ROOT

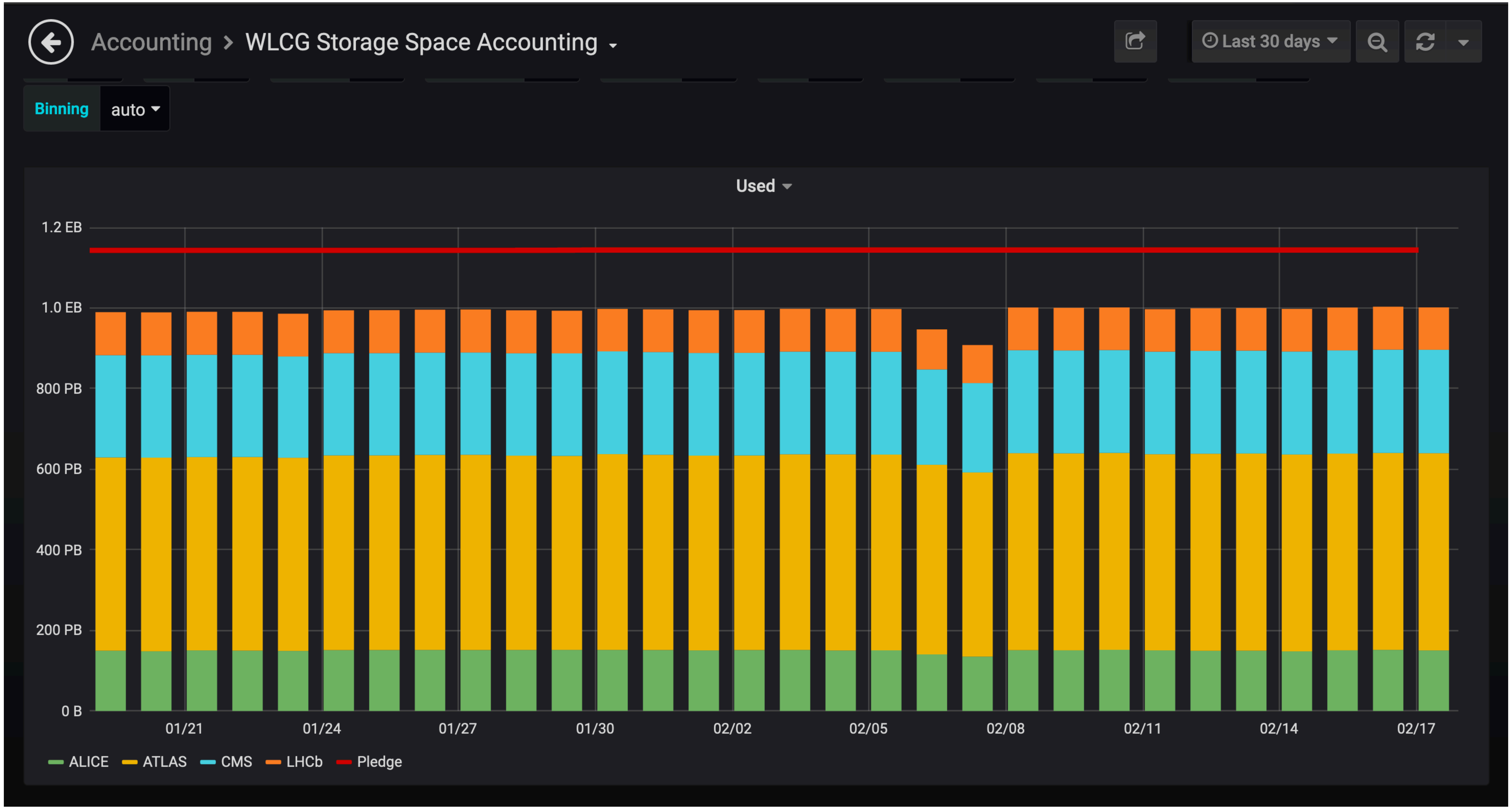
Axel Naumann, LHCC, 2020-02-18

Content

- Role
- Customers
- Team
- Investing

ROOT's Role

Data



Accounting > WLCG Storage Space Accounting
Last 30 days
Binning auto
Used
1.2 EB
1.0 EB
800 PB
600 PB
400 PB
200 PB
0 B
01/21
01/24
01/27
01/30
02/02
02/05
02/08
02/11
02/14
02/17
ALICE
ATLAS
CMS
LHCb
Pledge

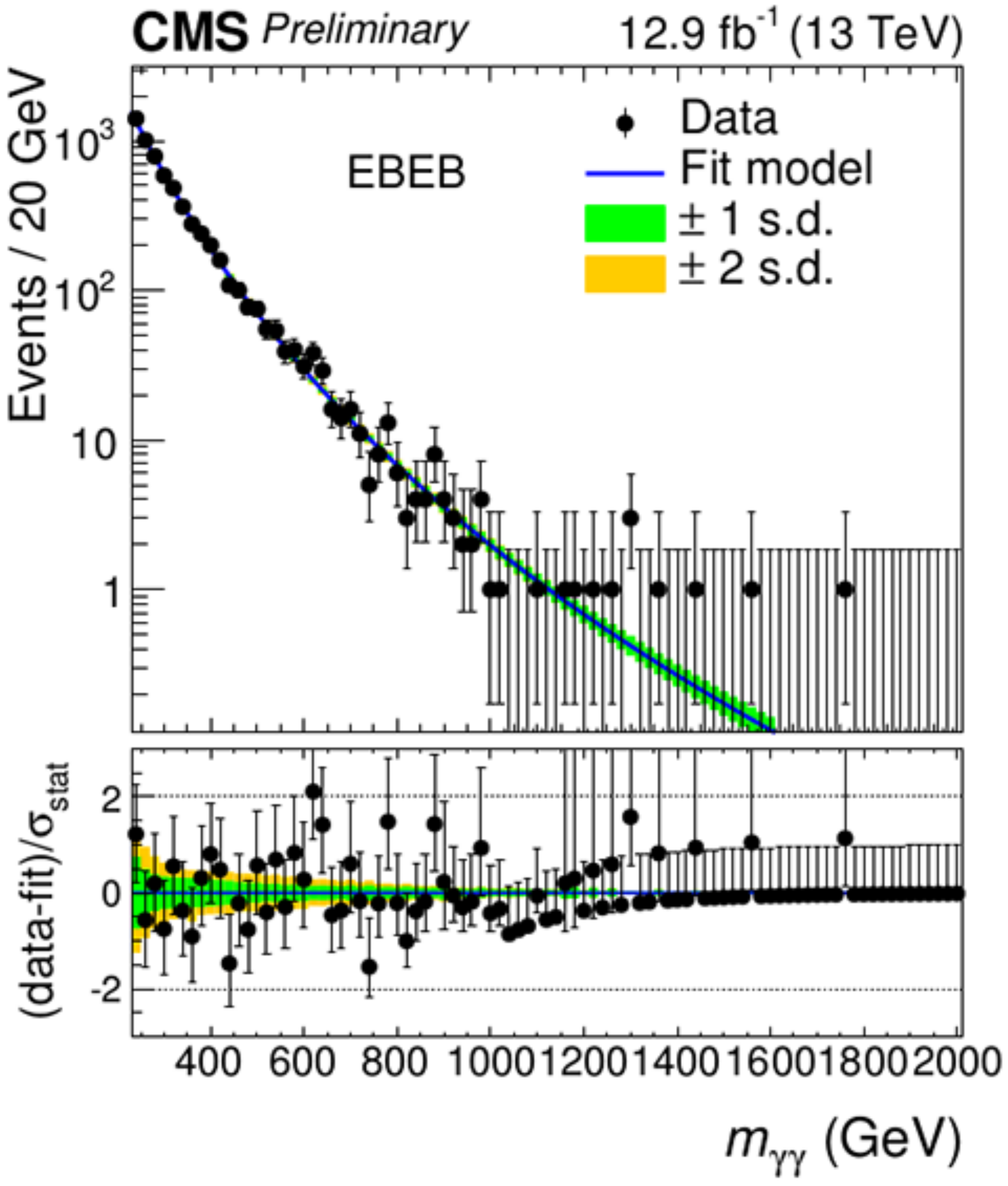
Data

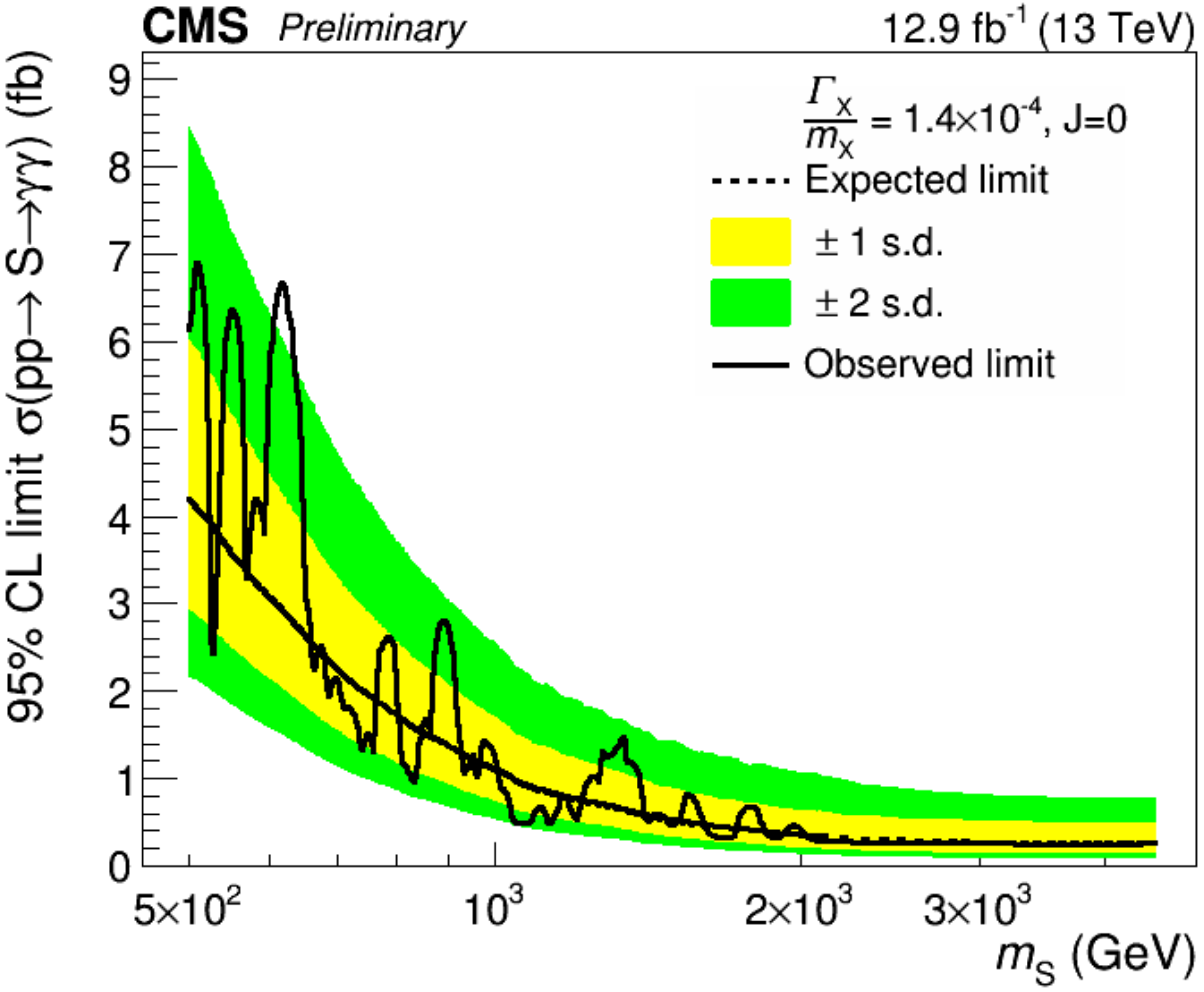
> 1 Exabyte of data in ROOT files

Data

- Extremely efficient data source, faster than anything else out there
  - That's for HEP's data: repetitive, collections of collections, statistically independent events
  - https://iopscience.iop.org/article/10.1088/1742-6596/1085/3/032020/meta
- Close collaboration with experiments: mini/micro/nano-AOD data models, parallelism, workflow optimization
- Common format helps data preservation

Graphics


CMS Preliminary
12.9 fb^{-1} (13 TeV)
EBEB
Data
Fit model
± 1 s.d.
± 2 s.d.
Events / 20 GeV
(data-fit)/σ_{stat}
$m_{\gamma\gamma}$ (GeV)


CMS Preliminary
12.9 fb^{-1} (13 TeV)
$\frac{\Gamma_X}{m_X}$ = 1.4×10^{-4}, J=0
Expected limit
± 1 s.d.
± 2 s.d.
Observed limit
95% CL limit σ(pp→ S→γγ) (fb)
m_S (GeV)

Graphics

- Define the language to communicate results, convince reviewers, to get analysis published
- Cornerstone of HEP

Python

- Massive libraries of highly efficient C++ code
  - Experiments' code, id-groups, Geant4, ROOT, mini-frameworks

versus

- Physicists wanting to get their analysis done

Python

- ROOT bridges C++ and Python, for any code, dynamically: **UNIQUE**
  - Enables simple analysis for physicists: more and more analyses in Python, now ca. O(50%)
  - While using highly efficient C++ code behind the scene for bulk computing
- Orders of magnitude acceleration compared to "production" Python-only analysis, e.g. https://indi.to/gQL7P

Math

- HEP's foundational math library
- Modeling with RooFit: parametrizations ("fits"), correlations, combinations
- Machine learning with TMVA

Math

- RooFit: unique.
- TMVA, otoh:
  - Optimized for high-throughput from ROOT files
  - Provides sustainable, stable interface across Keras, Tensorflow 1/2,...
  - Plus boosted decision trees: backbone of many MVA analyses

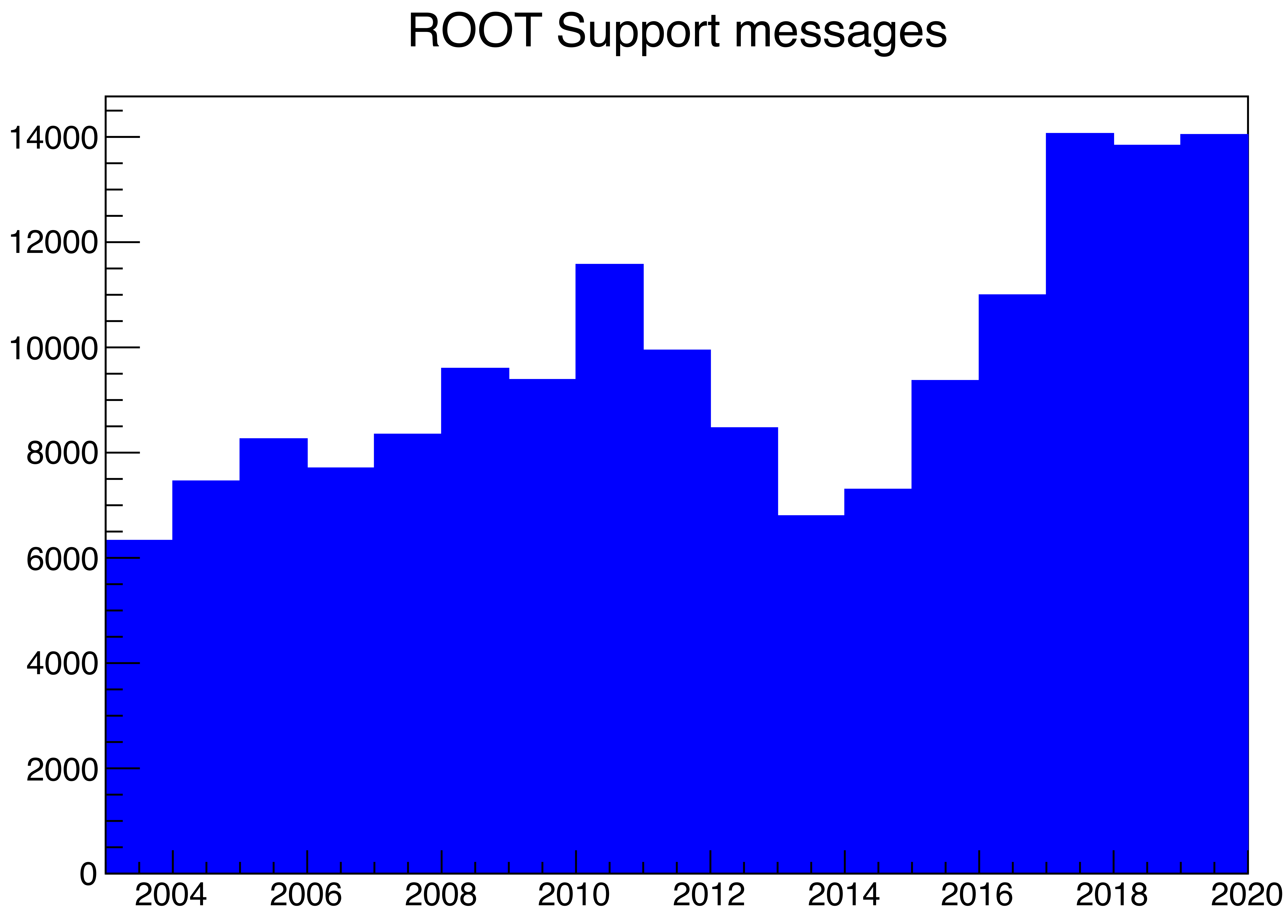
ROOT's Customers

Experiments' Software

- Excellent, continuous exchange
  - CMS invests in ROOT

Physicists

- Massive user base, estimated to be 30'000
- ROOT team provides massive user support: 56 messages/work day
  - Direct interaction with +/- all PhD students of the world



Physicists

- Large range of needs
    - From "getting started with coding" to background modeling how-to
- We are aiming for reduction of support load
    - See "future of ROOT"

ROOT's Team

Team Founded around 2000

- Almost all permanent contributors will retire in 5-10 years
- ROOT evolution needs hands
  - Recruited many temporary contributors, also externally
  - Hard to sustain backbone of HEP software (and evolution thereof) without continuity

The Team @ CERN, Begin 2019

ROOT@CERN?

- Majority at CERN - provides sustainability and constant rotation of fellows and students
- Developers at CERN, Fermilab, GSI, Princeton, University of Nebraska Lincoln, LAL/IN2P3, UC San Diego

Investing in ROOT

State of ROOT: The Challenge

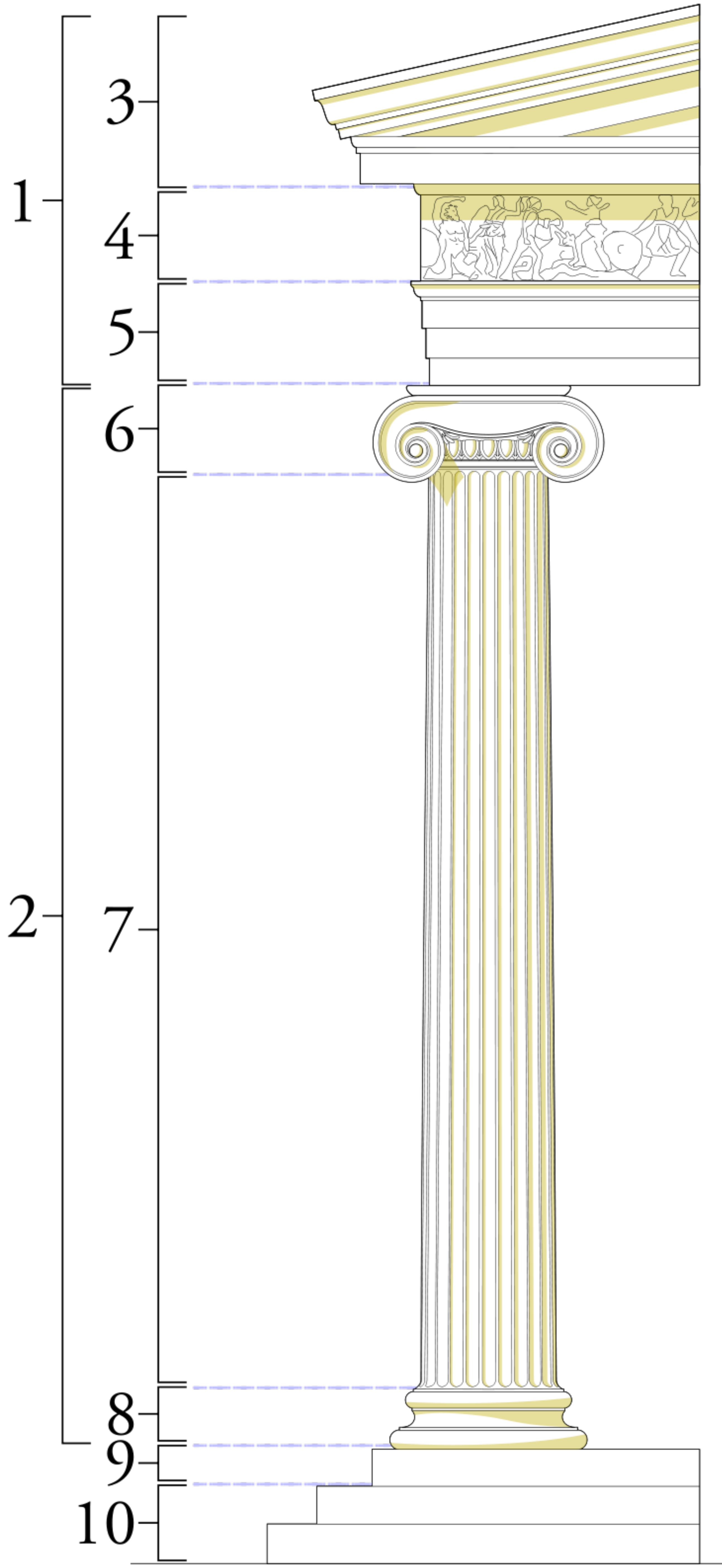
- ROOT is everywhere, centrally developed for HEP
- But physicists spend too much time debugging
- ROOT *perceived* as "dated", "of the past"
  - Despite state-of-the-art technology!

State of ROOT: The Solution

- Brand new implementations
- Interfaces are simple to use, hard to misuse
- Aggressively communicating innovation

"ROOT7"

- Massive, multi-year development effort
- Focused on main ROOT columns:
  - Analysis
  - I/O
  - Graphics
  - Foundational math: histograms



Mental Model

- Highly efficient implementations, sacrificing feature bloat
- Simple, robust interface abstractions
- Learning from >20 years of usage
- Optimizing for current and *future* requirements
    - HDD → SSD; 1GB → 4GB RAM; 1 → 32 cores; "all of C++"→ efficient subset; NVRAM; HPC

Code Rejuvenation

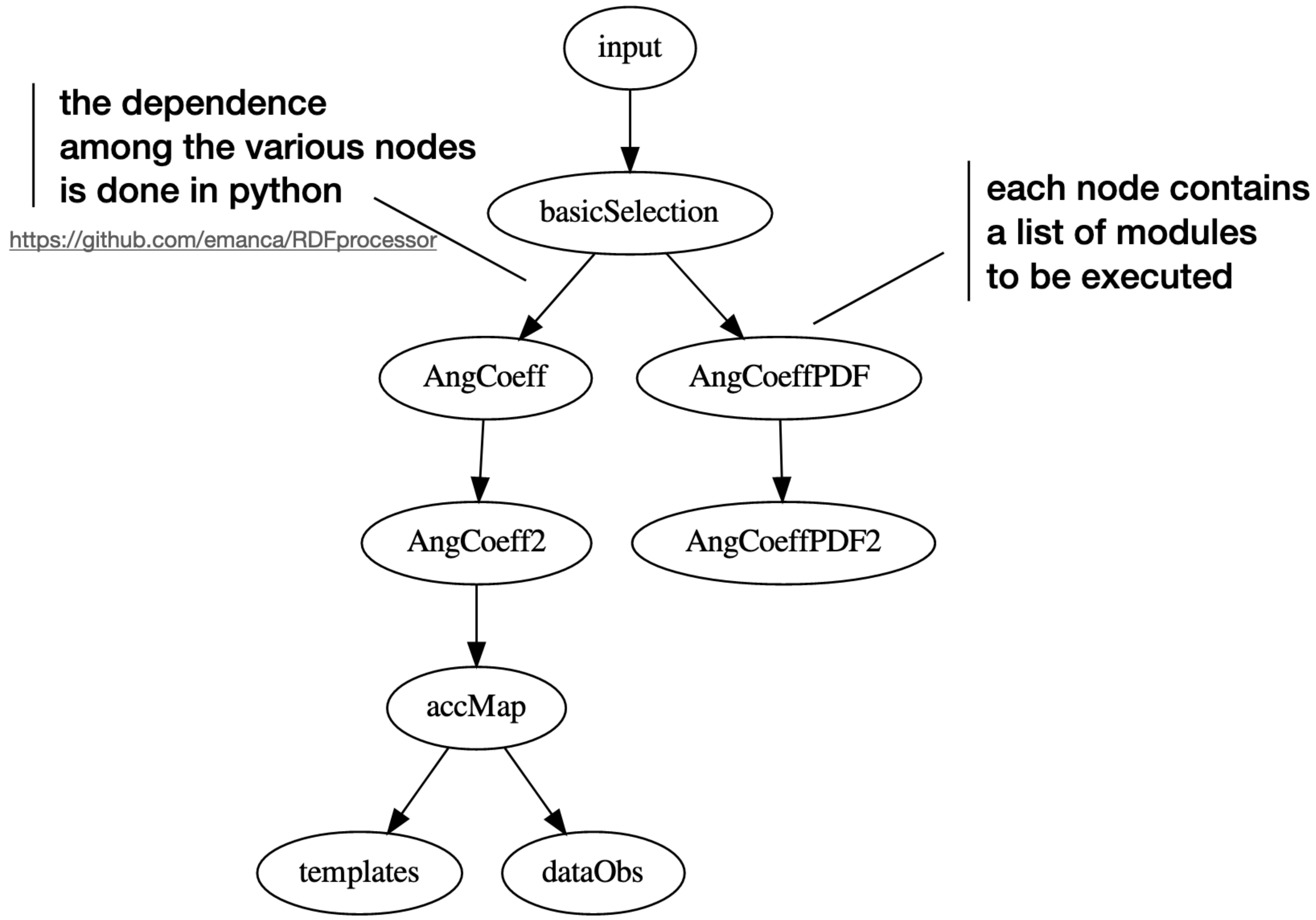
- New implementations break backward compatibility
  - TTree stays, but will be superseded; similar for other features
- Better documentation, better test coverage, better design
- Rewriting software from scratch is a mistake: https://www.joelonsoftware.com/2000/04/06/things-you-should-never-do-part-i/
  - But here instead: implementing new, focused features; reducing real, existing complexity accumulated over 20 years

Team Rejuvenation

- Hand-over of existing code (3 million lines of code) near impossible
- Sustainability by renovation
  - Many features rely on external libraries instead of in-house solutions
  - Others developed in teams to share knowledge; modular and minimal

Example Goals

- Dramatic simplification of writing analyses: "declarative"
  - See EP Software Seminar https://indico.cern.ch/event/849610/
  - Including scale-out across clusters / HPC (R&D), see e.g. https://link.springer.com/chapter/10.1007/978-3-030-29400-7_18

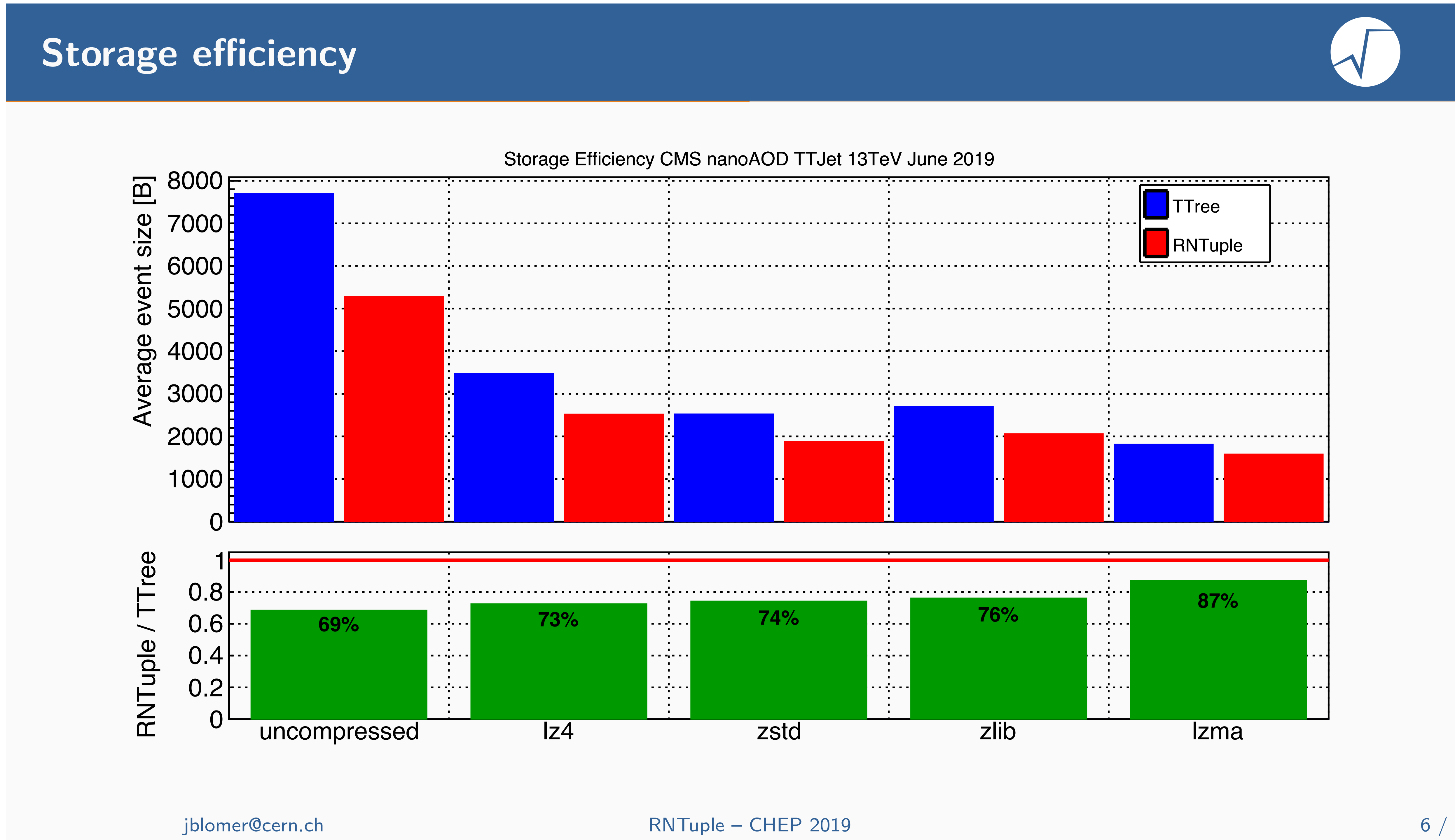


Example Goals

- Use of HPC, heterogenous computing also for analysis: physicists define "what", ROOT decides "how"
- ROOT's C++ interpreter cling allows live migration of code to GPUs
  - In use outside HEP: https://www.hzdr.de/publications/Publ-29350 https://zenodo.org/record/1412256
- Building powerful ingredients across sciences, to enable fundamentally new features, e.g. modeling + machine learning with HL-LHC's many parameters

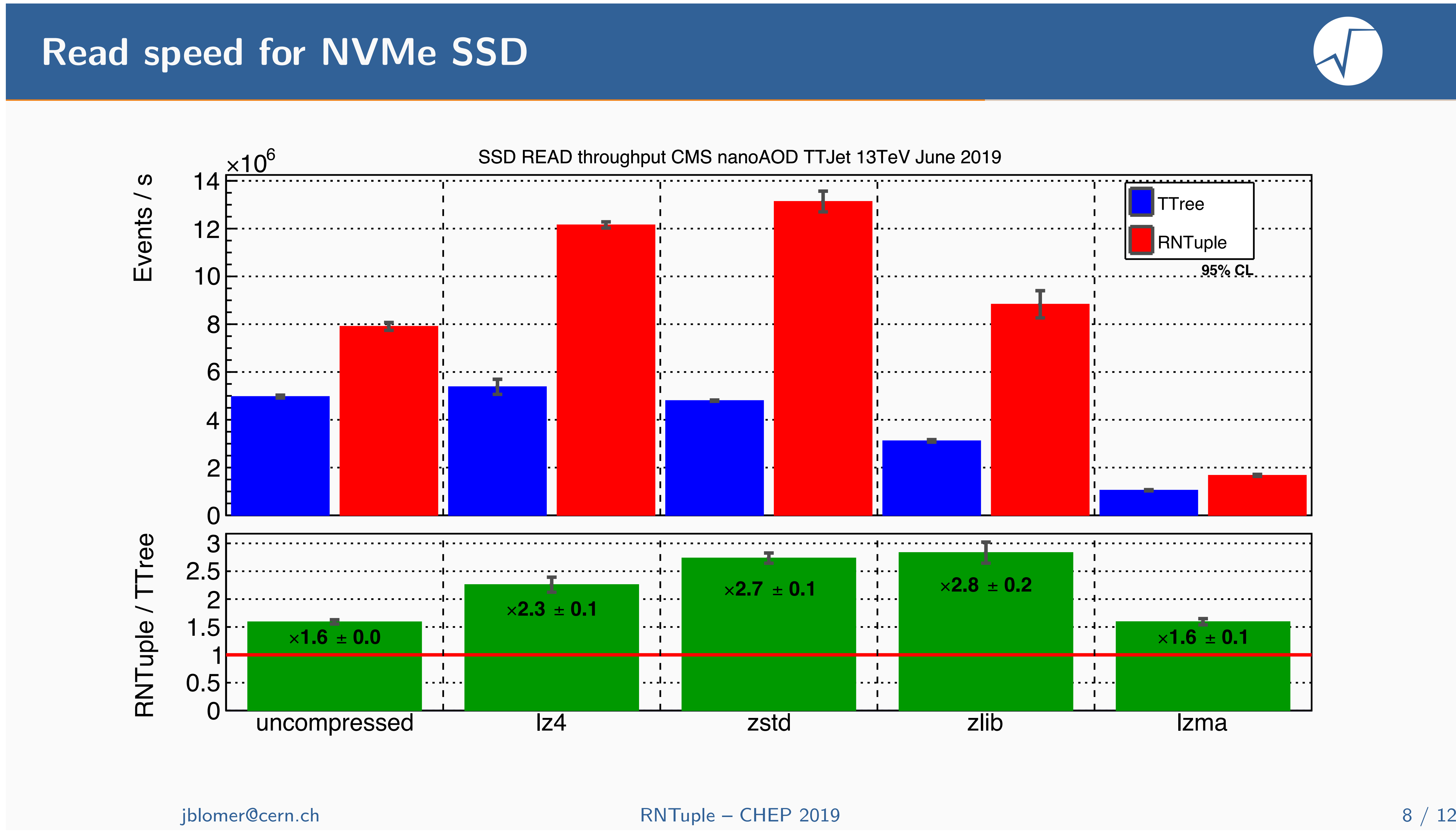
Example Goals

- Considerable storage space reduction and throughput increase
  - See https://indico.cern.ch/event/773049/contributions/3474746/attachments/1937507/3211341/rntuple-chep19.pdf



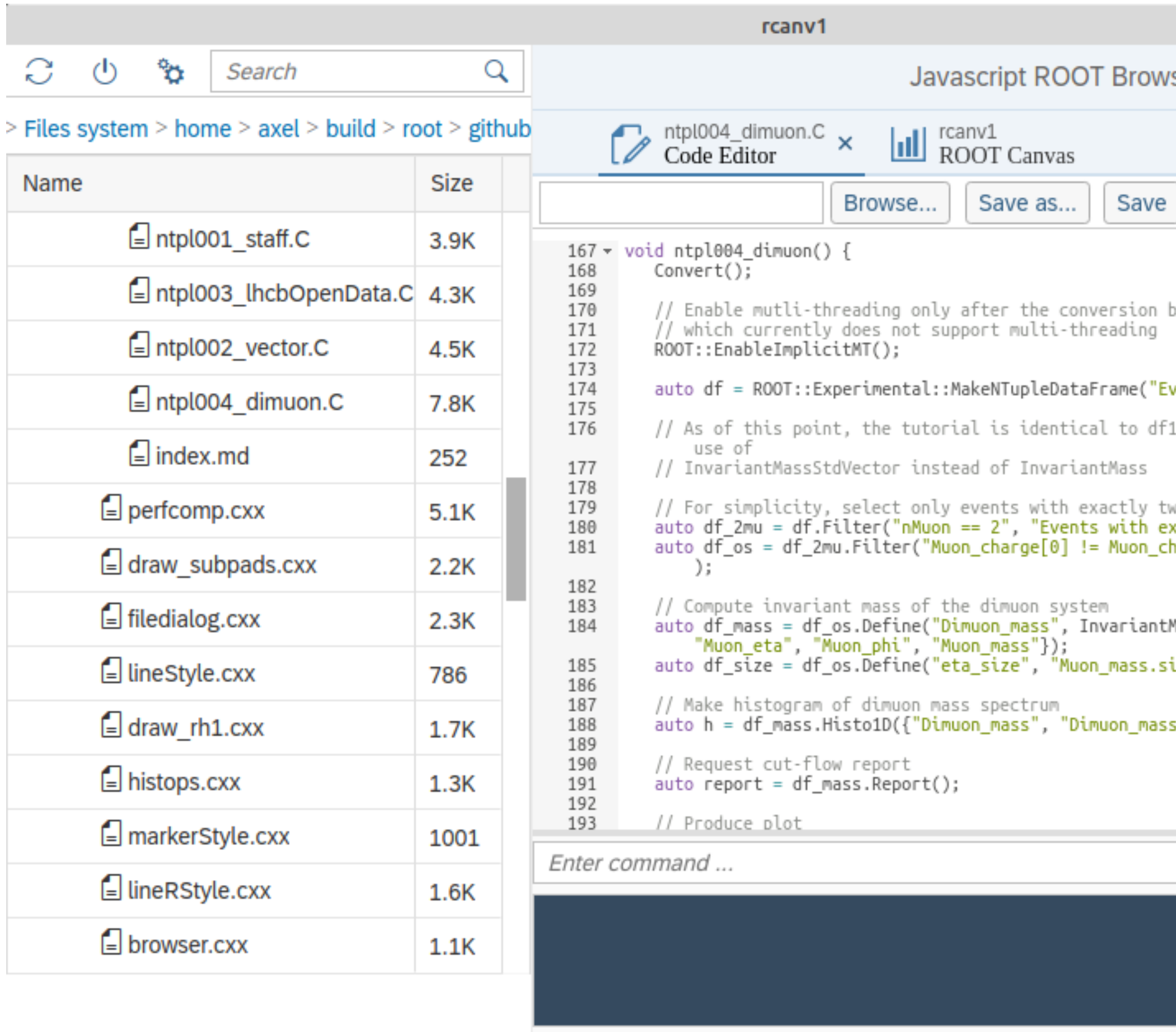
Example Goals

- Considerable I/O speedup
  - See https://indico.cern.ch/event/773049/contributions/3474746/attachments/1937507/3211341/rntuple-chep19.pdf



Are We There Yet?

- Multi-year developments
- Lots of R&D; user feedback wash & rinse; optimizations
- Long way from demonstrator, to production, to feature-complete
  - Example WebGUI: builds on JSROOT (2012), web-serving (2014), now experimental RBrowser (2019)



Are We There Yet?

- ROOT is investing now
  - To research, demonstrate and develop solutions for HL-LHC
  - Incredibly productive
- Relying on new generation, largely temporary contributors

Conclusion

ROOT on the way to HL-LHC

- Innovation is **required**, or else
  - Losing physicists and experiments to perceived "state-of-the-art" solutions: "industry standard", but optimized for non-HEP data and non-HEP analyses
  - Lack of common tools means duplication of effort
  - Loss of in-house expertise
  - Reliance on "industry tools" risks maintainability and longevity

ROOT @ HL-LHC Innovation

- Focus on data and throughput, to increase efficiency of physicists, luminosity, (computing) silicon, and CHF
- Considerable effort even for a long-term project
  - Challenges for sustainability during generation hand-over
- Central expertise and investment for the good of all HENP