



ROOT: Back to the Future

Axel Naumann for the ROOT Team
2018-07-12, CHEP 2018, Sofia

Outline

- Context: ROOT's role in the past, now and in the future
- Consequence and vision: what to expect from ROOT (and what not)
- Progress: where we are and what we do
- Conclusion



Analysis of Analysis Software in HENP

- ROOT is the centerpiece of HENP
 - virtually every physicist uses ROOT for analysis
 - > 1 exabyte of data in ROOT format
- Common (also graphics) language, common data format, common grounds
 - allows physicists to switch between experiments
 - enables tools applicable across experiments



Analysis of Analysis Software in HENP (2): non-ROOT

- Alternative open source "big data" software is now production grade
 - but in many aspects, "big data processing" != physics analysis
 - analysis as a program; usability (e.g. debugging); CPU efficiency; data delivery; setup-cost / scalability; event-based / stochastically independent; cannot skip data points;...
- Options are adapt us to them or them to us
 - traditionally, "adapt them to us" fails - see dev teams of Google, Facebook to influence development by *driving* it



Analysis of Analysis Software in HENP (3): In-House

- Common tools = common knowledge
 - expertise within the community, e.g. math, data format, visualization
- Allows to predict changes, adapt and benefit
 - targeted solutions (with tailored R&D) to our very own problems
- Interface with and learn from other tools
 - become part of the industry landscape, rather than suffer from efficiency hit due to non-HENP tools that are "almost good fits"



Summary: ROOT's Role in Past, Present and Future

- Past: ROOT became *the* tool for HENP data analysis
- Present: same, but computing-oriented physicists explore non-ROOT territory
- Future as ROOT wants it:
 - steer ROOT to remain competitive with alternative solutions
 - dev investment, public benchmarking, smart R&D
 - advantage: community knows its challenges and has tremendous expertise



Consequence and Vision

Investing in Development?

- ROOT dev's daily life:
 - support
 - develop small features as time allows
- Then came ROOT6 with cling
 - proved to be able to invest in multi-year dev targets, and deliver
 - within the current project, but at a cost (ROOT6: no more CINT bug fixes)



Team

- Kim Albertsson, CERN
- Guilherme Amadio, CERN
- Bertrand Bellenot, CERN
- Iliana Betsou, CERN
- Philippe Canal, Fermilab
- Olivier Couet, CERN
- Enrico Guiraud, CERN
- Sergey Linev, GSI
- Lorenzo Moneta, CERN
- Axel Naumann, CERN
- Danilo Piparo, CERN
- Oksana Shadura, Uni Nebraska Lincoln
- Yuka Takahashi, CERN+Uni Cincinnati
- Enric Tejedor, CERN
- Xavier Valls, CERN
- Vassil Vassilev, Princeton/CMS
- Stefan Wunsch, CERN

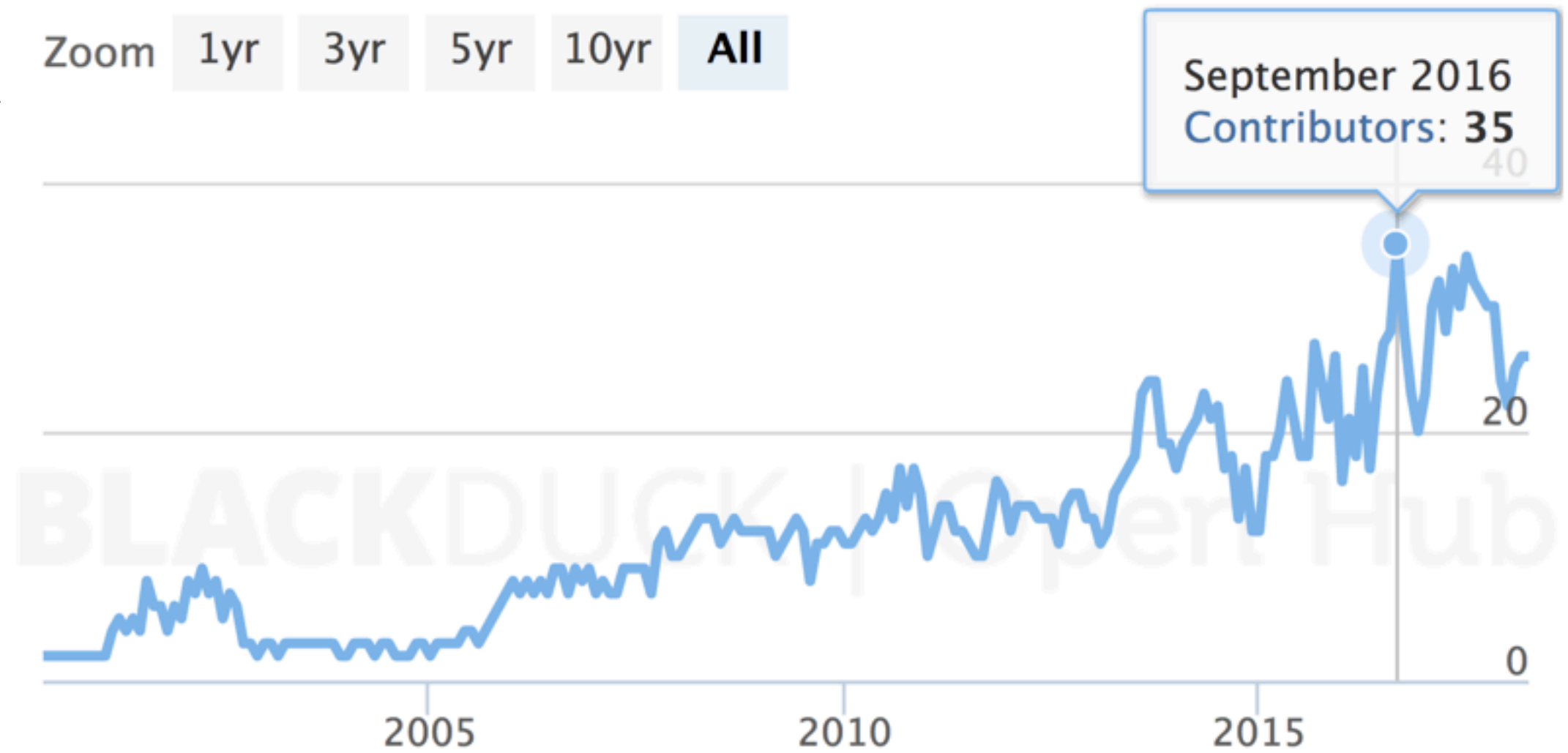


Contributions

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

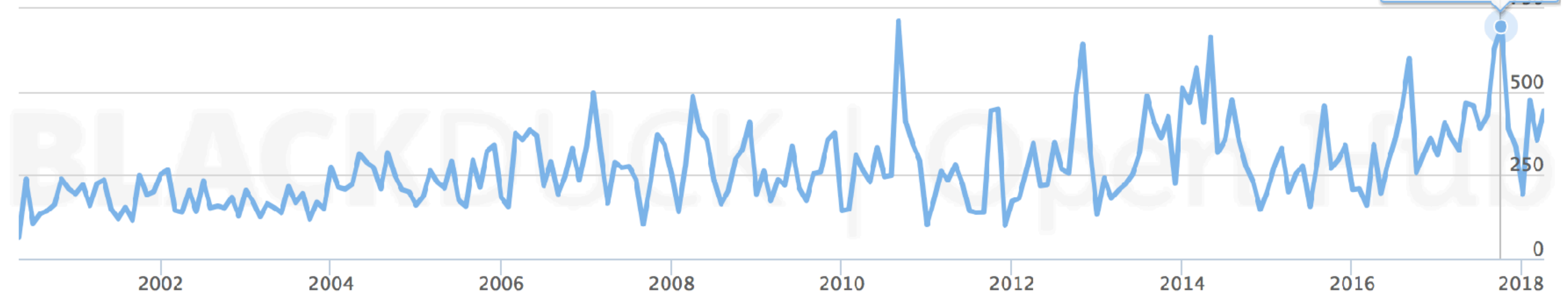
Contributors / month

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



Commits / month

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



Focus

- We want to support and maintain
- We cannot implement every useful feature *and* do major development
- Focus on big developments that matter *for users*
 - no next rewrite of the build system, no 101st graphics option



“People think focus means saying yes to the thing you've got to focus on. But that's not what it means at all. It means saying no to the hundred other good ideas that there are. You have to pick carefully.”

–Steve Jobs

Conclusion: what to expect from ROOT and what not

- Support remains the key ingredient of our work
 - Development focus on key areas, replaces "small feature" growth (reduced even for contributions)
 - histograms
 - web-based graphics / GUI / event display
 - RForest, new TTree-like container
 - RDataFrame, TMVA, Python
- } "ROOT7"
(because everything worth talking about needs a name)



Why, why those?

- Physicists invest in working around ROOT's personality
 - costs time + effort, both writing analysis and education
- Community and computing context evolved, time to take stock and change some defaults
- Histograms, WebGUI/Eve, RForest, RDataFrame/TMVA/PyROOT are the main parts of ROOT, visible to every physics analysis
- Make sure that they are homogenous, designed consistently



Progress: Where we are and what we do

Histograms: ROOT::Experimental::RHist

- Speed up by moving conditional branches to compile-time
- Massively reduced interfaces by separation of concerns: storage, binning, buffering, graphics attributes, internal operations
- Remove anachronisms: named objects, pointers, TObject, virtual functions
- Extensible: uncertainties, algorithms
- Looking for DQM adoption to guide development!

```
// Create a 2D histogram with an X axis with
// equidistant bins, and a y axis with irregular
// binning.
Experimental::TH2D hist({100, 0., 1.},
                        {{0., 1., 2., 3., 10.}});

// Fill weight 1. at the coordinate 0.01, 1.02.
hist.Fill({0.01, 1.02});
```



WebGUI, WebGraphics, WebEve



- HTML5 + CSS + JavaScript, using existing libraries: three.js, D3.js, OpenUI5
- Prototype development in ROOT::Experimental
 - graphics painters, in the context of JSROOT, painting of v6 and v7 objects, for instance histograms and primitives: root --web
 - GUI: fit panel, TBrowser
 - Eve: a first geometry + track viewer + editor



[Sergey: JSROOT](#)
[Sergey: WebGUI](#)

RForest

- Early prototype state, but telling
- Picking up the challenge of "my simplistic data model is way faster with XYZ than with ROOT!"
 - be faster than everyone else even for simple cases
 - optimize on current and remove anachronistic use cases
- Better usability by type-safe interfaces, separation of concerns (data model, serialization, backend storage, etc)
- Prototype for struct of vectors<primitive>




RDataFrame / TMVA / PyROOT

- TMVA is the data highway from ROOT files to machine learning
 - paving the highway to increase throughput; several new exits to new custom ML implementations as well as external (TensorFlow/Keras, scikitlearn etc)
- RDataFrame is the tool for writing highly efficient TTree-based (and in the future RForest-based) analyses in a simple yet composable way
 - re-use RDataFrame for TMVA
- Improve PyROOT as real analysis frontend alternative with ROOT, providing everything for quick coding and highly efficient data transfer



RDataFrame / TMVA / PyROOT

- RDataFrame is the tool for writing highly efficient TTree-based (and in the future RForest-based) analyses in a simple yet composable way
- Uses all your (for now: local) cores  [Enrico: RDataFrame](#)
- Simple: analysis becomes compact, modular code that doesn't bother about reading data, iteration etc; +/- same code for C++ in Python
- Robust: type safety lets us complain if code does not match data
- Published (i.e. out of ROOT::Experimental) in June's production **release v6.14**



RDataFrame / **TMVA** / PyROOT

- Better numpy integration
- Cross validation, in parallel (multi-processing)!
- Adapters to external backends: TensorFlow, Keras, scikitlearn
 - for training *and* evaluation or only training (and evaluation in TMVA proper)
- New ML backends in TMVA: CNN (v6.12), RNN (v6.14), work ongoing for GAN
 - using CPU (openBLAS) or GPU (CUDA)



[Kim: TMVA](#)

RDataFrame / TMVA / **PyROOT**

- Lambdas, move semantics - things we need for efficiently using C++ through simple Python code
- Better numpy integration; extensible pythonizations for new C++ code, e.g. histograms
- Re-base PyROOT on top of cppyy, Wim Lavrijsen's "fork" of PyROOT, sharing development and goals
 - started: `cmake -Dexperimental_pyroot=On`
- Provide support!



FOUR GIANT DEV AREAS for that small a team?

- Histograms, WebGUI/Eve, RForest, RDataFrame/TMVA/Python are HUGE chunks of work for a dev team of our size
- Needs long-term team members from multiple affiliations
- Prototype-driven development over years: ROOT6 taught us not to oversell nor underestimate the work, but that we can trust us and the community
- Building on that experience plus early exposure
 - learned how to trigger early feedback: worked better for RDataFrame than before for new histograms



In Other News

- LZ4 new default compression algorithm v6.14
 - fast fast fast for reading (analysis), try it out!
 - back-ported read support to most old branches: pick up their patch release!
- cling gets a CUDA backend
- ROOT becomes a package manager supporting build-on-demand
- ROOT works with C++ modules as dictionary



[Oksana: root-get](#)
[Yuka: C++ Modules](#)



In Other News (2)

- Parallelism everywhere:

- I/O, e.g. tree merging across threads

- fitting, e.g. vectorized evaluation and parallelized minimization

- TF1 fitting improvements

- RooFit is back on track, with team around Wouter Verkerke plus new CERN fellow starting September: support and development

- Continuous performance testing



[Lorenzo: Vectorization](#)

[Danilo: Future Parallel](#)

[Philippe: R/W Lock](#)

[Guilherme: BufferMerger](#)



[Lorenzo: Fitting/Modeling](#)



[Oksana: ROOTBench](#)



Conclusion

ROOT Users' Workshop

- **2018-09-10 to 13, Sarajevo**
- Feedback; present and discuss plans: **shape the future of ROOT!**
- All developers plus invited speakers, for instance **Chandler Carruth, Google!**
- **Students:** contact root-2018@cern.ch for **sponsorship!**



FUN WITH DATA!

ROOT USERS WORKSHOP 2018
SARAJEVO, SEPTEMBER 10 - 13

WORKSHOP TOPICS

- ROOT ROADMAP
- TDATAFRAME, A MODERN TTREE::DRAW()
- ON THE WAY TO ROOT 7:
GRAPHICS, HISTOGRAMS, TREES
- NEW PERSISTENCY FEATURES
- PARALLELISM IN AND WITH ROOT
- ROOT IN JAVASCRIPT
- MACHINE LEARNING IN AND WITH ROOT
- MATH TOOLS AND TECHNIQUES
- USER FEEDBACK

ORGANIZERS

- WOLF BEHRENHOF
- PHILIPPE CANAL, FNAL
- AXEL NAUMANN, CERN
- EDMOND OFFERMANN
- CINZIA PINZONI, CERN
- DANILO PIPARO, CERN

CONTACT

CERN.CH/ROOT2018
ROOT-2018@CERN.CH

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, robust ROOT
 - address the real issues of physicists in a relevant and applicable way, instead of a series of blue-sky developments
 - yes R&D, but guided by prototypes and early feedback



World Domination, Final Words, Acknowledgements

- TMVA, cling and soon other projects help us conquer the world, despite being targeted to high energy & nuclear physicists
- We sincerely enjoy delivering high quality, production grade features that have an impact, and we will continue to do that, see numerous CHEP contributions
- Only possible due to CERN, DIANA-HEP, Fermilab, Google SoC, unnamed companies, and many physicists and volunteers
- And thank **you** for reporting bugs, providing patches, helping out on the forum, for constructive criticism as well as for cheers and praise



ROOT

- <https://root.cern>
- <https://root-forum.cern.ch>
-  @root-project
-  <https://www.linkedin.com/groups/1826455>
- root-dev@cern.ch

ROOT Users' Workshop:
<https://cern.ch/root2018>

