

# ATLAS Distributed Computing in LHC Run2

Simone Campana – CERN  
on behalf of the ATLAS collaboration

# The Run-2 Challenge



**Resources constrained by “flat budget”  
(no increase in funding for computing)**

## A new detector



**e.g. tracking, calorimeters**

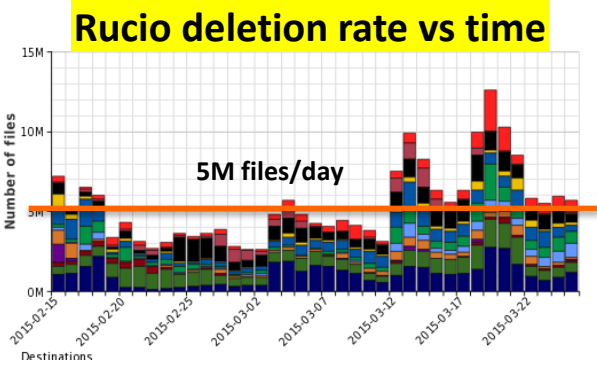
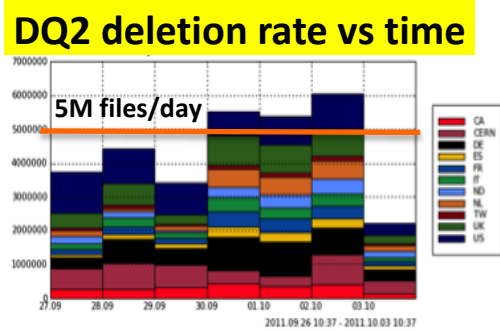
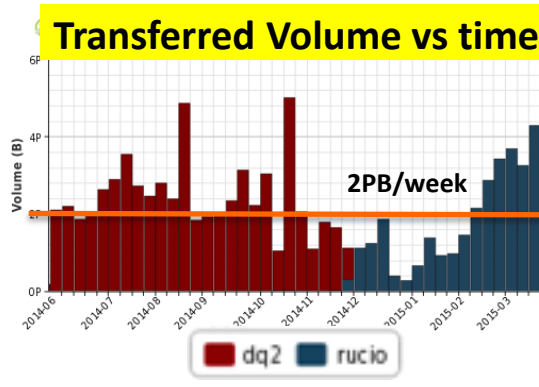
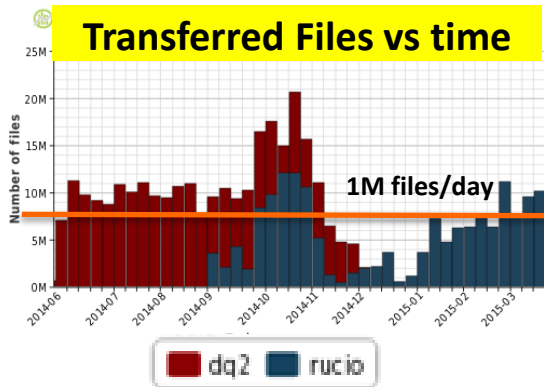
# How to face the Run-2 challenge

- New ATLAS distributed computing systems
  - Rucio for Data Management
  - Prodsys-2 for Workload Management
  - FAX and Event Service to optimize resource usage
- More efficient utilization of resources
  - Improvements in Simulation/Reconstruction
  - Limit resource consumption (e.g. memory sharing in multicore)
  - Optimize workflows (Derivation Framework/Analysis Model)
- Leveraging opportunistic resources additionally to pledged ones
  - Grid, Cloud, HPC, Volunteer Computing
- New data lifecycle management model

# New ATLAS distributed computing systems

# Distributed Data Management: Rucio

The new ATLAS Data Management system, Rucio<sup>[1]</sup>, is in production since 1<sup>st</sup> December 2014



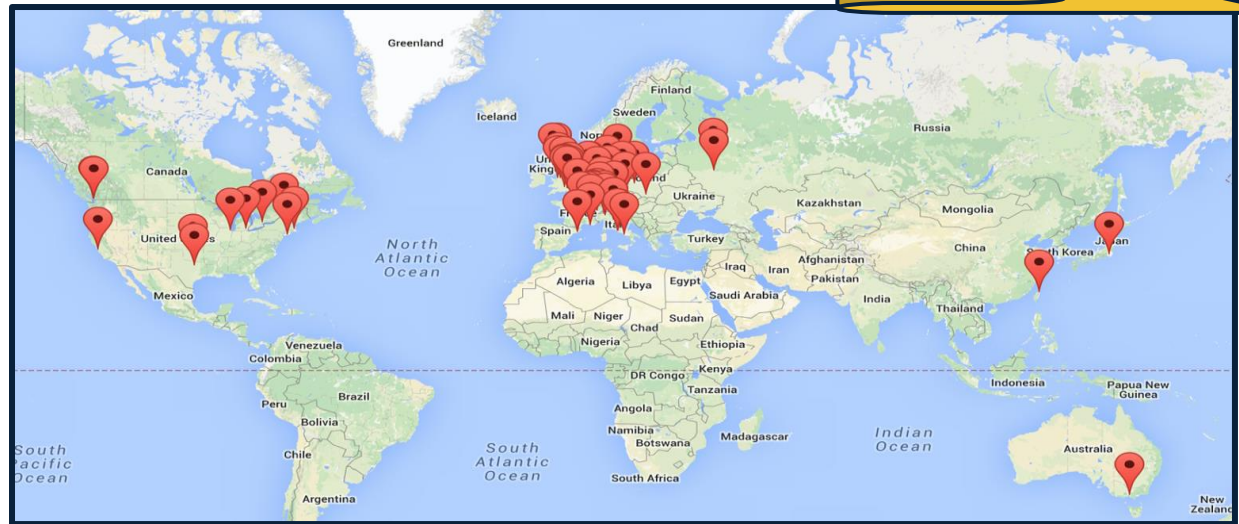
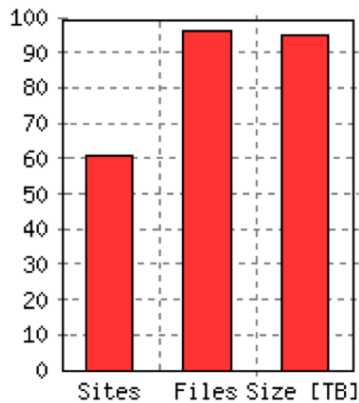
Rucio: a sophisticated system (offers more features than the previous one)

Already at early stage, equivalent performance as previous DDM in core functionalities

Most of Rucio potential (still unexplored) will be leveraged in production during Run-2

# Remote data access: the Xrootd ATLAS Federation (FAX)

Goal reached ! ~100% data covered



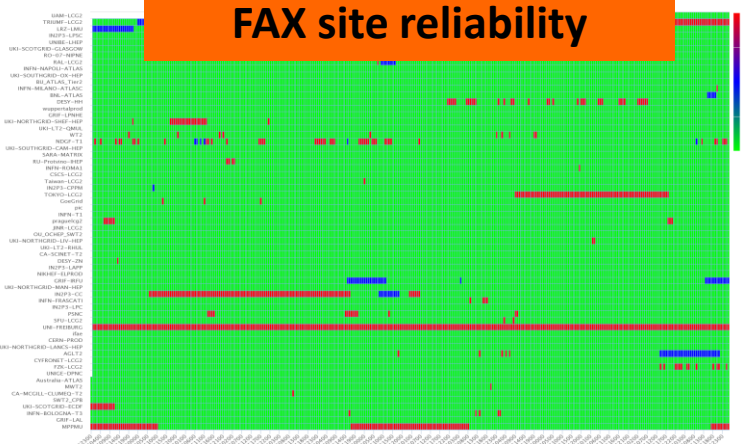
We deployed a Federate Storage Infrastructure: all data accessible from any location

Increase resiliency against storage failures: FAILOVER

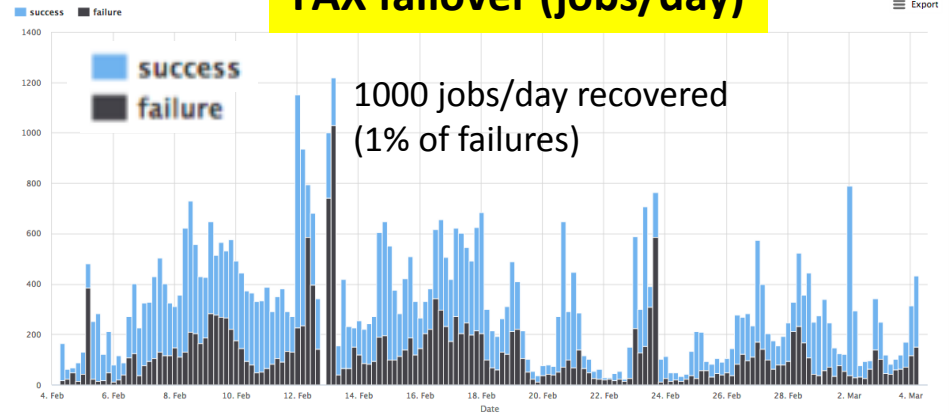
Jobs can run at sites w/o data but with free CPUs: OVERFLOW (up to 10% of jobs)

# Remote data access: FAX

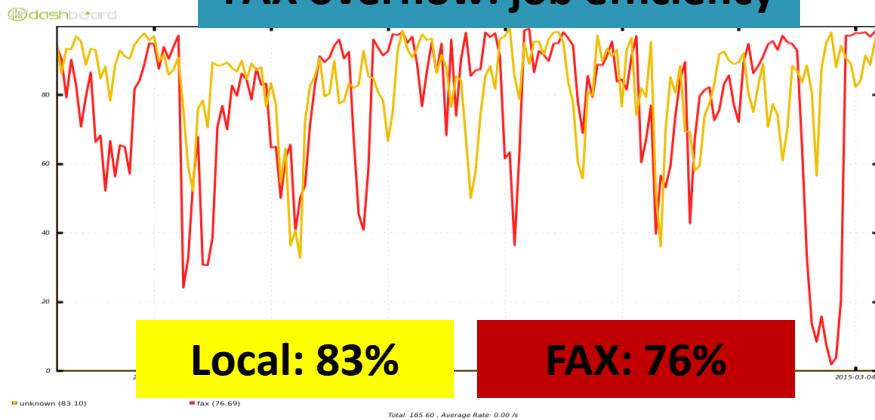
## FAX site reliability



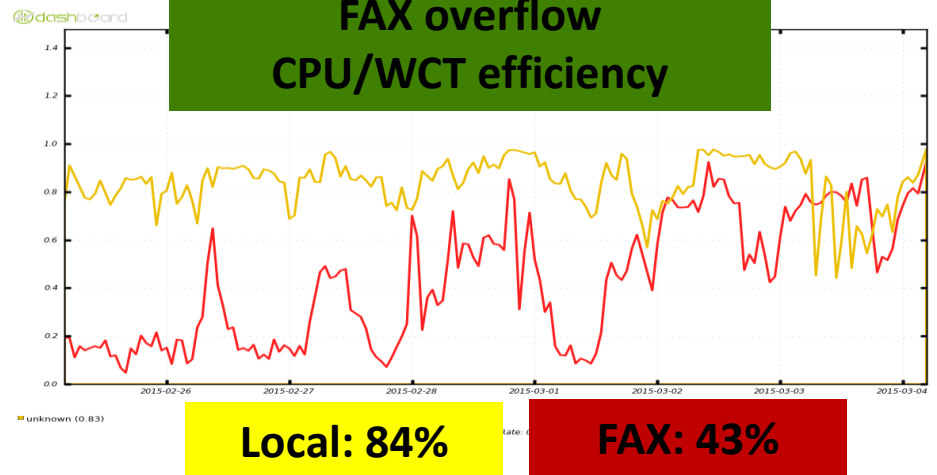
## FAX failover (jobs/day)



## FAX overflow: job efficiency



## FAX overflow CPU/WCT efficiency

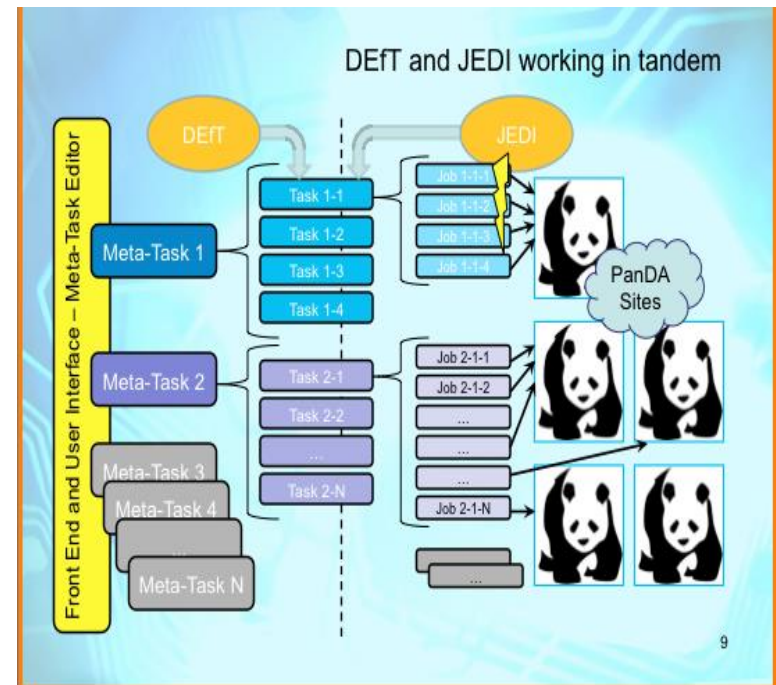




# Distributed Production and Analysis

We developed a new service for simulated and detector data processing: Prodsys-2<sup>[2]</sup>

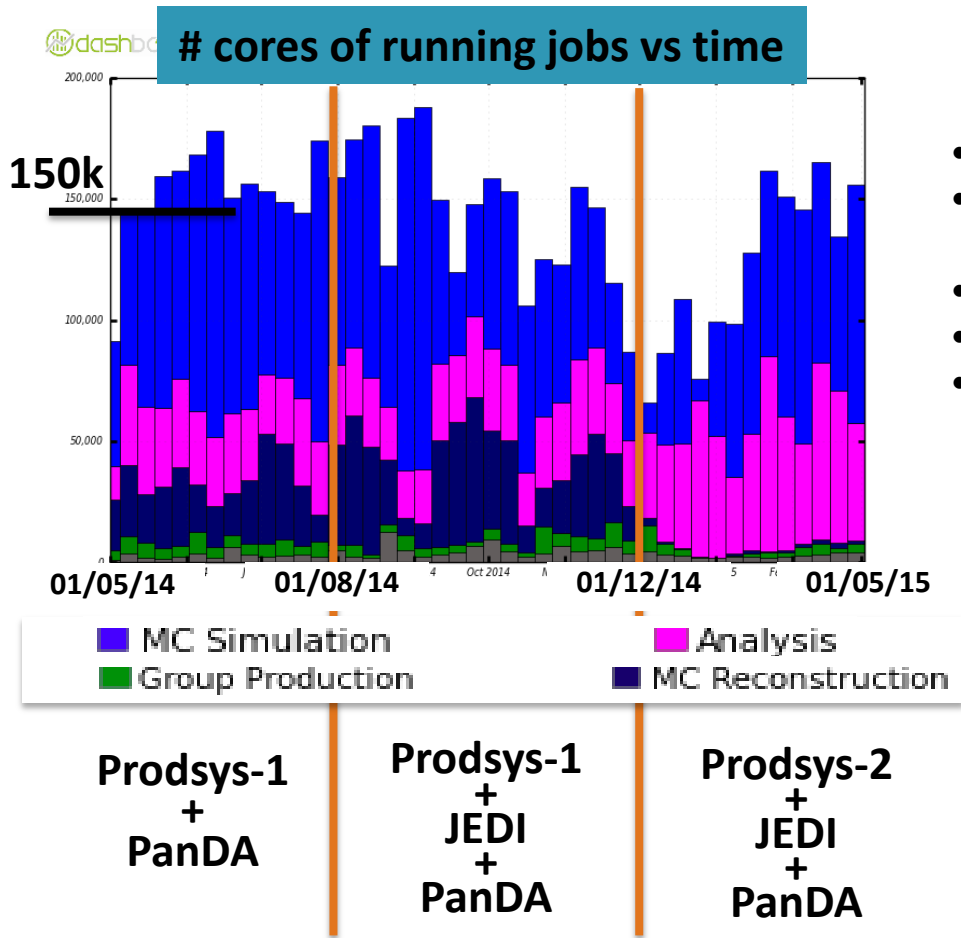
- Prodsys-2 core components
  - Request I/F: allows production managers to define a request
  - DEFT: translates user request into task definitions
  - JEDI: generates the job definitions
  - PanDA: executes the jobs in the distributed infrastructure
- JEDI+PanDA will provide the new framework for Distributed Analysis





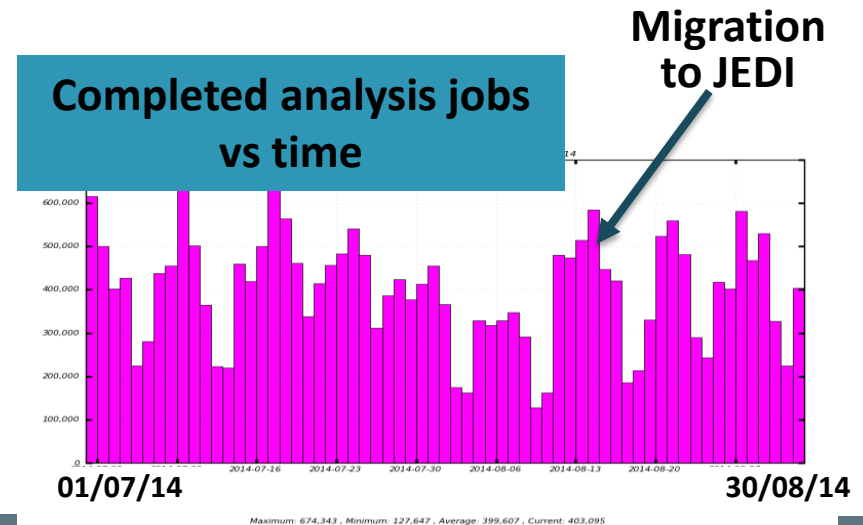
# Prodsys-2 is in production since 1<sup>st</sup> December 2014

## JEDI is in use for analysis since 8<sup>th</sup> August 2014



Prodsys-2 and JEDI offer an extremely large set of improvements

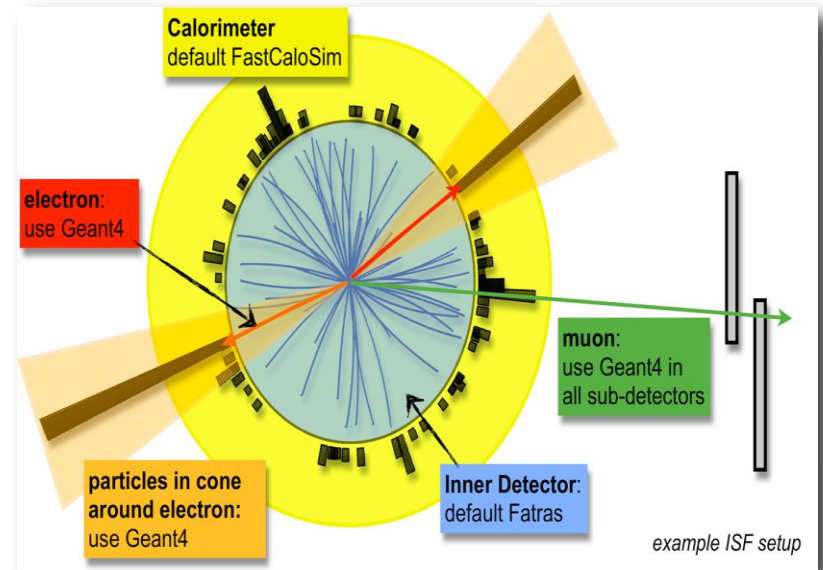
- Built-in file merging capability
- Dynamic job definition optimizing resource scheduling
- Automated recovery of lost data
- Advanced task management interface
- New monitoring



# More efficient utilization of resources

# Simulation

- Simulation is CPU intensive
- Integrated Simulation Framework
  - Mixing of full GEANT & fast simulation within an event
- Work in progress, target is 2016



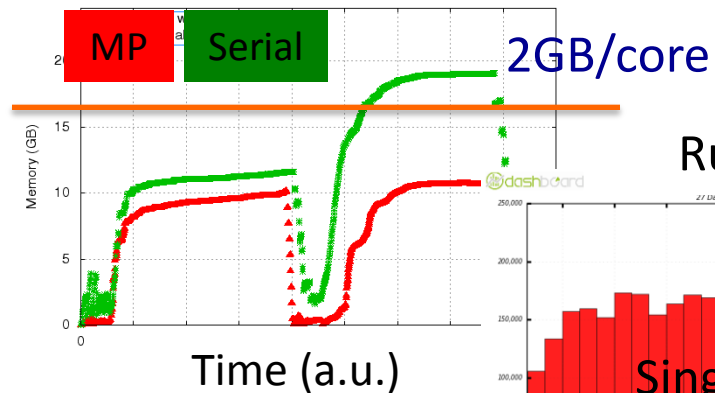
- More events per 12h job, larger output files, less transfers/merging, less I/O
- Or shorter, more granular jobs for opportunistic resources

# Reconstruction

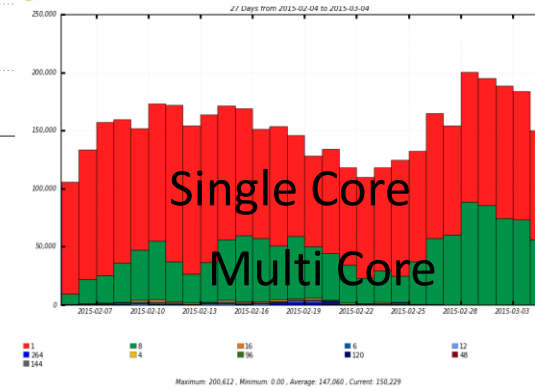
Reconstruction is memory eager and requires non negligible CPU (40% w.r.t. simulation, 20% of ATLAS CPU usage)

AthenaMP<sup>[3]</sup>: multi-processing reduces the memory footprint

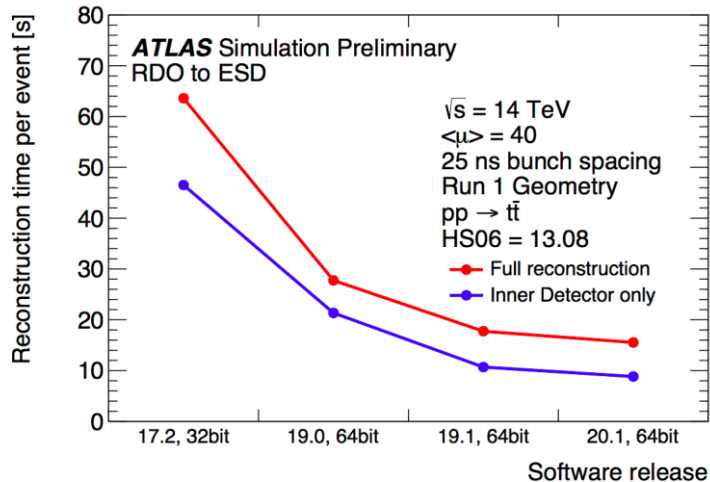
Athena memory Profile



Running Jobs



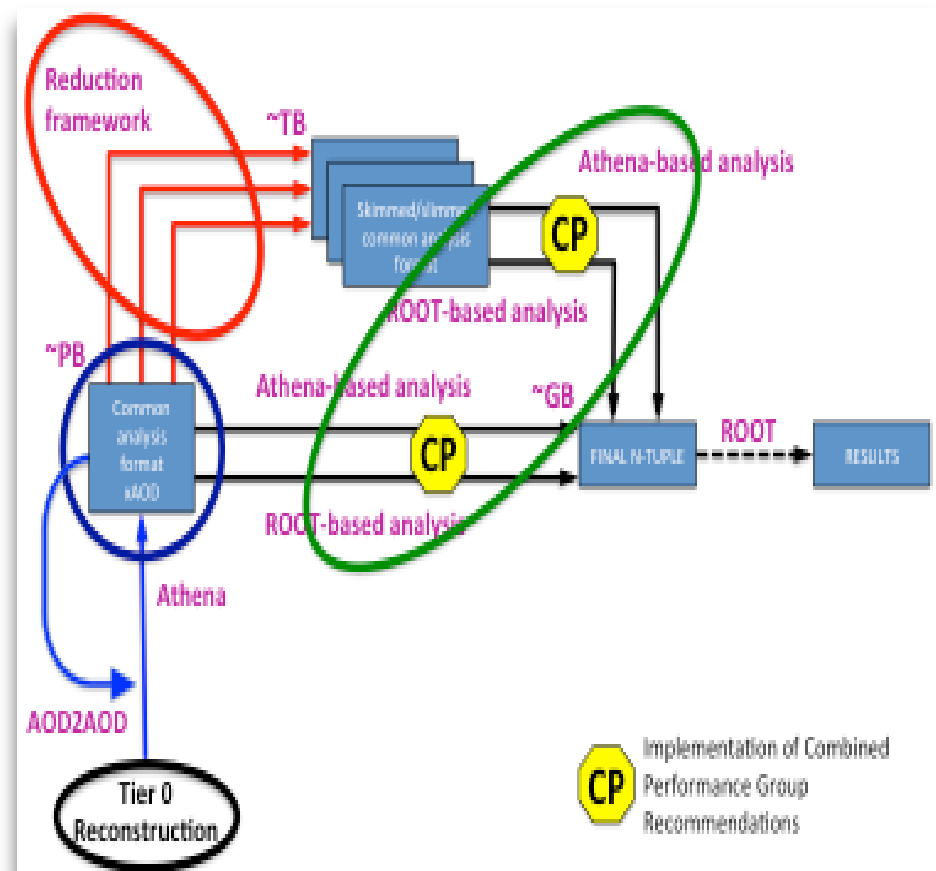
Reconstruction Time (s/event)



Code and algorithms optimization largely reduced CPU needs in reconstruction<sup>[4]</sup>

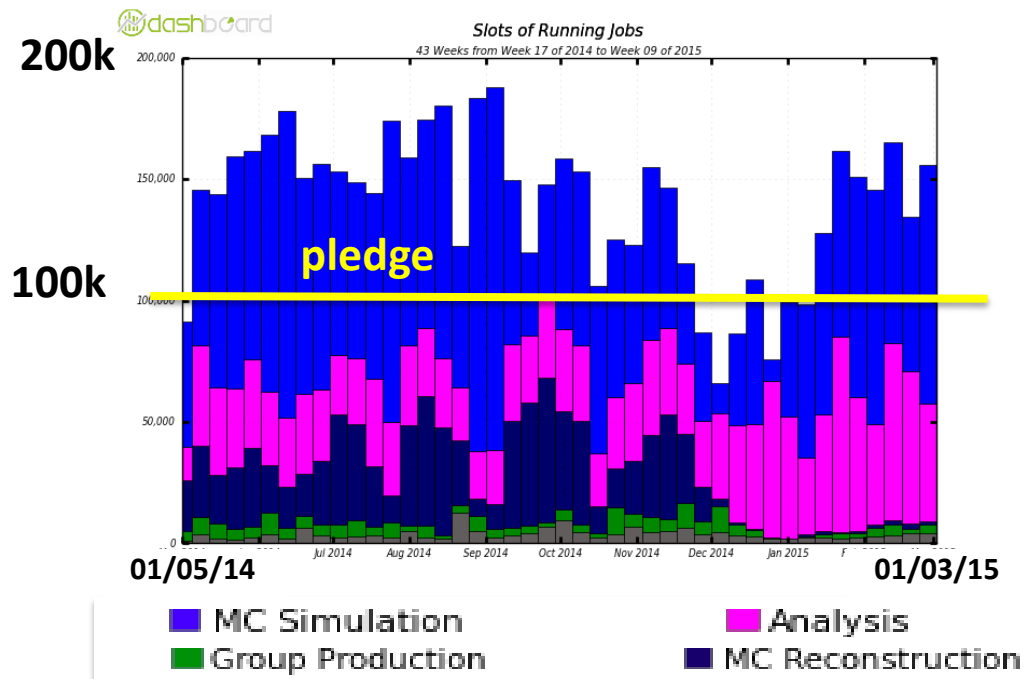
# Analysis Model

- Common analysis data format: xAOD
  - replacement of AOD & group ntuple of any kind
  - Readable both by Athena & ROOT
- Data reduction framework<sup>[5]</sup>
  - Athena to produce group derived data sample (DxAOD)
    - Centrally via Prodsys
  - Based on train model
    - one input, N outputs
    - from PB to TB



# Leveraging opportunistic resources

# of cores for ATLAS running jobs



Almost 50% of ATLAS production at peak rate relies on opportunistic resources

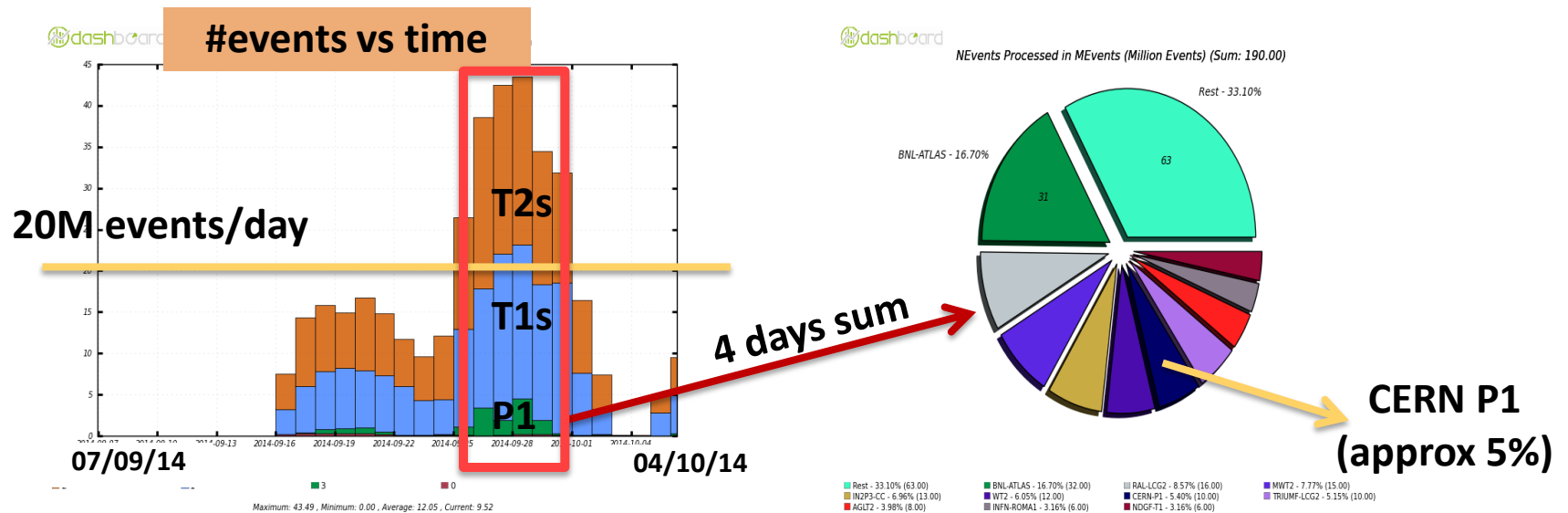
Efficient utilization of the largest variety of opportunistic resources is vital for ATLAS

Enabling utilization of non-Grid resources is a long term investment (beyond opportunistic use)

# (Opportunistic) Cloud Resources

We invested a lot of effort in enabling usage of Cloud resources<sup>[6]</sup>

The ATLAS HLT farm at the CERN ATLAS pit (P1) for example was instrumented with a Cloud interface in order to run simulation: Sim@P1<sup>[7]</sup>



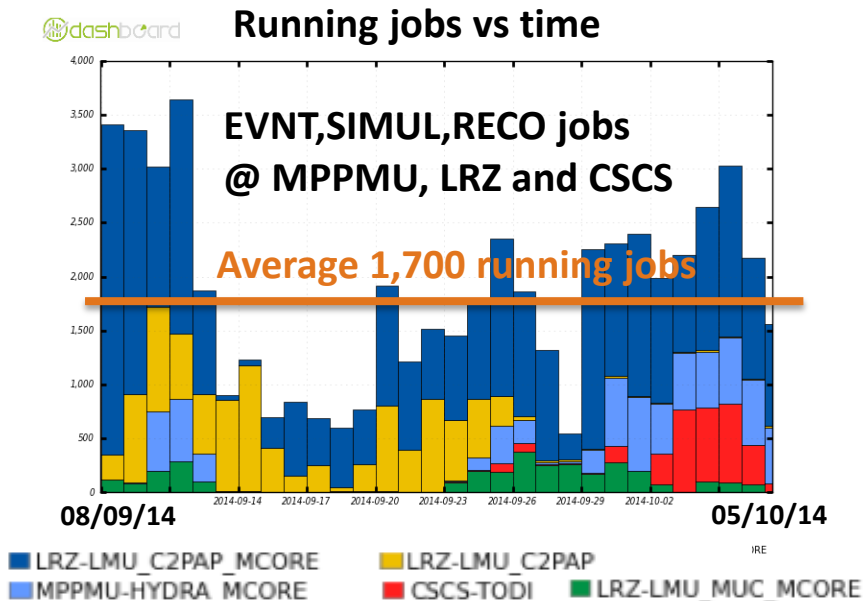
The HLT farm was dynamically reconfigured to run reconstruction on multicore resources (Reco@P1). We expect to be able to do the same with other clouds



# HPCs

High Performance Computers were designed for massively parallel applications (different from HEP use case) but we can parasitically benefit from empty cycles that others can not use (e.g. single core job slots)

The ATLAS production system has been extended to leverage HPC resources<sup>[8]</sup>



24h test at Oak Ridge Titan system (#2 world HPC machine, 299,008 cores). ATLAS event generation: 200,000 CPU hours on 90K parallel cores (equivalent of 70% of our Grid resources)

Mira@ARGONNE: Sherpa Generation using 12244 nodes with 8 threads per node, so 97,952 parallel Sherpa processes.

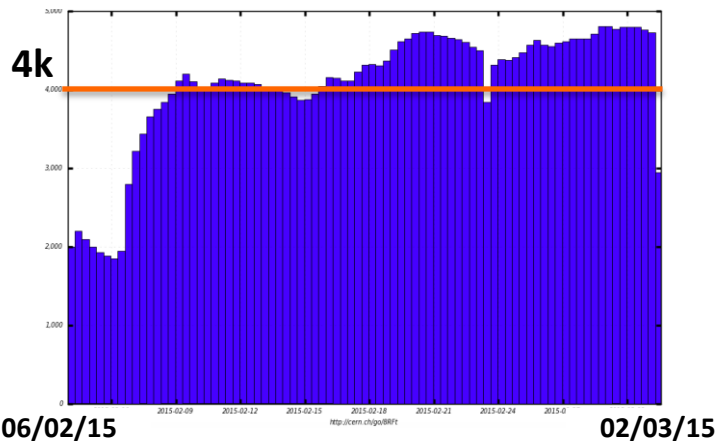
The goal is to validate as many workflows as possible.  
Today approximately 5% of ATLAS production runs on HPCs

# Volunteer Computing

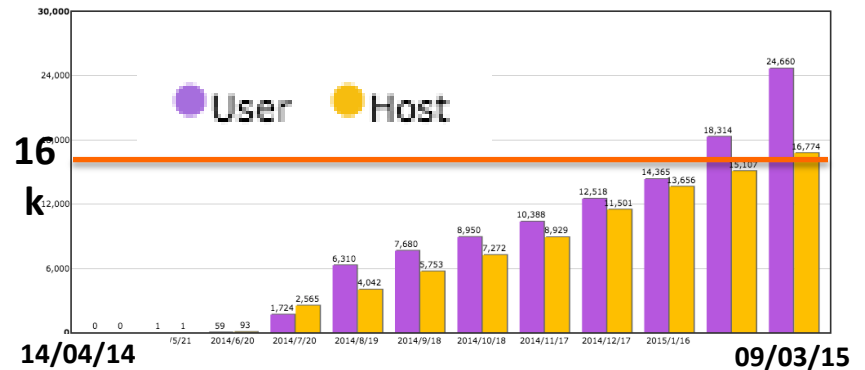
Enabling users laptops and desktops to run ATLAS simulation<sup>[9]</sup>



# running jobs vs time



# users/hosts vs time



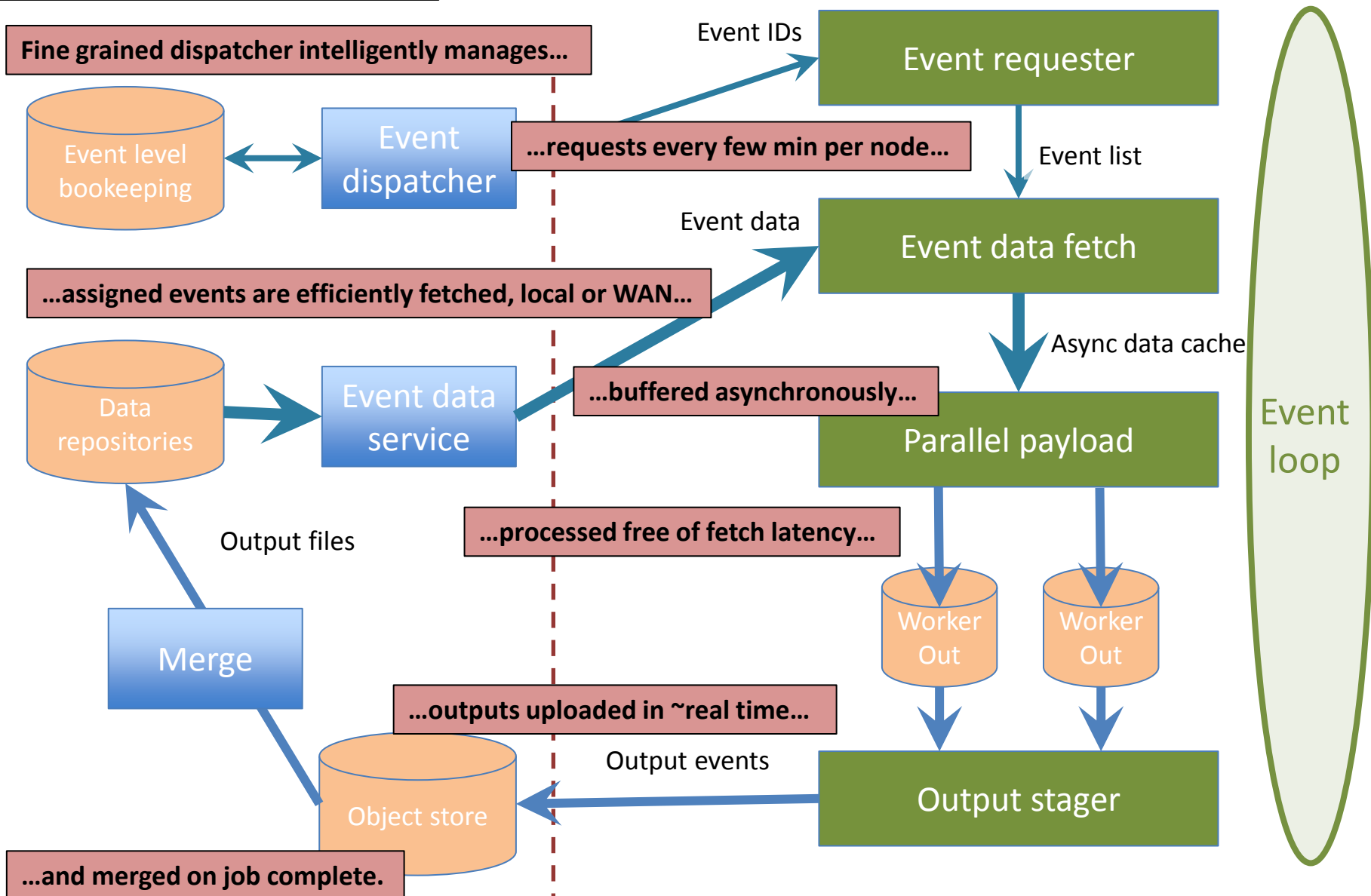
# Event Service

**Efficient utilization of opportunistic resources implies short payloads  
(get out quickly from the resources if the owner needs it)**

**We developed a system to deliver payloads as short as the single event:  
the **Event Service**<sup>[10]</sup>.**

**The Event Service will be commissioned during 2015**

# Event Service Schematic



# **New data lifecycle management model**

**a.k.a.**

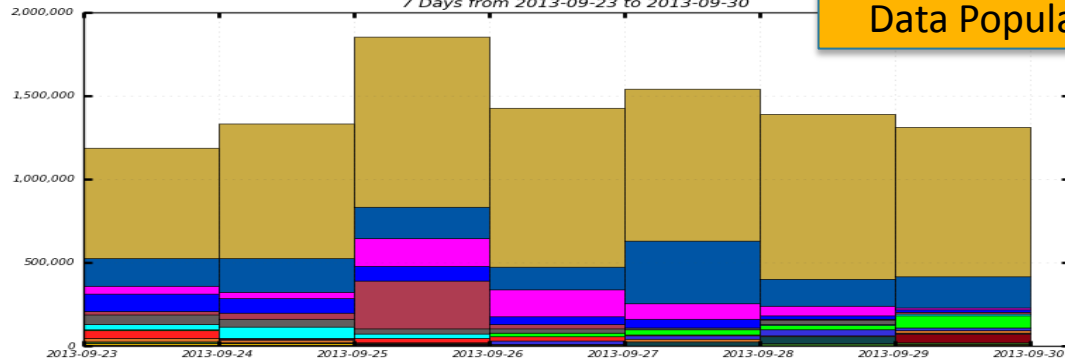
**“you can get unpledged CPU but not so much unpledged disk”**

# Dynamic Data Replication and Reduction

Dynamic Replication

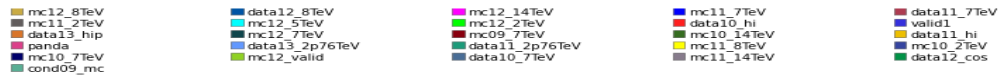
dashboard

Number of files accessed  
7 Days from 2013-09-23 to 2013-09-30



Data Popularity

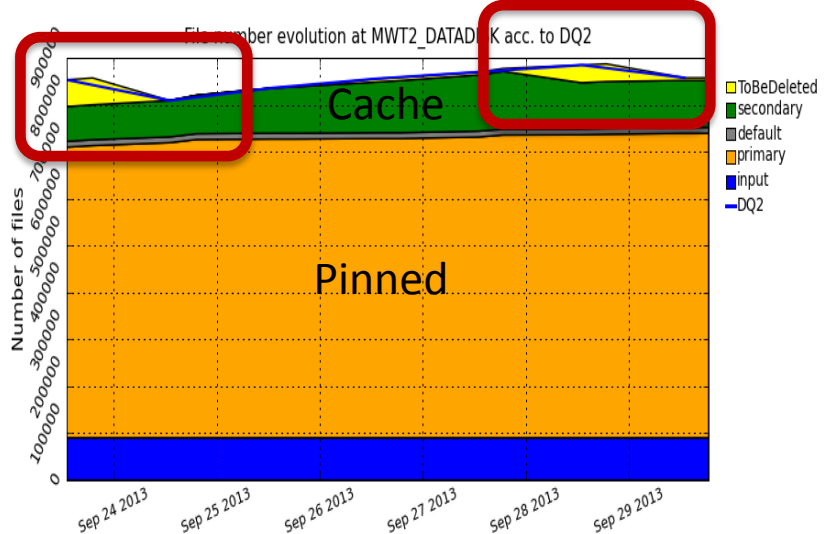
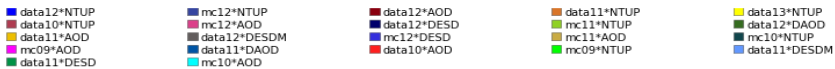
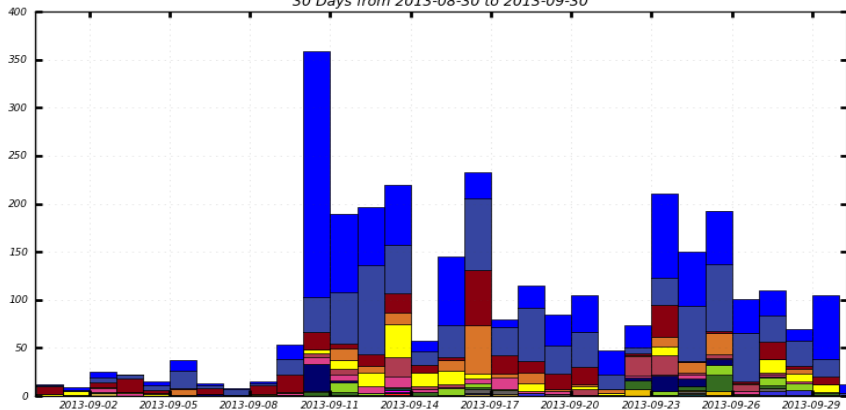
Dynamic Reduction



Maximum: 1.854,082, Minimum: 1.185,875, Average: 1.435,492, Current: 1.312,328

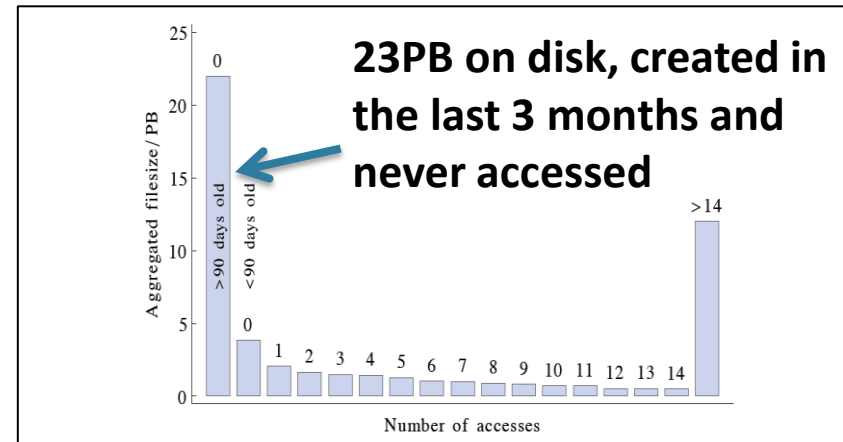
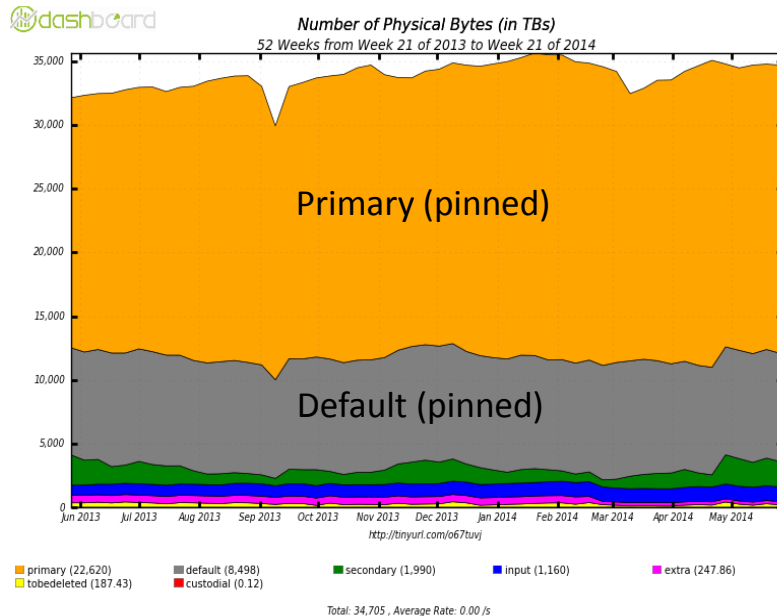
dashboard

Number of Physical Bytes (in TBs)  
30 Days from 2013-08-30 to 2013-09-30



# 18 months ago ...

## Disk occupancy at T1s vs time



8 PB of data on disk never been touched

- T1 dynamically managed space (**green**) is unacceptably small
  - It compromises our strategy of dynamic replication and cleaning of popular/unpopular data
- Large fraction of primary space is occupied by old and unused data



# The new data lifecycle model

- Every dataset has a lifetime set at creation
  - The lifetime can be infinite (e.g. RAW data)
  - The lifetime can be extended if the dataset is accessed
- Datasets with expired lifetime can disappear at any time from disk and tape.
- ATLAS Distributed Computing flexibly manages data replication and reduction, within the boundaries of lifetime and retention
  - Increase/reduce the number of copies based on data popularity
  - Re-distribute data at T2s rather than T1s and viceversa
  - Move data to tape and free up disk space

# Implications of the model

- We will use more tapes
  - Access to tape remains “centralized”
- For the first time we will “delete” tapes
- In the steady flow we will approximately delete as much as we will write
- Access through storage backdoors is today not accounted
  - We will improve this, but most people use official tools (PanDA/Rucio)



# Conclusions

- A lot of hard work went in preparing the ATLAS Software and Computing for Run-2
  - A balanced mixture between evolution and revolution
- Commissioning of new systems was carried on in non disruptive manner
- Our systems are ready for the new challenges
  - Still we have not yet explored many new capabilities

# References to relevant ATLAS contributions

[1] CHEP ID 205 - The ATLAS Data Management system - Rucio: commissioning, migration and operational experiences (Vincent Garonne)

[2] CHEP ID 100 - Scaling up ATLAS production system for the LHC Run 2 and beyond: project ProdSys2 (Alexei Klimentov)

[3] CHEP ID 165 - Running ATLAS workloads within massively parallel distributed applications using Athena Multi-Process framework (AthenaMP) ([Vakhtang Tsulaia](#))

[4] CHEP ID 147 - Preparing ATLAS Reconstruction for LHC Run 2 (Jovan Mitrevski)

[5] CHEP ID 164 - New Petabyte-scale Data Derivation Framework for ATLAS (James Catmore)

[6] CHEP ID 146 - Evolution of Cloud Computing in ATLAS (Ryan Taylor)

# References to relevant ATLAS contributions

[7] CHEP ID 169 - Design, Results, Evolution and Status of the ATLAS simulation in Point1 project (Franco Brasolin)

[8] CHEP ID 92 - ATLAS computing on the HPC Piz Daint machine (Michael Arthur Hostettler)

[8] CHEP ID 153 - Bringing ATLAS production to HPC resources - A use case with the Hydra supercomputer of the Max Planck Society (Luca Mazzaferro)

[8] CHEP ID 152 - Integration of PanDA workload management system with Titan supercomputer at OLCF (Sergey Panitkin)

[8] CHEP ID 140 - Fine grained event processing on HPCs with the ATLAS Yoda system (Vakhatang Tsulaia)

[9] CHEP ID 170 - ATLAS@Home: Harnessing Volunteer Computing for HEP (David Cameron)

[10] CHEP ID 183 - The ATLAS Event Service: A new approach to event processing (Torre Wenaus)