Quo Vadis?

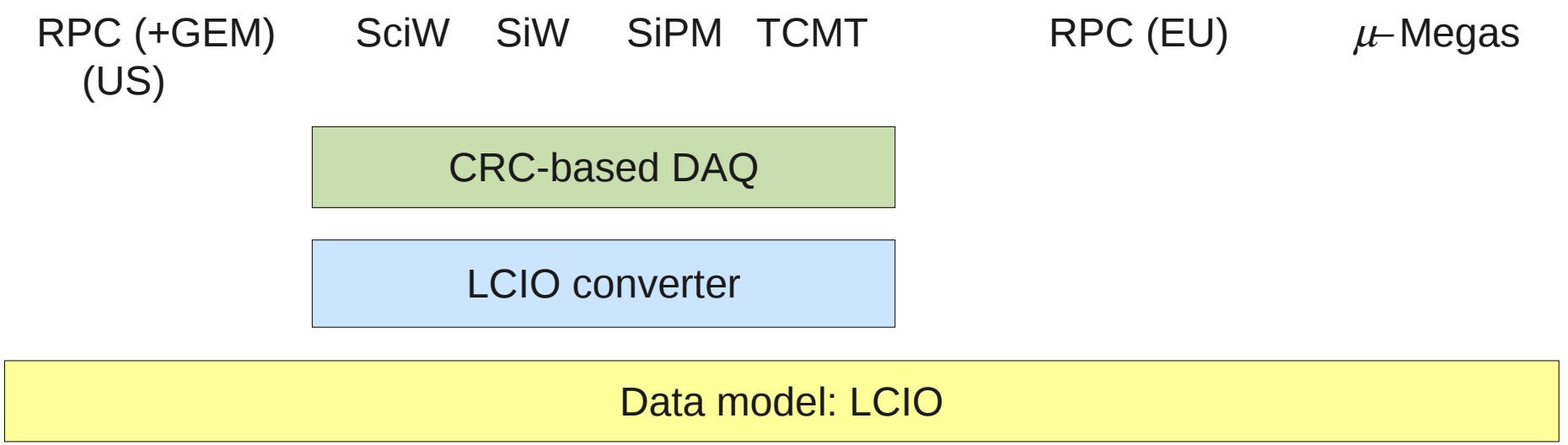
The future of CALICE offline computing

Niels Meyer, DESY

TechBoard Review, FNAL

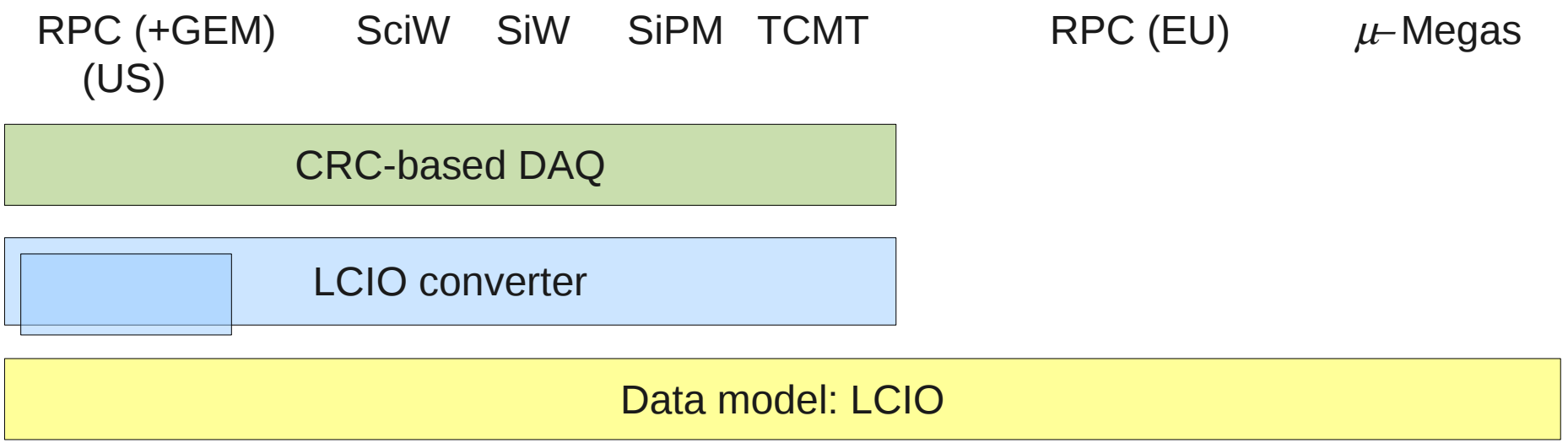
June 16, 2009

DAQ - Offline Interface



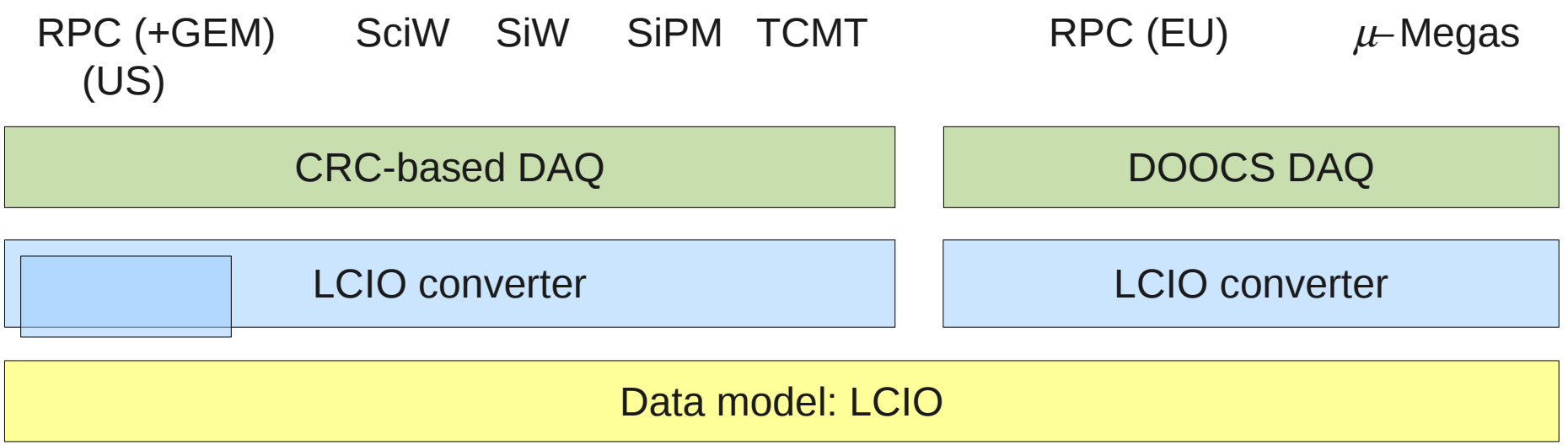
- Consensus: LCIO as data model
- Old DAQ has own format -> converter to LCIO in place

DAQ - Offline Interface



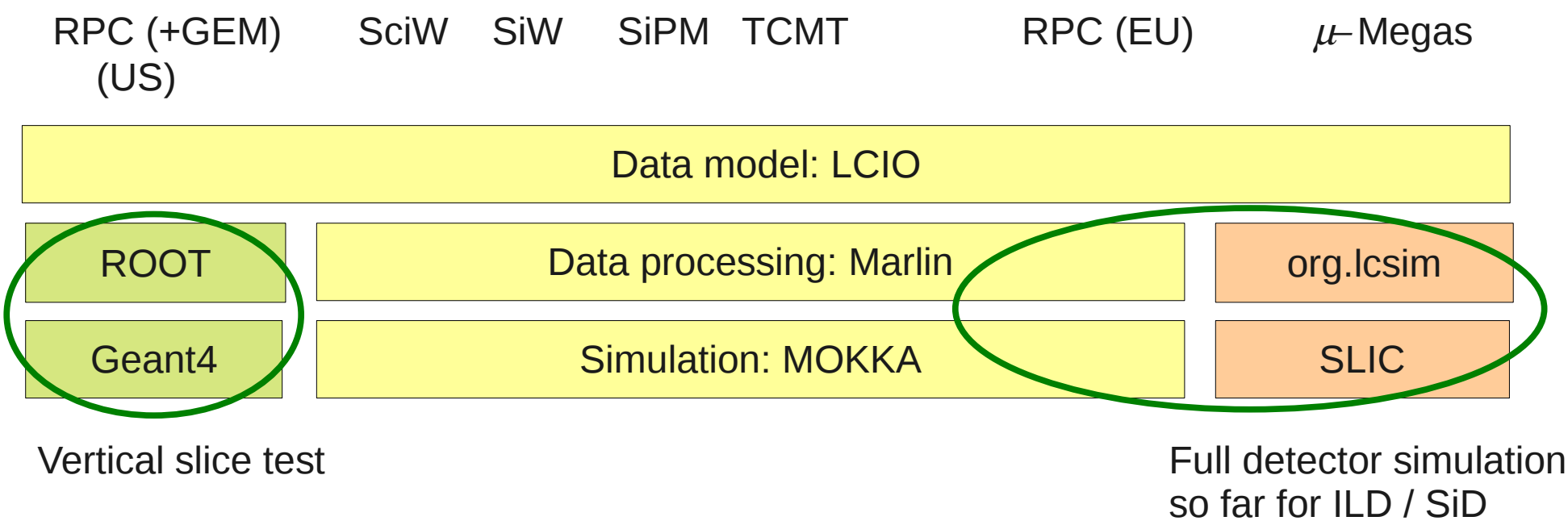
- Consensus: LCIO as data model
- Old DAQ has own format -> converter to LCIO in place
- US RPC uses CRC-DAQ framework, will implement dedicated event building as module in present LCIO converter

DAQ - Offline Interface



- Consensus: LCIO as data model
- Old DAQ has own format -> converter to LCIO in place
- US RPC uses CRC-DAQ framework, will implement dedicated event building as module in present LCIO converter
- DOOCS DAQ: XDAQ raw format plus new converter to LCIO

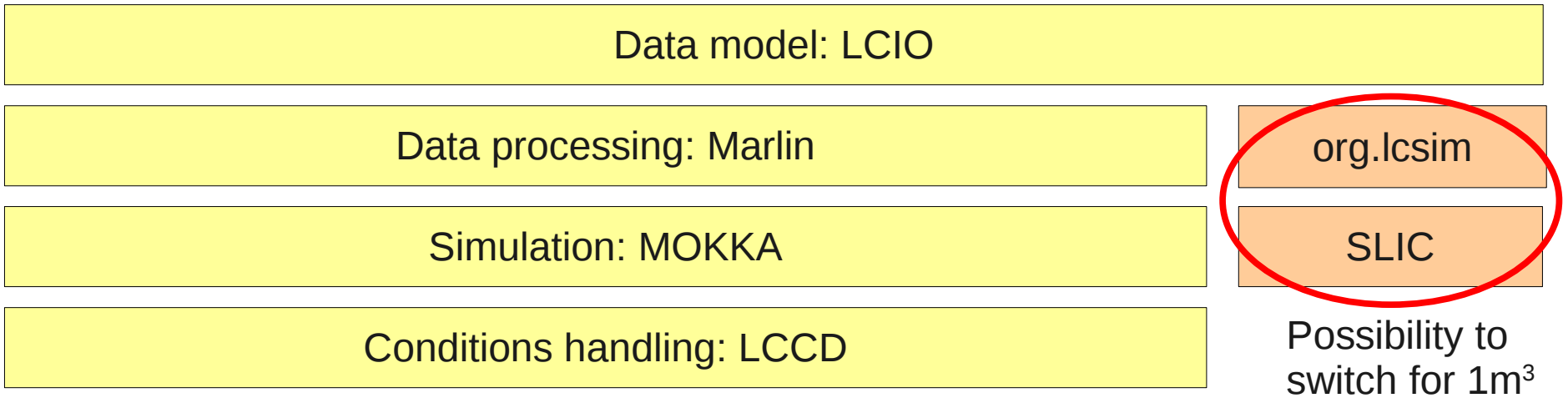
Data Processing



- LCIO: common development by ILD and SiD, in C++ and JAVA
- Combined TB uses ILD software (C++ throughout)

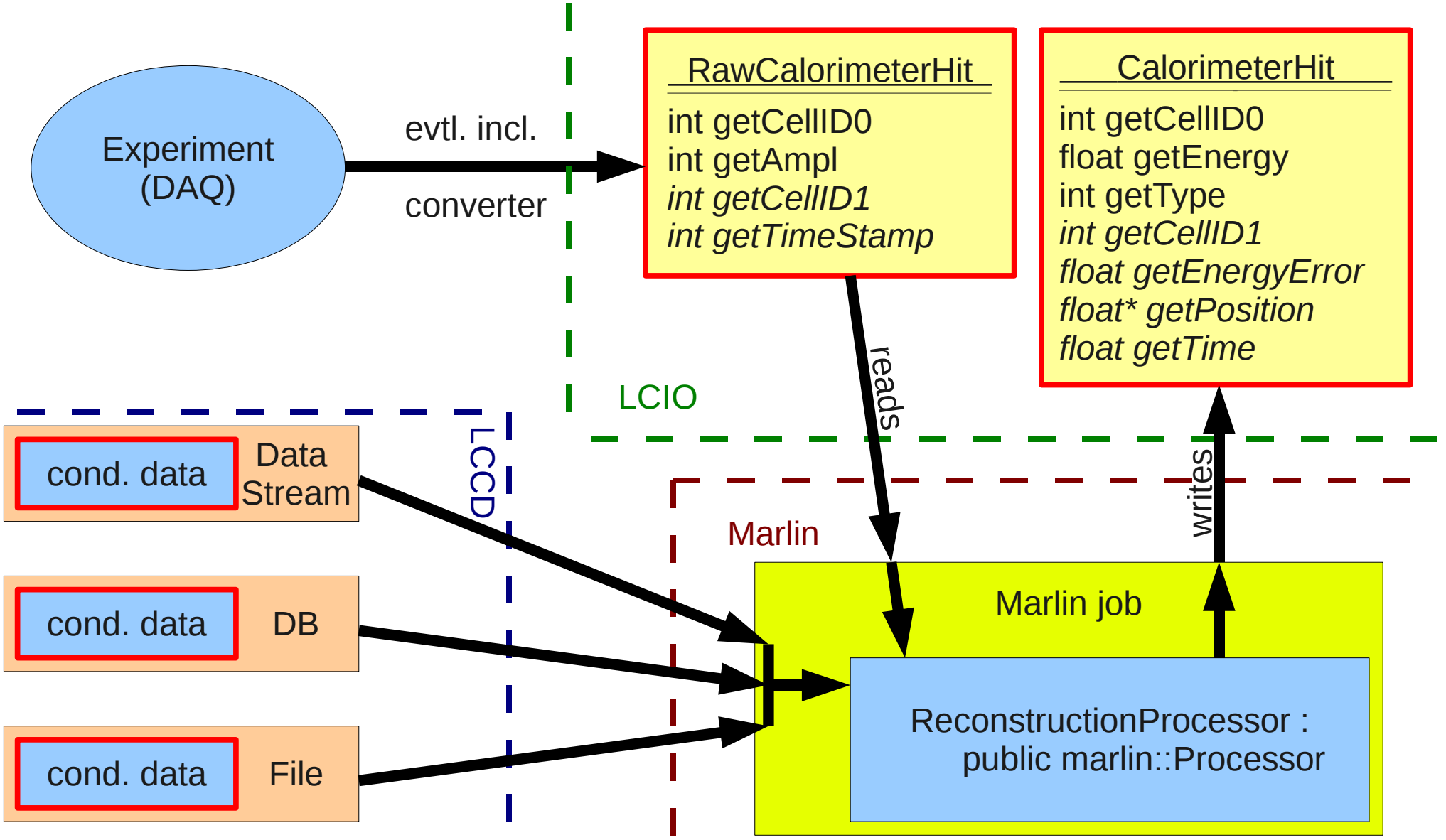
Data Processing

RPC (+GEM) (US) SciW SiW SiPM TCMT RPC (EU) μ–Megas



- LCIO: common development by ILD and SiD, in C++ and JAVA
- Combined TB uses ILD software (C++ throughout)
- ILD is the only one providing a package for conditions handling
- TechBoard decision at Daegu: continue with ILD for physics prototypes, no double implementation in JAVA

Idea of Hit Reconstruction



Generic Hit Reco

- Raw data characterized by *electronic channel*
- Calibration constants characterized by *detector cell*
- Hit characterized by *positon in shower / detector*
- The two translations inbetween are *cabeling* and *assembly*
- There are two 'generic' calibration algorithms
  - scalar operation on single cell amplitude (ped.subtraction, weighting, non-linearity correction, unit conversion, ...)
  - matrix operations on all amplitudes of one module (cross-talk correction, correction of coherent movement, ...)

Learning from the Past

- All detectors implemented this scheme with little common tools, but lots of code copy-and-paste plus modifications
- Few agreements made (coordinate system, geometrical cell ID)
- Most of effort went to 'infrastructure', while the formulae and matrices are the important part for detector studies and quality of data passed on to analyzers
- Integration of SciECal into offline reconstruction and simulation still not accomplished, due to test beam campaigns, limited man power, and little usable tools from present packages
- What isn't common in reconstruction also is not universal on analysis level and prevents development of common tools (e.g. track finder, event display, density clustering, ...)

DHCal Offline Task Force

- Proposed at Daegu, consists of one SW expert per project:
  - Jan Blaha (MicroMegas)
  - Gerald Grenier (RPC EU)
  - Lei Xia (RPC + GEM US)
  - N.M. (Analog & SW coordinator)
- Had two phone meetings with basic discussions
- Preliminary conclusions:
  - LCIO is fit for DHCals, i.e. no alternative to CalorimeterHit needed
  - Different to analog: timing is very important - eventually new non-hit classes needed
  - Common interest: weather conditions (temperature, air pressure)

Data Handling

- Currently rely on GRID for storage and processing to handle large data volums plus MyDQL server to store and provide conditions data
- Low-level scripts available for conversion and reconstruction; ok for experts (conversion and reconstruction), analyzers often use local resources
- Concept of database servers is good, but need to be clever to avoid bottle-necks. Some development work done for LCCD and upstream packages done already
- Problem is contents, not infrastructure (lack of man power)
- Incorporation of additional test beam campaigns straight forward, more convenient tools sometimes would be desirable

Summary

- Would not be where we are without ILD software for the physics test beam campaigns - should commit to this decision also for the future
- GRID resources -with all the childhood deseases and the inconvenience- are inevitable for our data handling
- Very successful examples from the electronics side and from the DAQ systems: CALICE is strong thanks to common interfaces. This has less been considered in offline computing, and we currently pay the price
- Promising attempts underway to avoid mistakes from the past for future DHCal prototypes
- You get what you pay for - clearly need more momentum for combined software effort in CALICE as a whole