VEOS DESIGN GOALS

William Bricken
July 1992

Presented as part of the SIGGRAPH'92 full-day tutorial (described below)

VEOS consists of several software subsystems. The kernel manages processes, memory, and communication. The entity interface permits modeling objects in the environment, and the environment itself, in a consistent, object-oriented manner. The interaction tools empower a participant within the virtual environment. The design of VEOS reflects multiple objectives, many practical constraints, and some compromises. Most importantly, VEOS is a research prototype, intended to show the way rather than to be sold for profit. Thus it is constantly undergoing revision and iterative refinement.

SIGGRAPH'92 SHORT COURSE DESCRIPTION

Steve Bryson and William Bricken
July 1992

Implementation of Immersive Virtual Environments

Course #4: Full day course

Course chair

Steve Bryson -- NASA Ames (Virtual Windtunnel)

Lecturers

Chuck Blanchard -- VPL Research
William Bricken -- HITlab
Lew Hitchner -- NASA Ames (virtual planetary exploration)
Rick Jacoby -- NASA Ames (VIEW lab)
Creon Levit -- NASA Ames (Virtual Windtunnel)
Warren Robinett -- UNC

Description

This course will describe the implementation of fully immersive virtual environments. The integration of hardware, software, and program design resulting in creating the illusion of virtual worlds will be covered. Developers who have designed the most famous and successful virtual worlds (that's us!) will discuss their work on a detailed level.

Objectives

After taking this course, the attendee will have a greater understanding of how to develop a fully immersive interactive virtual environment. Solutions to difficulties in the integration of hardware and software into a responsive, high performance system will be covered. Through several examples, various options in the development of virtual worlds will be learned. The attendee will know how to select the hardware for a particular virtual environment, outline the appropriate software structure, and implement that structure in a way which will give the greatest possible performance.

Difficulty

This course will require a moderate maturity in graphics programming and some awareness of interfacing serial devices to computers. Concepts such as transformation matrices, use of graphic libraries and basic Cartesian coordinates will be assumed. No knowledge of virtual environments will be required.

Who should attend

This course is intended for those who wish to know how to design and implement working high-performance immersive interactive virtual environments.

Course level

intermediate

TENTATIVE Syllabus

I. Introduction to Virtual Environment Technology and Software Concepts -- Steve Bryson

Introduction
Body tracking hardware and software
Immersive display hardware and software
Other interface technologies
Human Factors issues
Computation and graphic environments
Basic system integration

II. How successful applications and systems were designed, by those who designed them

This will be a series of short (40 min) talks by the lecturers. They will describe the design philosophies behind their systems, the relationship between the task those systems perform and their design, and the tradeoffs that made their systems work.

TENTATIVE examples (not in order)

The virtual telerobot, control and the integration of many interface concepts (Jacoby)

The virtual windtunnel, using computationally intensive virtual visualization tools (Levit)

A general system for virtual environment research (Robinett)

Virtual terrain environments and large database handling (Hitchner)

Virtual environments involving networked systems (Bricken)

Developing interesting and fun environments for the commercial world (Blanchard)