

# A Graphics Sampler for the DØ Experiment

*G. Alverson*<sup>1</sup>, *K. Bos*<sup>2</sup>, *T. Burnett*<sup>3</sup>, *F. Canelli*<sup>4</sup>, *D. Coppage*<sup>5</sup>, *L. Duflot*<sup>6</sup>, *Y. Gershtein*<sup>7</sup>, *N. Graf*<sup>8</sup>, *S. Hagopian*<sup>9</sup>, *P. Hamel*<sup>10</sup>, *S. Kahn*<sup>8</sup>, *T. McMahon*<sup>11</sup>, *M. Narain*<sup>12</sup>, *N. Oshima*<sup>13</sup>, *O. Peters*<sup>2</sup>, *H. Prosper*<sup>9</sup>, *D. Shpakov*<sup>14</sup>, *T. Trippe*<sup>10</sup>, *J. Womersley*<sup>13</sup>, *S. Youssef*<sup>9</sup>

<sup>1</sup> Institute1, Northeastern University, USA

<sup>2</sup> Institute2, FOM-Institute NIKHEF and University of Amsterdam/NIKHEF, The Netherlands

<sup>3</sup> Institute3, University of Washington, USA

<sup>4</sup> Institute4, University of Rochester, USA

<sup>5</sup> Institute5, University of Kansas, Lawrence,

<sup>6</sup> Institute6, Laboratoire de l'Accélérateur Linéaire, IN2P3-CNRS, France

<sup>7</sup> Institute7, Brown University, USA

<sup>8</sup> Institute8, Brookhaven National Laboratory, USA

<sup>9</sup> Institute9, Florida State University, USA

<sup>10</sup> Institute10, Lawrence Berkeley National Laboratory, and University of California, Berkeley, USA

<sup>11</sup> Institute11, Langston University, USA

<sup>12</sup> Institute12, Boston University, USA

<sup>13</sup> Institute13, Fermi National Accelerator Laboratory, USA

<sup>14</sup> Institute14, State University of New York, Stony Brook, USA

## Abstract

The DØ Experiment has a variety of displays for various purposes. A graphics toolkit and interface for 3-D event displays, including detector elements, reconstructed hits and tracks and physics objects, has been implemented on top of OpenInventor and OpenGL. Displays of all major detector subsystems have been integrated, and levels of detail can be selected. Specialized displays for muon tracking and triggering and for silicon microstrip tracker alignment have also been written. An interactive 3-D event display based on HEPVis using Open Inventor and OpenGL is under development. A web-based virtual DØ control room JAVA applet is also available for remote viewing of the DØ control room.

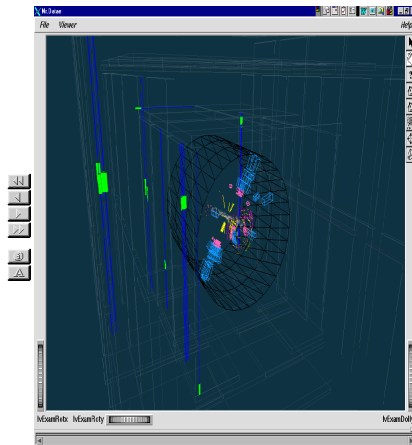
Keywords: graphics, data presentation, visualization, C++

## 1 Introduction

Graphical displays are used in collider experiments for detector design and commissioning, for online data quality control, for development and testing of algorithms for triggering and reconstruction and for the study of events for physics analysis. The DØ experiment at Fermilab has been working for several years to prepare for Run II, which is scheduled to take place 2001-2005, with analysis continuing for several years past the end of data taking. Displays are being developed which are not only useful now but during the course of the experiment.

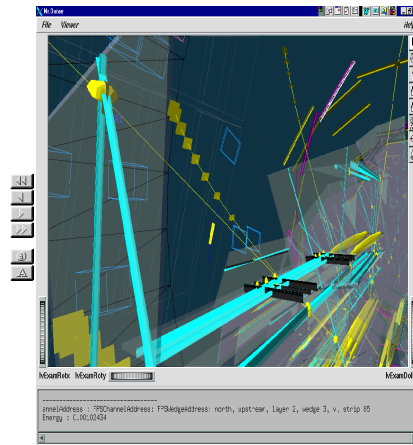
DØ graphics packages are written in C++, and are designed to be modular and object-oriented. OpenGL[1] is the basic graphical library underlying the displays. Many of these also use Open Inventor[2] and the HEPVis class library[3]. The graphical user interface (GUI), supported by HEPVis is also used in several of the displays. Most of the displays are constructed so that OpenGL can be replaced by some more advanced graphics library, if need be, at a later date. Most DØ displays run on Silicon Graphics (SGI) computers and PCs running Linux .

Some displays run under Windows NT.



Slide 3 of 5

(a)



Slide 2 of 2

(b)

**Figure 1:** (a) Display of a single muon track showing segments of the silicon detector (grey wedges), fiber tracker (blue rods) and calorimeter cells (wireframe boxes). Hits are represented by yellow error ellipsoids. The text window below the display provides pick feedback. (b) Display of a  $Z \rightarrow \mu\mu$  event showing hit silicon strips (aquamarine), reconstructed silicon hits with exaggerated scale (yellow ellipsoids), reconstructed tracks with exaggerated errors (yellow lines with surface patches), calorimeter cells (wireframe boxes), and preshower scintillators (yellow and cyan strips).

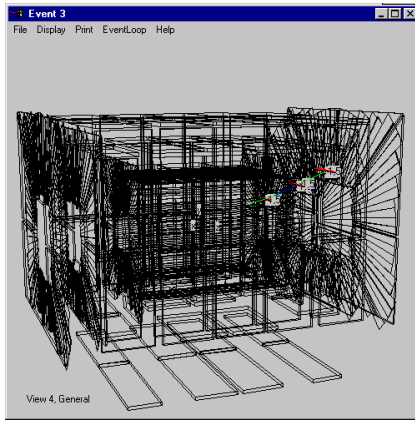
## 2 Mr.Datae – a 3–D Graphical Toolkit

Mr.Datae[4] is a graphics toolkit and interface for event displays for the Run II D0 detector and data. It provides displays for each of the sub-detector systems, which can be used in stand-alone displays or in an integrated display of the whole detector. It can also be easily used to make new displays tailored to specify needs. It achieves three goals: 1) it makes full use of existing software 2) it hides, behind an extremely simple interface, the complexity of 3D graphics software and 3) it cleanly isolates DØ graphics code from the underlying graphics package which may change in the future. Mr.Datae uses OpenInventor and HEPVis classes. It uses one of the HEPVis graphical user interfaces for a menu system. The basic graphics classes include "Shapes" such as hits, box and tracks, "Appearance" such as color, style and fonts, and "Auto Morphs" such as translation, rotation and scaling. Users can also include non-graphical objects in a display for pick feedback and can introduce pick-revealed details of a display using the "Details" class.

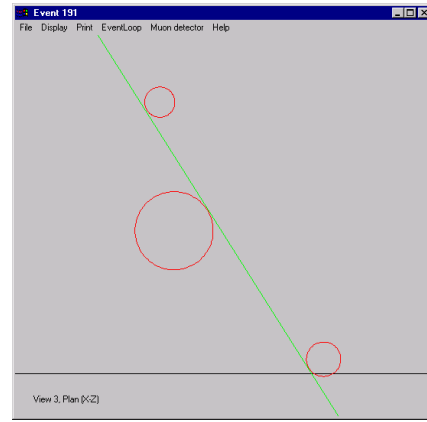
The system includes displays for the Silicon Microstrip Tracker, the Central Fiber tracker, the Muon Detector, the Central Preshower Detector, the Forward Preshower Detector and the Calorimeter as well as displays for global tracks, vertices and Lego plots of energy vs eta and phi. See Fig. 1 (a) and (b) for examples of Mr.Datae displays. Two dimensional displays for the calorimeter, inspired by the Aleph event display, DALI, are also being developed[5]. The framework, infrastructure, and sub-detector displays were implemented by various members of the D0 graphics group. See the D0 WWW Run II graphics page[4].

## 3 DØ VE – DØ Visualization Environment)

DØVE[6] is a visualization and control package designed for debugging of reconstruction code. It defines a 3-D display of lines, surfaces, and character data following the OpenGL model, and



(a)



(b)

**Figure 2:** (a) The muon geometry of the D0 detector. A MC track is also shown here as the blue line, with the wires which are hit in red. Part of the track reconstruction is shown in green. (b) A cosmic ray passing through one of the wire chambers of the D0 detector. The red circles are the drift circles measured and the green line is the reconstructed path of the cosmic ray. The two black lines are part of the chamber.

implements a default wire-frame viewer with keyboard control of orientation, perspective, and magnification of four independent views, which can be displayed simultaneously or individually. Its strength is in the rapid feedback allowed by tight integration with the code that is generating the geometrical data to be displayed, in that D0VE can be used with a debugger, and it provides functions that allow stepping through stages of reconstruction. That is, the user can step through code that is generating data with geometrical representations, and watch the evolution of the display. There is convenient access to the menu system of the native GUI (Windows or X), allowing user-defined buttons and actions. This control is used by the package to provide menu buttons that hide/show individual graphical element in the default display, with a cascading menu option.

D0Ve is currently being used to display the Silicon Tracker, the Fiber Tracker and the Muon Detector. These representations have been used to debug the software implementation of the geometry. See Fig. 2 (a) for a display of the muon detector geometry and a monte carlo track. It is also being used to display events for use in developing muon triggering algorithms and for a muon cosmic ray commissioning test (see Fig. 2 (b))

## 4 D0Scan

D0Scan[7] is an event viewer for D0 using OpenInventor and the HEPVis toolkit. The D0Scan design is based upon the HEPVis Twig class. This class provides a very flexible set of tools for organizing an OpenInventor scene, creating a GUI to control visibility of items in a scene, and centralizing picking callback Items such as detector components, hits, tracks and so forth are implemented as Twigs and arranged in a hierarchical structure similar to an OpenInventor scene graph. This structure is then automatically represented on a visibility control button panel. This approach provides a means to separate physics information from graphics information, so that, for example, picking callbacks can retrieve information about an object in the scene without the need to store that information in the graphics object itself. D0Scan is an integrated program. The overall goal is to provide a robust, consistent, relatively full-featured program to allow the average experimenter to examine the physics of individual events at D0.

## 5 Visual Alignment Package

A visual alignment package, which uses 3-D residual vectors is being developed for DØ using the Silicon Microstrip Detector (SMT) as a prototype[8]. The 3-D residuals contain all the information about track orientation and detector geometry needed to do alignment. The method is therefore detector independent. At each alignment step the user can display selected objects, tracks, related residual vectors and the  $\chi^2$ , rotation and translation for this step. The geometry display for the DØ Silicon Microstrip Detector was developed for this package, using HEPVis and Open Inventor. Since the DØ detector has trapezoidal wedge shaped disks as well as rectangular barrel ladders, two new shapes were developed and added to HEPVis, one for single sided wedges and one for double sided wedges. Using the SMT display, an inconsistency between the engineering drawings being used to build the detector, and the Monte Carlo geometry, was discovered. This display will be used to do a visual alignment of the silicon detector, and later extended to other sub-detectors.

## 6 DØ Virtual Control Room

A prototype web-based virtual DØ control room JAVA applet has been developed for remote viewing of the DØ control room[9]. It has screens showing the Main Control room, an Accelerator Status Screen, a DØ Run Status Screen, screens for triggering information, and a Event Display. Typically, screen data is automatically updated every 5-20 seconds. Other data, such as an event display, is updated on demand by the user of the VCR client.

## 7 Summary

DØ has a variety of graphical displays corresponding to a variety of needs for visualizing the detector and data. But all the displays have a common basis, using available state-of-the art graphics libraries. These displays should serve the collaboration well for the upgraded Tevatron era.

## 8 Acknowledgements

The DØ graphics group would like to thank the Fermilab Physics Analysis group, especially Jeff Kallenbach and Amber Boehnlein, for their support and expertise, which has helped us over the last few years. Special thanks to Jim Kowalkowski for his work on the interactive framework.

## References

- 1 <http://www.tgs.com/Products/opengl-index.html>
- 2 <http://http://www.tgs.com/Products/openinv-index.html>
- 3 <http://cactus.phyast.pitt.edu/~joe/hepvis/hepvis.html>
- 4 S. Youssef, et. al.,<http://www-d0.fnal.gov/~youssef/graphics/d0graphics.html>
- 5 L. Duflot; <http://www-d0.fnal.gov/~duflot/evdisp2D/evdisp2D.html>
- 6 T. Burnett and O. Peters; <http://www-d0.fnal.gov/d0dist/dist/packages/d0ve/v00-04-07/doc/>
- 7 G. Alverson; <http://www-d0.fnal.gov/~alverson/Graphics/D0Scan.html>
- 8 T. Trippe and P. Hamel; [http://d0lbt.lbl.gov/wwwtrippe/trippe\\_home.html](http://d0lbt.lbl.gov/wwwtrippe/trippe_home.html)
- 9 T. McMahon; <http://192.135.141.16/~mcmahon/vvcr/vcr.html>