# *High Performance Computing (i.e. supercomputing) in ATLAS*

*ATLAS Sites Jamboree*
*January 27, 2016*

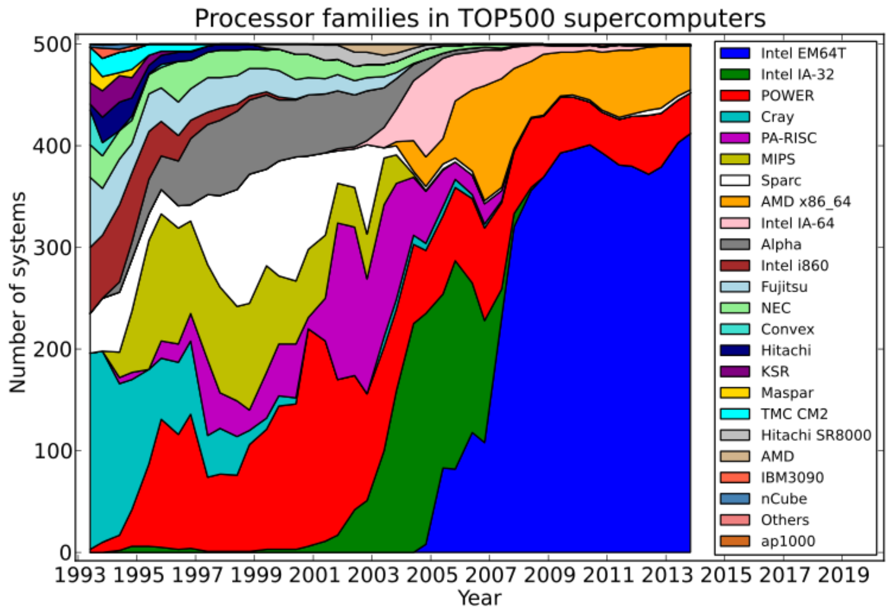
**Alexei Klimentov, BNL**
*for HPC Working Group*

# Thanks

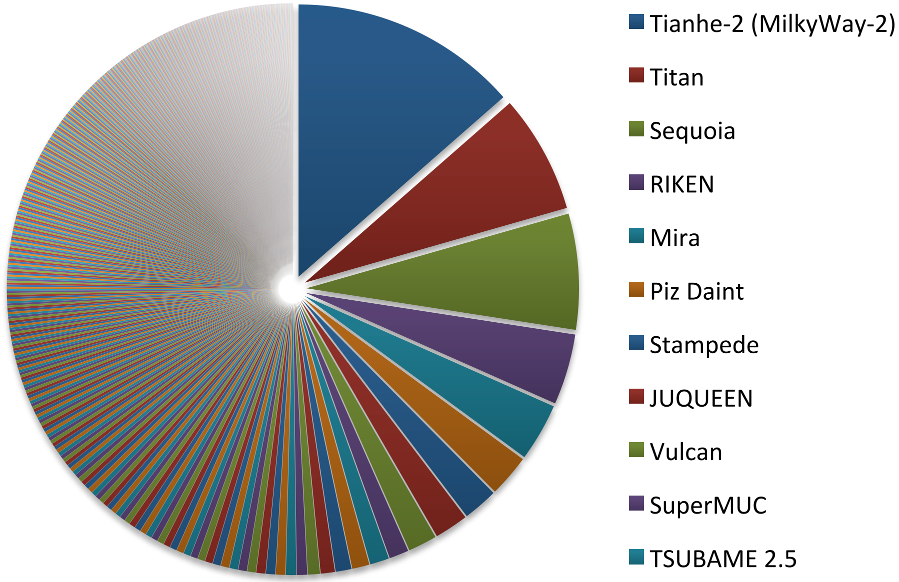
- D.Benjamin, T.Childers, K.De, R.Konoplich, D.Krasnopevtsev, M.Lassnig, T.Le Compte, T.Maeno, R.Mashinistov, P.Nilsson, D.Oleynik, S.Panitkin, H.Severini, V.Tsulaia, V.Velikhov…
  - For slides, materials and comments

# Outline

- Supercomputers
  …and ATLAS Distributed Computing
- Distributed SW major highlights related to HPC integration
- Major (but not ALL) activities at Leadership Class Facilities and SC
  – NERSC (and Cori)
  – Mira
  – Edison
  – Titan
  – SC@NRC-KI
    - *See Rod's talk for EU activities*
    - *See Wen's talk for Event Service & Yoda progress*
- University and detectors groups feedback to HPC activities
- GPUs : possible ML applications
- Summary and Conclusion

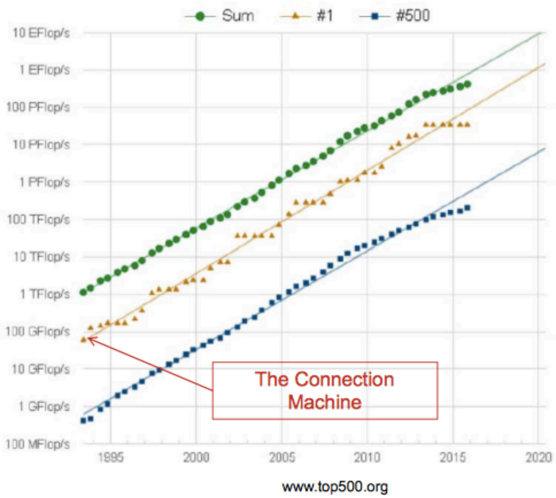



# Top 500




Seymour Cray :
"supercomputer, it is hard to define, but you know it when you see it"

## Top500 system performance evolution

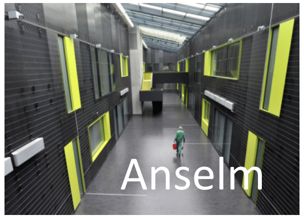



Performance doubling period on average:

No 1 – 13.64 months

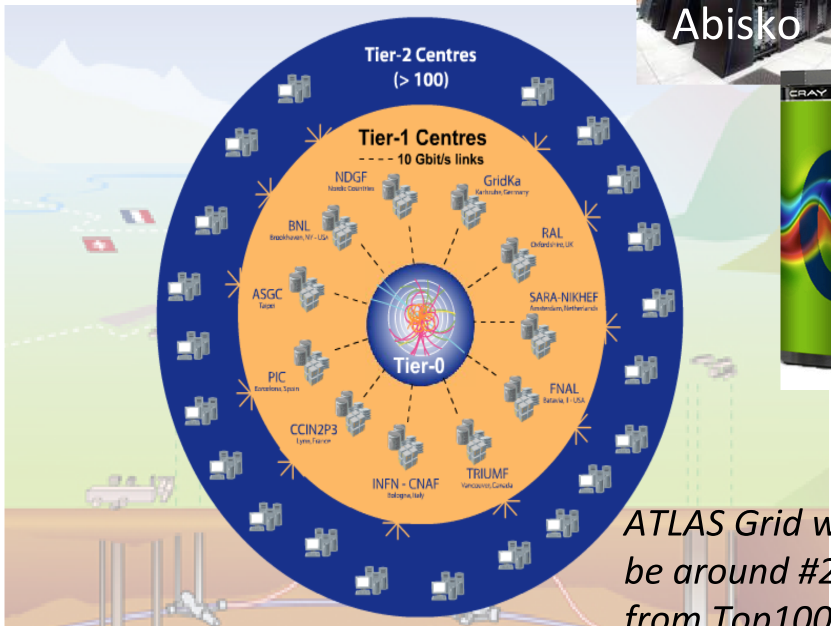
No 500 – 12.90 months

*Large HPCs use a variety of architecture*
*Half of computational power is concentrated in a small number of machines;*
*Small HPCs use x86 architectures. Typically, these are ordinary server racks, with Infiniband interconnects. 94% of the bottom 400 of the Top 500 (including the last 130) are all x86*



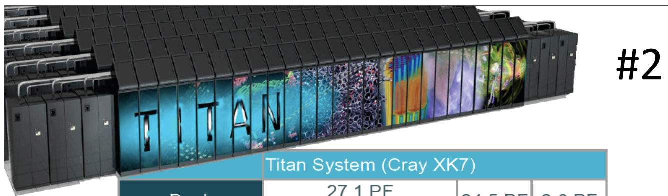





*ATLAS Grid would be around #27 from Top100*








| Titan System (Cray XK7) | | | |
|---|---|---|---|
| Peak Performance | 27.1 PF 18,688 compute nodes | 24.5 PF GPU | 2.6 PF CPU |
| System memory | 710 TB total memory | | |
| Interconnect | Gemini High Speed Interconnect | 3D Torus | |
| Storage | Lustre Filesystem | 32 PB | |
| Archive | High-Performance Storage System (HPSS) | 29 PB | |
| I/O Nodes | 512 Service and I/O nodes | | |

12 OLCF 20



***The ATLAS collaboration have members with access to these machines and to many others…***

# Interfacing Supercomputers to ATLAS Distributed Computing

- Cluster-like HPC
    - x86 cores
    - Worker-node TCP/IP connectivity
    - Small minimum partition sizes (sometimes one core)
  - The plan is to treat them as clusters and have them join the Grid
    - Since many new HPC sites are not grid sites
      - need a cost benefit analysis
      - large task manually or semi-automatically submitted needs proper registering outputs (interaction with Rucio)
      - Jobs can be queued for long time
  - Many successful stories, already in production and analysis for OU_OSCER (Oklahoma U) and NorduGrid clusters. Others in (pre)production and validation for production and analysis
  - No issues (but Operational) for integration and considering them as opportunistic resource

# Interfacing Supercomputers to ATLAS Distributed Computing. Cont'd

- Supercomputers
  - Many of them (the largest) are not x86 machines
    - The code requires at minimum to recompile
  - SSH-like access by members of the collaboration
  - Integrated with Production System. Jobs submission via
    - PanDA
    - aCT/ARC-CE

    'transparent to users'
  - Pre-installed ATLAS SW releases
    - Some issues with ATLAS releases installation on CVMFS @ Titan
  - many supercomputers in EU (Hydra, SuperMUC, CSCS) are running main production workflows : Full Simulation, Reco, as well as evgen
  - Many machines can be used today to offload Grid (if needed) and to run
    - Alpgen, Pythia, Sherpa, Powheg, MadGraph
  - Important milestone has been reached in October 2015,Titan was integrated with the ATLAS Production System

# Fundamental Questions

- How to get time on supercomputers ?
- How to interface supercomputers to ATLAS Distributed Computing ?
- How to run ATLAS code on supercomputers and how to do it efficiently ?

# Supercomputers Resource Allocation

- Resource allocation is very competitive
  - Many strong technical and scientific cases for the time
  - Allocation given to projects
    - Typically small group of people

  …and we are very successful in getting it
    - ~70M hours allocation in 2014-2015
      - ~8% of ATLAS Grid use and ~50% of our Monte-Carlo event generators
- Leadership Class Facilities (LCF) Usage in 2015 : NERSC, Mira, Titan

  Overall that the LCF community delivered more than ~99M grid-equivalent hours
    - ATLAS used 81.3M hours on Mira
    - ATLAS used ~5M hours on Titan
    - ATLAS used ~5M hours at NERSC
- ALCC proposal for 2016 is in preparation (Taylor, Tom et al)

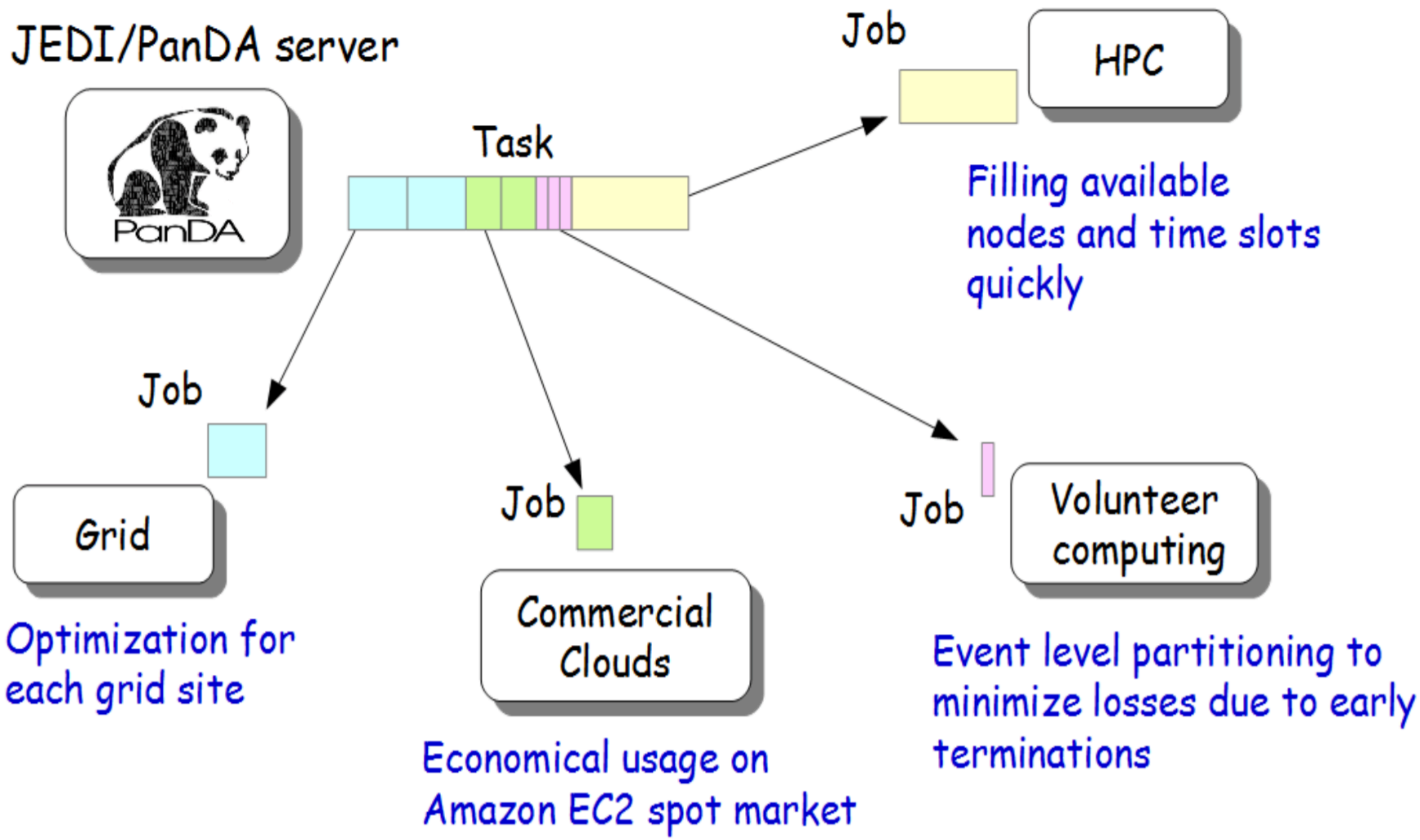
*There is an interest from HPC community in us.*

- *Common proposal and projects under DOE ASCR umbrella (BNL, UTA, ORNL, Rutgers University)*
- *Many invited talks/demos at the Supercomputing Conferences SC14 (New Orleans), Special PanDA booth at the SC15 (Austin, TX)*
  - *HENP applications on Titan.*
  - *Granular data processing on HPCs using an Event Service*
  - *Integrating Network Services with PanDA*
  - *Intelligent Networks*
- *Access to GPUs @ NRC-KI in 2016 for ML applications*

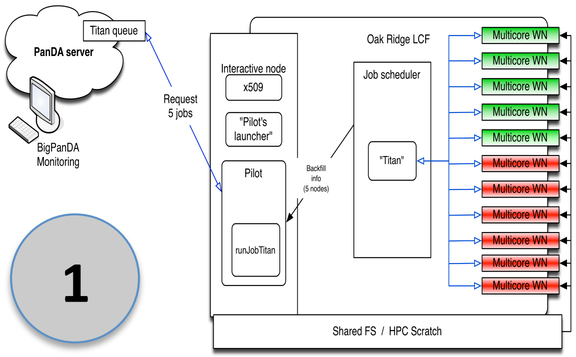
*ALCC – DOE ASCR Leadership Computing Challenge*
*ASCR – Advanced Scientific Computing Research*

# SW Highlights.

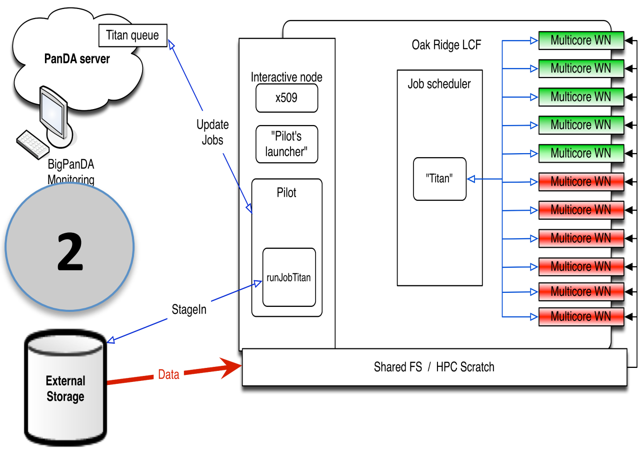
## *PanDA  & Heterogeneous Computing Resources*



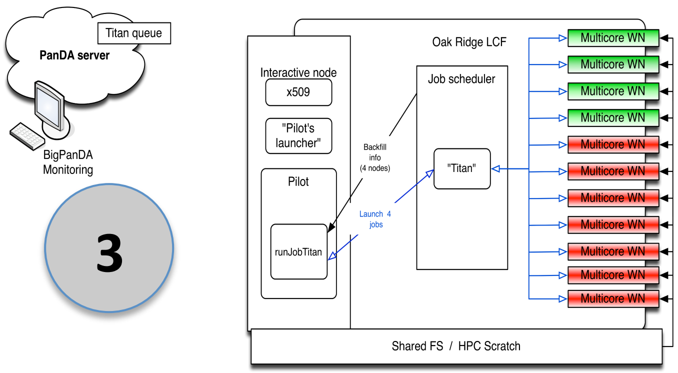

T.Maeno



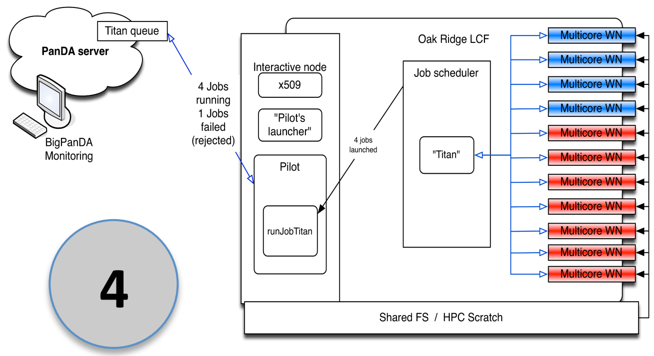

*Pilot requests information about available resources (backfill)*
*According to available resources, pilot requests jobs from PanDA server*



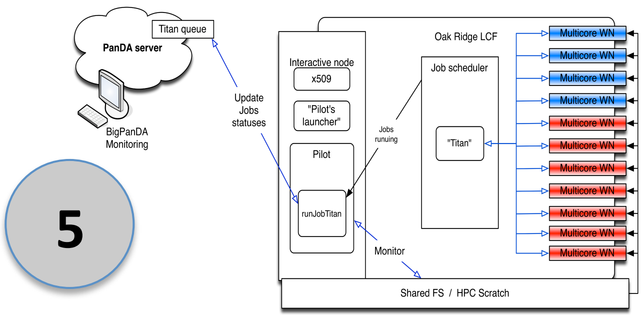

*Pilot stage-in and prepares environment before launching*
*Jobs updates with initial info (space report etc.)*



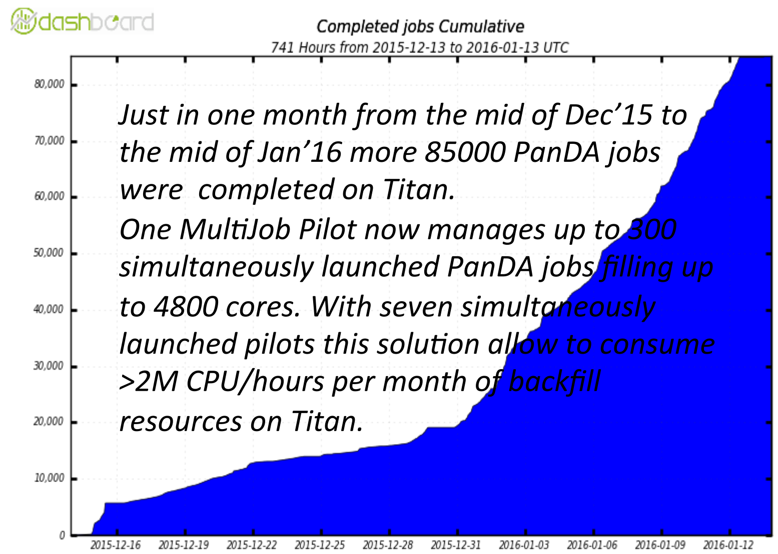

*Request availble backfill (situation may change since pilot started)*
*Adjust number of jobs for launching*




*Pilot updates jobs sub status, as only jobs runs on HPC*




*Pilot monitors available space, output and sends hearbeat during payload execution*




*Just in one month from the mid of Dec'15 to the mid of Jan'16 more 85000 PanDA jobs were completed on Titan.*
*One MultiJob Pilot now manages up to 300 simultaneously launched PanDA jobs filling up to 4800 cores. With seven simultaneously launched pilots this solution allow to consume >2M CPU/hours per month of backfill resources on Titan.*

Average Rate: 0.03 /s

## SW Highlights. PanDA multijob pilot workflow, as it implemented on Titan

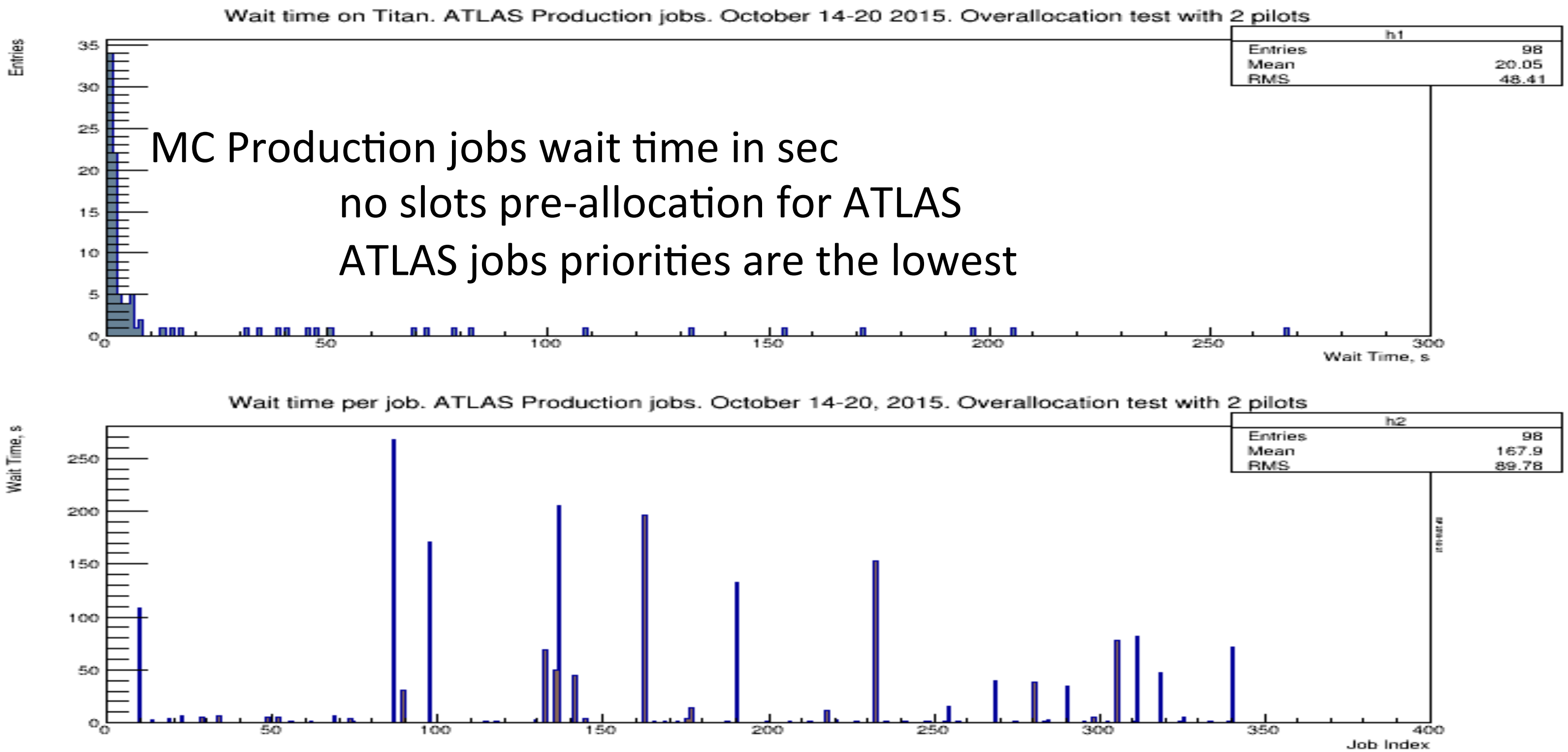
D.Oleynik et al

# Supercomputers Resource Allocation. Running ATLAS Payload in Backfill Mode

Several demonstrators on Titan in 2014-2015.

- Were able to collect ~ 200,000 core hours
- Max number of nodes per job – 5835 (93360 cores)
  - Close to 75% ATLAS Grid in size!
- Used ~14.4% of Titan free core hours

PanDA has potential to generate 300M hours per year on Titan



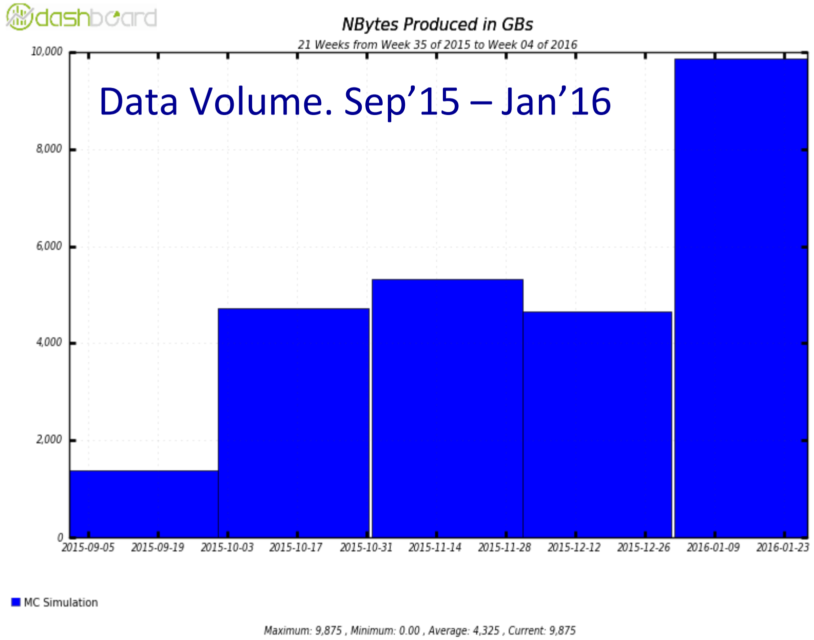

MC Production jobs wait time in sec

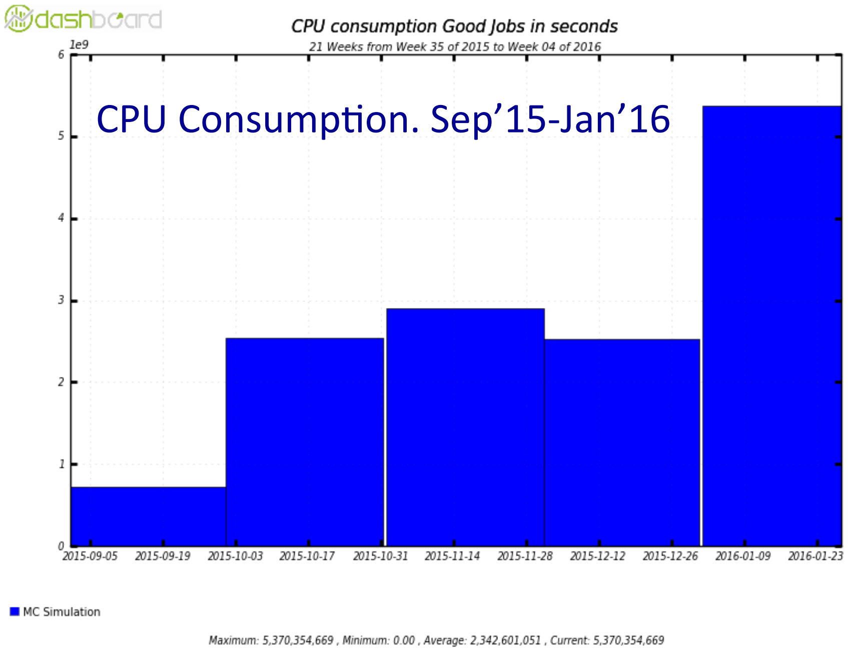
no slots pre-allocation for ATLAS

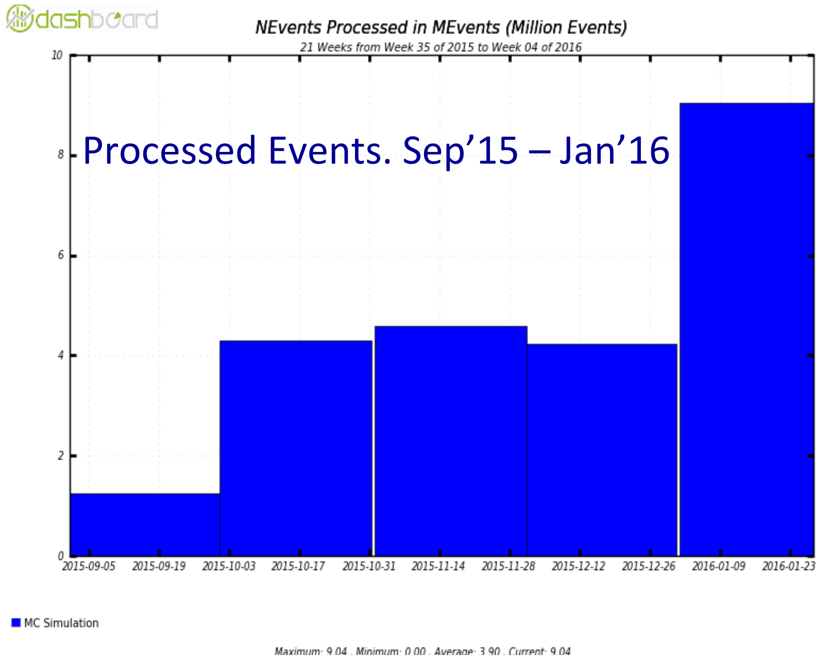
ATLAS jobs priorities are the lowest

S.Panitkin, D.Oleynik et al


dashboard
Successful Jobs
21 Weeks from Week 35 of 2015 to Week 04 of 2016
Running Successful Jobs. Sep'15-Jan'16
MC Simulation
Maximum: 90,644 , Minimum: 0.00 , Average: 43,973 , Current: 90,644


dashboard
NBytes Produced in GBs
21 Weeks from Week 35 of 2015 to Week 04 of 2016
Data Volume. Sep'15 – Jan'16
MC Simulation
Maximum: 9,875 , Minimum: 0.00 , Average: 4,325 , Current: 9,875


dashboard
CPU consumption Good Jobs in seconds
21 Weeks from Week 35 of 2015 to Week 04 of 2016
CPU Consumption. Sep'15-Jan'16
MC Simulation
Maximum: 5,370,354,669 , Minimum: 0.00 , Average: 2,342,601,051 , Current: 5,370,354,669


dashboard
NEvents Processed in MEvents (Million Events)
21 Weeks from Week 35 of 2015 to Week 04 of 2016
Processed Events. Sep'15 – Jan'16
MC Simulation
Maximum: 9.04 , Minimum: 0.00 , Average: 3.90 , Current: 9.04

# OLCF Titan.

**Titan (internal) statistic about allocation consumption.**

| Year-Month | Titan CPU/hours | CPU consumption (dashboard) | Wall Clock consumption (dashboard) |
|---|---|---|---|
| 2015-09 | 1 766 780 | 200 724 | 1 355 243 |
| 2015-10 | 2 989 861 | 709 420 | 1 890 663 |
| 2015-11 | 2 848 813 | 916 219 | 2 308 317 |
| 2015-12 | 2 147 640 | 719 289 | 2 609 463 |
| 2016-01 | 4 109 746 | 1 499 459 | 3 439 029 |

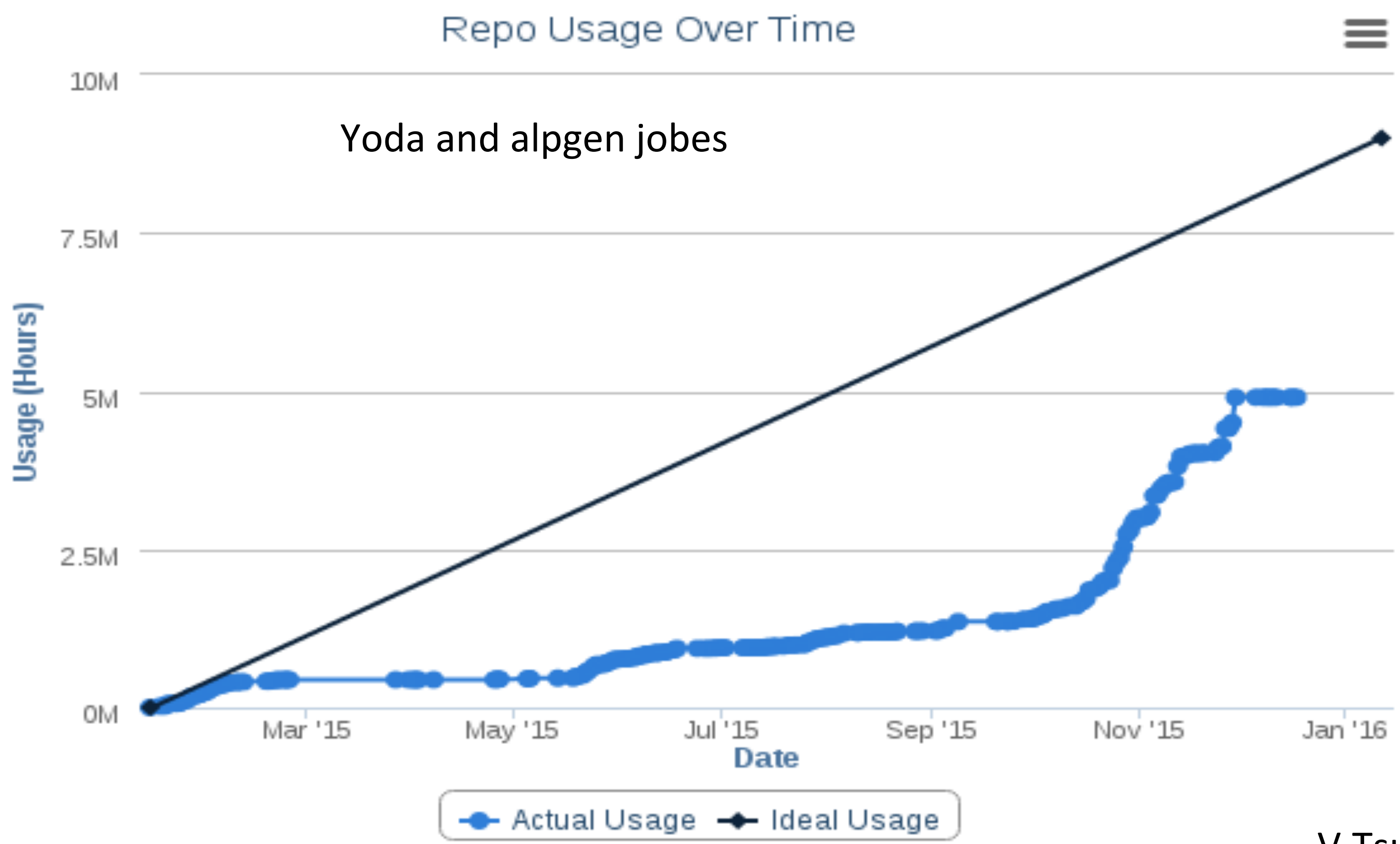
- Integrated with ProdSys2 and monitoring
- Storage cache is Lustre at OLCF
  - destination ATLAS SE is BNL.
- Data are shipped to BNL
- ATLAS SW releases installation
  - In order to run ATLAS production workloads, we need to install and maintain official software releases and corresponding databases on Titan. The ATLAS software releases have been distributed using the pacman package manager. Usage of native CVMFS was not possible because the FUSE module was missing on the Titan kernel modules, while CVMFS and Parrot proved not to be a stable installation.
- ANY ATLAS MC payload can be run @Titan
  - as soon as Titan is validated, we want to run full "online" mode (not special tasks in "brokeroff")
  - validation is the real bottleneck for better HPC utilization, Titan still waiting after 6 months
- 2016 plans
  - BigPanDA project is approved for Titan
  - we plan to run on Titan in pure backfill mode and no allocation.
    - Primary running mode since Oct 2015.
      - ~1.7M wall hours and ~70k sim. jobs per month
    - 2M hours/month in 2016
      - Plan to increase the number (currently 7) of simultaneously running pilots, in order to improve resource collection efficiency
      - Plan to increase IO efficiency
      - Multi-job pilot (new developers joined 'pilot team')

# G4 Simulation at NERSC

- Yoda running in production mode since mid October 2015
  - ~2.5M CPU-hours in Oct-Nov 2015, ~0.5M CPU-hours in January
  - More details about the status of Yoda in Wen's talk
- Athena ported to Cori
  - Smooth transition, no issues observed
  - Batch scripts migrated from Torque to SLURM
- ATLAS is participating in the Cori Burst Buffer Early Users program
  - Currently focused on the scalability of Athena initialization with number of concurrent starts
  - Discovered few problems with Burst Buffer
    - Either already fixed or currently being looked at by Burst Buffer experts from NERSC and Cray

V.Tsulaya

# NERSC. 2015




Yoda and alpgen jobes

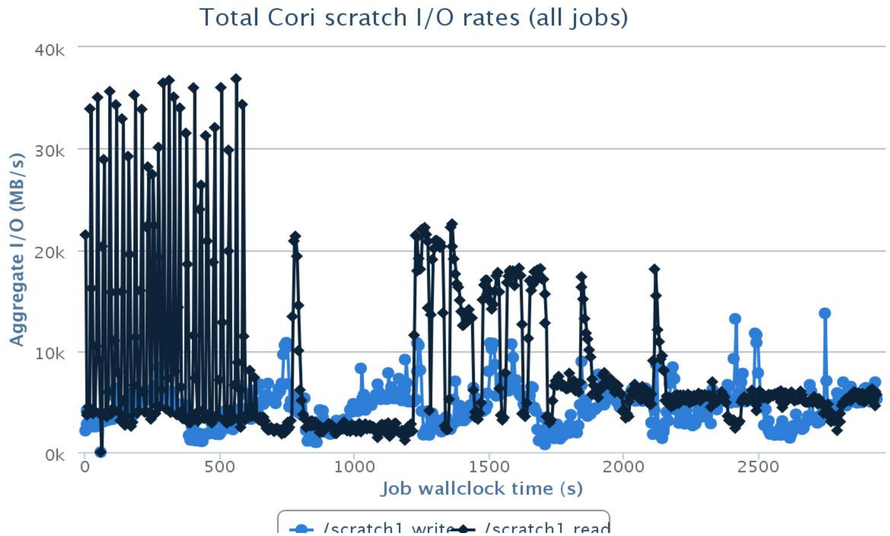
V.Tsulaya

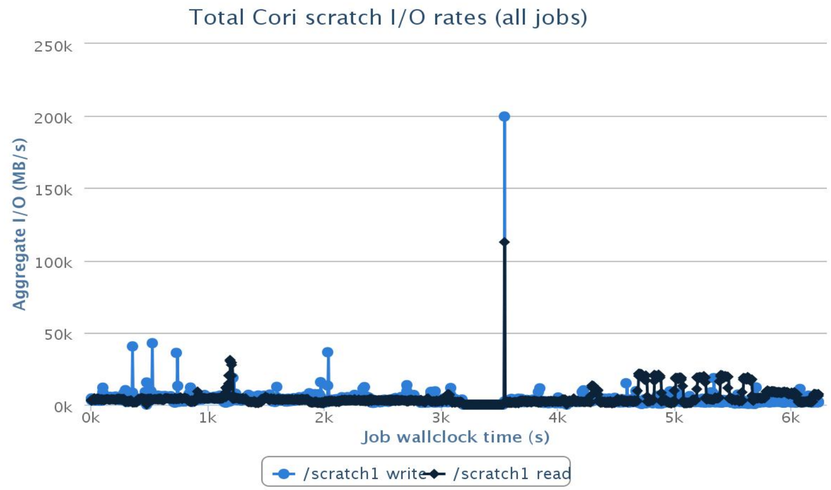
# Derivation Production Tests on Cori

- ATLAS derivation release 20.1.9.4
  - Manually installed
- Input locally stored (mc15 AOD 240GB)

mc15_13TeV:mc15_13TeV361639.MadGraphPythia8EvtGen_A14NNPDF23LO_Ztautau_lowMll_Np1.merge.AOD.e4442_s2608_s2183_r7326_r6282

- Output TOP Group : DAOD_TOPQ1
  - # events : 796600
  - 57 GB
  - 130.7 core-hours (to get unmerged output)
  - 110.7h to merge files

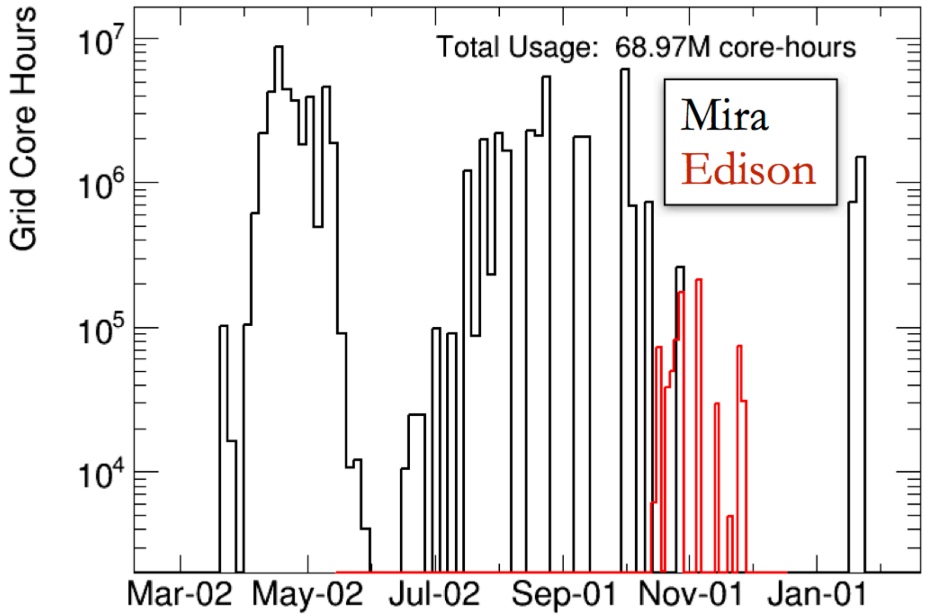


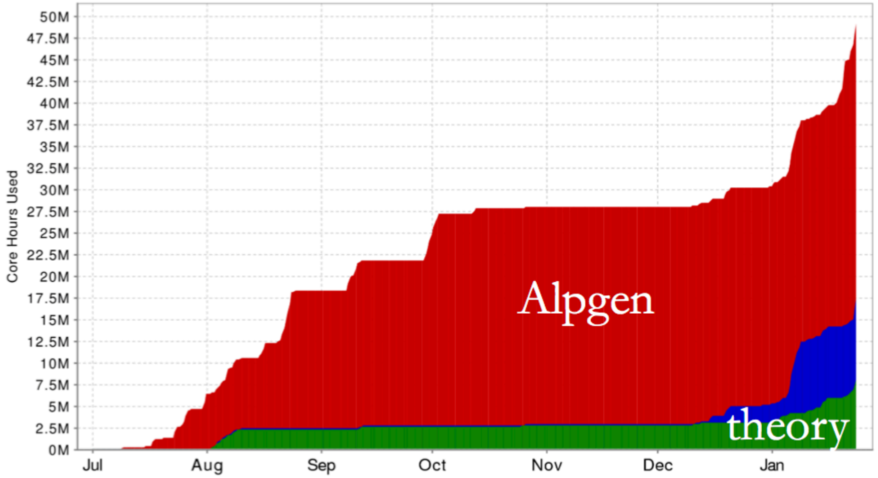




D.Benjamin

# Argonne Opportunistic Usage

- 70M core-hours of Alpgen delivered (16B events) to ATLAS PMG in the last year. Equivalent to 5% annual grid usage.
- Normal job size is 262,144 cores, with 4 threads per core. 1.7x the Grid.
- A new request was received in mid-January for another 10M core-hours of Alpgen.
- New Alpgen version being released. Up to 10 jets possible. New requests possible. Would dwarf current usage stats. Not possible on the Grid.
- Data output averaging 1.6TB/month.
- Sherpa optimization continues, but production use has begun. 192 integrations delivered.
- Working with Eddie to add Mira usage to monitoring plots.
- Panda Integration completed. Thanks Danila.
- ProdSys Integration coming next for EVGEN jobs. Thanks Doug.

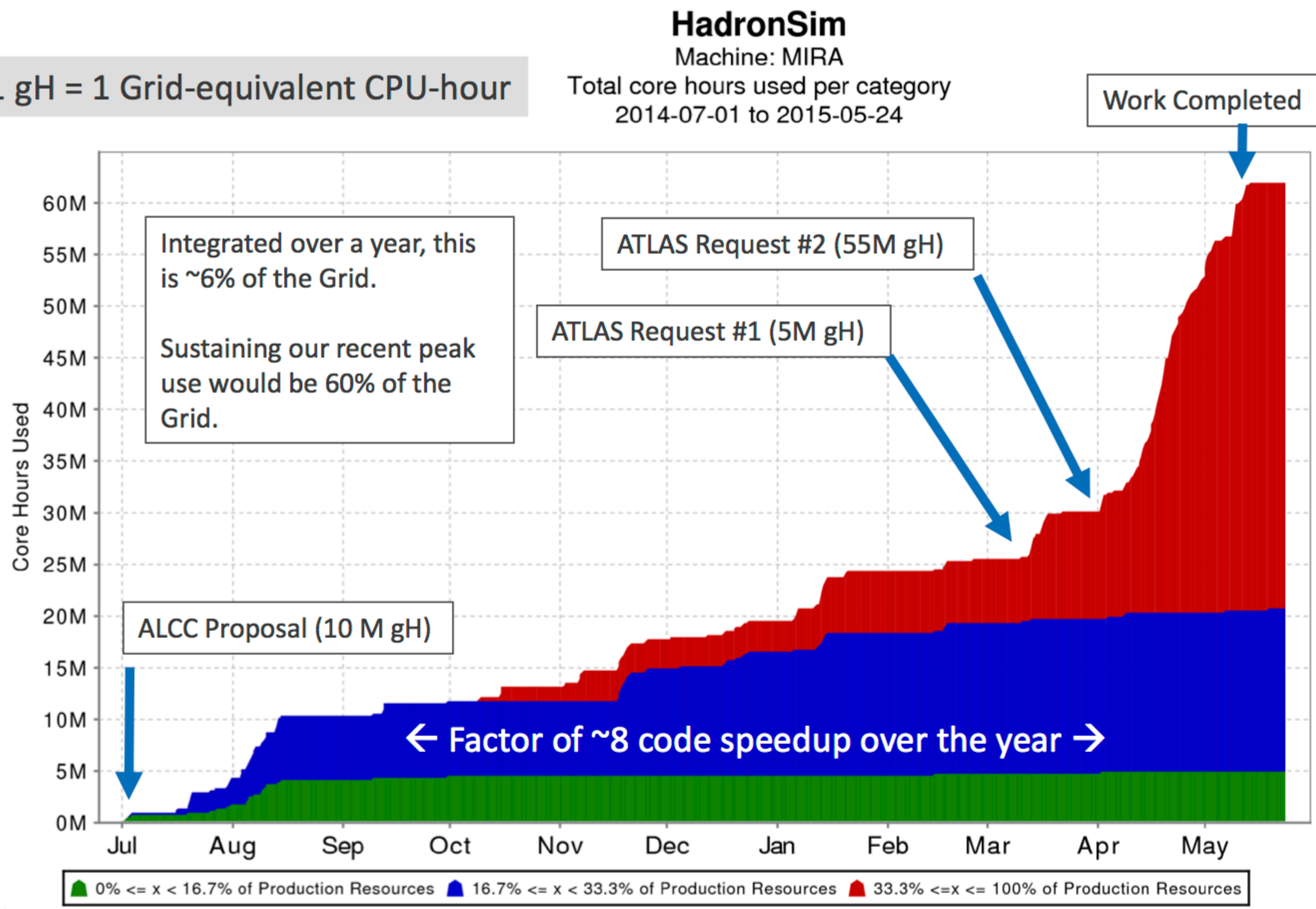


# Mira ALCC Use - Visually

T. Le Compte

1 gH = 1 Grid-equivalent CPU-hour

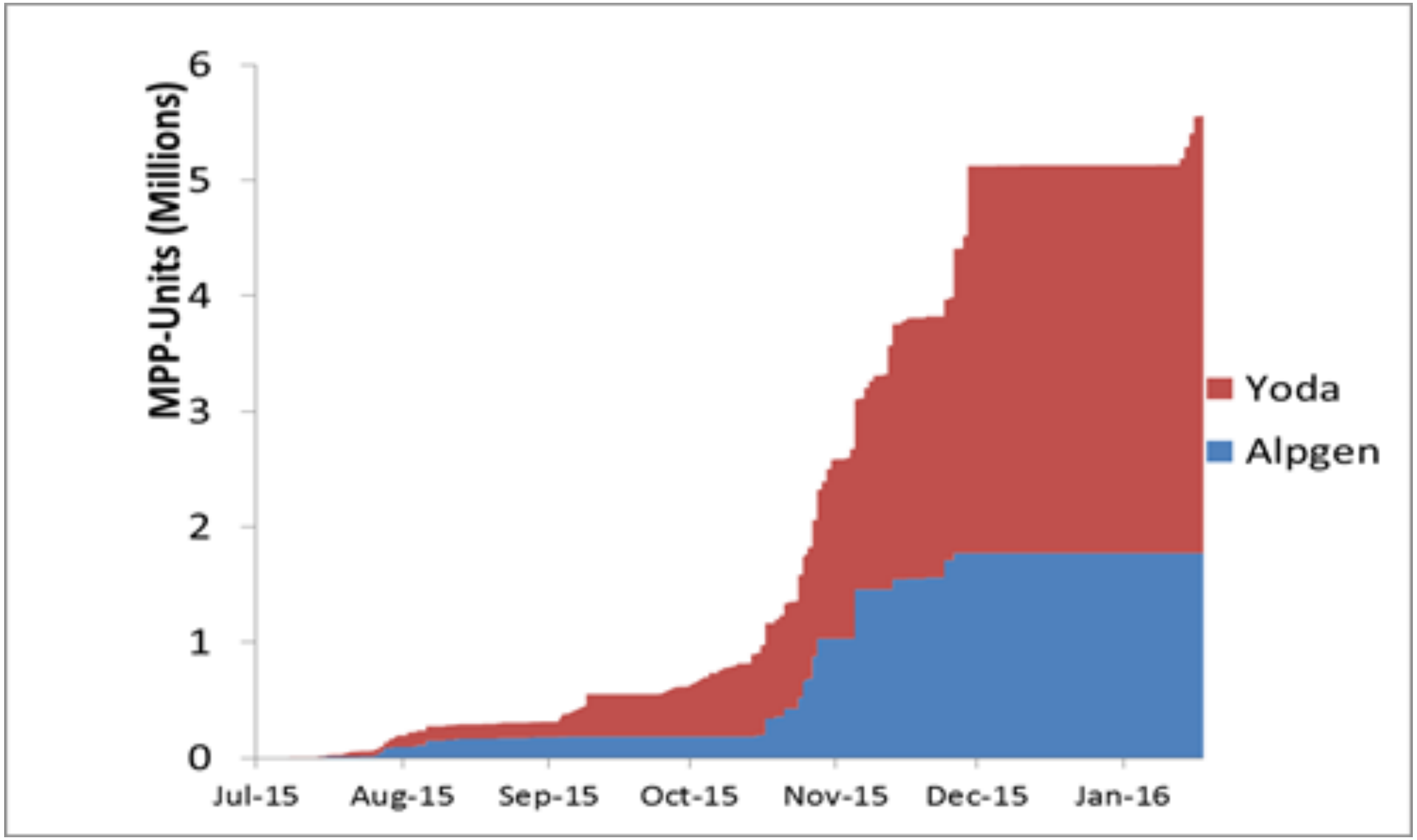





*ALCC – DOE ASCR Leadership Computing Challenge*

T. Le Compte

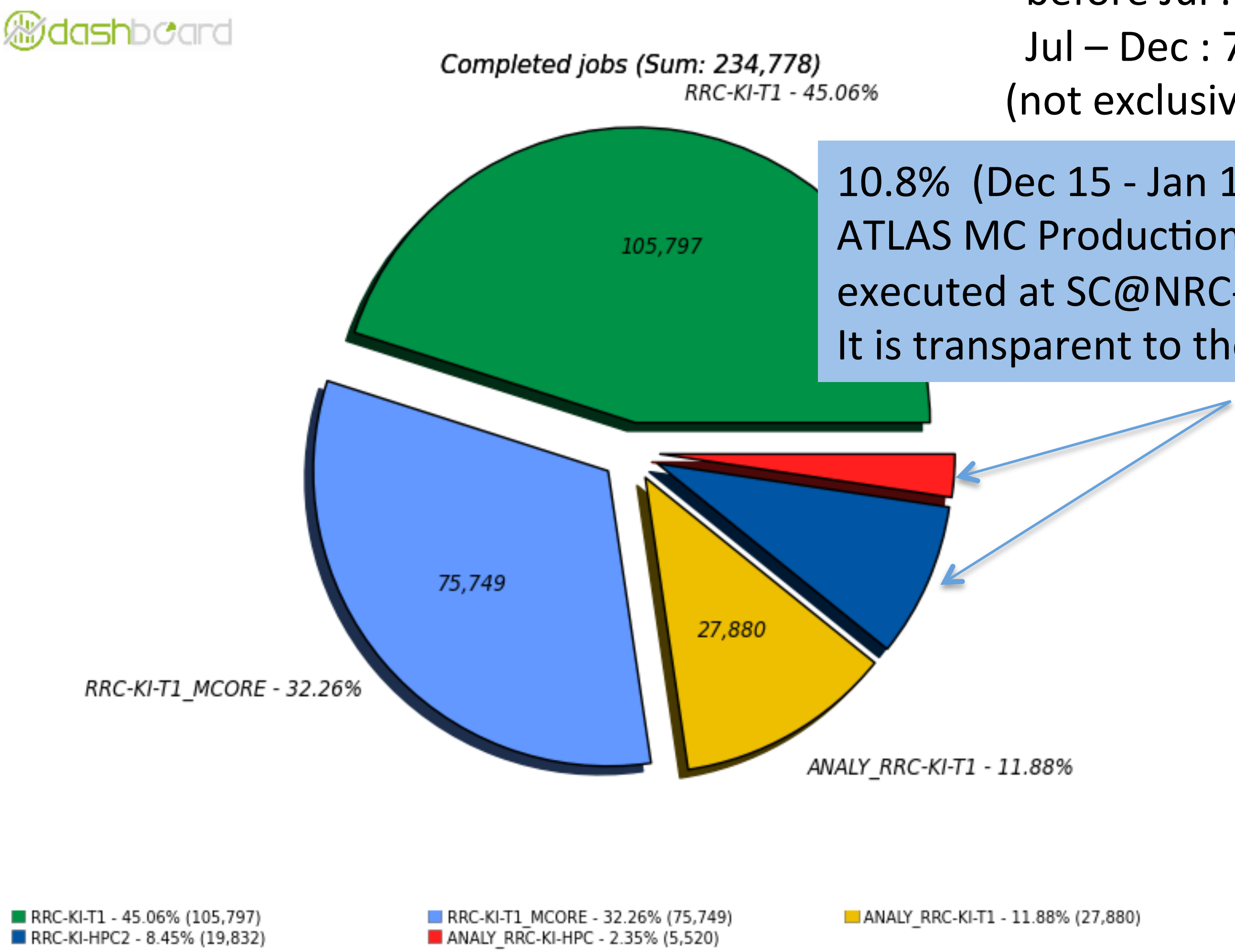
# Mira. Yoda and Alpgen



# ATLAS Site : Tier-1 and Supercomputer at NRC-KI

Possible access to GPUs in 2016 :
before Jul : 150 TFLOPS
Jul – Dec : 700 TFLOPS
(not exclusive access)

10.8% (Dec 15 - Jan 16)
ATLAS MC Production and Analysis executed at SC@NRC-KI
It is transparent to the ProdSys and Users



# ATLAS Physics Groups. EFT.

- ATLAS physicists from NYU and Manhattan College led by Prof. R. Konoplich calculated the Vector Boson Fusion channel for Higgs productions and delivered more than 15 million fully simulated events. All production was done on Titan. The statistics were enough to start development of a new ATLAS physics analysis
  - 12M events with Powheg+Minlo for VBF H->4l studies
  - Full simulation with 18.9.0 release
  - Official ATLAS dataset (data have been copied to Rucio end-point and registered)
- The main idea of effective field theory (EFT) approach is to add some higher dimensional operators of new physics to the lagrangian of the Standard Model. Additional coupling parameters in the Higgs interaction to Standard Model particles change the predicted cross section, as well as the shape of differential distributions.
- For Run 2, it is envisioned to have signal models which depend on a larger number of coupling parameters, in order to account for possible correlations among them. For this purpose, a morphing method has been developed and implemented. It provides a continuous description of arbitrary physical signal observables such as cross sections or differential distributions in a multidimensional space of coupling parameters. The morphing-based signal model is a linear combination of a minimal set of orthogonal base samples spanning the full coupling parameter space. The weight of each sample is derived from the coupling parameters appearing in the signal matrix element.

Rostislav Konoplich

# ATLAS Physics Groups. EFT. Cont'd

- The number of base samples is rapidly increases with the number of additional coupling. For example for a Higgs boson production we need:
  - Gluon fusion + 0 jet
    - (1 coupling in production, 4 couplings in decay) 10 base samples
  - Gluon fusion + 1,2 jets
    - (2, 4) 30 base samples
  - Gluon fusion + 1,2 jets
    - (2, 13) 273 base samples
  - Vector boson fusion
    - (6, 4) 105 base samples
  - Vector boson fusion
    - (13, 9) 1605 base samples

Rostislav Konoplich

# ATLAS Physics Groups. TRT SW

- *WLCG resources are fully utilized and it is important to integrate opportunistic computing resources such as supercomputers, commercial and academic clouds no to curtail the range and precision of physics studies (R.Mount et al)*
- Among the most important Inner Detector ATLAS studies dedicated to be solved on a supercomputer are several urgent tasks for ATLAS Transition Radiation Tracker Software group. They are reconstruction of proton-proton events with large number of interactions (high occupancy conditions), drift circle errors calibration study, particle identification (PID) studies and the production of xAOD group derivations.
- Shortly: we need disk space and CPU time on high performance computers/clusters for our studies. It is always better to have them co-located with Tiers (NRC-KI is our best example) than request some space each 15 days on CERN Scratch disk.

Dimitrii Krasnopevtsev

# ML @ HPC/GPUs for computing support

## Primary interest — workload analysis across large combinatorics

**Early anomaly detection:** e.g., backlog in system *A* leads to delays in system *B* or overflow in system *C*

**Forecasting:** e.g., expected throughput and packet-loss on a given link in the next *n* hours

**Decision making:** e.g., when/where/if should we place additional replicas of a dataset

**Resource optimisation:** e.g., which branches can we prune from a file

## We are collecting and aggregating all this data centrally

PanDA jobs, Rucio transfers, Job & CLI traces, xAOD traces, perfSonar, … — on Hadoop and ElasticSearch

Data aggregated in batch (Hadoop) and on the fly (ES) — produces reasonably sized inputs for ML algorithms

## Inherently parallel — many algorithms infeasible on our small-scale VMs

e.g., simple HoltWinters throughput forecast for all network links: 10'000 potential links x 3 seconds each

The frameworks we are already using for this have GPU support (R: CUDA, Theano: CUDA & OpenCL)

Shipping these workloads to an HPC with GPUs (even without!) would make a dramatic difference

# Summary and Conclusions

- Supercomputers offer important and necessary opportunities to ATLAS
- Integrate more HPCs into production environment
  - Assess pro/cons per project/machine
    - Local support is essential for the success
    - Network connectivity is essential
    - ‘Nearby’ ATLAS site capability to get and to store ‘extra’ data is essential
  - Many Supercomputing centers are very interested to collaborate with us.
  - Three technical thrusts
    - Integrate HPC into production environment
    - Port ATLAS code to each HPC system
    - Learn how to exploit accelerators where present
- Great progress with ‘HPC’ pilot module in 2015 led to the Titan integration with the ATLAS Production System
  - Mira is the next target
- Non x86 machines for ATLAS simulation needs more attention
  - ATLAS MC release
- Validation can be done faster