Challenges For Future HEP Analysis Facilities

Chris Hollowell <hollowec@bnl.gov>

OSG All-Hands Meeting
March 5, 2021

BROOKHAVEN NATIONAL LABORATORY | Scientific Data and Computing Center

U.S. DEPARTMENT OF ENERGY

Existing and Future HEP Analysis Facilities

- A number of Analysis Facilities (AFs/T3s) exist for ATLAS and CMS
  - Includes a mix of local and larger shared analysis facilities
  - **Typically provide compute services using the standard HTC interactive/batch and grid model**

- It is foreseen that the HL-LHC will greatly increase the demand for resources at analysis facilities
  - HL-LHC era facilities will also have additional requirements
    - Support for new/modern analysis software tools
    - Federated ID?

- R&D effort in IRIS-HEP, HSF, LHC computing operations programs and facilities on developing/deploying new software/services for future AFs

- Ongoing collaboration (organized by B. Bockelman) between and CMS and ATLAS to identify common requirements and areas where we can work together for HL-LHC AFs in terms of services/infrastructure, and defining best practices

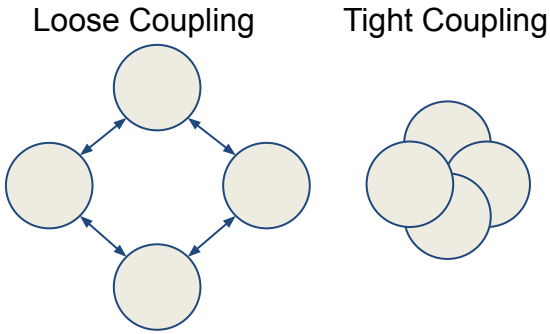
Future Challenges at Existing HEP Analysis Facilities

- Creates some adaptation challenges for existing analysis facilities
  - How to support new services/infrastructure/requirements at existing AFs while continuing to providing the standard HTC/grid environment users have become accustomed to and will require for the foreseeable future?
  - AF requirements should be defined in such a way that they do not conflict with existing regulations at sites
  - Grid model has had been greatly successful in part because sites are “loosely coupled”

Challenges for at Existing HEP Analysis Facilities (Cont.)

- Future AFs will likely couple/federate functionality

- Currently, in order to meet local requirements and to permit local optimization of resources (including leveraging synergies between local projects), grid sites retain the freedom to choose the specific methodology used for:
  - Deploying infrastructure
  - Configuration management (Puppet, Chef, Ansible, etc.)
  - Batch system (HTCondor, Slurm, PBS, etc.), and configuration
  - Etc.



- This loose coupling should continue to be preserved in future AFs where possible
  - **It is the services themselves, and the not how they are implemented/deployed that should be the focus**

Kubernetes at Analysis Facilities

- A number of IRIS-HEP projects we want to run at future AFs have been developed for k8s
  - REANA
  - ServiceX
  - Etc.

kubernetes

reana

- **It is clear that future AFs will need to support k8s in some form**
  - Vanilla k8s, Openshift, OKD etc.

- **k8s administration is complex**
  - May require training or staff increases to support at existing sites

- **k8s is difficult to deploy securely**
    - Effectively no security/restrictions for users by default
  - Particularly difficult to secure when used in conjunction with an existing/typical HTC/HPC environment
  - **Openshift/OKD helps in this area**

Issues Integrating k8s into Existing HTC/HPC

- **k8s doesn't care much about UIDs, but in HTC/HPC environments they are very important**
  - Vanilla k8s containers run as root by default, with full privileges

- **There is usually heavy use of UID-auth based network filesystems (i.e. NFS, GPFS, etc.) at traditional/existing HTC/HPC deployments**
  - Can be complicated to map containers running in k8s to users' UID, and securely allow them to access UID-auth network filesystems
  - Root on a node with network FS mounted, or root anywhere on a network with access to that FS (can open privileged port <1024) == Root on the FS
    - Can delete/modify arbitrary files
  - One reason Singularity is used in our community instead of Docker
    - Regular users are never root in containers (without a user namespace mapping), and thus never root on our hosts or network filesystems

- HPC/HTC Batch systems also typically control access/policy via UID

Issues Integrating k8s into Existing HTC/HPC (Cont.)

- **Openshift/OKD solve some of these issues**
  - Secure out of the box
  - Containers never run as root by default
  - Users have limited privileges/capabilities in the system, access restricted to their projects/namespaces by default
  - Can utilize keycloak with IPA for user access
  - Long-term support - insulation from large changes between k8s versions

- ***However, still difficult to map containers to running user's UID for compute cluster network FS access***
  - **Eventual transition to object-based-storage with token-auth would help**
    - Requires user buy-in, and can't easily replace POSIX home directories

- Some helm charts expect full k8s privileges (i.e. namespace-wide) for deployment and do not function with unprivileged Openshift/OKD

Could You Run A Whole Site in k8s?

- **If one needs to support k8s at future analysis facilities, could you run your whole site in k8s?** ***Maybe for small sites***
  - WMS like Panda now support scheduling directly via the k8s API - no CE

- But not really possible/realistic to have users submit processing jobs directly to k8s
  - **Complexity of k8s YAML over traditional batch JDF/commandline**
  - **Multi-tenancy security issues in vanilla k8s**
    - In GKE/EKS tenants get their own k8s cluster where they are full administrators - not one big k8s cluster for all tenants
    - Critical PodSecurityPolicy functionality needed for multi-tenancy not enabled by default on the kube-apiserver commandline, and still considered beta in v1.20 release
    - Openshift/OKD's contains default functionality/configuration make it more suitable for multi-tenant use

Could You Run A Whole Site in k8s (Cont.)?

- **Batch systems like HTCondor and SLURM are more mature schedulers, and tailored to our environments**
  - Support complex batch policies we require in large shared environments
  - Fairshare is lacking in k8s - per-namespace resource quotas are likely not enough
    - k8s primary focus on scale-out of web applications

- **Could you just run HTCondor/SLURM as a service inside k8s?**
  - ***Potentially***
    - The HTCondor developers have demonstrated this:
    - https://indico.cern.ch/event/936993/contributions/4022096/attachments/2109009/3547280/k8s.pdf
  - **Greatly increases the complexity of managing a farm with little benefit**
    - What services need to scale to the level of compute (on compute-style hardware) that can't use batch systems directly?

Traditional Batch Systems as Container Orchestrators

- **Batch systems like HTCondor and SLURM are also “container orchestrators” themselves**
  - Thanks to tools like Singularity
  - Don’t get k8s advanced networking/CNI capabilities
    - But not needed for a number of applications
      - Particularly those that are compute-job-like
    - Can simplify the integration with existing/traditional analysis facilities

- **BNL and SLAC have production JupyterHub instances which utilize HTCondor and SLURM to orchestrate notebook execution**
  - k8s not needed for JupyterHub service

- DASK also interfaces with batch systems
  - Functional with HTCondor at BNL

- While REANA requires k8s for the front-end, it can also schedule user job containers to HTCondor/SLURM

Hybrid k8s & Traditional Batch Model

- **The REANA approach simplifies adoption at existing analysis facilities**
  - Front-end may require k8s
  - But back-ends that need to run processing should support running containers on batch systems where possible, but also on k8s
  - Suggest future analysis compute tools adopt similar functionality
    - ***HTC/HPC batch systems aren't going away anytime soon***

- **Hybrid model adopted by BNL ATLAS Analysis Facility**
  - Use batch system for container orchestration for AF services where possible
    - Jupyter, DASK
  - Staff-only (single-tenant) k8s cluster available to deploy trusted services which require it
    - REANA testbed available on this cluster
      - Ongoing work to integrate with our batch farms/clusters
  - OKD test cluster available for multi-tenant regular user use
    - Allows users to internally deploy services like MySQL DBs, web apps

Federated ID at Analysis Facilities

- Increasing demand for federated ID/login at analysis facilities
  - Simplifies user experience and expedites access to disparate resources

- Somewhat problematic at DoE National Labs
  - Most DoE labs have strict account creation requirements for interactive shell access, and have local user/visitor administration centers which need to be involved in the account creation/approval process
    - **A policy issue - not a technical problem**
      - For example, BNL already has federated access to several web services
  - Distributed Computing and Data Ecosystem (DCDE) ASCR pilot project is looking to change that
    - Eventual goal to allow federation of accounts between labs through OneID
      - Could potentially allow federation of BNL accounts with other labs (and vice-versa) for interactive access
  - **In parallel to DCDE, BNL and SLAC CIOs also in discussions regarding federating access between their AFs via InCommon**
    - Including ongoing technical collaboration between the SLAC & BNL AFs

Commercial Clouds

- What about running Analysis Facilities in commercial clouds?
  - Fair cost comparisons are difficult to establish and include many factors
  - **However, previous studies have shown that for large numbers of dedicated resources, on-site datacenters are more economical than utilizing commercial cloud providers**
    - A. Wong *et al.* The role of dedicated data computing centers in the age of cloud computing 2017 *J. Phys.: Conf. Ser.* **898** 082009 https://iopscience.iop.org/article/10.1088/1742-6596/898/8/082009/pdf
    - Holzman, B., Bauerdick, L.A.T., Bockelman, B. *et al.* HEPCloud, a New Paradigm for HEP Facilities: CMS Amazon Web Services Investigation. *Comput Softw Big Sci* **1**, 1 (2017). https://doi.org/10.1007/s41781-017-0001-9 https://www.osti.gov/pages/servlets/purl/1418149

Commercial Clouds (Cont.)

- Expensive network egress charges also tend to make cloud options less appealing
- **Moving to the cloud does <u>not</u> allow for significant savings in staff cost**
  - Few FTE required for hardware/datacenter operations
  - Still need highly trained/costly developers and IT admins to develop/run services
- **However, cloud-bursting when additional non-dedicated resources are needed is valuable, and should be supported in future AFs where possible**
  - ***k8s helps in this area***
  - ***But so do tools like condor_annex, and Parsl/funcX***

Conclusions

- Future HL-LHC analysis facility resource requirements, and services will necessitate adaptation from traditional facilities to support new/modern tools and software being designed by IRIS-HEP

- **Clear that support for k8s will be necessary**
  - Consider use of Openshift/OKD to ensure secure deployment where multi-tenancy needed?
  - ***But it should not be required that whole facilities move to k8s***
    - Hybrid model that allows for continued use of batch systems desirable to simplify adoption at sites - BNL has shown this is possible

- **Federated ID/login will likely be an important part of future analysis facilities**
  - ***SLAC and BNL are working to support this***

- Important to enable cloud-bursting for peak demand at AFs, but it is more economical to operate equipment at site datacenters for dedicated resources