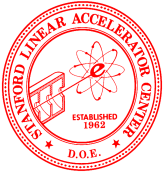# Extending EGS with SVG for track visualization

Ray F Cowan
rfc@slac.stanford.edu

Kanishka Shrivastava
kanishka@stanford.edu

Bebo White
bebo@slac.stanford.edu

Stanford Linear Accelerator Center, Stanford, CA 94309, USA.

Abstract:

The Electron Gamma Shower (EGS) Code System at SLAC is designed to simulate the flow of electrons, positrons and photons through matter at a wide range of energies. It has a large user base among the high-energy physics community and is often used as a teaching tool through a Web interface that allows program input and output. Our work aims to improve the user interaction and shower visualization model of the EGS Web interface. Currently, manipulation of the graphical output (a GIF file) is limited to simple operations like panning and zooming, and each such operation requires server-side calculations. We use SVG (Scalable Vector Graphics) to allow a much richer set of operations, letting users select a track and visualize it with the aid of 3-D rotations, adjustable particle display intensities, panning and zooming etc. A considerable advantage of our method is that once a track is selected for visualization, all further manipulations on that track can be done client-side without requiring server-side calculations. We hence combine the advantages of the SVG format (powerful interaction models over the Web) with those of conventional image formats (file size independent of scene complexity) to allow a composite set of operations for users, and enhance the value of EGS as a pedagogical tool.

Introduction: EGS

➢ **Simulates flow of e-, e+ and photons through matter at high energies**

➢ **Used by high energy physics community, and as a teaching tool**

Web Interface: EGS to order
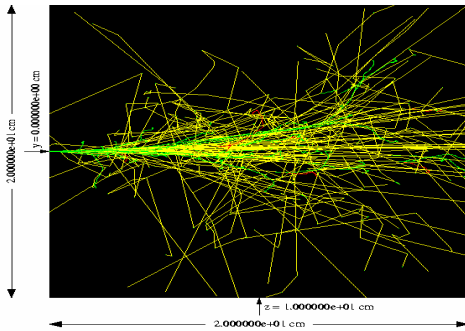
➢ **Enter parameters to EGS through a web form**

➢ **Should then allow operations like:**

 ✓ **Panning, zooming to see parts of interest**

 ✓ **Focus on the particles of interest only**

 ✓ **Pick a track and visualize it in detail**

 ✓ **Time-wise development of interactions**

➢ **Ideally, should be usable interactively**

➢ **Note: large data sets (typical: 140,000 points; 9000 tracks)**

Scalable Vector Graphics (SVG)

➢ **2-D graphics language**

➢ **Vector format (geometric shapes)**

➢ **W3C Recommendation**

➢ **Rich set of event handlers**

➢ **Can be added to HTML Web pages**

➢ **Native support in "modern" browsers**

Solution 1: GIF 'snapshots'
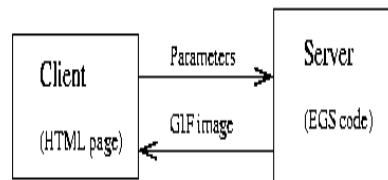
 ✓ **Allows panning and zooming**

 ✓ **Simple operations like turning off all e-s etc**

 ✓ **Can handle complex showers (GIF file size is constant)**

 ✘ **All computations done server-side (even for simple pan or zoom)**

 ✘ **Response time of several seconds => Not interactive**

Solution 2: SVG-based interface
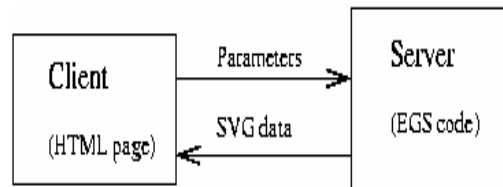
➢ **Entire EGS output is sent to the client**

➢ **Interface is built using SVG scripting**

➢ **Embedded in an HTML page**

 ✓ **No further server-side processing**

 ✓ **Richer and more intuitive interface**

 ✓ **3-D rotations for better visualization**

 ✓ **Adjustable intensities of e-, e+, photons**

 ✘ **NOT interactive (data sets too large)**

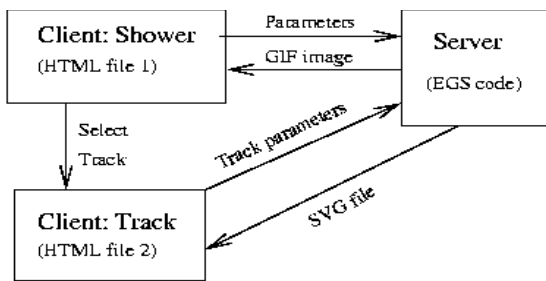 ✘ **Most action is at sub-pixel resolution, but still have to send all the data**

Insight: 2 kinds of operations

	Types of operations	Acceptable Delay	File sizes	Format
Shower level	Pan, zoom, view parameters	< 10 sec	Large (complete shower)	GIF
Track level	3-D rotation, adjustable intensities, pan, zoom, time-based animation	< 1 sec (interactive)	Small (one track + descendants)	SVG
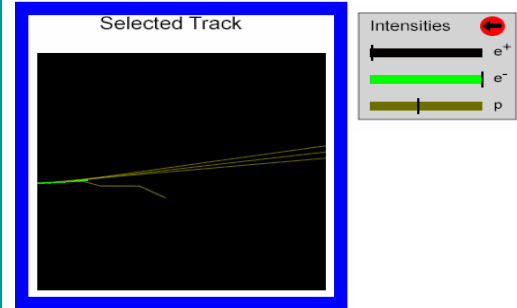
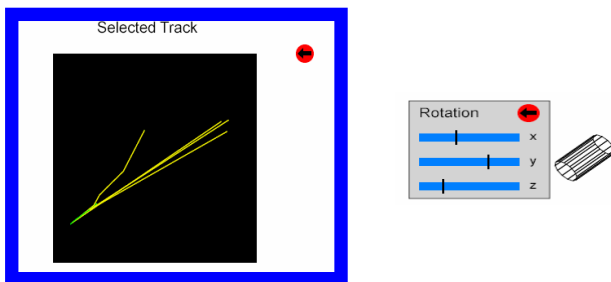### Solution 3: Best of both worlds

Advantages

✓ **No server-side processing after track is selected**

✓ **Only send constant size GIF snapshots of the full shower**

✓ **Interactivity where it is most needed**

Features of the new interface

✓ **3-D rotations**

✓ **Easy panning and zooming**

• **Just click and drag!**

✓ **Adjustable pixel intensities**

Implementation

➢ **Each track has an 'ID' and 'parent ID' for selection assigned by EGS**

➢ **SVG inherently 2-D, but XML based: can add z co-ordinate as an attribute**

Extensions and Future Work

➢ **Time-based animation**

➢ **Identify type of interaction (Compton scattering, etc) and display**

➢ **Maintaining the track ID and parent ID has uses beyond the Web interface**

Conclusions

➢ **SVG useful for event visualization**

  ✓ Rich set of event handlers

  ✓ Good for geometrical shapes

  ✓ Embeddable in HTML pages

➢ **EGS enhanced as pedagogical tool**

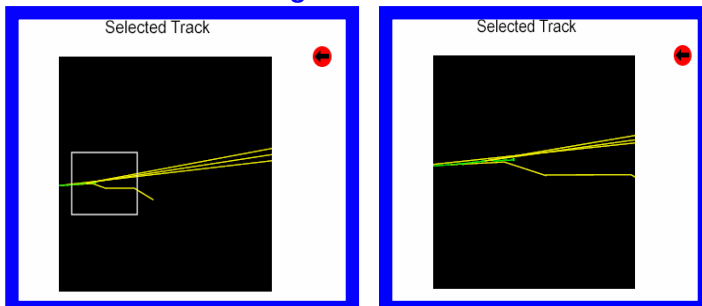 ✓ Can now visualize tracks better

  ✓ Combine advantages of SVG and GIF

References

• **SVG:** www.w3.org/Graphics/SVG

• **EGS:** www.slac.stanford.edu/egs/