Geant4 in ATLAS

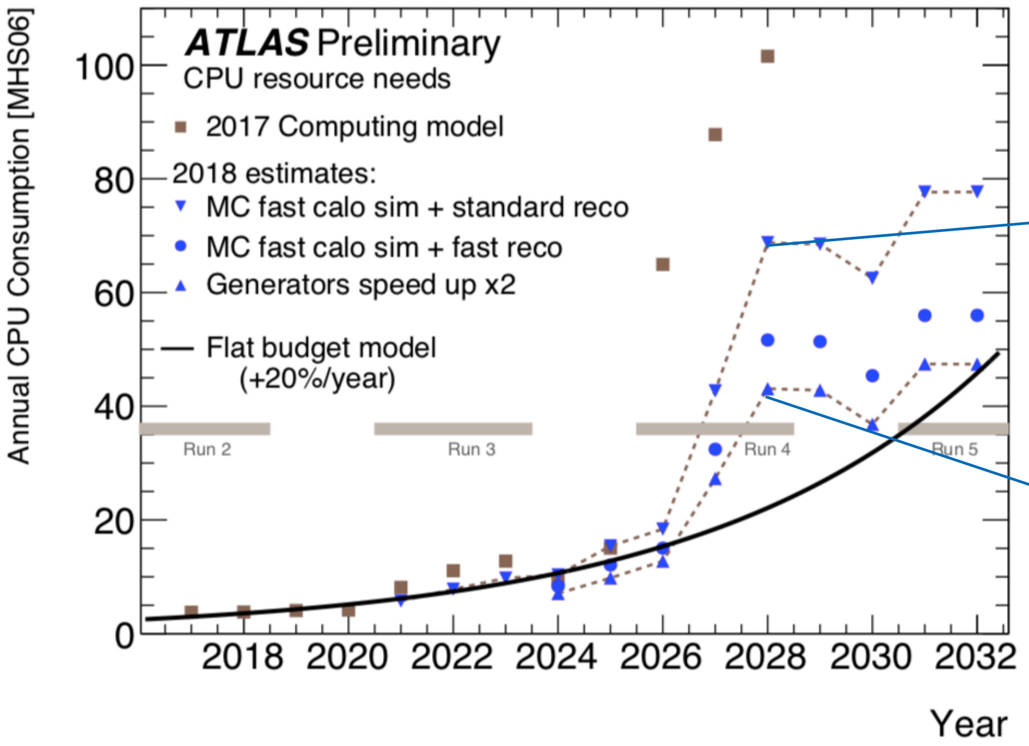
Geant4 Technical Forum

16th January 2020

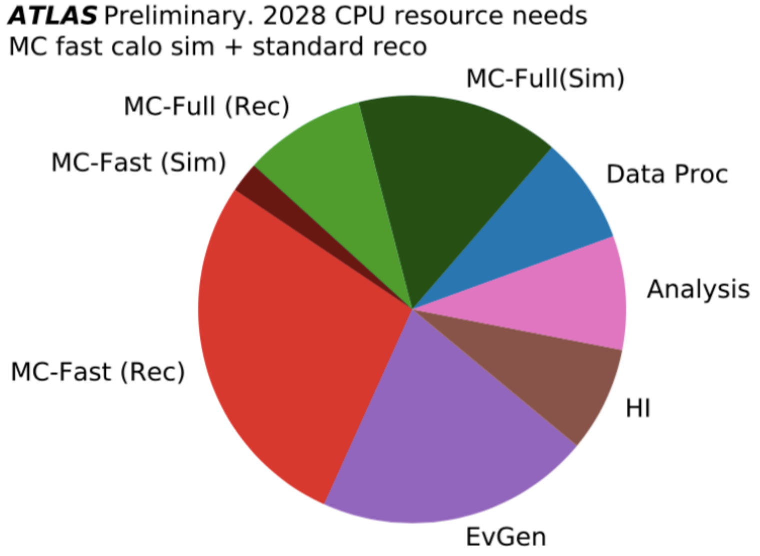
Marilena Bandieramonte

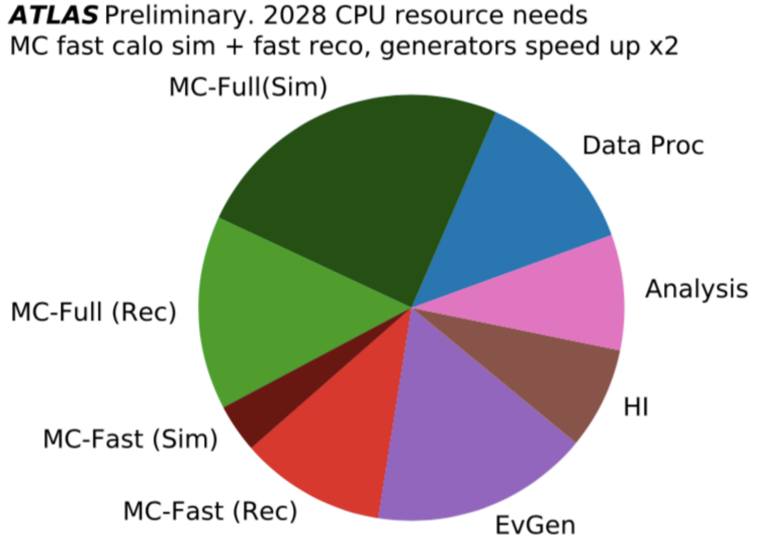
on behalf of the ATLAS collaboration

Computing complexity challenges



[ATLAS public]





- In Run3, we plan to run at least 50% of simulation with fast techniques (we aim to reach ~75%), but full Geant4 simulation will be heavily used regardless
- In Run 4, Full Simulation is expected to be the largest CPU consumers (20-25%)
  - Together with FastSim and FastReco it amounts to ~40% of all expected CPU consumption.
- Any performance optimizations of ATLAS simulation have a big impact on the overall picture.

Geant4 optimisations for Run2

We expect to continue using Geant4 10.1.patch03 for Run2 studies

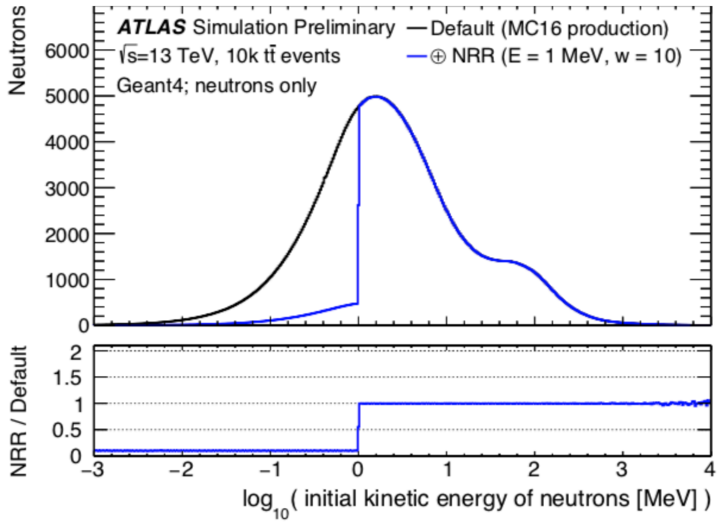
Recent Geant4 optimisations have been adopted as they passed successfully the Physics Validation

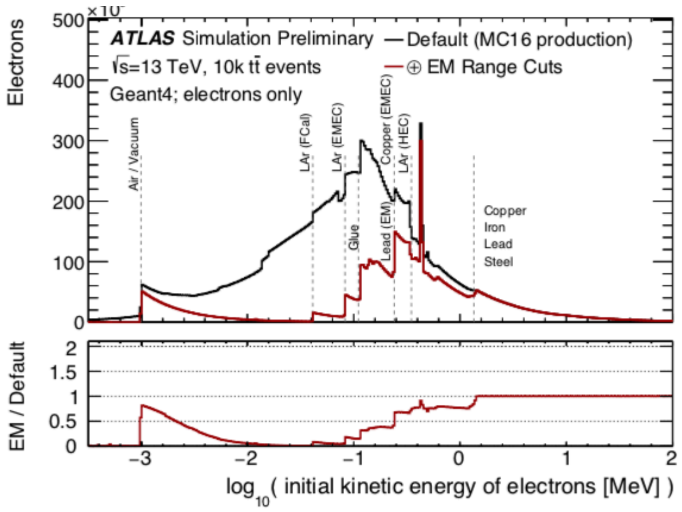
- Neutron Russian Roulette
- EM range cuts

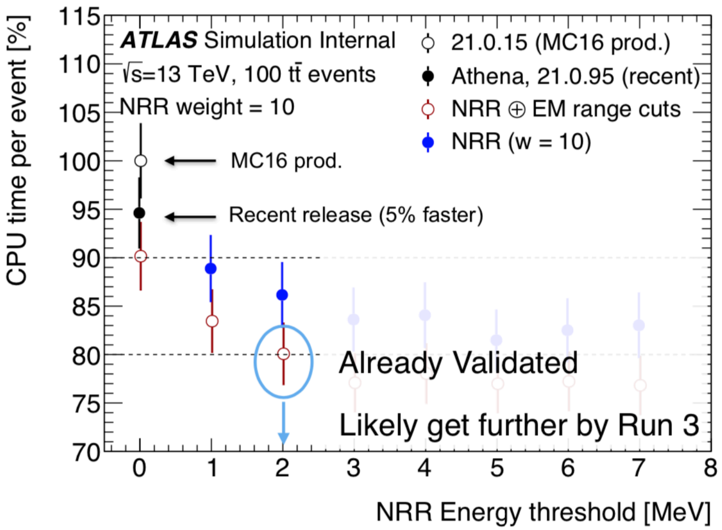
We will use them in future Run2 simulations (platform is x86_64-slc6-gcc62-opt)

Work in progress for Run3: Photon Russian Roulette, Geometry Optimizations, VecGeom

More details here (M. Muskinja)







Testing new Geant4 versions for Run3

We are going to decide which **G4 release we will use for Run3** *(converge in Summer 2020)*

Our plan is to build **Athena 21.0.X** and **Athena 22.0.X** releases based on the following **Geant4** versions candidates:

- 10.1.patch03.atlas07 (MC16 version)
- 10.4.patch03.atlas01 (current R22 version)
- 10.5.patch01 (or latest patch)
- 10.6.patch00 (or latest patch)

Get *testing workflow* working on limited number of configurations: **G410.1.p03.atlas07 in 21.0 and 22.0 + FTFP_BERT_ATL + AtlasRK4**

- (Re-)Calculate sampling fractions
- Look at single particle samples and phys val samples
- Update Birks' constant

Testing new Geant4 versions for Run3

Tuning Birks' Constant values:

- Use **Tile TestBeam** data and potentially **HEC TestBeam** data (and p-p data)
- Change value, recalculate sampling fractions, compare **ratio of EM to Hadronic response** to that of data. Iterate.
- Potentially **one tune** per physics list/G4 version combination (independent values for LAr and Tile)
- Any suggestions of **sensible ranges** of values of Birks' constant to test for different materials like LAr or the Tile scintillators?

Test different steppers in later G4 releases:

- Integrate any recent studies from **Tracking** (e.g. hadronic interactions, secondaries), **Jet/MET** (e.g. E/p studies, cluster variable studies), **MCP** (e.g. energy loss), *discuss with Geant4 experts*.
- Offer new samples to Combined Performance groups as far in advance as possible for cross-checks to ensure that **data/MC agreement is not harmed.**
- Produce **large-scale** (>1M event) **ttbar** sample production to evaluate **crash rate** of new G4 version.
- If crash rate sufficiently low (below 1% of jobs, better below 0.1%), launch **"Physics Validation" checks** comparing old and new Geant4 versions, some checks expected to fail until conditions have been fully updated for new Geant4 physics list, but it should flag key issues.

Recalculate the sampling fractions, re-do the Frozen Showers tune, and the FastCaloSim parameterisation to match the new Geant4 version

AthenaMT: why multi-threading?

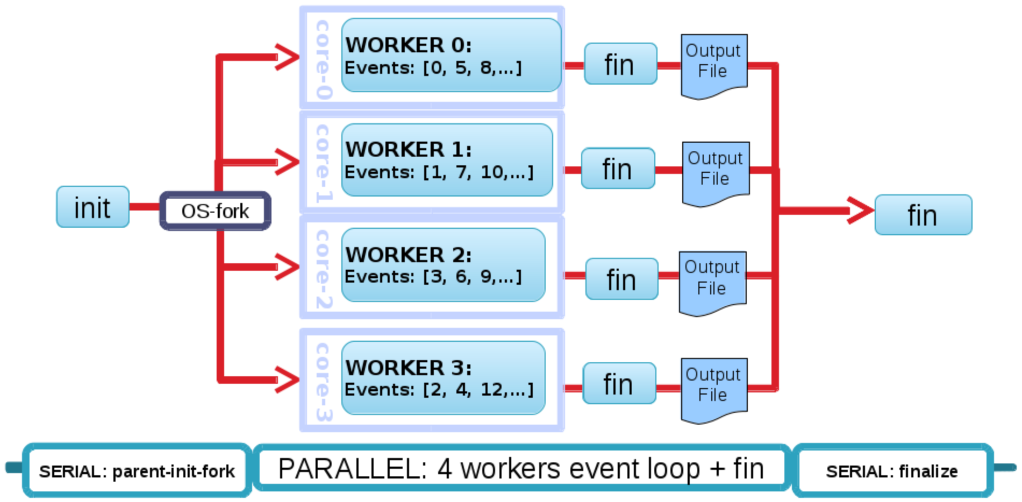
- The amount of **Monte-Carlo** that can be produced already **limits many physics analyses** and this will get **worse** with the increased luminosity expected

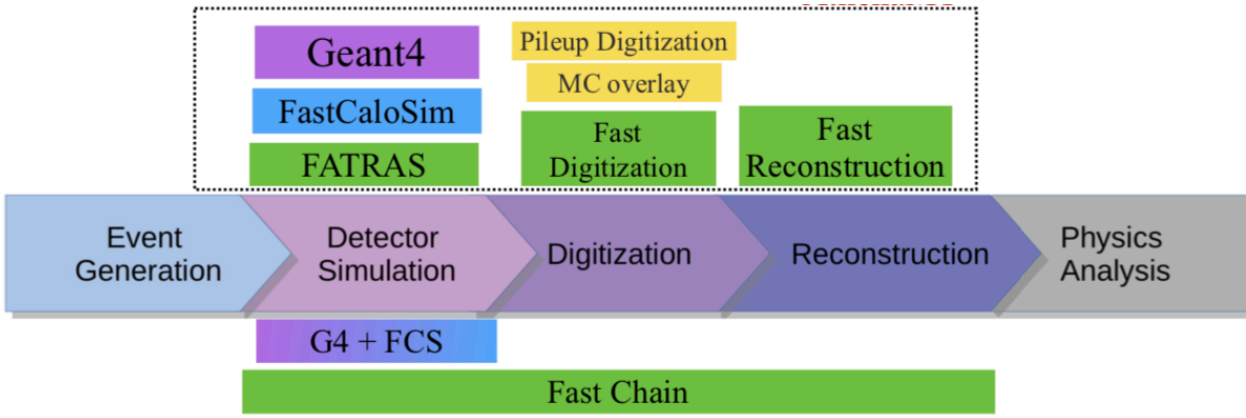
- The current model, **AthenaMP,** relies on Linux's *copy-on-write* mechanism for sharing memory pages between forks:
  - won't scale for Run-3 and beyond

- Ongoing effort to migrate **ATLAS** computing model to ***multi-threaded*** **AthenaMT**
  - Finer-grained *task parallelism*, minimised memory footprint
  - Only execute() is concurrent
  - *Scheduler-driven*, by dependency graph

- **Simulation**, **Digitization** and **Reconstruction** moving to MT paradigm using the AthenaMT/GaudiHive infrastructure.
  - Better scaling in terms of memory footprint (leverage new architectures)
  - Easy the investigation of heterogeneous computing architectures (e.g. use GPUs, FPGAs etc)

Schematic View of ATLAS AthenaMP

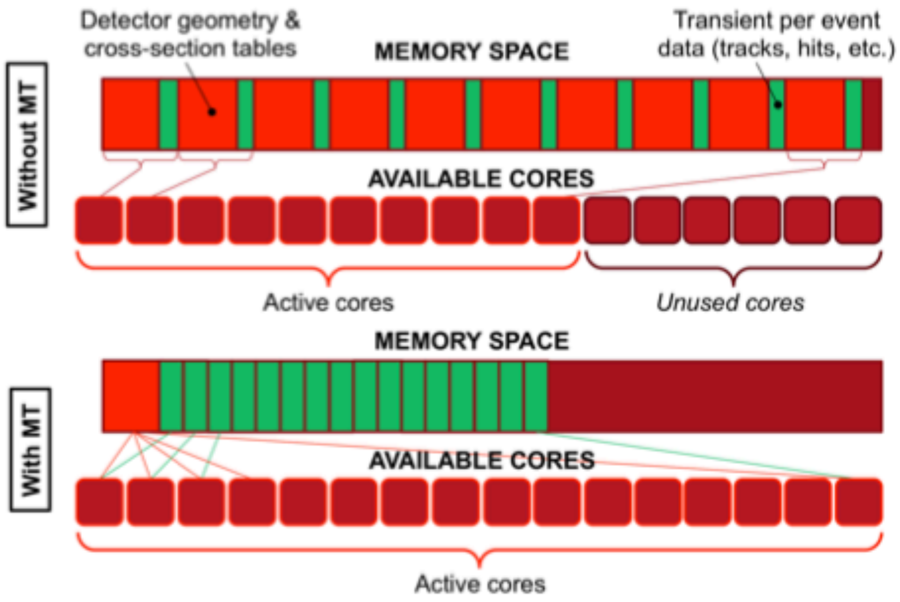




Thread coupling AthenaMT and G4MT

- **Geant4MT** has been successfully integrated in **AthenaMT outside of the Integrated Simulation Framework (ISF), first**
    - Inter-event rather than intra-event parallelism:
        - memory saving coming from sharing geometry and cross-section tables between threads

- **Segfaults** during execution or finalization of MT jobs, due to the way TBB starts new threads:
    - During **execution** of a MT job:
        - **TBB can spawn new threads** even after initialization is complete
            - The simulation was aborted because the geometry was released after the initialization but it is always needed to initialize new threads
    - When **finalizing** a MT job:
        - TBB creates extra-threads that are not catched by the ThreadPoolSvc -> no call to G4ThreadInitTool::initThread
            - Crashes when G4ThreadInitTool::terminateThread is called for those threads



AthenaMT & Geant4MT validation

- **The Athena Multi-threaded simulation with Geant4MT is fully functional**

  - **Outside of ISF:**
    - Fixed: thread-unsafety causing difference in HITS of LAr sensitive detector (~1-2%)
    - Fixed: thread-unsafety causing difference in HITS of Tile sensitive detector (~1-5%)
    - Fixed: simulation with CaloCalibrationHit (~50% of Dead material hits)
    - Confirmed/Fixed: reproducibility of simulation with **SUSY/Exotics G4Extensions** enabled
    - The G4 single threaded vs multi-threaded output has been confirmed **to be identical**
    - 100k grid test were ran with 8 cores without reported issues **(Physics Validation in progress)**

  - **Inside ISF:**
    - Revision of the **Geant4 initialisation step in MT mode** inside the Integrated Simulation Framework
    - Fixed: thread-unsafely causing differences in HITS of EntryLayerTool
    - Simulation runs correctly in multi-threaded mode with more than 1 thread and the output **has been validated** with 1000 ttbar events.
    - Next steps: *Physics Validation Campaign*

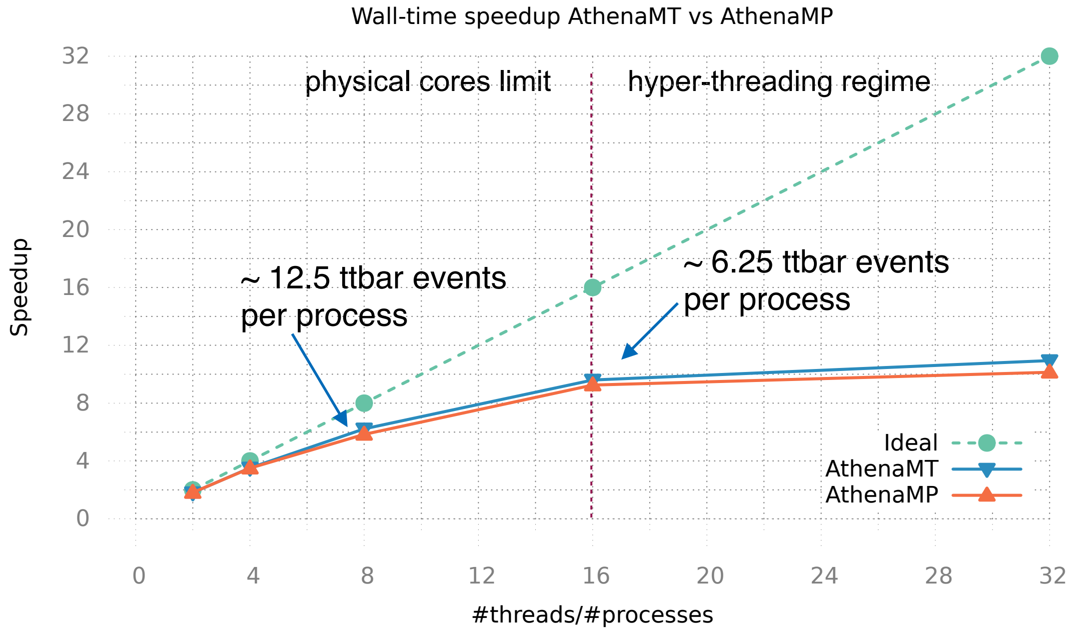
AthenaMT vs AthenaMP benchmarks

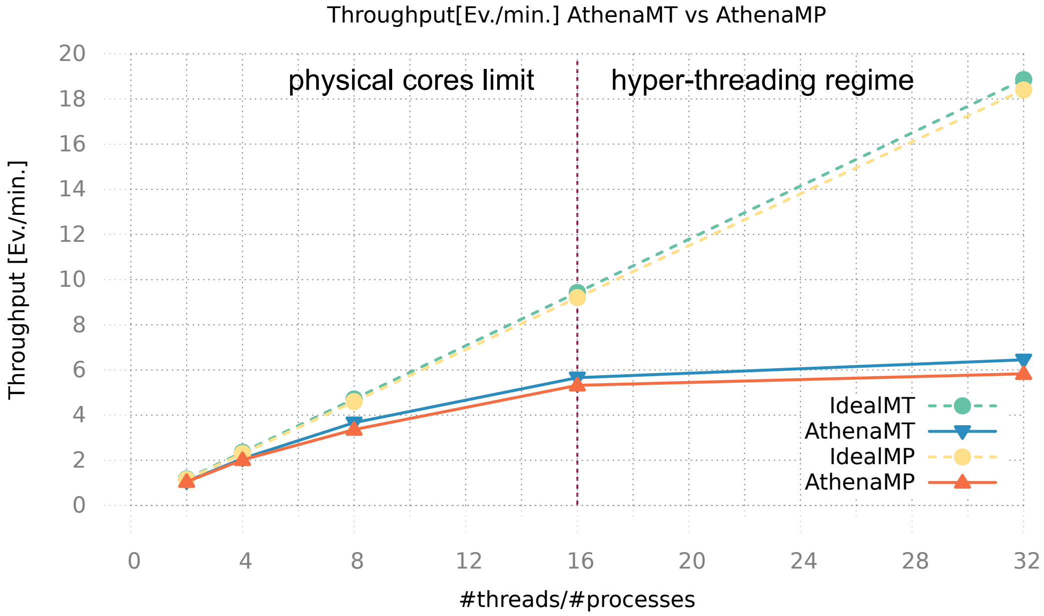
```
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                32
On-line CPU(s) list:   0-31
Thread(s) per core:    2
Core(s) per socket:    8
Socket(s):             2
NUMA node(s):          2
Model:                 79
Model name:            Intel(R) Xeon(R)
                         CPU E5-2620 v4 @ 2.10GHz
Test on 100 ttbar events, with prom
Athena, r2019-09-30T2130,master

results are AVG of 5 separate runs (from 1-32
threads/processes) - the machine was quiet all the
time (me as only user)
```

$$\text{AthenaMT Speedup}_{th\_n} = \text{Wall-time}_{th\_1} / \text{Wall-time}_{th\_n}$$

$$\text{AthenaMP Speedup}_{proc\_n} = \text{Wall-time}_{proc\_1} / \text{Wall-time}_{proc\_n}$$

Ongoing weak scaling benchmarks: with 50 ttbar events per thread

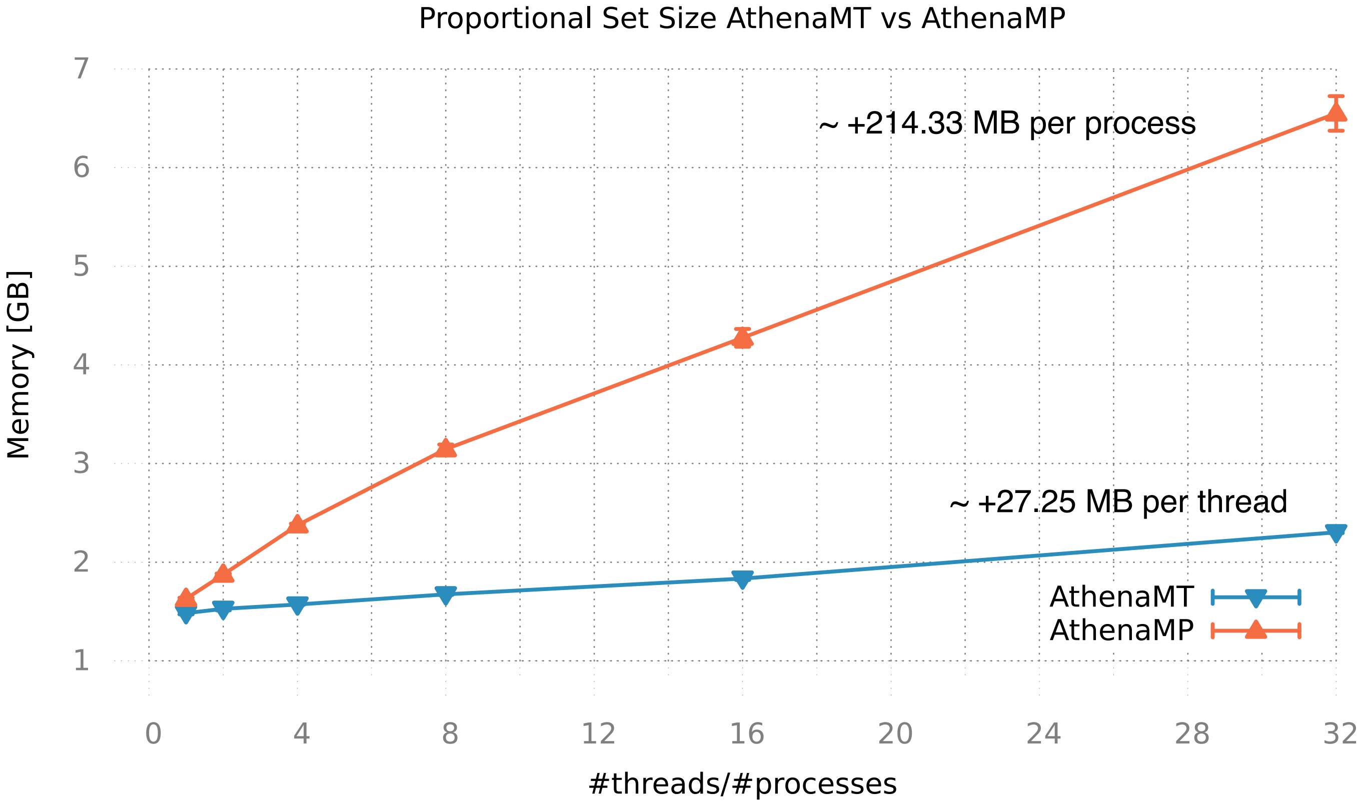




AthenaMT vs AthenaMP benchmarks

```
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                32
On-line CPU(s) list:   0-31
Thread(s) per core:    2
Core(s) per socket:    8
Socket(s):             2
NUMA node(s):          2
Model:                 79
Model name:            Intel(R) Xeon(R)
                         CPU E5-2620 v4 @ 2.10GHz
Test on 100 ttbar events, with prom
Athena, r2019-09-30T2130,master

results are AVG of 5 separate runs (from 1-32
threads/processes) - the machine was quiet all the
time (me as only user)
```

PSS[GB]	1 thread/process
AthenaMT	1.482771301
AthenaMP	1.628312683



The Proportional Set Size (PSS) is the portion of main memory occupied by a process and is composed by the private memory of that process plus the proportion of shared memory with one or more other processes

Quasi-stable particle simulation

- **Quasi-stable particles** (e.g. B mesons) can have a significant **lifetime** and can decay even beyond the first Pixel layers

- **Decays occur in the generator**, so these particles are not correctly simulated since only stable (status == 1) particles get simulated

- **QS particle simulation** adds non-stable particles to the simulation:
    - Particles treated by QS-sim are pre-assigned the lifetime and decay products from the generator so that the GenEvent truth record is reproduced

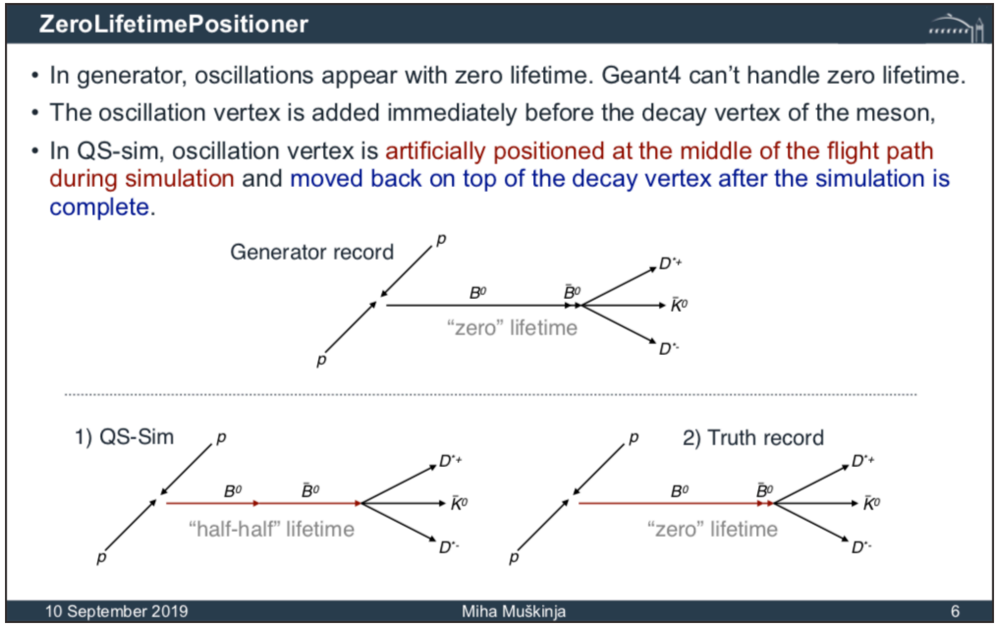
- Differences between the **default** and **QS sim**:
  - Charged QS particles bend in the magnetic field
  - QS particles can interact with the detector material and deposit energy if the appropriate physics processes are implemented in Geant4.

- Mainly useful for ***b*-tagging efficiency** measurements and **B physics**

- Currently, **all quasi-stable particles** (except status == 3) are propagated through **Geant4** simulation

Quasi-stable particle simulation

- **ZeroLifetimePositioner is** a fix for lifetime vertices in neutral meson mixing
  - Without the ZeroLifetimePositioner, we would preset the lifetime of **B0 mesons** immediately after the oscillation vertex to zero
  - In this case, **Geant4 does not immediately decay B0s** as it "should" but rather keeps propagating them,
    - This artificially increases the lifetime



ZeroLifetimePositioner **off**

```
Vertex:      -284 ID: 1201 (X,cT)=+5.82e+00,-8.24e-01,-3.50e+01,+1.46e+01
 Wgts(1)=2.8e+02
 I: 1        279      511 +1.27e+05,-6.59e+03,-2.64e+05,+2.93e+05  2      -284
 O: 1      10019     -511 +1.27e+05,-6.59e+03,-2.64e+05,+2.93e+05  2      -285
Vertex:      -285 ID: 1201 (X,cT)=+7.20e+00,-8.96e-01,-3.79e+01,+1.78e+01
 Wgts(1)=1e+04
 I: 1      10019     -511 +1.27e+05,-6.59e+03,-2.64e+05,+2.93e+05  2      -285
 O: 6      10020     -311 +2.30e+04,-9.21e+02,-4.84e+04,+5.36e+04  2      -286
           10022      323 +1.82e+04,-1.06e+03,-3.83e+04,+4.24e+04  2      -287
```

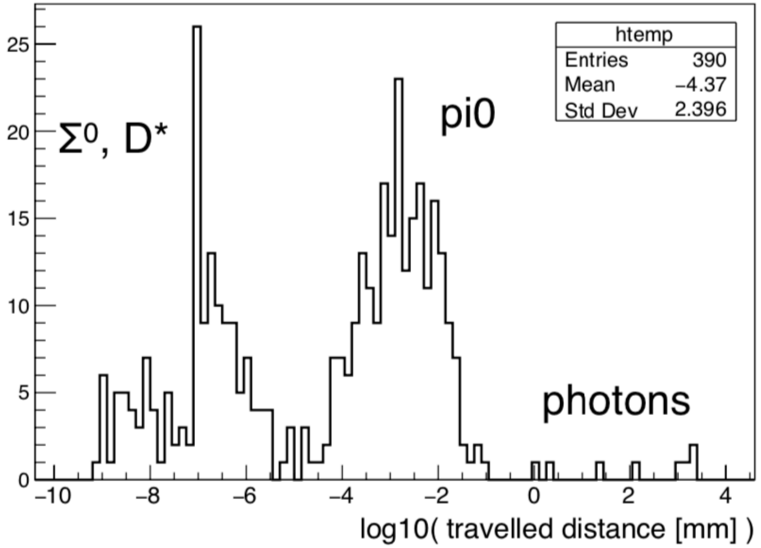
ZeroLifetimePositioner **on**

```
Vertex:      -284 ID: 1201 (X,cT)=+5.82e+00,-8.24e-01,-3.50e+01,+1.46e+01
 Wgts(1)=2.8e+02
 I: 1        279      511 +1.27e+05,-6.59e+03,-2.64e+05,+2.93e+05  2      -284
 O: 1      10019     -511 +1.27e+05,-6.59e+03,-2.64e+05,+2.93e+05  2      -285
Vertex:      -285 ID: 1201 (X,cT)=+5.82e+00,-8.24e-01,-3.50e+01,+1.46e+01
 Wgts(1)=1e+04
 I: 1      10019     -511 +1.27e+05,-6.59e+03,-2.64e+05,+2.93e+05  2      -285
 O: 6      10020     -311 +2.30e+04,-9.21e+02,-4.84e+04,+5.36e+04  2      -286
           10022      323 +1.82e+04,-1.06e+03,-3.83e+04,+4.24e+04  2      -287
```

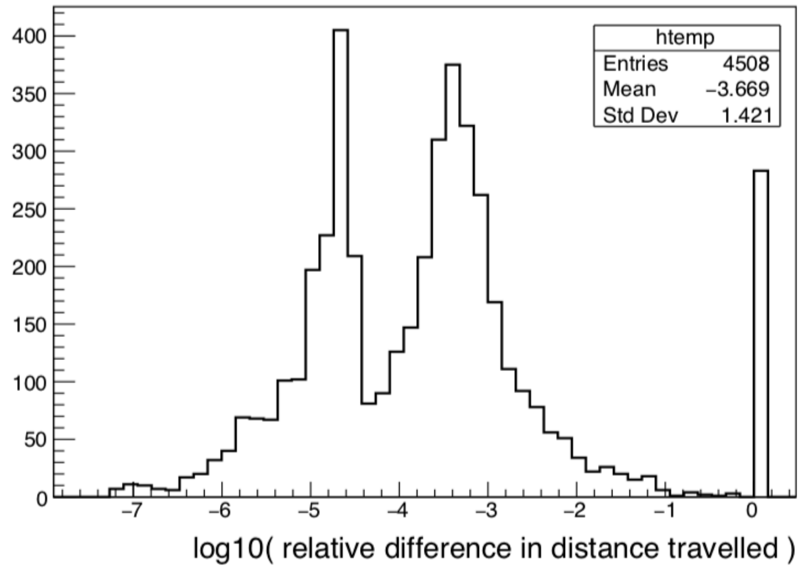
More details here (M. Muskinja) and here (S. Gargiulo)

Quasi stable particle simulation

- **Other zero lifetime cases:**
  - Zero lifetime in the generator, but gets a non-zero lifetime in QS simulation: Geant4 always takes **at least one step** with a new particle, artificially offsetting vertices.
- **Non-zero lifetime cases:**
  - Neutral particles do not travel the same distance in QS-sim and in the generator: relative difference at the order of 10^{-3} to 10^{-5}



zero lifetime particles



non-zero lifetime particles

- Would it be possible to update the code related to **pre-defined decays** to be more robust against things that the generators create (Like particles with zero or very small lifetimes)?

- Related request is the **implementation of b-physics models** (important news in this respect?)

Summary

- **We are starting a study to decide what Geant4 version to use for Run3**
  - Geant4.10.5 and Geant4.10.6
  - Update of Birk's constant

- **The Athena Multi-threaded simulation with Geant4MT is fully functional**
  - Output of single-thread simulation confirmed to be identical to the one produced with more than one thread both outside the ISF and inside.
  - Physics Validation campaigns are in progress

- **Quasi-stable particle simulation**
  - Request that the code related to pre-defined decays is updated to be more robust against things that the generators create
  - How to deal with other cases of zero lifetime in the generator? Only simulating particles with status < 3 would help, but it does not solve all cases:
    - Figure out exactly what happens when a particle is aligned 0 lifetime in Geant4
  - In case of non-zero lifetime, flight path changes slightly in the simulation,
    - Why exactly does this happen and is there a way to fix it?

Thanks for your attention!

Marilena Bandieramonte
marilena.bandieramonte@cern.ch

Backup slides

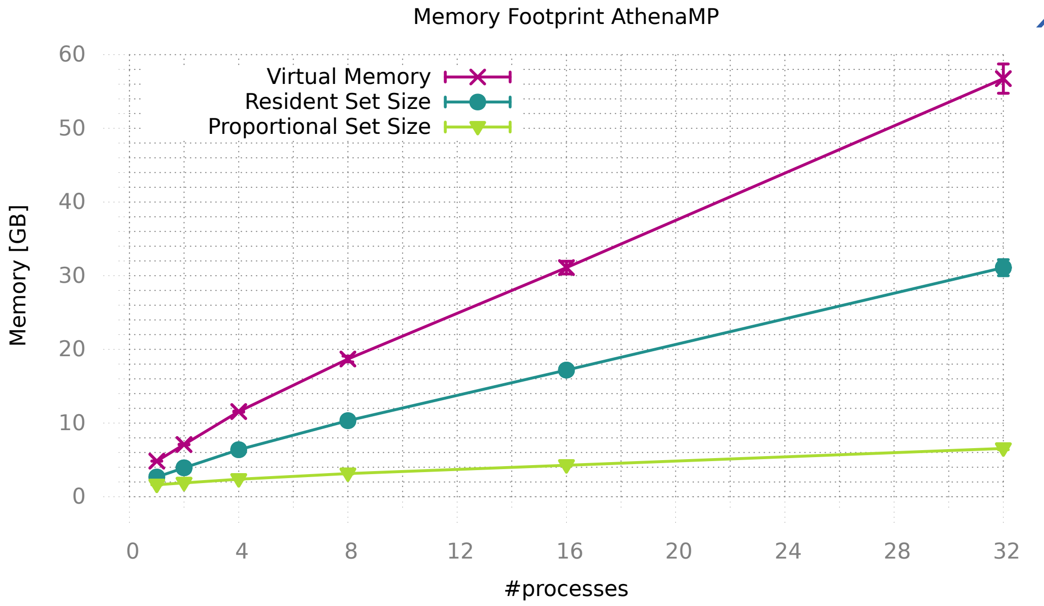
'ExtraParticles' package

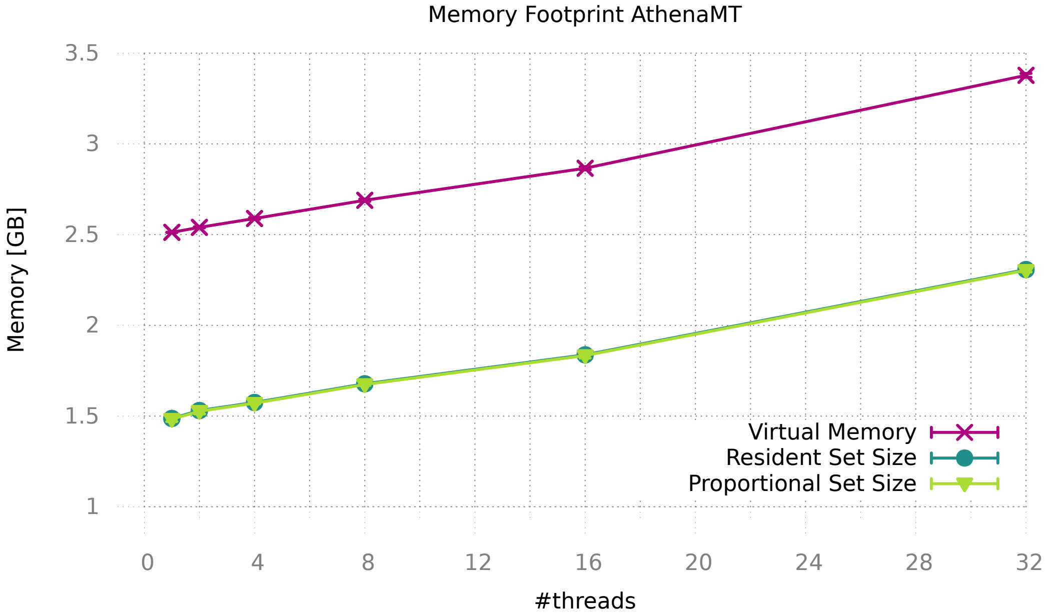
- Geant4 does not include all SM particles by default,
- Particularly, it does not include vector mesons such as D^* or D^{*+}, so B mesons decaying into vector D mesons could not be simulated,
- The ExtraParticles package adds all missing SM particles in Geant4 using the PDG table,
  - Charged particles are assigned EM physics and neutral particles have no physics processes assigned,
- ART test: test_MC16_FullG4_QS_ZPrimebb_ExtraParticles.sh.

```
02:57:06 ---------> Dumping now track #932400 barcode 10001 pdgID -511 in event 5
02:57:06     1     7.16     7.27    -13.2  1.08e+05        0     11.2       11.2  SectionC01 Decay
02:57:06     :----- List of 2ndaries - #SpawnInStep=  6(Rest= 0,Along= 0,Post= 6), #SpawnTotal=  6 -------------
02:57:06     :       7.16      7.27     -13.2   2.7e+04               rho-
02:57:06     :       7.16      7.27     -13.2  3.34e+03                pi+
02:57:06     :       7.16      7.27     -13.2  9.43e+03                eta
02:57:06     :       7.16      7.27     -13.2  1.34e+04              kaon0
02:57:06     :       7.16      7.27     -13.2  6.37e+03              kaon-
02:57:06     :       7.16      7.27     -13.2  4.98e+04      D_star(2010)+
02:57:06     :------------------------------------------------------------- EndOf2ndaries Info -------------
```

AthenaMT vs AthenaMP benchmarks

```
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                32
On-line CPU(s) list:   0-31
Thread(s) per core:    2
Core(s) per socket:    8
Socket(s):             2
NUMA node(s):          2
Model:                 79
Model name:            Intel(R) Xeon(R)
                         CPU E5-2620 v4 @ 2.10GHz
Test on 100 ttbar events, with prom
Athena, r2019-09-30T2130,master

results are AVG of 5 separate runs (from 1-32
threads/processes) - the machine was quiet all the
time (me as only user)
```





AthenaMT vs AthenaMP benchmarks

```
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                32
On-line CPU(s) list:   0-31
Thread(s) per core:    2
Core(s) per socket:    8
Socket(s):             2
NUMA node(s):          2
Model:                 79
Model name:            Intel(R) Xeon(R)
                         CPU E5-2620 v4 @ 2.10GHz
Test on 100 ttbar events, with prom
Athena, r2019-09-30T2130,master

results are AVG of 5 separate runs (from 1-32
threads/processes) - the machine was quiet all the
time (me as only user)
```