Computing Infrastructure

Warning! have not done anything for muon collider detector studies.
Came here to learn and find out the challenges and the Requirements. Have experience in setting up computing infra structure at Fermilab for CMS!

Hans Wenzel

Fermilab

Muon Collider 2011

Physics — Detectors — Accelerators

June 27-July 1, 2011 — Telluride

Problems Detector R&D for μ -Collider is facing

- Huge amount of BGR. (several hundred TeV dumped into the detectors.) What can we learn from simulation for SLHC/LHC.
- Huge amount of CPU resources are required, need infrastructure and storage to overlay bgr. Events.
- ILC detector concepts not a realistic model for muon collider detectors (LHC-detectors a more realistic model?). Fast timing requirements → a higher thermal load → needs cooling and will mean a more massive detector.
- Timing needs Hit Classes which allow for timeslices (as the LHC experiments).
- LC-colliders are advertised as high precision machines but how much is the precision compromised after BGR. is added and more realististic detectors are used that can deal with the BGR. Level → Needs detailed studies including full simulation and reconstruction Signal + Bgr.

Computing resources currently available for μ–collider Detector R&D at Fermilab

- Modest:
  - Fermigrid; ilc/lc VO right now there's an allocation of 200 slots. Plus opportunistic use.
  - 4th concept disk resources are being used for ilcroot studies.
  - Two ILC frameworks available for muon collider related detector R&D studies ilcroot + lcsim → both good to jumpstart a detector R&D effort.

    But probably both need development.

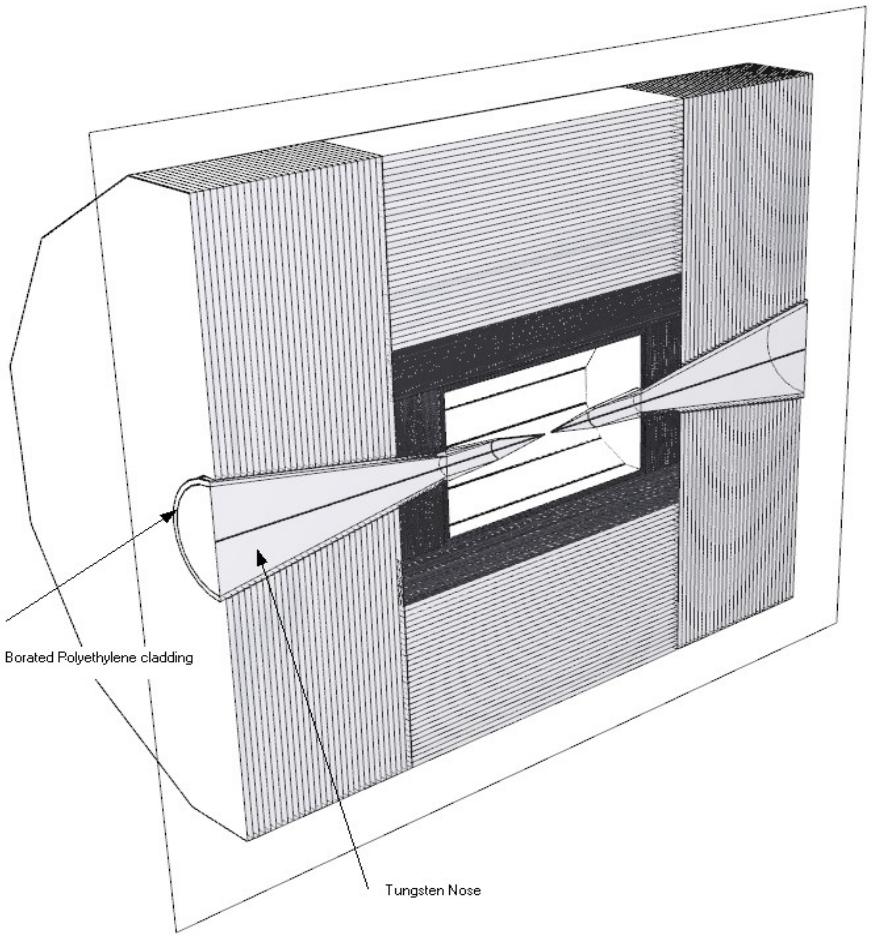
Needs to be scaled up to be able to do the necessary studies.

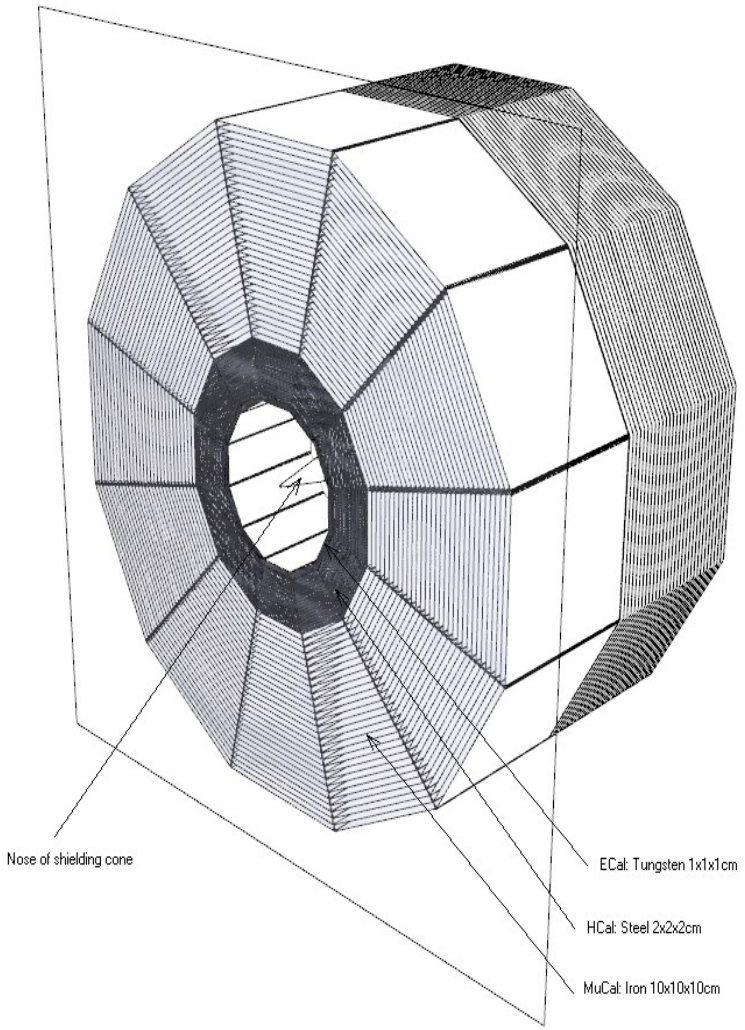
But I was impressed with what was already achieved with the limited resources (see talks at this workshop)

But computing resources are only one aspect

- If we want to make progress we need to take this seriously!
- Goal should be to create the infrastructure to attract physicists to work on the muon collider:
  - Guarantee that software is working (on a reference platform)
  - Kept up to date
  - Documented
  - There are experts who can help
  - Enough computing resources are available.
  - documentation: use automatic systems like doxygen, lxr,
  - code repositories : SVN
  - ...
- Start using the software/computing resources should be painless and easy
  - Procedure for visitors ID, computing account creation, grid certificate, VO membership etc. has to be in place and well documented. (web based)
  - Get started instructions have to be in place
  - Well advertised.

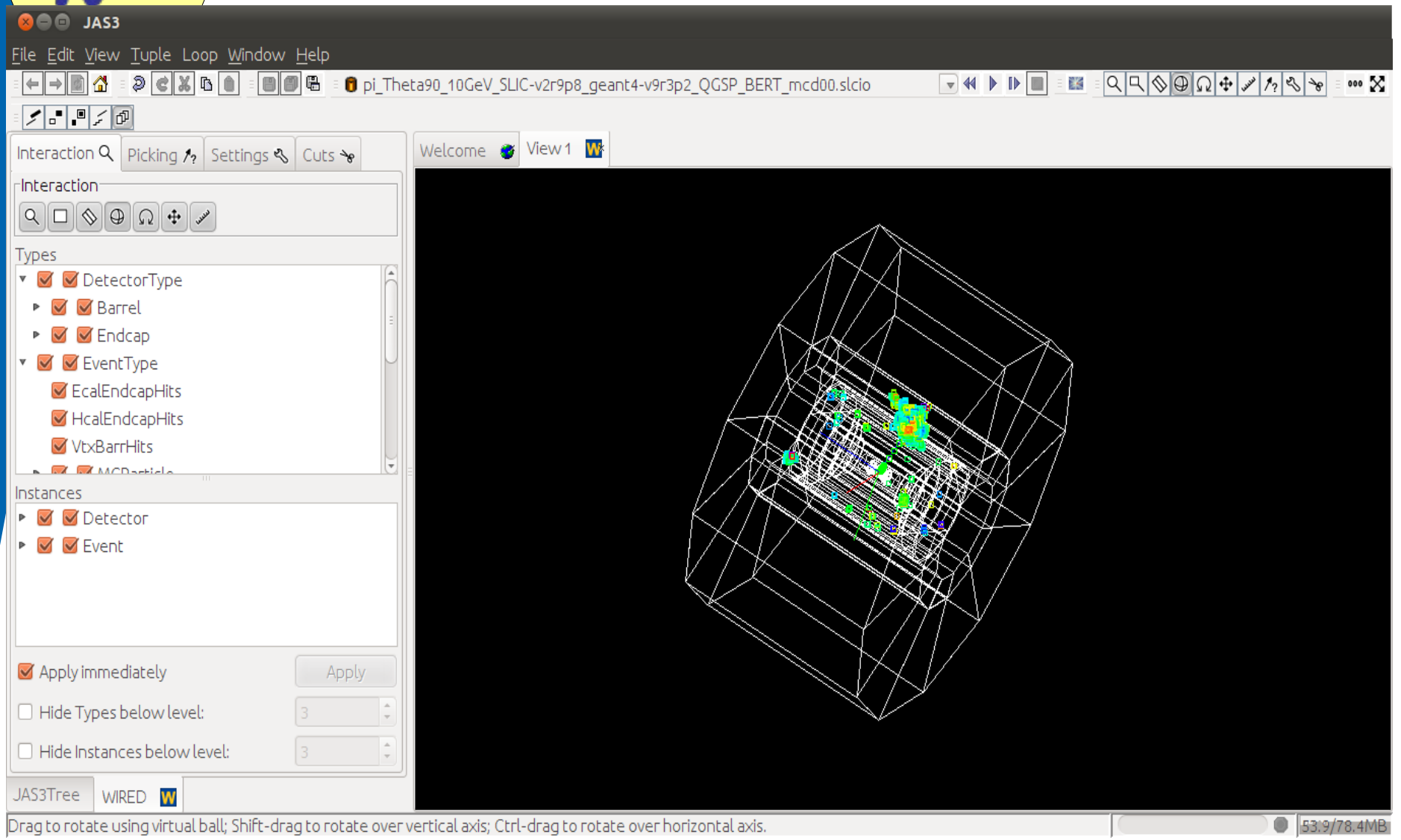
Muon Collider Detector in lcsim



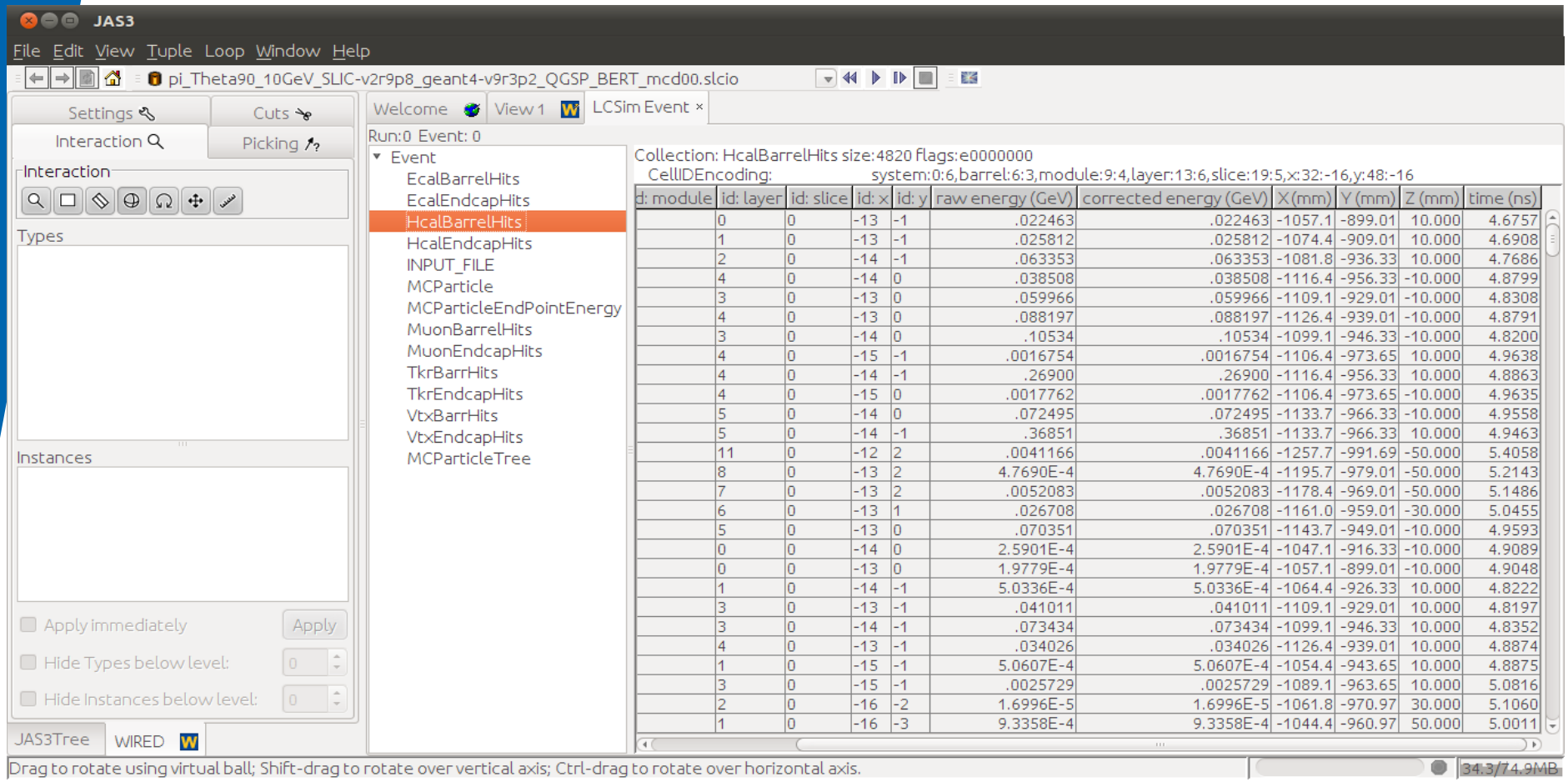


Norman Graf

Wired Event display



LCIO-Data Browser



Requirements to decide on what framework to use

That;s just my personal view. Before deciding on a framework One should investigate how it meets our requirements and will it will help you to do the studies Required.

Framework Requirements (1)

muon collider Detcor studies require frameworks where e. g. Geometry and Segmentation can get changed very quickly.
The system should be lightweight.
Be aware frameworks are to be used by physicists.

(don't need a complete experimental framework e.g. for studies you don't need interfaces to calibration databases etc.)

geometry:
- simple to define and change geometries, materials, detector segmentation in space and time. what is measured etc.
  (probably an xml-based description lcdd, gdml ...
  (human readable, usable for event display, simulation and reconstruction)
  should be complete (support e.g. all geometry types and material properties that geant 4 supports.

Creating a new detector should be easy and well documented

Framework Requirements (2)

language: mixed languages? currently the tools of the trade
geant 4, root are all written in c++, complete c++ frameworks exist
e.g. the framework used by mu2e (art), ilcroot... should be evaluated

Should be a complete framework?
event generation -> simulation -> reconstruction --> analysis can be run in a single job where modules communicate via an memory resistent event data model (EDM) that new objects can be added to.
(don't depend on files to communicate between the different stages

The logic flow should allow to simulate realistic trigger decisions, event filtering etc.

support batch and interactive interaction with framework (like e.g. geant 4)

persistancy: use root io, reserve the possibilty to easily define new classes to be made pesistent. Having a static format doesn't make sense when things are in flux. The expert should decide what information needs to be kept.

Framework requirements (3)

IDE/software development tools: integrate with an IDE like netbeans, eclipse .... this tools really help tonavigate the code.
Module templates could be provided to get people started.

support: When deciding on a framework who is supporting it?
Who is using it? Do you have free access to developers?
Has there been recent activity developping the framework/tools.
What are the plans for development (plan to support e.g.
multithreading, new CPU platforms

muon collider: - need a system to add bgr events (pile up) this is e.g. done at the hit level for LHC experiments (CMS) (order of a few hundred min bias events)
Probably muon Collider needs a different approach. Can the Bgr. be parameterized

- calorimeter hit classes need to provide fine time windows to allow for background rejection.

provenance: keep all information that make it possible to reproduce a result with the data.

Discussion

What kind of computing infrastructure/environment do you need to do the required studies?