Simulation Studies Update

20 October-26 October

Context

- The spectrometer sensitivity and beam steering studies aim to ascertain the optimal precision in spectrometer coil positioning.
- Number of configurations to explore in spectrometer studies = nStudies*nToroids*nOffsetType*nOffsetValues *nGenerator

  = 2x2x6x11x3

  =792
- For 10 million events per configuration, each configuration run consumes 1000 ch (core hours) and approximately **3GB memory per core** (for simple geometry without shielding and detector array). Output file occupies 800 MB on disk per configuration (already optimized by taking out irrelevant variables).

  Total required core years = 792*1000 ch = 90.35 core years

  Total Storage required = 792*800 MB = 633.6 GB

Optimization of Disk Space Usage

- Used G4AnalysisManager to produce output root files instead of remollIO class.
- Adding additional output variables entails to adding one line of code in remollRunAction.cc and one line of code in remollEventAction.cc.

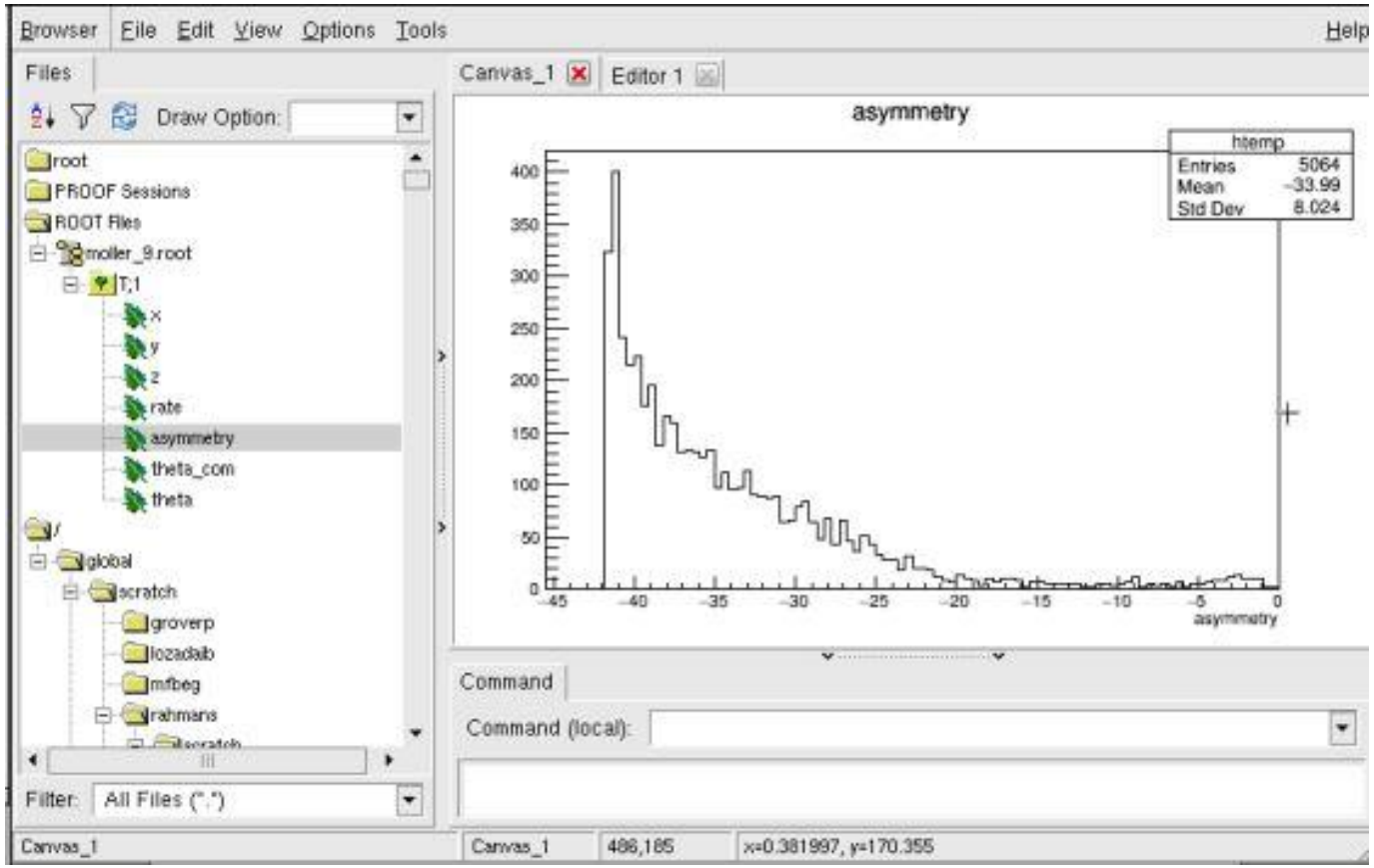
```
void remollRunAction::BeginOfRunAction(const G4Run* run)
{
  // Cast into remollRun
  const remollRun* aRun = static_cast<const remollRun*>(run);

  // Print progress
  G4int interval = 100; // Print this many progress points (i.e. 100 -> every 1%)
  G4int evts_to_process = aRun->GetNumberOfEventToBeProcessed();
  G4RunManager::GetRunManager()->SetPrintProgress((evts_to_process > interval)
                                                  ? evts_to_process/interval
                                                  : 1);

  if (IsMaster())
  {
    G4cout << "### Run " << aRun->GetRunID() << " start." << G4endl;

    fTimer->Start();

    G4AutoLock lock(&remollRunActionMutex);

    /*-----Implementing analysis manager-Sakib-----------------*/
    auto analysisManager = G4AnalysisManager::Instance();
    analysisManager->SetVerboseLevel(1);
    analysisManager->SetFirstHistoId(1);
    analysisManager->SetFirstNtupleId(1);

    analysisManager->CreateNtuple("T", "T");
    analysisManager->CreateNtupleDColumn("x");
    analysisManager->CreateNtupleDColumn("y");
    analysisManager->CreateNtupleDColumn("z");
    analysisManager->CreateNtupleDColumn("rate");
    analysisManager->CreateNtupleDColumn("asymmetry");
    analysisManager->CreateNtupleDColumn("theta_com");
    analysisManager->CreateNtupleDColumn("theta");
    analysisManager->FinishNtuple();
    analysisManager->OpenFile();
    /*-----------------------------------------------------*/


    //remollIO* io = remollIO::GetInstance();
    //io->InitializeTree();

    remollRunData *rundata = remollRun::GetRunData();
    rundata->SetNthrown( aRun->GetNumberOfEventToBeProcessed() );
    rundata->Print();
}
```

Open ▾ | remollEventAction.cc | /w/work/halla/parity/disk2/rahmans/remoll-root/remoll-2.0.0-edep/src | Save

```
70      for (unsigned int hidx = 0; hidx < thiscast->GetSize(); hidx++) {
71    //    io->AddGenericDetectorHit((remollGenericDetectorHit *) thiscast->GetHit(hidx));
72
73   /*-----------------Implementing Analysis Manager-Sakib---------------*/
74        remollGenericDetectorHit_t thishit= (((remollGenericDetectorHit *) thiscast->GetHit
  (hidx))->GetGenericDetectorHitIO());
75
76        if ((thishit.det==28||thishit.det==3001||thishit.det==3002||thishit.det==3003||
  thishit.det==3004||thishit.det==3005||thishit.det==3006||thishit.det==3007)&&
  thishit.trid<=maxtr &&thishit.pid==11){ //hits plane detector, is primary moller, is electron
77
78          analysisManager->FillNtupleDColumn(1,0,thishit.x);
79          analysisManager->FillNtupleDColumn(1,1,thishit.y);
80          analysisManager->FillNtupleDColumn(1,2,thishit.z);
81          analysisManager->FillNtupleDColumn(1,3,event->fRate*s);
82          analysisManager->FillNtupleDColumn(1,4,(event->fAsym)*1e9);
83          if (thishit.trid==1){
84              analysisManager->FillNtupleDColumn(1,5, (event->fThCoM)*180/pi);
85              }else{
86              analysisManager->FillNtupleDColumn(1,5, 180-(event->fThCoM)*180/pi);
87              }
88          analysisManager->FillNtupleDColumn(1,6, event->fPartRealMom[thishit.trid-1].theta
  ());
89
90          analysisManager->AddNtupleRow(1);
91
92        }
```

Analysis manager can be controlled with messenger commands to turn on/off ntuple branches. Did not implement this since not urgent to studies.

Optimization of Disk Space Usage



G4AnalysisManager produces identical output to remollIO class.

Additional HPC Resources

- Compute Canada Systems:
  - Graham (36160 cores and 320 GPU devices, 1127 nodes, Min 125 GB memory per node)
  - Cedar (58,416 cores and 584 GPU devices, 1542 nodes, Min 125 GB memory per node)
- University of Manitoba System:
  - GREX (3712 cores, 316 nodes, 4 GB memory per core)

GREX
Geant 4.10.04.p02
ROOT 6.14.04-gcc5.2
CLHEP 2.4.1.0

GRAHAM and Cedar
Geant 4.10.04.p02
ROOT 6.14.04-gcc5.4
CLHEP 2.4.1.0

Reducing Wait Time in the Queue

- Optimize resource consumption per run.
- GEANT4 Multithreading: Reduce memory footprint while maintaining linearity of event throughput vs number of threads.
- Better to run a MT sim with 5 threads all accessing a shared memory of 3GB on a node than 5 sequential sims each needing 3 GB of memory.

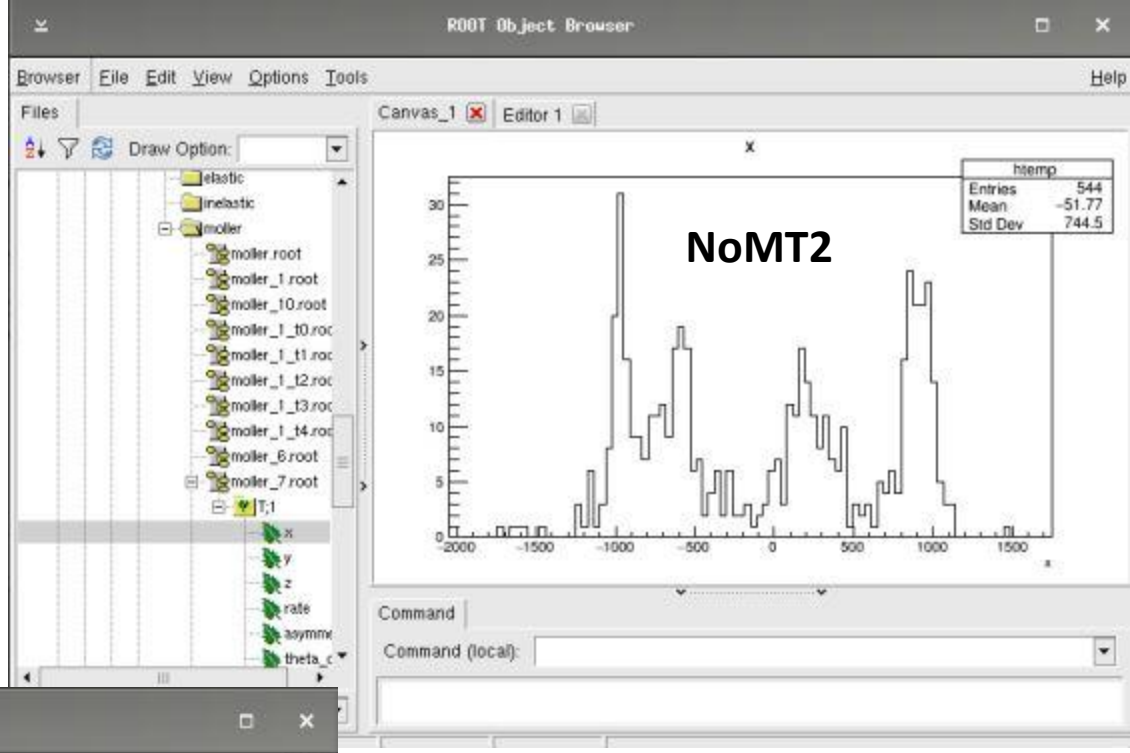
Questions

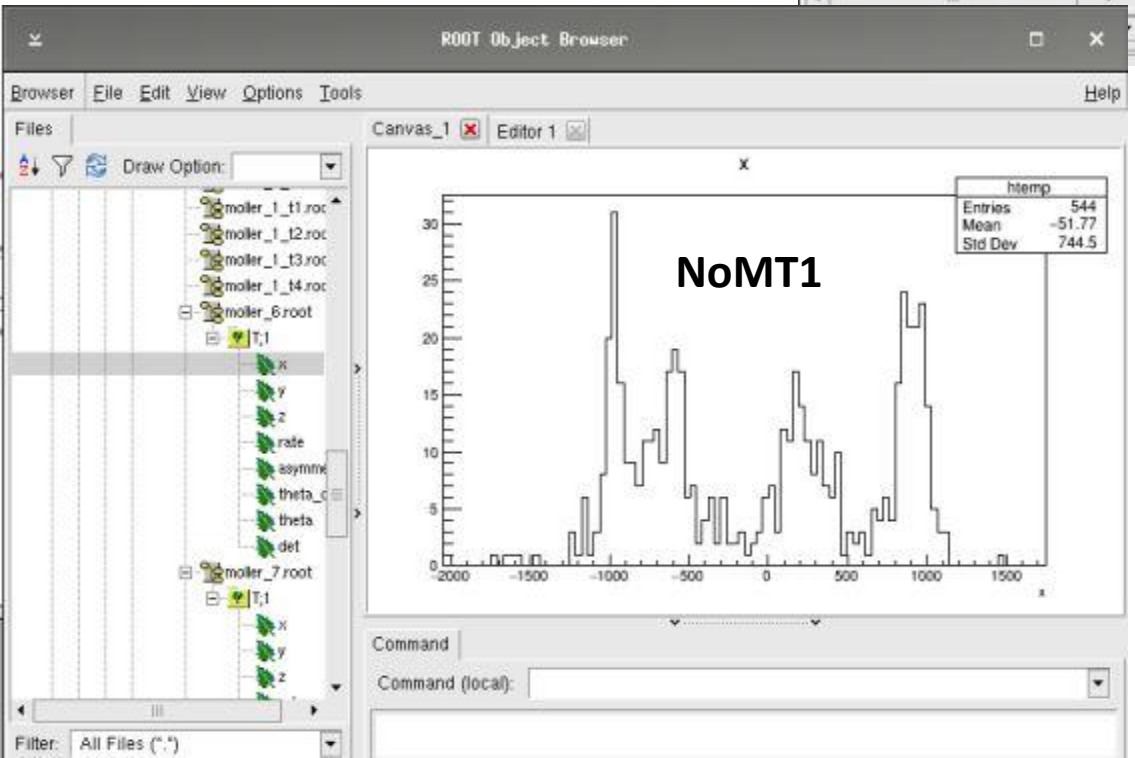
- Is it possible to combine results from NoMT and MT GEANT4 simulations?
- Is it possible to combine results from two separate clusters with identical compute environment?

Observations (Within the same cluster GREX)

1. Two NoMT runs will produce the same result with the same initial seed.
2. Two MT runs will produce the same overall result with the same initial seed. However, the threads within the runs will not produce identical results within or across runs.
3. The NoMT and MT sims use different random number generators. Can their results be compared?

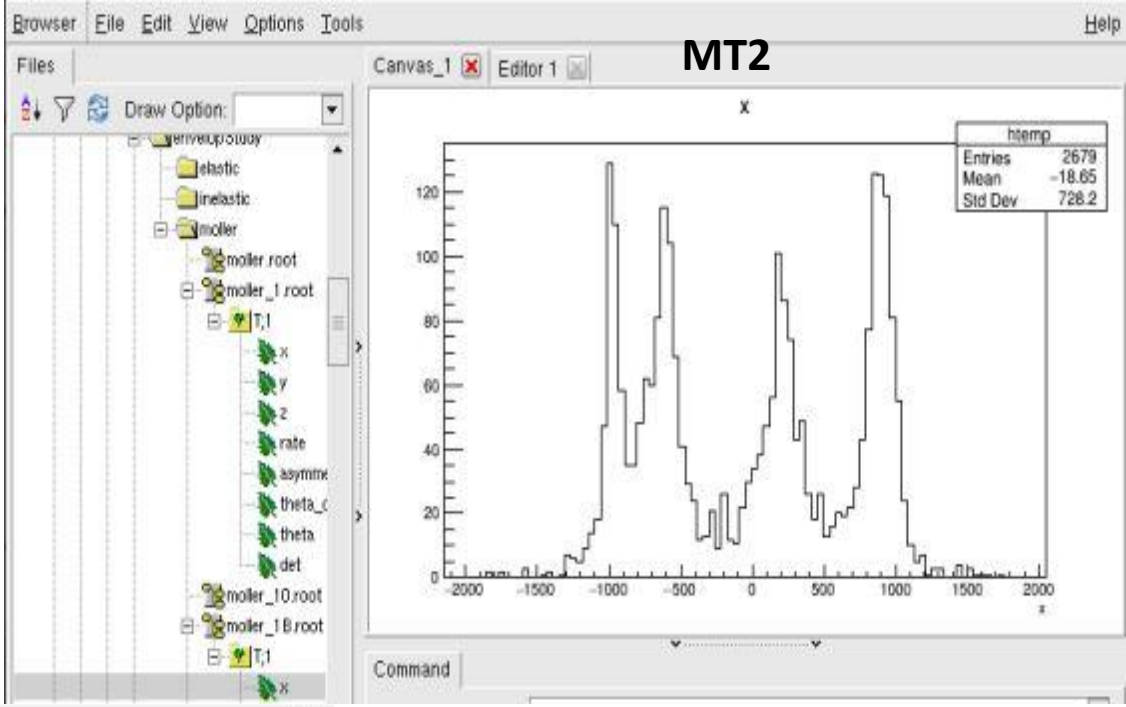
1. Two NoMT Runs Produce Identical Results With the Same Seed

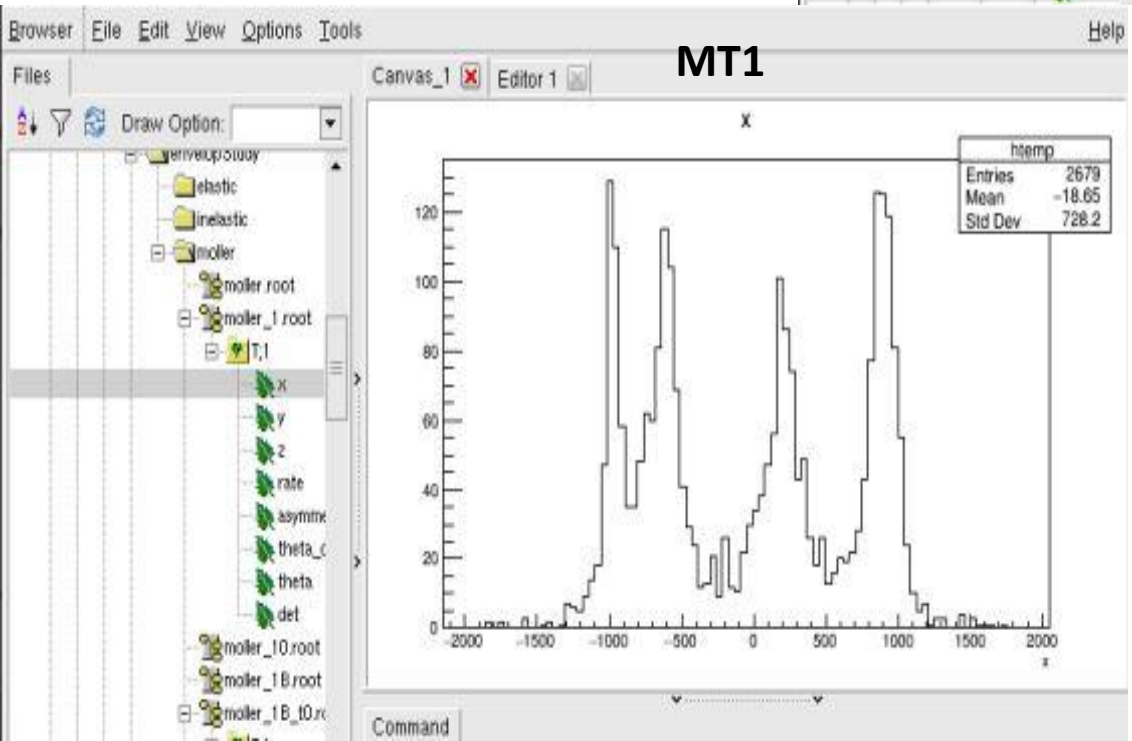




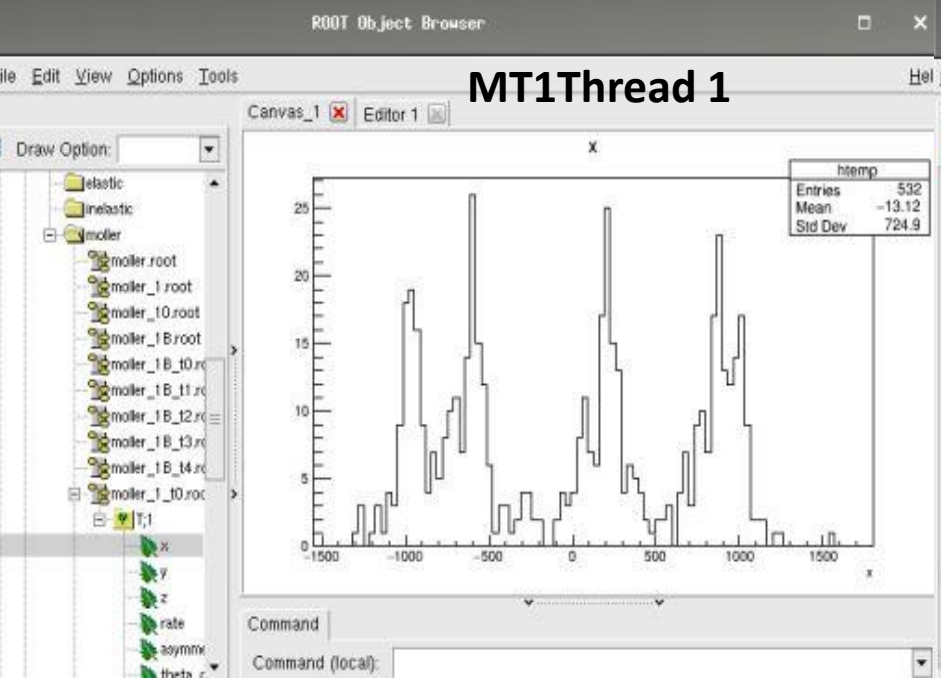
1000 event runs
Seed=123456
Looking at x-position
Of hits on detector plane.

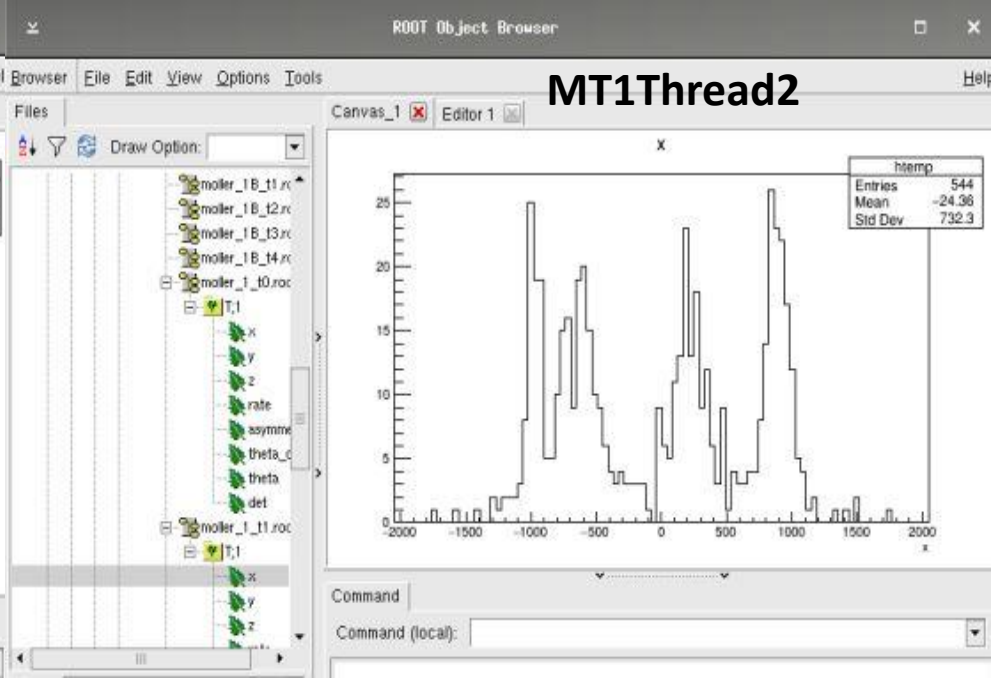
2. Two MT Runs Produce Identical Results With the Same Seed



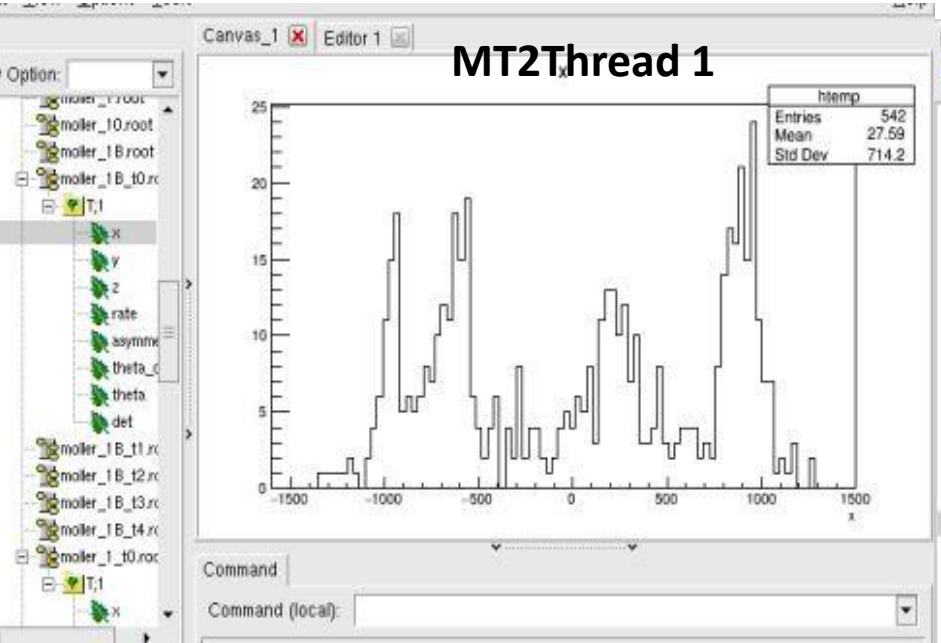


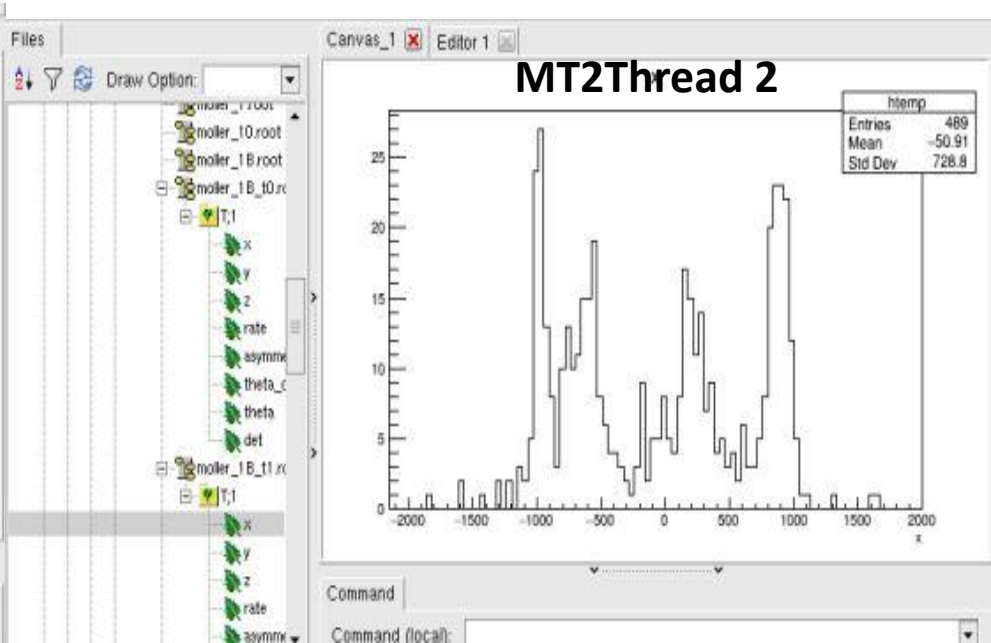
5000 event runs
Seed=123456
Number of Threads=5
Number of Events Per Thread=1000
Looking at x-position
Of hits on detector plane.



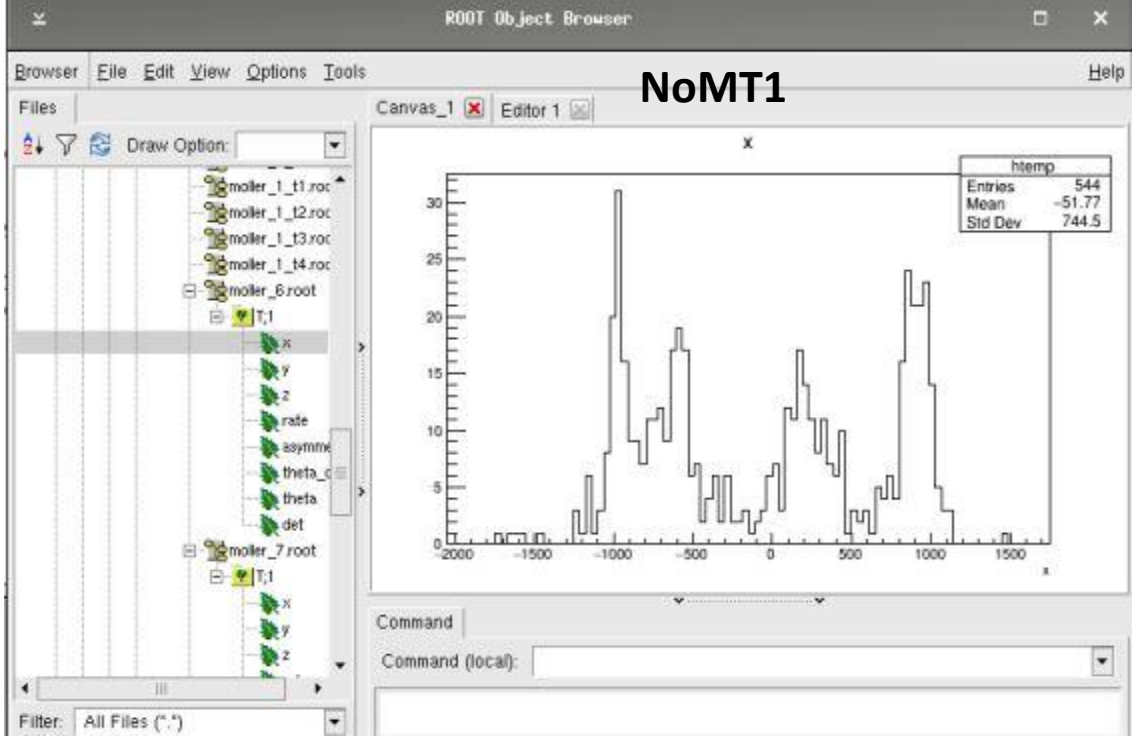


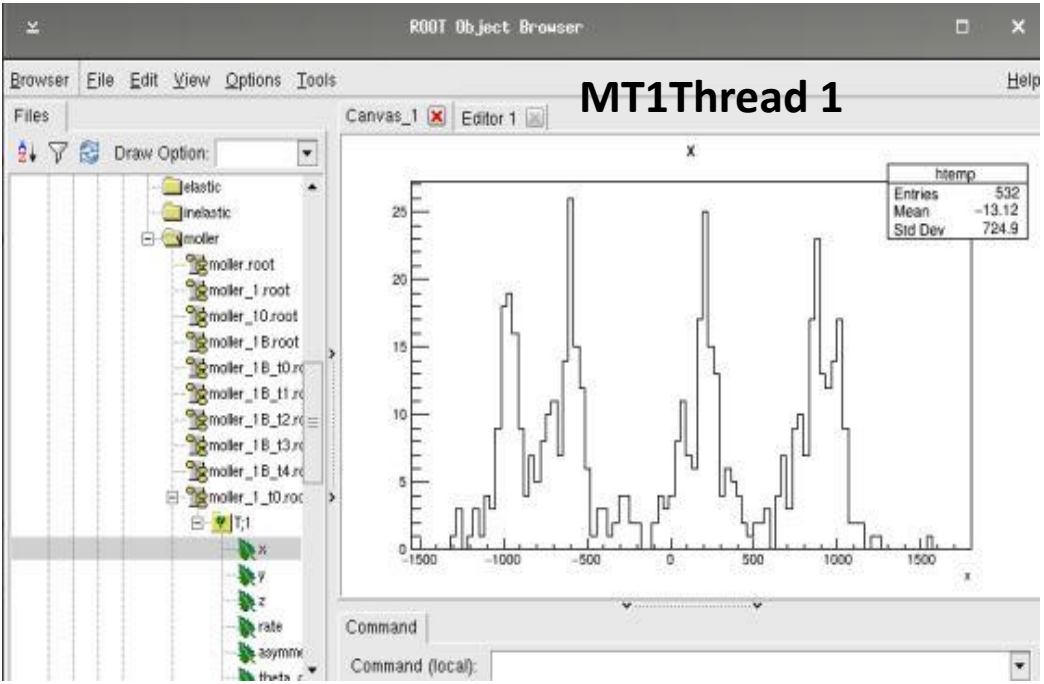
2. Threads within and across runs don't produce identical results with same seed.



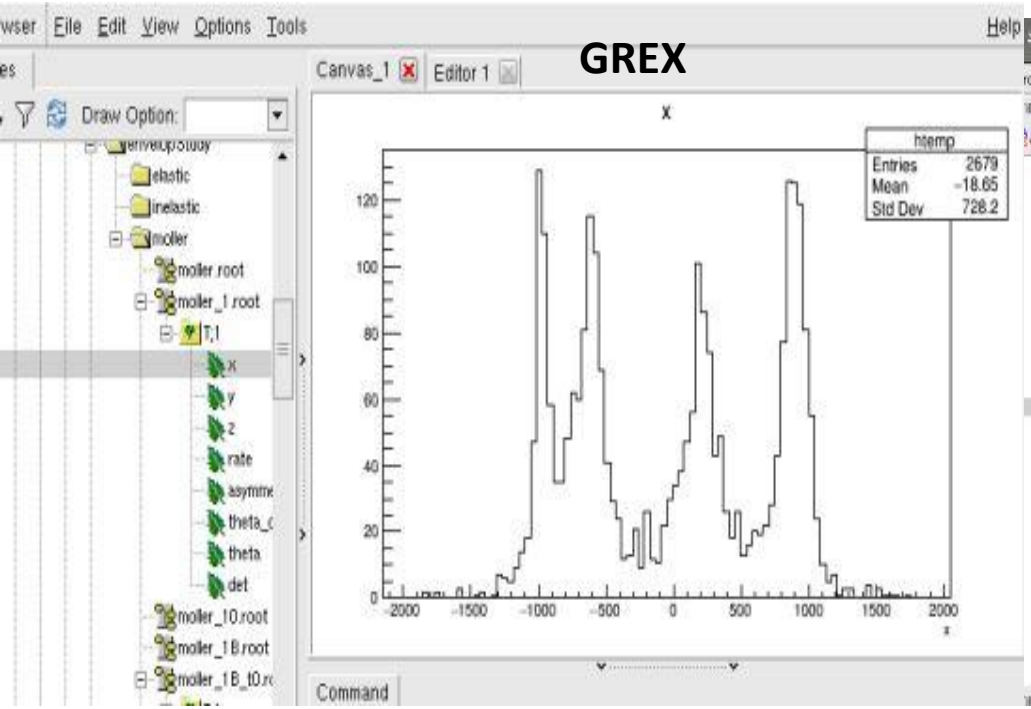


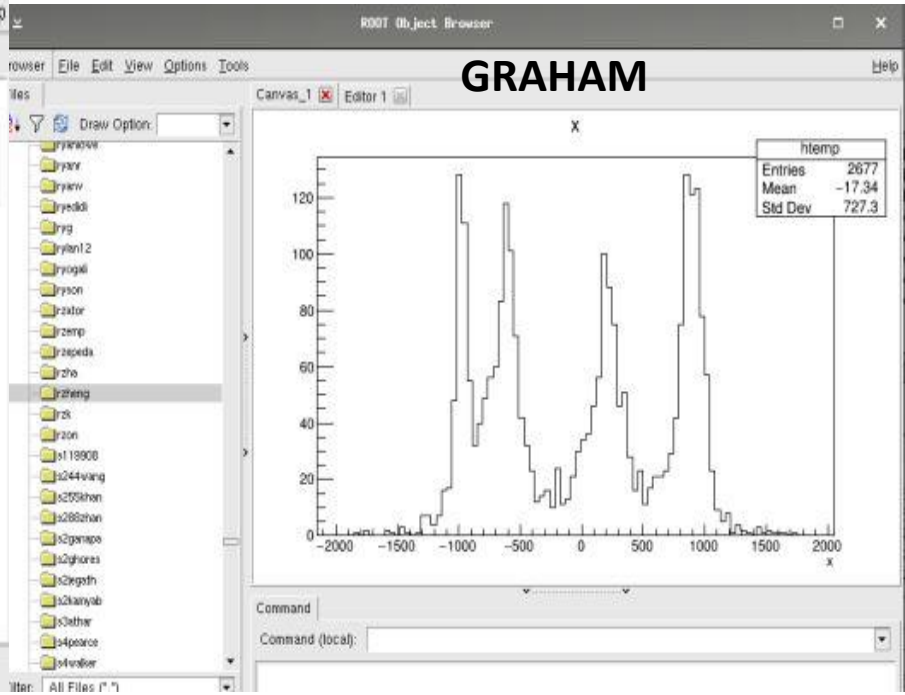
3. Results from MT and NoMT runs which use different random number generators





GREX vs Graham (Different Clusters)





May be the negligible difference in the output statistics can be attributed to difference in compiler versions