

Autodocumenter Installation and Usage

Author: Lev Solomovich

CONTENTS

BACKGROUND	3
INSTALLATION	3
REQUIREMENTS	3
PROCESS	3
USAGE	4
RESULTS	5
KNOWN LIMITATIONS.....	6
SUPPORT CONTACT	6

BACKGROUND

Alteryx users continually need some form of documentation generator from workflows. Users need this form of documentation generator for many reasons, ranging from convenience to compliance. Keyrus has set out to make the process easier by automatically generating documentation based on the content of a workflow.

When we presented our original version of the Autodocumenter tool for Alteryx, feedback from the community was positive but there were shortcomings. The key issues with the original documentation utility were:

- It did not support workflows with containers
- It did not provide an order of tools (the document would not be helpful in rebuilding a workflow)
- It did not provide enough details about the configurations of various tools.

The new workflow addresses these shortcomings. Additionally, the newest version generates a large image of the full workflow as part of the documentation. If required, a larger version of this image is saved to the Autodocumenter's outputs directory).

INSTALLATION

REQUIREMENTS

Alteryx 2018.3 or later is required as the Python tool is required for the Autodocumenter

There are only a few installation steps, but special care may be required if you are using an ADMIN installation of Alteryx versus a USER installation. ADMIN installation will require you to run the application with administrator privileges to allow the Python tool to install required libraries. The final part of the installation will be performed by a Python script embedded in one of the Autodocumenter workflow steps. If your organization blocks access to pip repository, you will have to find a way to manually install the required libraries "Pillow" and "xmldict" into the Alteryx Python environment.

The Autodocumenter was developed with the following versions, but any subsequent version should work:

- Pillow-6.0.0-cp36-cp36m-win_amd64.whl
- xmldict-0.12.0-py2.py3-none-any.whl

PROCESS

STEP 1. Download the latest version from Keyrus public Gitlab [here](#). As of writing, the latest version is "Imager_2019_08_07.zip"

STEP 2. Extract contents of zip file into any directory.

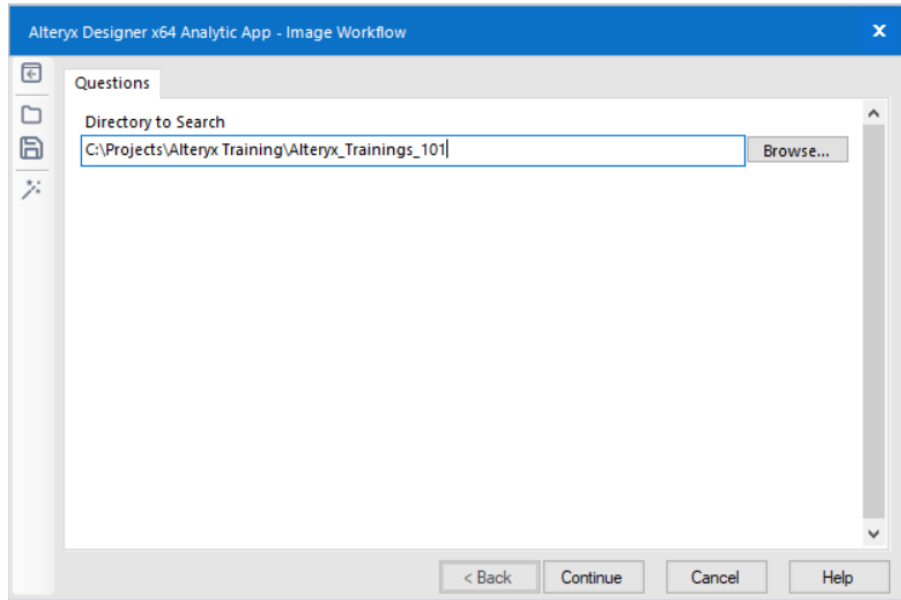
STEP 3. Disable Python tool timeouts for your Alteryx installation. Please see the post by Patrick Digan [here](#). Change the code on line 68 in execute.py by setting the timeout value to -1. The exact location can vary depending on installation type and Alteryx version, but this file is usually here: *C:\Program Files\Alteryx\2019\bin\Miniconda3\PythonTool_venv\Lib\site-packages\nbconvert\preprocessors\execute.py*

STEP 4. If you are using a USER installation of Alteryx, run Image” Workflow.yxwz” and follow Usage steps. The correct libraries will be installed the first time the Autodocumenter runs. It may take a little bit longer the first time.

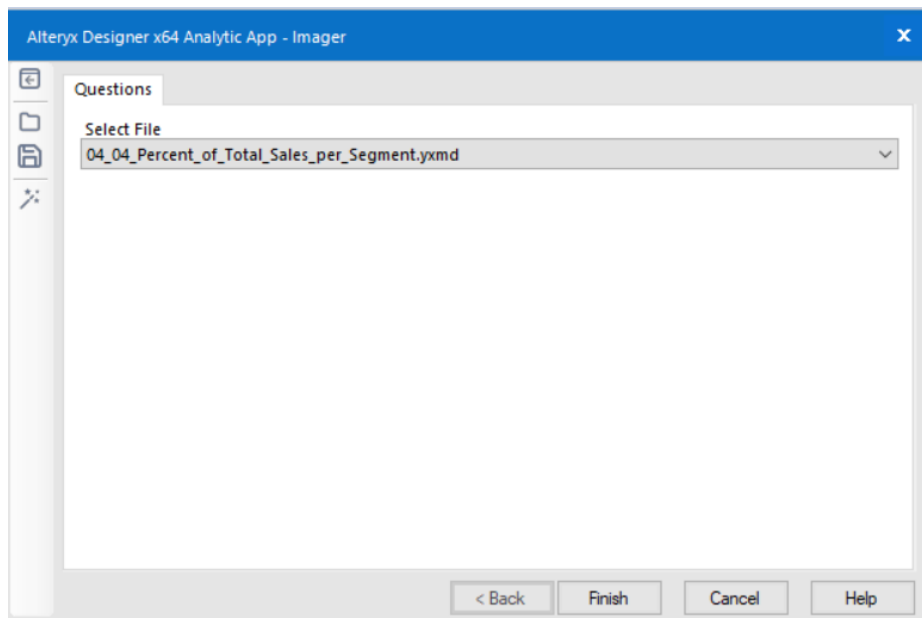
USAGE

After installation, we are ready to run the Autodocumenter.

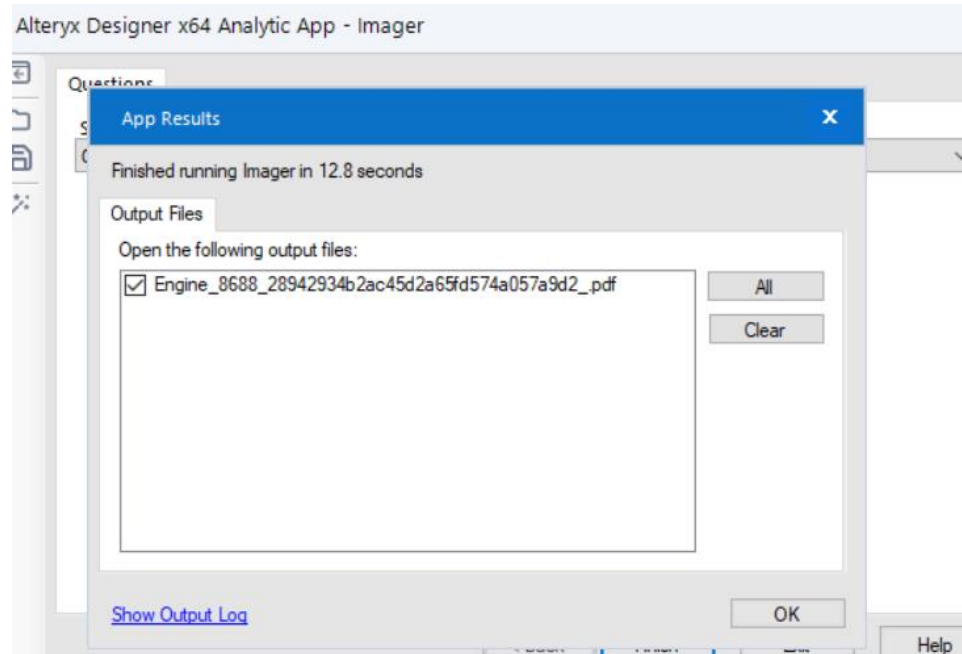
STEP 1. Run Image “Workflow.yxwz.” A dialogue box will appear asking you to select a folder to browse for workflows. Locate and select your workflows of interest. When pressing “Continue” at the bottom right, Alteryx will recursively search the directory all Alteryx workflows.



STEP 2. After pressing “Continue”, you should see a dropdown list with all workflows that were found. Select a workflow and press “Finish” to produce a document for this workflow.



STEP 3. Once finished, the Autodocumenter will run in the background and produce images and data required for the documentation. Alteryx will then produce a report based on this output using built-in reporting tools. The result should be a PDF file as shown below.



Note: This file is temporary and is not automatically saved anywhere (hence the funny looking name). You will need to open this file and save it if you wish to keep the result.

STEP 4. Make sure the PDF output file is selected, and press “OK.”

RESULTS

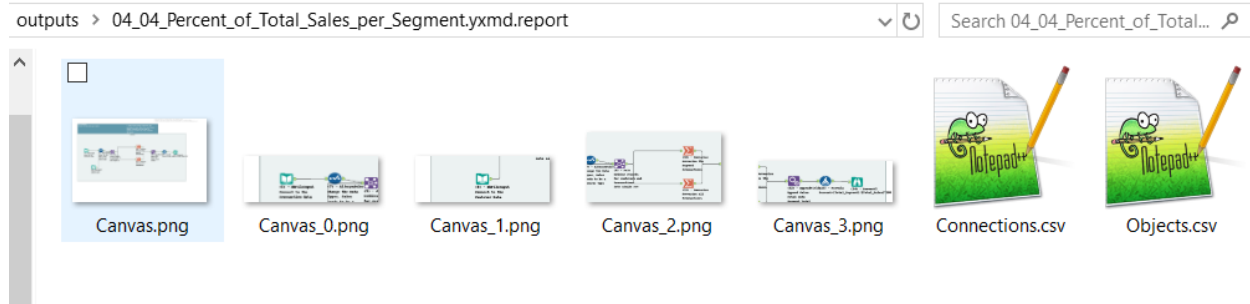
The resulting .pdf file will contain:

- Title Page
- Workflow Image
- Table of Contents
- Tool Summary (counts by type)
- Input/Output Summary (including directory paths and connections, passwords are not visible)
- Potential Problems (with suggestions)
- Section-by-Section Summary
 - Note: A “section” is discovered somewhat arbitrarily. Any piece of a workflow that starts with an input, join, or a few other tool types as the start of is called a “section”. If there is enough interest, we may build this feature out to be more nuanced and configurable
- Detailed Tool Configuration

The outputs directory will also contain a .report type file. The outputs directory should appear in the same directory where you unzipped the Autodocumenter tool. For example, if you extracted the zip to *C:\MyStuff\Autodoc*, the outputs directory will be created as *C:\MyStuff\Autodoc\outputs*. As shown by the example below, the .report file is a directory that can be opened:

04_04_Percent_of_Total_Sales_per_Segment.yxmd.report

The directory contains a full-sized PNG image of the workflow (this image can be very large), as well as section images and CSV files for Connections and Objects:



Between the PDF output, the CSV files, and PNG images, you should have enough material to put together your own documentation.

KNOWN LIMITATIONS

- If you have ADMIN installation of Alteryx, you must run as admin to allow Python tool to install required libraries on first run.
- If you have a workflow where an interface tool connects to a container (to activate/deactivate), the Autodocumenter may not work.

SUPPORT CONTACT

If you would like for this product to be supported or any modifications, paid consulting is available from Keyrus. Please reach out to us at sales@keyrus.us