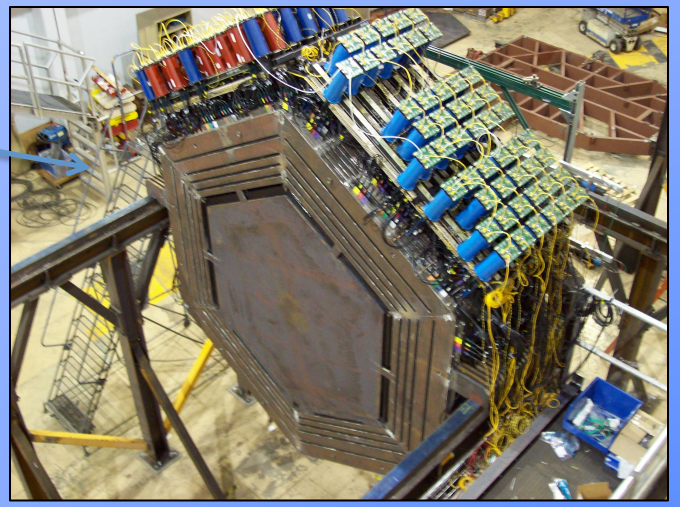
How We Get Things Done at MINERνA

Dave Schmitz – Fermilab

Workshop on Computing for Neutrino Experiments
Fermilab, March 12-13, 2009

MINERνA

- Purpose
  - a finely segmented, fully active detector for study of **neutrino-nucleus** interactions in the GeV energy range
  - provide important information for current and future **oscillation experiments** (MINOS, T2K, NOνA, DUSEL)
- Timeline
  - build a prototype detector (Wednesday!)
  - install prototype detector at NuMI (this month)
  - install full detector (spring 2010)
  - physics run and analysis (2010 –)
- People (see Heidi's talk)
  - currently ~20-30 software users and developers
  - more everyday



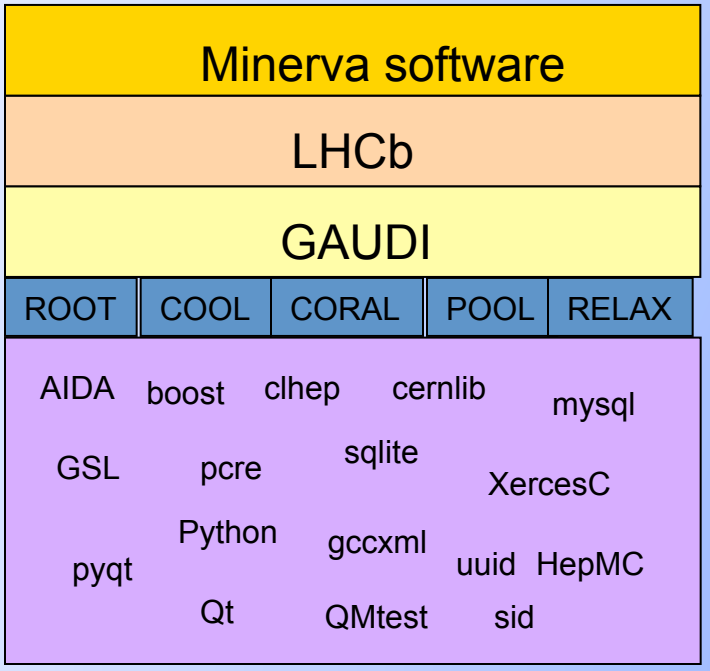
Software Framework

- Brett has done all the work for me. Thanks, Brett!

- MINERvA also uses the GAUDI framework developed by the LHCb collaboration and used by ATLAS
- MINERvA is a fairly small experiment with limited manpower and the decision was made some time ago to take advantage of the available GAUDI framework
  - LHCb had already spent a decade developing it
  - had been successfully used by other small experiments
  - with LHCb and ATLAS using it, will continue to be supported well beyond lifetime of MINERvA
- also can take advantage of many tools from LHCb

Software Framework

- what we haven't done yet – gone through the necessary process to build everything from source…
- LHCb software is built on a particular version of GAUDI
- GAUDI is built on a set of applications and external software products available in a bundle from LCG



LCG

LHCb (v26r1)

↓

GAUDI (v20r4)

↓

LCG v55c

so we currently use binary distributions available for SL4 and build our software on top

SL5 available soon from LHCb

MAC OSX planned

Software Framework

- as at Daya Bay, we also use some general tools from LHCb :
  - GaudiObjDesc – define data model in xml format
  - DetDesc – xml geometry and materials
  - GiGa – interfaces GEANT4 to framework
  - Panoramix – geometry visualization and event display

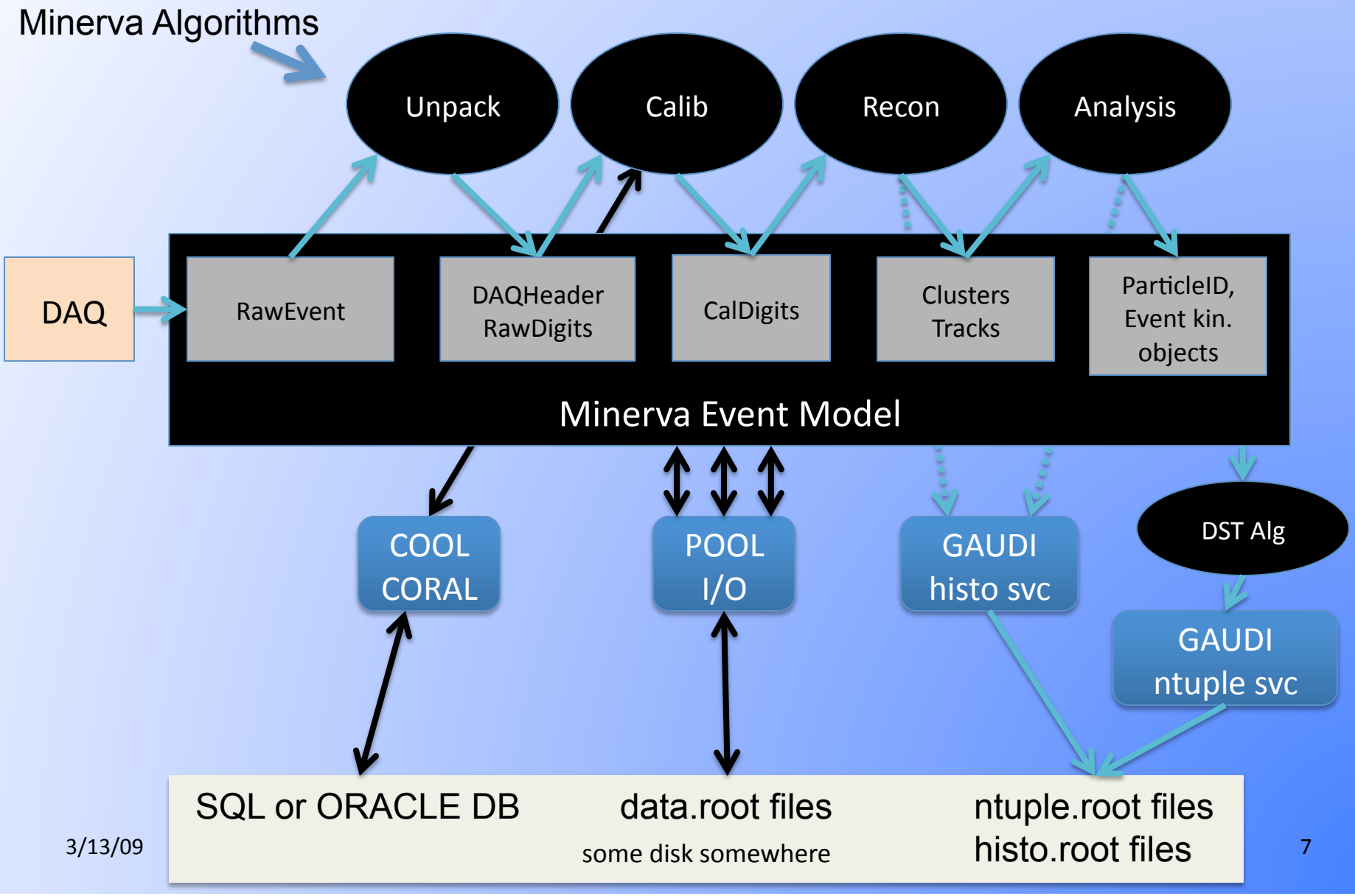
- but because we use binaries, have not extracted these components from rest of LHCb base ☹ working on it now.

- but if running SL4, very easy to install. source an install script, go for a coffee – or maybe dinner.

Framework

- What framework is used for reconstruction?
- What is used for simulation?
- What is used for data analysis?
- GAUDI allows that all three can be nicely integrated into a single framework
- POOL ➔ automatic I/O of entire event model at any stage of processing (Persistency for arbitrary transient C++ objects. No need to write converters)
- built in histogram and ntuple services makes doing high level analysis within the framework reasonable
  - avoid many independent ROOT-based analyses which each build analysis tools from scratch
- did create a DSTWriter (flat root tree) for doing studies of data, making plots, developing analysis, etc.

Software Framework



Describe a full analysis chain from raw data to paper

- we have papers?!
- see previous slide

Describe a full simulation chain from random seed to paper

- NuMI fluxes as 1D energy histogram – looking forward to moving to full numi ntuple fluxes
- GENIE neutrino event generator (more later)
- generated events loaded into framework and GEANT4 detector simulation (GiGa interface)
- custom optical model of scintillator, PMT and electronics readout simulation coded up as GAUDI algorithms
- and into the reconstruction and analysis stages…

Describe your calibration procedures

- working on implementing and testing all of this now, but…
- use a light injection system to monitor PMT gains
- measured the charge to ADC conversion of each front-end-board by injecting charge directly onto board
- attenuation in scintillator measured by systematic scan with radiation source of each module before installing
- timing calibration done with light injection system
- currently building a small, replica test detector to run at MTest and measure response to pions, protons of known energy

Describe alignment procedures

- have taken survey data as constructed on surface, will again below ground
- working now on alignment using cosmic ray data taken at surface
- can see evidence of small rotations and shifts in planes
- working on feeding back the measured corrections now

Fortran or C++?

- C++

What works really well?

- we are just getting started exercising and developing the system with our first real data, so as with many of the speakers, "ask again in a year or so."
- but the GAUDI framework seems a nice environment for analysis chain so far
- automatic Event Model object I/O into ROOT format using POOL. No hand written converters necessary.
- flexibility of COOL to connect to MySQL or ORACLE data bases
- small experiment able to use available products from other experiments and get something up and working quickly

What would you not do again?

- get something up and working so quickly
- but seriously, would like to move towards a more tailored set of code which we can build from source… and we are.