DOMA: workshop outcomes and action items

Brian Bockelman

DOMA: **Organization**, **Management**, and **Access**.

Over the last 3 days,

- **Organization**: Typically this is where we have long discussions about Parquet vs ROOT – but it was quiet this time! Instead, the focus was on “column joins”.
- **Access**: The usual suspects: ServiceX, XCache.
- **Management**: Each facility has their own approach:
  - Most have some sort of shared filesystem (EOS, CephFS, etc) for sharing data.
  - Caching (XCache) is also a way to manage production datasets.
  - Working on higher-level languages (Malloy) and mechanisms (SkyHook) is ongoing.

Column Joins

A rollicking discussion on column joins covered different ways to make it work – but few claimed it to be a bad idea!

Proposed Action Items:

- **Minimum useful item** – build a ‘join’ from two trees within the same file. Immediately helpful for the UT-Austin folks doing uncertainties.
- **Simple case** (suggested by Brian):
    - Have two directories, one a “reference” dataset and the other an extra column. Assume they are from compatible datasets.
    - Join the files on Run/Lumi/Event.
    - Throw an error if the ordering of the column differs from the reference dataset. **NO SORTING**.

ServiceX

Two different presentations showed concrete examples of using ServiceX.

“ServiceX leading order time-saver”

Many noted improvements this year – highlight is multiple code generators.

Action Items:

- Logging as a persistent pain. How do we move logs back to the user’s environment?
  - Without training the user how to use ElasticSearch.
  - Without training the user how to use “kubectl logs”.
- Handling multiple systematics trees.
- **September: Give estimates for resources needed**.
- **September: demo higher concurrency of multiple user jobs**.
- September: demo no account creation, output to POSIX.

ServiceX - What should we do for September?

Ideas discussed:

- Measure the ratio of (XCache => ServiceX => Coffea) to (XCache => Coffea) for the same Uproot-based AGC query.
    - **Goal**: Measure the ‘overhead’ of ServiceX in this context (as the AGC queries currently have trivial CPU).
    - Measure again for ServiceX output files already cached.
- xAOD-based demonstrator? We didn’t dive in – but can we define something to highlight the ServiceX strengths as well?
    - ~~Gordon’s immediate answer was “no” in the September timeframe…~~
    - To be clear, a **demo** is possible for xAOD in this timeframe but not make it as part of the tutorial.
- Lindsey: Need help to do NanoEvents for PHYSLITE in dask-awkward in this timeframe.

ServiceX - Is it “ready”

(What is ready anyway?)

We have enthusiastic developer+users that can use ServiceX to do amazing things. Open Question:

What’s missing to have it take over the world?

Short of world domination, how does it get to the point where 5 groups use it for their analysis?

XCache and Friends

Not much discussion specific to XCache during the workshop. A few observations:

- ATLAS and CMS still use XCache in incompatible modes. Is it too late for unification?
- A handful of open HTTP protocol bugs (opened in the last month but long pre-existing) are making HTTP-based access ugly in uproot.

Anything else? Seems XCache was ‘just working’ this year.

Model management

Much discussion around how we manage ML models (e.g., for upload to Triton)

- Is MLFlow ‘the’ training bookkeeping tool? Or a flash in the pan?
- Is there a standardized, reusable way to make Triton multitenant?
  - FNAL has a webapp that solves this – is it ready for everyone else to use it?
- Gordon had good points about keeping model provenance and tracking them long-term.

Is this worth discussing now? Obvious action items?
Or is it too early to try to converge?

Come to the Coffea Shoppe!

The OSG-LHC ensures there's a unified, integrated software stack for HTC services.

- What's the "One stop shop for an analysis facility"?
  - Example: ServiceX is under the "ssl-hep" umbrella, Coffea-Casa is under "CoffeaTeam". **Triton, MLFlow? Go Figure It Out Yourself!**

Idea:

- Let's develop the "**Coffea Shoppe**" stack, covering the integrated software needed to run new analysis facility services.
  - Let Coffea-Casa be an opinionated way to run Coffea; this would also include a collection of other useful services.
  - Target platform would be Kubernetes + Helm Charts.

We need release management! At least –

- CI/CD and end-to-end tests.
- Release announcements.
- Integrated tutorials and videos.
- Support Desk

Badge that shows all these things work together!

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant Nos. 1148698, 1836650 and 2030508. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.