Contribution ID: **15** Type: **not specified**

Case for awkward array - efficient I/O for highly hierarchical data structure

We present from our developer experience, a case for adopting awkward array abstraction in data I/O library for highly hierarchical data set commonly as seen in HEP.

We will discuss both technical, performance-related lessons learned in implementing TTree/RNTuple reading, these are universal principles for all columnar data format I/O.

Specifically, we discuss the the importance of minimizing allocation, and lazy materialization due to sparse access pattern in HEP analysis.

Then we move to high-level, design challenges and how "owning" the entire data representation by using awkward array enables efficient / lazy data I/O that otherwise cannot be easily achieved otherwise.

Specifically, we will use the example of accessing "subset of complex data structures" (e.g. only jets.pt when event.jets isa Vector{LorentzVector}) to demonstrate the weakness of conventional approach and how awkward array provides a systematic way to do exact I/O instead of overtouching.

Authors: LING, Jerry ⊠ (Harvard University (US)); MATO, Pere (CERN)

Presenter: LING, Jerry ⊠ (Harvard University (US))