CMS Tier-2 Analysis Facility Demo

October 28, 2022

Stefan Piperov

spiperov@purdue.edu

Erik Gough

goughes@purdue.edu

PURDUE UNIVERSITY | Research Computing INFORMATION TECHNOLOGY

What is an ANALYSIS FACILITY ?

- **How we did Physics Analysis in the past**
  - Local machine (laptop/desktop/workstation)
  - SSH access to login nodes, ideally with X11-support
  - Write C++ analysis code
  - Run batch jobs for full-scale analysis (*crab*, etc.)
  - Run ROOT using data on local or shared file systems
- **How can we improve this today**
  - Interactive, browser-based computing with immediate feedback
  - Python instead of C++
  - Provide a set of common environments for analysis
  - Ability to fully scale analysis via *Slurm* or *Kubernetes*, GPUs
  - While maintaining current analysis methods (you can still run ROOT in the old-fashioned way and submit batch jobs if desired…)
- **These are new technologies being investigated across the field, some are still under development**
  - In this demo, we want to show you what is possible today at Purdue
  - We will have a follow up survey to gather your feedback for future plans

Interfaces to the AF

Analysis Facility Landing Page

- **JupyterHub**
  - Provides immediate access to notebook environments
  - Shared resource - runs on Hammer front-end nodes
  - https://notebook.hammer.rcac.purdue.edu

- **Open OnDemand - NEW!**
  - Provides queued access to notebook and desktop environments
  - Exclusive access - runs on Hammer compute or Gilbreth GPU nodes
  - https://gateway.hammer.rcac.purdue.edu
  - https://gateway.gilbreth.rcac.purdue.edu

- **Composable JupyterLab - NEW!**
  - Provides immediate access to notebook and desktop environments
  - Exclusive fine grained access - runs on Geddes Kubernetes
  - https://cms.geddes.rcac.purdue.edu

Environments

- **What are they, and how they get referred to**
  - Conda environments
  - LMOD modules
  - Jupyter kernels
- **Why do we need them**
  - Add functionality
    - missing software packages
    - different versions (needed vs installed at system level)
  - Manage complexity
    - not all software packages are compatible with one another
    - finding compatible versions, and storing them in one place is the essence of building those new environments
- **Available pre-defined analysis environments (kernels)**
  - Python 3 - PyROOT
  - ROOT - C++
  - COFFEA
  - CMSSW
  - Python Machine Learning (ML)
- **Create your own environments!**
  - Share with colleagues via Data Depot, or experiment privately
    - using ‘conda create’
    - using ‘conda-env-mod’

Today's Demo

- **How to access the Analysis facility**
- **Example #1** - Typical Analysis Procedure - standard steps of a CMS analysis, performed locally on single files and single CPU
- **Example #2** - Full-size CMS Physics analysis of large datasets, using GPUs for training a DNN, and DASK for scaling-out to multiple CPUs/computing nodes.
- Purdue Analysis Facility Demo Materials in GitHub
- Analysis Facility Landing Page

Analysis Example

- **Includes main steps of a typical CMS analysis**
  - Read data and MC root files from storage element (*COFFEA* like: using *uproot* and *awkward array*)
  - Perform selection on physics objects
  - Fill and plot histograms using *ROOT* or *matplotlib*
  - Perform event selection (on multiple objects)
  - Compute a new quantity (invariant mass) from existing quantities (using *vector* library)
  - Train a DNN to distinguish between signal and background
  - Use DNN score in event selection and plot final distribution for data and MC
  - Save final plot

Scale-out Techniques

- **From single files to full datasets**
  - Use xrootd protocol to access local (or remote) datasets
  - Read in parallel using DASK
- **Using GPUs**
  - Switch to GPU-enabled version of pytorch, and make use of the local GPU for the training the DNN
- **Parallelize analysis to run on multiple CPUs (cluster)**
  - Split the whole processing in multiple chunks using DASK, and run it in parallel on a small cluster with automatic dynamic scaling of the number of CPUs.

Current status and Future plans

- Currently available at Purdue:

Analysis Facility Platforms			
Resource	**Cluster**	**Access**	**GPUs**
JupyterHub	Hammer	Shared Front-Ends	Yes (T4)
OnDemand	Hammer	Dedicated Nodes	Yes (T4)
	Gilbreth	Dedicated Nodes	Yes (A100, A30, V100, P100)
	Bell	Dedicated Nodes	No
Composable	Geddes	Dedicated Pods	Yes (A100)

- Future plans
  - Will be formulated based on your needs and feedback
  - Consolidate OnDemand into a single entry point for all clusters
  - Permanent DASK queues reservation, Jupyter DASK extension
  - Incorporate AMD GPUs that are becoming available

Summary

- We showed you what the AF at Purdue looks like, and how you can access it to run a simple CMS Physics analysis on it.
- We also demonstrated how you can scale that analysis to run on full datasets in real time, using multiple compute nodes and GPUs for speeding it up.
- The future shape and size of the facility will strongly depend on your needs.
- We want to hear from you
  - Do you find it useful?
  - What additional features you want to see?
  - Is the capacity enough?
- Survey coming soon!

- Special thanks to Amandeep Kaur and Dmitry Kondratyev for preparing the Physics Analysis examples!

- Email us for support at cms-support@lists.purdue.edu

PURDUE UNIVERSITY