Geomview Interface for Epics
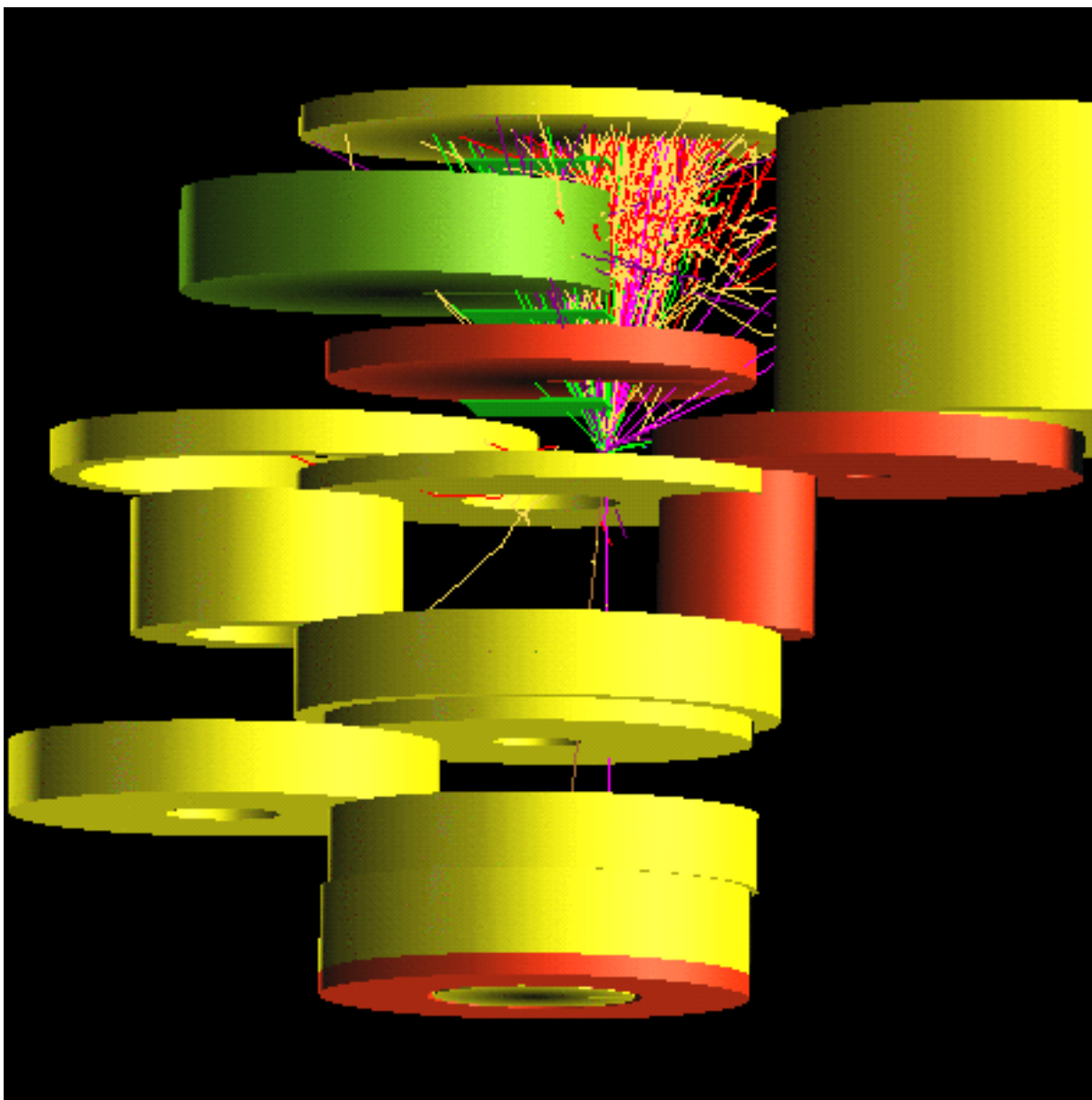

1 What is Geomview ?

This is explained in the manual 'Geomview Interface for Cosmos'. The important URL's are
http://www.geom.umn.edu/ and http://www.geomview.org/. The latest version for PC Linux is
1.8.1-2.

2 How to use Geomview within Epics

2.1 Preparation

Some of the following processes are the same as in Cosmos, and you may skip such ones.

- You have to setup an environmental variable GEOMVIEW to be the path to the Geomview executable For example, if you use csh,

  setenv GEOMVIEW /usr/bin/geomview (probably, for Linux)

  or

  setenv GEOMVIEW /usr/local/Apps/Geomview.app/Geomview (probably for NEXTSTEP)

- Geomview should normally be used in the *Epics*/Util/Geomview directory. If you are a NEXTSTEP user, you have to copy '.geomview' in this directory to your home directory. On other systems, having '.geomview' in the home directory is also not a bad idea.

- Setup an environmental variable

  GEOMVIEWTEXT

  to be the path to a perl script 'hvectext'.

  For examle, if in csh,

  setenv GEOMVIEWTEXT /usr/bin/hvectext (probably for Linux)

  or

  setenv GEOMVIEWTEXT /usr/local/Apps/Geomview.app/bin/next/hvectext (probably for NEXTSTEP).

  If you cannot find 'hvectext', it can be downloaded from the Epics Web site. If GEOMVIEW-TEXT is not given, you cannot display the color and media correspondence (not so important).

- The scripts in *Epics*/Util/Geomview assumes 'tcsh' in /usr/local/bin/. (On some systems, 'csh' would not run correctly). You might have to modify `#!/usr/local/bin/tcsh` in the top of the scripts or you might have to establish the link of 'tcsh' to /usr/local/bin/tcsh.

- Geomview is used to display detector configuration and particle tracks independently or simultaneously. For track display, you have to establish a working directory by editing 'disptracebygeomv' file where you will see 'vdir' at the top part. The default place is ../Work where some data for detector display is always put. The system stores particle track data for Geomview in that directory as 'ptcl2p1.vect' etc. You would also like to adjust the particle type to be displayed. The default setting is seen in 'ptcls' below 'vdir'; it dose not include neutrinos (code 7 and 8).

It may be worthwhile to remember that
    $GEOMVIEW object1 object2 ...
invokes Geomview to display Geomview objects specified by object1, object2,...

2.2 Displaying detector configuration

To display detector configuration, we utilize the data prepared for gnuplot so that we must go to *Epics*/Util and issue

- drawconfig

- If you are going to use new volume-shapes first time, issue

  ./usenewvol *config* followed by drawconfig.

- In drawconfig, you should determine how you would like to display the configuration, and finally select Menu 1. This will create data which can be used by gnuplot. You should be careful about that if the number of components is too large, some deeper level components may not be put in the data. In such a case, drawconfig shows a message that you can change the level.

- Open a new window and go to Epics/Util/Geomview.

- Issue

  ./dispconfigbygeomv

  This will show the detector; in some case you need to push the Look button.

- Optional parameters may be added as

  ./dispconfigbygeomv [-e][-i] | [-h]

  If -e is added, every component can be recognized in Geomview. If not, only the level 0 component is displayed; you can see the subdetector but cannot see the components contained in the subdetector. If the number of components are very large, -e would not be practical. If -i is added, you will see another Geomview window which displays media and color correspondence (color index). The -h is used to show the brief usage of the command.

  The color index display can be shown by

  ./showcolormap

  independently.

- If you have data for gnuplot in a directory other than ../Work, you can specify it as

  ./dispconfigbygeomv [-e] [-i] |[-h] *dir config.X config.Y...*

  where *dir* is the working directory, *config.X, config.Y...* represent config.Pb, config.Al etc in that directory.

2.3   Displaying tracks optionally with the detector

In what follows, you have to give an explicit data for an italicized parameter, and block letters must be given as it is. *tracedata* means the path to the trace data, say, /tmp/foo/trace1.

1. Only trace data is displayed.

   ./disptracebygeomv [-c|z|b] *tracedata*

   You may add an optional parameter -c, -z or -b. The -c (default) is to show only charged particles. That is

   ./disptracebygeomv *tracedata*

   and

   ./disptracebygeomv -c *tracedata*

   are equivalent. The -z is to show only neutral (zero charge) particles. The -b for showing both charged and neutral particles. Therefore, if 'ptcls' defined in 'disptracebygeomv' contains 1 (code for gamma ray), the default invocation of disptracebygeomv will not show gamma rays.

2. Trace data + detector

   ./disptracebygeomv [-{c|z|b}] *tracedata* det *option*

   where *option* is the same as the option to 'dispconfigbygeomv'.

If you issue
    ./disptracebygeomv
a brief summary of the command usage is displayed.

2.4   Some further details

- The disptracebygeomv command creates ptcl1z1.vect, ptcl2p1.vect, ptcl2n1.vect etc in the working directory. The number after 'ptcl' is the particle code; 1 means gamma rays, 2 electrons etc. Then, 'z', 'p', or 'n' comes;. 'z' is for zero charge, 'p' for positive and 'n' for negative charge. The number that comes after this is normally 1. However, if there is a very long consecutive track, we may see ptcl2p1.vect, ptcl2p2.vect etc. This is due to the Geomview limitation that it cannot treats 32k vectors at once and we have to divide the data into more than two files. Such a long track may not be produced when we trace particles in an ordinary detector.

  Be careful that these vector data can grow very large and is left in the working directory.

- You can specify the media color by 'colormap' and the particle color by 'colortab' file. (The color can also be changed within the Geomview window).
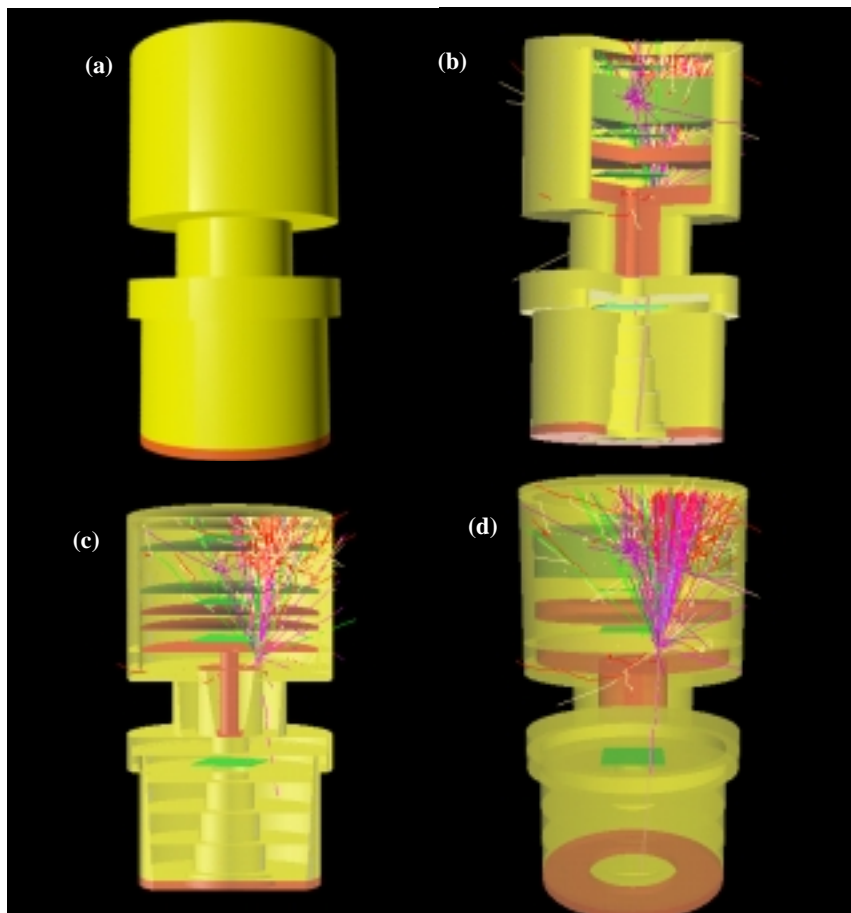

Figure 1: Various ways for displaying a detector and cascade

3   How can you examine the configuration ?

Beside demonstration purpose, Geomview display can be used to examine whether you could setup a config file correctly and the cascading there is as expected or not. You can use various approaches for this purpose. Take a detector shown in Fig.1(a).

- At 'drawconfig', make some part of box, sphere, or cylinder-like objects transparent so that we can see the inside of the detector. (b)

- Clip the view by adjusting the near clip plane of the Camera object. (c)

- Use semi transparent display at Geomview. (d)

- Move a desired component and see the inside. (Fig. in the front page).

It may be noted that it is sometimes better to use orthographic view rather than default perspective view of the 'Camera', especially if you want to make a view which shows detector dimensions[1].

4   Creating animation and taking still pictures

Among the external modules available for Geomview, StageTools would be one of the most useful ones. One module from that, StageManager, enables us to make mpeg animation, and StageStill to take a snapshot in various formats (eps, pnm etc).

To make these work, mpeg_encode and tcl/tk must be available for use. The Linux environment is ready for it.

To be able to use these for texture mapped image, one has to enable 'hardware shading' (though shading is actually performed by software).

Again, texture mapping is capriccio and in some case you cannot make a desired movie from such a image. In that case, you may use StageManager only for preview, and capture the image by the 'mkmovie' command (in *Epics*/Scrpt; the import command for capturing must be available). You have to increase the number of frames specified in StageManager by about 10 times (since capture rate is slow).

An example of animation is on the web page.

---

[1] Controlling orthograpic view on the NEXTSTEP verion of Geomviw is somewhat difficult due to unknown reason.