Geant4 in ATLAS

Zach Marshall (LBNL) for the ATLAS Sim Team

Geant4 Technical Forum

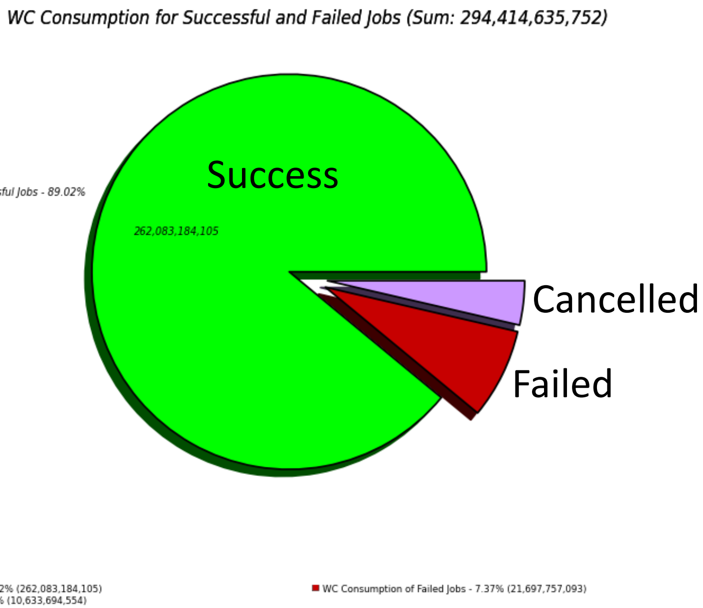
2 April 2015

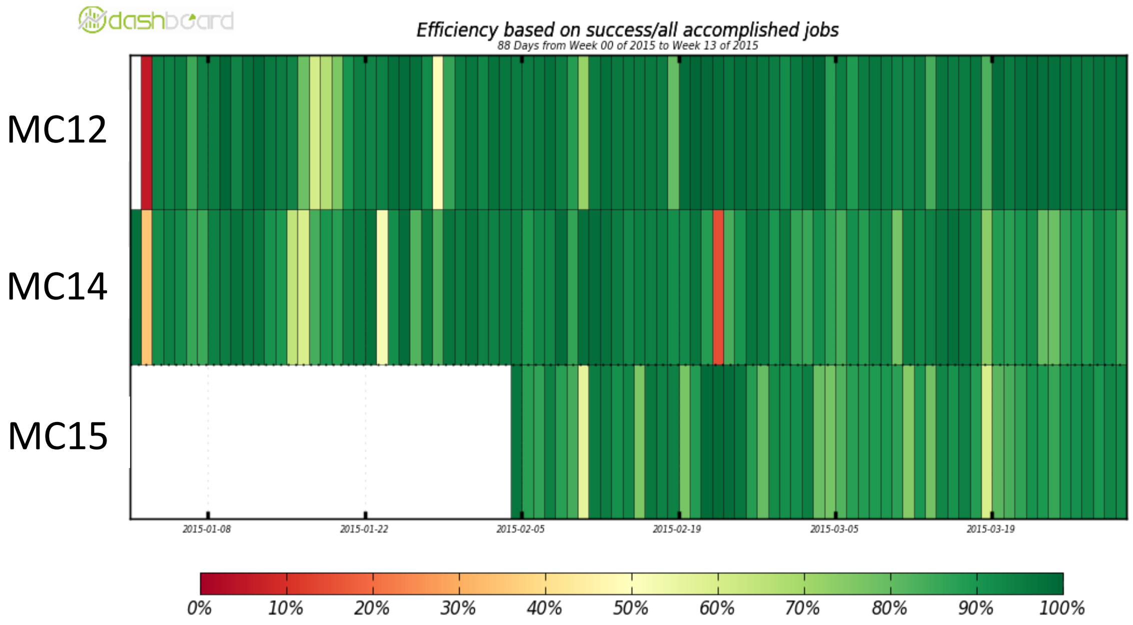
Current Production

- Ramping up 13 TeV MC production for the 2015 run (MC15)
  - Geant4 9.6 patch03 (validating patch04 soon), CLHEP 2.1, 64-bit, gcc 4.7, SLC6, C++11; this is our platform through 2015
  - Enormous number of ATLAS-specific updates (geometry and detector response), including several speed ups
  - Moving to our new "ISF" infrastructure by default
  - Starting to run production on more exotic machines: HPCs, Amazon cloud, BOINC all being validated
- Still running tails of older productions
  - Geant4 9.4+ patches for "MC12" production
- Getting ready for tests of MC16 (!)
  - Geant4 10.1, CLHEP 2.2, 64-bit, gcc 4.9 (?), SLC6, C++14
  - Expecting this to be the main production platform through 2016
  - Still testing ICC, Clang, Mac OS X builds (still no production plans)

Production Statistics

- Generally production running with 80-90% efficiency
  - No clear CPU gain in production w.r.t. previous years either ☹
- No major signs of differences between MC campaigns; no sign of major differences between Geant4 versions in crash rate
  - Most large failures you see are still grid / site failures, which are significantly more common than Geant4 failures
  - *Significantly* improving our monitoring tools now (as you can see), and we hope that soon we will be able to give much more precise crash rate information





Production Plans

- Working on a reasonable physics list variations
  - FTFP_BERT_EMZ/EMY for EM physics variations in testing
  - Received hadronic physics variation options for systematic studies (thanks Alberto!); now preparing to test these in the production system
  - The next run will include particles hitting a calorimeter that have higher energy than ever before. Are we sure the Geant4 hadronics and EM models are going to be up to it? We will feed back as quickly as possible, and we hope you all can help respond and improve the high-energy models as we get new information!
  - By the next G4 technical forum we should have MC produced, and we may also have *data* to compare it to!
  - We still have long-standing issues with lateral shower shapes; we'd love to hear about any progress on those (first tests of EMY are not terribly promising)
- Working on ApplyCuts for calorimeter simulation speed up
  - Is it clear whether this will just cancel out the gain from _EMZ?
- Completing validation of the simulation of quasi-stable particles (e.g. Bs, Ds) that could bend in a magnetic field or leave deposits in silicon but need to be decayed by the generator. Important for highly boosted systems – noticeable effects on b-tagging. Part of default simulation soon.

Bugs / Frustrations / Questions

- Does G4 have any plan to use RandGaussZiggurat (Available in CLHEP 2.2) ?
  - Our tests showed RandGaussQ to not be very Gaussian; at least worth checking.
- Need a patch to allow the use of granular CLHEP libraries
  - New segv in our tests from the same symbol being defined in multiple places
- "Hyperspace" bug is causing a fairly high rate of crashes and appears more prevalent than previously thought
  - Particles entering a volume and then continuing for km. Originally thought to be due to G4PolyCone problems, but now suspecting G4MultiLevelLocator
- Very long outstanding issue of tiny steps
  - Many steps that are order fm long. Still not understood. Does not appear to be ATLAS-specific, but also isn't showing up in the simple Geant4 examples.
- Apparent issue in field management and stepping (incorrect field manager being used in navigation), still being worked out with John A (to be confirmed)
- We have many materials that are copies of each other with different names. Is there a way to reduce the memory consumption in this sort of situation?
  - Design choice, to be able to change the density of the materials in a region of the detector that we do not understand well; different materials allows this in principle, though we have not used the handle yet.
  - Starting to also look into non-uniform density materials, as our forward services have gotten rather complicated in run 2. Want to avoid simulating every cable.
- Can the hadronic cross section speed improvements (from ASCR analysis) be put into production soon? What about taking cross sections from a database instead of many small files?

Longer-term Future

- We are right now trying to understand our own plans a bit better, and hope to be able to show you a five-year plan for simulation in the next few weeks
- G4MT Trial / prototype in testing for some time now
  - Making some serious progress on GaudiHive running with 10.1; first muon through the simulation recently (no SDs yet, but real geometry)
  - Learning a lot about multi-threading. Unfortunately, we find different design choices in the manner of multi-threading in GaudiHive (thinking of many algorithms flocking together across many threads) vs Geant4 (thinking of many threads doing work with a single algorithm)
  - This is already raising some interface and patch questions (mostly transmitted by Andrea Dotti so far), and we expect as we learn more we will need to request more patches and modifications from Geant4. This obviously has some impact on discussions about release schedules, as we do not want to carry around private interface changes for very long
- Infrastructure upgrades, mentioned last time, are well underway
  - Rewrite of simulation code to be more Athena/Gaudi-friendly; introducing concepts of tools and services, matching Geant4 concepts like sensitive detectors and user actions
  - Looking into parallel navigation as a solution to a technical problem – will report next time on progress

Feedback on 10.2 Goals

- We strongly support upgrades and improvements to transportation. Recent experience has taught us that most of our longest-standing frustrations come from somewhere inside the transportation code, or in its interaction with multiple scattering and energy loss.
- We strongly support improvements and tuning to high-energy physics models, some of which will only be possible after some run 2 data becomes available. Hadronics and EM physics may both be significantly affected.
  - Part of this is ensuring that we have a *reasonable* set of variations. Pythia and Herwig have always worked as variations because each is well maintained and well tuned. We should not let QGSP / BIC fall by the wayside even if FTFP / BERT are perfect.
- We support efforts to improve throughput, through reduction of memory churn, algorithm modifications and re-writes, general optimization, etc
  - We think it would be helpful to perhaps have a discussion with CMS and other heavy users of the targets for optimization. We would be willing to share (as we have in the past) CPU profiles of "modern" jobs if it would help in this effort.
- With our apologies, it is very difficult to be more specific than this without knowing what the impact of some proposed development might be inside of ATLAS.
- Has anyone gone back through the LPCC slides to see if the issues raised there have (at least mostly) been addressed?