

Visualization Developments and Plans

Joseph Perl

for the Geant4 Visualization WG

Contents of this Session

- ✦ Joseph Perl:
 - ✦ Highlights of the Current Beta
 - ✦ and what's left for the December release
 - ✦ gMocren and other new Volume Visualization Developments
 - ✦ from Satoshi, Akinori and Kyoko
- ✦ Laurent Garnier:
 - ✦ Qt
 - ✦ Posters from GL

Highlights of the current Beta - 1

- Added new `/vis/open` command for OpenGL viewers/`/vis/open OGL`
 - This new form allows the same macro to be used whether one is on Linux or Windows (calling `OGLSX` on Linux and `OGLSWin32` on Windows).
 - When the example has been built with Qt, it will use the Qt form of OpenGL.
 - When the example has been built with Motif, it will use the Motif form of OpenGL.
 - Also added options `OGLI` and `OGLS` to force Immediate or Stored use while still being somewhat generic.
 - The user is still free to use the older, specific viewer names, such as `OGLIX`, but the simple form, `OGL`, will give the most portable macros.
- Added new `/vis/open` command for OpenInventor, `/vis/open OI`
- Speed improvements for OpenGL
 - removing redundant transmissions of points and unnecessary redraws

Highlights of the current Beta - 2

- ✦ Modified trajectory modeling command
 - ✦ `/vis/modeling/trajectories/create/drawByParticleID`
 - ✦ Used to default all trajectories to grey, with the user having to explicitly set any other color code.
 - ✦ While the user can still set any color code they wish, the default has been improved to:
 - ✦ e- - red
 - ✦ e+ - blue
 - ✦ proton - cyan
 - ✦ gamma - green
 - ✦ neutron - yellow
 - ✦ pi+, pi-, pi0 - magenta
 - ✦ other - grey
- ✦ Added new commands
 - ✦ `/vis/scene/add/digis` and `/vis/filtering/digi`
 - ✦ analogous to the already-existing commands
 - ✦ `/vis/scene/add/hits` and `/vis/filtering/hits`
 - ✦ Just as the user of hits visualization has to provide a hits class, the user of the digitization visualization has to provide a digi class

Revised Vis Usage in Most Examples

- Removed code from EventAction that can now be done by commands
 - such as adding trajectories
 - or setting color code
- Improved Vis macros
- We did not change the resulting behavior
 - except for adding step points to the trajectories in appropriate cases
- The new vis macros demonstrate the best and most flexible current usage

Typical new Vis.Mac (1)

- # Use this open statement to create an OpenGL view:
- /vis/open OGL 600x600-0+0
- #
- # Use this open statement to create a .prim file suitable for viewing in DAWN:
- #/vis/open DAWNFILE
- #
- # Use this open statement to create a .heprep file suitable for viewing in HepRApp:
- #/vis/open HepRepFile
- #
- # Use this open statement to create a .wrl file suitable for viewing in VRML viewer:
- #/vis/open VRML2FILE
- #
- # Disable auto refresh and quieten vis messages whilst scene and
- # trajectories are established:
- /vis/viewer/set/autoRefresh false
- /vis/verbose errors
- #
- # Draw geometry:
- /vis/drawVolume
- #
- # Specify view angle:
- #/vis/viewer/set/viewpointThetaPhi 90. 0.

Typical new Vis.Mac (2)

- # Specify zoom value:
- #/vis/viewer/zoom 2.
- #
- # Specify style (surface or wireframe):
- #/vis/viewer/set/style wireframe
- #
- # Draw coordinate axes:
- #/vis/scene/add/axes 0 0 0 1 m
- #
- # Draw smooth trajectories at end of event, showing trajectory points
- # as markers 2 pixels wide:
- /vis/scene/add/trajectories smooth
- /vis/modeling/trajectories/create/drawByCharge
- /vis/modeling/trajectories/drawByCharge-0/default/setDrawStepPts true
- /vis/modeling/trajectories/drawByCharge-0/default/setStepPtsSize 2
- # (if too many tracks cause core dump => /tracking/storeTrajectory 0)
- #
- # Draw hits at end of event:
- #/vis/scene/add/hits

Typical new Vis.Mac (3)

- # To draw only gammas:
▪ #/vis/filtering/trajectories/create/particleFilter
▪ #/vis/filtering/trajectories/particleFilter-0/add gamma
▪ #
- # To invert the above, drawing all particles except gammas,
▪ # keep the above two lines but also add:
▪ #/vis/filtering/trajectories/particleFilter-0/invert true
▪ #
- # Many other options are available with /vis/modeling and /vis/filtering.
▪ # For example, to select colour by particle ID:
▪ #/vis/modeling/trajectories/create/drawByParticleID
▪ #/vis/modeling/trajectories/drawByParticleID-0/set e- blue
▪ #
- # To superimpose all of the events from a given run:
▪ #/vis/scene/endOfEventAction accumulate
▪ #
- # Re-establish auto refreshing and verbosity:
▪ /vis/viewer/set/autoRefresh true
▪ /vis/verbose warnings
▪ #
- # For file-based drivers, use this to create an empty detector view:
▪ #/vis/viewer/flush

News from Satoshi, Akinori and Kyoko

- ✦ DAWN has been updated.
 - ✦ Linux and Mac updates done.
 - ✦ Windows update on the way.
- ✦ gMocren now runs on all platforms
 - ✦ Windows
 - ✦ Linux
 - ✦ Mac
- ✦ Some gMocren images and a preview of something new
 - ✦ volume/surface fusion