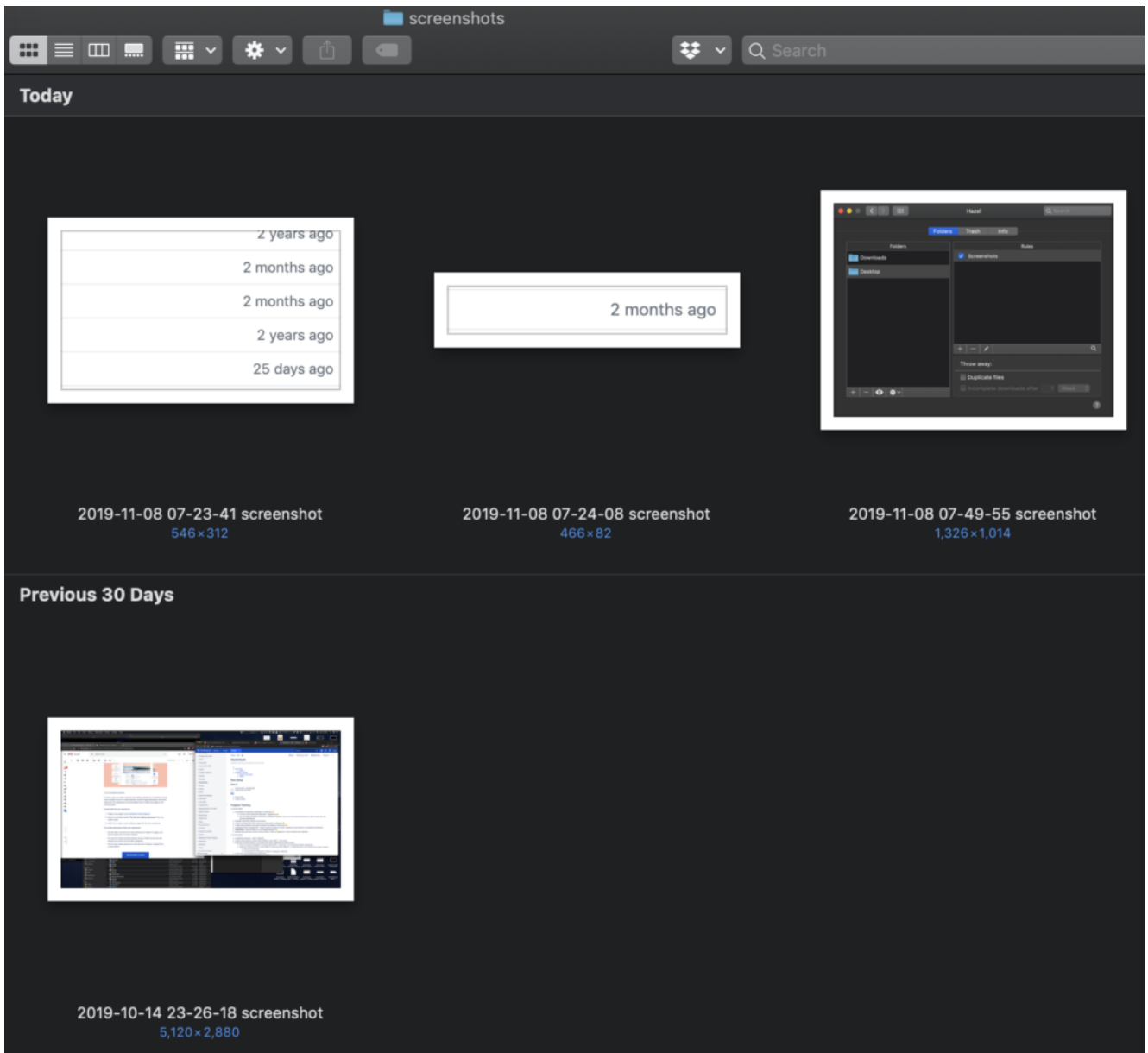


# Automating macOS Screenshots with Hazel

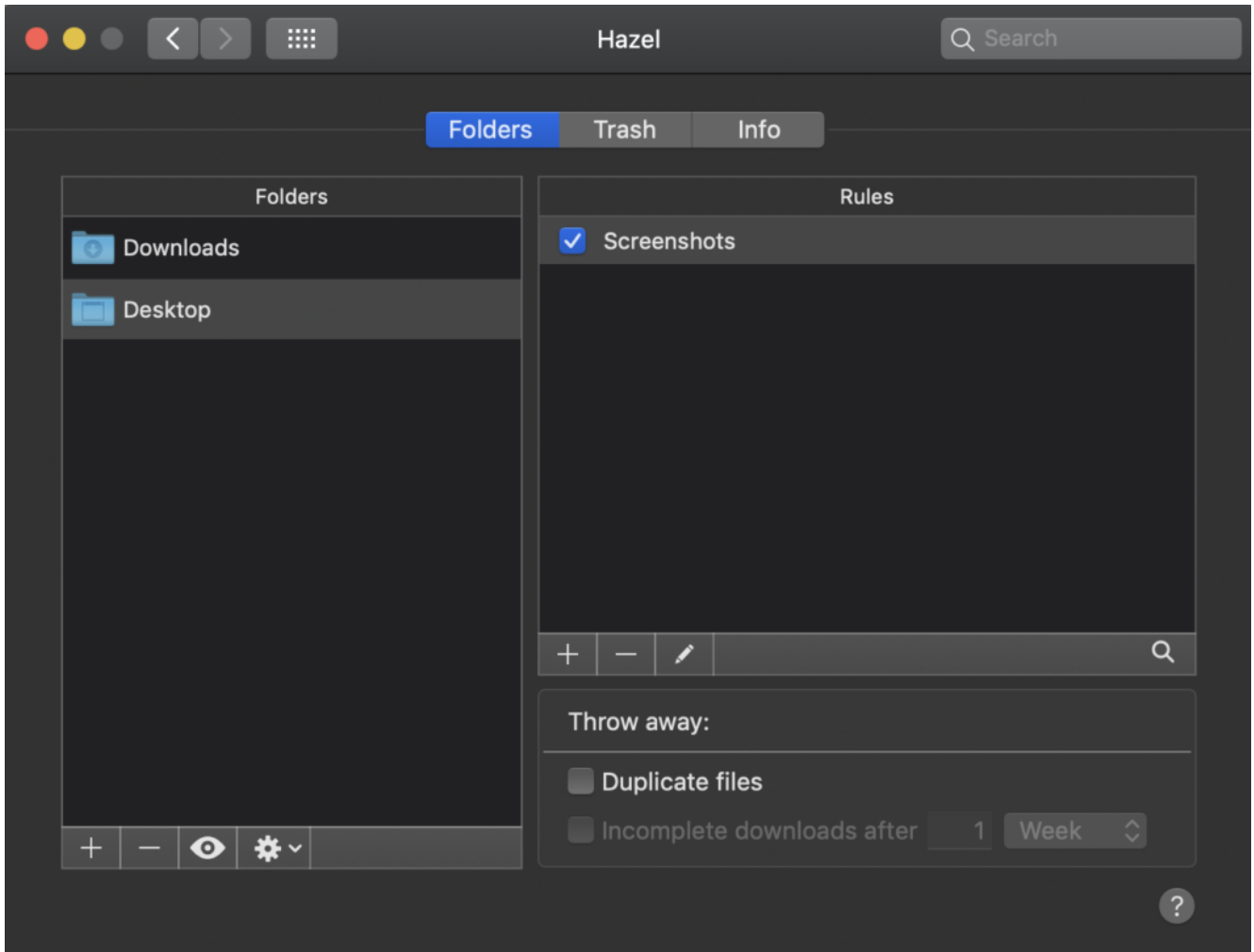


screenshots folder

I've used a number of screenshot applications on my computers in the last few years, and it's a never ending search for the perfect balance of flexibility and performance. Today I'm testing a new approach using native macOS grab functionality together with Hazel app.

# Hazel by Noodlesoft

In case you haven't heard about it, [Hazel](#) is an unbelievably powerful and flexible file management and automation tool for [macOS](#):



Hazel by Noodlesoft

You select incoming folders to process (in my screenshot: **Downloads** and **Desktop**) and then define rules for automatically processing files. Hazel is running in background, monitoring folders for matching rules you configured and applying them as it finds new files – within seconds of a new file matching a Hazel rule it gets dispatched.

Here's just a few of the workflows I have:

- Rename PDF files to a specific name based on some keywords found inside the PDF document itself (great for automatically renaming utility bills)
- Move ISO file downloads into specific folder on my NAS server
- Move large files from Downloads folder into special archive area on NAS (to save space on laptop)
- Rename all sorts of files to include datestamp – great for making these files easily found from [Bash scripts](#) later

## Common Issues with Screenshot Software

I've found that most screenshot making software (for macOS) is not flexible enough in one way or another:

- many apps are using proprietary database for storing images instead of standard folder
- there's no easy way to add screenshots from other software or folders – so unless you used the app to make a screenshot (or retake a screenshot of an existing image file) – it won't make it
- lots of apps allow you to export and import images but metadata is lost (like the date/time when the screenshot was taken)
- there's the problem of bloating up apps too, so something that started as a perfect screenshot tool can become an unusable (for me) collection of all sorts of

image and video capturing functionality that's just too slow or complex to use any longer

## The Problem with Cloud Sync

I would also like to have screenshots from different laptops and PCs to be synced if at all possible – and it's quite hard to find a screenshot app capable of doing this intelligently enough.

There's a few great solutions that are geared towards team work and collaboration, but the problem with them is I have to create separate account for each computer or that all the images end up in a cloud with potentially auto-generated links.

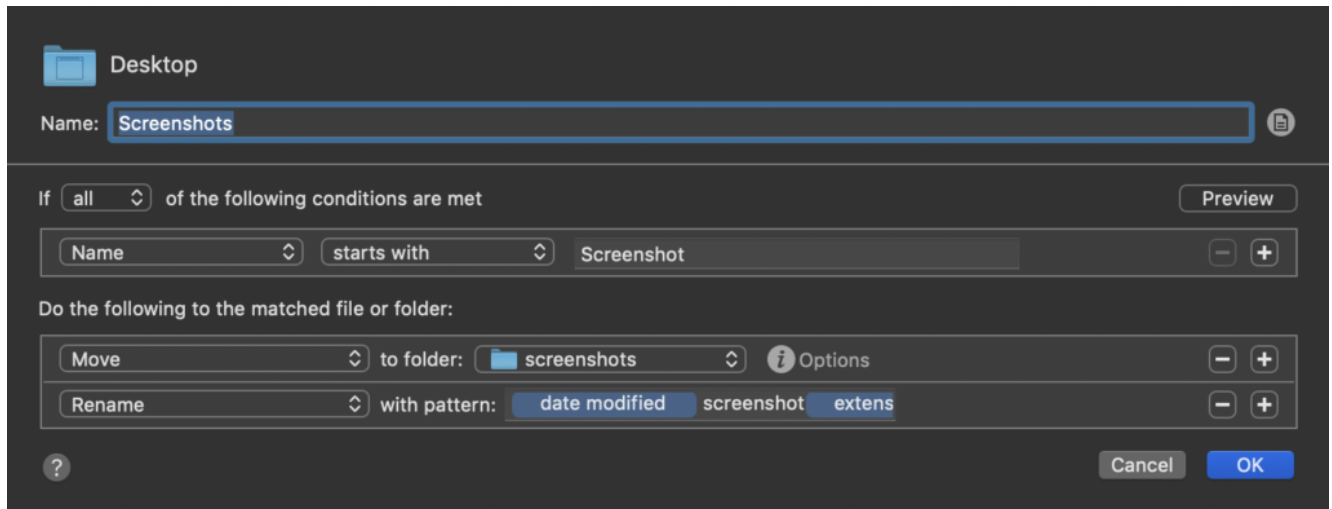
## Approach I'm trying Today

I decided to try a much simpler approach:

- use built-in screenshot tool in macOS
- use Hazel app to auto-collect screenshots from Desktop and place them into Pictures/screenshots folder
- use Hazel app to auto-rename screenshots for better file sorting
- Use Finder for accessing the screenshots folder with previews

# Configure folder processing rule in Hazel

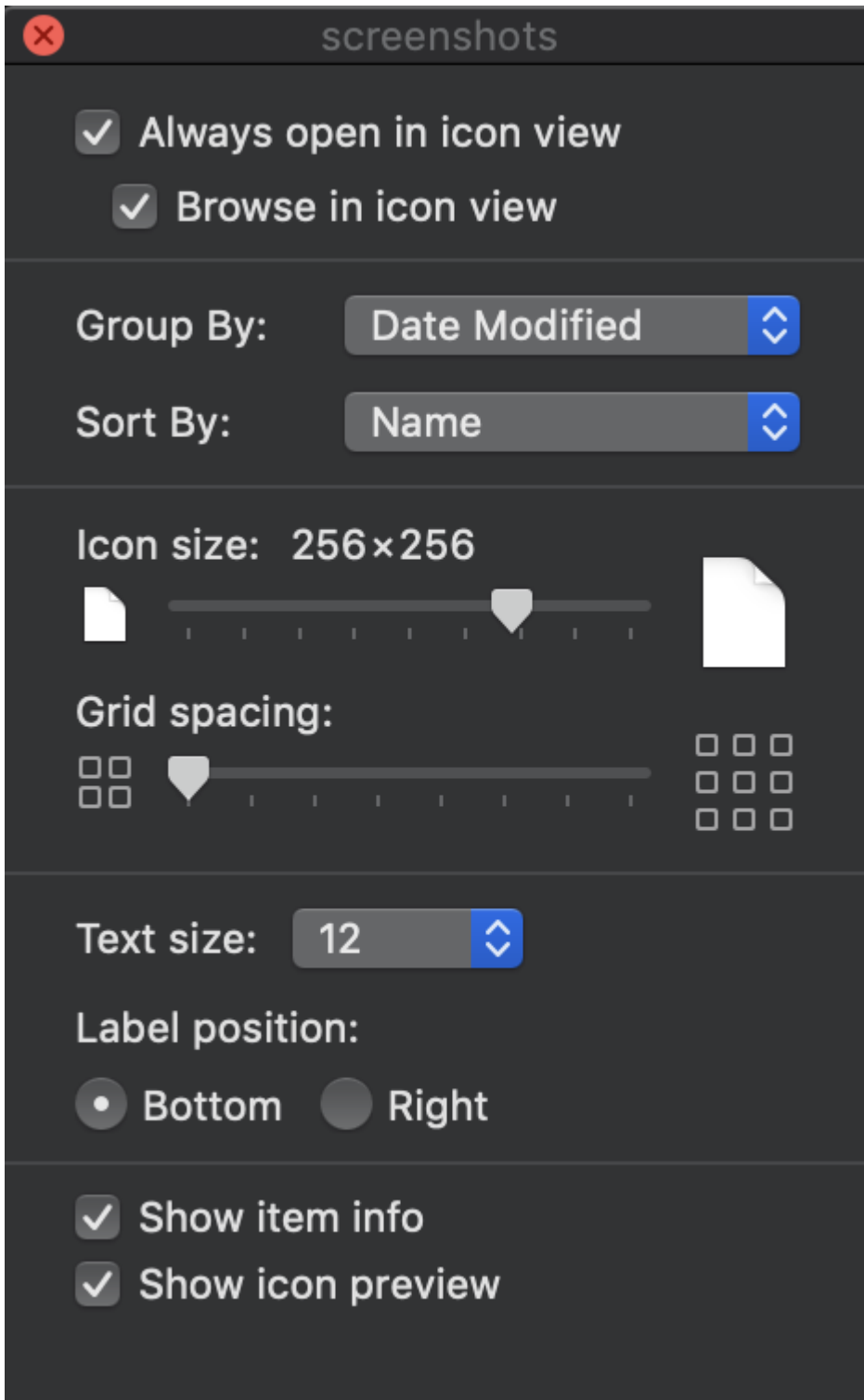
Here's how my rule looks:



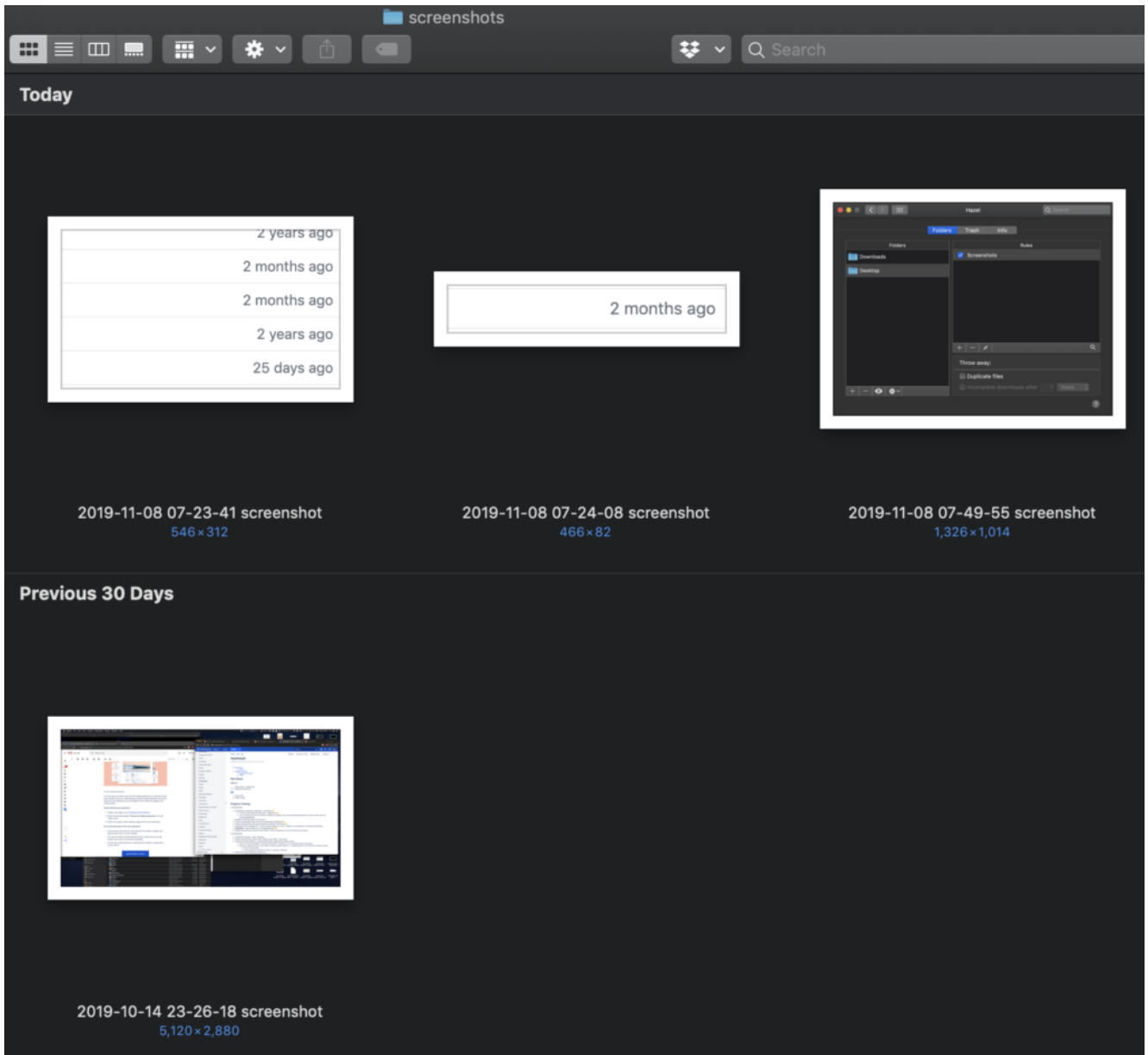
Hazel processing for screenshots

## macOS Finder settings for the screenshots folder

And these are the settings I've selected for the screenshots folders:



I think it's working out nicely: all my screenshots are sorted by the date they're taken, there's image size indication and thumbnails large enough for quickly finding the screenshot I want:



screenshots folder in macOS Finder

## Syncing screenshots folder

Documents folder is iCloud synced, so I'm hoping to configure this setup on both desktop and laptop, this way both of them will always have the same screenshot collection. Potentially, I can even use my Ubuntu 19.10 XPS laptop to capture screenshots into Dropbox folder, which can be processed by Hazel on macOS to import images.

That's it for now! Let me know how you're dealing with screenshots and what tools you use!

## See Also

- [Hazel by Noodlesoft](#)
- [macOS](#)
- [Assign keyboard shortcut to screenshot in Ubuntu](#)
- [Show off Unix distros with screenFetch](#)
- [Unix Tutorial](#)