Coffea-casa: an analysis facility prototype

Mat Adamec, Ken Bloom, Oksana Shadura,
University of Nebraska, Lincoln

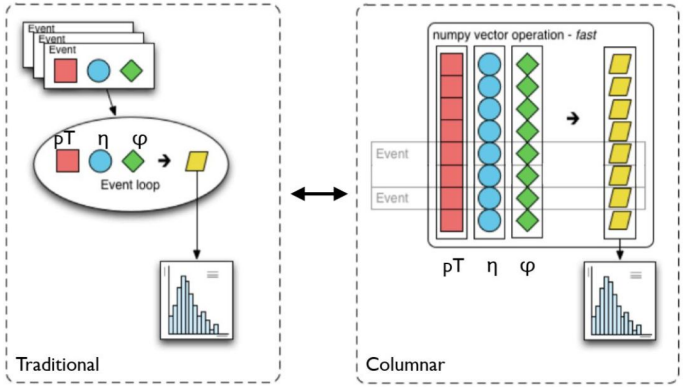
Garhan Attebury, Carl Lundstedt, John Thiltges
University of Nebraska Holland Computing Center

Brian Bockelman
Morgridge Institute

Building blocks used for designing AFs

- Modern authentication (AIM/OIDC), tokens, macaroons
- Efficient data delivery and data management technologies
- Columnar analysis and support new pythonic ecosystem
- Modern deployment and integration techniques
- Support for object storage
- Efficient data caching solutions
- Easy integration with existing HPC resources

Building blocks: Coffea Analysis Framework



DASK

APACHE Spark™

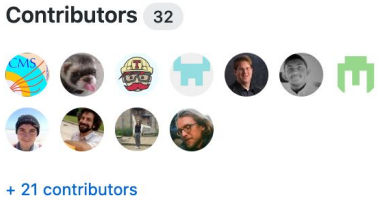
Parsl

New columnar data analysis concepts!

User just needs to define a high-level wrapper around user analysis code: **the coffea processor** and coffea framework will take care of everything incl. ***scaling-out***

Distributed executors!

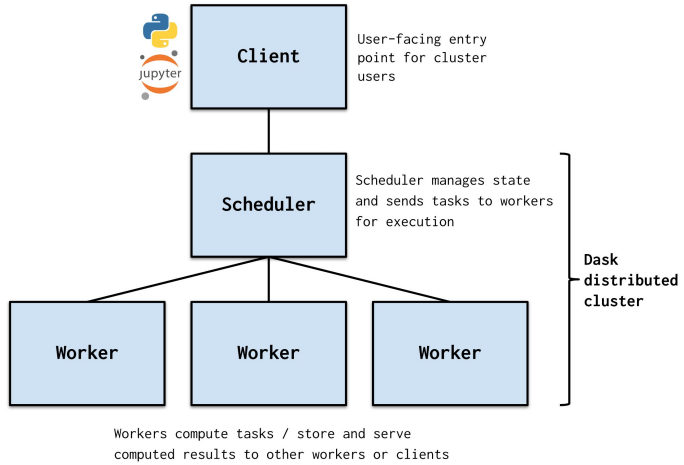
Coffea developers: Lindsey Gray, Matteo Cremonesi, Bo Jayatilaka, Oliver Gutsche, Nick Smith, Allison Hall, Kevin Pedro (**FNAL**); Andrew Melo (Vanderbilt); and others



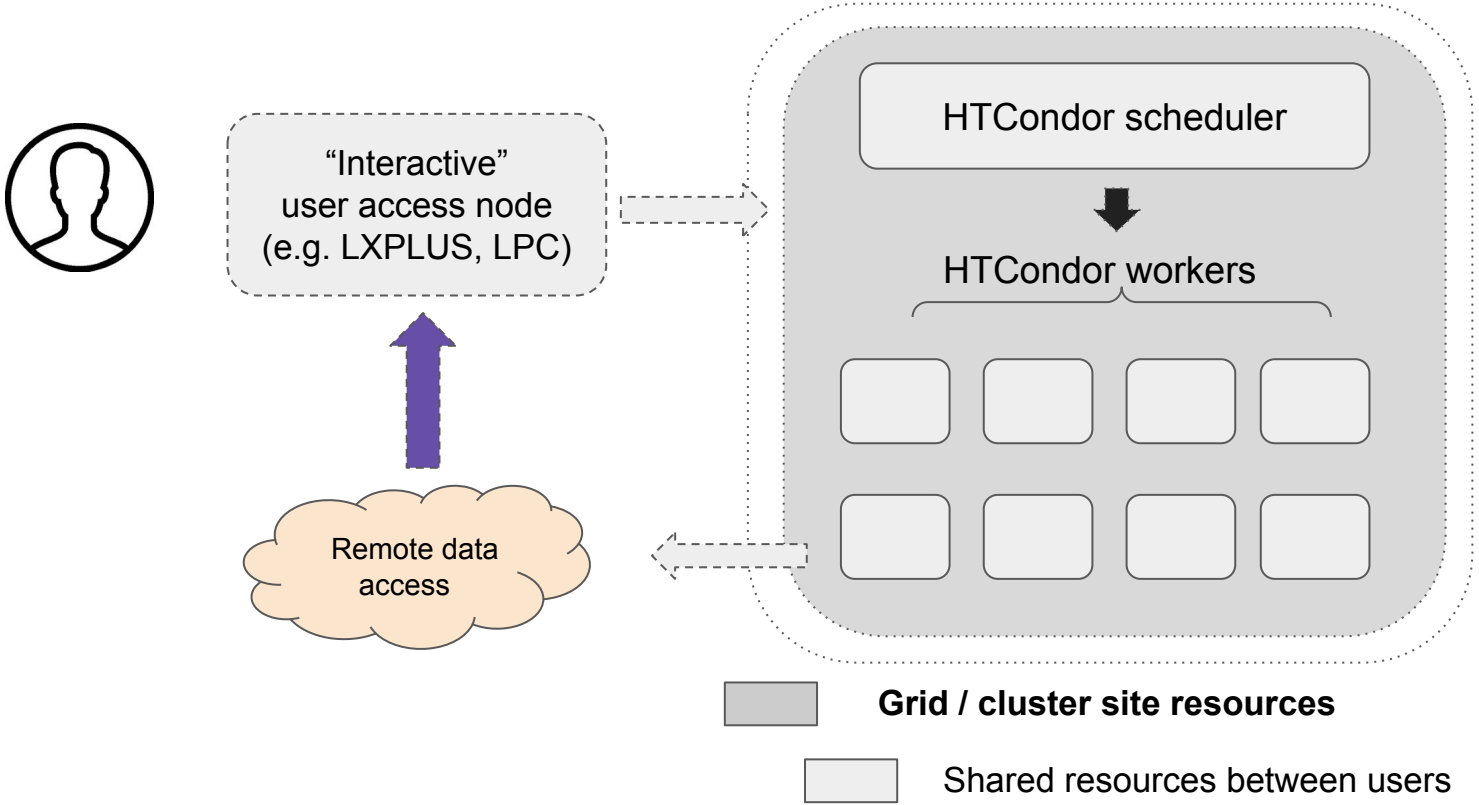
Building blocks: Dask

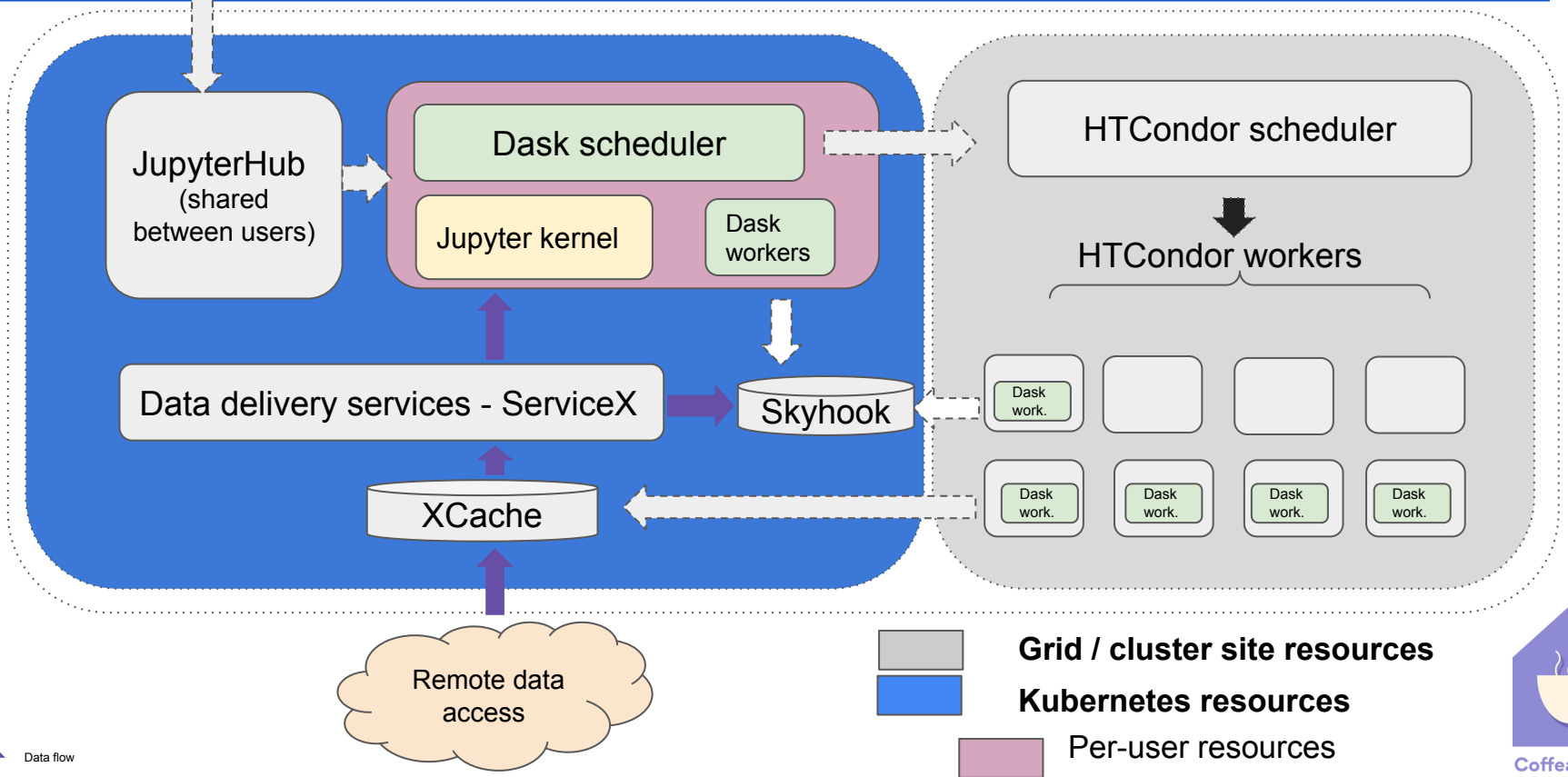
- Dask provides a task-management computational framework in Python based on the manager-worker paradigm
- Dask exposes lower-level APIs letting to build custom systems for in-house applications (!)
- Integrates with HPC clusters, running a variety of schedulers including SLURM, LSF, SGE and *HTCondor via "dask-jobqueue"*
- ***This allows us to create a user-level interactive system via queueing up in the batch system***

Dask can be used inside Jupyter or you can simply launch it through Jupyter and connect directly from your laptop



Simplified diagram of hypothetical Analysis Facility currently used by users




JupyterHub
(shared
between users)
Dask scheduler
Jupyter kernel
Dask
workers
HTCondor scheduler
HTCondor workers
Data delivery services - ServiceX
Skyhook
XCache
Dask
work.
Remote data
access
Grid / cluster site resources
Kubernetes resources
Per-user resources
Shared resources between users
Data flow
X sends requests to Y
Coffea-Casa

Analysis Facility Coffea-Casa

- **Authentication inside the system is independent of grid credentials**
  - Coffea-casa facility uses **OpenID Connect (OIDC)**
    - CMS, ATLAS AIM
  - **Enabled token authentication** for HTCondor:
    - Generated a token for authentication with HTCondor, required for Dask scale-out to the larger resources
  - **Generated a data access token for authentication with a local XRootD server**
  - Generated X.509 credentials (including a CA, host certificate, and user certificate) for use in Dask for TLS as well for user communication to Dask scheduler endpoint
- Security: TLS enabled communication between workers and scheduler by default
- Kubernetes pod customization ‘hook’ to create secrets for services
- Highly customized **“Analysis” Docker container(s)**
- All features are **incorporated into a Helm chart** (Kubernetes packaging format)

Analysis Facility Coffea-Casa

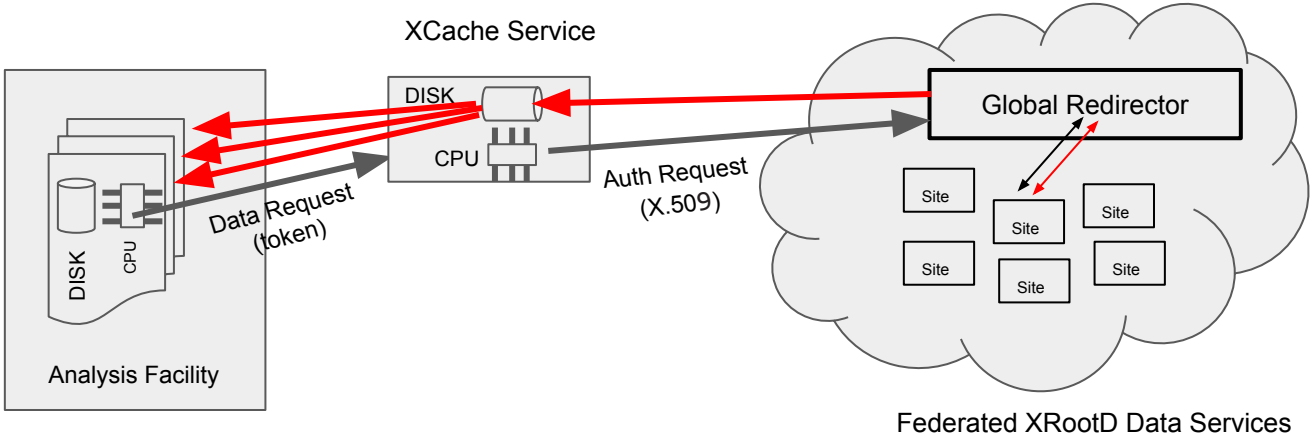
- ***CoffeaCasaCluster:* extending HTCondorCluster integration for Dask**
  - To handle the customizations needed for the Coffea-casa environment, we developed the *CoffeaCasaCluster* object, an extension of ***Dask-jobqueue***'s *HTCondorCluster* object.
  - *CoffeaCasaCluster* ensures the Dask worker starts with the appropriate Docker container in the HTCondor batch system with appropriate configurations and with the firewall ports configured correctly.
- **Looking into new backends:**

  Workqueue (http://ccl.cse.nd.edu/software/workqueue/)

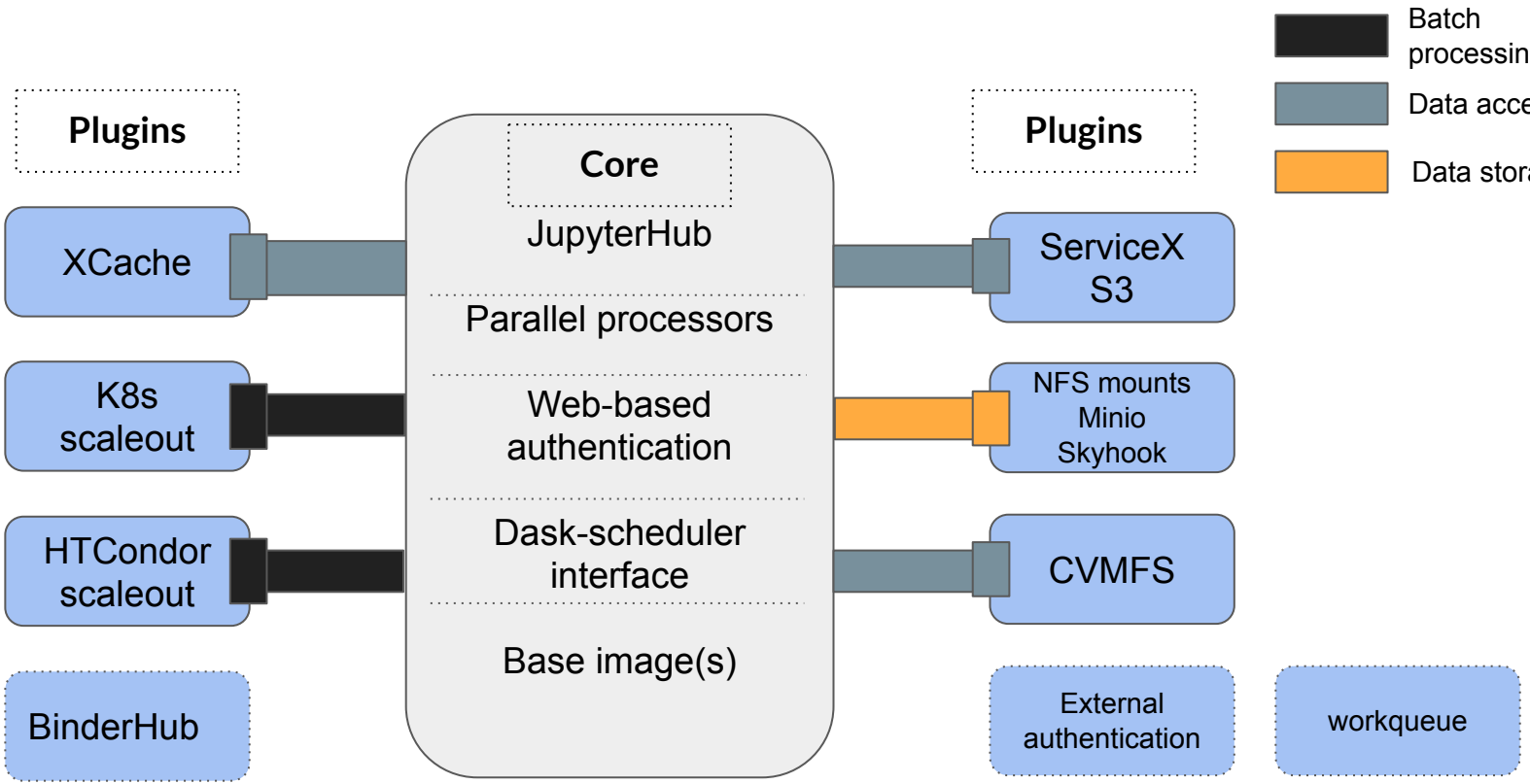
- ***More features as work in progress***

Data access for Analysis Facility @ UNL

- For speeding up data access Nebraska Tier-2 hosts an **XCache service with 90TB of cache space**
- Access data hosted by an HEP experiment:
  - *no GSI credential within the facility,* **the auto-generated data access token can be used to authenticate with an proxy service based on XRootD/XCache**

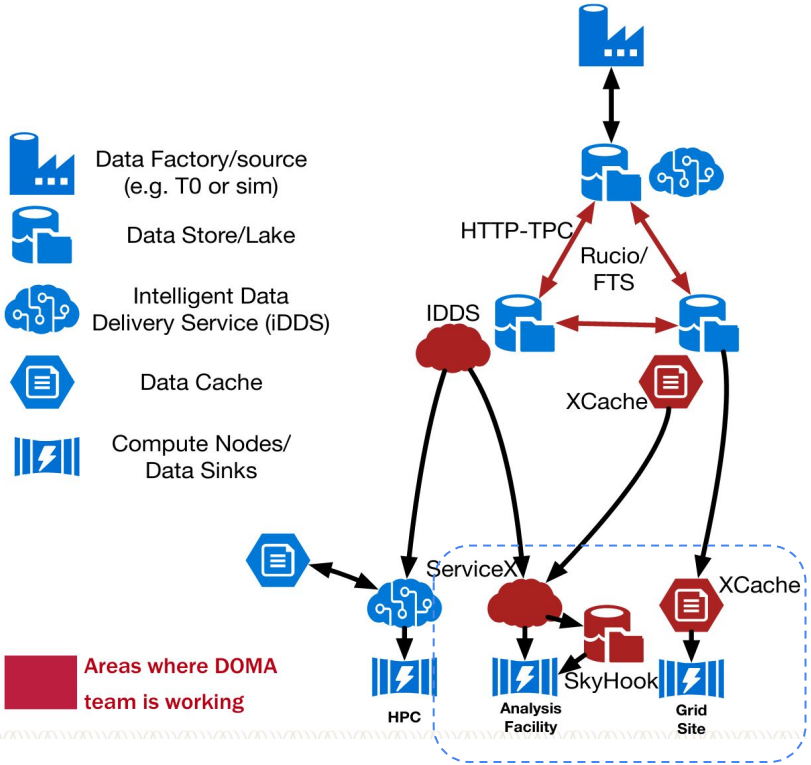


Coffea-casa components

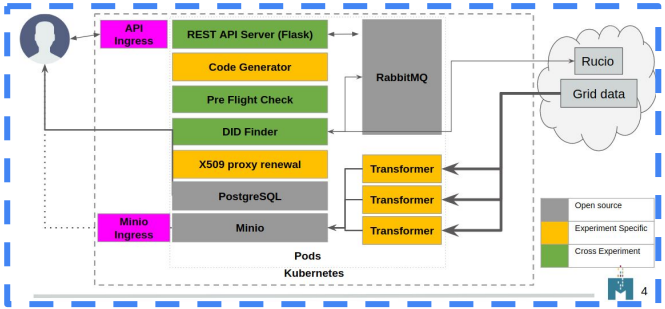


[Coffea-casa team]

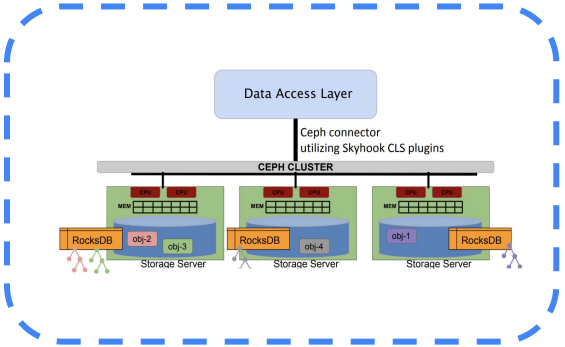
Analysis Facility and Distributed Ecosystem (Data Lakes)



Coffea-casa



Servicex

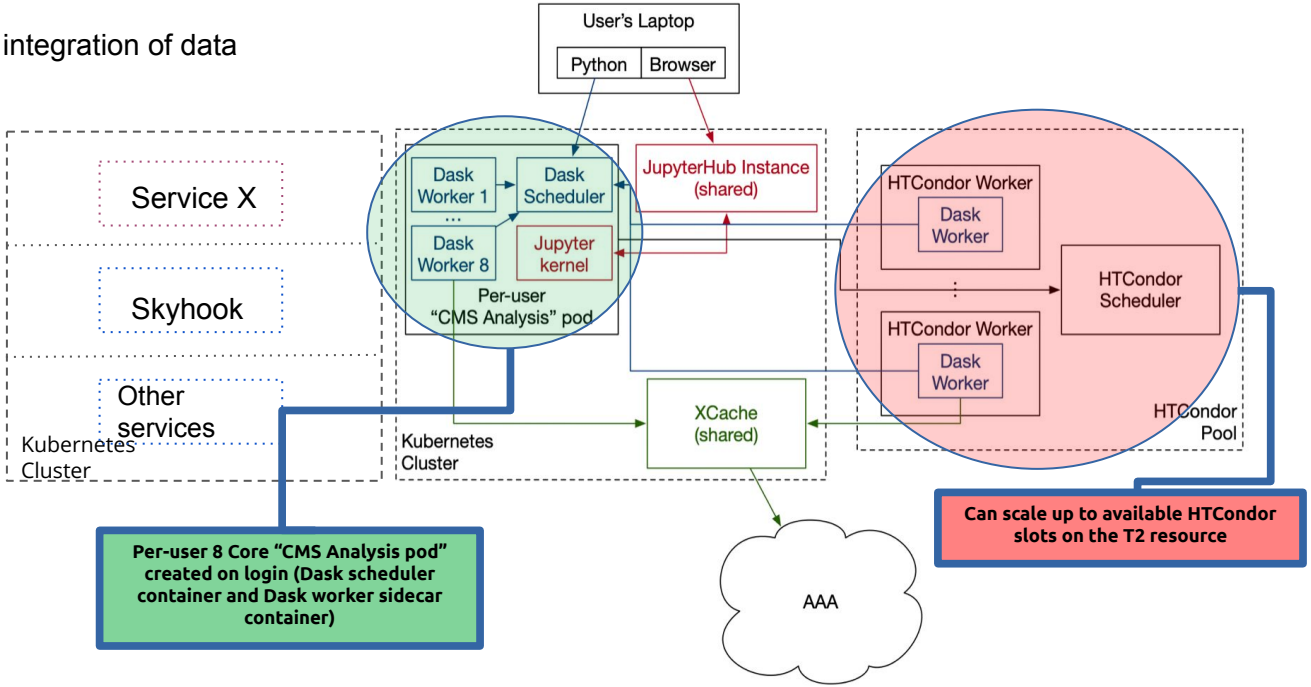


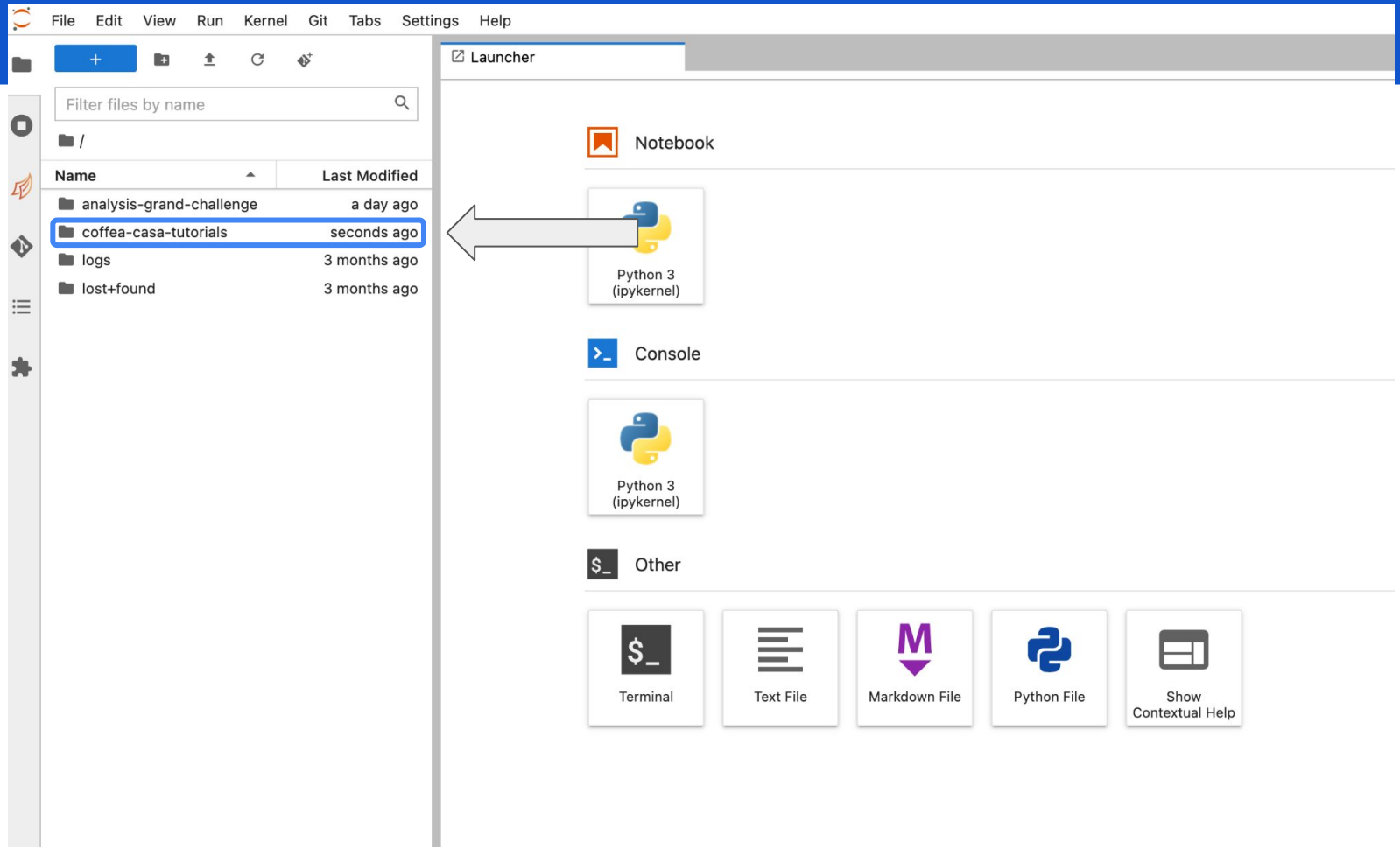
Skyhook

Analysis Facility Coffea-Casa @ UNL

We are easily bridging K8s resources with UNL Tier2 resources, while providing interactive environment!

On-going work on integration of data delivery services




File Edit View Run Kernel Git Tabs Settings Help
Filter files by name
/
Name
Last Modified
analysis-grand-challenge
a day ago
coffea-casa-tutorials
seconds ago
logs
3 months ago
lost+found
3 months ago
Launcher
Notebook
Python 3 (ipykernel)
Console
Python 3 (ipykernel)
Other
Terminal
Text File
Markdown File
Python File
Show Contextual Help

Thank you!

Coffea-casa webpage

GH discussions