



EDWARD MOYSE

SOFTWARE OVERVIEW

INTRODUCTION

- ▶ No significant changes expected to 21.0.X branches expected this year
 - ▶ For run-2 data taking, you should expect more of the same from our SW
 - ▶ ... however fast sim development is happening in rel 21 & some I/O optimisations e.g. shared writer
- ▶ For run-3 we will need to significantly revamp our software to include:
 - ▶ Migration to AthenaMT
 - ▶ Software for Phase-1 detector upgrades
 - ▶ ...other technical & physics improvements
- ▶ But of course we need to continue to optimise our current software...

SOFTWARE PERFORMANCE OPTIMIZATION TEAM (SPOT)

- ▶ New team to monitor performance of software, both CPU and storage

Work plan for specific tasks

Urgent tasks for the first quarter of 2018

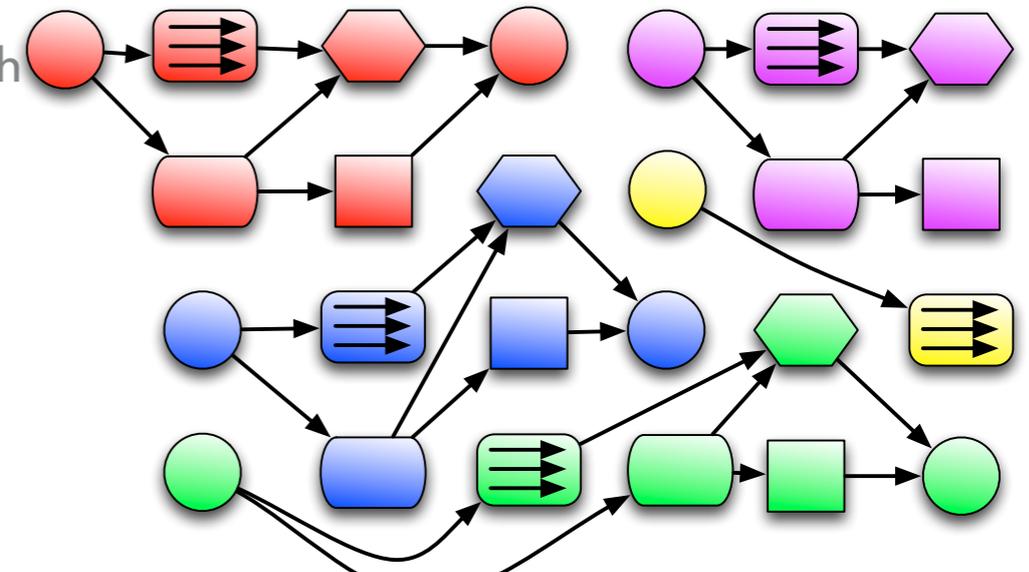
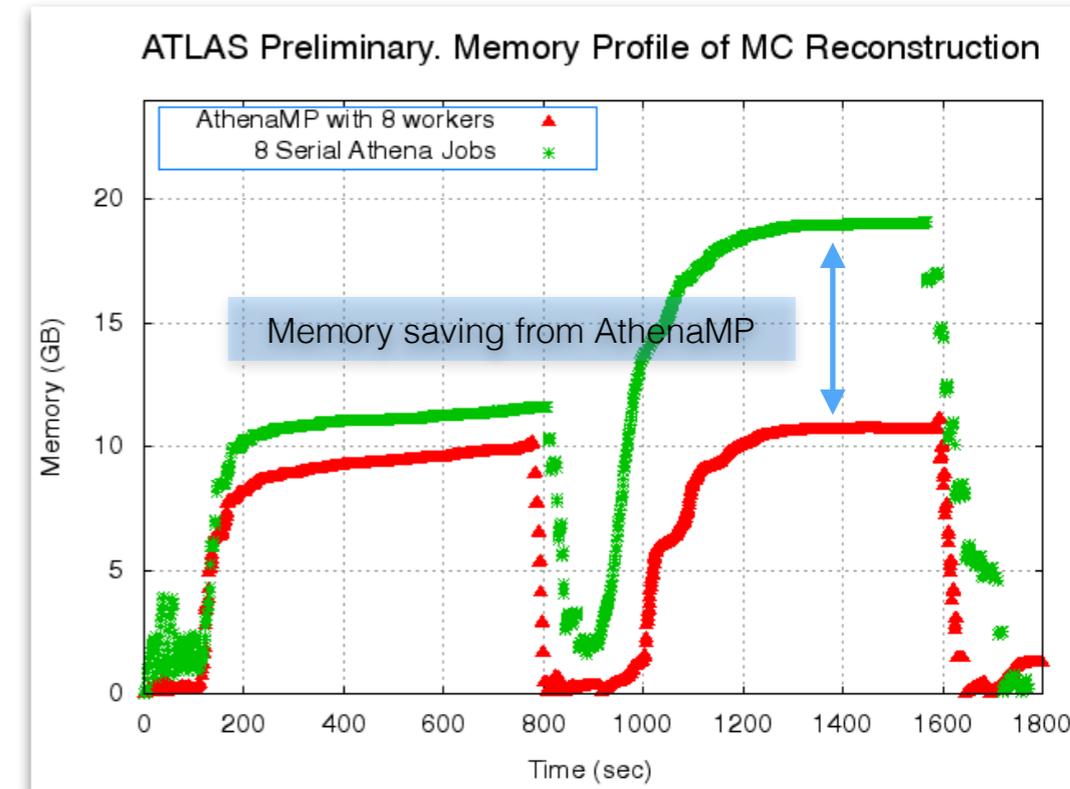
Task description	Impact	Aim to complete by end	JIRA	Indico/Twiki
Precise assessment of the impact of "Spectre" and "Meltdown" on ATLAS		January		691361
Adjust compression settings for AODs	Up to 10% reduction in size, with read-speed penalty of x2-x5 (5% penalty overall)	January	ATEAM-420	691361
Implementation of shared writer (DAOD production)	Dropping of merge step in DAOD production (up to 50% of production time)	January	ATLASDPD-1011	
Review and rationalisation of production error codes	Avoids pointless re-trying of jobs	April		
Full implementation of DAOD size reductions	Up to 30% reduction in AOD size	March		DaodRecommendations
Continue work related to the G4 BigLib	Up to 10% G4 speed up	April		

<https://twiki.cern.ch/twiki/bin/view/AtlasComputing/SoftwarePerformanceOptimization>

- ▶ More in the Thu morning joint ADC/SW [session](#)
- ▶ Other important meetings this week: [SIT](#) and [Core SW](#)

ATHENA MT

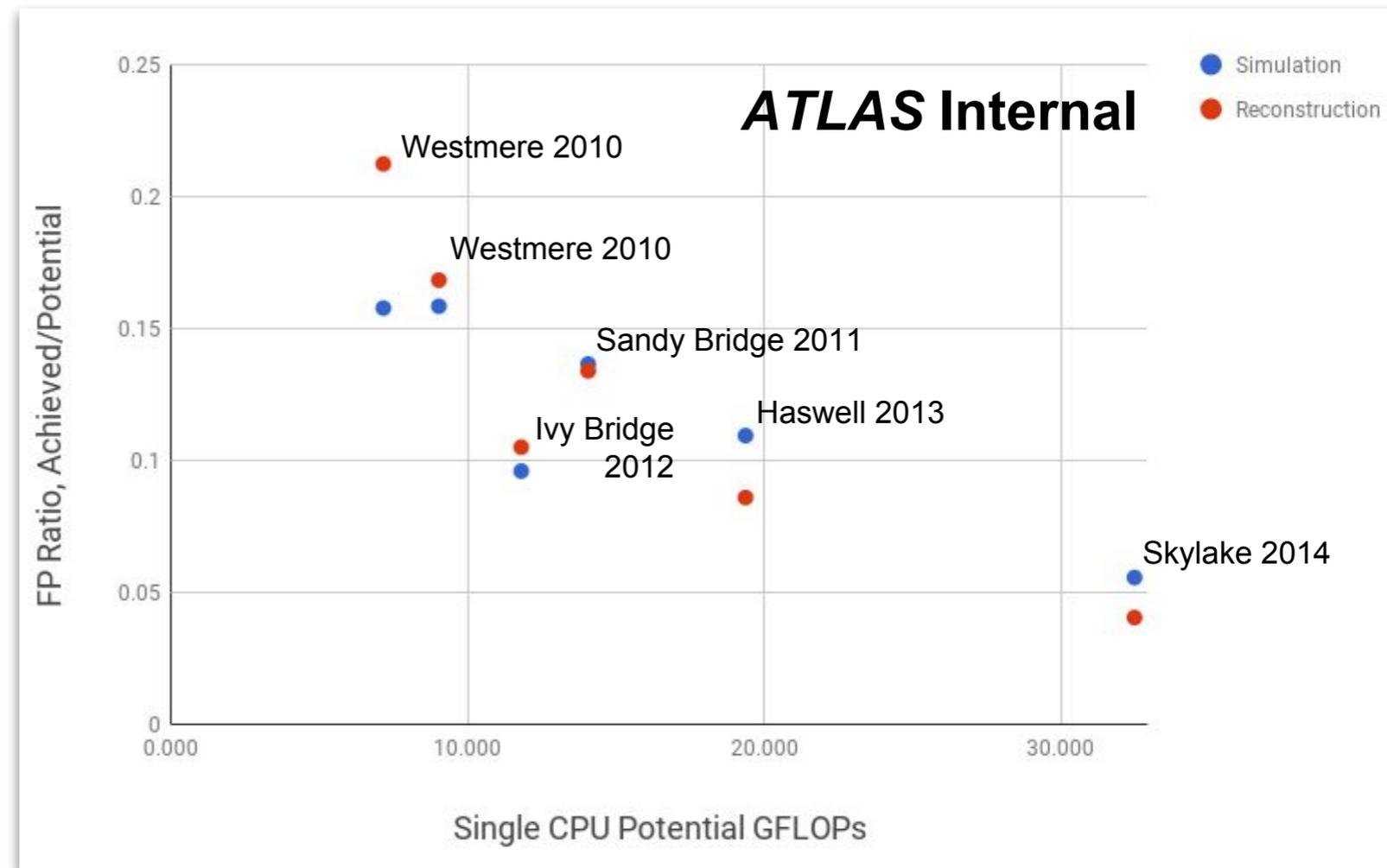
- ▶ Next major upgrade for software is a multi-threaded version of Athena, and migrating to this is the major task for ATLAS software for the next few years
- ▶ Reminder:
 - ▶ In **Run 1** we relied on *serial* Athena running on multiple CPU cores
 - ▶ In **Run 2** we use *multi-process* Athena (**AthenaMP**), that shares large static memory structures like magnetic field and geometry
- ▶ However, even with **AthenaMP** our memory consumption is very high (4GB for full reconstruction + 1.0-1.5GB per worker process)
 - ▶ We barely fit into current grid resources!
- ▶ Multi-threading shares memory much more efficiently than multi-process
 - ▶ Not only solves a current problem, but prepares us for different architectures with less memory per core
- ▶ Writing multi-threaded code is difficult!
 - ▶ So to reduce difficulty for developers, we rely on new multi threaded framework, **AthenaMT**



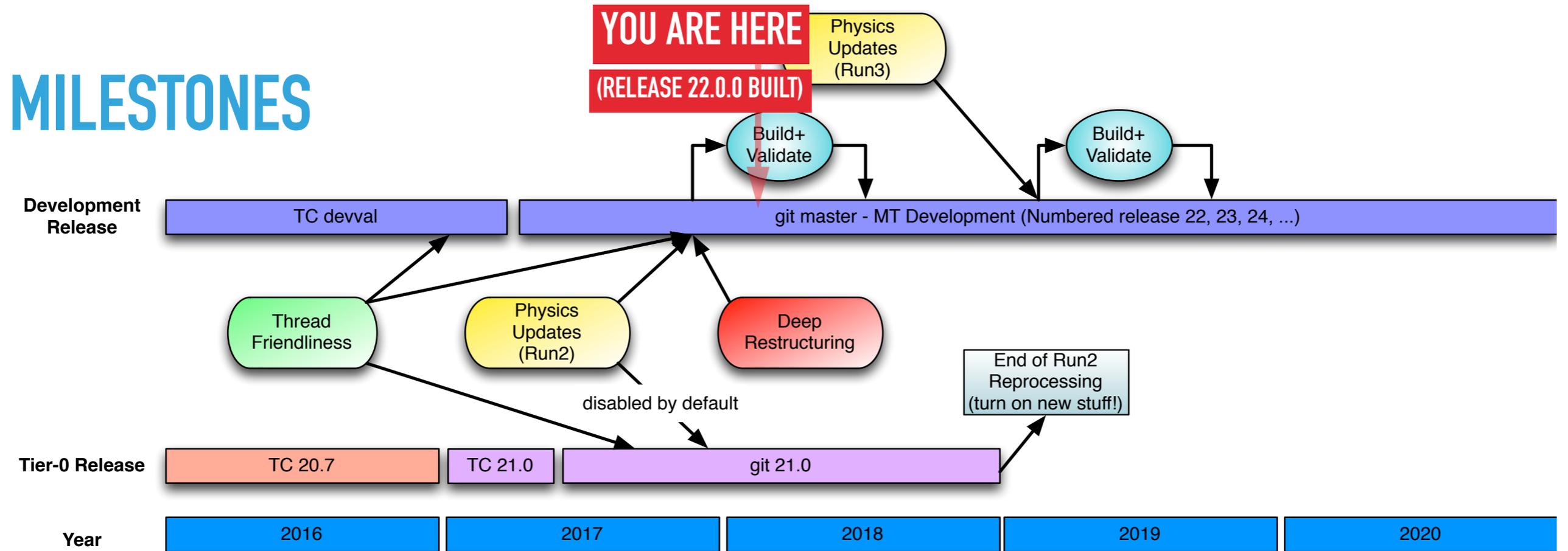
Roughly, view each row as a thread, each colour as an event, each box as an event processing step. A 'scheduler' determines what is run and where, trying to utilise CPU and memory resources as optimally as possible.

ATLAS FLOP PERFORMANCE OVER 5 YEARS OF INTEL CPUS

- ▶ Fraction of the potential floating point performance we use has been dropping over time
- ▶ CPU manufacturers add wider vectors that we do not take advantage of, or deep pipelines where cache misses are very costly
- ▶ Confirms what we have long suspected about the growing performance gap on modern architectures



MILESTONES



- ▶ **Q4 2017:** Finish migration to DataHandles. **MOSTLY DONE**
- ▶ **Q4 2017, Trigger M2:** Few e/gamma, Jet & Muon chains HLT workflow fully integrated into scheduler. Single thread
- ▶ **Q1 2018:** Start integration of ACTS into Athena **REVIEW STARTING SOON**
- ▶ **Q2 2018:** Replace all callbacks by conditions algorithms. That will also kill a number of public AlgTools
- ▶ **Q2 2018:** NSW prototype reconstruction running
- ▶ **Q3 2018:** First physics validation of e.g. ACTS integration
- ▶ **Q3 2018:** Get rid of remaining public tools by making them private or convert them to services.
- ▶ **Q4 2018:** Making AlgTools cost-(correct), implies killing any local cache
- ▶ **Q4 2018:** Finish migration of tracking code (+ clients) to ACTS

- ▶ **Q4 2018:** Make Services fully thread safe
- ▶ **Q4 2018:** Make ByteStream converters thread-friendly
 - ▶ **Q4 2018, Trigger M3:** Concurrent data access demonstrated. First integration with online system. Tested in online partition (single thread).
- ▶ **Q1 2019:** Cleaning out remaining thread-hostile constructs: Usage of incidents, local caches in Algorithms
- ▶ **Q2 2019:** Start physics validation comparing release 22 MT vs release 22 ST vs release 21
 - ▶ **Q4 2019, Trigger M4:** Algorithms and their tools migrated (reentrant or clonable), tested in online partition, multiple threads
- ▶ **Q3 2019:** MT compliant DataQuality Monitoring
- ▶ **Through 2020:** Large-scale validation and final bug-fixes and optimisation
- ▶ **Q1 2021:** Start of Run 3

ACTS REVIEW

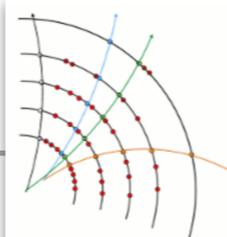
- ▶ ACTS is track reconstruction software, based on ATLAS's, but made detector agnostic and multithreaded
- ▶ Using ACTS for run-3 will hopefully save ATLAS a lot of development effort, so we have a set up a review board.
- ▶ Aim is for the review to be complete by the end of March

▶ Mandate

Migrating ATLAS's current tracking software to work in AthenaMT would be a significant challenge. The ACTS review board will instead evaluate the possibility of using ACTS as an alternative to such a rewrite.

The board should report on the appropriateness of ACTS's design for use within ATLAS, and in particular, of using it in conjunction with the existing tracking software. Any perceived deficiencies should be carefully detailed.

Plans for migration should be discussed, including required validation campaigns and the compatibility of the ACTS migration with the R22 (and Run-3) schedules.



A COMMON TRACKING SOFTWARE PROJECT

WELCOME TO THE ACTS PROJECT!

The ACTS project is a free, open-source software project for track reconstruction in high-energy physics experiments. It is designed as high-level toolkit which is independent of the detector design and the magnetic field configuration. By using modern C++ standards and emphasizing thread-safety during the development process, the resulting software is usable in multi-threaded applications. Data structures are designed to facilitate vectorisation whenever possible which leads to superior computing performance. Special care is taken to ensure that the software is not only useful for academic purposes but also applicable to real-life experiments (e.g. allow hooks for re-calibration of measurements during the track fit procedure).

The modular design allows the user to pick any needed components without being forced to use the tracking geometry description or the event data model of ACTS. The toolset includes

- a surface-based description of the tracking geometry (including converters from most common detector description languages),
- a slim event data model for track parameters and measurements,

RELEASES

- latest development version
[Open issues](#) | [git repository](#) | [Documentation](#)
- Version 0.04.01
[Release Notes](#) | [Download](#) | [Documentation](#)
30 May 2017 14:57 CEST.
- Version 0.04.00
[Release Notes](#) | [Download](#) | [Documentation](#)
9 May 2017 15:15 CEST.
- Version 0.03.02
[Release Notes](#) | [Download](#) | [Documentation](#)
24 Mar 2017 14:40 CET.
- Version 0.03.01
[Release Notes](#) | [Download](#) | [Documentation](#)
28 Nov 2016 14:26 CET.
- Version 0.03.00
[Release Notes](#) | [Download](#) | [Documentation](#)
11 Nov 2016 17:25 CET.
- Version 0.02.01
[Release Notes](#) | [Download](#) | [Documentation](#)

<http://acts.web.cern.ch/ACTS/>

▶ Membership

Chair:	Edward Moyses
Trigger:	Stewart Martin-Haugh
Reco :	Mark Hodgkinson and Goetz Gaycken
Muon:	Will Leight and Jochen Meyer
ID:	Nick Styles and Matthias Danninger
Core:	Scott Synder and Vakho Tsulaia
Egamma:	Anthony Morley and Christos Anastopoulos
Physics:	Bill Murray

RECONSTRUCTION

- ▶ Regular reports on progress in reconstruction [meetings](#)
- ▶ Good progress with first milestone (almost all code is migrated to data handles)
 - ▶ Important step, since we can now really start to use the scheduler in earnest (key component of AthenaMT)
 - ▶ The offline reconstruction group has been working closely with the trigger, and this cross-domain effort has been absolutely critical to the success so far
- ▶ Much less has been done for the next milestone, migration to condition handles
 - ▶ Will be concentrating on this shortly
- ▶ We are still very far away from running in a true multi-threaded environment, and some intimidating challenges remain
- ▶ Some areas of the software are a significant concern, due to lack of developers

TRIGGER

- ▶ Trigger is also migrating to AthenaMT, and will use offline algorithms seamlessly
 - ▶ Migration more difficult than for offline, because need to remove existing steering and wrappers
- ▶ Significant progress made:
 - ▶ TrigUpdateTests running in master
 - ▶ Portions of ID, calo, muon code migrated, whilst jets are work in progress
- ▶ However 2018 is a very important year, with critical milestones coming
 - ▶ Continuing close integration with offline reconstruction will be vital
- ▶ Trigger menus (~50k lines of python) need to be completely migrated, and trigger database to be streamlined
- ▶ For more details, please see Stewart's [talk](#) in a recent reconstruction meeting

UPGRADE SOFTWARE

- ▶ Ongoing effort to integrate upgrade software into our nightlies (which can be difficult, as often it is developed under tight timelines, outside of normal software oversight):
 - ▶ For example, significant work to integrate the ITK release (20.20.X) into 21.9. Not yet complete, and in the meantime the 20.20.X branches need to be maintained, which requires effort from overworked Software Infrastructure Team (SIT)
 - ▶ Appointment of Andi Salzburger as Upgrade Software Coordinator is really helping, as is the productive and positive attitude from Upgrade communities
- ▶ Major concern for run-3 is New Small Wheel (being developed in 21.3):
 - ▶ A tremendous amount of effort has been made in software and the situation is much improved
 - ▶ Some of this effort has come from Muon SW ... which has implications for the Muon SW MT migration
 - ▶ **However** many challenges remain and more people are needed to ensure success - we are still far from having functional reconstruction and complete simulation
- ▶ The possibility of only installing one wheel is extremely worrying - this could require substantial effort for the software, and even evaluating how much effort is required would be a significant task:
 - ▶ *Very real risk that preparing for all possible scenarios will mean that we are ready for none*

SIMULATION

(Slight digression away from MT)

Fast Simulation

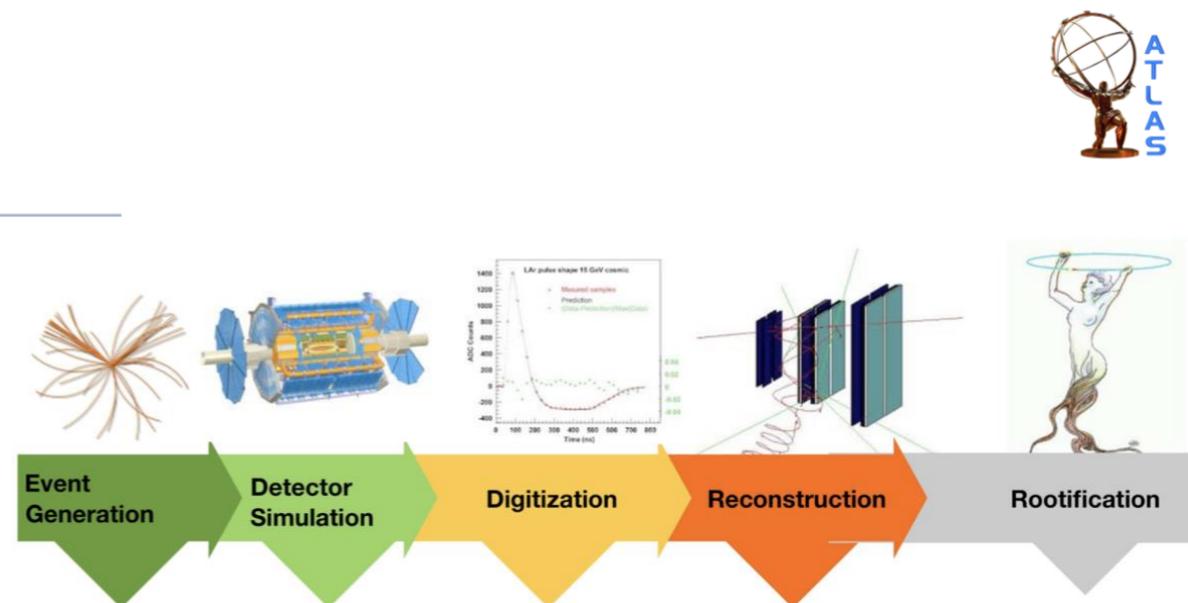
Overview and motivation

Most physics and performance studies require very **large datasets of simulated events**

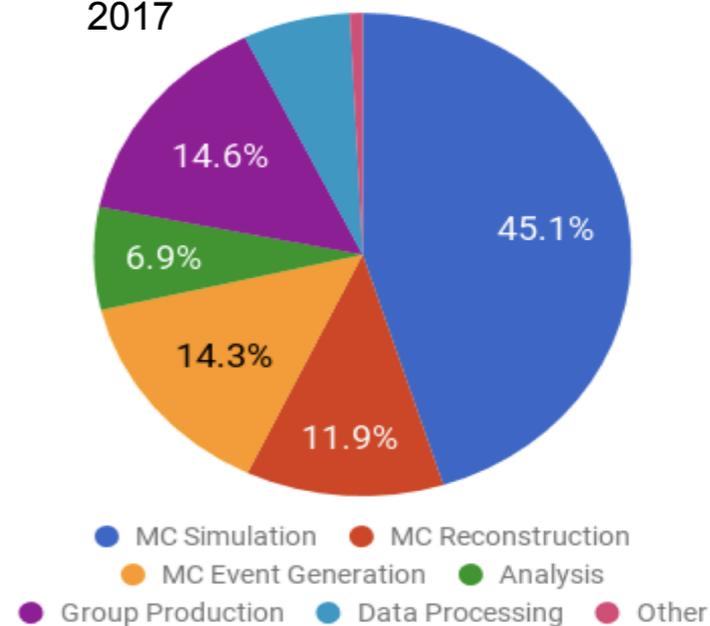
- Production in **Geant4** (~30m volumes) is **highly CPU-intensive** → $O(15\text{min})$ / event
- **Increasing luminosity** poses serious challenge to produce **sufficient Monte Carlo statistics**

A **highly-detailed detector** description is **not** always **necessary**

- Signal MC samples, alternative backgrounds for systematics
- Detector upgrade studies



Wall Clock consumption per workflow 2017



RECENT SIMULATION DEVELOPMENTS IN REL 21

- ▶ **Geant4** speed optimisation has been ongoing, so this is likely to improve the CPU requirements for full simulation.
- ▶ **FastCaloSim** Reduced CPU, potentially larger memory requirements and no change to output file size
- ▶ **FastChain**
 - ▶ Further reduced CPU, memory requirements not assessed yet, likely no change to output size in first iteration
 - ▶ This could be improved by not storing intermediate formats, but requires further discussion.

SIMULATION / DIGI / OVERLAY – MT DEVELOPMENTS

- ▶ **AtlasG4** - Running well in 22.0.0 - now need to do larger scale tests, and investigating performance on different Intel architectures at NERSC
- ▶ **ISF** - Now need to concentrate on migrating Integrated Simulation Framework (default for current production) to be multi-threaded
 - ▶ Significant work to do, however key developer is leaving, and even after replacements are found, the resultant total FTE will be lower.
- ▶ **Digitisation** - very challenging technical problems to solve, such as the design of a multi-event store gate. Critically low on effort.
- ▶ **Data/MC overlay** - being developed in 21.0 and swept into 22.X, but depends upon digitisation efforts (so is lower in the queue)
- ▶ For more details, please look at John Chapman's [slides](#) in a recent software coordination meeting.

AVAILABLE EFFORT

- ▶ In many key areas we are dangerously short of experts
 - ▶ Rely on a very few key people, who we absolutely cannot afford to lose
- ▶ We had very interesting discussion with various project leaders about improving the situation
 - ▶ In some areas, we have seen new people join (but of course they need to be trained up, and then encouraged to stay)
- ▶ Please don't forget that our expensive detector relies on well-written software to function optimally:
 - ▶ We cannot rely solely on qualification tasks and/or students.
 - ▶ We need experienced people who will stay and guide the work and newcomers.
 - ▶ Inefficient use of our computing resources has significant real world costs (and impact on quality of our physics)

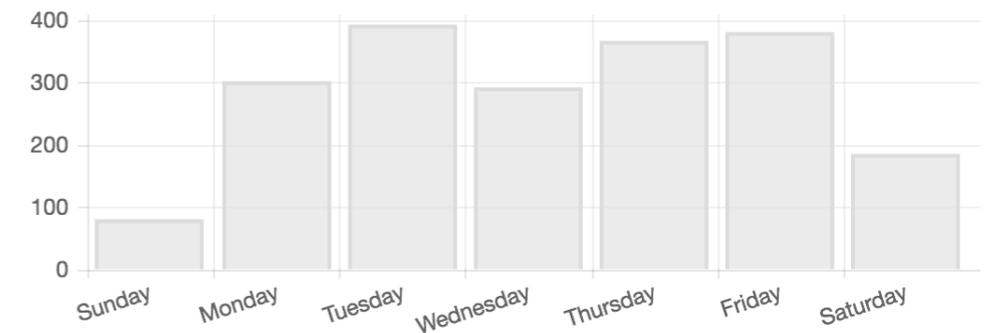
Commit statistics for master Nov 20 - Feb 22

(release 22.0.0)

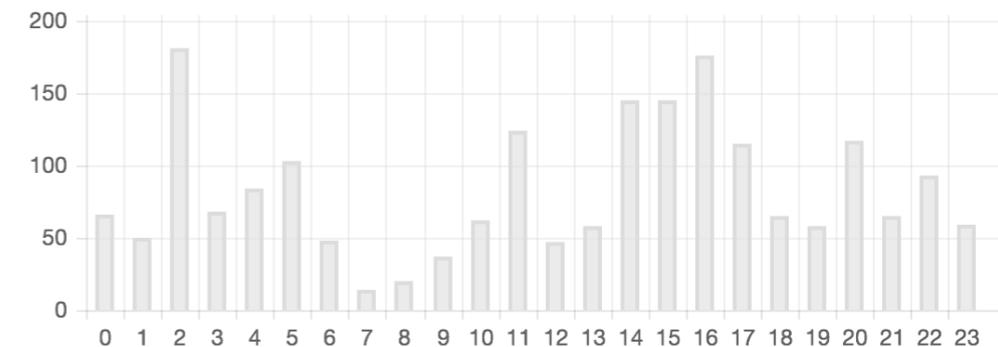
- Total: **2000 commits**
- Average per day: **21 commits**
- Authors **125**

~3% of collaboration

Commits per weekday



Commits per day hour (UTC)



<https://gitlab.cern.ch/atlas/athena/graphs/master/charts>

SUMMARY

- ▶ We have a challenging but vital programme of work for run-3
 - ▶ And are still working on optimisations for run-2
- ▶ Personpower is very tight, but through the hard work of a small team we are still on track...
 - ▶ However we are very vulnerable to slippage: we cannot afford to lose any more experts, and we cannot have too many unforeseen problems
 - ▶ Many tasks (such as integration of ACTS) the difficulty/FTE required is hard/impossible to quantify
 - ▶ We have had some injection of new people, but more is needed (and the more expert the better)
- ▶ [Offline](#) and [trigger](#) hackathons in this week - please feel free to join!

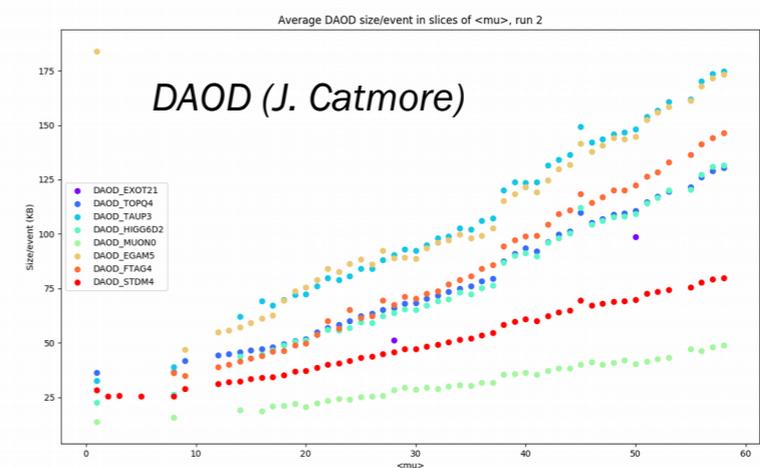
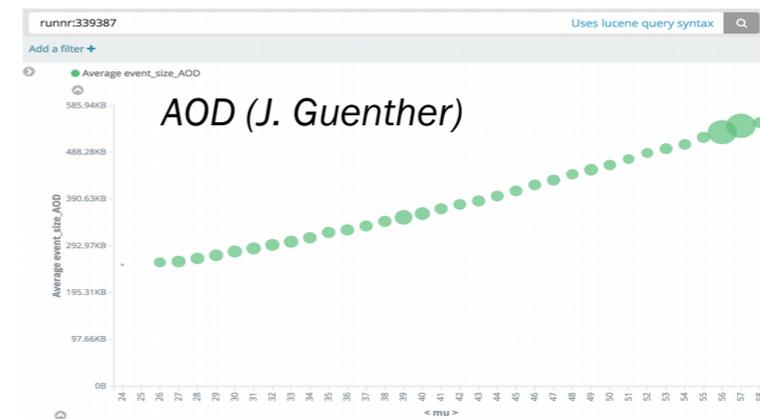


BACKUP

DP TALK, LAST ATLAS WEEK

Offline Disk & MC

- High μ data with a relaxed trigger menu gives a higher event rate and larger individual events
 - mean rate for fill > 1300 Hz;
AOD > 500 kB/evt
 - need to understand if (in particular) DAOD event size can be controlled
- Effects on MC still being checked
 - digi+reco increase reduces CPU available for other MC tasks slightly



FAST SIM TIMELINES

► Busy year for fast sim!

FastCalSimV2

Timeline



Input sample production without FCal -- ongoing (February)

Produce parameterisations -- 1000s of them (March)

First physics validations, CP (egamma, tau, JetMET) feedback; FCal production (April)

Long(er)-term

More physics validations, incorporate CP feedback, first tunings (Summer)

[Even more] validations, including forward physics (Autumn)

Large-scale production of samples for physics analyses (Winter)



Fast Simulation (ATLAS Week)

vrp

10

Fast Chain

Status and timeline



New EDM migration, HMPL (late February/early March)

- A lot of progress being made here
 - MIG13 (20.20) → 21.3
 - RTTs getting better day-by-day
- Ready within the month

Pseudotracking (late February/early March)

- Dependent on HMPL EDM changes
- Validate following EDM migration

FATRAS (March)

- Showing reasonable agreement; still affected by memory leaks
 - Repeat validation
 - Investigate material effects to improve q/pT bias (not a show-stopper)

Fast Si Digitisation (March)

- Tuning of configuration parameters ongoing; good accuracy in local tests

Fast TRT Digitisation (March)

- Good for hard scatter; some PU distributions look OK, but further tuning needed

Pileup-OTF in Pythia8

- Emulation of multiple BCID support (out-of-time PU); still in progress

ART

- Architectural differences → non-reproducibility
- Implement DCube

Full Chain

Begin validation of complete set of components (April)

Large-volume test sample production (May/June)

CP tuning, calibrations, and physics group feedback (Summer)

More details on recent work in backup slides.

Fast Simulation (ATLAS Week)

vrp

15

CONDITIONS DB

- ▶ CERN IT will maintain COOL, our conditions DB until end of LS2
- ▶ Current implementation has scalability issues, likely related to poor cachability of requests
- ▶ After review, conclusion is that ATLAS will have to in-house COOL/CORAL, so we can support it for run-3
- ▶ Should (hopefully) implement CREST like cachability improvements
- ▶ Person power is needed
- ▶ See [talk](#) from Richard for more details