

Simulation Status Report and Plans

A. Morsch

ALICE Offline Week

19/11/2014

Overview

- Geant4 Validation and Multi Threading Tests
- Geometry
 - Parallel Navigation
 - Refinements
 - Detector Upgrade

Geant4 Transition

- Factorized Strategy

Physics Validation with AliRoot

- Geant4 interfaced via Virtual Monte Carlo (VMC) + TGeo

Multi Threading Performance tests with thread safe VMC application

- Based on standard geant4_vmc example
- Use reduced set of AliRoot features allowing for realistic performance tests:

- full AliRoot geometry imported from file

- realistic magnetic field

- realistic event generator (based in AliGenHIJINGpara)

- Basic TPC response (hit) generation and I/O

- Easy to extend to more detectors

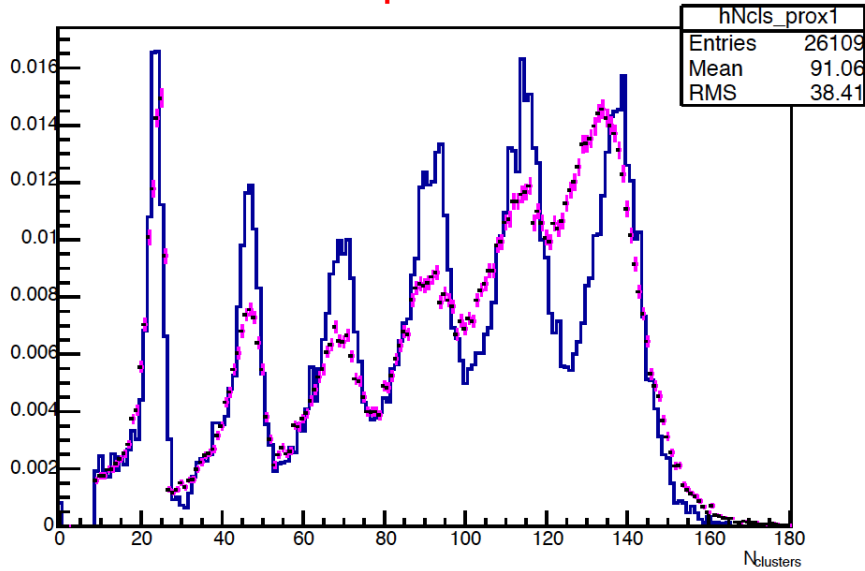
Status

- MT tests and physics validation with Geant4 v10
- Physics Validation
 - Main problem so far TRD response
 - Too high mean energy loss leads to unrealistic cluster per tracklet distribution
 - testing now simple rescaling
- Multi Threading
 - Testing suite implemented and running
 - performance tests ongoing

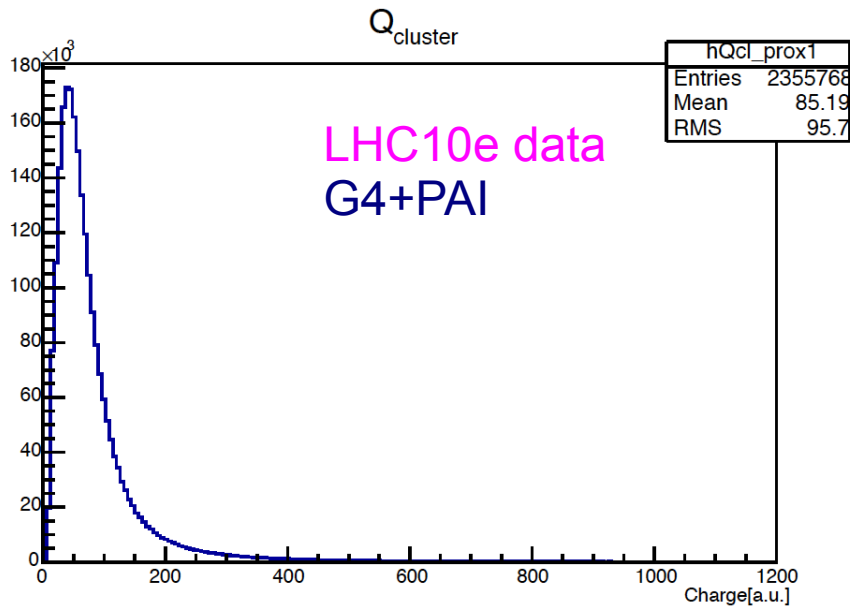
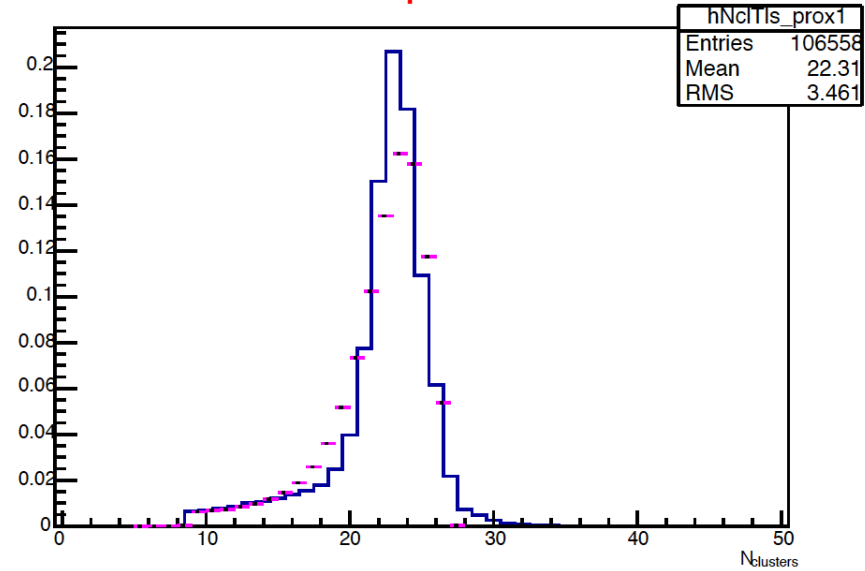
Geant4/AliRoot Validation

TRD Response

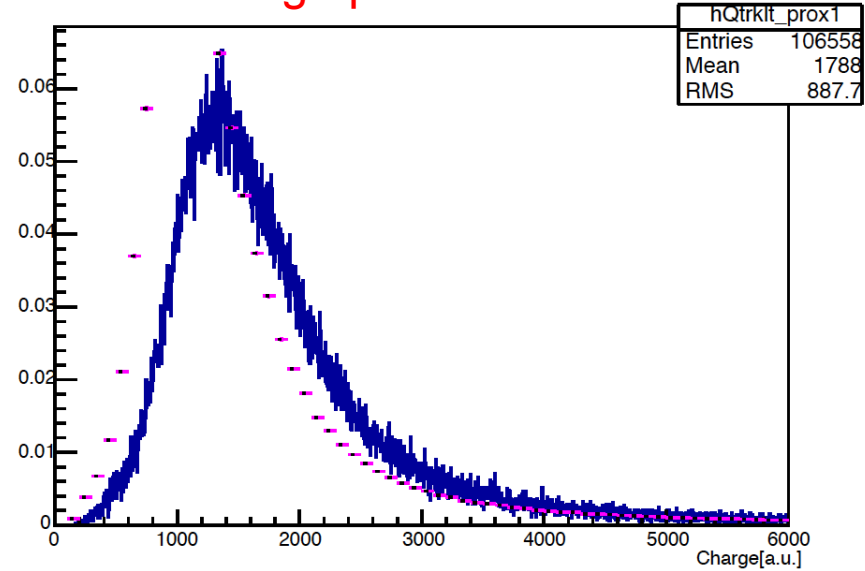
Clusters per track



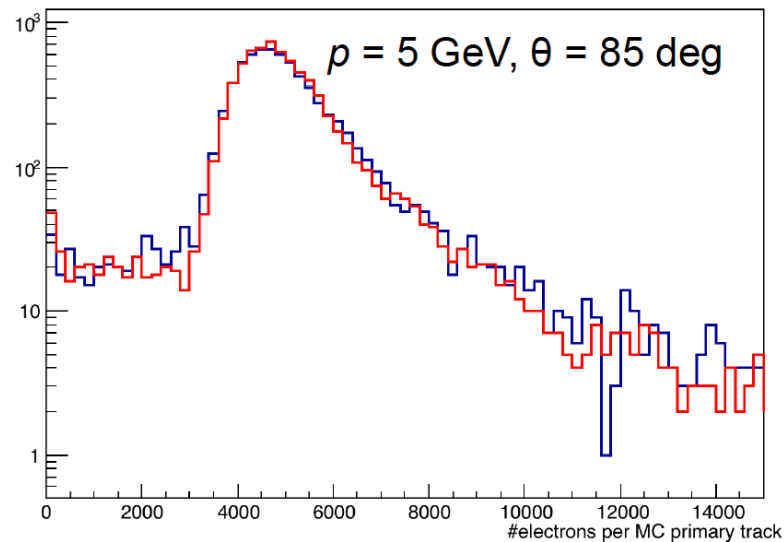
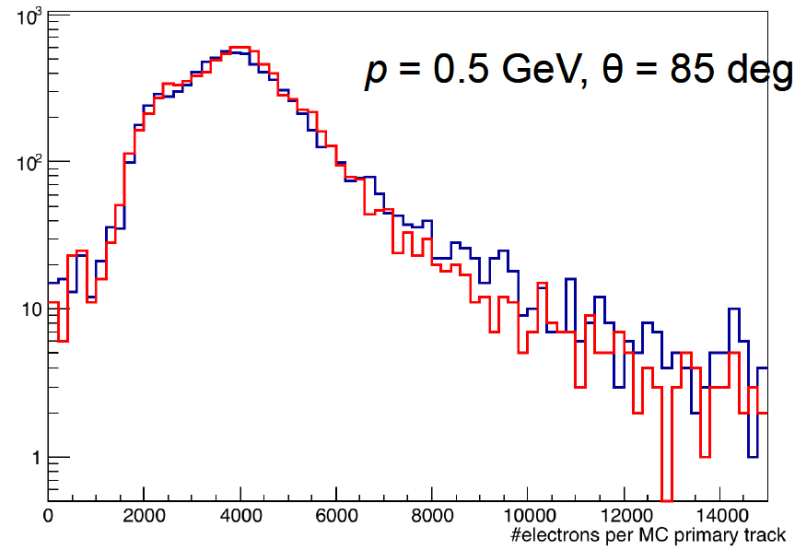
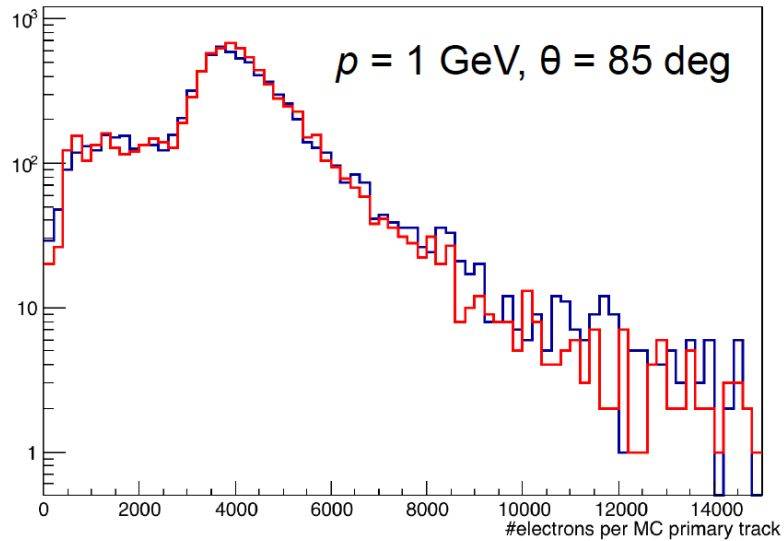
Clusters per tracklet



Charge per tracklet



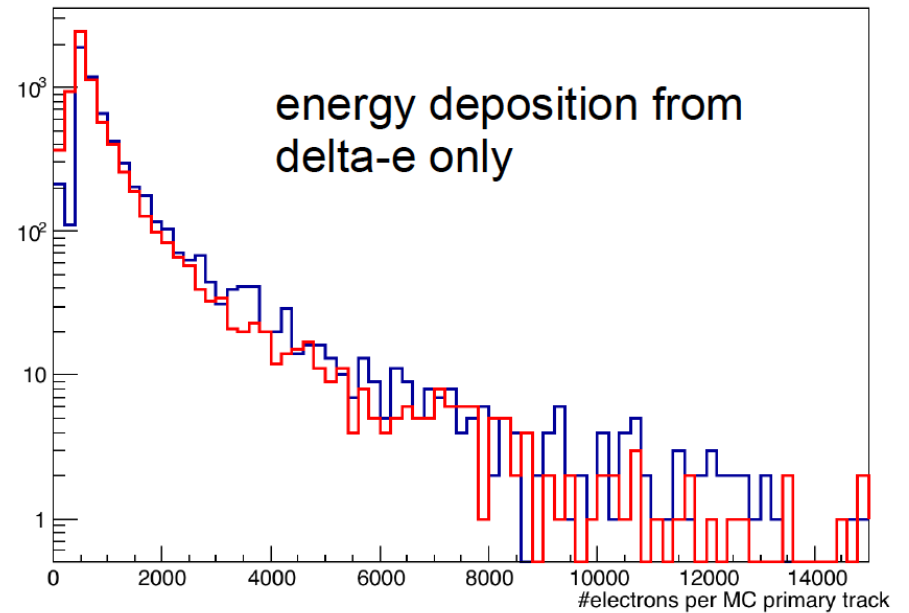
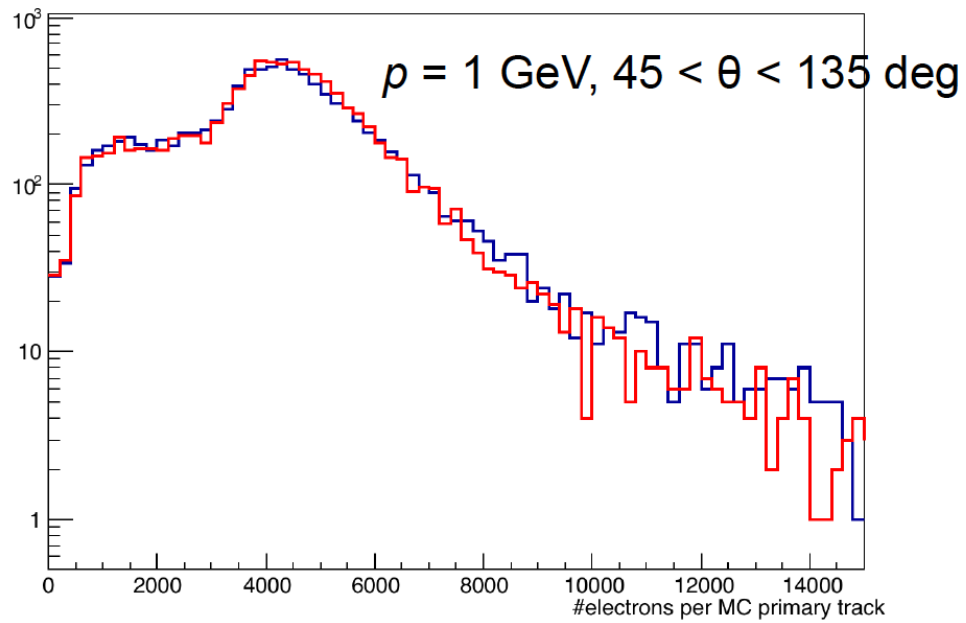
MC Tracks



— Geant3
— Geant4+PAI / 1.12

Geant4 dE/dx scaled down

MC Tracks



— Geant3
— Geant4+PAI / 1.12

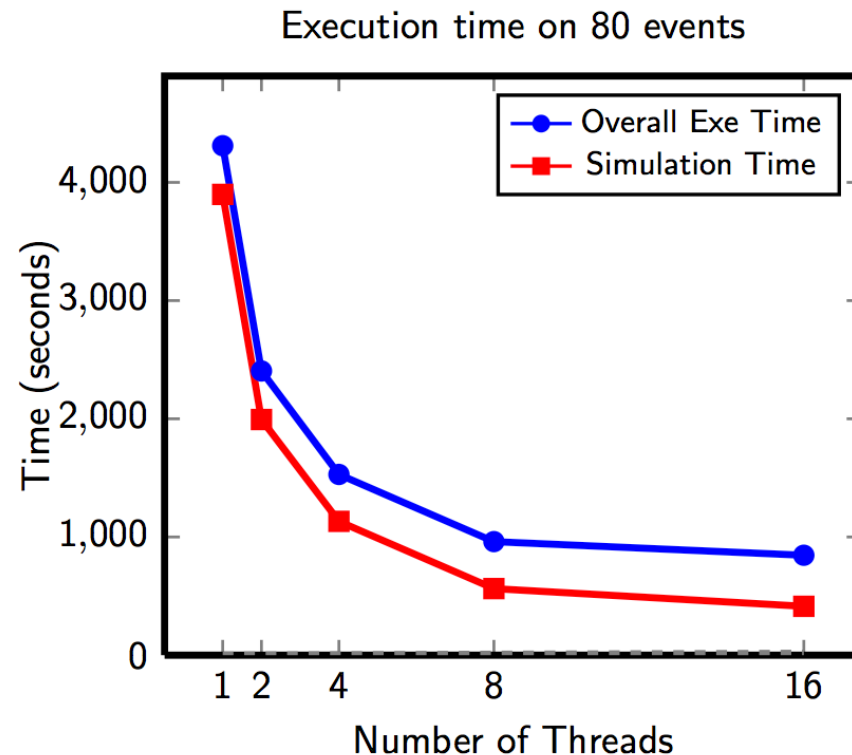
Plans

- Standard QA validation
 - Now running further TRD test
 - with QA we found the most obvious problems
 - need to get all detector experts involved
- Full analysis with Geant4
 - mirror existing Geant3 production
 - use and validate in analysis
 - PWG help needed

Multi-Threading Performance

Simulation Time vs Number of Threads

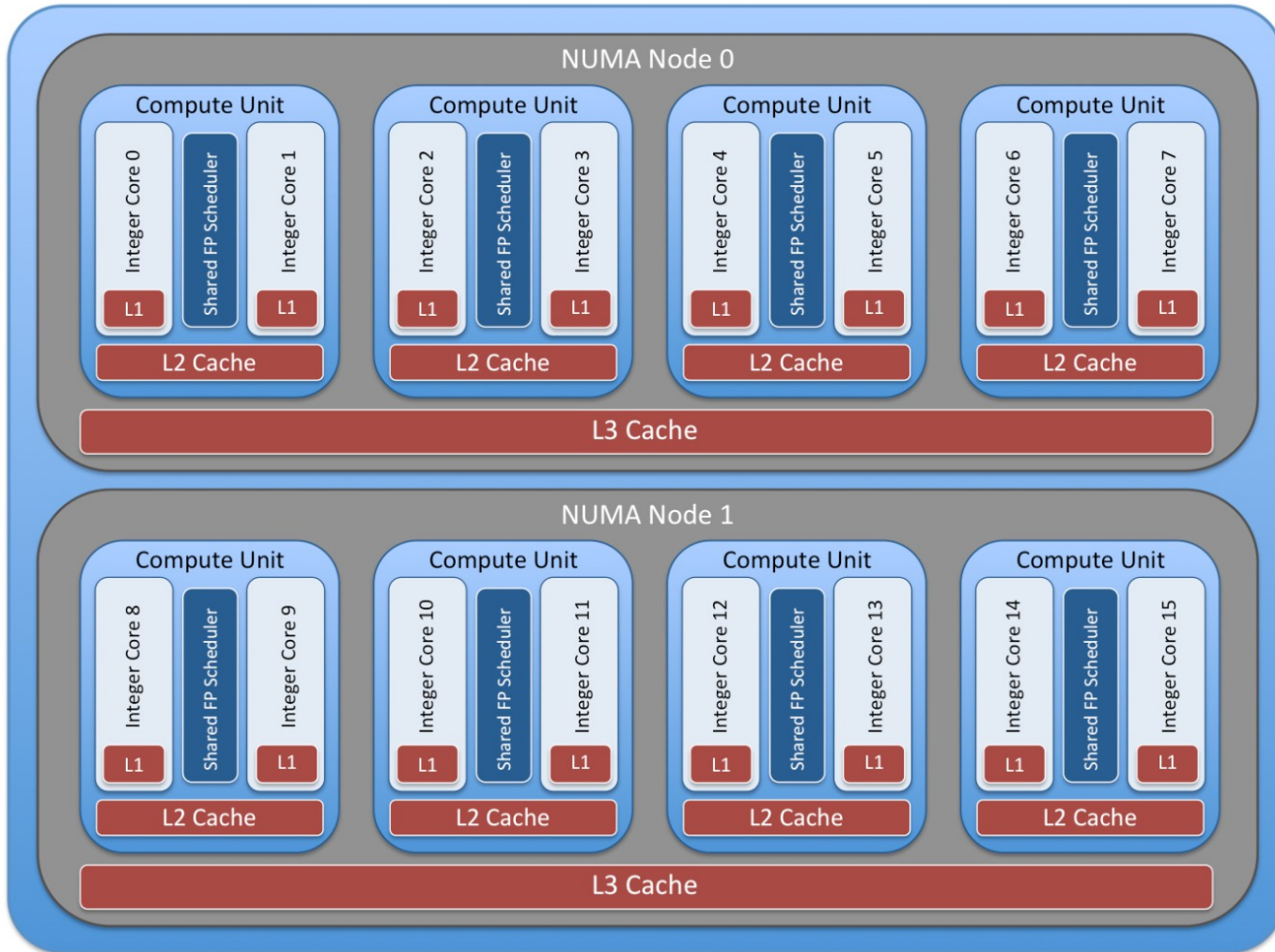
Work in progress



Titan
16-core 2.2 GHz AMD
Opteron 6274
(Interlagos)
32 GB of RAM
Cray Linux

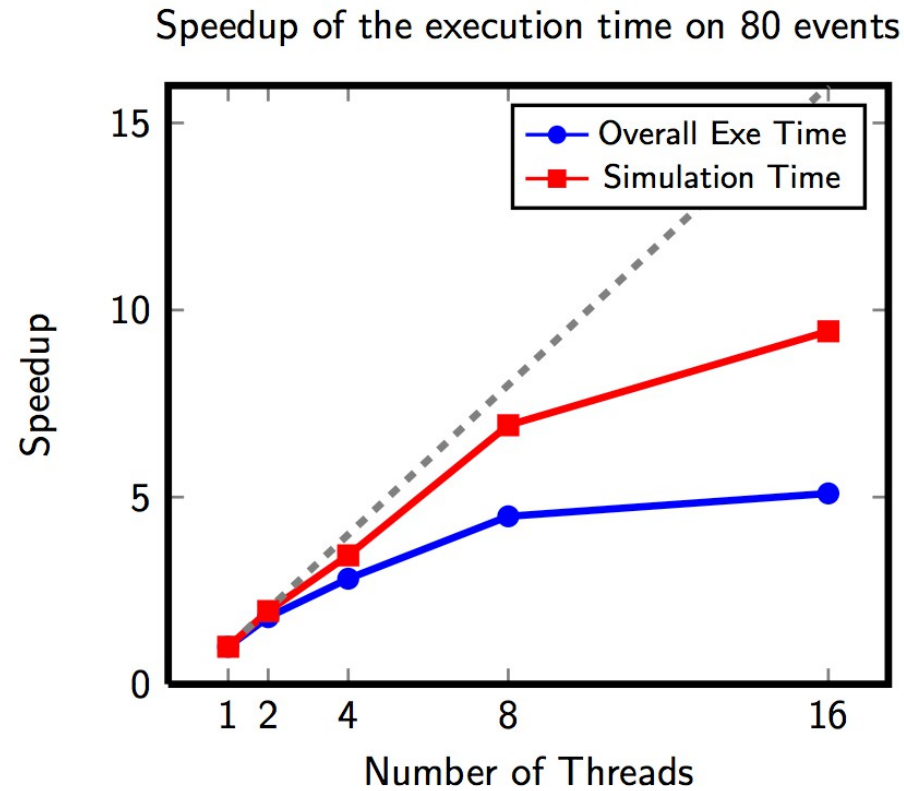
- Overall execution time – The batch time measured by Linux wall clock timer
- Simulation time – The time reported by VMC simulation

AMD Opteron™ 6274 (Interlagos) CPU



Speed-up vs Number of Threads

Work in progress



To investigate

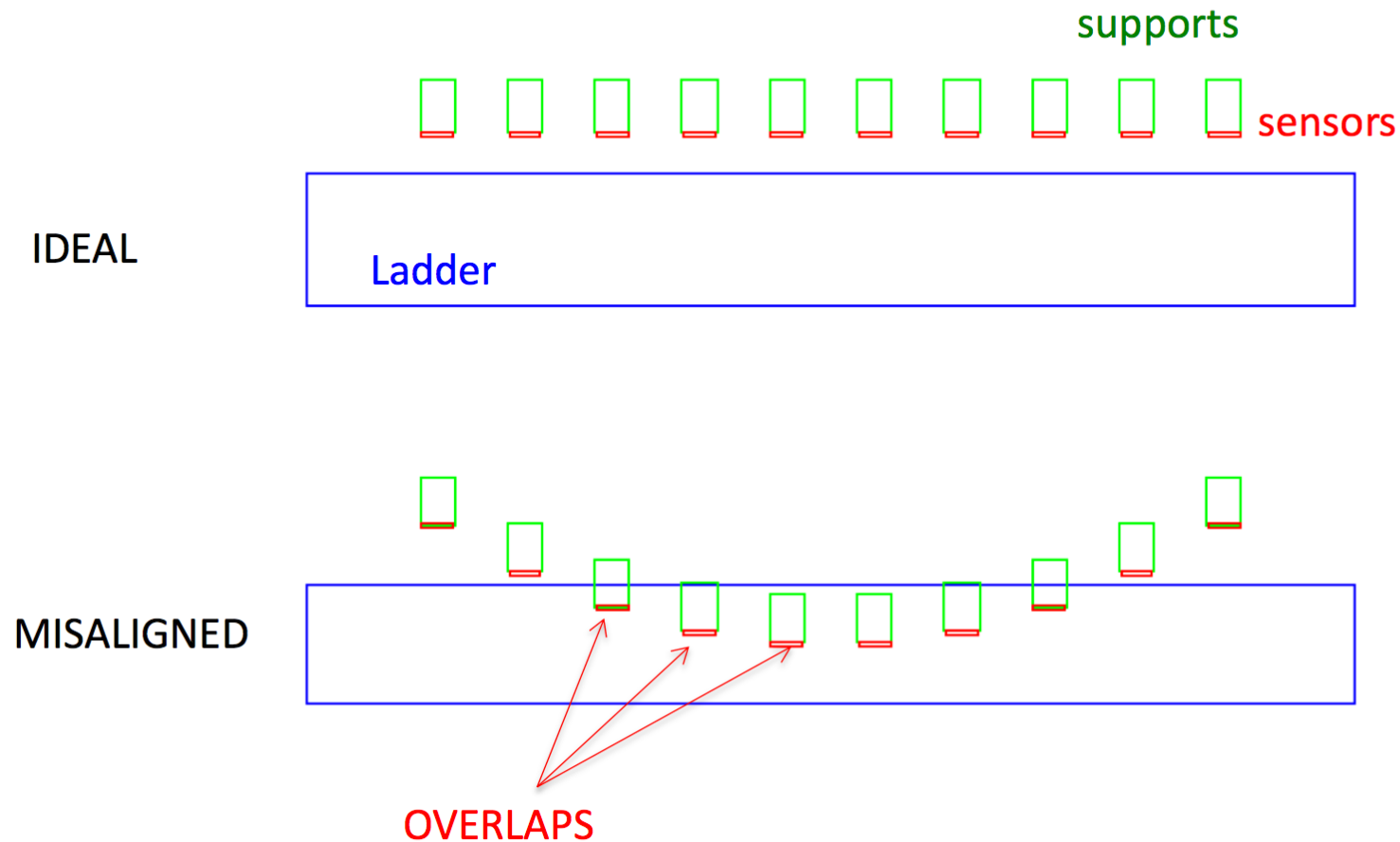
- Panality due to
 - global-dynamic Thread Local Storage (TLS) model
 - the effect of shared vs static libraries
 - locking of primary event generation (gRandom)

Geometry

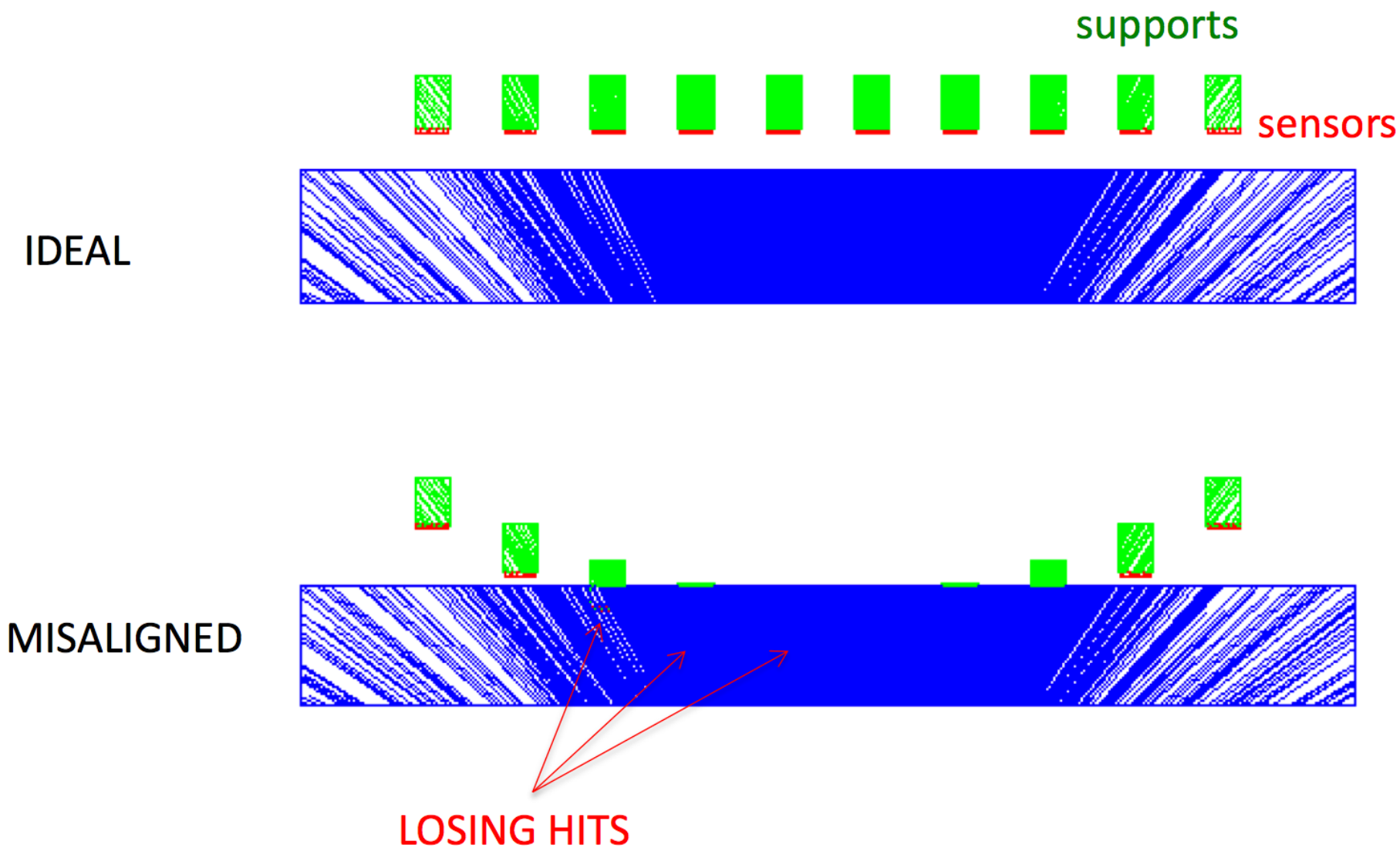
- Prioritized Overlaps
- Refinement of existing geometry
- Coordination of Run III geometry

Prioritization of Overlaps with TGeo

The problem



Overlaps and tracking



Solution:

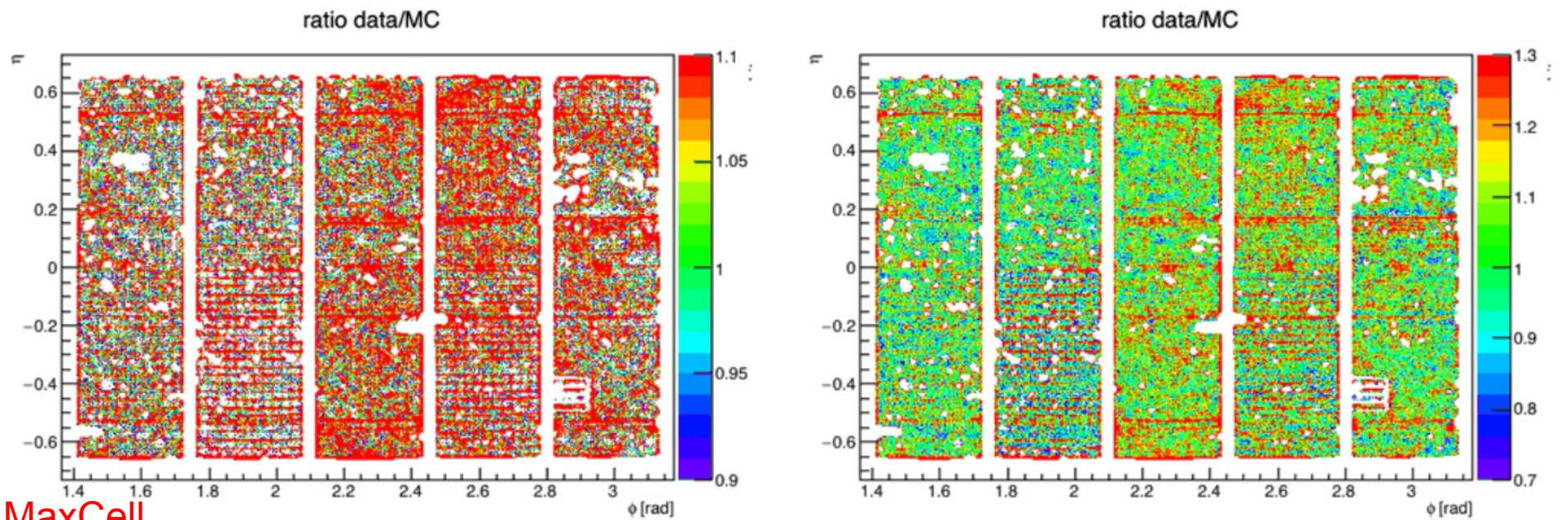
Parallel Geometry Navigation

- Declare some objects as VIO (very important)
- These objects are always considered when computing the location of a particle distance the the next boundary
- Navigation in the standard geometry has to be validated against the parallel geometry
- Create parallel world (will be done from AliSimulation)
`TGeoParallelWorld *pw = gGeoManager->CreateParallelWorld("priority_sensors");`
- Add alignable (physical) nodes to the PW (detector code)
`TGeoPhysicalNode *node = gGeoManager->MakePhysicalNode("/TOP_1/chip_1");`
`pw->AddNode(node);`
- Declare the volumes which can possibly overlap the priority ones (detector code)
`pw->AddOverlap("ladder")`
 - This will **HIGHLY** increase performance

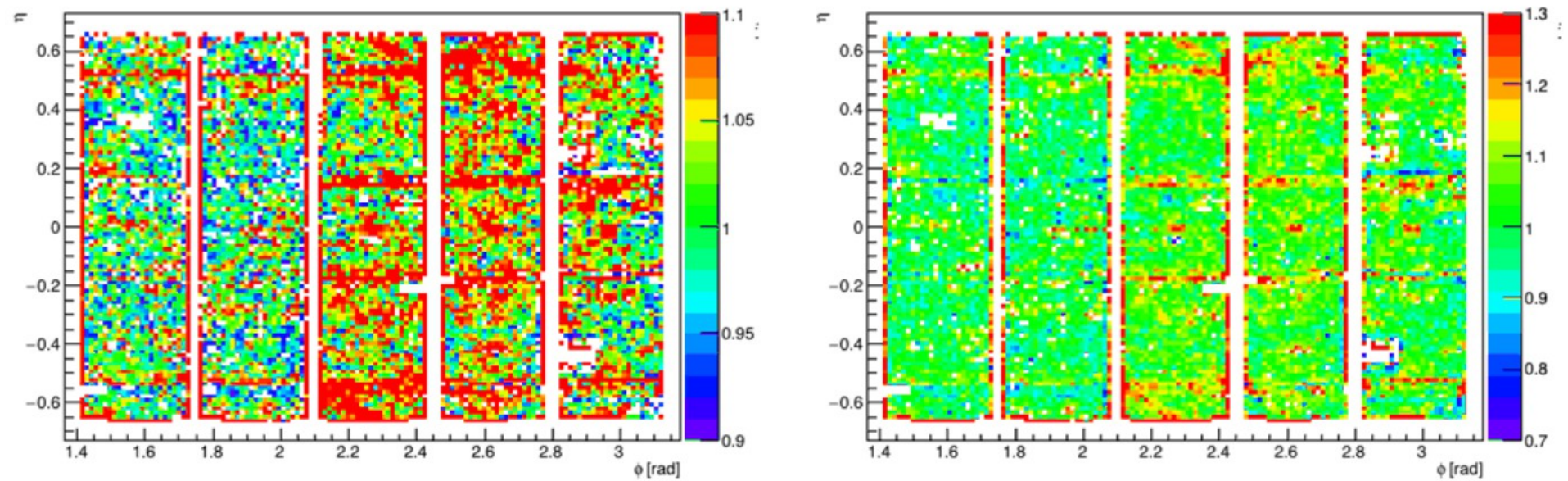
Ongoing geometry updates

EMCAL Cluster Map

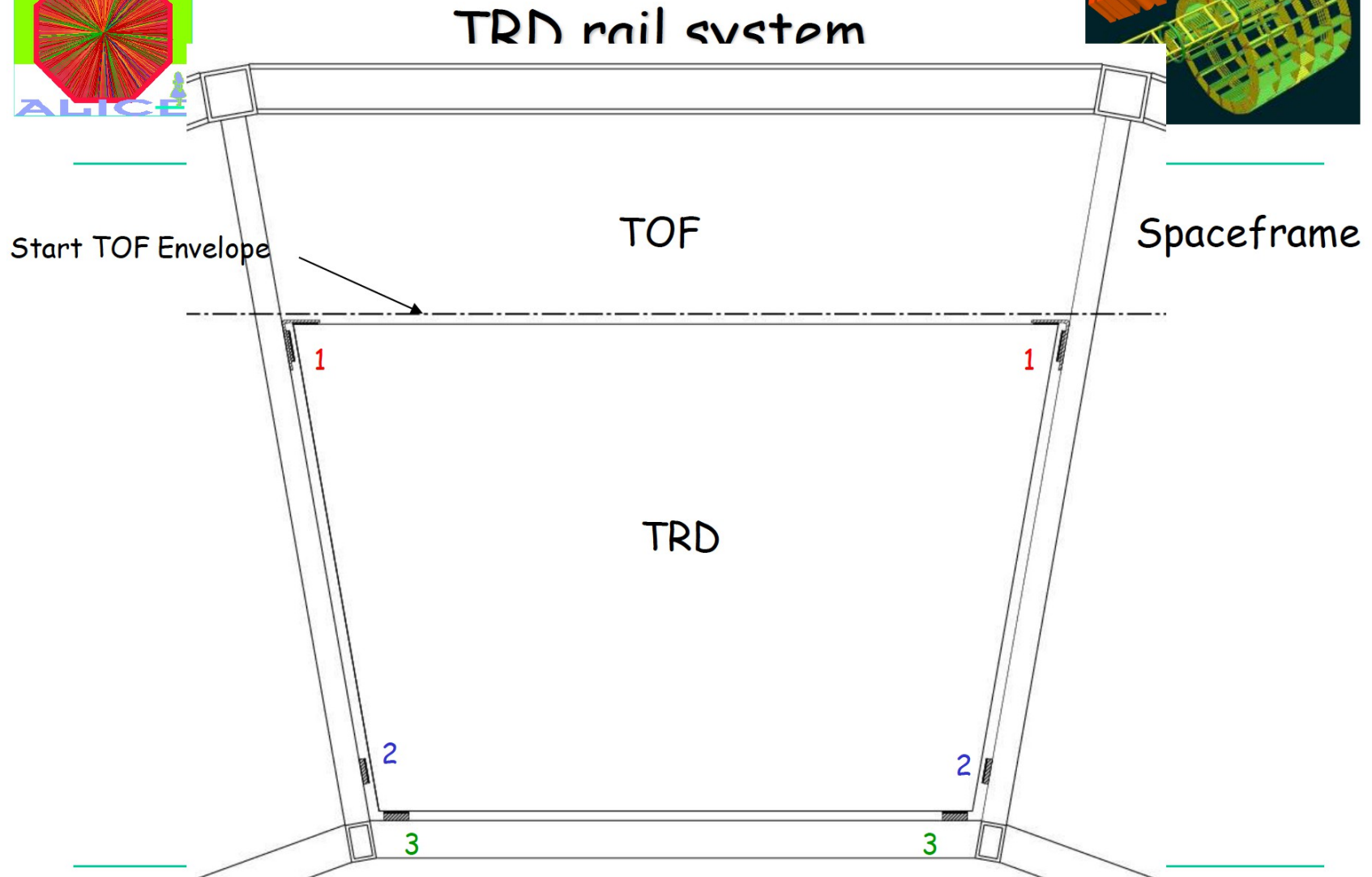
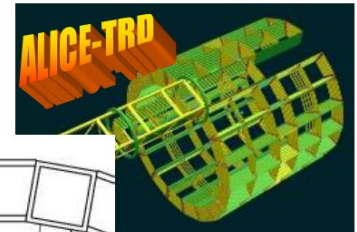
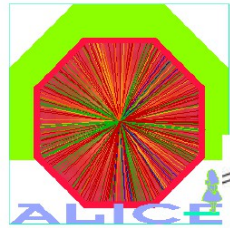
Clusters

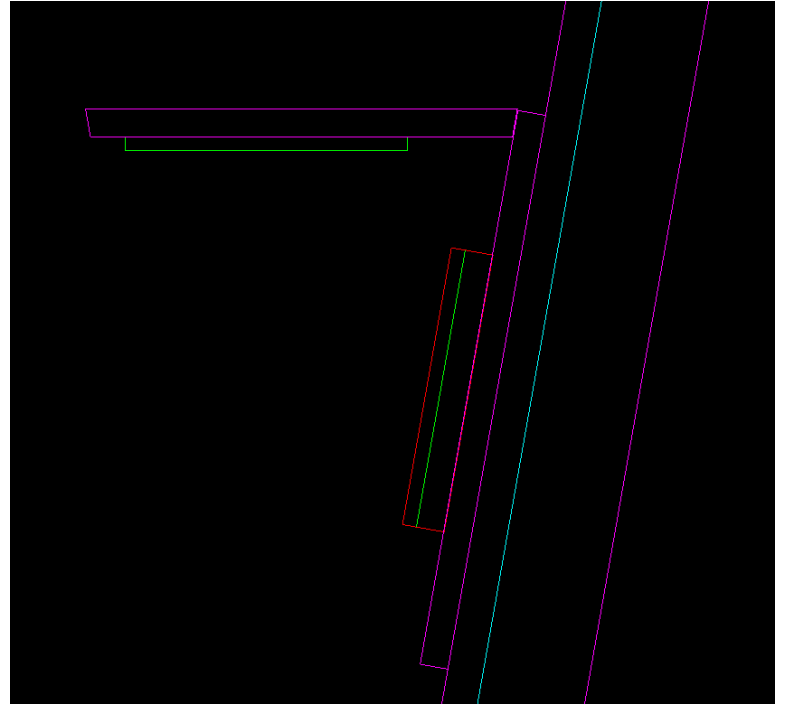
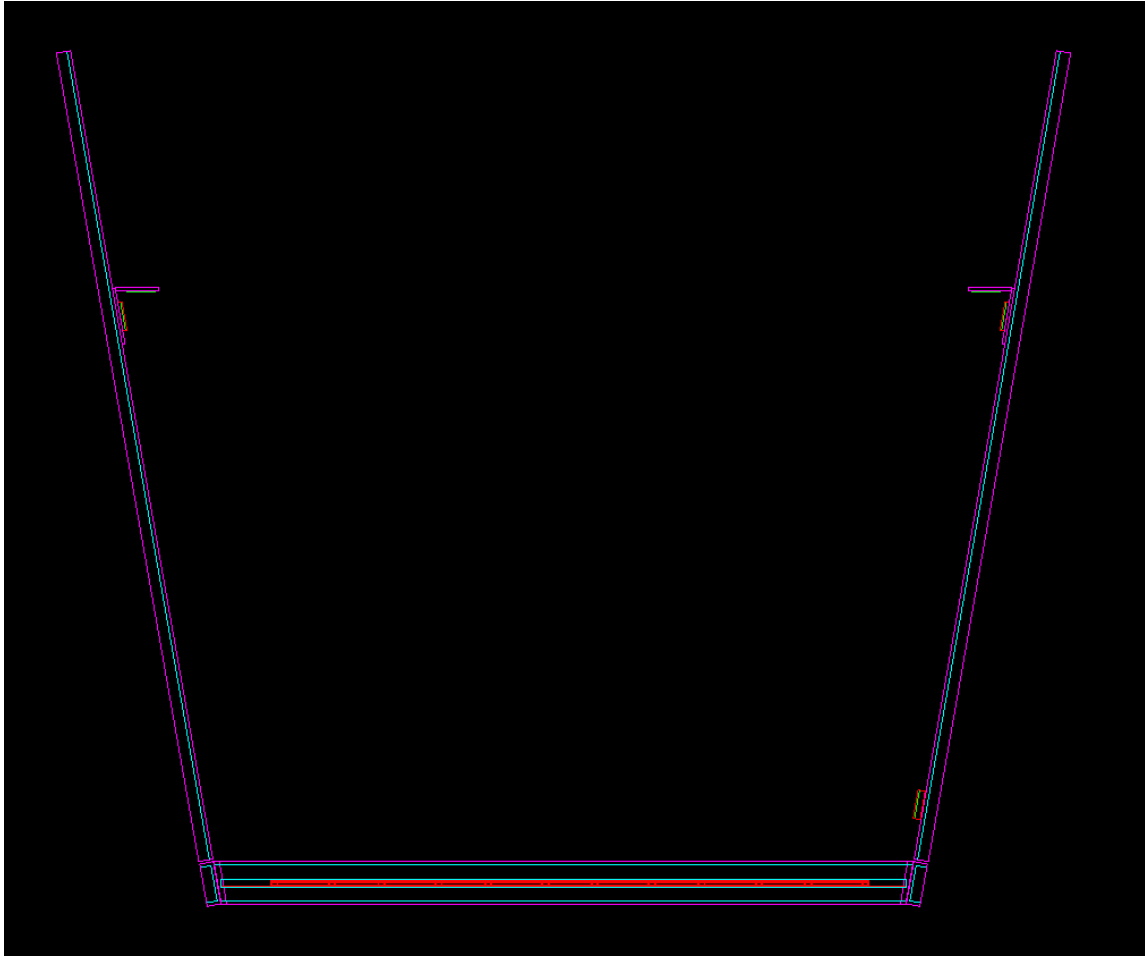


MaxCell



Geometry Refinements





Geometry for Detector Upgrades

- Improved coordination between TC, Upgrade, CWG8 and PWG-PP needed
 - implementation
 - versioning
 - documentation and common drawing data base
 - do not repeat the errors of the past