A WEB BROWSER FOR LHC DESIGN INTEGRATION

SILVANO DE GENNARO

Computer & Networks Division
CERN - European Organization for Particle Physics Research
Geneva, Switzerland

silvano@vxcern.cern.ch

This paper describes ***i3D***, a WWW 3D graphic browser which is currently being developed within the scope of the VENUS project at CERN. ***i3D*** allows VR style navigation within the LHC Virtual Prototypes (see abstract 103), loaded over standard WWW protocols from the VENUS server. The user can explore these data models, converted from EUCLID to the *i3D* format, and trigger WWW standard hyperlinks by simply clicking on graphic objects. Some of these objects will cause the loading of new 3D *worlds*, extending the concept of hyperlinks to the one of *hyperworlds*. VENUS intends to develop *i3D*'s functionality and performance in order to use it as a graphic model browser within the scope of LHC's Engineering Design Integration.

1 The background

The World Wide Web and Virtual Reality are two emerging fields of computing that attract wide attention from the media and several fields of Science, Art and Communication. In the world of the Web there has been for some time an open discussion on the possibility of integrating the two fields, by implementing a standard that would allow the carrying of Virtual Environments over WWW. These discussions, initiated during the WWW'94 conference at CERN, gave origin to a number of initiatives, one of which, named VRML (Virtual Reality Markup Language) [1], became particularly successful and was used in the implementation of the first 3D Web browser, named WebSpace and produced by Silicon Graphics.

WebSpace is based on another SGI product, OpenInventor, which is available on a number of multivendor platforms. Although this allows WebSpace to be easily portable, it introduces an added complexity at the level of the rendering, that limits WebSpace in terms of graphic speed.

During the summer of 1994 the VENUS project and CRS4 [2] signed a joint collaboration agreement for the development of a high performance 3D web browser, particularly equipped to deal with large territorial and engineering models. This tool was named ***i3D*** [3] and was intended to be used within CRS4's "Virtual Sardinia" project, and for LHC Engineering Design Integration at CERN.

2 The project

i3D V1.0 was made available at no charge on the public domain in June '95. Its first implementation is limited to SGI systems running IRIX 5.2 or later.

Like image or video viewers, *i3D* is spawn by HTML browsers when clicking on an i3D type hyperlink. After loading the chosen model, i3D will let the user fly within the virtual environment and explore it from within, like conventional VR tools. The navigation is controlled with the mouse, mapped on a 3D abstraction named the "flyball", or with a Spaceball. During the flight, the User can click on objects associated with standard WWW hyperlinks, recognizable by a blue wireframe around them. This will cause the HTML browser to load WWW pages, or to spawn other viewers, depending upon the hyperlink type. If the hyperlink is of *i3D* or VRML type, it will cause *i3D* to load a new *world.* We will call this a *hyperworld* and we will refer to the object associated to its hyperlink as to a *hyperdoor*. The hyperworld mechanism allows to user to explore large and complex virtual environments without the need for expensive high end VR platforms. Each hyperworld loaded during the flight is kept in a "world stack", that the user can browse through using the arrow keys.

3 The implementation

i3D is targeted to work efficiently on low cost 3D graphics workstations, and combines the most effective optimization techniques of high-end Virtual Reality navigation tools and WWW browsers. Its web interface uses CGI to communicate with Mosaic or Netscape. On the VR side it supports its own file format *".3d"* or VRML *".vrml"* or *".wrl"*, and features culling, automatic LOD (level of detail), texture mapping and stereo view. It also supports a limited number of VR metaphors such as the Spaceball and Stereographics shutter glasses. We will briefly cover some of these techniques.

Culling: During the navigation, the user's viewpoint is attached to a "camera" object, whose parameters can be specified in the model description. A camera has a given "Field of View" (FOV), defined as a rectangular prism with its vertex in the camera and the base towards the direction where the camera is pointing. The FOV is truncated by a given "Near Clipping Plane" and a "Far Clipping Plane", delimiting the visibility in terms of distance. **Culling** is the technique of displaying and rendering only the objects whose bounding box intersects the Field of View. If the model is properly optimized, by breaking large assemblies into smaller objects, culling can be very effective and can increase the frame rate enormously.

Level Of Detail: Each object in the scene can be associated to a number of different geometry descriptions, featuring an increasing level of detail, i.e. a higher number of vertexes. For example, to render a sphere smoothly you may need 250 facets, however if you move away from the sphere, you may not need that amount of detail to have the illusion that it is round. 20 metres away, you may represent it with 50 facets only, and 40 metres away it may become just a dot and you may need perhaps 8 facets or less. The usage of vertex normals help greatly the illusion of smoothness. *i3D* automatically selects the appropriate level of detail, depending upon the distance from the object and the system's stress level.

Texture Mapping: This technique is used to improve the visual aspect of an object and to reduce the amount of geometry needed to display objects. By mapping a wood texture on a wooden object, it becomes photorealistic. A fully textured world augments greatly

the illusion of "being there". On the other hand, if you need to represent a beam with 100 screws, you may model each single screw, resulting in a very high polygon count, or simply take a photograph of the screws and map it onto the beam. It should be noted however that texture mapping requires a special graphic card. Textures can penalize enormously the rendering times if they have to be computed in the CPU.

Stereo View: the illusion of immersion in Virtual Reality is based on stereoscopic view. In real life we see in stereo, which means that each eye sees the world around us from a viewpoint a little bit away from the other. In VR, the application computes two images, one per eye, taken from viewpoints offset by an angle of a few degrees, corresponding to the distance of your eyes. This distance can usually be adjusted with a parameter.

Spaceball: The Spaceball is a 6 degrees of freedom device, ideal for on-screen navigation in 3D worlds. Different from game-like joysticks, the Spaceball allows control of the movement by acting on all the 6 degrees at the same time. The handling is instinctive and mastered with a little practice.

Shutter glasses: The user wears a pair of LCD glasses, that can shut each eye on an off at a frequency of 30Hz, in synchrony with the appearance on the screen of the right or left eye images, computed by the application supporting stereo view.

4 *i3D* and LHC

The LHC detectors will be designed in hundreds of different institutes, scattered all over the planet. In these conditions, the design validation and integration is an extremely difficult task. By its ability to load geometry via hyperlinks, *i3D* is the ideal integration stage. Its data models don't have to reside in any place in particular, but can be scattered everywhere. Each object can be loaded on line from wherever it is being designed, and the whole scene integrates and materializes on your screen. In fact the user's screen is the only place where the virtual prototypes really exist.

The advantages of using *i3D* for LHC design integration are enormous. By using a database server it is possible to manage centrally the hyperlinks and hyperworlds structure, in order to keep control of the overall project. For instance, the experiment's management controls the naming of the parts and associates URL's linking to geometry, drawings, home pages etc. *i3D* will then query this database to obtain the appropriate information whenever the user selects a graphic hyperlink or a hyperdoor. Navigating through LHC with *i3D* on your workstation will not only allow you to look around, but also to obtain all sorts of technical or organizational information on each single part.

5 i3D's future

The *i3D* project has recently moved from CRS4 to CERN, as a research subproject of VENUS (see abstract 104). Its medium range programme of work includes porting to other UNIX™ workstations featuring OpenGL, improving its web interface by supporting other protocols than HTTP, feature parallel URL loading, and extending peripherals support to other VR metaphors, such as HMD and 3D mouse.

Extensive information on *i3D*'s usage and data formats can be found in *i3D*'s Home Page [3]. A more detailed presentation of its programming techniques is in [4].

Acknowledgements

i3D was designed and developed by Jean-Francis Balaguer [5] and Enrico Gobbetti [6]. The input format converters from *Wavefront* and *VRML* were provided by Antonio Ticca [7].

References

1 http://www.sdsc.edu/SDSC/Partners/vrml
2 http://www.crs4.it/
3 http://sgvenus.cern.ch/i3d/i3d-help.html
4 *"i3D: a High-Speed 3D Web Browser*", JF. Balaguer and E. Gobbetti, to appear in Proceedings of VRML'95 Symposium, San Diego, Dec. 95.
5 http://sgvenus.cern.ch/~balaguer
6 http:// www.crs4.it /~gobbetti
7 http://www.crs4.it/~mav