Why HDFhub?

JupyterHub for HDF

Why Jupyter Notebooks?

- Step by step notebook for a user to understand and replicate
- Switch between Python, BASH, etc in one GUI, sharing variables between languages along with other "magics"
- Opportunity to inject markdown cells to explain in plain English what is happening in each chunk of code.
- Images, video are embeddable, so you could have a short video that explains a section of code or the slides for a presentation

All of this is really cool, but…

- A notebook is dependent on a Notebook or Lab server.
- The application is usually run locally, and on multiple operating systems, so syntax differences exist
- This results in a situation where the notebook might not work in the environment that someone else is using.

A notebook is not a notebook

- Homegrown
  - Specific focus to environment
  - unsupported
  - contributable
- Open Source
  - most commonly used packages
  - supported
  - contributable
- Enterprise
  - most comprehensive
  - Supported
  - Contributable

Why JupyterHub?

"But the plans were on display . . ."
"On display? I eventually had to go down to the cellar to find them."
"That's the display department."
"With a torch."
"Ah, well the lights had probably gone."
"So had the stairs."
"But look, you found the notice, didn't you?"
"Yes," said Arthur, "yes I did. It was on display in the bottom of a locked filing cabinet stuck in a disused lavatory with a sign on the door saying Beware of the Leopard."
—Douglas Adams, *The Hitchhiker's Guide to the Galaxy*

A Notebook is not enough

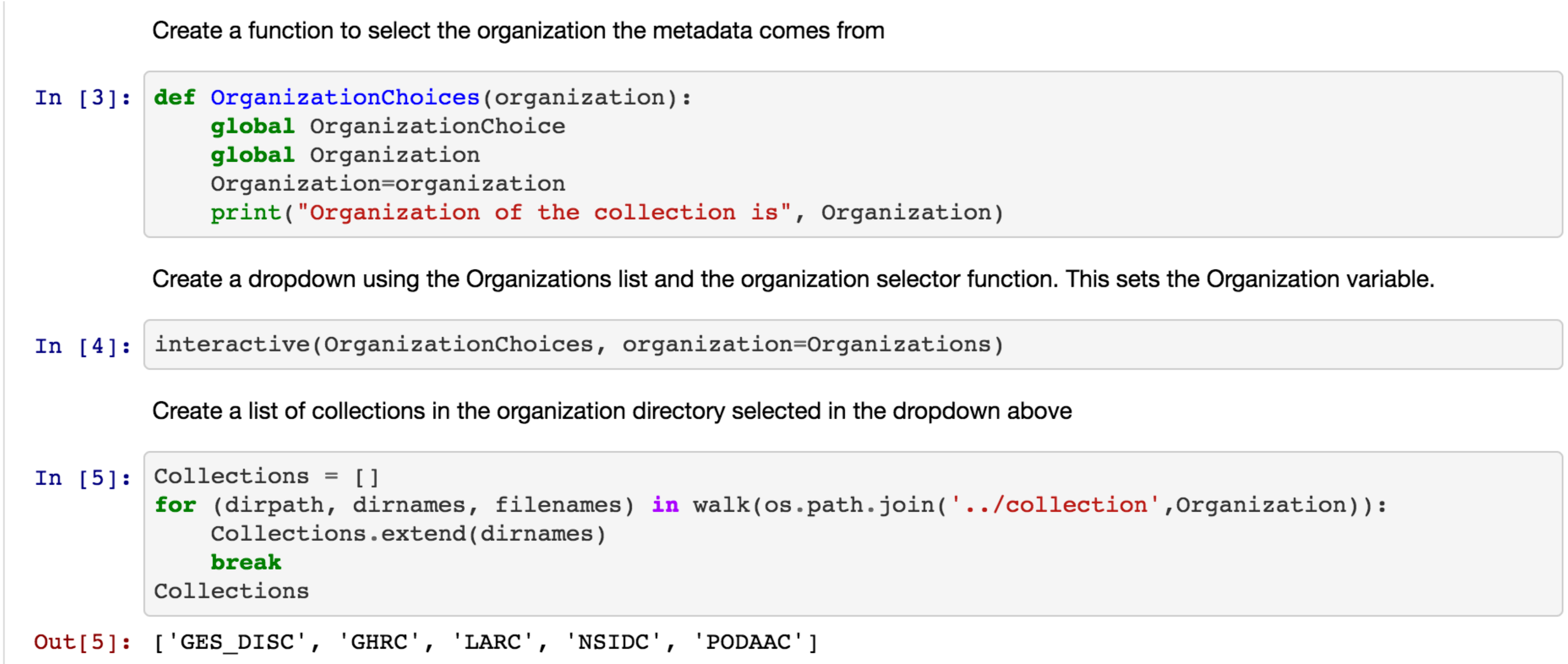
- The addition of JupyterHub allows us to control the user’s environment and data
- This results in redoability not just reproducibility

Science is show me not trust me. - Philip Stark

Other Options to improve notebooks

nbviewer is integrated with Github

- Renders a static version of the last run, committed state of a notebook in the branch you are looking at
- Doesn't allow interactivity
- Doesn't share the environment



Create a function to select the organization the metadata comes from

```
In [3]: def OrganizationChoices(organization):
            global OrganizationChoice
            global Organization
            Organization=organization
            print("Organization of the collection is", Organization)
```

Create a dropdown using the Organizations list and the organization selector function. This sets the Organization variable.

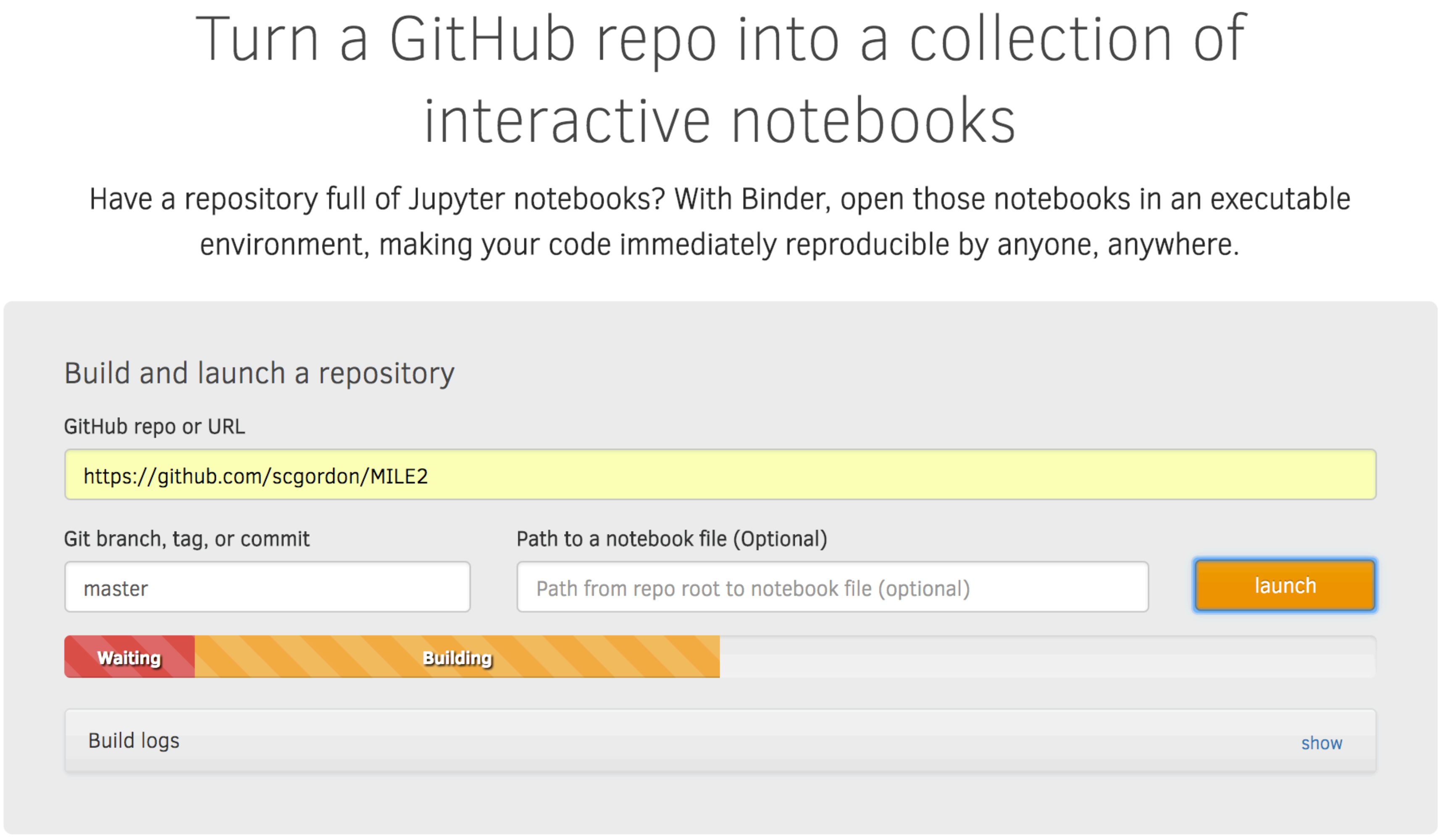
```
In [4]: interactive(OrganizationChoices, organization=Organizations)
```

Create a list of collections in the organization directory selected in the dropdown above

```
In [5]: Collections = []
        for (dirpath, dirnames, filenames) in walk(os.path.join('../collection',Organization)):
            Collections.extend(dirnames)
            break
        Collections

Out[5]: ['GES_DISC', 'GHRC', 'LARC', 'NSIDC', 'PODAAC']
```

Projects like Binder are promising



Turn a GitHub repo into a collection of interactive notebooks

Have a repository full of Jupyter notebooks? With Binder, open those notebooks in an executable environment, making your code immediately reproducible by anyone, anywhere.

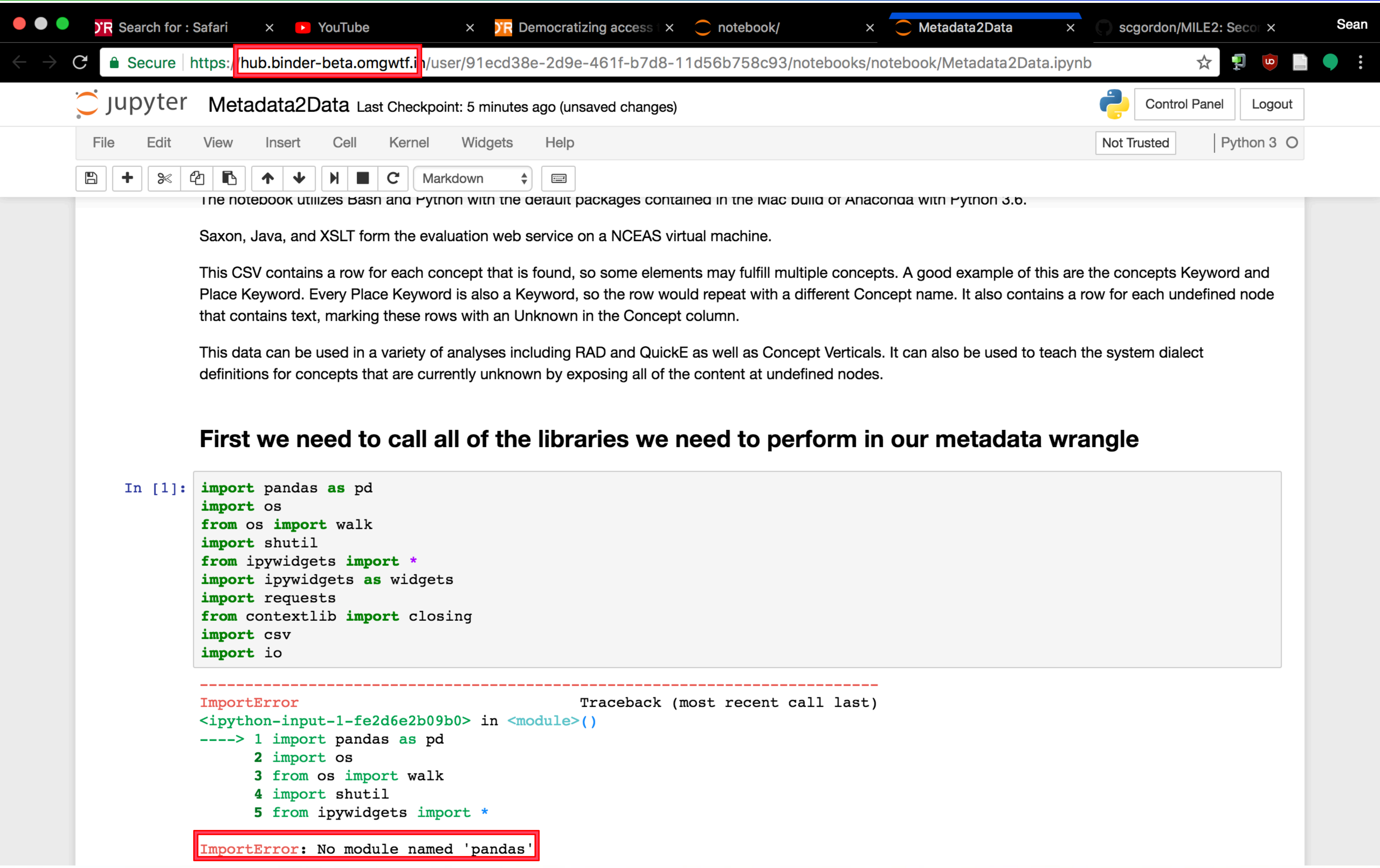
Build and launch a repository

GitHub repo or URL: https://github.com/scgordon/MILE2

Git branch, tag, or commit: master

Path to a notebook file (Optional): Path from repo root to notebook file (optional)

But it is beta…



Secure https://hub.binder-beta.omgwtf.in/user/91ecd38e-2d9e-461f-b7d8-11d56b758c93/notebooks/notebook/Metadata2Data.ipynb

jupyter Metadata2Data Last Checkpoint: 5 minutes ago (unsaved changes)

The notebook utilizes Bash and Python with the default packages contained in the Mac build of Anaconda with Python 3.6.

Saxon, Java, and XSLT form the evaluation web service on a NCEAS virtual machine.

This CSV contains a row for each concept that is found, so some elements may fulfill multiple concepts. A good example of this are the concepts Keyword and Place Keyword. Every Place Keyword is also a Keyword, so the row would repeat with a different Concept name. It also contains a row for each undefined node that contains text, marking these rows with an Unknown in the Concept column.

This data can be used in a variety of analyses including RAD and QuickE as well as Concept Verticals. It can also be used to teach the system dialect definitions for concepts that are currently unknown by exposing all of the content at undefined nodes.

First we need to call all of the libraries we need to perform in our metadata wrangle

```
In [1]: import pandas as pd
        import os
        from os import walk
        import shutil
        from ipywidgets import *
        import ipywidgets as widgets
        import requests
        from contextlib import closing
        import csv
        import io
```

```
---------------------------------------------------------------------------
ImportError                               Traceback (most recent call last)
<ipython-input-1-fe2d6e2b09b0> in <module>()
----> 1 import pandas as pd
      2 import os
      3 from os import walk
      4 import shutil
      5 from ipywidgets import *

ImportError: No module named 'pandas'
```

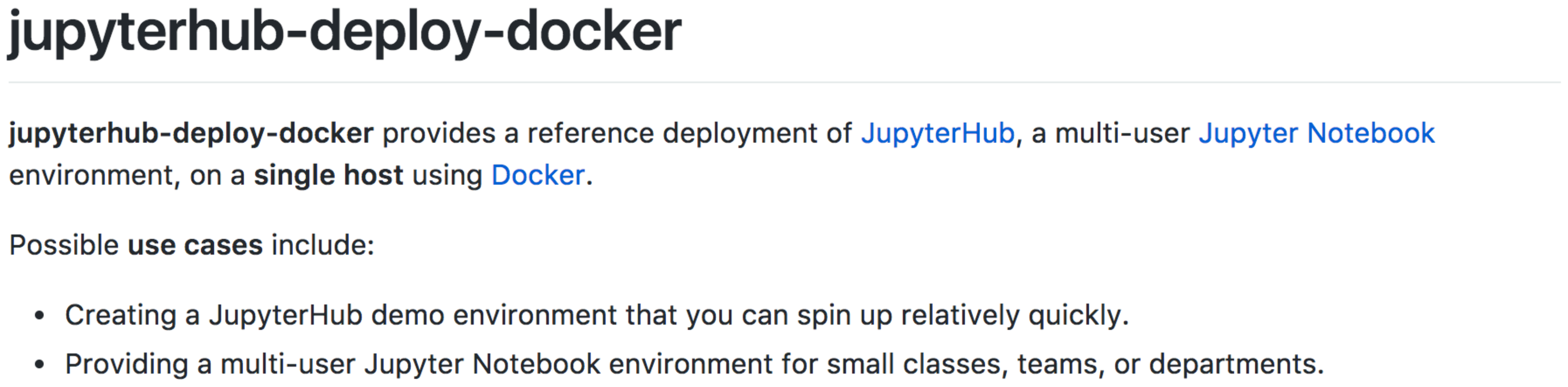
Building HDFhub

Build a JupyterHub

You can build your own from scratch

- Can change the authenticator to the website's
- Can choose spawner
- Can build the environment
- Can use Notebook or Lab for the GUI

Or just grab a docker container and get started



jupyterhub-deploy-docker

jupyterhub-deploy-docker provides a reference deployment of JupyterHub, a multi-user Jupyter Notebook environment, on a **single host** using Docker.

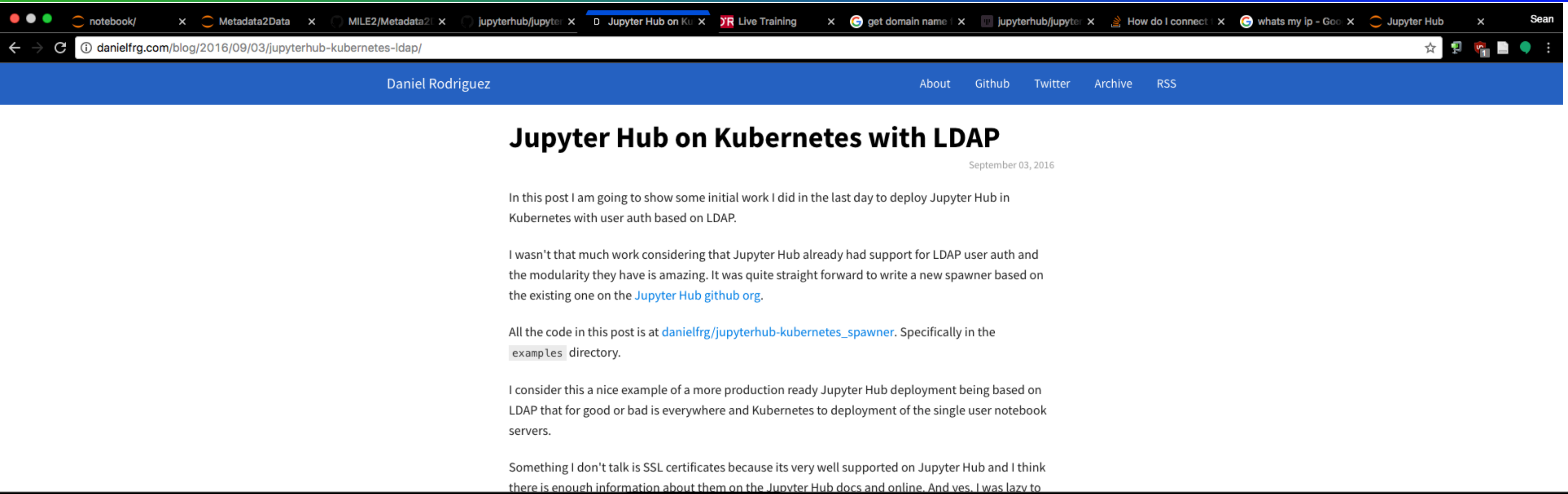
Possible **use cases** include:

- Creating a JupyterHub demo environment that you can spin up relatively quickly.
- Providing a multi-user Jupyter Notebook environment for small classes, teams, or departments.

Kubernetes

- Yuvi Panda: Jupyter Hub can be made safer using Kubernetes bc Kubernetes has a bunch of ways to limit impact. No root, remove SMTP access, limit number of sockets. Google or other cloud handles persistence of data if desired
- Can create multiple JupyterHubs with different environments
- Treats the vm like cattle not pets
- Likely to see the most active development and optimization as Yuvi is a core developer for kubespawner as well as Jupyter and Data8

Kubernetes with Authentication and Storage



Jupyter Hub on Kubernetes with LDAP

September 03, 2016

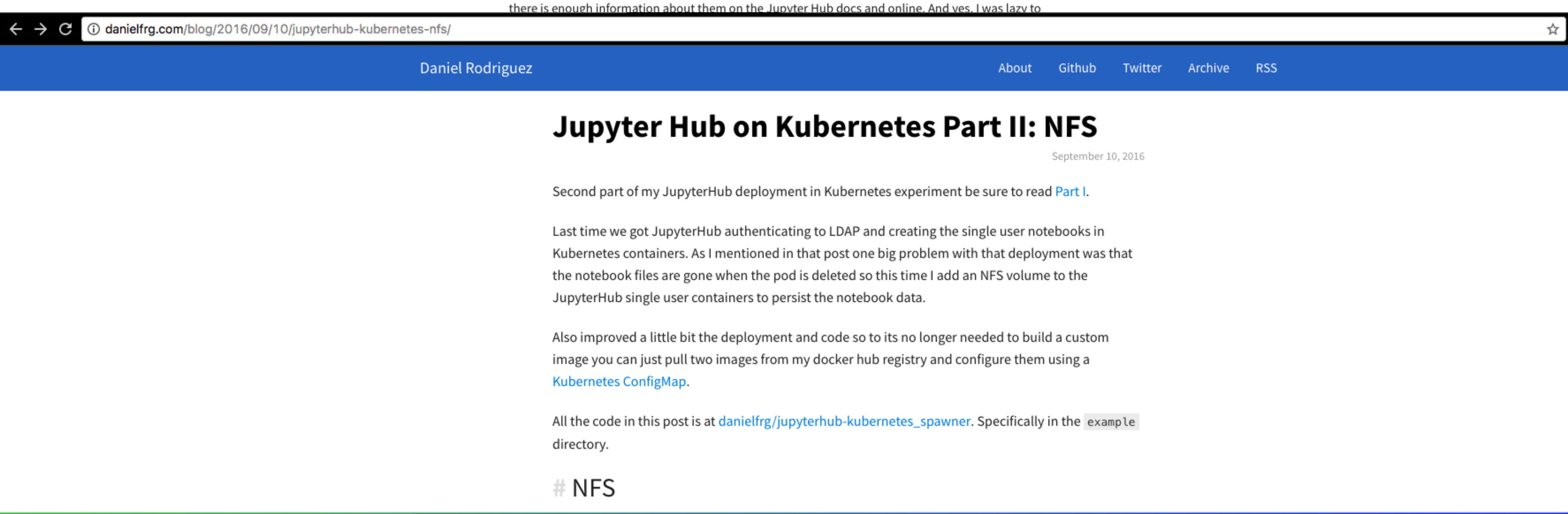
In this post I am going to show some initial work I did in the last day to deploy Jupyter Hub in Kubernetes with user auth based on LDAP.

I wasn't that much work considering that Jupyter Hub already had support for LDAP user auth and the modularity they have is amazing. It was quite straight forward to write a new spawner based on the existing one on the Jupyter Hub github org.

All the code in this post is at danielfrg/jupyterhub-kubernetes_spawner. Specifically in the `examples` directory.

I consider this a nice example of a more production ready Jupyter Hub deployment being based on LDAP that for good or bad is everywhere and Kubernetes to deployment of the single user notebook servers.

Something I don't talk is SSL certificates because its very well supported on Jupyter Hub and I think there is enough information about them on the Jupyter Hub docs and online. And yes, I was lazy to



Jupyter Hub on Kubernetes Part II: NFS

September 10, 2016

Second part of my JupyterHub deployment in Kubernetes experiment be sure to read Part I.

Last time we got JupyterHub authenticating to LDAP and creating the single user notebooks in Kubernetes containers. As I mentioned in that post one big problem with that deployment was that the notebook files are gone when the pod is deleted so this time I add an NFS volume to the JupyterHub single user containers to persist the notebook data.

Also improved a little bit the deployment and code so to its no longer needed to build a custom image you can just pull two images from my docker hub registry and configure them using a Kubernetes ConfigMap.

All the code in this post is at danielfrg/jupyterhub-kubernetes_spawner. Specifically in the `example` directory.

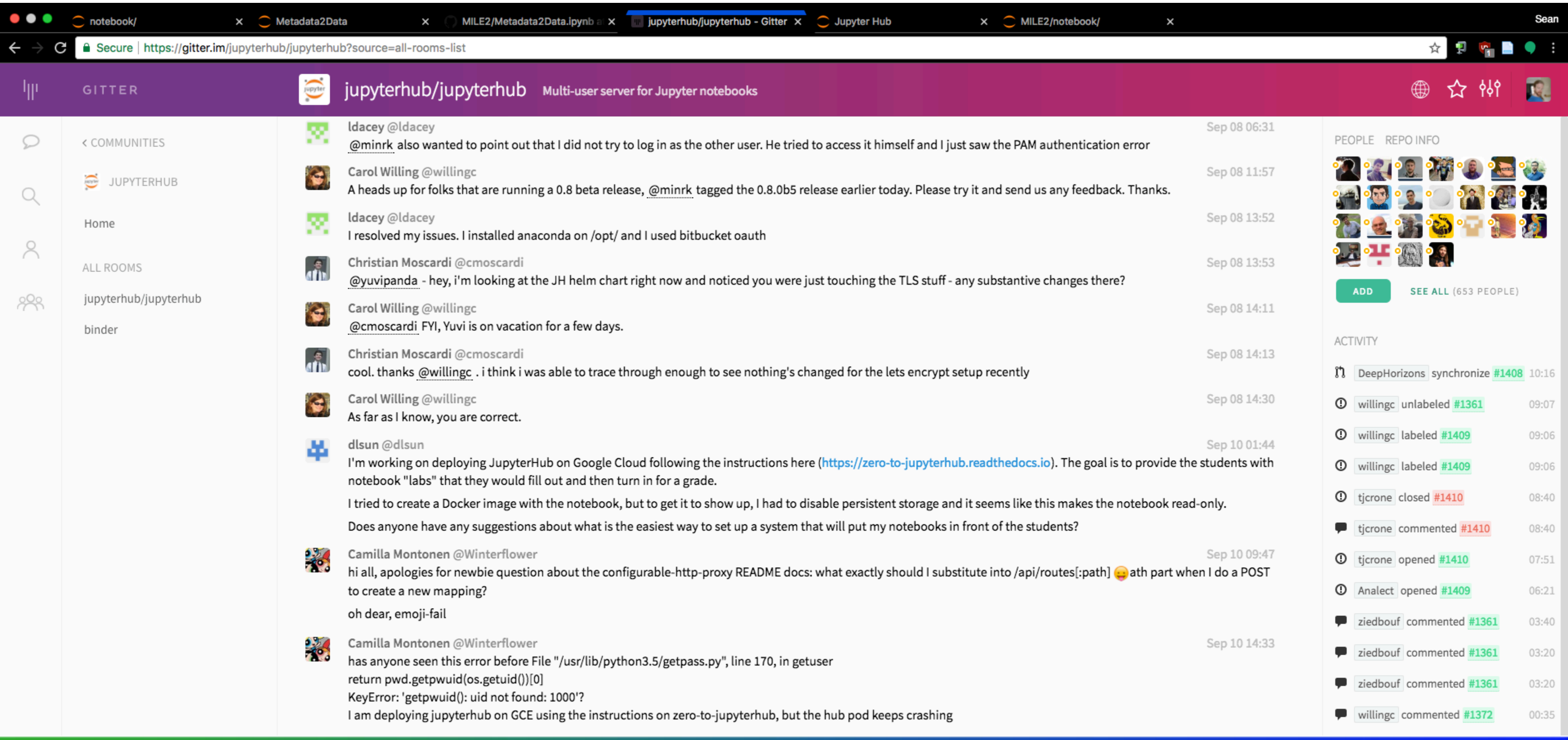
NFS

VM based tutorial got me a lot further

http://localhost:8080/user/vagrant/tree/MILE2

Gitter for all the technical questions

- The majority of Jupyter developers are active in this Gitter Channel: https://gitter.im/jupyterhub/jupyterhub



https://gitter.im/jupyterhub/jupyterhub?source=all-rooms-list

jupyterhub/jupyterhub Multi-user server for Jupyter notebooks

< COMMUNITIES
JUPYTERHUB
Home
ALL ROOMS
jupyterhub/jupyterhub
binder

ldacey @ldacey Sep 08 06:31
@minrk also wanted to point out that I did not try to log in as the other user. He tried to access it himself and I just saw the PAM authentication error

Carol Willing @willingc Sep 08 11:57
A heads up for folks that are running a 0.8 beta release, @minrk tagged the 0.8.0b5 release earlier today. Please try it and send us any feedback. Thanks.

ldacey @ldacey Sep 08 13:52
I resolved my issues. I installed anaconda on /opt/ and I used bitbucket oauth

Christian Moscardi @cmoscardi Sep 08 13:53
@yuvipanda - hey, i'm looking at the JH helm chart right now and noticed you were just touching the TLS stuff - any substantive changes there?

Carol Willing @willingc Sep 08 14:11
@cmoscardi FYI, Yuvi is on vacation for a few days.

Christian Moscardi @cmoscardi Sep 08 14:13
cool. thanks @willingc . i think i was able to trace through enough to see nothing's changed for the lets encrypt setup recently

Carol Willing @willingc Sep 08 14:30
As far as I know, you are correct.

dlsun @dlsun Sep 10 01:44
I'm working on deploying JupyterHub on Google Cloud following the instructions here (https://zero-to-jupyterhub.readthedocs.io). The goal is to provide the students with notebook "labs" that they would fill out and then turn in for a grade.
I tried to create a Docker image with the notebook, but to get it to show up, I had to disable persistent storage and it seems like this makes the notebook read-only.
Does anyone have any suggestions about what is the easiest way to set up a system that will put my notebooks in front of the students?

Camilla Montonen @Winterflower Sep 10 09:47
hi all, apologies for newbie question about the configurable-http-proxy README docs: what exactly should I substitute into /api/routes[:path]😛ath part when I do a POST to create a new mapping?
oh dear, emoji-fail

Camilla Montonen @Winterflower Sep 10 14:33
has anyone seen this error before File "/usr/lib/python3.5/getpass.py", line 170, in getuser
return pwd.getpwuid(os.getuid())[0]
KeyError: 'getpwuid(): uid not found: 1000'?
I am deploying jupyterhub on GCE using the instructions on zero-to-jupyterhub, but the hub pod keeps crashing

PEOPLE REPO INFO
ADD SEE ALL (653 PEOPLE)

ACTIVITY

DeepHorizons	synchronize #1408	10:16
willingc	unlabeled #1361	09:07
willingc	labeled #1409	09:06
willingc	labeled #1409	09:06
tjcrone	closed #1410	08:40
tjcrone	commented #1410	08:40
tjcrone	opened #1410	07:51
Analect	opened #1409	06:21
ziedbouf	commented #1361	03:40
ziedbouf	commented #1361	03:20
ziedbouf	commented #1361	03:20
willingc	commented #1372	00:35

Examples of JupyterHub in the wild

- Data8 http://data8.org/

https://github.com/data-8/jupyterhub-deploy

- Harvard https://github.com/harvard/cloudJHub
- Quantopian Research Platform https://libraries.io/github/quantopian/jupyterhub
- Lawrence Berkeley Labs https://goo.gl/GBQ5gS
- Enterprise solutions Anaconda, Immuta, DataScience offering free trials
- O'Reilly using Hub and Notebooks for webinars now

Why HDFhub for website engagement and beyond

Hub as community engagement

- Interactive with a lower barrier to entry

- People can easily and quickly build on each other's ideas
- This makes HDF more accessible in the perception of our user base as a format and support

- Likely to stimulate conversation on the forum since they are also on the website

- Identify development opportunities to expand HDF capabilities and tools by interacting with current and future users

No more workshop software carpentry

- Allows us to focus on the why immediately
- Only a connected web browser is needed
- Provides an understood GUI with powerful documentation capabilities to drive interactions with products like HSDS or the metadata evaluator
- Gives tangible results to users
- Easily tweakable to particular use case

Larger set of potential presenters

- This means we can reach out on more fronts at any conference/workshop/etc sending fewer participants.
- People more directly involved in identifying potential revenue can show all off our work easily rather than just tell about it with potential customers.
- Allows us all to play to our strengths rather than trying to fill all roles.

Where do we go now?

- Jupyter Notebooks are necessary but not sufficient
- The right environment is needed as well
- What we could do
  - Use the free Binder beta with all it's warts while it's free and hope it works when it's needed
  - Get our own Binder instance running so developers can keep their own work and requirements document up to date
  - Create a JupyterHub that gives each website user and employee persistent a notebook/lab server to attract website users and identify needs and trends while engaging with a broader audience
  - Create a JupyterHub that gives each website user the ability to read and execute featured Notebooks in a Notebook/Lab environment but no persistence and create persistent cloud environments for employees

Binder

- This presentation has been hosted on Binder.

- It’s part of the repository I used to give people a user interface to our metadata evaluation for improvement at the ESIP Summer meeting this year