Hello RNTuple and friends: Status and Plans for ROOT

Axel Naumann for the ROOT Team
2020-10-27, CEPC 2020, Shanghai (sorta)

Outline

- ROOT's role
- Vision: where ROOT is heading
- News: how far we got
- Conclusion

Introducing ROOT

- ROOT is a centerpiece of HEP, virtually every HEPicist uses ROOT for analysis, > 1 exabyte of data in ROOT format
- Common (also graphics) language, common data format, common grounds
- Coherently designed, integrated solution with optimized interplay
- Core in C++, with dynamic Python bindings

Why to bet on ROOT

- Targeted for HEP: simplicity, efficiency, support
- Allows to predict changes, adapt and benefit: solutions and R&D tailored to our very own problems
- Interface with and learn from other tools
- Single point of improvement: contribute here to have an impact, coherency and synergies (experiment vs analysis etc) guaranteed
- Advantage: community knows its challenges; gets a coherent, reliable, performant and agreed solution

Team

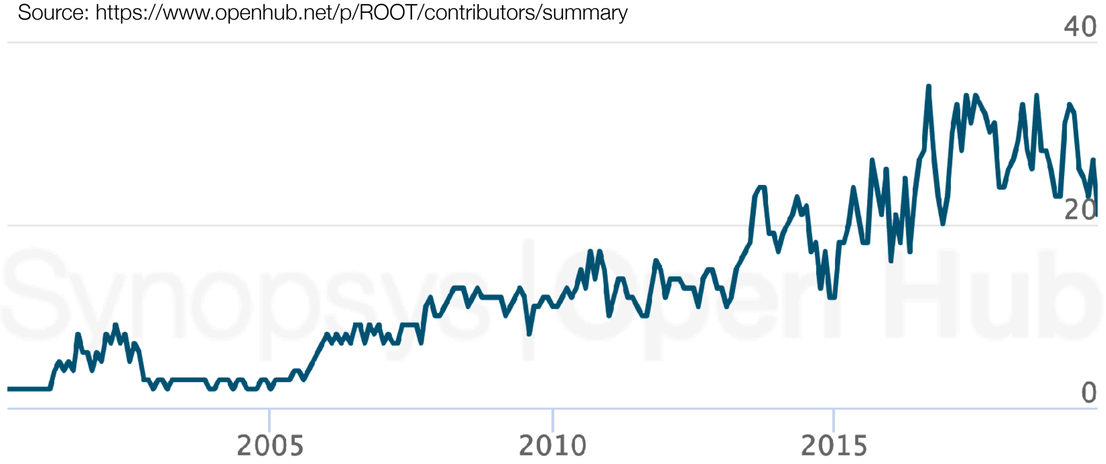
- Sitong An, CERN
- Bertrand Bellenot, CERN
- Jakob Blomer, CERN
- Philippe Canal, Fermilab
- Olivier Couet, CERN
- Bernhard Gruber, CERN / TU Dresden
- Enrico Guiraud, CERN
- Stephan Hageboeck, CERN
- Sergey Linev, GSI
- Lorenzo Moneta, CERN
- Axel Naumann, CERN
- Vincenzo Padulano, CERN
- Oksana Shadura, Uni Nebraska Lincoln
- Enric Tejedor, CERN
- Vassil Vassilev, Princeton Uni
- Stefan Wunsch, CERN

Contributions

- Many, many part-time contributors
- Extremely active also due to them!
- Very sustainable dev model, for decades

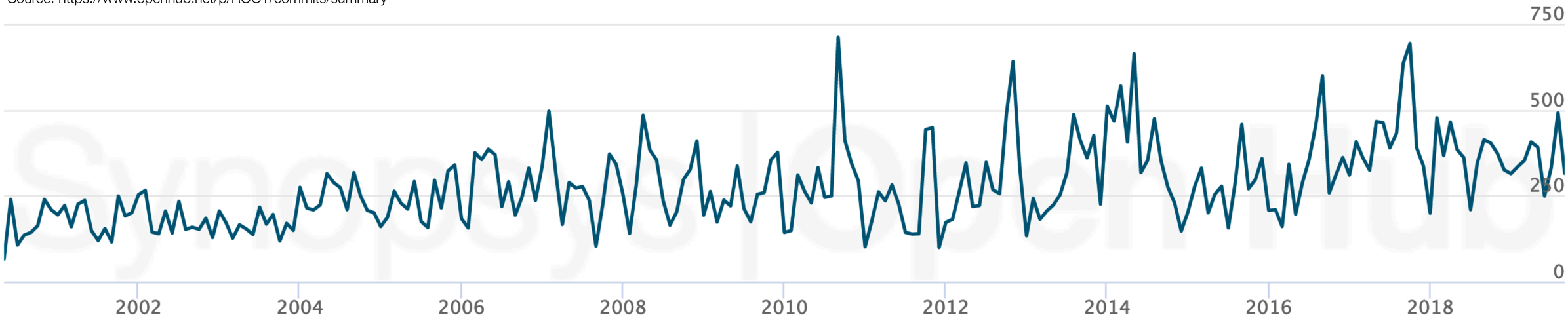
Contributors / month

Source: https://www.openhub.net/p/ROOT/contributors/summary



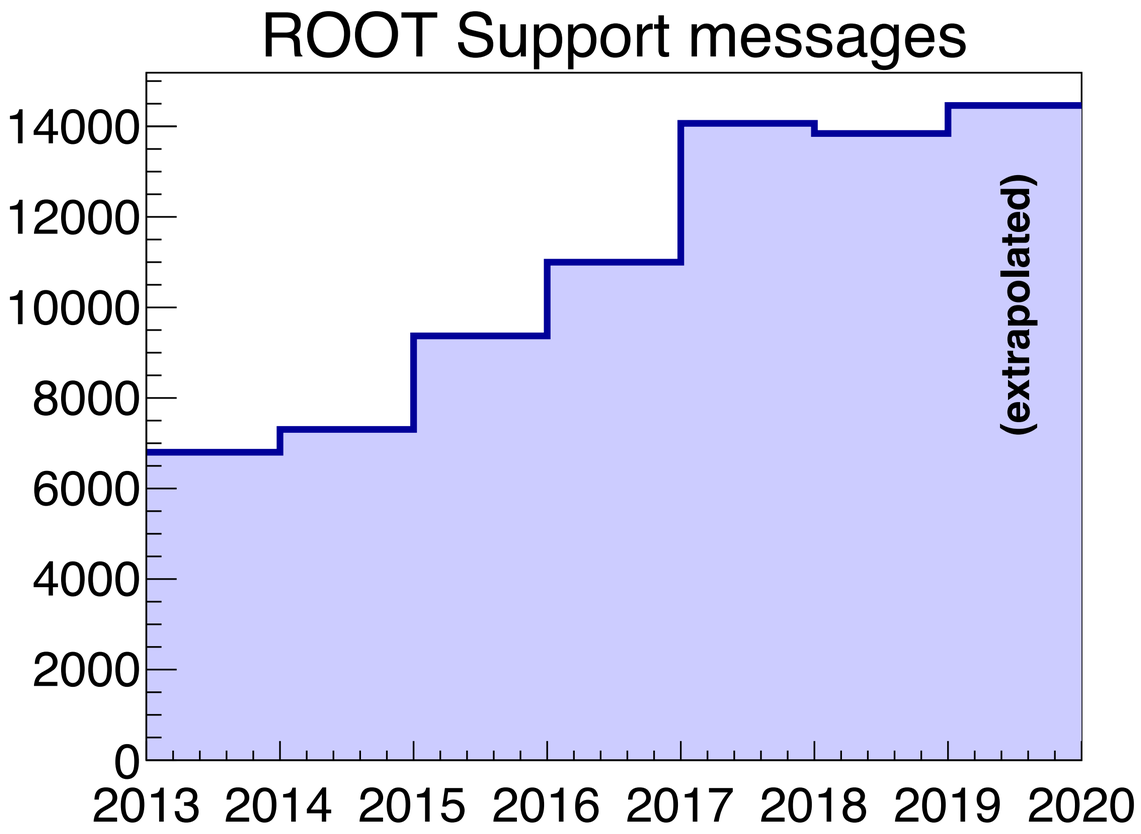
Commits / month

Source: https://www.openhub.net/p/ROOT/commits/summary



Support

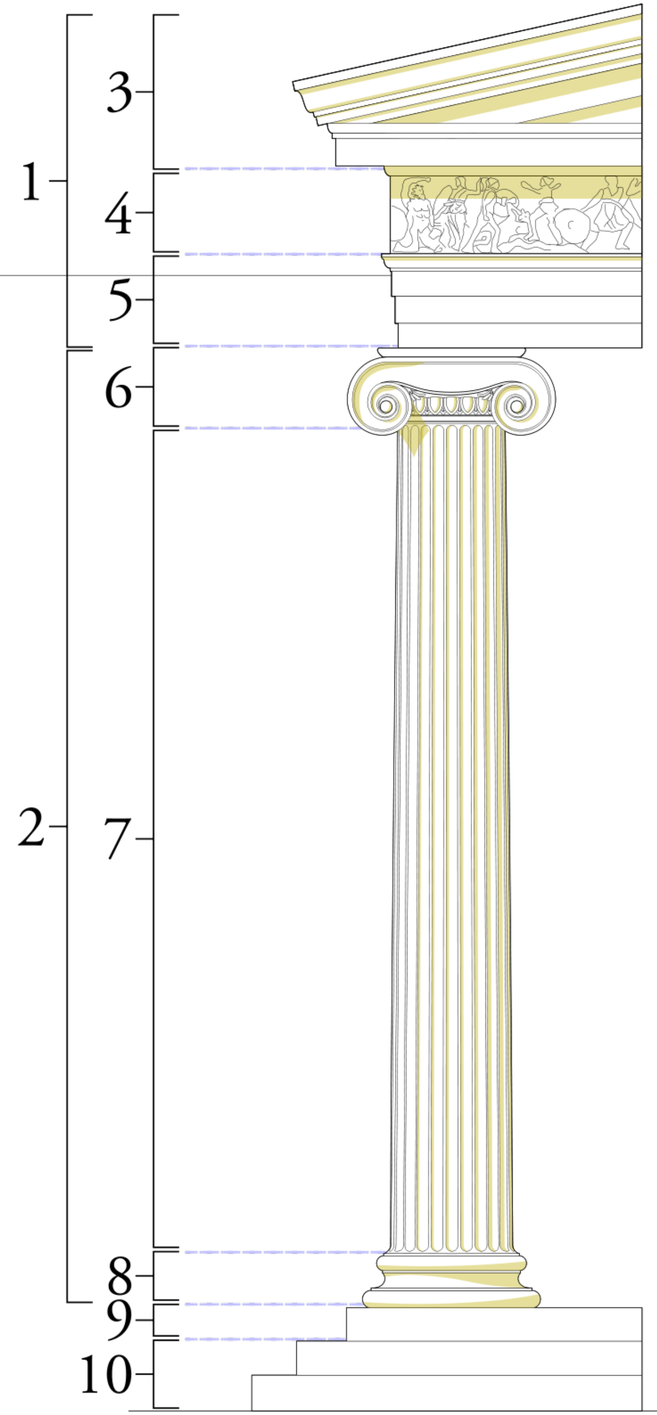
- https://root-forum.cern.ch
- Approx 56 messages per work day in 2018...2020



Consequence and Vision

"ROOT7"

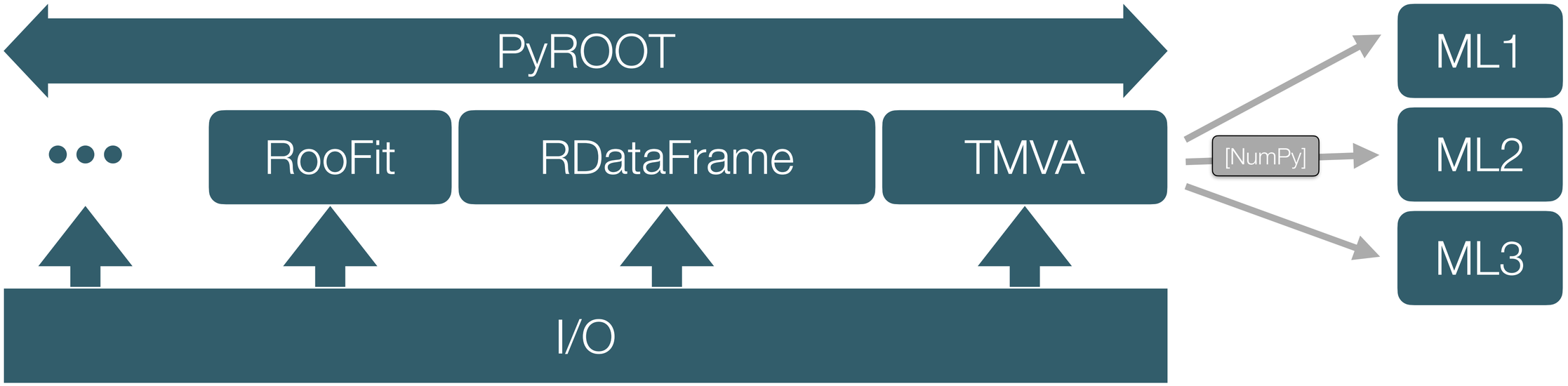
- Massive, multi-year development effort
- Focused on main ROOT columns:
  - Analysis: parallelism, Python, RDataFrame, RooFit, TMVA
  - I/O: TTree successor RNtuple
  - Graphics: web-based graphics, GUI, event display
  - Foundational math: histograms



Why, why those?

- Most relevant for physicists
- We can save your time: better defaults, simpler, faster
- Ensure homogenous, consistent design: like "TObject*", only 2020s style

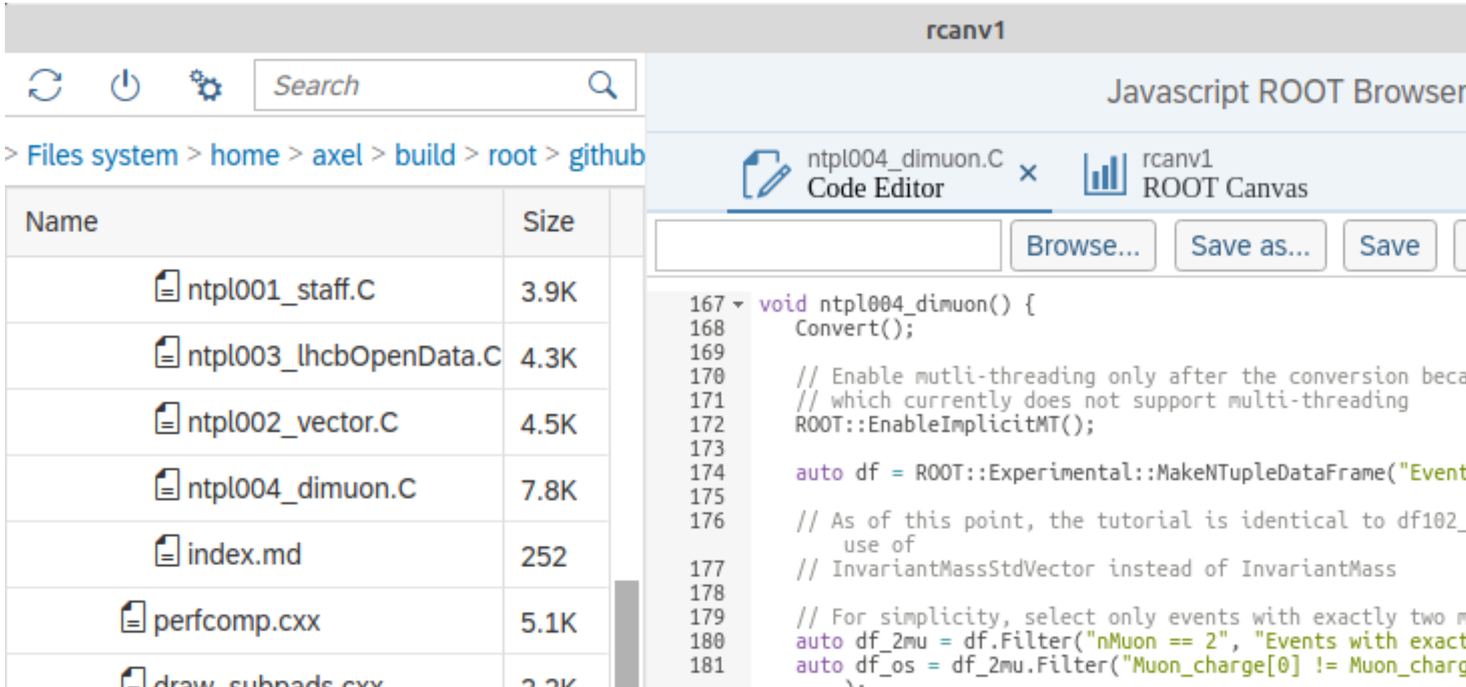
I/O is the basis

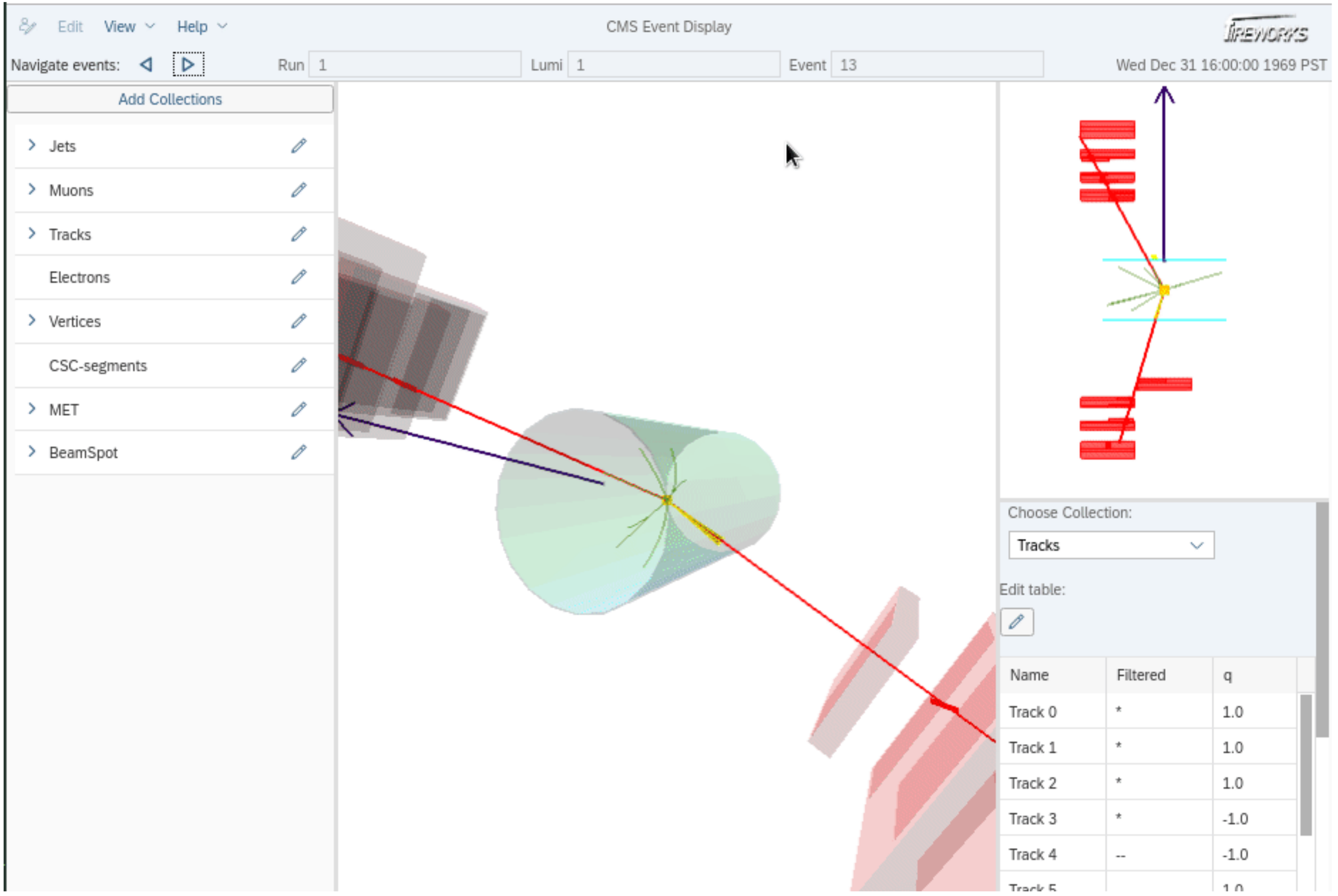


News: how far we got

WebGUI, WebGraphics, WebEve: ongoing

- HTML5 + CSS + JavaScript, using existing libraries: three.js, D3.js, OpenUI5
- Prototype development in ROOT::Experimental::
  - graphics painters, based on JSROOT: root --web
  - GUI: fit panel, RBrowser,...
  - Eve: a first geometry
    + track viewer + editor



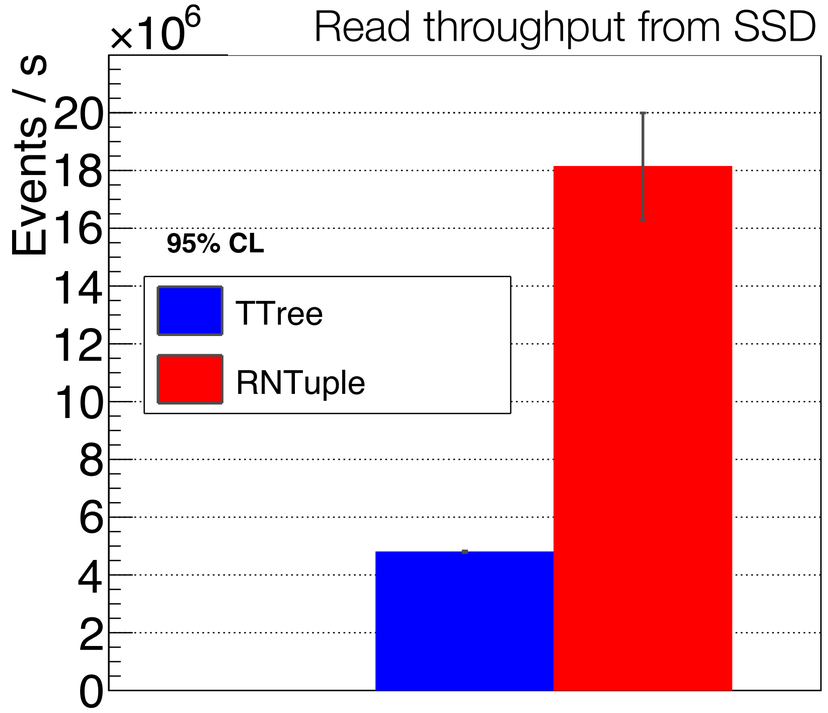

Edit
View
Help
CMS Event Display
Navigate events:
Run 1
Lumi 1
Event 13
Wed Dec 31 16:00:00 1969 PST
Add Collections
Jets
Muons
Tracks
Electrons
Vertices
CSC-segments
MET
BeamSpot
Choose Collection:
Tracks
Edit table:
Name
Filtered
q
Track 0 * 1.0
Track 1 * 1.0
Track 2 * 1.0
Track 3 * -1.0
Track 4 -- -1.0

RHist: ongoing

- Simplify: less documentation to sift through, e.g. make only 2D methods available for RH2F; separate "data" from "graphics"
- More usable: re-use axis definition for multiple histograms, circular axes ("modulo 2 pi"), counting axes ("4 jets")
- Accelerate

RNtuple: ongoing

- See this article for why HEP uses ROOT as data format. That was TTree.
- RNtuple is faster than anything else, even for simple cases
- Optimized for current use cases, e.g. tweaking compression, parallel I/O
- More usable: simpler, sturdier, type-safe interfaces
- But you might not even care: use RDataFrame (which knows to use RNtuple internally)



RDataFrame: since 2018's v6.14

- Highly efficient TTree/RNtuple analyses in a simple yet composable way, see this article
- Compact, modular, declarative code. Don't bother with reading data, iteration etc; +/- same code for C++ and Python
- Robust: type safety lets us complain if code does not match data
- Offers wonderful tutorials
- See e.g. CERN EP Software Seminar for "yes it actually works really well"

RDataFrame Example

```
ROOT::EnableImplicitMT();                    // Run a parallel analysis
ROOT::RDataFrame df(dataset);                // on this (ROOT, CSV, ...) dataset
auto df2 = df.Filter("x > 0")                // only accept events for which x > 0
             .Define("r2", "x*x + y*y");     // define r2 = x² + y²
auto rHist = df2.Histo1D("r2");              // plot r2 for events that pass the cut
df2.Snapshot("newtree", "out.root");         // write the skimmed data and r2
                                             //   to a new ROOT file
```

TMVA: since v6.20

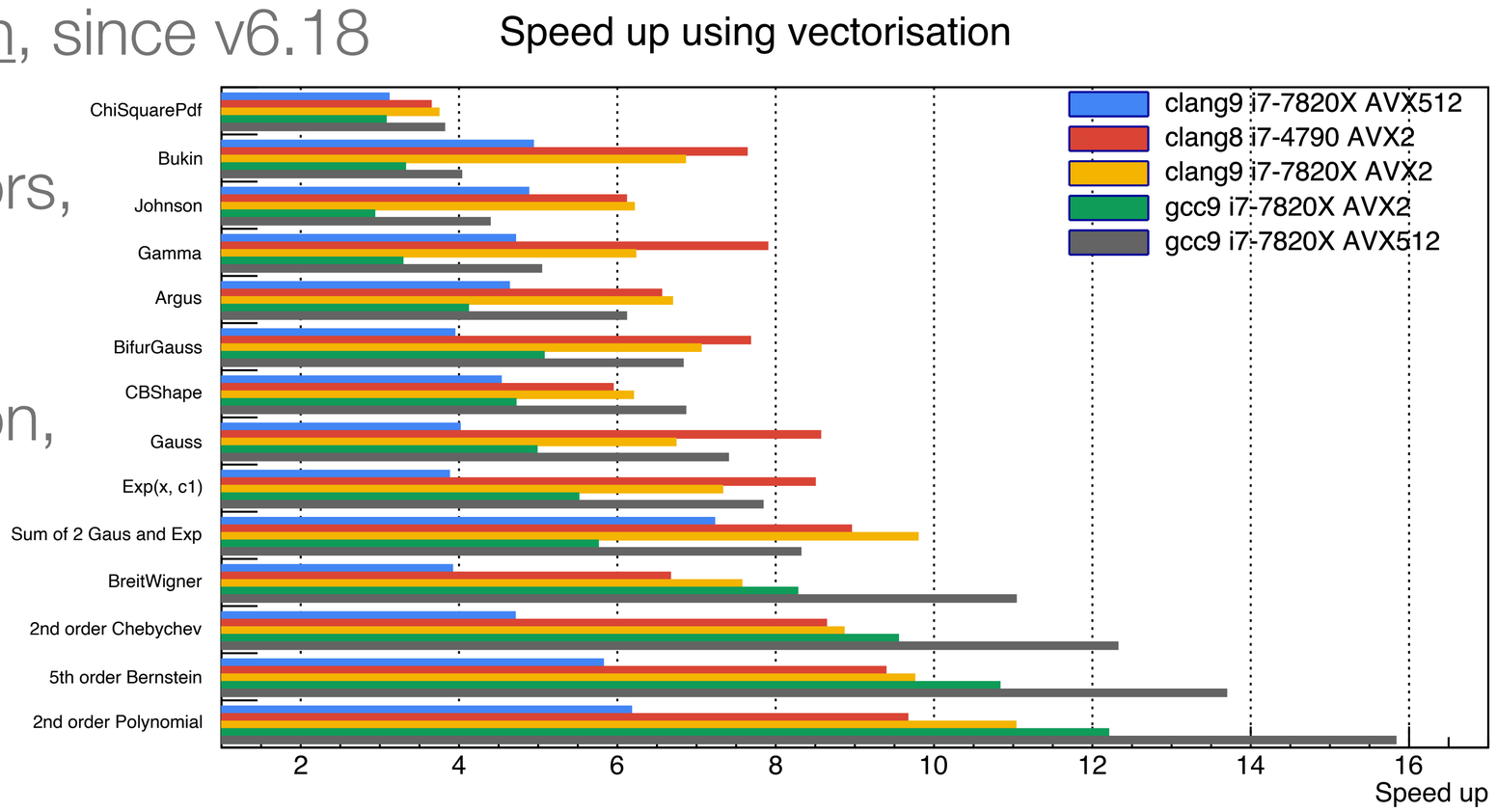
- Adapters to external backends: TensorFlow, Keras, scikitlearn; for training *and* evaluation or only training (and evaluation in TMVA proper)
  - inference benefits from TMVA's knowledge of ROOT I/O
- Employing CuDNN and C++ JIT for highest performance; ongoing work on integrating cling's automatic differentiation
- Example achievements: better numpy integration; cross validation, in parallel (multi-processing)!
- See this talk for practical examples

PyROOT: since v6.22

- All new PyROOT
  - C++ lambdas, move semantics for efficient use of C++ through Python
  - extensible pythonizations, e.g. RooFit
- ROOT built for both Python 2 *and* 3

RooFit

- New PDFs, e.g. RooJohnson, since v6.18
- Internal acceleration by factors, since v6.20 and ongoing
- Improved PyROOT integration, since v6.18 and ongoing



Other News

- `.help TTree::Draw` and version switching in ROOT doc, new website
- Added MacOS 11 + Silicon support, progressing with Windows support
- Redeclarations supported, esp useful for Jupyter / JupyROOT:

```
root [0] int i = 42;
root [1] double i = 17;
```

- Working on HPC benchmark - looking for testbeds!
- Ongoing upgrade to llvm 9: support for C++17, scheduled for 6.24 around end of 2020

Conclusion

ROOT: Back to the Future

- 1990s, ROOT started by needing to prove itself against alternatives
  - we are back in that situation, and we accept the challenge!
- Delivering a simpler, friendlier, more robust ROOT
  - address the real issues of physicists in a relevant and applicable way
  - guided by prototypes and early feedback
- **Lots** of ongoing work, for you, keep an eye on https://root.cern!

ROOT and You

- ROOT isn't just *for* you: ROOT is not MS Word, software you buy and have to deal with
- ROOT is *with* and *by* you: it's HEP's common tool, influence how it evolves!
  - If you see a bug, please report it so it gets fixed
  - If you have a fix, please hand it in for everyone to benefit
- Please complain if you are unhappy with something
- And praise has an effect, too ;-)

ROOT

- https://root.cern
- https://root-forum.cern.ch
- https://github.com/root-project
- @ROOT_Project
- https://www.linkedin.com/groups/1826455
- rootdev@cern.ch