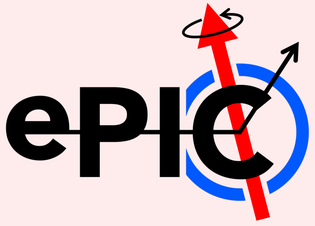
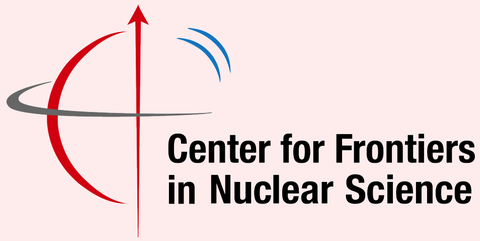
My inclusive physics analysis workflow at ePIC

Win Lin
Stony Brook University

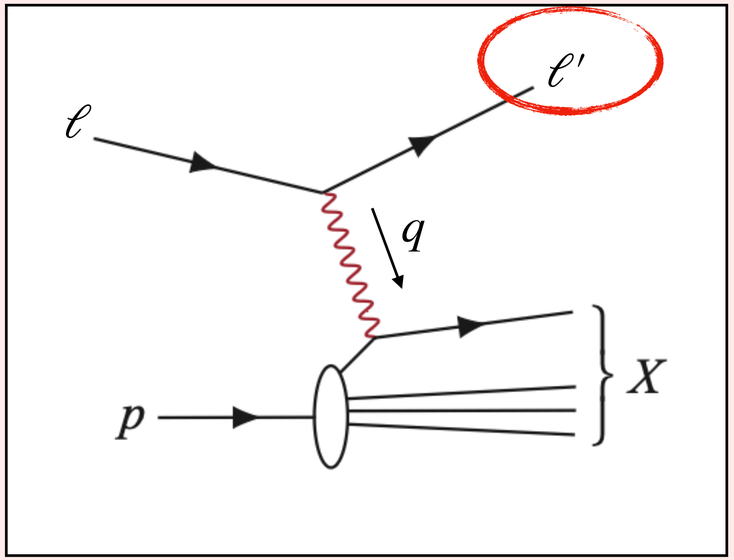
Physics, Software and Computing Discussion
10/28/2026

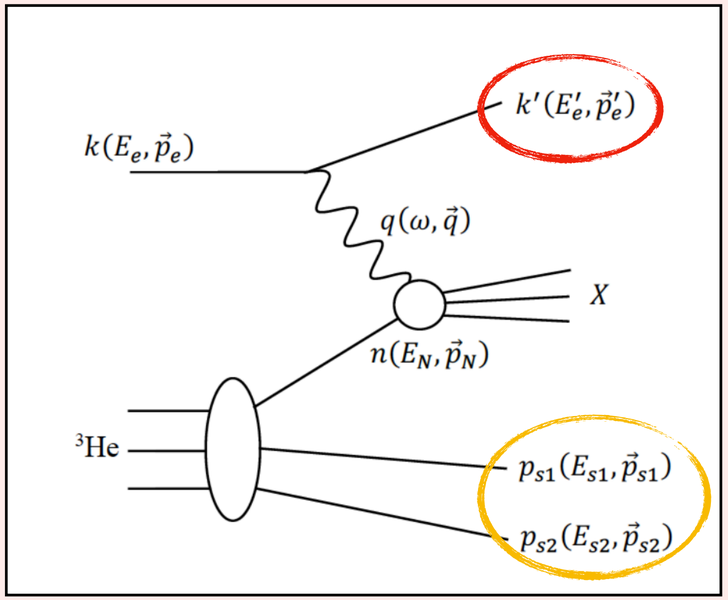






Inclusive physics analysis for ePIC



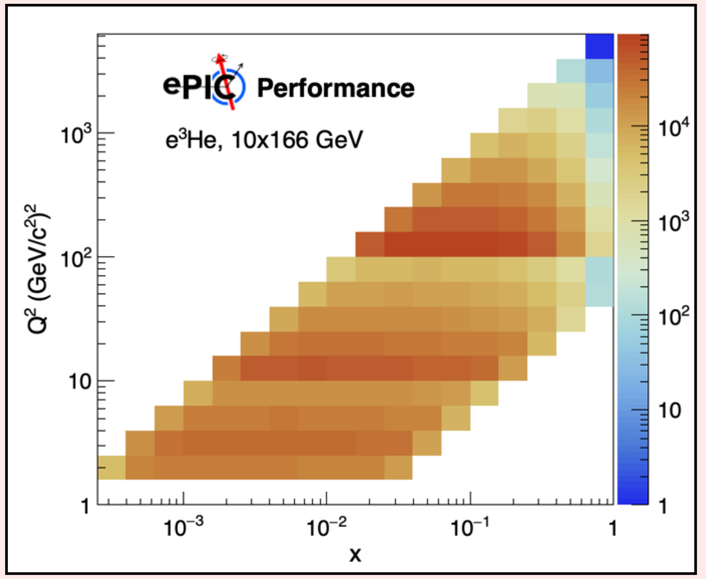


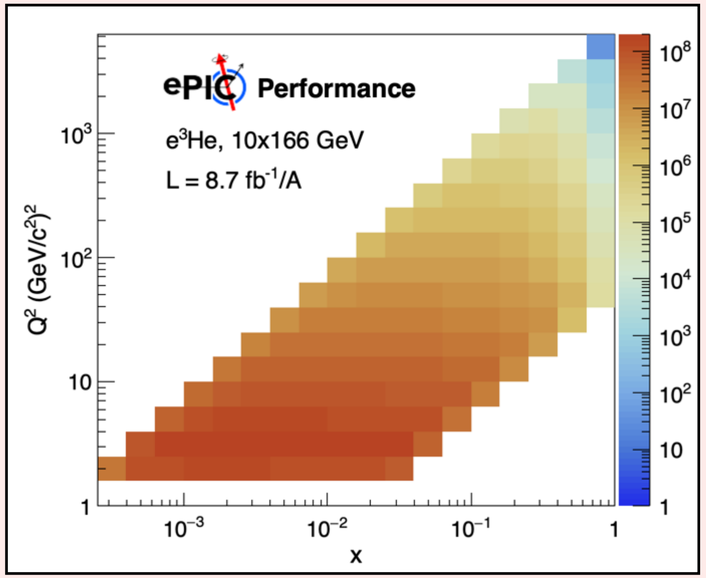
For inclusive analysis, we mainly need to:

- Find scattered electrons
- Reconstruct DIS kinematic: x, Q^2, y, W
  - beam kinematics
  - scattered electron kinematics
  - HFS kinematics for additional reconstruction methods
- Scale events to target luminosity

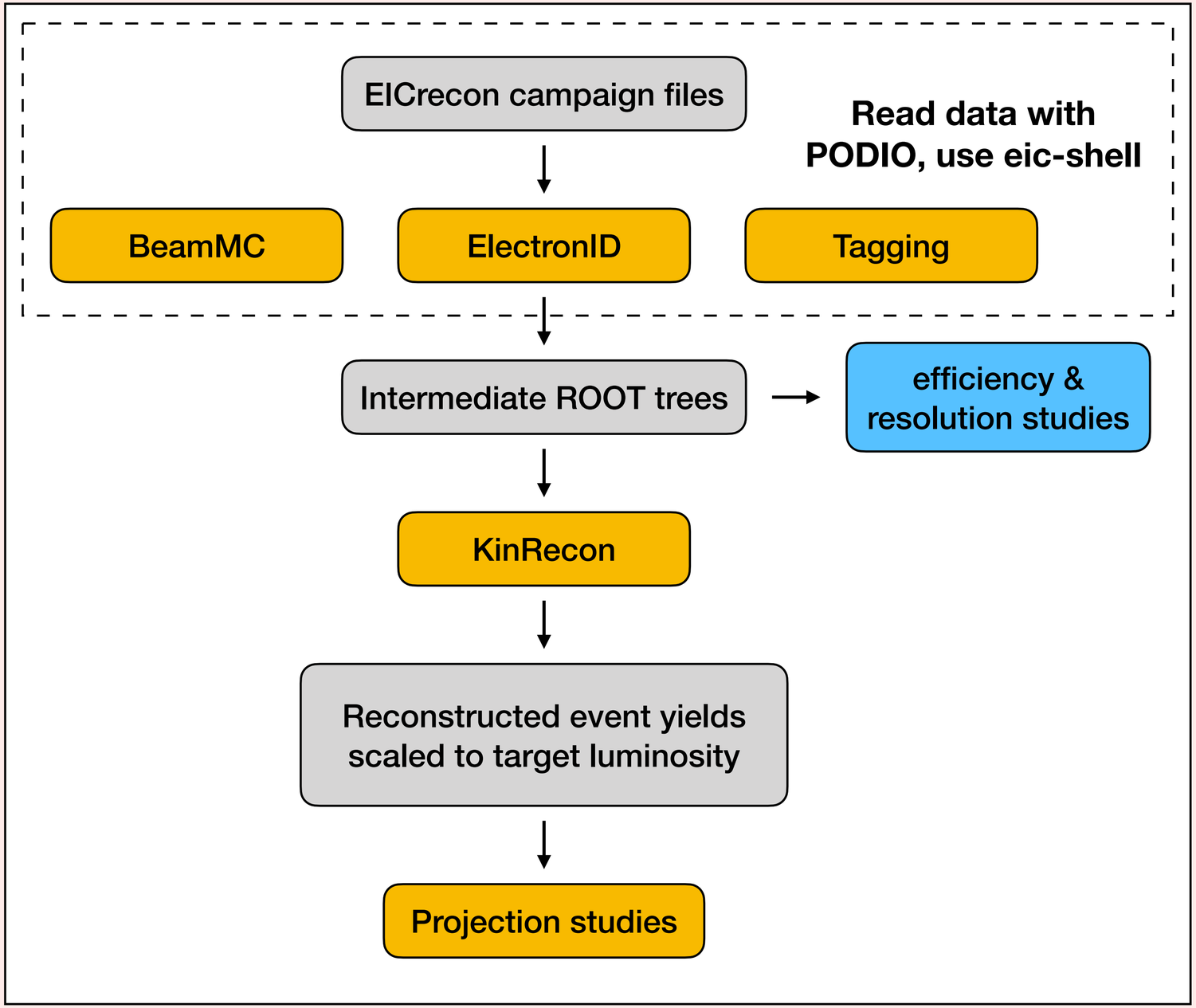
For spectator tagging:

- Identify spectators in RP and OMD





Workflow summary



My analysis code:
https://github.com/Meatbally2/eicDIS

- ROOT based code with C++ classes
- Modular analysis structure:
  - Check performance at different stages
  - Rerun individual steps as needed
- Test code on small samples, then process full datasets at BNL SDCC

- Shared global util:
  - AnaManager: manage sim. input
    - Search XRootD storage for files matching a given path pattern
    - Filter out corrupted files
  - HistManage: manage hist. and unify style
  - luminosityTable: numbers for event norm.

Some challenges: software

- Analysis relies on multiple libraries and data models (ROOT, PODIO, EDM4eic, EICrecon)
  - Need to manage headers, libraries, and compatible software versions
- Analysis code depends on the eic-shell environment
  - Updates to upstream libraries or data models can break existing code
  - Sometimes difficult to tell whether a crash comes from my code or a software compatibility issue

- Difficult to figure out what information is available in various ROOT branches, and how different groups perform their analysis

Example of what I need to load for eID analysis

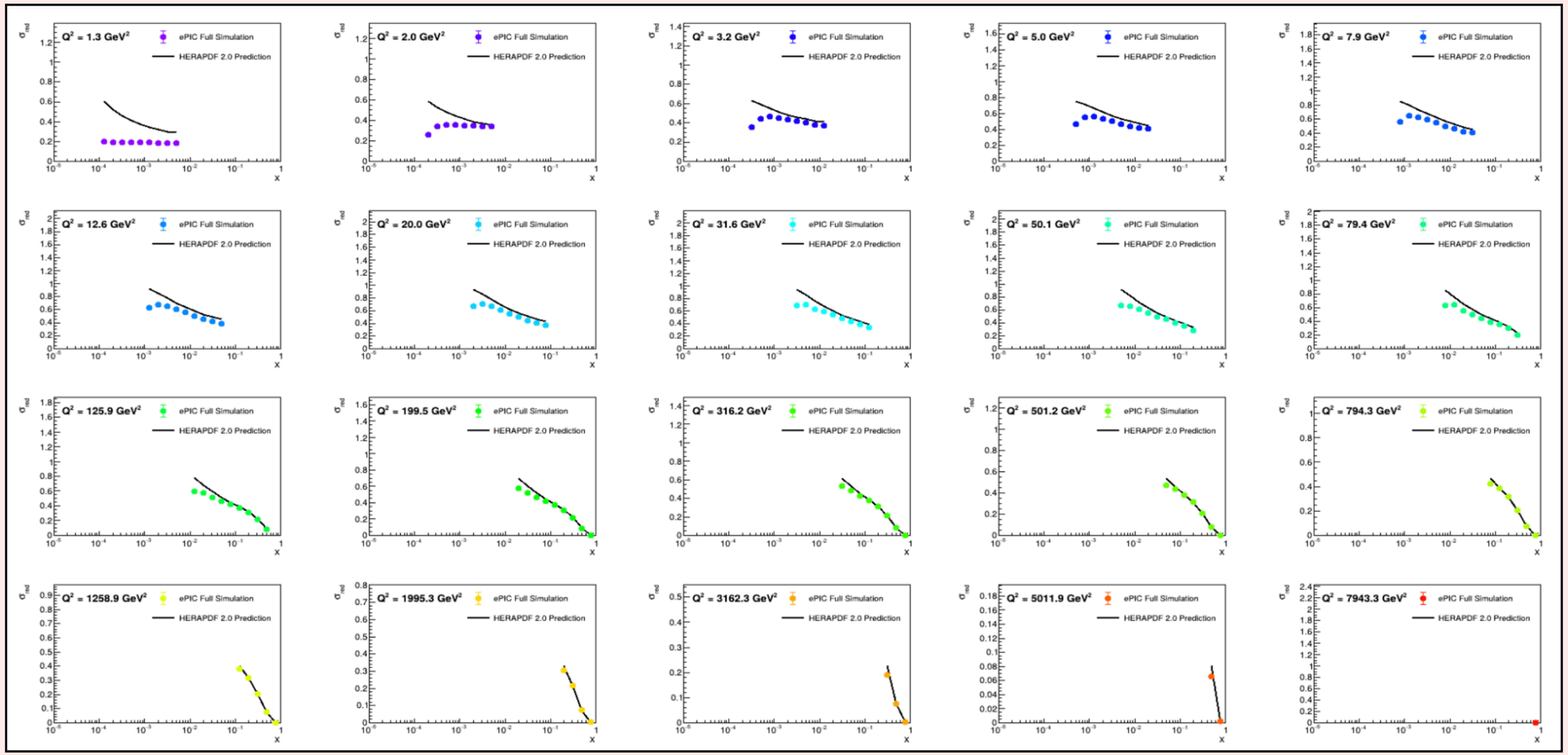
```
R__LOAD_LIBRARY(libpodio.so)
R__LOAD_LIBRARY(libpodioRootIO.so)
R__LOAD_LIBRARY(libedm4hep.so)
R__LOAD_LIBRARY(libedm4hepDict.so)
R__LOAD_LIBRARY(libedm4hepUtils.so)
R__LOAD_LIBRARY(libedm4hepRDF.so)
R__LOAD_LIBRARY(libedm4eic.so)
R__LOAD_LIBRARY(libedm4eicDict.so)
```

```
#include "podio/Frame.h"

#include "edm4eic/ReconstructedParticleCollection.h"
#include "edm4eic/MCRecoParticleAssociationCollection.h"
#include "edm4hep/MCParticleCollection.h"
```

Some challenges: simulation

- Need multiple beam configurations, generators, and reconstruction campaigns.
  - Difficult to track sample availability and completeness
- MC particle ancestry and generator level information is not all preserved. Difficult to identify the physical origin of particles
- Known issue with Pythia 8 cross section at low Q^2



Analysis by Stephen M.

Ideas for future analysis work

- Moving towards vibe coding ..
  - AI working speed is much much faster. AI coding capabilities are improving rapidly
  - Less time on figuring out library dependencies, software installation, ROOT trees, and branch structures
  - More time on defining analysis requirements, physics interpretation, and validating results

- Perhaps provide ePIC analysis skills and guidelines for AI
  - How to access campaign files, read data, use recommended software environment
  - Recommended analysis procedures and validation procedures

https://claude.com/programs/team-plan-for-scientists

