



# CERN and HEP Training Needs

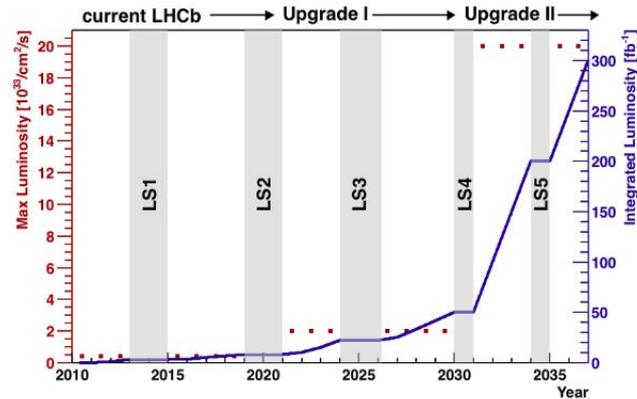
Graeme Stewart, CERN EP-SFT  
Stefan Roiser, CERN IT-SC-RD  
Maria Girone, CERN IT-DI-OPL



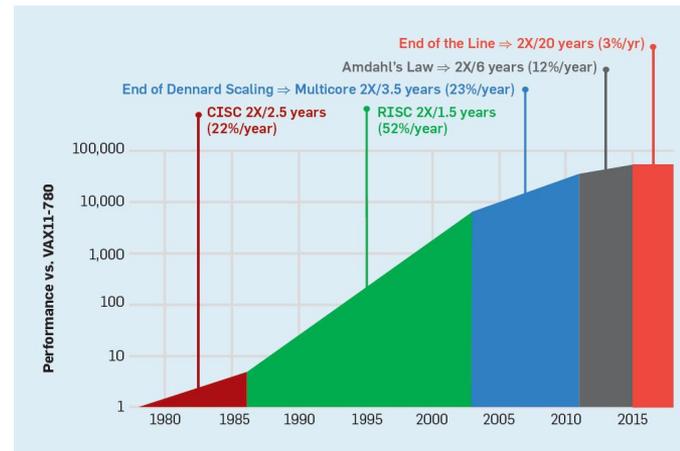
PRACE-CERN-GÉANT-SKAO kick-off  
workshop on HPC, 2020-09-29

# Challenges for HEP Computing

- We have challenges coming from our physics programmes
  - Extremely high rates from detectors, 30MHz with need for accurate physics reconstruction and data reduction
    - Trend to move from hardware systems (“triggers”) to ones based on software
  - Precision physics requires very accurate simulations (scaling with data rate), but constrained within the overall computing budget
- And from the evolution of the computing hardware we run on
  - CPU performance stalling, co-processors becoming more and more prevalent
  - Ongoing software challenge for our field



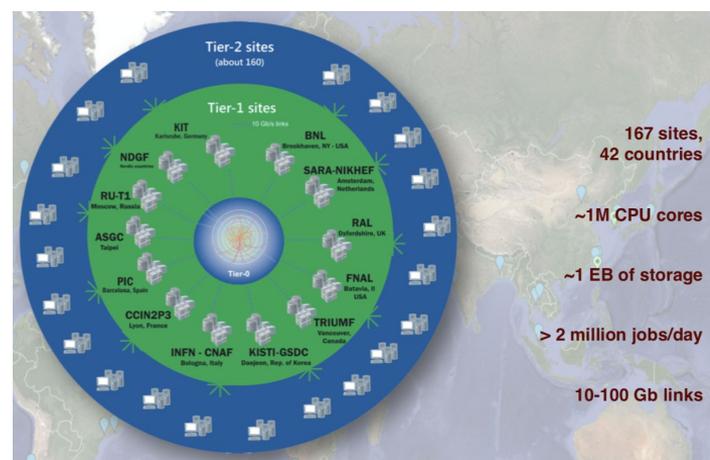
[arXiv:1808.08865](https://arxiv.org/abs/1808.08865)



[A New Golden Age for Computer Architecture, ACM](#)

# CERN and HEP as a software development community

- High Energy Physics has a vast investment in software
  - Estimated to be around 50M lines of C++
  - Run at huge scale, 1M cores, 1 exabyte of data
- Tens of thousands of experiment collaborators
  - Very high fraction write *some* software
  - Far fewer are dedicated to software development at >0.5 FTE
- High turnover of PhD students and postdocs
  - Software is identified as one of the key skills they acquire in our field
  - We need to train these people *effectively* and *usefully*



athena

ATLAS Experiment main repository for Athena



Gaudi  
Project ID: 38 | Request Access

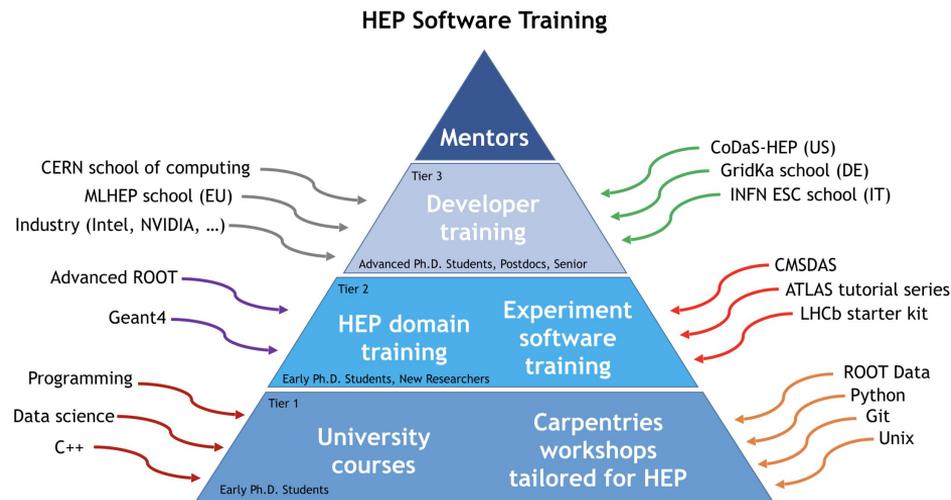
Unstar 39 Fork 58

7,007 Commits 36 Branches 459 Tags  
46.3 MB Files 10.9 GB Storage 26 Releases



# Training Needs

- Wide spectrum of needs for training
  - Basic key skills shared across the field (even between sciences)
    - Good C++ training is a particular issue for us
  - Specific training in HEP domain and experiment software toolkits
  - Advanced training for programming accelerators and optimising the performance of key components
    - New Compute Accelerator Forum starts this week
- The number of people narrows as one advances up the chain
  - In addition, today's students can be tomorrow's teachers



- HPC training has to cover
  - General engineering to make effective use of these machines
  - Specific points on job submission, data access, etc.
    - Which needs integrated into our experiment production systems

# Key Points: Modern, Scalable, Sustainable

- Training needs to be modern
  - We should not be bound to the practices of the past
  - Modern training is both optimal for HEP (best practice in software development) and for learners (up to date skills)
- Training has to be scalable
  - We have many people to teach and few dedicated (human) resources to do it
    - There is a *great demand* for training in the field - our C++ course was oversubscribed by x5
  - People should be able to become trainers with minimal efforts
    - Early postdocs passing on their knowledge to new PhD students
  - Delivering training needs to be rewarded
- Training has to be sustainable
  - One-shot courses consume a lot of time
  - Material for training should be a collective effort, following best didactic practice
    - We have collaborated with The Carpentries for good models
  - And should also be suitable for self-study
- Today we also have to do all of this in a *remote setting*, so innovation in teaching and tutorial setup is needed (we are learning!)