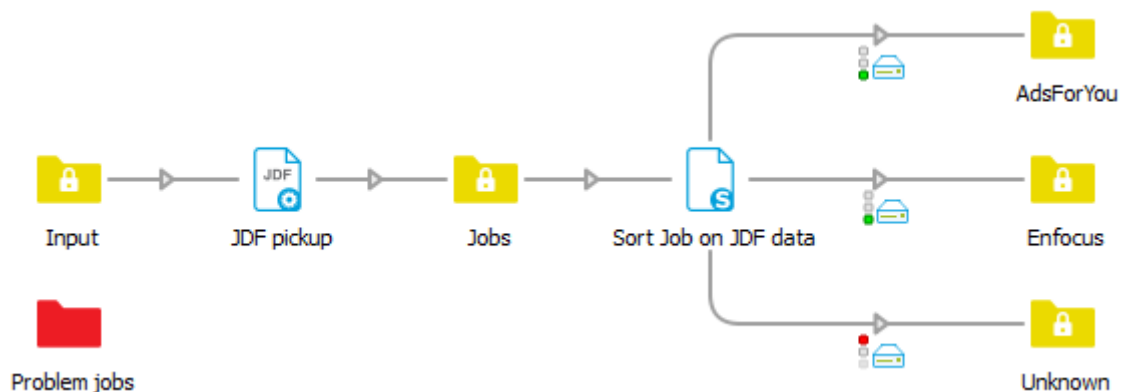
4.7 Sort jobs based on JDF metadata

Script language

JavaScript

Supported operating systems

Mac, Windows

Example flow and description

In this example flow, the input should be:

• A ZIP file containing a folder with an asset and associated JDF file
• A mime file containing a folder with an asset and associated JDF file
• A folder with an asset and associated JDF file

The tool JDF pickup adds the JDF file to the job as metadata. Next, the job is moved to the Script element containing the "Sort jobs based on JDF data" script. Note that, before running the flow, you must indicate which field you want to look up in the JDF dataset by using the "XPath to sort field" property.

The script will read the value of this property and look for that specific field in the JDF dataset. Afterwards it will get a list of all valid connections and it will try to match the JDF information with one of the connections and move the file along the matching connections. If no match was found, the job is moved along the "Error out" connection (in this example, to the "Unknown" folder).

Download link

14

Download flow and script for Switch 12 and earlier.

Download flow and script for Switch 13.

5. JavaScripts for Adobe Creative Suite applications
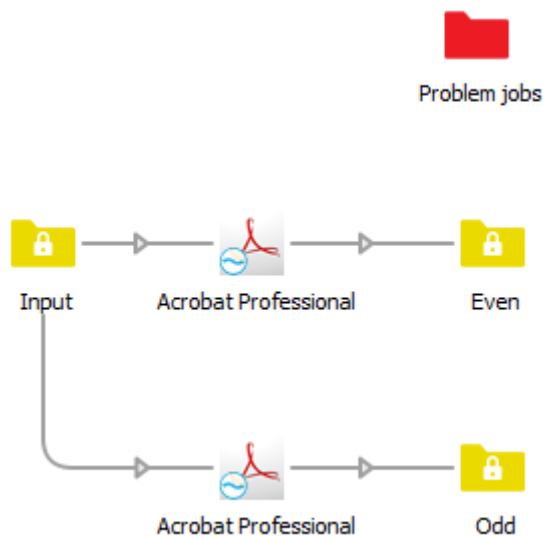
5.1 Extract even and odd pages - Adobe Acrobat

Script language

JavaScript

Supported operating systems

Mac, Windows

Example flow and description

The Acrobat configurator has a **Save as** property, which can be used to call a script. In this example, "Extract even_odd.jsx" takes care of splitting the PDF in two files (one with the odd and one with the even pages).

Download link

Download flow and script for Switch 12 and earlier.

Download flow and script for Switch 13.

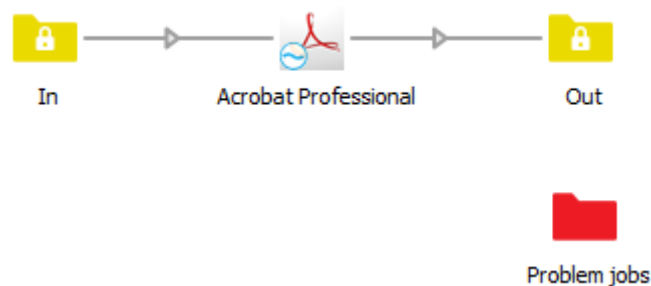# 5.2 Add document info - Adobe Acrobat Professional configurator

Script language

JavaScript

Supported operating systems

Mac, Windows

Example flow and description

The Acrobat configurator has a **Command** property, which can be used to call a script. In this example, "Acrobat_AddDocumentInfo.js" takes care of adding information to the following PDF info fields:

- Title
- Author
- Subject
- Keywords
- Creator

The new content for this fields has to be set in the Flow Element properties (Argument 1, 2, 3, ...).

Download link

Download flow and script for Switch 12 and earlier.

Download flow and script for Switch 13.

6. Visual Basic scripts
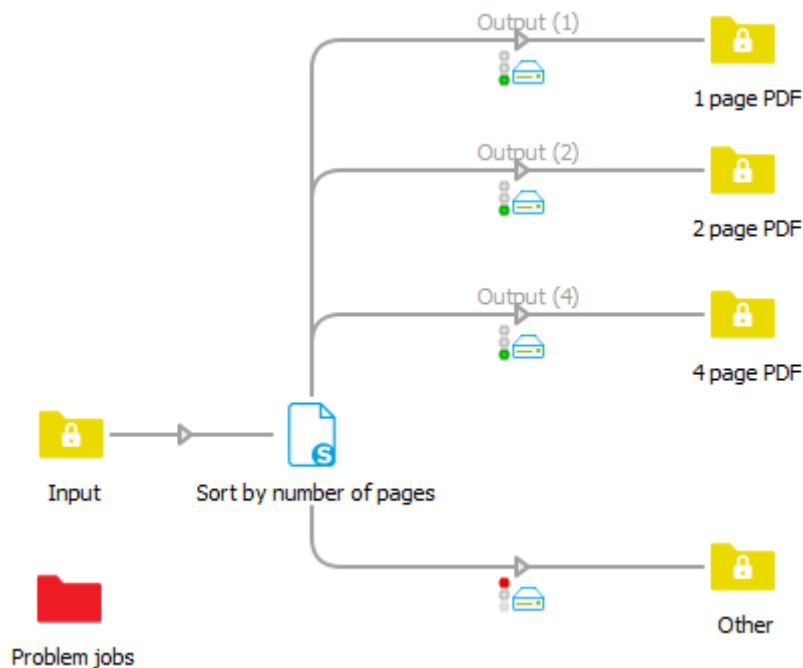
6.1 Sort PDF by number of pages using Adobe Acrobat

Script language

VB Script

Supported operating systems

Windows

Example flow and description

This script first launches Adobe Acrobat and retrieves the number of pages from the incoming job. It then checks the list of output connections and tries to find a match. A matching connection is called "Output (x)" where x is the number of pages in the PDF file. If a match is found, the PDF file is moved along that connection, if not found it is moved along the "Error out" connection ("Other").

Note: This script is meant as an example of using VBScript, not as a way to sort PDFs in Switch. If you want to sort PDFs by number of pages, it's much easier to set conditions on the connections, using the Stats.NumberOfPages variable. For more information about 'defining conditions with variables', refer to the Switch Reference Guide.

Download link

Download flow and script for Switch 12 and earlier.
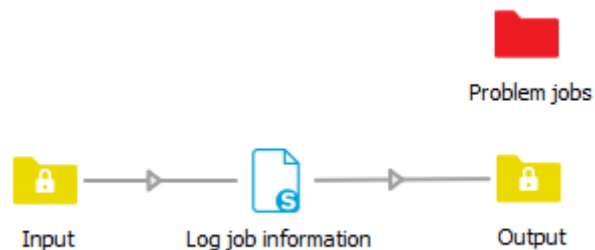
Download flow and script for Switch 13.

6.2 Log job info in Excel

Script language

VB Script

Supported operating systems

Windows

Example flow and description

The script (contained in a Script element) defines an extra Flow Element property ("Spreadsheet Path") to define the path to the Excel spreadsheet that needs to be used for logging the requested information. Note that you need to create this Excel file upfront and that it must contain a worksheet called "Log". Alternatively, you can use the example file (Job log.xls(x)) provided with the script (see "Download link" below). Do not forget to set the appropriate path.

The following information is retrieved for each job:

- Date/Time
- Job Name
- Mac Type
- Mac Creator Job Type
- Files in Job
- Size (bytes)
- Job State
- User Name

A new row is added to the Excel spreadsheet for each job that is processed.

Download link

Download flow and script for Switch 12 and earlier.

Download flow and script for Switch 13.