

New web-based widgets in ROOT

Bertrand Bellenot (CERN),
Iliana Betsou (CERN),
Olivier Couet (CERN),
Sergey Linev (GSI, Darmstadt),
Axel Naumann (CERN)

Eve-7 and FireworksWeb

The screenshot displays the FireworksWeb interface for CMS Event Display. The main window shows a 3D visualization of particle tracks originating from a central vertex. The tracks are color-coded and extend through various detector components, including the Beam Spot, Electrons, MET, Jets, and CSC-segments. A hierarchy browser on the left side allows users to navigate through these collections. The bottom right corner features a table of track data, which is currently filtered to show tracks with $p_T > 1$.

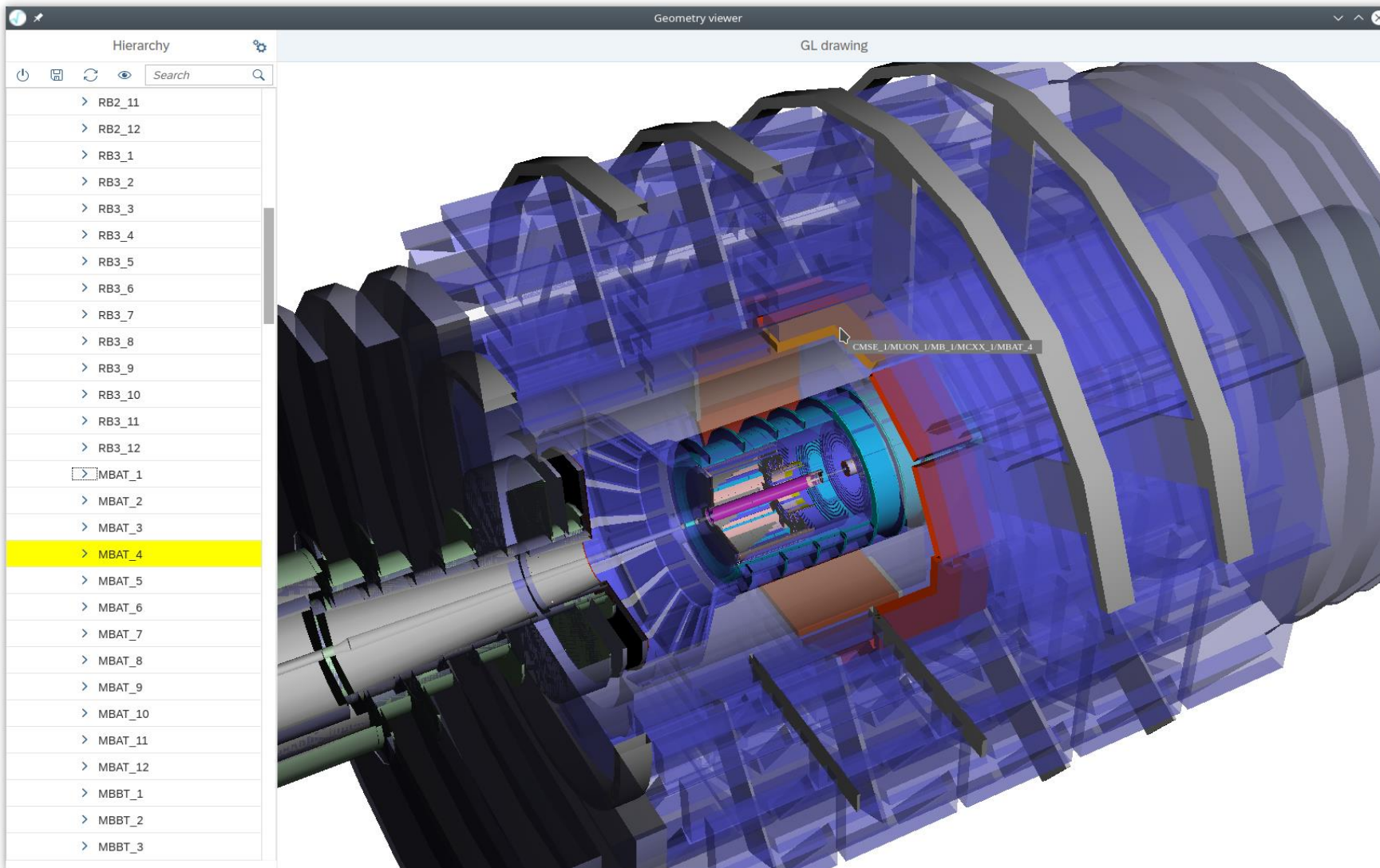
Collection	q	pt	eta	d0	d0Err	dz	dzErr	ndof	
Jets	-1.0	0.8	-1.388	-0.05899	0.00828	-0.51414	0.01775	25.0	
CSC-segments	-1.0	0.7	1.170	0.04339	0.01013	-0.46447	0.01623	10.0	
Vertices	1.0	0.7	1.268	0.06039	0.00900	-0.37631	0.01701	19.0	
Tracks	-1.0	4.0	-1.565	0.01518	0.00232	-0.59473	0.00590	33.0	
Track 4	-1.0	0.8	0.328	-0.01966	0.00927	-0.40351	0.01015	19.0	
Track 5	--	1.0	0.9	1.244	-0.01257	0.00858	-0.31874	0.01456	7.0
Track 6	--	-1.0	0.6	1.753	-0.04614	0.01497	-0.28649	0.04349	45.0
Track 7	--	-1.0	0.7	1.918	-0.05606	0.01252	-0.24402	0.04244	47.0
Track 8	--	1.0	0.8	2.069	-0.26707	0.01362	-1.46759	0.05349	43.0
Track 9	*	1.0	3.4	1.557	-0.07005	0.00294	-0.43480	0.00689	31.0
Track 10	*	-1.0	2.3	1.518	-0.06882	0.00358	-0.43616	0.00879	33.0
Track 11	*	-1.0	2.4	1.788	-0.06718	0.00471	-0.50191	0.01246	43.0

- 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

The screenshot displays the ROOT7 web panel interface. On the left, a window titled 'Fit panel drawings' shows a histogram of data points (px distribution) with a red Gaussian fit curve overlaid. The plot is titled 'This is the px distribution'. A statistics box for the fit is visible:

hpx	
Entries	25000
Mean	-0.004011
Std Dev	0.9978

On the right, the 'ROOT7 web panel' control interface is shown. It includes dropdown menus for 'Data Set' (TH1F::hpx) and 'Function' (prev1-gaus). Below these are tabs for 'General', 'Minimization', 'Pars', and 'Advanced'. The 'Method' section is set to 'Chi-square'. There are checkboxes for 'Linear fit', 'Integral', 'Best errors', 'All weights = 1', 'Empty bins, weights=1', 'SAME', 'No Drawing', and 'Do not store/draw'. A 'Robust' checkbox is checked with a value of 0.95. A slider at the bottom ranges from -4 to 4. At the bottom of the control panel are buttons for 'Update', 'Fit', 'Draw', and 'Close'.

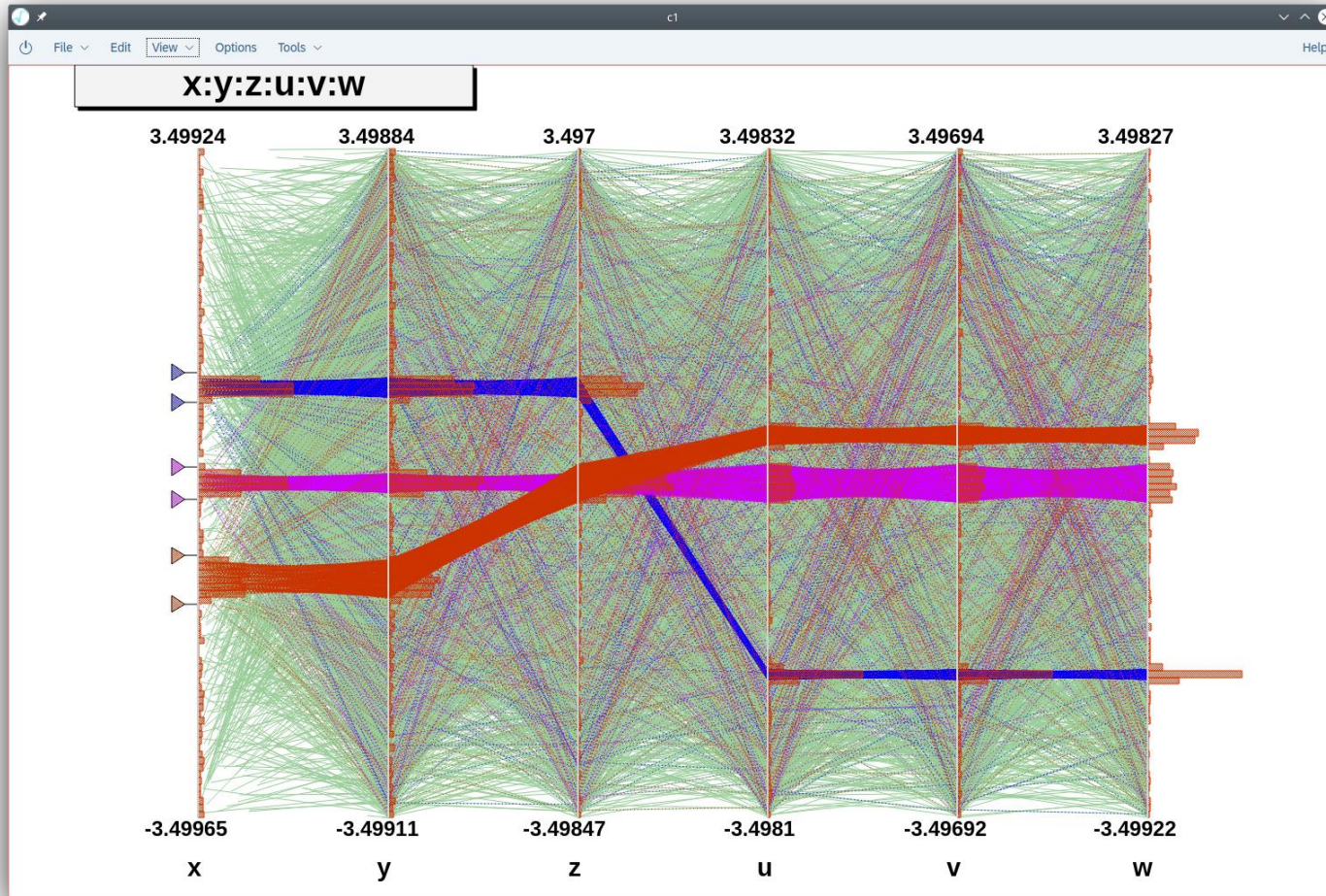
```
root [0]
processing fitpanel6.cxx...
assign UI5 dir /home/linev/build/webgui/ui5
info in <THttpEngine::Create>: Starting HTTP server on port 9220
showing web window in browser with:
usr/bin/chromium --window-size=400,650 --no-first-run --incognito --app='http://localhost:9220/win1/?key=948520' &
root [1] Opening in existing browser session.
libva error: va_getDriverName() failed with unknown libva error_driver_name=(null)
21776:21776:1023/105215,212304:ERROR:vaapi_wrapper.cc(400)] vaInitialize failed: unknown libva error
set panel ready rootui5.fitpanel.view.FitPanel
showing web window in browser with:
usr/bin/chromium --window-size=506,522 --no-first-run --incognito --app='http://localhost:9220/win2/?key=750399' &
info in <TCanvas::MakeDefCanvas>: created default TCanvas with name c1
FCN=44.0116 FROM MIGRAD STATUS=CONVERGED 60 CALLS 61 TOTAL
EDM=2.53564e-09 STRATEGY= 1 ERROR MATRIX ACCURATE
EXT PARAMETER
NO. NAME VALUE ERROR STEP FIRST
1 Constant 7.98830e+02 6.41237e+00 1.66427e-02 7.92036e-06
2 Mean -4.47451e-03 6.57466e-03 2.15375e-05 8.10805e-03
3 Sigma 9.96890e-01 5.25043e-03 4.64663e-06 3.10295e-02
```

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

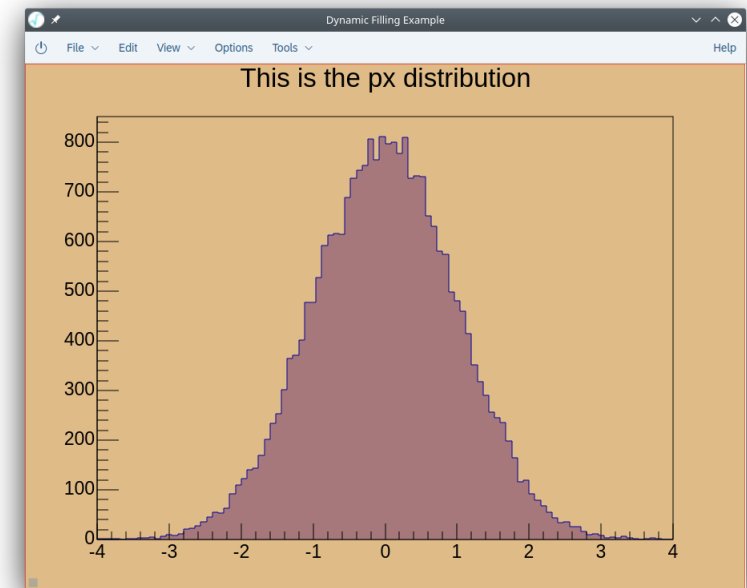
- Access fit functionality via web widget
 - very similar to original TFitPanel
 - use ROOT6 data classes for fitting
- Example of model/view separation
 - model is C++ class
 - converted to/from JSON
 - used as is for view configuration
- Display fit results in TCanvas
 - x11 or web-based

TWebCanvas

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

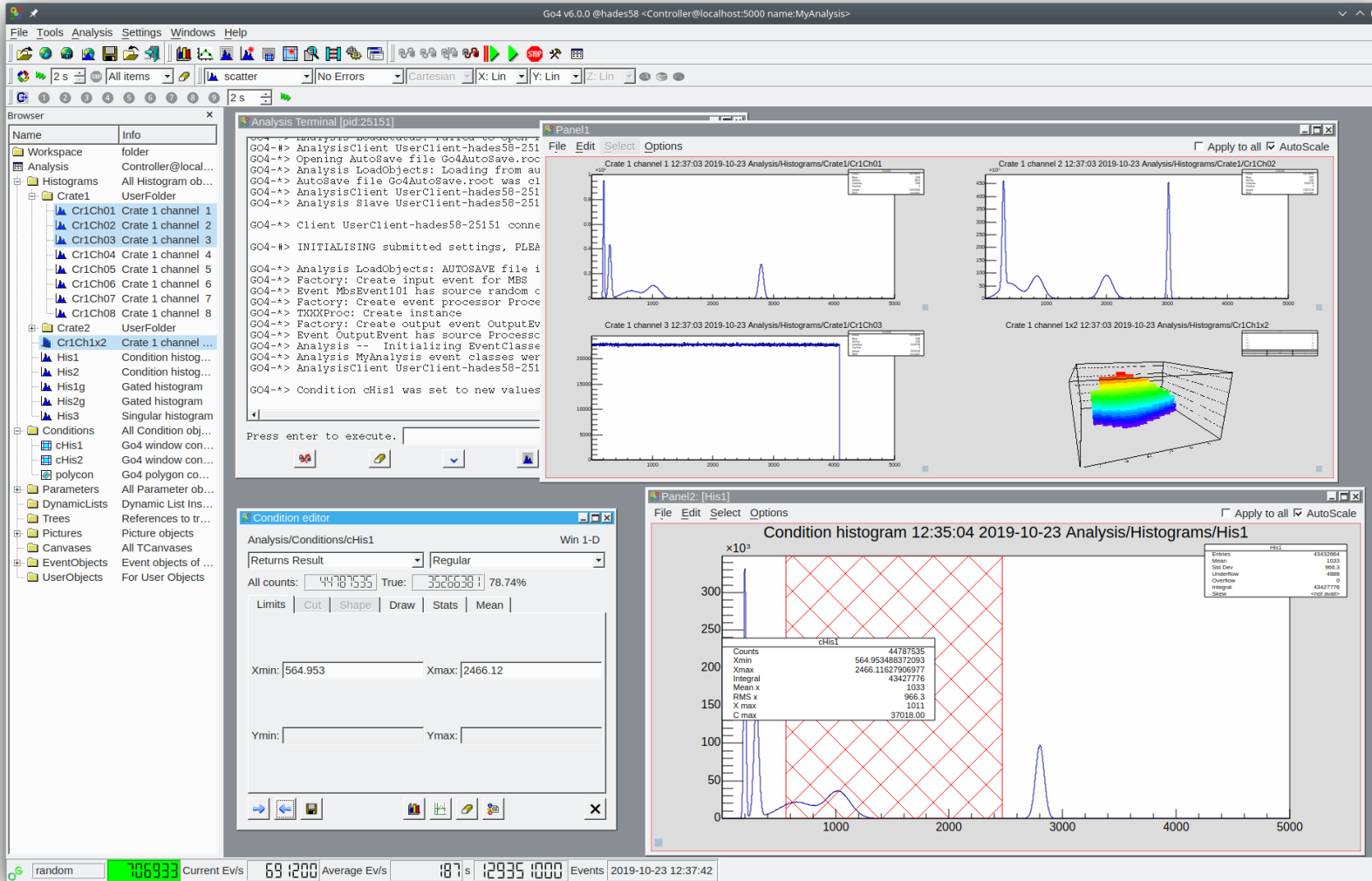


`root --web $ROOTSYS/tutorials/tree/parallelcoord.C`



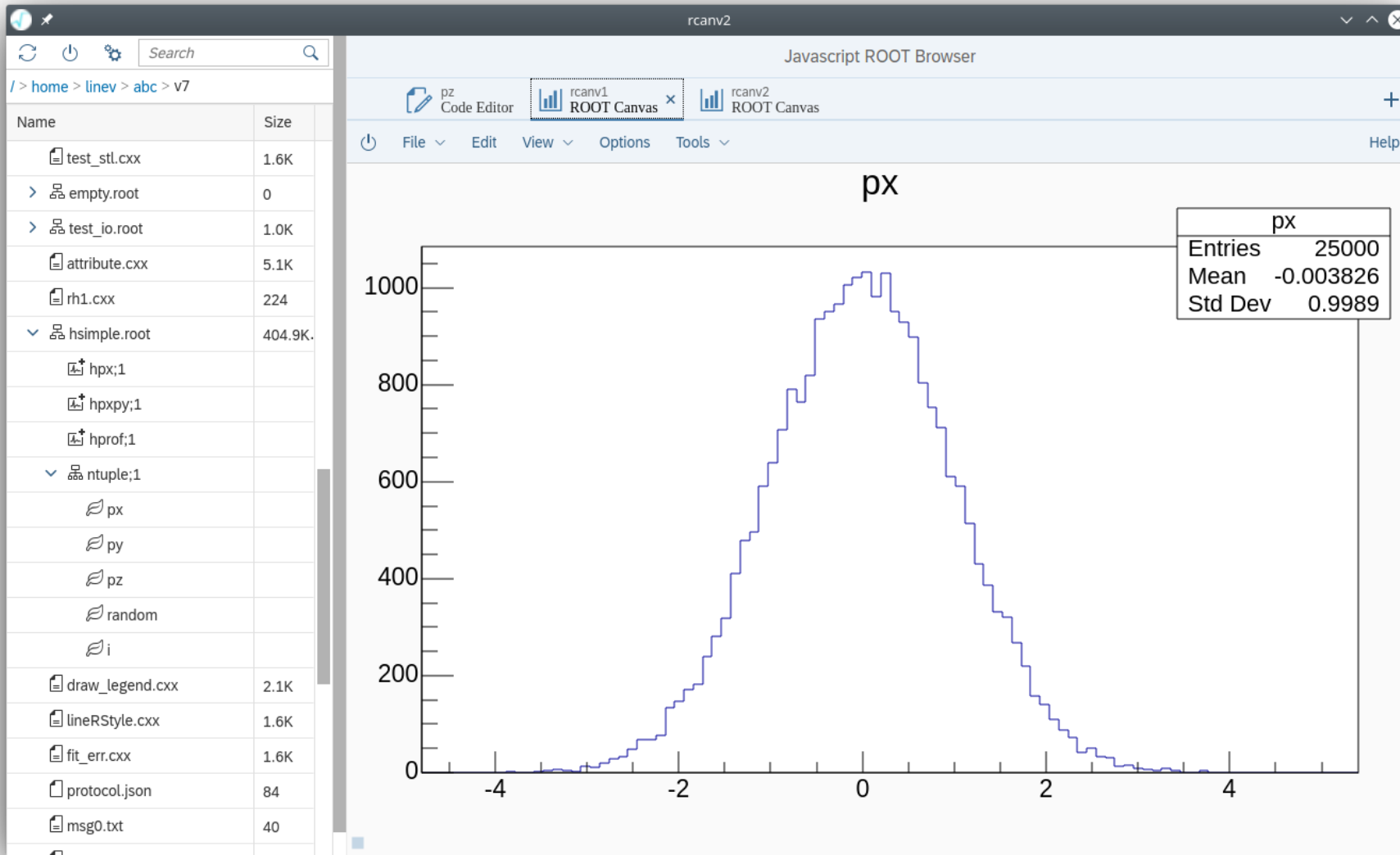
`root --web $ROOTSYS/tutorials/hsimple.C`

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



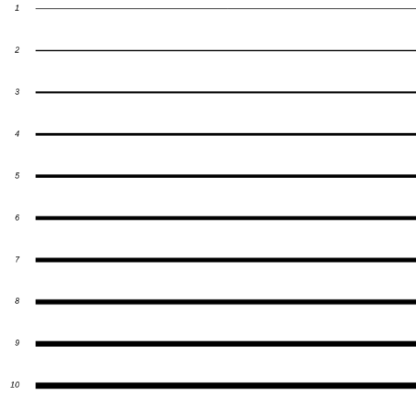
root \$ROOTSYS/tutorials/v7/browser.cxx

- Browse:
 - file system
 - ROOT files
- Display objects with:
 - RCanvas (ROOT7)
 - TCanvas (ROOT6)
- Edit text files
 - openui5 code editor
- View images
- RBrowsable classes:
 - object management
 - structure iterators
 - support old `Browse(TBrowser*)`
 - custom client info

RCanvas



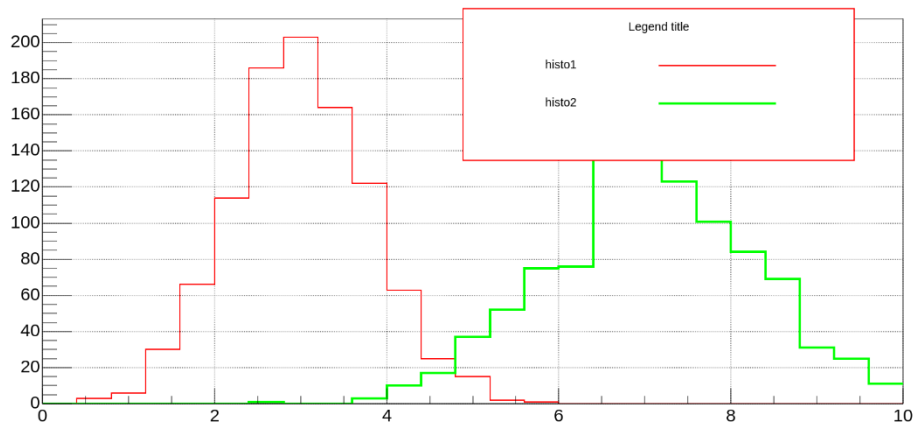
root \$ROOTSYS/tutorials/v7/text.cxx



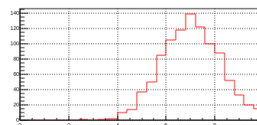
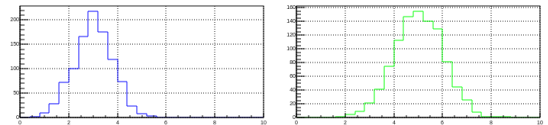
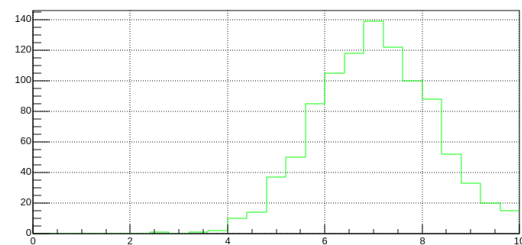
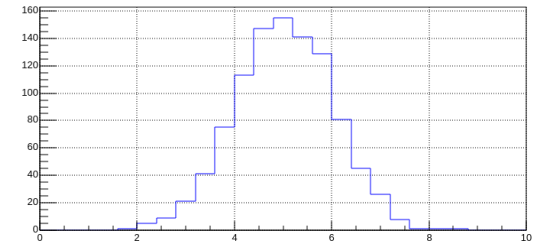
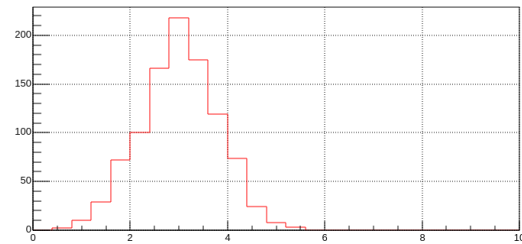
root \$ROOTSYS/tutorials/v7/lineWidth.cxx



root \$ROOTSYS/tutorials/v7/markerStyle.cxx



root \$ROOTSYS/tutorials/v7/draw_legend.cxx



root \$ROOTSYS/tutorials/v7/draw_subpads.cxx

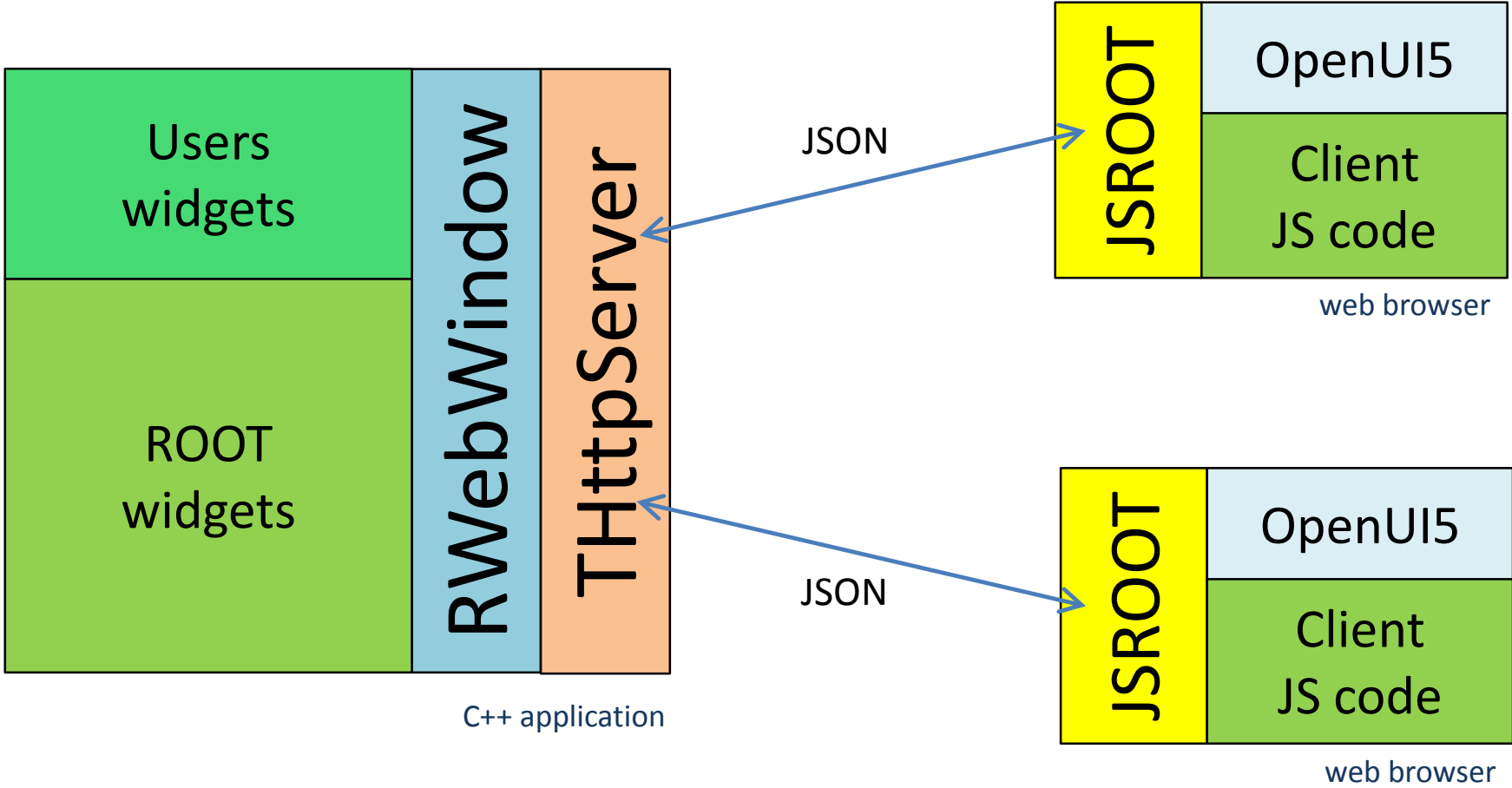
RCanvas

- Full redesign of TCanvas class
- No gPad!
- Separate data 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
- 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

RWebWindow role



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
 - compile ROOT master with **-Droot7=on** flag
- Any feedback is welcome!

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

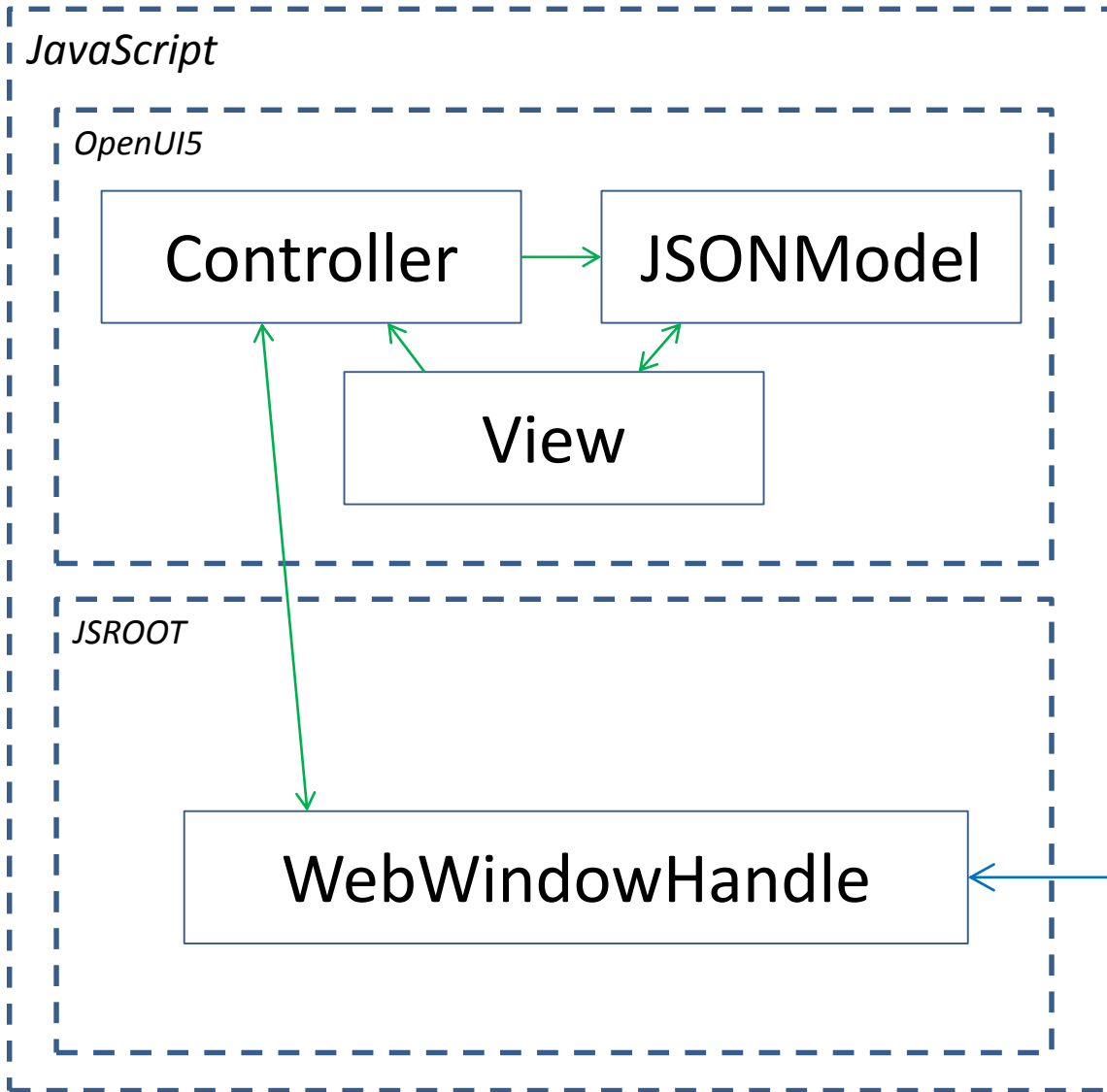
```
#include "ROOT/RCanvas.hxx"
#include "ROOT/RText.hxx"
#include "ROOT/RLine.hxx"

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

