The State of ROOT

http://root.cern

Axel Naumann for the ROOT Team

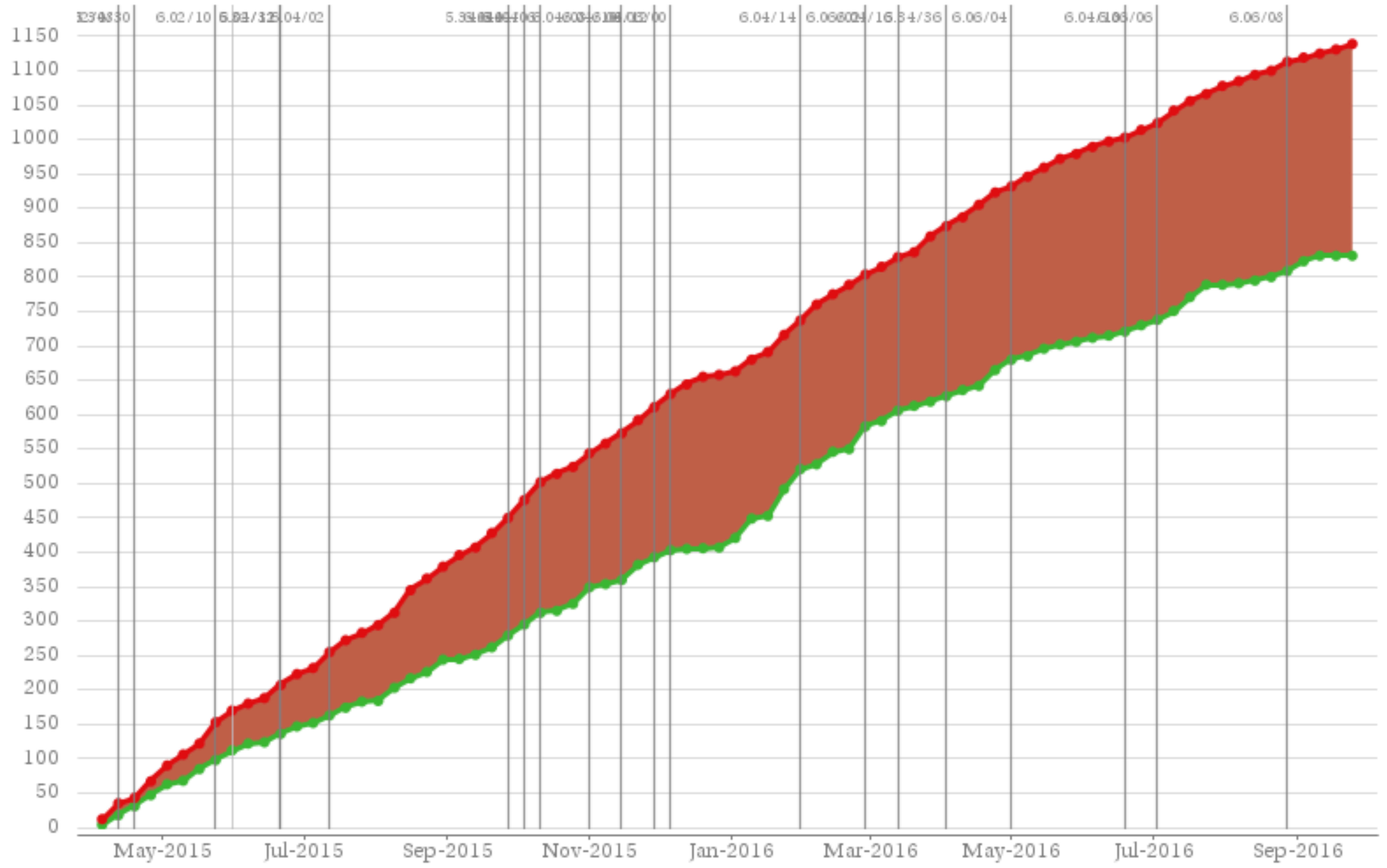
2016-11-02, ALICE Offline Week

The last 18 months

- v6.04: June, 2015; v6.06: Dec, 2015; v6.08: Oct, 2016; ROOT 5 frozen

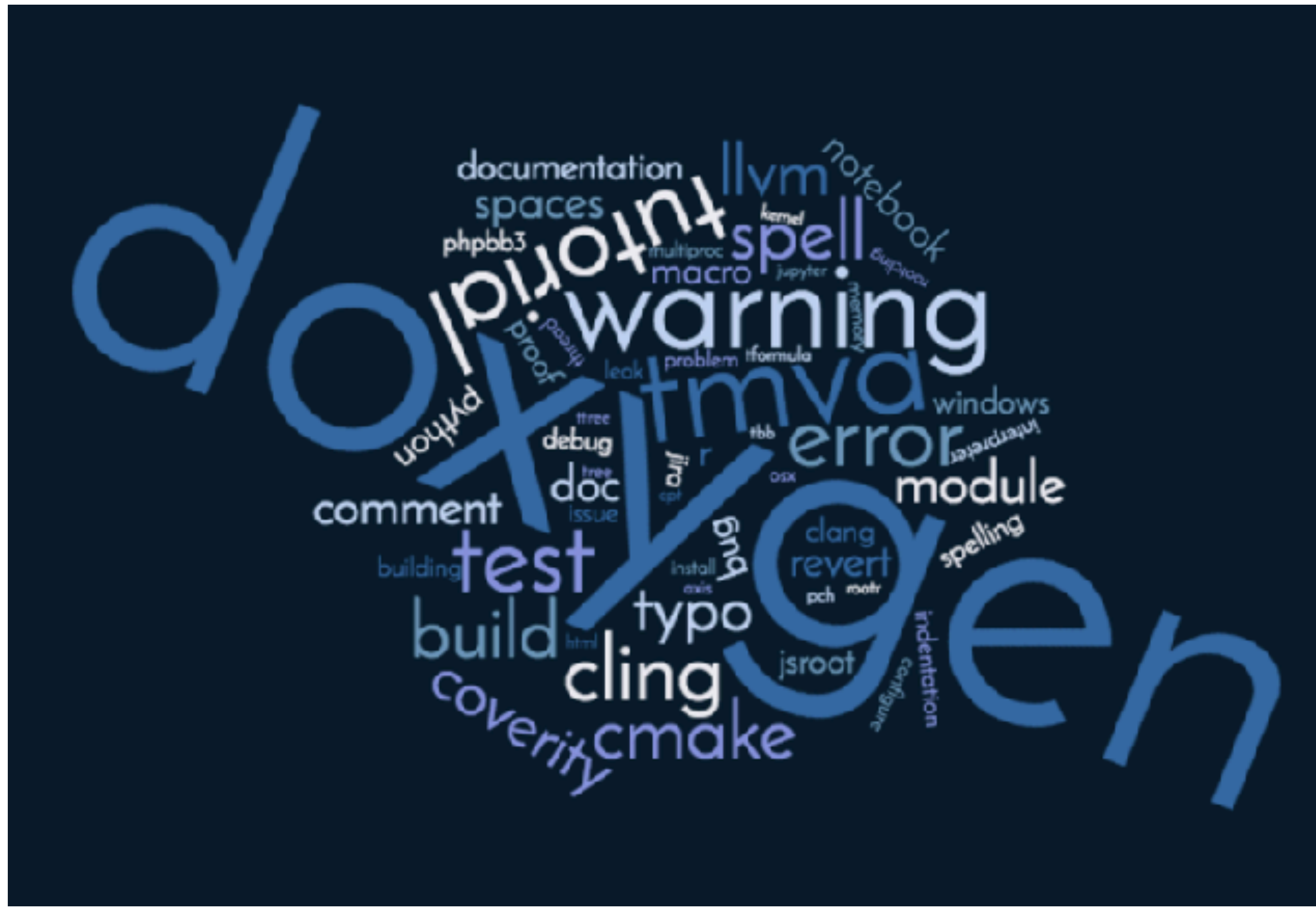
- 1150 new bugs, 850 resolved

- Forum: 14k posts in 18 months, now 10k users



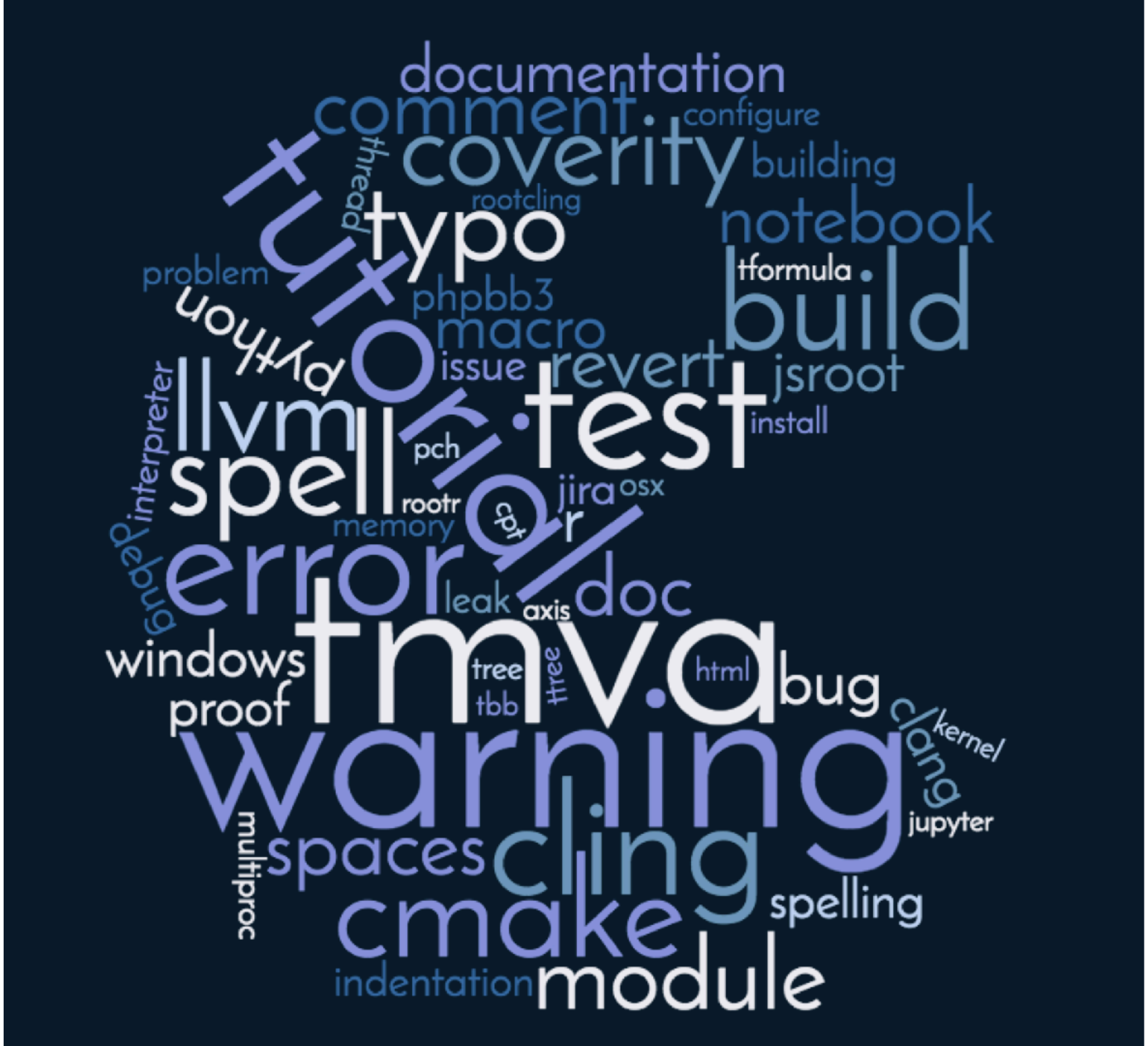
The last 18 months (2)

- >5k commits
- About 90 authors
- >200 pull requests @ github.com/root-mirror
  - with 34 open ones



Word cloud of commit logs

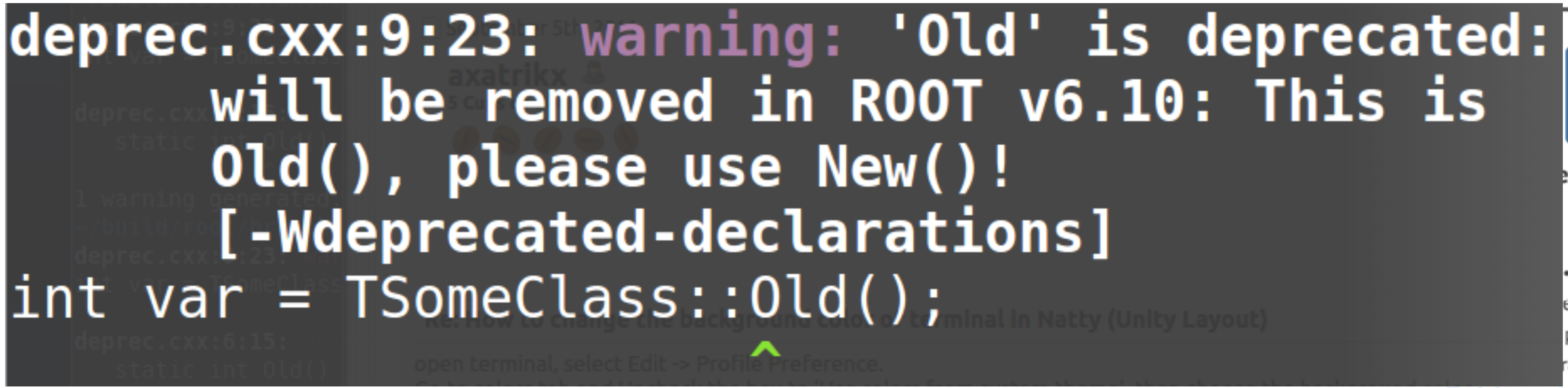
Word cloud of commit logs minus "doxygen"



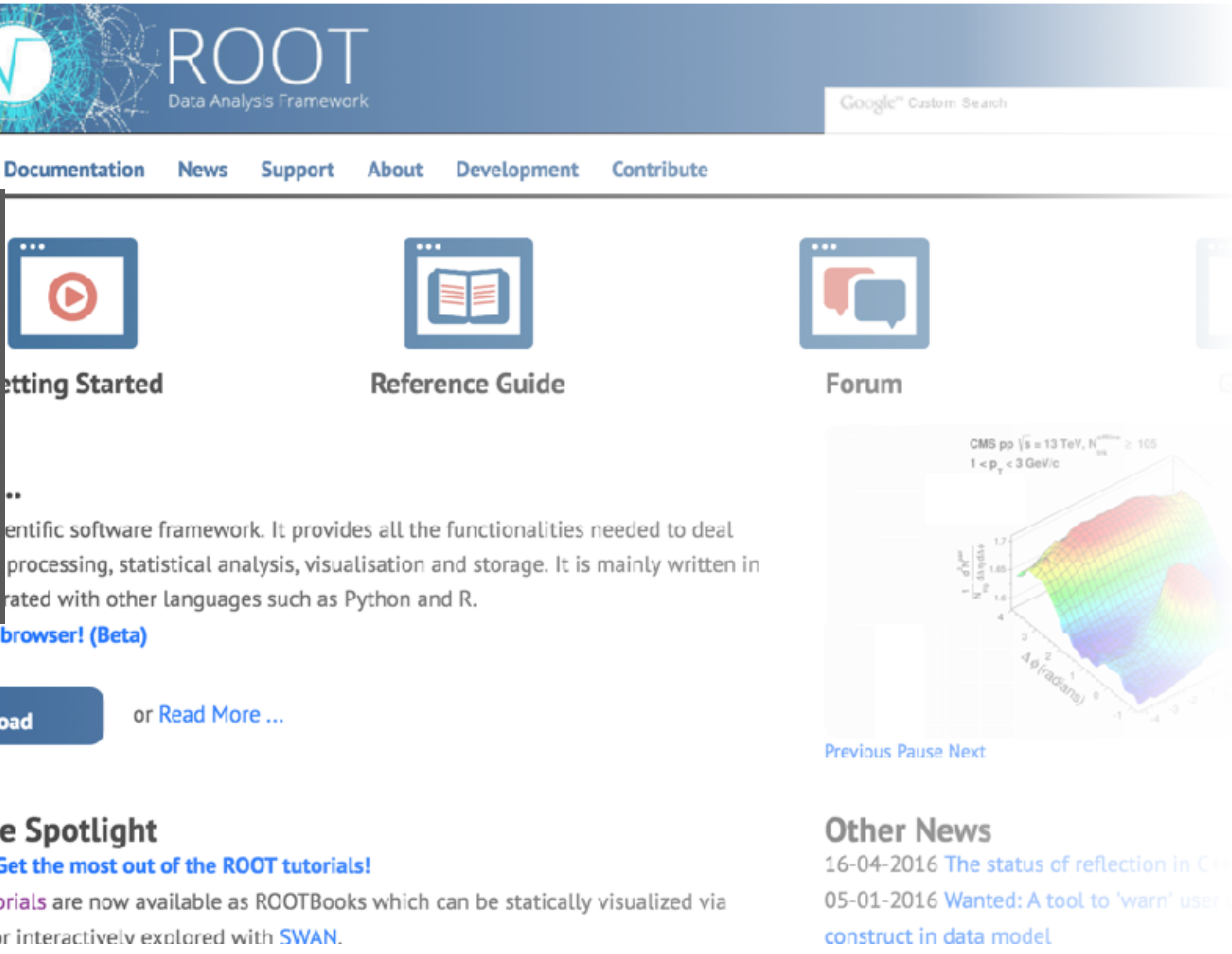
News

- New-style interfaces: #include “ROOT/TFoo.hxx”, ROOT::TFoo

- Started to deprecate interfaces



```
deprec.cxx:9:23: warning: 'Old' is deprecated:
     will be removed in ROOT v6.10: This is
     Old(), please use New()!
     [-Wdeprecated-declarations]
int var = TSomeClass::Old();
                      ^
```



ROOT
Data Analysis Framework

..

entific software framework. It provides all the functionalities needed to deal
processing, statistical analysis, visualisation and storage. It is mainly written in
rated with other languages such as Python and R.

Try it in your browser! (Beta)

or Read More ...

Under the Spotlight

05-09-2016 Get the most out of the ROOT tutorials!

All ROOT tutorials are now available as ROOTBooks which can be statically visualized via NBViewer or interactively explored with SWAN.

Other News

16-04-2016 The status of reflection in C

05-01-2016 Wanted: A tool to 'warn' use construct in data model

- New web site

Core

- Pulling in std::string_view, std::array_view, std::apply from the future

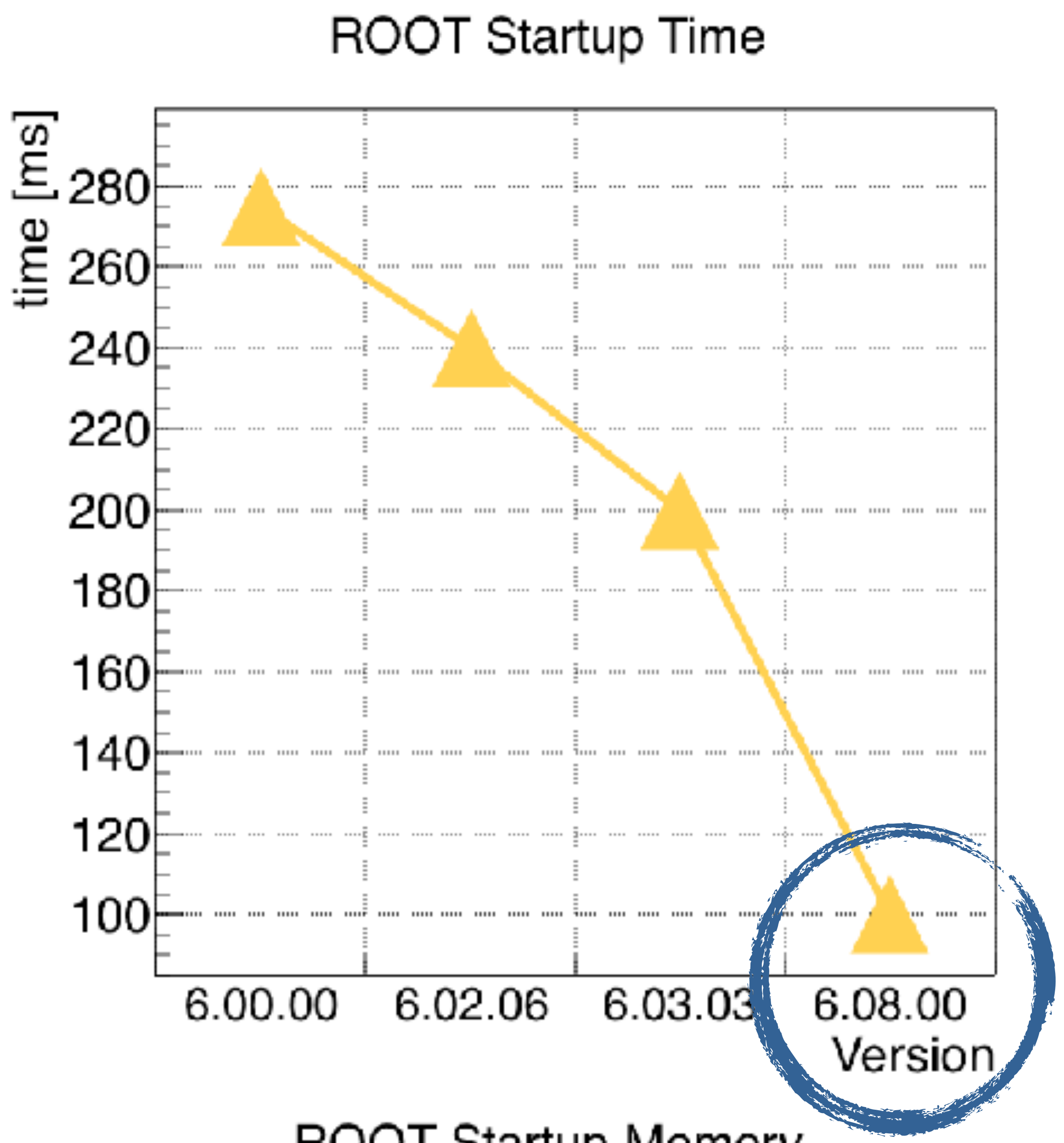
- Support for builds with C++ modules (clang only) - thanks, US-CMS!

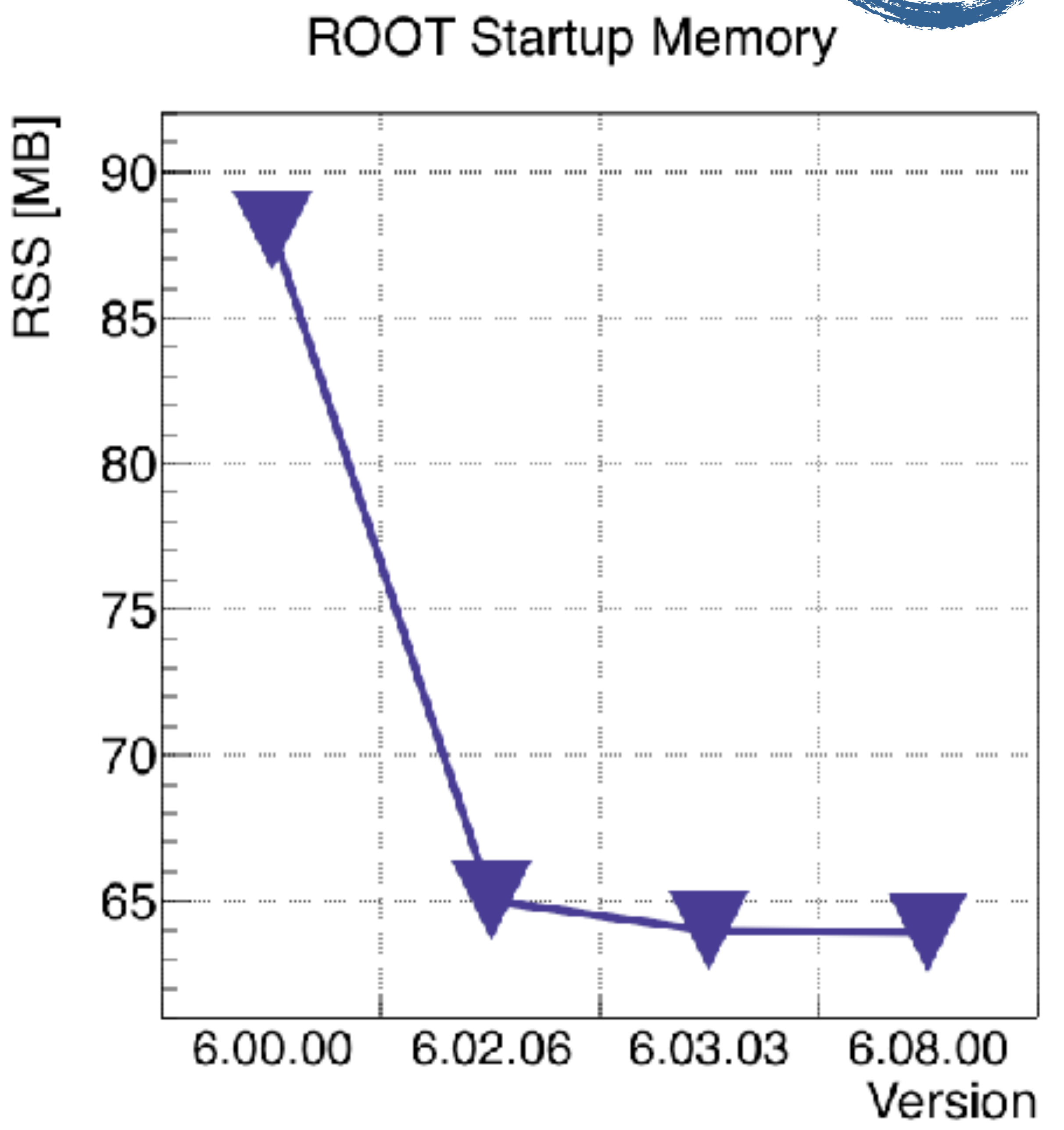
- ROOT::TSeq: integer range a la python xrange;
  ROOT::TThreadedObject: one object per thread, merge on demand

- **Still coming**: thread safe ROOT memory management ("list of cleanups"), const == thread safe

I/O, Dictionaries

- New C++ types for I/O: unique_ptr, array, tuple (already had unordered_*, forward_list)
- Memory reduction; more thread safety
- **Still coming**: zero-copy I/O, byte swapping, atomic<T>, shared_ptr, parallel write, schema evolution write rules, improved type system, use JIT, I/O for interpreted classes, improve compression of branch of unsplit collections, reduce overhead for deep hierarchy, thread safe (slower) version of TFile

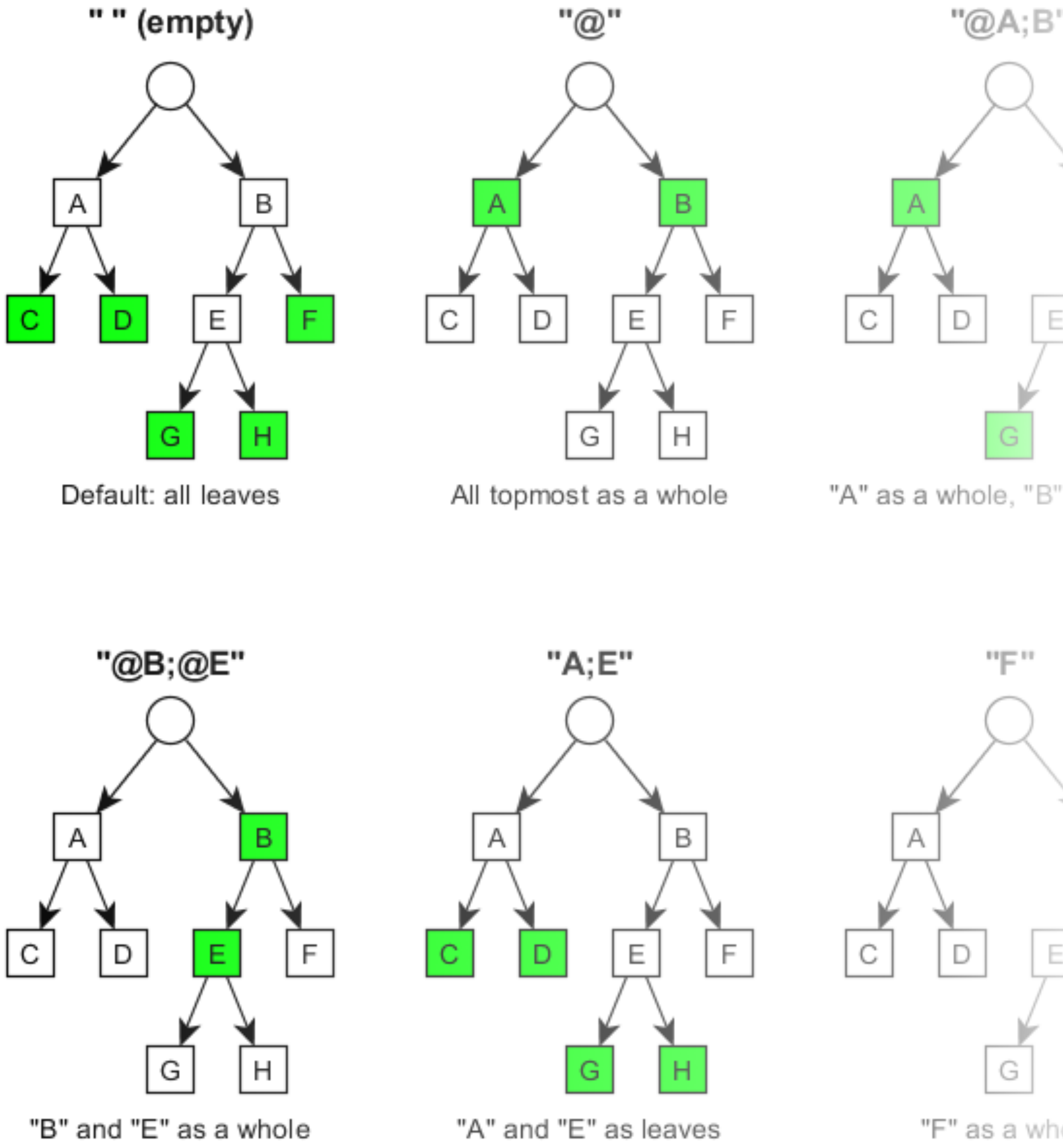




Tree

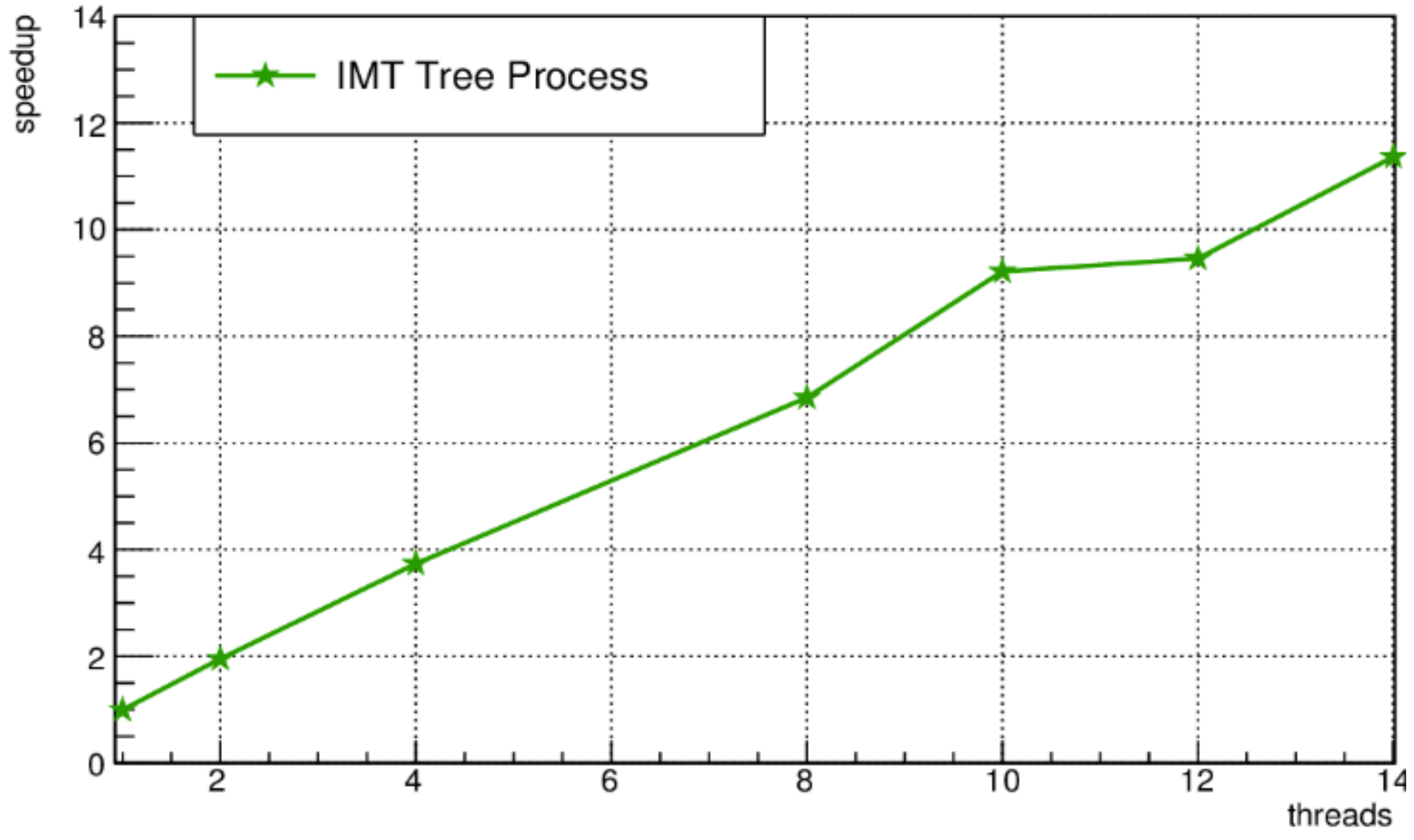
- MakeSelector uses TTreeReader (thank you, summer student Akos!)

- **Still coming**: parallel basket compression, compression of individual entries, production quality Parallel Tree Merger, faster + smarter TTreeFormula (using JIT), TTree::SetBranchAddress matching input type and branch content, vectorization in TTree::Draw and TTree interfaces



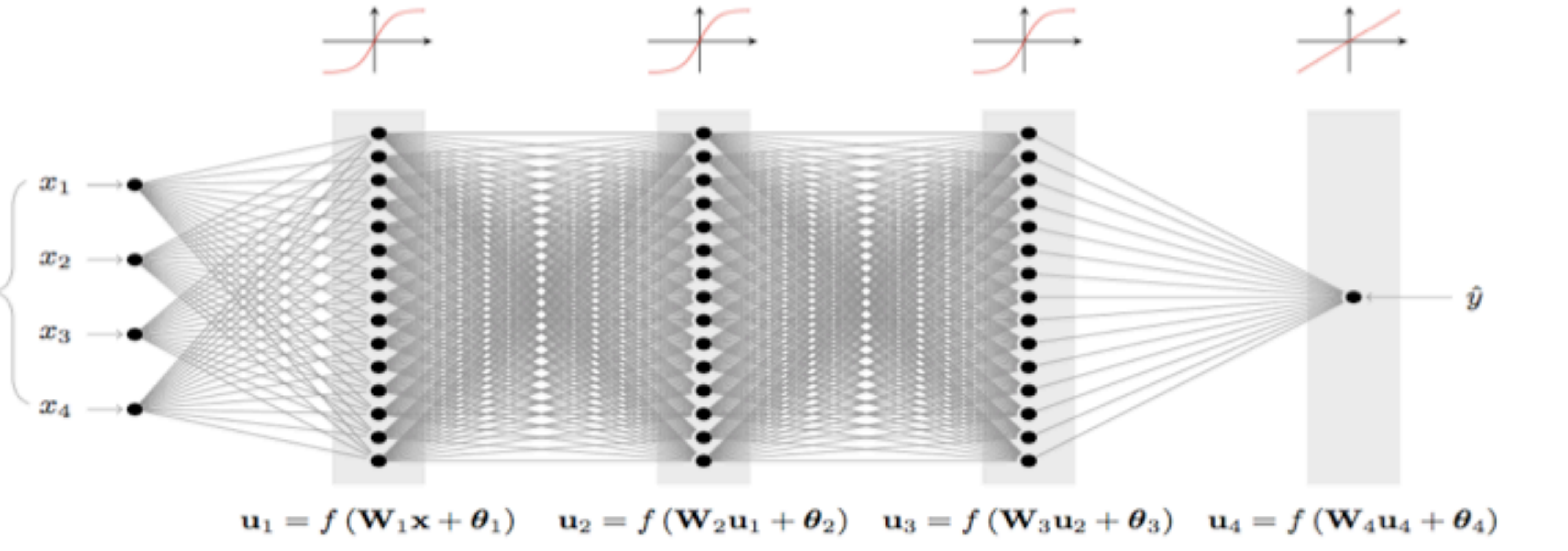
Parallel ROOT

- Multi-threaded TTree::GetEntry(), lambda-on-a-tree
  - enable with -Dimt=ON and ROOT::EnableImplicitMT()
- MultiProc tree analysis

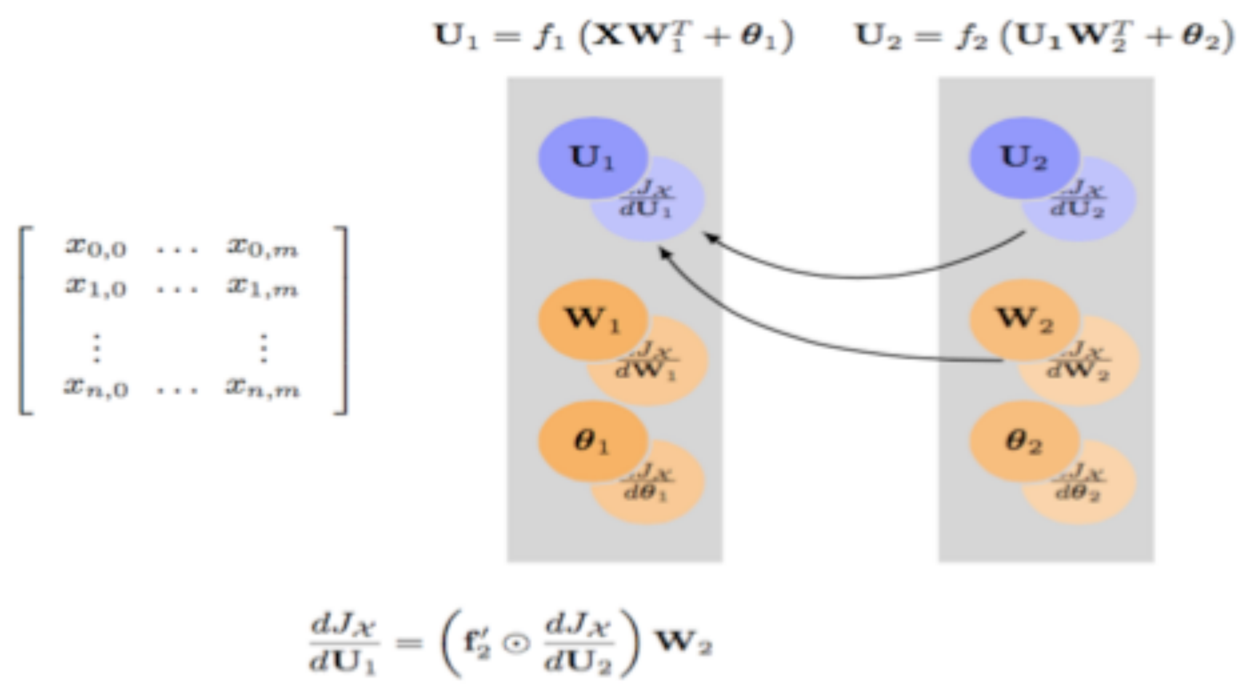


- **Still coming**: parallel analysis with functional programming, focusing on Python

Math


x_1
x_2
x_3
x_4
$\mathbf{x}$
$\hat{y}$
$\mathbf{u}_1 = f(\mathbf{W}_1\mathbf{x} + \boldsymbol{\theta}_1)$
$\mathbf{u}_2 = f(\mathbf{W}_2\mathbf{u}_1 + \boldsymbol{\theta}_2)$
$\mathbf{u}_3 = f(\mathbf{W}_3\mathbf{u}_2 + \boldsymbol{\theta}_3)$
$\mathbf{u}_4 = f(\mathbf{W}_4\mathbf{u}_4 + \boldsymbol{\theta}_4)$

- TMVA: deep learning in TMVA, RMVA, PyMVA (scikit-learn), JsMVA 20/90 authors! First time on GPU! Thank you, summer students + GSOCs: Omar +Simon + Attila!

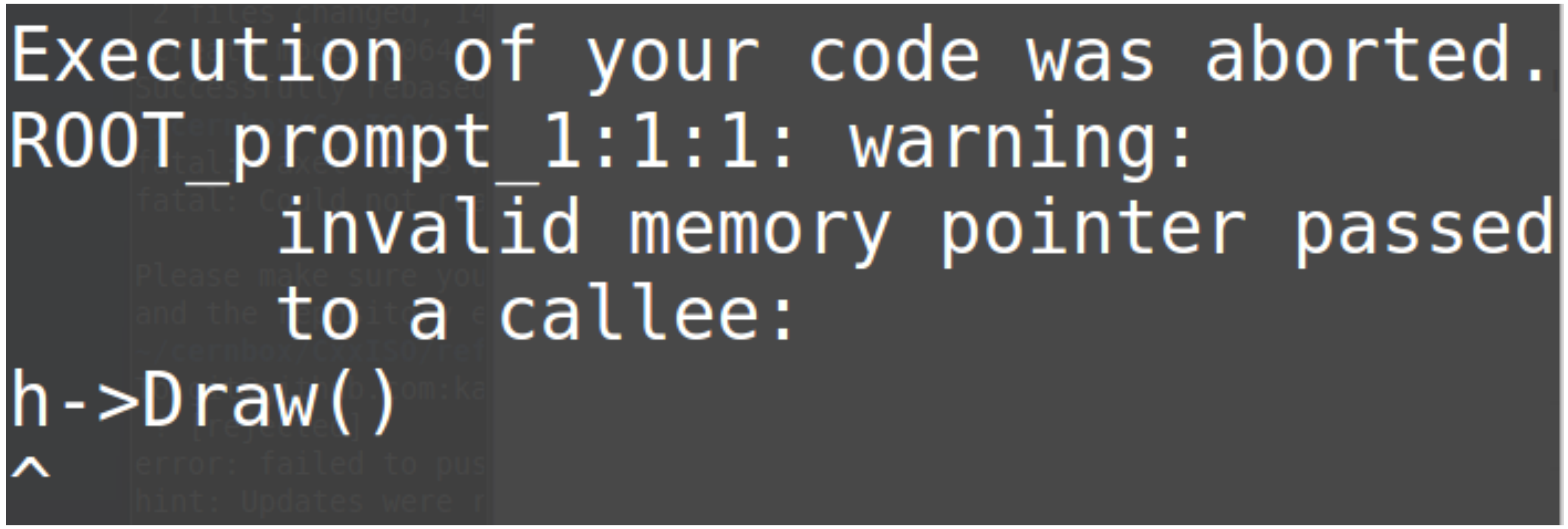

$\mathbf{U}_1 = f_1(\mathbf{X}\mathbf{W}_1^T + \boldsymbol{\theta}_1)$
$\mathbf{U}_2 = f_2(\mathbf{U}_1\mathbf{W}_2^T + \boldsymbol{\theta}_2)$
$x_{0,0}$ … $x_{0,m}$
$x_{1,0}$ … $x_{1,m}$
$x_{n,0}$ … $x_{n,m}$
$\mathbf{U}_1$
$\frac{dJ_\mathcal{X}}{d\mathbf{U}_1}$
$\mathbf{W}_1$
$\frac{dJ_\mathcal{X}}{d\mathbf{W}_1}$
$\boldsymbol{\theta}_1$
$\frac{dJ_\mathcal{X}}{d\boldsymbol{\theta}_1}$
$\mathbf{U}_2$
$\frac{dJ_\mathcal{X}}{d\mathbf{U}_2}$
$\mathbf{W}_2$
$\frac{dJ_\mathcal{X}}{d\mathbf{W}_2}$
$\boldsymbol{\theta}_2$
$\frac{dJ_\mathcal{X}}{d\boldsymbol{\theta}_2}$
$\frac{dJ_\mathcal{X}}{d\mathbf{U}_1} = \left(\mathbf{f}_2' \odot \frac{dJ_\mathcal{X}}{d\mathbf{U}_2}\right)\mathbf{W}_2$

- MixMax random number generator, see Computer Physics Communications 196 (2015) 161 / http://arxiv.org/abs/1403.5355

- **Still coming**: vectorization + parallelization of fitting; TMVA + Keras / Theano

Interpreter

- Exception support, GCC5 ABI, *pointer validity check, more value printing



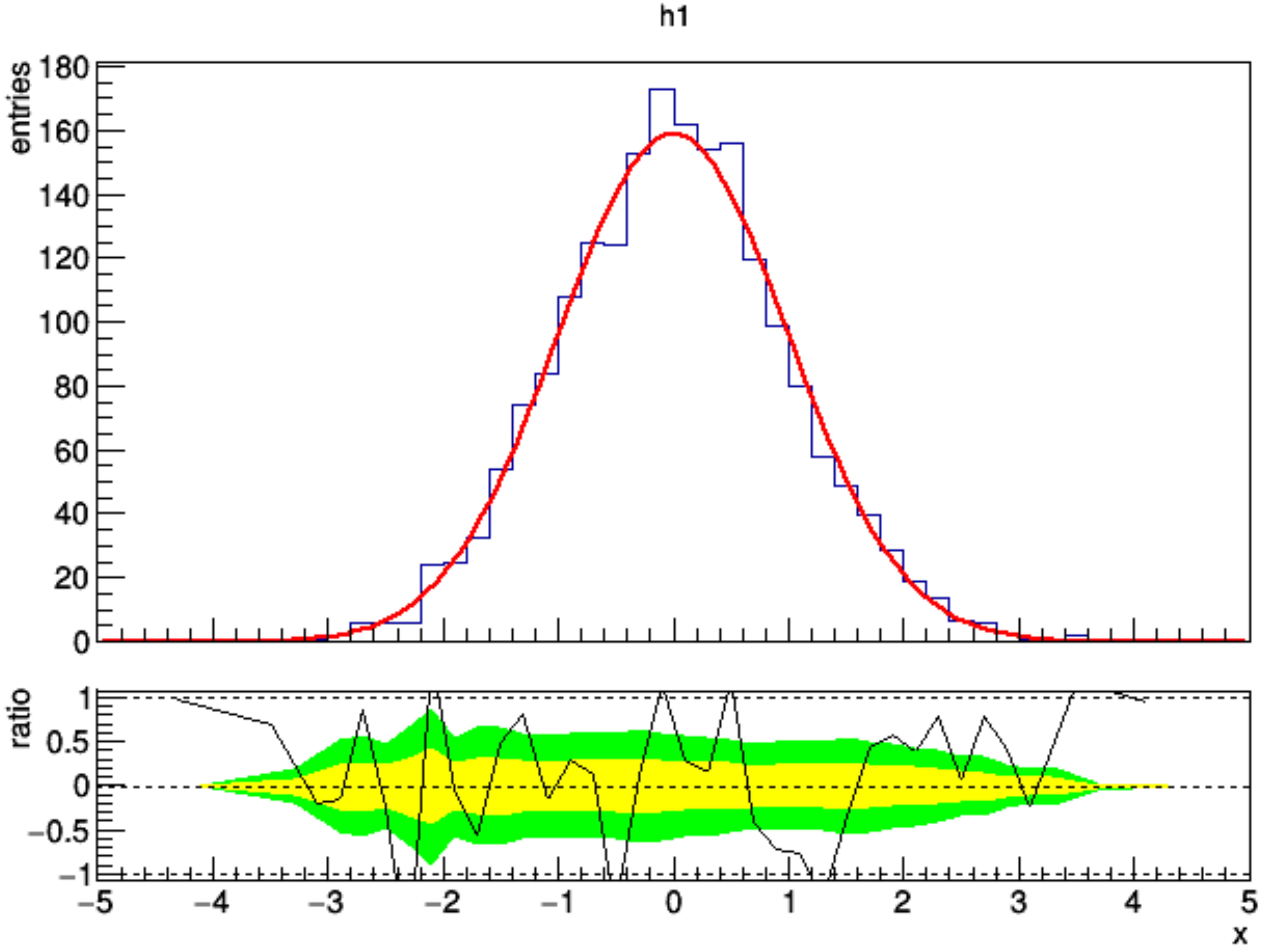
```
Execution of your code was aborted.
ROOT_prompt_1:1:1: warning:
      invalid memory pointer passed
      to a callee:
h->Draw()
^
```

- Python 3 support

- **Still coming**: unloading, Windows, even lazier compilation (a la Julia), modules for dictionaries

Graphics

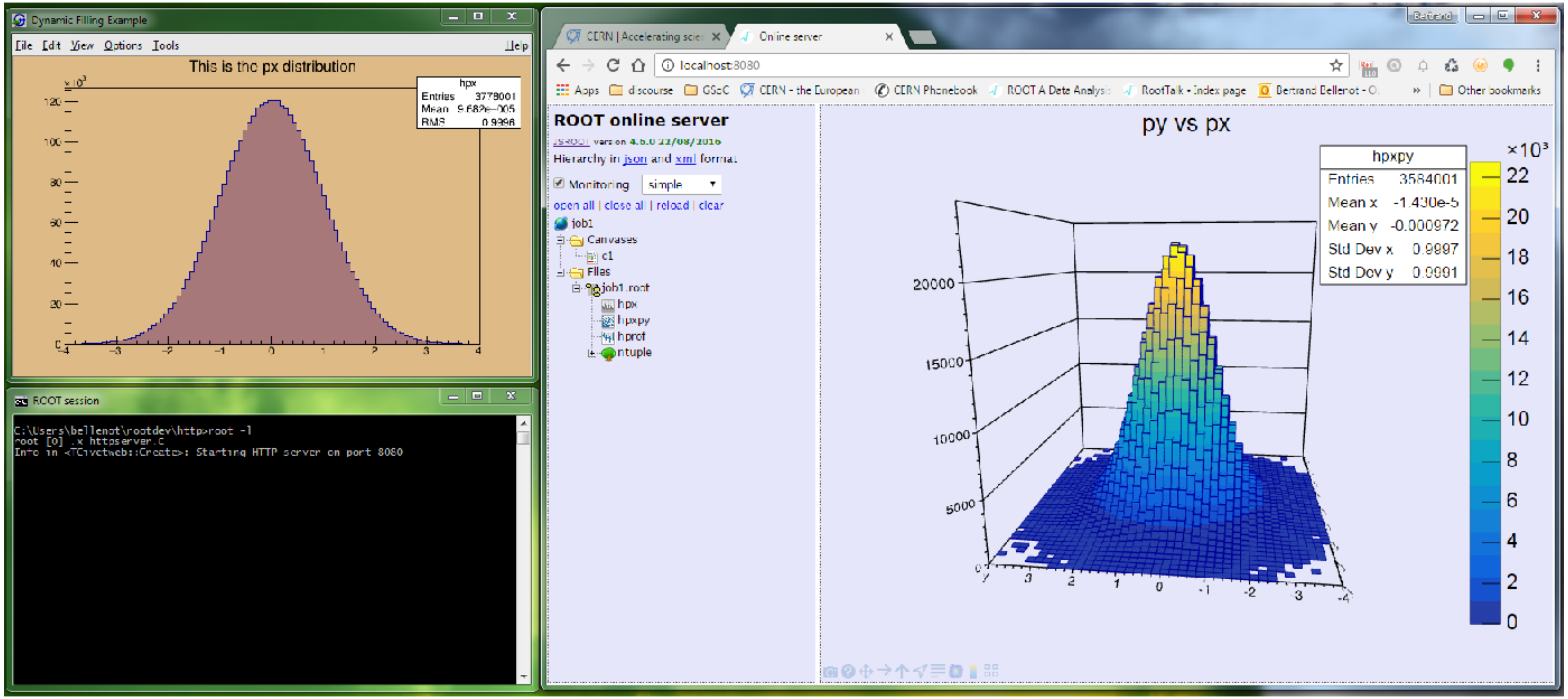
- TRatioPlot (thank you, summer student Paul!)



- **Still coming**: auto-coloring of multiple histograms, auto-placement of legend, fixed size fonts by default; switching to a web-based UI

Net

- THttpServer: bi-directional communication enabling interactive use of ROOT from the browser



```
C:\Users\bellenot\rootdev\http>root -l
root [0] .x httpserver.C
Info in <TCivetweb::Create>: Starting HTTP server on port 8080
```

ROOT online server

JSROOT version 4.6.0 22/08/2016

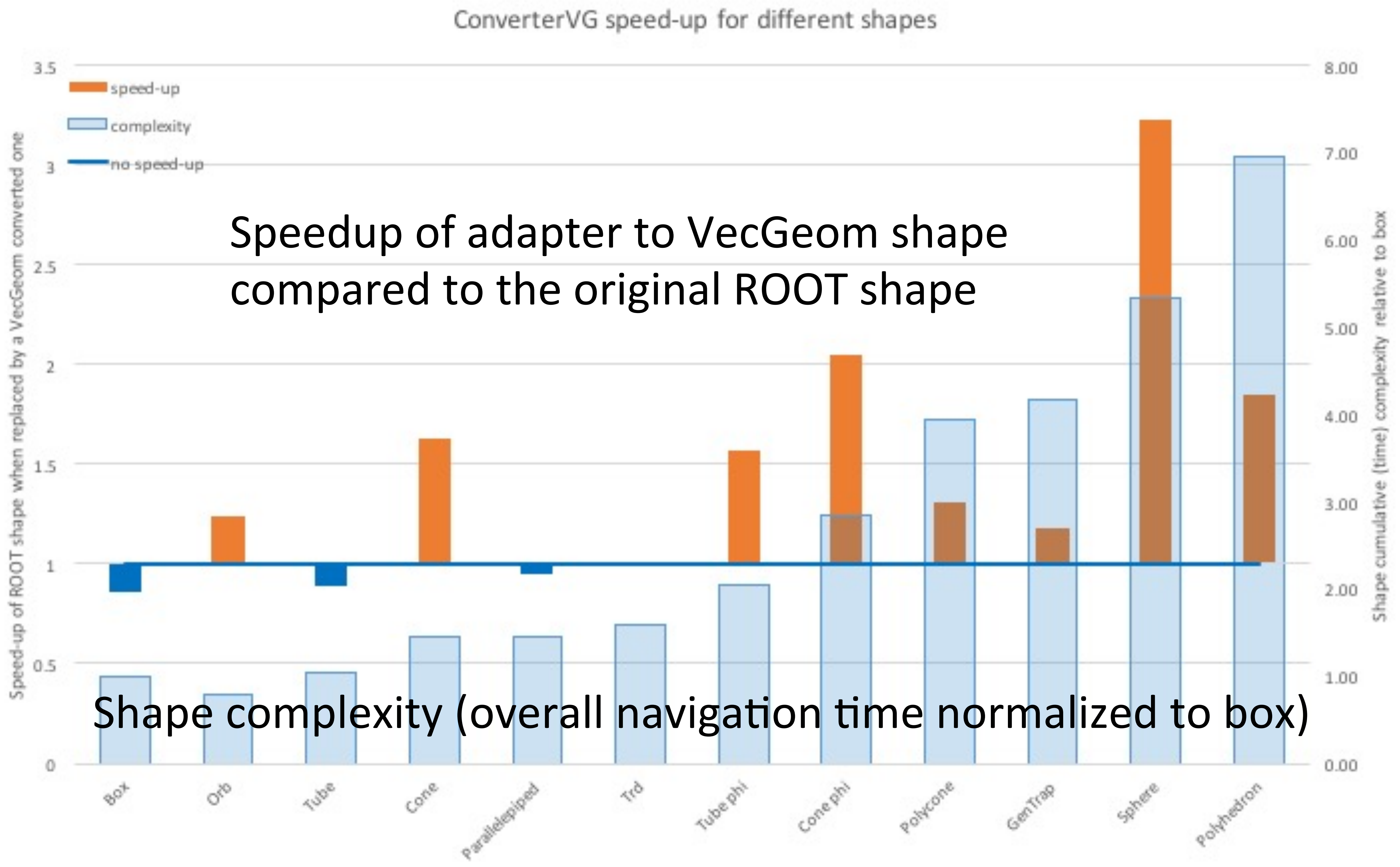
Hierarchy in json and xml format

☑ Monitoring simple ▾

open all | close all | reload | clear

Geometry

- VecGeom adaptor / conversion
- GDML expression parsing (JITted), treatment of global constants

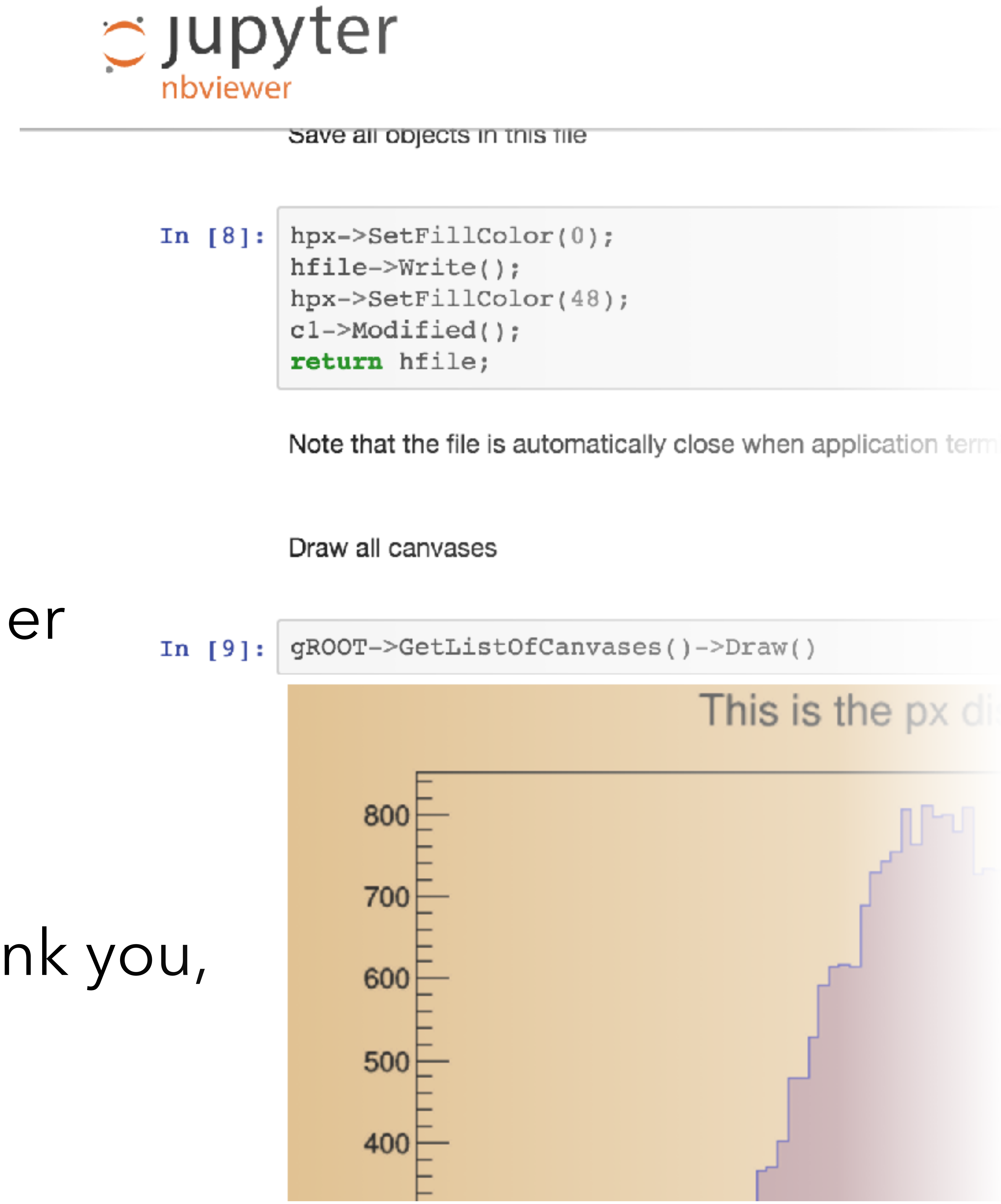


Speedup of adapter to VecGeom shape compared to the original ROOT shape

Shape complexity (overall navigation time normalized to box)

Usability

- Doxygen documentation
- Jupyter-ized tutorials (thank you, summer student Pau!)
- rootdrawtree -i *.root -o out.root -hs ‘histphi=jets.phi’ / TSimpleAnalysis (thank you, summer student Luca!)



Save all objects in this file

```
In [8]: hpx->SetFillColor(0);
        hfile->Write();
        hpx->SetFillColor(48);
        c1->Modified();
        return hfile;
```

Note that the file is automatically close when application term

Draw all canvases

```
In [9]: gROOT->GetListOfCanvases()->Draw()
```

CMake

- Many “built-in” external packages
  - downloaded + built with a simple -Dvc=ON etc
- configure/make now deprecated
  - about 1000 changes to CMake files since Okinawa
- **Still coming**: cross-build

New Interfaces: Status+Goals

- In ROOT::Experimental
  - TH1F and friends incl Draw(), TFile and friends
- Robust, concise
  - regular ownership, type-safe, thread-safe, compiler checkable, separation of public and internal parts, improved speed

WHEN YOU GET SOMETHING, IT'S NEW AND EXCITING. WHEN YOU HAVE SOMETHING, YOU TAKE IT FOR GRANTED AND IT'S BORING.

New Interfaces: Examples

```
TAxisConfig xaxis{10, 0., 1.};
auto hist = make_shared<TH2D>(xaxis);

hist->Fill({0.1, 0.2}, weight);

Canvas canv;
canv.Draw(hist);

TFilePtr file
  = TFile::Recreate("hist.root");
file->Write("TheHist", hist);
```

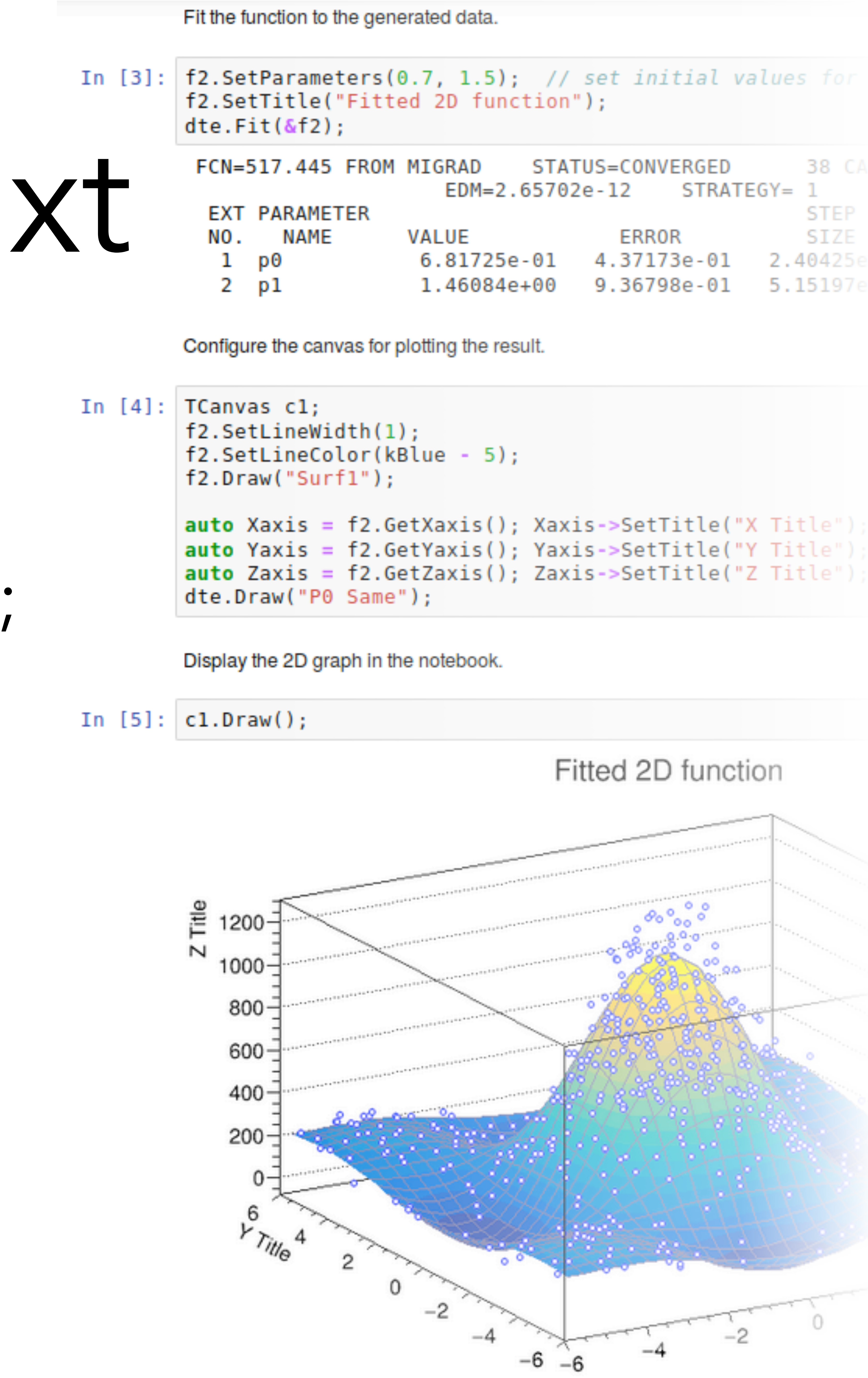
New Interfaces: Plan

- Will be released as part of ROOT 7
  - be used in experiment code developed for Run 3
- Gradual transition, followed by deprecation of old interfaces
  - re-use existing ROOT code where appropriate (e.g. fitting)
  - allow reading old data into new ROOT
  - exposure, feedback, discussions, patches! cern.ch/root7-signup

Extensions / Context

- JSROOT: can now draw simple TTree branches; geometry, tracks & hits (“JS-Eve”)

- Jupyter notebooks
  - SWAN



Fit the function to the generated data.

```
In [3]: f2.SetParameters(0.7, 1.5);  // set initial values for
        f2.SetTitle("Fitted 2D function");
        dte.Fit(&f2);
```

```
 FCN=517.445 FROM MIGRAD    STATUS=CONVERGED      38 CA
                     EDM=2.65702e-12    STRATEGY= 1
  EXT PARAMETER                                   STEP
  NO.   NAME      VALUE            ERROR          SIZE
   1  p0           6.81725e-01   4.37173e-01   2.40425e
   2  p1           1.46084e+00   9.36798e-01   5.15197e
```

Configure the canvas for plotting the result.

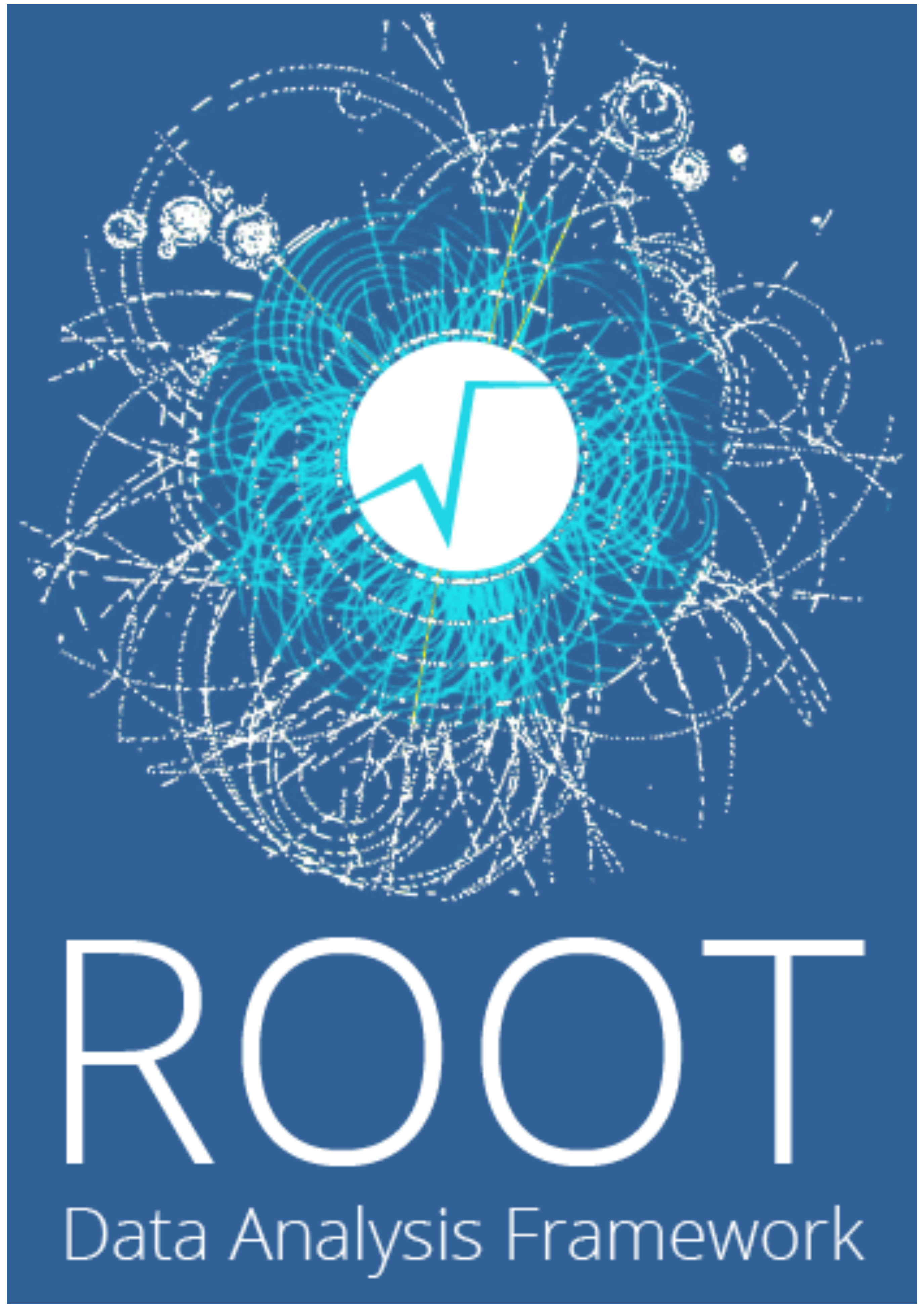
```
In [4]: TCanvas c1;
        f2.SetLineWidth(1);
        f2.SetLineColor(kBlue - 5);
        f2.Draw("Surf1");

        auto Xaxis = f2.GetXaxis(); Xaxis->SetTitle("X Title");
        auto Yaxis = f2.GetYaxis(); Yaxis->SetTitle("Y Title");
        auto Zaxis = f2.GetZaxis(); Zaxis->SetTitle("Z Title");
        dte.Draw("P0 Same");
```

Display the 2D graph in the notebook.

```
In [5]: c1.Draw();
```

Summary

- Ongoing development in many areas
- Keep the contributions coming!
  - feedback: root.cern/forum
  - bugs: root.cern/bugs
  - PRs: github.com/root-mirror/root
- Are we addressing your needs?



Backup

VecGeom adapter/converter

- New VecGeom solids adapter to TGeoShape
  - Providing only navigation functionality
  - Original ROOT shape pointer kept for other functionality (e.g. visualization)
- Converter for any transient TGeo geometry
  - Replacing shapes with the adapted matched in VecGeom
- New library built on demand
- One line usage: TVirtualGeoConverter::Instance()::ConvertGeometry()