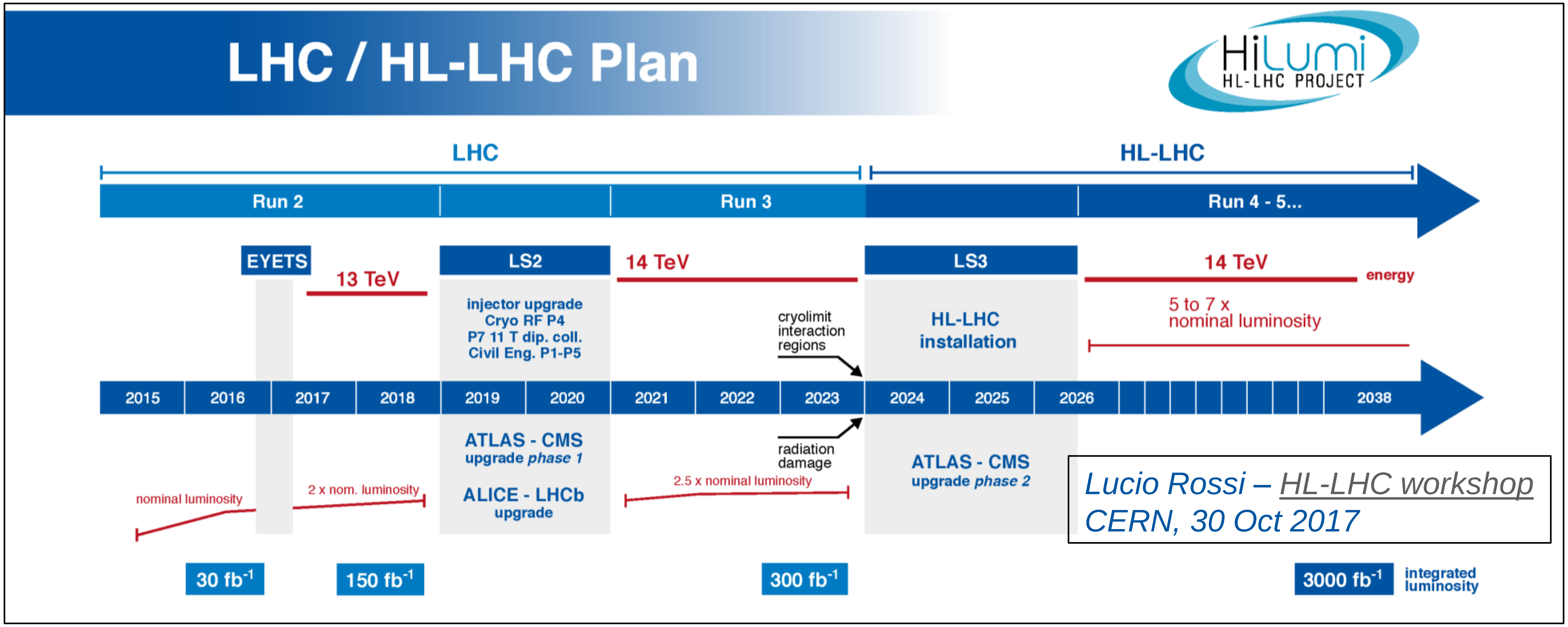
# HEP Software Foundation Community White Paper

Andrea Valassi (IT-DI-LCG)

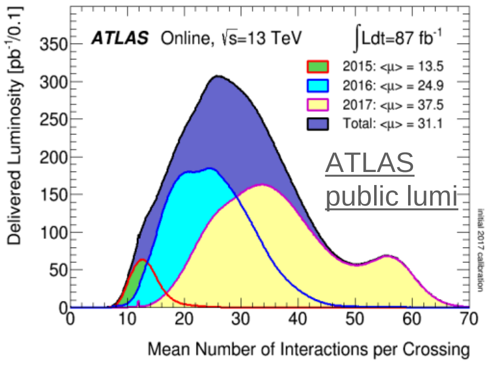
CERN IT Technical Forum – 3rd November 2017

*Disclaimer: credits to the CWP and WG authors;*
*misunderstandings, comments and opinions are my own!*
*(Apologies for not making it clearer what is verbatim in the CWP and what is my own.)*

# Exciting, but challenging, times ahead

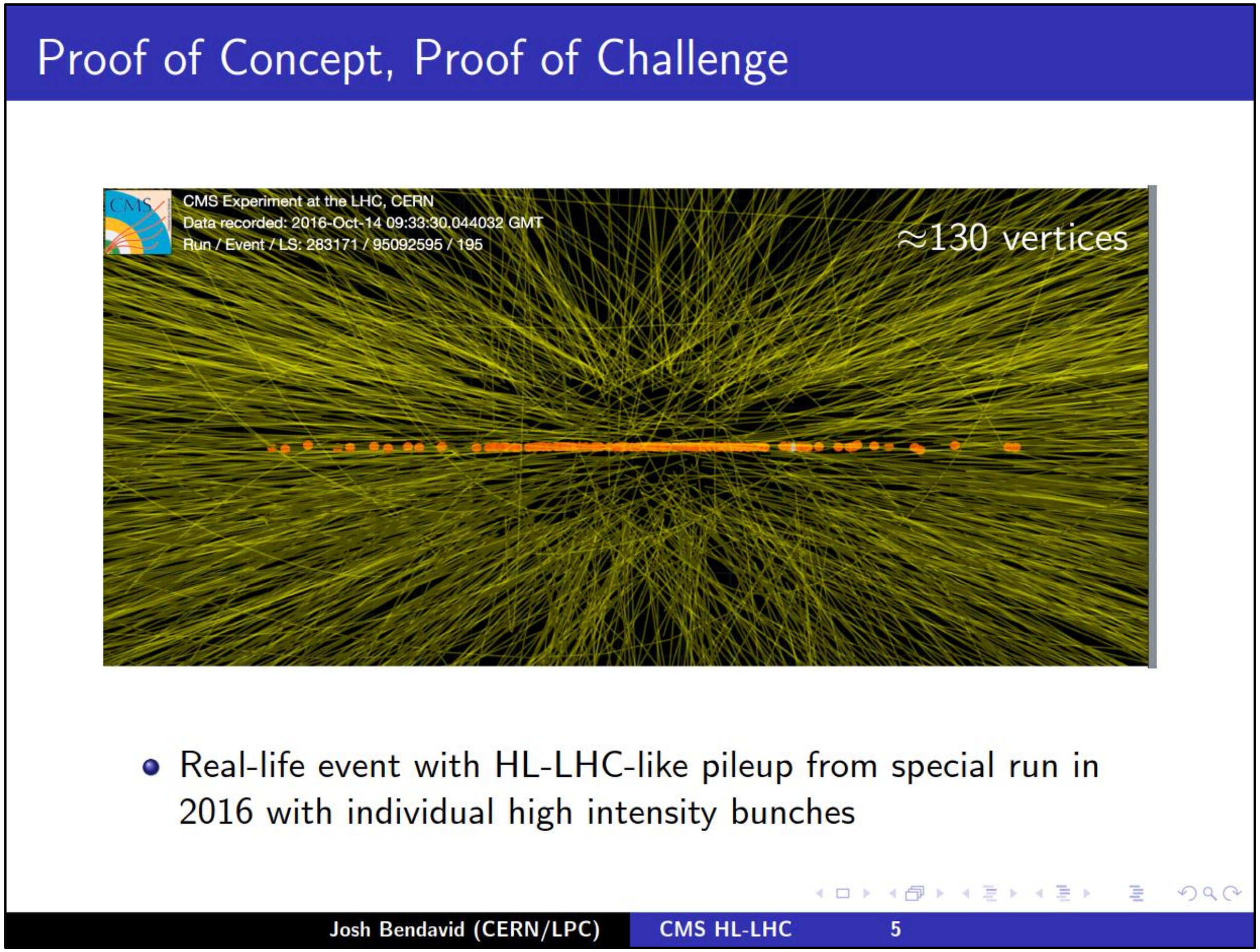



- Run3 (2021): major LHCb/ALICE upgrades
- Run4 (2026): major ATLAS/CMS upgrades, **high-luminosity LHC**
  - **more data** (higher luminosity) – and correspondingly more MC to generate/simulate
  - **more complex events** (higher pileup)




Pileup (2017):
μ ~ 30 to 60

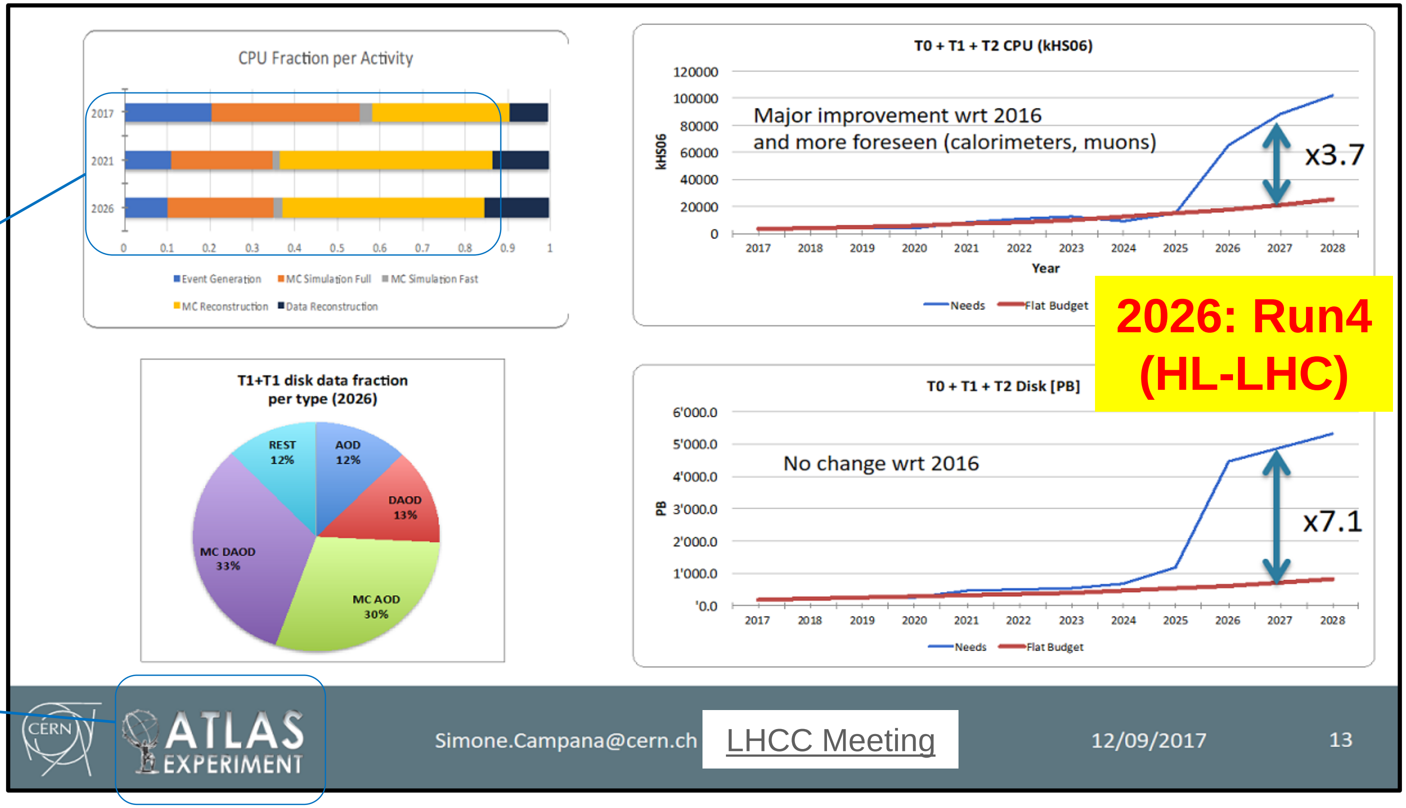
Pileup (2026):
μ ~ 130 to 200




*J. Bendavid – HL-LHC workshop – CERN, 30 Oct 2017*

# Computing resource challenges

(80-90% of CPU for MC, not data)




CPU

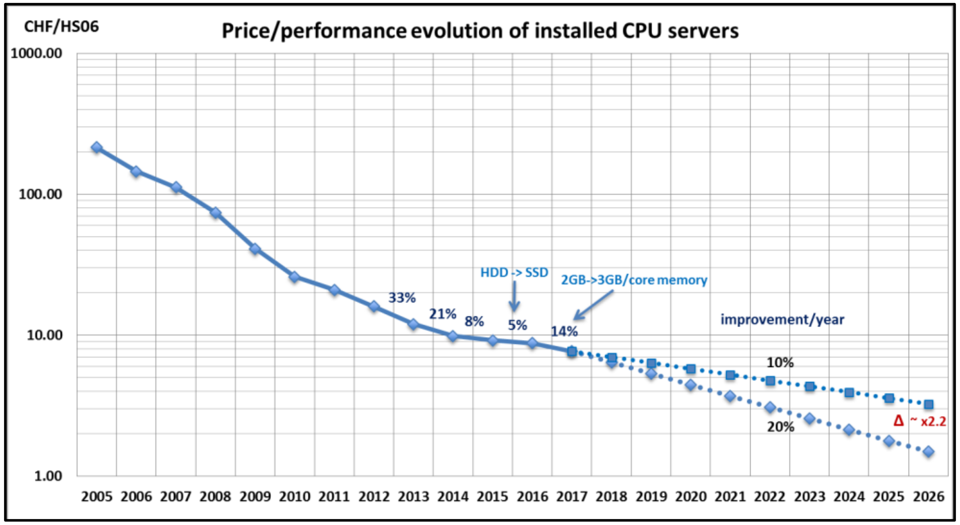
DISK

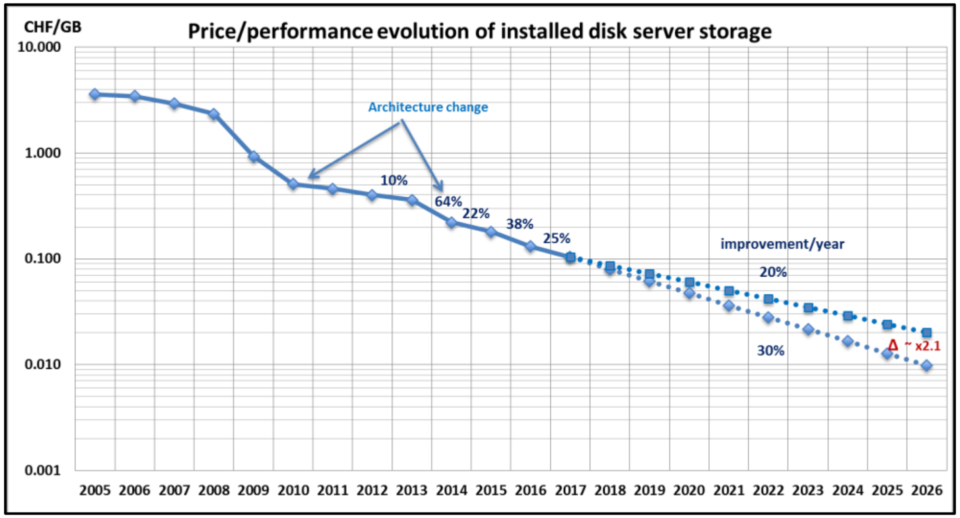
(similar expectations for CMS)

Shortfall of resources for HL-LHC: factor ~7 for disk, factor ~4 for CPU

***"The amount of data that experiments can collect and process in the future will be limited by affordable software and computing, not by physics"***

# Technology and market trends




*B. Panzer – Technology tracking twiki – May 2017*

- Improvement per year: ~10-20% for CPU and ~20-30% for disk
  - already included in ATLAS and CMS projections at the LHCC
  - Moore's law and Kryder's law are slowing down
- More diverse landscape, too
  - multi- and many-core processors
  - wide vector registries
  - GPGPUs, FPGAs, ARM, HPCs...
  - memory bandwidth relatively low
  - *many programming models*
- *Technology alone will not solve the HL-LHC resource challenge*
- *Need a "software upgrade"!*

# Software (and human) challenges

- Current software of the 4 LHC experiments
  - over 12M lines of code written over the last 15 years
  - mostly written for a single architecture (x86)
  - mostly written with serial processing in mind
  - mostly experiment specific (few exceptions, e.g. ROOT or GEANT4)
    - often HEP-specific, rather than using industry standards
  - often lacking tests and quality assurance
  - often ill documented
  - often legacy and poorly maintained (e.g. with authors no longer in HEP)
  - written by many developers with varying degrees of expertise

- *This makes a software upgrade more necessary and complex!*
  - must use new programming paradigms (concurrency, vectorization...)
  - must use new architectures and new types of resources
  - must try out new ideas and new algorithms
  - (while at the same time not disrupting the running systems)

# HSF CWP inception and goals

- CWP first proposed during the LAL HSF workshop (May 2016)
  - Community roadmap for HEP S&C – basis to pursue funds and projects
- WLCG charge for producing an HSF CWP (July 2016)
  - Overall strategy and roadmap for the required “software upgrade”
- *Prioritise software research and development needed to:*
  - *achieve <u>improvements in software efficiency and performance</u>*
  - *enable <u>new approaches in S&C to extend physics reach of the detectors</u>*
  - *ensure the <u>long term sustainability of the software</u>*
- Ultimate focus should be HL-LHC (Run4 and beyond)
  - Roadmap should also identify elements to be tested and used in Run3
  - Specific contact also made with FNAL muon and neutrino experiments, Belle II, Linear Collider community, national computing organisations
    - a primary goal of HSF is fostering communication across HEP and beyond

# HSF CWP in a wider context

- NSF S2I2-HEP
  - Conceptualization of a Scientific Software Innovation Institute for HEP
  - Significant contributions to the organization of HSF CWP workshops
- CERN Openlab White Paper published in September 2017
  - "Future ICT Challenges in Scientific Research"
  - Openlab also involved in the CWP process and organization
- Upcoming WLCG strategy document paper in Autumn 2017
  - Requested by LHCC, before the TDR for LHC Computing due in 2020
  - Will be largely based on HSF CWP, focusing more on short-term R&D

# Kick-off workshop – San Diego, January 2017


- HSF Workshop, SCSD/UCSD, 23-26 January 2017
  - 112 participants (86 US, 26 Europe/Asia)
- Plenaries and topical working group meetings
- Several other workshops on specific topics in the following 6 months

# CWP – WG reports and summary paper

- 13 Working Groups, producing their individual (10-50 pages) reports
  - Software Trigger and Event Reconstruction WG
  - Machine Learning WG
  - Data Access, Organisation and Management WG
  - Software Development, Deployment and Validation/Verification WG
  - Data Analysis and Interpretation WG
  - Conditions Database WG
  - Data and Software Preservation WG
  - Event Processing Frameworks WG
  - Physics Generators WG
  - Workflow and Resource Management WG
  - Visualization WG
  - Computing Models, Facilities, and Distributed Computing WG
  - Careers, Staffing and Training WG

*Reports may still be added from*
*- Security WG*
*- Workflow mgmt WG?*

- One overall roadmap document ("the" Community White Paper)
  - including short executive summaries (~3-5 pages each) from each WG
- See the CWP page on the HSF web site

# Final workshop – Annecy, June 2017

- HSF Workshop, LAPP Annecy, 26-30 June 2017
  - 90 participants (48 US, 42 Europe)
  - Reports (and further progress on writing!) from each of the 13 WGs
- The target to complete the process by August was a bit too optimistic...
  - Later decided to aim for individual WG reports by Sep and overall paper by Oct

# CWP first draft

A Roadmap for HEP Software and Computing R&D for the 2020s

*The HEP Software Foundation*

**Table of Contents**



- Most WGs have completed their individual reports (ready for arxiv)
  - Generally 3 sections: challenges, current practices, R&D programme
- Editorial team has been set up

**Editorial Team**

- Predrag Buncic (CERN) - Alice contact
- Simone Campana (CERN) - ATLAS contact
- Peter Elmer (Princeton)
- John Harvey (CERN)
- Frank Gaede (DESY) - Linear Collider contact
- Maria Girone (CERN Openlab)
- Roger Jones (Univ. of Lancaster) - UK contact
- Michel Jouvin (LAL Orsay)
- Rob Kutschke (FNAL) - FNAL experiments contact
- Dario Menasce (INFN-Milano) - INFN contact
- Mark Neubauer (U.Illinois Urbana-Champaign)
- Stefan Roiser (CERN) - LHCb contact
- Liz Sexton-Kennedy (FNAL) - CMS contact
- Mike Sokoloff (U.Cincinnati)
- Graeme Stewart (CERN, HSF)
- Jean-Roch Vlimant (Caltech)

- First CWP draft prepared by a small subset of the Editorial Team
  - Released on 20 October
  - Send your feedback!

# (1/13) Facilities and Distributing Computing

- Major challenge is finding the best configuration for facilities and computing sites that make HL-LHC computing feasible
  - Hardware cost now dominated by disk and CPU, followed by tape and network
  - Substantial differences in regional funding models

- Upcoming (and present) changes
  - *heterogeneous resources* (HPC, volunteer and cloud computing)
  - *heterogeneous computing architectures* (different CPUs, GPGPUs, FPGAs)
  - new developments in network (e.g. SDN) and storage (e.g. object stores)
    - also large increase in network traffic from other science domains (e.g. SKA)

- R&D programme
  - Understand better the relationship between costs and performance of WLCG
    - Activity starting in the *Performance and Cost Model WG* (October GDB presentation)
  - Define functionalities of a federated data center concept ("*data lake*")
  - Consolidate data mgmt, prototype site storage cache and event streaming
  - Investigate more scalable and uniform workload scheduling
  - Prototype and evaluate a quasi-interactive analysis facility

# (2/13) Data Organization, Management and Access

- Three main challenges
  - *Higher data volumes (~factor 10) and data rates* in HL-LHC era
    - Currently ~1 Exabyte (half tape, half disk) for all experiments
    - Storage and processing costs may restrict scientific output and physics reach
  - New CPU resources (*HPC, cloud*) require more dynamic data access/mgmt
  - *Machine Learning training* needs place new requirements on data access

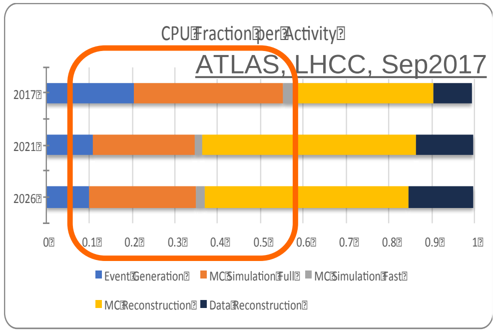
- Currently 3 largely independent domains: data organization (e.g. ROOT), management (transfer systems, catalogs), access (e.g. rfio, xrootd)
  - Blurring boundaries, will need to move to more global optimizations ("DOMA")

- R&D programme
  - Study event-based granularity (*event streaming* may simplify HPC integration)
  - Evaluate Big Data techniques (Spark-like, compression, column- vs row-wise)
  - Investigate *data caching* (both for reco/analysis/simulation and for ML training)
  - Exploit varied quality of service ("tactical" storage, alternatives to tape)
  - Globally optimize data access latency using fewer, larger sites ("*data lake*")
    - Higher availability of high-speed networks allows more extensive WAN access

# (3/13) Conditions Databases

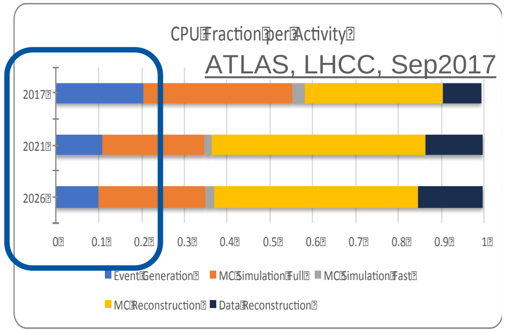
- Conditions data: non-event data required to process event data
  - Low data volume ~TBs, but high read rate ~10kHz from ~10k jobs
  - *Caching* essential, now used by Frontier (CORAL/SQL) and CVMFS (files)
    - Ultimately provided by Squid http caches in both cases
- Challenges and opportunities
  - Event streaming (e.g. HPC) prevents job-level condition access optimizations
  - Long-term maintenance: simplify requirements (too many use cases in COOL)
  - More frequent condition updates in Run3 (decreasing instantaneous luminosity)
  - Online full reconstruction reduces the need to access conditions data offline
- R&D programme
  - New *REST-based system* (CREST) for ATLAS, in collaboration with CMS
    - LHCb have moved to a git-based conditions database
  - Identify industry-standard technologies to replace HEP-specific components
  - Study how to leverage *advanced CVMFS features* for analysis jobs

# (4/13) Detector Simulation




- Presently this means GEANT4 for most experiments
  - Advantage of a *common toolkit* is that improvements benefit everyone at once!
- *Large fraction of CPU time*: ATLAS 35% now, 25% HL-LHC
  - Including(?) the simulation of new detectors from Run4 major detector upgrades
- Main R&D goal: *speed up simulation, improve performance and efficiency*
  - Especially: efficiency on new architectures (many-core, GPGPUs, SIMD, HPC)
  - *GeantV*: ambitious R&D, multi-threaded vectorised particle transport engine
    - beta release planned for 2019, early integration of beta in the experiments for 2020
    - vectorised geometry package (VecGeom) successfully integrated in Geant4 already
  - Also explore different *fast simulation* options, including some based on ML
- Other R&D areas
  - improve physics models to achieve higher physics precision
  - investigate new options for simulating pileup (real zero-bias events, pre-mixing)
  - new pseudo-random number generator, reproducible on parallel infrastructure

# (5/13) Physics Generators

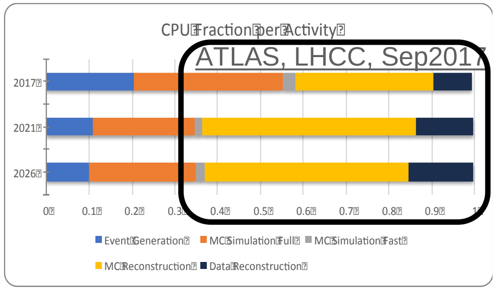



**(Work in progress - section omitted from the CWP first draft)**

*(I am therefore describing **my own – limited – understanding**)*

- Many generators with different features (e.g. Sherpa, MadGraph)
  - The very first step of simulation, prior to and independent of detector simulation
- *Large fraction of CPU time*: ATLAS 20% now (20B events), 10% HL-LHC
- (Some of the) main challenges for the future in my understanding
  - Need *more precise MC generators* (NNLO i.e. next-to-next-to-leading-order)
    - A very active area of theoretical physics research at the moment
  - These will be *even more computationally intensive* than current ones (NLO)
    - And will require more specialized computing such as massively parallel HPCs
  - Unclear to me if the needed resources have been (or can presently be) predicted
    - This may well be another high-priority area of work for software/computing
- See talks by F. Maltoni and G. Zanderighi at the recent HL-LHC Workshop

# (6/13) Software Trigger and Event Reconstruction




- *Largest fraction of CPU time* is event reconstruction
  - ATLAS 40% now, 60% HL-LHC (for MC ~3x as for real data)
    - Dominated by charged particle tracking (less by calorimeter, particle flow, particle ID)
  - Including the effect of higher pileup on the complexity of individual events
  - Including(?) event reconstruction in new detectors from Run4 major upgrades
  - Need benchmarks to measure *physics-per-EUR* and/or per-watt

- Major trigger upgrades upcoming
  - Run4 ATLAS/CMS: factor 10 increase, stream 1 MHz from hardware trigger
  - Run3 LHCb/ALICE: full bx rate (LHCb 40 MHz) to ~real-time *software triggers*

- R&D programme
  - Improve use of *vector units (SIMD)*
  - More efficient use of *many-core parallelism* by using multi-threaded frameworks
  - Investigate offload to dedicated heterogeneous resources (*GPGPUs, FPGAs*)
  - Improve on QA, QC, *testing and physics validation* (also using CI)
  - Evaluate tools and techniques to facilitate *real-time analysis*
  - Physics performance (also at low $p_T$) with high pileup, add timing information
  - Investigate radically new algorithms, also based on ML techniques

# (7/13) Data Processing Frameworks

- *Many common concepts, but several different implementations*
  - Gaudi developed by LHCb, but also used by ATLAS (Athena)
  - CMSSW developed by CMS, forked by Intensity Frontier experiments (art)
  - Basf2 used by Belle II
  - FairROOT and AliROOT closely related, ALICE and FAIR now developing O2

- Key idea to promote in this area is *commonality*
  - Vital to maintain cross-project communication (e.g. Concurrency Forum)
    - Many new ideas to try out, including concurrency and functional programming
  - If possible, try to encapsulate new developments as *common libraries*
    - So that they could be used from different experiment frameworks
    - IMO, "framework consolidation between experiments" beyond this is highly unlikely
  - IMO, frameworks also important to provide *commonality within each experiment*
    - Let the experts do the complex stuff once for all users in a collaboration (more later)

# (8/13) Training and Careers (CWP and – **a lot – my opinions**)

- *Human challenge*, in addition to resource and software challenges:
  - many developers with varying expertise in a collaboration
    - specific to HEP? – users (physicists) are developers, overlap with computing experts
    - solutions evolve continuously based on what has been observed
  - high turn around, people often leaving the field
    - “we” are left with their code, “they” are left with what they learnt in HEP (reusable?)

- Solutions include
  - *training*, share experts’ knowledge with non-experts, spread best practices
    - nice examples: LHCb starter kits (inspired by Software Carpentry) and hackathons
    - computing schools (CSC, Bertinoro, GridKa) are useful but not enough
    - experiment with different tools, including notebooks, WikiToLearn, MOOCs
  - encapsulating new paradigms in *framework libraries and interfaces*
    - let experts work out the technicalities and offer simple solutions to non-experts
  - developing *common libraries and projects* with other experiments
    - a common cross-experiment Q&A (like StackExchange) would have been nice
  - establishing links with *experts from other fields*
  - *improving recognition* of software (and training) work for academic careers
    - passes through proper journal publication, too

# (9/13) Software Development practices

- Challenges when O(100s) people in a collaboration develop code
  - Code quality, modularity, reusability, maintainability
  - Code validation (from unit tests to large scale physics performance tests)
  - Distributed development between geographically separated peers
  - Deployment and interaction with operation teams
  - Software licensing and distribution (and their impact on interoperability)
  - Inhomogeneous community with varying skills and funding models
    - Additional activity splits: production vs. user analysis, online vs. offline

- Some best practices that are used or are being adopted
  - Well defined dev environments, *support multiple platforms*
  - Moving to non-HEP-specific tools (git, CMake)
  - Promoting *code testing* (in advance of MRs) and *continuous integration*

- Ongoing and proposed R&D goals (in addition to those in "Training" and "Frameworks")
  - Continue with HSF working group on packaging
  - Strengthen links to communities outside HEP
  - Develop *tools to measure software performance* and collect profiling info
  - Common *toolkits for statistical regression testing* of reco/simu software

# (10/13) Data and Software Preservation

- Must *preserve the data and the knowledge* needed to derive results from it
  - Low-level bit preservation of the data is largely covered by DPHEP group
  - *Knowledge preservation (software, workflows, documentation) far from easy*
- Several benefits and use cases
  - Analyse old data, to resolve conflicts between results or test new theories
    - e.g. ATLAS exploring ability to reinterpret searches using services like RECAST
  - Ease transfer of knowledge within a collaboration – ensure *reproducibility*
  - Expose data for outreach purposes or to engage external experts (e.g. in ML)
    - All LHC experiments have released some of their data in the *Open Data Portal*
- R&D goals
  - Prototype analysis ecosystems including *analysis portals with a working UI*
  - Evaluate limits of *container technology* in the preservation arena
  - Demonstrate ability to execute production workflows including many containers
  - Collect analysis use cases, tracking evolution towards Big Data environments

# (11/13) Machine Learning

- HEP has used "ML" (e.g. neural networks to classify signal/bkg) for >20yrs
  - But now ML is ubiquitous in Big Data and many mature open source tools exist
    - BDT/NN are now in most HEP papers too, e.g. for pID or b/c/light jet classification
    - HEP-specific (TMVA in ROOT) and non-HEP (Scikit-learn, Keras...) packages
  - Challenge: *new, faster ML algos* for track/vertex reco, trigger, simulation?
    - While maintaining "sufficient fidelity" to the raw data
  - Other challenges from ML computing infrastructure (training, interactive UI...)

- R&D programme (coordinated by common *Inter-Experiment ML Forum*)
  - *Evaluation of data formats* (ROOT vs others) and impact on ML performance
  - Explore *ML as a Service (MLaS)*, starting from what is now provided by *SWAN*
  - Try out Deep Learning computer-vision techniques, e.g. for pID in LArTPCs
  - Evaluate ML for network anomaly detection to optimize data transfers
  - And many other ideas for simulation, trigger, tracking, anomaly detection...

# (12/13) Data Analysis and Interpretation

- Final stage of analysis – e.g. final data reduction and interactive plots
  - not part of “production” computing (e.g. not in ATLAS LHCC plots) – laptops, too
  - HEP currently gravitates around *ROOT* (I/O, ntuples, histograms, plots...)
    - challenges: *maintenance and sustainability* of a HEP-specific product
  - rise of many alternatives outside HEP (Python ecosystem, Spark...)
    - challenges: *interchangeability and interoperability* (“bridges” and “ferries”)

- R&D programme (largely discussed also at the Amsterdam workshop)
  - *Make Python a first-class citizen* in HEP (specifically in ROOT)
  - Enable dynamically *plugging in open-source tools* (interchangeability)
  - Interoperability: *bridges* (use external tools) and *ferries* (use external data)
  - Prototype *functional and declarative* programming models
  - *Analysis facilities* (e.g. Spark-like, query-based) beyond current “primitive” ones
  - Prototype an Interpretation Gateway, extending present recasting tools

# (13/13) Visualisation

- First main use case: *event displays* and *geometry visualisation*
  - 3d rendering via OpenGL (standalone) or WebGL (browsers)
  - Many experiment-specific tools
- Second main use case: statistical visualisation (e.g. histograms)
  - *ROOT* widely used in HEP
  - Increasing use of *Python (Matplotlib)* and JavaScript – e.g. *Jupyter notebooks*
- Common feature is *interactivity*
- Main challenges and [R&D goals]
  - Code *maintainability* and sustainability
  - Leverage *common packages* to decrease experiment-dependency
  - Improve access to geometry and event data through a streaming interface

# CWP - Conclusions

- ATLAS and CMS needs at HL-LHC exceed those achievable by incremental changes to code and computing facilities within the foreseen budget
- HEP S&C needs a *step change* and an *investment in people* to solve these issues
- Main goals should be
  - improvements in software (and computing) efficiency, scalability and performance
  - enable new approaches that can radically extend the physics reach of the detectors
  - ensure the long term sustainability of the software
- We must work together as a community and link to experts outside of HEP
  - "Together" includes all four LHC experiments, Belle II, DUNE, ILC...
  - Challenges will start sooner for LHCb, ALICE (and others) than for ATLAS and CMS

***"The amount of data that experiments can collect and process in the future will be limited by affordable software and computing, not by physics"***

# CWP and HSF outlook

- Second draft of the CWP expected fairly soon
  - Work still ongoing in some WGs and comments being received
  - Close 1st draft for comments on Nov 10? (then release 2nd draft)
- Other HSF activities are going on in parallel
  - Licensing, packaging, training...
  - But the emphasis in recent months has clearly been on the CWP
- Joint HSF-WLCG Workshop in Napoli (26-29 March 2018)
  - Bring together both the computing and software communities
  - The HL-LHC challenges must be addressed on both sides together!
- HSF session at CHEP2018 (probably on Sat-Sun 7-8 July )

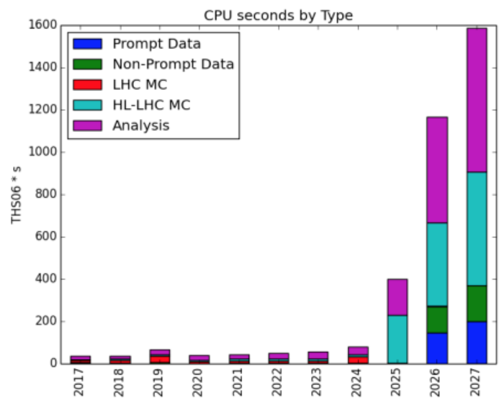
# Backup slides

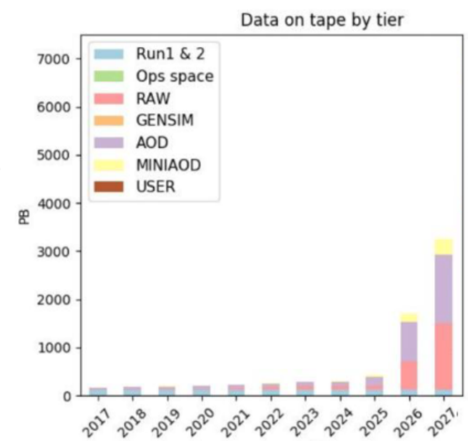
# HL-LHC expected shortfall of resources

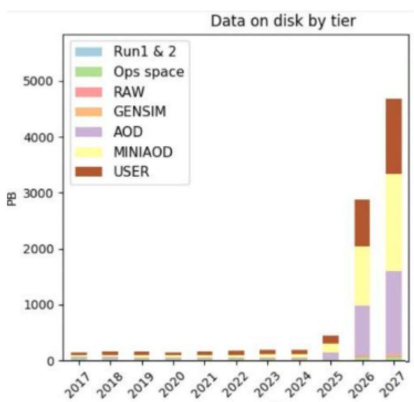
## CMS

### HL-LHC new working numbers

- CMS does not have newer officially blessed numbers for HL-LHC
- Still, work has been ongoing also due to the DOE request to US-CMS for long time planning
- Main changes wrt to older models (see for example ECFA presentation by S.Campana) are
  - Expectation of 10%/y code performance improvement
  - Rely largely on MiniAOD(SIM) for operations; AOD(SIM) an archival thing




Take home messages for 2027:
- 50 MHS06 CPU
- 5EB disk
- 3EB tape

- Wrt to 2017, assuming a +20%/y by Moore and friends, the excesses are ~6x for CPU, ~4x for storage

18

*Liz Sexton-Kennedy – LHCC Meeting – 12 Sep 2017*

# Run3 and Run4 upgrades

## Experimental upgrades

**Preparation in LS2 for Run 3**

LHCb
- Significant detector upgrade
  - Rebuid of all tracking detectors
- Triggerless readout @ 40MHz
- Trigger fully implemented in software

ALICE:
- Significant detector upgrade (inner tracking, TPC)
- Triggerless readout for Pb-Pb @ 50kHz
- New integrated DAQ-online system with smart data compression

**Preparation in LS3 for Run 4**

CMS
- New silicon tracker with trigger capabilities (at IA rate)
- Improved L1 latency and calorimeter end-caps

ATLAS:
- New inner tracker and readout improvements in calorimeters
- Track trigger operating at level-0 rates

*ACAT 2016*

22. January 2016 Johannes Albrecht 4/40

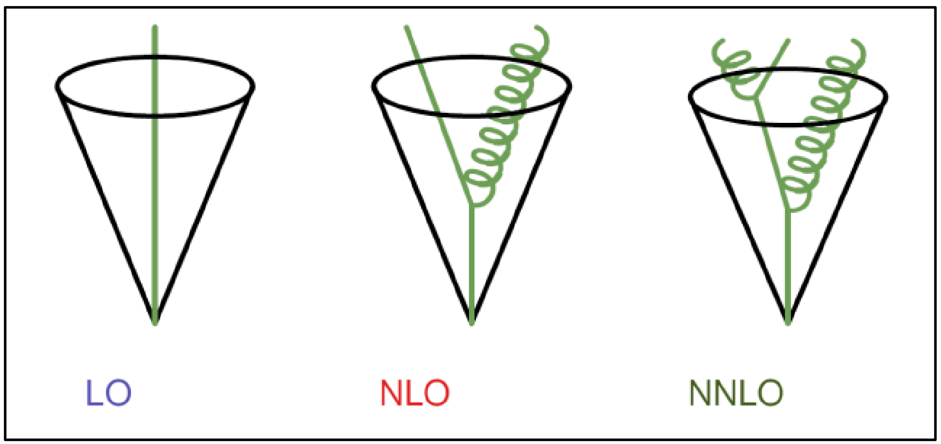
# LO, NLO and NNLO

- Perturbative QCD: expansion in powers of $\alpha_s \ll 1$
  - Expansion of observable

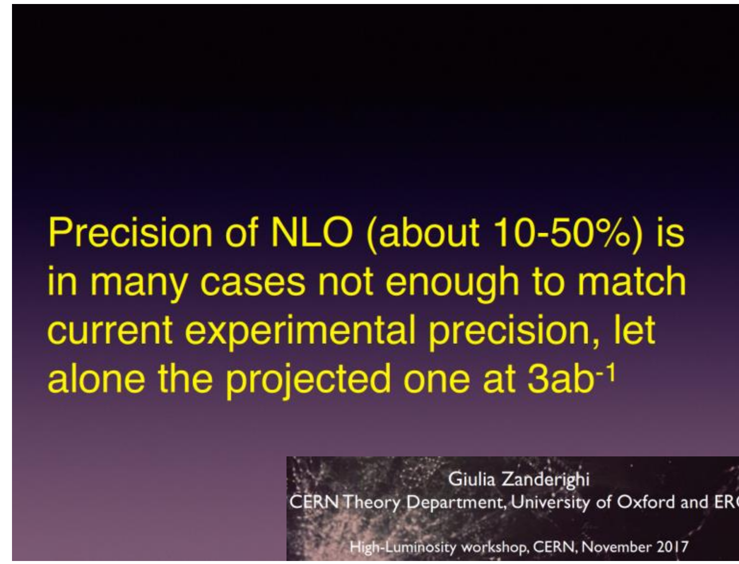
$$f = f_0 + \alpha_s f_1 + \alpha_s^2 f_2 + \alpha_s^3 f_3 + \ldots$$

  - often compute only the first one (leading order, LO) or two (next-to-leading order, NLO) terms
- Technique for calculations : using Feynman diagrams

Aude Gehrmann-De Ridder Perturbative QCD - Lecture




CERN Academic Training 2013




HL-LHC Workshop Nov 2017