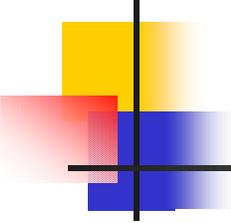


OnX & ROOT on behalf of Guy Barrand

ROOT Workshop 2001
June 13 FNAL

René Brun
CERN

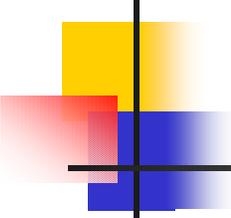


Context



- At the recent [HEPVIS](#) workshop in Boston, Guy presented his ideas on the medium/long term cooperation between various systems.
- We had the pleasure to welcome Guy for one week end of May.
- This was an opportunity to discuss with him several points, in particular the ROOT GUI and the 2-D, 3-D, Open Inventor interfaces
- During this week, Guy (and Fons) made some progress to discuss integration issues.
- Unfortunately, Guy could not come to the workshop. I asked him to prepare a summary of his work and ideas.
- The following slides based on Guy's URL:

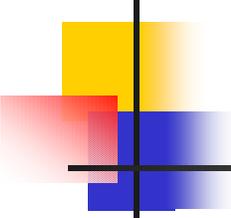
<http://www.lal.in2p3.fr/SI/OnX>



Overview



- Under development (07/06/2001) is the declaration of the ROOT GUI classes to OnX. Then from an OnX XML description of a GUI, the onx program will permit to construct a GUI by using the ROOT GUI classes.
- A first job had consisted to see the modularity of the ROOT GUI classes ; is it possible to put these classes in some stand alone package that could be compared to the packaging of other GUI toolkit like Xt, Gtk, Qt,..?
- From the code source of ROOT/3.06, the package [RootGui](#) (not yet released) had been formed. This package works over the CINT package and the [RioGrande](#) package which is a standalone packaging of the ROOT core classes.
- Having the [RootGui](#) package had been quite straightforward. Only one class "[TGApplication](#)" was missing, that permits to initialize the GUI system alone (without the ROOT graphic). This initialization class is something similar to XtAppInitialize for Xt, QApplication for Qt, etc...



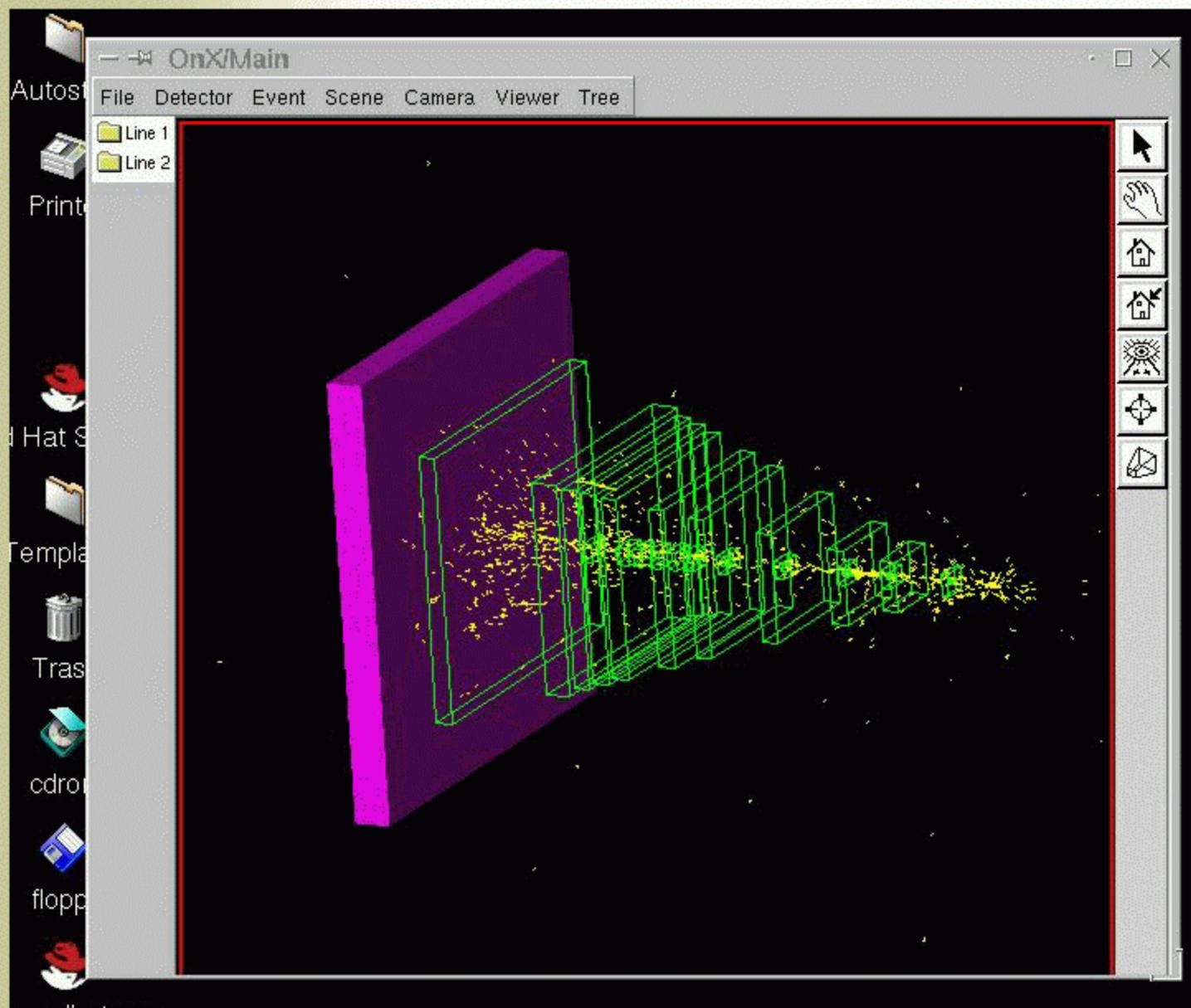
Overview



- OnX being able to handle [Inventor](#), a class to do the connection of [RootGui](#) to Inventor had been developed : TGSoExaminerViewer. This class is the equivalent of SoXtExaminerViewer that does the connection of Inventor kernel to Xt, SoQtExaminerViewer that does the same for Qt, etc... It had been developed from the source code of TRootOViewer, developed by Valeri Fine and Fons Rademakers that have made the tricky job of connecting the ROOT steering, the ROOT GUI, the existing ROOT 3D visualization, Inventor and Xt.
- For the steering of the GUI, the OnX ROOT driver uses the steering of ROOT Core.
- Some job is underway so that the source code used by the standalone packaging of RioGrande, RootGui be 100% the same than the ROOT distribution. With that the OnX/RootGui driver will be able to compile, link and run over a standard full ROOT distribution.
- A prototype runs right now that permits to have the LHCb event display (GaudiLab) running over the OnX/RootGui driver.
- Note that, within OnX, right now Inventor and HEPVis had been declared already to CINT. This permits to script Inventor scenes directly under CINT. It could be interesting to have the same in the ROOT distribution.

OnX gallery

At the forge. First view of the LHCb event display using the OnX/RootGui driver and Inventor (here SoFree implementation).



Open questions, further developments...



- On the structural point of view, we may question the need, at run time, of the CINT interpreter to build a GUI with the ROOT GUI classes. Despite the fact that CINT is used to build meta information for the classes, is it really needed at run time ? especially if someone may want to build a GUI with the ROOT GUI classes and have other interpreters connected to the component callbacks...
- Probably the ROOT 3D graphics, being over OpenGL, could be integrated in the system. A first draft could be reached by having one Inventor node (some SoROOT class) that triggers the OpenGL rendering of ROOT objects in its GLRenderAction method. The machinery is quite here in the [TRootOViewer](#) class. Having a SoROOT node (or a nodekit) will be very flexible in the sense that ROOT scenes may then be embedded in other Inventor scene graphs (for example HEPVis / SoPlotter histogram plotter!). We can imagine to refine the SoROOT node in order to restore picking and inject more scene graph logic over ROOT objects.

Open questions, further developments...



- A more serious desirable development will be to have the today ROOT 2D graphics (plotting) able to draw within the [OpenGL](#) and then [Inventor](#) context. This will permit to unify the ROOT graphics over one high level common layer, open the ROOT system toward experiments that have chosen Inventor, and most of all, have a huge potential of creativity in the graphics area...

preliminary job list for the next ten years...



- Declare Inventor and HEPVis to CINT within ROOT.
- Have all that running on Windows (could be done with SoFree).
- SoFree had been used right now ; have all that running with the SGI open source Inventor (on UNIXes and Windows).
- Have the SoROOT node.
- Have the ROOT 2D graphic over OpenGL and within Inventor.
- Restore picking.
- Have the ROOT "contextual popup menu" working over Inventor viewers.
- Study multithreading, non linear projection (theta-phi, fish eye, etc...) and "tracking problem" within Inventor
- Etc, etc, etc, etc, etc...