

Your Laptop is not a Bop-It!

•••

Leveraging the Power of Your Keyboard to
Improve the Developer Experience

[Geoff Cureton](#)

@gpcureton@mstdn.social



https://bopit.fandom.com/wiki/Bop_It_Extreme

Overview

- My Origin Story
- Minimising Cognitive Load
 - Economy of Movement
 - Economy of Search
- Questions

Part I: My Origin Story

My Origin Story

- Undergrad in Physics ('94 - '96)
 - Windows 3.1
 - MS Word / Excel
 - Mouse Centered Workflow
 - Floppy Disks



West Nevada Community College, 199?

My Origin Story

- Undergrad in Physics ('94 - '96)
 - Windows 3.1
 - MS Word / Excel
 - Mouse Centered Workflow
 - Floppy Disks
- Honours Year (1997): When I Try To Get Serious...

manage documents

outlines and tables



graphics and artwork

SAMS

Teach Yourself

Microsoft®

Word 2000

in **24**
Hours

Heidi Steele

... but then I met this guy...

 BeeWare [About](#) [Documentation](#) [Community](#) [Contributing](#) [News](#) [Donate](#) [Edit on GitHub](#) [...](#)

[Home](#) > [About BeeWare](#)

The Bee Team

The core team behind BeeWare development

So who are the people behind BeeWare? Well, there's a huge group of contributors, but the project is managed by the Bee Team.

Current team

Russell Keith-Magee

Dr Russell Keith-Magee is the founder of the BeeWare project. He joined the Django core team in 2006, and for 5 years, was President of the Django Software Foundation. He has been a member of the CPython core team since May 2024.

Russell is a Principal Software Engineer in the Open Source group at Anaconda (a BeeWare Gold sponsor).



 [freakboy3742](#)
 [@freakboy3742@cloudisland.nz](#)
 [freakboy3742](#)
 russell@beeware.org
Superpower: Bee-nevolent Dictator,
For Now

... who “ruined” my life by introducing me to LaTeX 😞

```
108 \section{Simulation of Linear Surfaces\label{sec:Linear_sim}}
109
110 Since we desire to simulate ocean surfaces in the spectral domain, our simulation
111 method can be based around one of two representations of the linear, Gaussian
112 process \cite{Rice54}:
113 \begin{equation}
114 \label{eqn:linear_one}
115 \eta(x) = \sum^{N}_{n=1} a_n \cos(k_n x) + b_n \sin(k_n x) , ,
116 \end{equation}
117
118 or
119 \begin{equation}
120 \label{eqn:linear_two}
121 \eta(x) = \sum^{N}_{n=1} c_n \cos(k_n x + \phi_n) , .
122 \end{equation}
123
124
125 In (\ref{eqn:linear_one}), \(\( a_n \) \) and \(\( b_n \) \) are independent random,
126 Gaussian variables, with zero mean and variance
127 \[
128 \sigma^2_n = \Psi(k_n) \Delta k , , \]
129
130 while in (\ref{eqn:linear_two}) the amplitudes \(\( c_n \) \) are given by
131 \[
132 c_n = \sqrt{2\Psi(k_n) \Delta k} , , \]
133
134
135
```

2 Simulation of Linear Surfaces

Since we desire to simulate ocean surfaces in the spectral domain, our simulation method can be based around one of two representations of the linear, Gaussian process (Rice 1954):

$$\eta(x) = \sum_{n=1}^N a_n \cos(k_n x) + b_n \sin(k_n x) , \quad (1)$$

or

$$\eta(x) = \sum_{n=1}^N c_n \cos(k_n x + \phi_n) . \quad (2)$$

In (1), a_n and b_n are independent random, Gaussian variables, with zero mean and variance

$$\sigma_n^2 = \Psi(k_n) \Delta k ,$$

while in (2) the amplitudes c_n are given by

$$c_n = \sqrt{2\Psi(k_n) \Delta k} ,$$

But seriously... why was exposure to LaTeX significant?

- It introduced me to the idea that a visual element and its representation/implementation can be decoupled (like HTML, Markdown...)

But seriously... why was exposure to LaTeX significant?

- It introduced me to the idea that a visual element and its representation/implementation can be decoupled (like HTML, Markdown...)
- Also...

But seriously... why was exposure to LaTeX significant?

- It introduced me to the idea that a visual element and its representation/implementation can be decoupled (like HTML, Markdown...)
- Also... LaTeX is cool.



But Also...

- I ran my LaTeX “distribution” em-TeX using a little GUI called “emTeXGI”.
- Sadly, emTeXGI has been memory-holed; no images of it remain, but it looked something like this...



But Also...

- I ran my LaTeX “distribution” em-TeX using a little GUI called “emTeXGI”.
- Sadly, emTeXGI has been memory-holed; no images of it remain, but it looked something like this...

- Alt-Tab between GUI and source
- Tab between “Save”, “Compile”

- I became irate when I couldn’t do this.



The Interregnum

- Eventually I went to grad school, and intermittently used Vim (oooh, foreshadowing!), but not in a serious way.
- Various jobs required me to use Windows, Visual Studio.NET, and other mousey things.
- This situation persisted for good while, punctuated by my occasional insistence on using LaTeX to make presentations (I'm cured of that), and conference posters (of which I'm still afflicted).
- Then came...

The Pandemic!

The Pandemic

- While I had tinkered around with keyboard-based workflows previously, I hadn't seriously put any time into it.
- By the middle of 2020, I found that I was spending a lot more time on my computer (for good or ill), and was itching to do something different.
- A few diversions started to take a more important role:
 - Coding YouTubers
 - Coding YouTubers who use Vim
 - Mechanical Keyboards
- Combining these interests led me to...

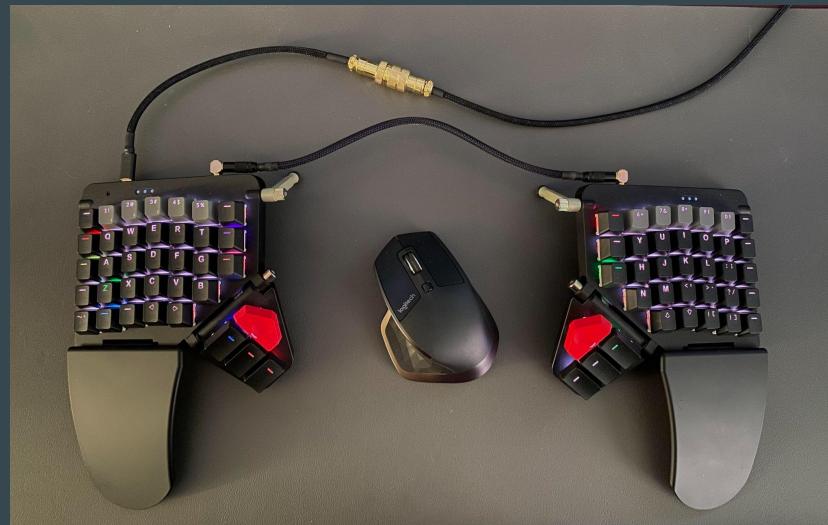
Part II: Minimizing Cognitive Load

Minimizing Cognitive Load

- Reducing behaviours and actions which interrupt your current task. People are bad multitaskers. They just are.
- The things I do to streamline my workflow fall into two broad categories:
 - Economy of Movement
 - Economy of Search

Economy of Movement

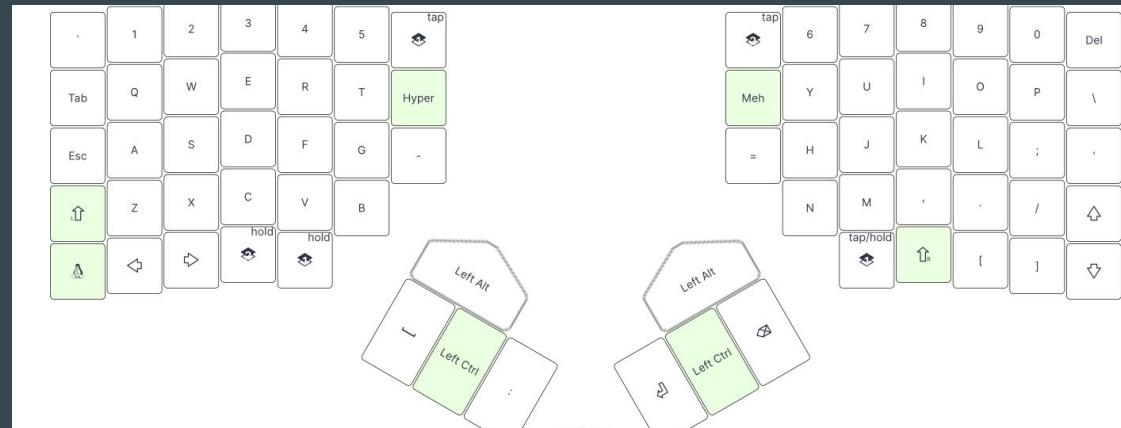
- This is where the concept of the Bop-It comes in: you will be more effective if you are not flailing away at your keyboard. Or between the keyboard and mouse.
- Use your mouse when you need to, and your keyboard when you don't
- For just the keyboard, economy of movement can be achieved either...
- Hardware (keyboards)



Economy of Movement

- This is where the concept of the Bop-It comes in: you will be more effective if you are not flailing away at your keyboard. Or between the keyboard and mouse.
- Use your mouse when you need to, and your keyboard when you don't
- For just the keyboard, economy of movement can be achieved either...

- Or Software (key-mappings)



Keyboards

- I don't get too invested in the notion of and “ergonomic” keyboard. I'm sure it's a thing, but 
- Split keyboards (as on the previous slides) are fine, but a good layout can be just as good.
- My most used keyboard...

Keyboards

- I don't get too invested in the notion of and “ergonomic” keyboard. I'm sure it's a thing, but 🤷
- Split keyboards (as on the previous slides) are fine, but a good layout can be just as good.
- My most used keyboard...



Keyboards

- Orthocolumnar! (keys are in vertical rows, not staggered)
- This is a 40% keyboard (w.r.t. a full-size keyboard's number of keys)
- How do we cram all the functionality we want into such a small size...



Keyboards

- Orthocolumnar! (keys are in vertical rows, not staggered)
- This is a 40% keyboard (w.r.t. a full-size keyboard's number of keys)
- How do we cram all the functionality we want into such a small size...
- LAYERS!



Keyboards

- You already use layers: the SHIFT key takes you to the “CAPS and SYMBOLS” layer.
- This keyboard has layer “raise” and “lower” keys either side of the space bar.
- LED lighting of the keys are used as a visual cue



Keyboards

- The “lower” layer converts the top row to the usual symbols you’d find along the top of a keyboard
- The “< >  <img alt="up arrow" data-bbox="232 188 272 200}” keys are Line start/end, Page down/up respectively.- “hjkl” are left/down/up/right respectively (more foreshadowing!)



Keyboards

- The “raise” layer converts the top row to the usual numbers.
- Makes an upside-down keypad: QWEASDZXCV ➔ 1-9,0
- “Keypad” is purple, as a visual cue



Keyboards

- Both layer keys simultaneously activates the “parenthesis” layer
- “RU”, “FJ” and “VM” give the “()”, “[]” and “[{}” pairs respectively. Note the symmetry...
- “D” and “K” give “^” and “\$” (IYKYK)



Hot Keys

- Sometimes you don't have your fancy programmable keyboard with you, what do you do then? Hotkey daemons!
- Sxhkd (<https://github.com/baskerville/sxhkd>), Linux
- Kanata (<https://github.com/jtroo/kanata>), Linux
- Skhd (<https://github.com/asmvik/skhd>), MacOS

```
#      MULTIMEDIA KEYS
#####
#
#Raises volume
XF86AudioRaiseVolume
    amixer set Master 10%+
#
#Lowers volume
XF86AudioLowerVolume
    amixer set Master 10%-
#
#Mute
XF86AudioMute
    amixer -D pulse set Master 1% toggle
```

Hot Keys

- Key mapping daemons are generally used to create mappings which are independent of the OS or desktop environment being used
- In the example below, the laptop media keys are being mapped to a particular audio mixer app.

```
#      MULTIMEDIA KEYS
#####
#Raises volume
XF86AudioRaiseVolume
    amixer set Master 10%+

#Lowers volume
XF86AudioLowerVolume
    amixer set Master 10%-

#Mute
XF86AudioMute
    amixer -D pulse set Master 1+ toggle
```

Web Browsing

- Navigating to links on a webpage might seem like “search”, but you already have to find the link with your eyes, so it’s just movement.



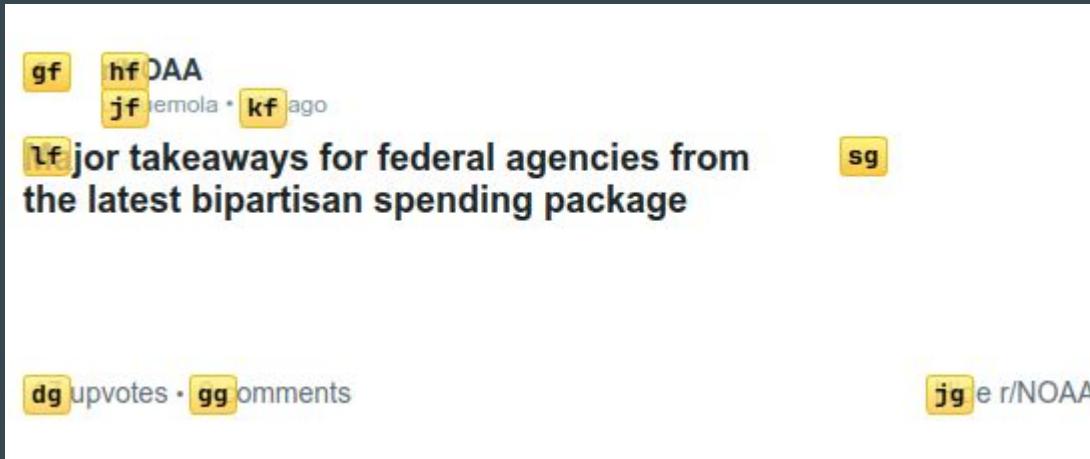
A screenshot of a Reddit post from the subreddit r/NOAA. The post is titled "Major takeaways for federal agencies from the latest bipartisan spending package". It was posted 19 hours ago by u/bluemola. The post has 17 upvotes and 0 comments. There are options to "Hide r/NOAA" and "Edit".

17 upvotes • 0 comments

Hide r/NOAA

Web Browsing

- Navigating to links on a webpage might seem like “search”, but you already have to find the link with your eyes, so it’s just movement.
- Shown below is “Qutebrower”, “f” for “find” brings up key combinations to select the various links. Note that each of the character pairs have either characters accessible from either hand, or by a comfortable rolling motion of the same hand. No awkward stretches.



Economy of Search

- Optimizing the process of searching can also be thought of as optimizing navigation: if we're looking for something, we'd rather be able to teleport there rather than walk.
- We can be searching for a particular string in a text file, or a workspace on our desktop, with the aim of going there
- Rather than inspecting a large list/table to find the thing we want (and clicking on it), it is preferable to type a few characters to zero in on what we want.

Fuzzy Finders

- The notion of “fuzzy finding” can be applied to a range of different contexts:
 - finding the app you want to load
 - finding a file on your file system
 - finding a string in a text file
- The utility of fuzzy finding depends on at least having some idea of what you are looking for, i.e.: part of the name
- There are some instances where a text-based search my be slower that just using your Mark-1 Eyeball to directly find the thing: they will be discussed as we get to them

FZF (<https://github.com/junegunn/fzf>)

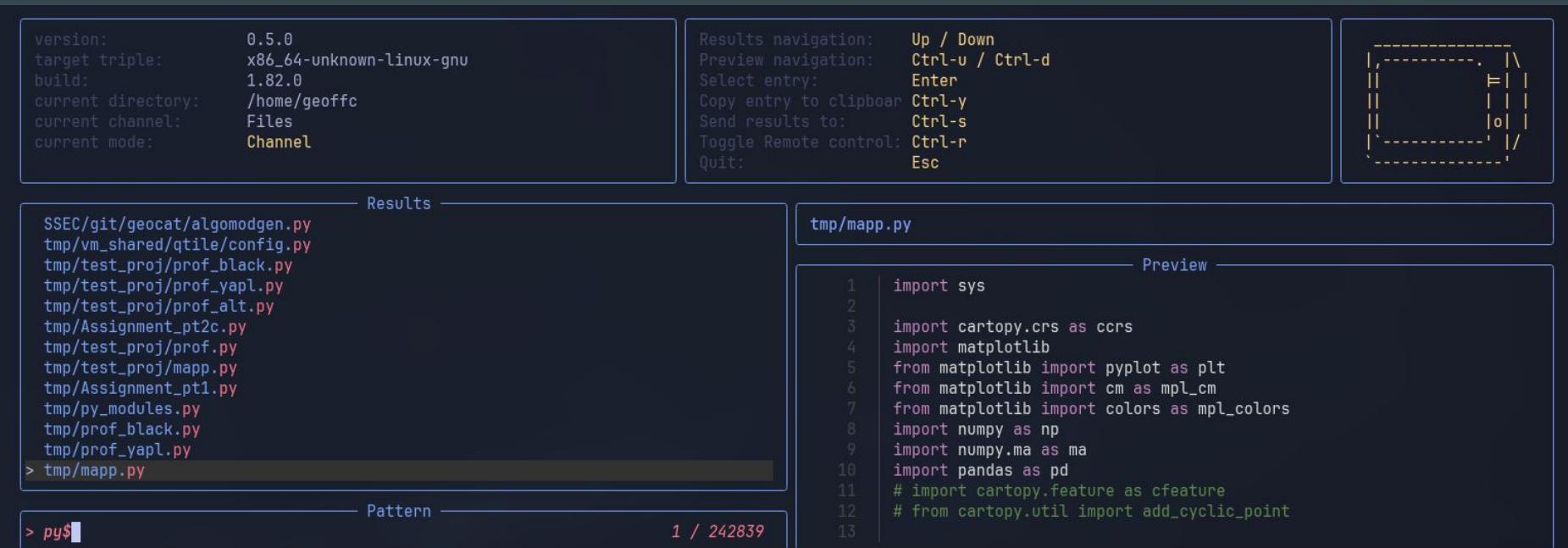
- A fuzzy finder for your command line
- Can search your command history, or for a file on your filesystem (and many things besides).
- CTRL-R and CTRL-T give history and filesystem fuzzy finding respectively

```
366 rm -f ./cspp-sounder-ql-1.4.tar.gz*  
367 cd /data/users/geoffc/CSPP_Sounder_QL_dev/git/cspp_sounder_ql-worktrees/legacy  
368 echo /data/users/geoffc/CSPP_Sounder_QL_dev  
369 cd /data/users/geoffc/CSPP_Sounder_QL_dev/work ; ls -la  
370 cd /data/users/geoffc/CSPP_Sounder_QL_dev ; ls -la  
371 cd /data/users/geoffc/CSPP_Sounder_QL_dev/sample_data ; ls -la  
372 cd /data/users/geoffc/CSPP_Sounder_QL_dev/package_tree/ ; ls -la  
373 export CSPP_SOUNDER_QL_HOME=/data/users/geoffc/CSPP_Sounder_QL_dev/package_tree/cspp-so  
374 cd /data/users/geoffc/CSPP_Sounder_QL_dev/git/cspp_sounder_ql-worktrees/legacy ; ls -la  
375 cd /data/users/geoffc/CSPP_Sounder_QL_dev/tarballs ; ls -la  
376 cd /data/users/geoffc/CSPP_Sounder_QL_dev/pkg_test ; ls -la  
377 echo $CSPP_SOUNDER_QL_HOME  
378 rm -rf cspp-sounder-ql-1.4  
379 sh cspp-sounder-ql/cspp-sounder-ql_packaging.sh cspp-sounder-ql 1.4 ..tarballs  
380 cd cspp-sounder-ql-1.4  
381 ls -la /data/users/geoffc/CSPP_Sounder_QL_dev/tarballs/cspp-sounder-ql-1.4  
382 nvim scripts/sounder_image.py  
383 git commit -a -m"Reversed the order of the colortable for Lifted Index, and reinstated  
384 sh /data/users/geoffc/CSPP_Sounder_QL_dev/setup/tmux_cspp_sounder_ql.sh  
385 nvim scripts/sounder_image scripts/sounder_packages/heap.py  
386  
387 98/303 +S  
sounder
```

```
> .py$  
77866/1293631 (0) —  
miniconda3/lib/python3.1/site-packages/pygments/lexers/d.py  
miniconda3/lib/python3.1/site-packages/pygments/lexers/j.py  
miniconda3/lib/python3.1/site-packages/pygments/lexers/q.py  
miniconda3/lib/python3.1/site-packages/pygments/lexers/r.py  
miniconda3/lib/python3.10/site-packages/pygments/lexers/d.py  
miniconda3/lib/python3.10/site-packages/pygments/lexers/j.py  
miniconda3/lib/python3.10/site-packages/pygments/lexers/q.py  
miniconda3/lib/python3.10/site-packages/pygments/lexers/r.py  
miniconda3/lib/python3.10/site-packages/pygments/lexers/r.py  
miniconda3/lib/python3.1.c~/site-packages/pygments/lexers/d.py  
miniconda3/lib/python3.1.c~/site-packages/pygments/lexers/j.py  
miniconda3/lib/python3.1.c~/site-packages/pygments/lexers/q.py  
miniconda3/lib/python3.1.c~/site-packages/pygments/lexers/r.py  
miniconda3/pkgs/pygments-2.19.1-pyhd8ed1ab_0/site-packages/pygments/lexers/d.py  
miniconda3/pkgs/pygments-2.19.1-pyhd8ed1ab_0/site-packages/pygments/lexers/j.py  
miniconda3/pkgs/pygments-2.19.1-pyhd8ed1ab_0/site-packages/pygments/lexers/q.py  
miniconda3/pkgs/pygments-2.19.1-pyhd8ed1ab_0/site-packages/pygments/lexers/r.py  
.local/share/oldshit/nvim/mason/packages/ruff-lsp/venv/lib/python3.10/site-packages/catt  
.local/share/oldshit/nvim/mason/packages/ruff-lsp/venv/lib/python3.10/site-packages/catt  
miniconda3/lib/python3.1/io.py  
miniconda3/lib/python3.1/os.py
```

television (<https://github.com/alexpasmantier/television>)

- “A very fast, portable and hackable fuzzy finder for the terminal.”
- Can search filesystem, code, processes, git repos, env vars, docker containers...



version: 0.5.0
target triple: x86_64-unknown-linux-gnu
build: 1.82.0
current directory: /home/geoffc
current channel: Files
current mode: Channel

Results navigation: Up / Down
Preview navigation: Ctrl-u / Ctrl-d
Select entry: Enter
Copy entry to clipboard: Ctrl-y
Send results to: Ctrl-s
Toggle Remote control: Ctrl-r
Quit: Esc

Results -

```
SSEC/git/geocat/algomodgen.py
tmp/vm_shared/qtile/config.py
tmp/test_proj/prof_black.py
tmp/test_proj/prof_yapl.py
tmp/test_proj/prof_alt.py
tmp/Assignment_pt2c.py
tmp/test_proj/prof_py
tmp/test_proj/mapp.py
tmp/Assignment_pt1.py
tmp/py_modules.py
tmp/prof_black.py
tmp/prof_yapl.py
> tmp/mapp.py
```

Pattern -

1 / 242839

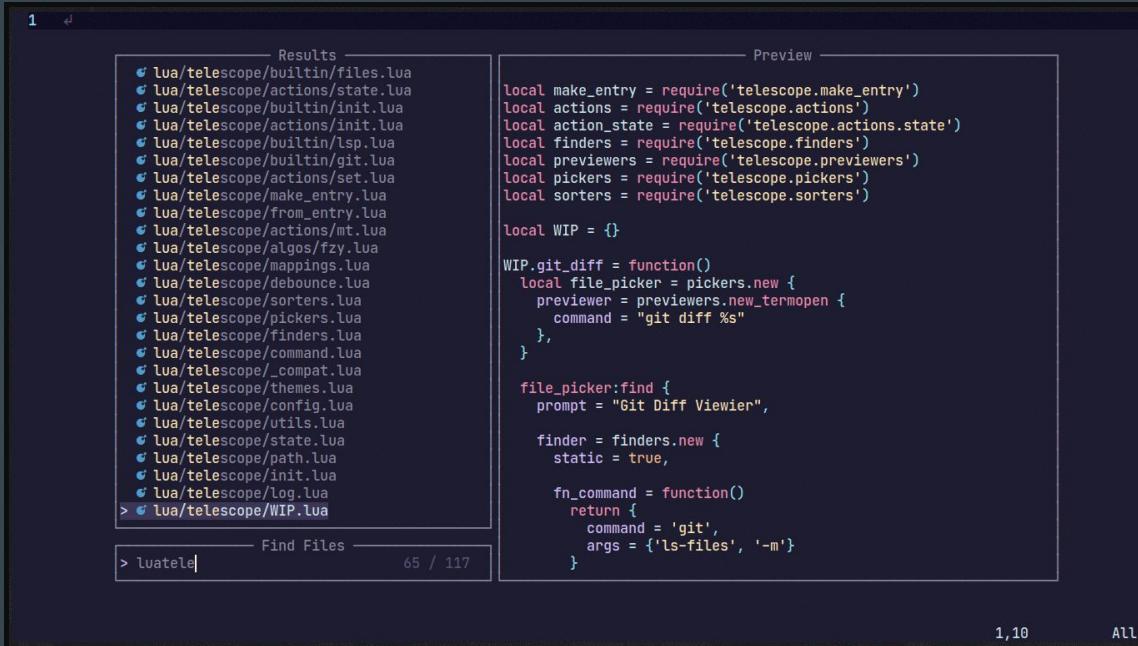
tmp/mapp.py

Preview -

```
1 import sys
2
3 import cartopy.crs as ccrs
4 import matplotlib
5 from matplotlib import pyplot as plt
6 from matplotlib import cm as mpl_cm
7 from matplotlib import colors as mpl_colors
8 import numpy as np
9 import numpy.ma as ma
10 import pandas as pd
11 # import cartopy.feature as cfeature
12 # from cartopy.util import add_cyclic_point
```

Telescope (<https://github.com/nvim-telescope/telescope.nvim>)

- The first general-purpose fuzzy finder plugin for NeoVim.
- You could add your own finders (say, for your Markdown notes)



The screenshot shows the Telescope fuzzy finder interface in NeoVim. The left panel, titled "Results", lists numerous Lua files within the telescope directory, including files like "builtin/files.lua", "actions/state.lua", "builtin/init.lua", "actions/init.lua", "builtin/lsp.lua", "builtin/git.lua", "actions/set.lua", "make_entry.lua", "from_entry.lua", "actions/mt.lua", "algos/fzy.lua", "mappings.lua", "debounce.lua", "sorters.lua", "pickers.lua", "finders.lua", "command.lua", "themes.lua", "config.lua", "utils.lua", "state.lua", "path.lua", "init.lua", and "log.lua". The file "WIP.lua" is highlighted with a blue selection bar. The right panel, titled "Preview", shows the content of "WIP.lua". The code defines a "WIP" local variable containing a table with a "git_diff" function. This function creates a file picker using the "pickers.new" function, sets its "prompt" to "Git Diff Viewier", and its "finder" to a "finders.new" function with "static" set to true. The "fn_command" of the picker is defined as a function that returns a command object with "command" set to "git" and "args" set to {"'ls-files', '-m'}. The status bar at the bottom shows "luatele" and "65 / 117".

```
local make_entry = require('telescope.make_entry')
local actions = require('telescope.actions')
local action_state = require('telescope.actions.state')
local finders = require('telescope.finders')
local previewers = require('telescope.previewers')
local pickers = require('telescope.pickers')
local sorters = require('telescope.sorters')

local WIP = {}

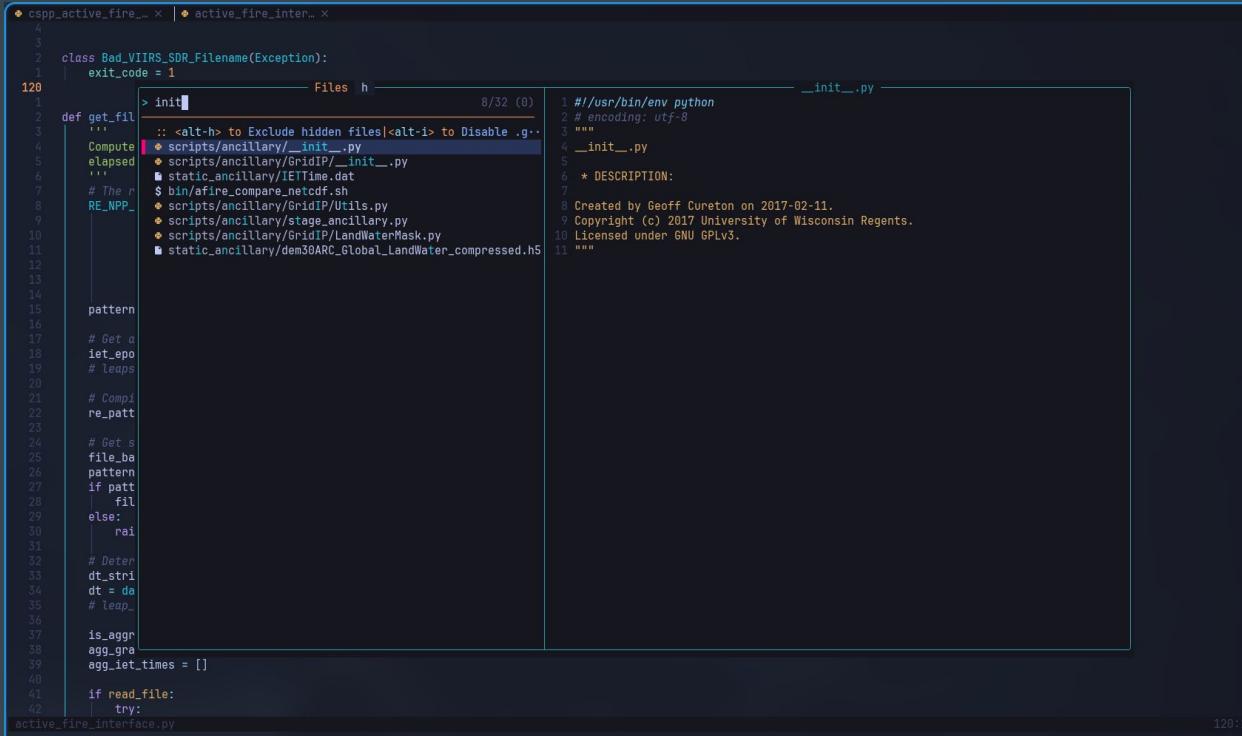
WIP.git_diff = function()
  local file_picker = pickers.new {
    previewer = previewers.new_termopen {
      command = "git diff %s"
    },
    file_picker:find {
      prompt = "Git Diff Viewier",
      finder = finders.new {
        static = true,
        fn_command = function()
          return {
            command = 'git',
            args = {'ls-files', '-m'}
          }
        end
      }
    }
  }
end
```

LazyVim (<https://www.lazyvim.org/>)

- A NeoVim “distribution” which provides a range of different search modes....

LazyVim (<https://www.lazyvim.org/>)

- Files



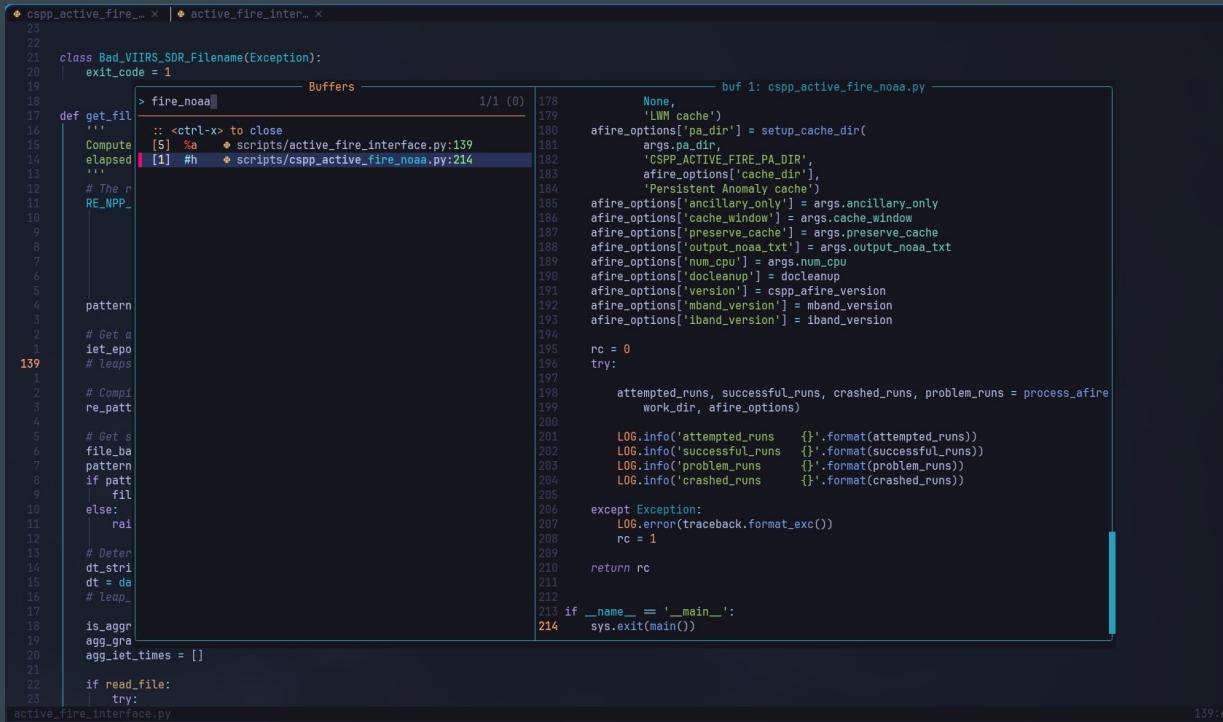
```
1  cspp_active_fire... x | active_fire_inter... x
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
120
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
120:1
class Bad_VIIRS_SDR_Filename(Exception):
    exit_code = 1

    def get_file(self, pattern):
        Compute_elapsed_time()
        # The n
        RE_NPP = re.compile(pattern)
        # Get a
        iet_epo = RE_NPP.findall(pattern)
        # leaps
        # Compi
        re_patt
        # Det
        file_ba
        pattern
        if patt
            fil
        else:
            rai
        # Deter
        dt_stri
        dt = da
        # leap_
        is_aggr
        agg_gra
        agg_get_times = []
        if read_file:
            try:
> init___.py
:: <alt-h> to Exclude hidden files|<alt-i> to Disable .gitignore
scripts/ancillary/_init__.py
scripts/ancillary/GridIP/_init__.py
static_ancillary/IETTime.dat
$ bin/afire.compare.netcdf.sh
scripts/ancillary/GridIP/Utils.py
scripts/ancillary/stage_ancillary.py
scripts/ancillary/GridIP/LandWaterMask.py
static_ancillary/dem30ARC_Global_LandWater_compressed.h5
# //usr/bin/env python
# encoding: utf-8
"""
__init__.py
* DESCRIPTION:
* Created by Geoff Cureton on 2017-02-11.
* Copyright (c) 2017 University of Wisconsin Regents.
* Licensed under GNU GPLv3.
"""

```

LazyVim (<https://www.lazyvim.org/>)

- Buffers



The image shows a screenshot of the LazyVim editor interface. It features a split window layout with two buffers. The left buffer is titled 'active_fire_interface.py' and contains Python code for a 'fire_noaa' function. The right buffer is titled 'buf 1: cspp_active_fire_noaa.py' and contains a similar Python script with some differences in variable names and logic. The code in both buffers is color-coded for syntax highlighting, and the interface includes standard Vim-style navigation and status bars.

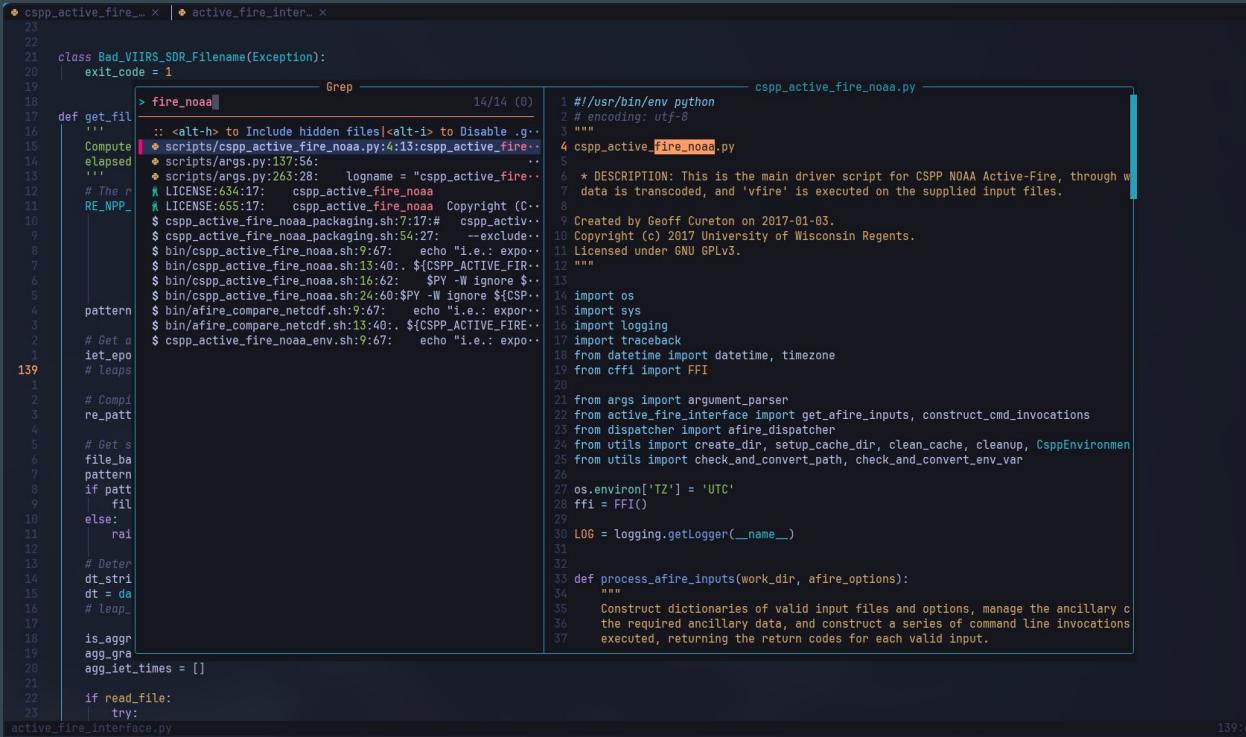
```
18  def get_file():
19      > fire_noaa
20      :: <ctrl-x> to close
21      Compute
22      elapsed
23      RE_NPP_
24      pattern
25      # Get a
26      ist_epo
27      # leaps
28      #
29      # Compil
30      re_patt
31      # Gets
32      file_da
33      pattern
34      if patt
35          fil
36      else:
37          rai
38      # Deter
39      dt_stri
40      dt = da
41      # leap_
42      is_aggr
43      agg_gna
44      agg_ieet_times = []
45      if read_file:
46          try:
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
```

LazyVim (<https://www.lazyvim.org/>)

- Search within current buffer

LazyVim (<https://www.lazyvim.org/>)

- Grep files in CWD (with ripgrep!)



The image shows a screenshot of the LazyVim editor. It has a dark theme with a split terminal window. The left pane shows the results of a ripgrep search for 'fire_noaa' across multiple files in the current working directory. The right pane shows the content of a Python script named 'cspp_active_fire_noaa.py'.

```
cspp_active_fire_noaa.py | grep fire_noaa
```

```
#!/usr/bin/env python
# encoding: utf-8
"""
cspp_active_fire_noaa.py

* DESCRIPTION: This is the main driver script for CSPP NOAA Active-Fire, through which
  data is transcoded, and 'vfine' is executed on the supplied input files.
* Created by Geoff Cureton on 2017-01-03.
* Copyright (c) 2017 University of Wisconsin Regents.
* Licensed under GNU GPLv3.

import os
import sys
import logging
import traceback
from datetime import datetime, timezone
from ffi import FFI
from args import argument_parser
from active_fire_interface import get_afire_inputs, construct_cmd_invocations
from dispatcher import afire_dispatcher
from utils import create_dir, setup_cache_dir, clean_cache, cleanup, CsppEnvironment
os.environ['TZ'] = 'UTC'
ffi = FFI()
LOG = logging.getLogger(__name__)

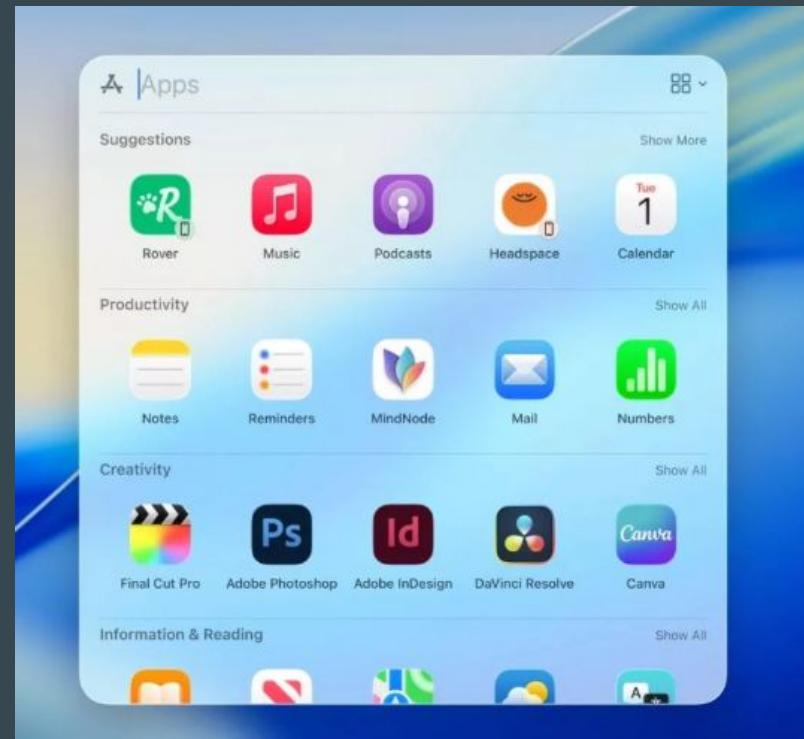
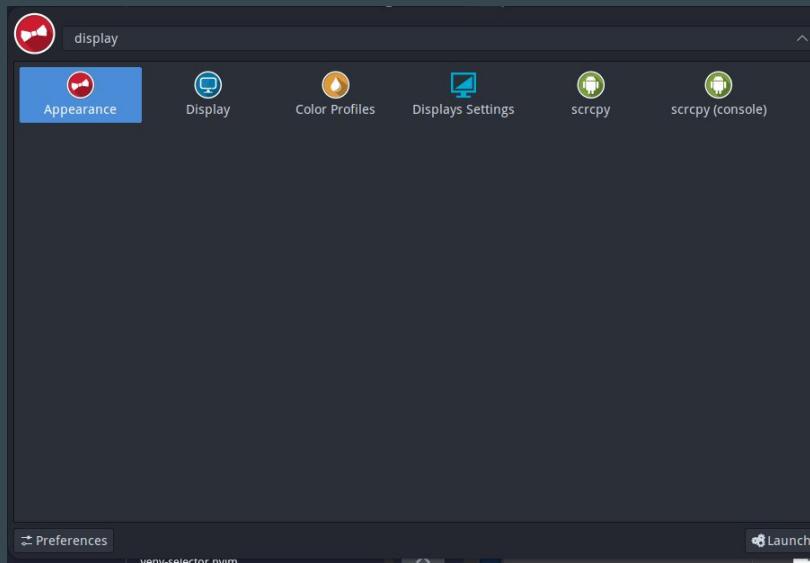
def process_afire_inputs(work_dir, afire_options):
    """
    Construct dictionaries of valid input files and options, manage the ancillary
    data, and construct a series of command line invocations
    executed, returning the return codes for each valid input.
    
```

Harpoon (<https://github.com/ThePrimeagen/harpoon>)

- When working in a large codebase, it's common to need to jump back and forth between one of a few points (say, a function usage and its definition)
- Harpoon is predicated on the idea that people cannot meaningfully use more than several points of context at a time.
- Each of those points can be made easily accessible using a hotkey: <something>-1, <something>-2 etc...
- Using these limited number of mappings, one can rapidly navigate code relevant to a task. As the task changes, so do the mappings.
- This workflow is an alternative to trying to have all of your context on screen at once... bigger monitors!, more monitors!

Application Pickers

- Beats navigating down a menu tree!



Miscellaneous

- There are some parts of my workflow that don't fit neatly a particular category.
- I'll just go through them and discuss why I find them useful

Vim / NeoVim / Emacs + TMUX

- A text editor with useful keybinds to navigate between the various elements (buffers, windows, dialogs, ...)
- Many editors outside of the main TUI based ones have plugins to replicate their behavior (VS Code - VIM keybinds)
- A terminal-based editor can be run “locally” on a remote server (inside a Screen/TMUX/Zellij session), you can log out, go home, and pick up where you left off.

```
Neo-tree
► ~/SSEC/CSPP/CSPP-ActiveFire/git/cspp_
  └─ __pycache__
  └─ bin
  └─ scripts
    └─ ancillary
      └─ active_fire_interface.py
      └─ afire_compare_netcdf.py
      └─ args.py
      └─ cspp_active_fire_noaa.py
      └─ dispatcher.py
      └─ log_common.py
      └─ unaggregate.py
      └─ utils.py
  └─ static_ancillary
  └─ vendor
  └─ LICENSE
  └─ README.md
  * VERSION
  $ cspp_active_fire_noaa_env.sh
  $ cspp_active_fire_noaa_packaging.sh
  $ cspp_active_fire_noaa_runtime.sh
  $ tag_version.sh
  (3 hidden items)
```

```
Neo-tree
▶ □ /data/users/geoffc/CSPP_Sounder_QL_de
  □ bin
  □ scripts
    □ sounder_packages
      □ args.py
      □ exceptions.py
      □ log_common.py
      □ ql_common.py
      □ sounder_image.py
      □ sounder_image_data.py
      □ sounder_profile.py
      □ sounder_skewt.py
      □ sounder_skewt_data.py
      □ sounder_slice.py
      □ sounder_slice_data.py
      □ thermo.py
    □ utils.py
  (1 hidden item)
  □ vendor
  □ LICENSE
  □ README.md
  * VERSION
  $ cspc-sounder-ql_packaging.sh
  $ cspc-sounder-ql_unpack_check.sh
  $ cspc_sounder_ql_env.sh
  $ cspc_sounder_ql_runtime.sh
  □ ruff.toml
  $ tag_version.sh
  (4 hidden items)
```

```
• sounder_image.py o 4 a 1 x • thermo.py a 1 x
25
24 Minimum commandline:
23
22 | python sounder_image.py INPUTFILE DATATYPE
21
20 where...
19
18 | INPUTFILES: The fully qualified path to the input files. May be a
17 | file glob.
16
15 | DATATYPE: One of 'iapp','mirs', 'hsrtv', 'nucaps' OR 'heap'.
14
13
12 | Created by Geoff Cureton <geoff.cureton@ssec.wisc.edu> on 2014-05-10.
11 | Copyright 2025 University of Wisconsin Regents.
10 | Licensed under GNU GPLv3.
9 | """
8
7 import os
6 import sys
5 import logging
4
3 from args import argument_parser_image
2
1 import sounder_image_data
49 from ql_common import granuleFiles
1 from ql_common import set_plot_styles
2
3 import sounder_packages
4
5 # every module should have a LOG object
6 LOG = logging.getLogger(__file__)
7
8
9 def main():
10 | """
11 | The main method.
12 | """
13
14 | # Read in the command line options
15 args, work_dir = argument_parser_image()
16
17 | input_files = args.inputs
18 | datatype = args.datatype
19 | dataset = args.dataset
20 | pressure = args.pressure
21 | quality_flag = args.use_quality_flag
22 | lat_0 = args.lat_0
23 | lon_0 = args.lon_0
24 | bounding_lat = args.bounding_lat
25 | extent = args.extent
26 | plot_type = args.plot_type
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
259
260
261
262
263
264
265
266
267
268
269
269
270
271
272
273
274
275
276
277
278
279
279
280
281
282
283
284
285
286
287
288
289
289
290
291
292
293
294
295
296
297
298
299
299
300
301
302
303
304
305
306
307
308
309
309
310
311
312
313
314
315
316
317
317
318
319
319
320
321
322
323
324
325
326
327
327
328
329
329
330
331
332
333
334
335
336
337
337
338
339
339
340
341
342
343
344
345
345
346
347
347
348
349
349
350
351
352
353
354
355
355
356
357
357
358
359
359
360
361
362
363
363
364
365
365
366
367
367
368
368
369
369
370
371
371
372
372
373
373
374
374
375
375
376
376
377
377
378
378
379
379
380
380
381
381
382
382
383
383
384
384
385
385
386
386
387
387
388
388
389
389
390
390
391
391
392
392
393
393
394
394
395
395
396
396
397
397
398
398
399
399
400
400
401
401
402
402
403
403
404
404
405
405
406
406
407
407
408
408
409
409
410
410
411
411
412
412
413
413
414
414
415
415
416
416
417
417
418
418
419
419
420
420
421
421
422
422
423
423
424
424
425
425
426
426
427
427
428
428
429
429
430
430
431
431
432
432
433
433
434
434
435
435
436
436
437
437
438
438
439
439
440
440
441
441
442
442
443
443
444
444
445
445
446
446
447
447
448
448
449
449
450
450
451
451
452
452
453
453
454
454
455
455
456
456
457
457
458
458
459
459
460
460
461
461
462
462
463
463
464
464
465
465
466
466
467
467
468
468
469
469
470
470
471
471
472
472
473
473
474
474
475
475
476
476
477
477
478
478
479
479
480
480
481
481
482
482
483
483
484
484
485
485
486
486
487
487
488
488
489
489
490
490
491
491
492
492
493
493
494
494
495
495
496
496
497
497
498
498
499
499
500
500
501
501
502
502
503
503
504
504
505
505
506
506
507
507
508
508
509
509
510
510
511
511
512
512
513
513
514
514
515
515
516
516
517
517
518
518
519
519
520
520
521
521
522
522
523
523
524
524
525
525
526
526
527
527
528
528
529
529
530
530
531
531
532
532
533
533
534
534
535
535
536
536
537
537
538
538
539
539
540
540
541
541
542
542
543
543
544
544
545
545
546
546
547
547
548
548
549
549
550
550
551
551
552
552
553
553
554
554
555
555
556
556
557
557
558
558
559
559
560
560
561
561
562
562
563
563
564
564
565
565
566
566
567
567
568
568
569
569
570
570
571
571
572
572
573
573
574
574
575
575
576
576
577
577
578
578
579
579
580
580
581
581
582
582
583
583
584
584
585
585
586
586
587
587
588
588
589
589
590
590
591
591
592
592
593
593
594
594
595
595
596
596
597
597
598
598
599
599
600
600
601
601
602
602
603
603
604
604
605
605
606
606
607
607
608
608
609
609
610
610
611
611
612
612
613
613
614
614
615
615
616
616
617
617
618
618
619
619
620
620
621
621
622
622
623
623
624
624
625
625
626
626
627
627
628
628
629
629
630
630
631
631
632
632
633
633
634
634
635
635
636
636
637
637
638
638
639
639
640
640
641
641
642
642
643
643
644
644
645
645
646
646
647
647
648
648
649
649
650
650
651
651
652
652
653
653
654
654
655
655
656
656
657
657
658
658
659
659
660
660
661
661
662
662
663
663
664
664
665
665
666
666
667
667
668
668
669
669
670
670
671
671
672
672
673
673
674
674
675
675
676
676
677
677
678
678
679
679
680
680
681
681
682
682
683
683
684
684
685
685
686
686
687
687
688
688
689
689
690
690
691
691
692
692
693
693
694
694
695
695
696
696
697
697
698
698
699
699
700
700
701
701
702
702
703
703
704
704
705
705
706
706
707
707
708
708
709
709
710
710
711
711
712
712
713
713
714
714
715
715
716
716
717
717
718
718
719
719
720
720
721
721
722
722
723
723
724
724
725
725
726
726
727
727
728
728
729
729
730
730
731
731
732
732
733
733
734
734
735
735
736
736
737
737
738
738
739
739
740
740
741
741
742
742
743
743
744
744
745
745
746
746
747
747
748
748
749
749
750
750
751
751
752
752
753
753
754
754
755
755
756
756
757
757
758
758
759
759
760
760
761
761
762
762
763
763
764
764
765
765
766
766
767
767
768
768
769
769
770
770
771
771
772
772
773
773
774
774
775
775
776
776
777
777
778
778
779
779
780
780
781
781
782
782
783
783
784
784
785
785
786
786
787
787
788
788
789
789
790
790
791
791
792
792
793
793
794
794
795
795
796
796
797
797
798
798
799
799
800
800
801
801
802
802
803
803
804
804
805
805
806
806
807
807
808
808
809
809
810
810
811
811
812
812
813
813
814
814
815
815
816
816
817
817
818
818
819
819
820
820
821
821
822
822
823
823
824
824
825
825
826
826
827
827
828
828
829
829
830
830
831
831
832
832
833
833
834
834
835
835
836
836
837
837
838
838
839
839
840
840
841
841
842
842
843
843
844
844
845
845
846
846
847
847
848
848
849
849
850
850
851
851
852
852
853
853
854
854
855
855
856
856
857
857
858
858
859
859
860
860
861
861
862
862
863
863
864
864
865
865
866
866
867
867
868
868
869
869
870
870
871
871
872
872
873
873
874
874
875
875
876
876
877
877
878
878
879
879
880
880
881
881
882
882
883
883
884
884
885
885
886
886
887
887
888
888
889
889
890
890
891
891
892
892
893
893
894
894
895
895
896
896
897
897
898
898
899
899
900
900
901
901
902
902
903
903
904
904
905
905
906
906
907
907
908
908
909
909
910
910
911
911
912
912
913
913
914
914
915
915
916
916
917
917
918
918
919
919
920
920
921
921
922
922
923
923
924
924
925
925
926
926
927
927
928
928
929
929
930
930
931
931
932
932
933
933
934
934
935
935
936
936
937
937
938
938
939
939
940
940
941
941
942
942
943
943
944
944
945
945
946
946
947
947
948
948
949
949
950
950
951
951
952
952
953
953
954
954
955
955
956
956
957
957
958
958
959
959
960
960
961
961
962
962
963
963
964
964
965
965
966
966
967
967
968
968
969
969
970
970
971
971
972
972
973
973
974
974
975
975
976
976
977
977
978
978
979
979
980
980
981
981
982
982
983
983
984
984
985
985
986
986
987
987
988
988
989
989
990
990
991
991
992
992
993
993
994
994
995
995
996
996
997
997
998
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1010
1011
1011
1012
1012
1013
1013
1014
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1020
1021
1021
1022
1022
1023
1023
1024
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1030
1031
1031
1032
1032
1033
1033
1034
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1040
1041
1041
1042
1042
1043
1043
1044
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1050
1051
1051
1052
1052
1053
1053
1054
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1060
1061
1061
1062
1062
1063
1063
1064
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1070
1071
1071
1072
1072
1073
1073
1074
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1080
1081
1081
1082
1082
1083
1083
1084
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1090
1091
1091
1092
1092
1093
1093
1094
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1100
1101
1101
1102
1102
1103
1103
1104
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1110
1111
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1350
1351
1351
1352
1352
1353
1353
1354
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1360
1361
1361
1362
1362
1363
1363
1364
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1370
1371
1371
1372
1372
1373
1373
1374
1374
1375
1375
1376
1376
1377
1377
1378
1378
1379
1379
1380
1380
1381
1381
1382
1382
1383
1383
1384
1384
1385
1385
1386
1386
1387
1387
1388
1388
1389
1389
1390
1390
1391
1391
1392
1392
1393
1393
1394
1394
1395
1395
1396
1396
1397
1397
1398
1398
1399
1399
1400
1400
1401
1401
1402
1402
1403
1403
1404
1404
1405
1405
1406
1406
1407
1407
1408
1408
1409
1409
1410
1410
1411
1411
1412
1412
1413
1413
1414
1414
1415
1415
1416
1416
1417
1417
1418
1418
1419
1419
1420
1420
1421
1421
1422
1422
1423
1423
1424
1424
1425
1425
1426
1426
1427
1427
1428
1428
1429
1429
1430
1430
1431
1431
1432
1432
1433
1433
1434
1434
1435
1435
1436
1436
1437
1437
1438
1438
1439
1439
1440
1440
1441
1441
1442
1442
1443
1443
1444
1444
1445
1445
1446
1446
1447
1447
1448
1448
1449
1449
1450
1450
1451
1451
1452
1452
1453
1453
1454
1454
1455
1455
1456
1456
1457
1457
1458
1458
1459
1459
1460
1460
1461
1461
1462
1462
1463
1463
1464
1464
1465
1465
1466
1466
1467
1467
1468
1468
1469
1469
1470
1470
1471
1471
1472
1472
1473
1473
1474
1474
1475
1475
1476
1476
1477
1477
1478
1478
1479
1479
1480
1480
1481
1481
1482
1482
1483
1483
1484
1484
1485
1485
1486
1486
1487
1487
1488
1488
1489
1489
1490
1490
1491
1491
1492
1492
1493
1493
1494
1494
1495
1495
1496
1496
1497
1497
1498
1498
1499
1499
1500
1500
1501
1501
1502
1502
1503
1503
1504
1504
1505
1505
1506
1506
1507
1507
1508
1508
1509
1509
1510
1510
1511
1511
1512
1512
1513
1513
1514
1514
1515
1515
1516
1516
1517
1517
1518
1518
1519
1519
1520
1520
1521
1521
1522
1522
1523
1523
1524
1524
1525
1525
1526
1526
1527
1527
1528
1528
1529
1529
1530
1530
1531
1531
1532
1532
1533
1533
1534
1534
1535
1535
1536
1536
1537
1537
1538
1538
1539
1539
1540
1540
1541
1541
1542
1542
1543
1543
1544
1544
1545
1545
1546
1546
1547
1547
1548
1548
1549
1549
1550
1550
1551
15
```

```
#!/usr/bin/env python
# encoding: utf-8
"""
thermo.py

Various thermodynamic relationship for ice and water.

Created by Geoff Cureton <geoff.cureton@ssec.wisc.edu> on 2009-04-04.
Copyright 2009-2025 University of Wisconsin Regents. All rights reserved.

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>.

# from scipy import log10
from numpy import log10
from numpy import exp, log
from numpy import min, minimum, where

# -----
# Here we go. A set of functions that I use from time to time to calculate
# the basic stuff that I'm sick of doing over and over! I'm going to
# endeavour to include references and global constants to make it all nice
# and legible.
# -----


Rs_da = 287.05 # Specific gas const for dry air, J kg^{-1} K^{-1}
Rs_v = 461.51 # Specific gas const for water vapour, J kg^{-1} K^{-1}
Cp_da = 1004.6 # Specific heat at constant pressure for dry air
Cv_da = 719.0 # Specific heat at constant volume for dry air
Cp_v = 1870.0 # Specific heat at constant pressure for water vapour
Cv_v = 1410.0 # Specific heat at constant volume for water vapour
Cp_lw = 4218 # Specific heat at constant pressure for liquid water
Epsilon = 0.622 # Epsilon=Rs_da/Rs_v; The ratio of the gas constants
degCtoK = 273.15 # Temperature offset between K and C (deg C)
rho_w = 1000.0 # Liquid Water density kg m^{-3}
grav = 9.81 # Gravity, m s^{-2}
Lv = 2.56e+05 # Latent Heat of vaporisation
boltzmann = 5.67e-8 # Stefan-Boltzmann constant
mv = 18.0153 # Mean molar mass of water vapor(g/mol)
```

<v/git/cspp_sounder_ql-worktrees/legacy
leo

NORMAL ➤ legacy < * scripts/sounder_image.py ^Wh < 23 25% 49:9 19:35 thermo.py
0: Home 1: Root 2: package 3: Repo* 4: Work- 5: pkg test 6: python 7: SampleData

9:18
2026-01-08 19:35:06

Tiling Window Managers

- Outside of my text editor (NeoVim), I try to manage my various desktop tasks in the same way as my editor: using the keyboard.
- The long-standing mode for app management is app-per-window, so how do I manage my windows?
- My preferred method is to use a tiling window manager, which manages where and how windows are drawn on the screen, so that the windows don't overlap.
- This allows the user to efficiently move windows around using screen, and between virtual desktops, using the keyboard.
- My preferred TWM is Qtile, written in python (<https://pawamoy.github.io/qtile/>)



 ZSA



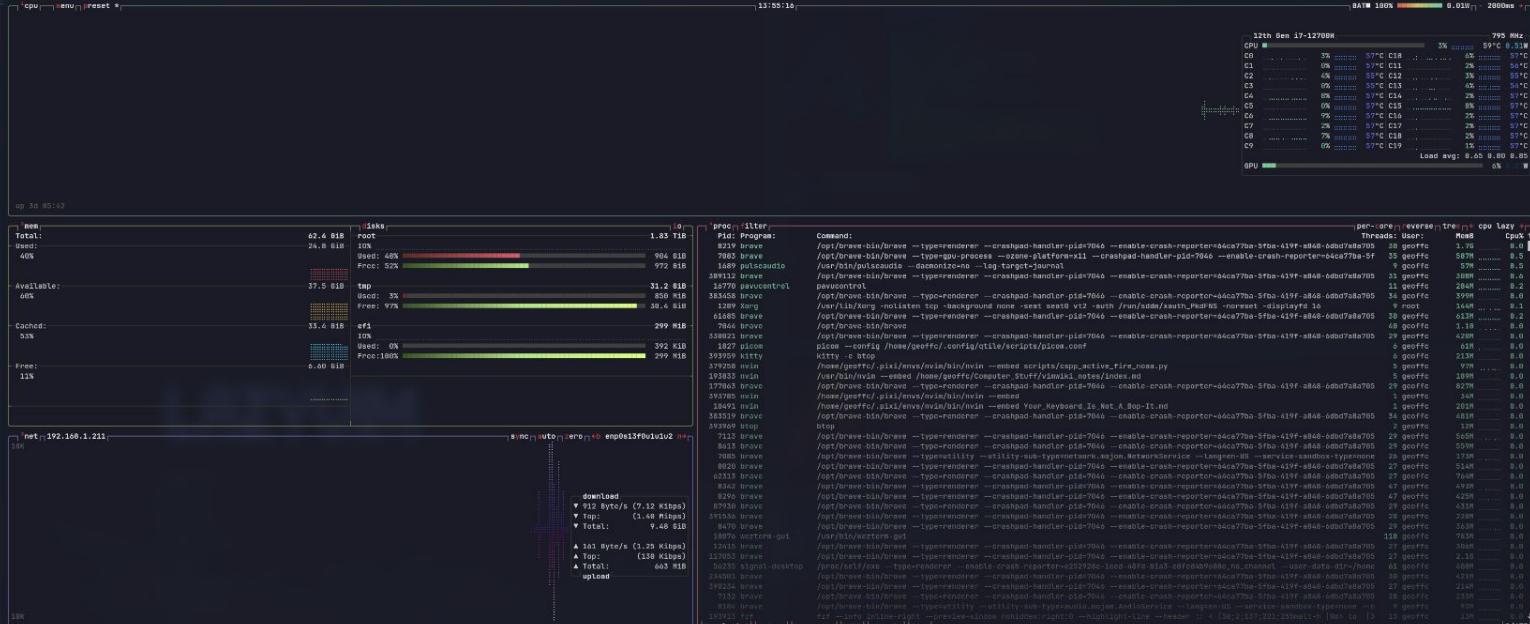
 ZSA

[13:51:23] [geoffc@subrahmanyam ~ ()]

LAZYUM



[13:51:23] [geoffc@subrahmanyam]



h

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q

q



Thank you!