



What I would like to use submit for

H. Moritz Günther
Kavli Institute for Astrophysics

Different computer uses in astronomy

A yellow pencil and a pink eraser are positioned in the top right corner of the slide, appearing to be on a white piece of paper against a blue grid background.

- Hardcore simulations
- Data pipelines, e.g. process all stars in the Chandra archive
- Reduce, model, and fit a single observation
- Write papers, slack, read the news

Different computer uses in astronomy



- Hardcore simulations
 - Think 1000's or millions of CPUs, GPUs, TPUs
 - Clusters, MPI, all the good stuff

**Not me
(at least currently)**

- Data pipelines, e.g. process all stars in the Chandra archive
- Reduce, model, and fit a single observation
- Write papers, slack, read the news

Different computer uses in astronomy



- Hardcore simulations
- Data pipelines, e.g. process all stars in the Chandra archive
 - Often needs observatory specific software (today: conda packages, Python, docker), that can be shared between users
 - Custom Python code
 - Develop and experiment interactively, then batch job
 - Not time critical
- Reduce, model, and fit a single observation
- Write papers, slack, read the news

Different computer uses in astronomy

- Hardcore simulations
- Data pipelines, e.g. process all stars in the Chandra archive
- Reduce, model, and fit a single observation
 - I want that fast and I want that now.
 - Interactive work, most projects can be done on laptop
 - In my case: Python, Jupyter notebooks
 - Software via conda, pip, custom code
- Write papers, slack, read the news



```
for cp, p in zip(csplots, parts):
    cp.prepare(dpha, rsp(model.lhs * p))

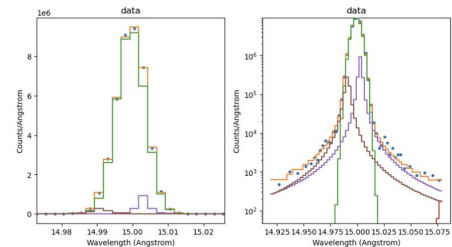
fig, axes = plt.subplots(ncols=2, figsize=(10, 5))

for i, ax in enumerate(axes):
    plt.sca(ax)
    dplot.plot(clearwindow=False, ylog=bool(i))
    mhplot.plot(overplot=True)
    for cp in csplots:
        cp.overplot()

_ = axes[0].set_xlim(evt[colname].mean() - 5 * evt[colname].st
                    evt[colname].mean() + 5 * evt[colname].st)
_ = axes[1].set_ylim(dplot.y.min()/10, dplot.y.max())
```

-8]

Python



-2]

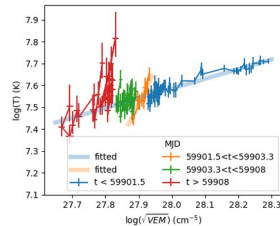
[dplot.y](#)

Python

```
array([4.66666708e+02, 1.00000009e+03, 9.00000080e+02, 1.40001
       1.60000014e+03, 2.40000021e+03, 2.00000018e+03, 3.60000
```

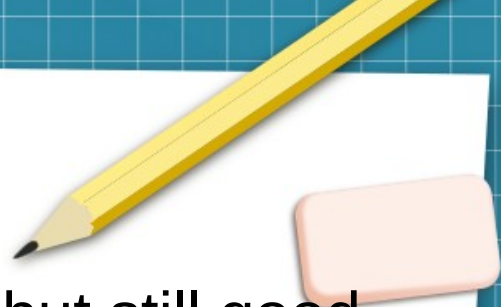

Different computer uses in astronomy

- Hardcore simulations
- Data pipelines, e.g. process all stars in the Chandra archive
- Reduce, model, and fit a single observation
- Write papers, slack, read the news
 - Laptop, not subMIT
 - But need to move data/plots from subMIT to local machine

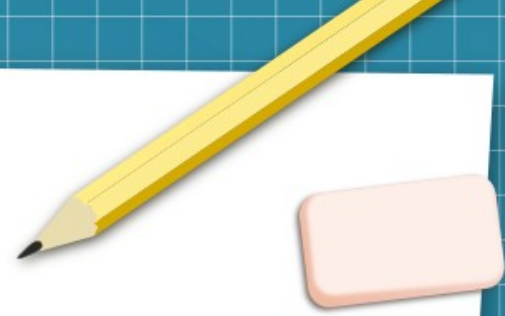


Do I still need a desktop?

- Desktop under my desk in NE-83 is 10 years old, but still good (64 GB RAM, 8 TB disk space)
- When I started at MKI, I did everything on that machine
- Now, laptop is sufficient for most tasks
- Desktop is idle > 98 % of the time.
- Cost > 3000 \$ in 2014, but worked for >10 years



subMIT



- Submit is great
 - even though I only need it a few times per year
 - I need it when I need it.
- Will not replace office desktop PC, that's unused > 98 % of the time.



Replace desktop with subMIT I



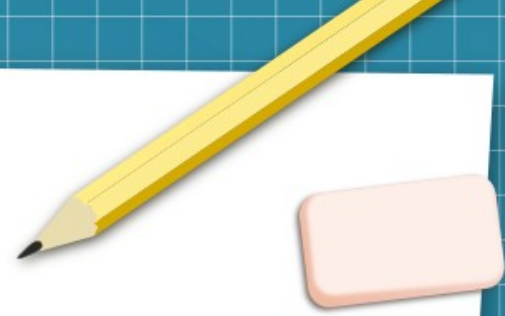
- Need ~20 GB for observatory software, calibration data. Can be shared in our group, maintained by one of us so we don't all have to keep track of when to upgrade.
- Desktop provides permanent storage, backed-up through CrashPlan. I use ~ 1-2 TB with observational data, simulations from projects years ago, etc. Could trim that, but sometimes it's good to "have it all".
- Need to keep for reproducible science, even if I look at it rarely.
 - How would I do that if I used subMIT exclusively?

Replace desktop with subMIT II



- Often work on many projects, leave on dormant for weeks or months, come back to that
 - But don't want to spend time to transfer data and code into and out of non-permanent storage.
 - The old model of a hard drive with folders works well for that.
- Software: Easy. Python...
- Moving data: scp.
 - Is there an easier way? Dropbox? Networked folders?

Summary



- Today, I do 95% of my work on my laptop.
- I'd move the remaining 5% from a desktop to subMIT.
- Software, interface, etc.
- But how to do long-term, backed-up data storage?



The slide design licensed under a Creative Commons Attribution-ShareAlike 3.0 Unported License.
It makes use of the works of Mateus Machado Luna.

