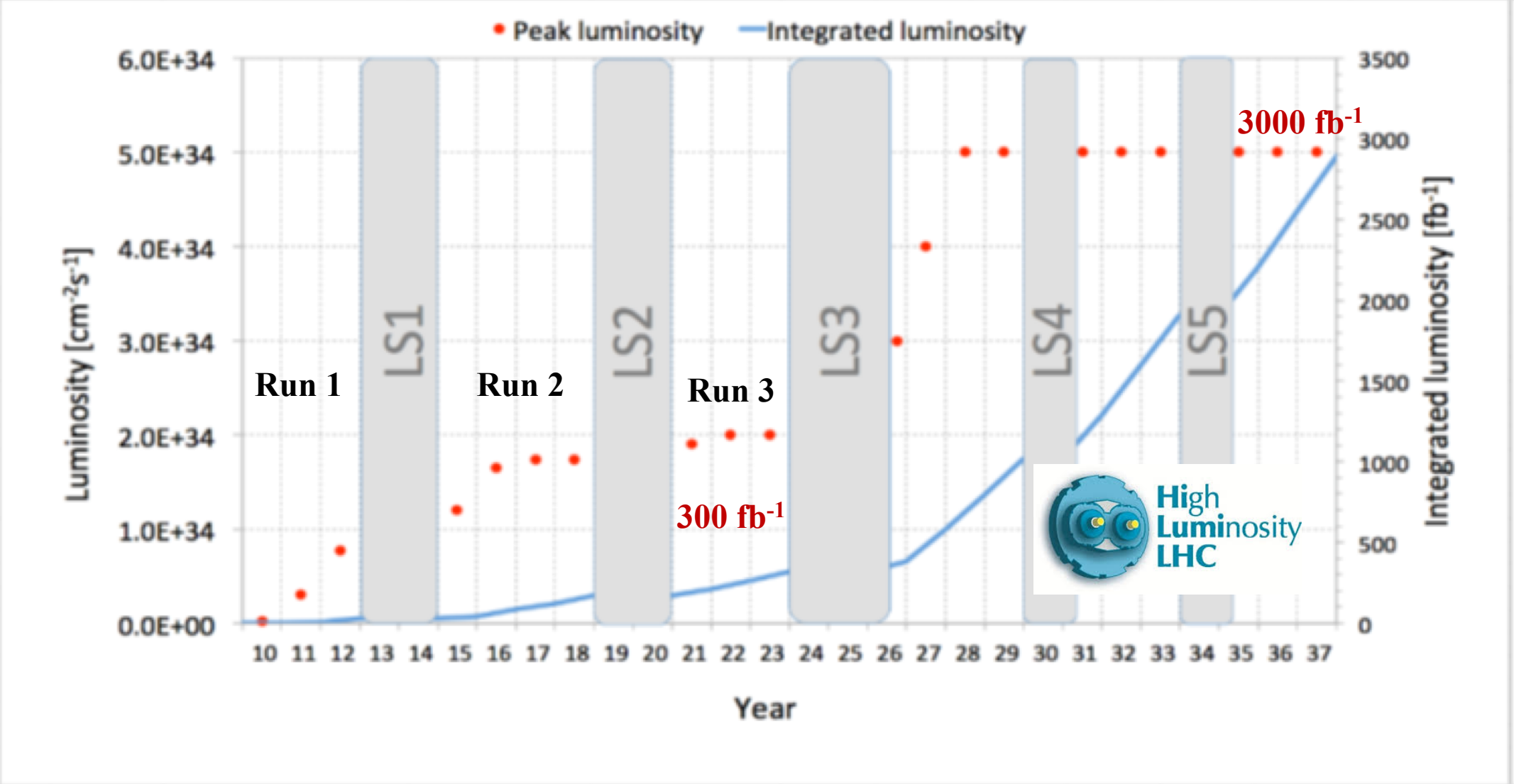


# HPC at LHC

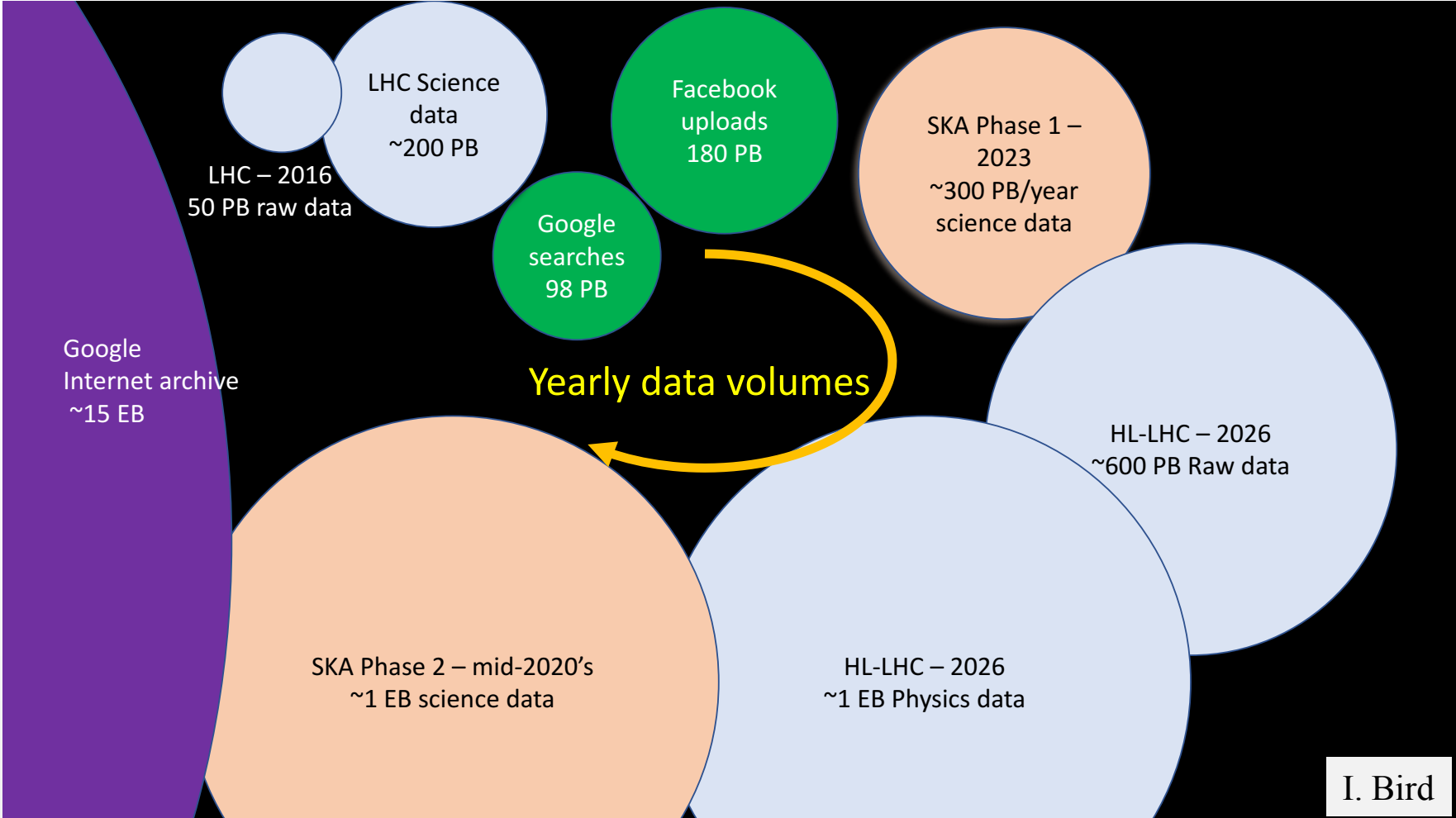
Donatella Lucchesi  
Università & INFN Padova

Slides stolen from several people presentations

# LHC Roadmap



# LHC in terms of data

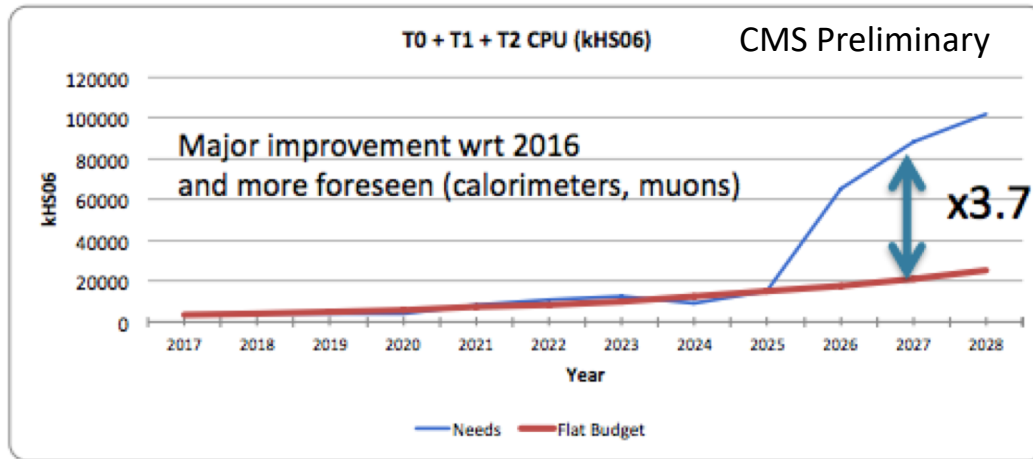


I. Bird

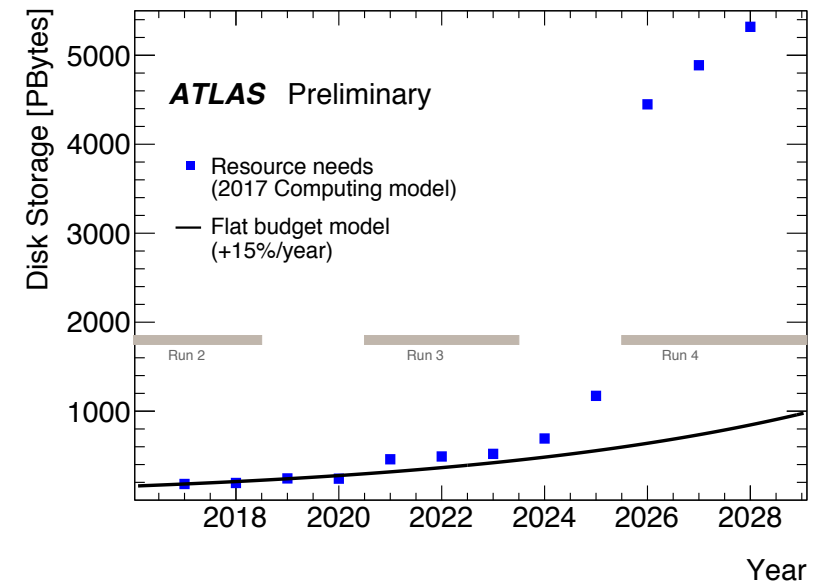
# Resources Needs

- In principle computing resources increase  $\sim$  linearly with data.
- Physicists already adopted several improvements, at the moment it is needed:

1) factor  $\sim 4$  of more CPU respect to the “20% per year growth”



2) factor  $\sim 5$  of more storage, cold storage (inactive data rarely used or accessed) concept already embedded



Question: Do we need HPC?

Answer: No.

But, if they built it, better we use it or we are always up for a challenge.

How LHC-users see HPC:

- different from the HTC world, not genuinely made for HEP software
- large clusters with accelerators and fast interconnected
- limited outbound connectivity, we are High Throughput!
- There are differences from machine to machine, we go from porting all code to “virtual grid site”

## Few examples of HPC usage by LHC

Major issues to solve:

- ❑ Job submission, interface of experiment framework with the HPC center
- ❑ Experiment software availability
- ❑ Data movement to and from the HPC center
- ❑ availability of controlled environments (light virtualization: docker, shifter, singularity...)
- ❑ full outgoing connectivity for unscheduled data access (data streaming, condition databases, ....)

- ATLAS has been using machines in the top500 list for many years
- They are coupled to PanDA workflow management system
  - ◆ All HPC's but one run G4 simulation
    - Mira has been used for years to generate Alpgen and Sherpa events
  - ◆ Some run all production workflows
  - ◆ Some run analysis

→ **Data transfers have a variety of solutions based on HPC centers needs:**

- ◆ Local Storage Element
- ◆ Remote Storage Element
- ◆ ARC-CE
- ◆ Xrootd cache
- ◆ Globus online

→ **we have several modes of execution:**

- ◆ normal grid jobs
- ◆ short backfill jobs
- ◆ event service jobs
- ◆ jumbo jobs,...

→ **Software delivery and installation is done in a variety of ways:**

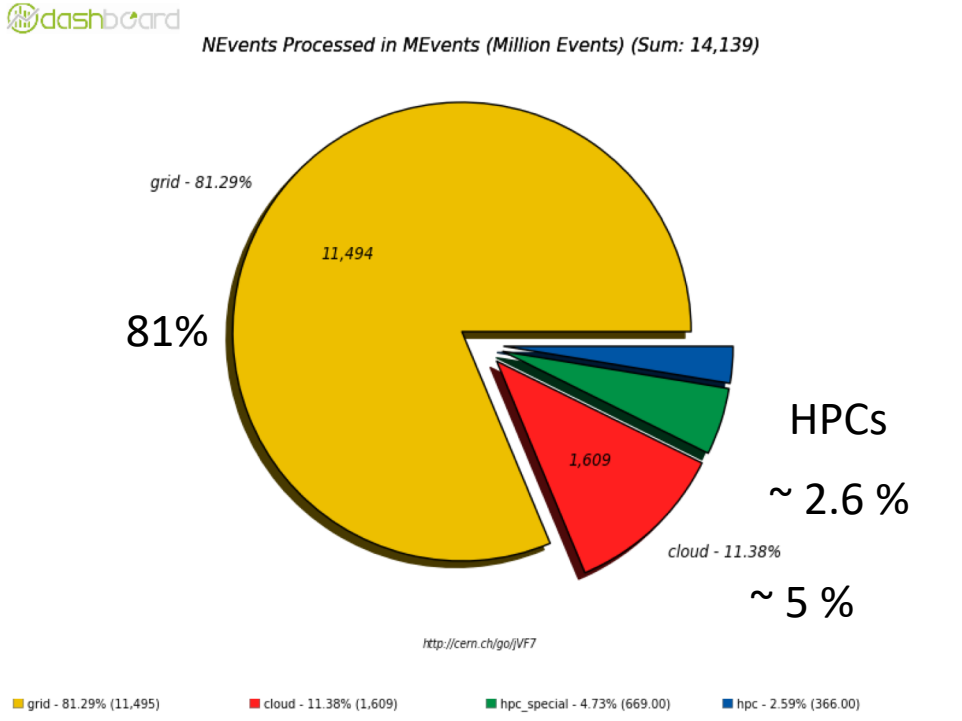
- ◆ grid-like with cvmfs
- ◆ with containers filled with software (both shifter and singularity)
- ◆ local ATLAS sw installation using rpms and tarballs create from the rpms.

Even with all of the various solutions needed - ATLAS demonstrated scalability on biggest of the sites (NERSC, Titan, Piz Daint, SuperMuc)

# HPC@ATLAS: 2017 usage

Doug Benjamin Duke University

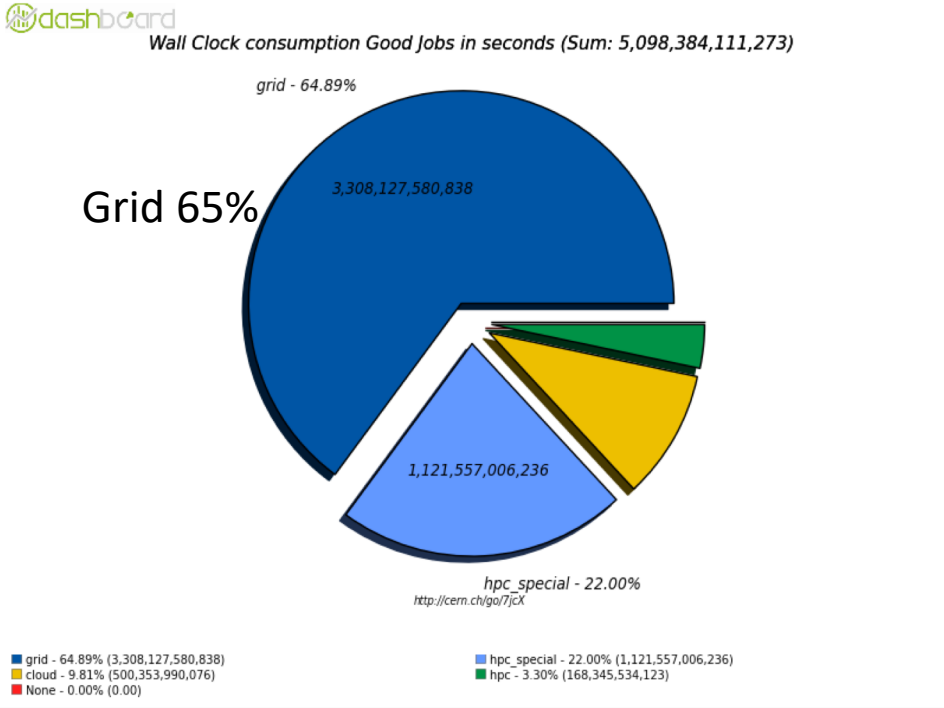
Events processed (> 14 B evts)



HPC's > 1B events

Walk clock

MC Production





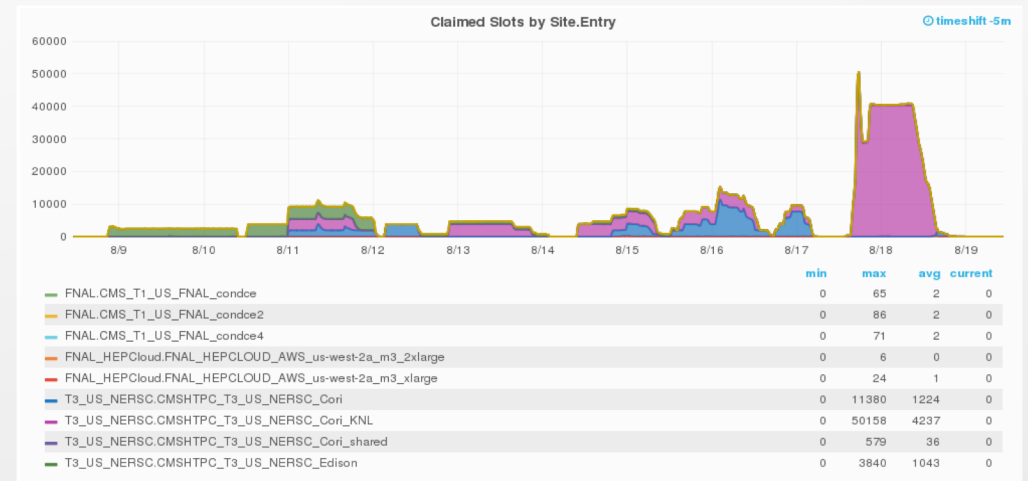
CMS is conducting R&D to commission HPC facilities for production during Run2. We are planning to fine-tune the operational model and use HPC for standard operations in Run3 and to rely on significant contribution from HPC for Run4.

## How to make HPC look like a grid site

- Pilot submission
- Runtime environment
  - Shifter(NERSC) and singularity (XSEDE)containers
- Squid caches for conditions access through Frontier
- Reading input from xrootd data federation or local HPC storage
- Job output written to external CMS site or to local HPC storage

CMS wants to be able to run the full chain of MC processing (reading pileup) and also data re-reconstruction (reading RAW) on HPC. Significant requirements on the HPC networking and IO (especially in combination with provisioning for peak)

- 50k Cori Haswell cores GenSimDigiReco, average ~8Gbps input
- 50k Cori KNL cores GenSimDigiReco, average ~2Gbps input



# HPC-Tier-2 integration from CMS

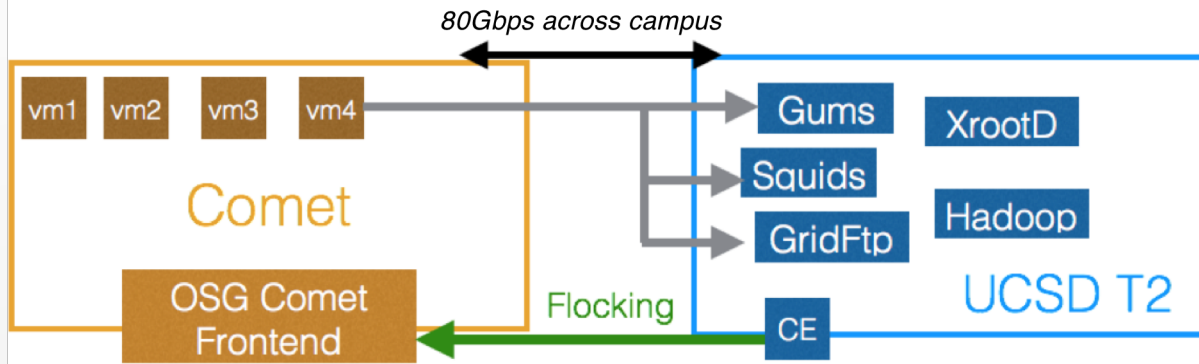
Frank Wuerthwein

## Science Gateways & Virtual Clusters

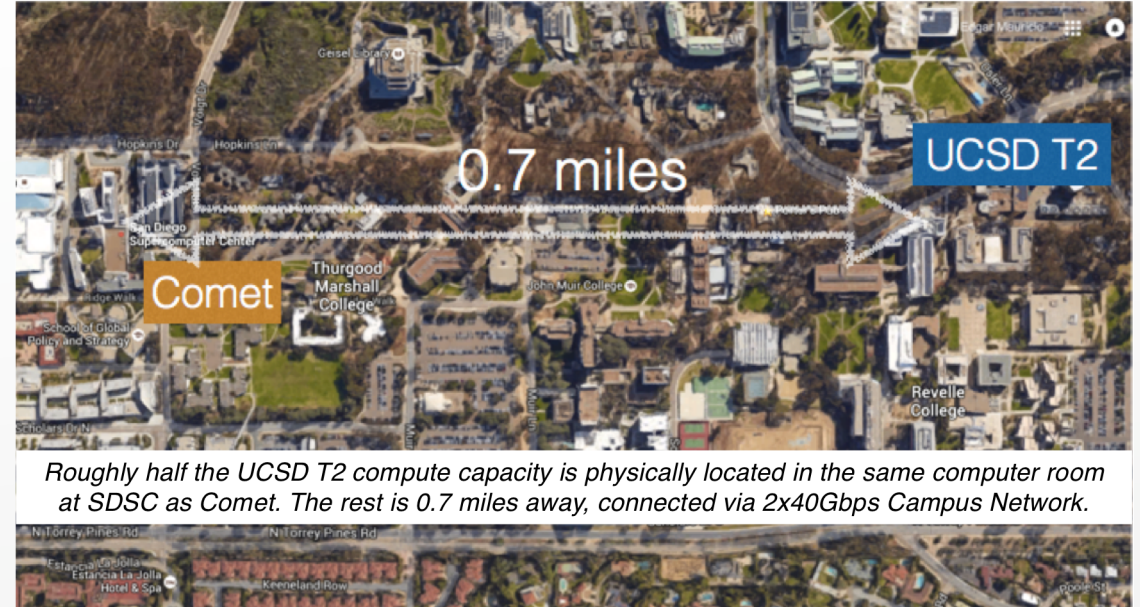
Comet and the UCSD Tier-2 Center for CMS are “co-located” on the UCSD campus.

... or how having a different culture at the institution leads to different use of HPC ...

### OSG Integration by re-using services at UCSD Tier-2 (some of these services are physically located in SDSC computer room)

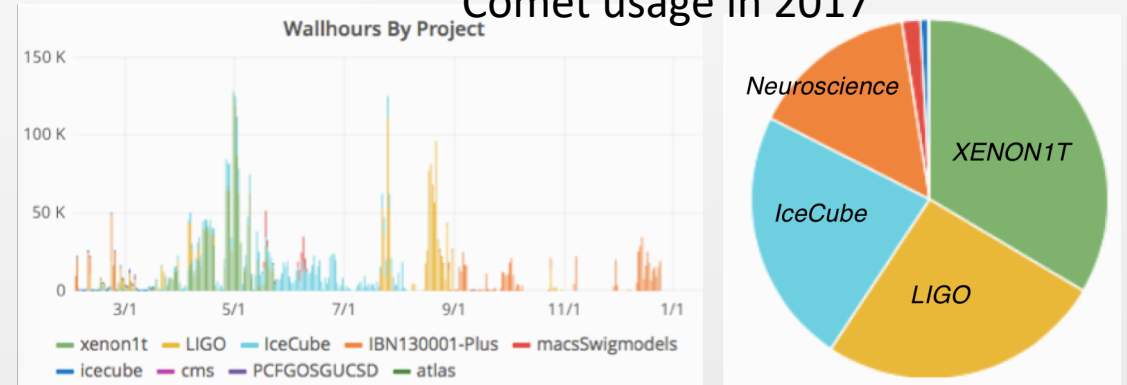


Comet virtual cluster (VC) interface provides a “cloud” like API to request resources from the Comet SLURM batch system. OSG team has full control over OS install in VC.

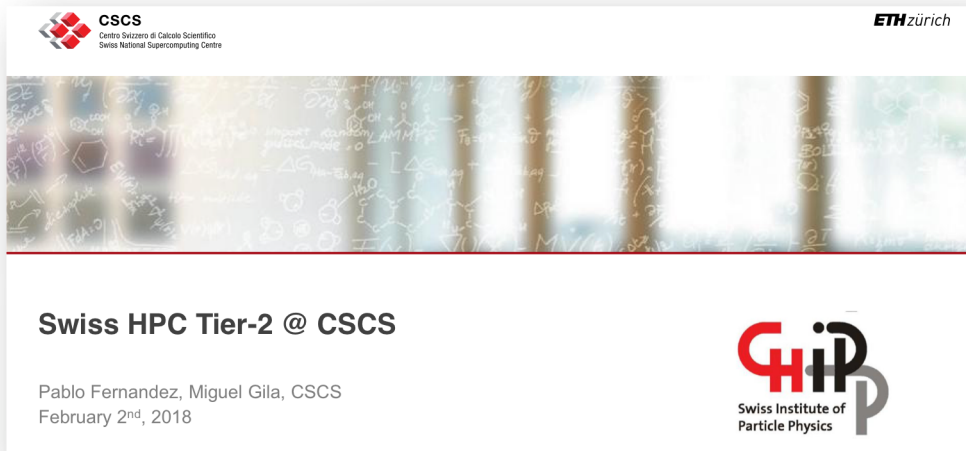


Roughly half the UCSD T2 compute capacity is physically located in the same computer room at SDSC as Comet. The rest is 0.7 miles away, connected via 2x40Gbps Campus Network.

### Comet usage in 2017



# Tier-2 HPC-based?



The slide thumbnail features the CSCS logo (Centro Svizzero di Calcolo Scientifico / Swiss National Supercomputing Centre) and the ETH zürich logo in the top left and right corners. The main content area has a background of mathematical formulas and a red horizontal line. Below the line, the text reads "Swiss HPC Tier-2 @ CSCS" and "Pablo Fernandez, Miguel Gila, CSCS February 2nd, 2018". The CHIP logo (Swiss Institute of Particle Physics) is in the bottom right corner.

## Solution Highlights

- Uses standard WLCG middleware (fully compatible with other sites)
  - Regular ARC CE in front of Piz Daint
- Software on the compute nodes is **containerized**
- Most components are shared with the previous cluster (Phoenix)
  - CVMFS
  - Scratch File System
  - Storage Element (dCache)
  - BDII, APEL, VO-BOXES
- Small/standard customizations are needed
  - Mainly on ARC Ces
- No local disks on the nodes
  - SWAP available via Cray's DataWarp
  - Other File Systems mounted from Scratch (FS-on-a-file)

## Why are we moving computing towards HPC?

- Challenge for LHC computing for the HL-LHC era (~50x needs in 8-10 years)
  - Waiting for the technology to improve will only give us ~5x
  - Switzerland wants to make active contributions
  - HPC is considered one of the main workforces for the future (see references at the end)
- CSCS is running a much bigger (~100 times) shared HPC system
  - Can grow/scale much better
  - Plenty of on-site expertise (some already in the cluster since long: Scratch FS, Infiniband...)
  - Benefit from very good economy of scales
  - Access to other high-performance technologies (GPU accelerators, in-node flash drives, high-speed networks...)
  - Bigger attention both inside and outside the datacenter (not a niche anymore!)

Study and detailed comparison of the performances obtained on GRID-Tier-2 and HPC-Tier-2 are in progress

# HPC@ALICE

ALICE tested local HPC cluster at Subatech, Nantes

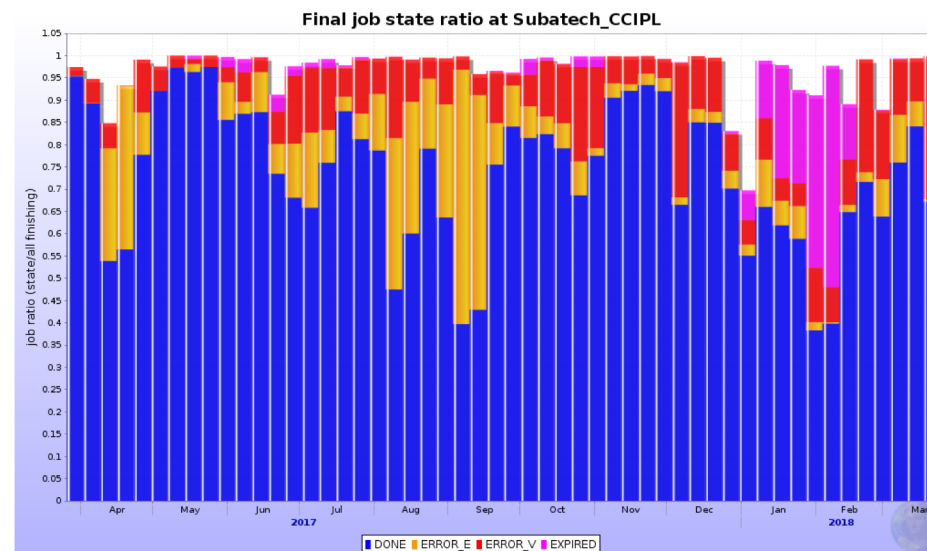
Tests are ongoing to exploit CORI @NERSC



## Our requirements

- As a partner, we had the opportunity to state our requirements and we asked for :
  - A « service box » with 2x10Gbits/s network adapt.
  - External network access via the service box (gateway)
  - CVMFS operational on the worker nodes
  - Local disk on the WN (CVMFS cache + ...)
  - Enough RAM/core to run Alice jobs
- We managed to have the storage installed at Subatech (+1PB under EOS in Nov 2016)

## Some graphs from Monalisa



# HPC@LHCb

Federico Stagni – 10th LHCb Comp Workshop  
(<https://indico.cern.ch/event/561982/>)



## ~easy integration when

- WNs have inbound/outbound connectivity
- LHCb CVMFS mounted on the WNs
- SLC6 “compatible”
- At least 2GB/core
- x86

This is the case for OSC and CSCS

When some of the requirements above are not met, we can try to go around them, but this requires dedicated work (and anyway it may not be possible, case by case)

## New activity in progress

- Santos Dumont (SDumont)
- Origin: French ATOS/BULL
- Located in Petrópolis/Rio de Janeiro – Brazil
- LHCb project for LHCb use, accepted: 2017



Renato Santana – 2/FEB/2018

# To conclude

## In Italy

- Intention to respond to a PRACE call to use CINECA.
- Preliminary test to verify the feasibility are foreseen.
- Group of people from INFN and CINECA are working together to overcome some of the issues:
  - Job submission
  - Experiment software availability.
  - Outgoing connectivity

## Considerations

As you have seen, LHC uses HPC not in a standard way.

If, in the future, LHC experiments have to use HPC centers for mission critical LHC computing, it will be necessary that it is involved in planning the infrastructure:

- accelerators that suit best our needs and software stack
- internal/external networking setups
- base system architecture (x86\_64 is our friend)
- availability of sizeable in-machine scratch disks