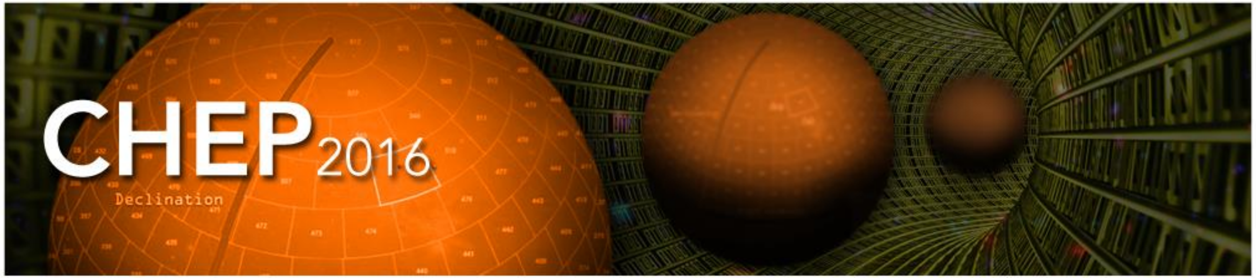
Experience Using Commercial Clouds in CMS

Maria Girone
October 13th, 2016



22nd International Conference on Computing in High Energy and Nuclear Physics, Hosted by SLAC and LBNL, Fall 2016

on behalf of the CMS collaboration

The CMS AWS team

L Bauerdick, B Bockelman, D Dykstra, I Fisk, S Fuess, G Garzoglio, M Girone, O Gutsche, B Holzman, H Kim, R Kennedy, D Hufnagel, N Magini, D Mason, P Spentzouris, A Tiradani, S Timm and E Vaandering

Opportunities with HEPCloud and AWS

- **HEPCloud** Facility as a portal to an ecosystem of diverse computing resources academic or commercial
  - Allows to burst to very high capacity
    - First type of external resources considered was Amazon Web Service with CMS being a driving use case

- **AWS** academic 9 to 1 matching grant awarded to CMS from June 2015 to March 2016
  - Size of award based on what it would cost to to do one month of large-scale processing **(56k cores, $20k invested by CMS)**
  - Conditional waiver for exporting data if export costs <15% of total monthly bill
    - Use of research networks like ESnet

See presentations from G. Garzoglio "The HEP Cloud Facility: elastic computing for High Energy Physics" and
S. Timm "Virtual Machine Provisioning, Code Management and Data Movement Design for the Fermilab HEPCloud Facility"

Goals of the CMS AWS Project

- One of the use cases of the Fermilab HEPCloud effort
  - IaaS through Cloud computing for elasticity, without over-provisioning local resources

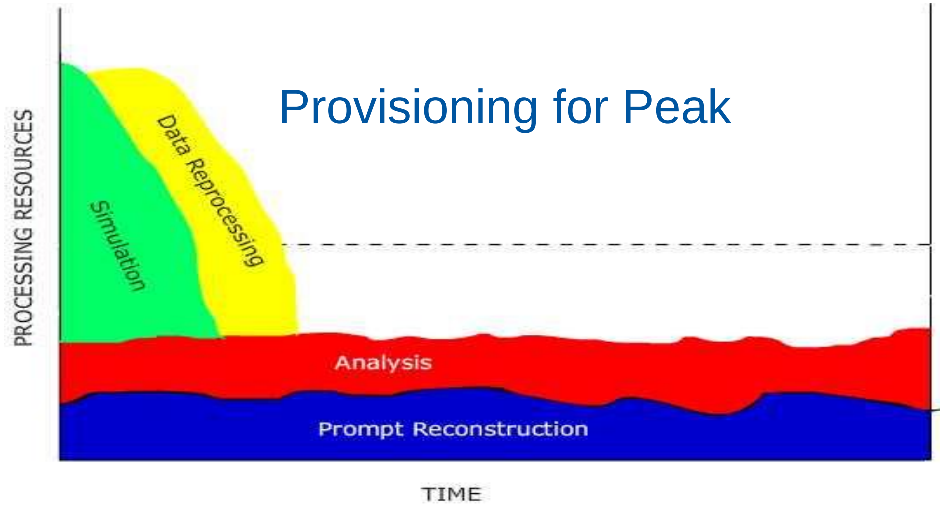
- Demonstrate the capability of executing in production any of the CMS centrally organized workflows on AWS
  - Demonstrate scalability up to additional 56000 compute cores, steady state
  - Contribute significantly to a official CMS data production for the 2016 winter conferences

- Demonstrate the ability to significant augment the peak capacity of the CMS production system using commercial computing resources
  - Investigate scheduling for peak
  - Investigate the cost, reliability, and efficiency of large-scale commercial computing resources

This work has been documented in the note FERMILAB-PUB-16-170-CD

Scheduling for Peak

- Experience from Run1 and 2 shows that computing needs are not constant over time
  - Many of the activities come in bursts
    - Software releases, conferences, data taking schedules
- Elasticity of bursting resources into commercial clouds would allow for shorter and more agile time schedules
  - **Powerful source for problem recovery**
- Large commercial resource pools could enable this if we are able to use them effectively and at similar costs of our dedicated resources



M. Girone, Plenary CHEP 2015, Distributed Data Management and Distributed File Systems
iopscience.iop.org/article/10.1088/1742-6596/664/4/042022/pdf

Workflows

Several production workflows based on physics value, current needs, difficulty of the workflow, effective use of resources

CPU-Intensive

- **Generation and simulation (GEN-SIM)**
  - CPU-bound, single core, requires only parameters as input

Data-Intensive

- **Data reconstruction (Data-RECO)**
  - Raw events served by a global data federation, every job needs to be completed to maintain the integrity of the dataset
- **Event crossing simulation and reconstruction (DIGI-RECO)**
  - Input data from previous step to be served over a data federation plus additional minimum bias events for full crossing to be served from local storage pre-staged to S3

PREFERRED WORKFLOW

- **Generation and event crossing simulation and reconstruction (GEN-SIM-DIGI-RECO)**
  - Combined *all* elements in a single chain, higher amount of processing per event wrt DIGI-RECO within the 15% data egress waiver
  - Requires only parameters as input data and produces summarized analysis formats

The Four-steps Workflow

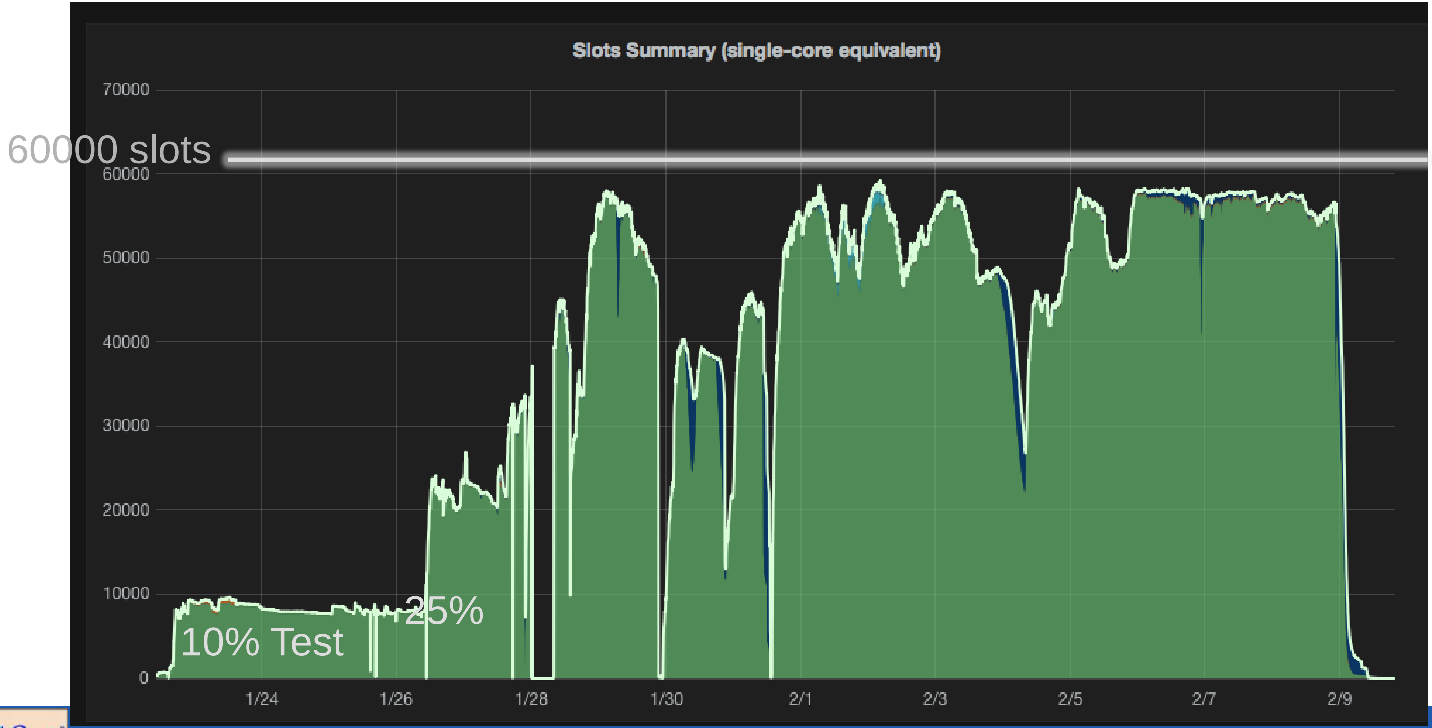
- For the large-scale run, four large Standard Model background samples were produced in full chain GEN-SIM-DIGI-RECO
  - Drell-Yan+Jets to dilepton for mass ranges of 10{50 GeV (DY M10-50)
  - Drell-Yan+Jets to dilepton for mass > 50 GeV (DY M50)
  - t-tbar to jets (TTJets)
  - W+Jets to lepton/neutrino (WJetstoLNu)

- Only AoD and mini-AoD were kept and staged out to the Fermilab EOS storage system

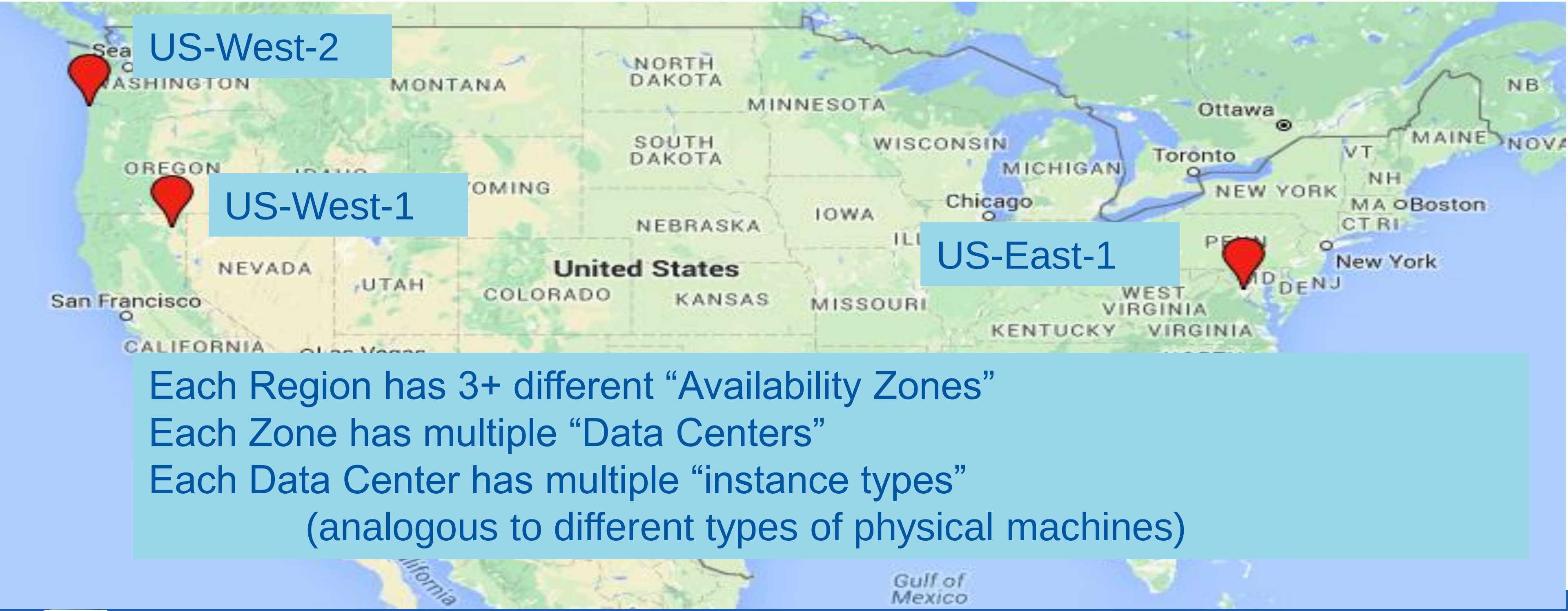
- IO CHALLENGES

- The classic mixing scenario of CMS reads hundreds of minimum bias event objects to simulate a crossing
  - The AWS S3 storage system was able to handle the load, but as AWS has a very small charge per read, a huge number of reads added up quickly!!!
- The minimum bias files needed to be staged to local storage attached to the VMs
  - Can be an interesting technique to apply to dedicated facilities struggling to keep up with the load

Reaching ~60k slots on AWS FNAL HEPCloud



AWS topology: 3 US “Regions”



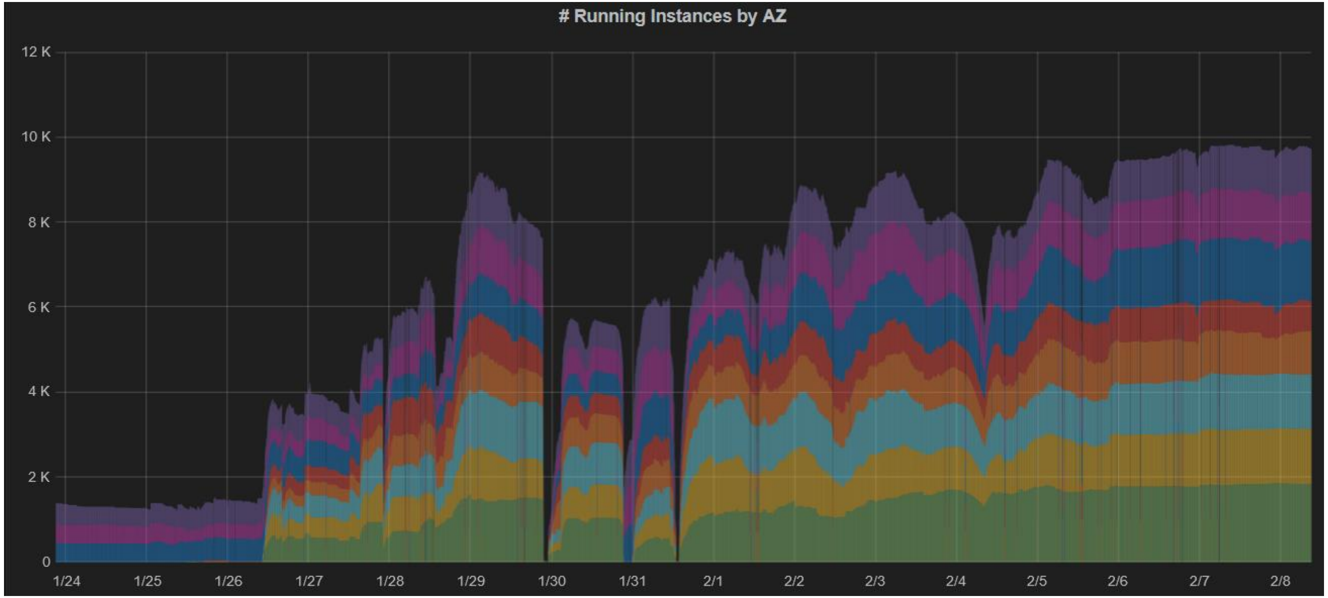
Each Region has 3+ different “Availability Zones”

Each Zone has multiple “Data Centers”

Each Data Center has multiple “instance types”

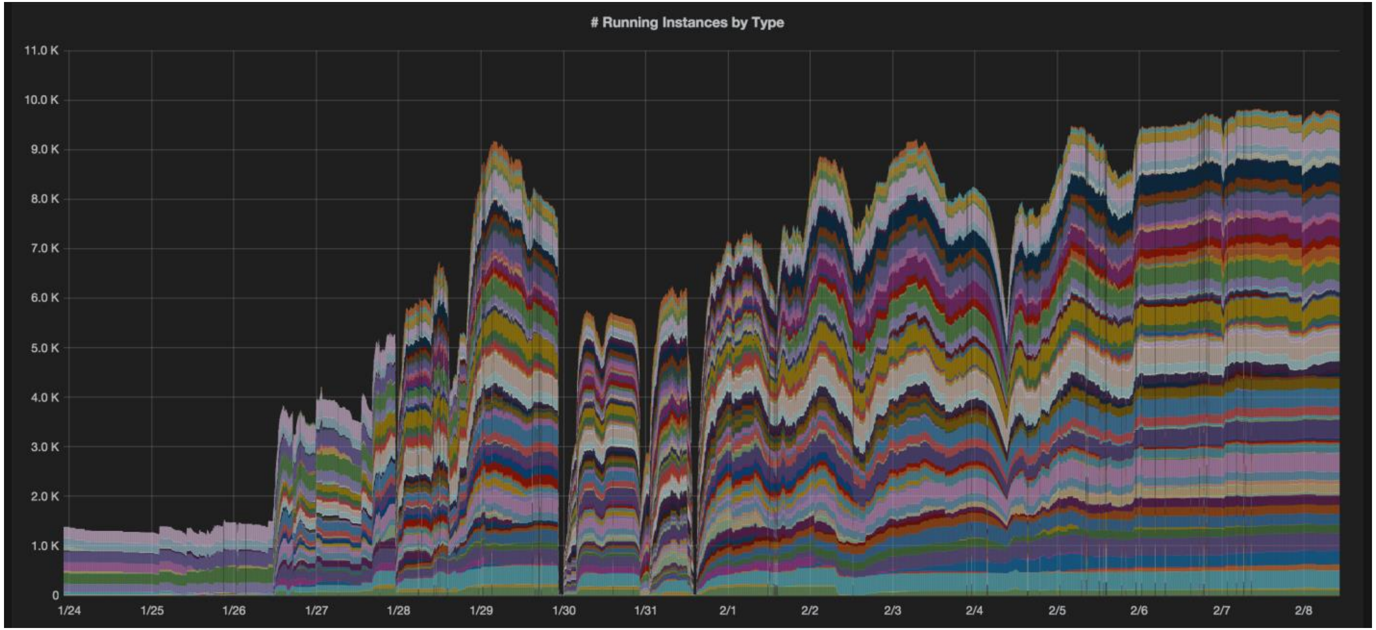
(analogous to different types of physical machines)

HEPCloud AWS slots by Region/Zone



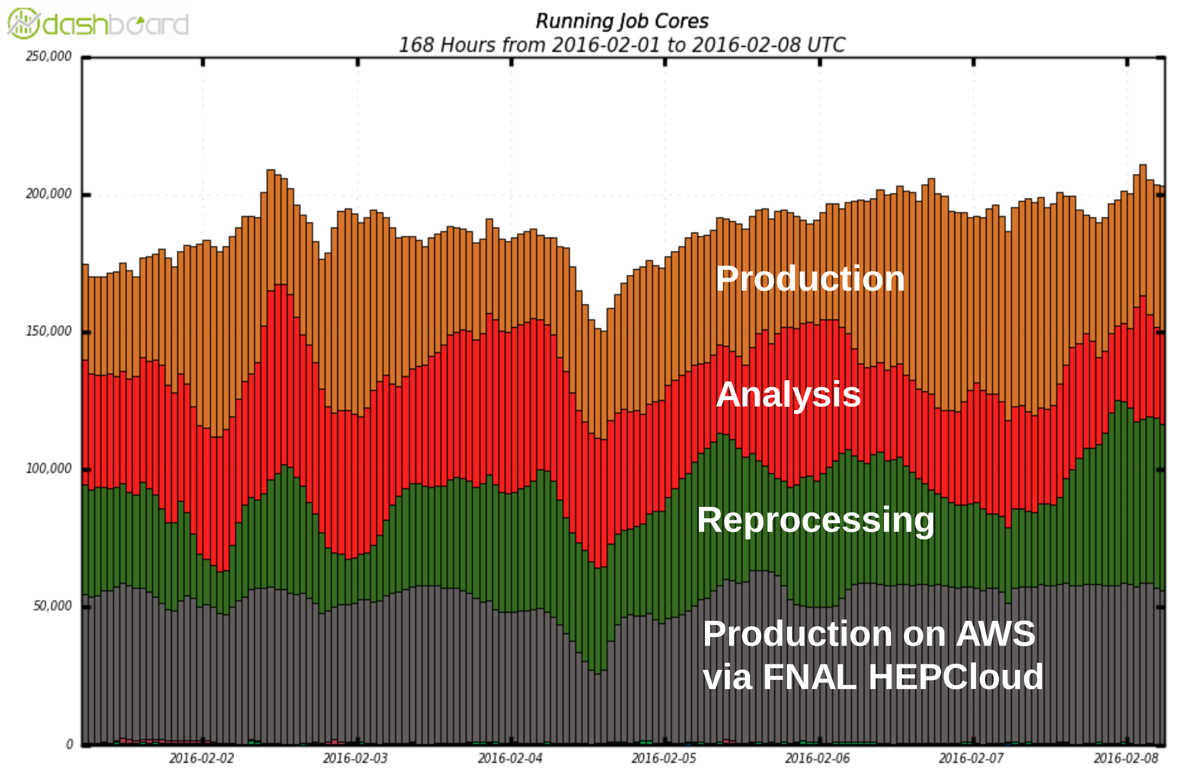
Each color corresponds to a different region+zone

HEPCloud AWS slots by Region/Zone/Type

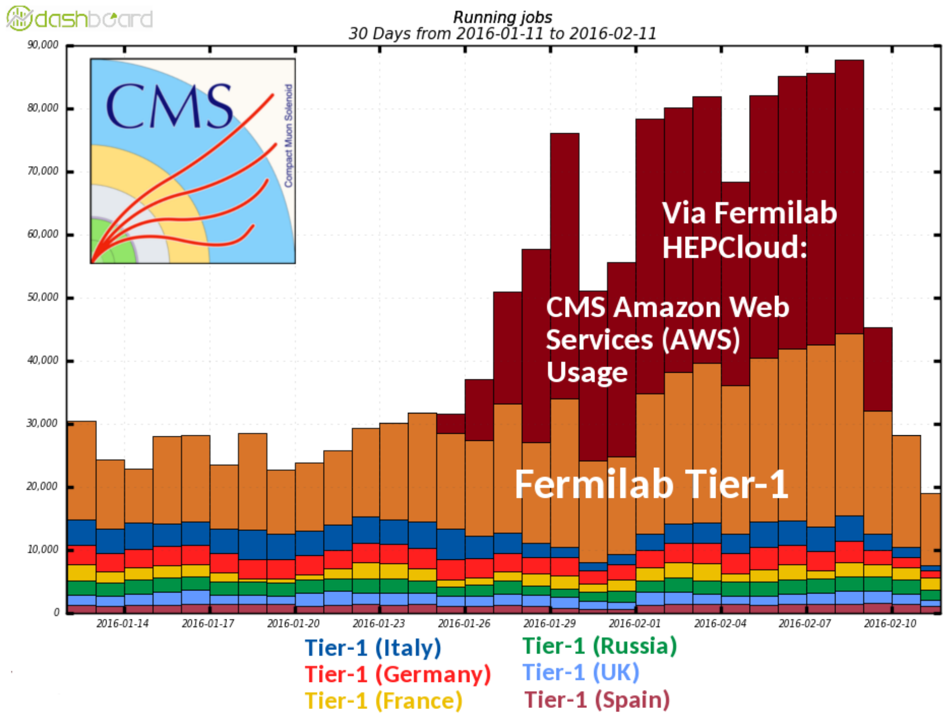


Each color corresponds to a different region+zone+machine type

HEPCloud AWS: 25% of CMS Global Capacity



Fermilab HEPCloud wrt Global CMS Tier-1

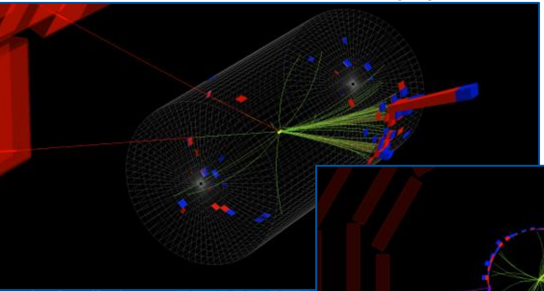


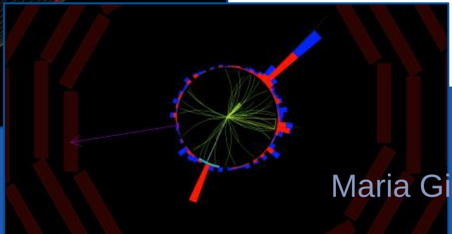
Scale achieved

- **All CMS simulation requests fulfilled for Moriond 2016**
  - **2.9 million jobs, 15.1 million wall hours**
    - 9.5% badput – includes preemption from spot pricing
    - 87% CPU efficiency
  - **518 million events generated**

/DYJetsToLL_M-50_TuneCUETP8M1_13TeV-amcatnloFXFX-pythia8/RunIIFall15DR76-PU25nsData2015v1_76X_mcRun2_asymptotic_v12_ext4-v1/AODSIM
/DYJetsToLL_M-10to50_TuneCUETP8M1_13TeV-amcatnloFXFX-pythia8/RunIIFall15DR76-PU25nsData2015v1_76X_mcRun2_asymptotic_v12_ext3-v1/AODSIM
/TTJets_13TeV-amcatnloFXFX-pythia8/RunIIFall15DR76-PU25nsData2015v1_76X_mcRun2_asymptotic_v12_ext1-v1/AODSIM
/WJetsToLNu_TuneCUETP8M1_13TeV-amcatnloFXFX-pythia8/RunIIFall15DR76-PU25nsData2015v1_76X_mcRun2_asymptotic_v12_ext4-v1/AODSIM

Number of times a job was executed





Cost Comparison

- Average cost per core-hour
  - On-premises resource: **.9** cents per core-hour (assumes 100% utilization)
    - 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: **0.163 ttbar/s**
    - Off-premises: **0.158 ttbar/s**

Raw compute performance roughly equivalent
Cloud costs larger – but approaching equivalence

Outlook

- The tests performed by Fermilab and CMS on AWS have demonstrated that is possible to utilize dynamically provisioned commercial Cloud resources
  - Proven capability to execute efficiently **both** data intensive and CPU intensive workflows
  - HEPCloud facility was able to add by 25% total CMS resources

- AWS required the workflows to be carefully chosen and tuned and the workflow system needed to be adjusted to handle a larger dynamic range
  - We could not identify any workflow that could not be executed

- AWS was much more competitive in terms of cost than in previous investigations
  - Potentially an interesting resource to supplement our dedicated resources

Backup Slides

Services

- Services deployed in AWS
  - Squid and Frontier
    - AWS CloudFormation as orchestrator
      - Elastic Load Balancer
      - Auto Scaling Group and polices
      - Squid-optimized Amazon Machine Image
      - Route 53 DNS CNAME Record Set
      - CloudWatch alarms
    - AWS Network Configuration
    - AWS Limits
    - AWS Spot Instances
  - Services deployed at Fermilab
    - Accounting and billing
    - glideinWMS
    - Reporting
    - Operational Setup with HTCondor pool

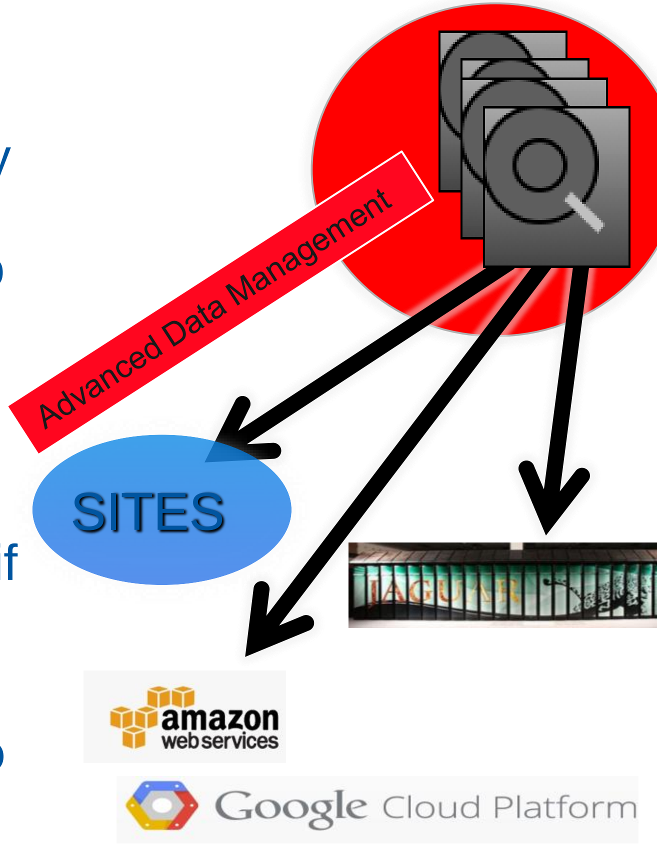
The Four-steps Workflow

- For the large-scale run, four large Standard Model background samples were produced in full chain GEN-SIM-DIGI-RECO
  - Drell-Yan+Jets to dilepton for mass ranges of 10{50 GeV (DY M10-50)
  - Drell-Yan+Jets to dilepton for mass > 50 GeV (DY M50)
  - t-tbar to jets (TTJets)
  - W+Jets to lepton/neutrino (WJetstoLNu)
- Only AoD and mini-AoD were kept and staged out to the Fermilab EOS storage system

GEN: Madgraph 4-vector event generation
SIM: Propagation of particles through the GEANT detector simulation
DIGI: Electronics simulation and event digitization, including mixing in pile-up data pre-staged to S3 storage
RECO: Reconstruction of the data into physics quantities used in analysis

Data Management

- Data storage and data export on AWS costs money
  - Data ingest does not
- The AAA project allows input data to be served into running applications on AWS
  - Does not use us resources storing data locally and advanced read-ahead and good networking into AWS maintains high CPU efficiency
- Data export on AWS is expensive and only waved if the charges are less than 15% of the processing costs
  - Workflows that produced small fnal output like AOD and MiniAOD were economically more favorable than RECO or any intermediate simulation formats



Challenges

- Running on AWS in this test represented a 50% increase in the total Tier-1 computing resources
  - Since we were concentrating on 3 large workflows it also involved ramping a lot of jobs quickly
  - Unsurprisingly there were some adjustments needed in the workflow management system and pilot factories

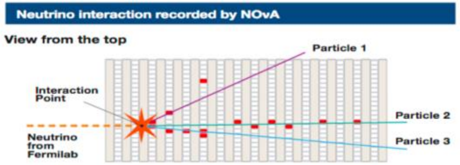
- One of the most interesting cost limitations was in IO

- The classic mixing scenario of CMS reads hundreds of minimum bias event objects to simulate a crossing
  - The AWS storage system was able to handle the load, but they have a very small charge per read, but very small charges multiplied by a huge number of reads add up!
- The minimum bias files needed to be staged to local storage
  - interesting technique to apply to dedicated facilities struggling to keep up with the load

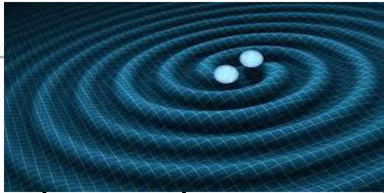
Initial HEPCloud Use Cases

NoVA Processing

Processing the 2014/2015 dataset
3 use cases: Particle ID, Montecarlo, Data Reconstruction
Received AWS research grant



Dark Energy Survey Gravitational Waves



Search for optical counterpart of events detected by LIGO/VIRGO gravitational wave detectors (FNAL LDRD)
Modest CPU needs, but want 5-10 hour turnaround
Burst activity driven entirely by physical phenomena (gravitational wave events are transient)
Rapid provisioning to peak

CMS Monte Carlo Simulation

Generation (and detector simulation, digitization, reconstruction) of simulated events in time for Moriond conference 58000 compute cores, steady-state
Demonstrates scalability
Received AWS reasearch grant