Subject: Re: First results of TPC code profiling Posted by StefanoSpataro on Wed, 09 Feb 2011 13:01:07 GMT View Forum Message <> Reply to Message

Hi,

I am still trying to understand what is going wrong with the TPC digitization. For this job I am using a i7 with 8Gb of memory, and I am visually monitoring the memory using the System Monitor given by the Ubuntu distribution.

I have run the run\_digi\_tpccombi.C, with all the detector digitization tasks, simulating 10k protons between 0.2 and 5 GeV/c.

I have seen that the used memory is in general quite stable. When I have started to take a look, it was stable around 750 Mb, without any variation. After around 1000 events, I have seen that the digitization was stopping on a well defined event, in particular at this stage:

XXXXX electrons arriving at readout Aggregating drifted electrons into avalanches finished. XXXXX Avalanches created 0 aggregations done. XXXXX Signals created PndTpcElectronicsTask::Exec Building up padmap ...finished. XXX pads hit

In the middle of TPC digitization, still filling the line with dots ".". This procedure has taken few minutes, and at the end the memory has raised up to 1.3Gb... just this task! The memory has remained stable at 1.3Gb for a while. After a bit, a "long" tpc event has raised it up to 1.4Gb, then stable again for many events, and after another "heavy" event up to 1.8Gb, then to 2.1Gb, then finally 2.7Gb: increasing only during specific events and in the middle of tpc tasks.

The following is the 1.8Gb->2.1Gb case (the only one I have copied):

PndTpcClusterFinderTask::Exec number of digis: 2878 371 cluster created containing 2878 digis from 2878 Hit array contains 26 hits PndEmcMakeCluster, event: 7078

From this check, I can argue the following points:

The memory consumption is stable for almost all the events, for all the tasks For some well defined events the memory jumps up, in the middle of TPC digitization The memory does not go down after the heavy events, it is not cleaned and it remains blocked

I think it should be important to understand what is making the memory growing inside the "guilty" TPC task, and how to clean the memory to go back to the original size. If not, it is easy to understand that if one has a "limited" memory and not 8Gb like my machine, then there is a crash. In this case, if I would use a 2Gb limited machine, i.e. GSI Ixi or my VM -> crash.

If I remember some time ago there was some discussion because in that TPC task many links were created, then maybe also them could be the reason... but I think still that we should find some way to reduce the memory somehow during processing.

I leave the word to the experts...