CERN openlab

Computing and Analysis Challenges

Pisa School on Future Colliders

Sofia Vallecorsa

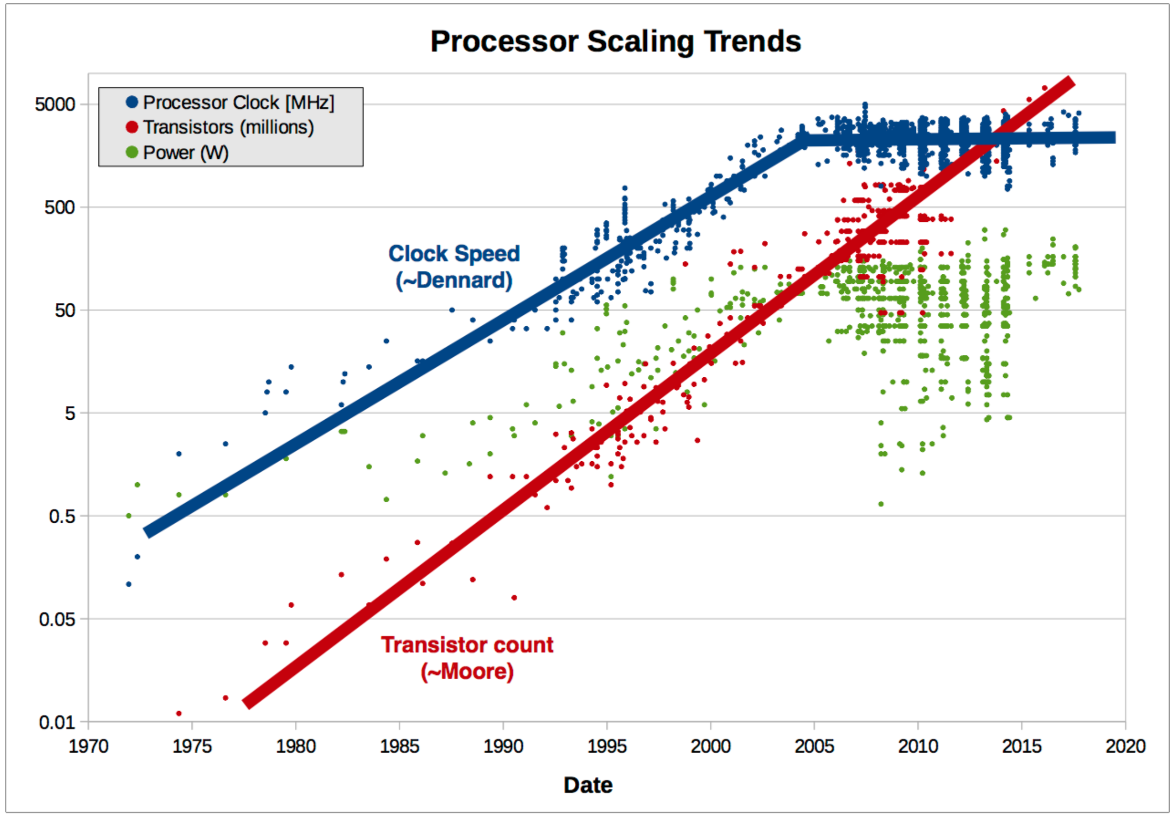
19/08/2018

Outline

- Setting the scene – The Power Wall
  - Physics requirements
- The HL-LHC challenge
- Coordinated efforts: HSF and openlab
- Current computing model: WLCG, OpenStack
- Evolving the computing model
- Finding new strategies to improve code efficiency
  - Examples from the Online, Reconstruction, Simulation
- Data Analysis
- Summary

The power wall

1965: G. Moore noted that the number of electronic components which could be crammed into an integrated circuit doubled every year



Number of transistors per chip is going up

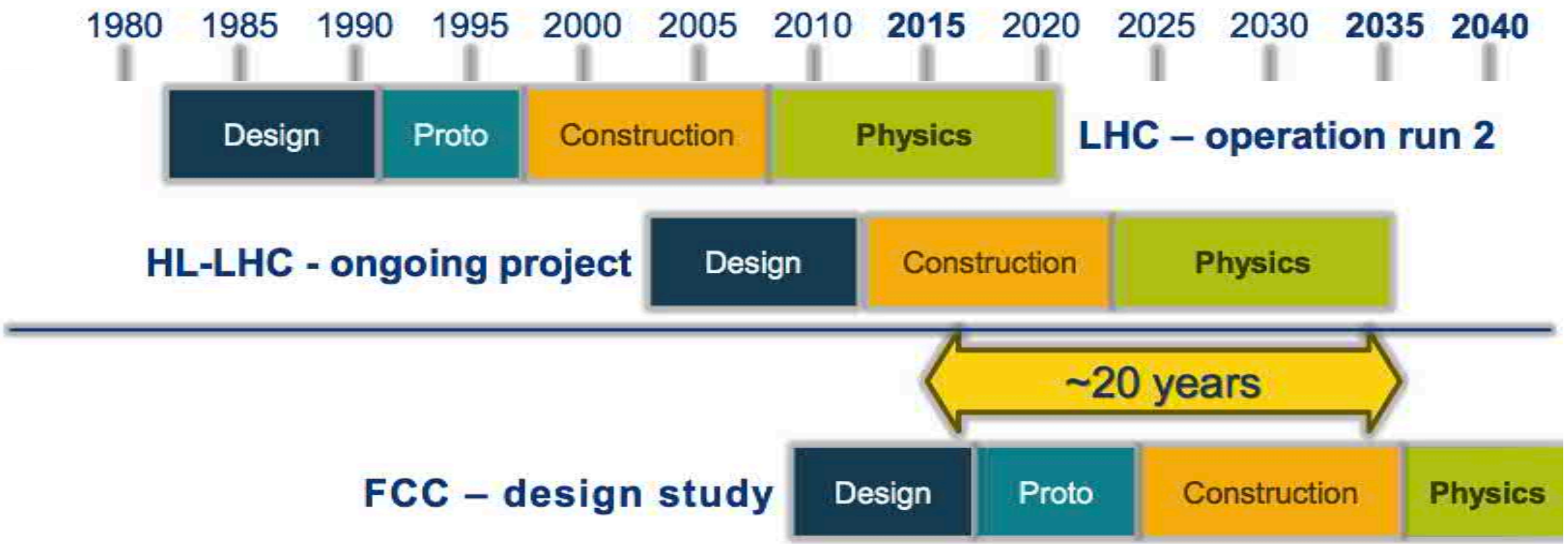
- Clock speed has flattened at ~3GHz

Amount of dissipated energy is the limiting factor (power wall)

The HEP plan

Relied mostly on clock speed increase to simply see code running faster on more performant hardware..

Massive data processing and simulation



Modified from CERN Courier May 2017

There are a number of different options for new machines

- Lepton colliders (ILC, CLIC, FCC-ee) have overall less serious computing challenges
- Hadron colliders (HE-LHC, FCC-hh) bring a massive data rate and complexity problem

HEP computing model needs to evolve

Physics requirements: recap 1

Huge particle/data rates ~ 1-2 PB/s

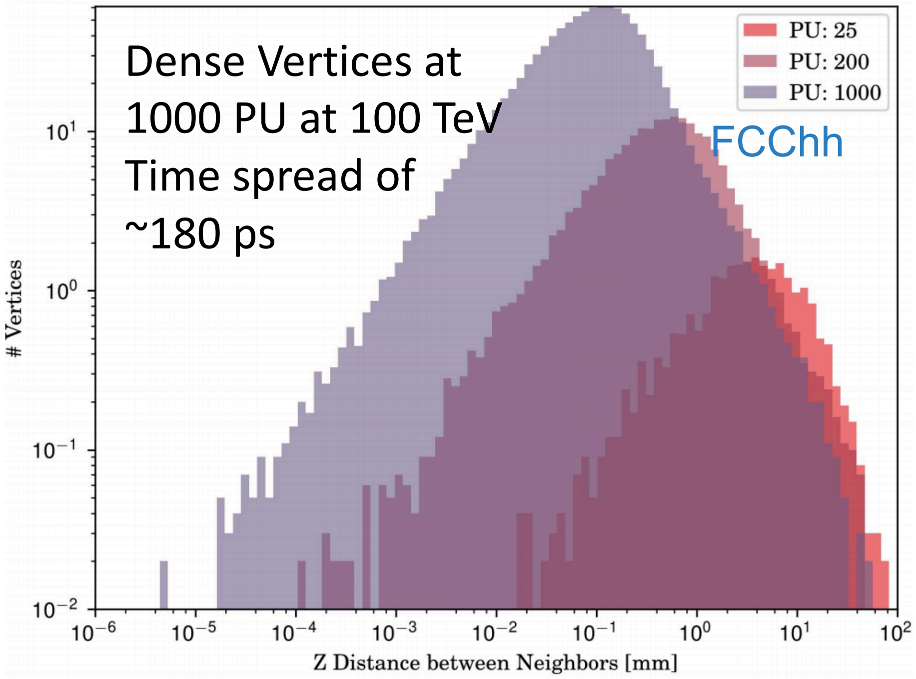
Large pile-up

- Dense vertices

Depending on configuration, events from different bunch crossing will likely overlap

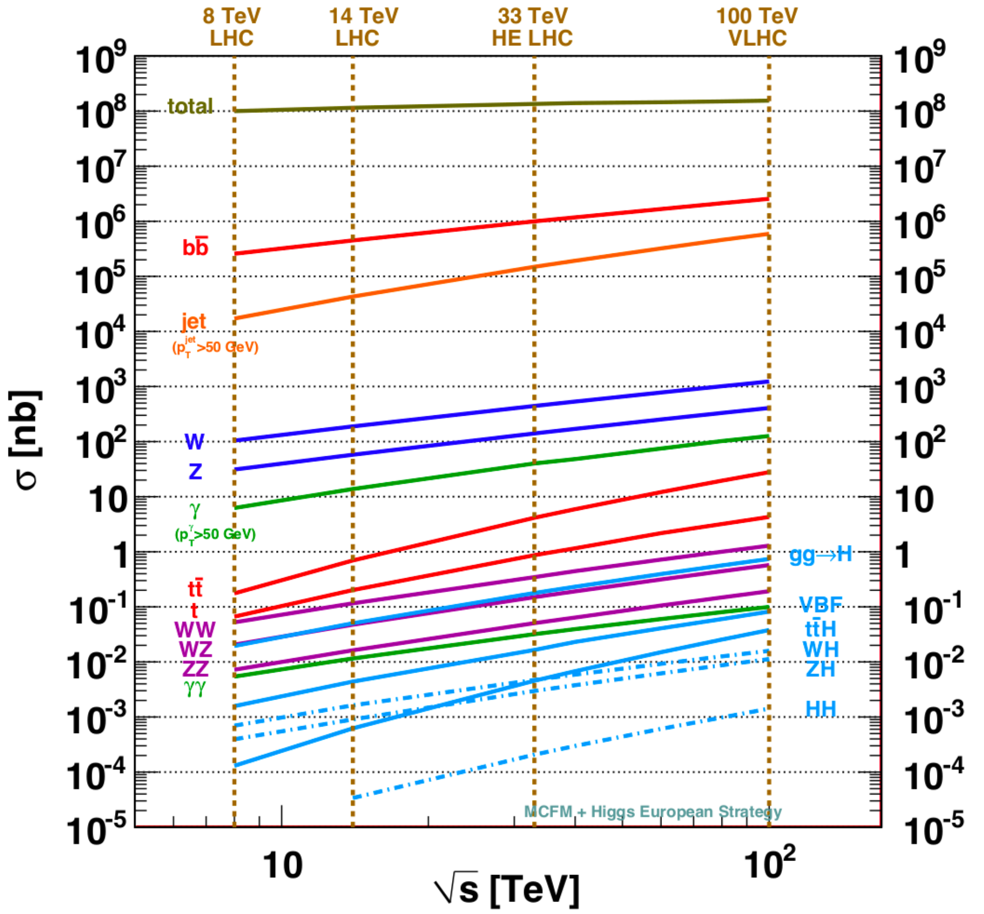
- 4D reconstruction to disentangle

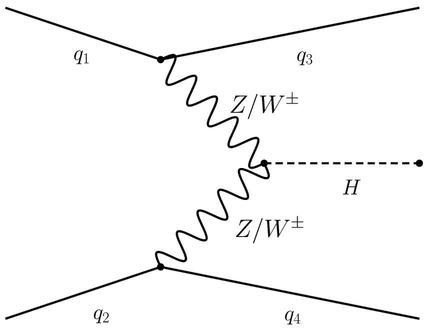
Parameter	unit	LHC	HL-LHC	HE-LHC	FCC-hh
E_{cm}	TeV	14		27	100
Peak luminosity $\times 10^{34}$	$\mathrm{cm}^{-2}\mathrm{s}^{-1}$	1	5	25	30
bunch spacing	ns	25			
σ_{inel}	mbarn	85		91	108
σ_{tot}	mbarn	111		126	153
$\langle p_{\mathrm{T}} \rangle$	GeV/c	0.6		0.7	0.76
$dN_{ch}/d\eta\|_{\eta=0}$		7		8	9.6
Number of bunches		2808			10600
BC rate	MHz	31.6			32.5
Peak pp collision rate	GHz	0.85	4.25	27.3	32.4
Peak avg PU events/BC		27	135	864	997
Goal integrated luminosity	ab^{-1}	0.3	3	10	20

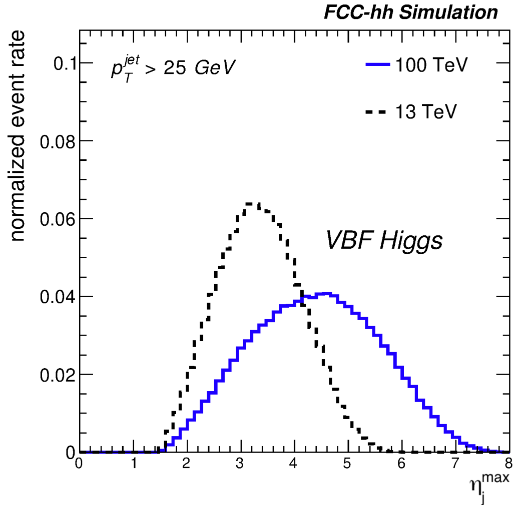


A. Zaborowska, FCC week, 2018

Physics requirements recap 2

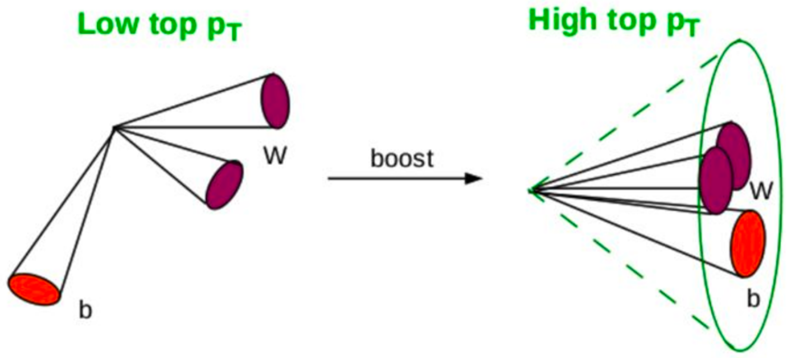






Forward physics: large acceptance – large number of readout channels

Boosted objects: high granularity tracking and calorimetry



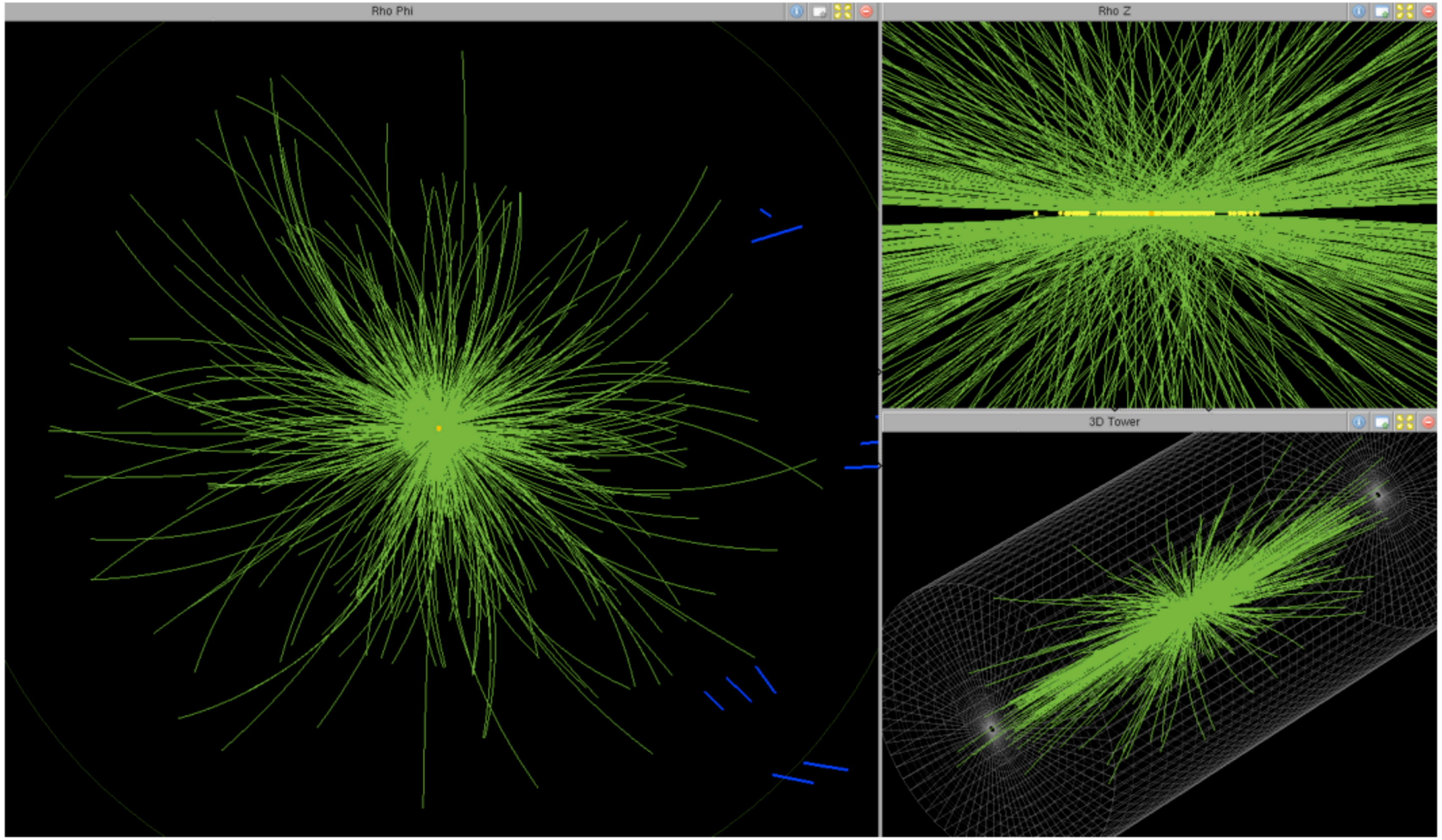
A. Zaborowska, FCC week, 2018

openlab

Ex: Tracking in CMS

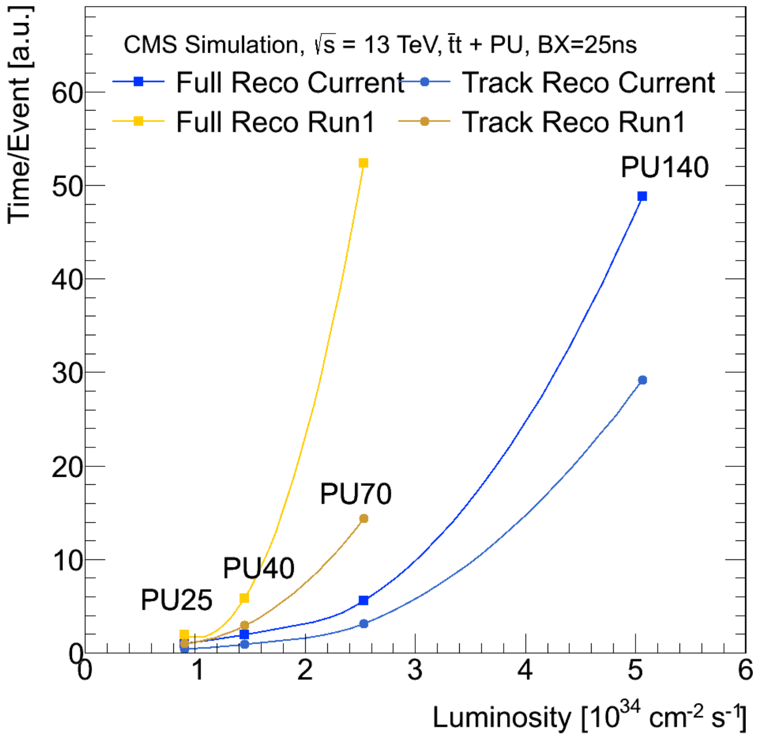
Reconstruction of CMS Simulated Event

$t\bar{t}$ event at <PU>=140 (94 vertices, 3494 tracks)



2023 CMS Tracker

- Higher granularity (x6), extended coverage, hardware trigger capability

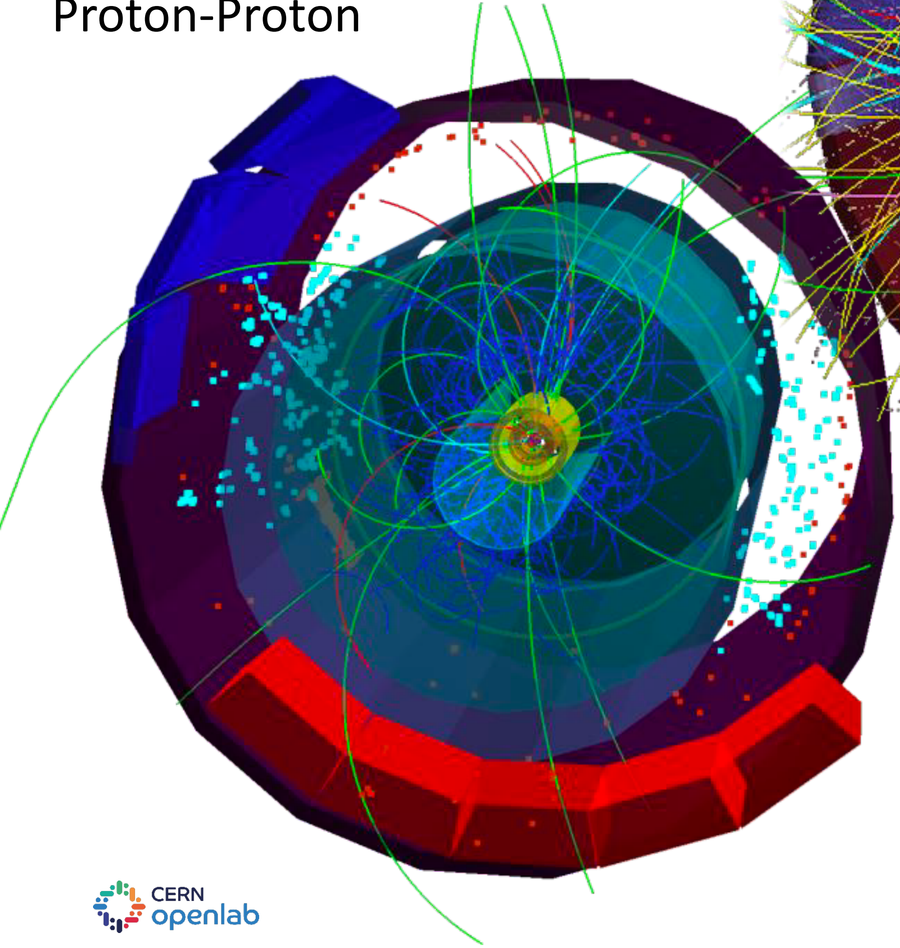


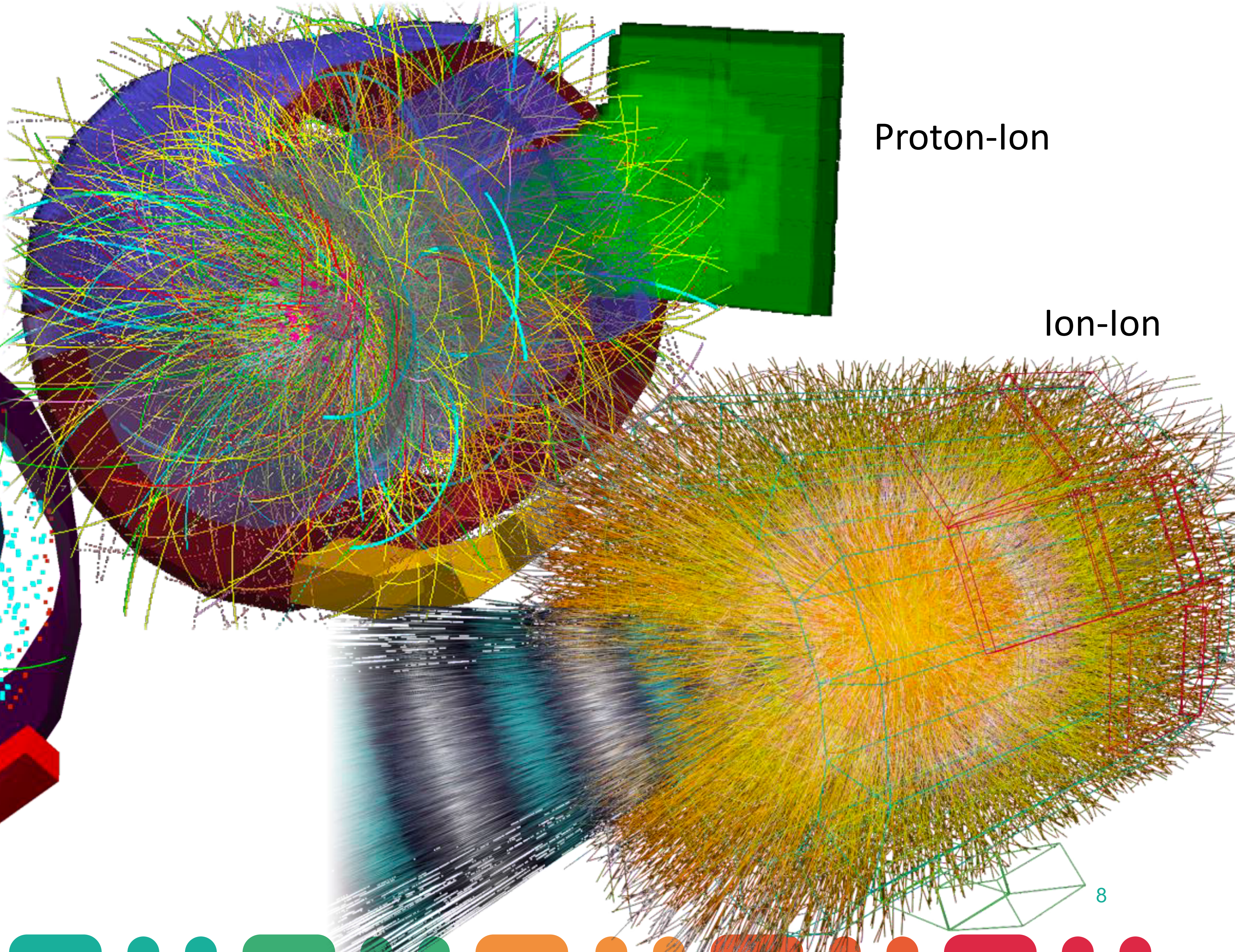
Ex: ALICE ...

Proton-Ion

Proton-Proton

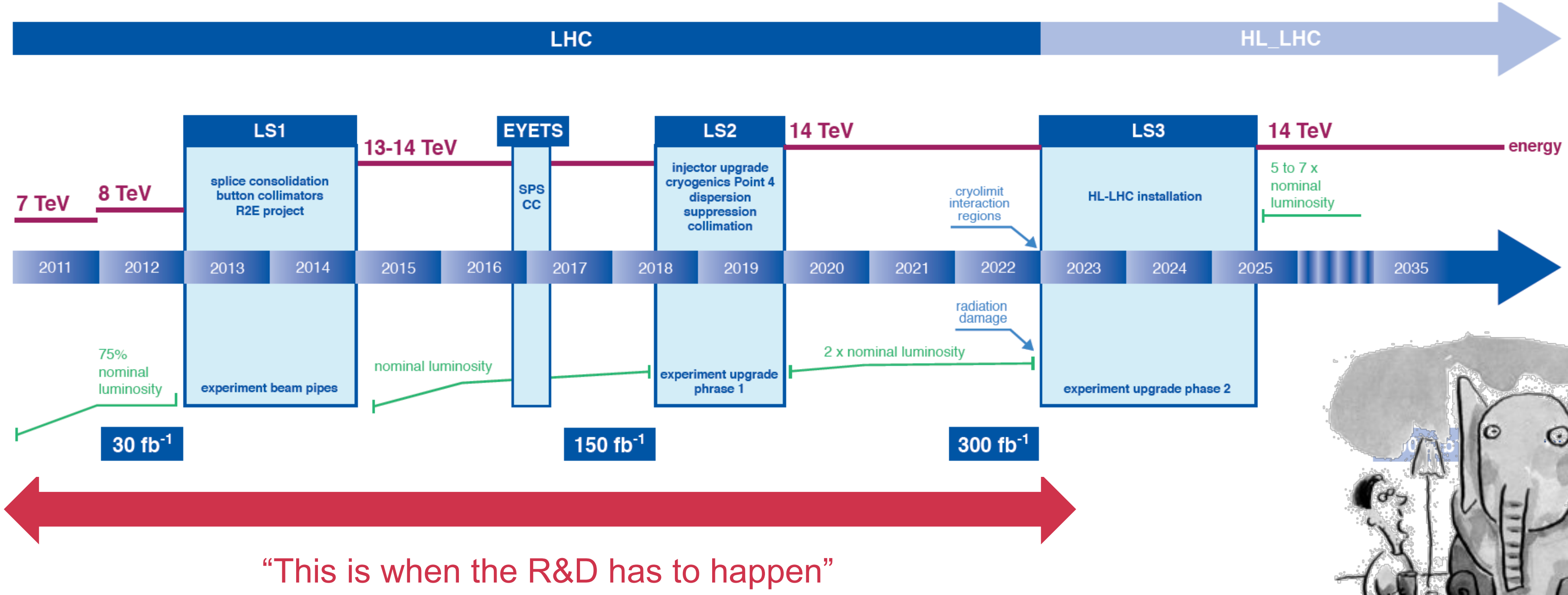
Ion-Ion

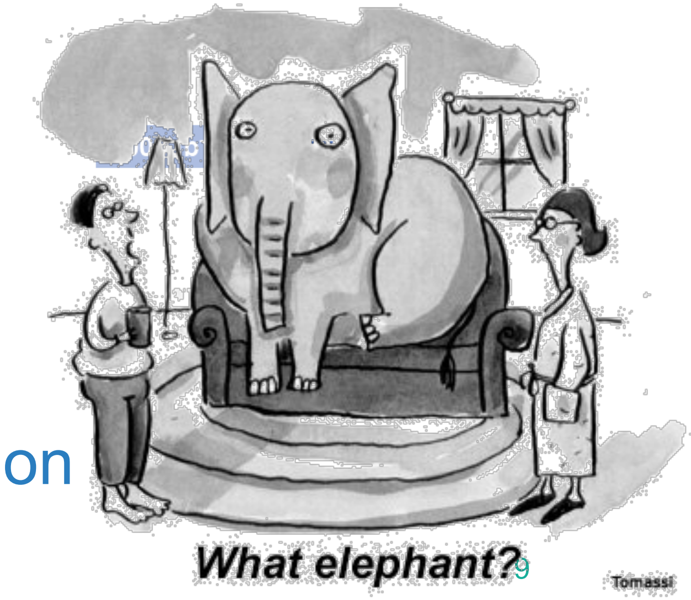




HL-LHC

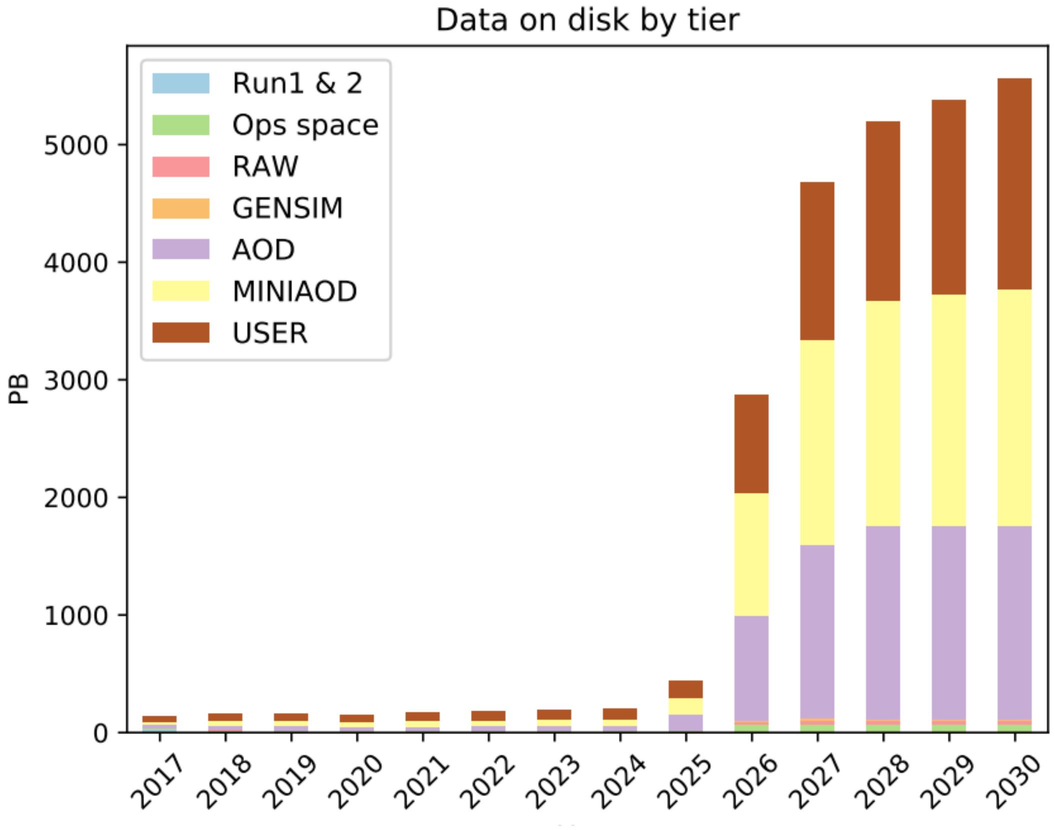
"The elephant in the room"

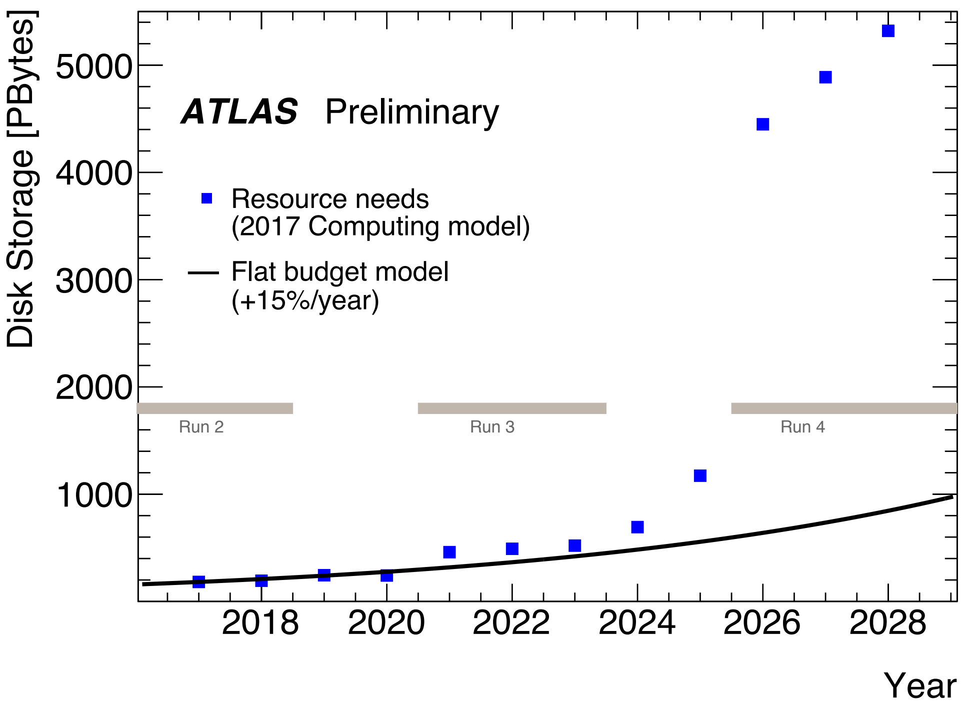




Detectors development must be matched by software innovation

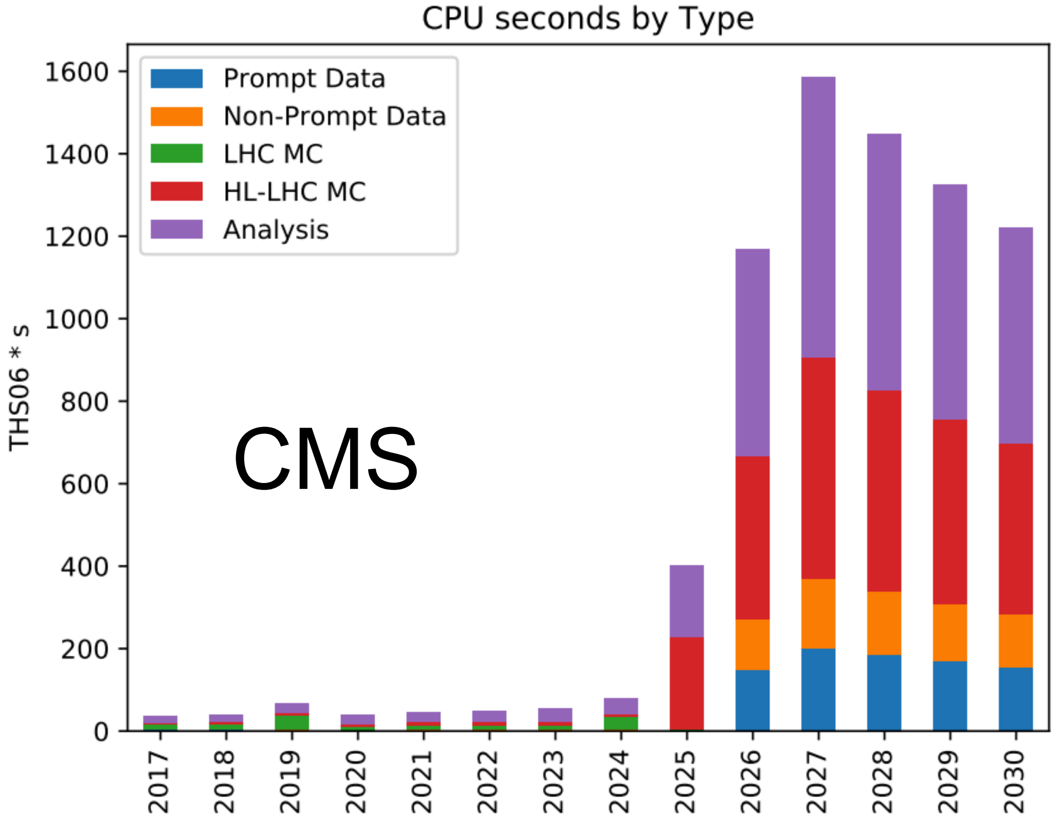
HL-LHC: data volume





https://arxiv.org/pdf/1712.06982.pdf

HL-LHC: computing power



- Raw data volume increases exponentially
  - Processing and analysis load
- Technology at ~20%/year will bring x6-10 in ~10 years
  - Estimates of resource needs x10 above what is realistic to expect

High-Luminosity LHC is far from being a solved problem for software and computing

Beyond HL-LHC, Whatever the future, we pass through the HL-LHC on the way

Coordinated efforts

Hep Software foundation

HSF established in 2015 to facilitate common efforts and improve coordination

→ Community White Paper

Contents

manage, process, and analyse the shear amounts of data to be recorded. In planning for the HL-LHC in particular, it is critical that all of the collaborating stakeholders agree on the software goals and priorities, and that the efforts complement each other. In this spirit, this white paper describes the R&D activities required to prepare for this software upgrade.

for the coming decades. This programme requires large investments in detector hardware, either to build new facilities and experiments, or to upgrade existing ones. Similarly, it requires commensurate investment in the R&D of software to acquire, manage, process, and analyse the shear amounts of data to be recorded. In planning for the HL-LHC in particular, it is critical that all of the collaborating stakeholders agree on the software goals and priorities, and that the efforts complement each other. In this spirit, this white paper describes the R&D activities required to prepare for this software upgrade.

CERN openlab

https://hepsoftwarefoundation.org/organization/cwp.html

CERN openlab

A science – industry partnership to drive R&D and innovation

Evaluate state-of-the-art technologies in a challenging environment and improve them

Test in a research environment today what will be used in many business sectors tomorrow

Training

Dissemination and outreach

CERN openlab

PARTNERS	CONTRIBUTORS	ASSOCIATES	RESEARCH
HUAWEI	rackspace	COMTRADE	INFN, King's College London
intel	IBM	Yandex	Fermilab, Simula
ORACLE	E4 Computer Engineering	open systems	Newcastle University, EMBL-EBI
SIEMENS	Extreme		GSI, IVI Innovation Value Institute

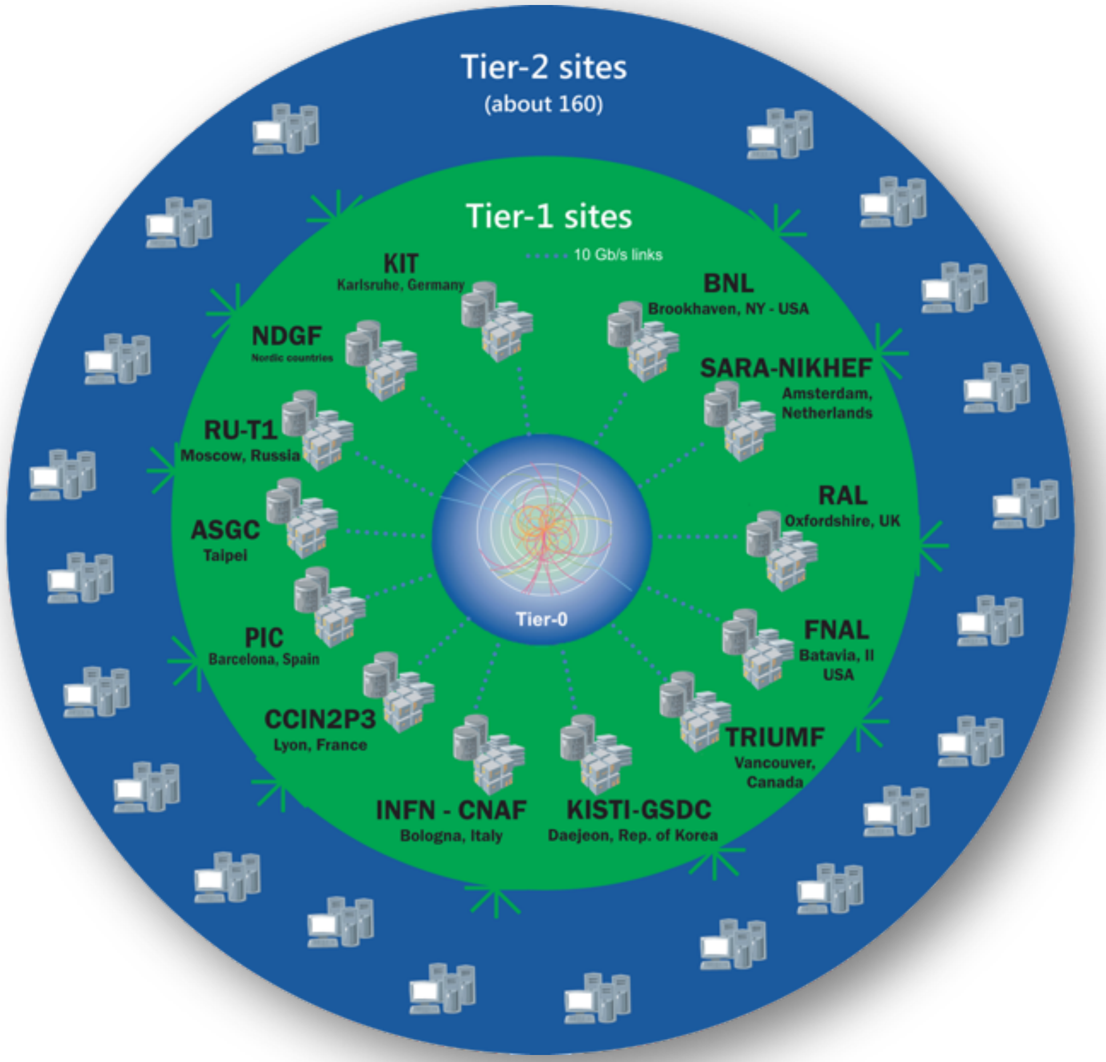
Current model

The Worldwide LHC Computing Grid

Tier-0
(CERN and Hungary):
data recording,
reconstruction and
distribution

Tier-1: permanent
storage, re-
processing,
analysis

Tier-2: Simulation,
end-user analysis



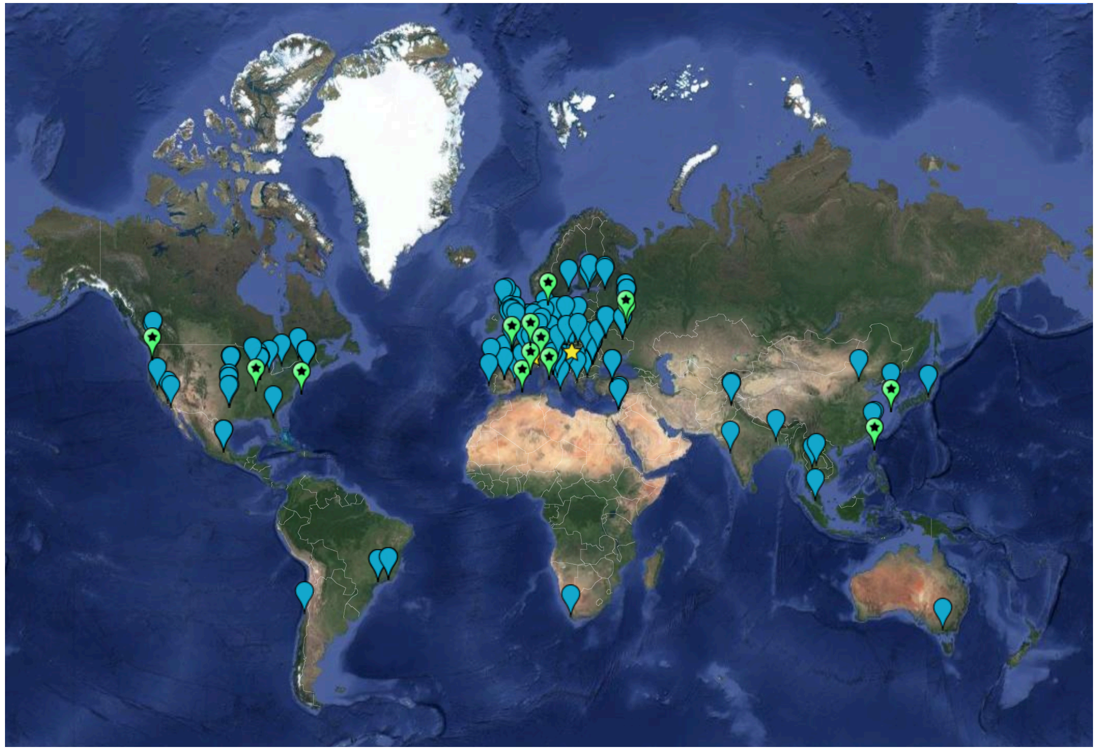
WLCG in numbers

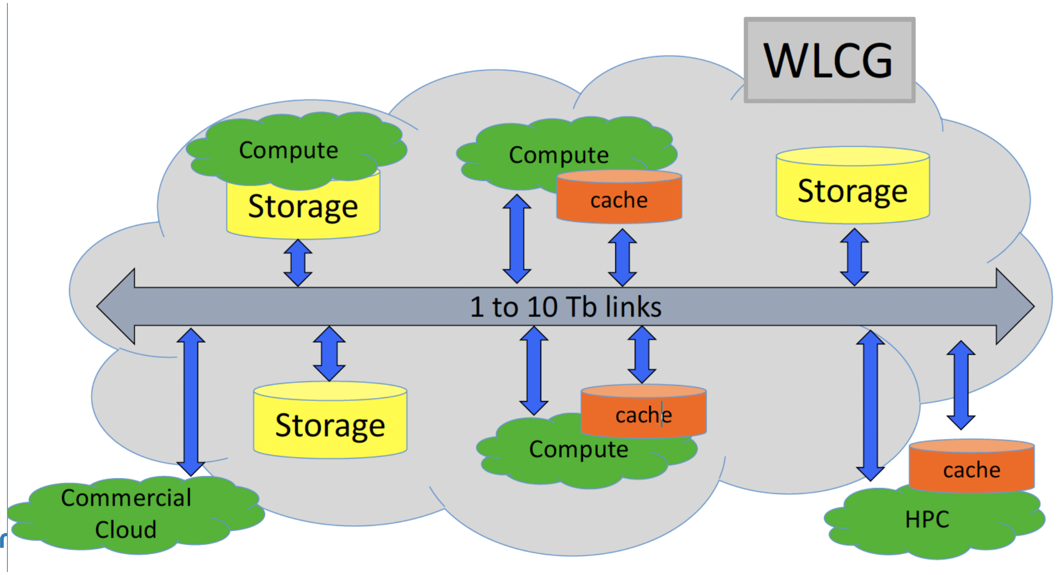
~170 sites, 42 countries

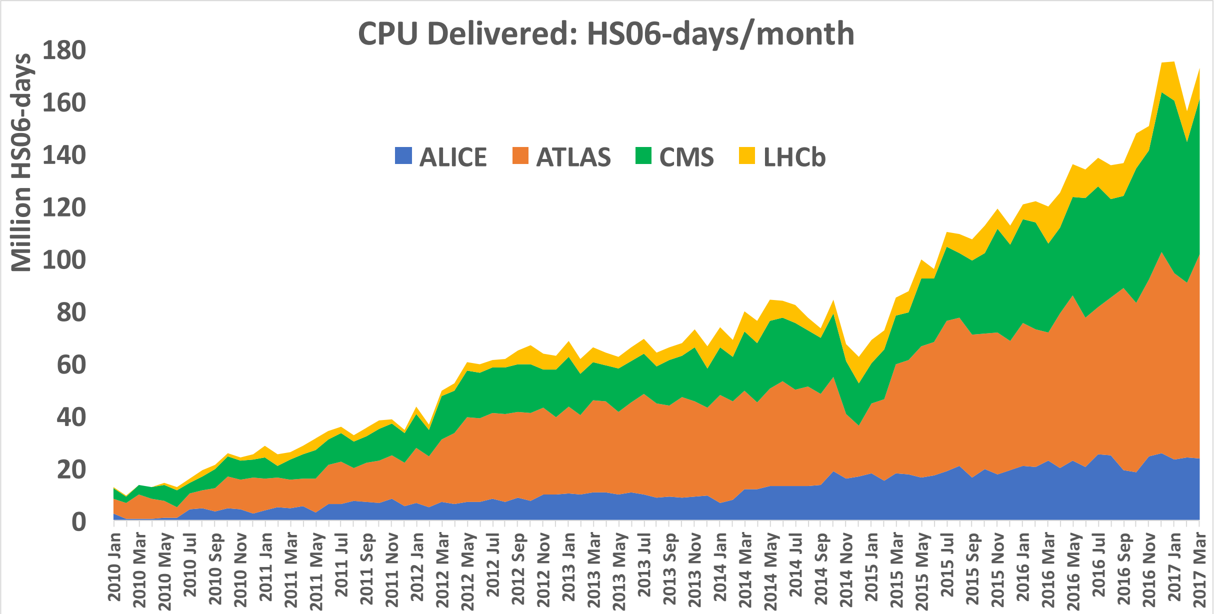
~800k CPU cores, 600 PB of storage

2 million jobs/day

10-100 Gb links







Distributed model

Performant & reliable networks

- 10 Gb/s → 100 Gb/s at large centers
- >100 Gb/s transatlantic links in place

Originally strict hierarchical Tier structure

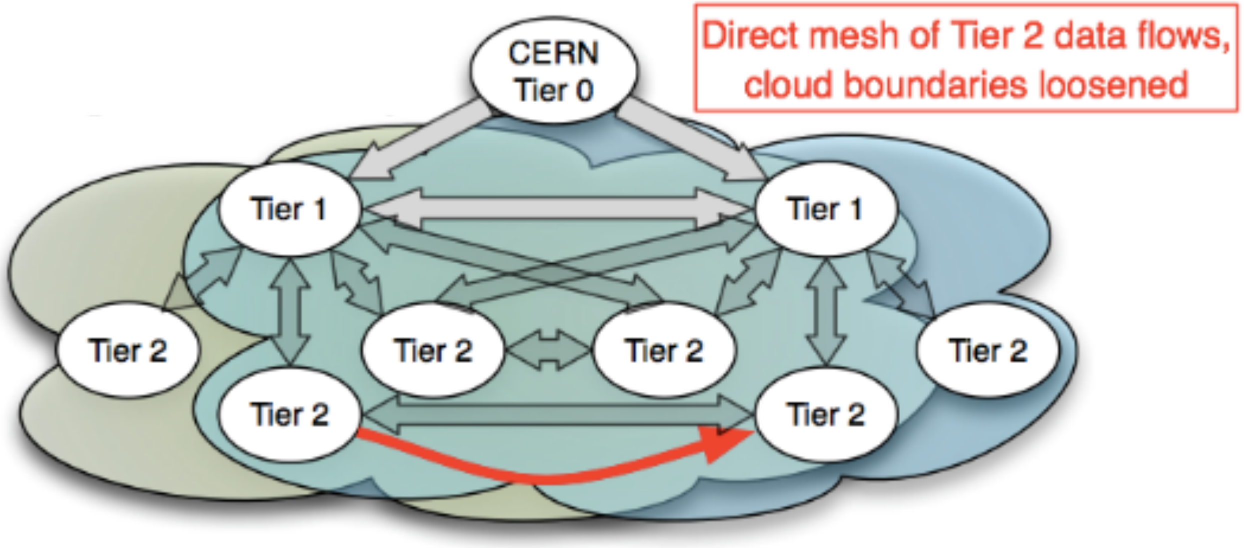
- Role based

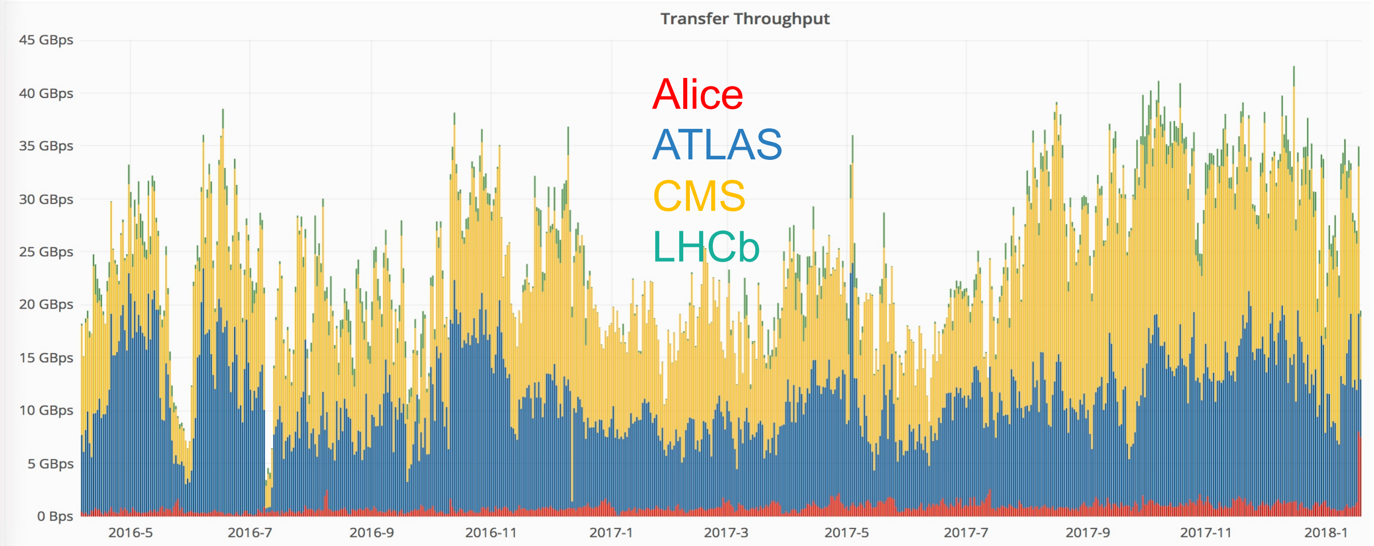
Now focus on use of resources & capabilities

- Data access peer-peer
- Optimise overall distributed resources
- More functional and service quality based

Currently moves more than **3PB/day**

ATLAS since 2011:

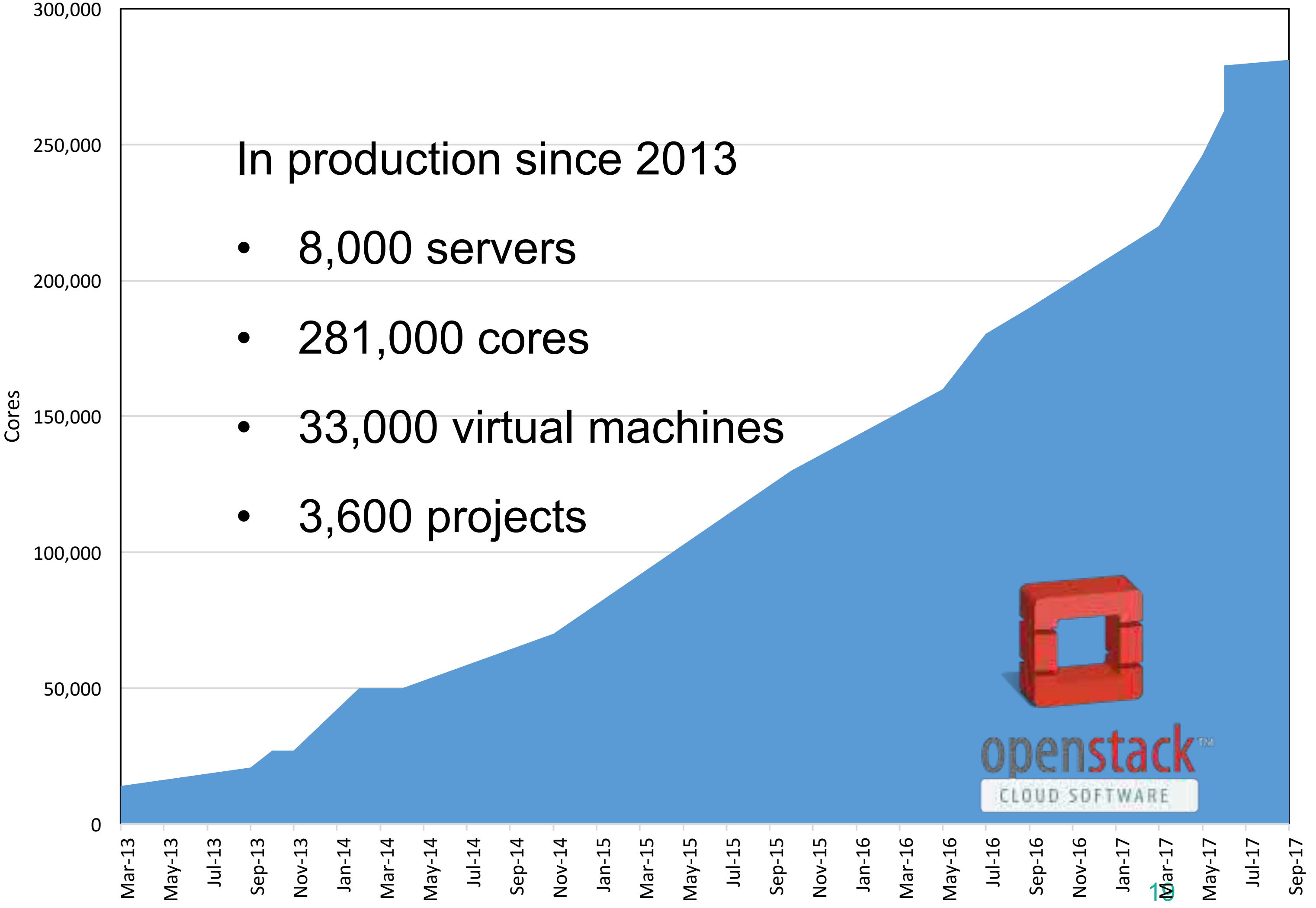




CERN openlab

CERN OpenStack Private Cloud

- One of the early adopters and largest contributors
  - 90% of the resources are provided through a private cloud
- Flexible and dynamic deployment
  - Moving to containers (investigations within CERN openlab)

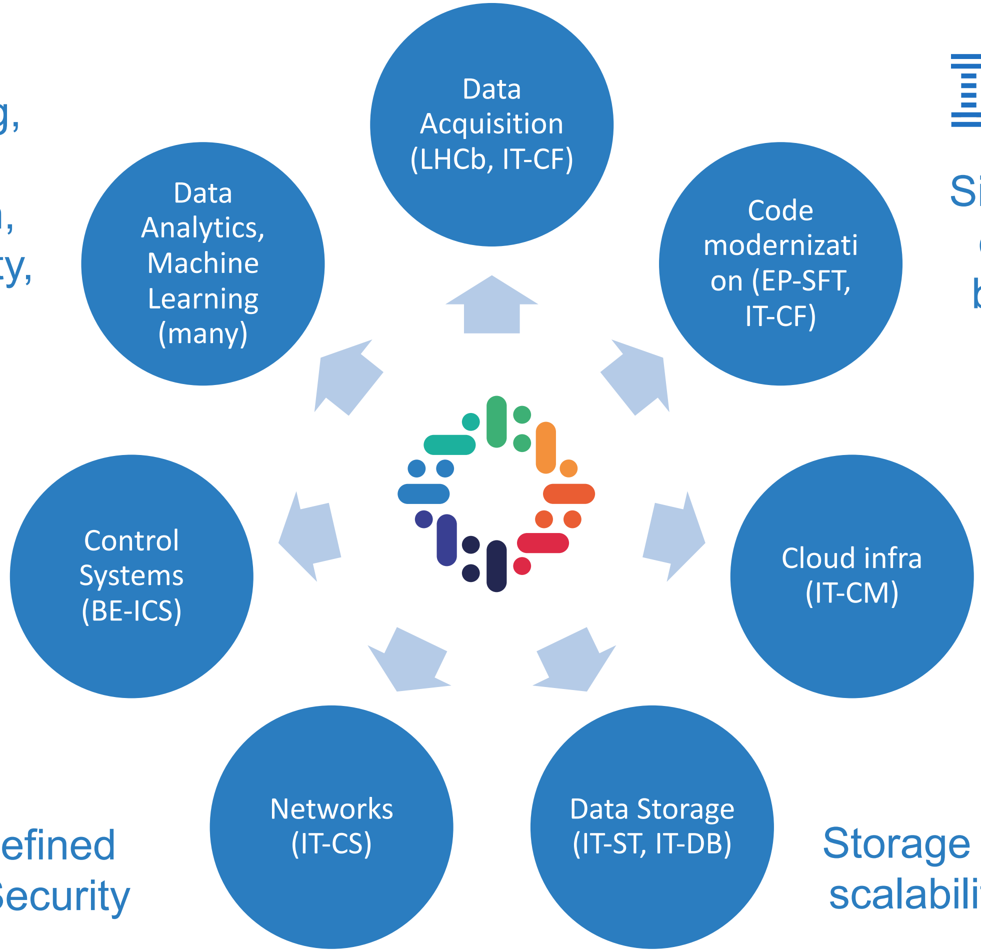


JOINT R&D PROJECTS

High-bandwidth fabrics, accelerated platforms

Data quality monitoring, anomaly detection, physics data reduction, benchmarking/scalability, systems biology and large-scale multi-disciplinary platforms

Simulation, HPC on the Cloud, benchmarking



Predictive/proactive maintenance and operations

Cloud federations, containers, scalability

Software Defined Networks, Security

Storage architectures, scalability, monitoring

CERN openlab

Evolution of computing models

Infrastructure optimisation

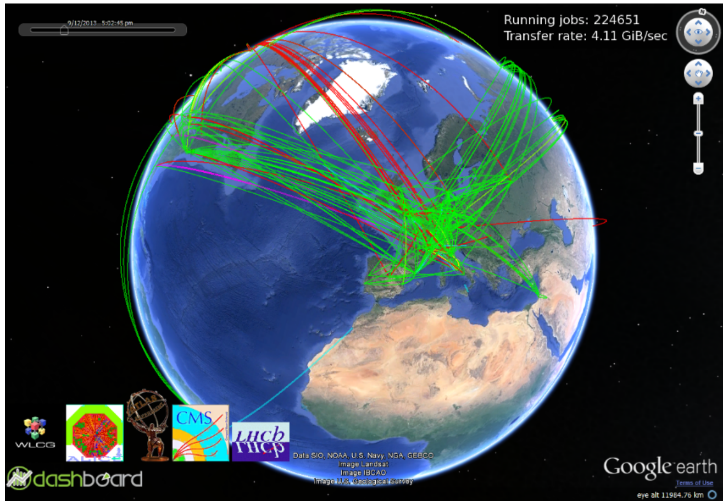
Data storage

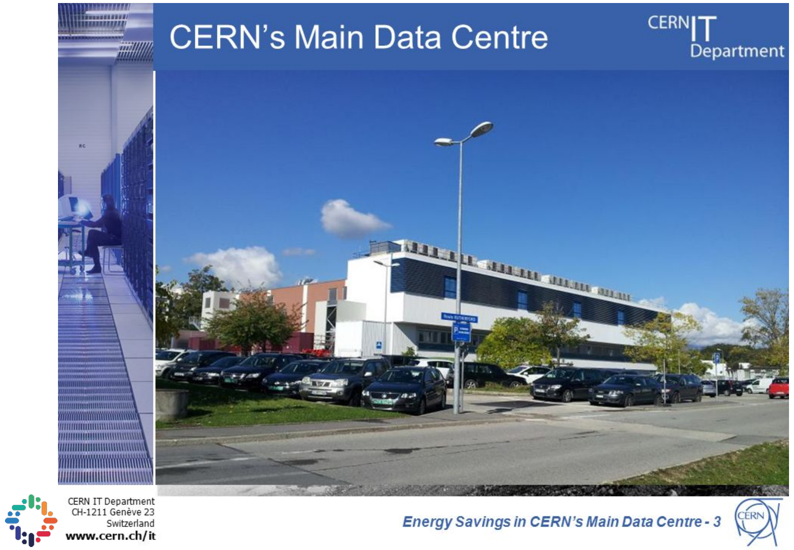
Commercial / Public Cloud

HPC

Diversifying hardware

ML for Infrastructure Monitoring





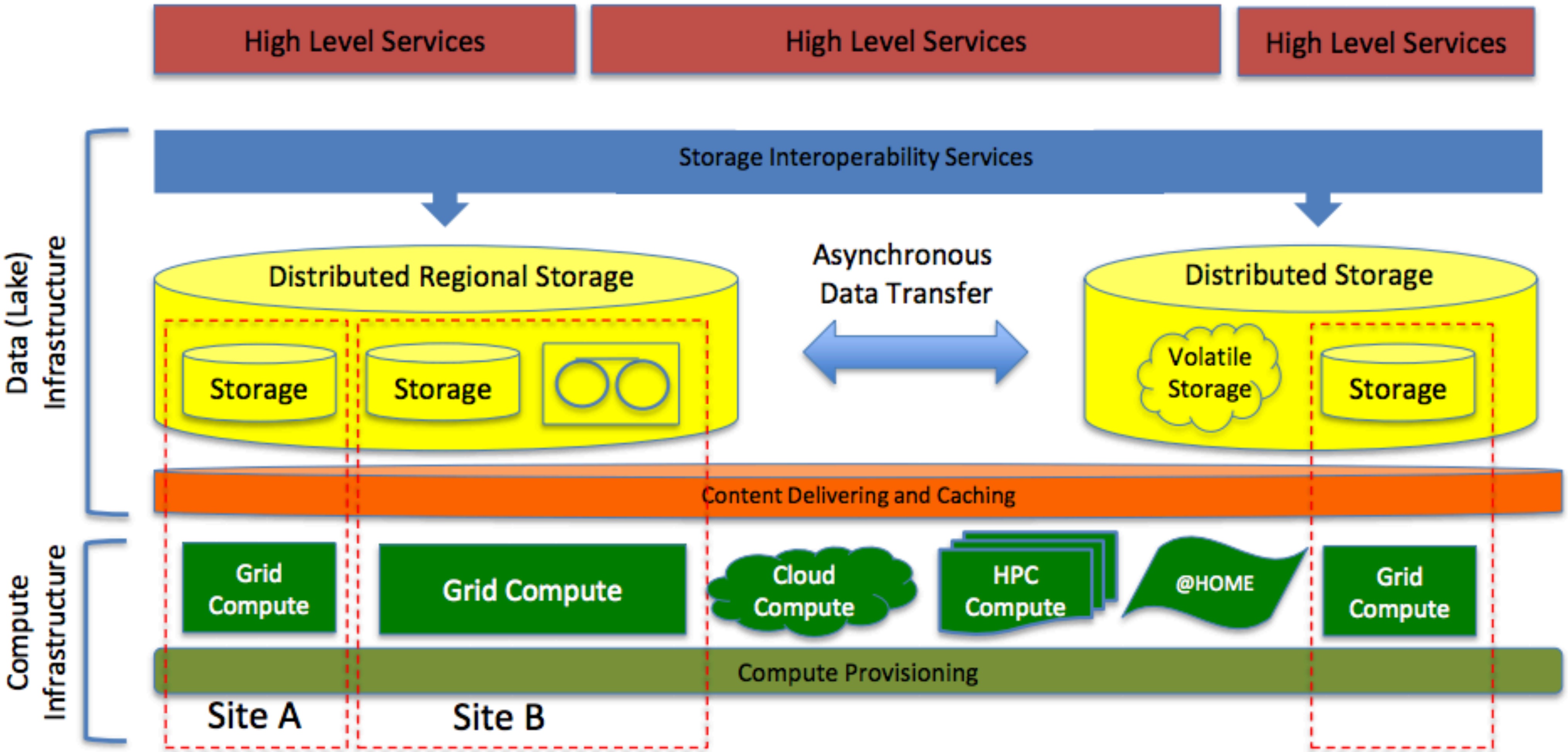
ML is being evaluated to optimize several infrastructure tasks

Data placement: use smart data analysis to predict where to move data across the WLCG infrastructure

Network security: analyse traffic patterns to detect anomalies and intrusions

Data Centre optimization: optimize job allocation, resource utilization, energy consumption, etc.

Evolution of data storage: Data-lakes

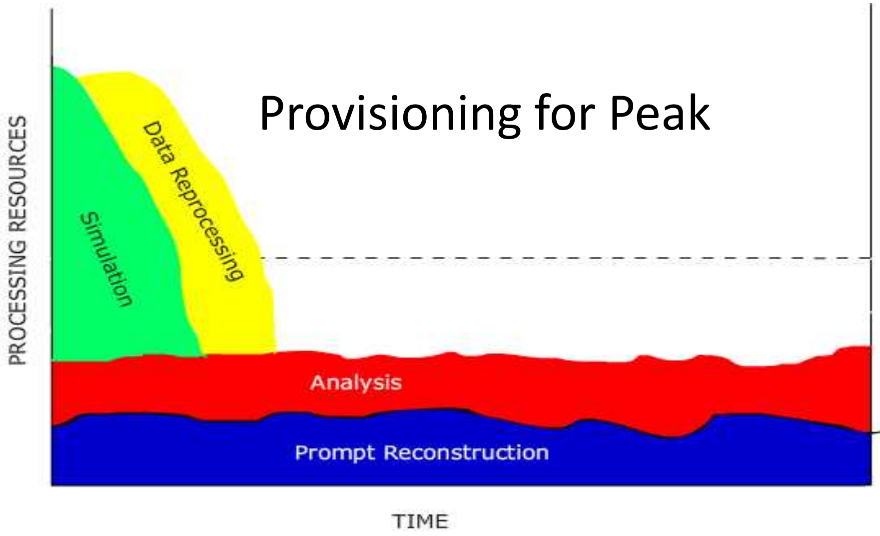


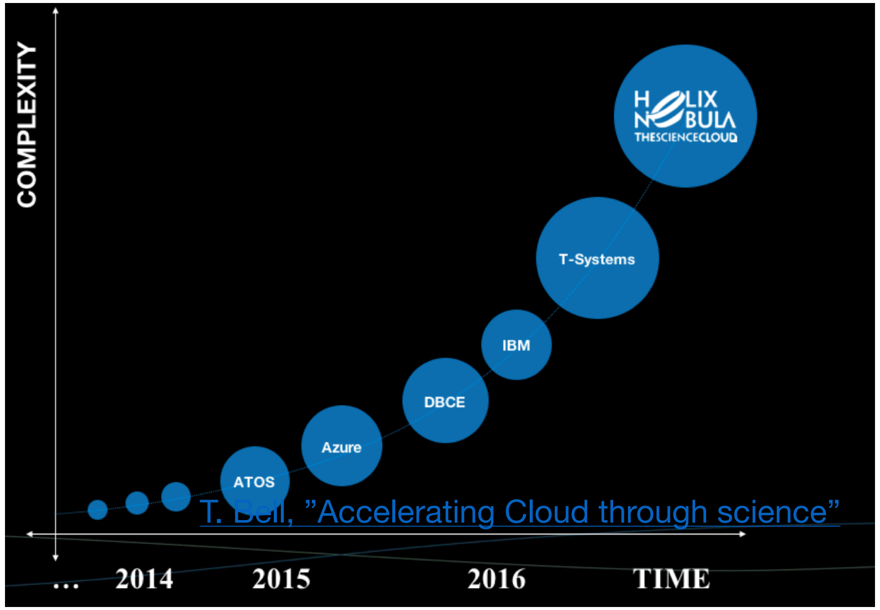
CERN openlab

Public Cloud Investigations

- Investigate scale-out with public providers without impact on users
- **Helix Nebula** –a Pre-Commercial Procurement tender for a European hybrid cloud
  - support deployment of high-performance computing and big-data capabilities for scientific research
  - Available to multiple user groups in HEP, astronomy, life sciences, …

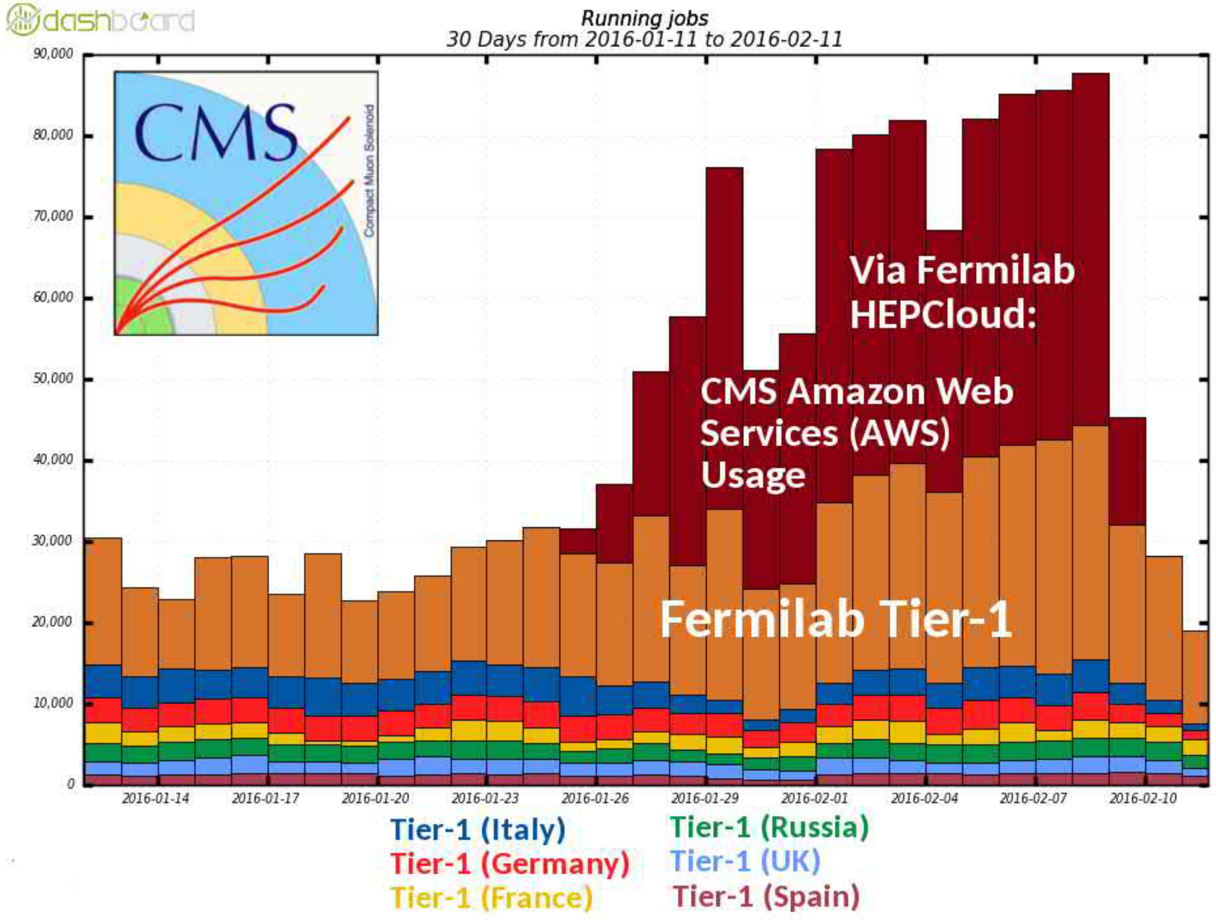
www.HNSciCloud.eu

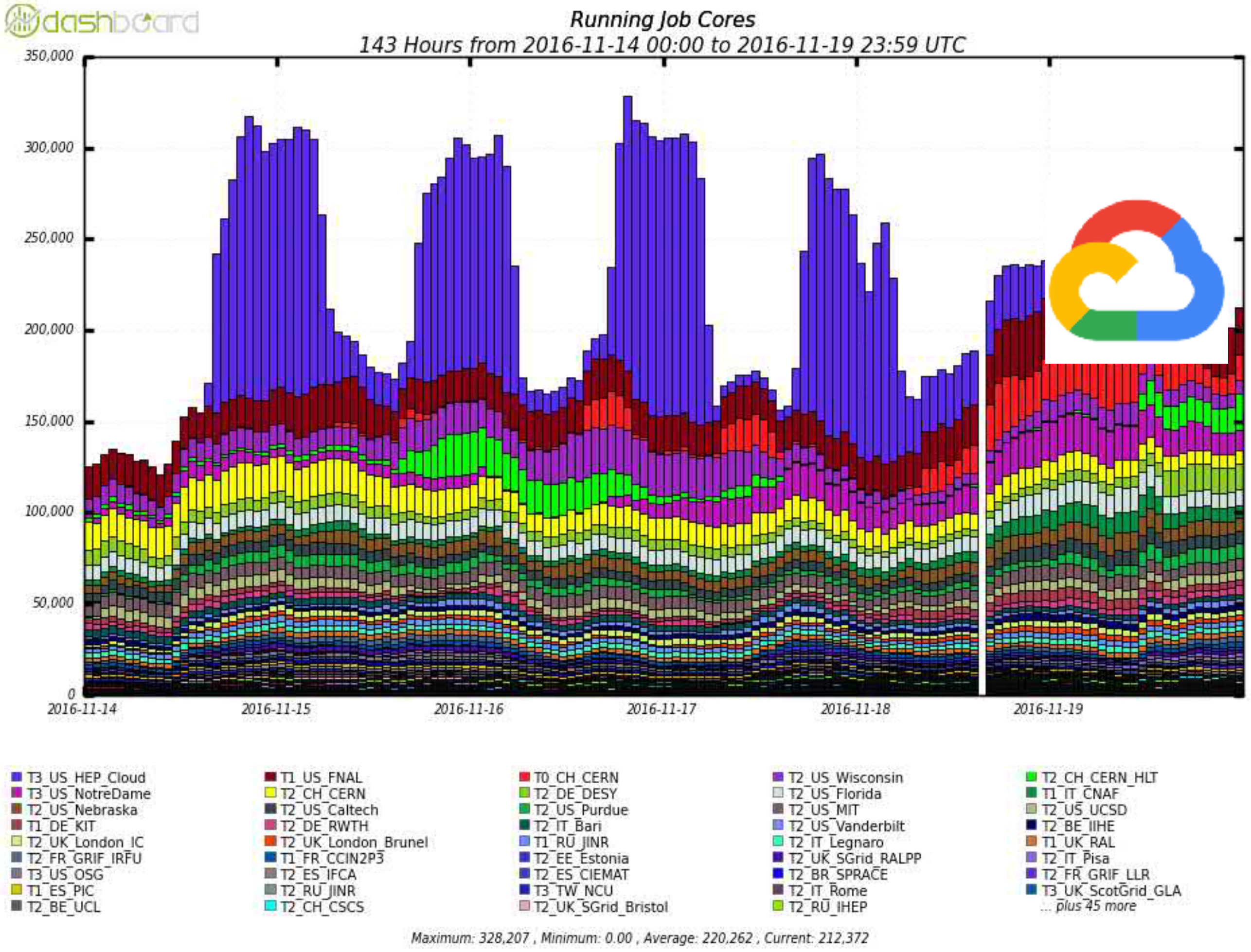




T. Bell, "Accelerating Cloud through science"

Scale out tests to commercial clouds





Explore Opportunistic use of:

HPC facilities, Large cloud providers, crowd-computing ?

CERN openlab

HPC resources

Our typical computing approach has been so far HTC oriented

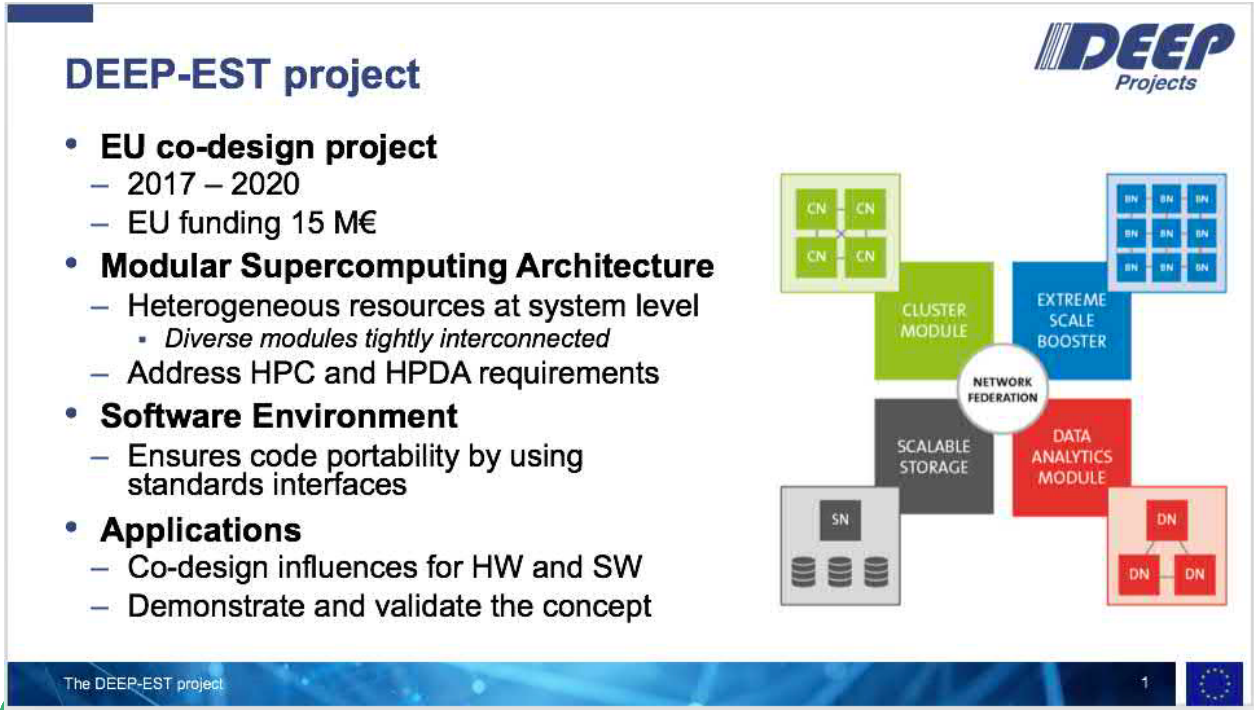
- HPC centers constitute an important resource

Being tested by the experiments

- For scalability and heterogeneous architectures
- ATLAS reached more than 200k traditional x86 HPC cores for simulation

CERN is part of EU funded DEEP-EST

- Research on modular HPC systems



Ex: Summit @Oak Ridge

#1 2018 Top500 list

System Performance	Each node has	The system includes
• Peak performance of 200 petaflops for modeling & simulation • Peak of 3.3 ExaOps for data analytics and artificial intelligence	• 2 IBM POWER9 processors • 6 NVIDIA Tesla V100 GPUs • 608 GB of fast memory • 1.6 TB of NVMe memory	• 4608 nodes • Dual-rail Mellanox EDR InfiniBand network • 250 PB IBM Spectrum Scale file system transferring

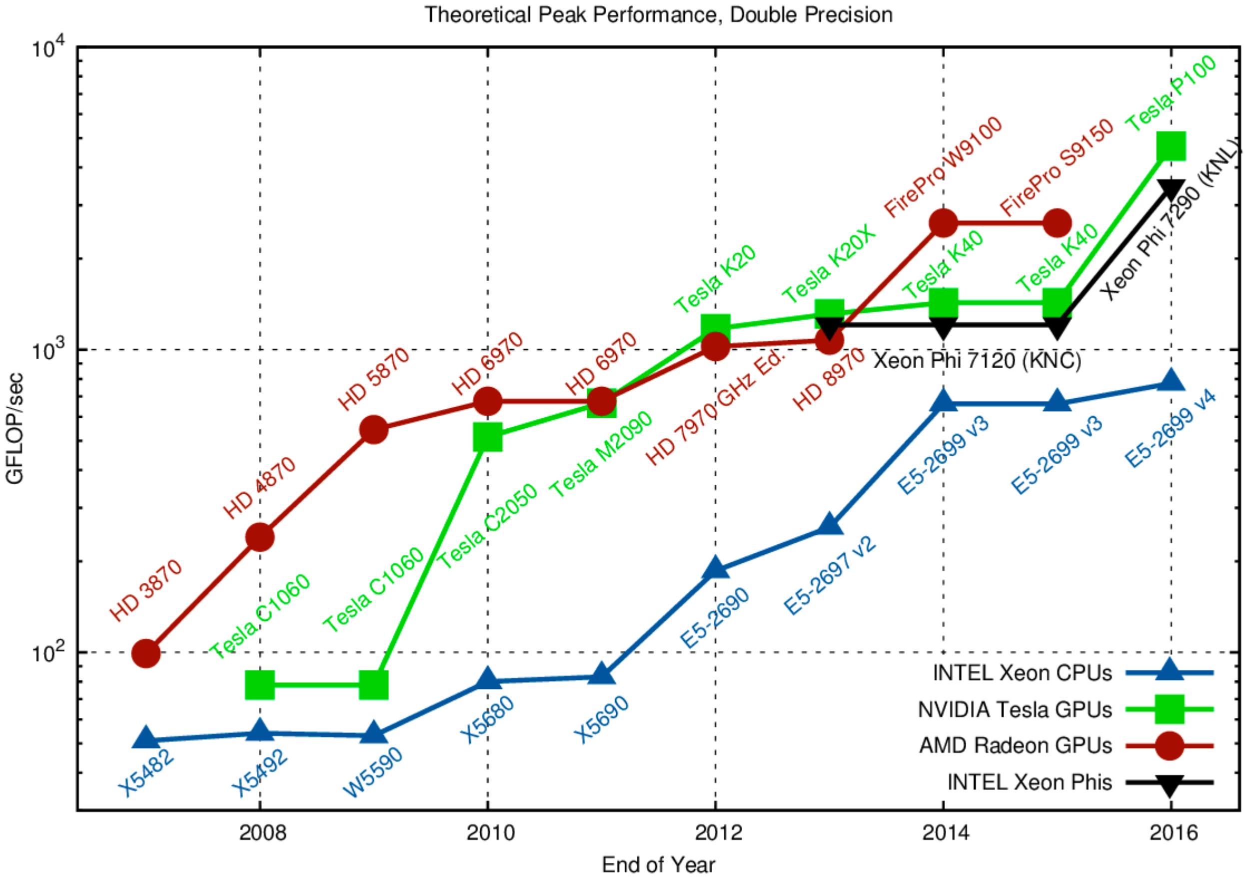
https://www.top500.org/static/media/uploads/top500_ppt_201806.pdf

H/W Accelerators

- Accelerators have different computing model than CPU
  - Many cores, high floating point throughput

- Ex. NVIDIA TESLA Kepler K40
  - 1.4 TFLOPS DP peak throughput
  - 288 GB/s peak off-chip memory access bandwidth
  - 36 G DP operands per second

- In order to achieve peak throughput, need 1,400/36 = ~39 DP arithmetic operations for each operand value fetched
  - In most of current code is 0.5 (fetch two operands, rarely use them again) ☹



Resource Starvation !

Software optimisation

Case by case investigation is needed

- Sometimes better start from scratch!

We need to rethink our algorithms in terms of

- Scalabilty
- Efficient use of resources
- Portability across platforms

traffic deadlock in Tel Aviv, 2011

Software

A few selected examples

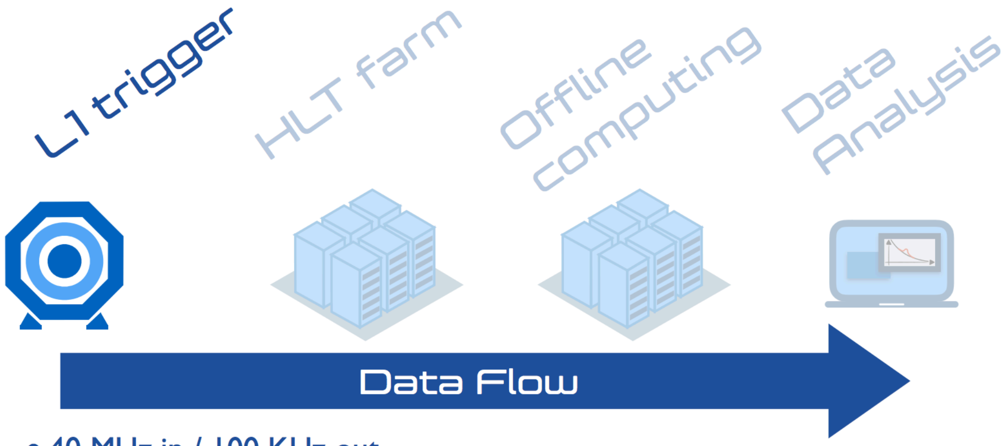
Trigger

Tracking

Simulation

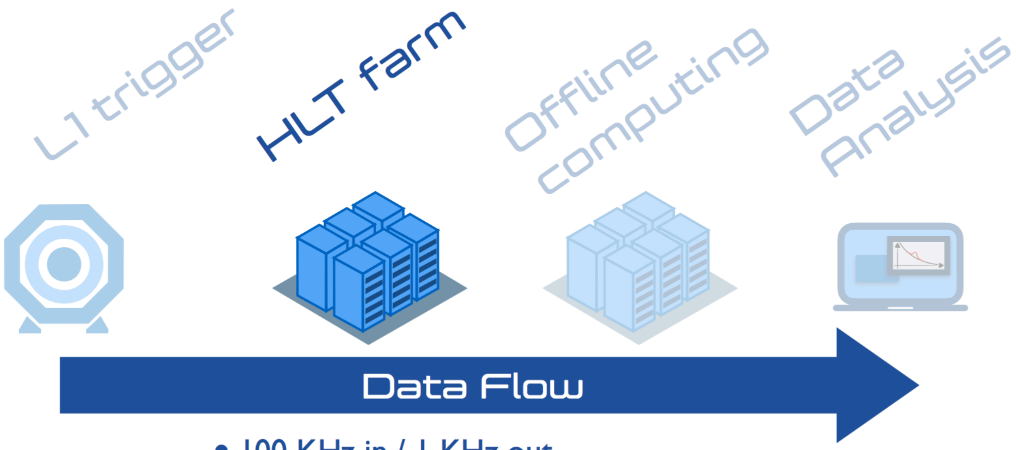
Analysis

Trigger: real time processing



- 40 MHz in / 100 KHz out
- ~ 500 KB / event
- Processing time: ~10 µs
- Based on coarse local reconstructions
- FPGAs / Hardware implemented

3

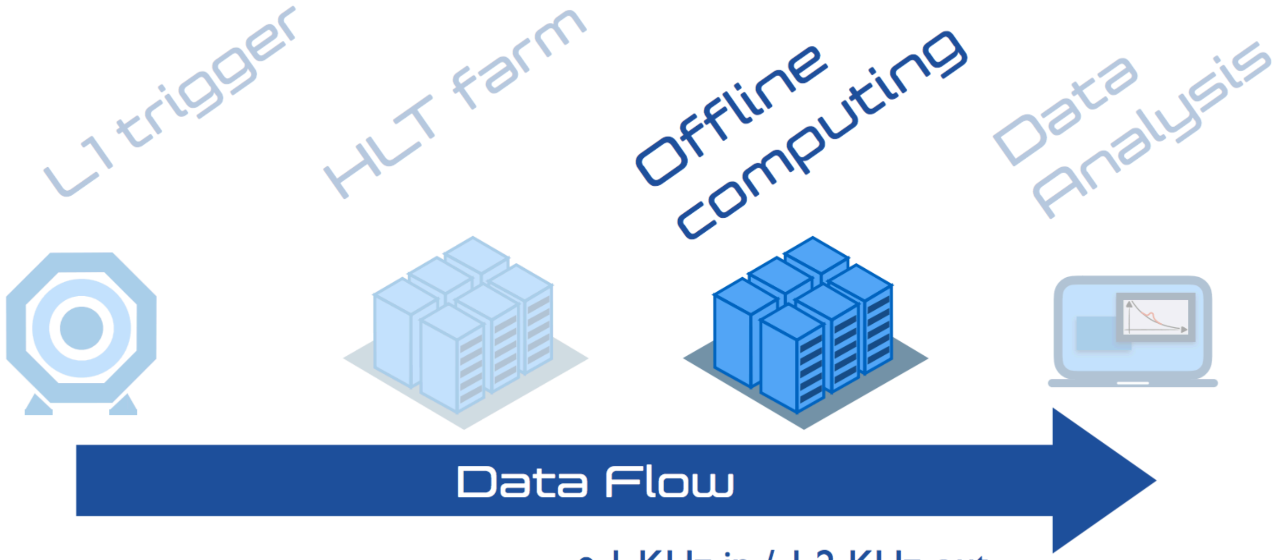


- 100 KHz in / 1 KHz out
- ~ 500 KB / event
- Processing time: ~30 ms
- Based on simplified global reconstructions
- Software implemented on CPUs

4

Experiment currently implement both hardware and software stages (“cascade”)
Feature-building in custom electronics (e.g. FPGAs) reduces rate

Offline processing



- 1 KHz in / 1.2 KHz out
- ~ 1 MB / 200 KB / 30 KB per event
- Processing time: ~20 s
- Based on accurate global reconstructions
- Software implemented on CPUs

5

Organized processing, one software stack/framework per experiment (C++), one or few output sets

Mostly done on WLCG

Trigger challenges

Data rates

	Incoming rate (kHz)	Outgoing rate (kHz)	Reduction factor
L1	40000	$10^2 – 10^3$	400-10,000
HLT	2-1000	1 -10	10-2000

HL-LHC

HLT	Event size (kB)	rate (kHz)	Bandwidth (Gb/s)	Year
CMS	4000	10^3	32000	2023
LHCb	100	40000	32000	2019

LHCb is investigating FPGAs and GPUs for real time reconstruction of 5GB/s

CMS is porting heavy ”offline” tasks to real-time processing

- Integrate GPUs in the HLT farm to achieve 100 msec latency (now O(10) sec)

Online vs Offline

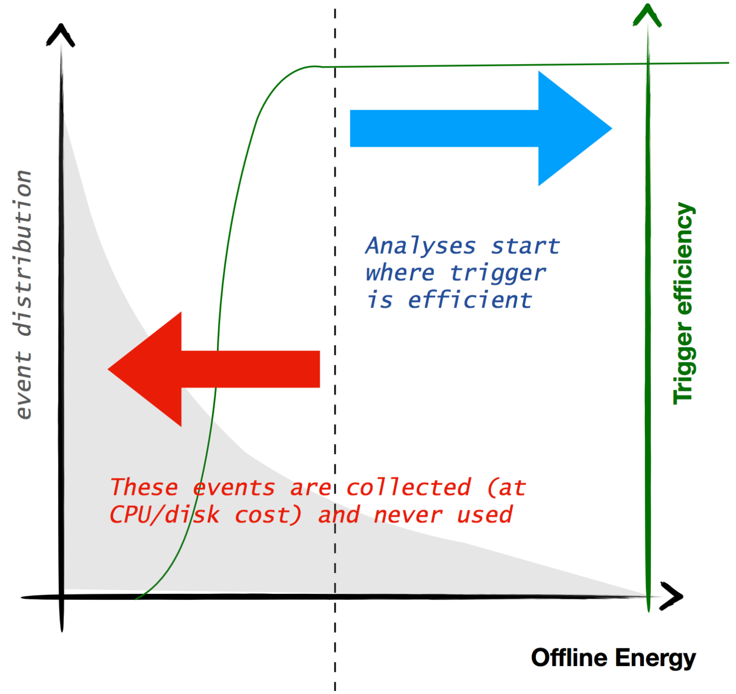
Trigger efficiency

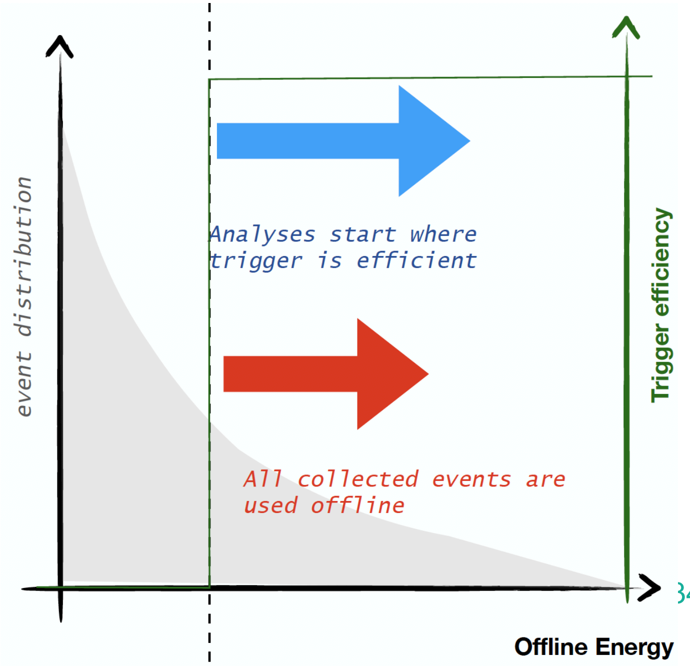
Online vs offline reconstruction differences are limiting discovery reach

- Online selection reduces sensitivity to new physics (tails of event distribution)
- Not optimal use of resources

Having the same reconstruction at L1/HLT/Offline would recover lost sensitivity

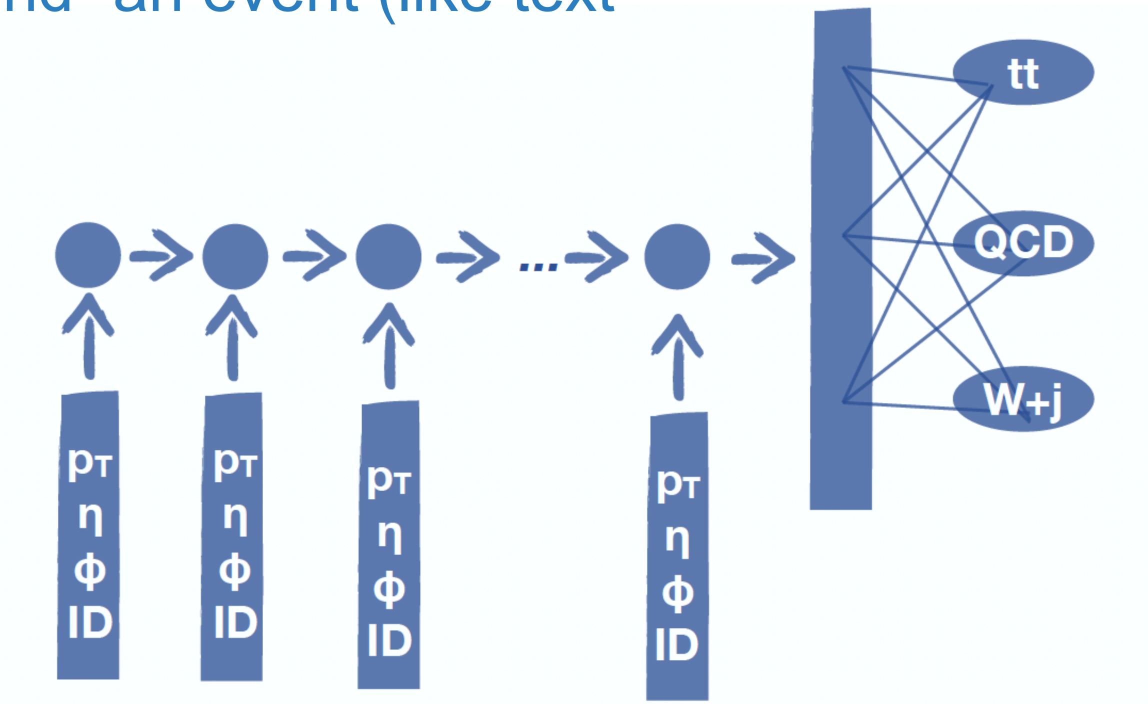
- This cannot be done exactly offline code too slow
- It could be done “in average” → ML algorithms





EX R&D: ML/DL for Trigger

- Event as a sentence
  - Events are made of particles like sentences are made of words
  - Physics is the grammar that dictates the order
- Use recursive neural networks to "understand" an event (like text understanding applications)



Images courtesy of Maurizio Pierini, CERN CMS

D. Weitekamp, 2017 CERN OpenLab Summer Student

NN based Trigger

Topology trigger!

tt events are a tiny fraction in single-lepton datasets

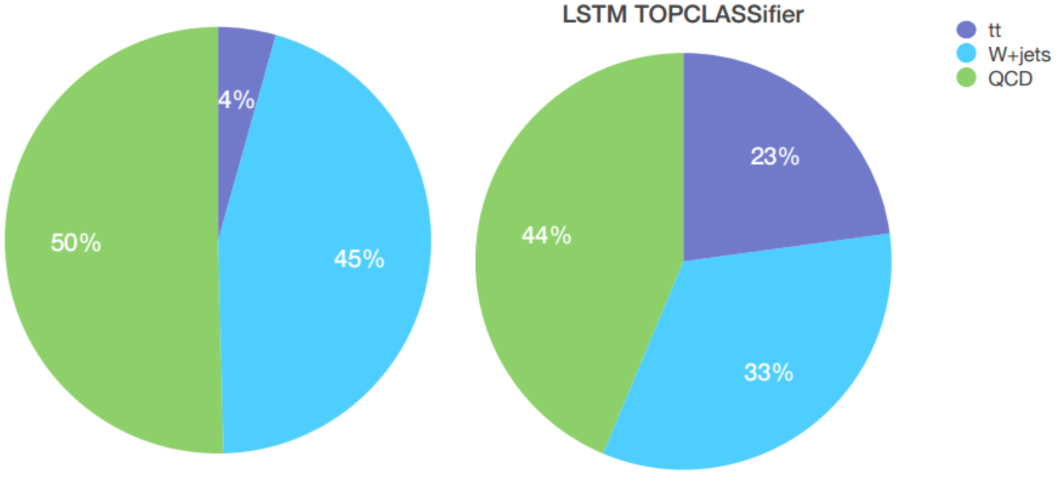
Most triggers are object- and not topology-based

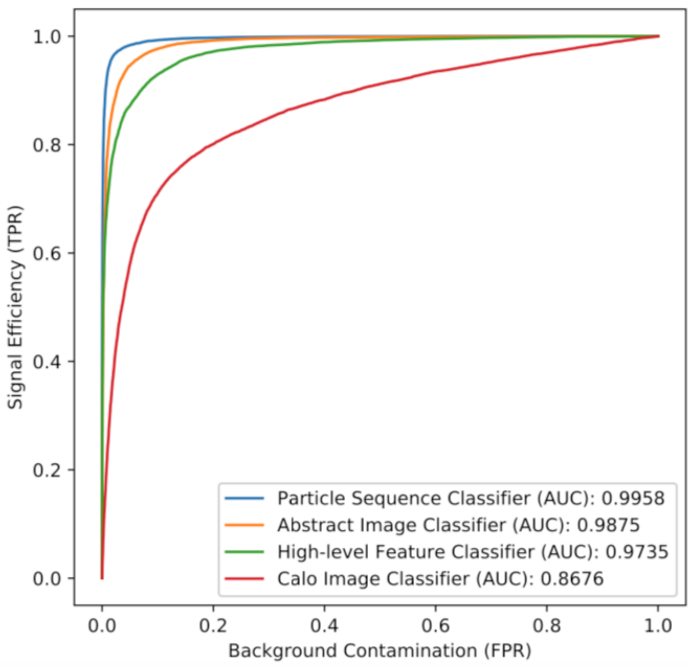
Represent the topology in a DL-compliant way

DL"designs" the best classification criterion

Strong QCD/W+j background rejection for 99% efficiency on tt events

Such a filter at trigger level could save x10 downstream resources



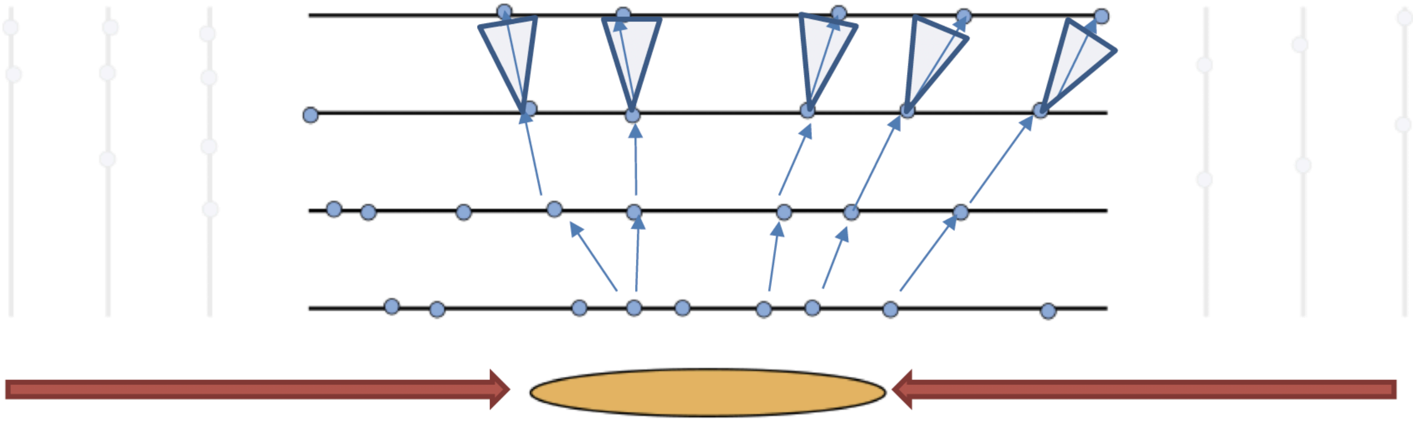


Track Seeding and pile-up

Typical approach is not easily parallelisable

First create doublets from a pair of layers

- propagate generated doublets to third layer
- propagate triplets to fourth layer and store
- start from another pair of layers

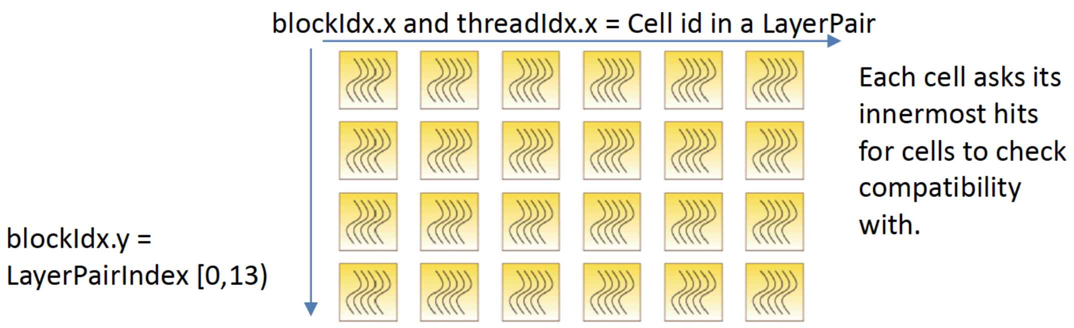


Absence of massive parallelism

- Poor data locality
- Synchronizations due to iterative process

https://indico.cern.ch/event/656491/contributions/2939163/attachments/1631960/2602215/18-04-11_FCCWeek_TrickTrack_1.pdf

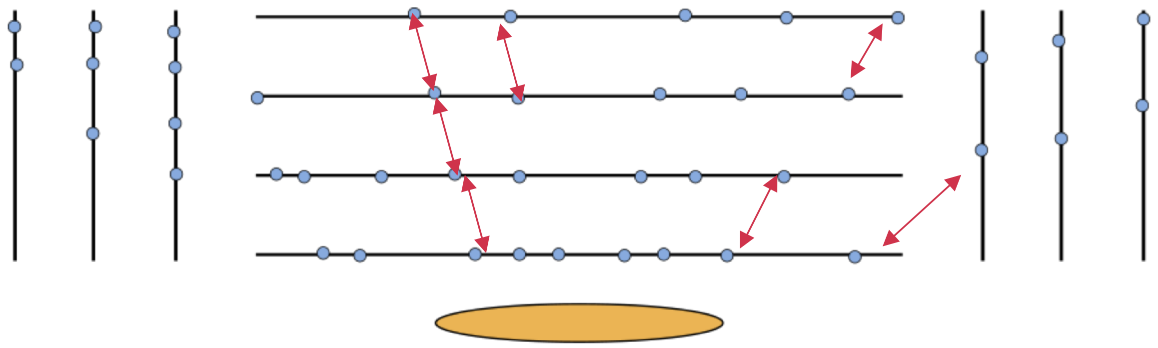
Ex: Parallel tracking

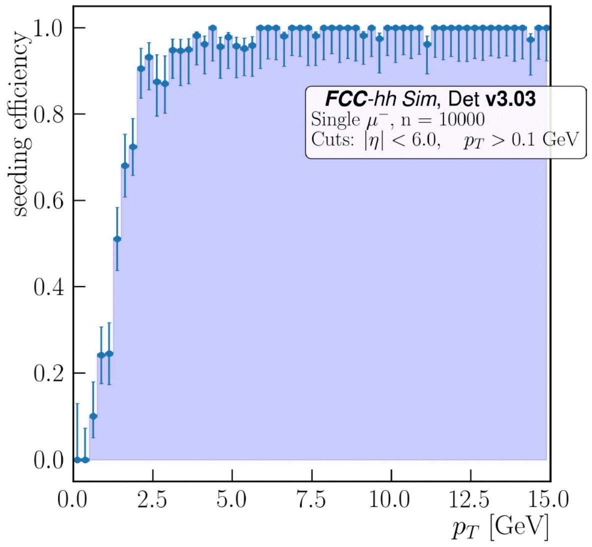


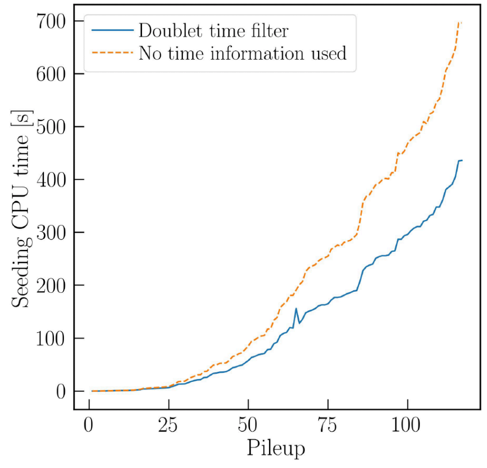
Parallelization requires algorithmic design

- Cellular Automaton (CA): parallel track seeding algorithm
- Doublets (Cells) are created for each pair of layers (compatible with a region hypothesis)
- Fast computation of the compatibility between two connected cells
- No knowledge of the world outside adjacent neighboring cells required

easy to parallelize

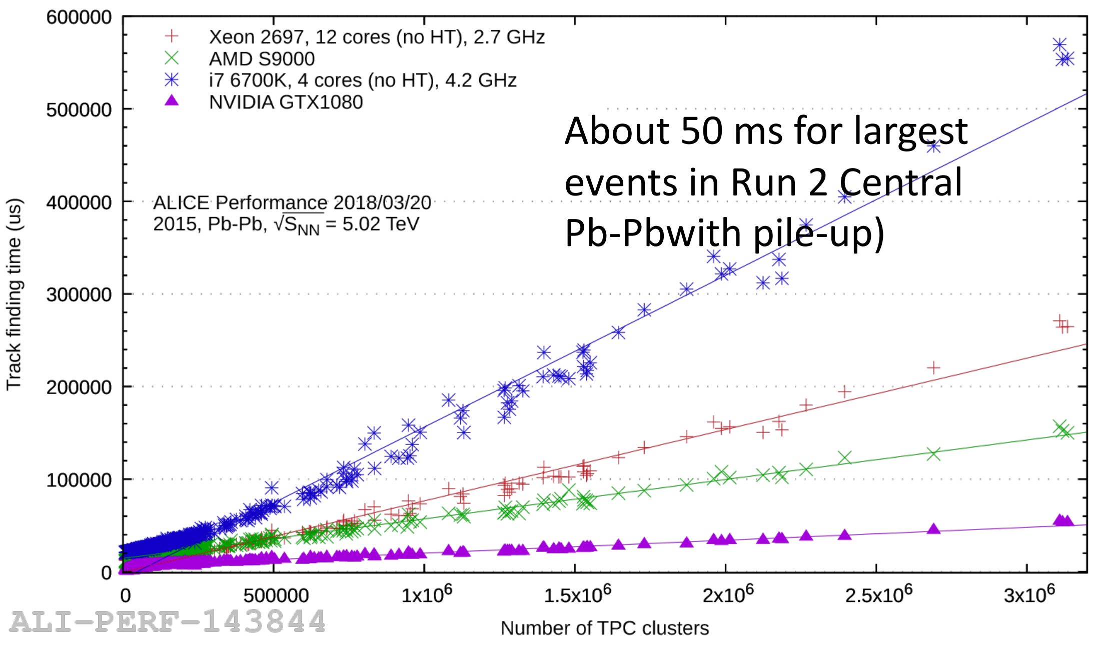


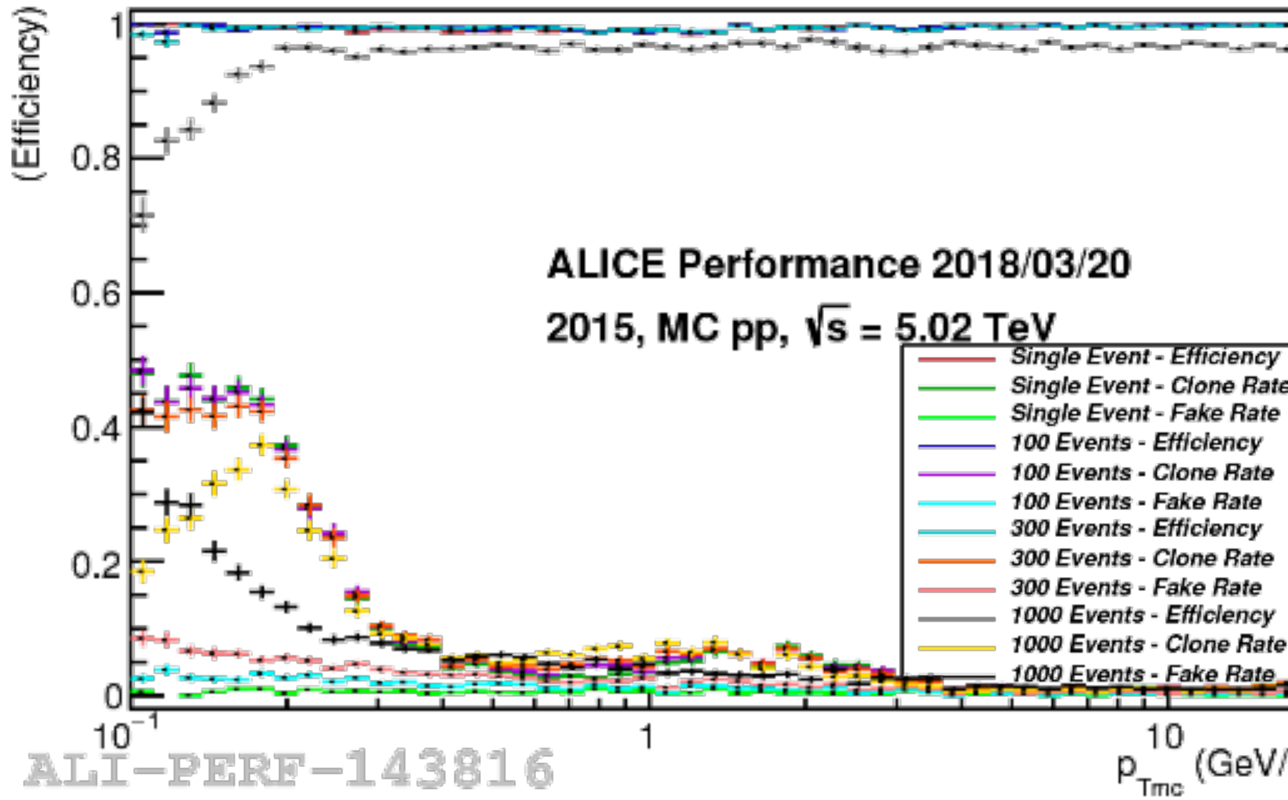




GPU accelerated HLT in Alice

CA based tracking implemented and tested on Pb-Pb events





https://indico.cern.ch/event/658267/contributions/2813689/attachments/1621144/2579443/2018-03-21_CTD_2018.pdf https://ctdwit2017.lal.in2p3.fr/

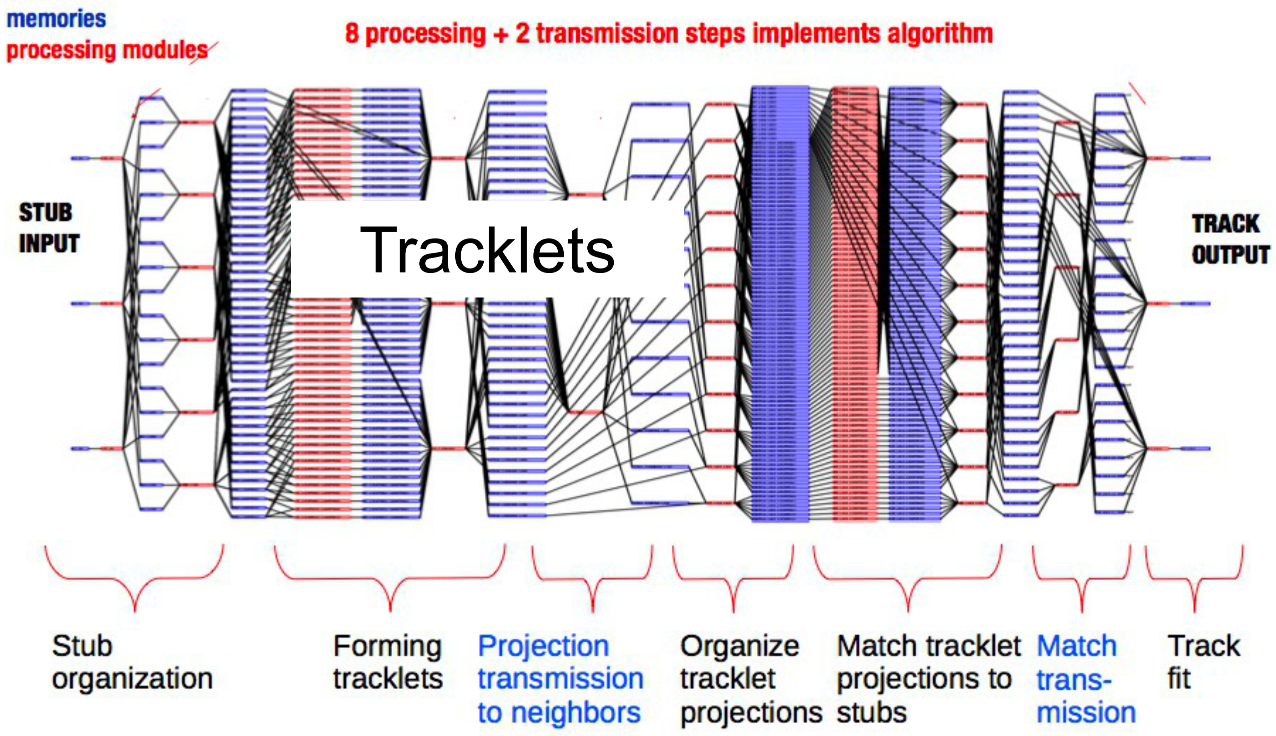
Hardware tracking

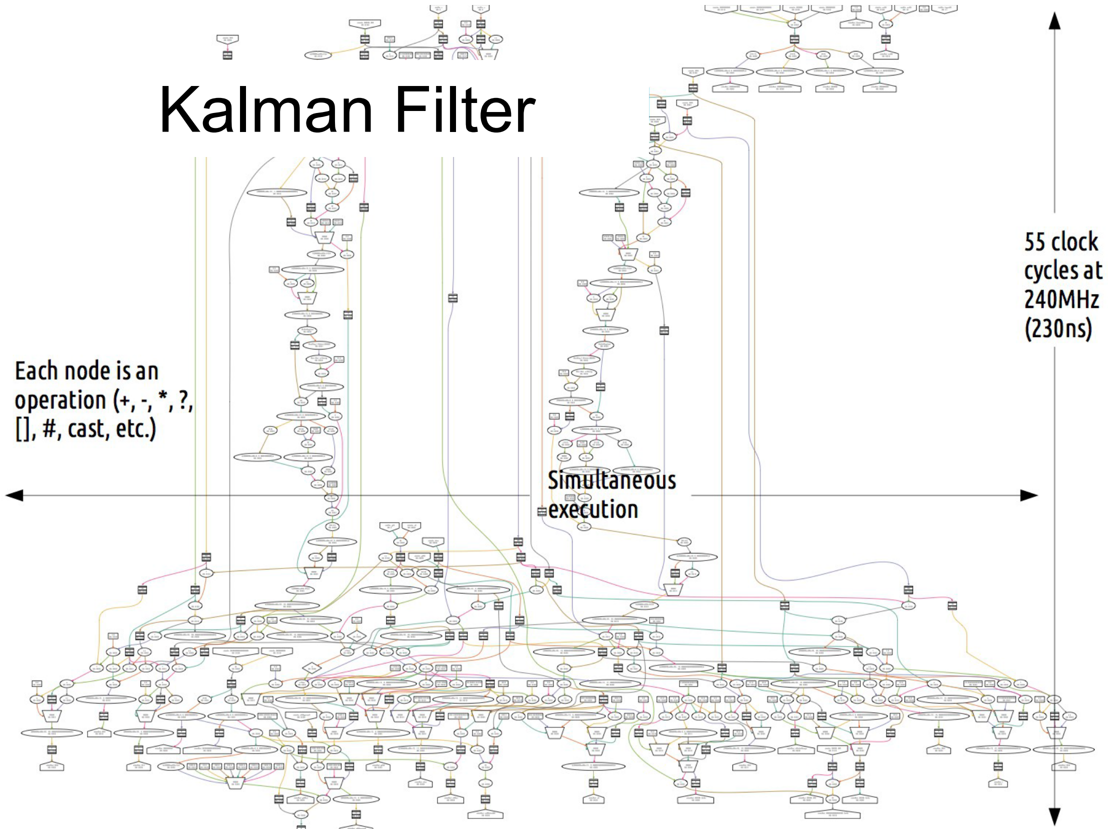
Track trigger implementation for trigger upgrades development on-going

- Several approaches investigated

Dedicated hardware is the key to fast computation

- Not applicable for offline processing unless by adopting heterogeneous hardware.





ML for tracking

Recent work applies ML/DL to particle tracking:

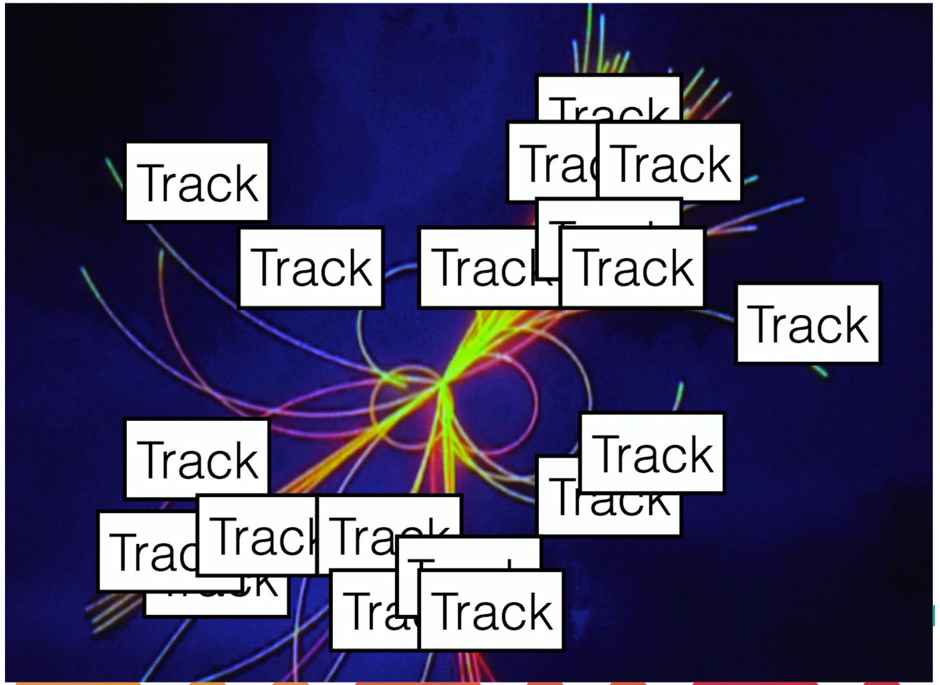
Hopefield network http://inspirehep.net/record/300646/

CNN in NOVA https://arxiv.org/abs/1604.01444

HEP.TrkX : https://heptrkx.github.io/

TrackML RAMP : https://tinyurl.com/y84yd5hn

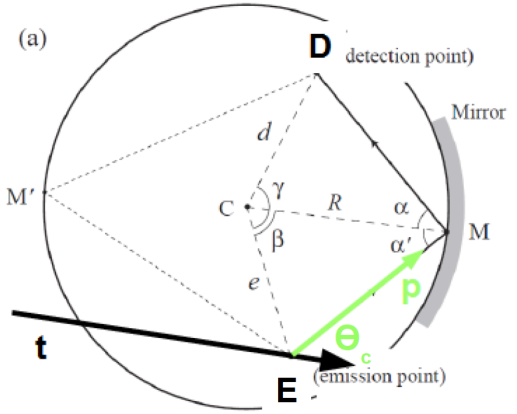
TrackML challenge on kaggle!



https://indico.cern.ch/event/658267/timetable/#20180322.detailed
https://www.desy.de/dvsem/WS1213/pantaleo_talk.pdf
https://indico.cern.ch/event/656491/contributions/2939164/attachments/1631963/2602408/JuliaHrdinka_FCCweek2018.pdf
https://indico.cern.ch/event/658267/contributions/2813689/attachments/1621144/2579443/2018-03-21_CTD_2018.pdf

Accelerating PiD with FPGA

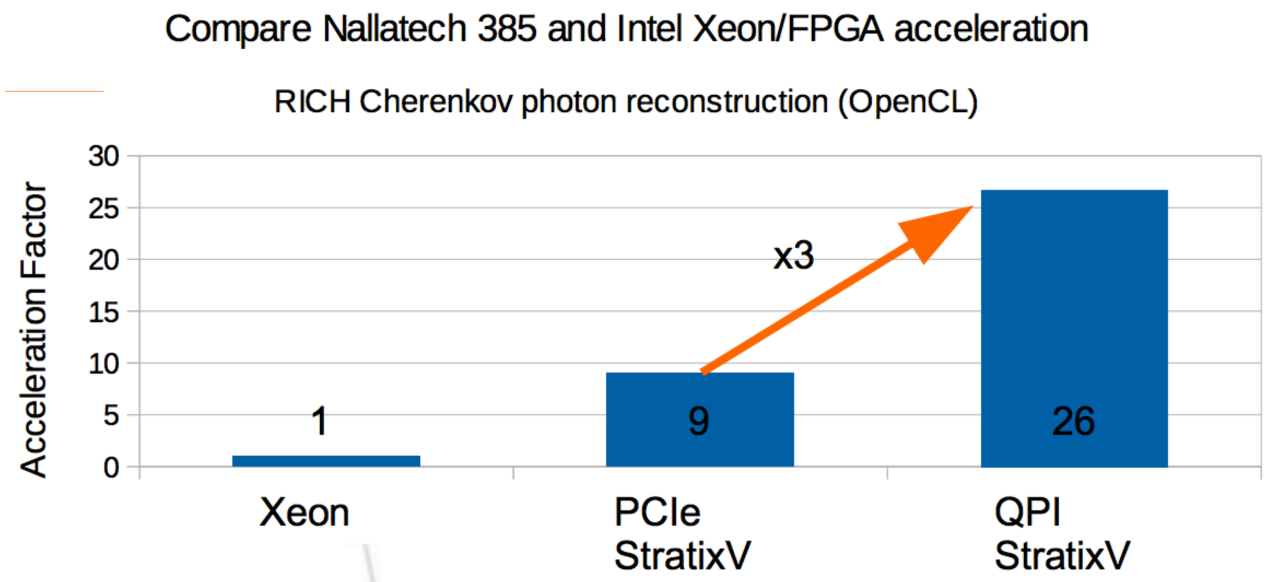
LHCb RICH

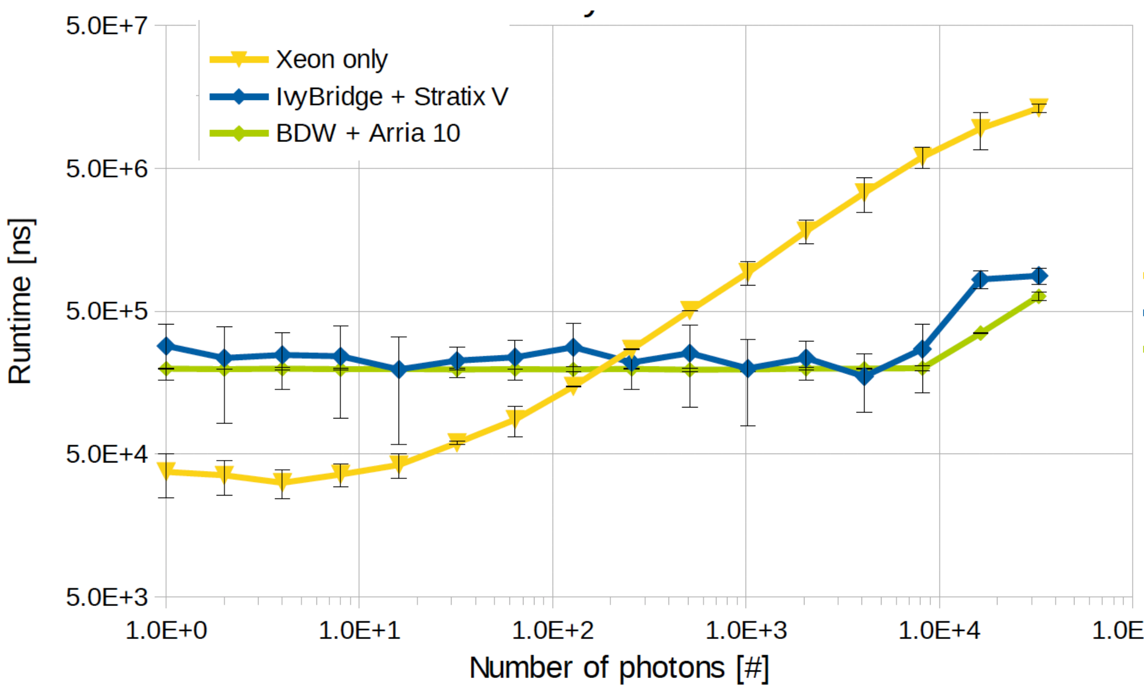


Reconstruction of Cherenkov angle

- Acceleration up to x35 with Intel Xeon-FPGA wrt single Xeon

Bottleneck: Data transfer bandwidth to FPGA





C. Farber, LHCb https://indico.ihep.ac.cn/event/6387/session/56/contribution/50/material/slides/0.pdf

Simulation

V. D. Elvira, CHEP 2018
https://indico.cern.ch/event/587955/contributions/2937511/attachments/1678317/2695427/DE-T2Offline-Abs30.pdf

Economic impact/cost of simulation in HEP collider experiments

We define "simulation chain" physics generation, interaction with matter (G4), readout modeling, reconstruction, analysis

- Took 85% of CPU resources used by CMS, while G4 module took 40% of total (Run 1, 2)
- ATLAS's Geant4 module was 8-9 times slower than CMS's and the experiment uses significantly more resources than CMS in physics generation
- Rest of resources used in reconstruction and analysis of real collider data

CMS in more detail taken from (analysis of 2012, and May 2015-May 2016 periods)

- 540k/860k core months corresponding to 45/70k CPU cores at full capacity (half in G4)
- Purchasing cost is 5/8 million dollars
- Cost of physical hardware including life-cycle, operation, maintenance
  - 0.9 cents/core hour (FNAL), or 1.4 cents/core hour (what FNAL paid industry in 2017)
- Annual cost of simulation in CMS: 3.5-6.2/5.5-10 million dollars
- Improvements of 1%, 10%, 35% in G4 time performance would render 50-80k, 500-800k, 1.8-2.8M dollars savings to CMS

Computing needs of HL-LHC program are 10-100 times higher depending on simulation and reconstruction solutions implemented – reconstruction will take a larger fraction (pileup)

Fermilab

CERN openlab

Speeding up simulation

See A. Dotti G4 tutorial

Intense R&D activity on code modernisation

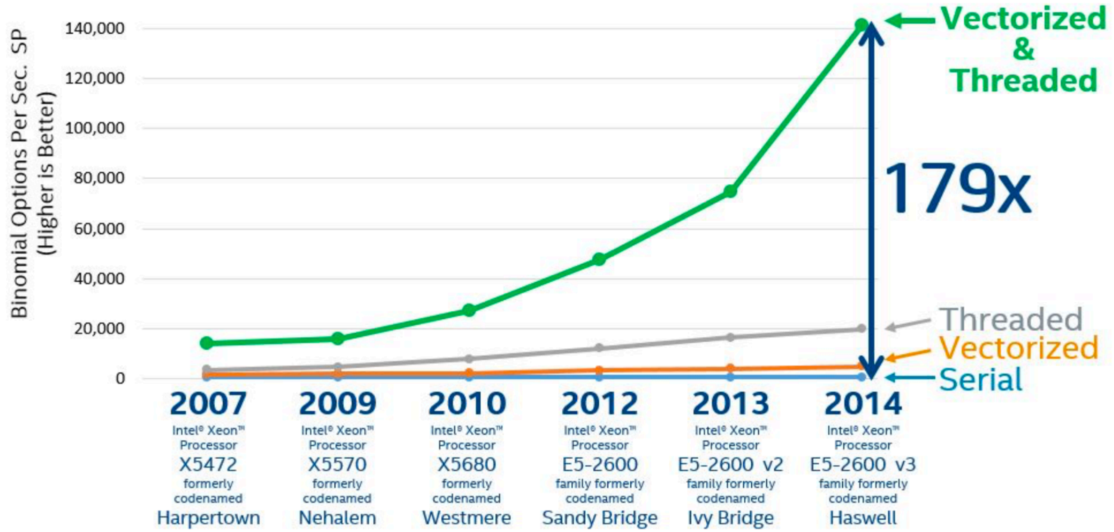
- Improve existing code (**Geant4** - scalar processing)
  - Reduce memory consumption
  - Implement event level parallelism

Speeding up simulation

Intense R&D activity on code modernisation

- Improve existing code (**Geant4** – scalar processing)
  - Reduce memory consumption
  - Implement event level parallelism

We need to approach the problem at multiple levels!



source: Intel

CERN openlab

Speeding up simulation

Intense R&D activity on code modernisation

- Improve existing code (**Geant4** – scalar processing)
  - Reduce memory consumption
  - Implement event level parallelism
- Prototype fine grained parallelism through the **GeantV** “project”
  - Improved, vectorised physics models
  - Improved, vectorised geometry (**VecGeom**)
  - Smart track level parallel transport
- Back-propagate improvements to Geant4

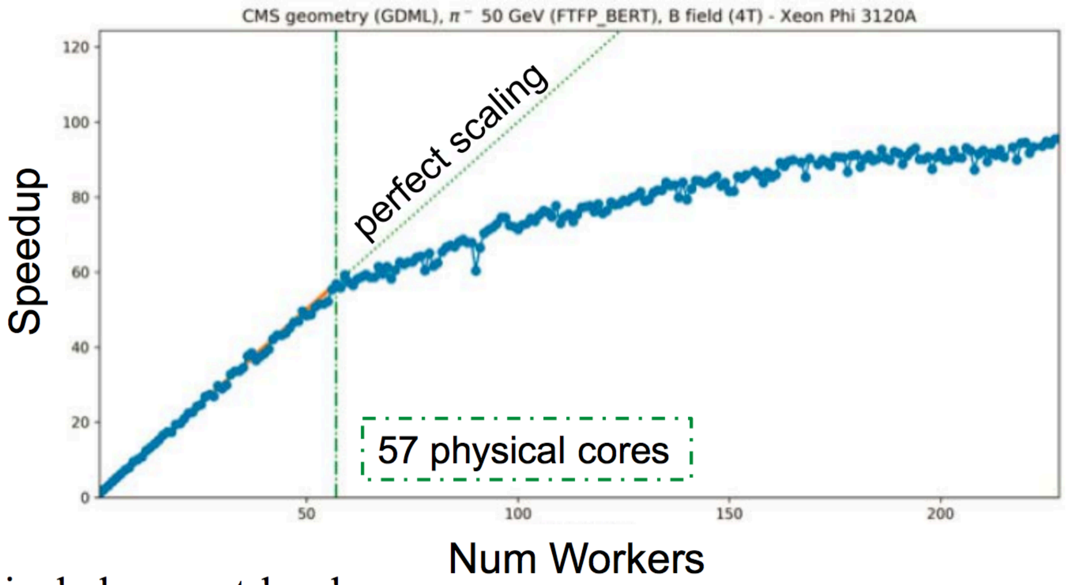
http://geant.cern.ch

CMS case study

Test improvements in a real-life scenario: CMSSW

K.Pedro, CHEP 2018.
https://indico.cern.ch/event/587955/contributions/2937652/attachments/1679306/2697284/CMS_simulation_performance_CHEP2018.pdf

Multithreading



- Geant4 includes event-level multithreading
- Nearly perfect scaling with physical cores, further 30% gain from hyperthreading
- Memory reduced by factor of 10 (vs. multiprocessing approach)

- CMSSW framework supports multithreading
- Similar gains in throughput observed, memory usage remains under 2GB
- ➢ More efficient use of grid resources (included in CMS production releases)

Speeding up simulation: CMS case study

Results of Existing Improvements

	Relative CPU usage	
Configuration	**MinBias**	**ttbar**
No optimizations	1.00	1.00
Static library	0.95	0.93
Production cuts	0.93	0.97
Tracking cut	0.69	0.88
Time cut	0.95	0.97
Shower library	0.60	0.74
Russian roulette	0.75	0.71
FTFP_BERT_EMM	0.87	0.83
All optimizations	0.21	0.29

- From HEP Software Foundation Community White Paper
  - CMS Phase 0 detector, Geant4 10.2
- HF shower library, Russian Roulette have largest impacts
- Cumulative effects: with all improvements, simulation is 4.7× (3.4×) faster for MinBias (ttbar)
- CMS simulation takes 4.3 sec†/event (24.6 sec†/event) for MinBias (ttbar)

†1 sec = 11 HS06 for test machine

New Improvements: Geometry

VecGeom: new library for detector geometry

- Supports vectorization and new architectures
- Code rewritten to be more modern and efficient (vs. Geant4, ROOT, USolids)
- Can be used in scalar mode with Geant4
- CMS observes 7–13% speedup with similar memory usage

  → Just from code improvements, no vectorization!

➢ Included in latest CMS production releases
  - First mainstream use of vectorized library by experiment

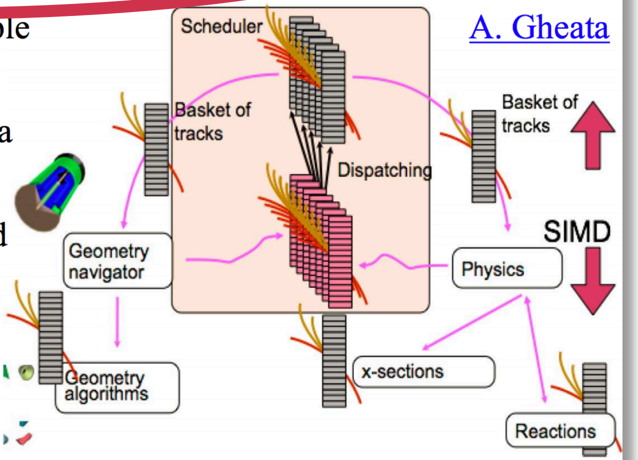
	Relative CPU usage	
Geometry library	**MinBias**	**ttbar**
Native	1.00	1.00
VecGeom	0.87	0.93

CMS case study: GeantV integration

Potential Improvements: GeantV

- CMS has already achieved significant speedups in Geant4 and enabled event-level multithreading for more efficient use of resources
- However, even this will not suffice for the demands of Phase 2
- Enter GeantV: Vectorized Transport Engine
  - o Track-level parallelism: process multiple events simultaneously
  - o Exploit single instruction, multiple data (SIMD) vectorization
  - o Group similar tracks into *basket* (based on particle type, geometry/material)
  - o Send entire basket to algorithm: process particles in parallel

A. Gheata



Conclusions

- CMS has substantially reduced CPU usage of Geant4 full simulation
  - o ~3–5× speedup using various technical improvements and physics-preserving approximations
  - o Continue to find ~10% improvements, e.g. from VecGeom and magnetic field stepper/tracking optimizations
- HL-LHC and Phase 2 upgrades bring significant challenges:
  - ➢ Need more events, more accuracy, in more complicated geometry… w/ relatively smaller fraction of total CPU usage
- GeantV is one promising approach to speed up full simulation even further
  - o Track-level parallelism (rather than event-level), vectorized components
  - o Alpha release is available, beta release planned for 2019
  - ➢ Successful early integration in CMS software framework!
  - o Aim for 2–5× speedup with final version

Fast Simulation

Already used for searches, upgrade studies,…

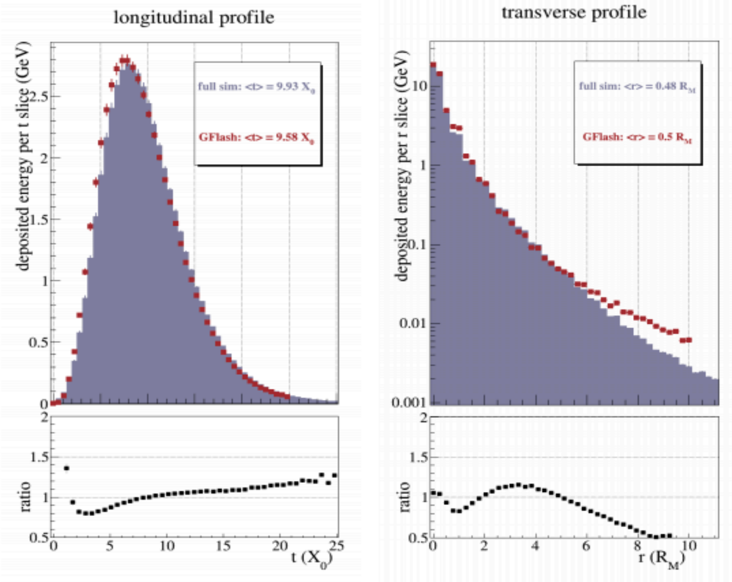
Different techniques

- Shower libraries (pre-simulated EM showers, fwd calorimeters in ATLAS/CMS)
- Shower shapes parametrizations (GFlash,..)
- Fast trackers simulation (ATLAS FATRAS, ..)
- Look-up tables
  - Hit library (LHCb)
- Fully parametrized simulation (DELPHES - see tutorial)

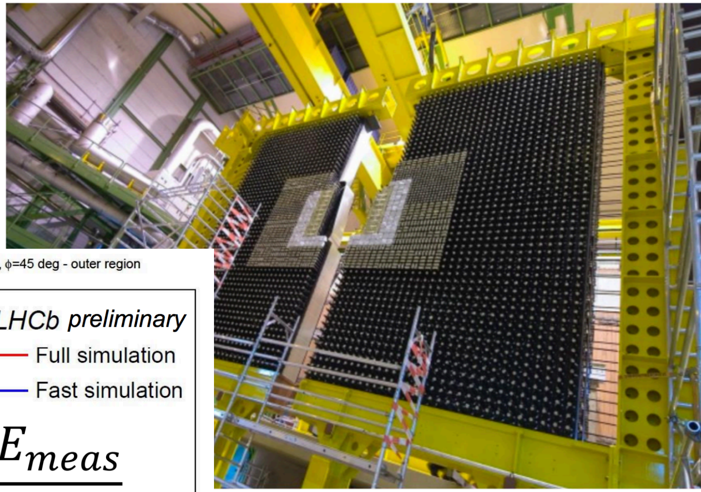
Different performance

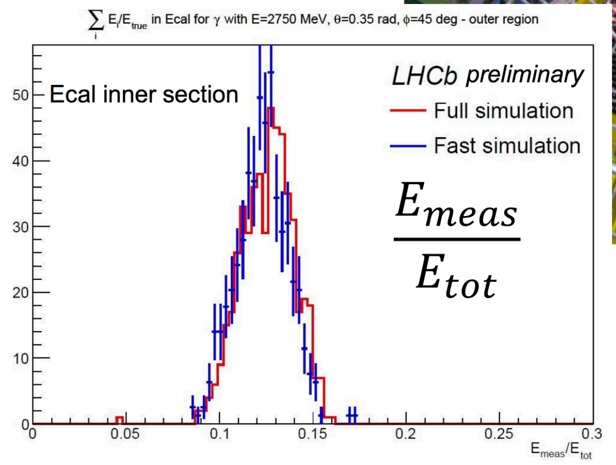
- Different speed improvements (x10 - x1000)
- Different levels of accuracy (~10% wrt full sim)

FCChh



Zaborowska, CHEP2016





M. Rama, LHCb, CHEP2018

CERN openlab

A generic framework?

MC need to integrate fast simulation

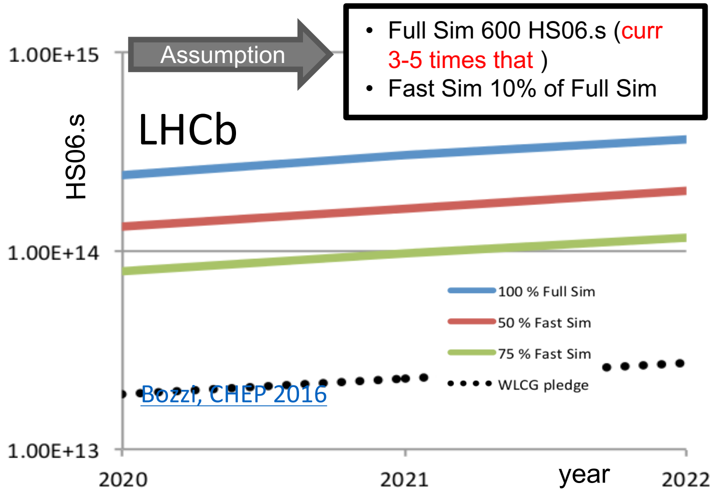
- Geant4 has mechanism to mix fast and full simulation: user-defined models within “envelopes” → few use it

Towards a common framework providing

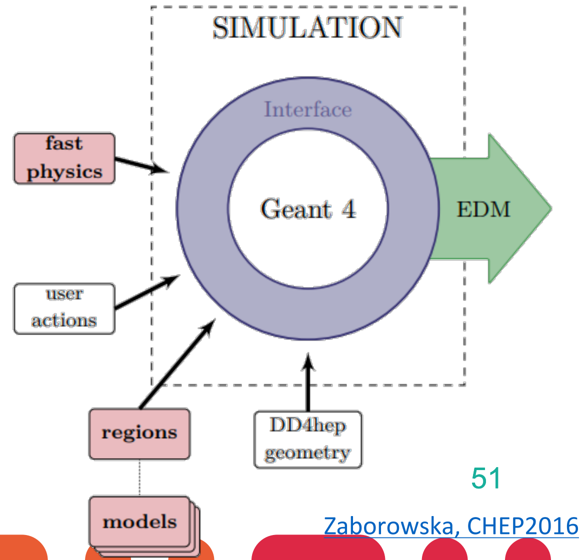
- Algorithms and tools
- Mechanism to mix fast and full simulation according to particle type and detector

R&D within CERN openlab to develop a generic fully customizable fast sim framework

- Deep Learning based



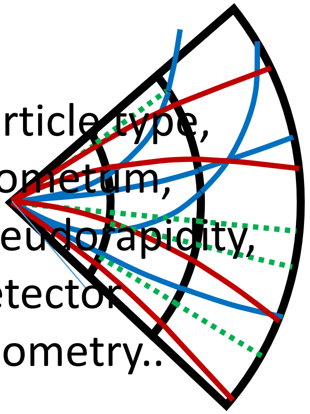
FCC Gaudi framework



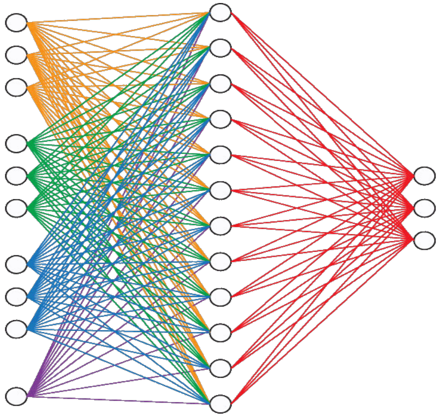
Zaborowska, CHEP2016

Deep Learning for fast sim

EX. SIMULATION OF A CALORIMETER

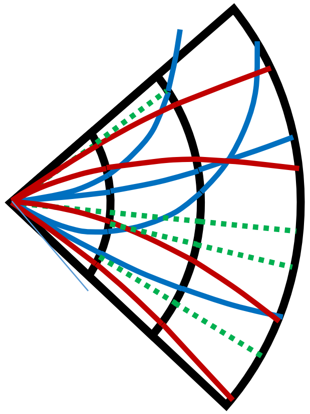


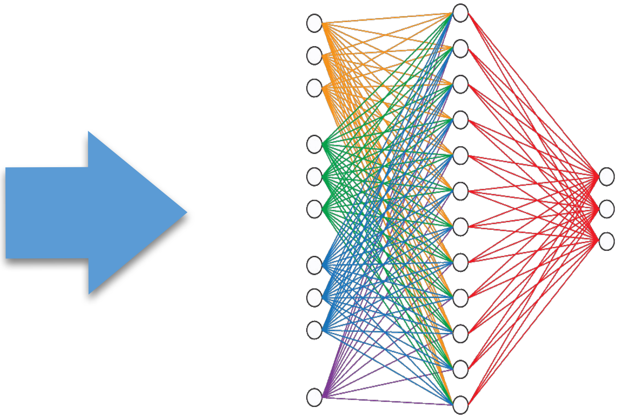
Particle type,
mometum,
pseudorapidity,
detector
geometry..



Energy depositions in cells

Deep Learning for fast sim





Generic approach

Can encapsulate expensive computations

DNN inference step is generally faster than algorithmic approach

Already parallelized and optimized for GPUs/HPCs.

Industry building highly optimized software, hardware, and cloud services.

Numerous R&D activities (LHC and beyond) (see results presented at CHEP2018)

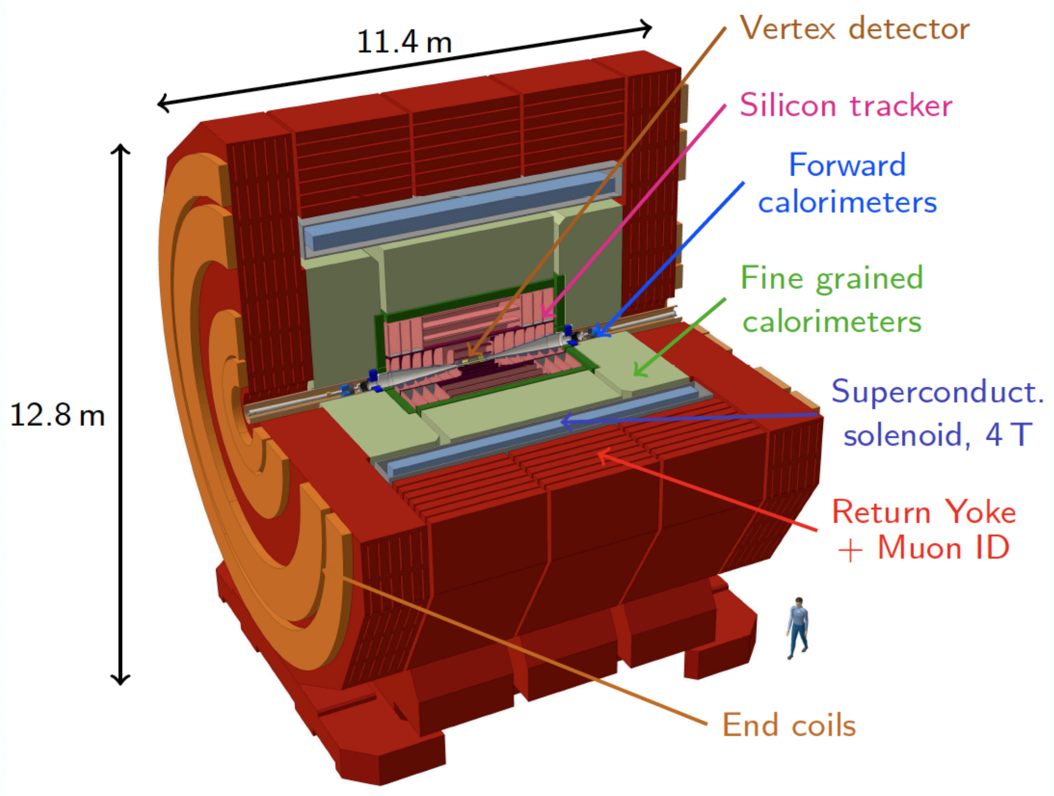
Example: Generative Adversarial Networks for CLIC high granularity calorimeter

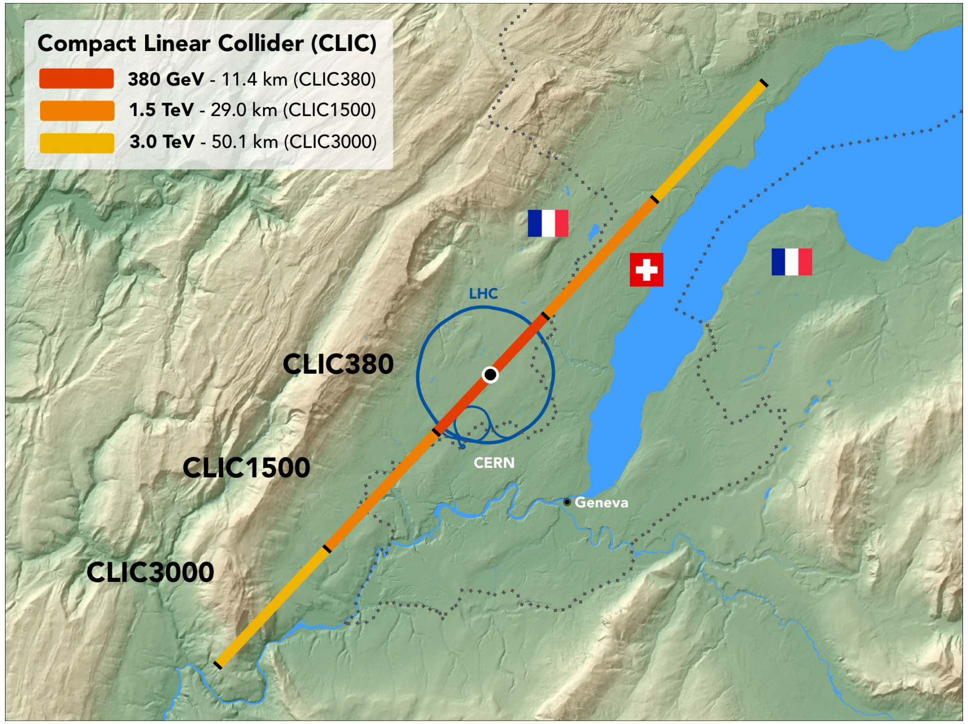
What is CLIC?

Compact LInear Collider

High-luminosity linear e+e- collider

Three energy stages up to 3 TeV





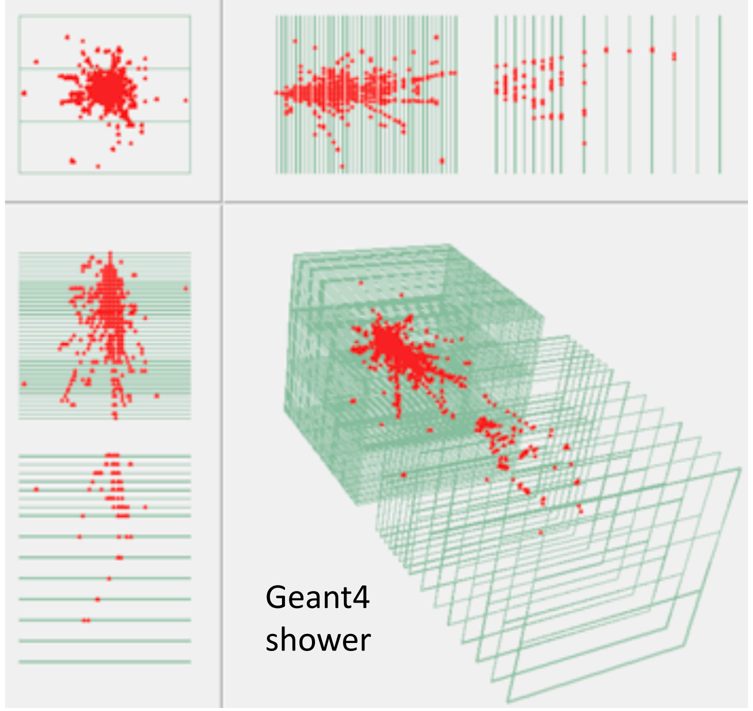
Electromagnetic calorimeter detector design

1.5 m inner radius, 5 mm×5 mm segmentation: 25 tungsten absorber layers + silicon sensors

http://cds.cern.ch/record/2254048#

CLIC calorimeter simulation

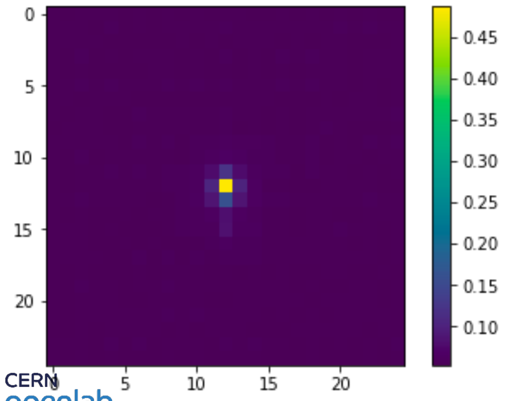
Data is essentially a 3D image

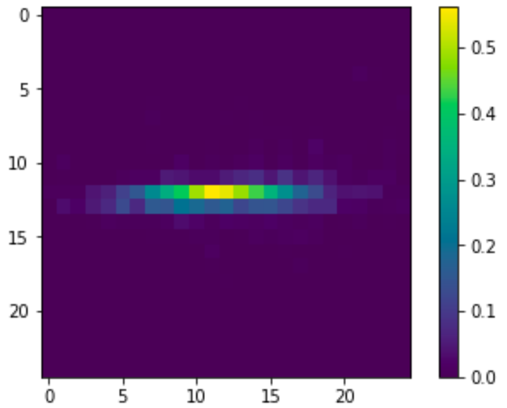


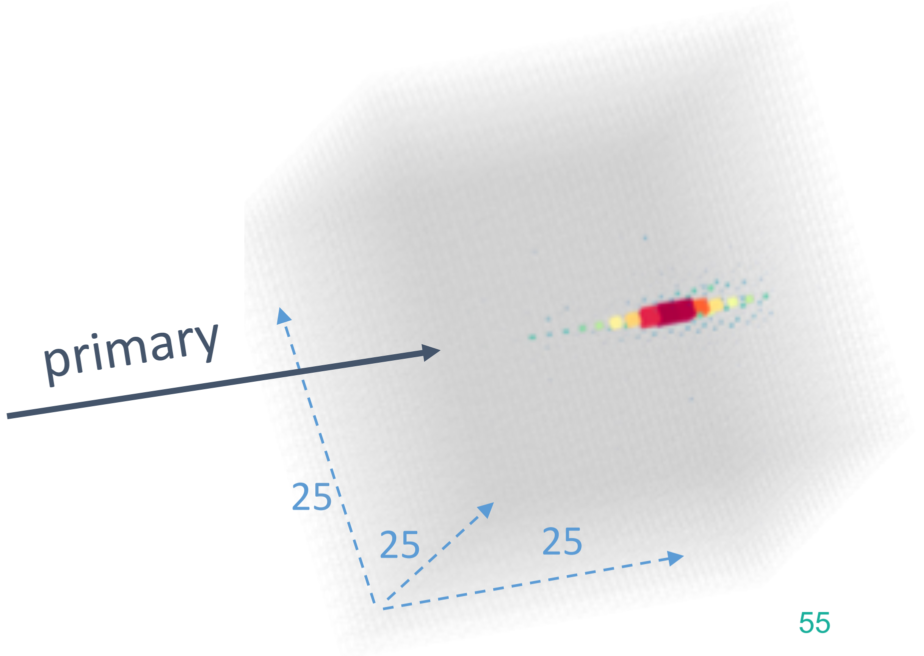
1M single particle samples (e,γ,π)

- Flat energy spectrum (10-500) GeV
- Orthogonal to detector surface
- +/- 10° random incident angle

Images are highly segmented and sparse







CERN openlab

arXiv:1406.2661v1

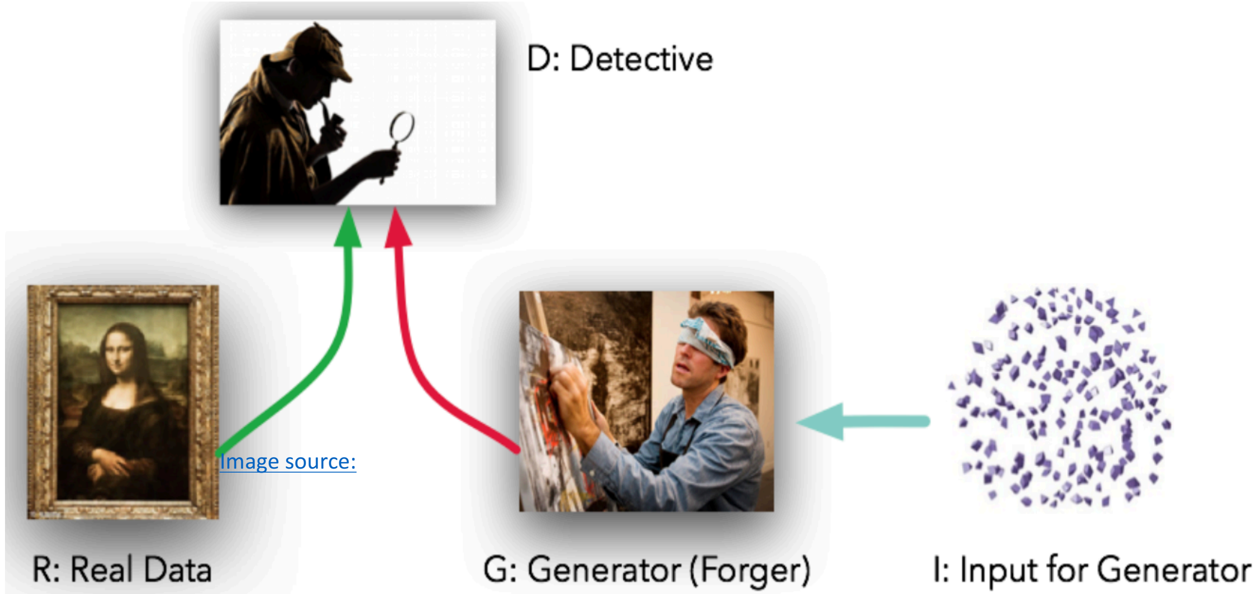
Generative adversarial networks

Simultaneously train two networks that compete and cooperate with each other:

Generator G generates data from random noise

Discriminator D learns how to distinguish real data from generated data

https://arxiv.org/pdf/1701.00160v1.pdf



The counterfeiter/detective case

Counterfeiter shows the Monalisa

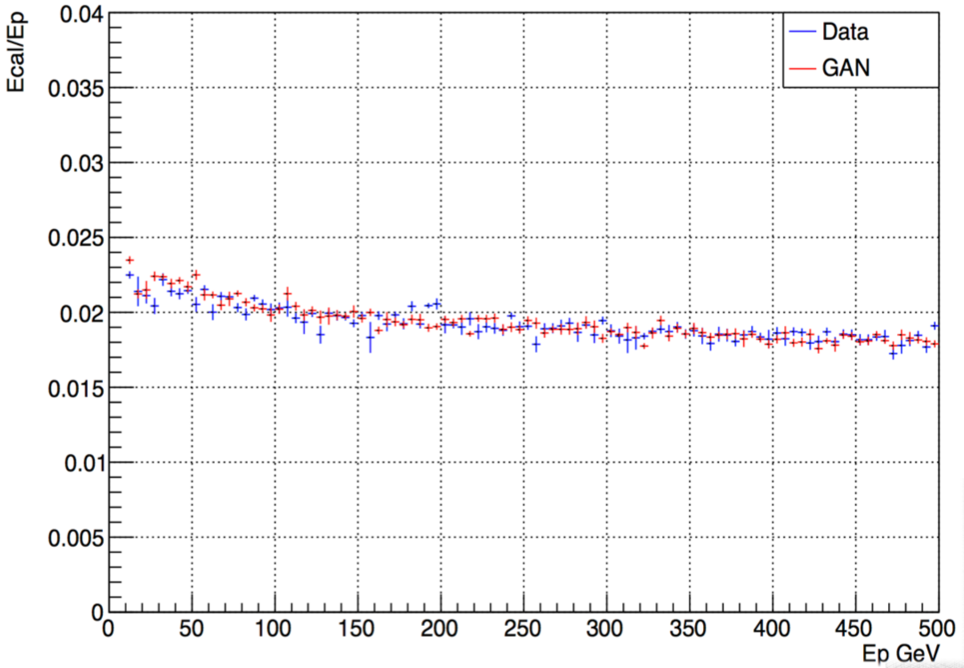
Detective says it is fake and gives feedback

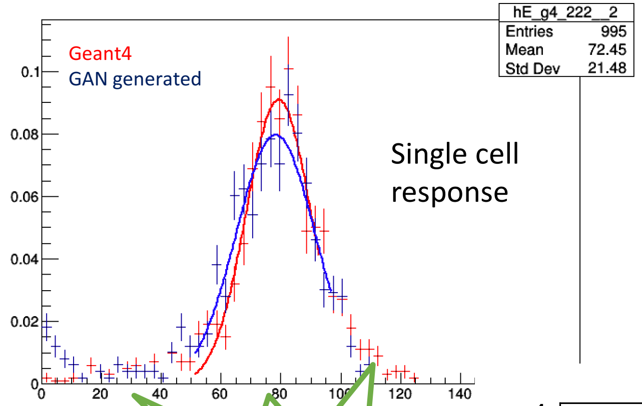
Counterfeiter makes new Monalisa based on feedback

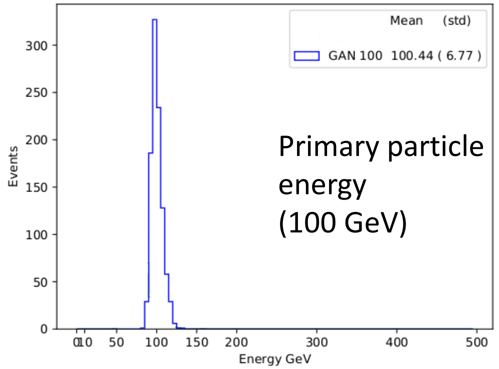
Iterate until detective is fooled

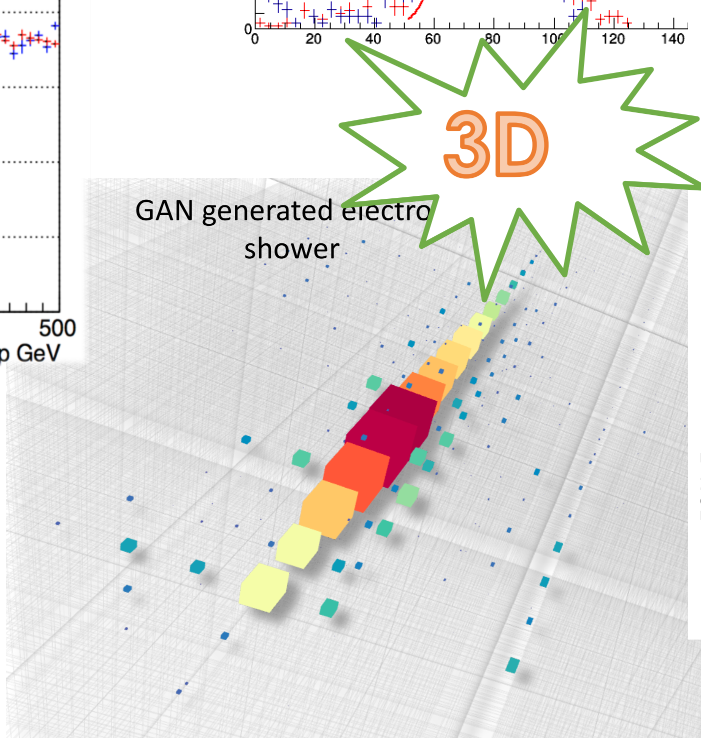
Results validation: Accurate!

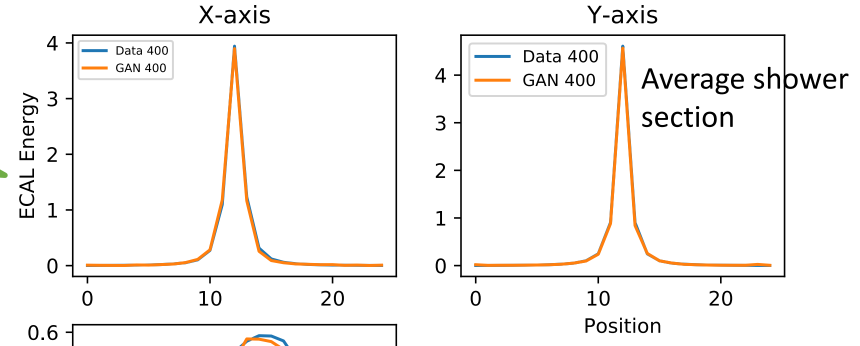
Comparison to Geant4 data

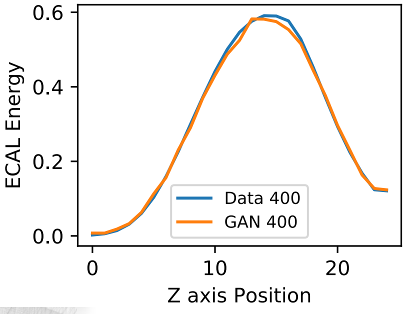


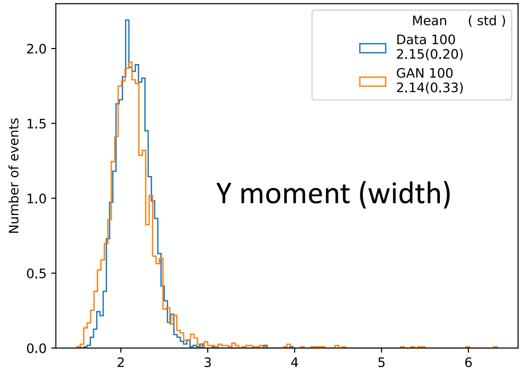












CERN openlab

Computing resources: Fast!

Using a trained model is very fast

Single node performance. Keras + TF 1.8

Inference:

Classical Monte Carlo requires 17 s/shower

3DGAN takes 7 ms/shower on Xeon

speedup factor > $2 \cdot 10^3$

0.04 ms/shower on NVIDIA P100

speedup factor > $4 \cdot 10^5$!!!

Training:

45 min/epoch on NVIDIA P100

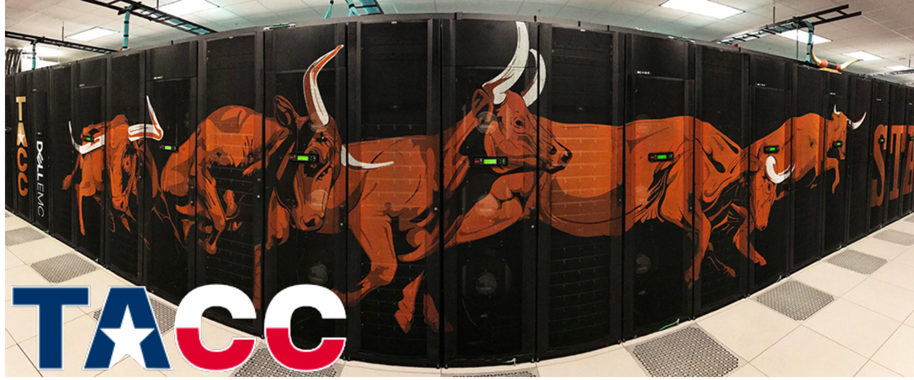
Only 200K G4 events are needed for training

Time to create an electron shower		
Method	Machine	Time/Shower (msec)
MC Simulation (geant4)	Intel Xeon Platinum 8180	17000
3D GAN (batch size 128)	**Intel Xeon Platinum 8180**	**7**
3D GAN (batch size 128)	**NVIDIA P100**	**0.04**

CERN openlab

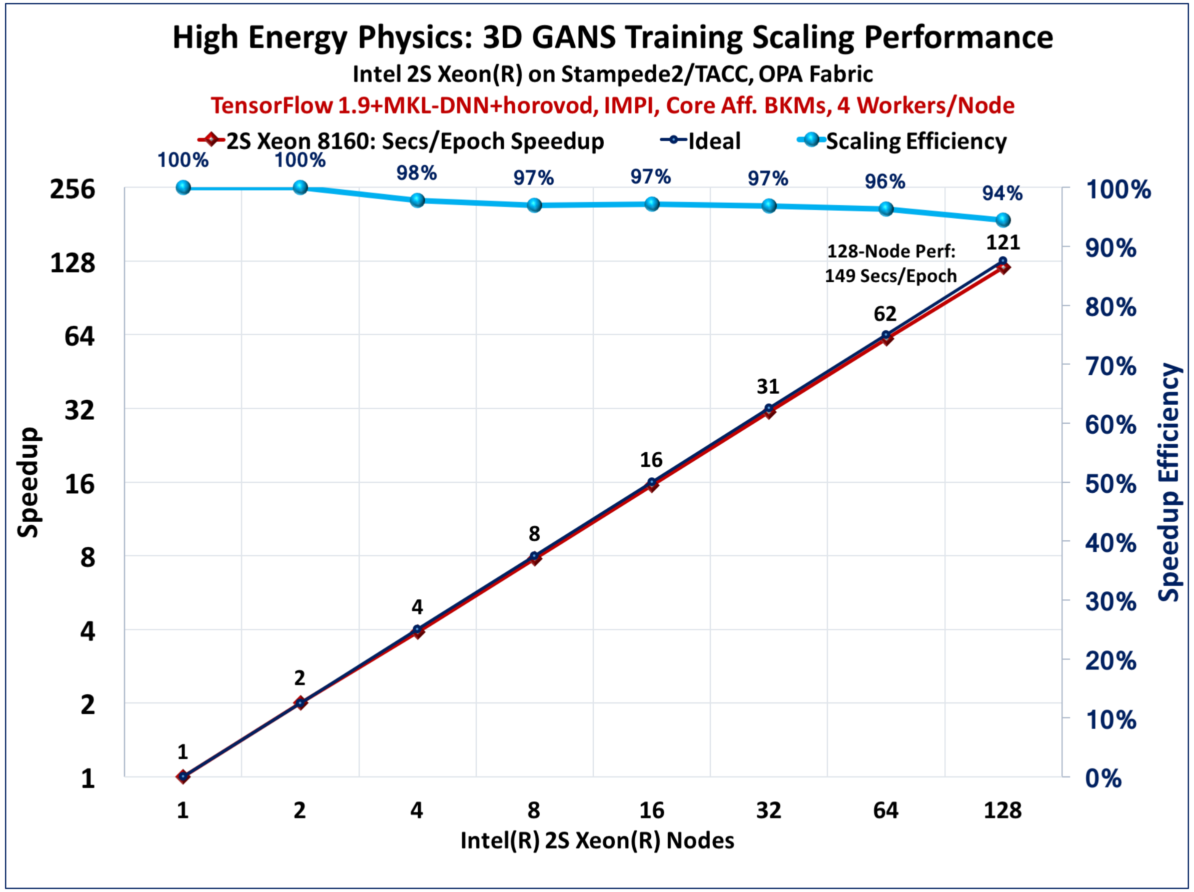
HPC friendly!

Distributed training using data parallelism

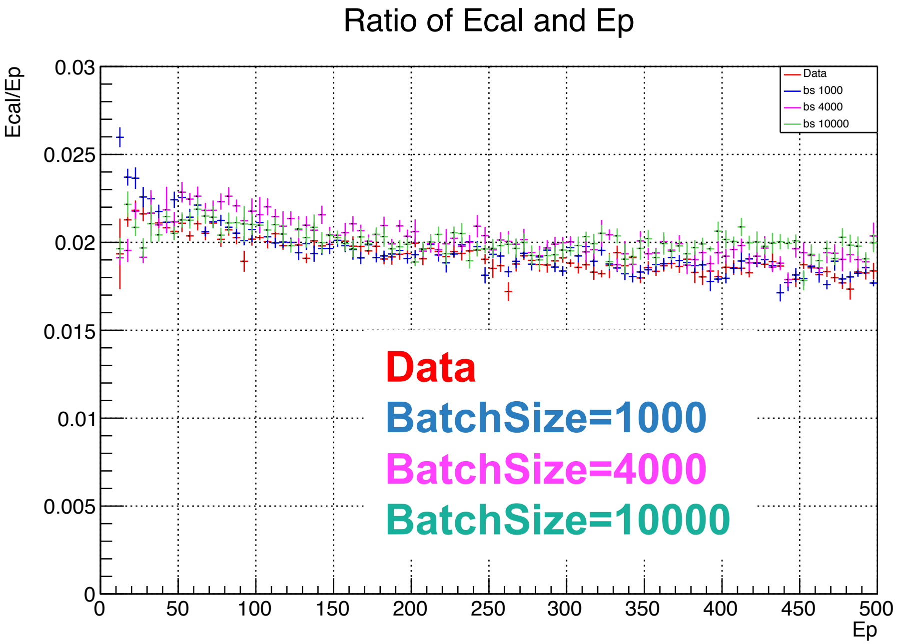


Run on TACC Stampede2 cluster:

- Dual socket Intel Xeon 8160
- 2x 24 cores per node, 192 GB RAM
- Intel® Omni-Path Architecture

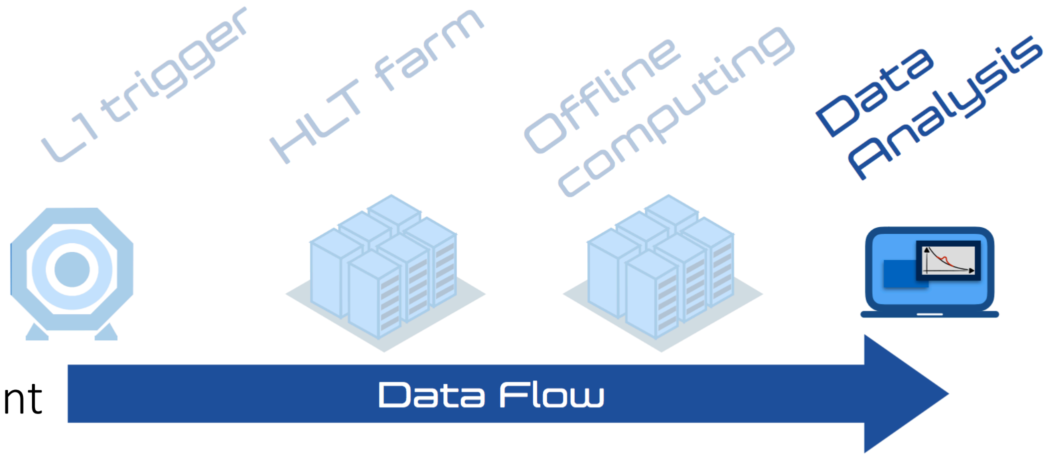


Study performance degradation



Analysis Workflow

- “Small” groups, individually implemented code
- Analysis dependent:
  - Subsets of the total data volume
  - Slimming (filter specific collisions) & Skimming (reduce content per collision)
  - Calculation of new quantities
- Complicated multi-step workflow because dataset is too large for interactive analysis
- Rerun framework code (e.g. with non-default parameters)
  - correct problems/ mistakes
- Can take weeks using GRID resources and local batch systems
  - Experiments started to centralize first step
- Not all time spent is actual CPU, a lot of time is bookkeeping, resubmission of failed jobs, etc.

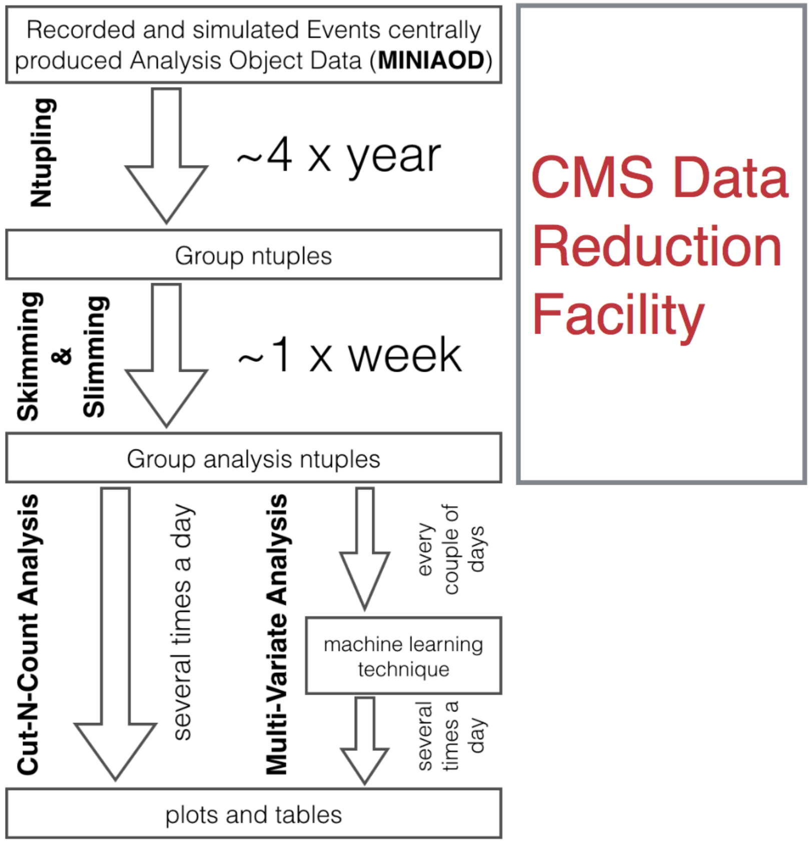


- Up to ~ 500 Hz In / 100-1000 events out
- <30 KB per event
- Processing time irrelevant
- User-written code + centrally produced selection algorithms

Currently based on ROOT, the community’s statistics, plotting and I/O toolkit

ROOT Data Analysis Framework

Ex: CMS Data Reduction facility

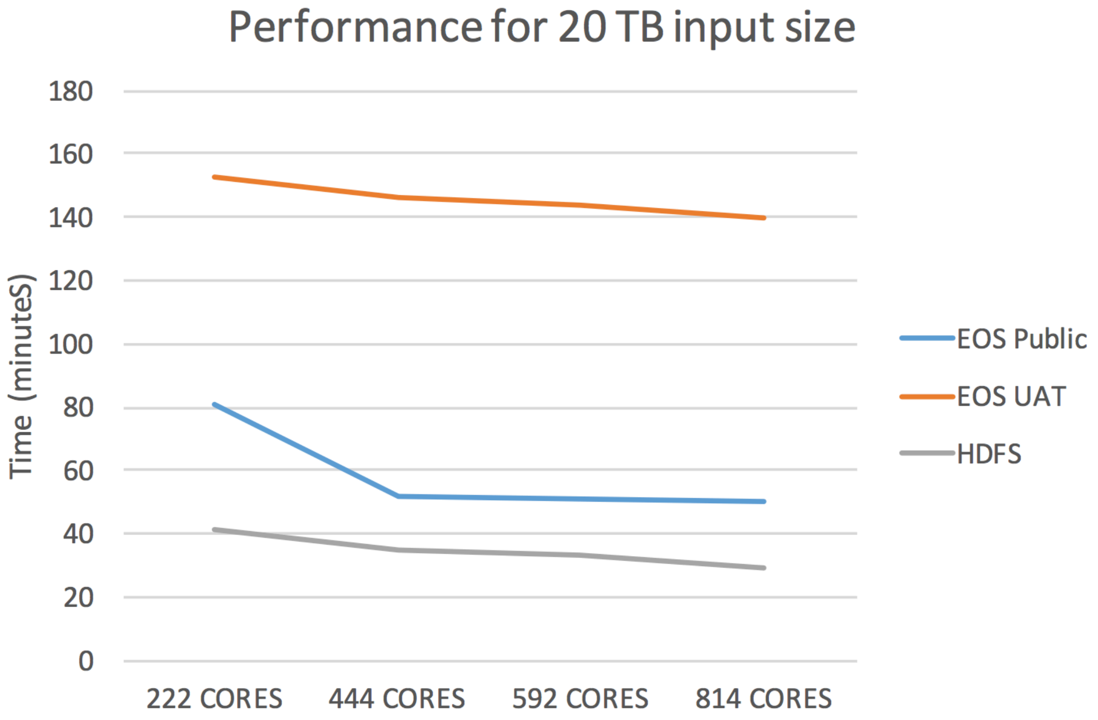


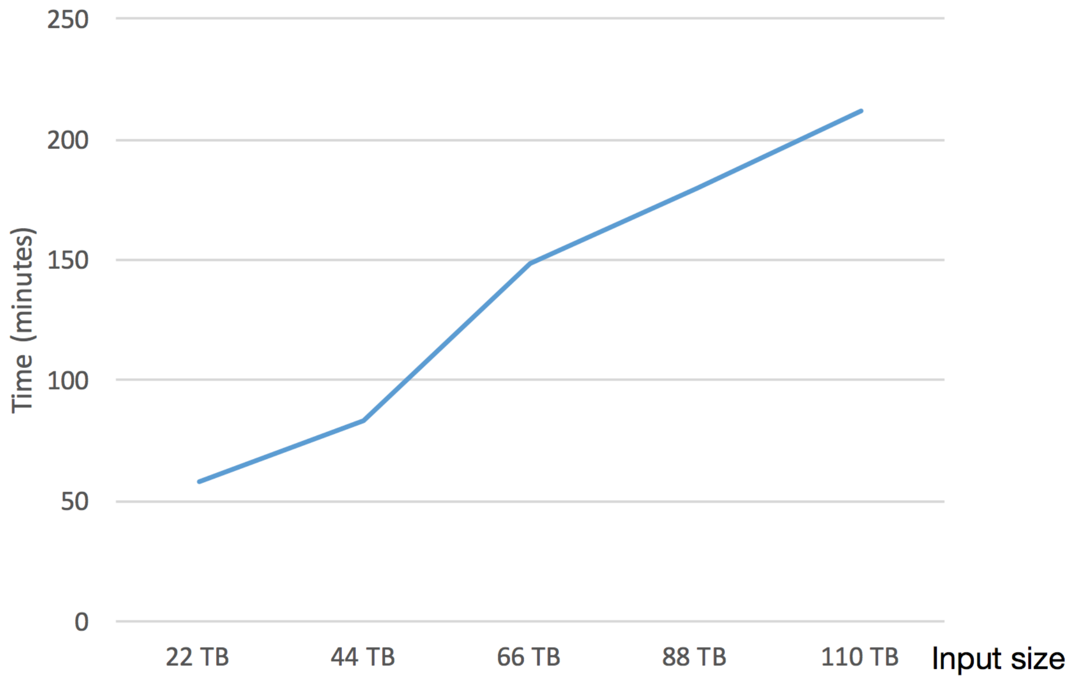
- CERN Openlab project with Intel (2 years)
- Demonstration facility optimized to read through petabyte sized storage volumes
  - Produce sample of reduced data based on potentially complicated user queries
  - Time scale of hours and not weeks as it currently requires.
- If successful, this type of facility could be a big shift in how effort and time is used in physics analysis
  - Same infrastructure and techniques should be applicable to many sciences

Scalability Tests

Spark analytix cluster @CERN, shared infrastructure with ~1300 cores, 7 TB RAM

HDFS and Remote EOS Public/UAT storage





M, Cremonesi, CHEP2018 https://indico.cern.ch/event/587955/contributions/2937521/

Further R&D

- Parallelisation of analysis frameworks (see ROOT contributions at CHEP2018)
  - Memory and I/O Optimisation (data format and memory structures, TDataFrame)
  - Improved features
  - User friendly (-ier…) APIs
  - Containerised analyses
- Exploration of “other” tools
  - HYPSTER : python-based data analysis framework (ML/DL integration)
  - Panda DataFrames
  - HPC-friendly: HYDRA, columnar data platforms (Numpy-like)
- Data Analytics platforms
- Exploration of optimised data-format
  - compression

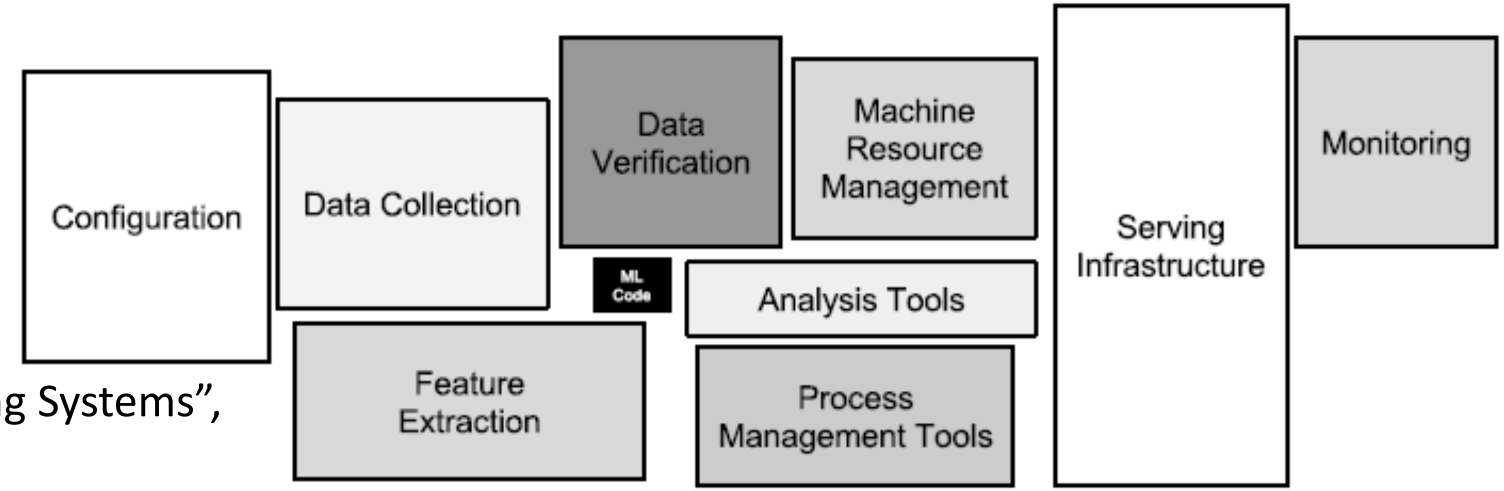
Data Analytics at scale – Challenges

When you cannot fit your workload in a desktop

- Data analysis and ML algorithms over large data sets
- Deploy on distributed systems (containers)

Need specialized components for:

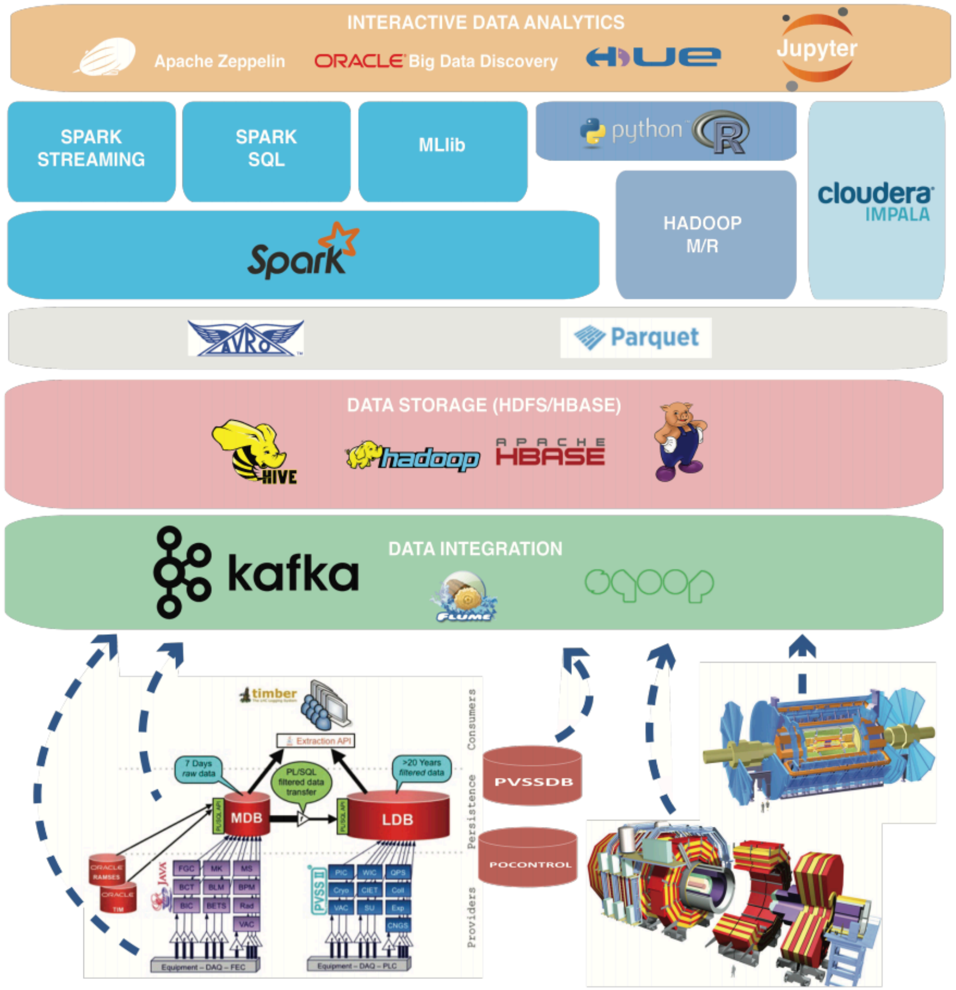
- Data ingestion tools and file systems
- Cluster storage and processing engines
- ML tools that work at scale



From "Hidden Technical Debt in Machine Learning Systems", D. Sculley at al. (Google), paper at NIPS 2015

Hadoop and Big Data Analytics at CERN

- New scalable data services being tested
  - Scalable databases
  - Hadoop ecosystem
  - Time Series databases
  - Interactive data analytics (Jupyter..)
- Activities and objectives
  - Support of Hadoop Components
  - Further value of Analytics solutions
  - Define scalable platform evolution
- Hadoop Production Service



Machine Learning with Spark

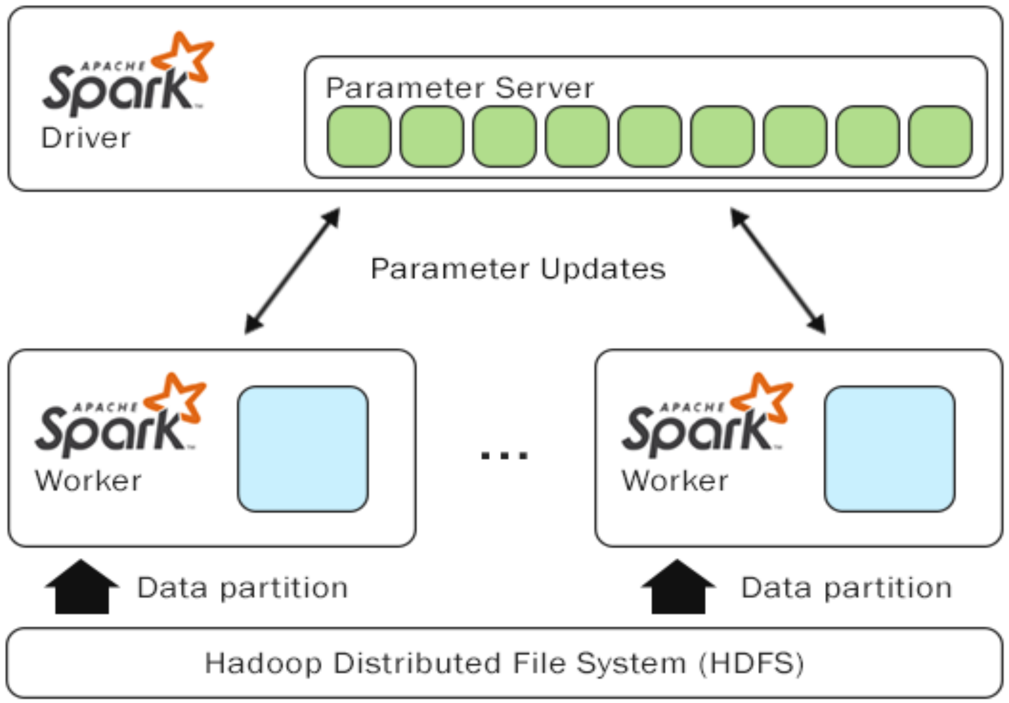
Spark has tools for machine learning at scale

Spark library MLlib

- Frameworks and tools for distributed deep learning with Spark available on open source:

TensorFrame, BigDL, TensorFlowonSpark, DL4j, ..

@CERN: Developed an interface to Keras



https://github.com/cerndb/dist-keras

Main developer: Joeri Hermans (CERN)

Containers

Containers can make analysis systems more useful and easily shareable.

Applications become self-contained and work on any number of platforms

ML applications exposed as services

Leverage external and distributed data access layers

S. Trigazis, CERN openlab openday 2017

Leverage Containers

in High Energy Physics and elsewhere

- Improve agility in deploying and rolling new software releases
- Isolation with kernel control groups and namespaces
- Faster than virtual machine, shared kernel
  - Non virtualization overhead
- Ease of use, microservices, container images, declarative deployments
- Integrate containers in the CERN cloud
  - Shared identity, networking integration, storage access, …
- Immutable Infrastructure

kubernetes

docker SWARM

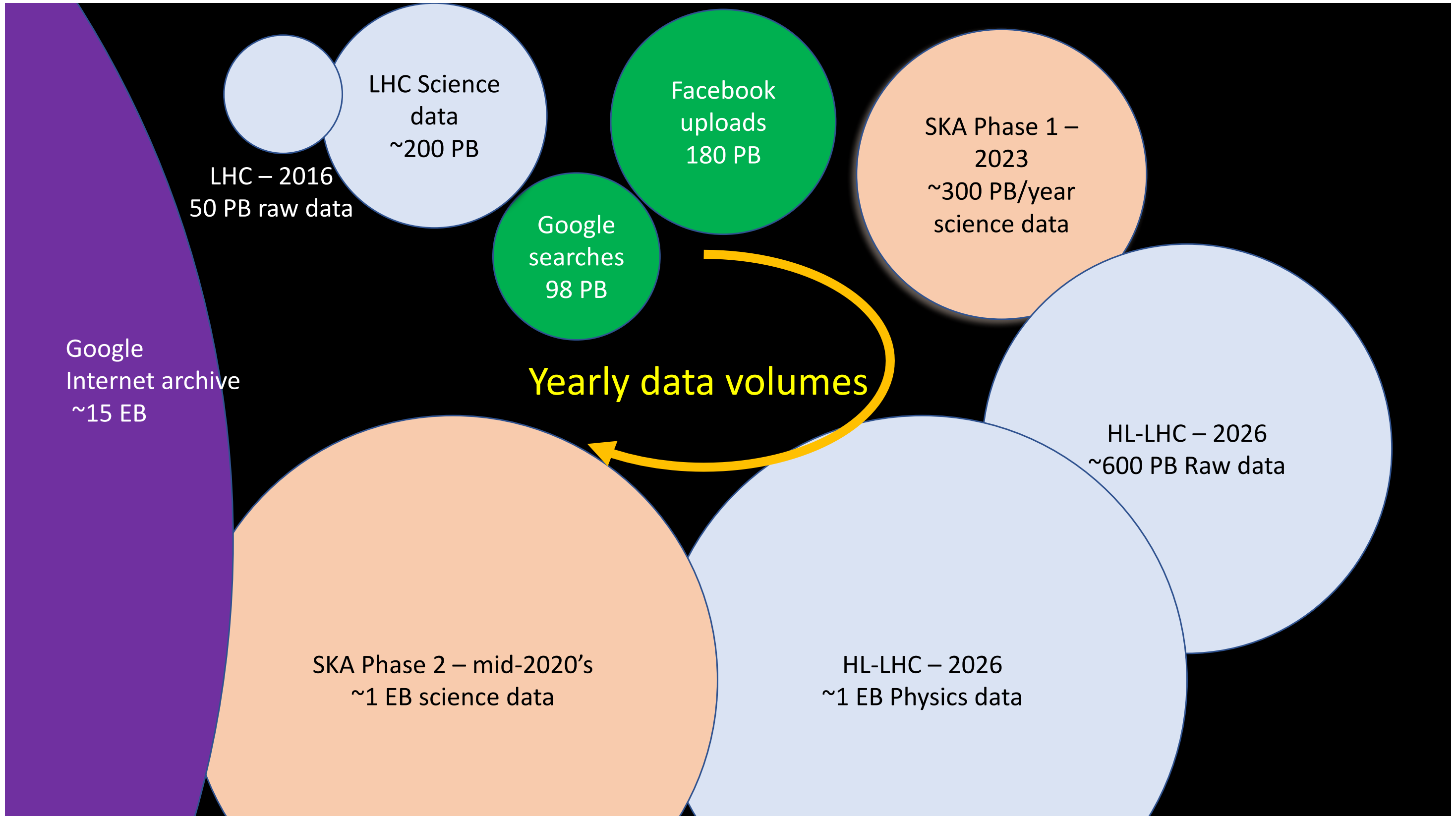
DC/OS

CERN openlab

LHC vs Big Data?

In the past CERN was at the forefront of the Big Data challenge

- Not so simple anymore
  - Growing number of actors
- More sciences are becoming “Big Data” sciences
  - Collaboration and community “building” is essential



https://indico.cern.ch/event/656491/contributions/2940766/attachments/1632534/2603674/summary_fcchhdet.pdf

Summary

The challenge is evident

- Will need significant R&D to
  - consolidate the models being investigated
  - Understand concrete implications as well on cost understanding and modelling
  - Exploring new techniques (ML), service delivery models and integrating them in the models will make decisive contributions to the overall cost and efficiency

- Data deluge is not a exclusive to HEP
  - Other sciences with similar challenges
  - Tech industry with exponentially data growth
  - Need to create synergies for common benefits

Thanks!

Questions?

3D convolutional GAN

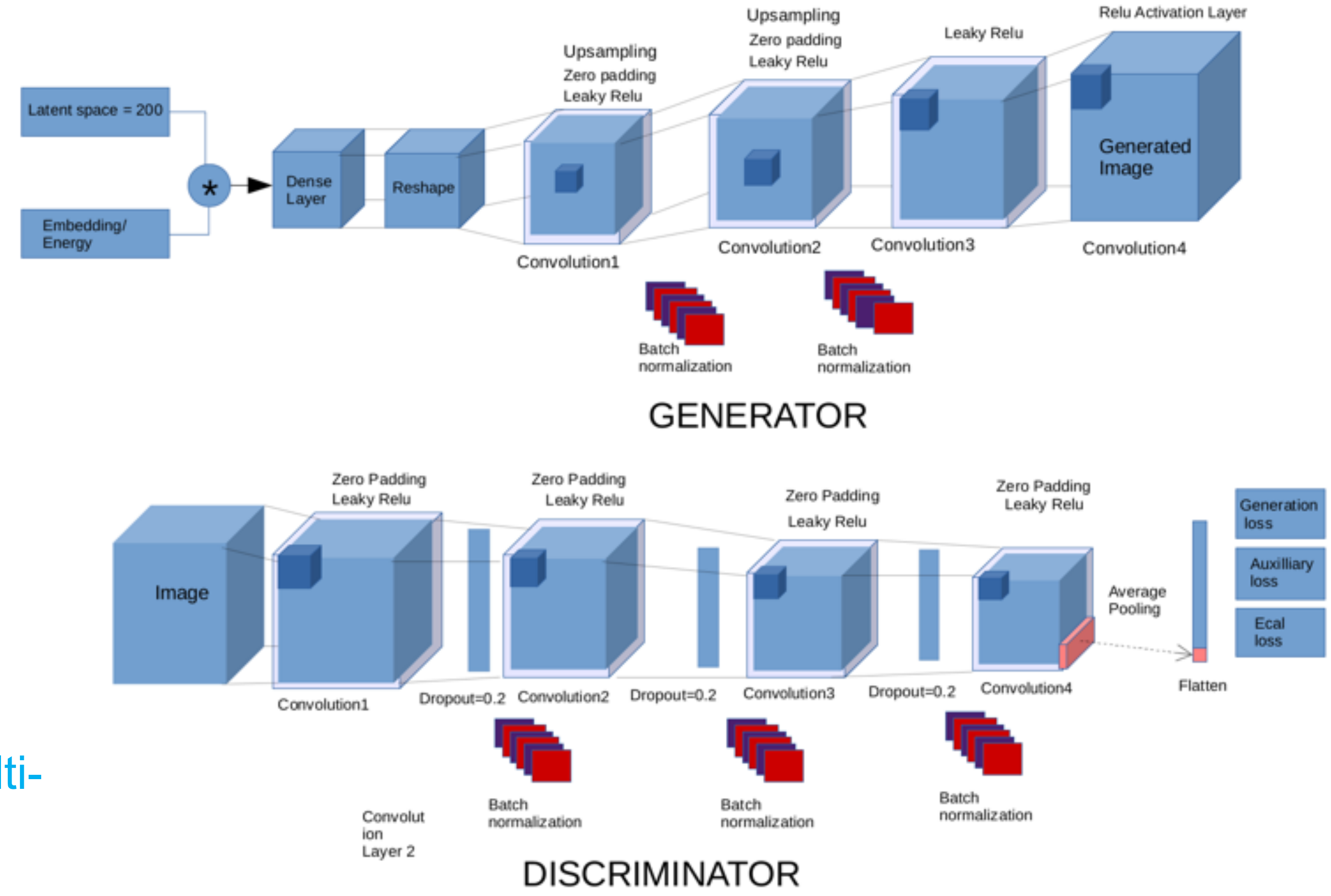
Similar discriminator and generator models

- 3d convolutions (keep X,Y symmetry)

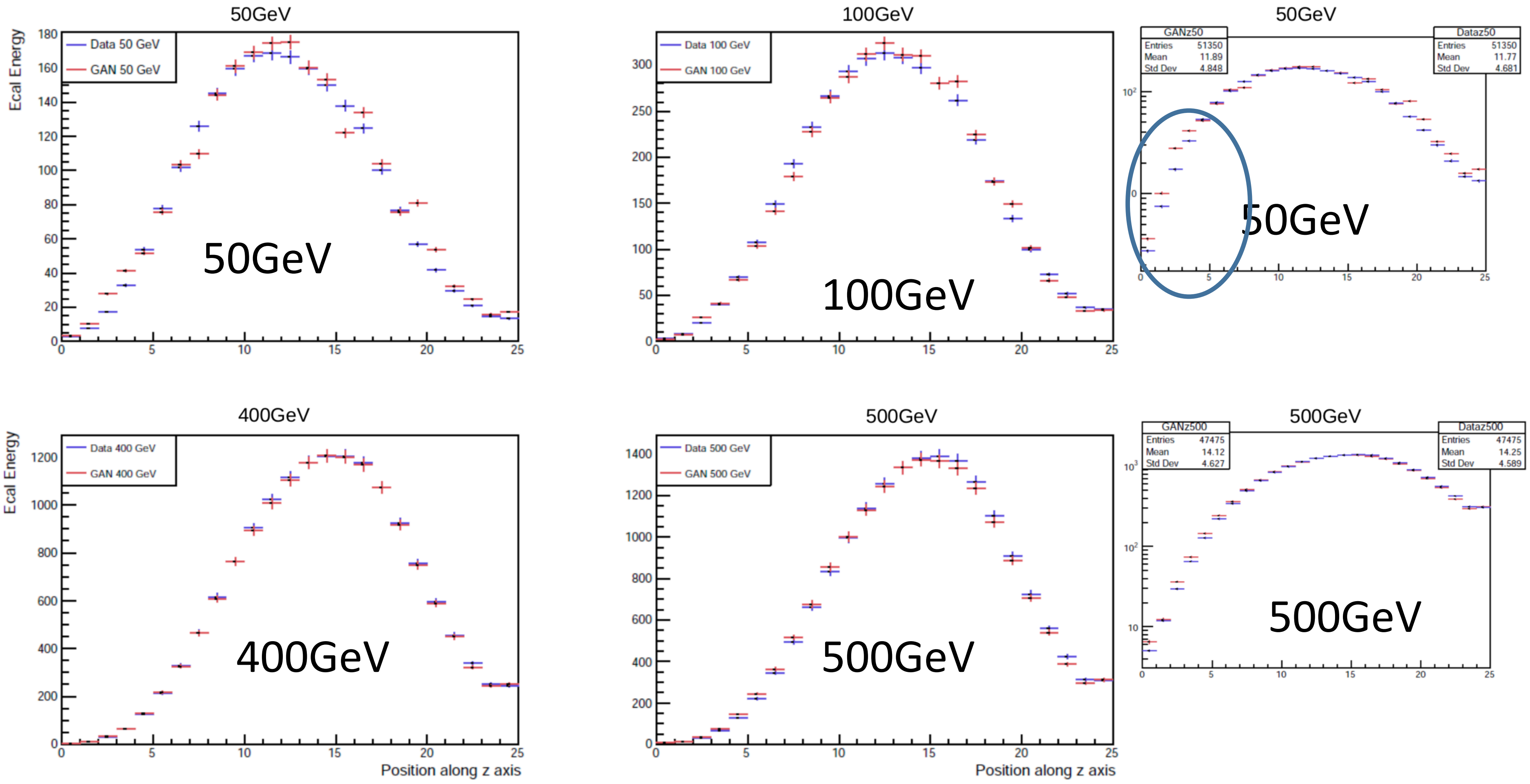
Condition training on several input variables

Auxiliary regression tasks assigned to the discriminator: cross check

Easily generalisable to multi-class approach (or multi-discriminator approach)

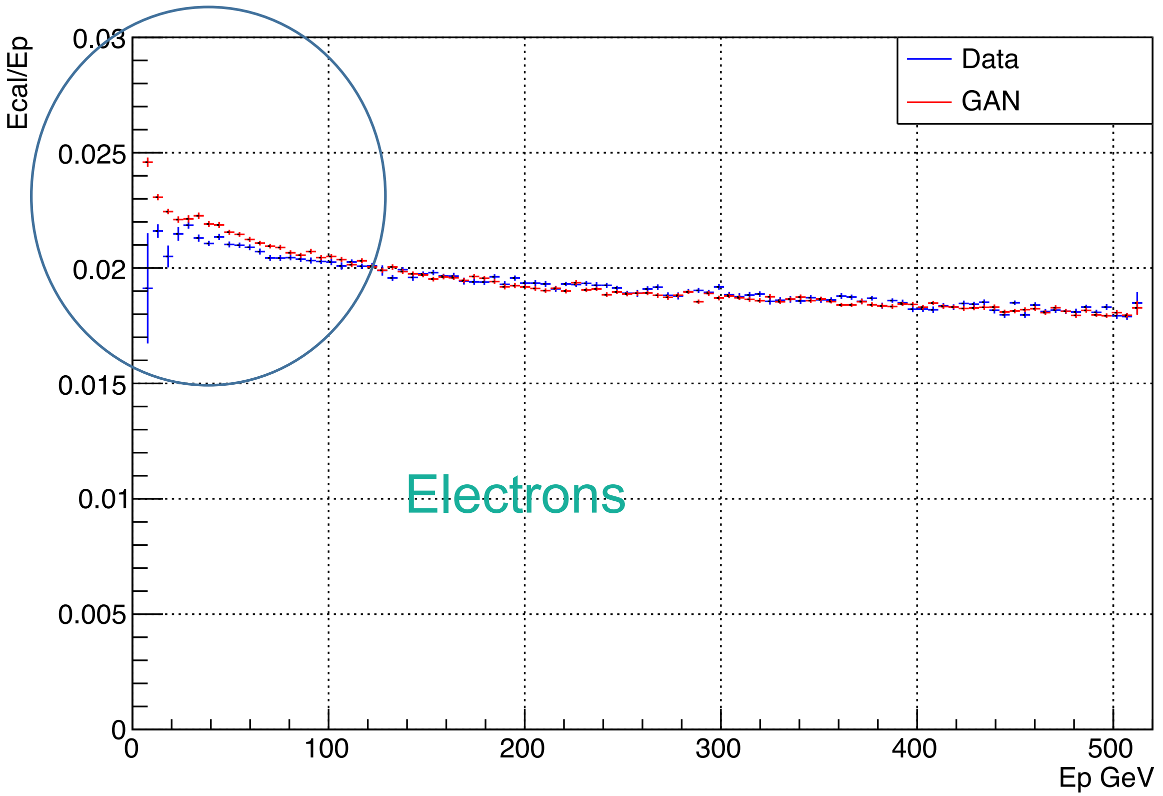


Electrons shower shapes

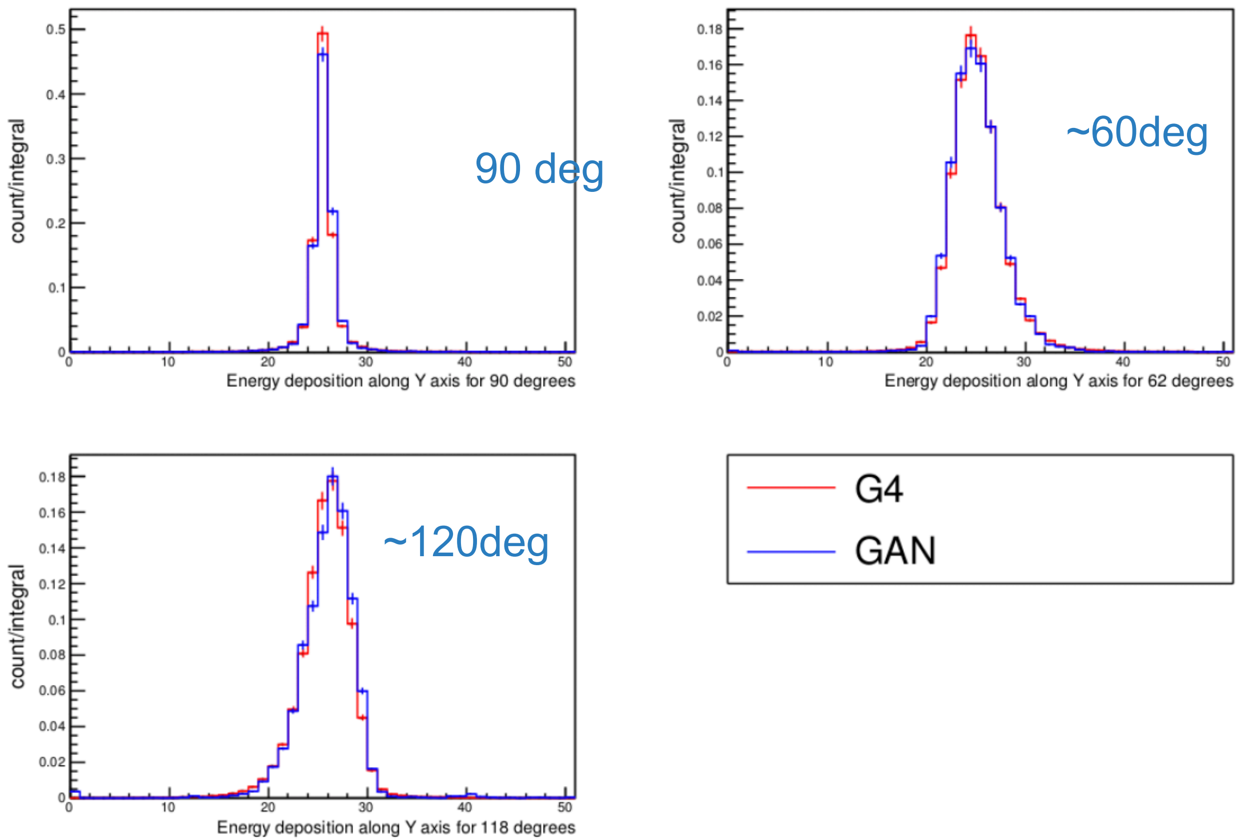


CERN openlab

Calorimeter sampling fraction



Incident angle



Media hierarchy

We still use tape! Why?

- $/PB (TCO incl. power)
- separate physical copy with high “destruction’

We stopped trying “automatic” HSM (Hierarchical Storage Management) for large experiment users

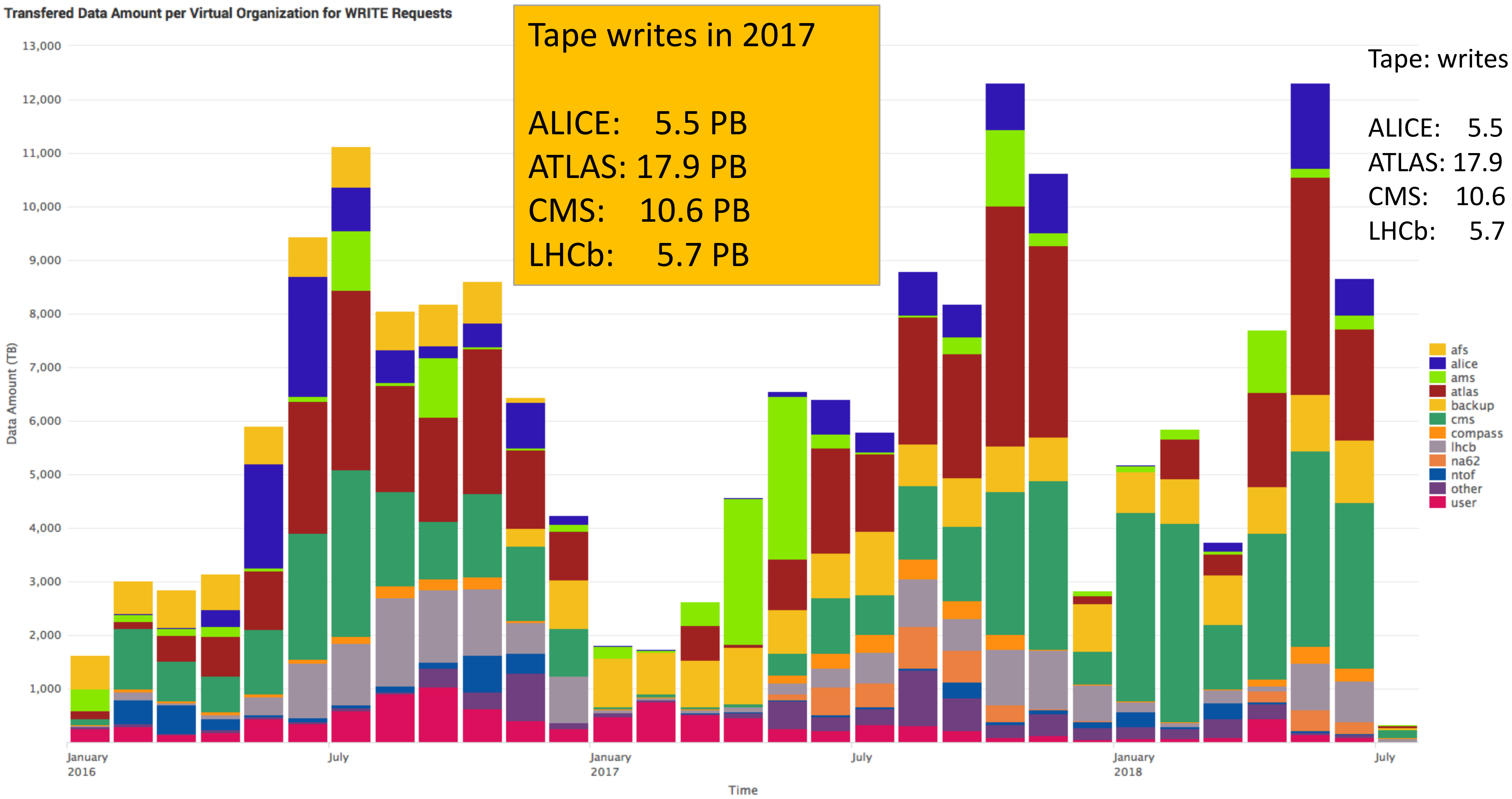
- file based HSM interface did not allow t

Disk content is stable (until the experime active data)

- thousands of job streams at relatively lo

Tape access enabled only for a few production activities

Scale Examples: Tape Archive



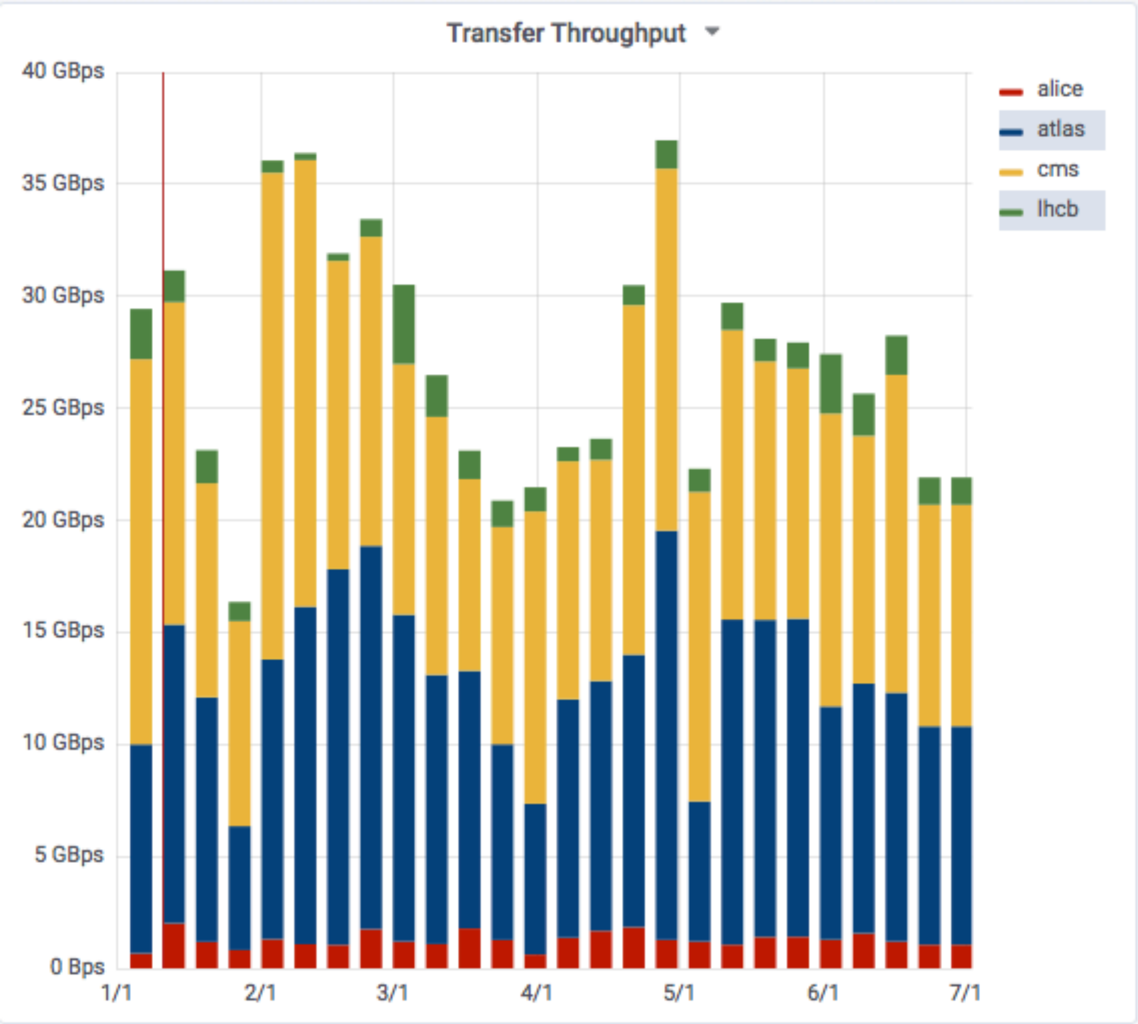
Tape writes in 2017

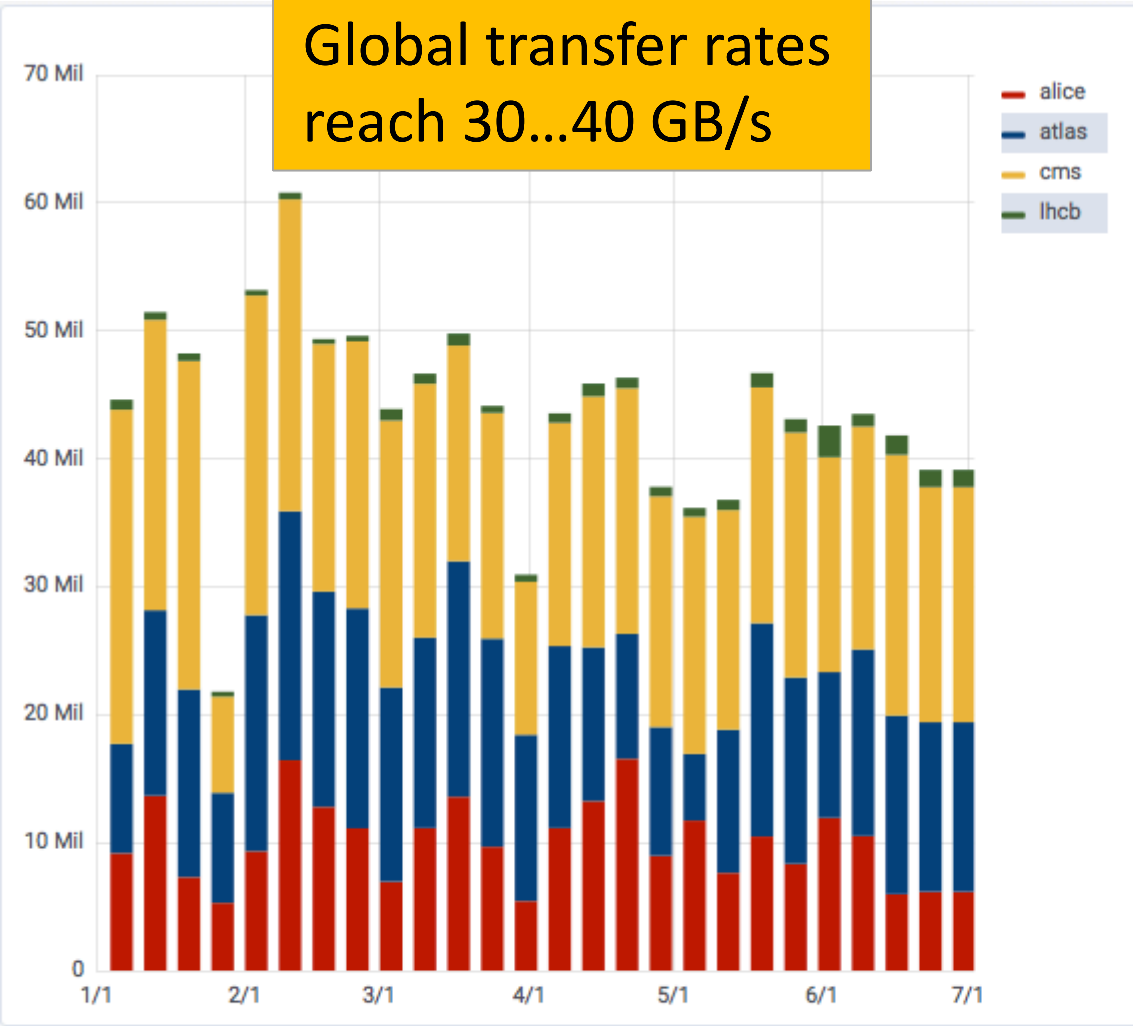
ALICE: 5.5 PB
ATLAS: 17.9 PB
CMS: 10.6 PB
LHCb: 5.7 PB

Tape: writes in 2017

ALICE: 5.5 PB
ATLAS: 17.9 PB
CMS: 10.6 PB
LHCb: 5.7 PB

Scale Example: Data Transfer





WLCG:
Global transfer rates reach 30…40 GB/s

Speeding up simulation

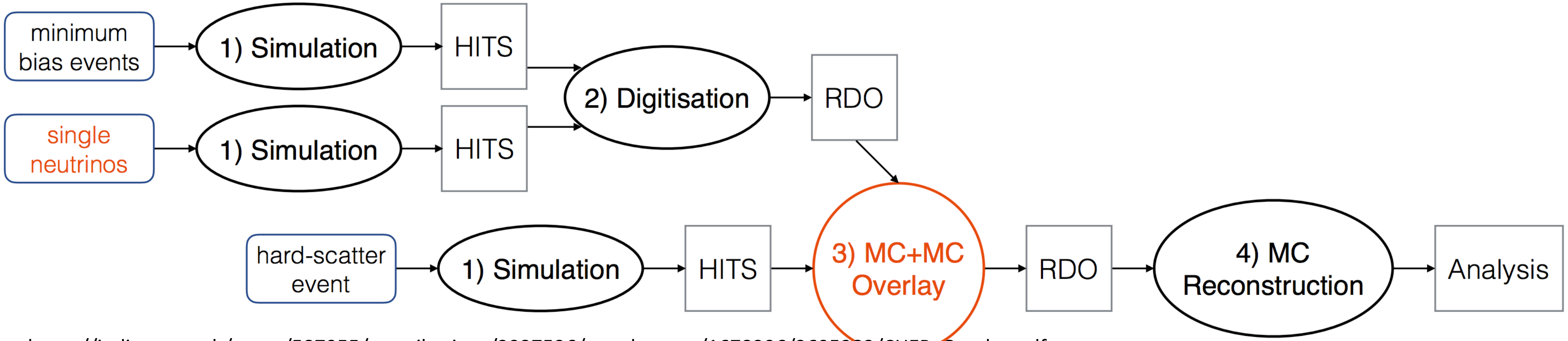
MC-MC overlay

Simulate a hard-scatter G4 event with usual configuration

Pre-mixing of pile-up events: Standard pile-up simulation of zero-hard-scatter events (e.g. single neutrinos). In the future this step should only require minimum bias events.

Digitise simulated hard-scatter event and overlay it on pre-mixed pile-up digits.

Re-use is the key!



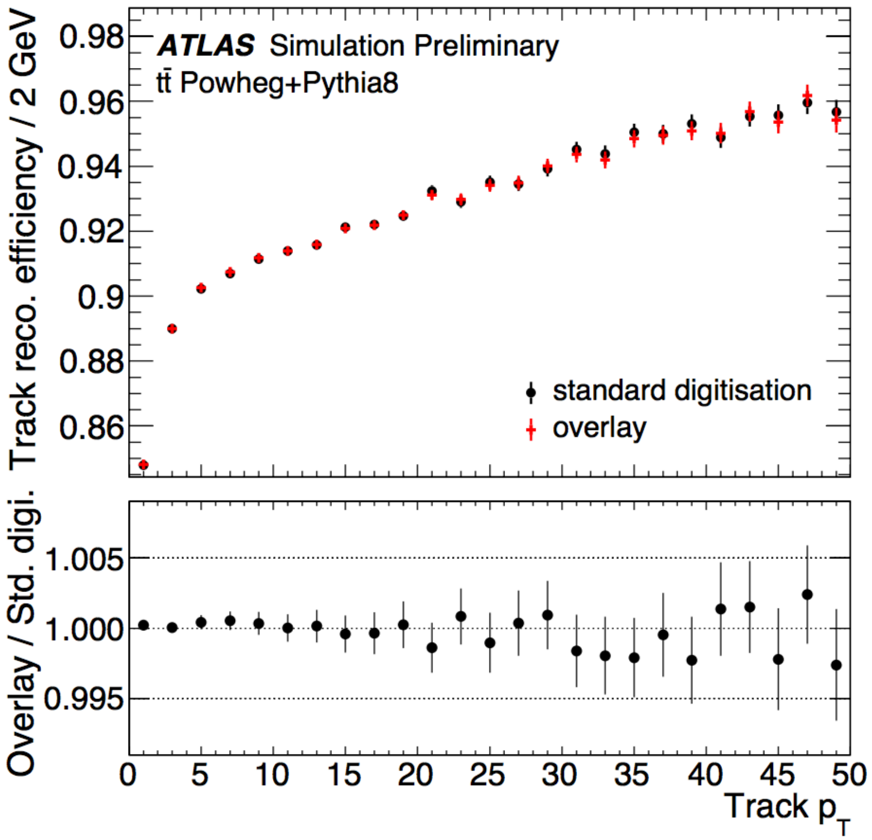
https://indico.cern.ch/event/587955/contributions/2937526/attachments/1678296/2695383/CHEP_Overlay.pdf

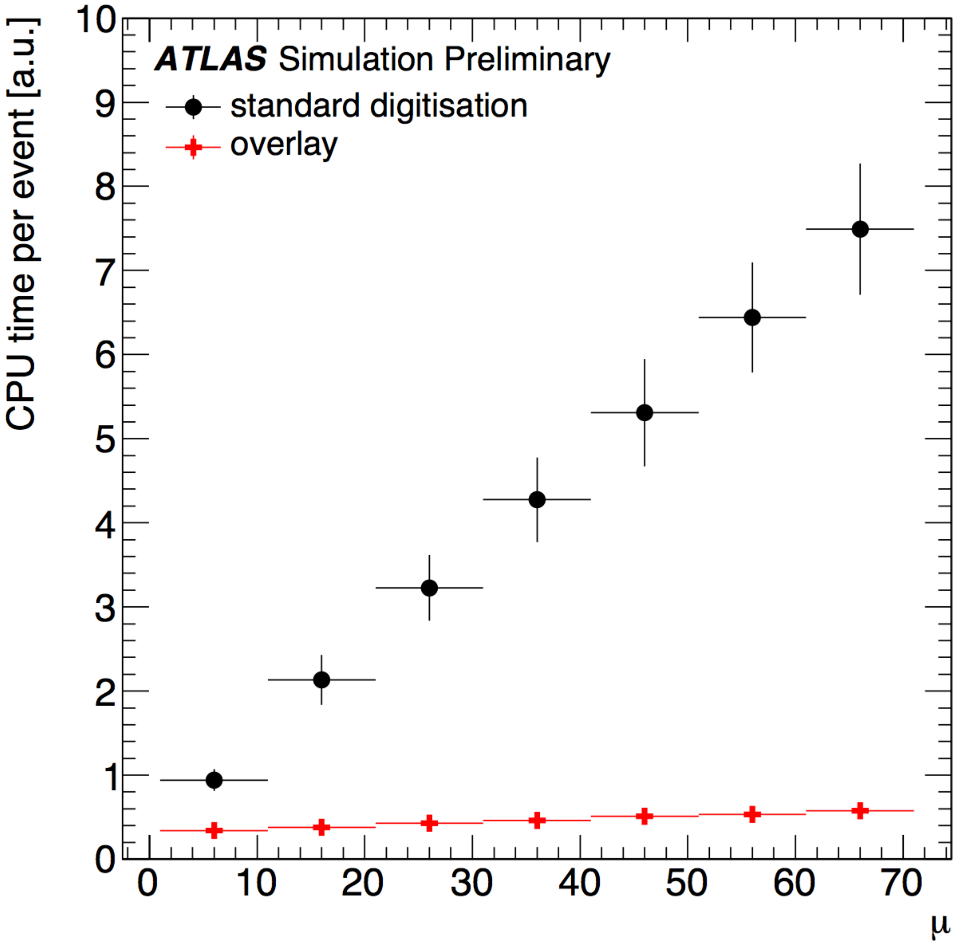
T. Novak, ATLAS, CHEP 2018

Speeding up simulation

MC-MC overlay

Close to nominal physics performance

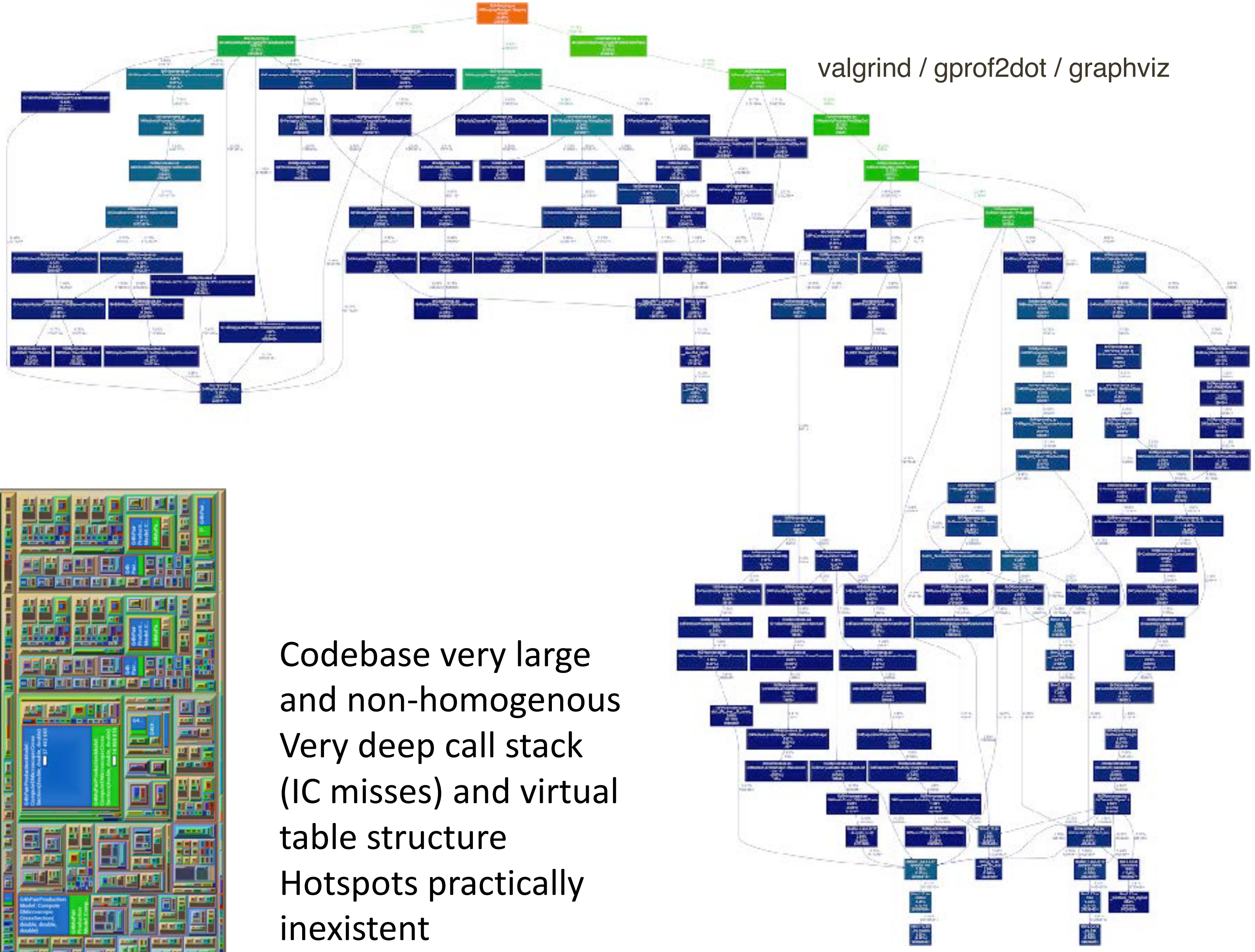




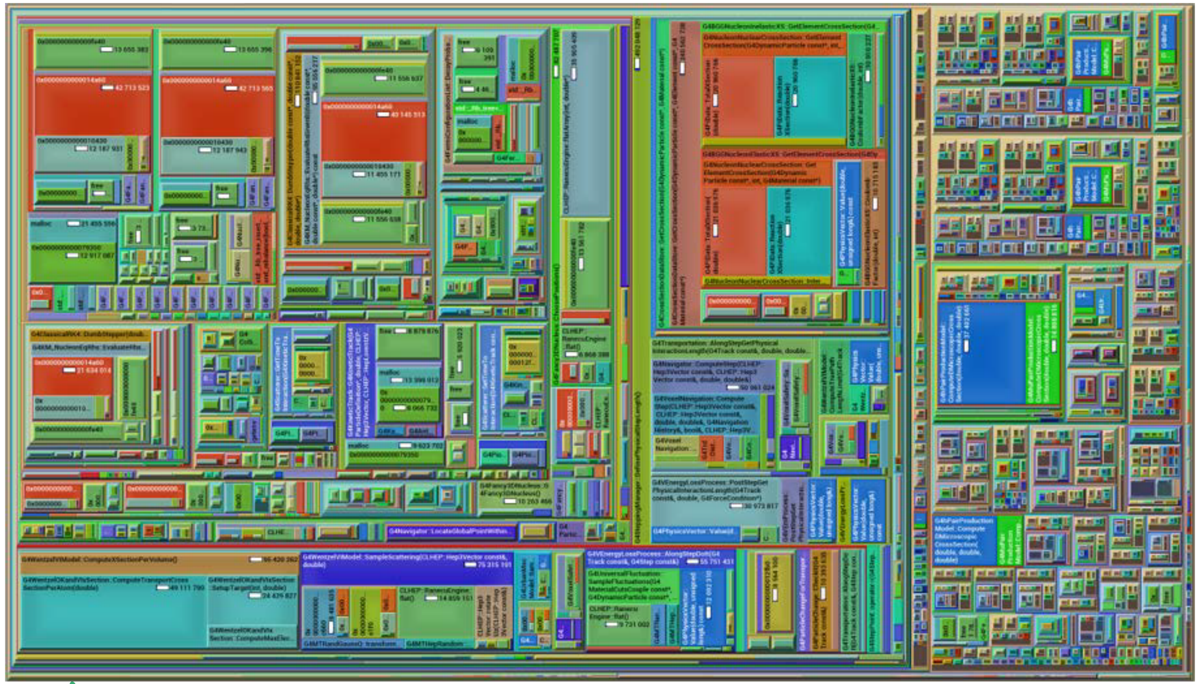
Data-MC overlay is also been developed!
See Haas, ATLAS, CHEP 2016

A simple G4 example

valgrind / gprof2dot / graphviz



Valgrind/kCachegrind



Codebase very large and non-homogenous
Very deep call stack (IC misses) and virtual table structure
Hotspots practically inexistent

http://geant4.web.cern.ch/geant4/UserDocumentation/Doxygen/examples_doc/html/ExampleB1.html

ALFA /Fair MQ