Cosmic reconstruction

Jamie Boyd CERN

SLAC Atlas mtg

19th Nov 2008

Outline

- Introduction
  - Tier0 specifications
  - Tier0 hardware
  - Atlas dataformats
- Commissioning with Cosmic rays
  - Brief history
  - Current setup
  - Reconstruction performance (CPU/memory/crashes)
  - Tier0 monitoring
  - Offline monitoring
  - Releases & Validation
  - Cosmic ray reconstruction statistics
- Commissioning with simulated physics data
  - FDR
  - Calibration loop
- Conclusions

Slide from David Rousseau

Tier0 reco requirements

- Tier0 reconstruction should be able to reconstruct all events (200Hz) from atlas within ~24 hours
- Tier0 was dimensioned
  - based on 15kSi2K CPU s per event (~6-7 second)
  - Multiple-core CPU with 2GB memory per core (⇒limit on on real memory)
- Crash rate should be as low as possible
  - <1% of jobs (for prompt reconstruction, these jobs should be reprocessed successfully later)
  - ⇒ one job being up to 10.000 events, crash rate should be less than 1 per million event
  - ⇒ given that there are about 100 reconstruction algorithms, crash rate per algorithm and per event should be less than 1 per 100 million

Info from Armin Nairz

Tier0 hardware

- Tier0 uses resources from lxbatch (cern batch workers)
- Uses lsf to submit jobs to specific tier0 queue
- Dedicated castor disk pool 200 TB t0atlas for
  - Incoming raw data (BS files) written by SFO's (at point1
  - Output files (ESDs etc…)
  - Acts as a buffer if tier0 processing behind
    - 200 TB = 7-10 days days at nominal rate
- Also 80 TB t0merge castor diskpool for logfiles and tempory files (files that get merged across streams/jobs)
- 2 types of processor used:

  - dual-cpu, quad-core node: e.g. lxb8418 CPU (count/logical): Intel(R) Xeon(R) CPU E5345 @ 2.33GHz (2/8) memory (swap): 16030 MB (4095 MB) model: ex_07_5 (probably Elonex)

  - dual-cpu, dual-core node: e.g. lxb7834 CPU (count/logical): Intel(R) Xeon(R) CPU 5160 @ 3.00GHz (4/4) memory (swap): 7971 MB (4095 MB) model: e4_07_1 (probably Elonex)
- Nominal resources are:
  - 1600 cores (100 dual-cpu, dual-core nodes, 150 dual-cpu, quad-core nodes)
  - 2GB RAM/core / ~2.4GB with swapping

Atlas Data Formats

- BS – Bytestream = RAW data format
  - size cosmic~1-10MB/evt (depending on num of LAr samples readout), CM~1.6MB/evt
- ESD – Event Summary Data = pool (root) file output of reconstruction
  - size: cosmic~0.5MB/evt CM~0.5MB/evt
- AOD – Analysis Object Data = pool (root) file summary physics objects
  - size: cosmic~40k/evt CM~100k/evt
- TAG – Physics TAG
  - size: CM~1k/evt
- TAG_COMM – Commissioning TAG
  - size: cosmic~0.2k/evt
- CBNT – Combined ntuple (like dump of ESD in ntuple format) – being discontinued
  - size: cosmic ~1MB/evt
- DPD – Derived Physics Data – format varies (mostly pool (root) file)
  - size: varies

CM = computing model (old ref 2005! – probably outdated!)

Reconstruction (history)

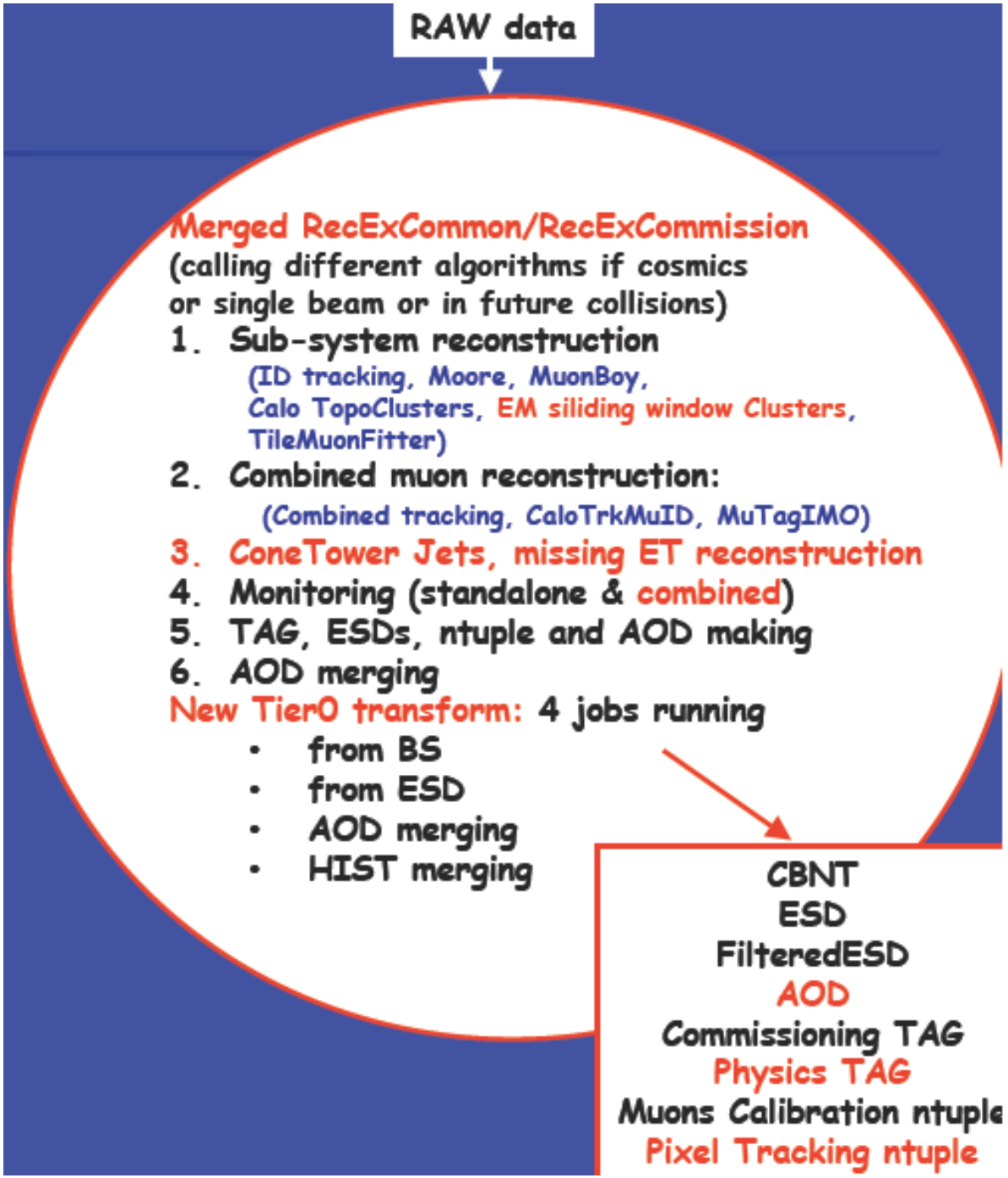
- Tier0 reco of cosmics started with Milestone weeks
  - M3, June 07, 2M evts
  - M4, Sept 08, 3M evts
  - M5, Nov 08, 12 M evts
  - M6 , Mar 09,
  - M7, May 09,
  - M8, July 09,

Increasing detector coverage…

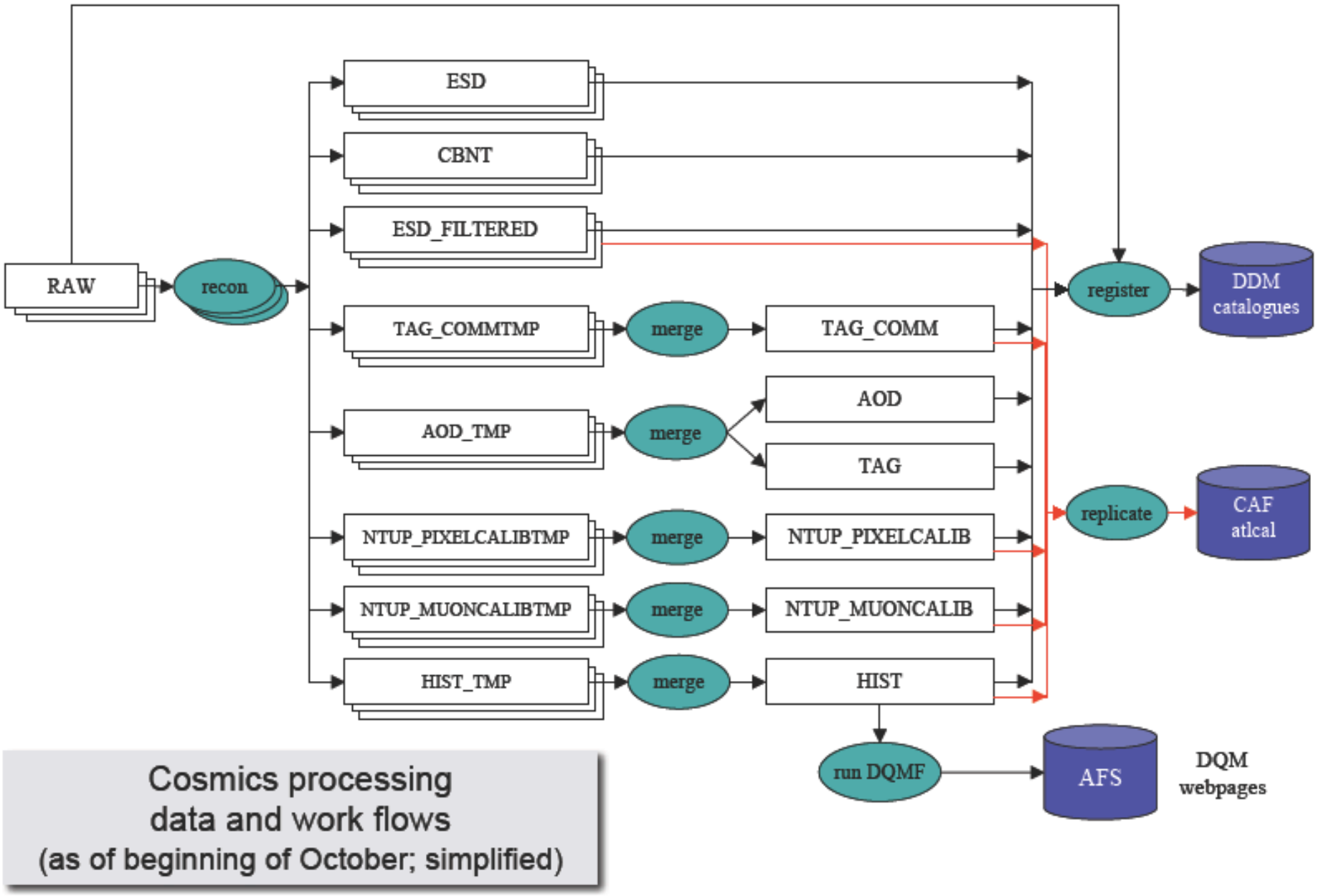
- Then continuous (mostly 24/7) cosmic running since (apart from 10-12 Sept when we had single beam) until Nov 6 09
- The code was running RecExCommission which does minimal reconstruction:
  - ID tracking, Calo cell energy determination (iterative due to random phase of cosmic) topo clusters, Muon tracking (2 algs)
  - Offline monitoring (of raw detector quantities as well as derived quantities like tracks)
  - All done in one job running from BS (outputs ESD, CBNT)
- In September switched to a RecExCommon style job (more like MC reco)
  - Still mostly minimal reco – but with additional reco:
    - Calo – sliding window clustering
    - Minimal Jet and missingEt reo
  - Also produced AODs in separate athena job running from ESD
  - Split monitoring so that most runs in ESDtoAOD stage (reduces memory use in BStoESD step)

Slightly more detailed look at what reconstruction is currently running at tier0.

Red are New since Sept



Slide stolen from Armin Nairz tier0 talk at s/w week – showing current tier0 workflow

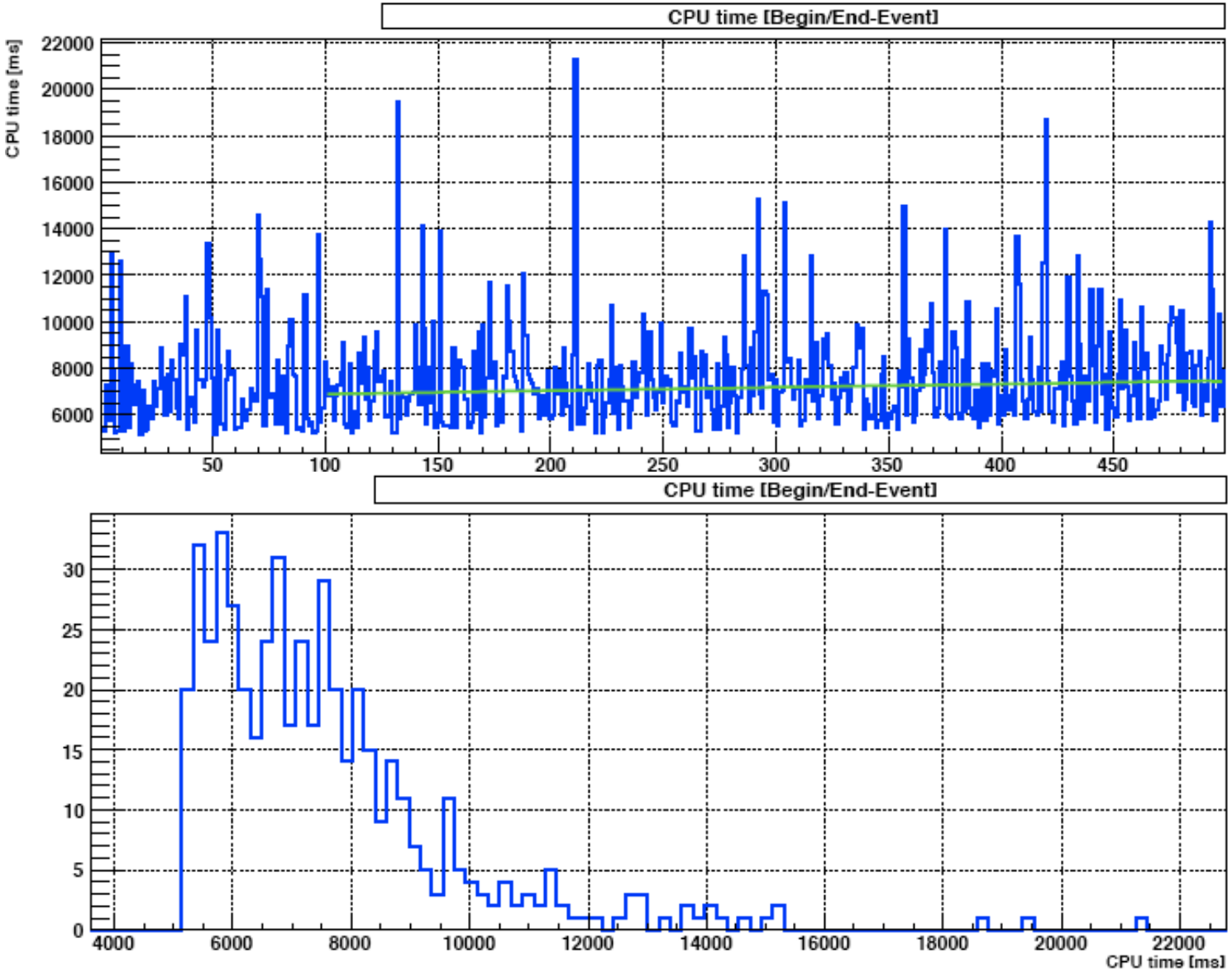


Not just running reconstruction – but a number of complicated and coordinated things!

Reconstruction performance

- A lot of work gone into making the code more stable and robust
  - A few cosmic shower events taking >30mins in muon reco!
  - Crashes due to corrupt data
  - Infinite hangs in track fits
  - Memory leaks and crazy memory use (100MB histograms from RPC monitoring! (bad binning of 2D histograms using TH2D instead of TH2F, …)
  - Crack down on ERROR messages and WARNINGs in the logfiles
- A huge amount of progress has been made
- Resulted in not a single crash in last 2 weeks of combine data taking (~10Million events)
- Tier0 reconstruction of the first beam 'splash' events (with >200k MDT hits, and >1000TeV Calo energy) ran with no crashes
  - Big success!

CPU performance (BS step)



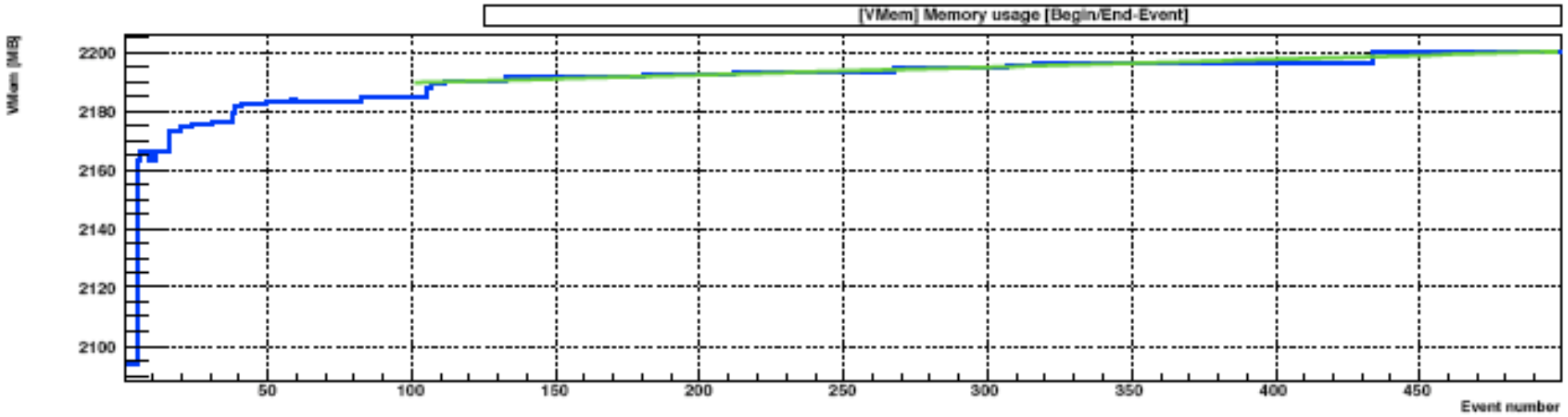
CPU time / evt for BS->ESD step.

Job takes ~7s/evt
Doesn't increase with event number.

Upto ~30s/evt for slow evts (showers)

Slowest algorithms
Mboy, LAr monitoring, LAr reco (cosmic specific)

Memory performance (BS step)



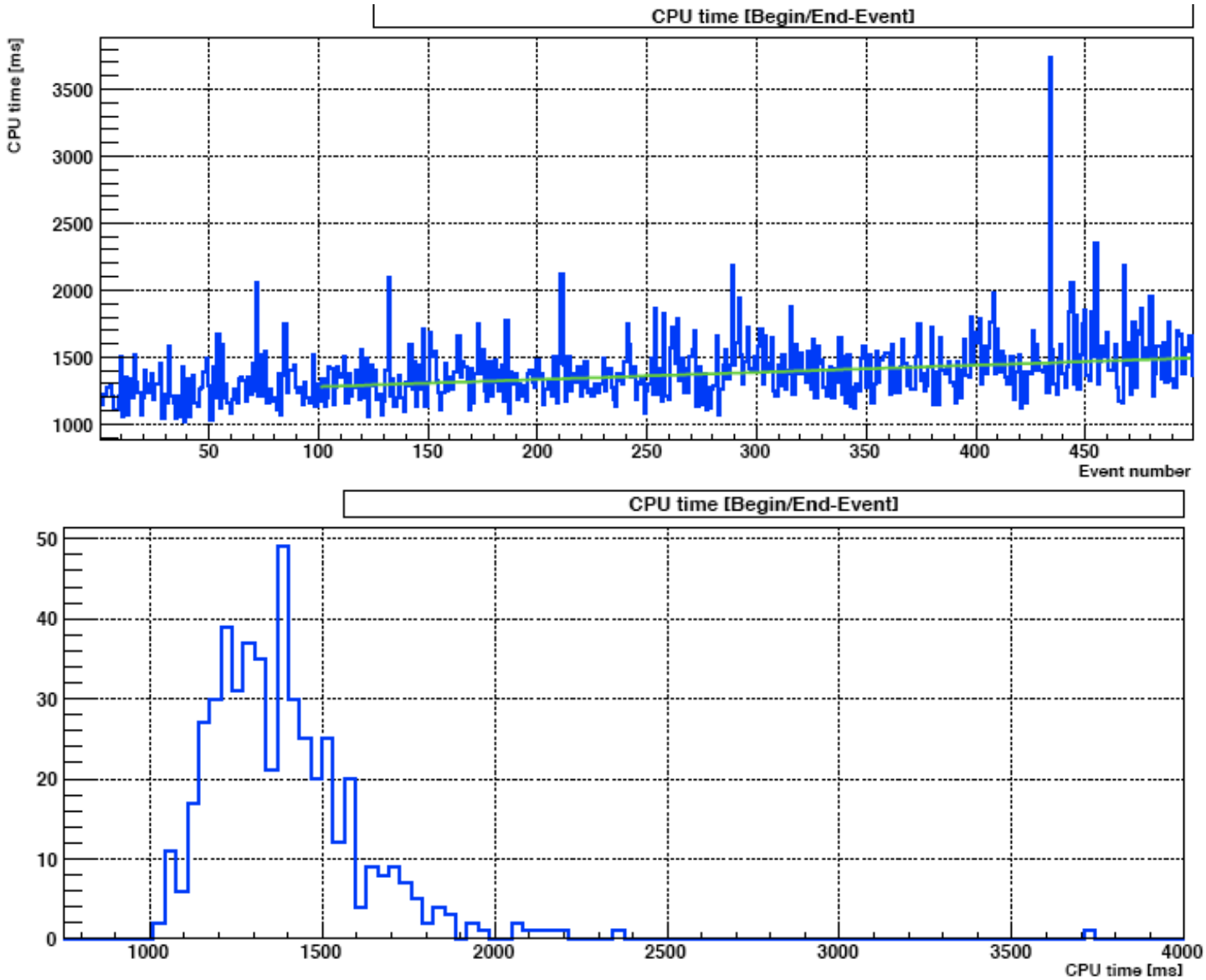
Memory use in the BS->ESD stage as function of event number

Memory flattens out at ~2.2GB (no significant memory leaks)

Some memory growth (~100MB) in the first 100 events – from caching in code

A significant amount of this memory comes from the fact that LAr energy calculation is being done offline (rather in DSP) using an iterative method (phase of cosmic is unknown) this uses ~150 MB of memory more than would be for collisions.
Monitoring also a big memory consumer.

CPU Performance (ESD step)

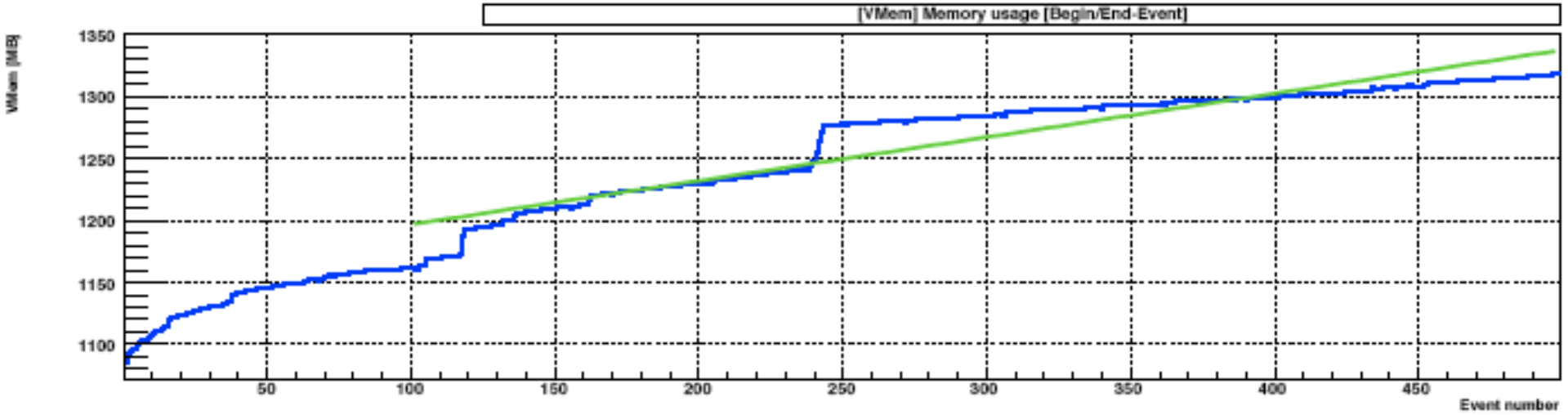


CPU per evt for ESD->AOD step

Average time per event ~1.4s

Slight increase in time per event (due to memory growth – see next slide)
1.3 -> 1.5 s/evt in 400 evts

Memory performance (ESD step)

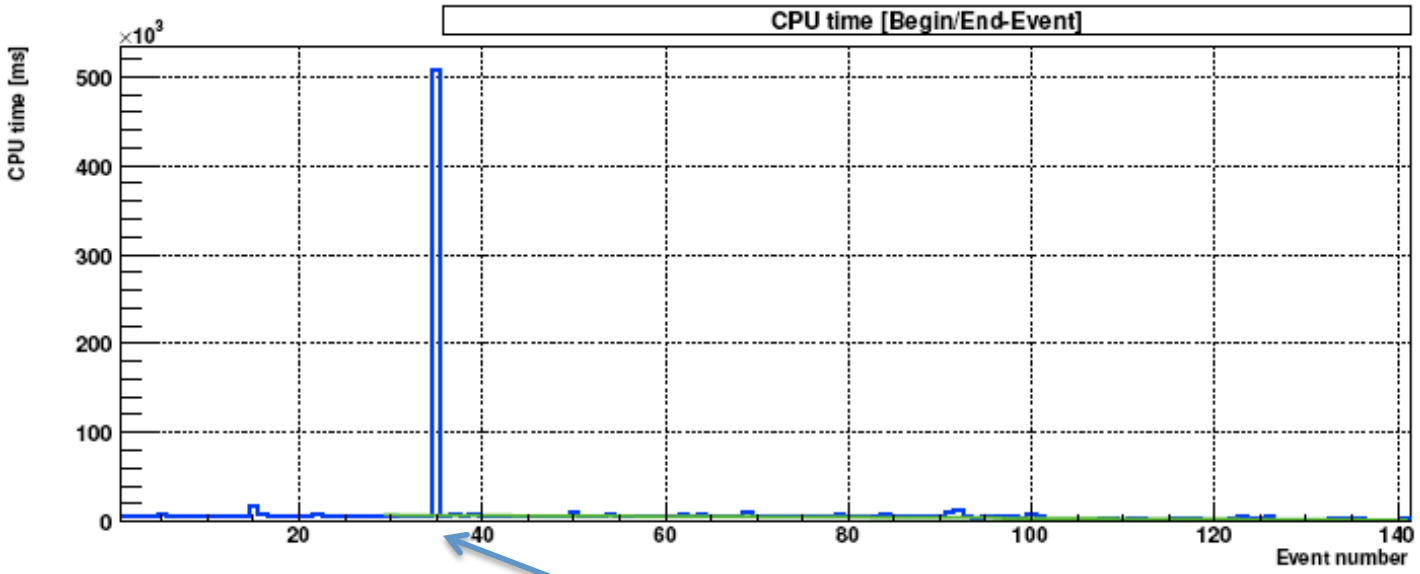


Memory use per event.

Clear memory leak of ~0.5MB/evt being traced – not one leak, some bug reports submitted

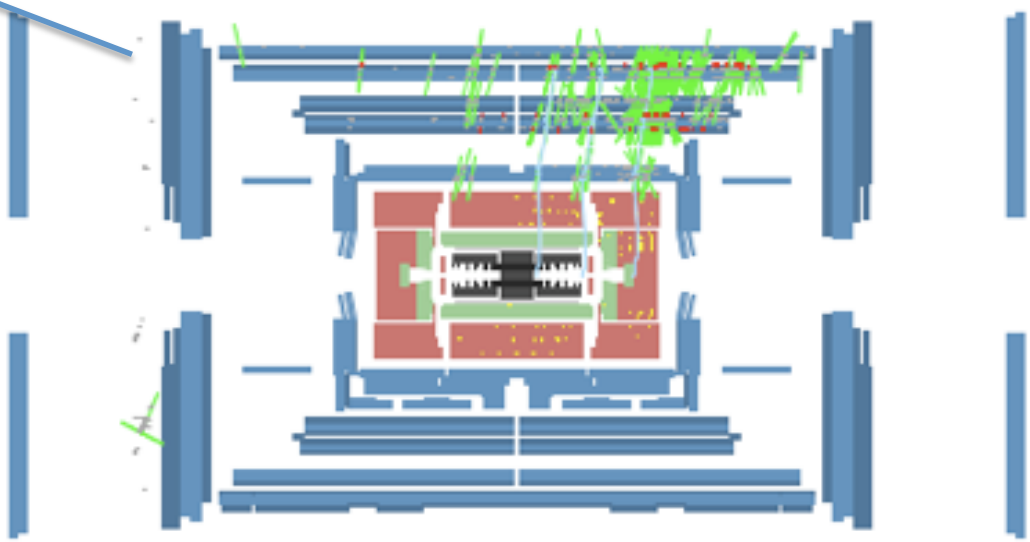
Initial memory use low ~1.1GB so memory leak not a huge problem for the size of jobs we run – still not good!

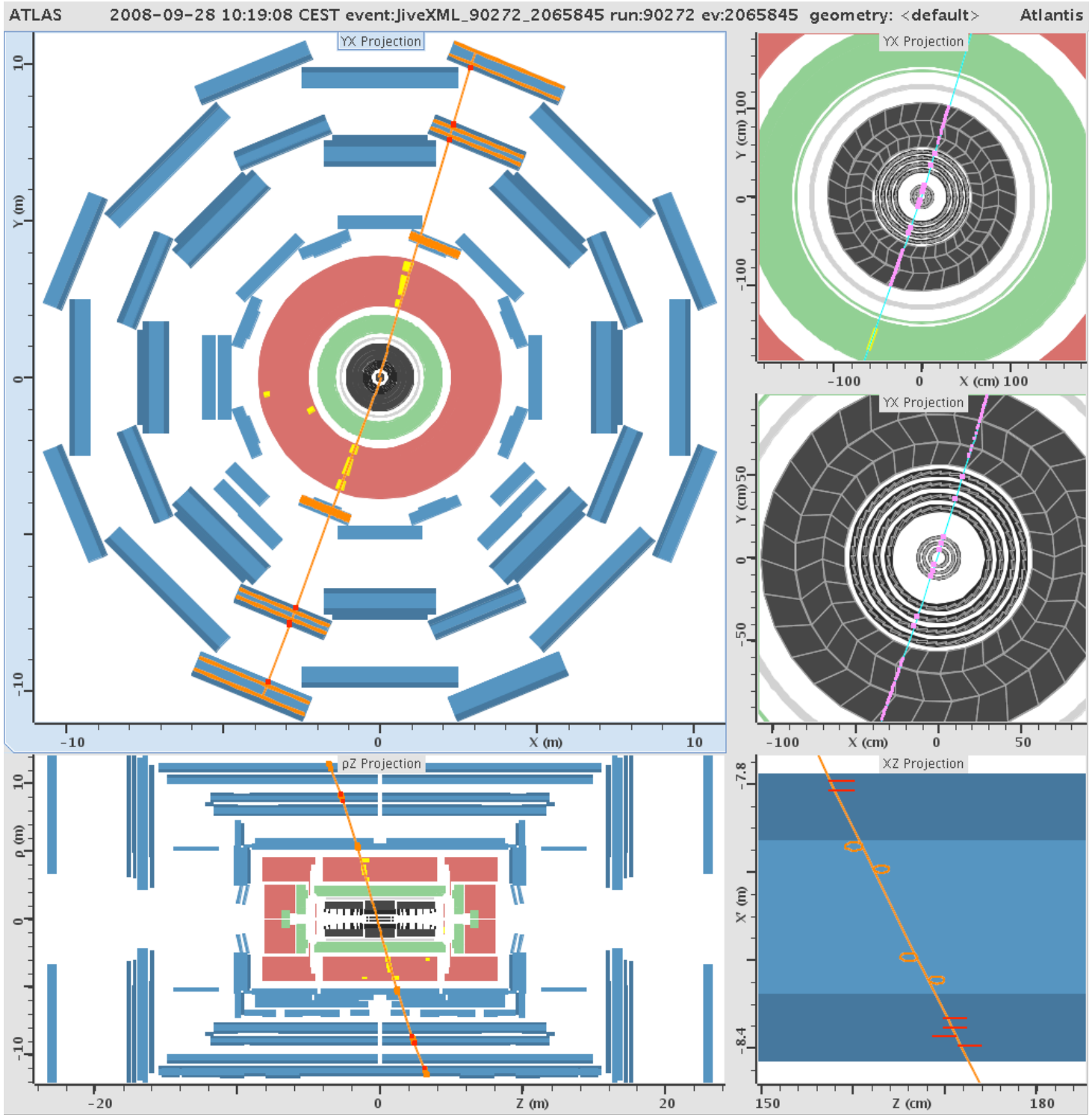
Performance improvements



This cosmic shower event was taking >500s to reconstruct (this was ~6months ago)!

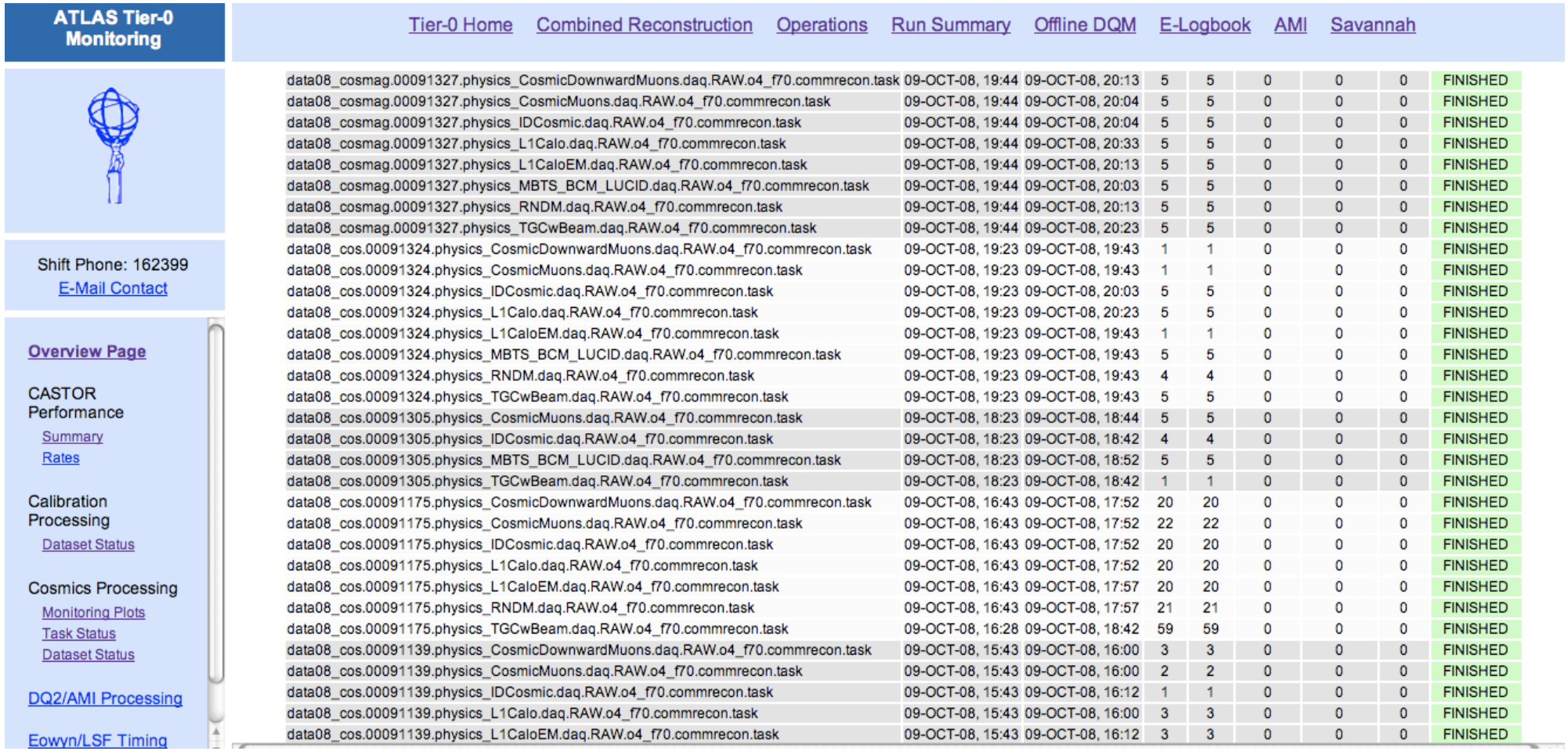
With improvements in Muon tracking code this event now takes ~10s





Proof that reco works

Tier0 monitoring



ATLAS Tier-0 Monitoring

Tier-0 Home | Combined Reconstruction | Operations | Run Summary | Offline DQM | E-Logbook | AMI | Savannah

Shift Phone: 162399
E-Mail Contact

- Overview Page
- CASTOR Performance
  - Summary
  - Rates
- Calibration Processing
  - Dataset Status
- Cosmics Processing
  - Monitoring Plots
  - Task Status
  - Dataset Status
- DQ2/AMI Processing
- Eowyn/LSF Timing

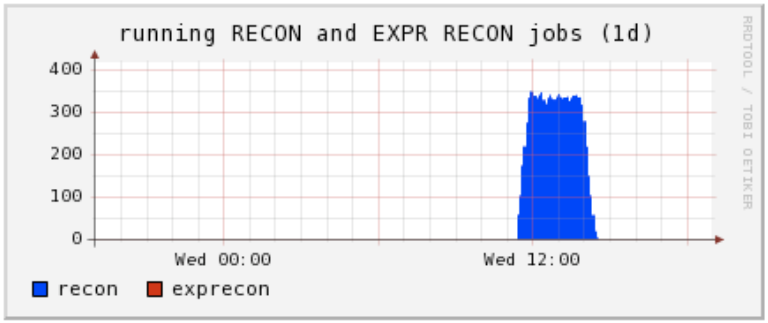
data08_cosmag.00091327.physics_CosmicDownwardMuons.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 19:44	09-OCT-08, 20:13	5	5	0	0	0	FINISHED
data08_cosmag.00091327.physics_CosmicMuons.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 19:44	09-OCT-08, 20:04	5	5	0	0	0	FINISHED
data08_cosmag.00091327.physics_IDCosmic.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 19:44	09-OCT-08, 20:04	5	5	0	0	0	FINISHED
data08_cosmag.00091327.physics_L1Calo.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 19:44	09-OCT-08, 20:33	5	5	0	0	0	FINISHED
data08_cosmag.00091327.physics_L1CaloEM.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 19:44	09-OCT-08, 20:13	5	5	0	0	0	FINISHED
data08_cosmag.00091327.physics_MBTS_BCM_LUCID.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 19:44	09-OCT-08, 20:03	5	5	0	0	0	FINISHED
data08_cosmag.00091327.physics_RNDM.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 19:44	09-OCT-08, 20:13	5	5	0	0	0	FINISHED
data08_cosmag.00091327.physics_TGCwBeam.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 19:44	09-OCT-08, 20:23	5	5	0	0	0	FINISHED
data08_cos.00091324.physics_CosmicDownwardMuons.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 19:23	09-OCT-08, 19:43	1	1	0	0	0	FINISHED
data08_cos.00091324.physics_CosmicMuons.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 19:23	09-OCT-08, 19:43	1	1	0	0	0	FINISHED
data08_cos.00091324.physics_IDCosmic.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 19:23	09-OCT-08, 20:03	5	5	0	0	0	FINISHED
data08_cos.00091324.physics_L1Calo.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 19:23	09-OCT-08, 20:23	5	5	0	0	0	FINISHED
data08_cos.00091324.physics_L1CaloEM.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 19:23	09-OCT-08, 19:43	1	1	0	0	0	FINISHED
data08_cos.00091324.physics_MBTS_BCM_LUCID.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 19:23	09-OCT-08, 19:43	5	5	0	0	0	FINISHED
data08_cos.00091324.physics_RNDM.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 19:23	09-OCT-08, 19:43	4	4	0	0	0	FINISHED
data08_cos.00091324.physics_TGCwBeam.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 19:23	09-OCT-08, 19:43	5	5	0	0	0	FINISHED
data08_cos.00091305.physics_CosmicMuons.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 18:23	09-OCT-08, 18:44	5	5	0	0	0	FINISHED
data08_cos.00091305.physics_IDCosmic.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 18:23	09-OCT-08, 18:42	4	4	0	0	0	FINISHED
data08_cos.00091305.physics_MBTS_BCM_LUCID.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 18:23	09-OCT-08, 18:52	5	5	0	0	0	FINISHED
data08_cos.00091305.physics_TGCwBeam.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 18:23	09-OCT-08, 18:42	1	1	0	0	0	FINISHED
data08_cos.00091175.physics_CosmicDownwardMuons.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 16:43	09-OCT-08, 17:52	20	20	0	0	0	FINISHED
data08_cos.00091175.physics_CosmicMuons.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 16:43	09-OCT-08, 17:52	22	22	0	0	0	FINISHED
data08_cos.00091175.physics_IDCosmic.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 16:43	09-OCT-08, 17:52	20	20	0	0	0	FINISHED
data08_cos.00091175.physics_L1Calo.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 16:43	09-OCT-08, 17:52	20	20	0	0	0	FINISHED
data08_cos.00091175.physics_L1CaloEM.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 16:43	09-OCT-08, 17:57	20	20	0	0	0	FINISHED
data08_cos.00091175.physics_RNDM.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 16:43	09-OCT-08, 17:57	21	21	0	0	0	FINISHED
data08_cos.00091175.physics_TGCwBeam.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 16:28	09-OCT-08, 18:42	59	59	0	0	0	FINISHED
data08_cos.00091139.physics_CosmicDownwardMuons.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 15:43	09-OCT-08, 16:00	3	3	0	0	0	FINISHED
data08_cos.00091139.physics_CosmicMuons.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 15:43	09-OCT-08, 16:00	2	2	0	0	0	FINISHED
data08_cos.00091139.physics_IDCosmic.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 15:43	09-OCT-08, 16:12	1	1	0	0	0	FINISHED
data08_cos.00091139.physics_L1Calo.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 15:43	09-OCT-08, 16:00	3	3	0	0	0	FINISHED
data08_cos.00091139.physics_L1CaloEM.daq.RAW.o4_f70.commrecon.task	09-OCT-08, 15:43	09-OCT-08, 16:12	3	3	0	0	0	FINISHED

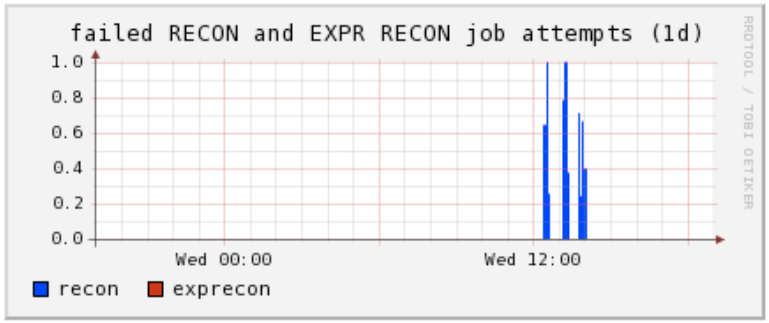
http://atlas.web.cern.ch/Atlas/tzero/prod1/monitoring/

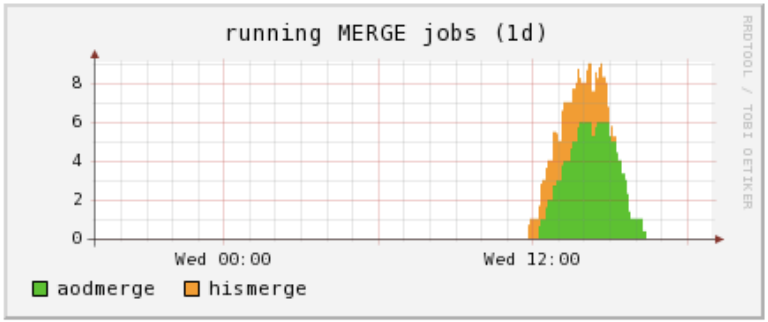
Tier0 Monitoring

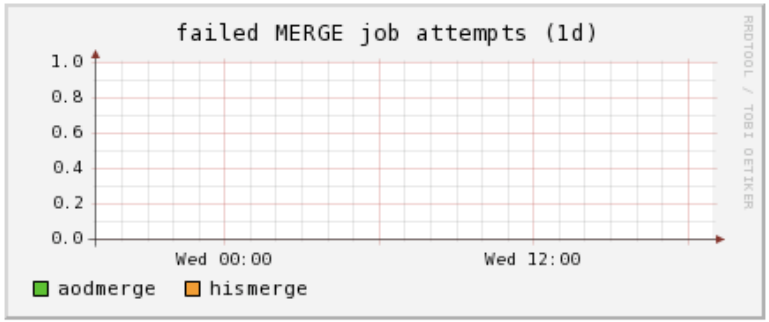
Plot from today (unusual)

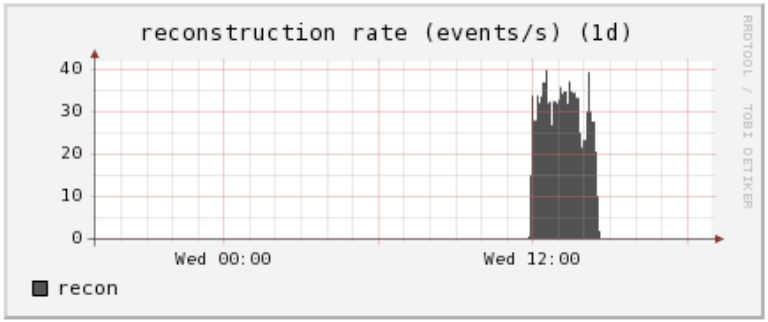
- not many jobs running (>1000 for all of combined running)
- some failures (no crashes in last 10M events processed of combined running)











Offline monitoring

- Offline monitoring plots produced in both BS->ESD and ESD->AOD steps these are then merged
- A different transform then merges the monitoring root files across runs
  - This is done incrementally (15min monitoring)
  - Allows monitoring of a longrun way before the full run is processed ← very important!
- Monitoring plots displayed on web at:

http://atlasdqm.cern.ch/tier0/Cosmics08/results_Cosmics08.html

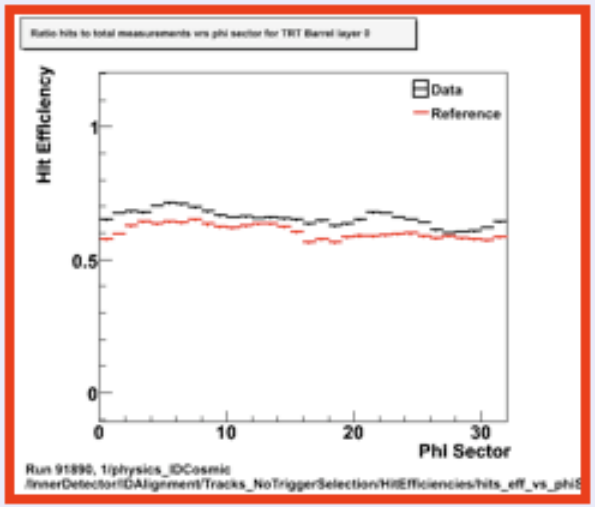
*** Indicates run still being processed

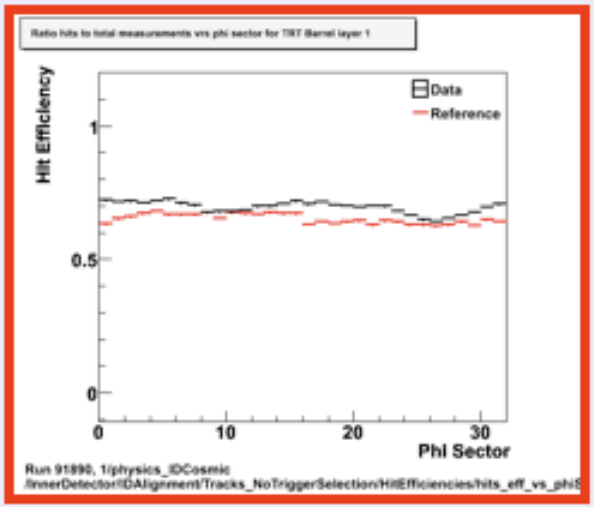
Run Number	T0 Iteration	Streams
95306	1	[physics_CosmicMuons***] [physics_L1Calo***] [physics_L1CaloEM***] [physics_MBTS_BCM_LUCID***]
95267	1	[physics_CosmicMuons] [physics_L1Calo] [physics_L1CaloEM] [physics_MBTS_BCM_LUCID]
94935	1	[physics_CosmicMuons] [physics_L1Calo] [physics_L1CaloEM] [physics_MBTS_BCM_LUCID]
94932	1	[calibration_LArCells] [physics_CosmicMuons] [physics_L1Calo] [physics_L1CaloEM] [physics_MBTS_BCM_LUCID]
94851	1	[physics_CosmicDownwardMuons] [physics_MBTS_BCM_LUCID] [physics_RPCwBeam]
94757	1	[physics_MBTS_BCM_LUCID]
94582	1	[physics_CosmicDownwardMuons] [physics_MBTS_BCM_LUCID] [physics_RPCwBeam]
94563	1	[physics_CosmicDownwardMuons] [physics_MBTS_BCM_LUCID] [physics_RPCwBeam]
94491	1	[calibration_LArCells] [physics_CosmicMuons] [physics_L1Calo] [physics_L1CaloEM] [physics_MBTS_BCM_LUCID]
94462	1	[physics_CosmicDownwardMuons] [physics_MBTS_BCM_LUCID] [physics_RPCwBeam]

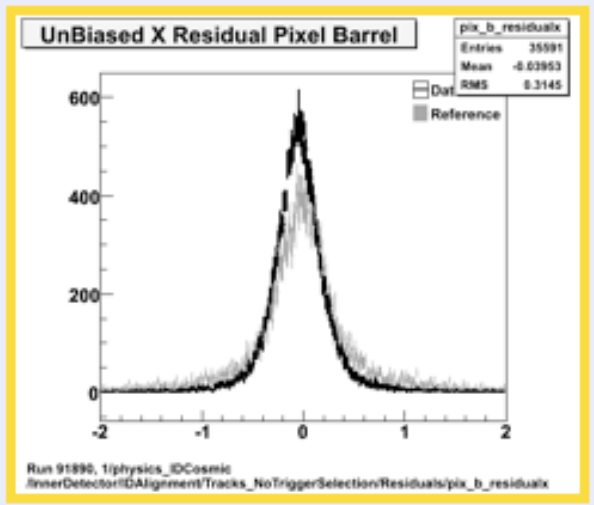
Monitoring plots per stream

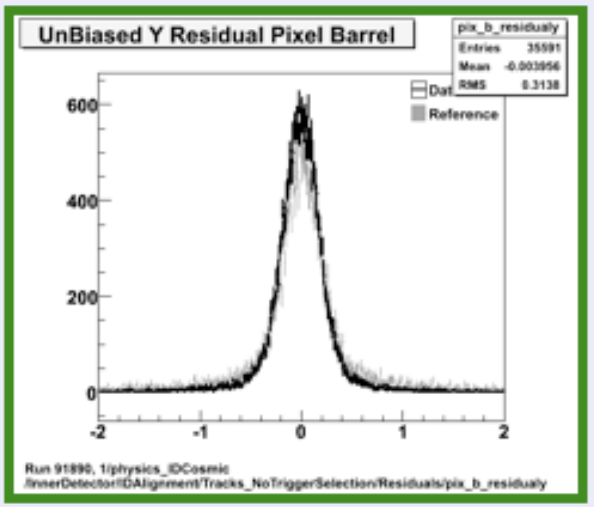
Offline monitoring – Plots…

- InnerDetector: Red
  - Global: Green
    - Hits: Undefined
    - Noise: Undefined
    - Synchronization: Undefined
    - Tracks: Green
  - IDAlignment: Red
    - Tracks_NoTriggerSelection: Red
      - GenericTracks: Green
      - GenericTracks_Detail: Green
      - HitEfficiencies: Red
      - Residuals: Red
  - Pixel: Red
    - DQMF: Red
      - BLayer: Red
      - Barrel: Red
      - ECA: Red
      - ECC: Red
    - DQShift: Green
    - PixelExpert: Green
      - DisabledModules: Undefined
      - General: Green
      - Occupancies: Green
      - Timing: Green
      - ToT: Green
  - SCT: Red
    - SCTB: Red
      - Efficiency: Red
      - Errors: Red
      - Noise: Red
      - TBin: Red
      - TBinTracks: Green
      - TrackHitMaps: Undefined
      - Tracks: Undefined
    - SCTEA: Red
      - Efficiency: Green
      - Errors: Red
      - Noise: Red
      - TBin: Red
      - TBinTracks: Green
      - TrackHitMaps: Undefined
      - Tracks: Undefined
    - SCTEC: Red
      - Efficiency: Green
      - Errors: Red
      - Noise: Red
      - TBin: Red









Colour of outer box of plot is indication of the DQ (from automatic comparisons)

Monitoring memory use

System Mon	Mem use in BS->ESD step (MB)	Mem use in ESD->AOD step (MB)
MuonRaw	9	72
LAr	79	0
Tile	26	29
CTP	22	0
Pixel	2	2
L1	2	0
TRT	17	0
Global	14	0
ID Global	13	0
SCT	12	0
HLT	11	0
L1Calo	10	0
Calo	0	80
Muon Comb/Track/Segment/ Phys	0	6 / 12 / 13/10
Jet	0	1
MissingEt	0	11
Total	218	242

Save ~250MB by running monitoring in ESD step (good).
But more monitoring could be migrated to ESD step.
Overall use ~460MB in monitoring (a lot!)

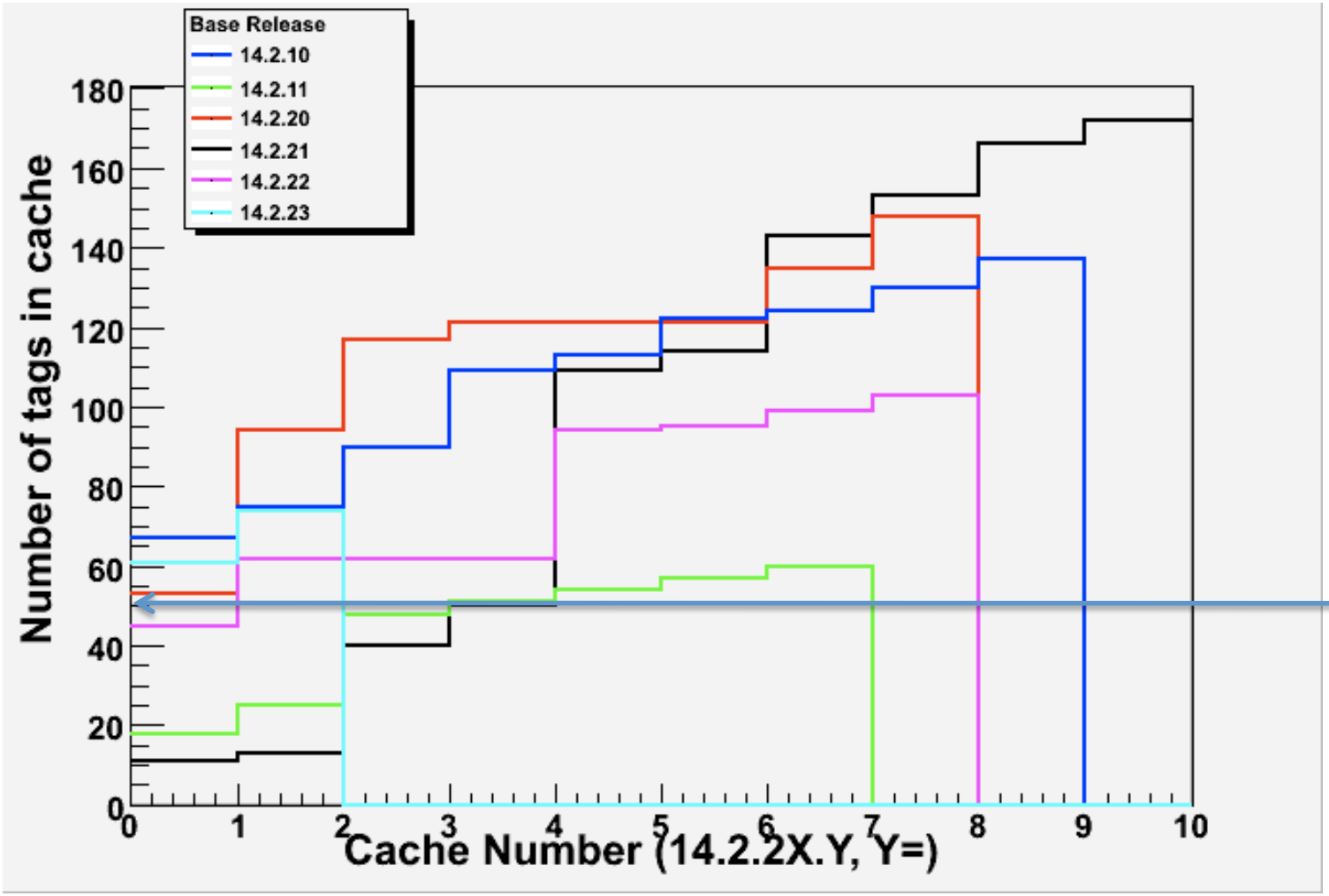
Need to have a clear policy to allow mon changes into tier0 processing. What memory use is acceptable?

LAr mon should reduce memory use – or move to ESD step.

Release strategy

- AtlasTier0 patching project (cache) sits on top of fixed release eg. 14.2.2X.Y built on top of 14.2.24
- Urgent bug fixes, improvements can be deployed at tier0 with ~24hr cycle
- Have 14.2.2X.Y nightly with large scale validation Tier0 Chain Test (TCT) running tier0 transform over ~10-50k events
- Problem with Patch projects are:
  - If has too many tags in it takes too long to compile
  - Changes with header file changes or which require many tags to be recompiled cant go in
- In this case need to move to a new base release (eg. 14.2.25)
  - Slow and requires much validation (this is where we merge in tier0 fixes with updates for GRID production and physics analysis) <- ALWAYS PAINFUL
  - David Quarrie suggested moving to fixed release when patch release contained 50 tags

Releases contd.



50 tags

Plot of the number of tags in the patch project for summer 2008 processing
Can see that we often (nearly always) exceeded the 50 tag limit!

Release Validation

- Of course crucial that s/w running at tier0 doesn't crash and produces correct results
- Use TCT to check the s/w before switching tier0 to use it
- AID summary page shows status of nightly:

http://atlas-computing.web.cern.ch/atlas-computing/links/distDirectory/nightlies/aid_tct/

- Checked by s/w validation shifter – posts savannah bugreports if problems / crashes
- For cosmics no-one has been checking physics output of validation tests (ie. Is tracking OK) – needs to change for collisions

Release	Job	Log Name	N Events	Log Check	Hist Check	Manager(s)
14.2.OLD.Y-VAL	BSreco-T0Test_1-14.2.OLD.Y	BStoESD	N/A	html hist	N/A	N/A
14.2.OLD.Y-VAL	BSreco-T0Test_2-14.2.OLD.Y	BStoESD	N/A	html hist	N/A	N/A
14.2.OLD.Y-VAL	BSreco-T0Test_4-14.2.OLD.Y	BStoESD	N/A	html hist	N/A	N/A
14.2.OLD.Y-VAL	BSreco-T0Test_0-14.2.OLD.Y	BStoESD	N/A	html hist	N/A	N/A
14.2.OLD.Y-VAL	BSreco-T0Test_3-14.2.OLD.Y	BStoESD	N/A	html hist	N/A	N/A
14.2.OLD.Y-VAL	BSreco-T0Test_6-14.2.OLD.Y	BStoESD	N/A	html hist	N/A	N/A
14.2.OLD.Y-VAL	BSreco-T0Test_5-14.2.OLD.Y	BStoESD	N/A	html hist	N/A	N/A
14.2.OLD.Y-VAL	BSreco-T0Test_8-14.2.OLD.Y	BStoESD	N/A	html hist	N/A	N/A
14.2.OLD.Y-VAL	BSreco-T0Test_10-14.2.OLD.Y	BStoESD	N/A	html hist	N/A	N/A
14.2.OLD.Y-VAL	BSreco-T0Test_9-14.2.OLD.Y	BStoESD	N/A	html hist	N/A	N/A
14.2.OLD.Y-VAL	BSreco-T0Test_11-14.2.OLD.Y	BStoESD	N/A	html hist	N/A	N/A
14.2.OLD.Y-VAL	BSreco-T0Test_12-14.2.OLD.Y	BStoESD	N/A	html hist	N/A	N/A
14.2.OLD.Y-VAL	BSreco-T0Test_13-14.2.OLD.Y	BStoESD	N/A	html hist	N/A	N/A
14.2.OLD.Y-VAL	BSreco-T0Test_14-14.2.OLD.Y	BStoESD	N/A	html hist	N/A	N/A
14.2.OLD.Y-VAL	BSreco-T0Test_1-14.2.OLD.Y	ESDtoAOD	N/A	html hist	N/A	N/A
14.2.OLD.Y-VAL	BSreco-T0Test_2-14.2.OLD.Y	ESDtoAOD	N/A	html hist	N/A	N/A
14.2.OLD.Y-VAL	BSreco-T0Test_4-14.2.OLD.Y	ESDtoAOD	N/A	html hist	N/A	N/A
14.2.OLD.Y-VAL	BSreco-T0Test_0-14.2.OLD.Y	ESDtoAOD	N/A	html hist	N/A	N/A
14.2.OLD.Y-VAL	BSreco-T0Test_3-14.2.OLD.Y	ESDtoAOD	N/A	html hist	N/A	N/A

Current release coordination & validation strategy is very tricky and time consuming – needs to be improved!

Crashes / Bugs at Tier0

- Crashes at tier0 are automatically detected by tier0 monitoring and the relevant part of the logfile is on the web
- Tier0 shifter assigns the bug to Tier0 savannah bug report (removing obvious duplicates):

https://savannah.cern.ch/projects/atlaspoint1t0/

- S/W validation shifter re-assigns the bug to the relevant bug tracker and follows its progress – making sure any fixes get into the Tier0 reco
- Quite successful for rare bugs – but when we have a sudden storm of crashes (eg. Data corruption for a certain run) not dealt with in a good way
- Better crash reporting scheme(s) under discussion

Bookkeeping

- Need to be able to tell what release & Tier0 setup was used to process a given run – Very important
- All data is in AMI (Atlas Metadata Interface)
- AMI tag is in dataset name:

data08_cosmag.00091890.physics_IDCosmic.recon.ESD.o4_f73._lb0018._sfo01._0002.1

= AMI tag = f73

- Interpreted on AMI webpage: http://ami.in2p3.fr:8080/opencms/opencms/AMI/www/ReferenceTables/

productionStep	tag	SWReleaseCache	DBRelease	Geometry	JobConfig	transformation
recon	f73	AtlasTier0-14.2.22.8	n/a	n/a	n/a	/afs/cern.ch/atlas/project/tzero/prod1/projects/data08_cos/trfs/Recon.v3.py, based on Reco_trf.py

- Current problem with this is Auto-configuration of job

Eg. Magnetic field status looked up in job config and DetDescrVersion / CondDB tag chosen depending on field status. This means whats in AMI for DetDescrVersion / CondDB can be wrong

DDM (dq2) used for bookkeeping of output datasets, which are also in AMI

Table from Armin Nairz

Tier0 cosmic statistics

Data Type	June 24th - October 28th, 2008		
	Files	Events	Tot. Size [MB]
RAW	693,509	464,362,139	1,104,305,943
CALRAW	17,028	110,463,488	17,558,049
ESD	651,109	434,868,635	240,578,525
ESD_FILTERED	474,716	n/a	7,795,534
CBNT	651,109	434,868,635	404,906,622
AOD	2,572	79,046,922	1,894,468
TAG	2,572	79,046,922	4,407
TAG_COMM	4,623	325,676,278	27,558
HIST	5,378	n/a	183,594
NTUP_PIXELCALIB	354	n/a	21,974
NTUP_MUONCALIB	5,022	n/a	2,285,878

1.1PB of RAW data

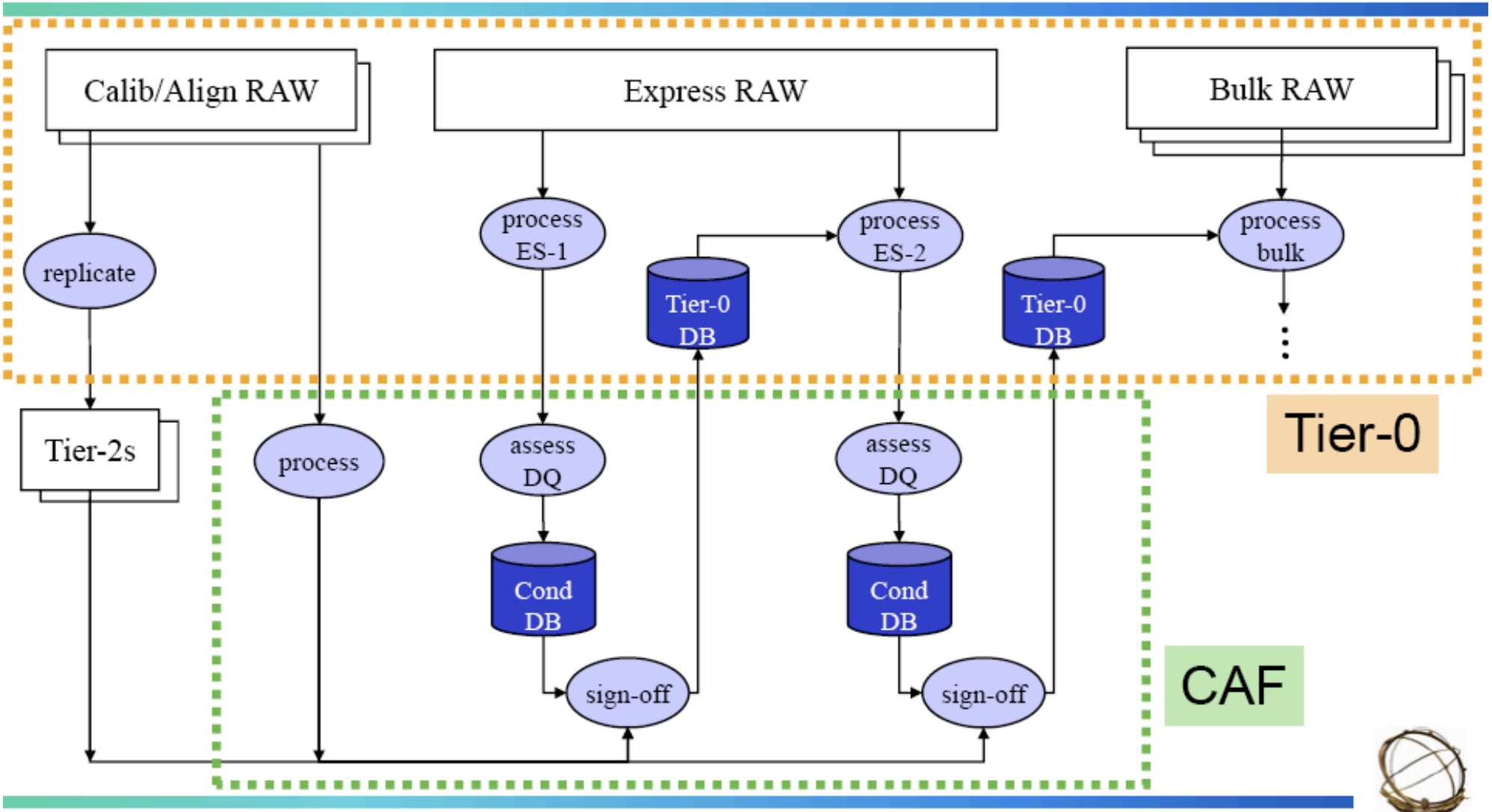
>0.5PB output data

>1450 M evts processed

Commissioning with Physics data (FDR)

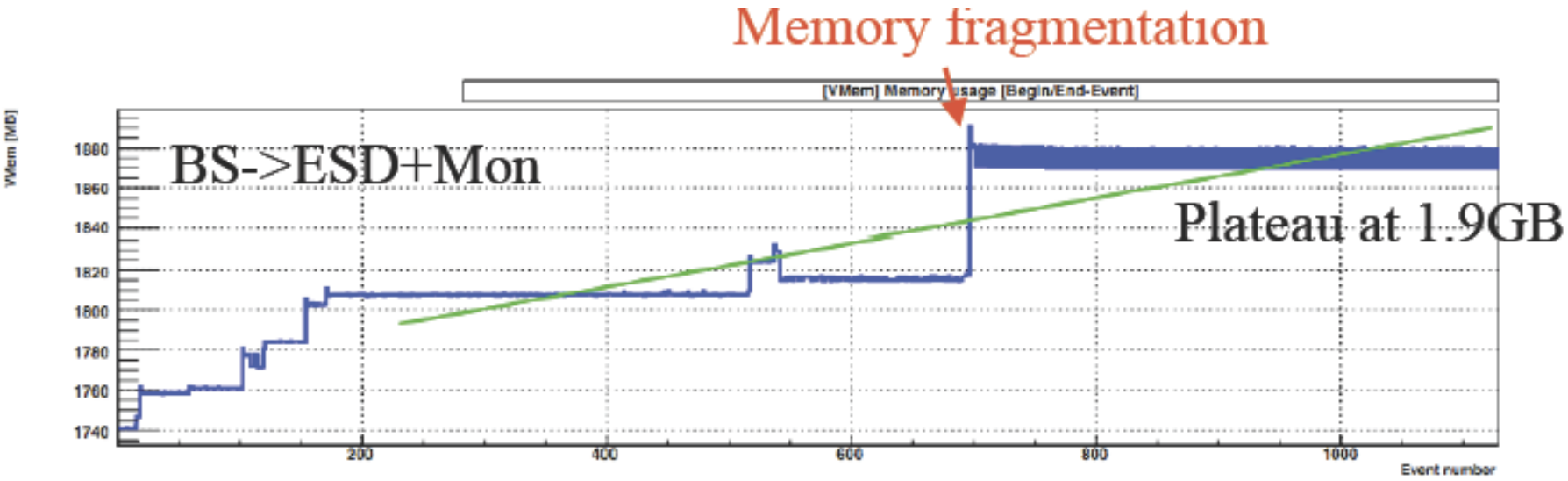
- I have been involved in Tier0 reco on the cosmic reconstruction side – very useful for testing the system with real data & interface between point1 and tier0
- In parallel there has been a number of complementary Full Dress Rehearsal (FDR) exercises which have been using simulated physics data to commission various things including tier0 reco & calibration loop
  - Use a (sort of) realistic mix of data (what would come out of HLT)
  - Test physics streaming
  - BS files from SFOs
  - Run 'real' calibration loop (see next slide)
  - Real physics monitoring (Z's, J/psi's etc..)
  - Running more physics code reco than for cosmics (e-gamma, Tau, b-tagging etc..)

Introduction: The "Calibration Loop"



Reco performance from FDR2

- CPU time for FDR2 reco (weighted average over Jet, El, Mu, MinBias streams for 10^31 lumi) ~20kSi2K
  - Reminder computing model assumes 15 (so 30% more than tier0 budget)
- Memory use ~1.9GB (just about OK, and less than cosmic reco)
  - no obvious memory leaks
  - Eg. For Jet stream:



Conclusions

- Tier0 commissioning using cosmic rays and FDR is well advanced in terms of operational procedures
- Tier0 reco of cosmics also provided crucial feedback to detectors for their commissioning in a reliable and timely manner
- Cosmic/FDR reco at tier0 have also helped to improve the code a lot (dealing with corrupt data, high multiplicity events, memory & CPU …)
- Still many things to get right
  - Release coordination and validation takes too much time and effort
    - Need to add in physics validation of all new code to run at tier0
  - Calibration loop not well tested in continuous running mode – many things need to become automated
  - Memory and CPU use in reco will be a continual battle

Further reading…

- https://twiki.cern.ch/twiki/bin/view/Atlas/CosmicCommissioningReconstructionStatus

Cosmic reco configuration + data-access

- http://atlas.web.cern.ch/Atlas/tzero/prod1/monitoring/

Tier0 monitoring

- http://atlasdqm.cern.ch/tier0/Cosmics08/results_Cosmics08.html

Offline monitoring DQ plots