Integrating AGC pipeline at BNL facility

Matthew Feickert
(University of Wisconsin-Madison)
matthew.feickert@cern.ch

IRIS-HEP / AGC Demo Day December 2022
December 16th, 2022

American Family Insurance Data Science Institute
UNIVERSITY OF WISCONSIN–MADISON

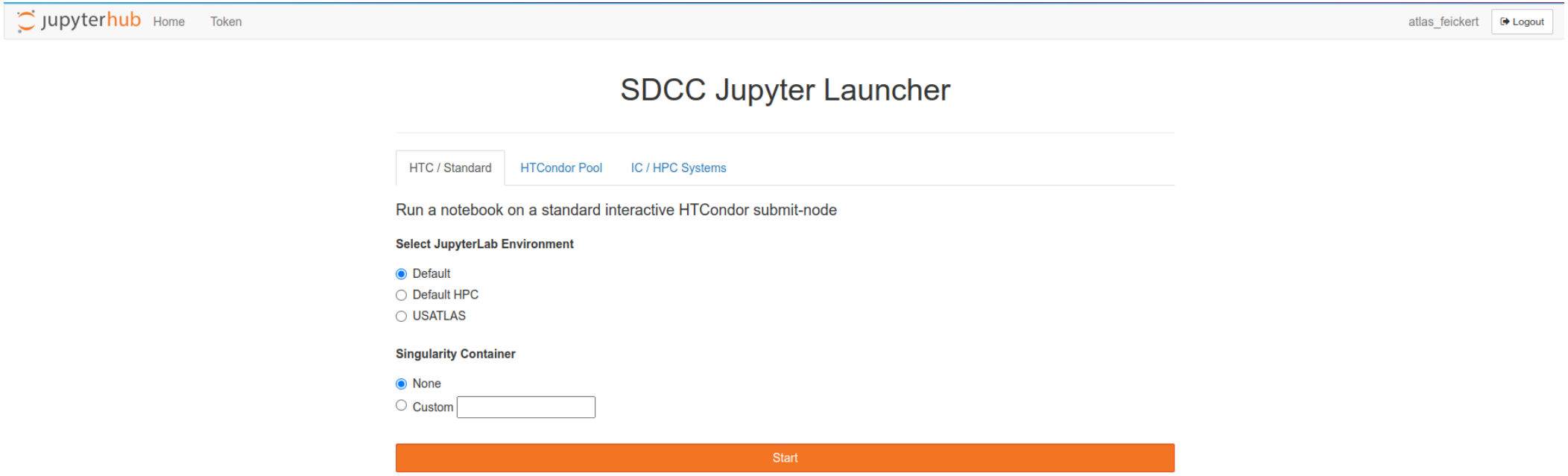
iris hep

ATLAS EXPERIMENT

Introduction and Overview

- This work is really all thanks to BNL team
  - Doug Benjamin
  - Ofer Rind
  - Chris Hollowell
- Ongoing process, but today just showing the first (unoptimized) steps that came together pleasantly quickly
- Today showing examples of running at BNL's SDCC Jupyter instance

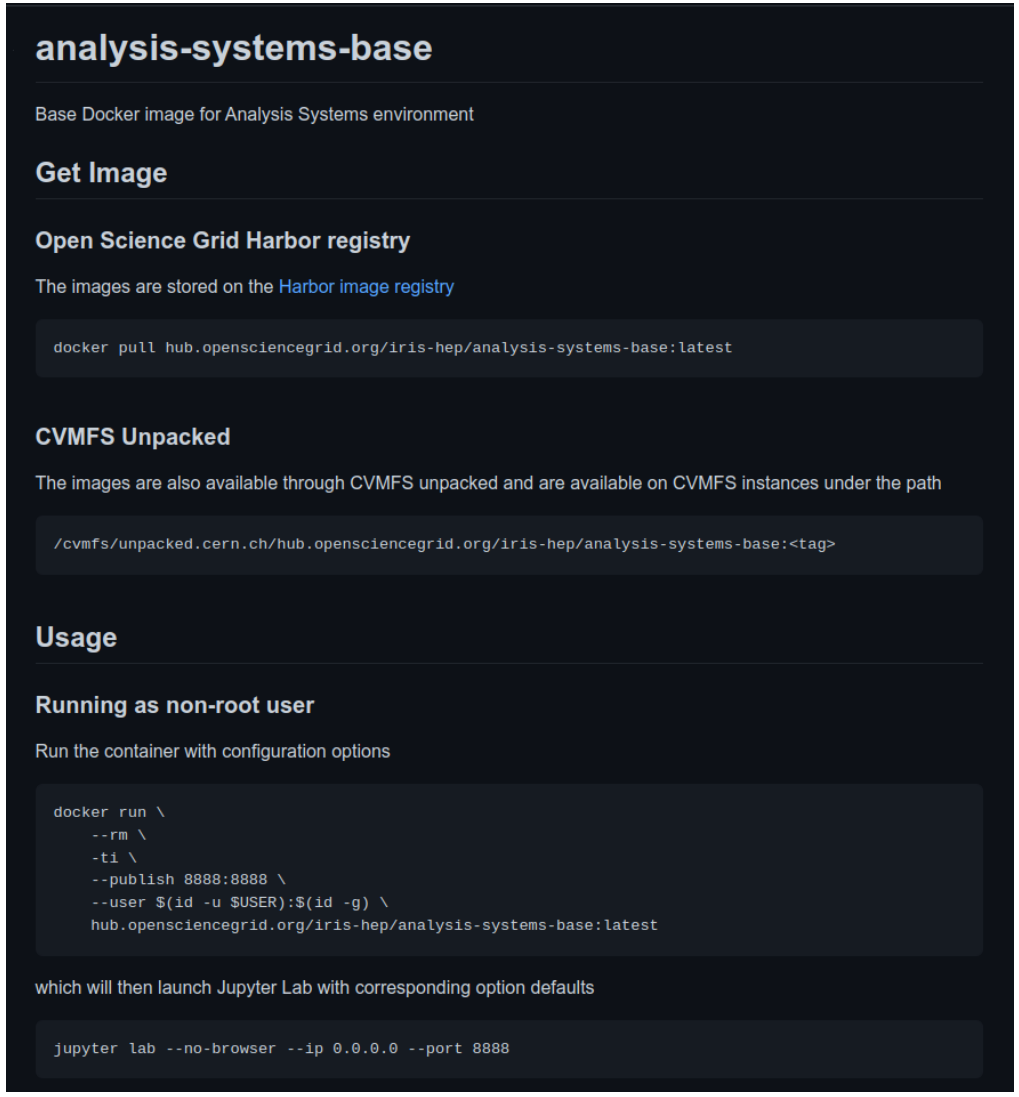
BNL SDCC Jupyter Launcher: custom images!



Allows for running custom images as Singularity containers
Can pull from public image registries or from CVMFS unpacked

Development image: analysis-systems-base

- As we can use custom images at BNL created the `analysis-systems-base` image (https://github.com/iris-hep/analysis-systems-base)
- Images are hosted on OSG Harbor under `iris-hep` org
  - `hub.opensciencegrid.org/iris-hep/analysis-systems-base`
  - Thanks Brian Lin for making this happen!



analysis-systems-base

Base Docker image for Analysis Systems environment

Get Image

Open Science Grid Harbor registry

The images are stored on the Harbor image registry

```
docker pull hub.opensciencegrid.org/iris-hep/analysis-systems-base:latest
```

CVMFS Unpacked

The images are also available through CVMFS unpacked and are available on CVMFS instances under the path

```
/cvmfs/unpacked.cern.ch/hub.opensciencegrid.org/iris-hep/analysis-systems-base:<tag>
```

Usage

Running as non-root user

Run the container with configuration options

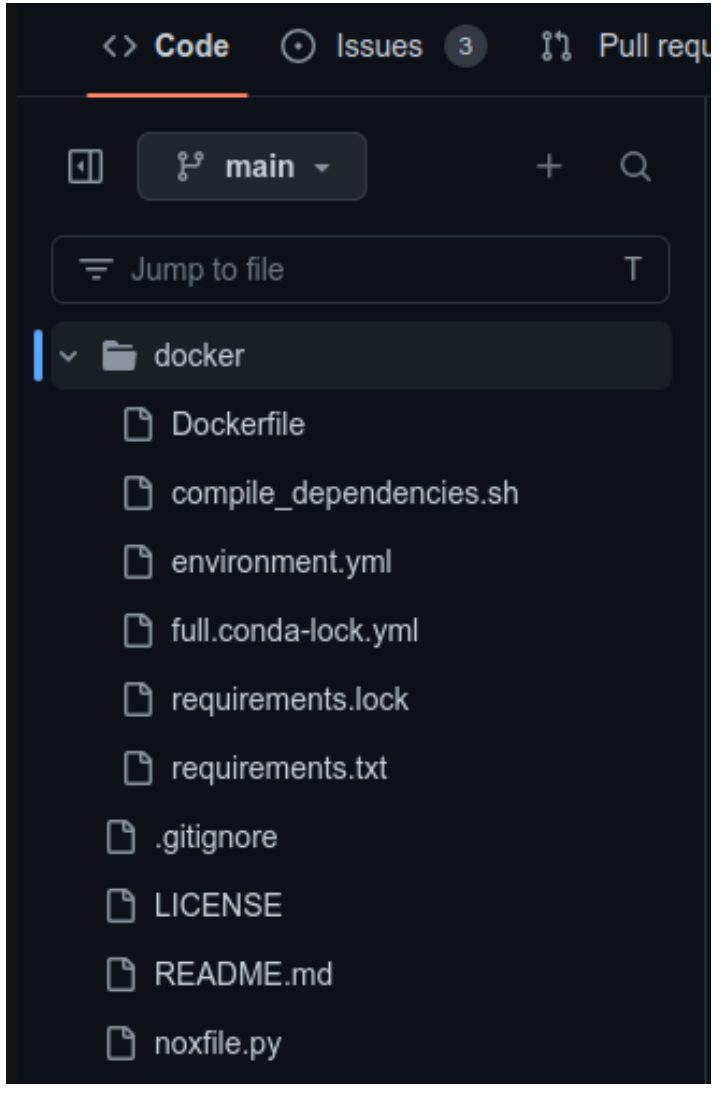
```
docker run \
    --rm \
    -ti \
    --publish 8888:8888 \
    --user $(id -u $USER):$(id -g) \
    hub.opensciencegrid.org/iris-hep/analysis-systems-base:latest
```

which will then launch Jupyter Lab with corresponding option defaults

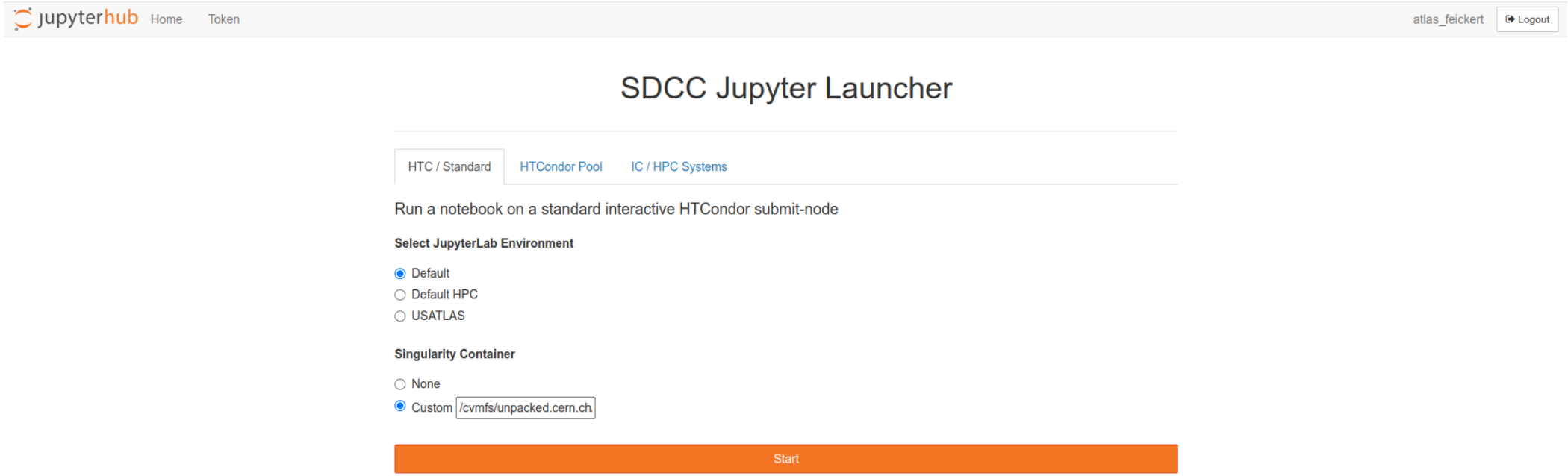
```
jupyter lab --no-browser --ip 0.0.0.0 --port 8888
```

Development image: analysis-systems-base

- As we can use custom images at BNL created the `analysis-systems-base` image (https://github.com/iris-hep/analysis-systems-base)
- Goal is to use lock files to make as much of the image as fully reproducible as possible and statically defined.



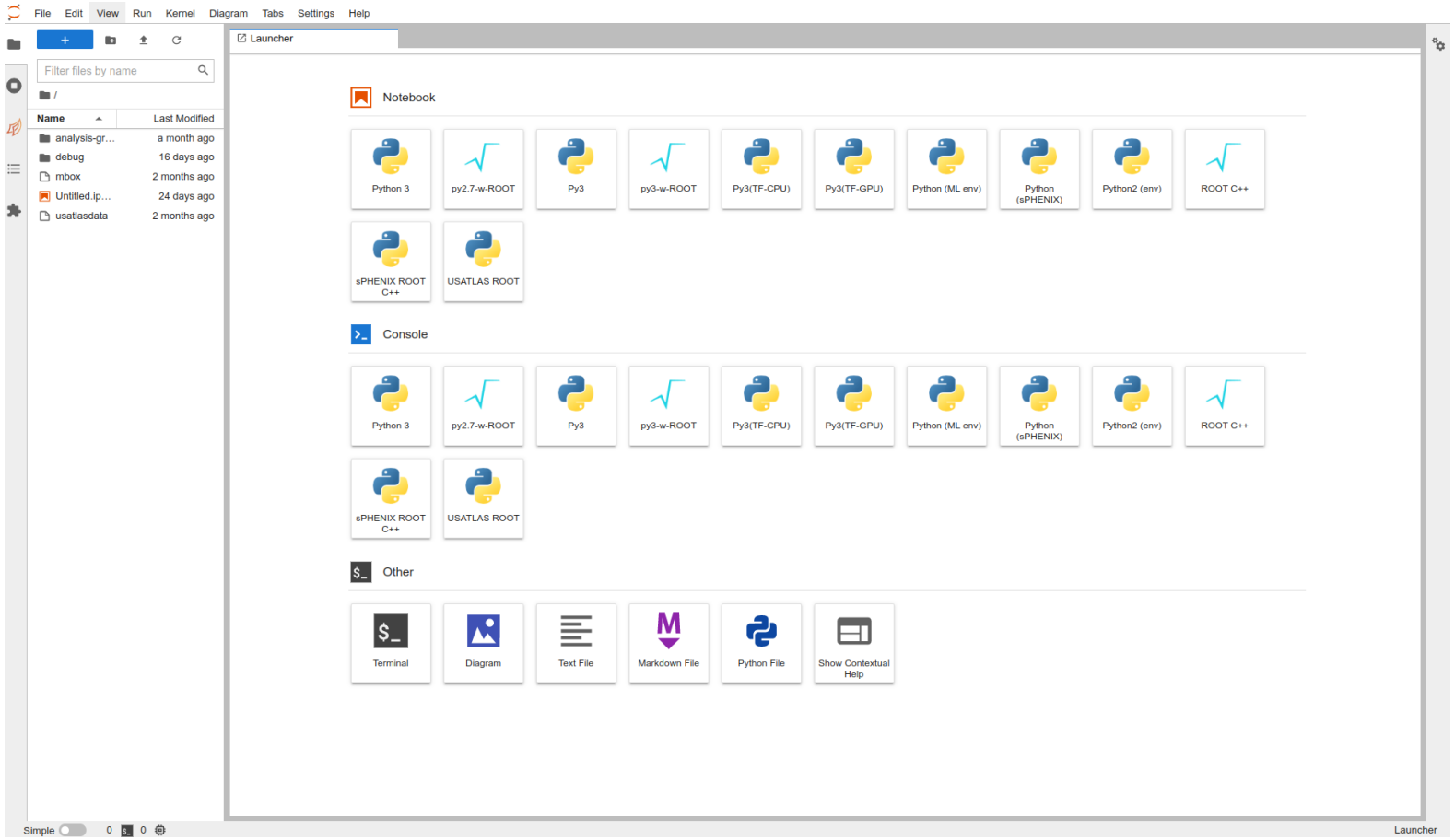
BNL SDCC Jupyter Launcher: custom images!



Today's demo:

`/cvmfs/unpacked.cern.ch/hub.opensciencegrid.org/iris-hep/analysis-systems-base:2022-12-15`

WIP: Custom kernel discovery



At the moment, lacking mounting (or something else) to mount custom kernels from user defined virtual environments

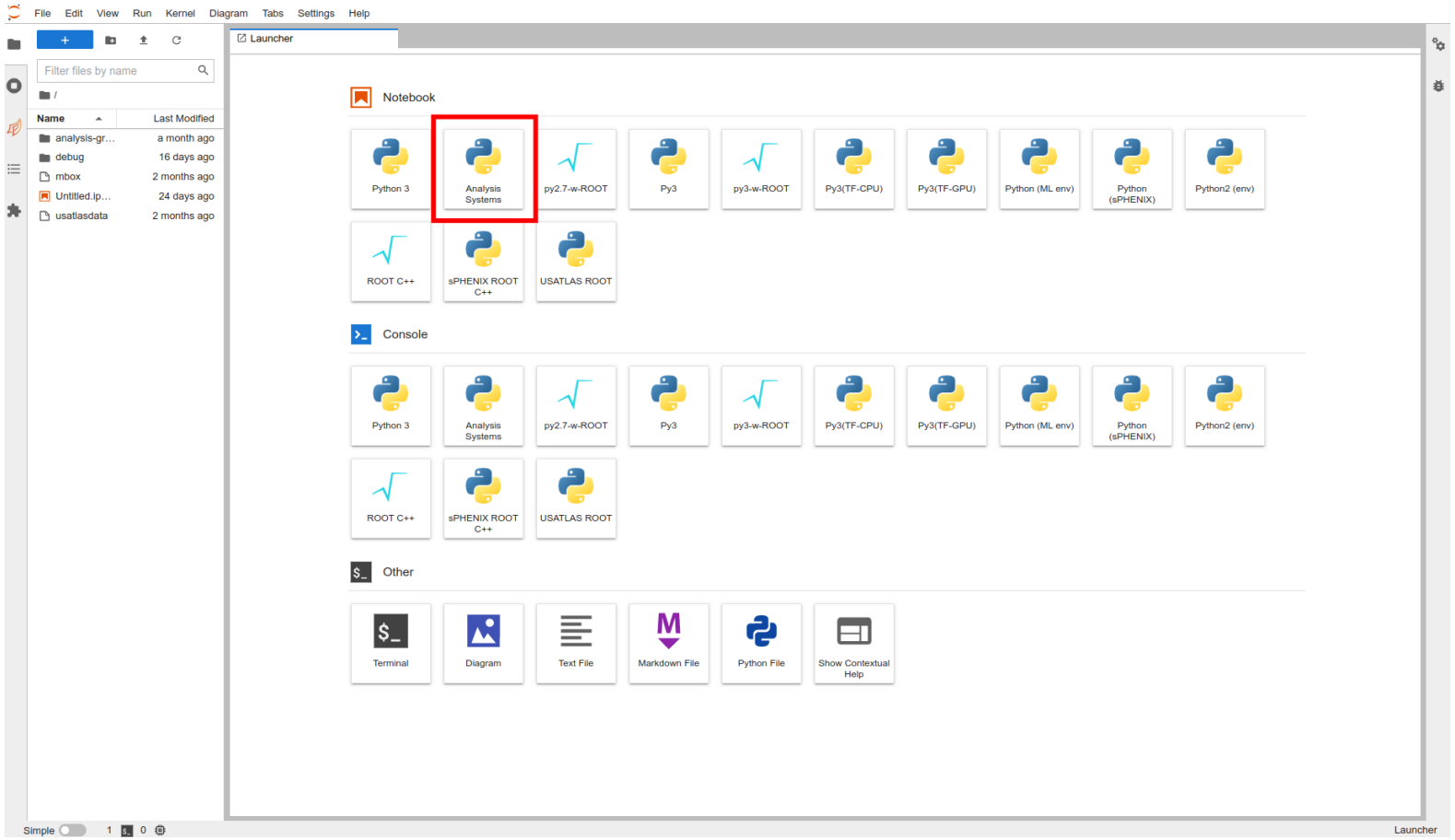
WIP: Custom kernel discovery

```
Singularity> mkdir -p ~/.local/share/jupyter/kernels
Singularity> ln --symbolic \
   /opt/micromamba/envs/analysis-systems/share/jupyter/kernels/analysis-systems \
   ~/.local/share/jupyter/kernels
Singularity>
```

Hack for time being: Create custom symlink first time

(c.f. https://github.com/iris-hep/analysis-systems-base/issues/12)

WIP: Custom kernel discovery



Hack for time being: Create custom symlink first time

(c.f. https://github.com/iris-hep/analysis-systems-base/issues/12)

Example: AGC CMS Open Data $t\bar{t}$ Analysis

- Doug has moved data to `/usatlas/atlas01/atlasdisk/users/benjamin/AGC/`
- Notebook runs end-to-end (good first start 🚀)
- BNL does not use Kubernetes, so not a Coffea-casa AF, so use global config

```
...
PIPELINE = "coffea"  # pure coffea setup
USE_DASK = True  # enable Dask
AF = "local"  # local setup, not coffea-casa
AF_NAME = "bnl"  # Added on Matthew's fork
...
```

Example: AGC CMS Open Data $t\bar{t}$ Analysis

- Execute the data delivery pipeline step

```
N_FILES_MAX_PER_SAMPLE = 10  # 157 GB
...
[########################################] | 100% Completed |  1min 32.6s
execution took 93.11 seconds

N_FILES_MAX_PER_SAMPLE = 50  # 678 GB
...
[########################################] | 100% Completed |  6min 14.1s
execution took 375.80 seconds

N_FILES_MAX_PER_SAMPLE = 100  # 1 TB
...
[########################################] | 100% Completed | 10min 10.0s
execution took 611.55 seconds
```

Example: AGC CMS Open Data $t\bar{t}$ Analysis

- Execute the data delivery pipeline step
- Nothing that I've done at BNL has been optimized yet (just doing defaults)
- **N.B.** Need to get better information on how scaling is done at BNL (so comparisons are not valid here yet)
  - UChicago AF is running with `AF = "coffea_casa"` and so scaling across k8

Data	BNL AF (sec)	UChicago AF (sec)
157 GB	93	65
678 GB	375	182
1 TB	611	243

(The input files are all in the 1–2 GB range)

Comparison of unoptimized numbers (don't try to 1:1 these)

Summary

- Have a runnable environment for AGC at BNL AF (🚀)
- BNL team is working to make drop-in with custom images well specified and easy
- Things work, but to understand how scaling is working will need to improve the monitoring story
  - Currently don't have Dask dashboards detecting the Dask cluster
- Also need to try other AGC analyses and expand / modify environment

Backup

The end.