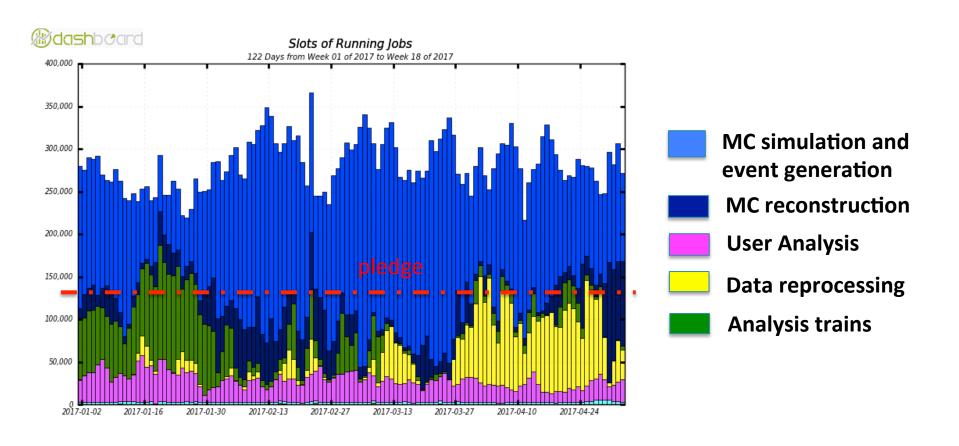
ATLAS Computing Status

Simone Campana

Torre Wenaus

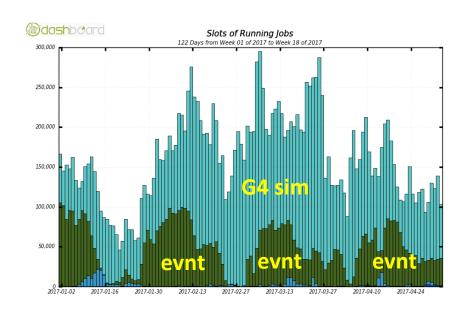


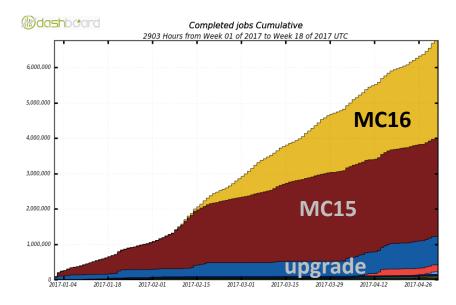
ATLAS computing activity in 2017





MC Simulation





MC16 hits production started mid Feb. Improved Run-2 geometry, new chambers

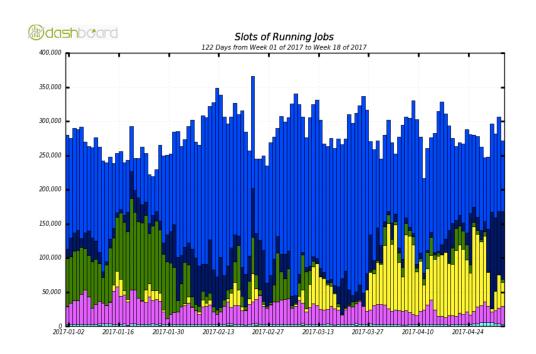
MC16 G4 simulation is 30% faster than MC15 (G4.10, new compiler, code improvements)

MC16 will be the baseline for remaining of Run-2 analyses

Focus now on Fast Simulation



Reconstruction and Reprocessing





Analysis trains

Final derivations with Athena20.7 done in Jan (data+MC)

Focus now on Athena21 derivations, for Combined Performance recommendations and early 2017 analyses



MC reconstruction

Reconstruction of MC16 with Athena R21

So far HITS were reconstructed with 2015/2016 conditions (MC16a)

Preparation of 2017 condition (MC16c) is underway



Data reprocessing

Reprocessing of 2015 and 2016 data with Athena R21 completed in 6 weeks

Only 6 events crashed

Athena21 now on Git, built with CMake



AOD sizes Task Force

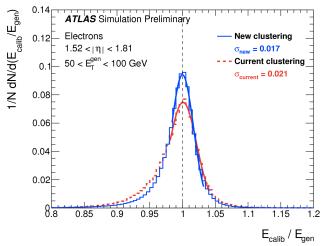
Size savings (KB/ev) w.r.t. baseline 21.0 AOD

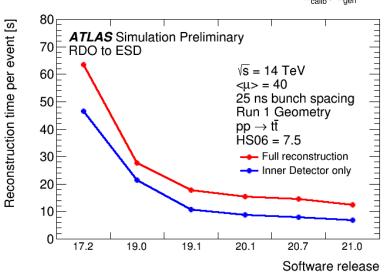
	Data (1st pass)	Data (reprocessed)	Standard Model MC (ttbar)	Signal MC (ttbar)
Track pT 400->500 MeV and covariance matrix compression	26	26	27	27
Removal of negative E caloclusters & unused moments	4	4	12	12
Removal of unused PFlow moments	4	4	12	12
Removal of most jet containers (retain 3)	19	19	40	40
Removal of most flavour tagging (retain 1)	29	29	33	33
Removal of G4 truth	0	0	65	65
Use of AODSLIM	0	45	0	0
Use of AODSUPERSLIM	0	0	0	55
Current AOD size	420	420	582	582
Nominal size/event	420	420	582	582
AOD size in Run-2 Computing Model	320	320	500	500
New AOD size in computing model	319	319	410	410

We achieved implementing the recommendations already in 2017. AODs from latest reprocessing (2015+2016 data) benefit of >20% reduction

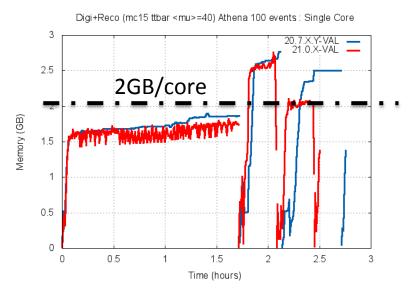


Athena R21 physics and software performance

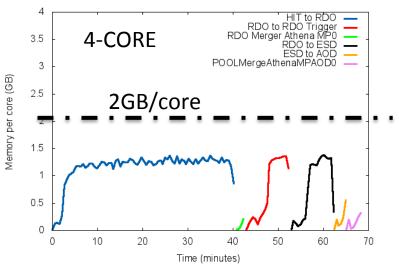




MC digi+trigger+reco, S-CORE



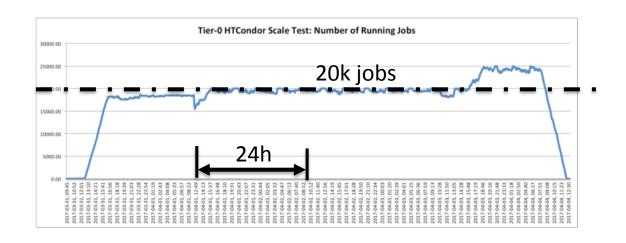
Digi+Reco (mc15 ttbar <mu>=40) AthenaMP 100 events : 4-Core





TO Condor scaling test

Dedicated, specially configured, heavily over-committed nodes, allowing to run up to 200 parallel jobs. No AFS, using EOSFuse



We managed to run the system stably over 4 days, with 20-25k jobs in parallel (2016 production: 12k, 2017 production: expected 15-16k)

With Condor and EOSFuse combined latencies staying well within <~1s total budget we have per job (for preparation, submission, status queries, post-processing, etc.)

Measurement indicate we can reach/exceed the necessary expected throughput of ~100k jobs/day

Will start running small fraction of T0 with Condor, then scale up based on experience



Conclusions

Ready for data

