FAIR Simulation & Analysis Framework FairRoot

Mohammad Al-Turany (GSI-IT)
Florian Uhlig (GSI-IT)

Overview

- FairRoot Features
- CMake: Testing and building system
- Grid
- GPU and CUDA
- Integrated Track follower (Geane)
- Event display
- Eve & Geane
- Summary

Features

- No Executable: (Only Rootcint)
  - Compiled Tasks for reconstruction, analysis, etc.
  - Root macros for steering simulation or reconstruction
  - Root macros for configurations (G3, G4, Fluka and Analysis)
- VMC and VGM for simulation:
- Reconstruction can be done directly with simulation or as a separate step
- RHO Package for Analysis
- TGeoManager in Simulation and Reconstruction

Features

- Dynamic Event structure based on ROOT tree
- Hades oracle interface and run time data base
- Hades Geometry Interface.
  - G3 Native geometry
  - Geometry Modeller (TGeoManager)
  - Different geometry input format
- Grid: we use AliEn!
- CMake: Makefiles, dependencies , QM
- Doxygen for class documentation

FairRoot Deliver:

- Main Application based on VMC Application
- Detector base classes that handle initialization, geometry construction, hit processing(stepping action), etc.
- IO Manager based on ROOT TFolder and TTree (TChain)
- Geometry Readers, ASCII, ROOT, CAD2ROOT
- Radiation length manager
- Generic track propagation based on Geane
- Generic event display based on EVE and Geane
- Oracle interface for geometry and parameters handling
- Fast simulation base services based on VMC and ROOT TTasks. (Full and Fast simulations can be mixed in one run)
- Interfaces for some event generators, Pythia, Urqmd, Evtgen, Pluto, Dpm, ...

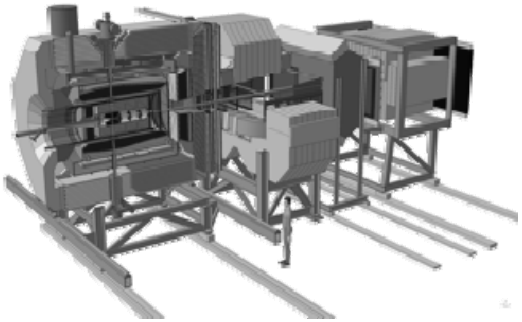
FairRoot

FairRoot

HadesRoot CbmRoot PandaRoot MDPRoot

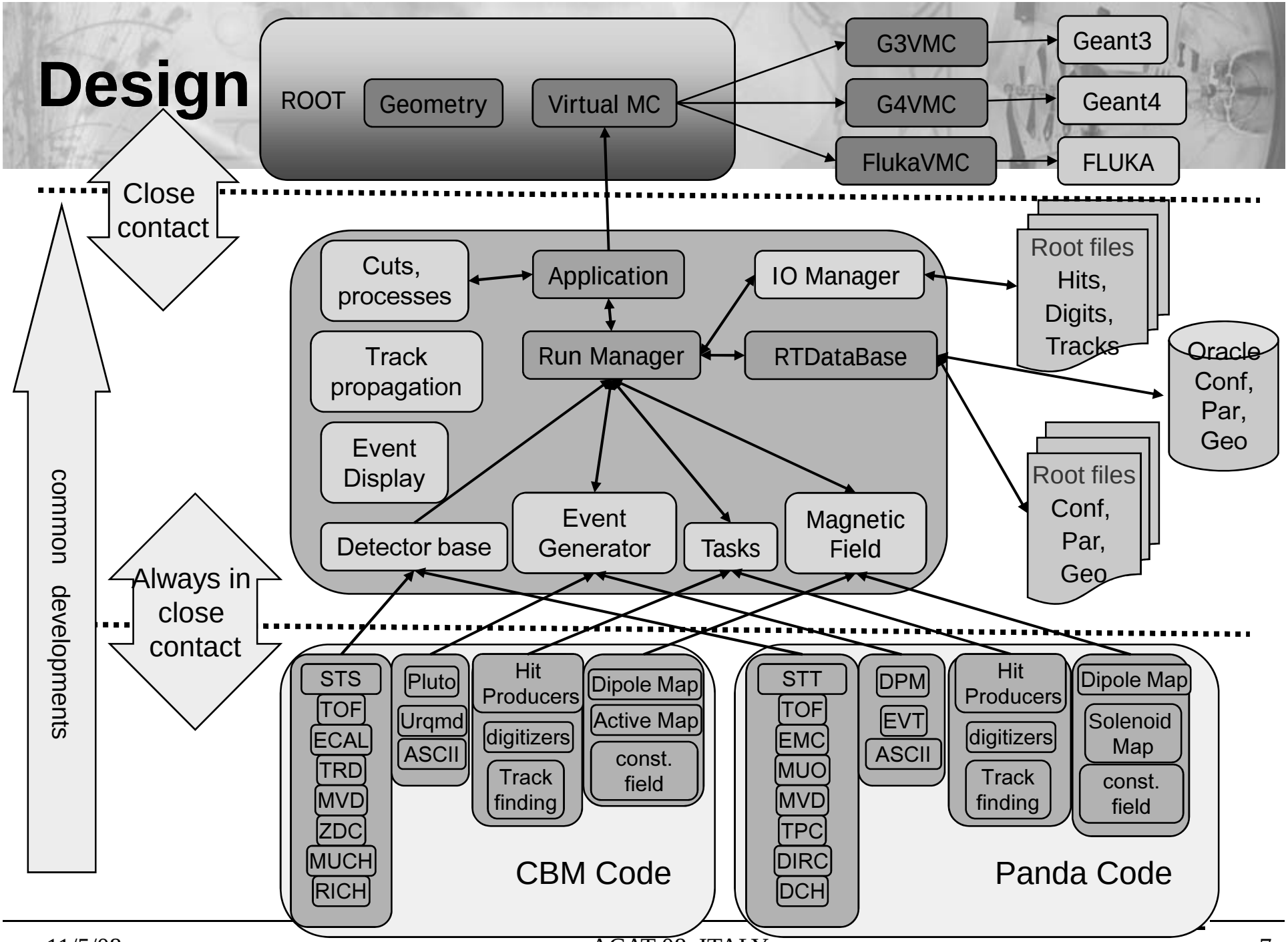
QuickTime™ and a
decompressor
are needed to see this picture.

QuickTime™ and a
decompressor
are needed to see this picture.



QuickTime™ and a
decompressor
are needed to see this picture.

Design


ROOT
Geometry
Virtual MC
G3VMC
G4VMC
FlukaVMC
Geant3
Geant4
FLUKA
Close contact
common developments
Always in close contact
Cuts, processes
Application
IO Manager
Track propagation
Run Manager
RTDataBase
Event Display
Detector base
Event Generator
Tasks
Magnetic Field
Root files Hits, Digits, Tracks
Oracle Conf, Par, Geo
Root files Conf, Par, Geo
STS
TOF
ECAL
TRD
MVD
ZDC
MUCH
RICH
Pluto
Urqmd
ASCII
Hit Producers
digitizers
Track finding
Dipole Map
Active Map
const. field
CBM Code
STT
TOF
EMC
MUO
MVD
TPC
DIRC
DCH
DPM
EVT
ASCII
Hit Producers
digitizers
Track finding
Dipole Map
Solenoid Map
const. field
Panda Code

Building system: History/Motivation

- Start with self written Makefiles
  - Need work when porting to another platform

- Autotools (autoconf, automake, etc.)
  - Standard for *ix systems
  - Easy to use for user (./configure && make && make install)
  - Different macro languages for different tools in chain
  - „Autohell“ if there is a problem, even a blank character at the wrong position
  - No test system

- Cmake/Ctest/Dashboard

CMake

- It supports great variety of platforms
  - (basically every *ix, Mac OS, Sun Solaris, Windows)
  - (Win: Borland, MS Visual C++, cygwin, mingw)
- CMake generates out of simple rules native Makefiles for all supported platforms
- produce project files for IDE's (KDevelop, XCode, VStudio)
- Input files (rules) are the same on all platforms
- Big community behind it, CMake is the build tool for KDE 4
- CMake has a testing framework

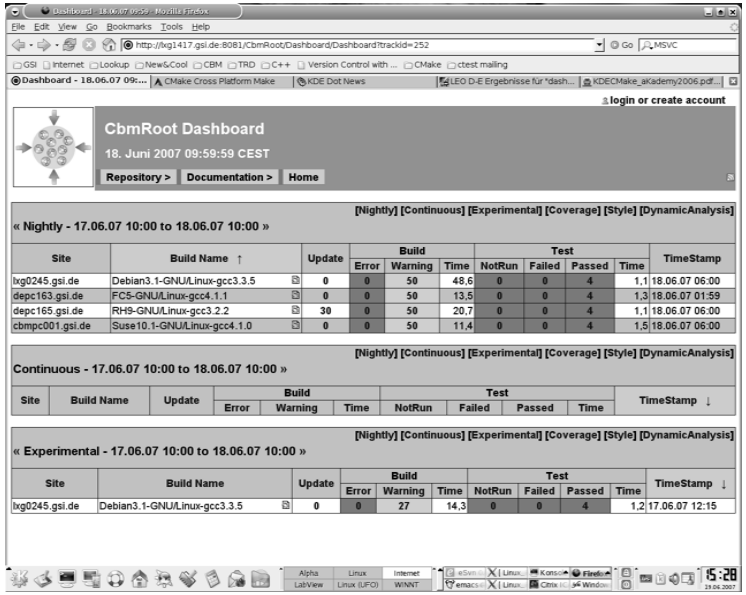
Daily tests:

- Large code base is too large/complicated for a single developer to understand/maintain
- Identify problems when they occur
- Project depends on external packages which can cause problems
- Provide direct feedback to the developers as they experiment with new features

Software Process Dashboards

SVN maintains source code revision

CTest/CMake compiles and test the newly commited source code on distributed clients



Typical developer checks in code

Developer reviews the results

Dashboards

- Client/Server architecture
- Cross platform testing
- Memory testing (purify, valgrind)

If it's not tested it's not working

- Coverage testing
- Create documentation on a nightly basis (Doxygen)
- Check coding conventions (Rule Checker)

Dashboard example

Build - demac004v1 - FC9-GNU/Linux-gcc4.3.0 - Nightly - 11/1/08 12:59 AM

http://lxg1417.gsi.de:8081/PandaRoot/Dashboard/Build?submissionid=6162

Apple Yahoo! Google Maps YouTube Wikipedia News (528) Popular

[Top] [Configure] [Build|E|W]

Build Errors (4)

« Nightly - 10/31/08 9:00 PM to 11/1/08

Site
lxg0245.gsi.de
opteron.cluster.rug.nl
opteron.cluster.rug.nl
demac004v1
depc165.gsi.de
demac004.gsi.de
kviiq41.kvi.nl

« Continuous

Site
lxir010.gsi.de

« Experimental - 1

Site
ponni
ponni
demac004v2.gsi.de
demac004v1
demac004v1
demac004.gsi.de

Updates - demac004v1 - FC9-GNU/Linux-gcc4.3.0 - Nightly - 11/1/08 12:59 AM

http://lxg1417.gsi.de:8081/PandaRoot/Dashboard/Update?submissi

Apple Yahoo! Google Maps YouTube Wikipedia News (528) Popular

login or create account

PandaRoot Update - demac004v1 - FC9-GNU/Linux-gcc4.3.0 - Nightly

November 1, 2008 12:59:00 AM CET

Repository > | Documentation > | Rule Checker | Home

ElapsedTime	1
EndDateTime	Nov 01 05:46 CET
StartDateTime	Nov 01 05:45 CET
UpdateCommand	"/usr/bin/svn" status
UpdateType	SVN

Changed files as of 11/1/08

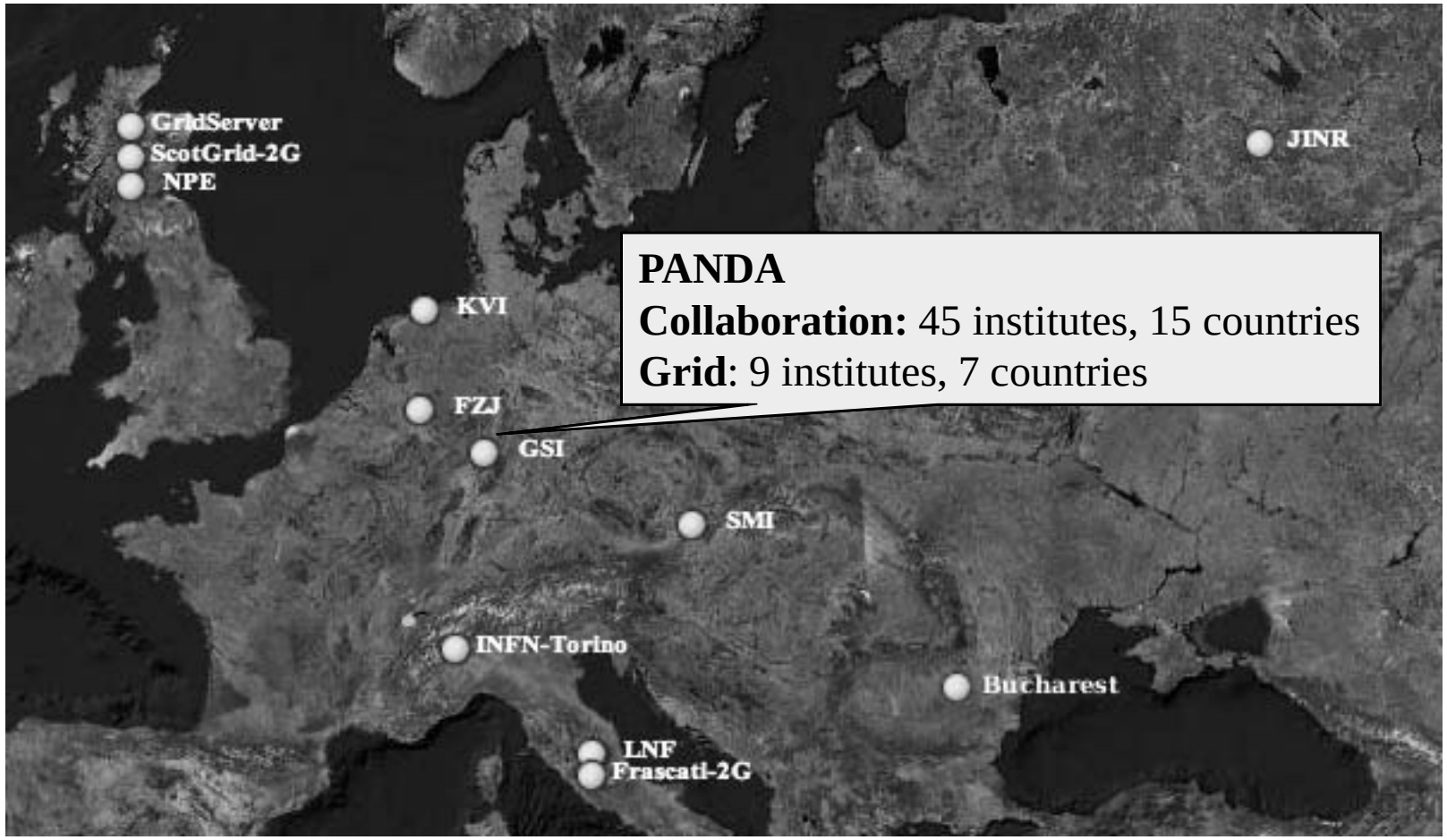
[Expand all | Collapse all]

Updated files (80)

tpc/TestChamber/hexReadout Revision: 3939 by hoeppner
Mapping tools for new hex readout of testchamber

tpc/TestChamber/hexReadout/Mapping.h Revision: 3939 by hoeppner
Mapping tools for new hex readout of testchamber

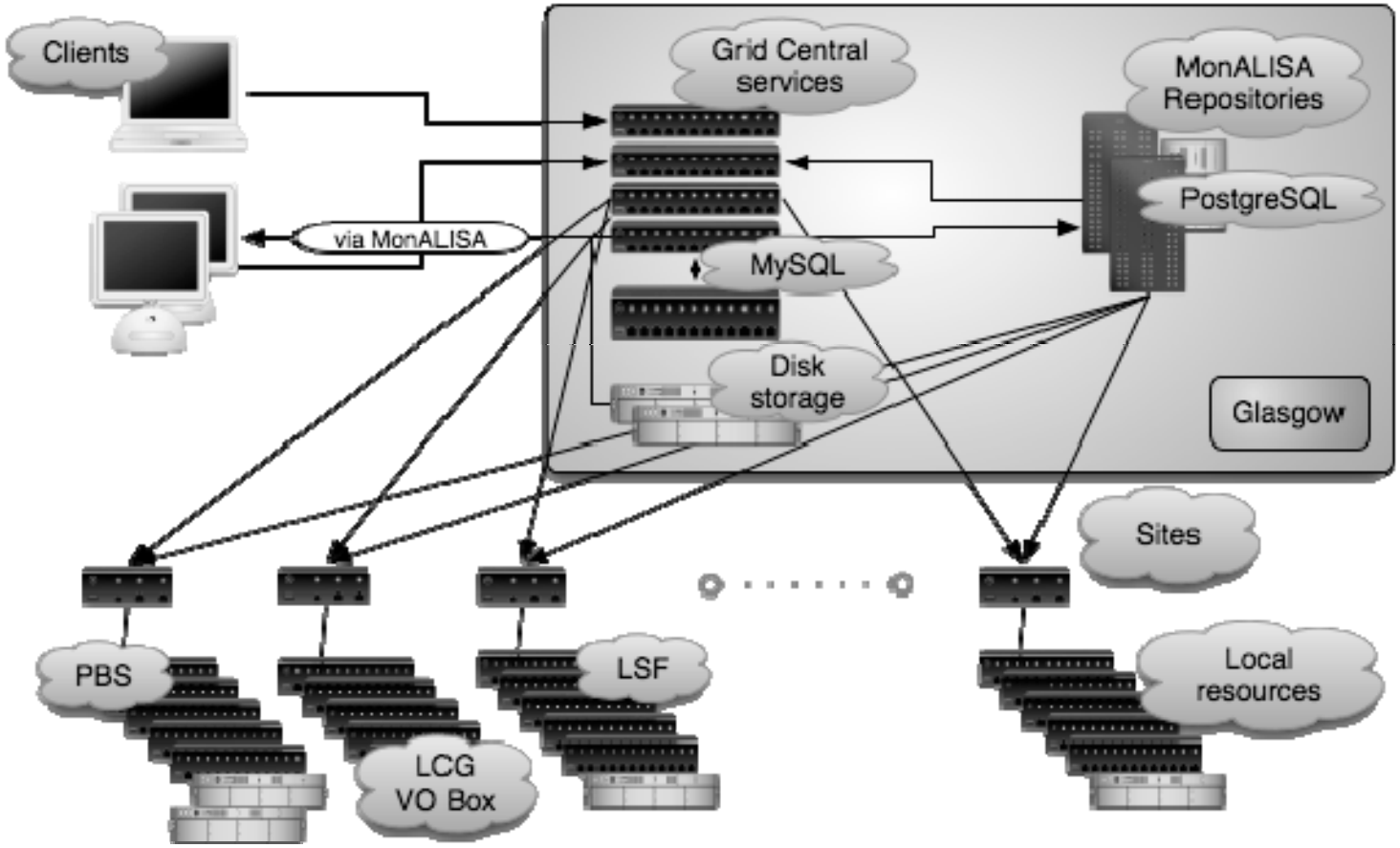
Grid



PANDA
Collaboration: 45 institutes, 15 countries
Grid: 9 institutes, 7 countries

Dan Protopopescu

Structure



Dan Protopopescu

Monitoring

MonALISA tools - with excellent support from Costin

MonaLisa Services Status

Site	Online	ML Version	Machine	Java Version	libc	auto update	Collected Values Rate (1/s)	Current Param No	ML CPU Time (%)	Free Memory (MB)	Memory (MB)	Memstore (hours)	Contact Person
1. Ateneo		1.8.6	medgrid.ateneo.edu	1.6.0_02	libc-2.3.4.so	true	4.08	1213	0.2	21.53 MB	94 MB	4.226	Allan Espinosa <allane@acm.org>
2. Bucharest		1.8.6	panda01.nipne.ro	1.6.0_02	libc-2.3.4.so	true	20.99	1271	1.482	18.08 MB	84.31 MB	1.876	Titi Preda <tpreda@nipne.ro>
3. Dubna		1.8.6	lxpanda.jinr.ru	1.6.0_02	libc-2.3.4.so	true	3.365	1092	0.716	15.38 MB	88.56 MB	1.785	Nikolay Kutovskiy <nikolay.kutovskiy@jinr.ru>
4. Frascati1		1.8.6	ed22lf.lnf.infn.it	1.6.0_02	libc-2.3.2.so	true	4.535	1150	0.465	26.42 MB	94.06 MB	3.351	Federico Ronchetti <dedro@lnf.infn.it>
5. Frascati2		1.8.6	ed22af.lnf.infn.it	1.6.0_02	libc-2.3.4.so	true	14.02	897	4.664	67.37 MB	93.81 MB	3.289	Federico Ronchetti <dedro@lnf.infn.it>
6. GSI	2008-02-21...	1.8.6	grid8.gsi.de	1.6.0_02	libc-2.3.2.so	true	4.526	1267	0.266	44.52 MB	95.31 MB	3.019	Kilian Schwartz <k.schwartz@gsi.de>
7. GridServer		1.8.6	panda.gla.ac.uk	1.6.0_02	libc-2.3.4.so	true	15.67	1407	0	20.94 MB	62.69 MB	1.348	Dan Protopopescu <protopop@jlab.org>
8. Juelich		1.8.6	ikp663.ikp.kfa-juelich.de	1.6.0_02	N/A	true	18.3	1116	1.166	19.01 MB	95.31 MB	3.555	Rene Dosdall <r.dosdall@fz-juelich.de>
9. KVI		1.8.6	kvip81.kvi.nl	1.6.0_02	libc-2.3.3.so	true	7.346	1158	0	18.29 MB	95.31 MB	3.452	Johan Messchendorp <messchendorp@kvi.nl>
10. NPE		1.8.6	npcfs.physics.gla.ac.uk	1.6.0_02	libc-2.3.4.so	false	20.57	1898	0	26.88 MB	61.88 MB	0.673	Dan Protopopescu <protopop@jlab.org>
11. Pavia		1.8.6	lxfocus14.pv.infn.it	1.6.0_02	libc-2.3.2.so	true	3.597	1212	0.083	54.71 MB	95.31 MB	3.261	Gianluigi Boca <boca@pv.infn.it>
12. ScotGrid		1.6.22	svr024.gla.scotgrid.ac.uk	1.5.0_12	libc-2.3.2.so	true	12.09	857	0.05	20.38 MB	63.56 MB	1.802	Graeme Stewart <g.stewart@physics.gla.ac.uk>
13. Torino1		1.8.6	pandafarm01.to.infn.it	1.6.0_02	libc-2.3.2.so	true	4.742	836	0.25	54.34 MB	95.31 MB	3.521	Antonio Amoroso <amoroso@to.infn.it>
14. Torino2		1.8.6	pandafarm02.to.infn.it	1.6.0_02	libc-2.3.4.so	true	4.781	814	0.233	20.4 MB	95.31 MB	3.837	Antonio Amoroso <amoroso@to.infn.it>
15. Vienna		1.8.6	smigrid01.smi.oeaw.ac.at	1.6.0_02	libc-2.3.2.so	true	3.356	194	0.366	81.03 MB	95.31 MB	0.042	Paul Buehler <paul.buehler@oeaw.ac.at>
Total	**14**						**142**	**16382**					

	DONE		ERRORS	
00%	100			
00%	100			
99%	99	1%	1	
00%	100			
00%	100			
00%	100			
		98%	49	
90%	9			
00%	100			

78900	pbartest	22.02.2008 17:30	**22.02.2008 21:00**	100	1												100%	100			
78879	pbartest	22.02.2008 15:01	**22.02.2008 16:15**	20	1												100%	20			
78858	pbartest	22.02.2008 15:01	**22.02.2008 15:30**	20	1												100%	20			
74868	pbartest	22.02.2008 15:01	22.02.2008 15:01	0	1																

http://mlr1.gla.ac.uk:7001/

PANDA Grid ▾ Alerts (2) ▾ 16 Jobs Running – ▾ Admin Section ▾ Wikis ▾

Dan Protopopescu

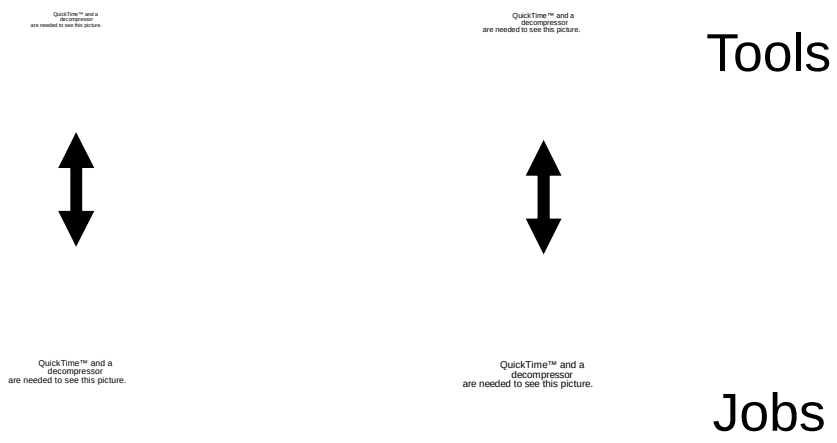
Nvidia's Compute Unified Device Architecture (CUDA)

- CUDA development tools work alongside the conventional C/C++ compiler, so one can mix GPU code with general-purpose code for the host CPU.
- CUDA Automatically Manages Threads:
  - it does not require explicit management for threads in the conventional sense, which greatly simplifies the programming model.
- Developers must analyze data structure
  - Determine how best to divide the data into smaller chunks for distribution among the thread processors.
- FindCuda.cmake (Abe Stephens SCI Institute)
  - Integrate CUDA into FairRoot very smoothly

See: www.MPRonline.com

CUDA & GPUs

- One of the major features of the GT200 GPU and cards based on it, is the support for hardware double precision floating point operations. These cards are already available!
- OpenCL is scheduled to be introduced in Mac OS X v10.6 ('Snow Leopard').



What is GEANE?

- Package to calculate the **average trajectories** of particles through **dense materials** and to calculate the **transport matrix** as well as the propagated errors **covariance matrix** in a given track representation.

- It is a **track follower**: it predicts the trajectory of a charged particle in terms of mean values and errors both in forward and in backward direction. **Three effects** are taken into account:

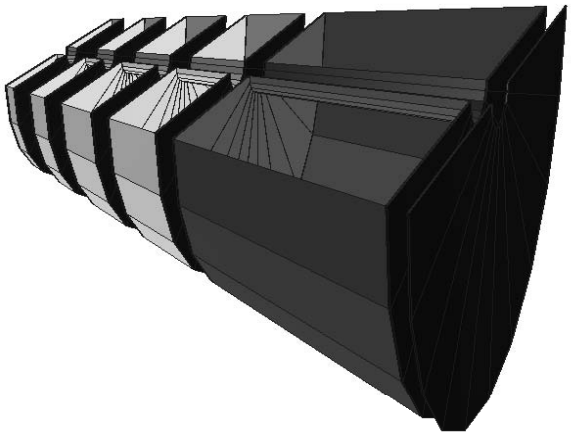
  - energy loss (affects mean values and errors)
  - Coulomb multiple scattering (affects errors only)
  - magnetic field (affects mean values only)

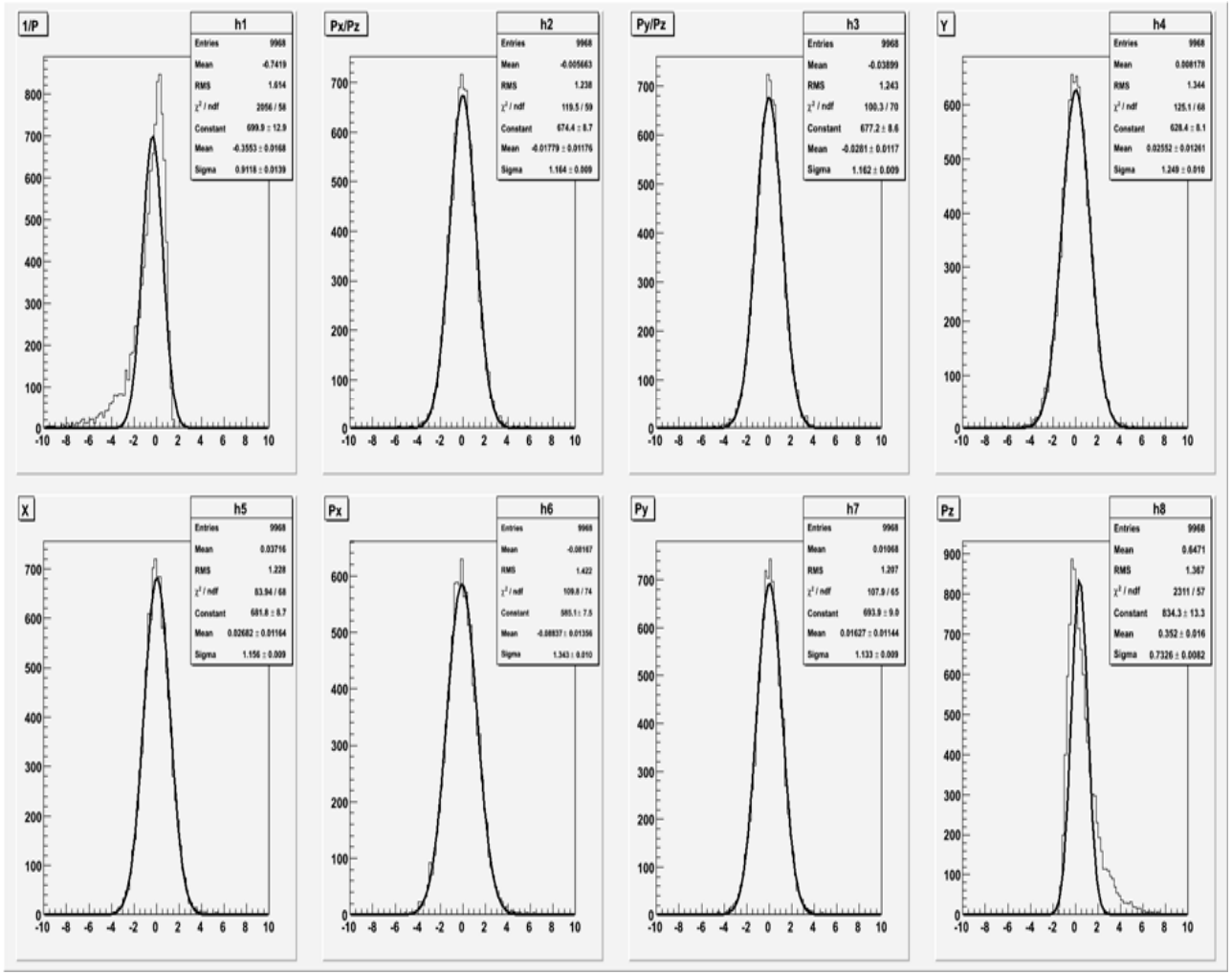
Geane Integration in FairRoot

- It is integrated in VMC (TGeant3)
- In FairRoot:
  - Geane can be used in the analysis or reconstruction
  - Propagation to
    - Length
    - Plane
    - Volume (Enter or Exit point)
    - To Line
    - To Point
    - Point of closest approach on a wire

New Methods introduced by Pavia group in PANDA

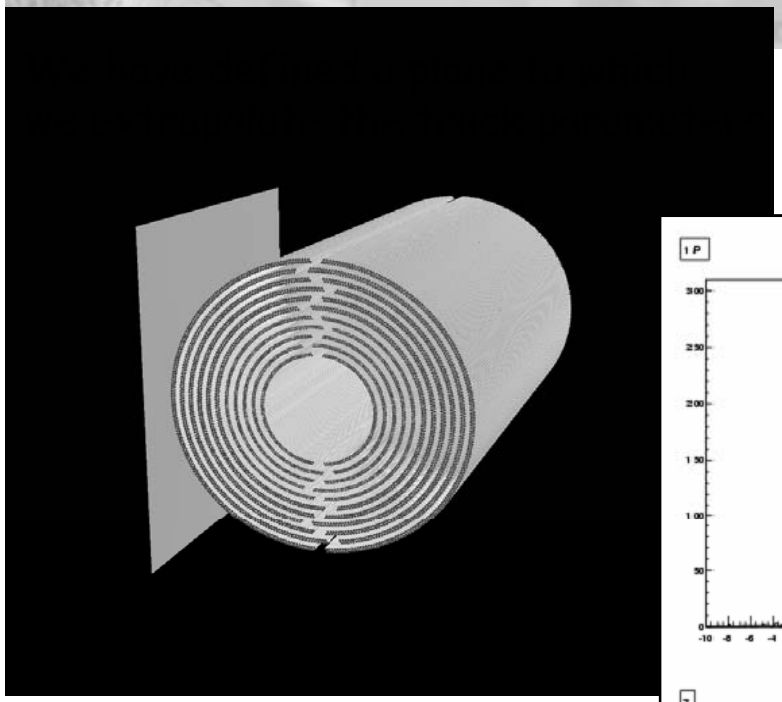
Muon Absorber in CBM

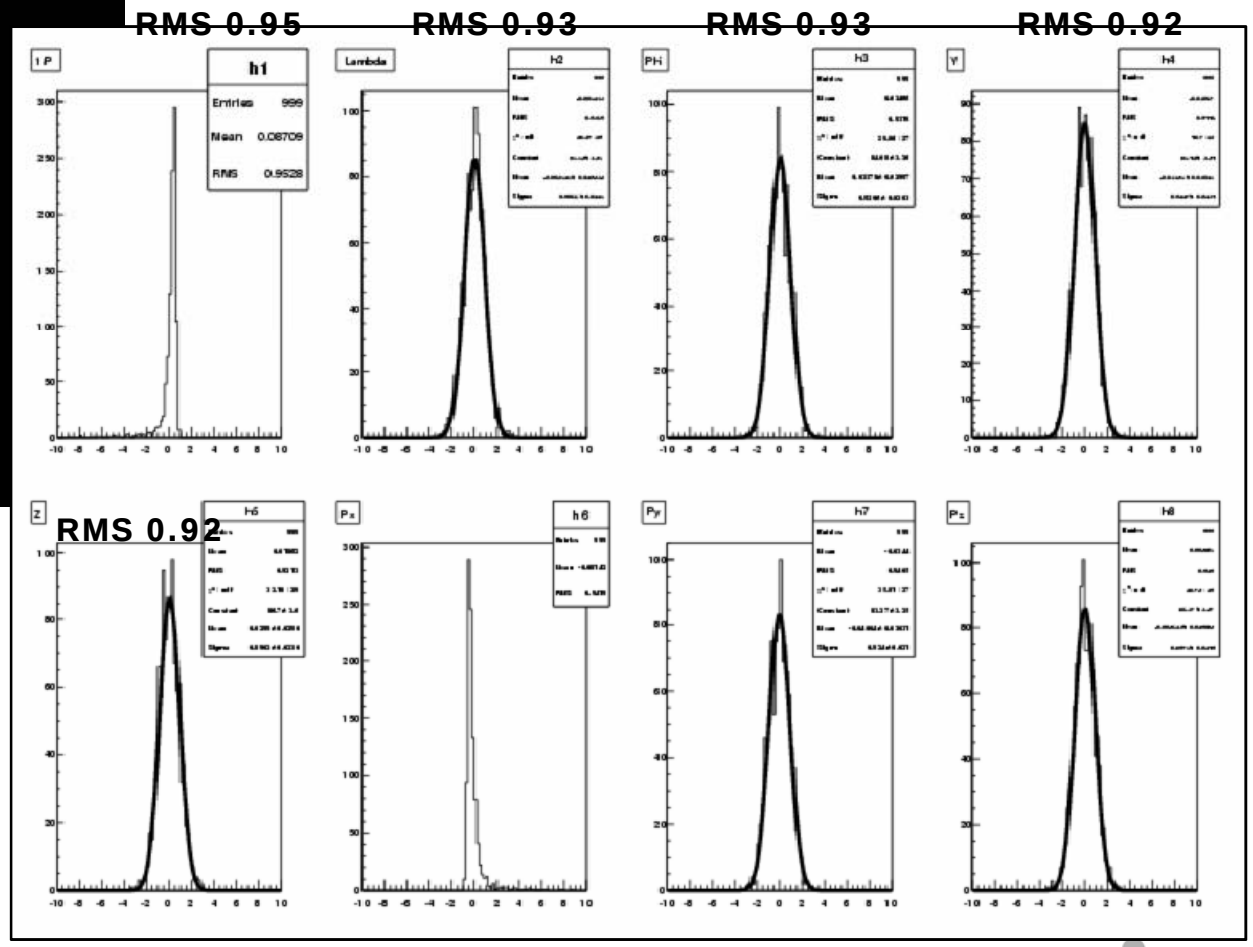




Pulls for the Panda STT

500 MeV/c

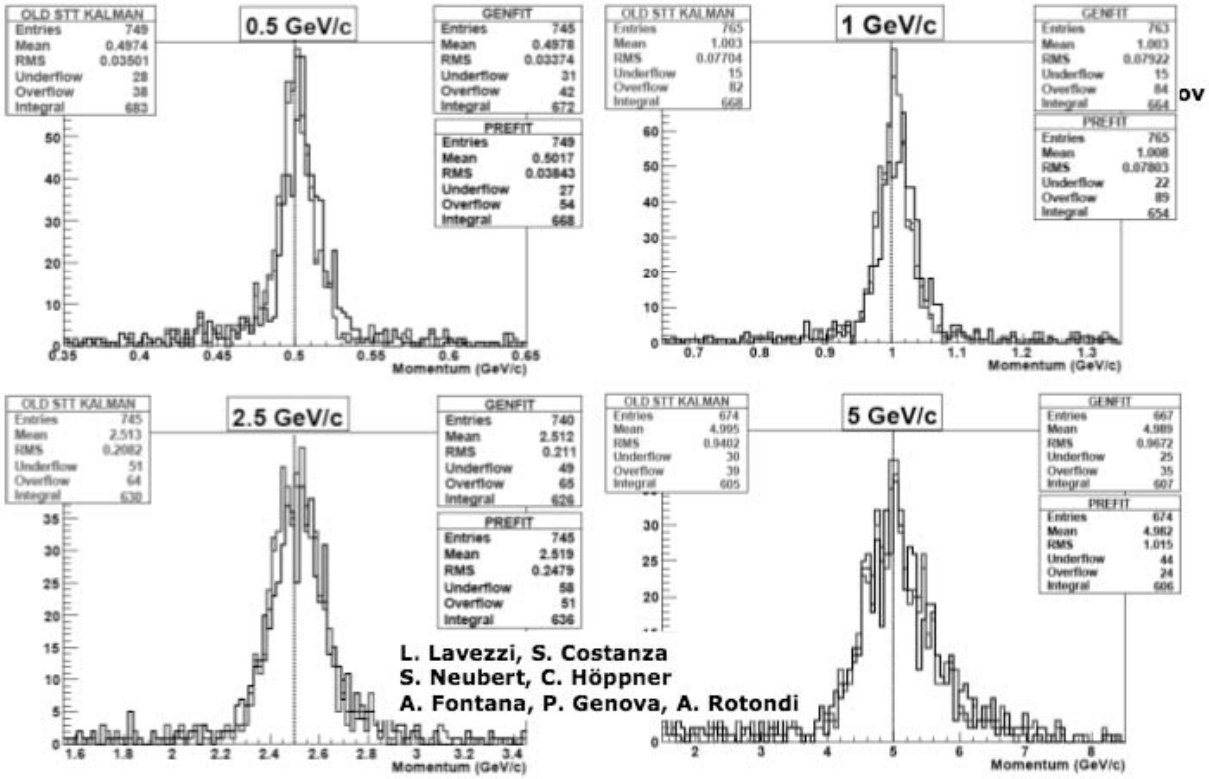


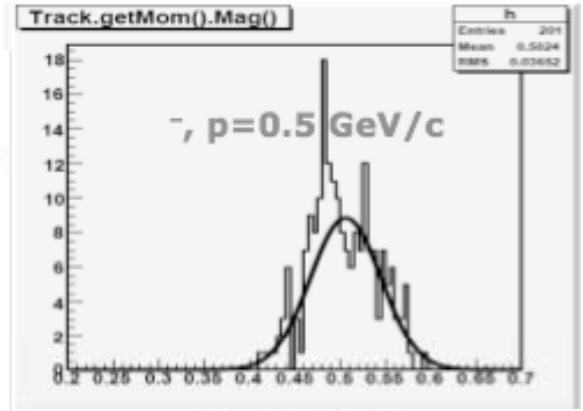


Kapton	1.42	g/cm3
Isobutan	2.7e-3	g/cm3

Geane + Kalman (Panda central tracker)

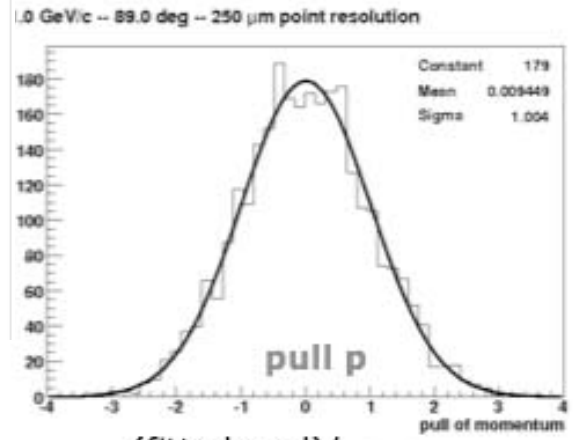
L. Lavezzi, S. Costanza
S. Neubert, C. Höppner
A. Fontana, P. Genova, A. Rotondi





p_{fit} / GeV/c

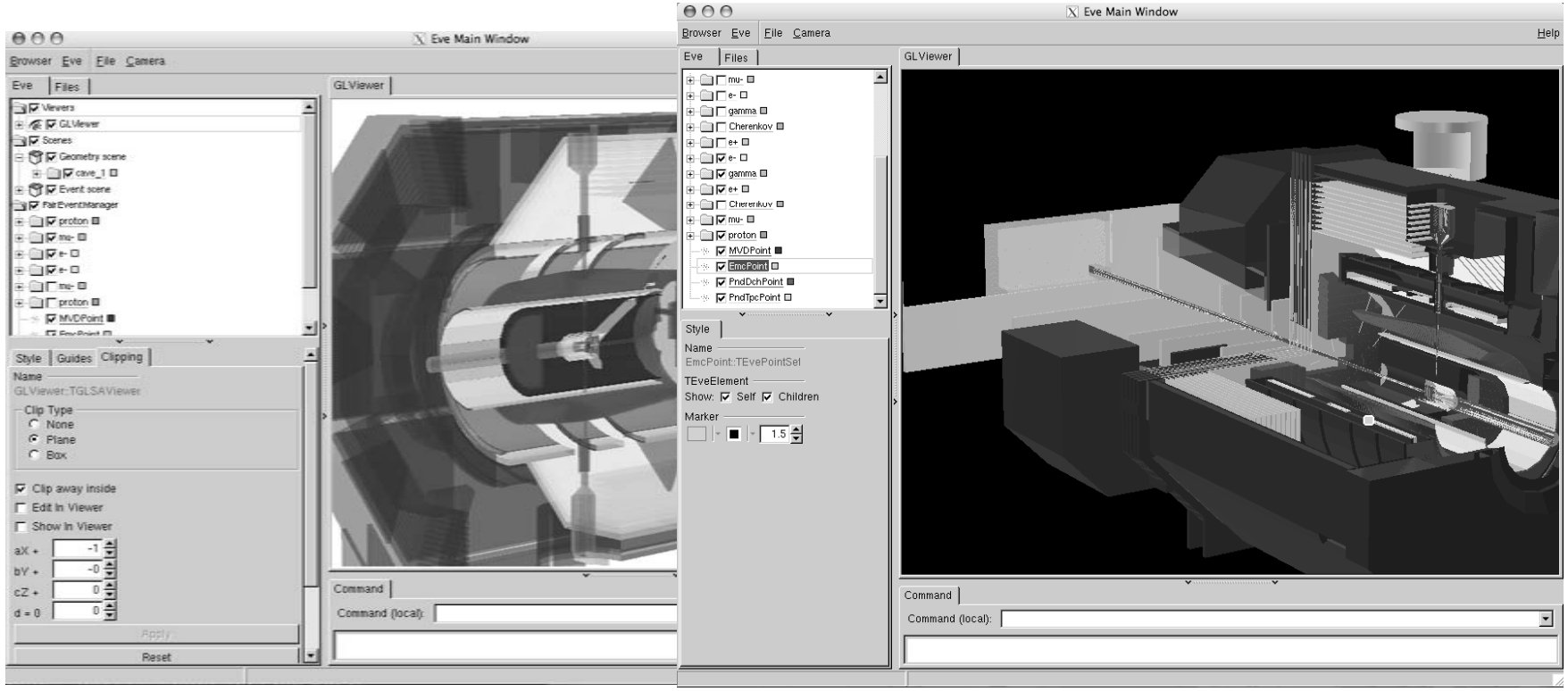
⁻, p=1.0 GeV/c



(fitted-real)/ fitted

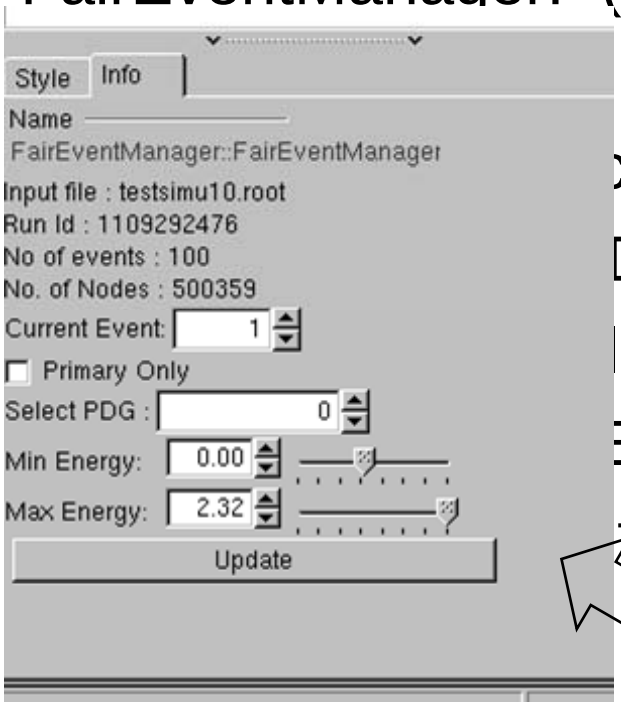
Eve in FairRoot

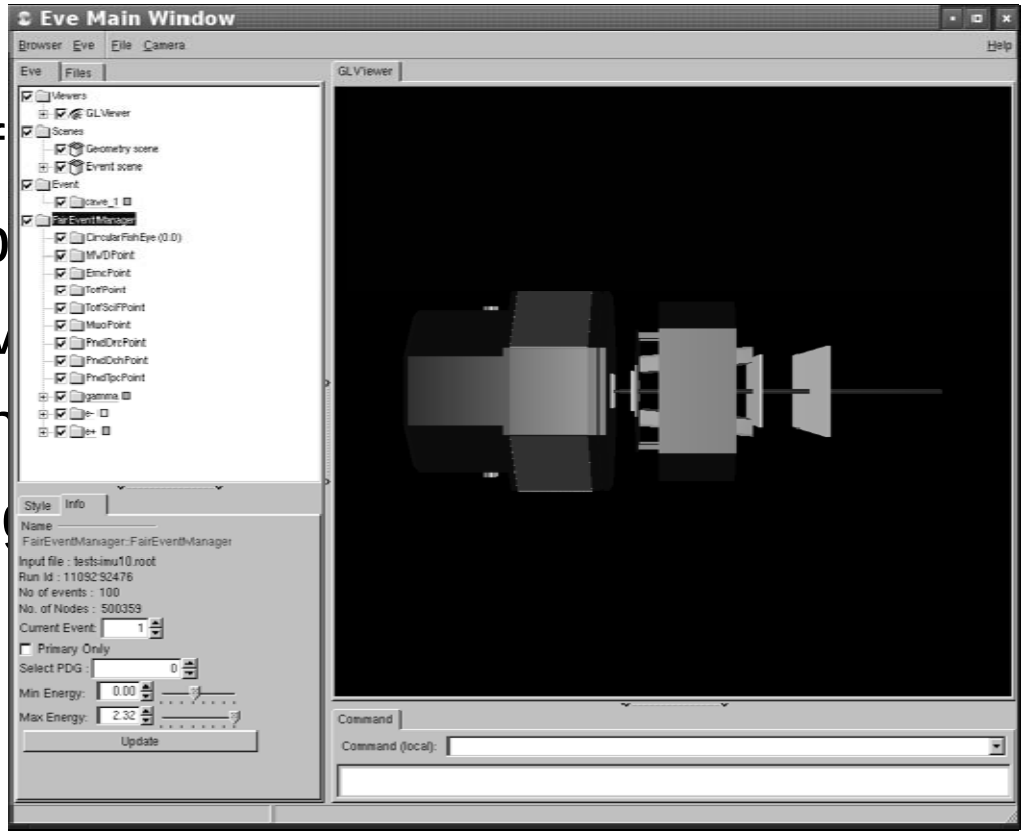
- Integration is straight forward
- Some features can be used directly (even from macro)



FairEventManager

- FairEventManager: (Sub-class of TEveEventManager)

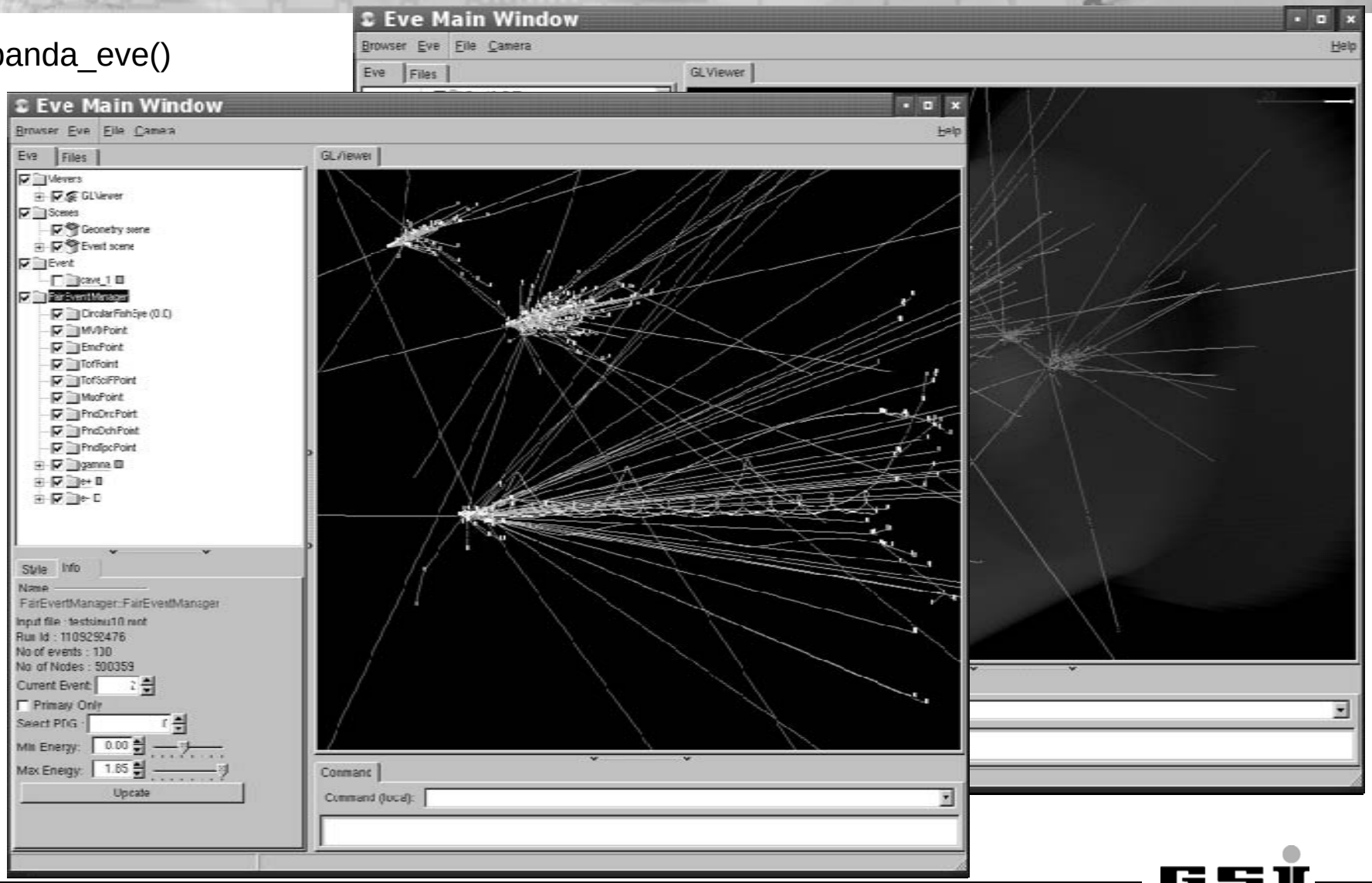




Event Display Macro

panda_eve()

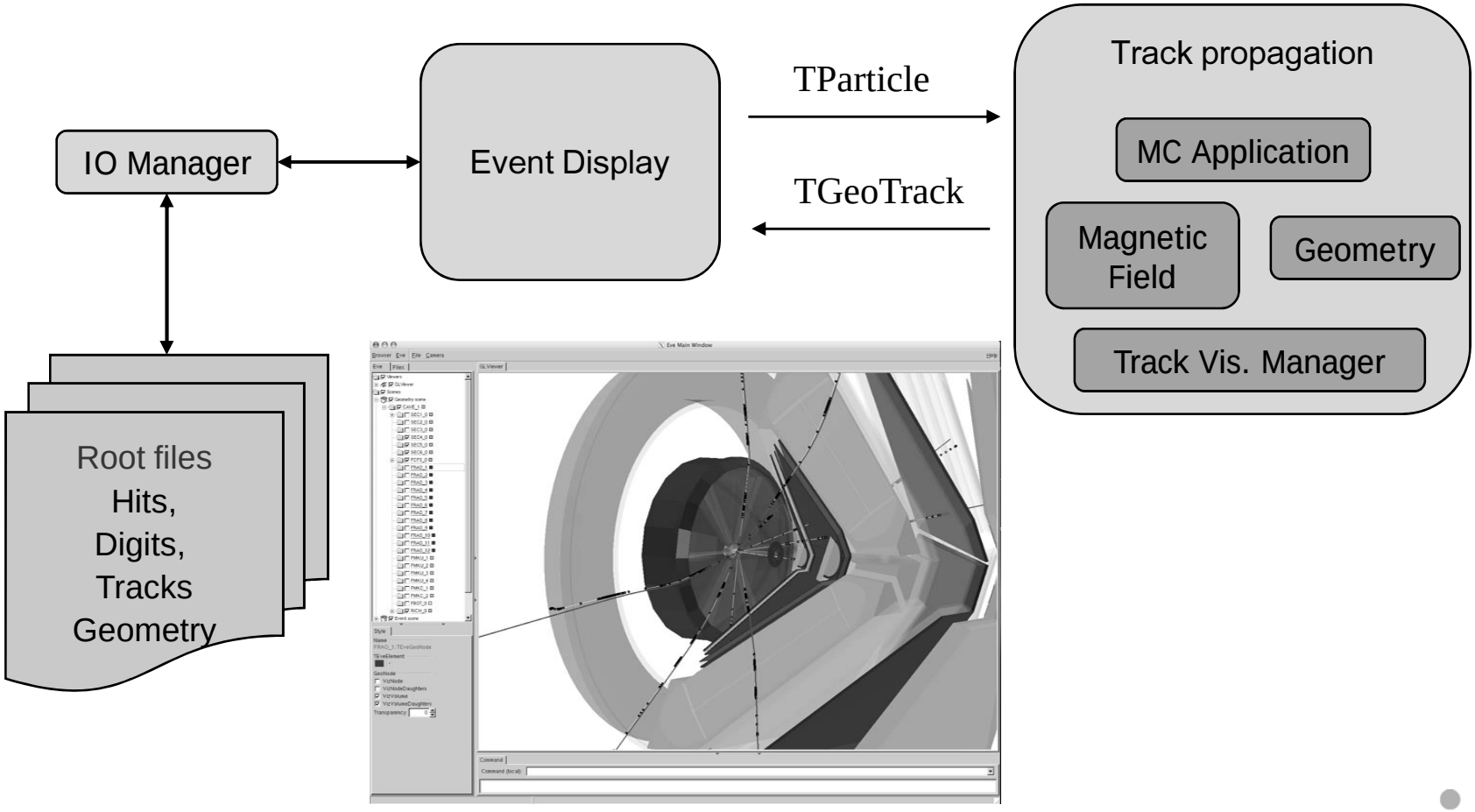
{



}

Eve & Geane

- Track propagation is done online for the selected tracks

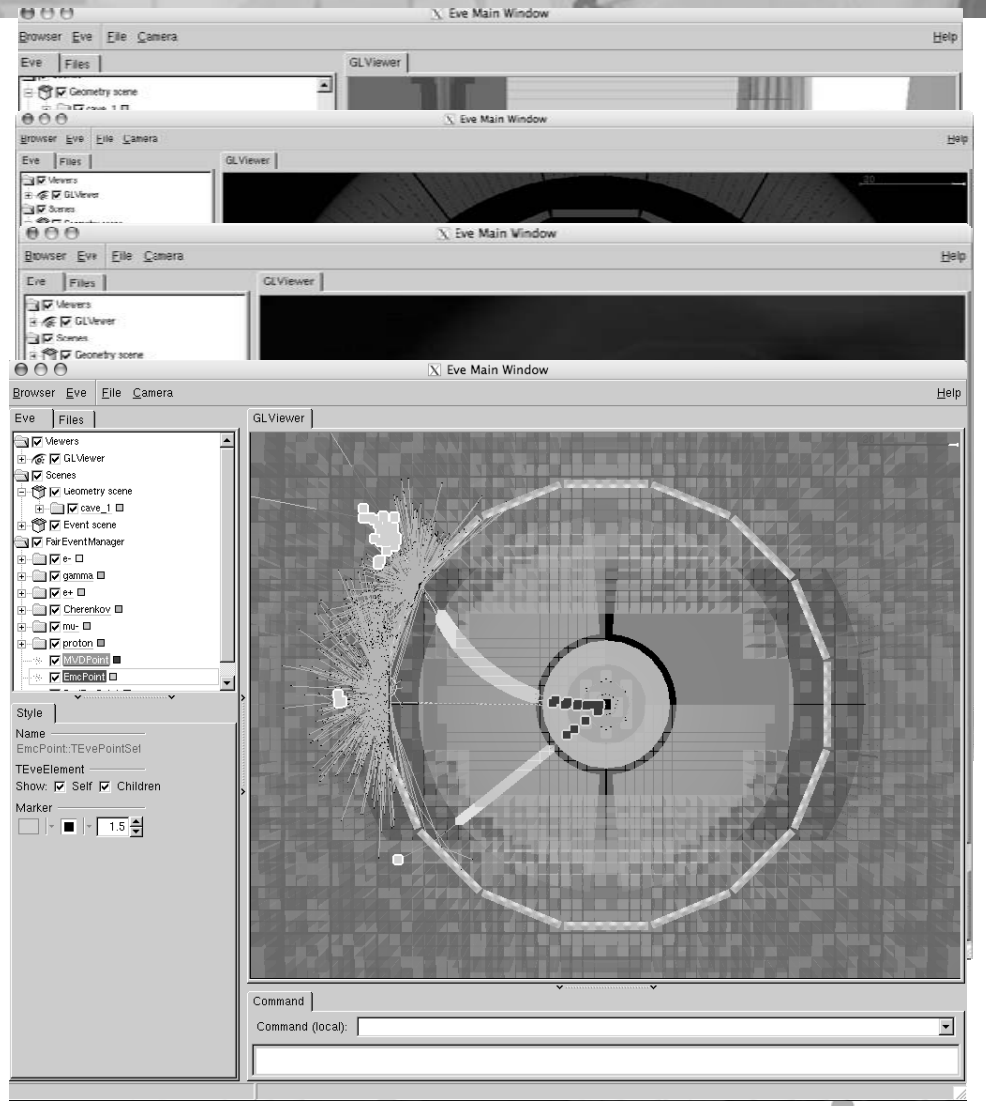


Geane Propagator for EVE

- Geane is used as propagator but there is no path markers during propagation!

- Track parameters at the vertex are enough to display the track.

- Using path markers will improve the quality of propagate tracks!



Summary

- Cmake/Ctest build and test system is doing well
- Geane is used with different tracking methods
- Integration of Geane and Eve proof to be useful, still need to have an editor for this propagator
- CUDA looks very attractive for us, but have to be careful where and how to use it!