The new Analysis Model for ATLAS

James Catmore
University of Oslo, Norway

What is ATLAS changing for Run-2?

- New tracking software model
  - including replacement of CLHEP with Eigen
  - integration of the new inner pixel layer (IBL) into the software
- Integrated simulation framework
- New data placement system and model
- New bulk production system
- New analysis model
  - including novel data format
- Memory usage optimisation
- Core software overhaul
- … and many other things

Why all of these changes?

BANQUE NATIONALE SUISSE
BANCA NAZIONALE SVIZZERA
20
Vingt Francs
Venti Franchi
05E8097967
04J1139611
04C5755783
100
Hundert Franken
Tschient Francs
99K6267433
50
Cinquante Francs
Cinquanta Franchi

RESOURCES

Very tight budgets for computing in Run-2
… but a lot more data to process, with higher pile-up

⇒ have to be much smarter about how we use our computing and human resources

SPEED

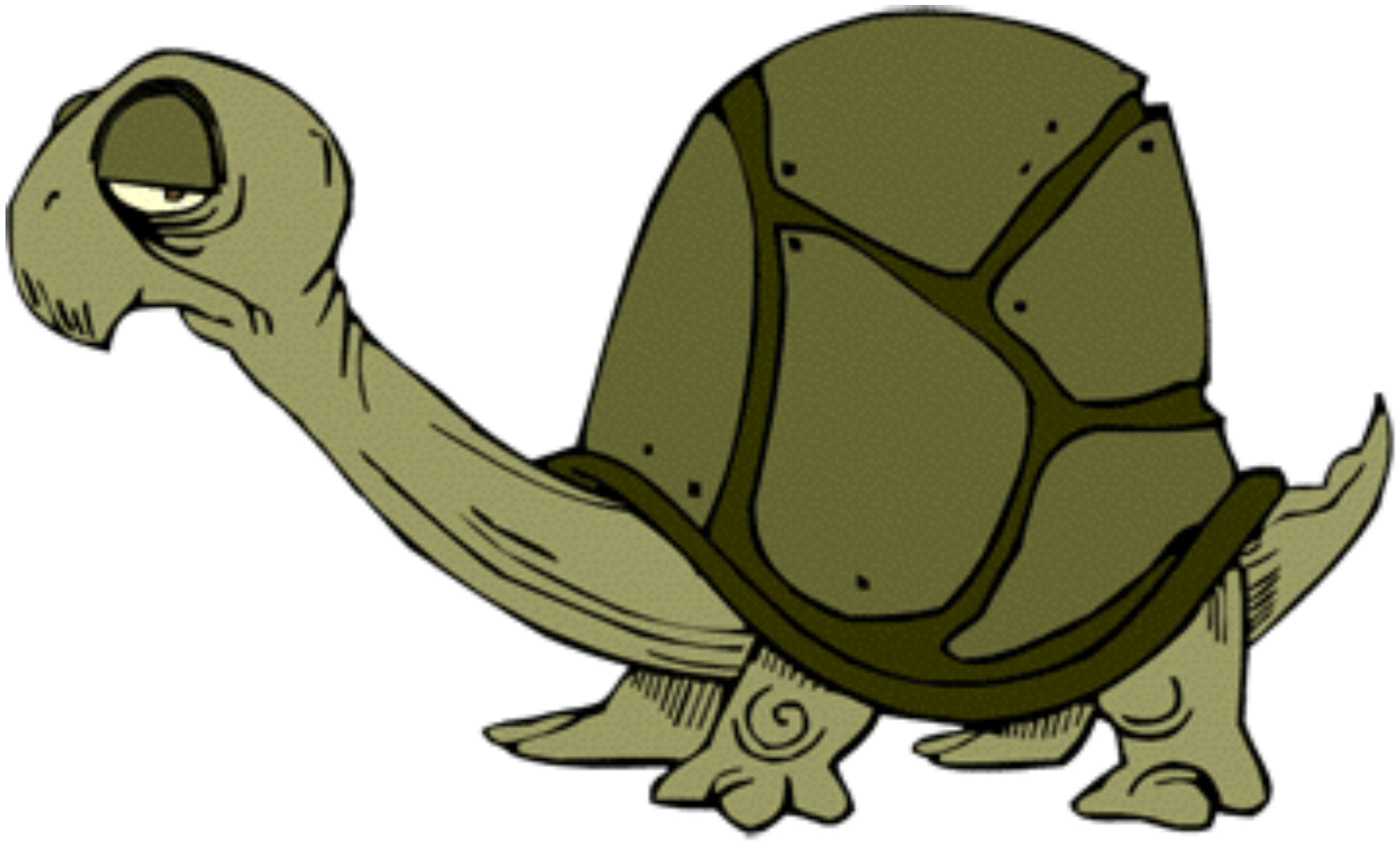
…. of reconstruction

… of simulation

… of distribution

… of access

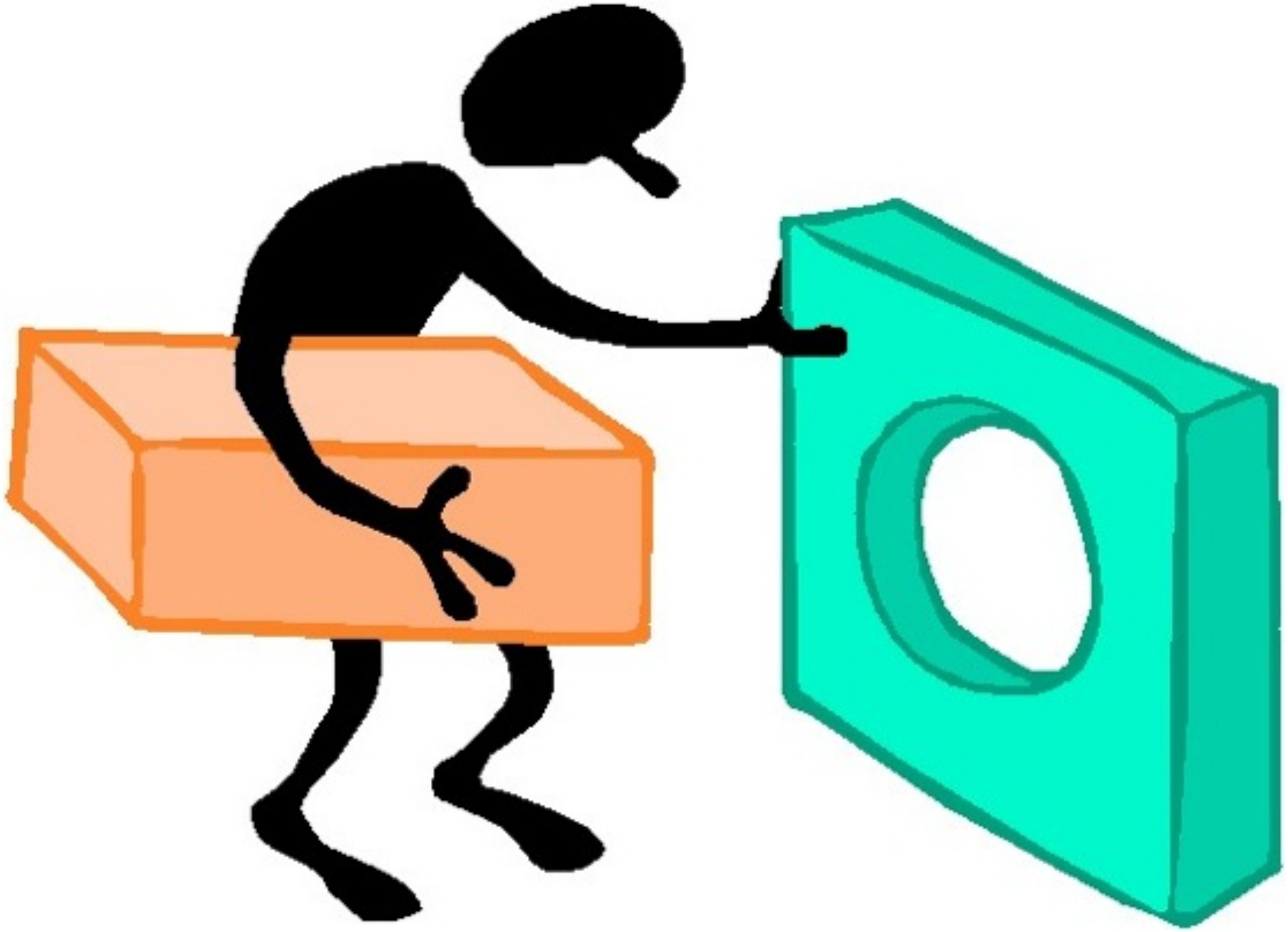
… of analysis



needs to be *significantly* faster than in Run-1

COMPATIBILITY

…of software used by different working groups, institutes, users



needs to be *much* better than in Run-1

What is ATLAS changing for Run-2?

- New tracking software model
  - including replacement of CLHEP with Eigen
  - integration of the IBL into the software

  } Speeding up reconstructon

- Integrated simulation framework } Speeding up simulation
- New data placement system and model } More efficient use of disk
- New interface to the Grid } Making job submission less laborious and more reliable
- New analysis model
  - including novel data format

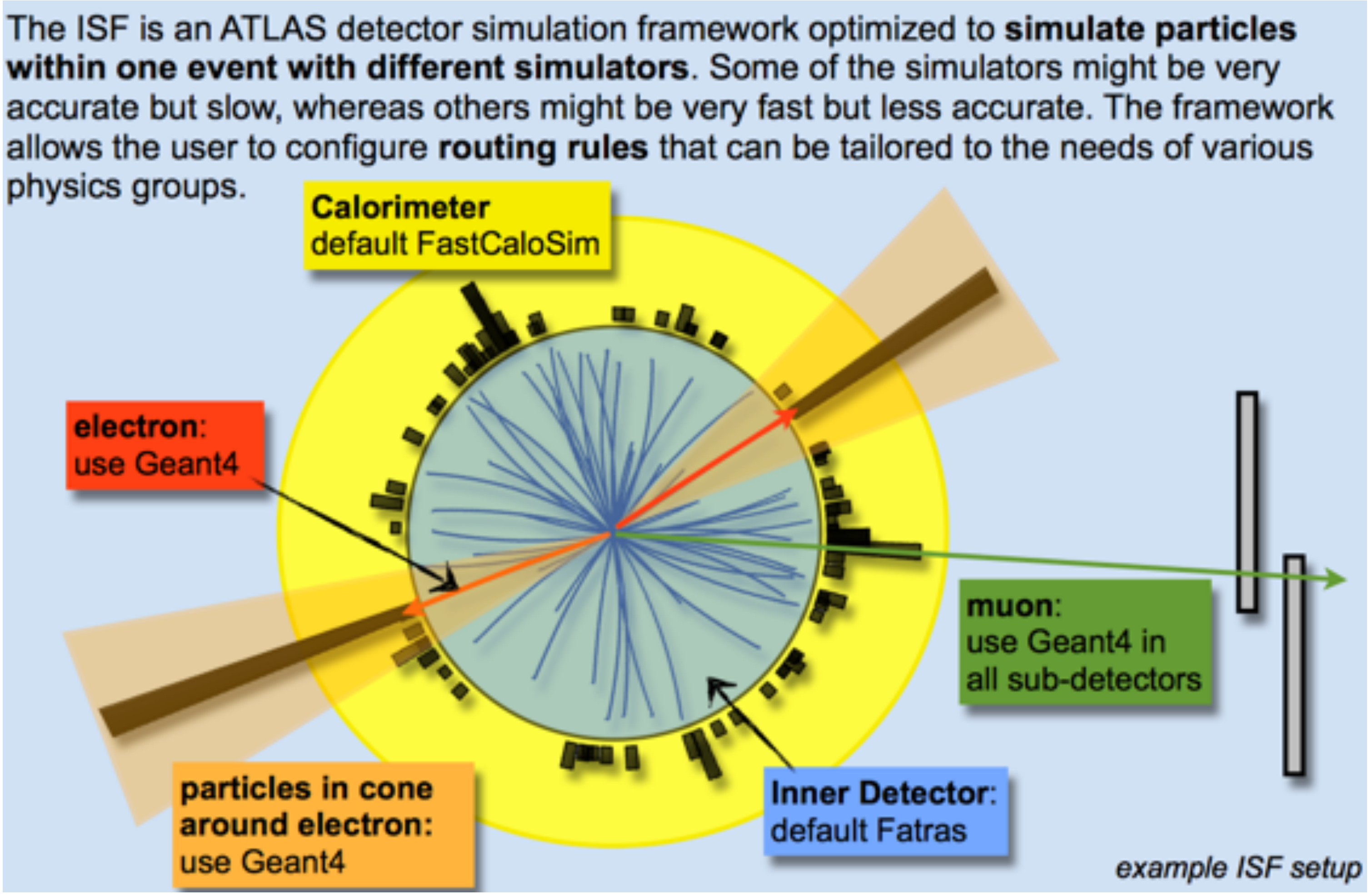
  } Saving disk, CPU, analysis time
Making a common analysis EDM/ framework

- Memory usage optimisation
- Core software overhaul
- … and many other things

Integrated Simulation Framework

In Run-1 ATLAS used homogeneous simulation, either "full" or "fast"

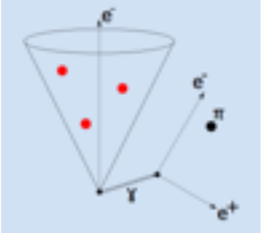
The ISF, to be deployed in Run-2, allows different simulation chains for different parts of the *same event.*

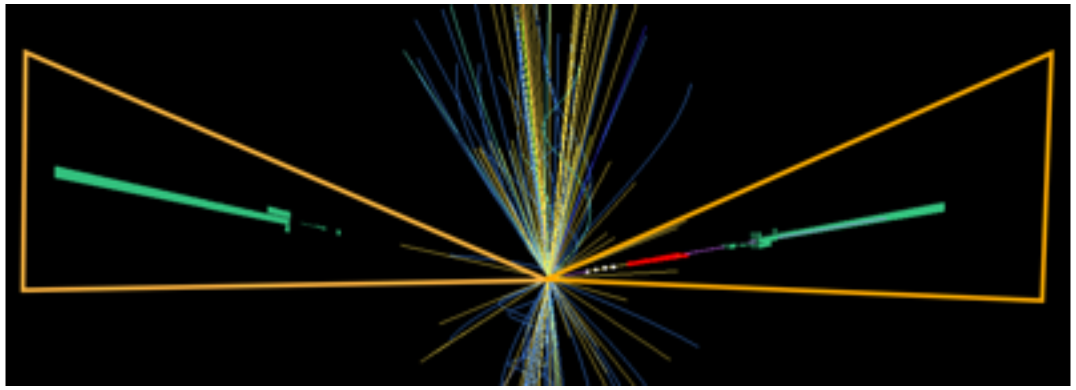


The ISF is an ATLAS detector simulation framework optimized to **simulate particles within one event with different simulators**. Some of the simulators might be very accurate but slow, whereas others might be very fast but less accurate. The framework allows the user to configure **routing rules** that can be tailored to the needs of various physics groups.

Diagrams from E. Ritch

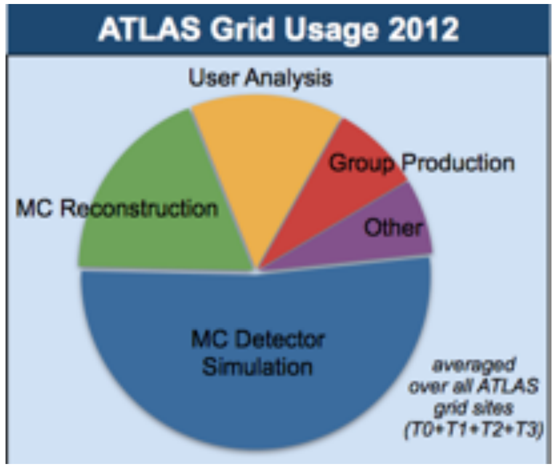
Possible to restrict simulation to certain particle types or regions around certain particle types





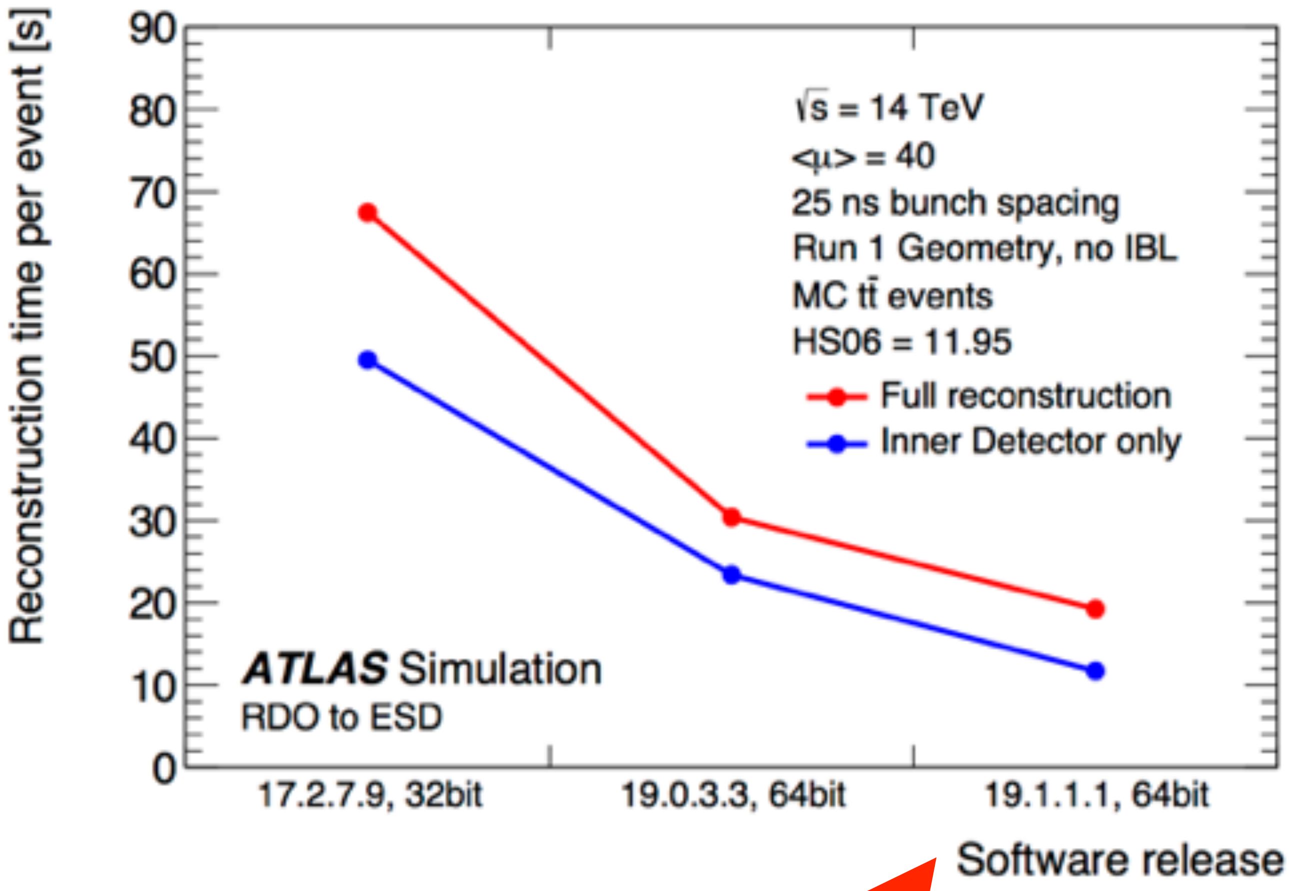
Combining these features, for certain signal MC samples it is possible to obtain speed-up factors of 2 orders of magnitude or more

This will have a significant impact on our CPU consumption overall



Reconstruction improvements

- Linear algebra libraries switched from CLHEP to Eigen
- Very significant simplification of the tracking software model
- New tracking strategies and cuts optimised for 14 TeV data taking

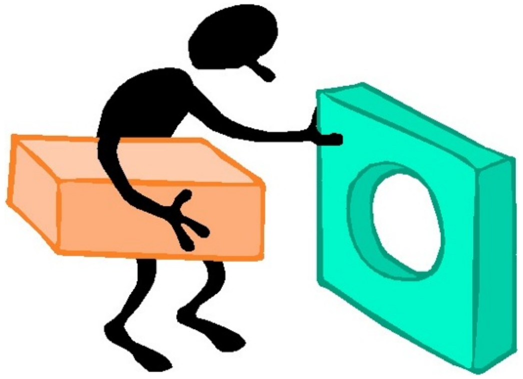


Factor of 3 faster!

and maybe we can do more before 2015…
compiler optimisation, vectorisation?

New analysis model: what was wrong with the old one?

Output of reconstruction not readable in ROOT

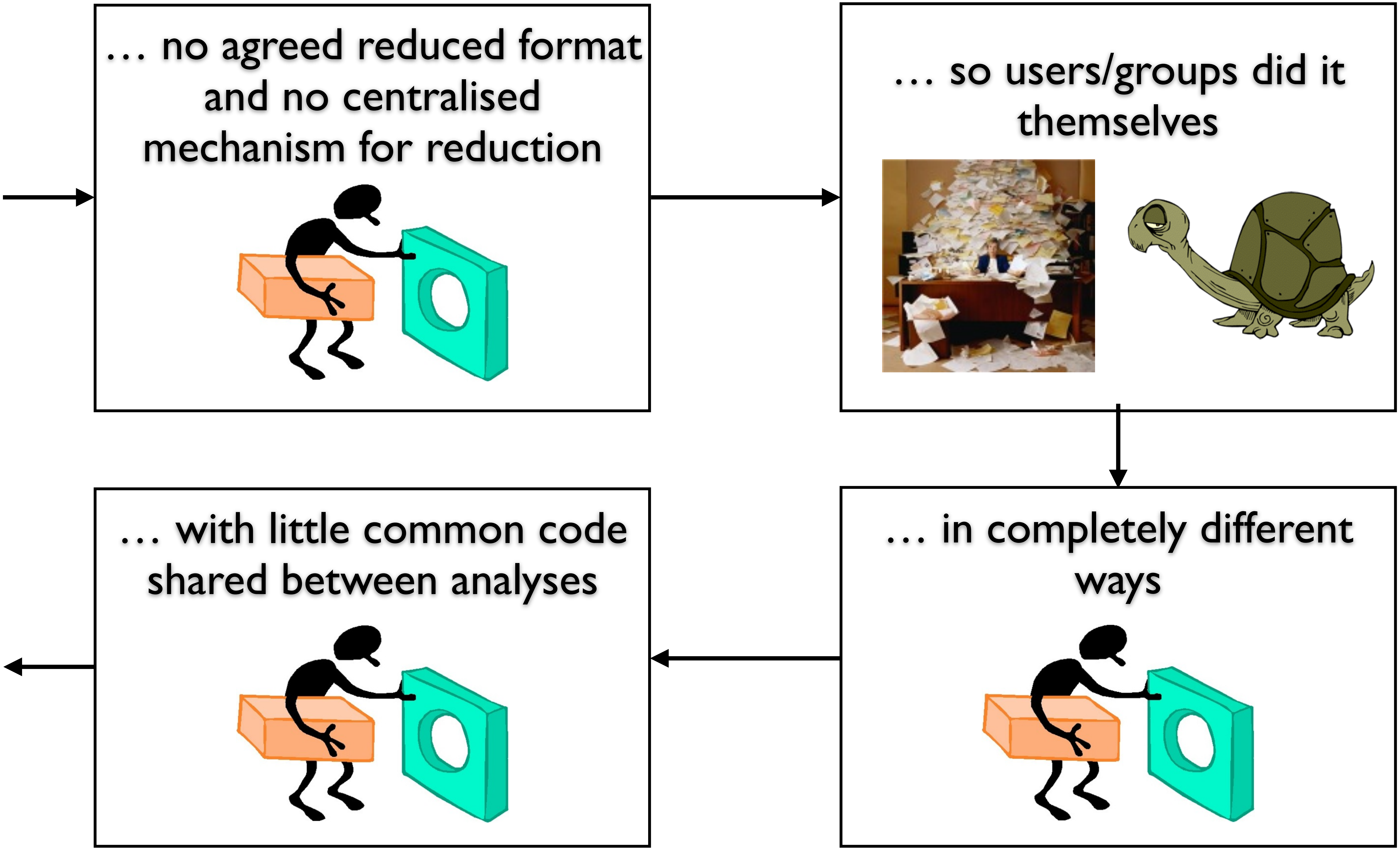


→ Massive format conversion needed

↓ ...but this was too big for users to analyse

← … so data reduction needed

New analysis model: what was wrong with the old one?



… no agreed reduced format and no centralised mechanism for reduction

… so users/groups did it themselves

… in completely different ways

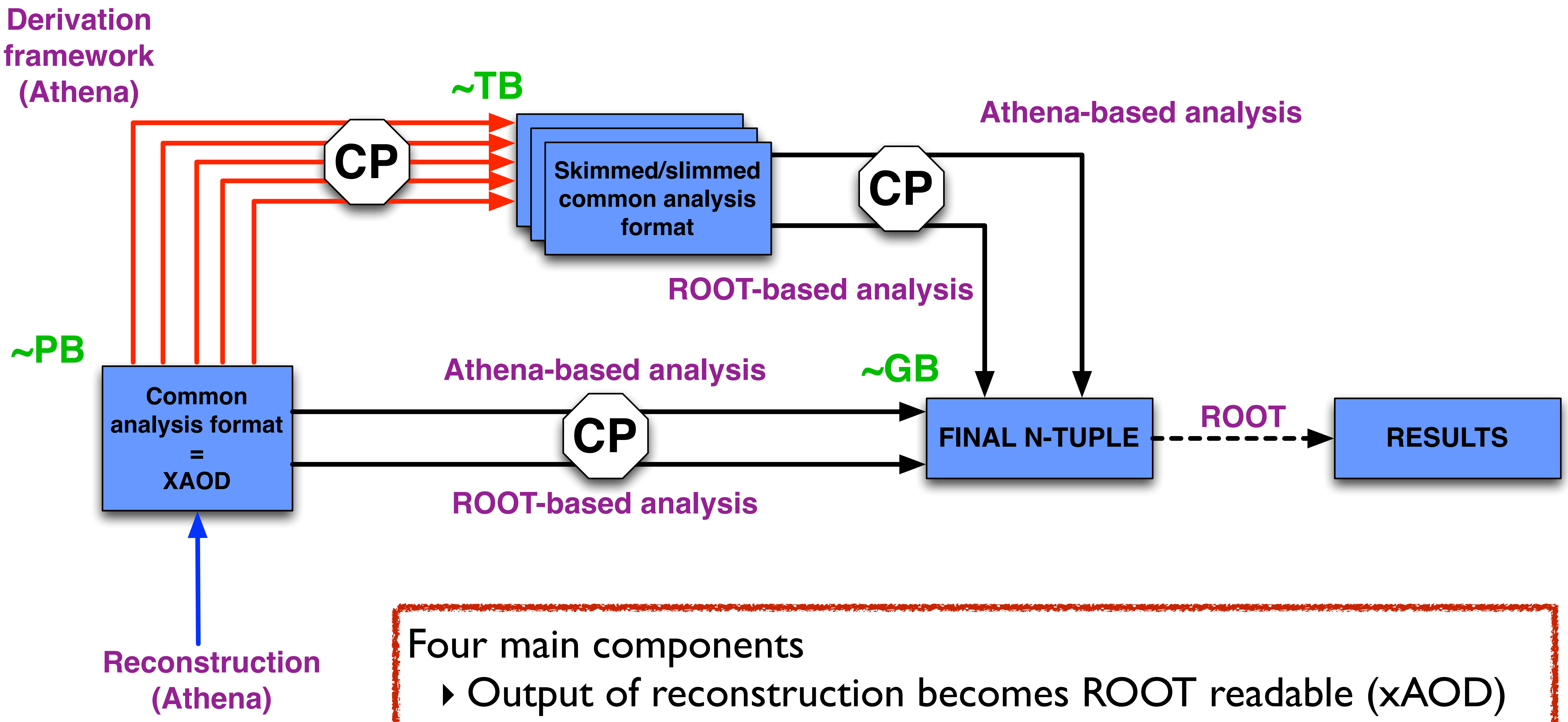
… with little common code shared between analyses

New analysis model: what was wrong with the old one?

- Results
  - Same data written out over and over again
  - Long delays waiting for format conversion to n-tuples
  - Users having to baby-sit production tasks for weeks on end
  - Plethora of incompatible analysis frameworks and event data models
  - No common mechanism for applying systematics

⇒ CHAOS

New analysis model



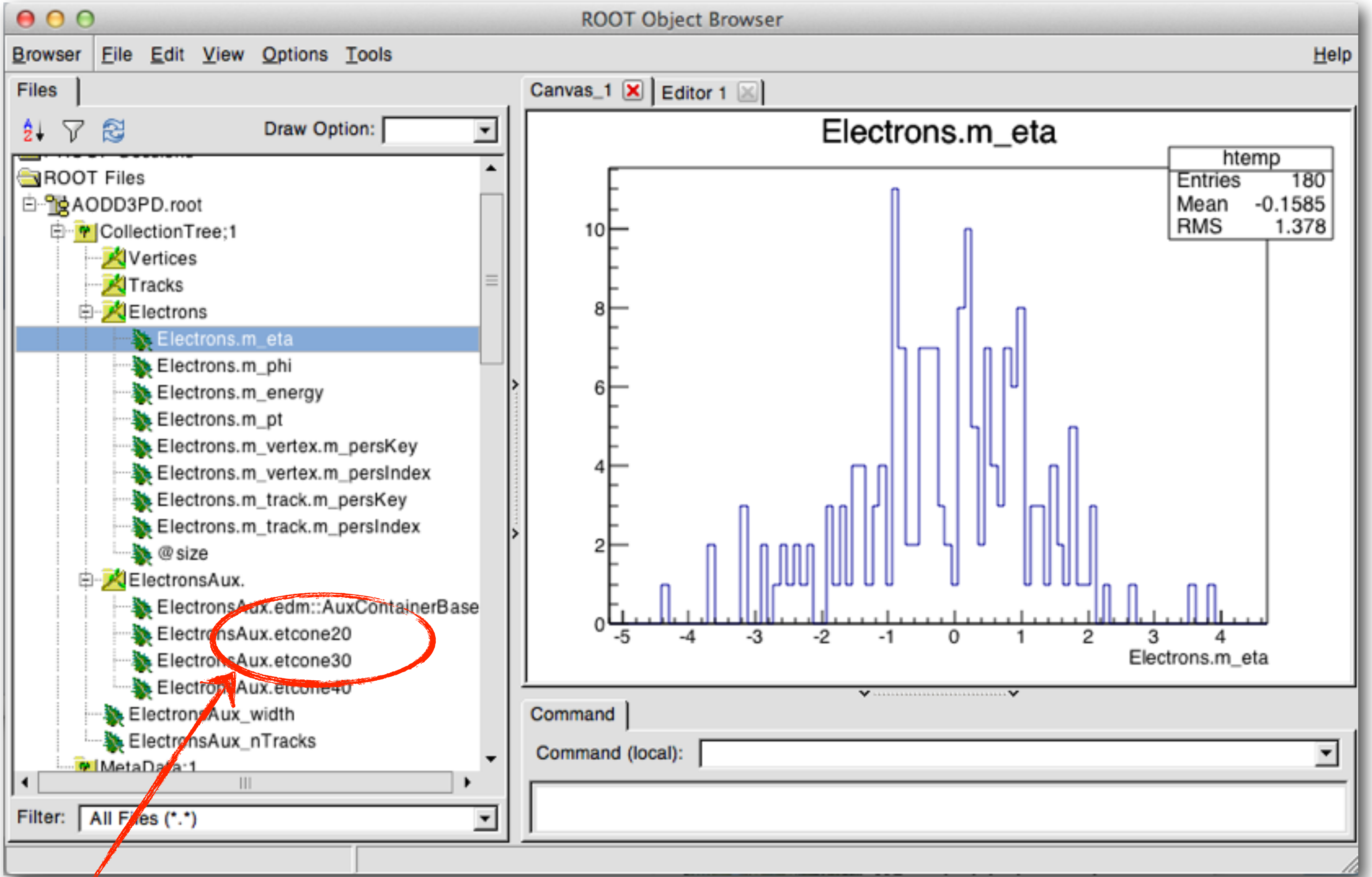
Four main components

- Output of reconstruction becomes ROOT readable (xAOD)
- Most analysis done on small, centrally produced *derivations* of the full data
- Common EDM and framework for user analysis
- Regular updating of xAOD with new conditions

The xAOD format

- Output of reconstruction immediately readable in ROOT
  - No need for format conversion; ROOT analysis code can use it as-is
- Includes re-designed event data model (EDM) - much simpler and more transparent than the old EDM
- Design concept
  - Interfaces to data objects (tracks, muons etc) and their actual payload of information are *split*
  - Payload is held in an *auxiliary store* but the user code only interacts with the *interface*
  - The auxiliary store can be interpreted directly by ROOT
  - Like an n-tuple it supports *partial reading* of an event
  - Auxiliary store contains two kinds of variable
    - *Static*, which are explicitly declared in the class
    - *Dynamic*, which are created on-the-fly

The xAOD format



- ROOT Files
  - AODD3PD.root
    - CollectionTree;1
      - Vertices
      - Tracks
      - Electrons
        - Electrons.m_eta
        - Electrons.m_phi
        - Electrons.m_energy
        - Electrons.m_pt
        - Electrons.m_vertex.m_persKey
        - Electrons.m_vertex.m_persIndex
        - Electrons.m_track.m_persKey
        - Electrons.m_track.m_persIndex
        - @size
      - ElectronsAux.
        - ElectronsAux.xAOD::AuxContainerBase
        - ElectronsAux.etcone20
        - ElectronsAux.etcone30
        - ElectronsAux.etcone40
      - ElectronsAux_width
      - ElectronsAux_nTracks
    - MetaData;1

Properties held by auxiliary branches

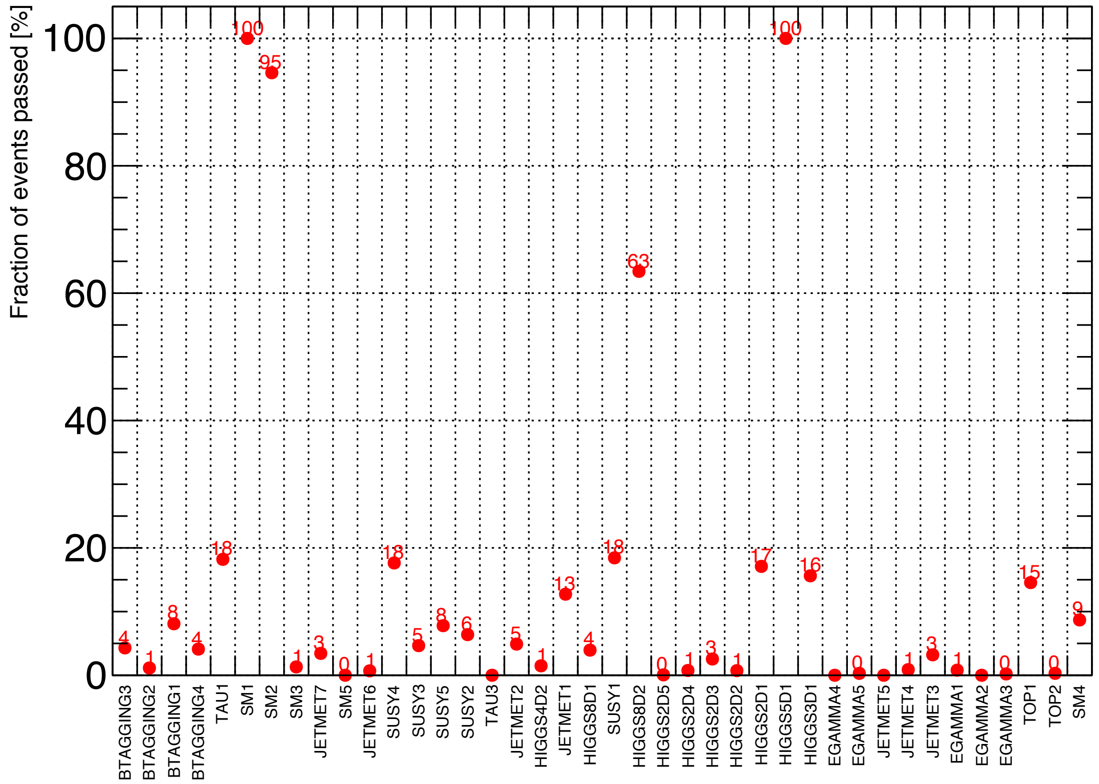
The Derivation Framework

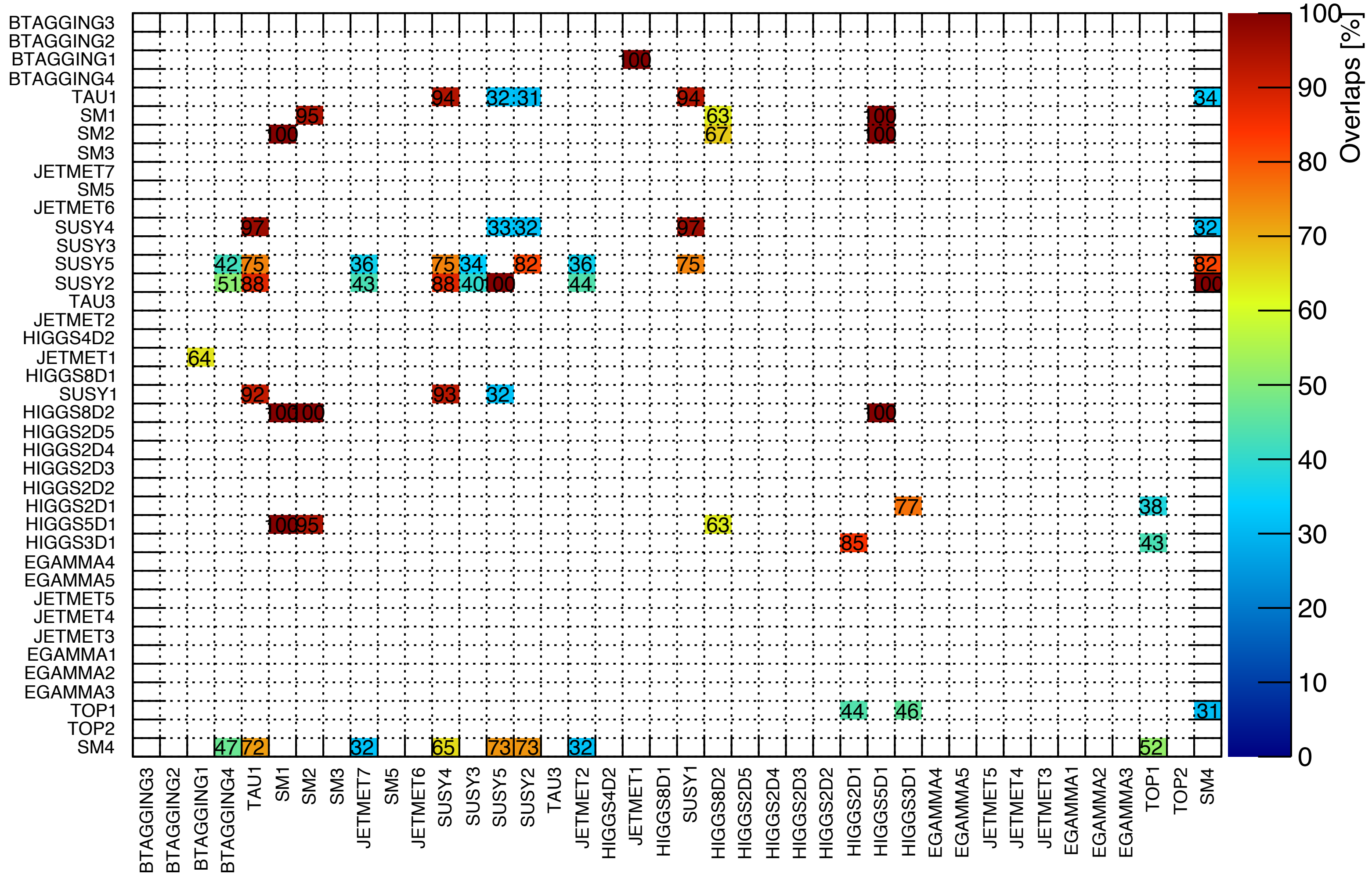
- Centrally run software for reducing PB-sized datasets down to TB for user analysis
- Benchmark: dataset should be small enough for a user to process a derivation in ~1 day with normal privileges, or be able to subscribe it to a Tier-3
  - Should be around 1% of the full xAOD size
  - "Derivations" are analysis-specific, and we foresee ~100 in total by 2015
  - Many derivations can be produced simultaneously (
- Runs four kinds of operation
  - Skimming (removing events)
  - Slimming (removing certain information from all events)
  - Thinning (removing whole objects)
  - Augmentation (adding extra information)

Derivation type	Implemented formats
Slimmed only	TOPQ1, STDM1, HIGG5D1, JETM1, EXOT{2,3,9}
Trigger-based skims	SUSY1, TAUP1, EXOT1, JETM{4,5}, SUSY4
Single (e,μ,τ) skim	STDM4, HIGG8D2, SUSY2, SUSY3, SUSY5
Single (e,μ) + τ skim	TAUP3, HIGG4D1
Di-lepton (e,μ) skim	STDM3, HIGG3D1, HIGG4D2, EGAM{1-4}, HIGG2D{1,3,4}, TOPQ2
Tri-lepton (e,μ) skim	STDM5
Quad-lepton (e,μ) skim	HIGG2D2
Di-lepton (e,μ)+γ skim	HIGG2D5
Single e/γ skim	STDM2, EXOT6
Di-photon skim	HIGG8D1
W→eν skim	EGAM5
W+jet skim	JETM2
Z+jet skim	JETM3
Single jet skim	EXOT{5,7}
Di-jet skim	EXOT8
Lepton (e,μ) + jet skim	JETM{6,7}

The Derivation Framework

Tools available to monitor event/content overlap and skim/size fractions





Should enable us to merge derivations which are very similar

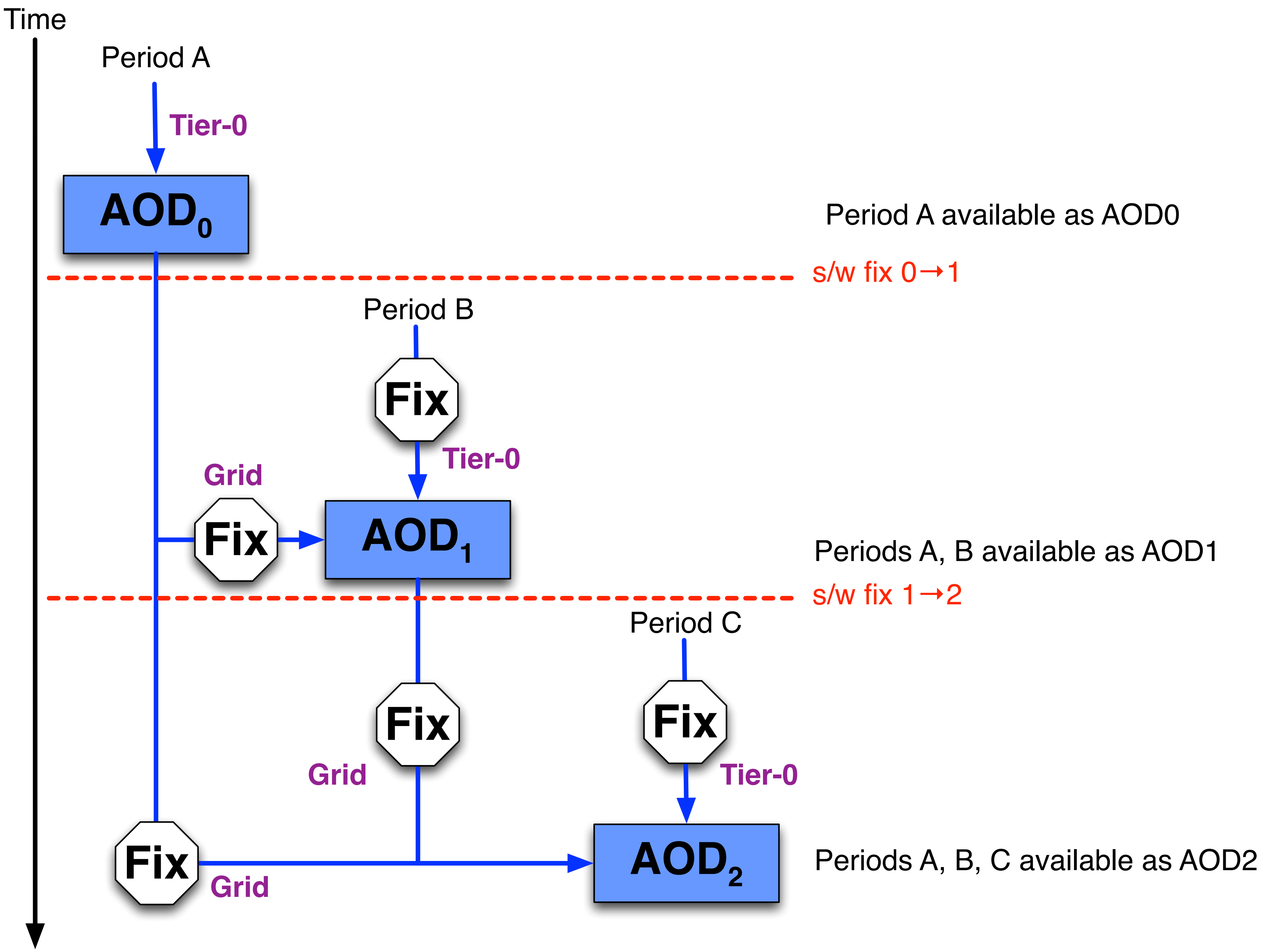
Common analysis EDM and framework

- By adding a few small libraries it is possible to access the xAOD EDM in ROOT
- Means that physics analysis can use *the same objects* that are used in reconstruction
- Means that *the same tools* can be used across ATLAS for applying calibrations and systematics
- The analysis framework (RootCore) previously available for analysing n-tuples has been ported to xAOD and significantly improved
  - Analysts now have the choice of using the full software framework (Athena) or RootCore - but using *the same code*

Allows code re-used, collaboration between groups, avoiding proliferation of DIY analysis EDMs and frameworks

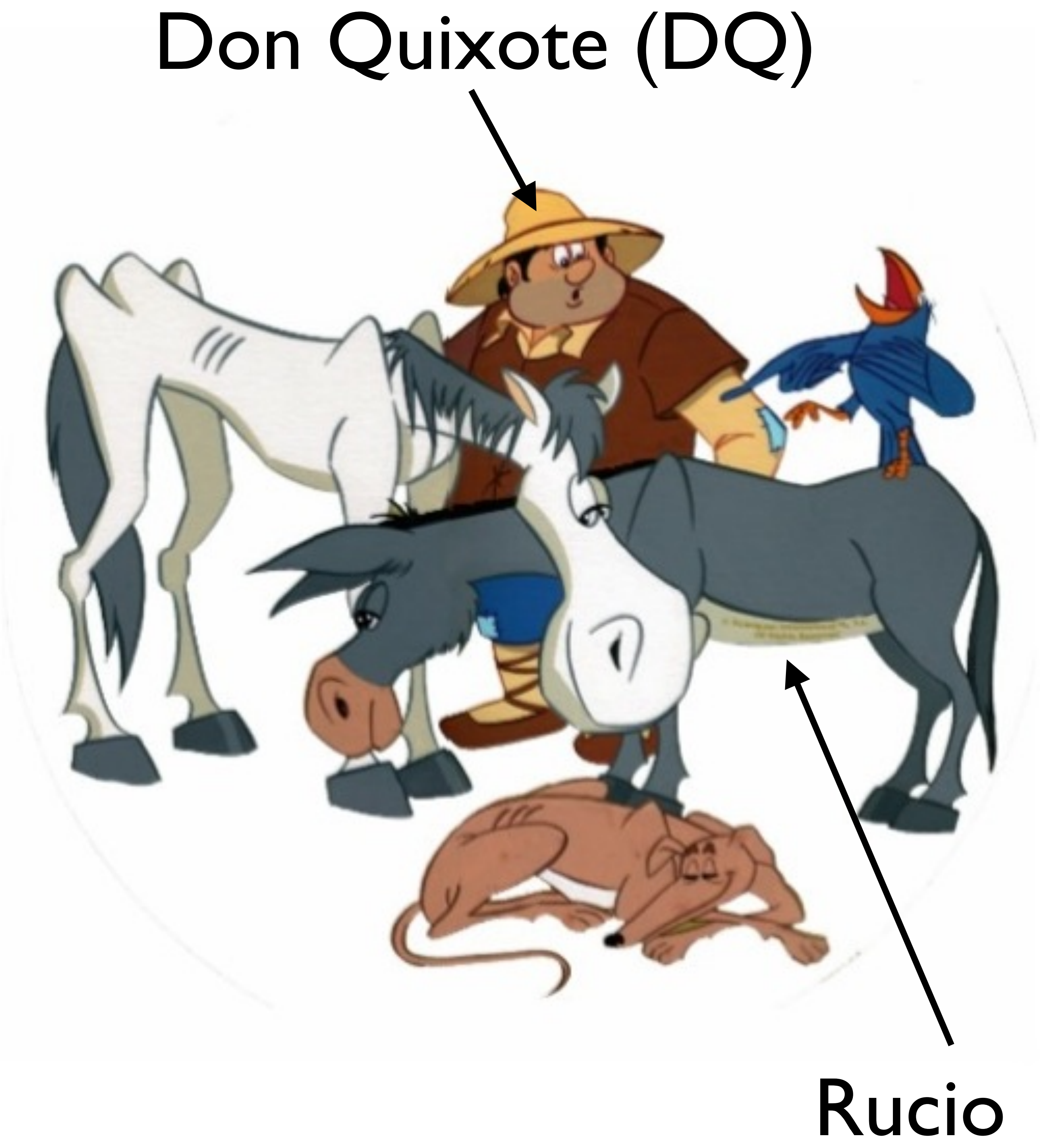
xAOD-to-xAOD reprocessing

- Idea: initially, modifications to the calibrations in the prompt reconstruction are applied in the derivations
- At some point the full xAOD should be remade with the corrections applied
    - In most cases this can run from the older xAOD
    - So we follow the pattern of pushing computation upwards: from users to derivations, and from derivations to reconstruction



New data management system

- Current data management system: *DQ2*
  - This has performed extremely well in Run-1, but it is difficult to extend and is operationally burdensome
  - Replaced with a new DDM called *Rucio*

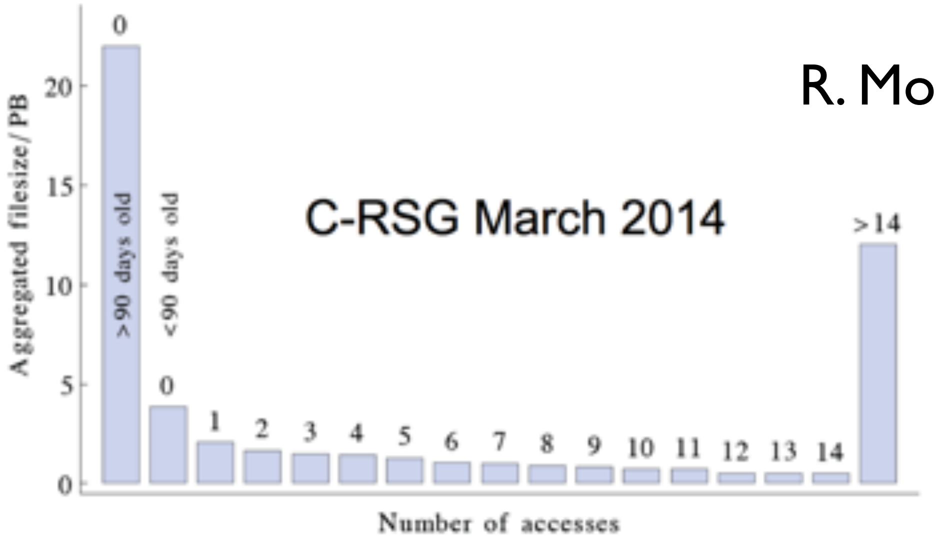


New data management system

- Main features of Rucio
  - Basic unit is the *file* rather than the *dataset*. Files are then grouped into datasets. A file can be shared amongst many datasets. Allows flexible scoping of files across different domains
  - Introduces searchable *metadata* on files/datasets
  - Introduces *automatic replica management*
    - Define replication rules (e.g. two replicas of dataset X on any T1 sites, etc)
    - Allows quota and lifetime rules to be enforced automatically
- These features will reduce the workload on ATLAS Distributed Computing personnel and group space managers who currently have to do much of the replica management manually

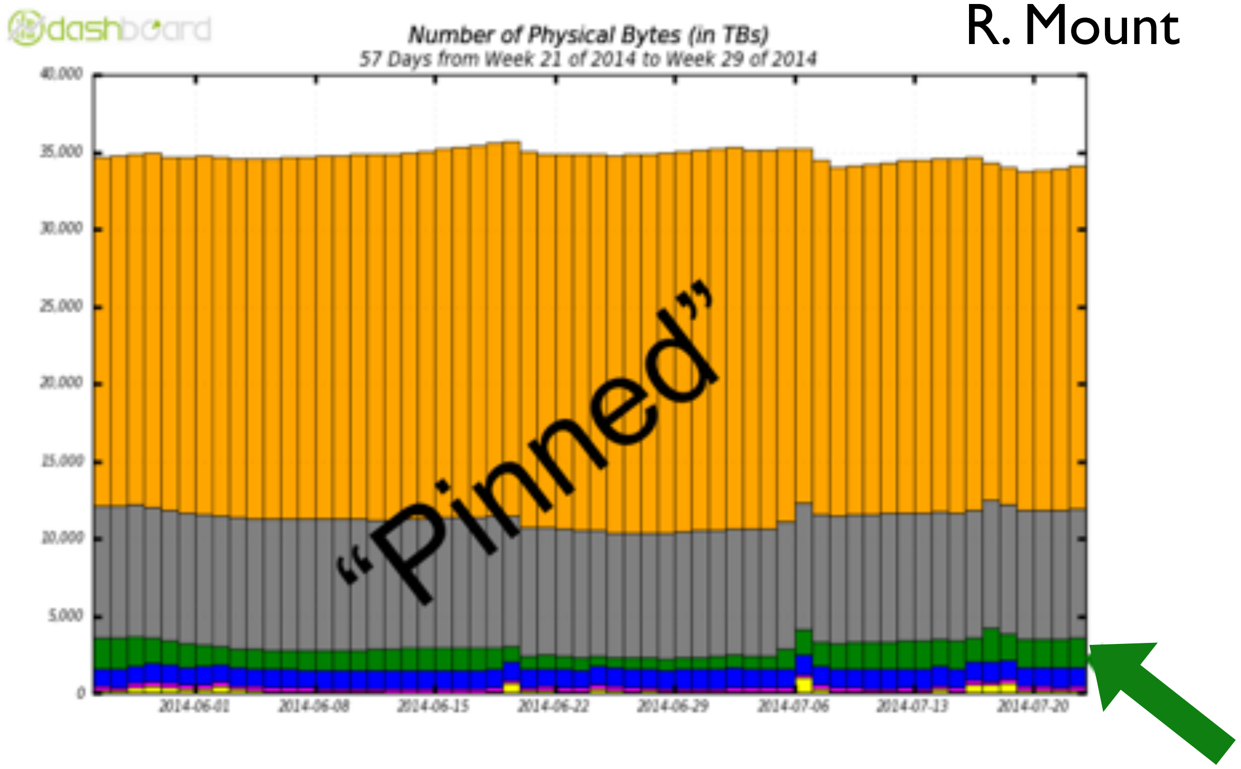
New data placement and lifetime policy

- Interrogation of the ATLAS data access patterns reveals that much of the data is never used
    - e.g. 26PB on DATADISK (T1+T2) had not been touched in the last 90 days, period ending 14th March 2014
    - 8PB never used at all
- Mostly there due to de facto “if in doubt, keep it” policy
- Causes acute problems at T1 sites, where pinned data blocks the disks from admitting new data
    - Seems to cause a roughly twice-annual panic when we realise we don’t have enough space to do new production



R. Mount

Figure 1 ATLAS DATADISK: volumes of data versus number of accesses in the 90 days ending 14 March 2014. Data created in the last 90 days but not accessed are in the second bin. The total volume of all DATADISK



R. Mount

Available for new data

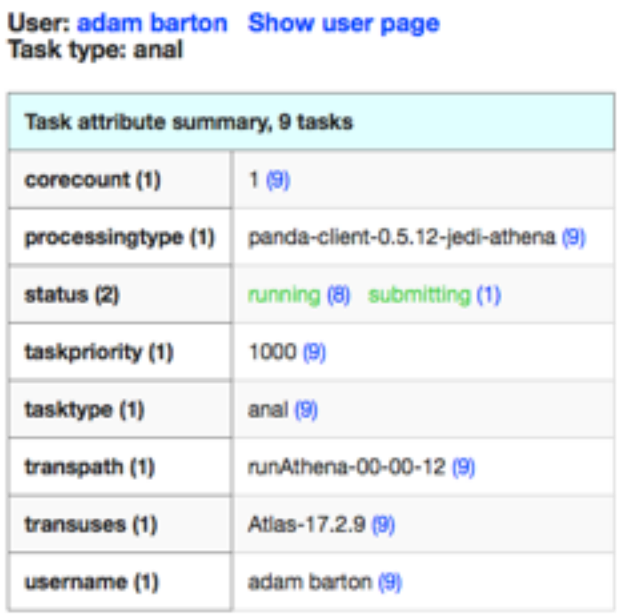
New data placement and lifetime policy

- New policy: *all data will have a lifetime*
  - Access to data resets the clock
  - Lifetimes vary for different kinds of data - set by relevant coordinators
  - After the lifetime expires, data deleted from disk/tape
- New policy: *all data will have a disk residency priority*
  - Calculated algorithmically using (e.g.) access patterns, predictions of future access, time left until final deletion, manual overrides, etc
  - When disk space is needed datasets with the lowest priority go to tape if they are within their lifetime, until there is enough space
- All of the above is made possible by the advances in the DDM
- Automatic replica creation/destruction based on popularity will continue as currently
- Aim to replace the current replication policy (hundreds of lines, many exceptions etc) with a much simpler, automatically executed policy

New interface to the Grid

- ATLAS Workload management system PanDA upgraded to BigPanDA
    - http://atlascloud.org:8080/pandawms/
    - Generalisation of PanDA to allow use beyond ATLAS
    - Most visible change to users: monitoring
        - New monitoring site: http://bigpanda.cern.ch/
- New submission interface: JEDI (Job Execution and Definition Interface) by which users and the central production team submit jobs
    - User analysis still done via the familiar tools GANGA and pAthena - they just use JEDI as their back-end
    - Central bulk production done directly with JEDI
    - "Jobs" become less important whilst "tasks" becomes the central concept: a task may contain many jobs, some of which may be killed and re-tried; the user should focus on the status of the task
    - *Scouting* now to be deployed for user analysis, so if a user makes a typo, the first few scouts will pick it up and prevent mass submission of jobs that are doomed to fail

New interface to the Grid

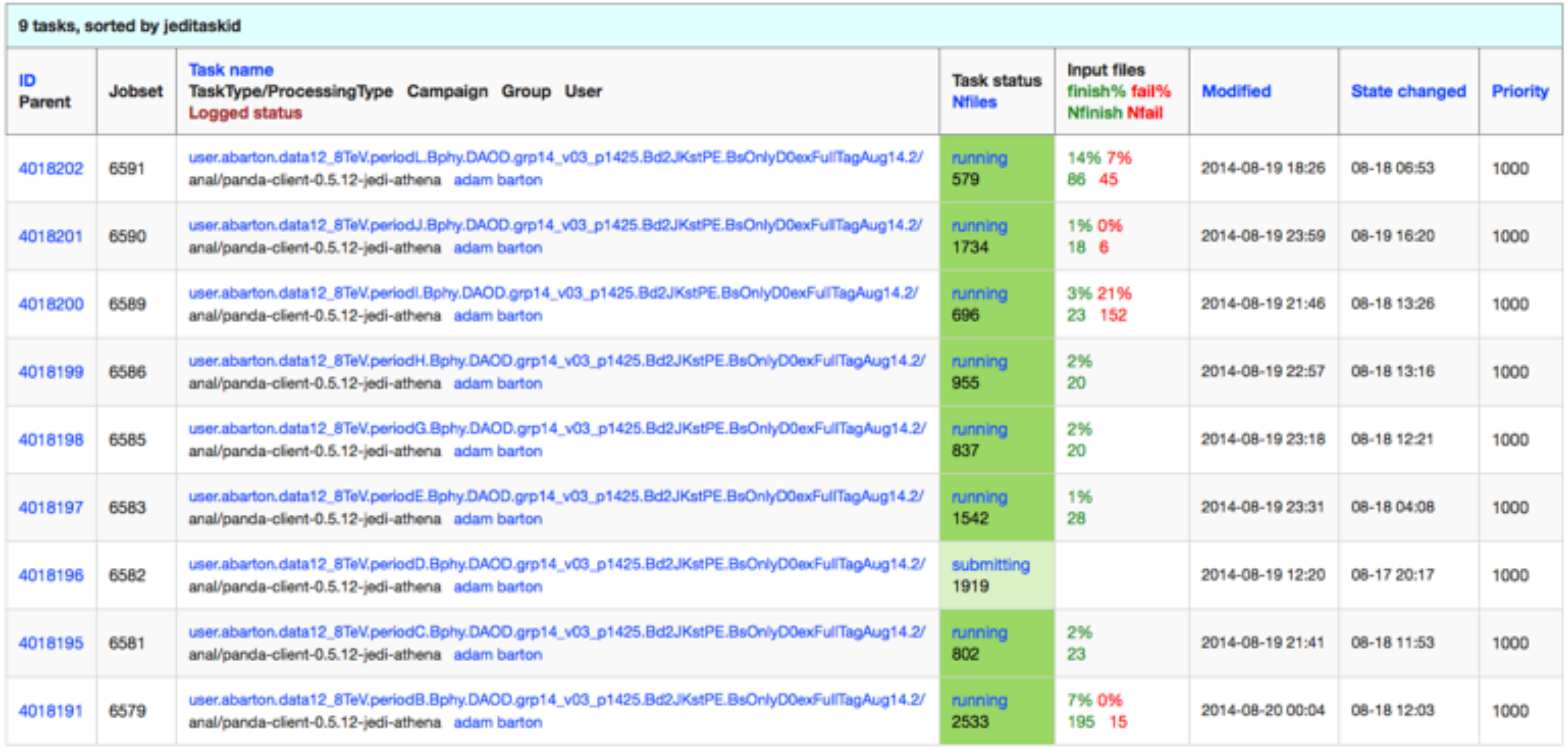


User: adam barton   Show user page
Task type: anal

Task attribute summary, 9 tasks	
corecount (1)	1 (9)
processingtype (1)	panda-client-0.5.12-jedi-athena (9)
status (2)	running (8) submitting (1)
taskpriority (1)	1000 (9)
tasktype (1)	anal (9)
transpath (1)	runAthena-00-00-12 (9)
transuses (1)	Atlas-17.2.9 (9)
username (1)	adam barton (9)

BigPanDA interface for
analysis tasks of a single user

Please contact atlas-support-cloud-uk@cern.ch with suggestions, improvements etc



9 tasks, sorted by jeditaskid

ID Parent	Jobset	Task name TaskType/ProcessingType Campaign Group User Logged status	Task status Nfiles	Input files finish% fail% Nfinish Nfail	Modified	State changed	Priority
4018202	6591	user.abarton.data12_8TeV.periodL.Bphy.DAOD.grp14_v03_p1425.Bd2JKstPE.BsOnlyD0exFullTagAug14.2/ anal/panda-client-0.5.12-jedi-athena adam barton	running 579	14% 7% 86 45	2014-08-19 18:26	08-18 06:53	1000
4018201	6590	user.abarton.data12_8TeV.periodJ.Bphy.DAOD.grp14_v03_p1425.Bd2JKstPE.BsOnlyD0exFullTagAug14.2/ anal/panda-client-0.5.12-jedi-athena adam barton	running 1734	1% 0% 18 6	2014-08-19 23:59	08-19 16:20	1000
4018200	6589	user.abarton.data12_8TeV.periodI.Bphy.DAOD.grp14_v03_p1425.Bd2JKstPE.BsOnlyD0exFullTagAug14.2/ anal/panda-client-0.5.12-jedi-athena adam barton	running 696	3% 21% 23 152	2014-08-19 21:46	08-18 13:26	1000
4018199	6586	user.abarton.data12_8TeV.periodH.Bphy.DAOD.grp14_v03_p1425.Bd2JKstPE.BsOnlyD0exFullTagAug14.2/ anal/panda-client-0.5.12-jedi-athena adam barton	running 955	2% 20	2014-08-19 22:57	08-18 13:16	1000
4018198	6585	user.abarton.data12_8TeV.periodG.Bphy.DAOD.grp14_v03_p1425.Bd2JKstPE.BsOnlyD0exFullTagAug14.2/ anal/panda-client-0.5.12-jedi-athena adam barton	running 837	2% 20	2014-08-19 23:18	08-18 12:21	1000
4018197	6583	user.abarton.data12_8TeV.periodE.Bphy.DAOD.grp14_v03_p1425.Bd2JKstPE.BsOnlyD0exFullTagAug14.2/ anal/panda-client-0.5.12-jedi-athena adam barton	running 1542	1% 28	2014-08-19 23:31	08-18 04:08	1000
4018196	6582	user.abarton.data12_8TeV.periodD.Bphy.DAOD.grp14_v03_p1425.Bd2JKstPE.BsOnlyD0exFullTagAug14.2/ anal/panda-client-0.5.12-jedi-athena adam barton	submitting 1919		2014-08-19 12:20	08-17 20:17	1000
4018195	6581	user.abarton.data12_8TeV.periodC.Bphy.DAOD.grp14_v03_p1425.Bd2JKstPE.BsOnlyD0exFullTagAug14.2/ anal/panda-client-0.5.12-jedi-athena adam barton	running 802	2% 23	2014-08-19 21:41	08-18 11:53	1000
4018191	6579	user.abarton.data12_8TeV.periodB.Bphy.DAOD.grp14_v03_p1425.Bd2JKstPE.BsOnlyD0exFullTagAug14.2/ anal/panda-client-0.5.12-jedi-athena adam barton	running 2533	7% 0% 195 15	2014-08-20 00:04	08-18 12:03	1000

Main questions for 2015 (in my opinion)

- User adoption
  - Will people be willing to re-write their analyses to use the new framework?
  - Will physics groups be willing to use the ISF widely with “aggressively” fast simulation?
- Derivations
  - Will we be able to make the derivations both small and also useful?
  - How well will MC fit in the model?
  - Will it be fast/flexible enough to satisfy the user community?
  - How will validation of these formats be done?
- xAOD-to-xAOD reprocessing
  - How often will this happen? How long will a campaign take to plan, validate and execute?
- Placement, deletion, task definition/submission: few worries about this

Main questions for 2015 (in my opinion)

- User adoption
    - Will people be willing to re-write their analyses to use the new framework?
    - Will physics groups be willing to use the ISF widely with "aggressively" fast simulation?
- Derivations
    - Will we be able to make the derivations both small and also useful?
    - How well will MC fit in the model?
    - Will it be fast/flexible enough to satisfy the user community?
    - How will validation of these formats be done?
- xAOD-to-xAOD reprocessing
    - How often will this happen? How long will a campaign take to plan, validate and execute?
- Placement, deletion, task definition/submission: few worries about this

DC14 should answer some of these questions

Into Run 3

- Many challenges await…
  - Luminosity for both reconstruction *and* simulation
  - Integrating the upgraded detector into the software
  - IT technology challenges… GPUs?
  - What about data preservation
  - Multi-threading needs to be aggressively pursued, already during Run-2
    - Work already under way with Gaudi-Hive
- **Lots of work for a lot of people!**

Conclusions

- During Long Shutdown 1, several hundred people have contributed to some huge improvements in the software and computing
  - Faster reconstuction
  - New analysis model
  - New interfaces to the grid
  - New data placement and management systems
- This puts ATLAS in an excellent position for Run 2
- Many challenges need to be addressed for Run 3
  - There has been a long-term decline in the number of people working on offline software. **This needs to be reversed.**

Additional material

Optimizing Memory Footprint

- Multi-process Athena – **AthenaMP** – our approach to saving memory in reconstruction and simulation jobs

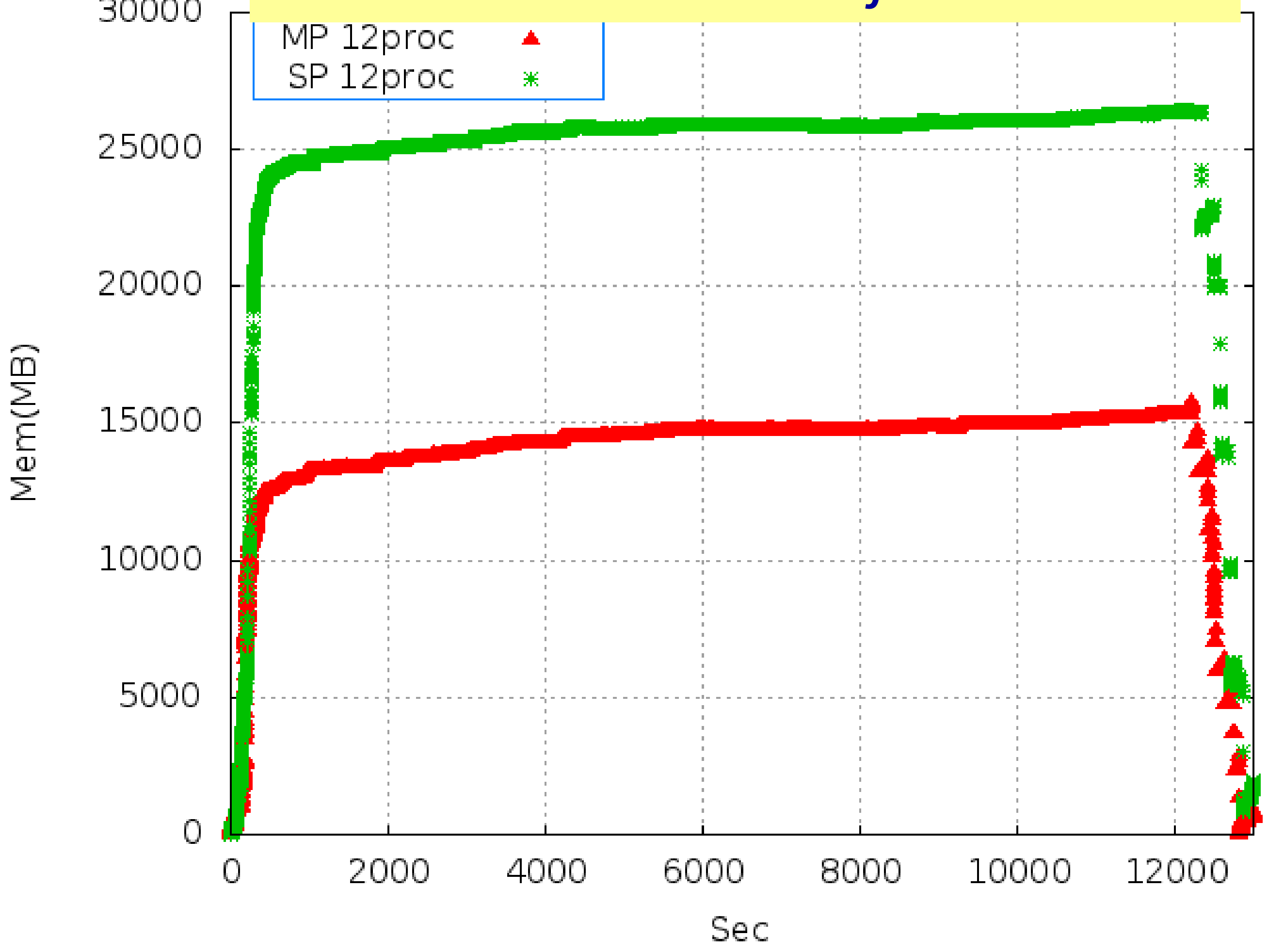
- Leverages Linux **fork()** and **Copy On Write** for sharing memory pages between worker processes

- Memory savings come **“for free”** with **no changes in the algorithmic code**

- Several **multi-core queues** enabled at ATLAS Grid sites for running AthenaMP jobs

- AthenaMP is currently used in 2014 to run **new G4 simulation**
  - The plan is to use it for **reconstruction too**

Reconstruction memory profile
MP vs serial jobs



Migration to ROOT6, CMake, ...

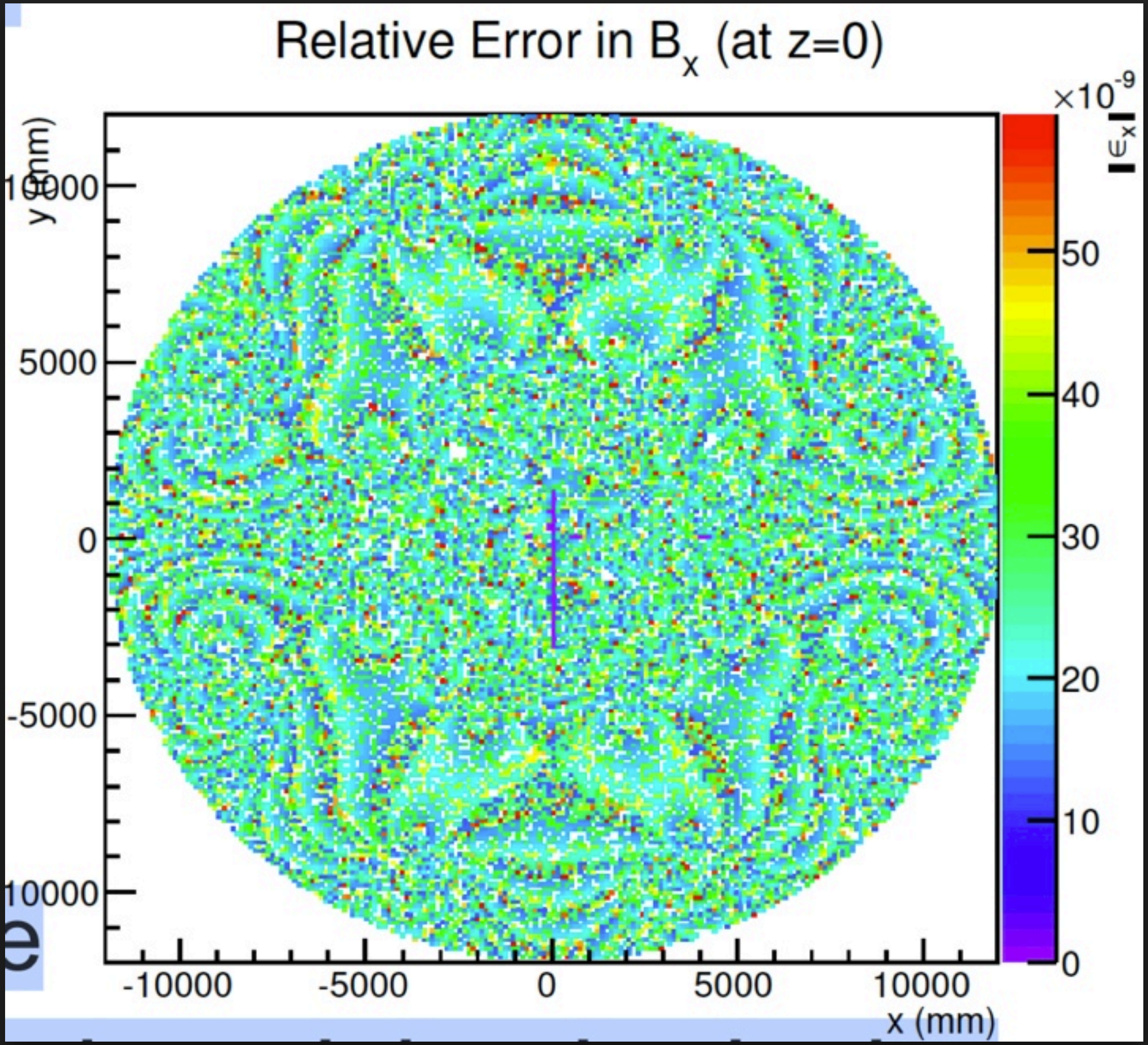
- ATLAS ROOT6 Task Force making good progress in integration with Athena
  - Plan is to **switch to ROOT6** in Release **19.3.0** (end of September)
  - **Release 20** will contain a production release **ROOT 6.02** (*if no problems have been identified!*)
  - ROOT6 will of course be available for **Analysis Releases**

- Migration **from CMT to CMake**
  - **New build system**, used by LCG software as well as LHCb and ALICE
  - The migration Task Force in place
  - The plan is to **switch to CMake in Release 19.3.0**

- Several other upgrades being worked on
  - **New Conditions Database** instance (*only for Data, not MC)*
  - **Tag Collector III** with improved user interface
  - **Nightly build** updates
  - Migration **from Savannah to JIRA**

Accessing Magnetic Field Map

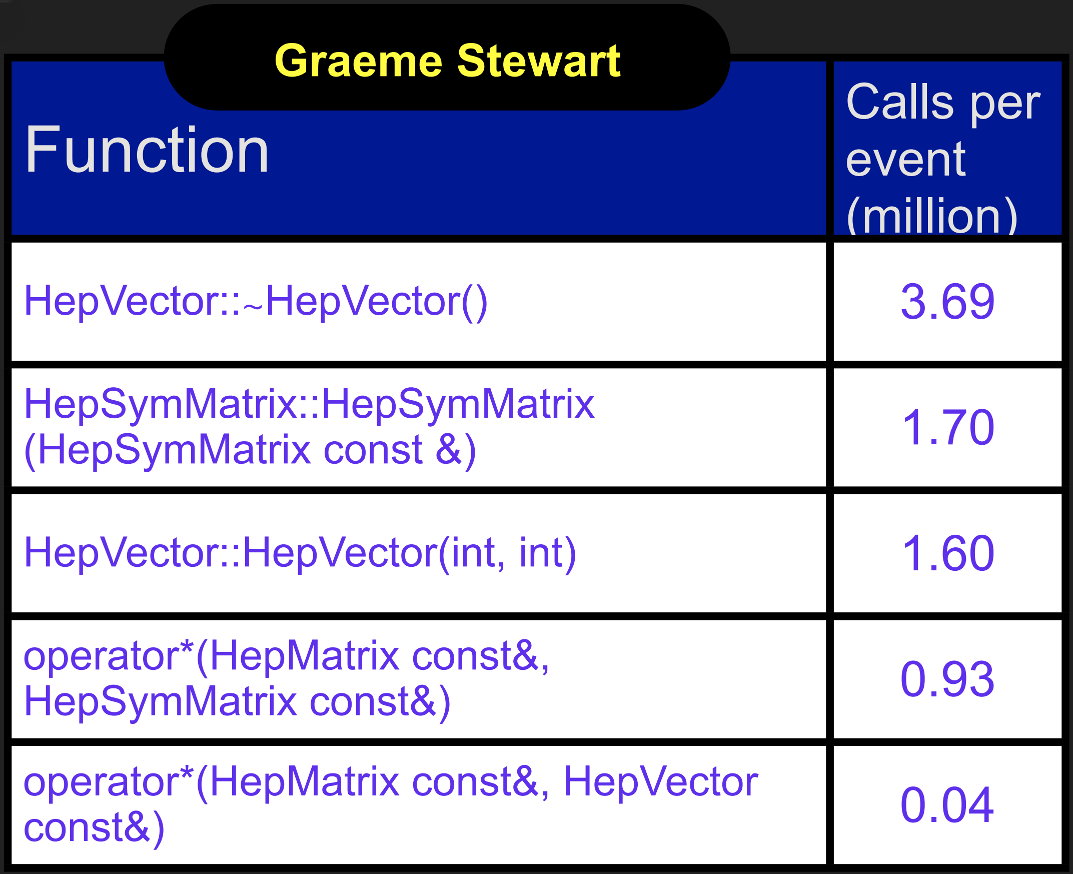
- New AtlasFieldSvc replaced the old MagFieldAthenaSvc
- Code converted from Fortran 77 to C++
- Adding field value cache
- Unit conversion minimisation
- Make code auto-vectorisable and applying intrinsics
- Speed-up of ~20% in simulation jobs



Masahiro Morii
Valerio Ippolito
Emma Tolley

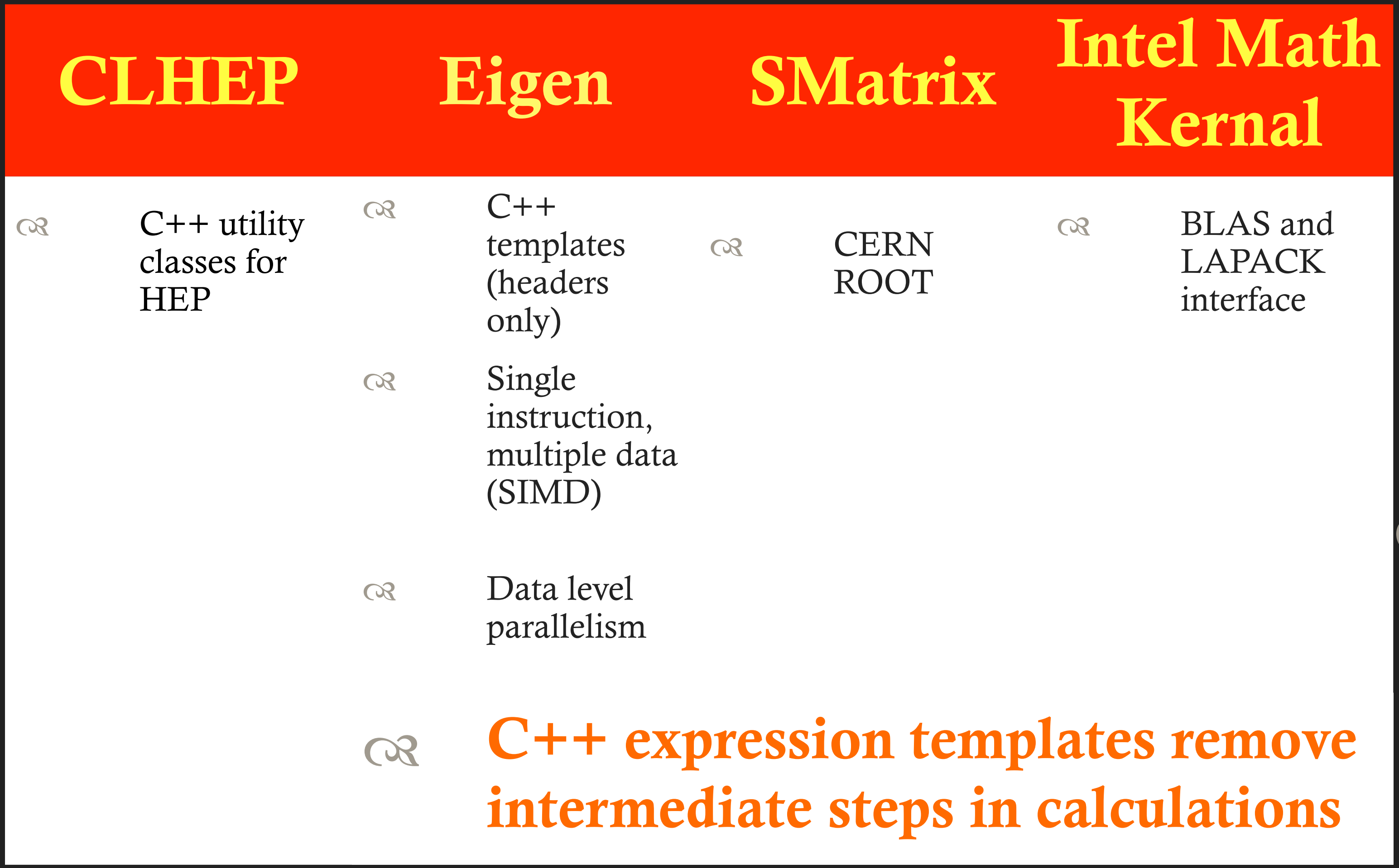
Monitoring CLHEP functions

- Monitor calls to CLHEP in 2012 data (JetTauEtmiss stream) reconstruction job



Graeme Stewart

Function	Calls per event (million)
HepVector::~HepVector()	3.69
HepSymMatrix::HepSymMatrix (HepSymMatrix const &)	1.70
HepVector::HepVector(int, int)	1.60
operator*(HepMatrix const&, HepSymMatrix const&)	0.93
operator*(HepMatrix const&, HepVector const&)	0.04



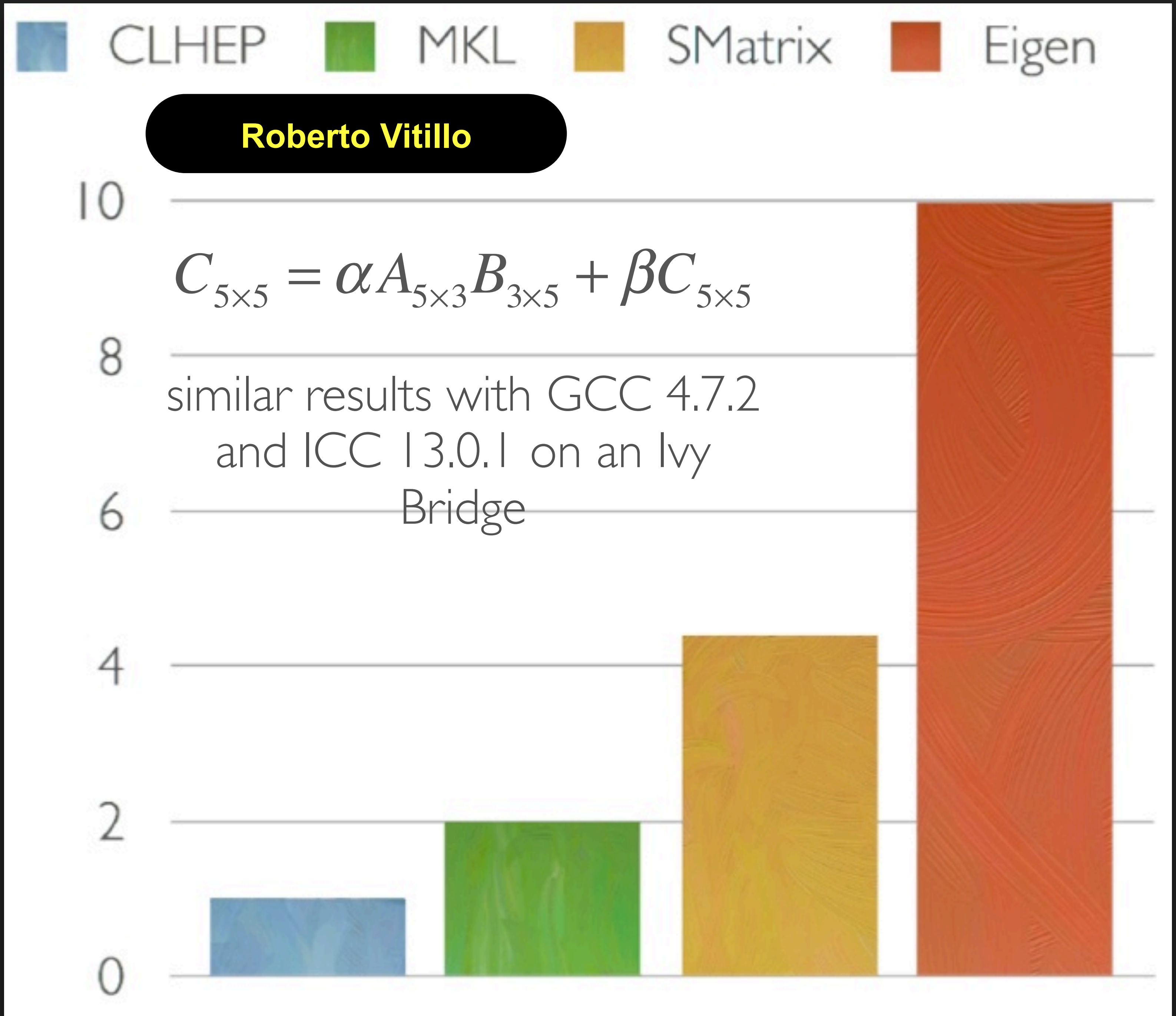
CLHEP	Eigen	SMatrix	Intel Math Kernal
C++ utility classes for HEP	C++ templates (headers only) Single instruction, multiple data (SIMD) Data level parallelism	CERN ROOT	BLAS and LAPACK interface

- **C++ expression templates remove intermediate steps in calculations**

- Seek alternative libraries for linear algebra as CLHEP is no longer supported

Speedup w.r.t CLHEP

- Isolated speed comparison with expression templates



$$C_{5\times5} = \alpha A_{5\times3} B_{3\times5} + \beta C_{5\times5}$$

similar results with GCC 4.7.2 and ICC 13.0.1 on an Ivy Bridge

- Matrix multiplication
- Replace CLHEP with Eigen
- Thousands of lines of code changed in 8 months spanning up to a thousand packages.
- CLHEP Lorentz vectors still necessary.

Trigonometric functions

- GNU libm used as default for trigonometric functions in ATLAS software

Graeme Stewart

Function	Calls per event (million)	Time per call (ns)	Time per event (s)
Exp	3.4	150	0.50
Cos	2.5	150	0.37
Sin	2.2	150	0.33
atanf	2.1	20	0.05
sincosf	2.1	20	0.05

- Total times of all trigonometric functions per event : 2.0 s of 14.1 s (before upgrade)

VDT

- Developed by CMS
- Designed for auto-vectorisation with fast calculations using Pade

libimf

- Performance optimised by Intel
- Can be used as a drop in replacement: set LD_PRELOAD to be loaded at runtime

libm

- Standard GNU
- Precise

Math library	Speed relative to libm
GNU libm	1.0
VDT	0.9
libimf	0.9

- Replace libm with libimf
- VDT available for further study