UC SANTA CRUZ

Central N-tuples + Coffea

atlas-schema: a python and conda package

Dr. Giordon Stark

PyHEP.users 2025

October 28th, 2025

indico://e/1566263/c/6736095

Run: 300800

Event: 2418777995

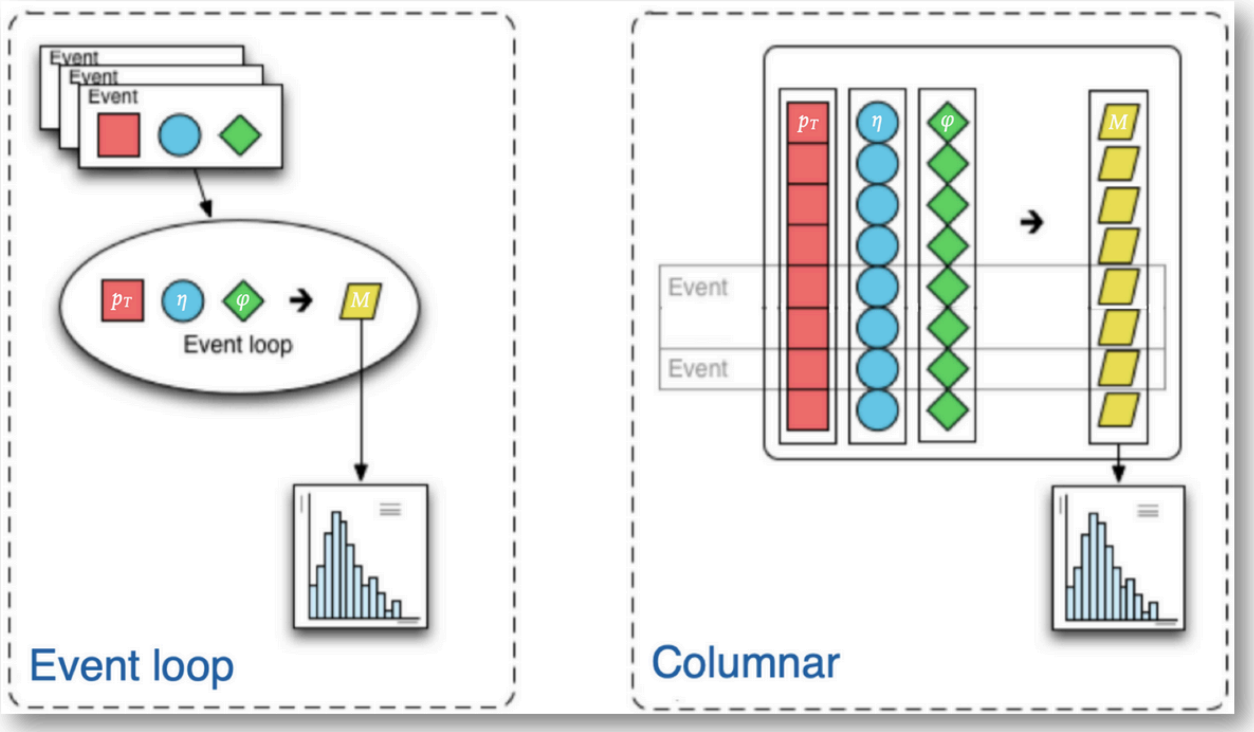
2016-06-04 03:47:03

if you can read this, you're too close

Why/What Columnar?

"columnar analysis" == "array programming for data analysis"

- Higher level APIs for physicists and improved user experience
  - People using columnar analysis on ntuples already seem to be loving it
  - Enable the same UX but without ntupling (save disk)
- Potential for higher performance
  - Enable on-the-fly combined performance (CP) tool corrections on PHYSLITE
- Broader scientific data analysis ecosystem integration
  - Extend and scale ATLAS tools with large and performant ecosystem



Different expressions/representations for same analysis result goals

(Nick Smith, 2019 Joint HSF/OSG/WLCG Workshop) *algs are highly-configurable which is both nice, but also hard to generalize in a schema in a robust fashion

Central ATLAS ntuple

- TTree populated with branch names typically* follow the simple pattern:

```
[a-zA-Z0-9_]+_[a-zA-Z0-9_]+_(NOSYS|[a-zA-Z0-9_]+(?:_1up|_1down)?)
 collection  _   variable   _               systematic
```

- It would be nice to be able to use `coffea` to work with objects that:
  - gives us their systematically-varied kinematic information
    - `jet_pt_NOSYS, jet_pt_JET_EffectiveNP_Detector1__1up, …`
    - `events.Jet.pt_syst.JET_EffectiveNP_Detector1__1up`
    - `events.JET_EffectiveNP_Detector1__1up.Jet.pt`
  - are lorentz-vector-like, and perform vectorized calculations
    - `[sqrt(jeta**2 + …) + sqrt(…) for jęta in jet_eta for weta in el_eta]`
    - `events.Jet.deltaR(events.Electron)`
  - have additional behavior to avoid remembering particular CP group quirks for some (non-kinematic) variables [such as origin or photon's isEM flag]
    - `ph_isEM & (ph_isEM&0x400) == 0`
    - `events.Photon.passes_isEM(atlas_schema.enums.PhotonID.Rhad)`

- and to get the full benefits of using dask, and all other modern tools that we're seeing in data science outside of our field

Enter `atlas-schema`

(credit to Dr. M. Feickert for the name suggestion to be explicit about its role)

- Quick install:
  - `pip install atlas-schema`
  - `conda install atlas-schema`
  - `pixi add atlas-schema`
- Some links:
  - https://github.com/scipp-atlas/atlas-schema/
  - https://github.com/conda-forge/atlas-schema-feedstock

Summary

- Half the battle is getting ATLAS users on-board with columnar analysis, **make it easier**
  - need feedback from users — what works or not
  - could imagine the DAOD-level coffee-schema being included in this package later
- **coffea could use some additional help to get ATLAS-specific functionality** working