THE SIMULATION- AND ANALYSIS-FRAMEWORK FAIRROOT

CHEP 2012, New York
Outline

- History
- Features
- New and Recent Projects
- Time Based Simulation
- Example
- Summary and Outlook
FairRoot: timeline

- Start testing the VMC concept for CBM
- First Release of CbmRoot
- Panda decided to join -> FairRoot: same Base package for different experiments
- MPD (NICA) start also using FairRoot
- R3B joined
- ASYEOS joined (ASYEOSRoot)
- EIC (Electron Ion Collider BNL) EICRoot
- GEM-TPC separated from PANDA branch (FOPIRoot)
Developer Team

Core Team:

- Mohammad Al-Turany, GSI-IT, 2003
- Denis Bertini, GSI-IT, 2003
- Florian Uhlig, CBM/IT, 2006
- Radek Karabowicz, PANDA/IT, 2008
- Dmytro Kresan, R3B/IT, 2011
- Tobias Stockmanns, PANDA

People participated in major features:

- Ilse König, HADES
- Volker Friese, CBM

Florian Uhlig

CHEP 2012, New York
FairRoot

- Framework for simulation, reconstruction and data analysis

- Very flexible
  - No executable
    - Use plug-in mechanism from Root to load libraries only when needed
    - Use Root macros to define the experimental setup or the tasks for reconstruction/analysis
    - Use Root macros to set the configuration (Geant3, Geant4, …)
  - No fixed simulation model
    - Use different simulation models (Geant3, Geant4, …) with the same user code (VMC)
Very flexible

- No fixed navigation engine / geometry management
  - Use Root TGeoManager for geometry management
    - Geometry can be defined using different input formats
      - ASCII files in format inherited from HADES
      - Root files
      - Defined directly in the source code
  - Use TEve as base for general event display

- Geometry is described once. Then it can be converted (VGM) to choose between different MC’s and different navigations
  - G4 native geometry and navigation
  - G4 native geometry and Root
  - ...
FairRoot cont.

- Very flexible
  - No fixed output structure
    - Use a dynamic event structure based on Root TFolder and TTree which is created automatically
      - Depend on registered data classes
    - Data output possible after each step
    - Simulation and reconstruction can be done in one go or in several steps
  - Use runtime data base
    - Developed for the HADES experiment
    - Decouple parameter handling in FairRoot from parameter storage
    - runtime data base IO to/from
      - ASCII files
      - Root files
      - Database
Parameter IO

Generic parameter container

Parameter

Object

different versions

name
type: Int_t, Float_t, Double_t, Char_t, Text_t, UChar_t, class type
value(s)

stored as byte array (RAW or BLOB)
number of values (single value or array)
class version
streamer info, root version for ROOT classes
own version management

any class derived from TObject decoded in the analysis interface by ROOT streamer

Florian Uhlig

CHEP 2012, New York

22.05.12
FairRoot DB

Run Manager

IO Manager

FairRoot

RunTime Database

triggers on changing RunId

Init registered parameter containers

ASCII File Configuration parameters.

Root File Configuration parameters.

Oracle

Root File
MC-points
Digits, etc
FairRoot DB extended

FairRoot

- Run Manager
- IO Manager

RunTime Database

DB Interface

TSQlServer

ASCII File Configuration parameters.

Root File Configuration parameters.

- Oracle
- MySQL
- Postgresql
Version management

The Query process
1. Context (Timestamp, Detector, Version) is the primary key
2. Context converted to unique SeqNo
3. SeqNo used as keys to access all rows in main table
4. System gives user access of all such rows
Proof in FairRoot

- PROOF - Parallel ROOf Facility.
- It allows parallel processing of large amount of data. The output results can be directly visualized (e.g. the output histogram can be drawn at the end of the proof session).
- The data which you process with PROOF can reside on your computer, PROOF cluster disks or grid.
- The usage of PROOF is transparent: you use the same code you are running locally on your computer.
- No special installation of PROOF software is necessary to execute your code: PROOF is included in ROOT distribution.
- Proof runs on computing clusters as well as on your local many core computer.
In FairRoot change one line in the macro to use it:

FairRunAna *fRun = new FairRunAna();
To

FairRunAna *fRun = new FairRunAna("proof");
PoD is shipped with a number of plug-ins, which cover all major RMSs, such as local cluster systems and Grid.

If you don’t have any RMS, then the SSH plug-in can be used.

The SSH plug-in is also used to setup PROOF clusters on Clouds.

PoD: Anar Manafov, GSI ([22] 21/05; [21] Postersession 1)
GPU support in FairRoot

- CUDA is fully integrated into the FairRoot build system
- CMake creates shared libraries for cuda part
- FairCuda is a class which wraps CUDA implemented functions so that they can be used directly from ROOT CINT or compiled code

- See talk of Mohammad Al-Turany
  - [353] 24/05, 5.25PM, Room 905/907
Radiation length info
FairRadLenManager

Example: Contributions of different Functional parts of the PANDA MVD to the overall material budget
Dose studies
FairRadMapManager

- What energy dose will be accumulated during a certain time of operation?
- Create all physical volumes with correct material assignment
- Run the simulation engine
- FairRadMapManager will sum up every deposited energy in each volume in the geometry
Determine the particle fluency through a certain boundary (surface) and deduce a map. Knowing the volume and density of the object of interest and the specific energy loss doses can be estimated.
FairRoot for real data

- FairRoot was designed from the beginning to combine simulation and analysis in one tool.
- Using the same internal structure the user can compare easily at any time/level the real data with the simulation.

Reconstructed Beam EVENT

The large GEM-TPC Prototype
L. Fabbietti for the GEM-TPC Collaboration
NeuLAND: The High Resolution Neutron Time-of-Flight Spectrometer for R3B

Comparison of LAND response to neutrons from 170 to 1050 MeV from experiment (black lines) and from R3BRoot simulations (red lines).

Florian Uhlig

CHEP 2012, New York

22.05.12
Triggered Experiment

- Use SHIP (Separator for Heavy Ion reaction products) as example
- If particle enters detector region the data tacking is triggered
- When DAQ reads out the detectors everything belongs to this event
Triggered Experiment

Projectiles -> Target wheel -> Lenses -> Electric Field -> Magnets -> Beam stop -> 7.5° Magnet -> TOF Detectors -> Si Detectors -> γ Detectors
Triggered Experiment
The Challenge

- Central events have up to 1000 charged particles inside acceptance
- Looking for rare probes require events rates up to $10^7$ per second
- Complicated trigger signature
- Searching for secondary vertex requires reconstruction of a large part of the event
The Challenge

- Central events have up to 1000 charged particles inside acceptance
- Looking for rare probes require events rates up to $10^7$ per second
- Complicated trigger signature
- Searching for secondary vertex requires reconstruction of a large part of the event
- Conventional hardware trigger not feasible: no dead time allowed
- Self-triggered autonomous front-ends pushing time-stamped data forwards to DAQ
The Challenge

Necessary to test this in simulation
Task for FairRoot
Provide functionality for the tasks “event mixing” in the digitization stage and “time sorting” in the reconstruction stage.
The absolute event time is calculated by the framework. Experiments define functions for event time calculation. Time of detector digi is this absolute time + the time inside the MC event.
Base class to store detector data (digis) between events

Buffer stores data together with absolute time until this data is active and can be influenced by later events

- This time is detector dependent and is defined individually

If the same detector element is hit at a later time the data can be/is modified

- Modifications are detector and electronics dependent

Result is a randomized data stream which is stored in a TClonesArray which would be the input to the DAQ
Randomized Digi Data

Same color = same event

Array index
Sorter – Technical Implementation

Number of storage cells given by the spread of the time stamps within the data stream.

Width of one storage cell given by time resolution of detector.

Storage Pointer position calculated numerically from time stamp.

0-Element Pointer

If a storage position is calculated which would override old data, the old data is saved to disk and the storage cell is freed.
Sorted Digi Data

![Graph showing sorted digital data with array index and h0 entries. The graph indicates a distribution of entries across different array indices.]

Florian Uhlig

CHEP 2012, New York

22.05.12
Sorted Digi Data

Florian Uhlig

CHEP 2012, New York
Task for Experiments
Event Builder
Read back the Data

Eventbuilder

- Read data from IO Manager using different functions
- Different algorithms already available
  - Read data up to a given absolute time
  - Read data in a given time window
  - Read data until next time gap of certain size
- Other algorithms can be (easily) implemented if needed
Example: Rutherford Experiment

- Scattering of 5MeV alpha particles at a 2μm gold foil
- Unexpected large scattering angles observed
- Implementation using FairRoot needs
  - 600 lines of c++ source code created mostly automatically (copied from a template)
  - 60 lines of code for the build system
  - 200 lines of code for the steering macros
  - 70 lines of code for the geometry and media definition
Rutherford Experiment

- Change experimental setup
- Change material properties
- Change simulation engine
- Change physical processes
Summary and Outlook

- Hope I could show you that FairRoot
  - is flexible
  - is easy to use
  - is easy to extend
- Special tools to do dose studies
- Tools for time based simulation are implemented
  - Calculation of event time
  - Mixing of events by automatic buffering and write out when needed
  - Fast sorting of data
  - Several event builder functions
Summary and Outlook

- Many more topics only touched or not showed at all
  - Proof integration
  - Database connectivity
  - GPU usage inside of FairRoot
  - Build and test system
  - ...

- Resources
  - Webpage: http://fairroot.gsi.de
  - Forum: http://forum.gsi.de
  - Test Dashboard: http://cdash.gsi.de/CDash
FairRoot related talks and poster

- [606] Future Experiments and Impact on Computing
- [394] Event reconstruction in the PandaRoot framework
- [353] Track finding and fitting on GPUs, first steps toward a software trigger
- [40] STEPtoRoot – from CAD to monte carlo simulations
- [399] Electron reconstruction and identification capabilities of the CBM Experiment at FAIR