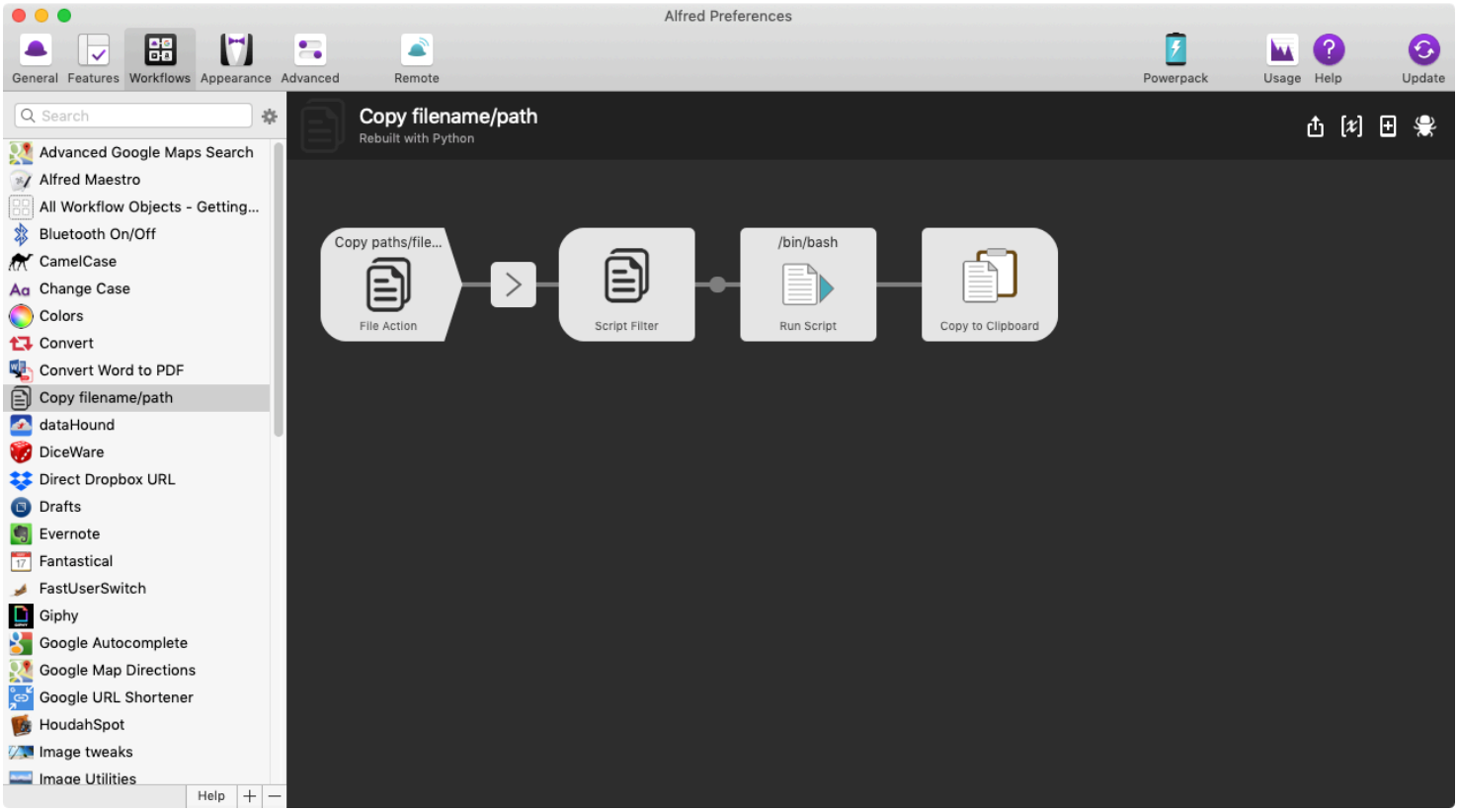
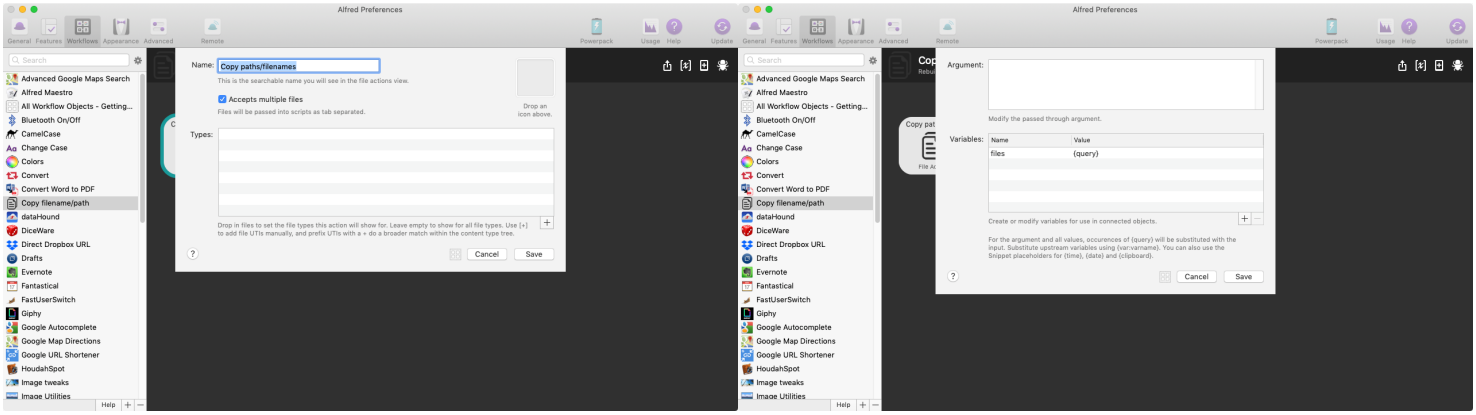
Copy paths or filenames (Python) - ALFRED Workflow

This workflow was a re-work of an earlier version that used a Python-based script filter and then acted on the selection via AppleScript. Here, the entire workflow operation is Python-based.



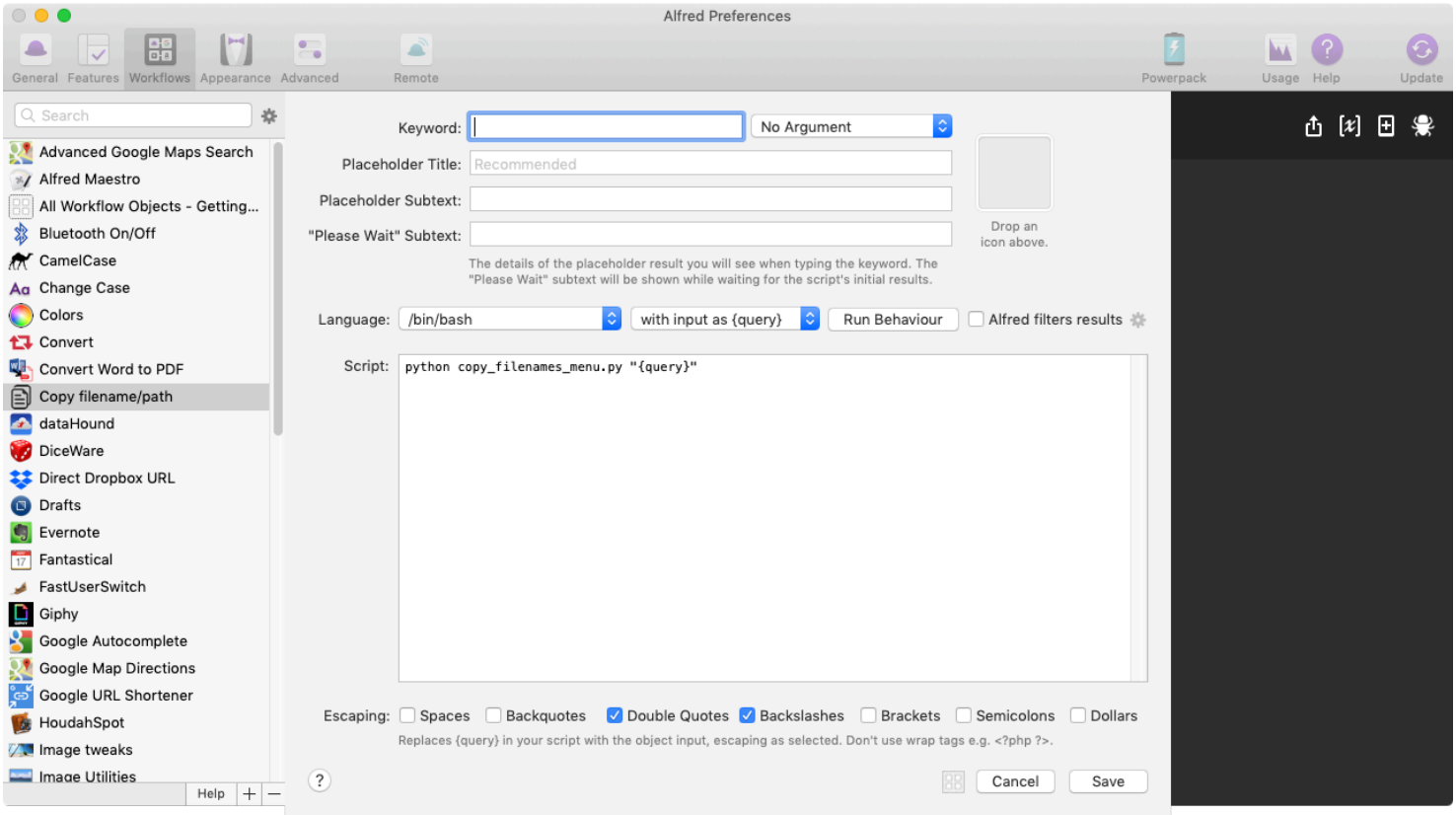
File Action

The workflow is activated by a file action and the file paths are pushed into a variable name **files**.



Script filter

A Python-based script filter provides a menu of options along with examples using the first file in the group.



The script filter runs `copy_filenames_menu.py` that in turn calls functions contained in `item_helpers.py` and `example_helpers.py`.

copy_filenames_menu

```
#!/usr/bin/env python
#
#   Script filter for copying paths/filenames of selected FINDER items
#   to the clipboard
#
from item_helpers import *
from example_helpers import *

query = sys.argv[1]

items = [pattern_item('Only filename(s)',filename_example(), 'fn','sign.png'),
        pattern_item('Abbreviated path(s)', abbr_path_example(), 'abbr', 'tilde.png')
,
        pattern_item('Full path(s)', path_example(), 'full', 'path.png')]

show(items)
```

item_helpers

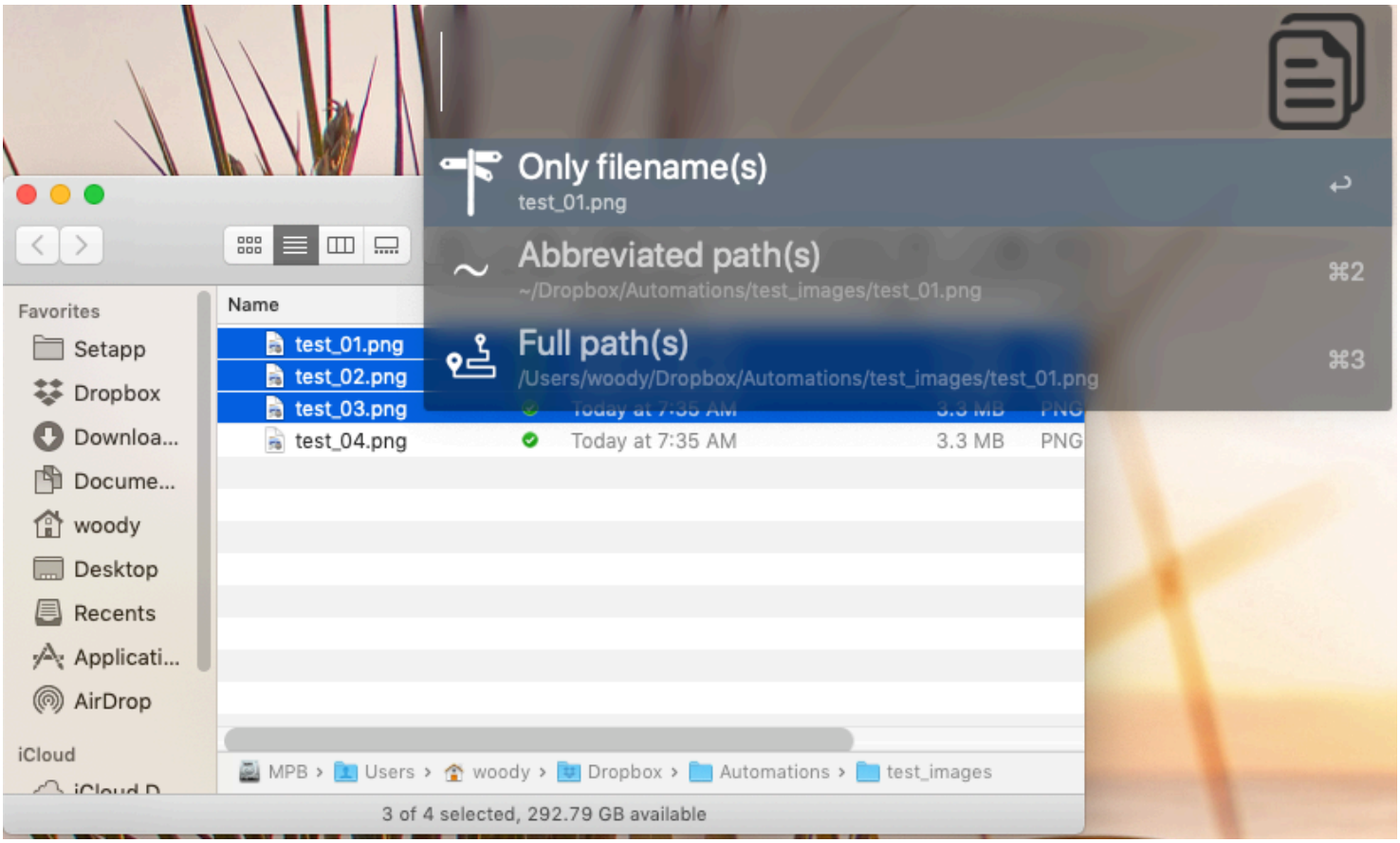
```
#
# Builds dictionary items for COPY_FILENAMES script filter
#
import sys
import json
import os

def pattern_item(title, pattern, arg, icon_name):
    item = {'valid': True,
            'title': title,
            'subtitle': pattern,
            'arg': arg,
            'autocomplete': arg,
            "icon": {
                    "path": icon_name
                    }
            }
    return item

def show(items):
    output = {'items': items}
    output_json = json.dumps(output)
    sys.stdout.write(output_json)
```

example_helpers

```
#!/usr/bin/python
#
#   Creates examples of path/filename handling for COPY_FILENAMES script filter code
#
import json
import os
import sys
import re

def filename_example():
    files = os.environ['files'].split('\t')
    str = files[0].split('/')
    return str[-1]

def path_example():
    files = os.environ['files'].split('\t')
    return files[0]

def abbr_path_example():
    files = os.environ['files'].split('\t')
    #   re.sub(pattern, repl, string, count=0, flags=0)
    str = re.sub('\/Users\/\w*', '~', files[0])
    return str
```

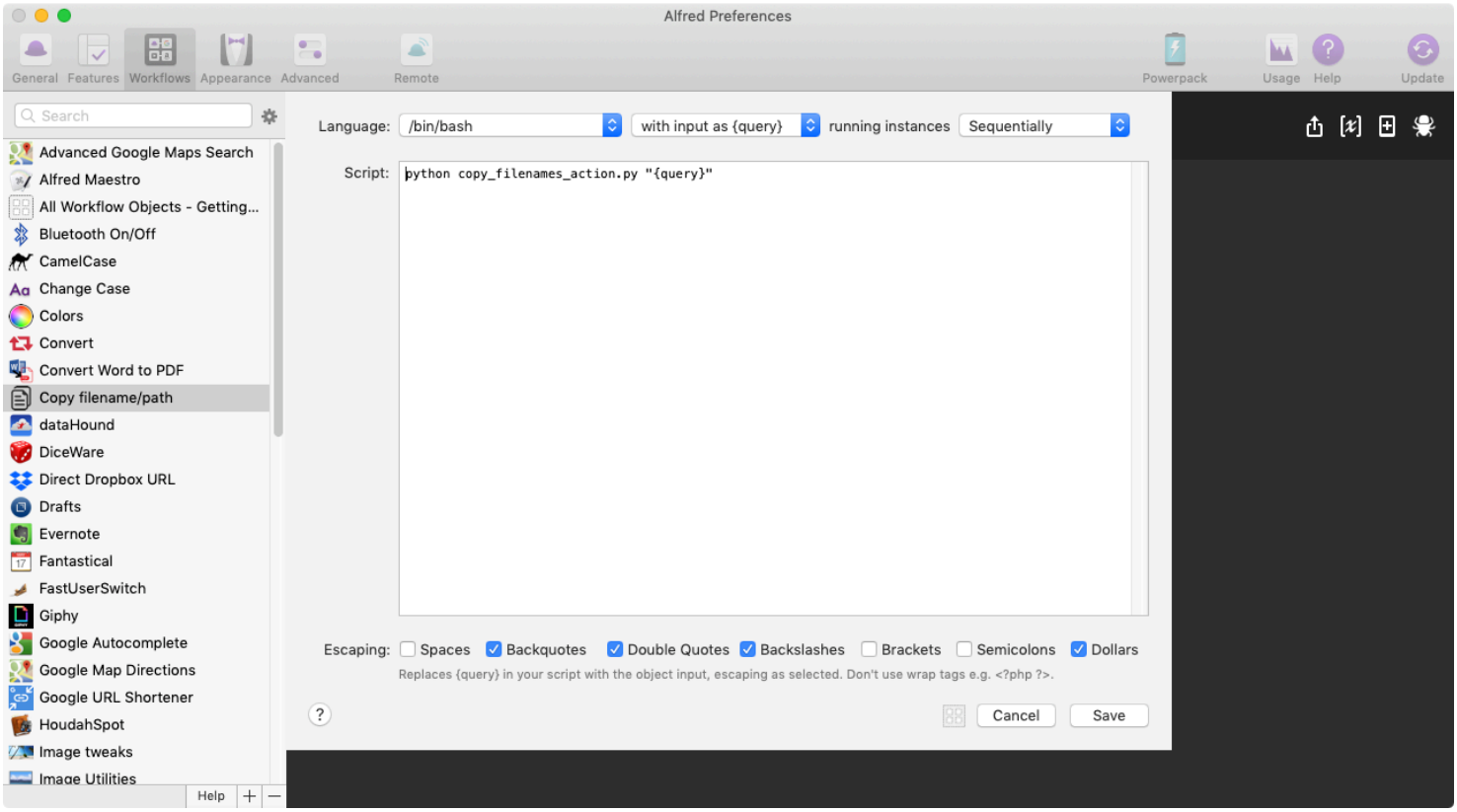
Menu

Selecting files and invoking the file action, yields the option menu depicted below. Choosing one of the options passes that choice to the subsequent **Run Script** action.



Run Script

A Python script, `copy_filenames_action.py`, processes the list of file paths per the selected action and pushed the result to the clipboard



copy_filenames_action

```
#!/usr/bin/python
#
#   Processes the selected path/filename per the selected treatment
#
import json
import os
import sys
import re

action = sys.argv[1]

files = os.environ['files'].split('\t')

if not action == 'full':
    for ii in range(len(files)):
        if action == 'abbr':
            files[ii] = re.sub('\/Users\/\w*', '~', files[ii])
        elif action == 'fn':
            str = files[ii].split('/')
            files[ii] = str[-1]

sys.stdout.write('\n'.join(files))
```