Detector simulation and Geant4

A. Dell'Acqua, CERN EP-ATC
dellacqu@mail.cern.ch

WG meeting on Dec. 1st

- Pretty well attended
- Quite a few interesting talk
- Most comforting fact: we reached the critical mass
  - People working on all parts of the Atlas detector
  - Some people have started from scratch and have reached a honorable level of productivity
  - we can start envisaging some productive work
- Lots of problems are emerging (foreseen)
- Can these problems be solved smoothly (next challenge)?

Current activities

- People concentrating on geometries, yet
- Need to start evaluating physics asap (and with the broadest spectrum)
  - First examples show problems which, although not surprising, are turning out to be quite nasty
- Training phase almost complete
  - ~100 people went through the course
  - winding down
- Chaos frozen, waiting for an architecture to fit into
  - just providing facilities, when a case arises

Highlights

- We have seen a complete geometry of the Accordion in G4, for the first time
  - but we spotted several problems, as well
- The Detector Description in XML is proceeding well and a tool is available for building a G4 description out of it
- The muon system has been described at least twice and a complete chain AMDB-G4-MuonBox-Persint has been realized

Geometry

- Placements work properly
  - The geometry of the accordion can be realized in this way, at least for test beam simulations
  - Extremely heavy in terms of memory and initialization time
- Parameterized volumes are OK, as far as detector geometry is concerned
  - but disastrous in terms of tracking performance…
  - 1-dimensional voxelization (wrt 2- and 3-dimensional voxelization for Placements)
  - We can’t live without (but we can’t use them either!)

Geometry (2)

- Boolean solids
  - They are great! People fall in love with them…
  - …but they are an endless source of bugs
  - …and doing so, they are slow, as well
  - …and until now we got no graphics for them
- We have not tried BREPs and other geometrical entities provided by G4, yet
- The possibility of defining our own solids and get them to work in the simulation as a snap is extremely appealing
  - Let’s create a G4ZigZag then...

Geometry (3)

- The answer (for complex geometries) is obviously in combining different techniques.
- Need to try them, in order to gather expertise
  - Looking for creative “architects”
- …but at the same time we need parameterized volumes and boolean solids to be made as performing as possible
  - High priority to implementing the 2-dimensional voxeling for parameterized volumes
  - Low priority to G4VolumeAssembly (which is just a way of masking placements in the blue…)

The G4 builder

- Very nice exercise by Stan
  - Allows to go from the XML description of a detector to its geometry description in G4
- Useful for a rapid implementation of a geometry to be tested/compared against another one
- For the moment it makes heavy use of boolean solids and it does not implement parameterizations
- My feeling is that the “ultimate” geometry will still be written by hand, with dirty tricks to gain time
  - still, we must give the possibility of utilizing the builder

Physics

- It is clear (see M. Leltchouk's and D. Barberis' talks) that we have a problem there
- It is not always clear which processes to use, when alternative implementations exist
- It is not clear what is the effect of cutting in stopping range (wrt kinetic energy as in G3)
  - it is encouraging to see that the mean value of the energy deposited in the LAr does not seem to depend on the cut value, though
- There seem to be problems with several processes
  - …and we haven't started with hadronics, yet

Physics (2)

- We have to slow down and understand and verify what we are doing
- Although shooting particles into the Accordion might be fancy, we probably have to step back to simpler geometries (material slabs, simple plate calorimeters) in order to simplify the problem
- Re-use knowledge and expertise accumulated by G4 developers to sort things out
  - Mini-workshop on Dec. 9th (2pm) to discuss about processes and cuts, at least for EM physics

Physics (3)

- We ought to set up a working group composed of “volunteers” who can spend their precious time exploring the G4 dungeons (cfr. M. Nessi, Aachen, 1990)
  - regular meetings
  - clear program of work
  - evaluation of the physics available in G4
  - work with G4 to correct bugs
  - report in N months from now
- …and concentrate on test beams!

Miscellanea

- Although there exists some fancy graphic stuff, aimed at improving the look of your presentation, basic utilities to facilitate user's life when implementing his/her simulation program are far below G3!
  - I miss DTREE, DCUT, an user interface whose commands can be abbreviated…
- The user must now implement parts of the simulation which were coming for free with G3 (#1 question I'm asked in the courses: "Is there a list of pre-defined elements and material in Geant4?")

Installation and distribution

- Until now, the patch policy (of the lack of it) has been the biggest problem I saw
  - We have to maintain our own version
  - We have to check out new versions as they become available
  - We have to build the libraries
- I still can't see why should we act as librarians for Geant4!
- The G4 CVS repository should be made public (read only)
- I can't see any improvement with the revised policy

Where should we go from here?

- Evaluation of the physics currently in Geant4 is the keystone for the success of this simulation
  - we need people willing to run a simulation program and understand the results
- Test beams are the next challenge
  - no need to worry about geometrical implications
  - a good agreement between data and simulation must be achieved before we can even think of collecting things into an Atlas simulation program
- Collaboration with the Geant4 group is fundamental and needed