ATLAS Cloud Computing R&D project

Sergey Panitkin[1], Fernando Barreiro Megino[2], Jose Caballero Bejar[1], Doug Benjamin[3], Alessandro Di Girolamo[2], Ian Gable[4], Val Hendrix[5], John Hover[1], Katarzyna Kucharczyk[2], Ramon Medrano Llamas[2], Peter Love[6], Henrik Ohman[7], Michael Paterson[4], ***Randall Sobie***[4], Ryan Taylor[4], Rodney Walker[8], Alexander Zaytsev[1]

on behalf of the ATLAS Collaboration

1: Brookhaven National Laboratory, Upton, USA
2: CERN, Geneva, Switzerland
3: Duke University, Durham, USA
4: University of Victoria, Victoria, Canada
5: Lawrence Berkeley National Lab, Berkeley, USA
6: Lancaster University, Lancaster, UK
7: Uppsala University, Uppsala, Sweden
8: Ludwig-Maximillian Universitat, Munich, Germany

CHEP 2013
Amsterdam, The Netherlands

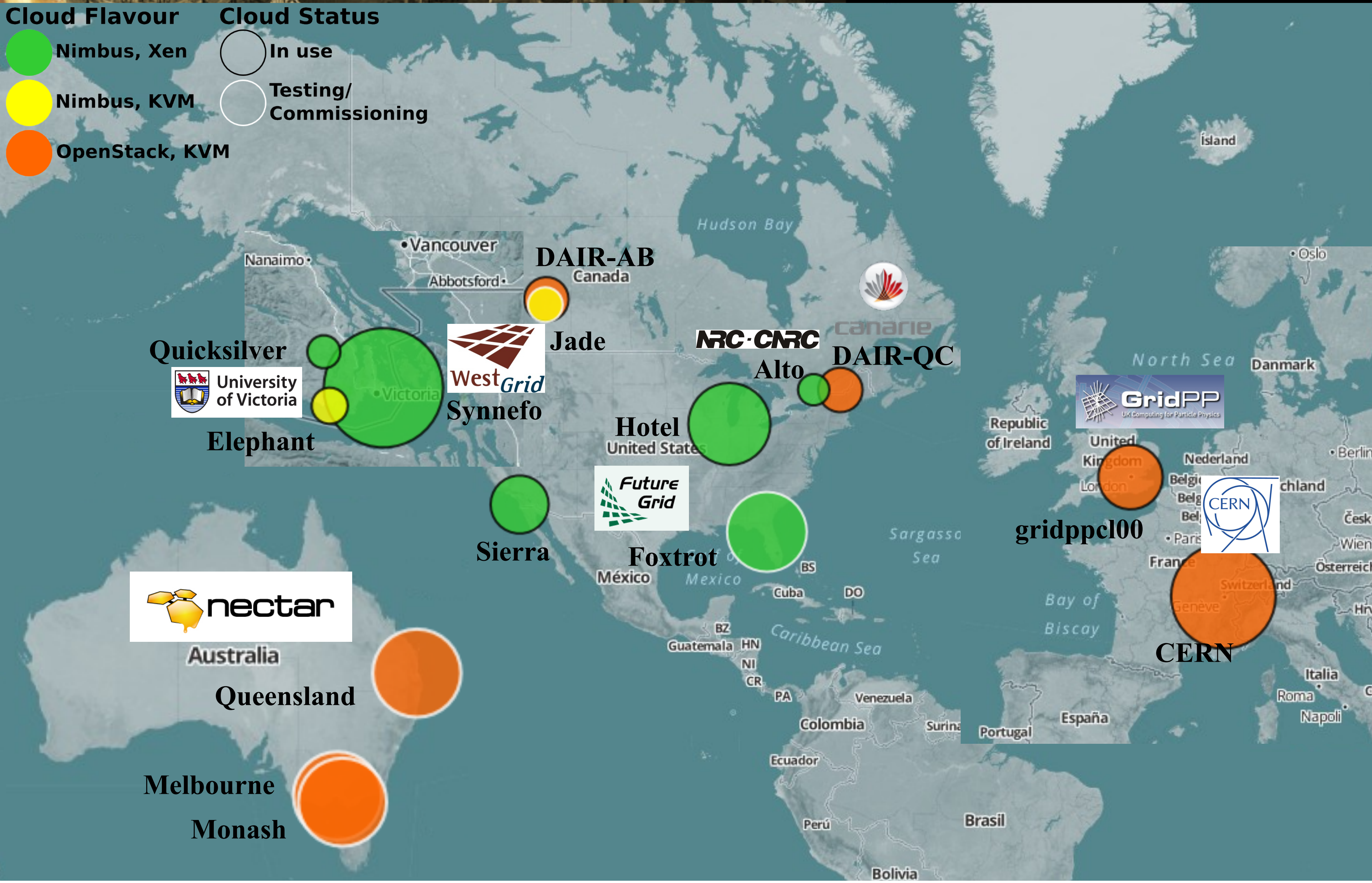
Outline

- Introduction
- Private and Academic Clouds
  - Cloud Scheduler and “Grid of Clouds”
  - ATLAS HLT farm
- Public and Hybrid Clouds
  - Amazon EC2
  - Google Compute Engine

Cloud Computing and ATLAS

- A few years ago the ATLAS Experiment set up cloud computing project to exploit virtualization and clouds
  - Utilize private and public clouds as an extra computing resource
  - Mechanism to cope with peak loads on the Grid
- Since then we gained experience with variety of cloud platforms
  - Amazon EC2
  - Helix Nebula project (CloudSigma, T-Systems and ATOS)
  - FutureGrid in USA, Synnefo cloud (U. Victoria)
  - Private clouds based on OpenStack, CloudStack, OpenNebula, etc…
- In this talk we will discuss ATLAS Cloud R&D activities that took place since CHEP 2012

A “Grid of Clouds”

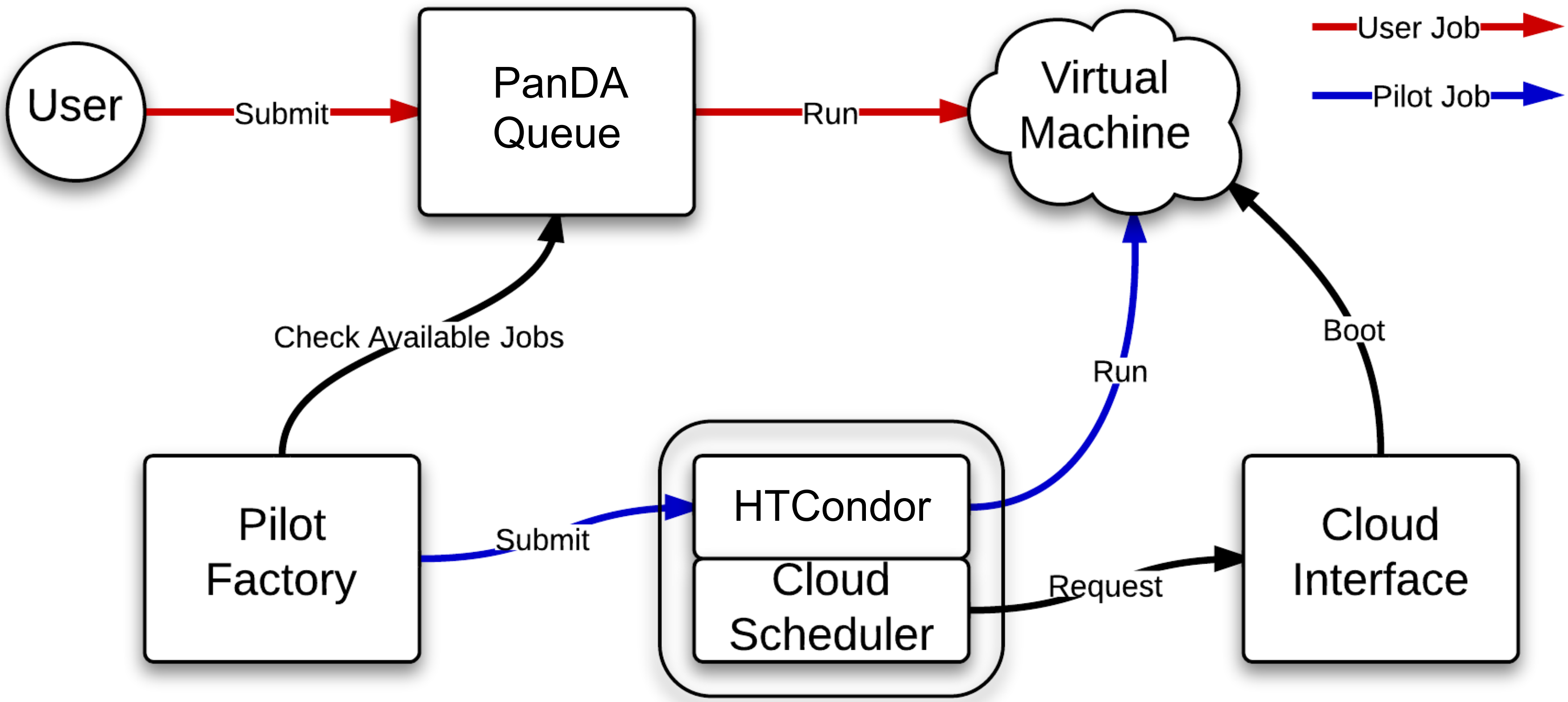


Powered by Cloud Scheduler

- A Python package for managing VMs on IaaS clouds, based on the requirements of HTCondor jobs
- Users submit HTCondor jobs, with additional attributes specifying VM properties
- Developed by UVic and NRC since 2009
- Used by ATLAS, BaBar, CANFAR
  - And possibly Belle II in the future

- More information about Cloud Scheduler:
  - https://github.com/hep-gc/cloud-scheduler
  - Research Computing in a Distributed Cloud Environment (Proc. of HPCS 2010)
  - A Batch System For HEP Applications on a Distributed IAAS Cloud (Proc. of CHEP 2010)
  - http://goo.gl/G91RA (ADC Cloud Computing Workshop, May 2011)

Cloud Job Flow



Key Features of Cloud Scheduler

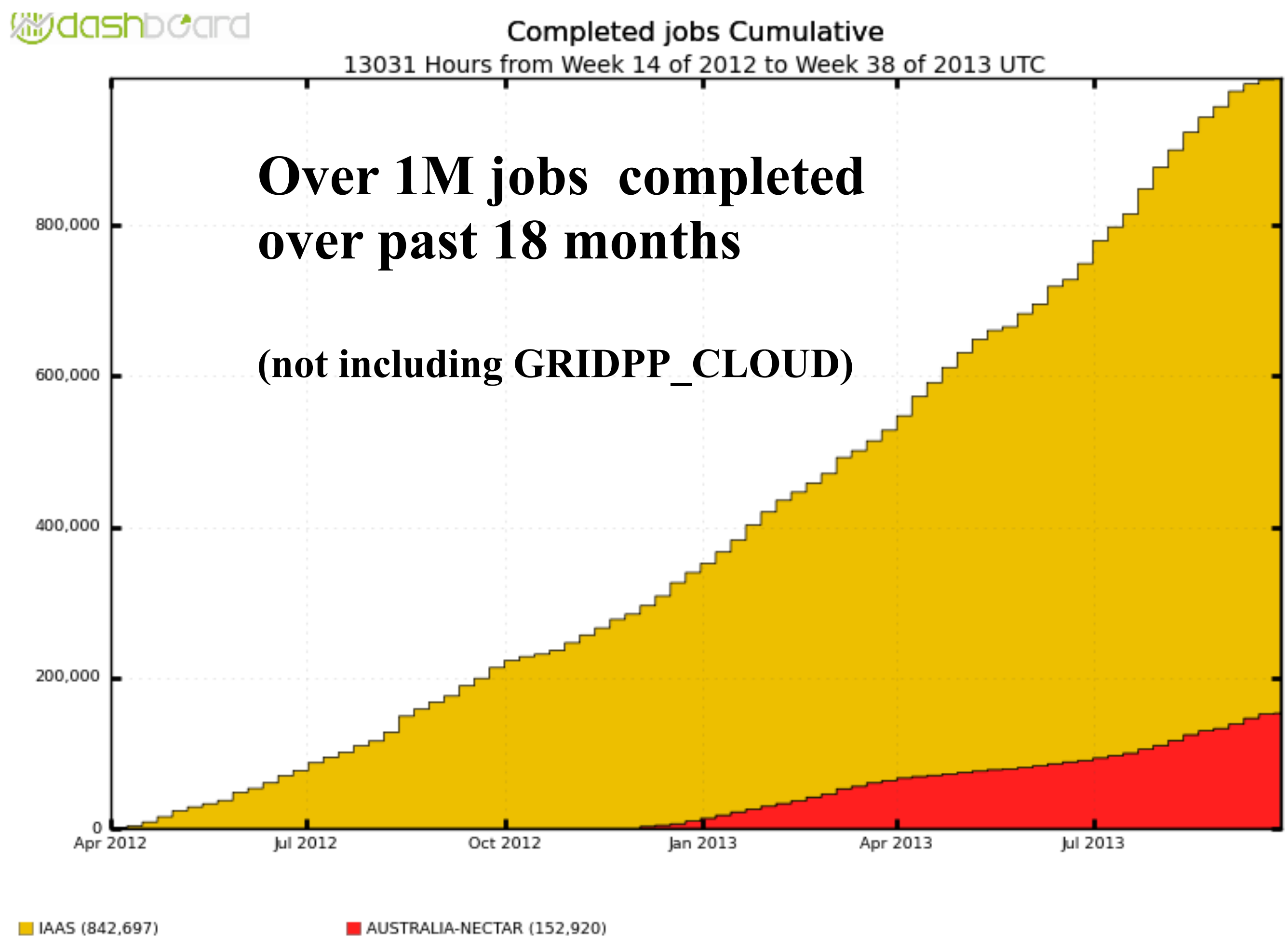
- Dynamically manages quantity and type of VMs in response to user demand
- Easily connects to many IaaS clouds, and aggregates their resources together
- Complete solution for harnessing IaaS resources in the form of an ordinary HTCondor batch system
- Generic tool, not grid-specific or HEP-specific
- `pip install cloud-scheduler`

Cloud Scheduler Image

- Dual hypervisor image. Can run on KVM or Zen
- Configuration management with Puppet
- Use one image on all (12+) clouds
- Customized CernVM batch node v2.7
- Use whole node VM for better efficiency
  - Cache sharing instead of disk contention
  - Fewer image downloads when ramping up

Production on IaaS Cloud

- Aust[redacted]
- preliminary tests Oct. 2011
- standard operation Apr. 2012
- GRIDPP_CLOUD: added May 2013
- CA-JADE: testing now

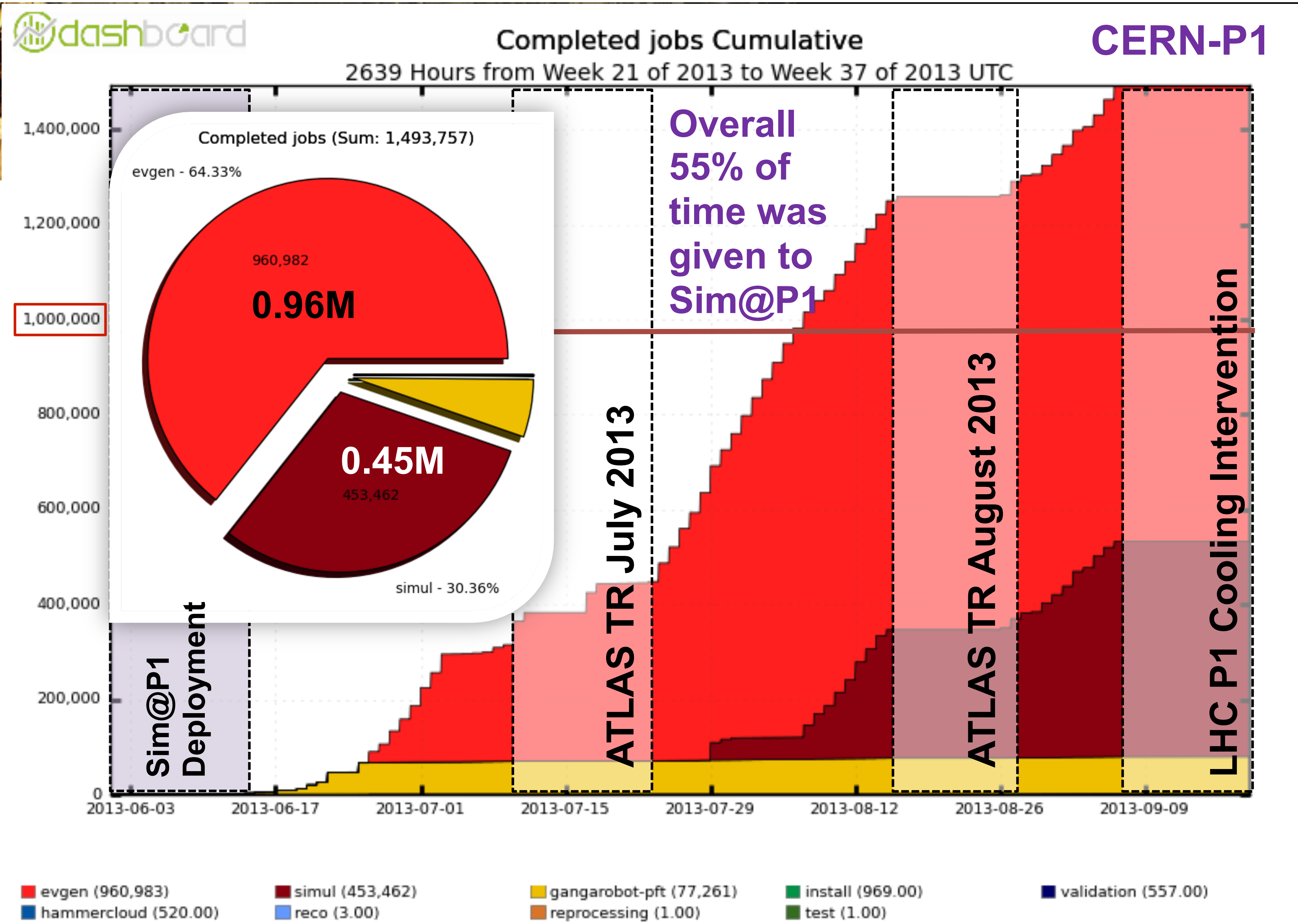


Cloud on ATLAS HLT Farm

- Long LHC shutdown provided an opportunity to utilize ATLAS High Level Trigger (HLT) farm for ATLAS offline computing
- Prototyping started in February 2013
- Production operation since June 2013
- Resources organized as an IaaS cloud using OpenStack
    - Resources are presented to users as a PanDA queue
    - ~1500 HT compute nodes located at Point 1
    - ~17.1k batch slots are available (~2.1k VMs) for production

Cloud on HLT farm status

- Production operation since June 2013
- Coordinated with ATLAS TDAQ activities. TDAQ has top priority
- The largest ATLAS Grid site when running
- As of now more than **1.4M** jobs were finished on HLT farm cloud
- **21.6%** job failure rate due to opportunistic nature of resources (hardware repurposed for TDAQ needs)
- That translates in **8.6%** CPU time loss over the period of running
- Workload related job failure rate is on the level of **0.1%**
- Fast switch between activities:
  - TDAQ to Sim@P1 under one hour.
  - Sim@P1 to TDAQ under 10 minutes.



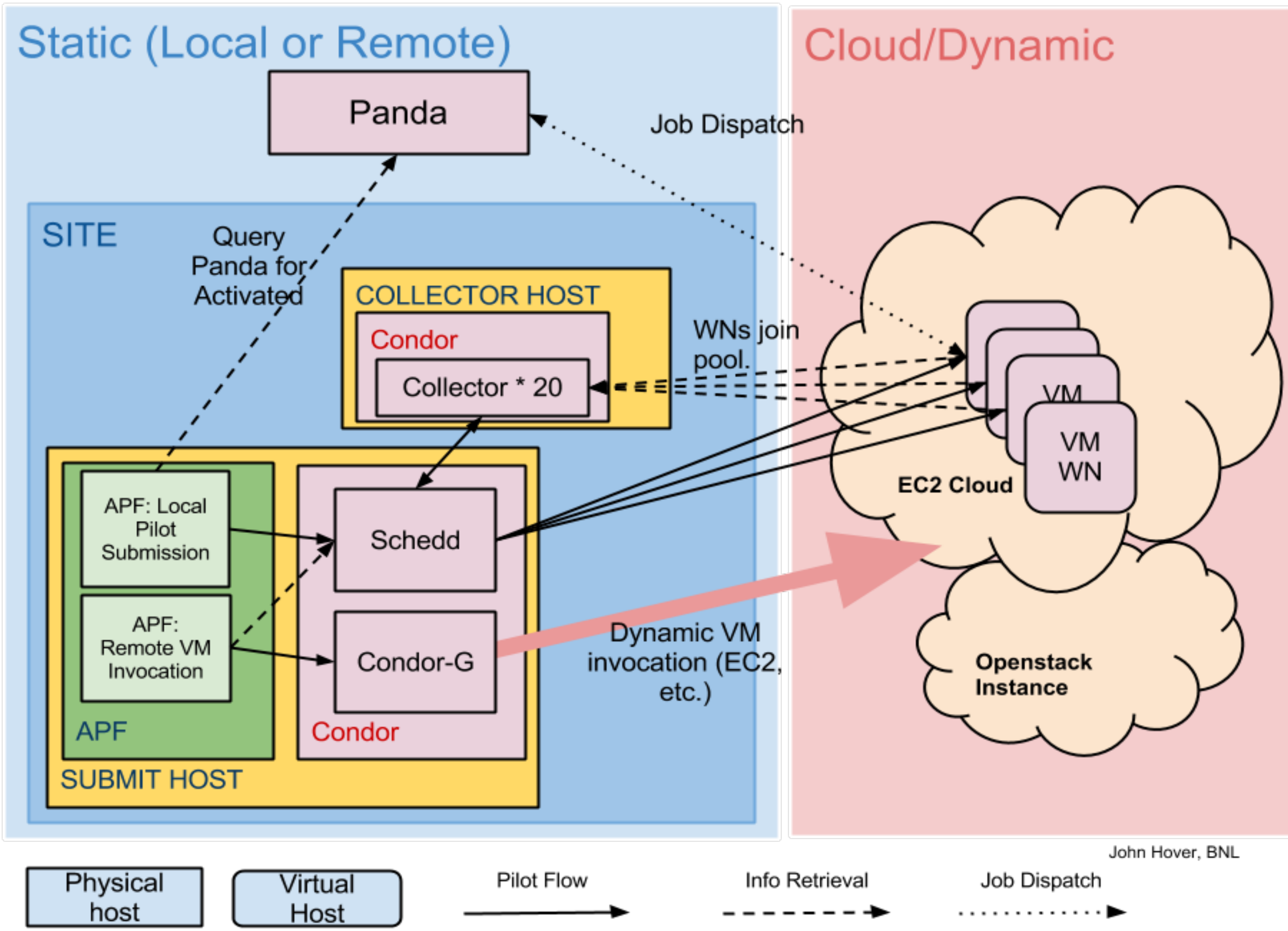
Total: 1.49M jobs

June 1, 2013 – Sep 18, 2013

Project on Amazon EC2

- RACF BNL group has received grant allocation from Amazon EC2 in 2013
- The idea of the project was to set up a hybrid cloud with some of the resources at BNL T1 and "elastic" part of the cloud on Amazon EC2
- We wanted to try Amazon EC2 spot market
    - Cheapest resources, but price and availability fluctuations
    - Dynamic environment - opportunistic resources
- We wanted to create one cloud spanning geographically distributed EC2 sites

Hybrid Cloud: BNL OpenStack + EC2



Hybrid cloud on EC2 results

- Ran ~5000 EC2 VMs for about 3 weeks
- Used 3 EC2 zones – Virginia, California, Oregon
- Added to the pool ~250 VM on OpenStack at BNL
- Ran ATLAS Monte Carlo production jobs
- Total cost ~ $13k , only $750 for data transfer
- Actual spot price paid was very close to baseline – less than $0.01 per hour for m1.small type of instance
- Reliable operations of EC2 platform
- Poor job efficiency on EC2 due to long running jobs.
    - It's better to run short jobs on the EC2 spot market!

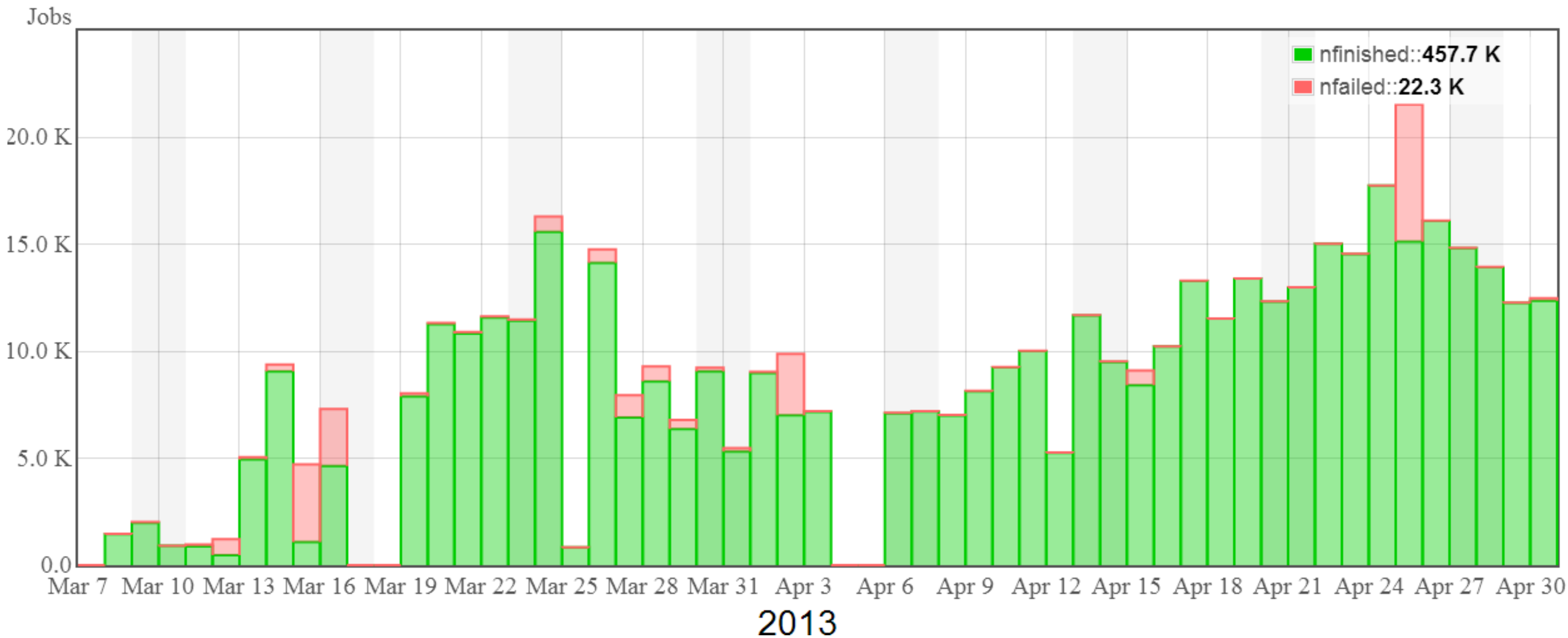
Google Compute Engine Project

- ATLAS was invited to participate in GCE closed preview period in August 2012
- Brand new cloud platform, new cloud API, modern hardware
- Google agreed to allocate additional resources for ATLAS after the initial period
  - ~5M core-hours, 4k cores for about 2 months, (original preview allocation was 1k cores)
- Resources were organized as HTCondor based PanDA queue
  - Centos 6 based custom built images, with SL5 compatibility libraries to run ATLAS software, CVMFS
  - Whole-node, 8 core instances with ~3.4 TB of ephemeral storage
  - HTCondor head nodes and web proxies at BNL
  - Output automatically transferred to storage at BNL
- Transparent inclusion of the cloud resources into ATLAS Grid
- The idea was to test long term stability while running a cloud cluster similar in size to Tier 2 site in ATLAS
- Planned as a production type of run. Delivered to ATLAS as a resource and not as an R&D platform

Running on GCE

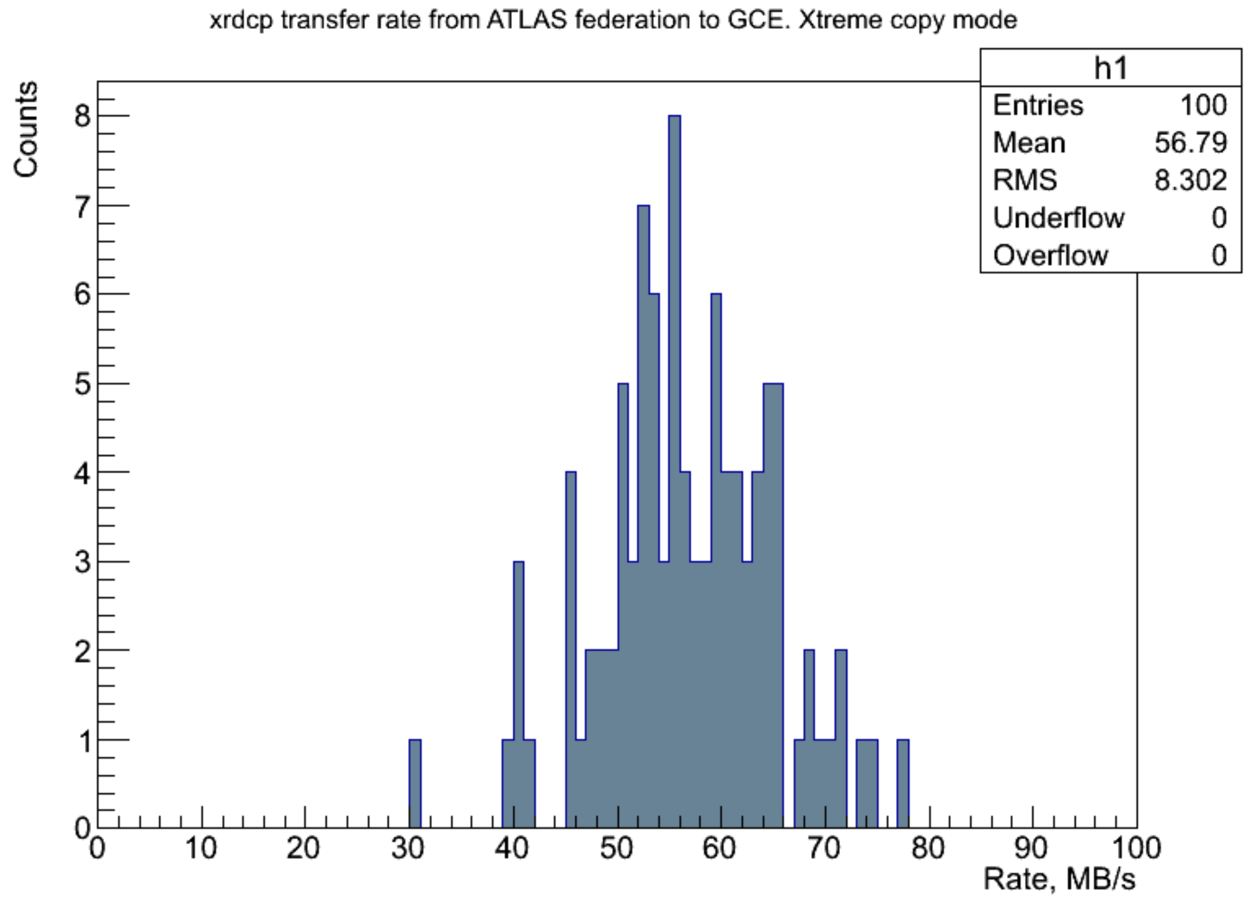
- We ran for about 8 weeks (2 weeks were planned for scaling up)
- We ran computationally intensive workloads
  - Physics event generators, Fast detector simulation, Full detector simulation
- Very stable running on the GCE side
- Most problems that we had were on the ATLAS side, not cloud related
- Overall failure rate ~6%, mostly during start up and debugging period
- Completed 458k jobs, generated and processed about 214M events
- Also ran several smaller projects
  - Large PROOF farms on GCE
  - Tested data transfer from Federated ATLAS Xroot (FAX)
- We were invited to give a talk at Google IO Conference in May 2013 about the project

Failed and Finished Jobs on GCE



- Most of the job failures occurred during start up and scale up phase – as expected
- Most of the failures were on the ATLAS side – file transfer, LFC problems, HTCondor
- No failures were due to GCE problems

Data transfers from FAX to GCE



- Data transfer from Federated ATLAS Xroot to GCE in multisource/multi-stream mode
- Xroot cluster on GCE using ephemeral storage with 1.7 TB volumes per node
- Average transfer rate: 57 MB/s (single source xrdcp rate 40 MB/s)
- Note, this is over public networks

Summary

- ATLAS Cloud R&D has been an active and successful project
- Significant computational resources were delivered to ATLAS in the form of IaaS clouds in the past 18 months
- Production scale projects on private and academic clouds
  - Cloud Scheduler/IaaS cloud has been running for the past 18 months on infrastructure distributed worldwide
  - Large scale OpenStack installation on ATLAS HLT farm at CERN
    - Comparable to Tier 1 in CPU capacity (2.1k VMs, 17.1k batch slots)
- Active engagement with commercial cloud providers
  - Large scale production run on Amazon EC2
    - Successfully utilized EC2 spot priced resources at scale (~5k running VMs at one time)
    - Operated hybrid private-public cloud spanning EC2 and OpenStack at BNL
  - Large scale production run on Google Compute Engine for about 2 months
    - ~500 VMs, ~4k cores on a new high performance cloud platform