ROOT

Axel Naumann axel@cern.ch
for the ROOT Team
Content

❖ Multithreading, Multiprocess, Vectorization
❖ I/O
❖ Math
❖ Graphics
❖ ROOT7: THist, TFile, TPad; TDataFrame
❖ Future of Graphics
ROOT In Numbers

- ROOT forum: 11’000 users, >100 new users / month, 1’300 posts / month
- 960 new bugs, 800 closed bugs since last ACAT

12 Month Summary

Aug 14 2016 — Aug 14 2017

4541 Commits
   Up + 1029 (29%) from previous 12 months

110 Contributors
   Up + 38 (52%) from previous 12 months

Source: https://www.openhub.net/p/ROOT
Diversion: Diversity

- Currently 17 team members
  - plus about 2 part-time contributors
  - plus currently 7 students (GSoC and CERN summer students)
- Out of those 26, 1 female. That’s 96% male. (CERN jobs: 100% male)
- we can improve in other dimensions, too - e.g. nationality - but nothing as striking
Three days of “what do we have, what could we need in 5 years”? Feedback:

- ascendancy of python (PyROOT support issue); C++ to stay main component
- ROOT I/O still the keystone of analysis
- RooFit, TMVA and interfacing with ML tools are also essential
- leverage (even better) external tools, developing bridge and connectors
- rise of the functional/declarative programming in analysis (i.e. decouple what you want from how you get it) - see TDataFrame later
Multithreading, Vectorization

- **TDataFrame!** (see later)
- **ROOT spawns parallel (tbb) tasks; e.g.**
  - I/O: Brian’s talk on Thursday (track 1!)
    "Increasing Parallelism in the ROOT I/O subsystem"
  - poster "Parallelisation and Vectorisation of ROOT Fitting classes"
- **Vectorization through interfaces: poster "Speeding up experiments software with VecCore, a portable SIMD library"**
Vectorization and Multithreading

- Currently aiming for matrix vectorization on/off, MT on/off
- Cost: increasing interface complexity, code growth
  - example adapted from VecGeom

```cpp
// scalar
void DistanceToIn(Vector3D<double> const &point,
                   Vector3D<double> const &direction,
                   double &distance);

// vector types
VGPRECISION DistanceToInVec(Vector3D<VGPRECISION> const &point,
                            Vector3D<VGPRECISION> const &direction);
```

- What are our options?
I/O

- Studying I/O performance characteristics: Jakob’s talk on Thursday (track 1!) “A quantitative review of data formats for HEP analyses"
- Compression format LZ4 (soon default?) reads much faster, ~15% larger files
  - see Brian’s poster "Optimizing ROOT I/O for Analysis"
- New, faster zlib implementation
- TBufferMerge: parallel writing into a TTree!
- Ongoing work with moving dictionary parts to clang C++ modules (Raphael, Vassil)
Math

- Multithread fitting: TH1::Fit(function, “MULTITHREAD”) uses all cores
- Vectorized function evaluation in TF1 with vectorized user functions/\n  functors, using Vc and VecCore, for fitting histograms, graphs, trees, etc. (but
  not Roofit)
- Easier creation of TF1 objects
- upcoming: vectorized TF1 objects

```
// Composition:
TF1 comp("sin( f1(x) )");
// Sum of normalized functions:
TF1 nsum("NSUM([A]*gaus, [B]*expo)",
    xmin, xmax);
// Convolution:
TF1 conv("CONV(expo, gaus)", xmin, xmax);
```
New features, most notably:

- new Deep Neural Network (working in parallel in CPU or GPU)
- Interfaces to Keras (PyKeras) which can use Theano or Tensorflow
- Improved support for multi-class classification

Upcoming (PR under evaluations):

- parallelization of Envelop methods (CrossValidation, Variable Importance and HyperParameter optimizations)
- internal parallelization of BDT
- deep auto-encoder for variable dimensionality reductions
- new framework to support existing standard DNN, Convolutional Layers (for CNN), Recurrent layers (for RNN)
Two major feature requests implemented

- automatic palette colors, e.g. line: `hist->Draw("PLC")`
- auto-placement, e.g. `canvas->BuildLegend()`
- “do the right thing” options!

Plus constant flow of smaller improvements, e.g. “BOX1” TH3 option
ROOT 7’s New Interfaces

- Fast, focused, safe, thread-safe, transition-friendly
- Backward incompatible, but sustainable new interfaces
- Exist since 6.08
  - in `ROOT::Experimental`
  - e.g. `#include <ROOT/THist.hxx>`
Current Features

- **THist, TFile, TPad / TCanvas** (new!)
  - with explicit “pixel” / “normal” / “user” coordinates
- **TDataFrame**: a better `TTree::Draw()`, see next talk by Guilherme!
- Deciding on new interface personality: ownership, separation of simple / advanced interfaces, safer code through array spans + `unique_ptr` +…
- Features added continuously
- Release separately as use demands and stabilization allow
Example: THist

- Fast: less virtual interfaces, more inlined, more bulk data operations
- Safe: 1D histogram has no `hist->GetBinError(x, y)`
- Transition: keeps most interface names `TH1F::Fill()`, `TH2D::GetEntries()`
- Thread-safe: no directory registration, no raw pointers, explicit ownership
- Focused: no `SetLineColor()`
- Yet composable and configurable for experts: statistics, storage
Future of Graphics

- Replace native UI with platform independent web standard
  - HTML5, CSS, JavaScript and web browsers
  - d3.js and ROOTJS
  - local sockets and HTTP web sockets
  - OpenUI5 for GUI
- Both a good time to start this (most ingredients exist) and early (e.g. “browser as a library”)
And It Works!

$ root -l tutorials/v7/draw_v6.cxx
root [0]
Processing tutorials/v7/draw_v6.cxx...
Info in <TClivetweb::Create>: Starting HTTP server on port 9504
Windows needs cling support

Few but very valuable customers

Lots of progress, still hoping for rudimentary support in 2017

until then, virtual machines or containers: https://hub.docker.com/r/rootproject/root-ubuntu16/

ROOT’s patches to llvm have been upstreamed (some of clang, too)
Conclusion - Main Evolution Items

- I/O: LZ4, vectorized zlib; TTree merging
- Parallelization: math, I/O, analysis
- Vectorization: math, user interfaces
- Math: see above, plus RooFit, TMVA with GNN, RNN
- Graphics: using web technology
Conclusion

❖ New interfaces == new momentum, even more under discussion!
❖ Several new non-CERN team members, especially visible in I/O
❖ TDataFrame!
Your Core ROOT Team

Xavi [1], Vassil [2], Sergei[3], Raphael [4], Philippe [5], Pere [1], Olivier [1], Oksana [6], Martin [1], Lorenzo [1], Guilherme [1], Gerri [1], Enrico [1], Enric [1], Danilo [1], Bertrand [1], Axel [1]

1: CERN
2: Princeton University
3: GSI
4: Chalmers University
5: Fermilab
6: University of Nebraska

Plus several regular + essential contributors!
@ROOT

- https://root.cern
- https://root-forum.cern.ch
- https://root.cern/bugs