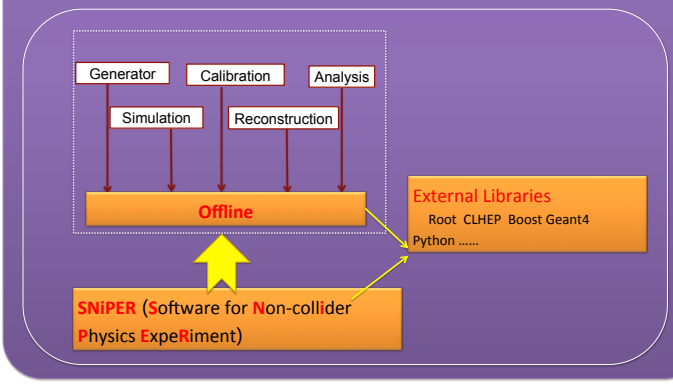
Offline Data Processing Software for the JUNO Experiment

Xingtao Huang(SDU), Jiaheng Zou(IHEP), Weidong Li(IHEP), Teng Li(SDU), Tao Lin(IHEP)
(On behalf of JUNO Collaboration)

The Jiangmen Underground Neutrino Observatory (JUNO) is a 20 kton liquid scintillator detector with multi-purpose measurements. The detector will be built in 700 m deep underground laboratory with primary physics goal for neutrino mass hierarchy determination.

JUNO Offline Software System

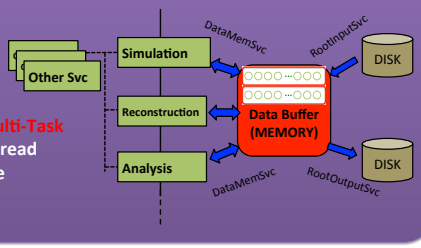
- SNiPER: the Framework for non-collider Physics experiments developed by JUNO Collaboration
- Offline: including event model, geometry, generators, simulation, reconstruction, etc.
- External Libraries: Interfaces to frequently used third-party software and tools



Data Management with Buffer

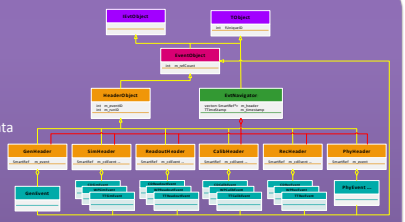
- Buffer is a dynamically allocated memory to hold multiple events
- Buffer size is configured with the Time Window
- Events in Buffer are trimmed dynamically
- Each Task has its own Buffer

- Multiple Events in Buffer supports physics correlation analysis between events
- Multiple Buffers and Multi-Task are suitable for multi-thread parallel computing to be implemented



Event Data Model

- Based on TObject of ROOT
  - Commonly used in memory and root storage
- Two-layer Scenario
  - Header stores small and characteristic data
  - Event stores large and detailed data
  - Highly reduce I/O burden and speed up event selection
- Event Object Correlation
  - Correlations of event data at different stages with SmartRef
  - Correlation between events with Navigators
  - Correlations is able to be saved into Root Files and recovers when reading back



- SmartRef
  - A new mechanism for objects correlation
  - Similar with TRef but supporting object correlation with different entry numbers
  - Correlate data objects in both memory and ROOT files
  - Support lazy loading of data objects very well

Three Functional Layers

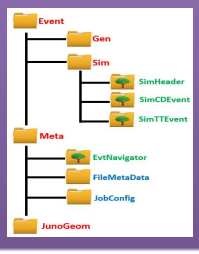
- Application Layer
  - Do not care where the data comes from
  - Application developers:
    1. Read data from data buffer
    2. Concentrate on applications
    3. Put data back to data buffer
  - Do not care where the data will go
- Core Software Layer
  - SNiPER
  - Application Management
    - Load and plug-in applications
    - Manage and execute applications
  - Data Management
    - Store data with data buffer
    - Send data to applications
    - Get data from applications
  - Framework Management
    - Define interfaces
    - Manage interfaces
    - Implement common interfaces
- User Interface Layer
  - Run applications in python batch or interactive modes

Event Layout in Buffer and Root Files

- Events in Buffer
  - Similar with the directory Structure
  - Each Event with an unique path
- Events in Root File
  - Tree Structure
  - Tree/Branch Name same with its path in buffer

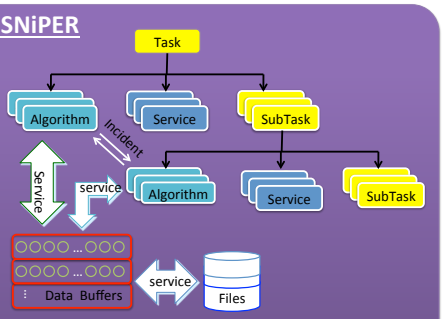
/Event
/Event/Gen
.....
/Event/Sim
/Event/Sim/SimHeader
/Event/Sim/SimCDEvent
/Event/Sim/SimTTEvent
.....

Root/IO



Main Components of SNiPER

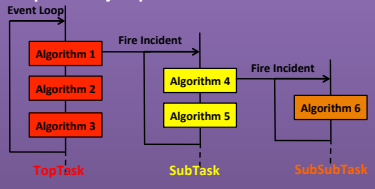
- Modular Design
  - Algorithm
  - Service
  - Task
  - SubTask
  - Incident
  - Data Buffer



Data Processing Controlling

- Task manages its data buffer, algorithms, services and subtasks.
- Task performs sequential execution of its algorithms
- Incident triggers SubTask to implement jump execution

- Multi-Task Mechanism
  - Perform Event Mixing of the Inverse Beta Decay (IBD) with several Backgrounds.
  - Simulate IBD Event Splitting



Status and Outlook

- SNiPER is a new Software for Non-Collider Physics Experiment designed and developed by JUNO Experiment
- Some new functionalities are fully implemented such as Multi-Task, Data Buffers, flexible data processing controlling etc.
- JUNO Event Data Model is based on ROOT and Event Correlations is implemented with a new type of pointer, SmartRef.
- JUNO detector simulation, digitization and reconstruction have been developed based on SNiPER and work very well.
- Software validation suite is in place and routinely runs before a new release
- SNiPER Parallel Computing is under investigation and will be implemented soon.