Software Update for Hypernuclear experiment

Sanghwa Park

Jefferson Lab

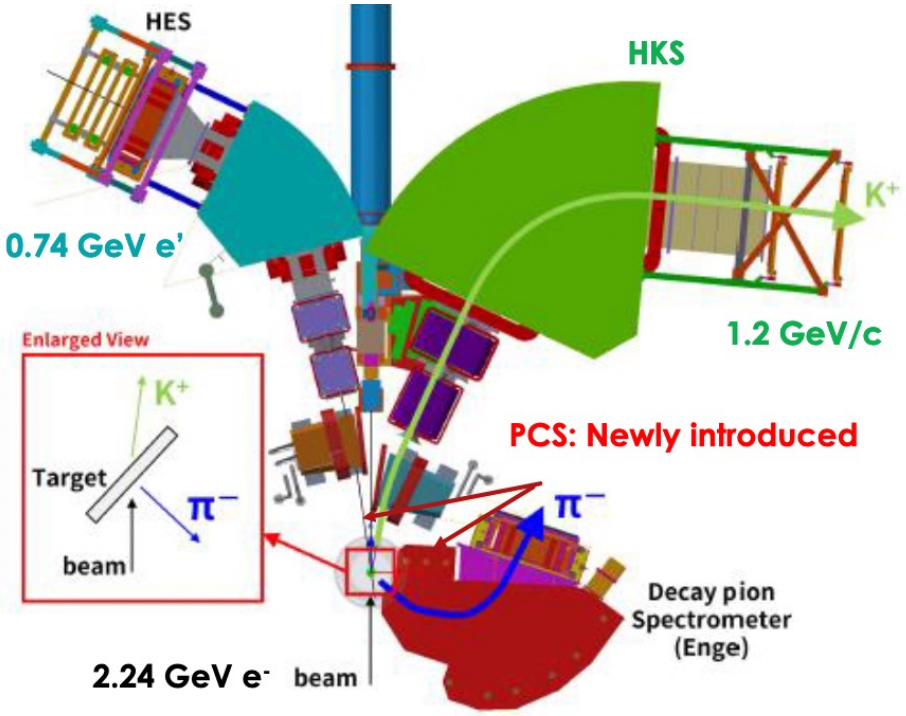
Feb. 12, 2026

Software requirement for Hypernuclear experiments

- The upcoming Hall C hypernuclear experiments need analysis software to replay data
  - Previous hypernuclear experiments (E01-011, E05-115) used the old fortran based Hall C analysis software package
  - Started developing the new software based on newer version of Hall A and Hall C analyzers (ROOT/C++)
  - Largely utilizing already existing codes

- Online monitoring tools for shifters
  - panguin has been used at Hall A/C for online monitoring
    - Framework already exists, no development work needed
  - Hydra provides well structured, easy online monitoring platform (also ML/AI support for data quality monitoring):
    - NPS deployment: https://epsciweb.jlab.org/hallc_offline_hydra/plotBrowser
    - Support provided from EPSCI gropu, but need dedicated personnel from collaboration for labeling

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

Detectors and Readout

Spectrometer	Detector	FADC (ADC/TDC)	Low resolution TDC	High resolution TDC
HKS	KDC	-	640+640	
	TOF	88	-	88
	AC	42	-	-
	WC	48	-	-
HES	EDC	-	1120	-
	TOF	116	-	116
Number of total channels		294	2400	204

Spectrometer	Detector	ADC	TDC
ENGE	Fiber tracker	-	832
	Drift chamber	-	360
	Timing counter	-	96
Number of total channels		0	1416

- ADC/TDC: FADC250
- TDC: CAEN1190, VETROC, VfTDC
- Readout electronics used for previous Hall A/C experiments
  - All decoders are implemented

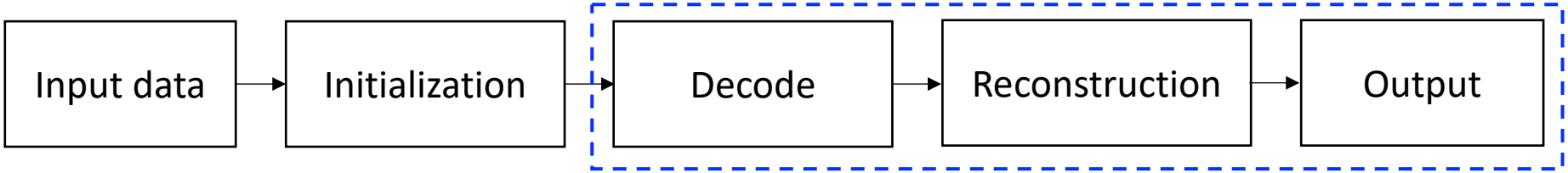
Software framework (Online/Offline Replay)

- Hall A analyzer(https://github.com/JeffersonLab/analyzer/):
  - ROOT/C++ based online and offline analysis package
- hcana (https://github.com/JeffersonLab/hcana):
  - Hall A analyzer (Podd) based software for standard HMS and SHMS spectrometers and detectors.
  - Many parts are written based on previous Hall C analyzer ENGINE (fortran).

- Software for Hypernuclear experiment:
  - https://github.com/JeffersonLab/HYPAnalyzer
  - Taking advantages of existing hcana, Podd support
  - Decoders: FADC250, VETROC, VfTDC
  - Detector types: Cherenkov, Drift Chamber, TOF, Sci Fiber

Workflow (Online/Offline Replay)

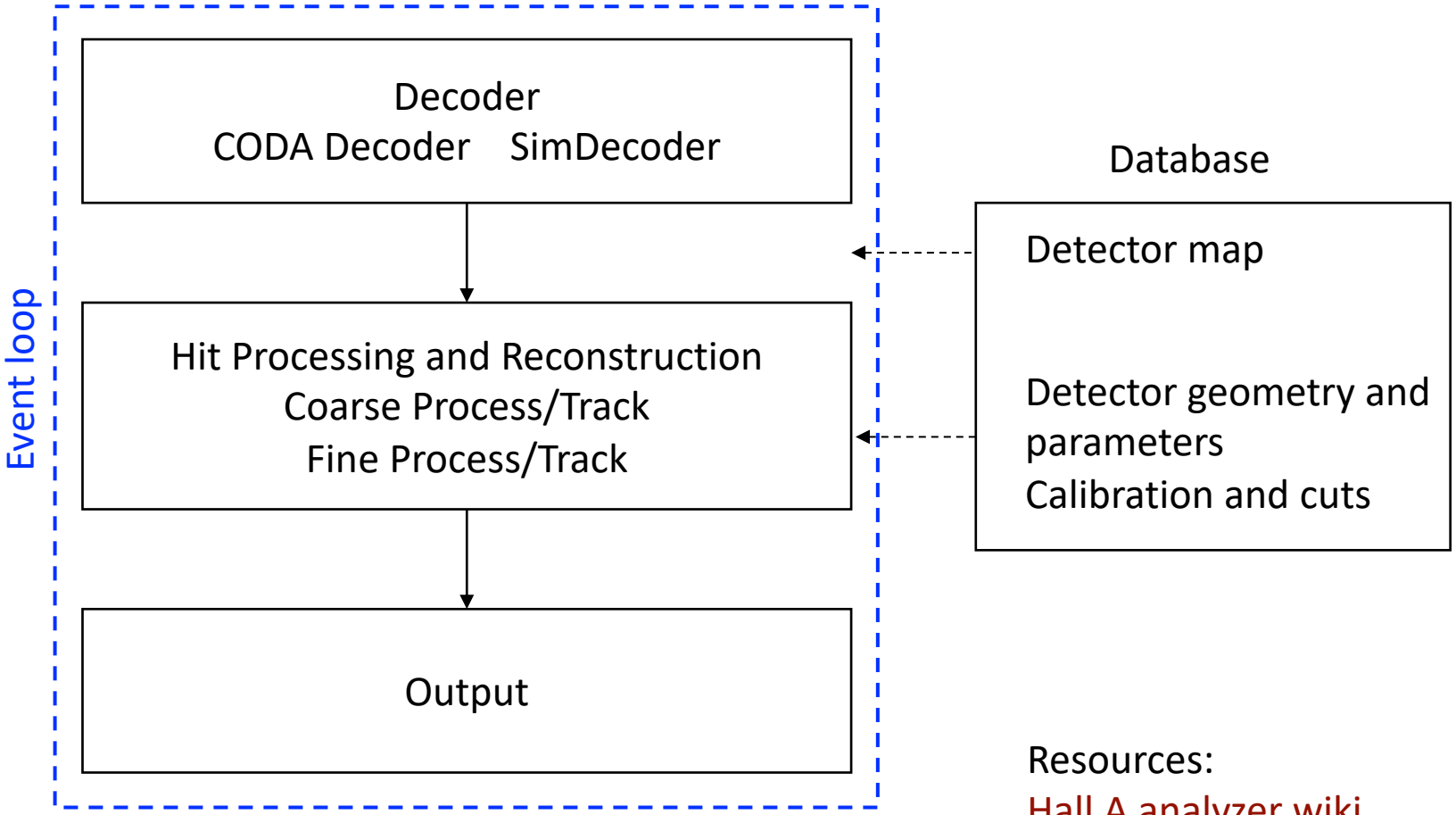
- General workflow of the analysis software



- EVIO file
- Initialize modules
- Set up detectors
- Load configuration and parameters
- Define output vars
- Decode raw signals, add to raw hit list
- Look through reference channels for reference time
- Event loop
- Calling CoarsePro cess/FineP rocess
- ROOT TTree, Histogra ms
- REPORT OUTPUT
- .def files to handle output variables

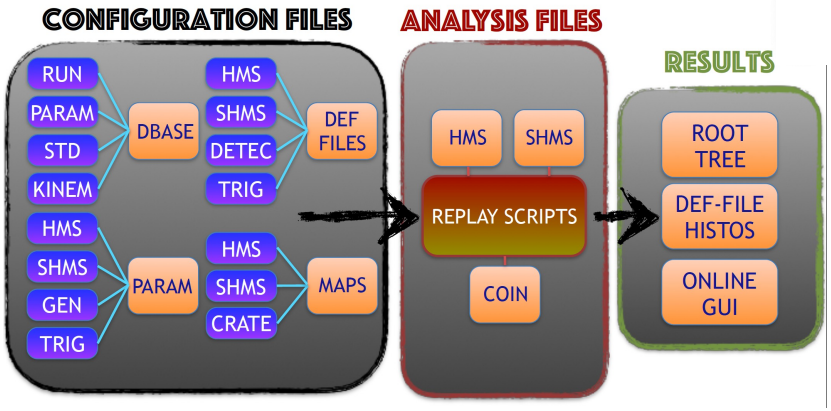
Worlflow (Online/Offline Replay)

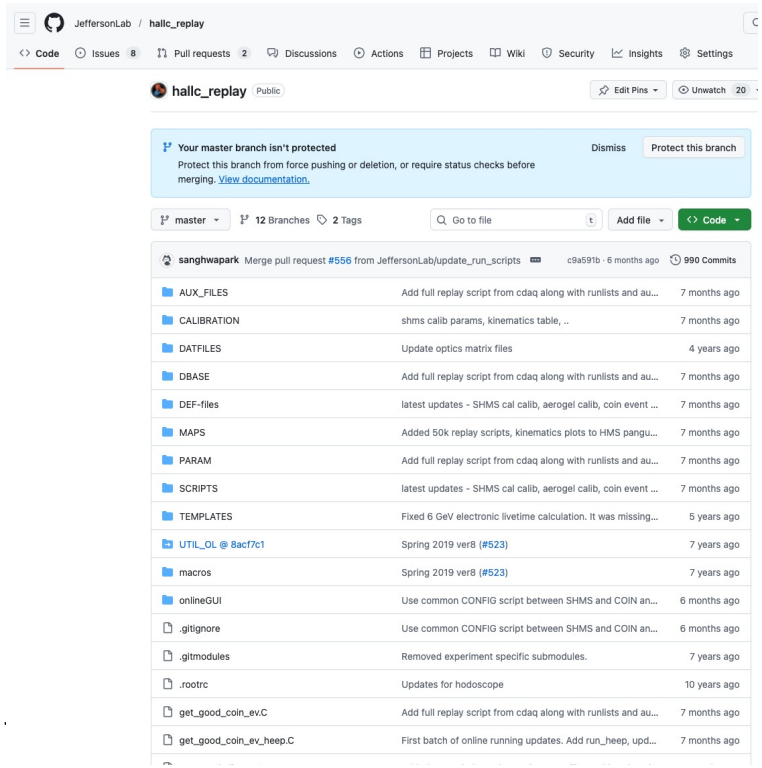
- General workflow of the analyzer



Resources:
Hall A analyzer wiki
Analysis Workshop (2018, 2017)

Replay workflow (traditional Hall C format)

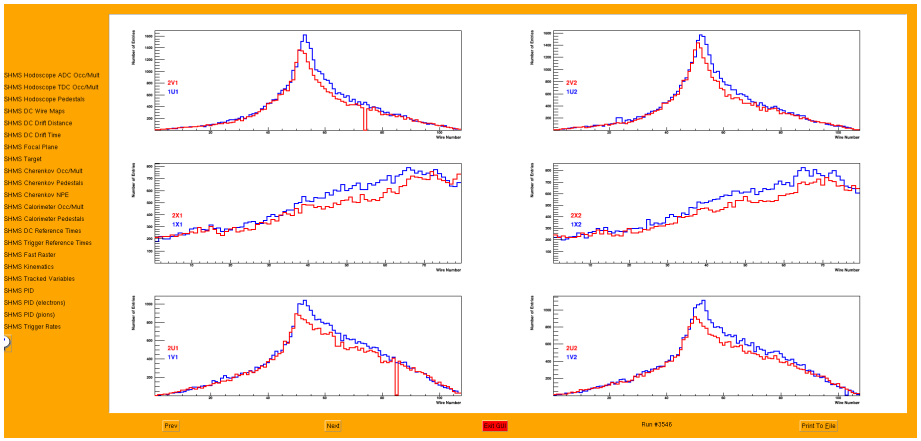


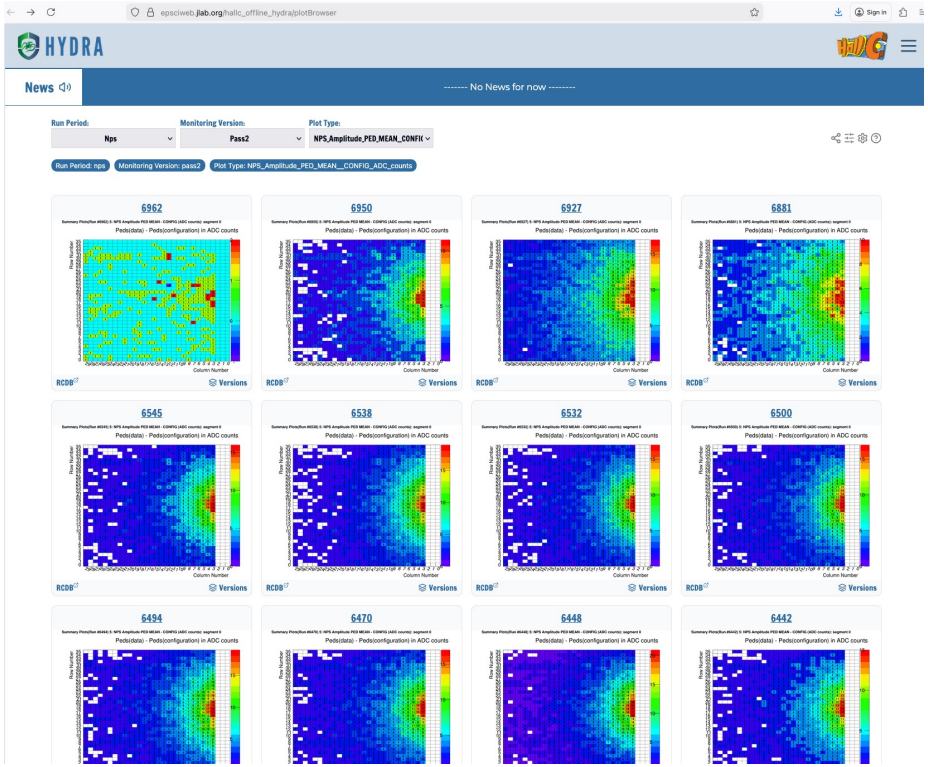


- Database files
  - DATFILES: spectrometer matrix elements
  - DBASE: standard kinematics, database file list to include
  - MAPS: Crate and detector maps
  - PARAM: parameter files for all detectors (geometry, cuts, calibration, reconstruction parameters)
  - DEF-files: output tree variables, histograms
  - TEMPLATES: template files for REPORT OUTPUT

Worlflow (Online/Offline Replay)

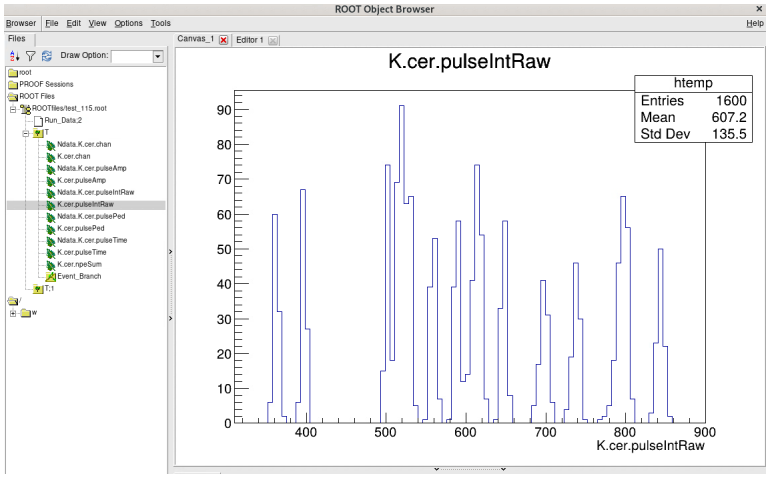
- Online monitoring:
  - panguin (https://github.com/JeffersonLab/panguin)
  - Hydra (AI supported data monitoring)

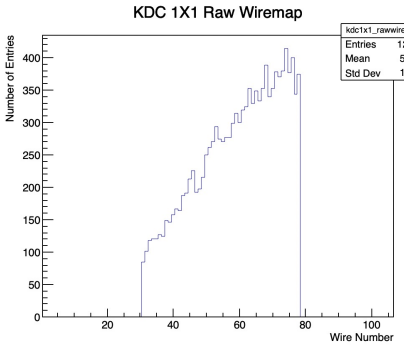


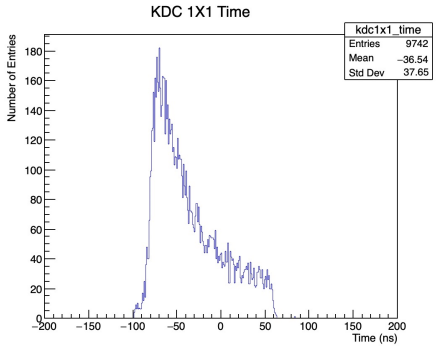


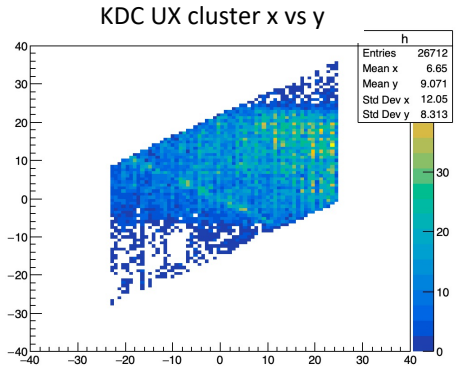
Jefferson Lab

Test with cosmic data

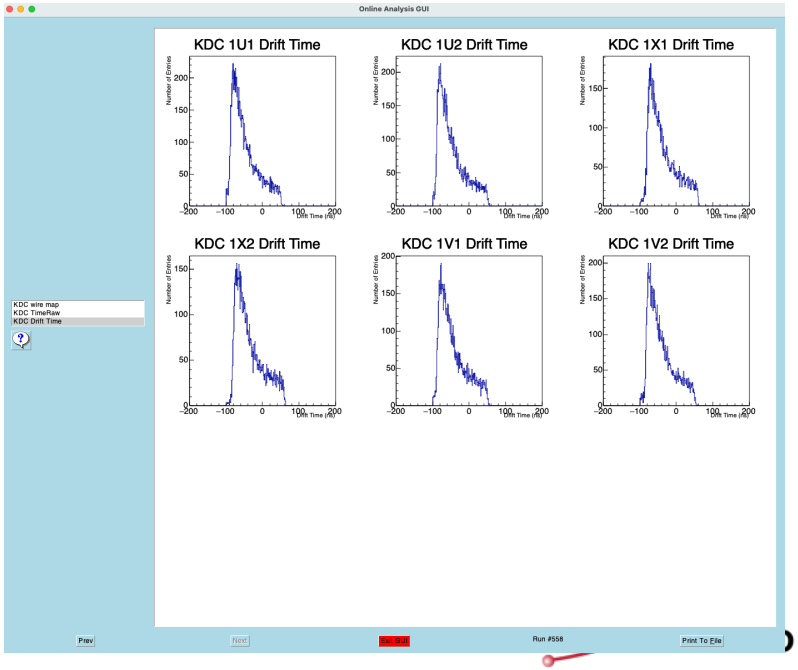








- Test replay outputs with ESB cosmic data:
  - Cherenkov, TOF: raw signal checked
  - DC: Testing/checking reconstruction output
- Using panguin to plot variables are straightforward
  - panguin –f <configfile.cfg> -r <runnumber>



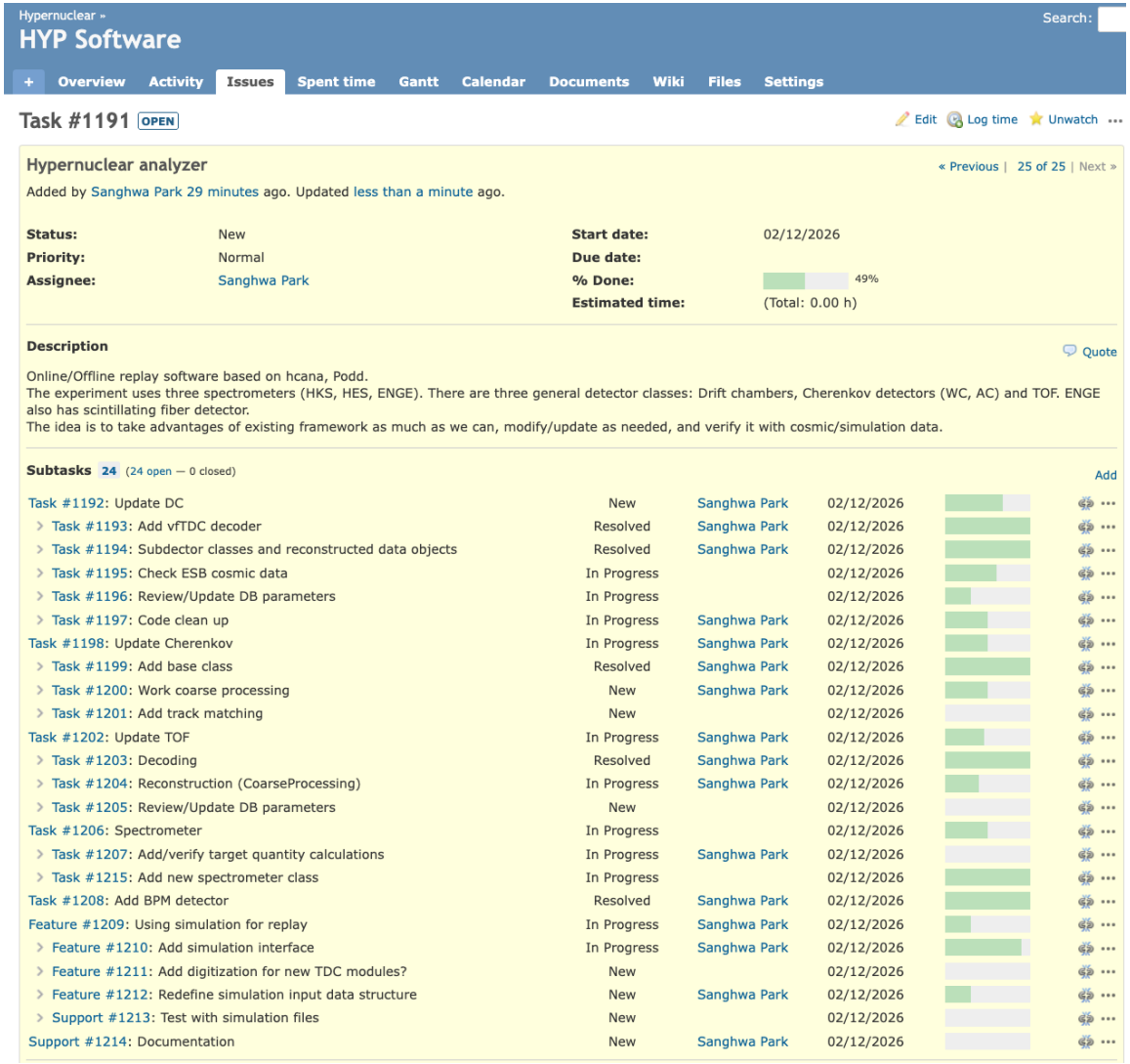
Software tasks

- **DC:**
  - Add vfTDC decoder (done)
  - Subdetector classes, hit/cluster objects based on hcana (done)
  - Check ESB test data (in progress)
  - Code clean up (in progress)
  - Review/Update DB parameters (geometry, cuts)
- **Spectrometer:**
  - Add new spectrometer class for hypernuclear (done)
  - Add/verify target quantity calculations, start with what we have in hcana (in progress)
- **Cherenkov:**
  - Add base class to handle decoding and basic hit processing (done)
  - Reconstruction (good hit selection and npe calculation, track matching)
  - Review/Update DB parameters (geometry, cuts)
- **TOF:**
  - Implement FADC250 and vfTDC decoding, checked with cosmic (done)
  - Reconstruction: Hit processing and track matching (in progress)
  - Review/Update DB parameters (geometry, cuts)

Software tasks

- **Beamline:**
  - Raster analysis module (already exist in hcana)
  - Implement BPM module to process with FADC signal (done)
  - Test with the data from upcoming run
- **Using simulation data in replay**
  - Implement support framework (done)
  - Define simulation input data structure (in progress)
  - Test with simulation files
- **Documentation**

Software tasks



Hypernuclear »

HYP Software

Search:

+ Overview Activity Issues Spent time Gantt Calendar Documents Wiki Files Settings

Task #1191 OPEN

Edit Log time Unwatch ...

Hypernuclear analyzer

« Previous | 25 of 25 | Next »

Added by Sanghwa Park 29 minutes ago. Updated less than a minute ago.

Status:	New	**Start date:**	02/12/2026
Priority:	Normal	**Due date:**	
Assignee:	Sanghwa Park	**% Done:**	49%
		Estimated time:	(Total: 0.00 h)

Description Quote

Online/Offline replay software based on hcana, Podd.
The experiment uses three spectrometers (HKS, HES, ENGE). There are three general detector classes: Drift chambers, Cherenkov detectors (WC, AC) and TOF. ENGE also has scintillating fiber detector.
The idea is to take advantages of existing framework as much as we can, modify/update as needed, and verify it with cosmic/simulation data.

Subtasks 24 (24 open — 0 closed) Add

Task #1192: Update DC	New	Sanghwa Park	02/12/2026
> Task #1193: Add vfTDC decoder	Resolved	Sanghwa Park	02/12/2026
> Task #1194: Subdector classes and reconstructed data objects	Resolved	Sanghwa Park	02/12/2026
> Task #1195: Check ESB cosmic data	In Progress		02/12/2026
> Task #1196: Review/Update DB parameters	In Progress		02/12/2026
> Task #1197: Code clean up	In Progress	Sanghwa Park	02/12/2026
Task #1198: Update Cherenkov	In Progress	Sanghwa Park	02/12/2026
> Task #1199: Add base class	Resolved	Sanghwa Park	02/12/2026
> Task #1200: Work coarse processing	New	Sanghwa Park	02/12/2026
> Task #1201: Add track matching	New		02/12/2026
Task #1202: Update TOF	In Progress	Sanghwa Park	02/12/2026
> Task #1203: Decoding	Resolved	Sanghwa Park	02/12/2026
> Task #1204: Reconstruction (CoarseProcessing)	In Progress	Sanghwa Park	02/12/2026
> Task #1205: Review/Update DB parameters	New		02/12/2026
Task #1206: Spectrometer	In Progress		02/12/2026
> Task #1207: Add/verify target quantity calculations	In Progress	Sanghwa Park	02/12/2026
> Task #1215: Add new spectrometer class	In Progress		02/12/2026
Task #1208: Add BPM detector	Resolved	Sanghwa Park	02/12/2026
Feature #1209: Using simulation for replay	In Progress	Sanghwa Park	02/12/2026
> Feature #1210: Add simulation interface	In Progress	Sanghwa Park	02/12/2026
> Feature #1211: Add digitization for new TDC modules?	New		02/12/2026
> Feature #1212: Redefine simulation input data structure	New	Sanghwa Park	02/12/2026
> Support #1213: Test with simulation files	New		02/12/2026
Support #1214: Documentation	New	Sanghwa Park	02/12/2026

https://redmine.jlab.org/issues/1191

ERR recommendation

a. The reconstruction and analysis software used by the collaboration for the past experiments will have to be ported to the software framework used currently by Hall C. The work has just started. While contributors to this task are listed, it is unclear how much work will be involved in FTEs and whether this would be compatible with the other commitments these contributors may have.
b. No estimate of the computing resources (storage and CPU) required for offline simulations, reconstruction and analysis. Traditionally, Hall C experiments are the smallest consumers at JLab but, since the experiment involves new detector packages and requires high precision, it would be advisable to provide an estimate to ensure this is compatible with the available resources.

Recommendations:

Provide a realistic estimate of the required FTEs and the identified manpower as well as the corresponding timeline with milestones.

Milestones, timeline, personnel

- Milestones:
  - Have a mature replay software --- Fall 2026
  - Integrate simulation data in analysis chain --- End of 2026
  - Documentation --- End of 2026
  - Setting up analysis, computing environment at CH --- End of 2027
    - Most of software already installed on cdaq machines and maintained
    - replay scripts for shift users, instruction

- Personnel:
  - Analysis software development:
    - Sanghwa Park (JLab, 0.4 FTE in 2026)
    - Ken Nishida (Tokyo)
  - Podd/general support: Ole Hansen (JLab), Mark Jones (JLab)
  - Geant4 simulation: Jin Takahashi (Tokyo)

Summary

- Making good progress to have the software ready
- Decoding and hit processing have been tested with cosmic data for all readout
- Working toward completing reconstruction parts
- Continue working with students to verity the reconstruction output with data
- Online monitoring framework already available; no development work needed, but list of plots and config files need to be prepared

Backup

Software environment

- JLab scientific computing: GSPDA computing bootcamp has nice instruction slides https://indico.jlab.org/event/954/

- Setting up analysis:
  - On ifarm:
    - Using modules:
      - /group/halla/modulefiles: ROOT, analyzer, hcana, ..
      - cvmfs: no hcana yet
    - Installation of personal version (for developers): /work/hallc/hks/your_directory
  - cdaq machines:
    - Used for online analysis during the experiment
    - Hall C software modules at /capps/modulefiles: ROOT, analyzer, hcana, panguin, epics, …
      Let me know if you need any other software to make it available

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