Elastic Analysis Facility

Burt Holzman, Fermilab

OSG All-Hands Meeting, Joint USCMS/USATLAS Session

Sep 4, 2020

Quick Disclaimer

- I haven't done much of this technical work

- The real people who have worked on this or adjacent work (and not an exhaustive list!)
  - Maria Acosta, Nick Smith, Lindsey Grey
  - Matteo Cremonesi and coffea team
  - Jim Pivarski and awkward array team
  - Chris Bonnaud, Ed Simmonds, Tony Tiradani, Farrukh Khan
  - Coffea-casa team @ UNL
  - Brian Bockelman

Analysis Facilities are not new

CAF specs

The intention of the CAF specs from 1997 seem to have been to define central analysis facilities that would provide a level of "user satisfaction" similar to Run I.

JAN
27
2004

The CMS User Analysis Farm (UAF)

- Introduction
- News
- How to get access to the cluster?

Analysis Facilities are also targeting our future

SnowMass2021

The Evolution of HEP Analysis Facilities

Paolo Calafiura <pcalafiura@lbl.gov>
V0, 8/31/2020

Analysis Facilities for Late-Stage Analysis

Kevin Lannon (klannon@nd.edu)[1], Paul Brenner[1], and Mike Hildreth[1]

[1]Univeristy of Notre Dame

Snowmass Letter of Interest - Analysis Facilities - CompF5

Burt Holzman <burt@fnal.gov> (Fermilab), Lindsey Gray (Fermilab), Bo Jayatilaka (Fermilab), Nick Smith (Fermilab)

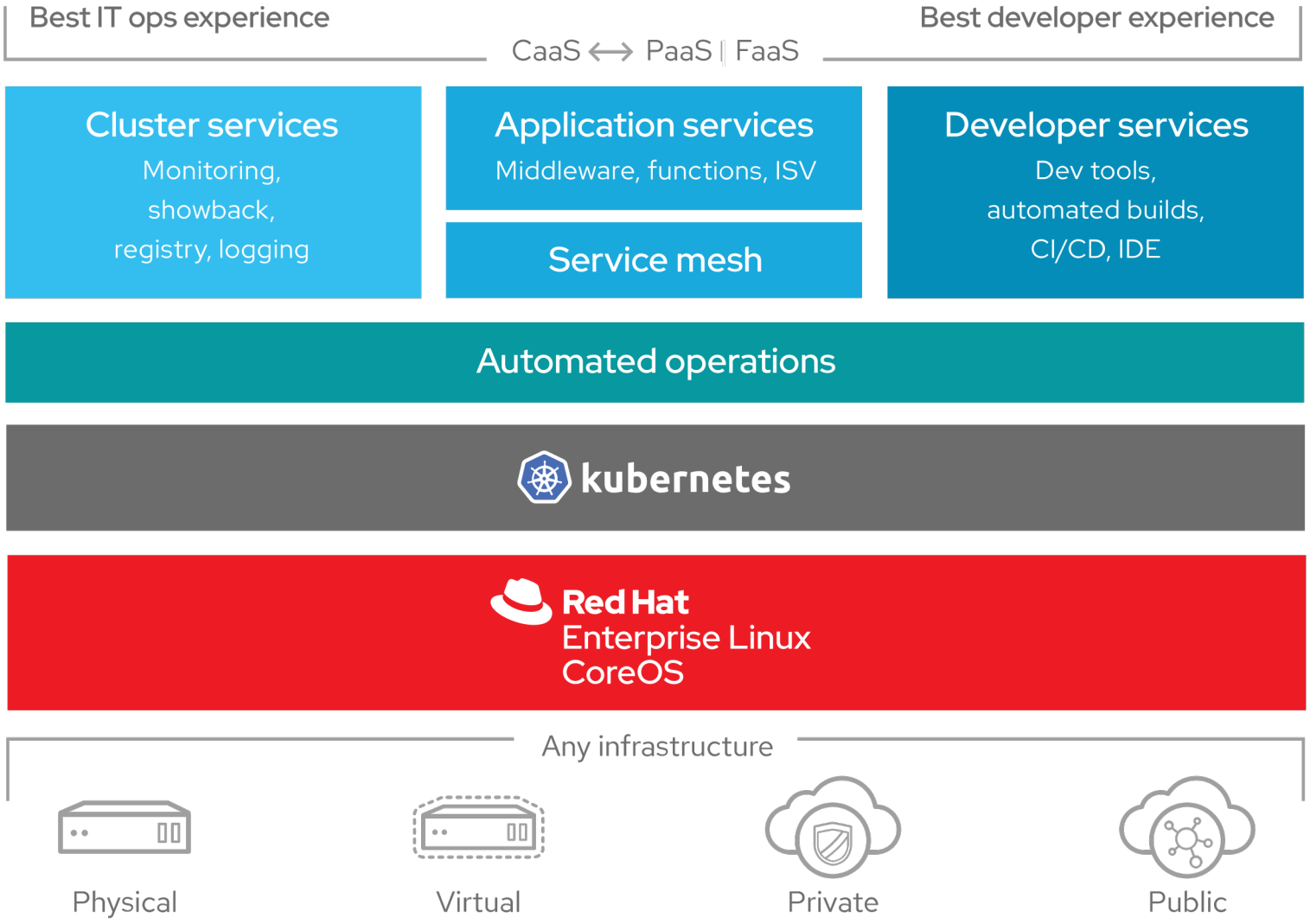
Elastic Analysis Facility: motivation

- Traditionally we have done **row-wise** analysis
  - Process one event at a time serially
  - Scripts and batch jobs (and now also notebooks)
  - We need to support this
- Trending towards **column-wise** (tidy/big data) analyses
  - **Low-latency** access to CPU resources
  - Fast access to a significant amount of **disk**
- Low latency implies <100% resource utilization
  - **Coscheduling** analysis with other computation is appealing

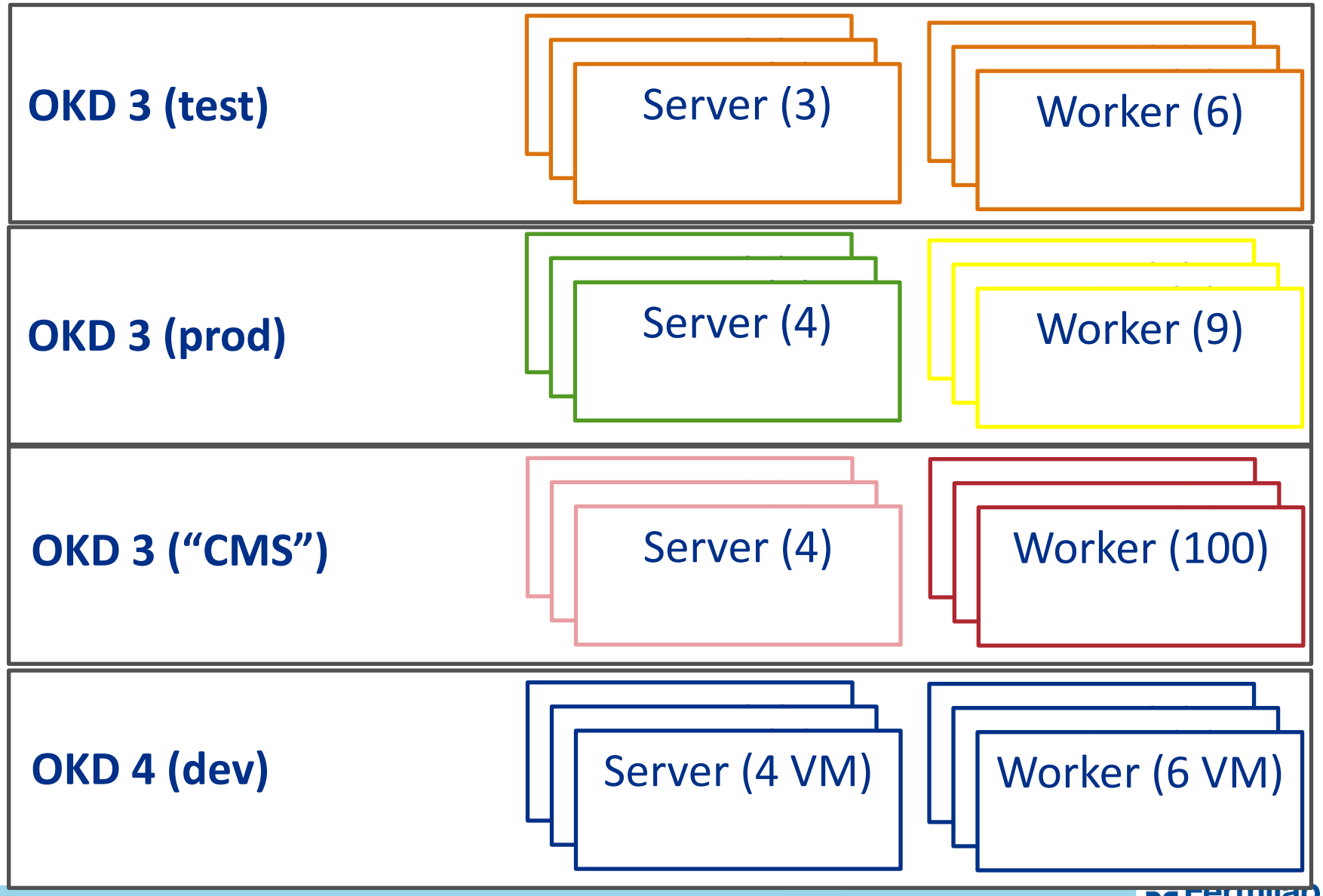
Elastic Analysis Facility goals

- As a foundation layer, use a **containerized infrastructure** (Kubernetes-based). Stable, in-production, operated and supported by routine SCD Facility operations staff. [“Routine” corresponds to following a set of documented procedures with responsibilities rotated on a regular basis with staff.]
- A **Jupyterhub** deployment that provides a number of different environments with LDAP authentication for Fermilab users. Allow users to customize the user-space environment.
- Support “typical” (**row-wise**) analysis needs
- Support a columnar analysis low latency application (e.g. Apache Spark, **Dask**, …)
- **Document and package** the Elastic AF so it can be deployed elsewhere

Elastic AF: layered architecture



State of the Elastic AF infrastructure

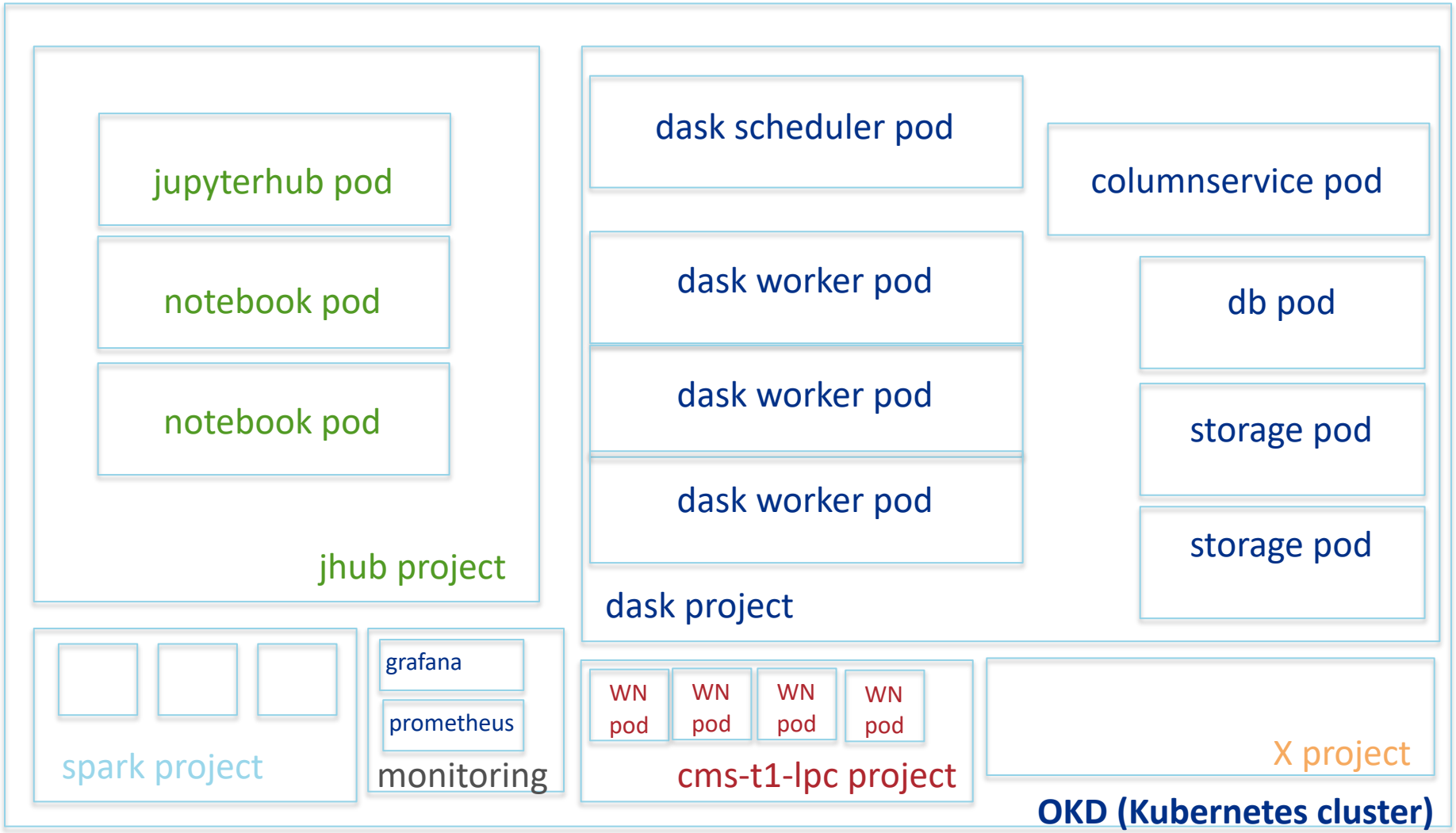


older

newer

VM

Current Elastic AF pre-prototype - static allocation



Current Elastic AF pre-prot

That looks familiar …

jup

note

notebook pod

ect

d

db pod

storage pod

storage pod

dask worker pod

dask project

WN pod

WN pod

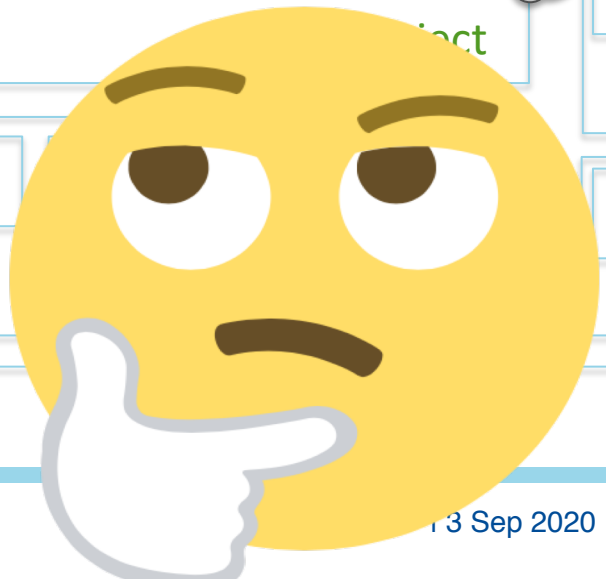
WN pod

WN pod

cms-t1-lpc project

X project

OKD (Kubernetes cluster)



A Prototype U.S. CMS analysis facility

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

Garhan Attebury, Carl Lundstedt, Derek Wietzel
University of Nebraska Holland Computing Center

Mátyás Selmeci
University of Wisconsin, Madison

Brian Bockelman
Morgridge Institute

Proposed Analysis Facility @ T2 Nebraska

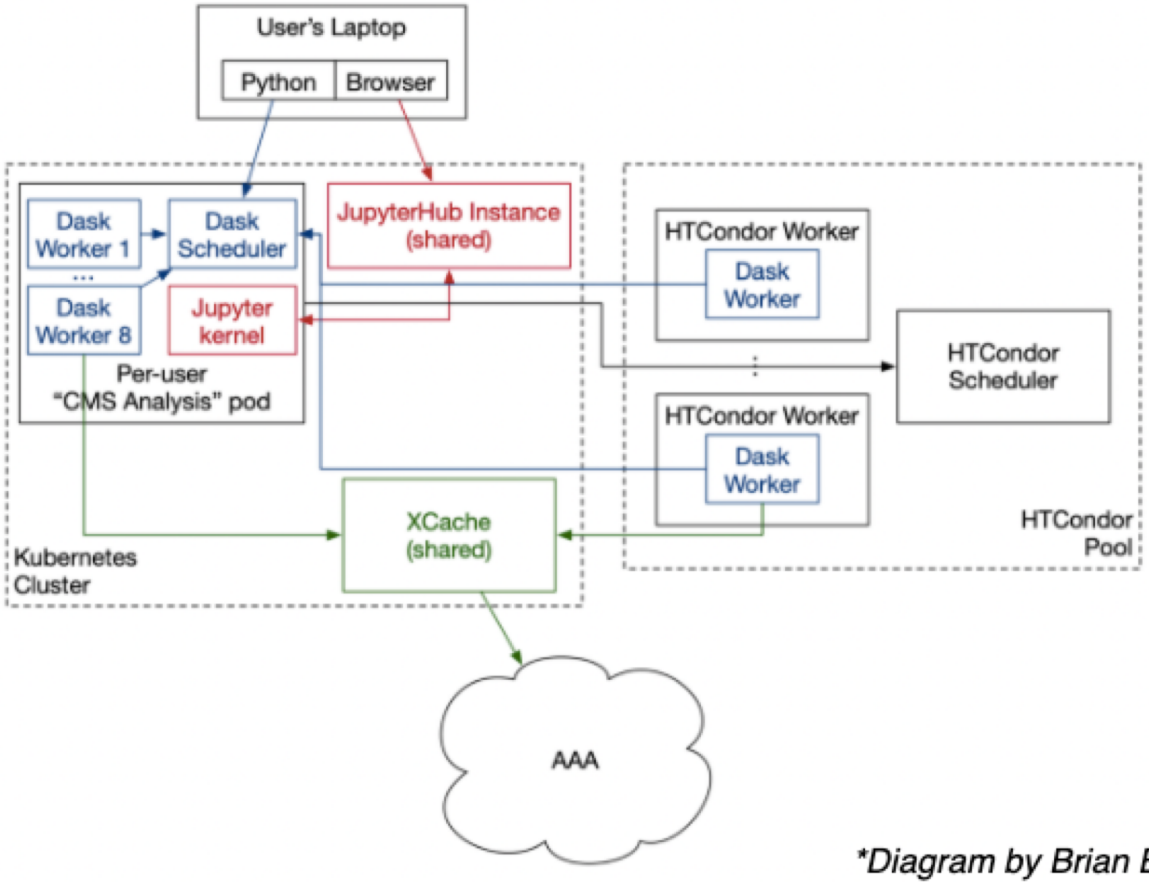


*Diagram by Brian Bockelman

Token-based solution (rather than X.509)

See demos of the UNL Facility (PyHEP 2020)

https://www.youtube.com/watch?v=CDIFd1gDbSc
https://www.youtube.com/watch?v=oPl0t8J36-Q

Near-term tasks

- Document resource requirements from existing applications
- Document (for users) resource constraints and acceptable use (e.g. storage is a cache with a finite lifetime for data)
- Survey Kubernetes platforms (Openshift/OKD, Rancher, kubeadm, ...)
- Monitoring
- Storage: survey underlying storage technology (including GlusterFS, Ceph, BeeGFS?, others?). Evaluate both technical strengths and support models based on resource requirements. Evaluate underlying storage platform (R&D NVME cluster)
- Deploy the token-based UNL facility as a project
- Benchmark NVME cluster (CephFS+minio vs. radosgw)
- Evaluate feasibility of cooperative scheduling with power users

Medium-term tasks

- Establish list of core infrastructure services. Purchase and deploy high quality redundant hardware.
- Deploy a stable, maintainable storage and kubernetes-based platform, keeping the existing prototype facility as a development/ITB cluster.
- Write documentation and package components when necessary -- a “zero to Elastic AF” for an outside facility to pick up and use.
- Document and deploy a policy for Jupyterhub python customizations (e.g. pip install) – such as pruning stale configurations after N weeks
  - (Looking forward to Wei’s talk)

A few thoughts and questions

- CMS is currently driving, but the Elastic AF needs to support multiple experiments (DUNE, Intensity Frontier)
- We are using Kubernetes on commercial cloud – and will continue to do so. Can we bridge the gap easily?
  - Hopefully Igor has already answered this!
- We share a foundation with Slate/SSL – potential for collaborative work beyond “k8s best practices show and tell”
  - E.g. trust and curation of container images?
- What’s the right scheduling model? (What are we trying to optimize?)