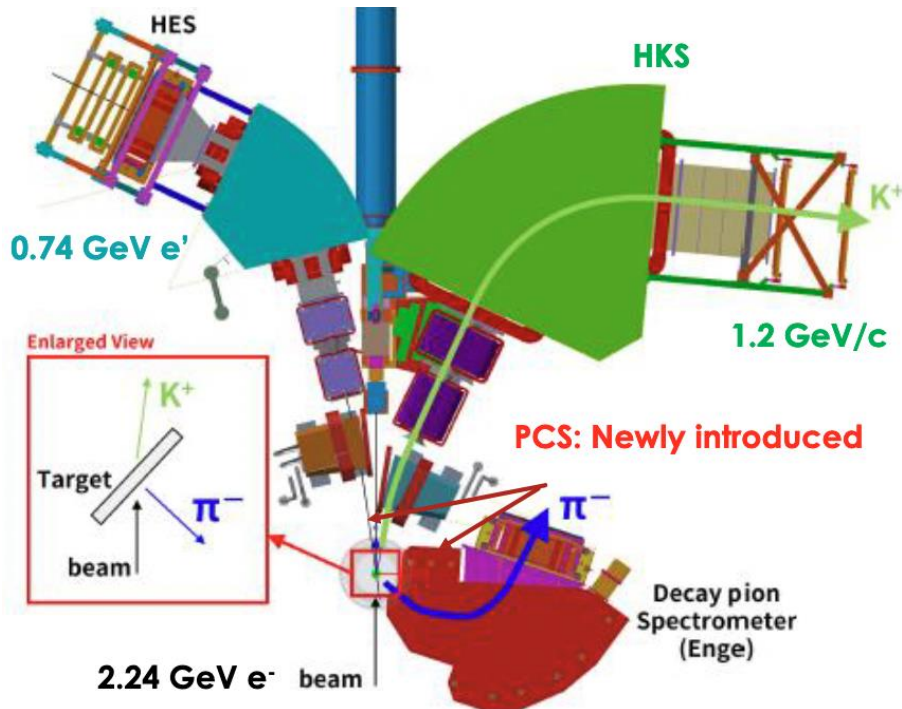


Hypernuclear Analysis Software Update

Sanghwa Park
Jefferson Lab

May 16, 2025

Experimental Setup



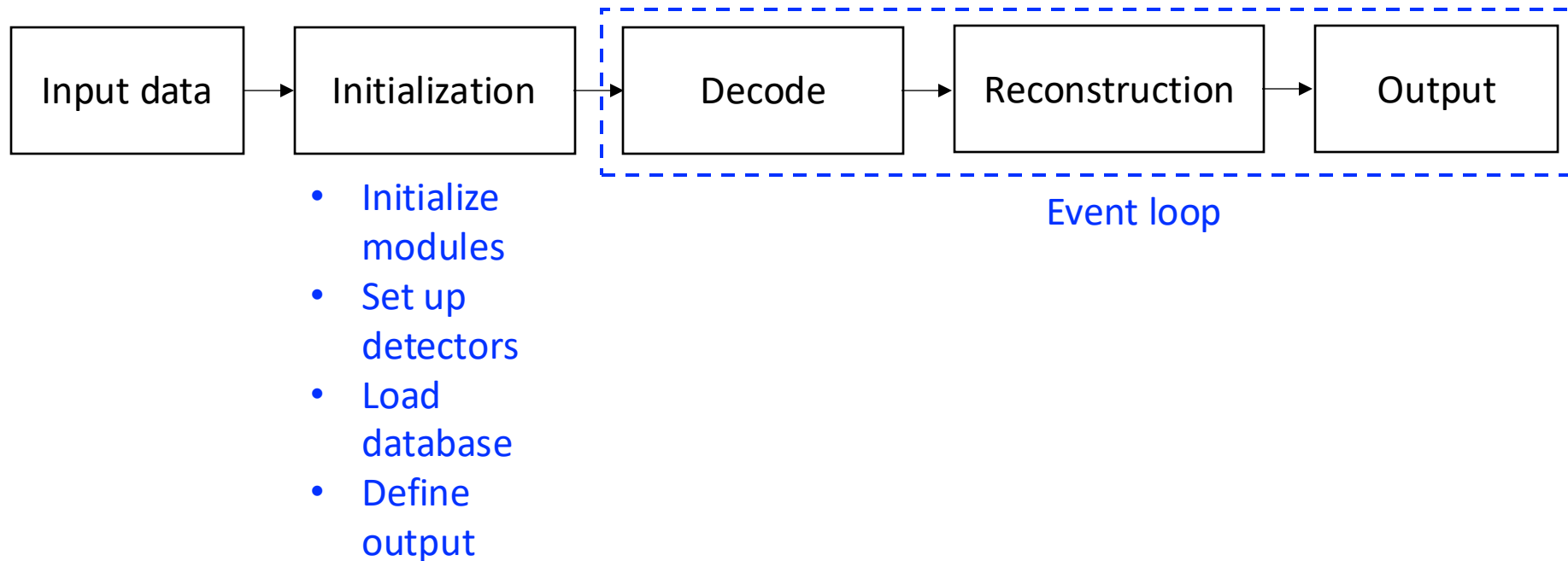
- Spectrometers and Detector subsystems:
 - HKS/HES/ENGE spectrometers
 - Drift Chamber, Cherenkov (water, aerogel), Plastic scintillator (TOF) detectors; **all commonly used detector types in Hall C**
- Readout electronics used for previous Hall A/C experiments
 - **Raw data decoding and base software exist**

Software framework (Online/Offline Replay)

- Previous hypernuclear experiments (E01-011, E05-115) used the old fortran based Hall C analysis software package
- HallA analyzer (ROOT/C++) based online and offline analysis software is being developed
- <https://github.com/JeffersonLab/HYPAnalyzer>
 - Largely taking advantage of already existing decoders and detector modules
 - Started with scintillator detector (to be used for EHodo and TOF detectors). Testing with simulation interface.

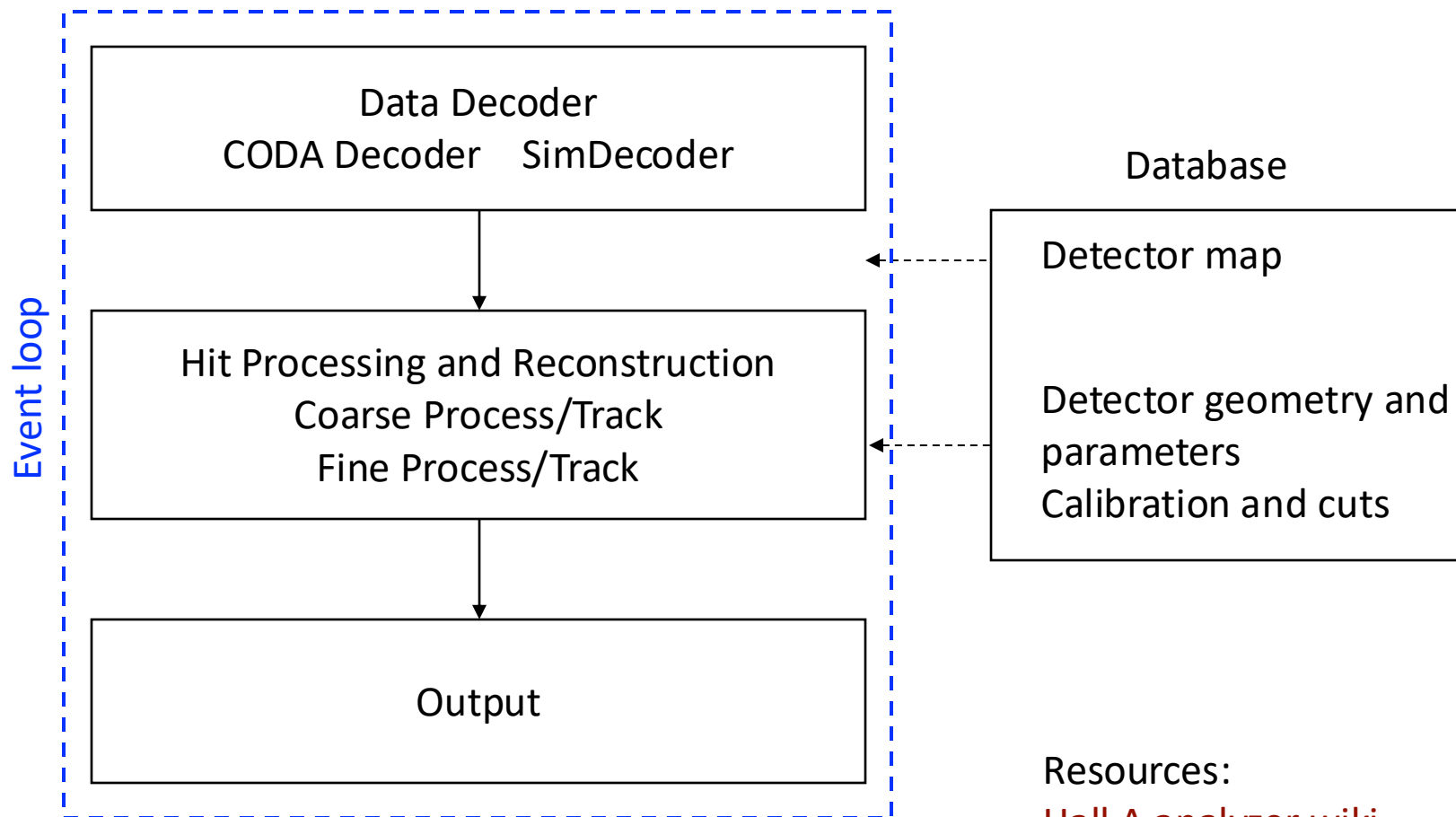
Software framework (Online/Offline Replay)

- General workflow of the analyzer



Software framework (Online/Offline Replay)

- General workflow of the analyzer



Resources:

[Hall A analyzer wiki](#)

Analysis Workshop ([2018](#), [2017](#))

Simulation interface

- Implement the simulation interface used for SBS experiment
 - Allow to use simulation data as an input in the analysis chain
 - Need digitization
- THcSimFile: THaRunBase. Use simulation output ROOT file to fill the event object (THcSimEvent) instead of CODA data
- THcSimDataDecoder: Provide sim adc/tdc data structure, Encoder/Decoder
- THcSimDecoder: Unpack the event object into slot data
- THcSimADC, THcSimTDC: Simulated ADC and TDC modules

- THcSimADC.cxx
- THcSimADC.h
- THcSimDataDecoder.cxx
- THcSimDataDecoder.h
- THcSimDecoder.cxx
- THcSimDecoder.h
- THcSimEvent.cxx
- THcSimEvent.h
- THcSimFile.cxx
- THcSimFile.h
- THcSimTDC.cxx
- THcSimTDC.h

Software framework (Monitoring and Calibration)

- Online monitoring:
 - Current Hall A/C experiments using penguin (<https://github.com/JeffersonLab/penguin>)
 - Hydra (AI based data monitoring): Deployed in all halls (actively used in Hall B and D). Initial deployment done for Hall C during NPS; Use histograms from penguin output as input.
- Calibration:
 - To be developed
 - Detector calibration: Benchmark existing codes from Hall A/C analysis
 - Spectrometer calibration:
 - Important to achieve the desired energy resolution.
 - Calibration method and procedure well established from the previous experiments
- Databases:
 - Run database: RCDB successfully implemented for Hall C experiments. Will have an interface with replay software
 - Calibration database: Plan to implement CCDB for Hall C experiments

Software tasks

- Software base:
 - Decoders: All planned readout were already used at JLab, decoders existing and would only need no/minor modification (mostly event format if any)
 - Spectrometer and detector modules, etc:
 - Spectrometers: basic class outline. Need to have track finding and reconstruction.
 - Detectors: we have the initial version of the scintillator detector classes (for Hodoscope and TOF). Other detectors have only a rough class outline at this moment, but we can use hcana modules as reference.
- Pseudodata and data from test bench for software testing:
 - Using Geant4 simulation: ready to test for Scintillator detectors, will extend the application to other detectors as well
 - Simulation output format needs to be updated
 - Using test bench data

Software tasks

- Detector geometry:
 - Need to collect the detector geometry information (ultimately we want the geometry parameters in the DB files)
 - Resources:
 - <https://doi.org/10.1016/j.nima.2018.05.042>
 - http://lambda.phys.tohoku.ac.jp/~db/human_resource/thesis/2008_M_2_D_1.pdf
 - http://lambda.phys.tohoku.ac.jp/~db/human_resource/thesis/2001_B_1_D_1.pdf
 - Workflow:
 - Set up the initial version of all detector modules (at least having raw hit processing)
 - Have them tested with pseudodata/data from test bench
 - Add/update coarse/fine process of hits
- Good starting point for user contribution

Timeline and milestones, personnel

Phase 1 (2025)

- Develop data replay software
- Test with pseudo data, bench test data

Phase 2 (2026)

- Have a mature replay software
- Develop calibration, monitoring codes
- Documentation

Phase 3 (2027-)

- Setting up analysis, computing environment at CH, ready for the experiment

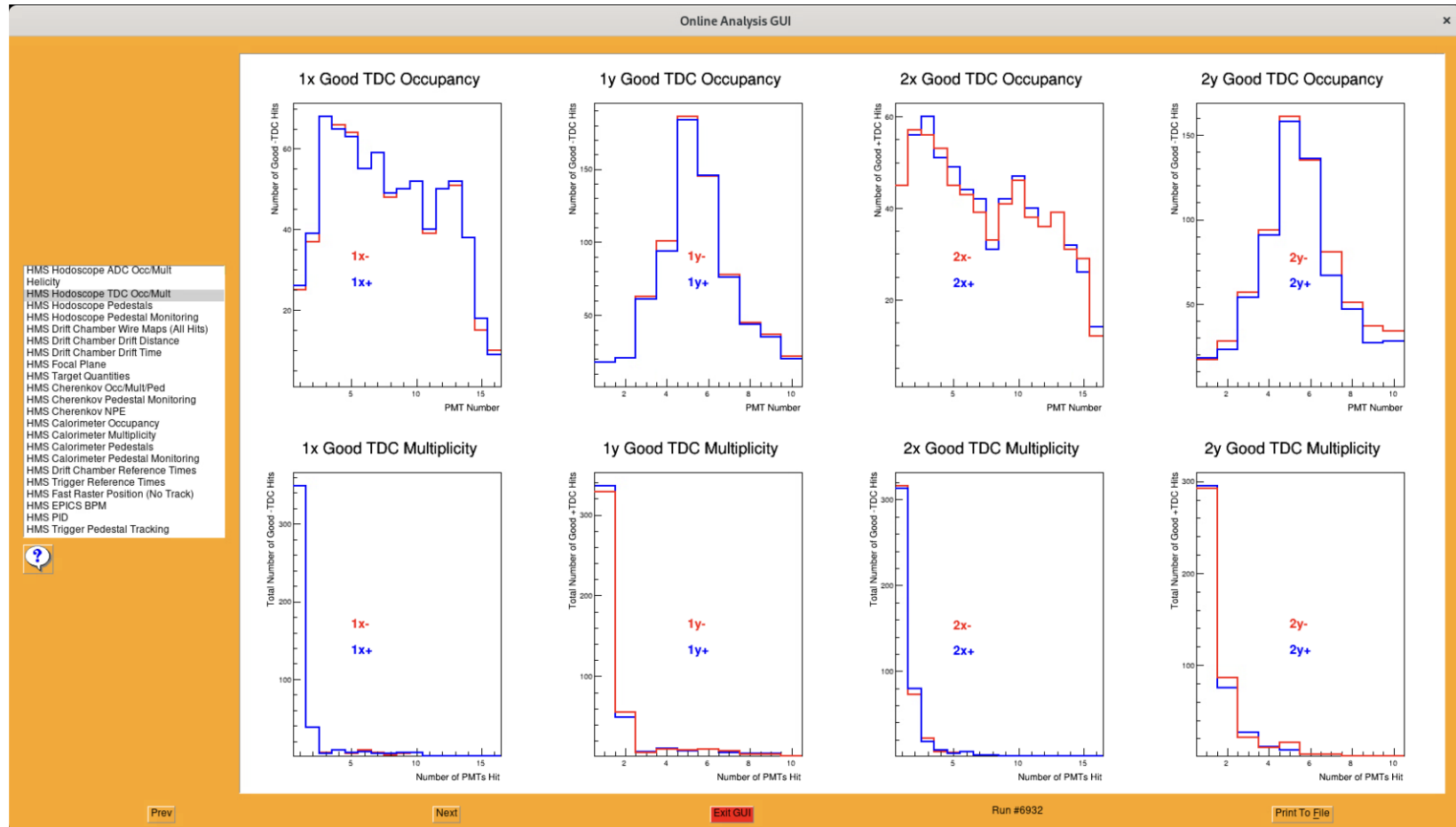
- Plan to get the replay software developed and tested by the first quarter of 2026 – this still looks reasonable to achieve
- Some of the development, such as physics analysis codes and calibration will be done in parallel.
- Personnel: S. Park (JLab), M. Jones (JLab), O. Hansen (JLab), T. Gogami (Kyoto), T. Iwamoto (Kyoto), T. Ishige (Tohoku), R. Kumaragamage (Hampton), .. and more.

Summary

- Online/Offline replay software for hypernuclear experiments is being developed
- Similar equipment and analysis technique from the previous experiments are used
 - Experience and references exist for new software development
- Taking advantage of the existing framework
 - Based on Hall A analyzer
 - Using simulation output for testing it, implemented the simulation interface used for SBS.
 - Could also utilize the data from test bench
- Replay example and resources were sent to the collaboration. Will start more regular discussion/meeting.

Backup

Panguin online monitoring tool



HallC Run Database

