3D viewer offers another dimension for PVSS

If you have ever watched a science-fiction film where the cockpit of a spacecraft is shown, you will have spotted an animated 3D display showing an overview of the complete apparatus, with flashing red highlighting the damaged parts.

And what about the LHC and the experiments at CERN? The operators in the control rooms sit in front of screens displaying mysterious, yet standard-looking, operation panels (many of these run on top of the common application platform, PVSS). With the new 3D viewer developed by IT-CO, such 3D displays will become available.

Let's have a look at an example of the 3D viewer in action. In this series of images (figure 1), the state of a fictitious subdetector is shown. The colours of the working elements signify the temperature: green being OK, yellow signalling that attention is needed and red signalling danger of damage.

The idea of a 3D overview display showing the state of the detector first emerged in the CMS experiment. It was implemented by Robert Gomez-Reino and immediately attracted the attention of many users. However, it had an important limitation – being a complex blend of Java and ActiveX technologies, it could only be used on the Windows platform. This limitation turned out to be a show-stopper preventing the adaptation of the CMS display for use in other experiments.

Luckily, in recent versions of PVSS the Qt library was harnessed to power the user interface. The Qt application framework, known for its excellent cross-platform portability, is a workhorse for projects such as the KDE desktop environment (a leading, open-source desktop environment for Unix) Lucasfilm, Skype and Google Earth. Its open architecture allows us to extend the set of "widget'' types (elements used to create user-interface applications), using the same code-base on all platforms. PVSS makes full use of this feature, allowing the PVSS platform to be extended with custom "external widget object" extensions, written with Qt. This has been done for the 3D viewer presented in this article.

The rendering and animation of the 3D scene in the viewer is powered by the Coin3D library – an open-source implementation of the OpenInventor high-level programming interface for 3D graphics. The mechanism to embed the Coin3D-based graphics scene in a Qt application is also readily available from the Coin3D webpages, which made the initial integration of these software components straightforward. However, the set of shape types, offered out-of-the-box, quickly appeared insufficient – shapes representing a segment of a cone,

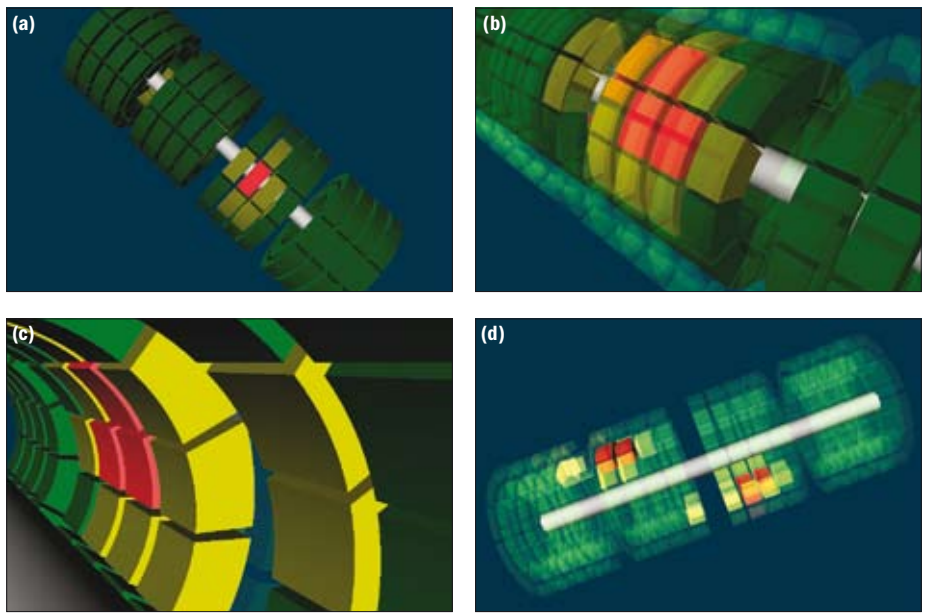

Fig. 1. (a) The operator spotted an anomaly in one of the cylinders. "Inner" elements are not visible. (b) Dimming the external layer reveals a problem in the inner cylinders. (c) "Flying" through the detectors allows inspection of the problem details. (d) Dimming all of the parts except the ones indicating problems allows for spatial correlation.

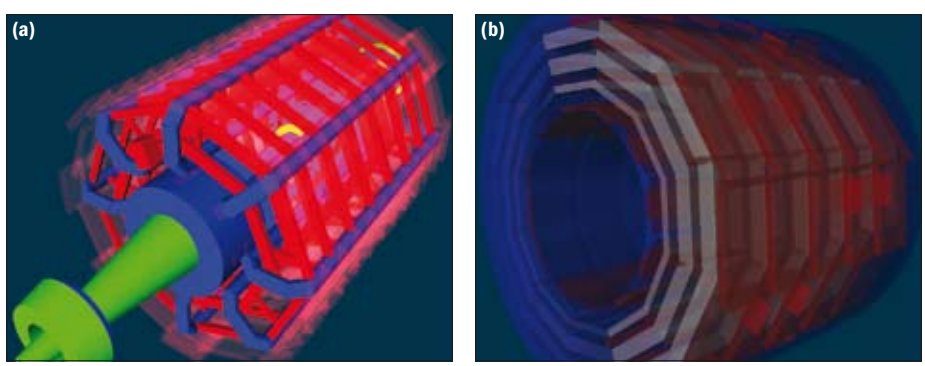

Fig. 2. Visualizations that make use of the data from the detectors' geometry databases. (a) Image from the ATLAS detector. (b) Image from the CMS muon subdetector.

polyhedra or trapezoids had to be added. Again, we were able to identify and reuse the code of an existing open-source project – HEPVis – that fitted our needs exactly.

The set of technologies being integrated does not, however, end here. To display the views of detectors, such as the ones in figure 2, the data stored in pre-existing geometry databases can be used. Unexpectedly, the data used mainly by the offline data-analysis programs find an exciting application in the detector online systems. All that is left to do is to establish the connection between the geometrical representation of the detector parts and actual parts of the existing control systems. Many subdetectors will soon have a 3D view of their hardware ready to be embedded into their operational panels.

The 3D viewer is not limited to visualizations of the detector. It is a generic, fully programmable UI widget, the power of which could be unleashed using PVSS scripting features, such as 3D histograms, charts and trend plots.

Useful links

3D viewer homepage (JCOP framework downloads page): http://itcobe.web.cern.ch/itcobe/Projects/Framework/Download/Components/3DViewer/welcome.html.
PVSS (process control and visualization): www.pvss.com.
Qt cross-platform application framework: http://trolltech.com/products/qt.
Coin3D high-level 3D graphics toolkit: www.coin3d.org/.
HEPVis class library extension to the OpenInventor toolkit; maintained in the OpenScientist code base: http://openscientist.lal.in2p3.fr/.

Piotr Golonka and Alvar Cuevas i Fajardo, IT-CO