The Fermilab HEPCloud, or How to add 240 TFlops in an hour or two*

Oliver Gutsche, for the Fermilab HEPCloud Team

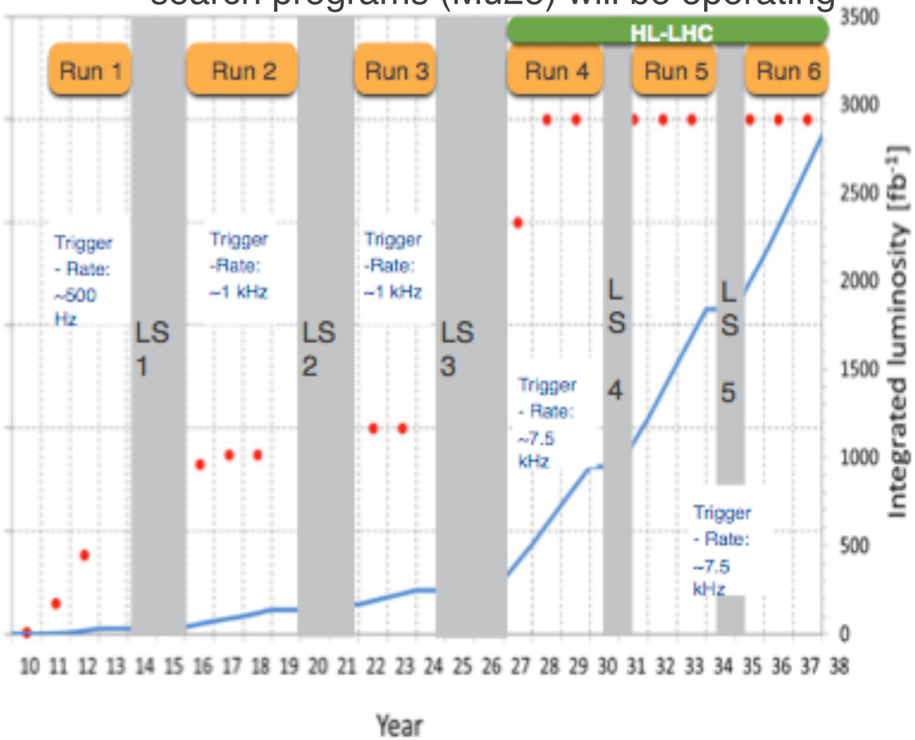
HEP Software Foundation – Community White Paper

January 23, 2017

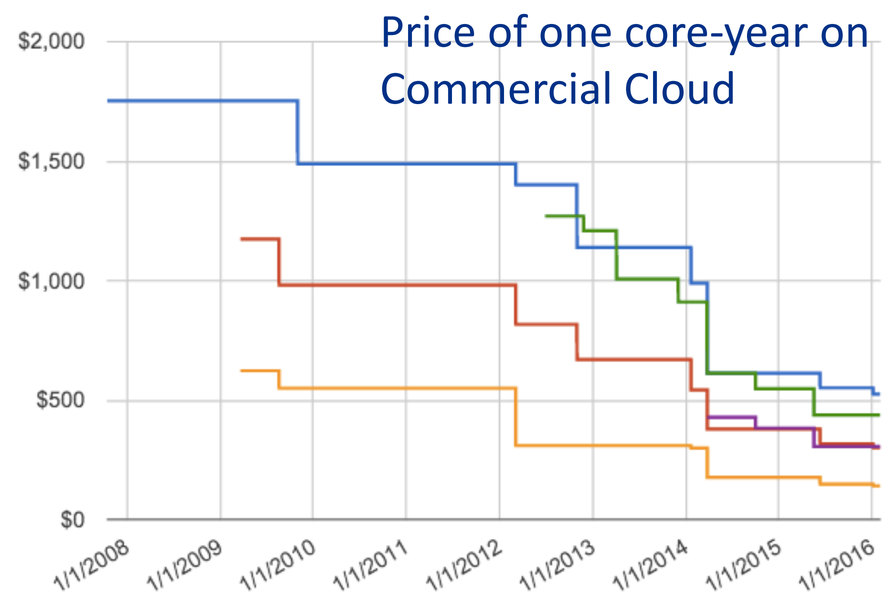
* Or three. Four at the most.

Drivers for Evolving the Facility: Capacity and Cost

- High Energy Physics computing needs will be 10-100x current capacity
  - Two new programs coming online (DUNE, High-Luminosity LHC), while new physics search programs (Mu2e) will be operating



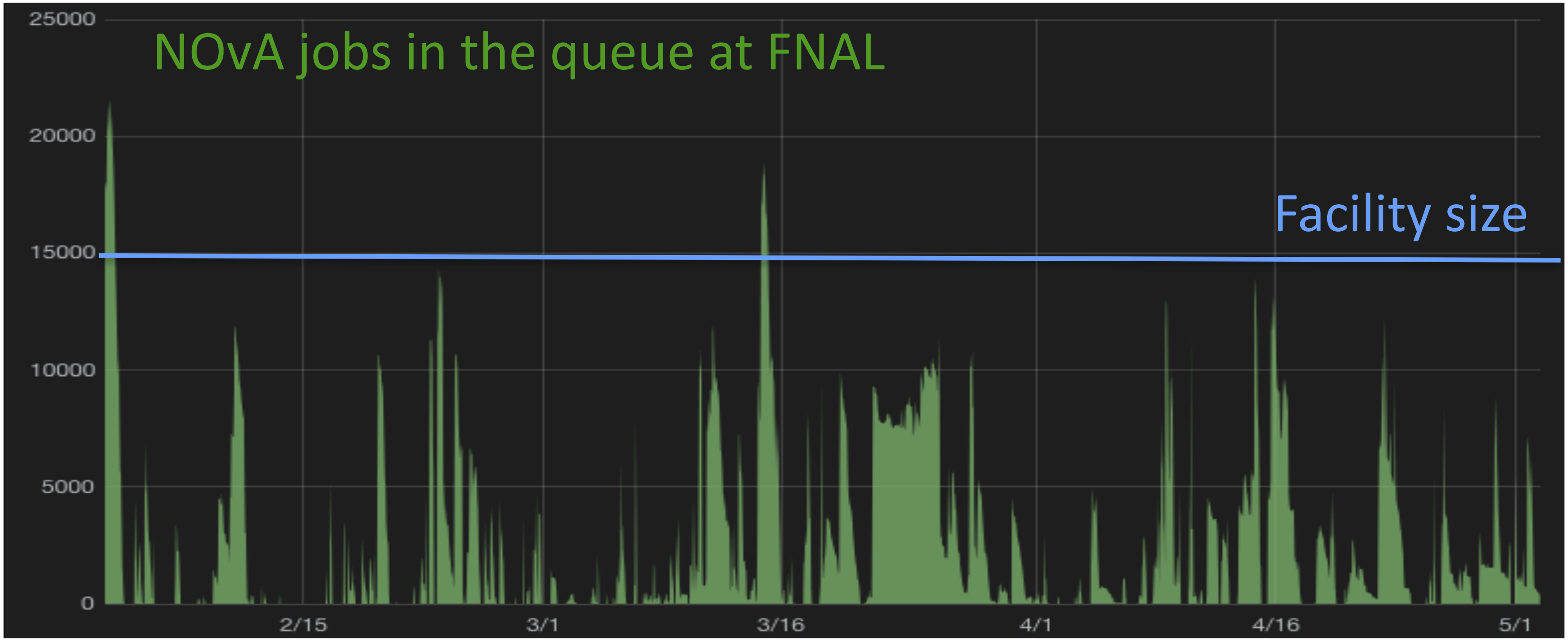
- Scale of industry at or above R&D
  - Commercial clouds offering increased **value** for decreased **cost** compared to the past



Fermilab

Drivers for Evolving the Facility: Elasticity

- Usage is not steady-state
- Computing schedules driven by real-world considerations (detector, accelerator, …) but also ingenuity – this is research and development of cutting-edge science



HEPCloud: the Evolved Facility

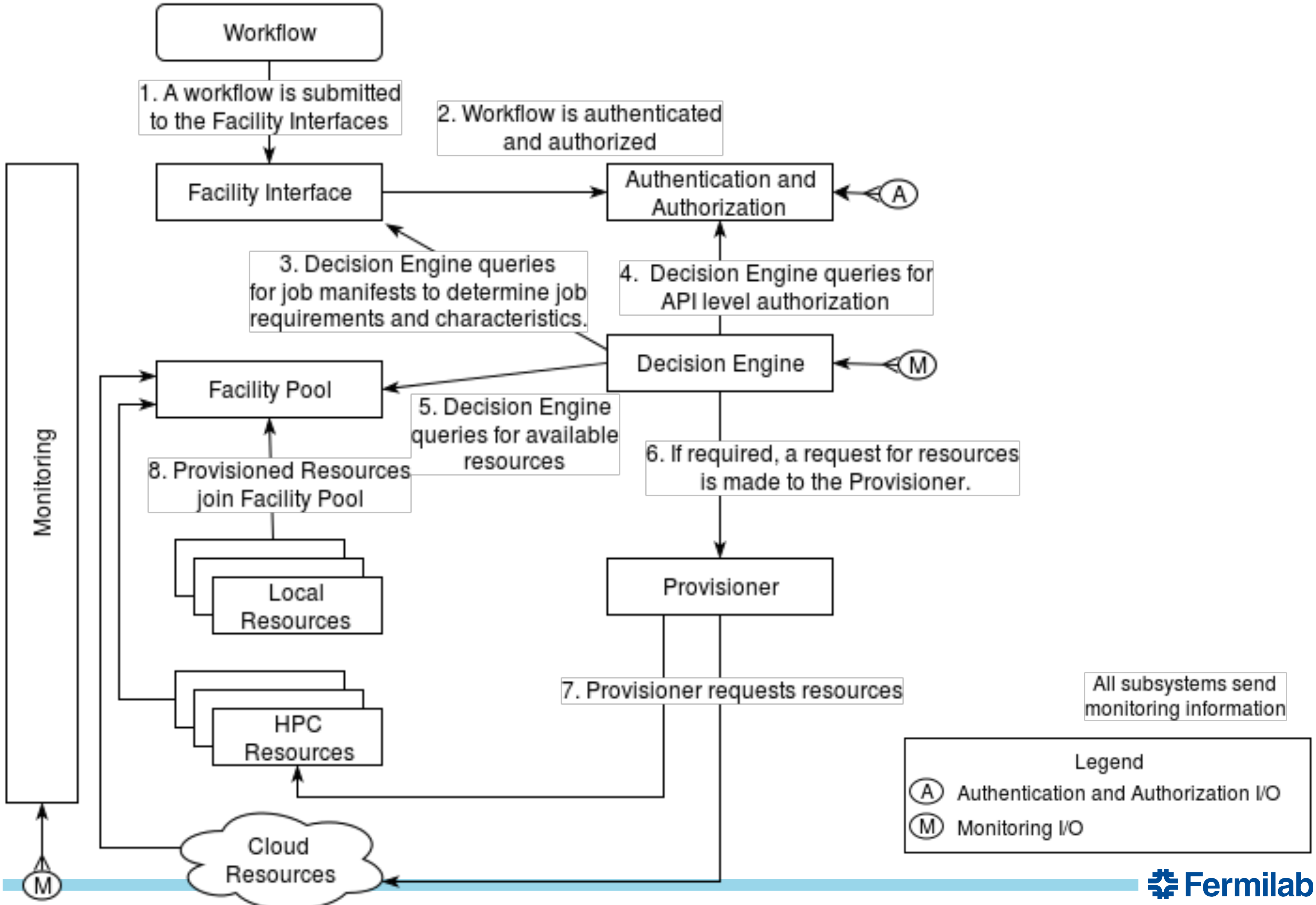
- Vision Statement
  - HEPCloud is envisioned as a portal to an ecosystem of diverse computing resources commercial or academic
  - Provides “complete solutions” to users, with agreed upon levels of service
  - The Facility routes to local or remote resources based on workflow requirements, cost, and efficiency of accessing various resources
  - Manages allocations of users to target compute engines
- Pilot project to explore feasibility, capability of HEPCloud
  - Goal of moving into production during FY18
  - Seed money provided by industry

HEPCloud Collaborations

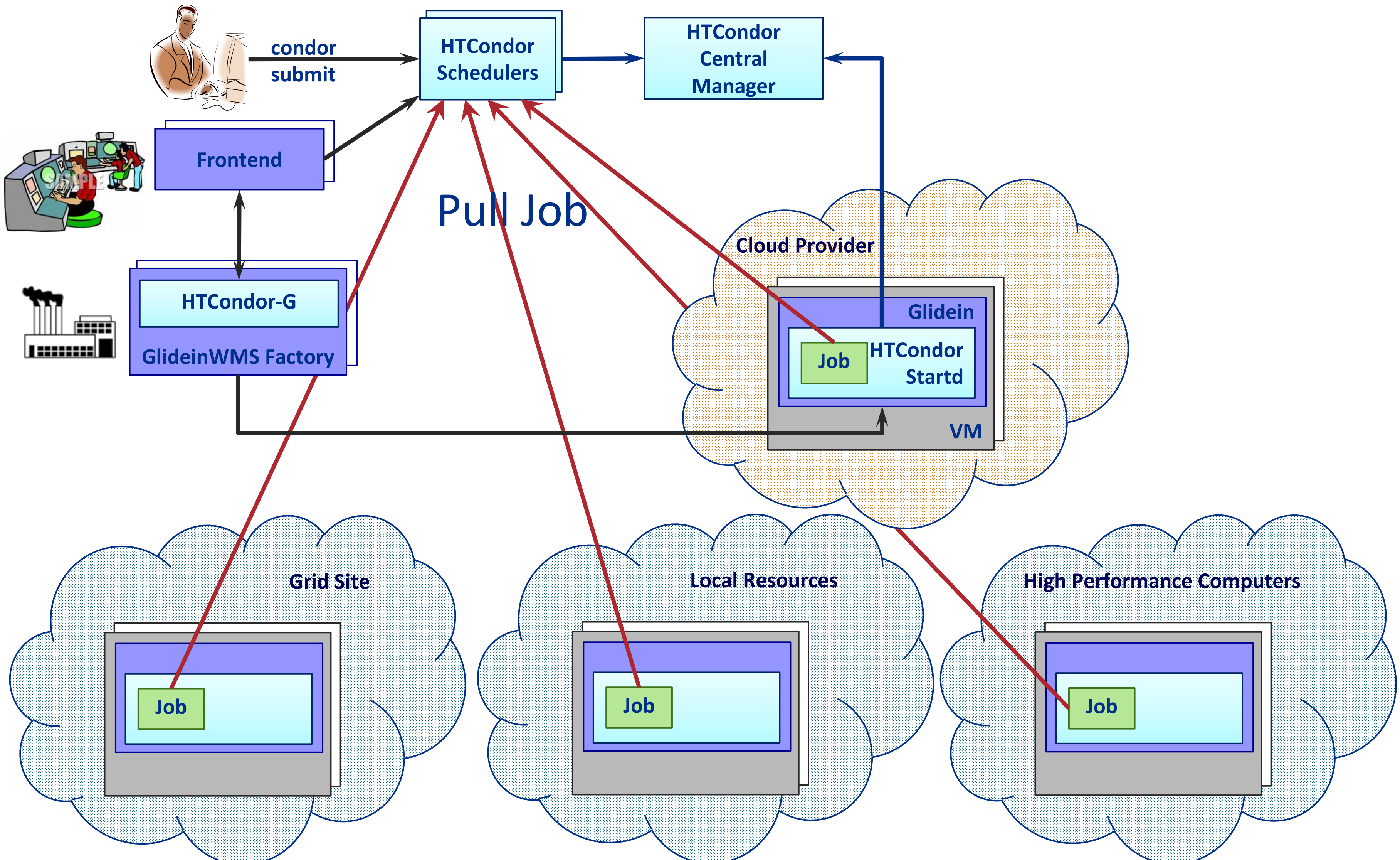
- Participate in collaboration to leverage tools and experience whenever possible

- Grid technologies – Worldwide LHC Computing Grid
  - Preparing communities for distributed computing
- BNL and ANL , ATLAS – engaged in next HEPCloud phase
- HTCondor – common provisioning interface
- CMS, Neutrino and Muon experiments – collaborative knowledge and tools, cloud-capable workflows
- CERN faces similar challenges and we are having productive conversations
  - For example - CERN openlab CTO is engaged in HEPCloud

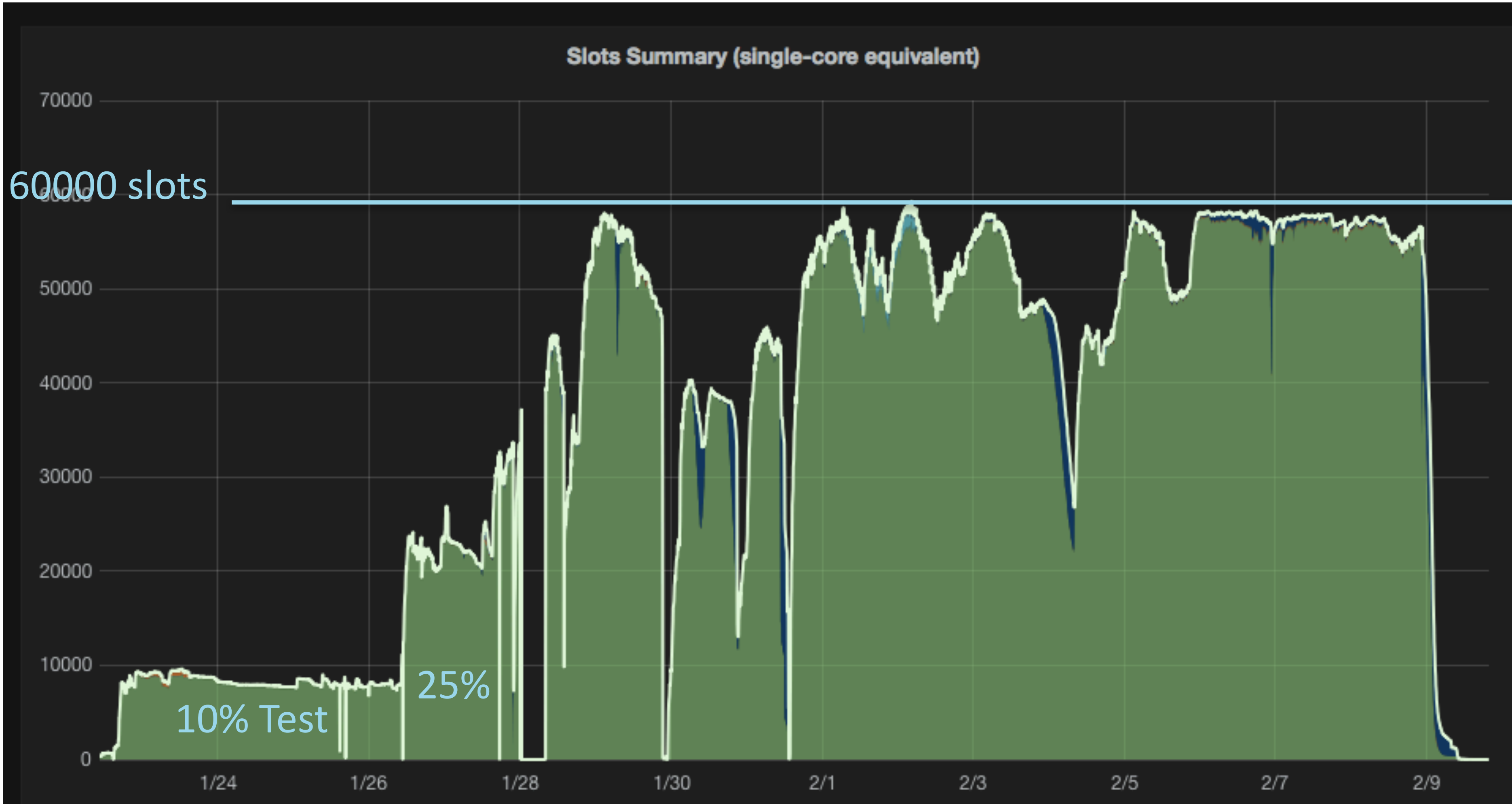
HEPCloud Architecture



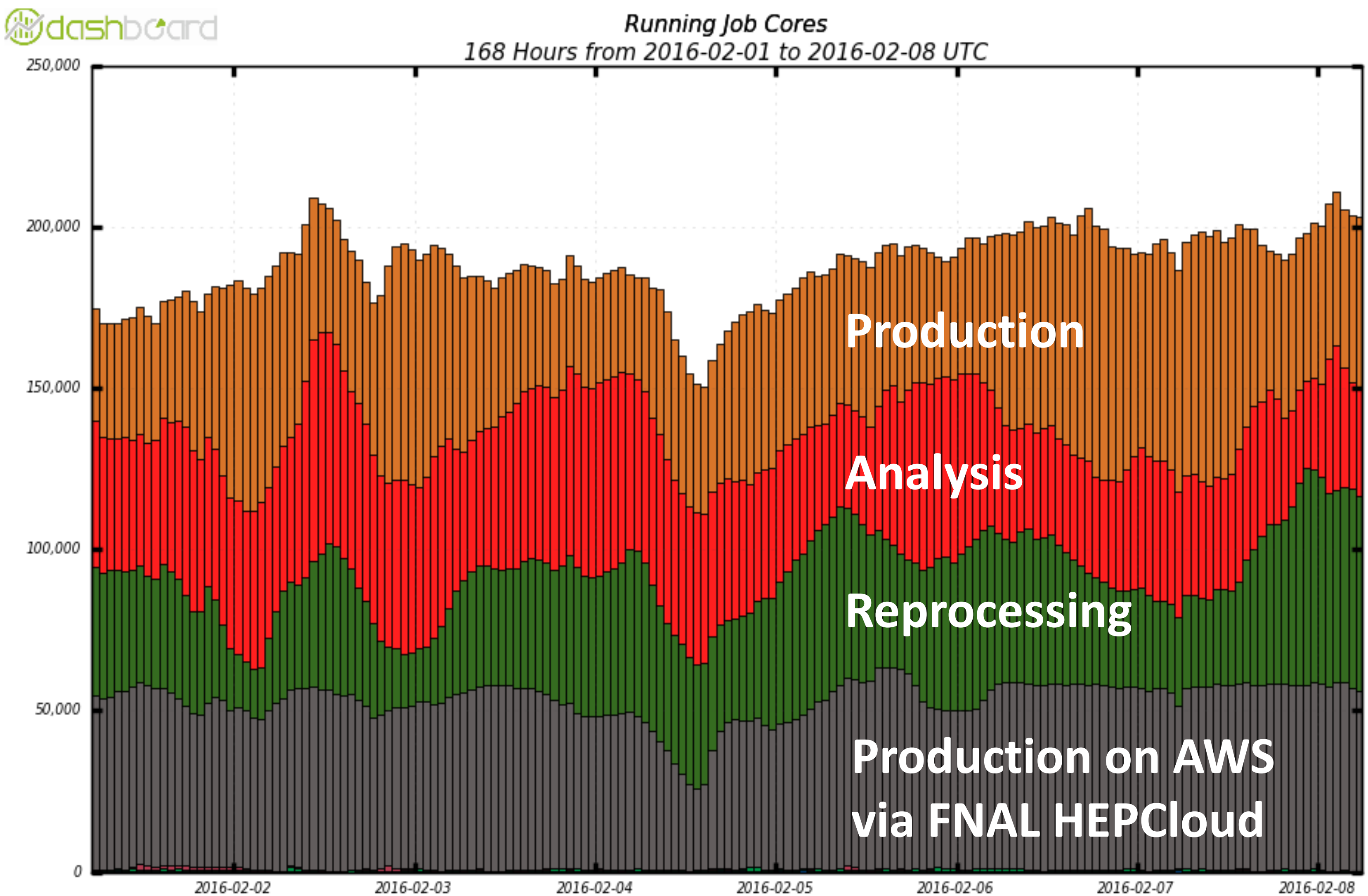
HEPCloud – glideinWMS and HTCondor



Reaching ~60k slots on AWS with FNAL HEPCloud

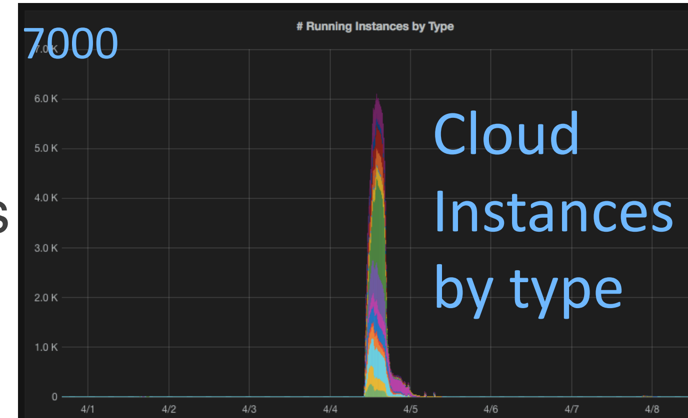


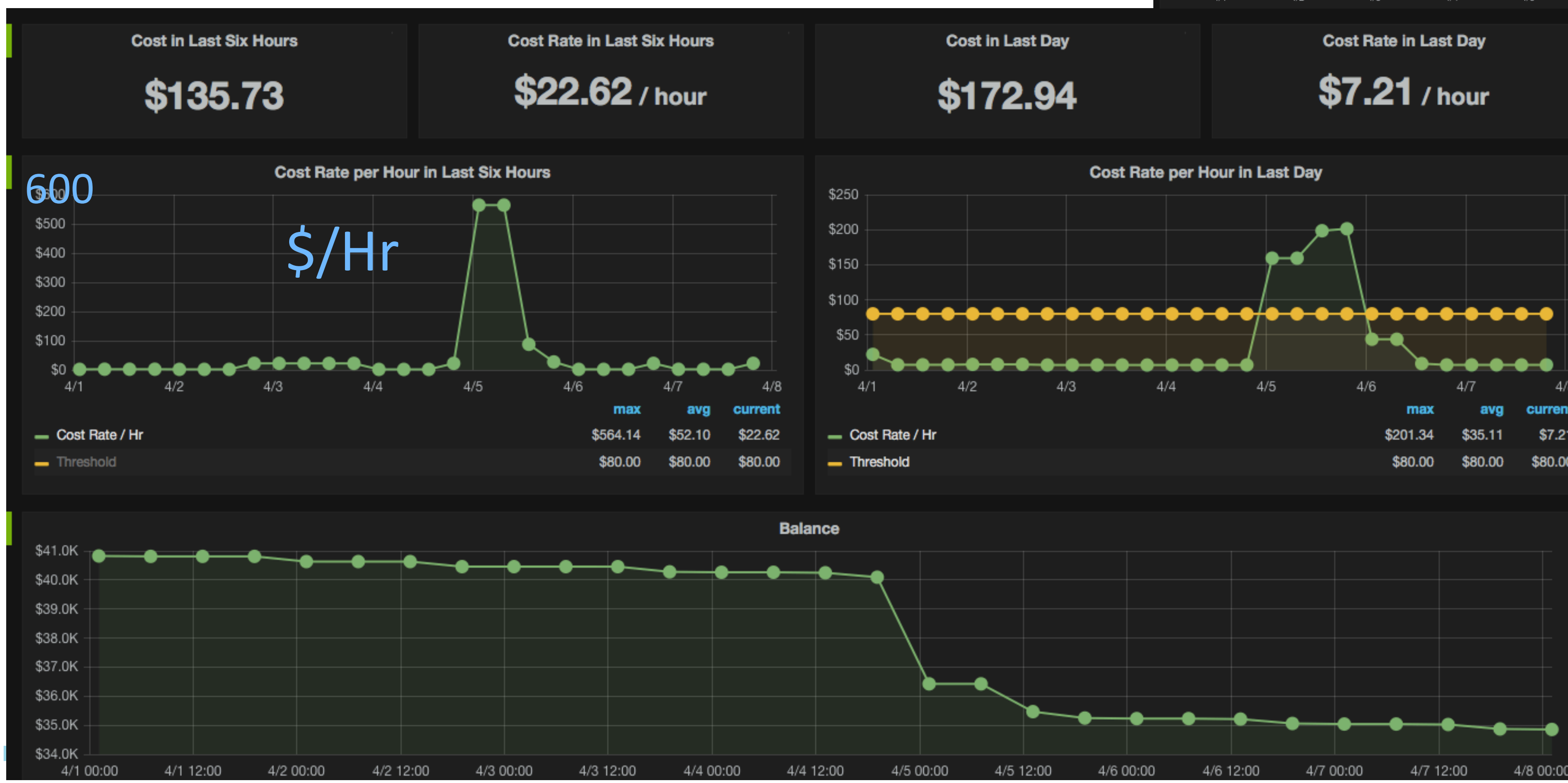
HEPCloud AWS: 25% of CMS global capacity



HEPCloud: Orchestration

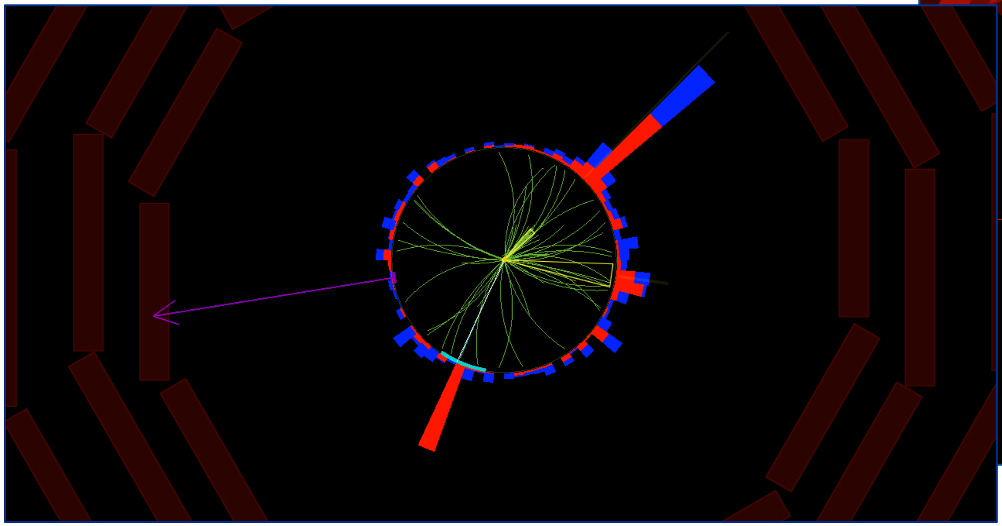
- Monitoring and Accounting
  - Synergies with FIFE monitoring projects
    - But also monitoring real-time expense
  - Feedback loop into Decision Engine

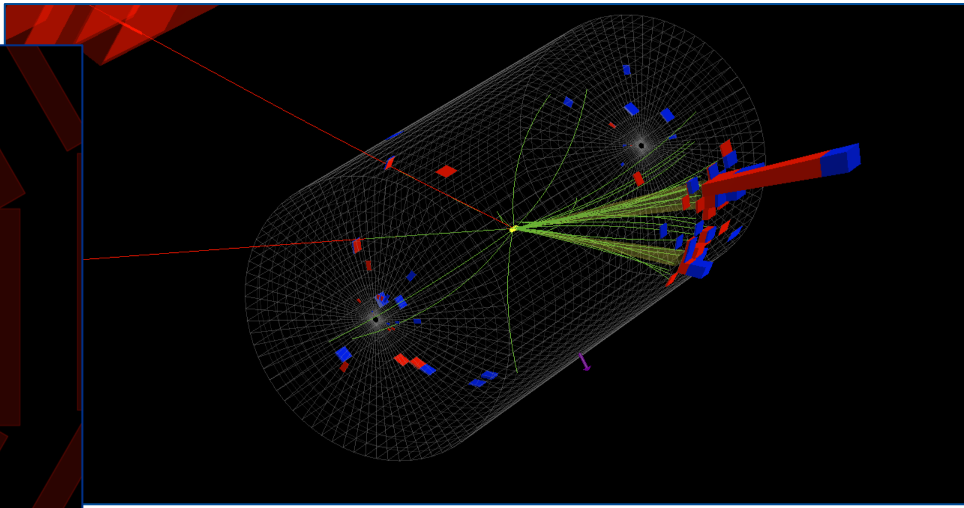




Results from the Jan 2016 CMS Use Case

- **All CMS simulation requests fulfilled for conference**
  - **2.9 million jobs, 15.1 million wall hours**
    - 9.5% badput – including preemption
    - 87% CPU efficiency
  - **518 million events generated**





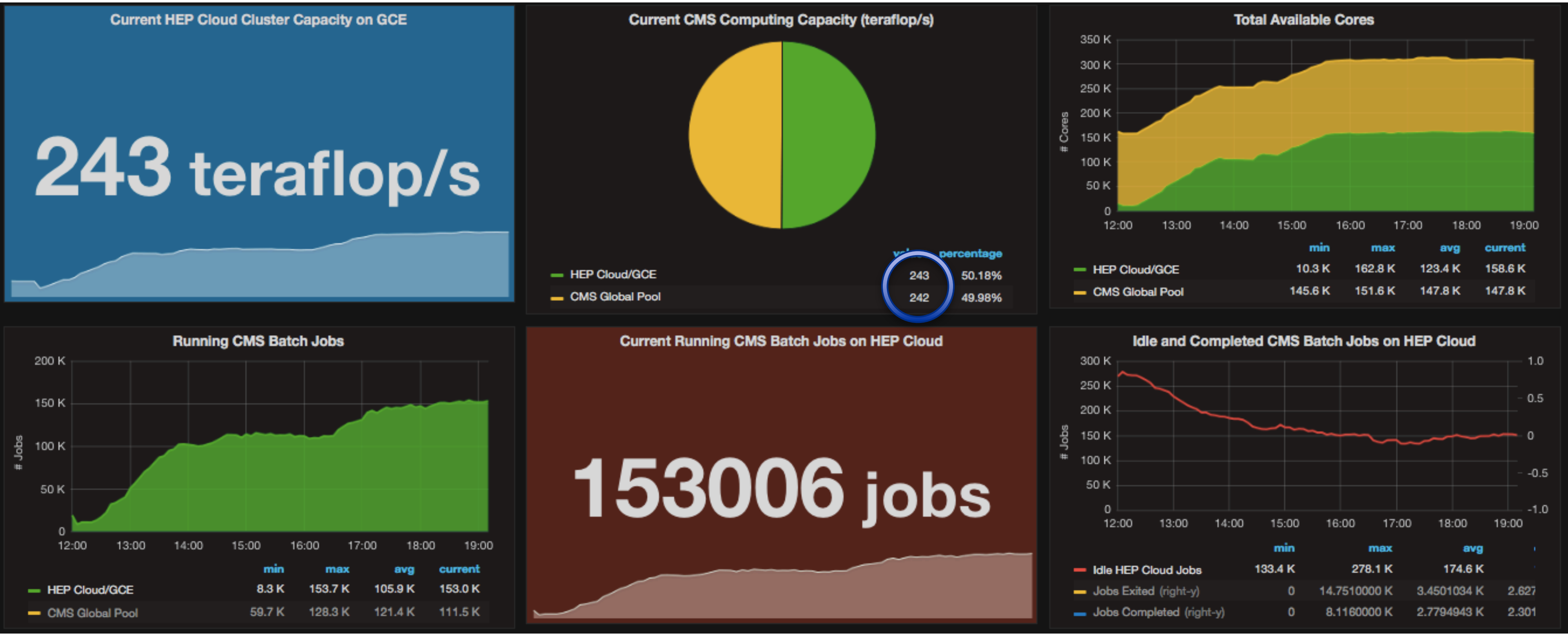
Results from the Jan 2016 CMS Use Case

- All CMS simulation requests fulfilled for conference
  - 2.9 million jobs, 15.1 million wall hours
    - 9.5% badput – including preemption
    - 87% CPU efficiency
  - 518 million events generated

- **Supercomputing 2016**
  - **Aiming to generate* 1 Billion events in 48 hours during Supercomputing 2016**
  - **Double the size of global CMS computing resources**

* 35% filter efficiency – 380 million events staged out

https://fifemon.fnal.gov/hcf/dashboard/db/hep-cloud-demo?from=now-12h&to=1479172229000



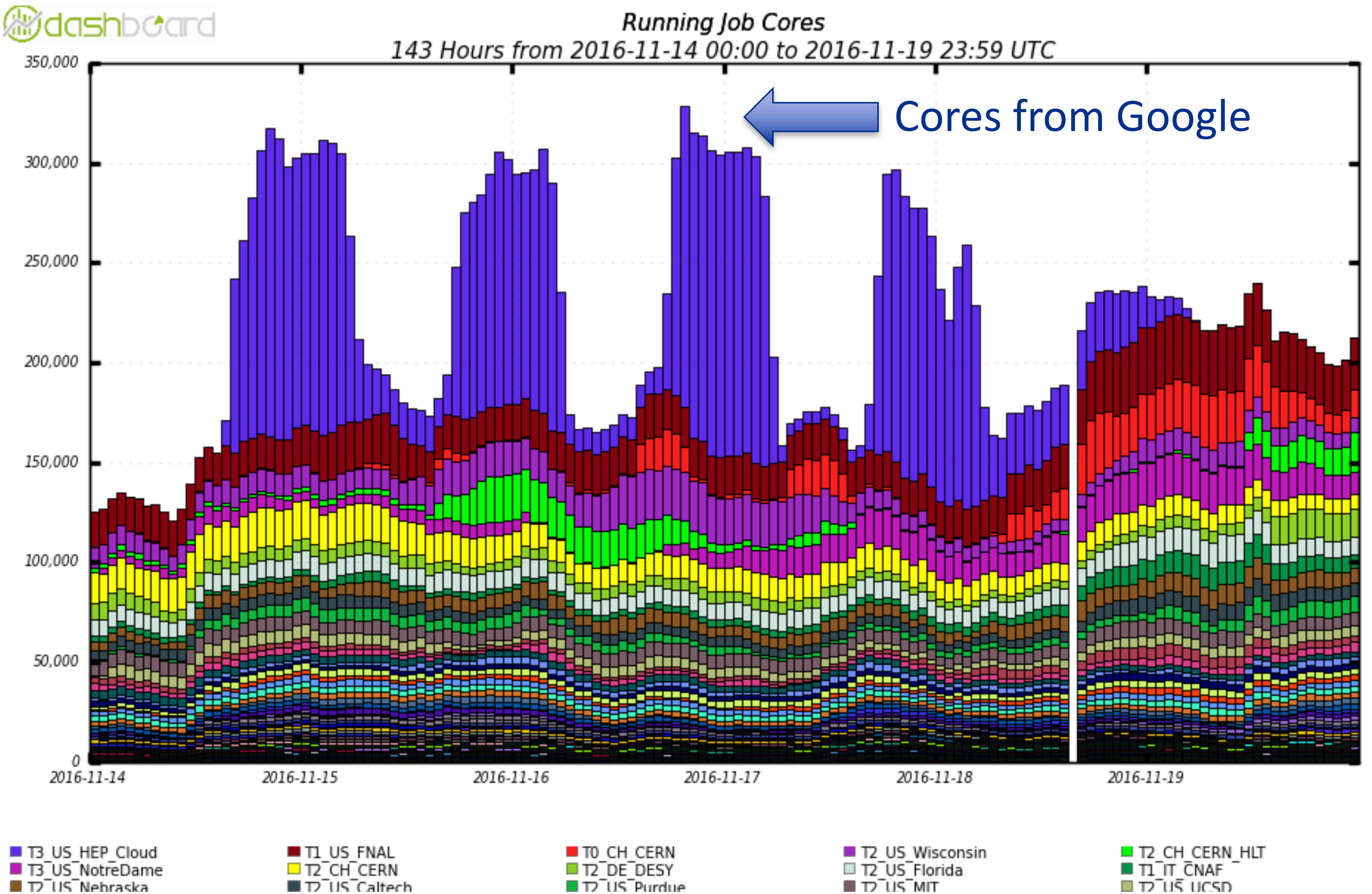
333	Software Company (M) United States	Cluster Platform DL380, Xeon E5-2673v3 12C 2.4GHz, 10G Ethernet HPE	19,320	485.3	741.9	644
334	Amazon Web Services United States	**Amazon EC2 C3 Instance cluster** - Amazon EC2 Cluster, Intel Xeon E5-2680v2 10C 2.800GHz, 10G Ethernet Self-made	26,496	484.2	593.5	609
335	Hosting Company United States	Cluster Platform DL380 Cluster, Xeon E5-2630v3 8C 2.4GHz, 10G Ethernet HPE	20,720	483.9	795.6	1,036

485

334 on the Top500 list?

Fermilab

Doubling CMS compute capacity



CMS @ Google – preliminary numbers

- 6.35 M wallhours used; 5.42 M wallhours for completed jobs.
  - 730172 simulation jobs submitted; only 47 did not complete through the CMS and HEPCloud fault-tolerant infrastructures
  - Most wasted hours during ramp-up as we found and eliminated issues; goodput was at 94% during the last 3 days.
- Used ~$100k worth of credits on Google Cloud during Supercomputing 2016
  - $71k virtual machine costs
  - $8.6k network egress
  - $8.5k disk attached to VMs
  - $3.5k cloud storage for input data
- 205 M physics events generated, yielding 81.8 TB of data

On-premises vs. cloud cost comparison - AWS

- Average cost per core-hour
  - On-premises resource: **.9** cents per core-hour
    - Includes power, cooling, staff
  - Off-premises at AWS: **1.4** cents per core-hour
    - Ranged up to 3 cents per core-hour at smaller scale
- Benchmarks
  - Specialized (“ttbar”) benchmark focused on HEP workflows
    - On-premises: **0.0163** (higher = better)
    - Off-premises: **0.0158**
- Raw compute performance roughly equivalent
- Cloud costs larger – but approaching equivalence
  - Still analyzing Google data; back-of-envelope ~ **1.6** cents per core-hour

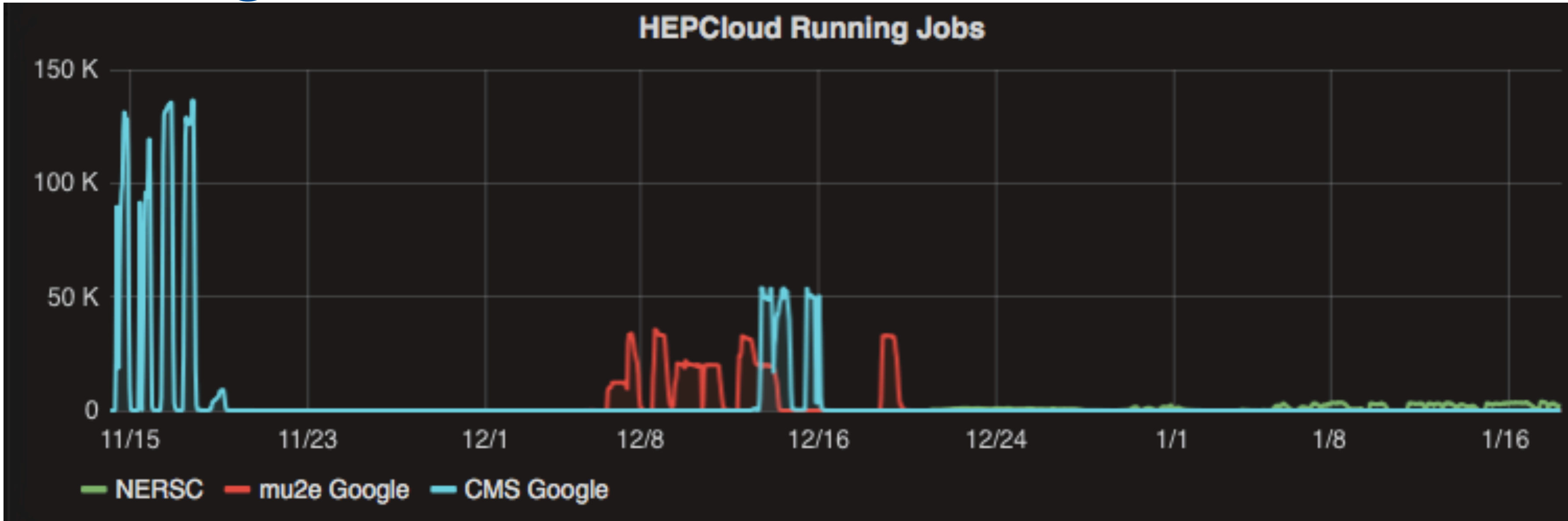
HEPCloud Compute and HPC

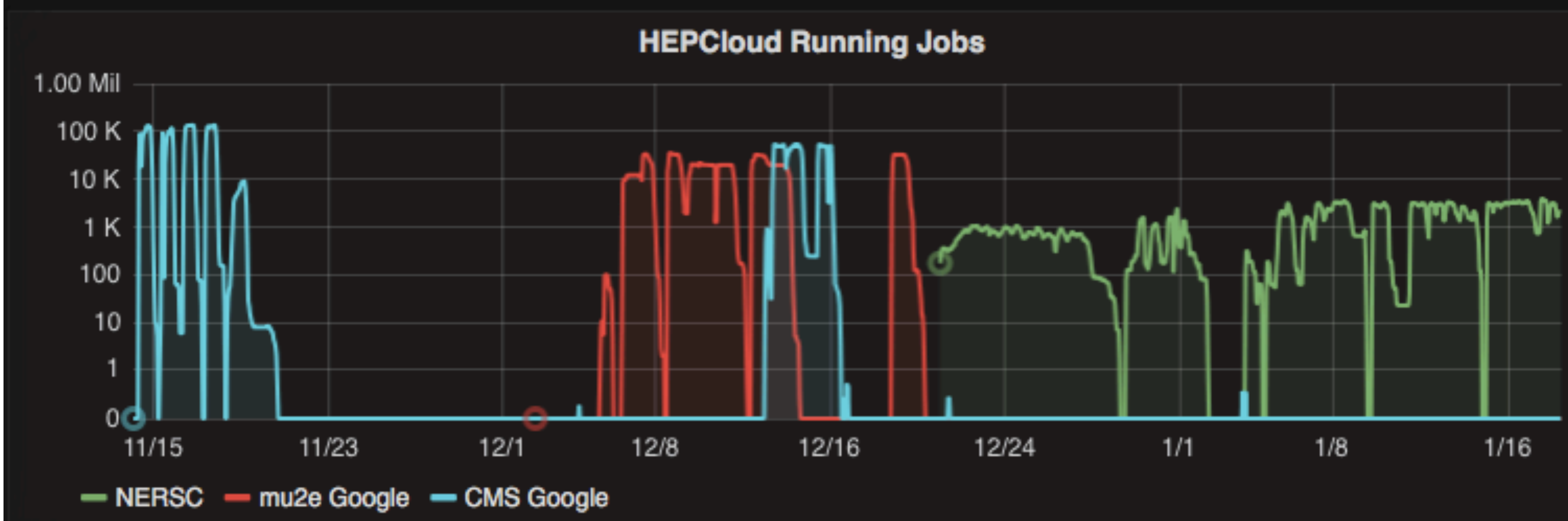
- A very appealing possibility, as we are approaching the exascale era, is to consider HPC facilities as a potential compute resource for HEPCloud
  - and, in the other direction, consider HEPCloud facility services (e.g. storage) as a potential resource for HPC facilities
- Investigate use cases with workflows that will allow such utilization within the constraints of allocation, security and access policy of HPC facilities.
- Initiate work with HPC facilities to fully understand constraints and requirements that will enable us to develop the HEPCloud process, policies and tools necessary for access of HPC resources

HEPCloud Compute and HPC

- Early steps: adapt HTC workflows to HPC facilities
  - MicroBooNE production on Cori @ NERSC
  - Pythia on Mira @ ALCF
  - CMS production on Edison, Cori @ NERSC
- **Plans for 2017:** HEPCloud provisioning @ NERSC
  - HEPCloud allocation granted for 28 million MPP-hours
    - 16 million MPP-hours for intensity frontier (mu2e, MicroBooNE, NOvA, …)
    - 12 million MPP-hours for CMS
      - CMS production will run Knight's Landing; experiment is working to optimize and maximize efficiency
  - Leverage experience with eye on leadership computing facilities

Running at NERSC





Thanks

- The Fermilab team:
  - Burt Holzman, Joe Boyd, Stu Fuess, Gabriele Garzoglio, Dirk Hufnagel, Hyun Woo Kim, Rob Kennedy, Krista Majewski, David Mason, Parag Mhashilkar, Neha Sharma, Steve Timm, Anthony Tiradani, Panagiotis Spentzouris
- The HTCondor and glideinWMS projects
- Open Science Grid
- Energy Sciences Network
- The Google team:
  - Karan Bhatia, Solomon Boulos, Sam Greenfield, Paul Rossman, Doug Strain
- The AWS team:
  - Sanjay Padhi, Jamie Baker, Jamie Kinney, Mike Kokorowski
- Resellers: Onix, DLT

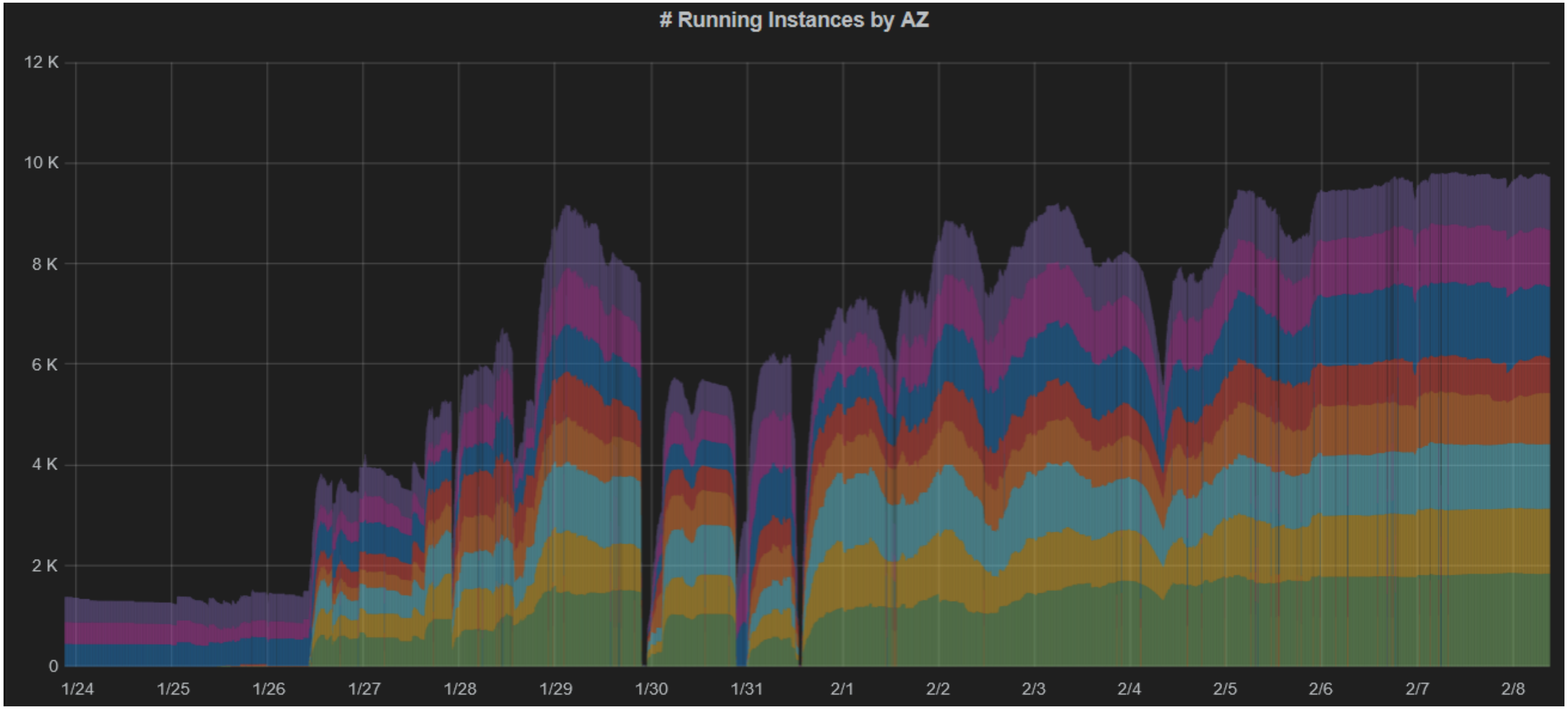
http://hepcloud.fnal.gov

Backup

HEPCloud Compute and HPC

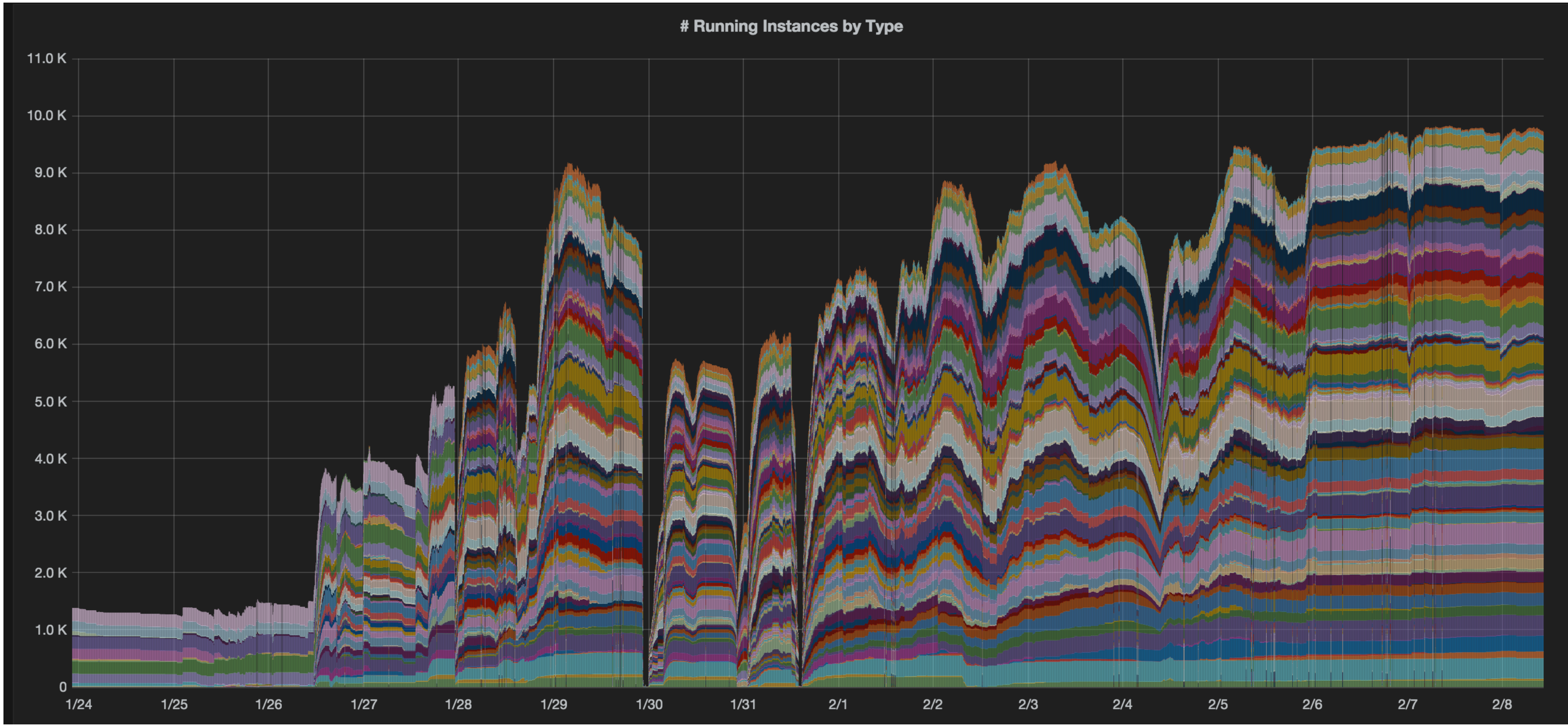
- **Early steps**: adapt HTC workflows to HPC facilities
  - MicroBooNE production on Cori @ NERSC
    - Successfully downloaded the entire MicroBooNE release, including LArSoft and the art framework onto Cori, using Shifter from dockerhub.
    - Executed single node tests of MicroBooNE Monte Carlo production, reading from and writing to the global scratch file system through the container
  - Pythia on Mira @ ALCF: multi-parameter tuning of event generators using collider data
    - MPI + multi-threading to execute 32k instances of Pythia and the Rivet analysis suite
    - Spirit of code-sharing – leveraged CMS contributions to multi-thread Pythia
  - **CMS production on Edison, Cori @ NERSC**: Provisioned resources and executed a variety of different GEN-SIM-DIGI-RECO workflows

HEPCloud AWS slots by Region/Zone



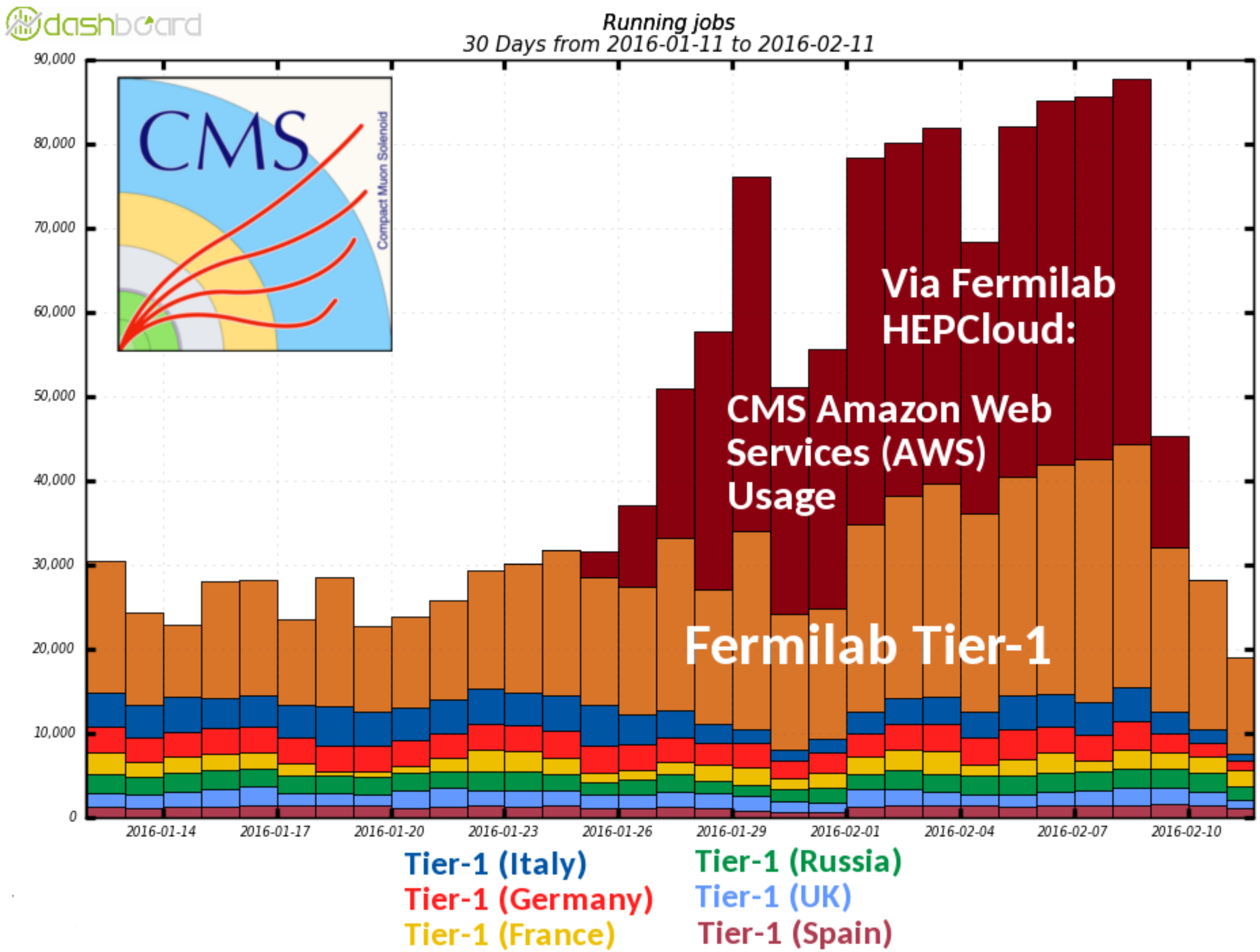
Each color corresponds to a different region+zone

HEPCloud AWS slots by Region/Zone/Type



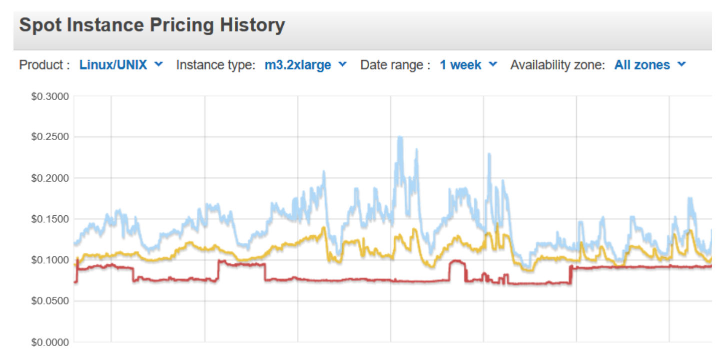
Each color corresponds to a different region+zone+machine type

Fermilab HEPCloud compared to global CMS Tier-1



VM Pricing: using the AWS "Spot Market"

- AWS has a fixed price per hour (rates vary by machine type)
- Excess capacity is released to the free ("spot") market at a fraction of the on-demand price
  - End user chooses a bid price
  - If (market price < bid), you pay only market price for the provisioned resource
    - If (market price > bid), you don't get the resource
  - If the price fluctuates while you are running and the market price exceeds your original bid price, you may get kicked off the node (with a 2 minute warning!)



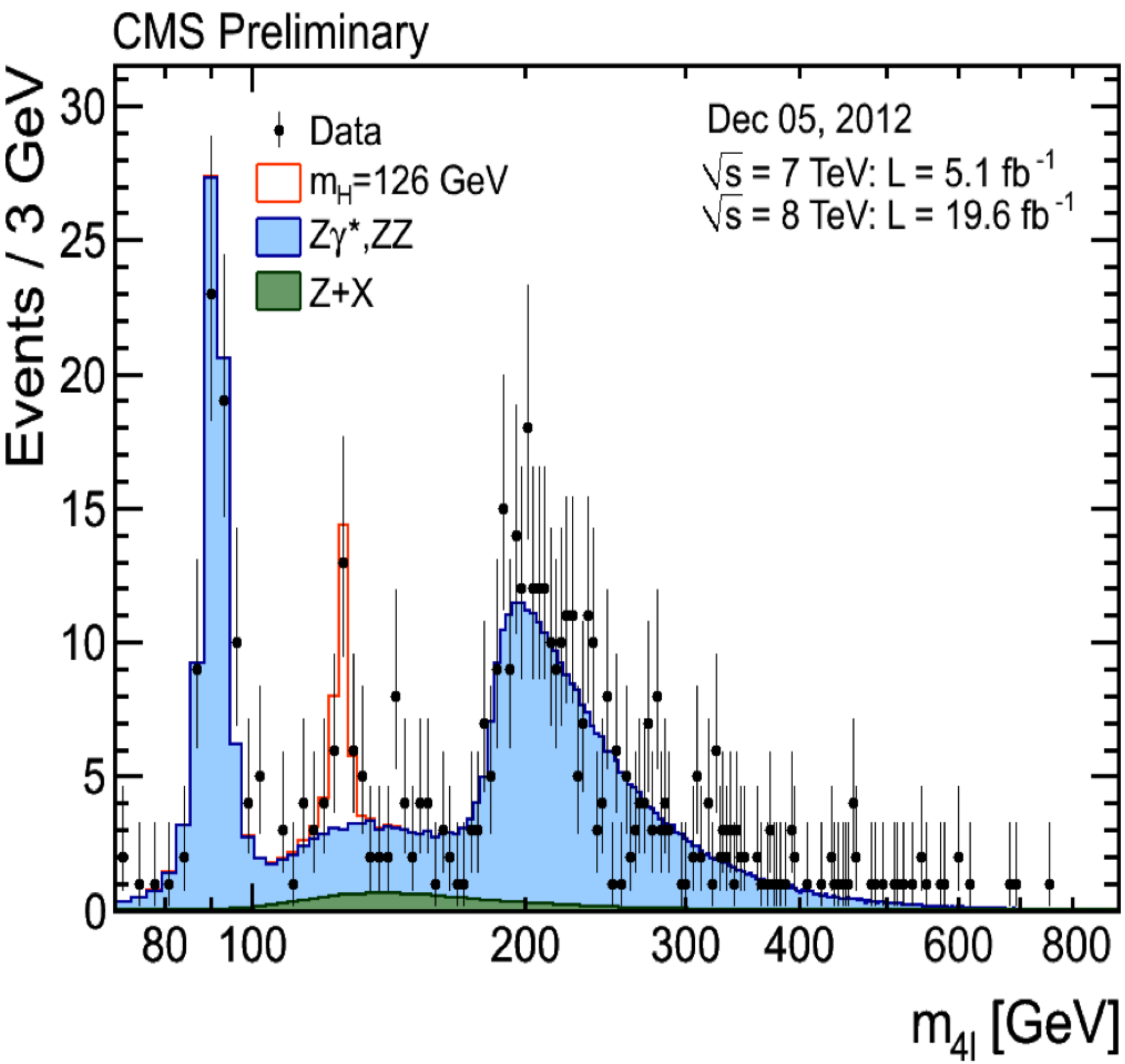
VM Pricing: using Google preemptible VMs

- Google VMs have a fixed cost (varies by machine types)
- Preemptible Google VMs are available at a significantly smaller fixed cost – 1 cent per core hour for a “standard candle”
  - We saved a few percent on cost by using custom VMs (2 GB per core instead of the standard 3.75 GB per core)

Running on Google Cloud - Google Services

- Distributing experiment code (many versions and codes)
  - CVMFS: caching layer using squid web caches
  - Scalable, easy-to-manage software distribution
  - Good fit for **Google Load Balancing**
- Reading input data
  - Staged 500 TB of input data to **Google Cloud Storage**
    - Standard HEP and CMS data management tools now speak http!
    - **Thanks to ESNet and Google** for upgraded (100 Gbit+) peering!
  - Mounted data using **gcsfuse**
    - Good for big serial reads
- Monitoring
  - **Stackdriver logging**
    - Splunk-like functionality – a big help for troubleshooting

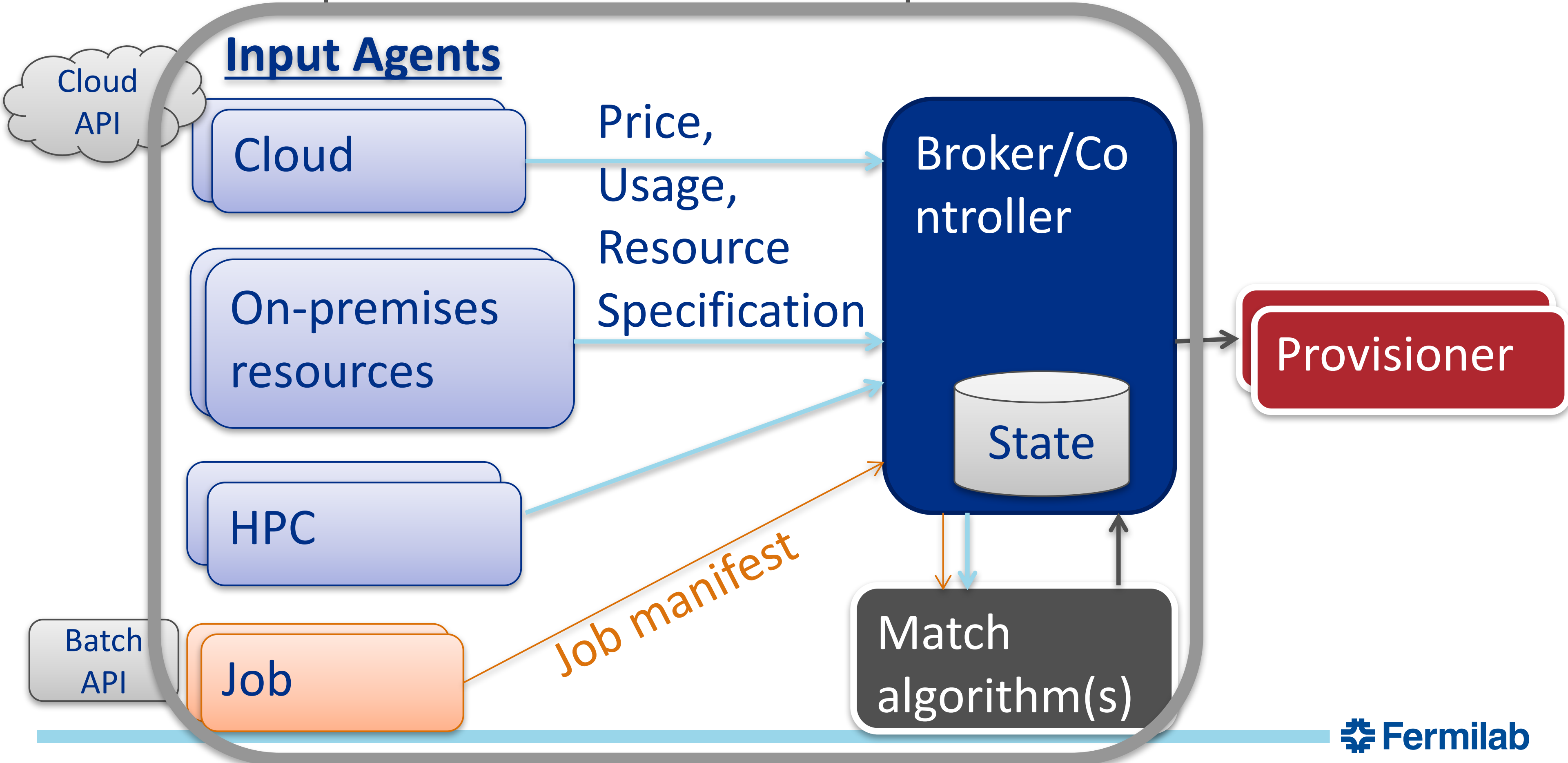
How is the science done?



- Particle Physics: **Statistical Science**
- Comparison with what we know (Standard Model)
- Analyze all data and look for deviations ➜ Needle in the Haystack

Decision Engine – design & architecture

- Decision Engine chooses what to provision next
  - v1.5 implementation: Strict matching based on processing type
  - v2.0 implementation: Zeroth-order prioritization based on cost



Pythia on Mira – Details

- We incorporated MPI into the main routines, using scatter and broadcast to send out unique parameters. The plan is to start one process on each node, running 64 threads, each with an instance of the pythia-based analysis. We will do this in chunk of 128 nodes, where each chunk is a gather collection point for writing to disk.
- Things were running on our x86 cluster - the porting to power PC of the build tools was the challenging part.
- ~150 core test

Description of CMS workflow

- Four chained steps (output of step N is input of step N+1)
  - Step 1 requires few GB input ("Gridpack") – same files per job
  - Step 2 requires additional input: "pile-up" data (simulating multiple events per bunch crossing), 5-10 GB

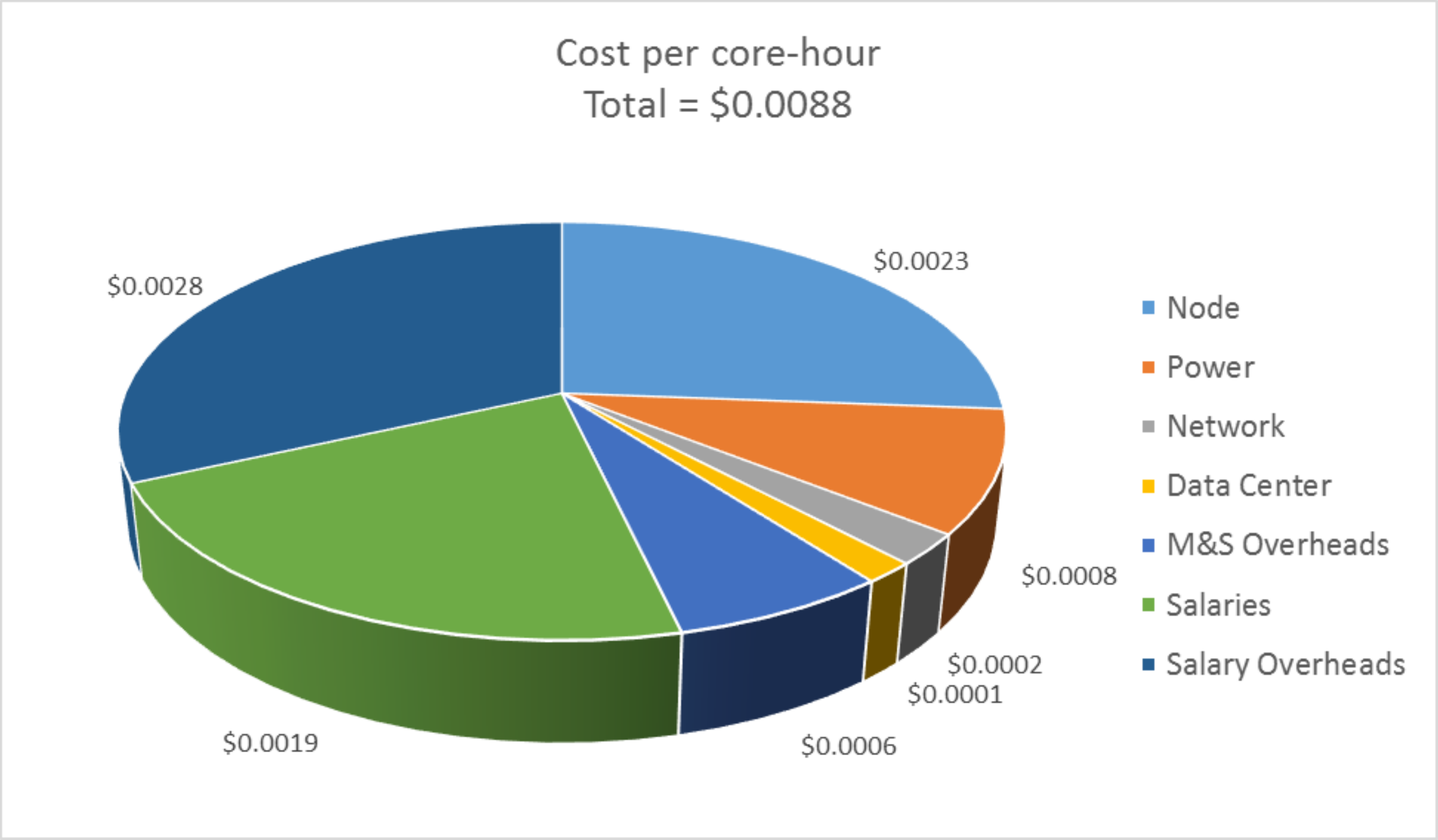
- Pile-up data is constructed on-the-fly by random seek and sequential reads into a 500 TB dataset
    - Staged pile-up datasets to Google Cloud Storage (storage service) ahead-of-time using FTS3 and PhEDEx – standard HEP grid tools and CMS data placement service

Reading pile-up from Google Cloud Storage

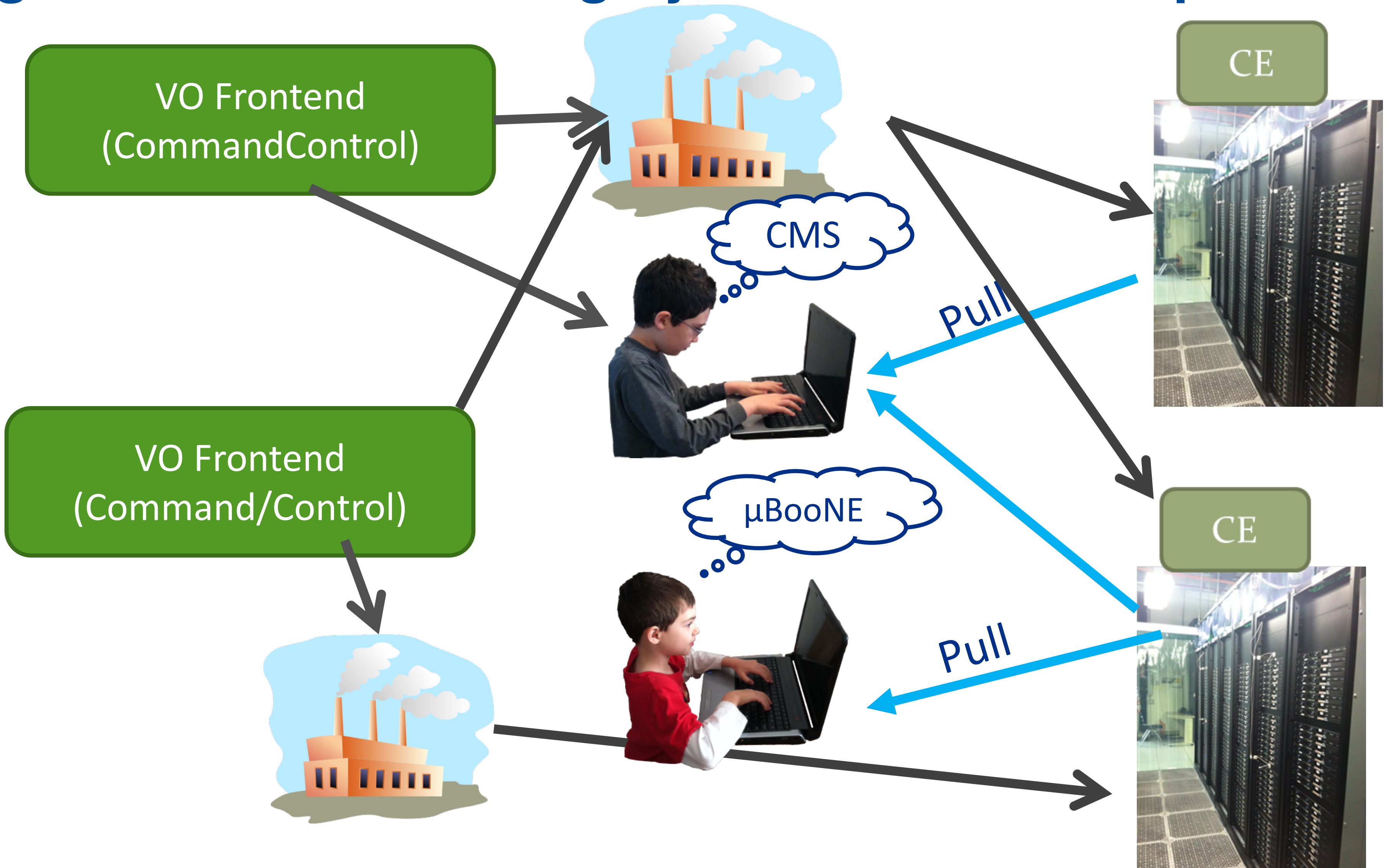
- Mounted regional bucket via gcsfuse on glide-in startup to /gcsfuse
- Used HTCondor “additional_json_file” functionality to specify role tied to image

Elements of the cost per core-hour

Based on Fermilab CMS Tier-1



glideinWMS – Building dynamic HTCondor pools



HEPCloud: Networking

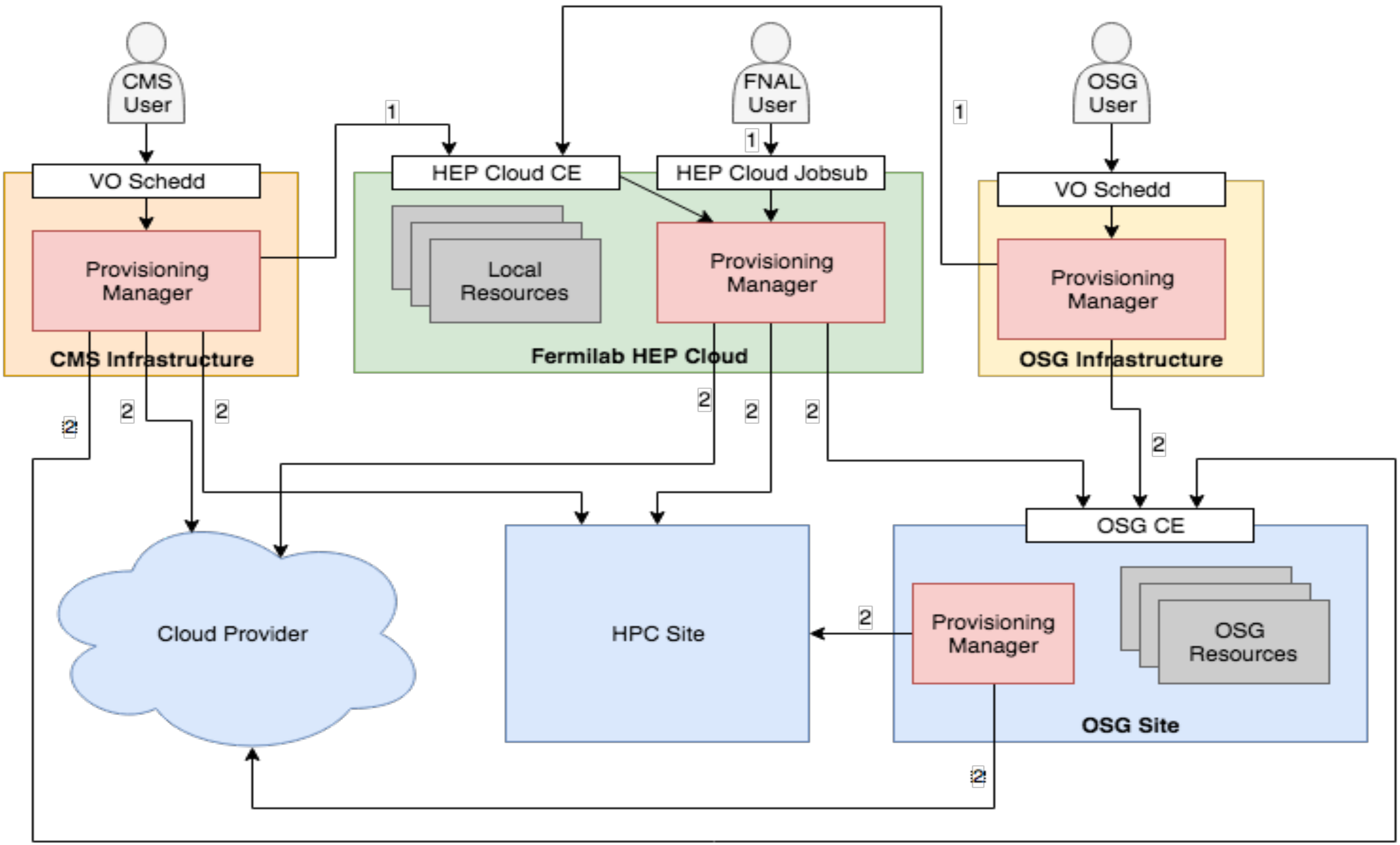
- All models of distributing computing rely on the performance of the underlying (local and wide-area) network
- Fermilab is approaching **1 Terabit** data center – connect to Energy Sciences Network (ESNet) at 4*100 Gigabit
  - ESNet enables distributed computing beyond ESNet sites: 100 Gigabit peering points with other networks

- Zone-based security protection of network resources
- On-demand (**Software Defined Network**-based) traffic controls
- Virtualization of network resources

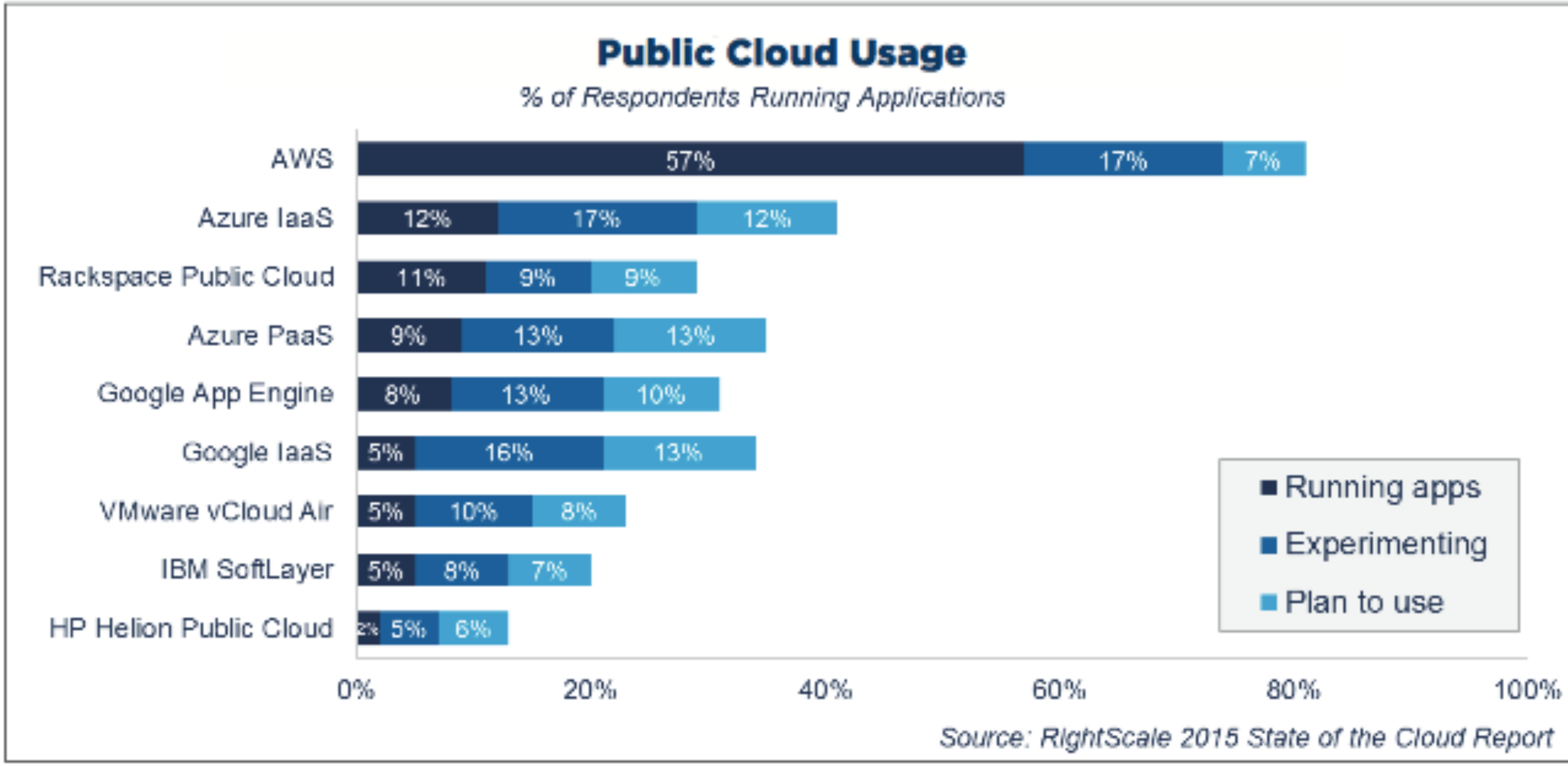
HEPCloud: Storage

- Data is the lifeblood of science
  - HEP experiments generate it by the station-wagon-load
  - Fermilab is a leader in the field in storing and serving petabytes of data to the world
- We are working with industry and other collaborators to modernize our services
  - Data storage and retrieval
  - Data cataloging
  - Support multiple-layer storage infrastructure approach
- One part of HEPCloud is to understand how to integrate all of these components – always driven by the experiment needs, both present and future

User’s View of HEPCloud

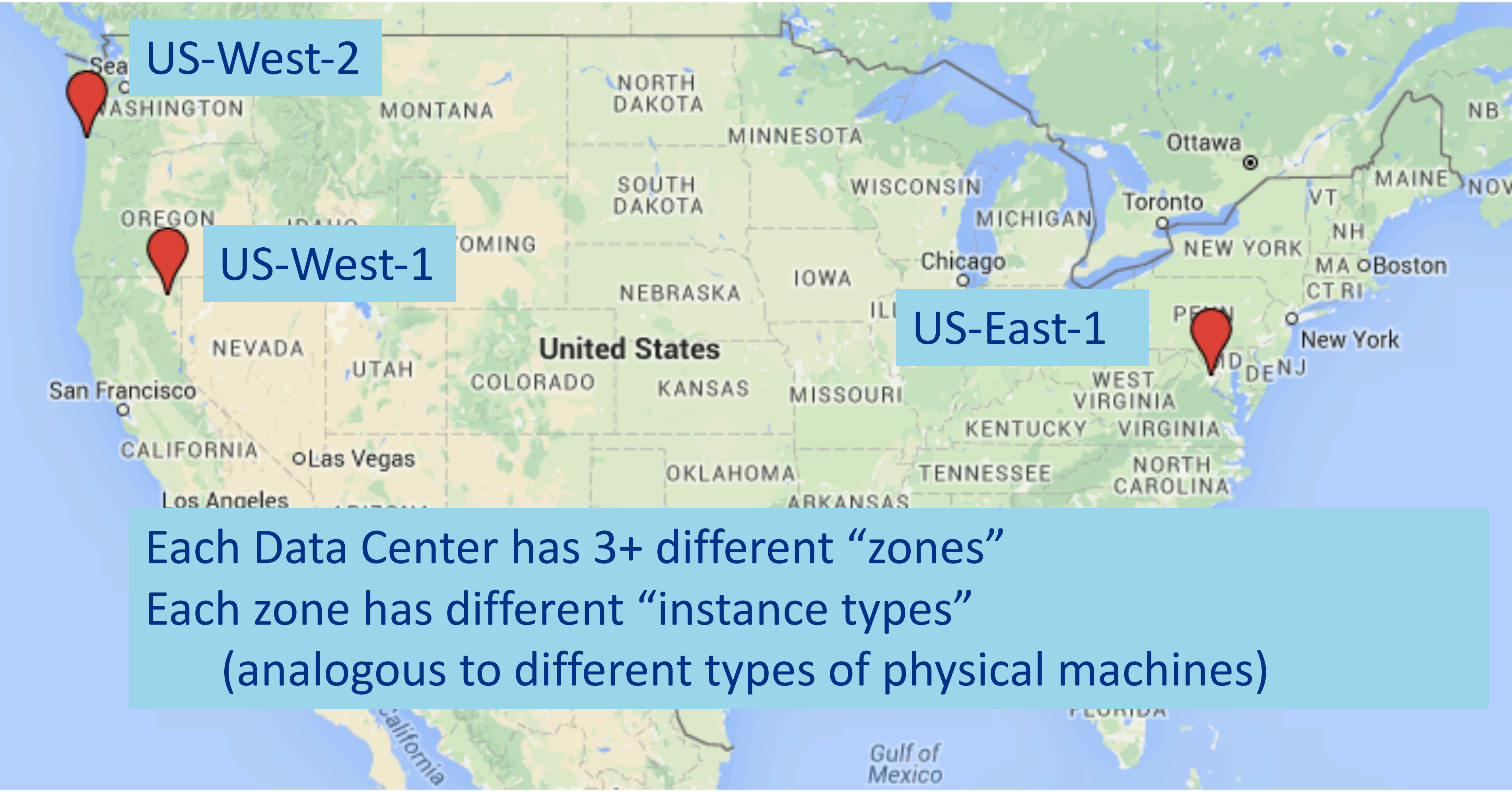


Fermilab HEPCloud: expanding to the Cloud



Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof.

AWS topology – three US data centers (“regions”)



Each Data Center has 3+ different “zones”

Each zone has different “instance types”

(analogous to different types of physical machines)

Fermilab

Reading pile-up from AWS S3 (storage)

- AWS worker nodes granted permission to read from AWS S3 folder ("bucket") via AWS Security-Token-Service (STS)
- ROOT has a TS3WebFile class!
  - But session key support was missing (needed for STS!)

Add support for session keys in TS3WebFile

(some minor revision also by the committer)

Browse files

master v6-07-04

holzman committed with **smithdh** on Dec 1, 2015

1 parent 55ed62b   commit fe169587a0dc681a33ecdd33544c32cbeb43d3b7

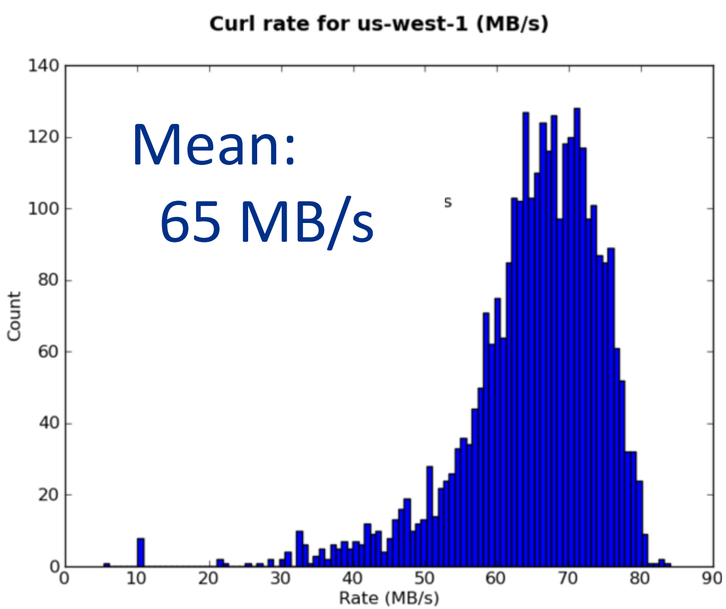
Showing 4 changed files with **73 additions** and **14 deletions**.

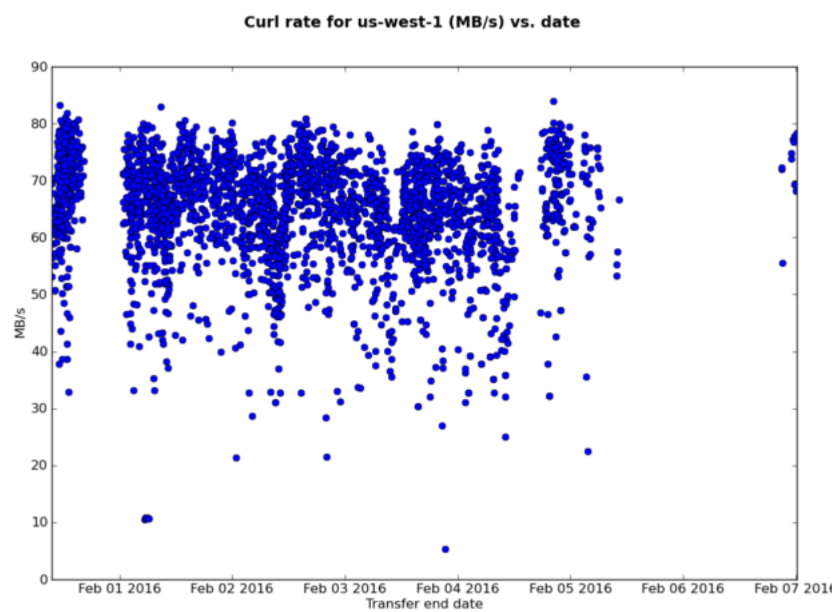
Unified Split

- This worked great!
  - Except…

Reading pile-up from AWS S3 (storage)

- Cost of data access was 30% of compute costs
  - 150 million HTTP GETs per hour is a lot!
- Wrote a curl wrapper to provide the custom AWS authentication headers
  - (Not often I can say I reduced costs by 5 orders of magnitude!)





Late 2016 HEPCloud Use Cases - Google

NoVA Processing

Processing the 2014/2015 dataset
16 4-day "campaigns" over one year
Demonstrates stability, availability, cost-effectiveness
Received AWS academic grant

CMS Monte Carlo Simulation

Generation (and detector simulation, digitization, reconstruction) of simulated events in time for Moriond16 conference
56000 compute cores, steady-state
Demonstrates scalability
Received AWS academic grant

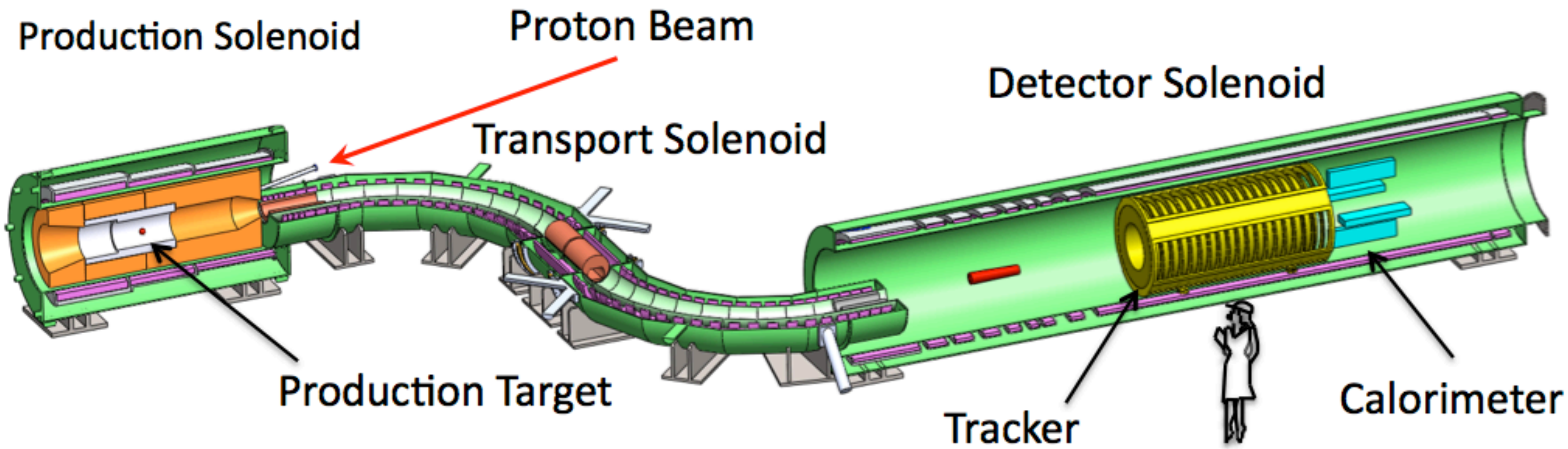
CMS Monte Carlo Simulation

Generation (and detector simulation, digitization, reconstruction) of simulated events in time for Moriond17 conference
160000 compute cores during Supercomputing 2016 conference (~48 h)
Demonstrates scalability, capability
Received Google Cloud Platform grant

mu2e Processing

Simulating cosmic ray veto detector and beam particle backgrounds
3M integrated core-hours
Demonstrates rapid on-boarding
Received Google Cloud Platform grant

Mu2e experiment



- Charged Lepton Flavor Violation is a near-universal feature of extensions to the Standard Model of particle physics
- Rare muon processes offer the best combination of new physics reach and experimental sensitivity
- **Search for muon (in bound state) converting to an electron ("mu" to "e")**

Mu2e – executing on Google Cloud