New web-based widgets in ROOT

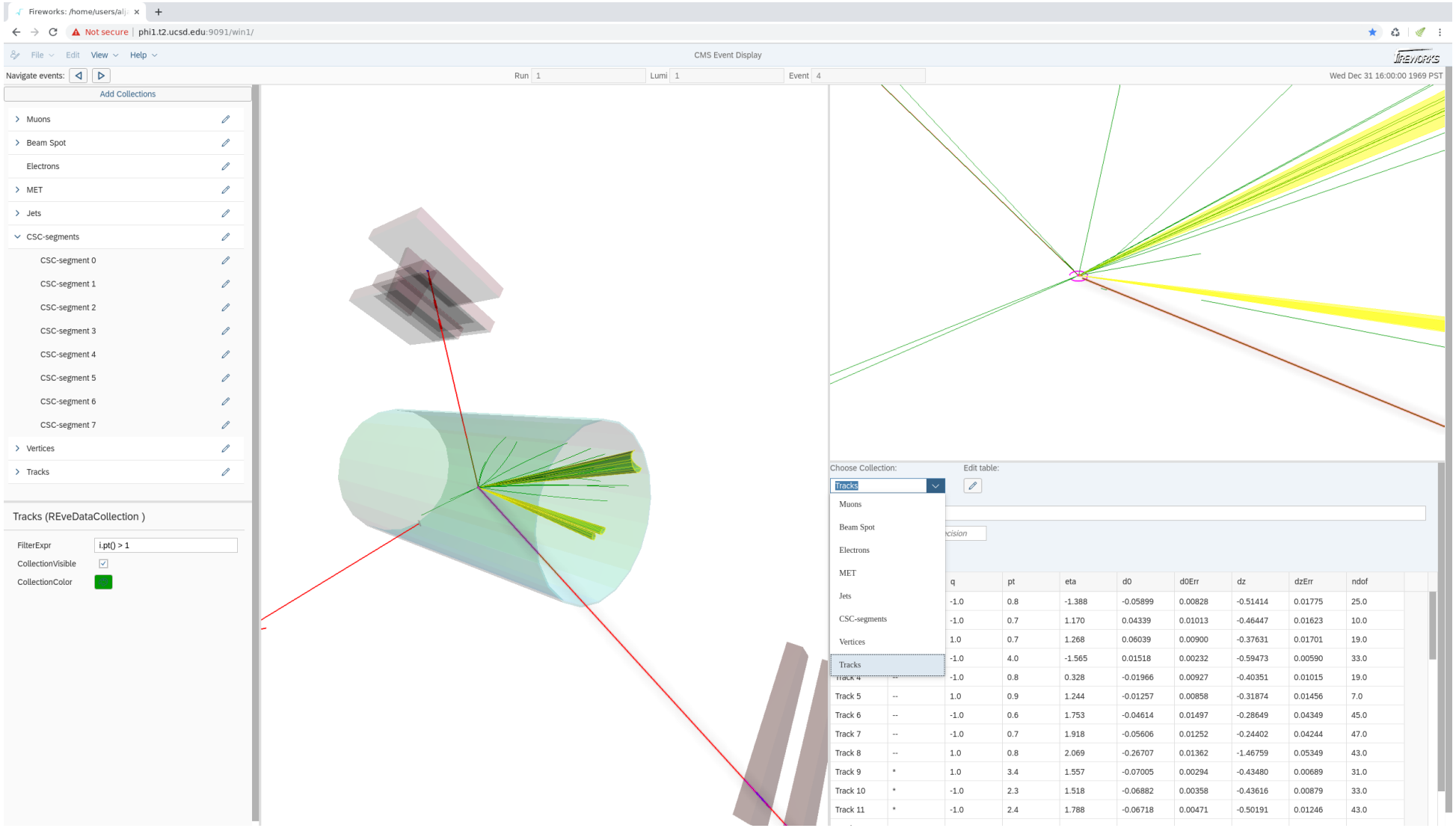
Bertrand Bellenot (CERN), Iliana Betsou (CERN),
Olivier Couet (CERN), Sergey Linev (GSI, Darmstadt),
Alja Mrak-Tadel (UCSD, San Diego), Thibault Souquet (CERN),
Matevz Tadel (UCSD, San Diego), Axel Naumann (CERN)

Introduction

- Full redesign of ROOT graphics
- Same code for all platforms
- Same code for visual and batch mode
- Threads-safe when feasible

- Take web technologies
- Reuse JSROOT
  - https://root.cern/js/

Eve7 and FireworksWeb



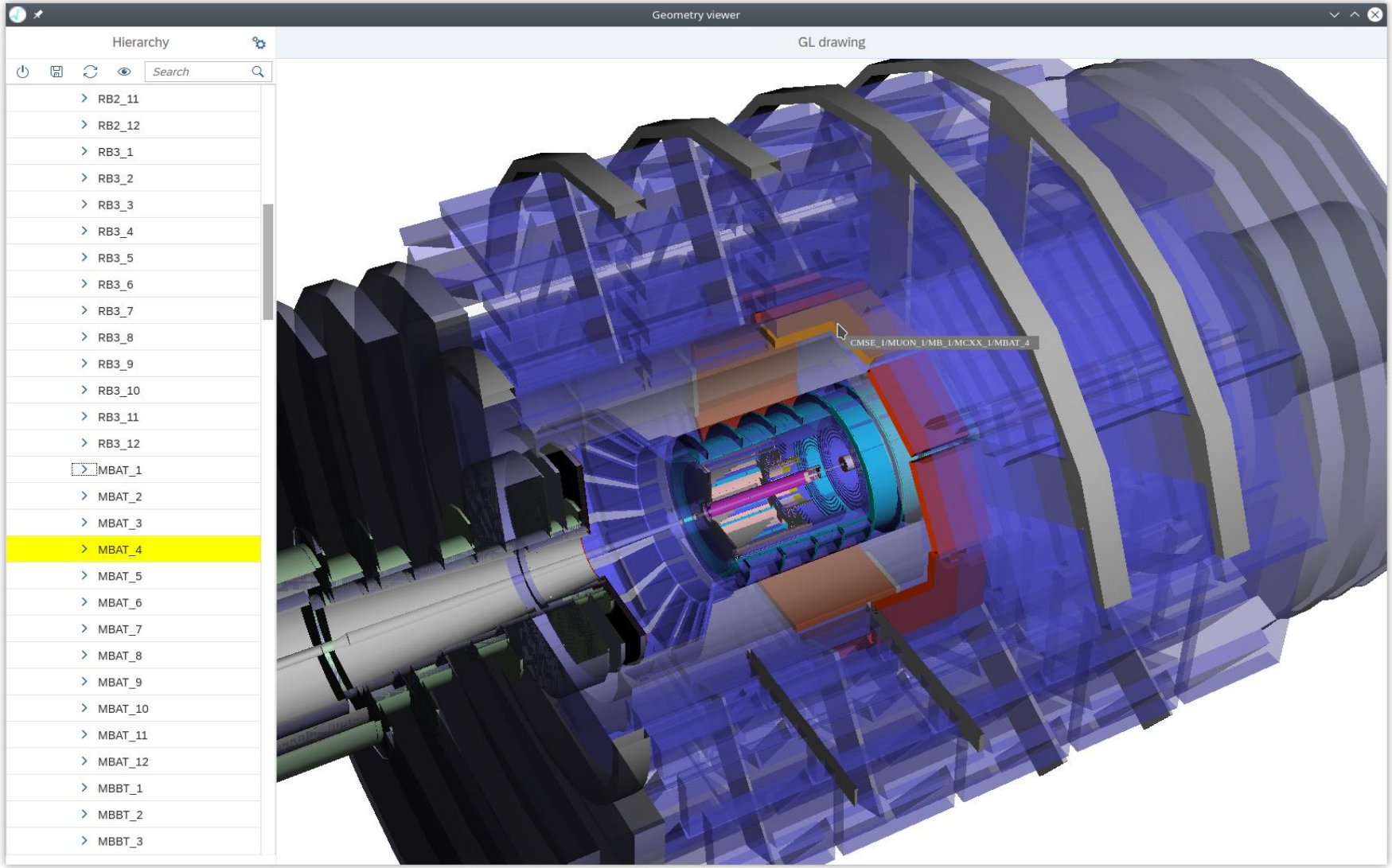
Online event display

- Hierarchy browser
- 3D views
- Projection views
- Table views
- Multiple clients
- Offline mode

https://linev.github.io/eve7/

See presentation of Matevz Tadel from Tuesday

RGeomViewer

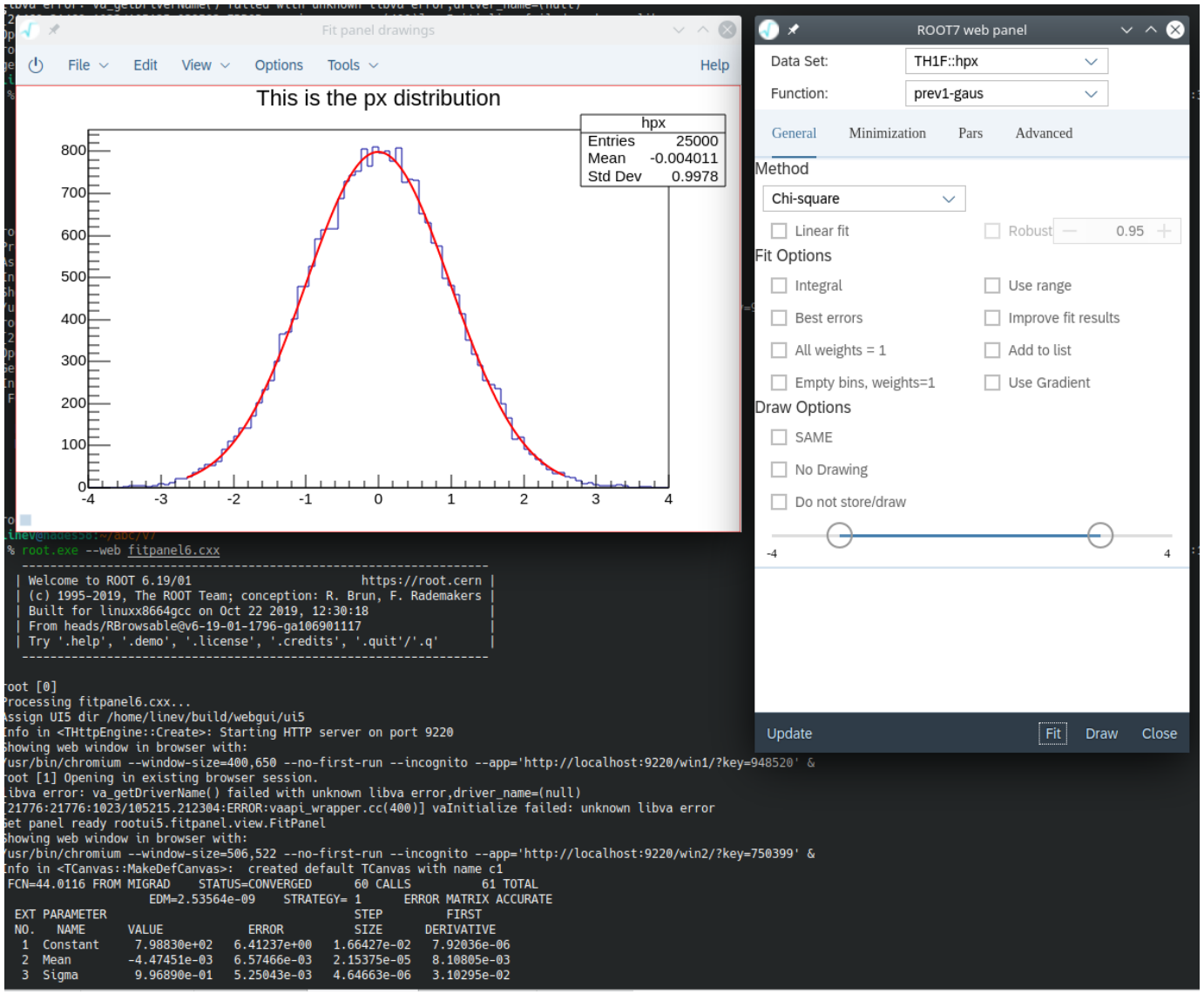


- Reuse eve7 and JSROOT code
- Browse hierarchy
- Search nodes
- Individual volume display
- Transparency
- Wireframes
- Offline mode

- See presentation later today

root $ROOTSYS/tutorials/eve7/viewer.cxx

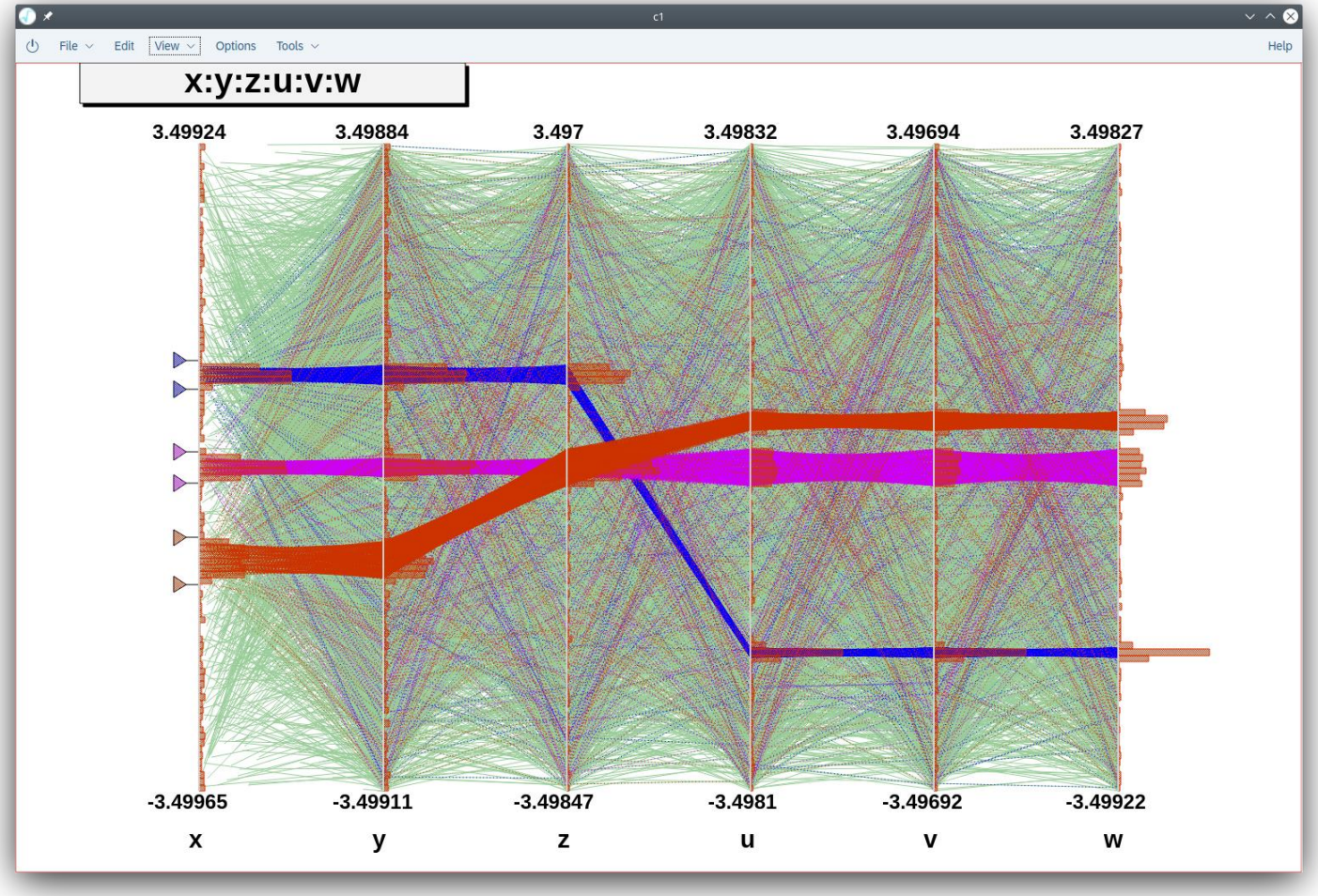
RFitPanel



root --web $ROOTSYS/tutorials/v7/fitpanel6.cxx

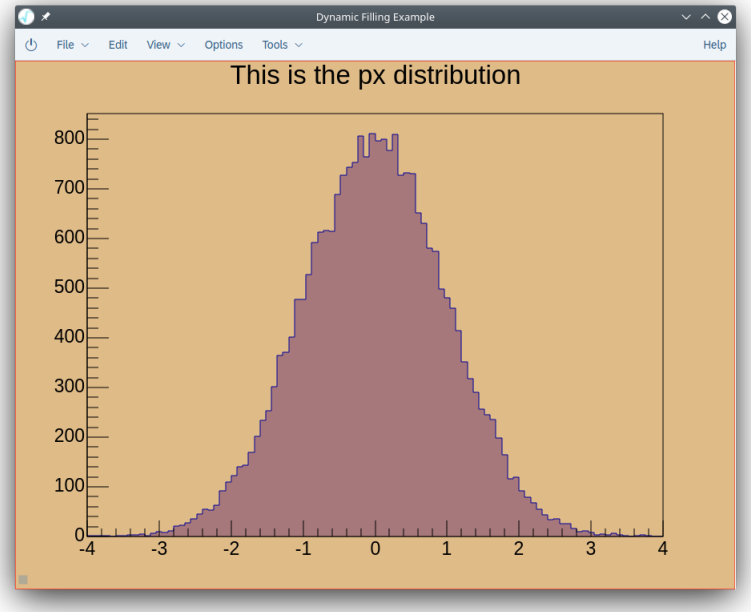
- Access fit functionality via web widget
  - very similar to original TFitPanel
  - use ROOT6 data classes for fitting
  - improve usability
- Example of model/view separation
  - model is C++ class
  - converted to/from JSON
  - used as is for view configuration
- Display fit results in TCanvas
  - x11 or web-based

TWebCanvas



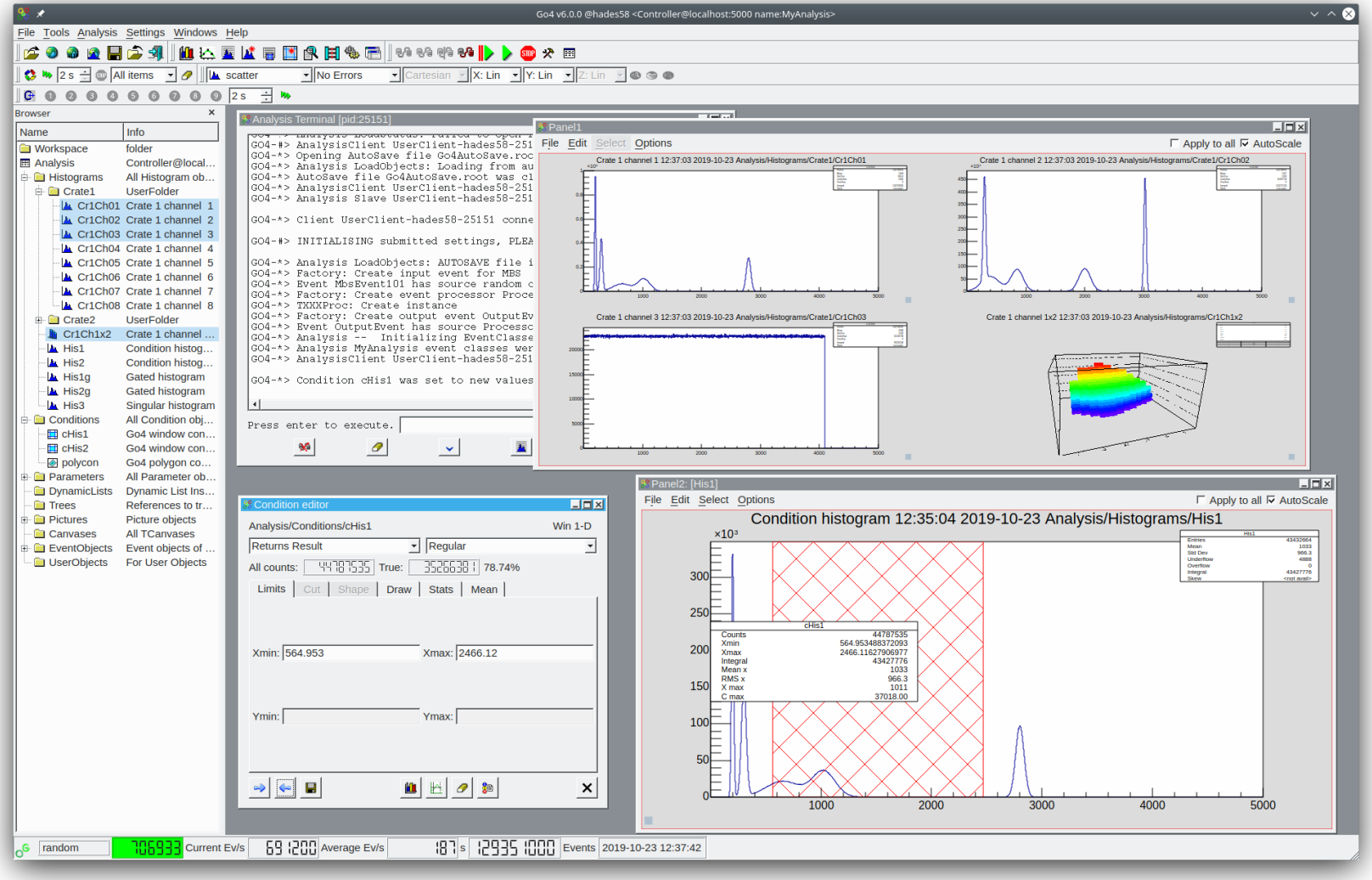
root --web [$ROOTSYS/tutorials/tree/parallelcoord.C]

- Show TCanvas in browser
  - web-based TCanvasImp
- Reuse JSROOT code
- Limited support of TVirtualX
  - custom Paint() may work



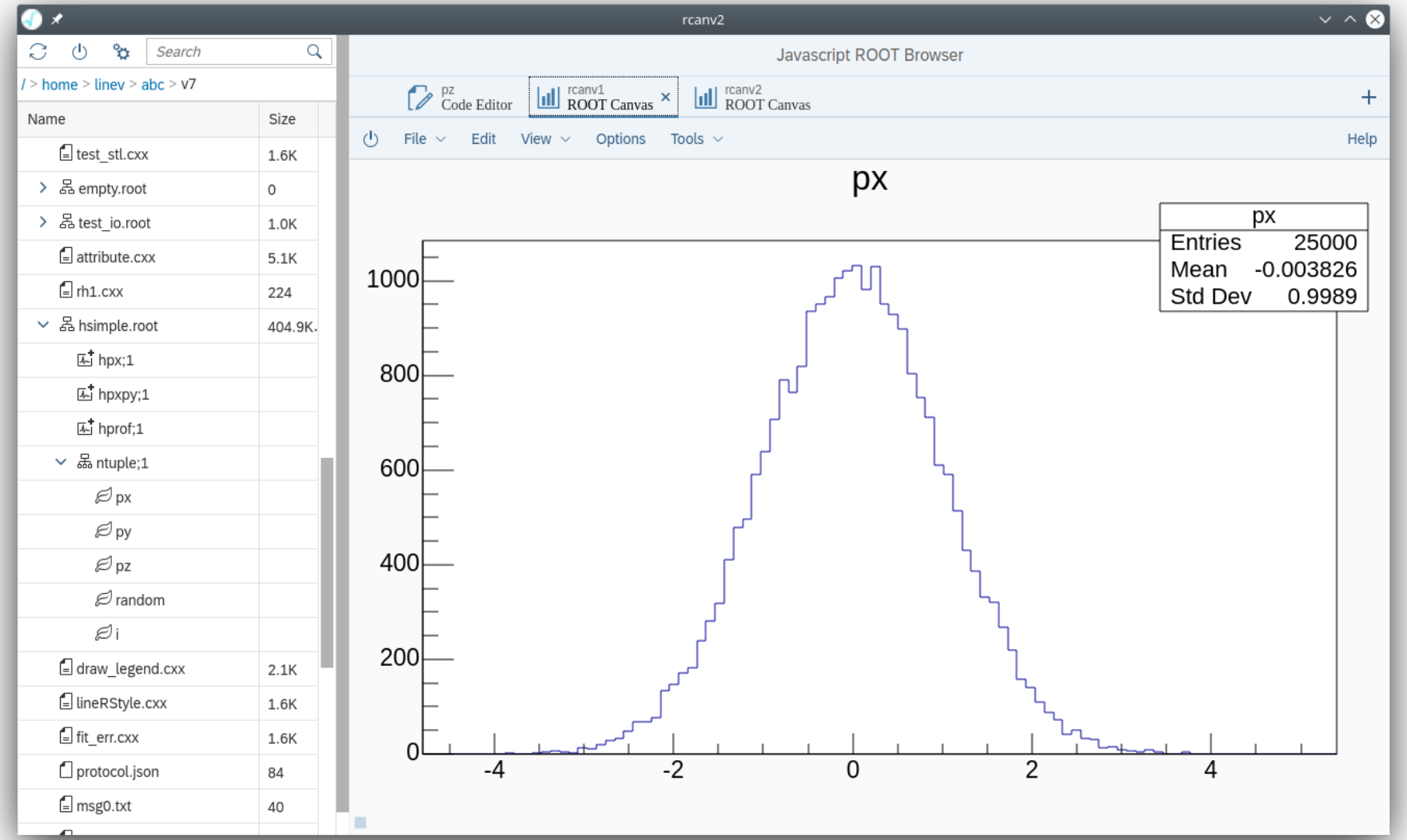
root --web [$ROOTSYS/tutorials/hsimple.C]

TWebCanvas with Qt5



- go4 v6.0
  - developed since 1999
  - http://go4.gsi.de
- Qt5-based GUI
  - QtROOT for ROOT graphics
  - since a while not working on Mac (missing x11 support)
- Solution:
  - embed TWebCanvas in QWebEngine
  - provide support for custom go4 classes
- Same code for:
  - Linux/Mac/Windows
- Any ROOT web widget can be embed in Qt5:
  - root --web=qt5 …

RBrowser

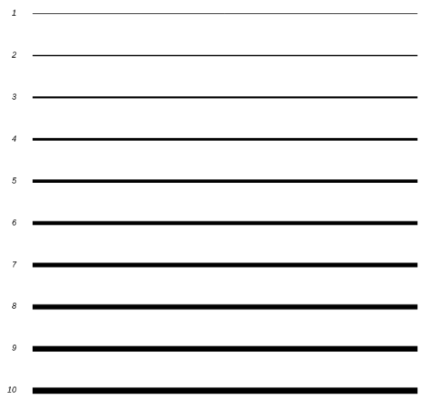


root $ROOTSYS/tutorials/v7/browser.cxx

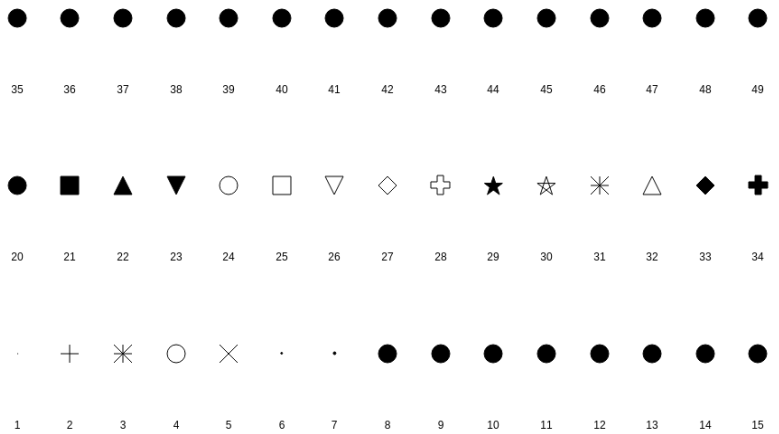
- Browse:
  - file system
  - ROOT files
  - TTree
- Display objects with:
  - RCanvas (ROOT7)
  - TWebCanvas (ROOT6)
- Edit text files
  - openui5 code editor
- View images
- Scalable hierarchy browser
  - load only visible items
- RBrowsable adapter classes:
  - object management
  - iterators over sub-elements
  - support old TObject::Browse(TBrowser*)
  - custom client info

RCanvas

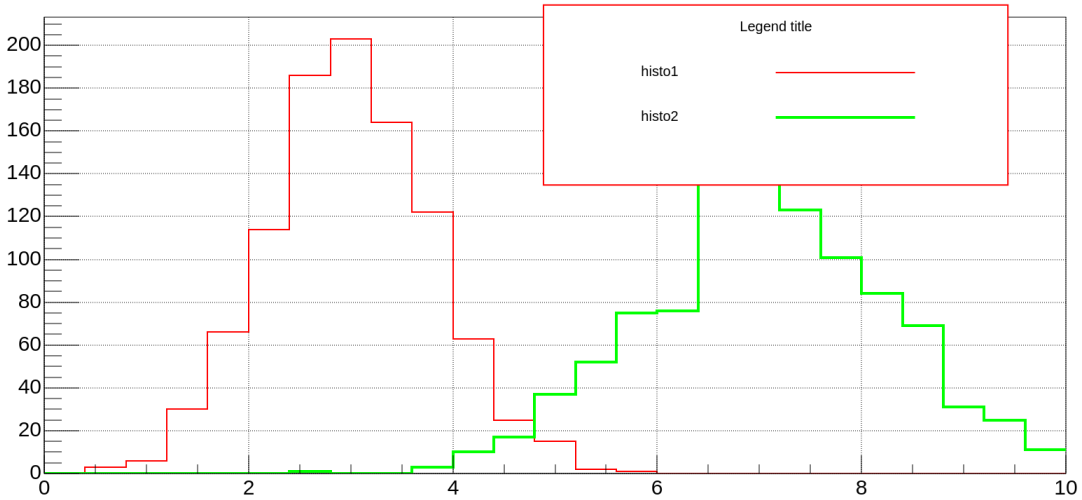
root $ROOTSYS/tutorials/v7/text.cxx



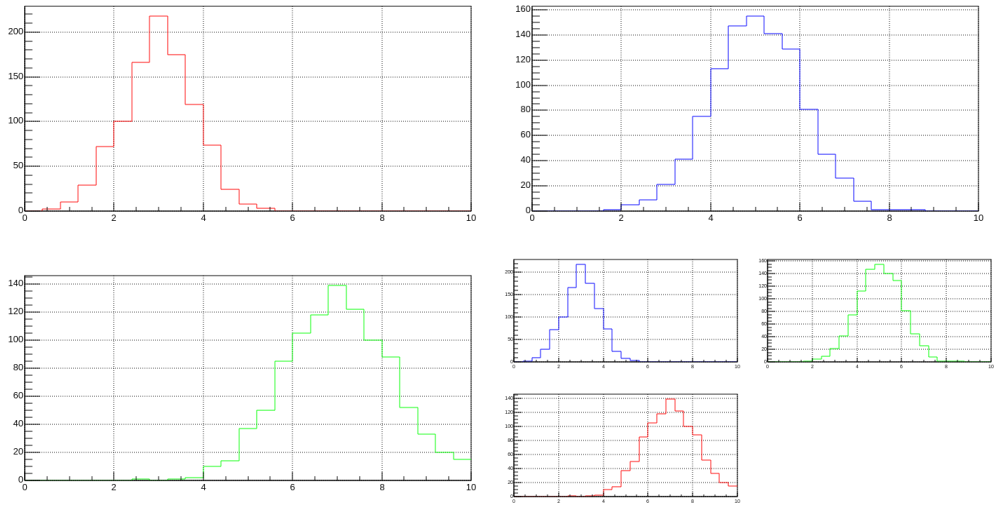
root $ROOTSYS/tutorials/v7/lineWidth.cxx



root $ROOTSYS/tutorials/v7/markerStyle.cxx



root $ROOTSYS/tutorials/v7/draw_legend.cxx



root $ROOTSYS/tutorials/v7/draw_subpads.cxx

RCanvas

- Full redesign of TCanvas class
- No gPad!
  - threads safety
- RDrawable
  - graphical primitive
  - attributes
  - reference data object
- RPadBase
  - maintain list of primitives

```
#include "ROOT/RCanvas.hxx"
#include "ROOT/RHistDrawable.hxx"

using namespace ROOT::Experimental;

void draw_rh1()
{
   RAxisConfig xaxis(25, 0., 10.);
   auto pHist = std::make_shared<RH1D>(xaxis);
   pHist->Fill(5);

   auto canvas = RCanvas::Create("Canvas Title");
   auto draw1 = canvas->Draw(pHist);
   draw1->AttrLine().SetColor(RColor::kRed).SetWidth(2);

   canvas->Show();
}
```

$ROOTSYS/tutorials/v7/draw_rh1.cxx

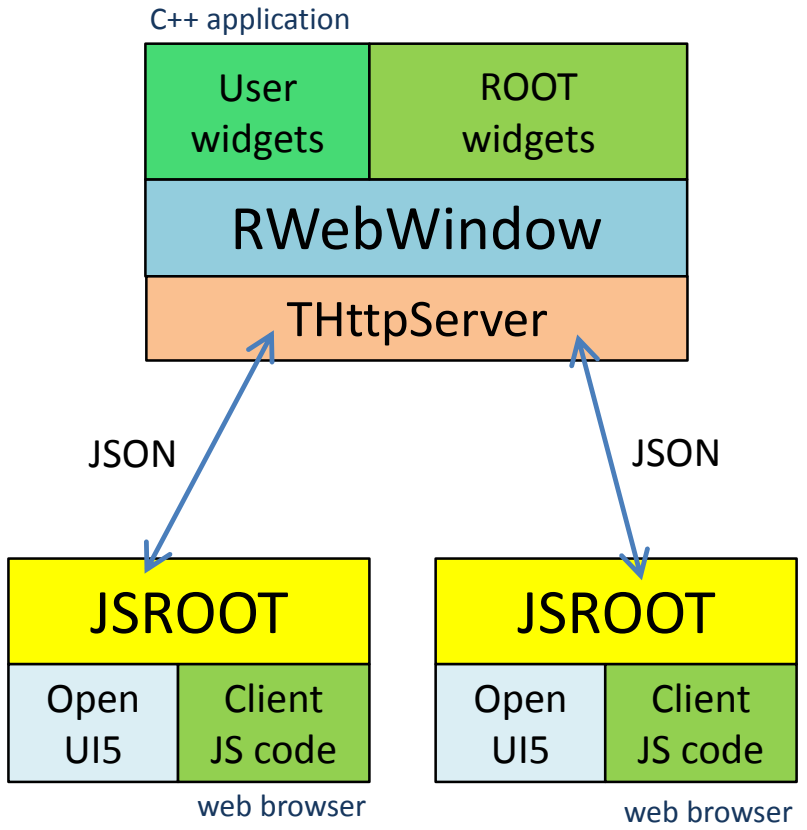
RCanvas

- Separate data (e.g. histogram) from view attributes
- Any attribute is optional
  - value can be provided with CSS-like syntax
  - default values provided in attribute classes
- Data can be shared via std::shared_ptr
  - provide I/O support, but only inside RCanvas
- Exactly same code for visual and batch mode

RWebWindow

- Gateway to web-based displays in ROOT

- Launch web browser(s)
  - special support for Chrome and Firefox
  - headless mode, used for batch mode
- Local web displays
  - Chromium Embedded Framework **CEF**
  - Qt5 *QWebEngine* – also chrome-based
- Communication via websockets
- Openui5 support
  - any other GUI framework can be used
- Offline support
  - client code can be used without running ROOT



Status and plans

- Functional prototypes and core interfaces are there
  - lot of changes in last three months
  - therefore still in ROOT::Experimental namespace

- All components can be tried now

  cmake –Droot7=on –DCMAKE_CXX_STANDARD=14 …

- Any feedback is welcome!

Useful links

- „Birds of a Feather“ meeting
  - today 13:30 – 15:00, Room **RB** (near R1)

- v7 tutorials
  - https://github.com/root-project/root/tree/master/tutorials/v7

- RWebWindow docu
  - https://root.cern/doc/master/classROOT_1_1Experimental_1_1RWebWindow.html

- Offline examples
  - https://linev.github.io/

Backup slides

RCanvas example

```
#include "ROOT/RCanvas.hxx"
#include "ROOT/RText.hxx"
#include "ROOT/RLine.hxx"

using namespace ROOT::Experimental;

void lineStyle() {
   auto canvas = RCanvas::Create("Canvas Title");
   double num = 0.3;
   for (int i=10; i>0; i--){
      num = num + 0.05;
      canvas->Draw<RText>(std::to_string(i))->SetPos({.3_normal, 1_normal*num}).AttrText().SetSize(13).SetAlign(32).SetFont(52);
      canvas->Draw<RLine>()->SetP1({.32_normal,1_normal*num}).SetP2({.8_normal, 1_normal*num}).AttrLine().SetStyle(i);
   }
   canvas->Show();
}
```

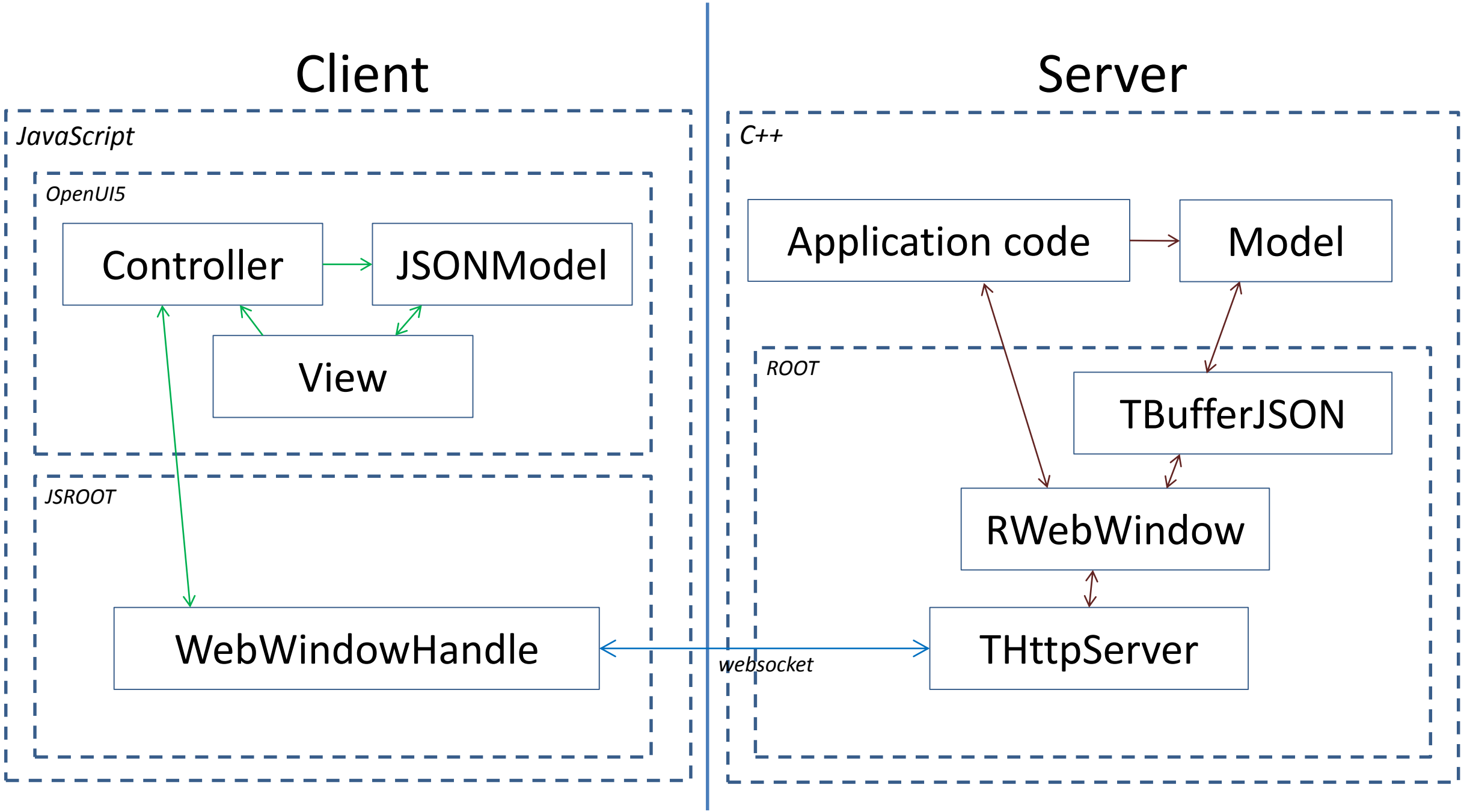
RCanvas example

```
#include "ROOT/RCanvas.hxx"
#include "ROOT/RText.hxx"
#include "ROOT/RLine.hxx"

using namespace ROOT::Experimental;

void lineStyle() {
   auto canvas = RCanvas::Create("Canvas Title");
   double num = 0.3;
   for (int i=10; i>0; i--){
      num = num + 0.05;
      auto text = canvas->Draw<RText>(std::to_string(i));
      text->SetPos({.3_normal, 1_normal*num});
      auto &atext = text->AttrText();
      atext.SetSize(13);
      atext.SetAlign(32);
      atext.SetFont(52);

      auto line = canvas->Draw<RLine>();
      line->SetP1({.32_normal,1_normal*num});
      line->SetP2({.8_normal, 1_normal*num});
      auto &aline = line->AttrLine();
      aline.SetStyle(i);
   }
   canvas->Show();
}
```


Client
Server
JavaScript
C++
OpenUI5
Controller
JSONModel
View
JSROOT
WebWindowHandle
Application code
Model
ROOT
TBufferJSON
RWebWindow
THttpServer
websocket