JUNO Offline Software for Data Processing and Analysis

Ziyan Deng, Wenxing Fang, Xingtao Huang, Teng Li, Weidong Li, Tao Lin, Zhengyun You, Jiaheng Zou

on behalf of the JUNO offline software team

May 8-12 , 2023

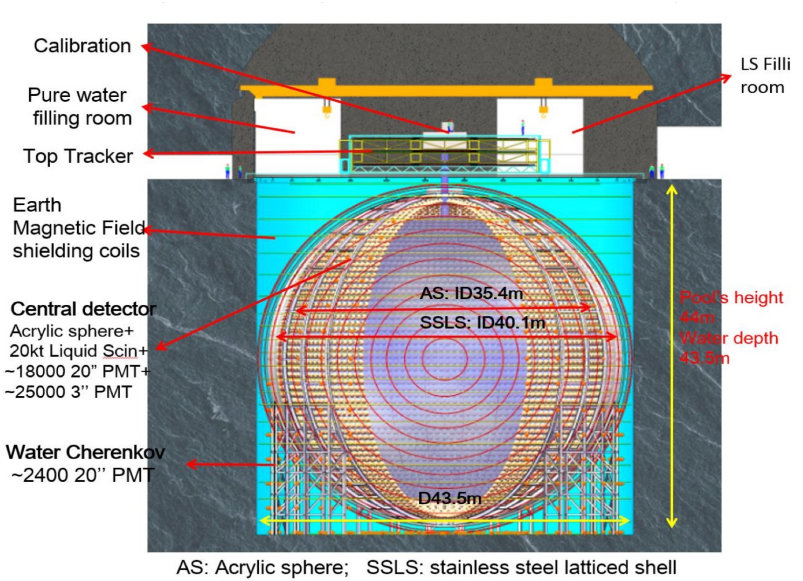
Contents

- JUNO Experiment
- JUNO Software (JUNOSW)
  - Framework
  - Event Data Model
  - Geometry Management System
  - Event Display Tools
  - Database System
  - Analysis Framework
  - Data Processing and Analysis Chain
- Summary

Jiangmen Underground Neutrino Observatory (JUNO)

- JUNO is a multiple-purpose neutrino experiment currently under construction in southern China
  - Large fiducial volume: 20 kton liquid scintillator detector
  - Excellent energy resolution :3% energy resolution at 1MeV
  - Rich physics programs:
    - Reactor neutrinos: Mass Hierarchy and precision measurement of oscillation parameters
    - Supernova neutrinos
    - Geoneutrinos
    - Solar neutrinos
    - Atmospheric neutrinos
  - Status:
    - Finished civil construction in Dec. 2021
    - Detector assembling goes smoothly now
    - Expected to start data taking in 2024

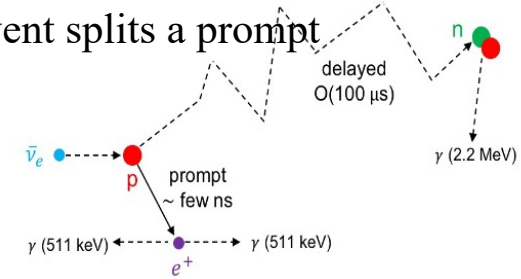
"JUNO physics and detector", Progress in Particle and Nuclear Physics, Volume 123, March 2022, 103927



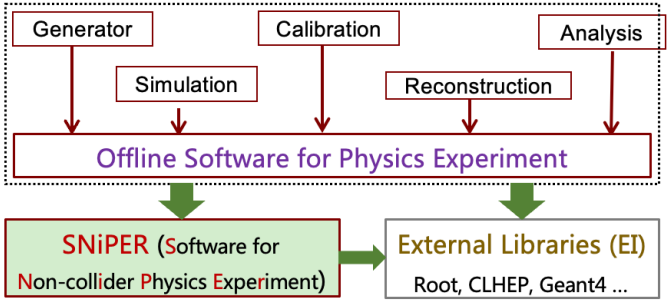
AS: Acrylic sphere; SSLS: stainless steel latticed shell

JUNO Software (JUNOSW)

- Data processing and analysis of neutrino experiments is different from the one of the collider experiments, i.e. reactor neutrino is detected with the Inverse Beta Decay (IBD)
  - M.C. production: Event splitting and mixing -> one neutrino event splits a prompt and a delayed event -> complicate relations
  - Physics analysis : time and vertex coincidence between the prompt and delayed events -> event buffering
  - Large data volume and rare signal: 2 PB/year * 20 years; only ~60 reactor neutrinos per day



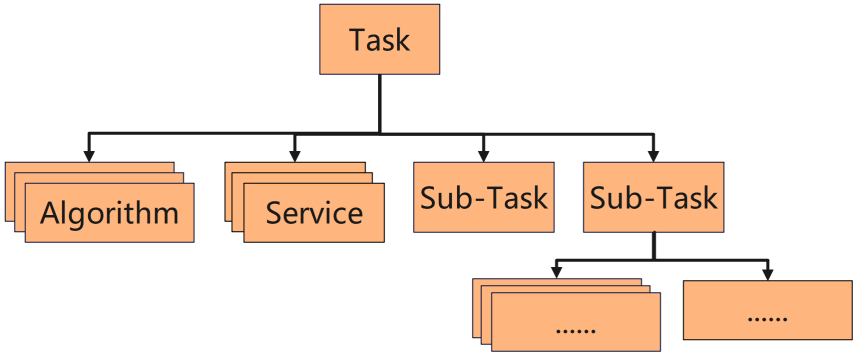
- JUNOSW (originally called as JUNO offline) has been developed based on NuWa (the offline software of the Daya Bay experiment) from scratch since 2012.
  - Core Software: Framework, Event Data Model, Geometry Management, Database System, Analysis Framework
  - Applications: simulation, calibration, reconstruction and analysis
  - Modern management technologies: C++17, python3, cmake, gitlab (https://code.ihep.ac.cn/JUNO/offline/junosw)



"Modern Software Development for JUNO offline software" by Tao Lin on May 9

SNiPER Framework

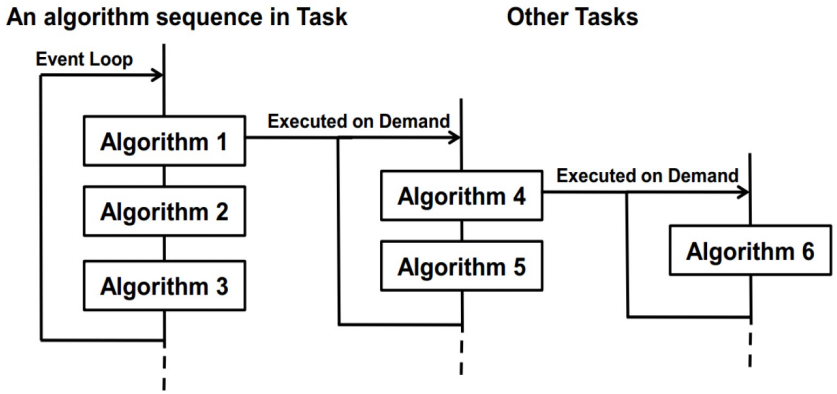
- SNiPER (Software for Non-Collider Physics Experiment) : a general-purpose software framework
  - Light weighted: only dependent on Boost.Python
  - Highly modularized and extensible
  - Originally developed for JUNO, but also adopted by several experiments: LHAASO, STCF, HERD, nEXO, etc.
- Key components
  - Algorithm: an unit of event data proceeding
  - Service: an unit for common functions that can be called by users, anywhere when necessary.
  - Task: a lightweight application manager to assemble specific algorithms, service as well as sub-tasks.
  - Data Buffer: a central place in memory for holding and sharing multiple events



J. H. Zou et al 2015 J. Phys.: Conf. Ser. 664 072053

SNiPER Framework

- SNiPER (Software for Non-Collider Physics Experiment) : a general-purpose software framework
  - Light weighted: only dependent on Boost.Python
  - Highly modularized and extensible
  - Originally developed for JUNO, but also adopted by several experiments: LHAASO, STCF, HERD, nEXO, etc.
- Key components
  - Algorithm: an unit of event data proceeding
  - Service: an unit for common functions that can be called by users, anywhere when necessary.
  - Task: a lightweight application manager to assemble specific algorithms, service as well as sub-tasks.
  - Data Buffer: a central place in memory for holding and sharing multiple events

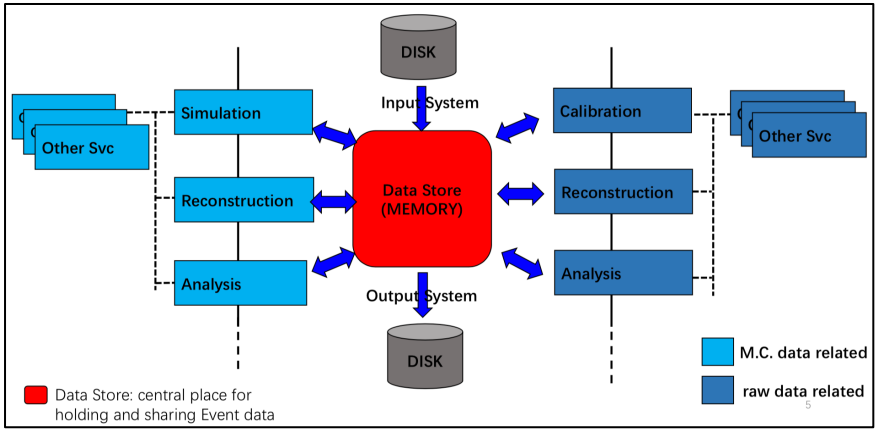


J. H. Zou et al 2015 J. Phys.: Conf. Ser. 664 072053

SNiPER Framework

- SNiPER (Software for Non-Collider Physics Experiment) : a general-purpose software framework
  - Light weighted: only dependent on Boost.Python
  - Highly modularized and extensible
  - Originally developed for JUNO, but also adopted by several experiments: LHAASO, STCF, HERD, nEXO, etc.

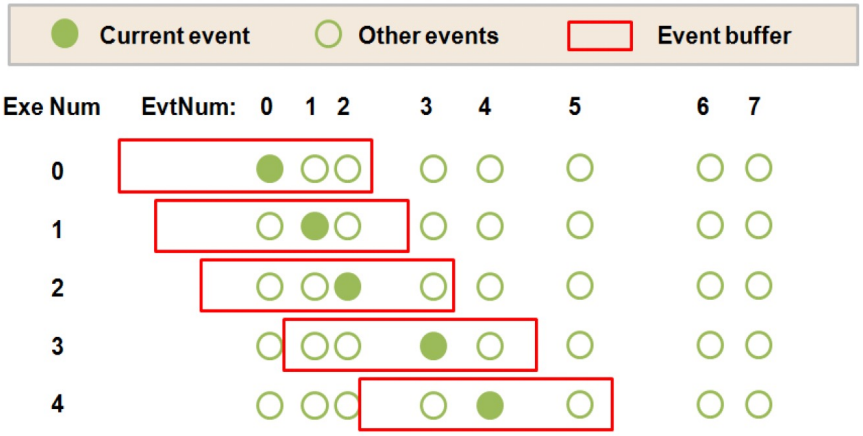
- Key components
  - Algorithm: an unit of event data proceeding
  - Service: an unit for common functions that can be called by users, anywhere when necessary.
  - Task: a lightweight application manager to assemble specific algorithms, service as well as sub-tasks.
  - Data Buffer: a central place in memory for holding and sharing multiple events



J. H. Zou et al 2015 J. Phys.: Conf. Ser. 664 072053

SNiPER Framework

- SNiPER (Software for Non-Collider Physics Experiment) : a general-purpose software framework
  - Light weighted: only dependent on Boost.Python
  - Highly modularized and extensible
  - Originally developed for JUNO, but also adopted by several experiments: LHAASO, STCF, HERD, nEXO, etc.
- Key components
  - Algorithm: an unit of event data proceeding
  - Service: an unit for common functions that can be called by users, anywhere when necessary.
  - Task: a lightweight application manager to assemble specific algorithms, service as well as sub-tasks.
  - Data Buffer: a central place in memory for holding and sharing multiple events

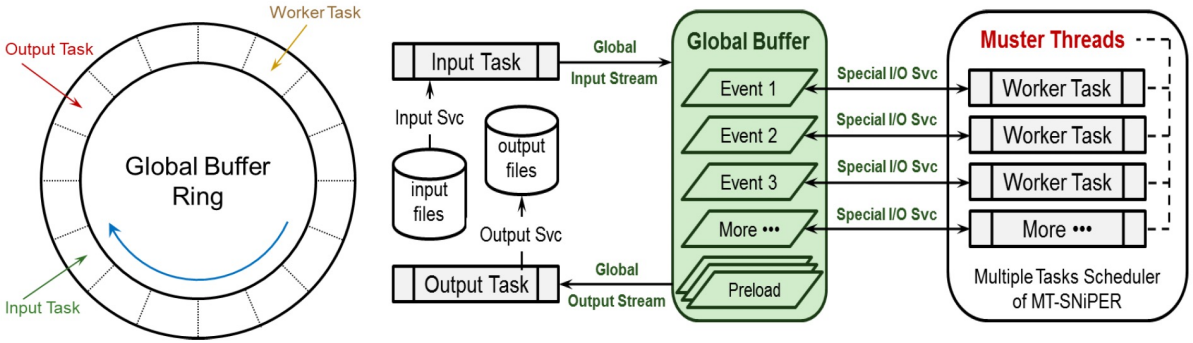


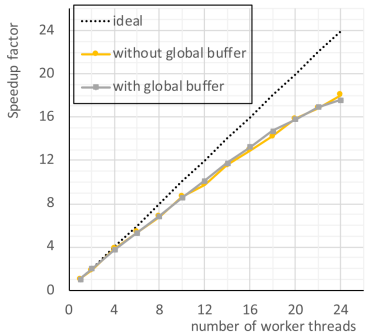
J. H. Zou et al 2015 J. Phys.: Conf. Ser. 664 072053

Multi-threaded SNiPER (MT.SNiPER)

- MT.SNiPER supports concurrent task of event processing based on the TBB
  - A thread-safe Global Buffer is designed and implemented with a ring and configurable capacity and cordon
  - All events in the Global Buffer are sorted by their time stamps
  - Allow the dedicated Input Task, Worker Task, Output Task to put, take, and popup events at the same time in different threads
  - Events are concurrently processed in different Worker Tasks
- The multi-threaded simulation and reconstruction have been working very well

J.H. Zou et al., J. Phys. Conf. Ser. 1085, 032009 (2018)
T. Lin et al., EPJ Web of Conferences 214, 02008 (2019)



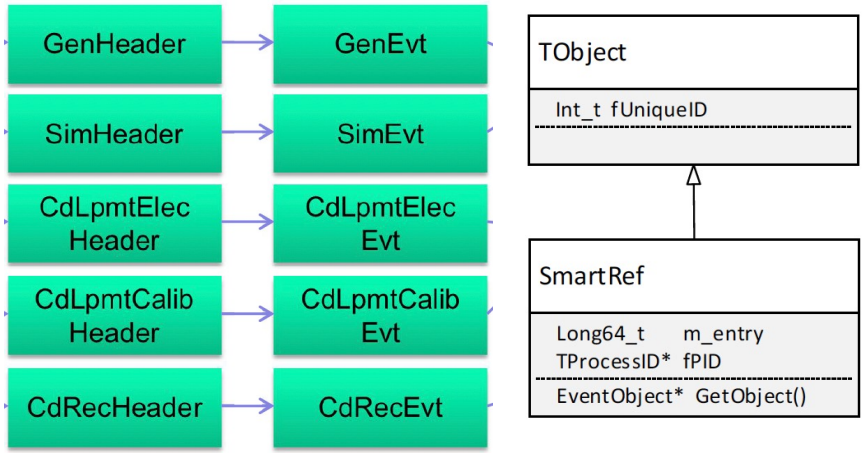


Event Data Model

- Event Data Model (EDM) defining event objects for different processing stages and correlations between different event objects takes very important roles on the whole data processing and analysis
- JUNO EDM is based on ROOT and takes advantage of its intrinsic powerful functions, persistency, IO streamer, scheme evolutions, run time type Information etc.

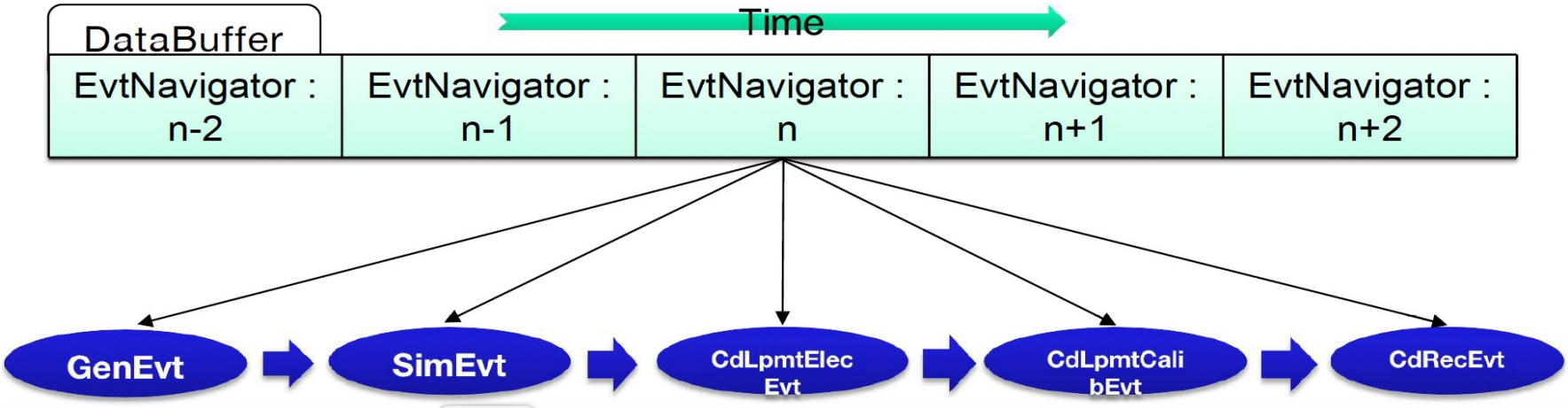
  - All EDM classes are derived from TObject
  - Two layer design (Header and Event) of EDM classes are adopted to speed up event selection
    - The Header holds light-weighted features (tag) of events while the Event holds heavy data
    - A smart references (SmartRef) based on GUUID is developed to build correlations between Header and Event
    - SmartRef also provides a lazy-loading mechanism to dramatically reduce I/O burden

T. Li et al., J. Phys. Conf. Ser. 762 (2016) 012001
T. Li et al., Chinese Physics C 41(6): 066201



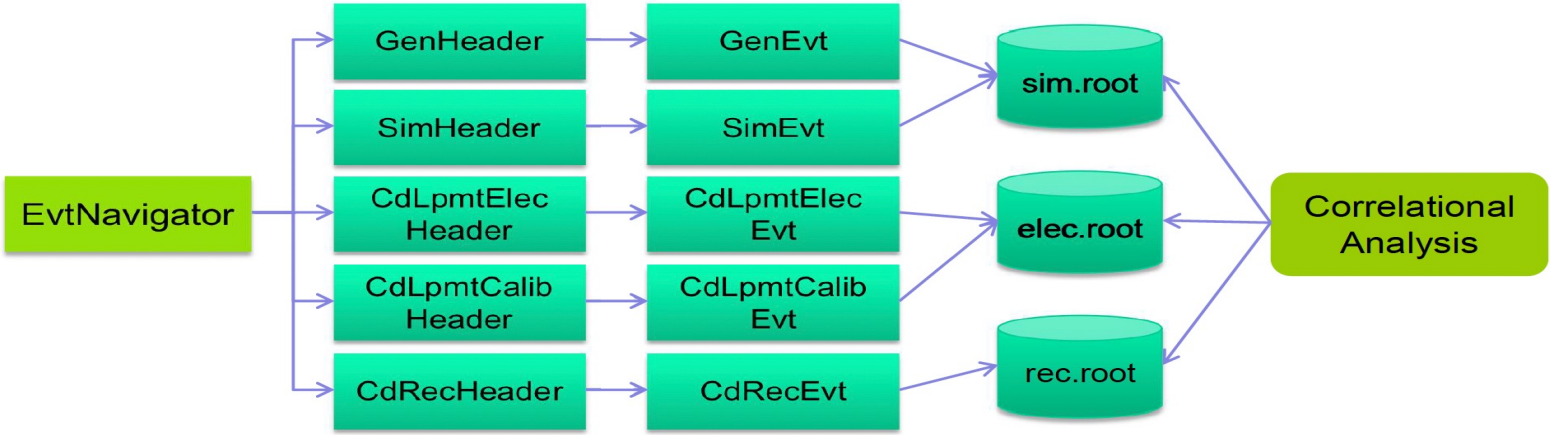
Event Data Model

- EvtNavigator is developed to serve as the catalogue for EDM objects
  - EvtNavigator also uses SmartRef to correlate EDM objects
  - A event-index mechanism based on Global UUID is implemented to facilitate cross-file correlation analysis
  - Provide the way to trace the MC truth of readouts at the event/particle/hit level with the help of external associations



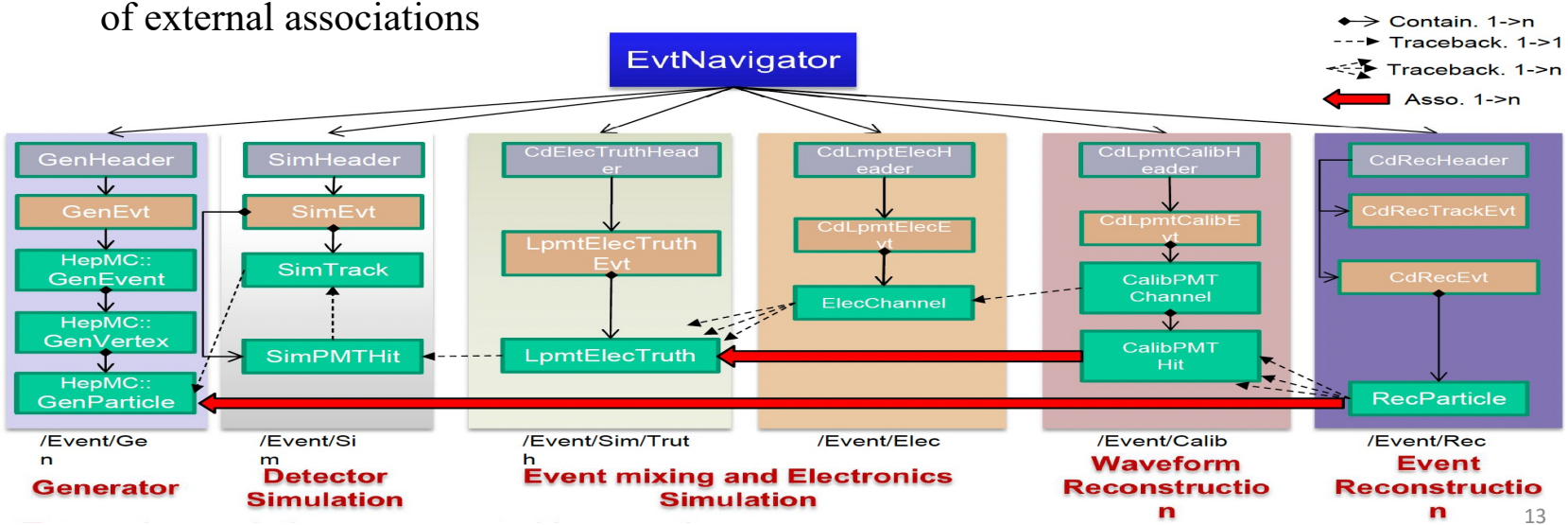
Event Data Model

- EvtNavigator is developed to serve as the catalogue for EDM objects
  - EvtNavigator also uses SmartRef to correlate EDM objects
  - A event-index mechanism based on Global UUID is implemented to facilitate cross-file correlation analysis
  - Provide the way to trace the MC truth of readouts at the event/particle/hit level with the help of external associations



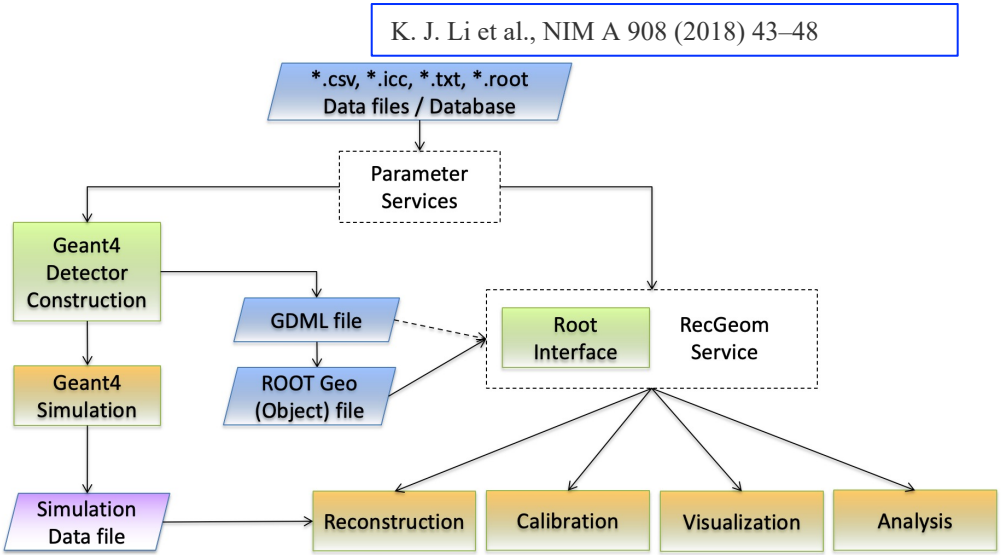
Event Data Model

- EvtNavigator is developed to serve as the catalogue for EDM objects
  - EvtNavigator also uses SmartRef to correlate EDM objects
  - A event-index mechanism based on Global UUID is implemented to facilitate cross-file correlation analysis
  - Provide the way to trace the MC truth of readouts at the event/particle/hit level with the help of external associations



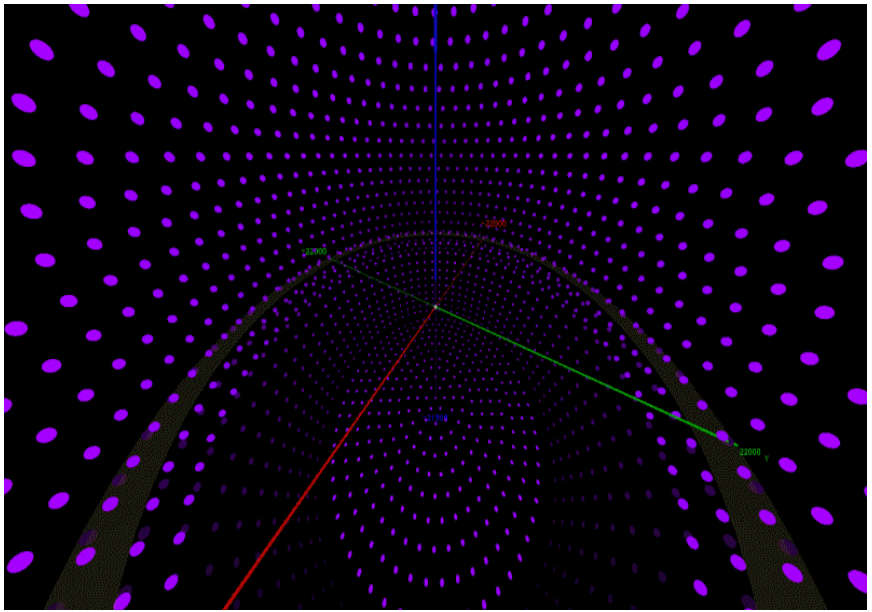
Geometry Management System (GMS)

- GMS is developed to provide a consistent and detailed detector description for different applications
  - Input from single source of geometry data : currently from ROOT file, plan to read from DB
  - Export from Geant4 detector construction into GDML/ROOT TGeo file format
  - Perform automatic conversion with Files – Geant4 – GDML – ROOT – Applications with different services
    - Parameter Service
    - RecGeom Service
  - Geometry information depending on different level of request
  - Some information not supported by GDML can be retrieved from Parameter Services
    - Optical Surface
    - Matrixes

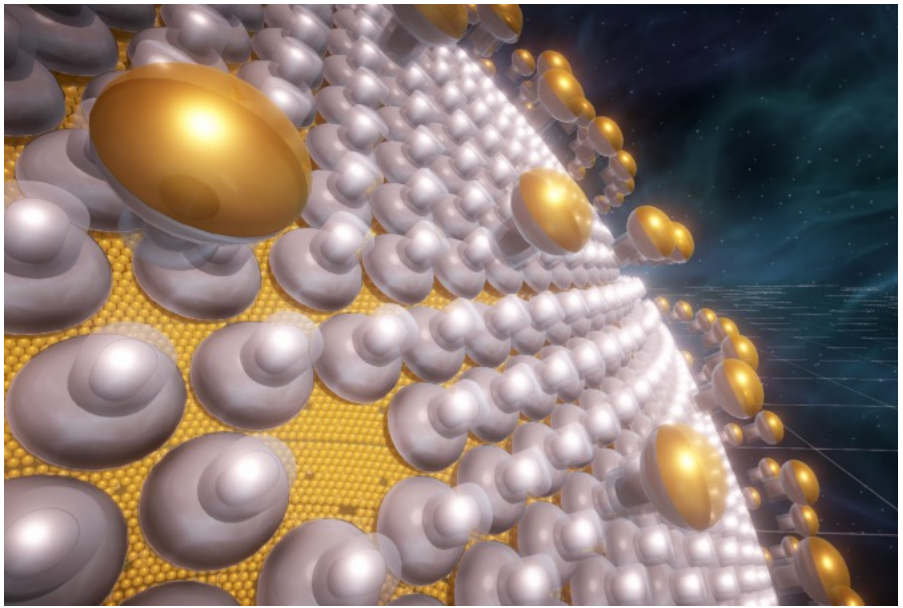


Event Display

- Root based Event Display: SERENA (Software for Event display with Root Eve in Neutrino Analysis)



- Unity based Event Display: ELAINA (Event Live Animation with unIty for Neutrino Analysis)

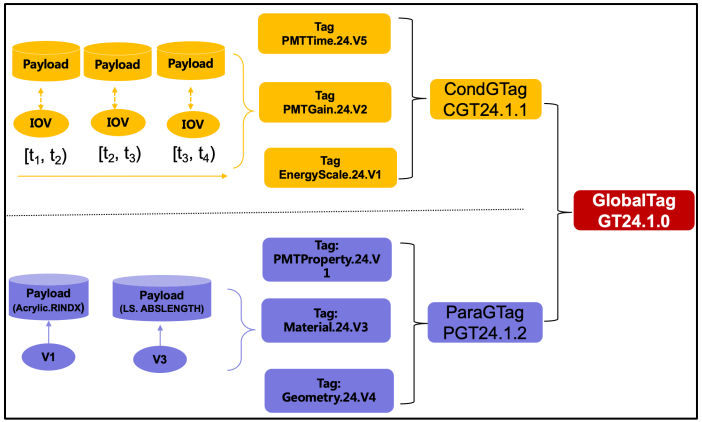


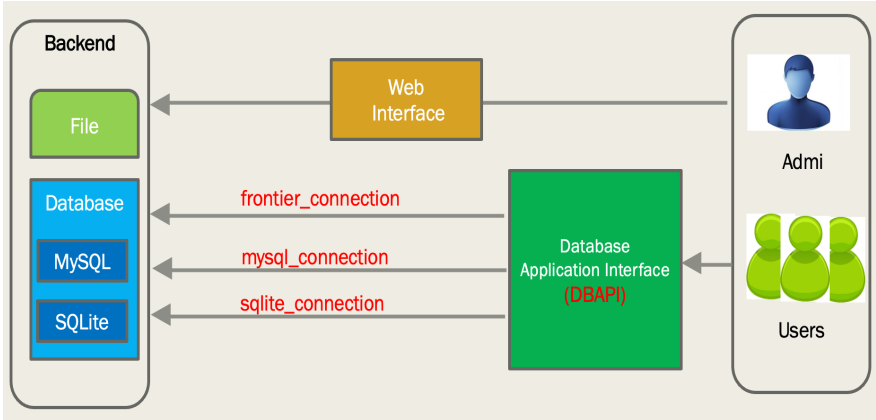
J. Zhu et al., JINST 14 (2019) 01, T01007

Database System

- OfflineDB is developed to unify management of Conditions Data and Parameter data
  - Two types of data are stored in Payloads in their suitable formats and deployed in CVMFS
  - Meta data are managed with DB tables: Payload, IOV, Version, Tag and GlobalTag
  - A GlobalTag combines the tag of Conditions data and Parameter data
  - Three backups are implemented to meet different applications
    - MySQL for quickly testing of database
    - Frontier+Squid for massive data production
    - SQLite for online event classification (OEC)

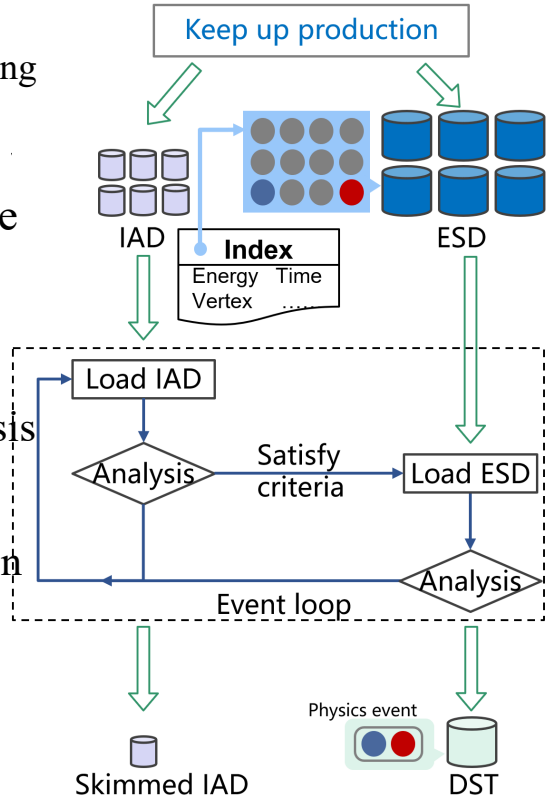
X.T. Huang et al., *EPJ Web of Conferences 245, 04030 (2020)*





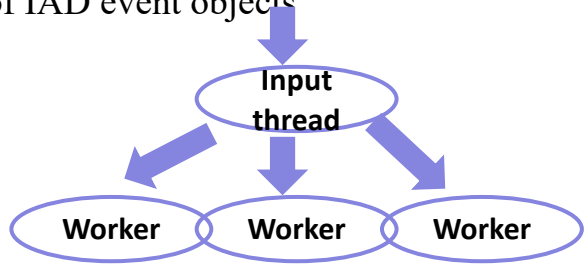
Analysis Software Framework

- Challenges of data analysis
  - Time correlation analysis (a prompt signal and a delayed signal within varying time window)
  - Rare signals (1kHz event rate, but only ~60 reactor neutrinos per day)
- Developed analysis software framework with an Indexing Technique
  - The Indexed Analysis Data (IAD) consists of two parts:
    - A pointer to the associated Event Summary Data (ESD)
    - A collection of key reconstructed variables needed by the analysis
- Analysis work flow
  - Step1: perform event pre-selection within the IAD stream based on its key variables
  - Step2: navigate back to the associated ESD and perform further selection in the ESD stream
  - Time correlation analysis could be applied in both steps

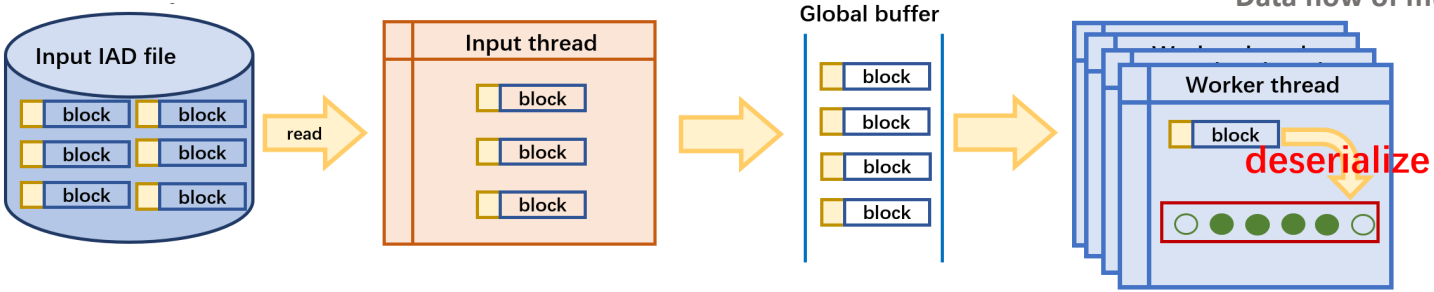


Analysis Software Framework

- Multi-threading in Analysis
  - One data input thread reads IAD blocks and puts them in the global buffer
  - Multiple worker threads
    - get IAD blocks from global buffer and convert them to a collection of IAD event objects
    - perform two-level event selections and run analysis algorithms
  - One or two data output threads
    - write skimmed IAD event objects into a file
    - write Data Summary Type (DST) objects into another file

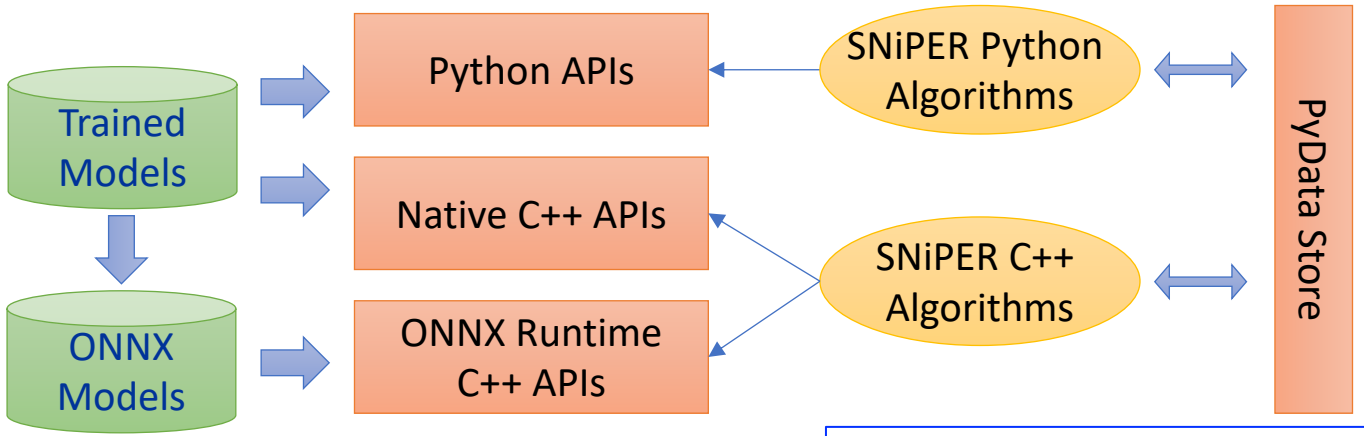


Data flow of multi-threading



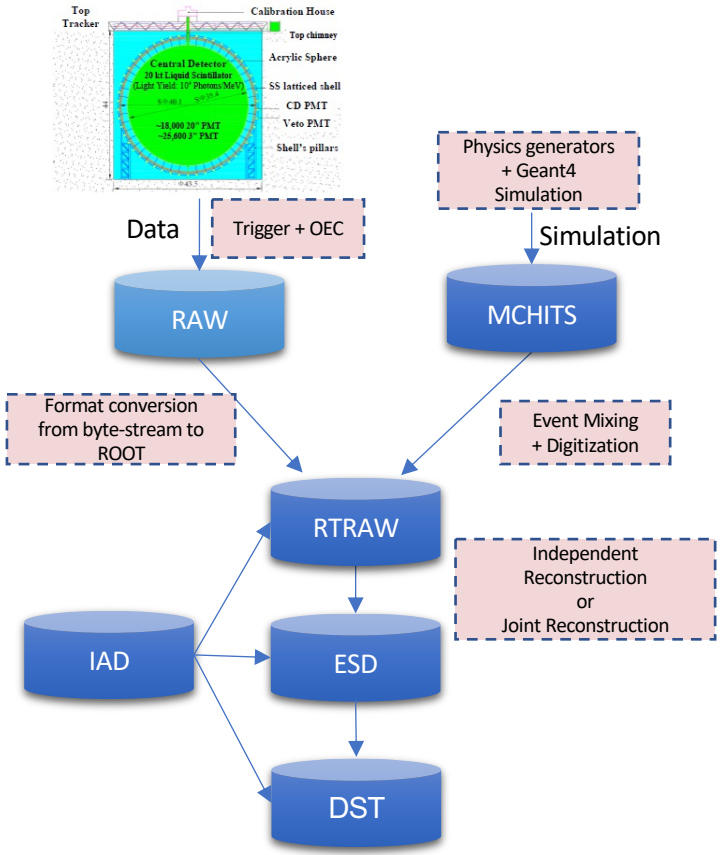
Integration of machine learning-trained models

- A python based datastore is developed for data sharing between C++ and Python
- Popular ML libraries are integrated into JUNOSW in C++ for inference with three solutions:
  - Python API for SNiPER Algorithm in python
  - Native C/C++ APIs: TensorFlow, PyTorch
  - Open standard for ML: ONNX + ONNX Runtime (C++)



T. Lin et al., *Integration of machine learning-trained models into JUNO's offline software, ACAT2022*

Data Processing and Analysis Chain



- MCHITS (Monte-Carlo Simulated Hits Data)
  - In ROOT format, contains GenEvt, SimEvt
- RAW (Raw Data from DAQ and OEC)
  - In byte-stream format. Also contains the OEC result.
- RTRAW (ROOT RAW)
  - In ROOT format (with EvtNavigator).
  - Sorted; contains ElecEvt, TrigEvt, OecEvt; may contain GenEvt and SimEvt for simulated data.
- ESD (Event Summary Data)
  - In ROOT format (with EvtNavigator), contains CalibEvt, RecEvt
- DST (Data Summary Type)
  - Contains information of physical events, Flexible definition and file format, Does not contain relationships
- IAD (Indexed Analysis Data) (Auxiliary data)
  - Contains the address to the RAW/ESD and the necessary physics variables

Summary

- The JUNO offline software (JUNOSW) has been developed with modern technologies to meet the specific requirements of JUNO experiment and other non-collider experiments
  - Data processing framework, SNiPER, supports buffering and management of multiple events, event splitting and mixing, TBB-based multi-threading, and integration of machine learning etc.
  - ROOT based event data model to character data representations at different processing stages and complicated relationships between them.
  - Unified detector geometry management system to support multiple applications including simulation, calibration, reconstruction and detector visualization.
  - Unified database system to manage the conditions data and parameter data
  - Event index based correlation analysis framework to support selection of sparse physics events from the large volume of data.
- The JUNO data processing and analysis chain was completed and has been used by several rounds of M.C. data challenge on either local computing clusters or the distributed computing infrastructure.

Thanks for your attention!