ESSnuSB WP5
Orsay progresses since March meeting

14 June 2019 video meeting

G.Barrand, CNRS/IN2P3/LAL

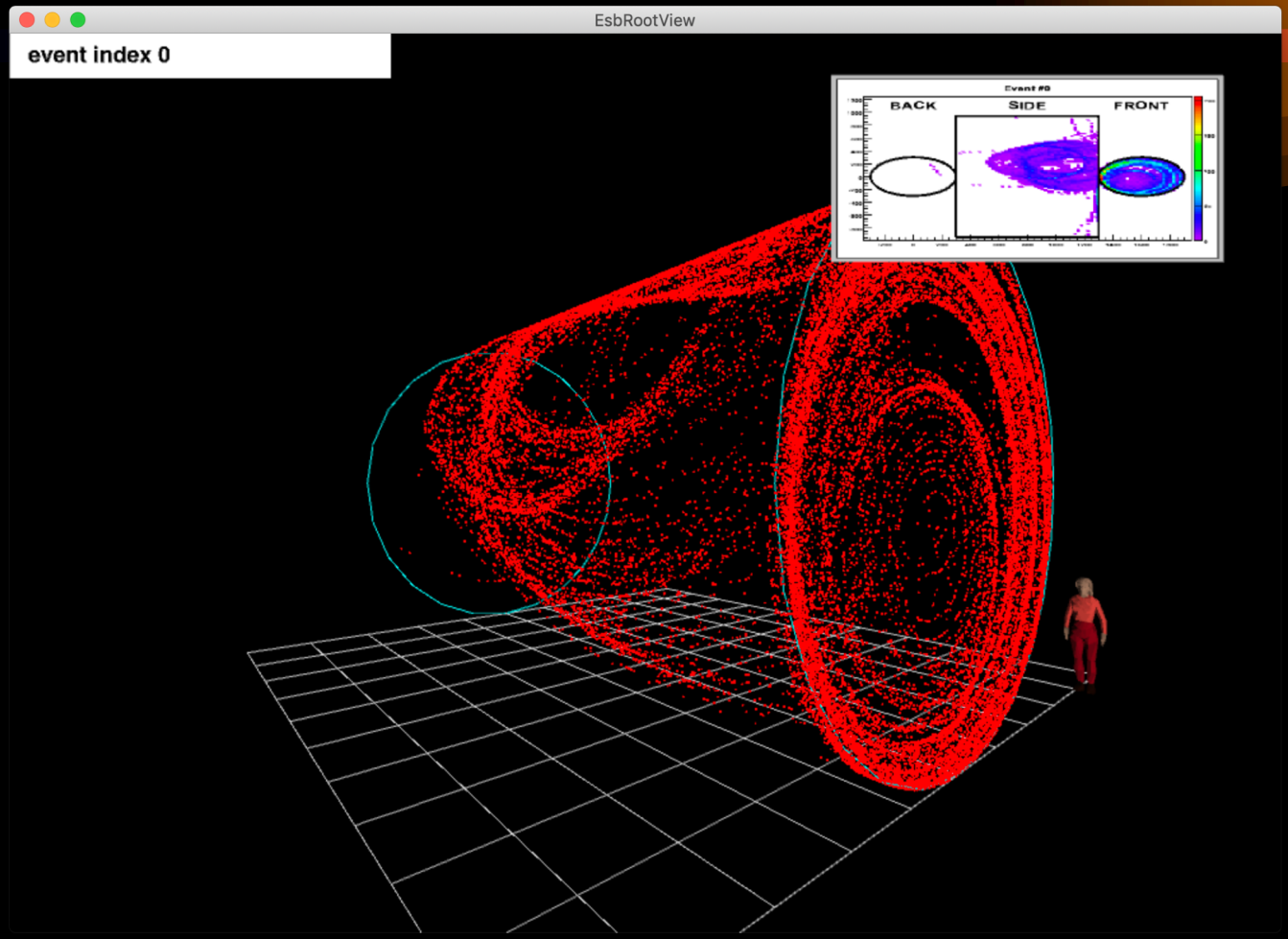
Done since March

- EsbRootView 1.0.0 release in May on GitHub gbarrand/EsbRootView
- Pages under https://gbarrand.github.io. See EsbRootView section.

EsbRootView/1.0.0

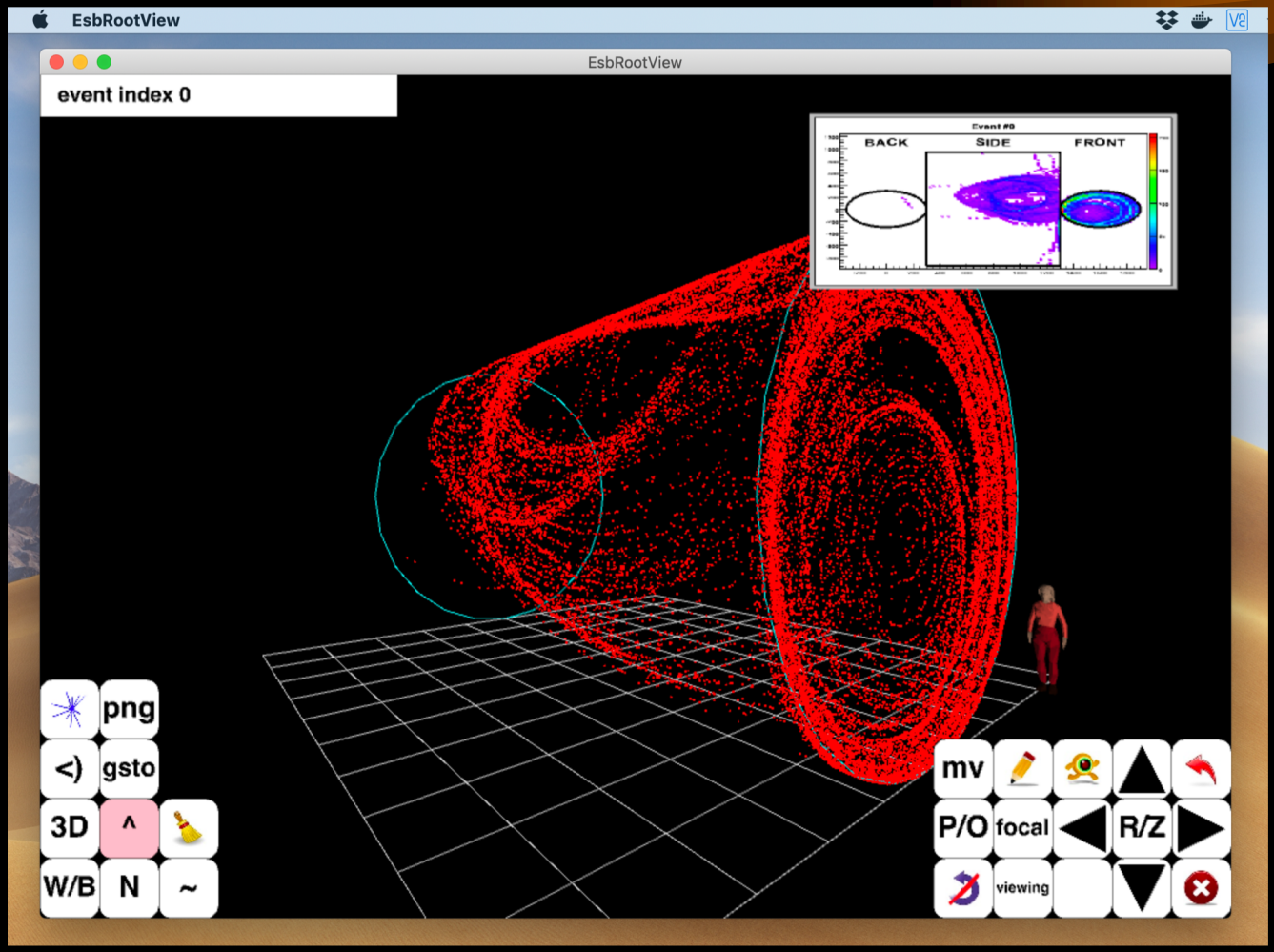
- Application « thought » as a smartphone/tablet like « app ».
- Today « reps » for :
  - Detector (today a cylinder) read from geo_full.root.
  - WCDetectorPoints (today colorised uniformly in red)
  - The « flat 2D projection » plot.
  - A grid to represent the « floor » to help position things in space.
  - Avatars, to help giving the scale of things.
- « goodies » to interact with views.
- Have a working « next event ».
- Reading an event files from the user « document directory ».
- Packing comes with a evetest.root file with five events (produced with EsbRoot/ess_sim.C).

EsbRootView/1.0.0



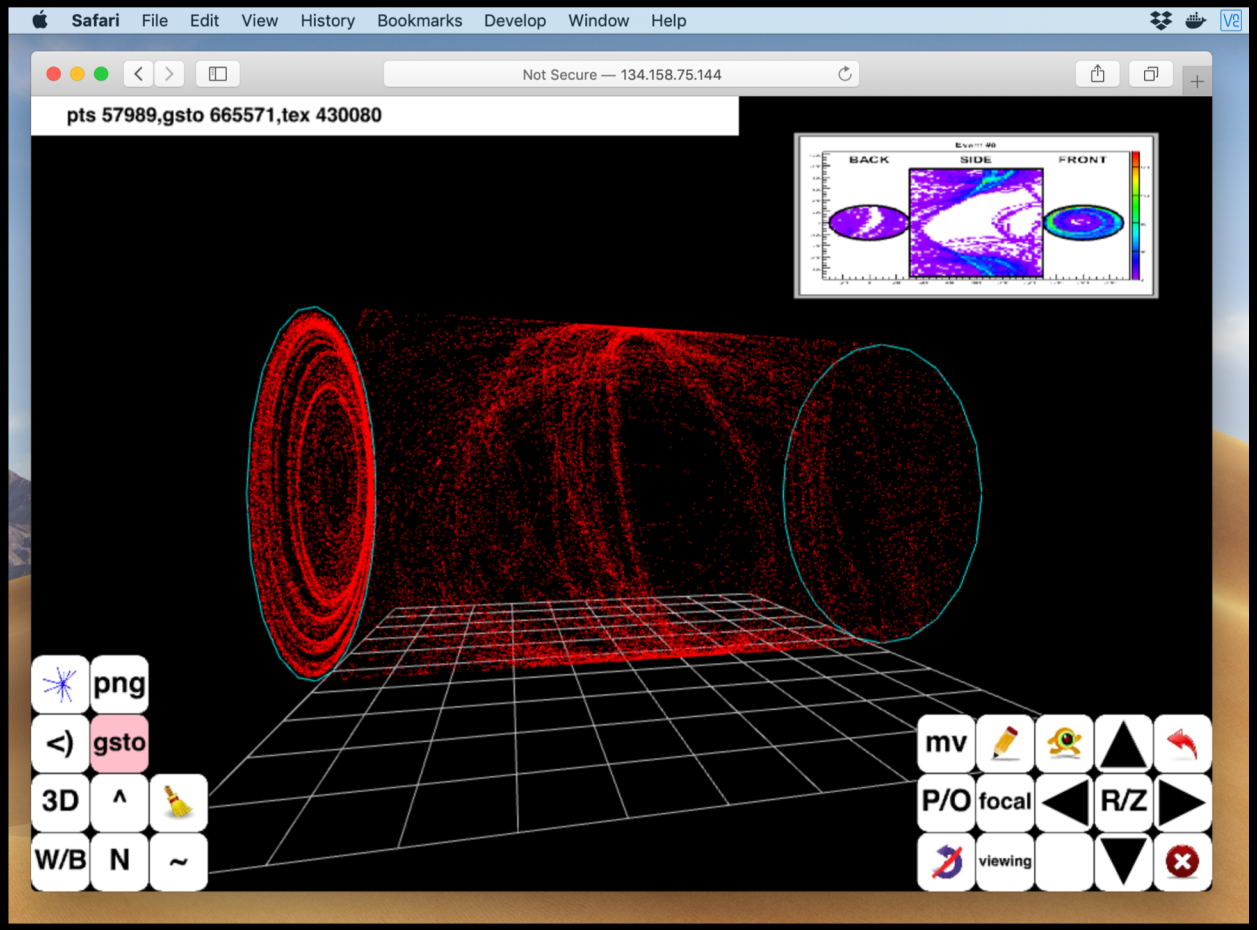
EsbRootView/1.0.0

With « camera panels » buttons.



EsbRootView/1.0.0

In a web browser (here macOS/Safari). Server side done with tntnet C++ lib to handle « web server logic » and WebGL. Here running on a LAL/OpenStack virtual machine.



Next…

- Have a specific demo on YouTube.
- Depose on GooglePlay (=> world wide distrib on Android).
- Probably too early to submit to Mac/iOS Apple app stores.
- More representations ?
  - MCTracks ?
  - WCPoints of an MCTrack : to « cross check » the event model ?
  - Colorise/filter WCPoints according to particle type ?
  - Primary tracks ? (« Where is the beam ? »)
  - The 2D plot in full view.
  - More plots ?
  - Other ideas ?

Next… (technicalities)

- IO : generic creation of an object from « streamer infos » deposited in a file. I want that.
- Big issue on macOS/iOS : have a « Metal » driver. (Apple OpenGL deprecated at WWDC-2018).