PHENIX Technical Note

P.O.V.: PHENIX Offline Visualization

by
Jeffery T. Mitchell (Brookhaven National Laboratory)
and
Jerome Lauret (SUNY-Stony Brook)

5/31/01

Abstract

This Technical Note will describe the details of an event display for PHENIX data and simulations that is suitable for generating Public Relations quality pictures for distribution.

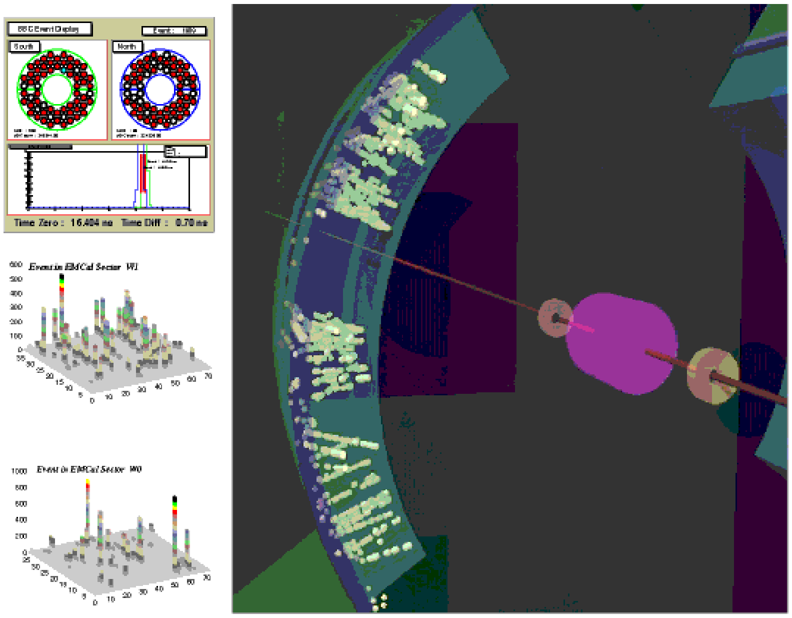
1. Introduction

P.O.V. stands for the **P**HENIX **O**ffline **V**isualization package. This package was developed to serve primarily as a Public Relations Event Visualization for PHENIX. This is the package that produced the picture of the first PHENIX collision within a couple of hours after it was recorded (see Figure 1). That picture was included in the press release about RHIC's first collisions. A P.O.V. display has also appeared in the October 2000 issue of the *CERN Courier* (see Figure 3).

P.O.V. has many advantages for PHENIX event visualization. It is very flexible in the customization of what it displays by simply editing the scene files - geometry and coloring are easy to manipulate. The produced images are rendered using a ray tracing program, so effects like lighting, transparency, shadowing, etc. can be used to enhance the image. Also, the resolution of the output image can be adjusted to as high as 1280 x 1024 pixels.

The package is named P.O.V. since it uses the freeware, Open-Source package called Pov-Ray, or **P**ersistence **O**f **V**ision Ray Tracer as its rendering tool. You can download this package and see what it can do by going to the Pov-Ray web site at *http://www.povray.org/*. You must have access to a computer with this ray tracer installed in order to produce displays. Pov-Ray was chosen since it is free software and it is supported on many platforms including LINUX and Windows 98/NT. Pov-Ray is also easy to use, especially for those familiar with GEANT geometry.

Pov-Ray is the same package that has been used to generate many RHIC and PHENIX animations that have appeared on many national newscasts, including the Virtual Tour of PHENIX. You can view these animations at the web site *http://www.phenix.bnl.gov/WWW/software/luxor/ani/*.



PHENIX evidence of RHIC collisions recorded Thursday morning, June 15 after accelerator operators brought beams into collision in the PHENIX interaction region.
Signals were recorded simultaneously in seven PHENIX detector subsystems: Zero-Degree Calorimeters (ZDC), Beam-Beam Counters (BBC), Time-of –Flight Detectors, Pad-Chambers, Drift Chambers, Time Expansion Chambers, and Electromagnetic Calorimeters (EMCal).
The event display (upper right) shows reconstructed tracks pointing to the collision.
The BBC (upper left) confirms that the collision occurred near the center of the PHENIX detector system. The EMCal displays (lower left) show the energy deposited by particles coming from the collision.

Figure 1: PHENIX P.O.V. event display produced within a couple of hours after PHENIX's first recorded RHIC collision.

2. The P.O.V. Analysis Package

The software necessary for generating a P.O.V. display from either real or simulated data can be found in the PHENIX repository under */offline/analysis/POV*. Up-to-date detailed information on the POV package can be found on the web at the URL *http://www.phenix.bnl.gov/WWW/software/luxor/uti/pov/.*

Classes are provided in the POV package to produce visualizations from PHENIX DSTs (using the class named *dstPOV*) and from PISA output ROOT files (using the class named *simPOV*). For both, the general procedure is the same (see the next section for more details on how to generate a picture). You must first have a DST or PISA file available. Then you must execute the corresponding ROOT macro (within ROOT) that will run the package on the event number that you specify. The package will produce a Pov-Ray hits file containing any specified hit and track information. Finally, you must run Pov-Ray on the Pov-Ray input file that contains the PHENIX geometry and pulls in the hits file. You will find that generating the Pov-Ray hits file within ROOT is nearly instantaneous once the requested event is read. However, for a central Au+Au event, it takes Pov-Ray nearly 5 minutes to render the event on a 450 MHz LINUX system at 460x460 pixel resolution.

3. Producing a Display From a DST File

This section outlines a step-by-step guide describing how to go from a PHENIX raw data file to a P.O.V. picture.

- Obtain the P.O.V. package by typing *cvs checkout offline/analysis/POV* in your working directory.

- Use the standard PHENIX procedure to build the *pov* package libraries. Make sure that your *LD_LIBRARY_PATH* environment variable contains the location of the libraries.

- Run the standard CAMERA chain on a raw data file to produce a DST file.

- Select the event number (in sequence on the DST) which you wish to display.

- Generate a Pov-Ray hits file from the DST file (default name: *povTest.pov*). In order to do this, follow the following steps:

  - Edit *povData.C* (in the macros directory of the package) in your running directory. This is the ROOT macro that will run the *dstPOV* methods on the DST file. Set the input parameters to the desired values (including the event number and the name of the output Pov-Ray file). See the web documentation for updated descriptions of the input parameters.

  - Run ROOT.

  - Type *.x povData.C()* to execute the macro within ROOT. The macro will skip to the event number you have selected and then analyze that event before stopping.

  - Type .quit to get out of ROOT.

- Run Pov-Ray on the output Pov-Ray hits file using the default PHENIX Pov-Ray set-up file that contains the geometry description (*/macros/PHENIX.pov* in the package). If you are not using the default file names, you will need to edit *PHENIX.pov* to change that name (near the bottom of the file). The camera position is defined near the top of the *PHENIX.pov* file. If you are on a UNIX system, you can generate a plot with the following sample command:

  ```
  povray +i PHENIX.pov -geometry 400x400 -display :0.0 &
  ```

See the Pov-Ray documentation for more details about how to run the rendering program. The *PHENIX.pov* file includes the generated Pov-Ray file from the previous step. You will see the resulting display, which will be saved to disk in the format specified in your Pov-Ray initialization file.

4. Producing a Display From a Simulation (PISA) File

This section outlines a step-by-step guide describing how to go from a PHENIX PISA output ROOT file to a P.O.V. picture.

- Obtain the P.O.V. package by typing *cvs checkout offline/analysis/POV* in your working directory.

- Use the standard PHENIX procedure to build the *pov* package libraries. Make sure that your *LD_LIBRARY_PATH* environment variable contains the location of the libraries.

- Run the standard CAMERA chain on a raw data file to produce a DST file.

- Select the event number (in sequence in the PISA file) which you wish to display.

- Generate a Pov-Ray hits file from the PISA file (default name: *povTest.pov*). In order to do this, follow the following steps:

  - Edit *povGEANT.C* (in the macros directory of the package) in your running directory. This is the ROOT macro that will run the *simPOV* methods on the PISA file. Set the input parameters to the desired values (including the event number and the name of the output Pov-Ray file). See the web documentation for updated descriptions of the input parameters.

  - Run ROOT.

  - Type *.x povGEANT.C()* to execute the macro within ROOT. The macro will skip to the event number you have selected and then analyze that event before stopping.

  - Type .quit to get out of ROOT.

- Run Pov-Ray on the output Pov-Ray hits file using the default PHENIX Pov-Ray set-up file that contains the geometry description (*/macros/PHENIX.pov* in the package). If you are not using the default file names, you will need to edit *PHENIX.pov* to change that name (near the bottom of the file). The camera position is defined near the top of the *PHENIX.pov* file. If you are on a UNIX system, you can generate a plot with the following sample command:

```
povray +i PHENIX.pov -geometry 400x400 -display :0.0 &
```

See the Pov-Ray documentation for more details about how to run the rendering program. The *PHENIX.pov* file includes the generated Pov-Ray file from the previous step. You will see the resulting display, which will be saved to disk in the format specified in your Pov-Ray initialization file.

5. Visualization Implementation Details

This section will describe the details of the implementation of the visualization of the hits and tracks for both the simulation and the data.

Although this package may have some uses as a tracking debugging tool, it is limited by the fact that several steps by the user are necessary to produce a picture, and the turn-around time is not immediate from data taking to the picture. That is, this package is not directly interactive. It is best used when you have a real cool event and you know where you would like to place the camera for visualization, and then letting it take the snapshot for distribution.

At the time of the writing of this Note, the package is supported for v03 DSTs and PISA99 GEANT files running within ROOT version 2.23 using the pro.6 libraries. For reference, the package in this configuration will analyze files used for analyses leading up to production of the QM2001 Proceedings. See the web documentation for the current support status of this package.

For visualization of simulated (GEANT) data, only the GEANT hits and tracks information is used. For each detector, the 3-dimensional Cartesian coordinates of each recorded GEANT hit are used in the display. Activation of the hits display for a given detector is handled through the user-controlled data members of the *simPOV* class, which can be set within the ROOT macro. Hit associations into tracks are facilitated by using the utilities in the *cge* (**C**entral Arm **G**lobal **E**valuations) package. Documentation on *cge* can be found in *PHENIX Technical Note 384* and on the web at the URL *http://www.phenix.bnl.gov/WWW/software/luxor/cge/package/index.html*. The *cge* utilities return GEANT hit-to-track relation information, which is then written to the output Pov-Ray hits file upon user request. Figure 2 shows a P.O.V. display from a central HIJING Au+Au PISA99 ROOT file in the Run 2000 detector configuration with no magnetic field with *dch+pc1+pc2+pc3* associated GEANT tracks shown.

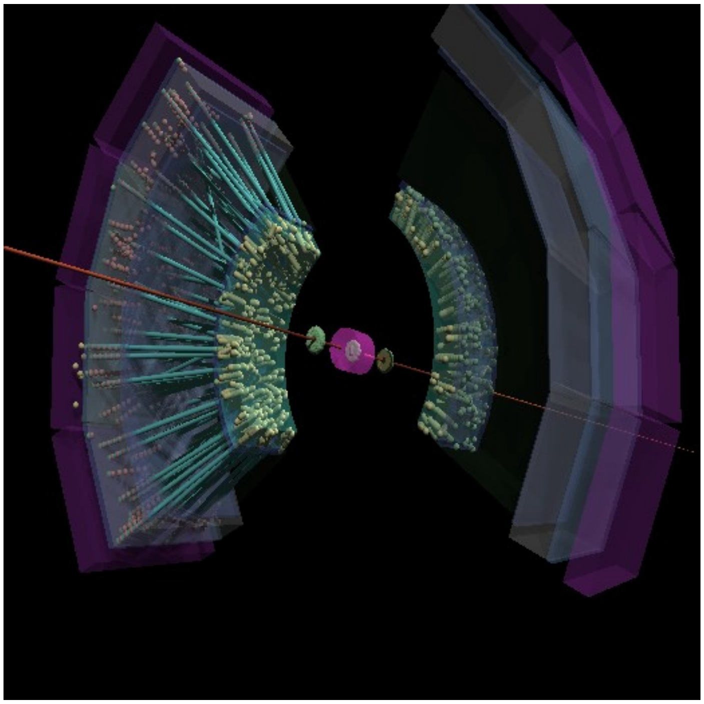


Figure 2: P.O.V. visualization from a Run 2000 central HIJING Au+Au PISA99 file (B=0). Shown are GEANT hits from the BBC, MVD, DCH, PC1, PC2, PC3, TEC, and TOF. Track associations between DCH+PC1+PC2+PC3 are also shown.

For visualization of analyzed DST data, the situation is complicated by the fact that the drift chamber and the time expansion chamber (TEC) both are projective detectors and provide only 2-dimensional information for their recorded hits. For drift chamber hits, it is assumed that the z information for that hit can be estimated by the drift chamber reconstruction. So, if the z information for a drift chamber hit is not available, P.O.V. will not plot it, nor will it attempt to estimate it. The same applies to drift chamber tracks. If the drift chamber track does not contain z information, it is not plotted. For TEC tracks, the PC3 reconstructed clusters are queried and the TEC tracks are assigned the z-coordinate of the closest PC3 cluster to the TEC track in phi. The PC3 cluster z-coordinate must be consistent with the sign of the TEC track z coordinate to be used in the assignment. If a z-coordinate cannot be assigned, the TEC track is not plotted. It is planned to install a query to the reconstructed global tracks followed by an assignment of the TEC track z-coordinate from the projection estimate provided by the track model in the near future. For the remaining detectors, the 3-dimensional information is obtained directly from the DST quantities. For the calorimeters, an option is provided to represent reconstructed clusters as spikes proportional to their energy. There is also an option to draw straight line projections of the drift chamber tracks to the beam-line vertical plane. In the near future, an option will be added to plot the poly-line track model estimates. Figure 3 shows a P.O.V. visualizations from a DST reconstructed from data taken during Run 2000 with the magnetic field turned off.

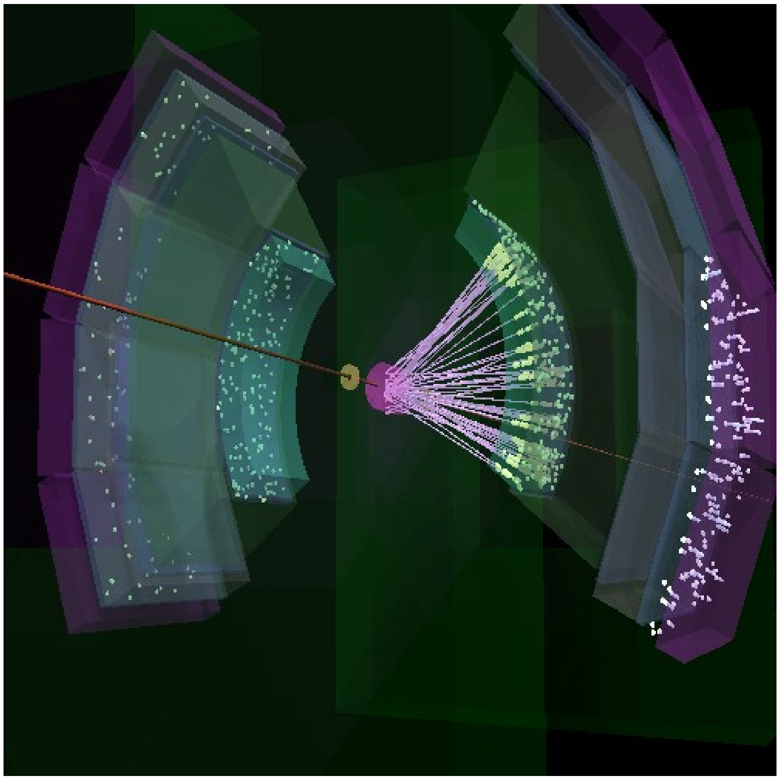


Figure 3: This picture shows one of the first events that PHENIX recorded. The yellow dots in the interior are drift chamber and pad chamber reconstructed hits. The white dots and lines on the exterior are pad chamber reconstructed hits and reconstructed calorimeter clusters with the line length proportional to the energy deposited. On the far side are hits from the pad chamber and the time expansion chamber. The pink lines are projections of the drift chamber reconstructed tracks to the vertex where the collision took place. For this event, there was no magnetic field.