IPT & EGVE Workshop (2005)
R. Blach, E. Kjems (Editors)

Colosseum3D – Authoring framework for Virtual Environments

Anders Backman

VRlab/HPC2N, Umeå University, Sweden

Abstract

This paper describes an authoring environment for real time 3D environments, Colosseum3D. The framework makes it possible to easily create rich virtual environments with rigid-body dynamics, advanced rendering using OpenGL Shaders, 3D sound and human avatars. The creative process of building complex simulators is supported by allowing several authoring paths such as a low level C++ API, an expressive high level file format and a scripting layer.

Categories and Subject Descriptors: I.3.4 [Graphics Utilities]: Software support

1. Introduction

Authoring a rich multimodal real time 3D environment can be a daunting task. Virtual Reality (VR) is a category of 3D graphics applications, where the user is immersed into the virtual works using display systems such as head mounted displays (HMD), CAVE [CSD93] or other projection systems. A 3D simulator must perform many complex tasks, such as parsing 3D geometry and drawing it efficiently, simulating dynamic objects, handling collision detection, reading sound files and playing them accordingly to the simulation, handling of system events etc.

This work concentrates on the authoring process of virtual environments with both physical as well as visual content. The framework described, Colosseum3D, is a modular system written in C++ for building large scale simulation environments, and is based on existing open source software. Also most of the modules that is a part of the framework is written as self contained libraries and can be downloaded and used in other projects independent of our framework.

Our framework has proven to be intuitive when building real-time applications. For instance it has been used by more than 200 students at Umeå University creating simulations with real-time rigid-body dynamics.

2. Architecture

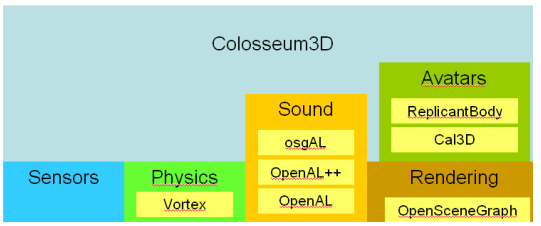

Figure 1: Components of Colosseum3D

2.1 Rendering

OpenSceneGraph (OSG) [Osg05] is used as the rendering toolkit. OSG is a versatile open source scenegraph written in C++ and is portable between most platforms.

2.2 Physics

Vortex from CM-Labs is a fully featured rigid-body dynamics simulation toolkit with a C API. As we have implemented a C++ layer for the dynamic simulation, we have the possibility of switching dynamics toolkit if we decide to. We are also working on the integration of Open Dynamics Engine (ODE) [Ode05] which would make all the ingoing parts of the framework open source based.

2.3 Sound

Currently we are using OpenAL [Ope05] to handle 3D sound in the framework. OpenAL is a portable API for spatial sound with a programming interface similar to OpenGL. On top of OpenAL we have developed OpenAL++ [Häm02] which is an object oriented abstraction layer written in C++. To integrate OpenAL++ into the scenegraph we have written yet another abstraction layer, osgAL [OAL05].

2.4 Avatars

Cal3D [Cal05] is a skeleton based 3D character animation library written in C++. Our framework extends the toolkit with features such as queuing and blending of animations and ground following in a module named ReplicantBody [Sun02]

2.5 Scripting

For scripting of actions and events and even creation of objects we are using Lua [Lua05]. Lua is well suited for embedding into real time applications such as games and simulators due to its efficiency and small memory footprint. To expose all C++ classes to Lua we are using Tolua++ [Tol05] which is a small utility library written in Lua to parse C++ headers and create wrapper code that enables the use of C++ code from within Lua. The wrapper libraries can then be requested at runtime from Lua. This results in a smaller memory footprint as only the required libraries will be loaded during runtime.

2.6 Sensors

Sensors is a module that handles external VR devices such as 6DOF tracker systems and gloves.

3. Authoring

In Colosseum3D a simulation is created by declaring a *state*. A state can be seen as a snapshot of the simulation at time t_0. When the simulation starts, successive states are calculated by the simulation engine. Each state describes objects with their visual and their physical attributes at a given time. The visual attribute can be as simple as referencing an existing 3D model in a file format supported by the rendering tool. The physical attribute describes the attributes necessary to simulate the object in a rigid body simulation. This includes mass, material, geometry, etc. An object does not have to have both visual and physical attributes. A pure physical object can be created simply by omitting the visual attributes. Part of the physical description of the system is also joints. These places a constraint on any attached object. A joint can for example describe a hinge, where an object is free to rotate around a given axis. The simulation state can be entered into an extendible descriptive file format.

3.1 OpenGL shader development

OpenGL Shading Language (GLSlang) is a high level language used to partly replace the fixed pipeline of OpenGL. It can be used to access vertex and fragment primitives efficiently on the graphics processor unit (GPU). As OpenSceneGraph supports GLSlang, we have exposed all of the features of GLSlang to the Lua scripting environment. Also by adding the required declarations to the descriptive file format, shaders can be declared and used as visual attribute for any object.

References

[Cal05] CAL3D, http//cal3d.sourceforge.net, 05/05/18

[CSD93] CRUZ-NIERA C., SANDIN D., DEFANTI T., Surround-screen Projection-based Virtual Reality: The Design and Implementation of the CAVE. *In Computer Graphics, Proceedings of SIGGRAPH 93*, August 1993.

[Häm02] HÄMÄLÄ T., OpenAL++ - An Object oriented toolkit for real-time spatial sound, *Master's thesis UMNAD 391*, Department of Computing Science, Umeå University, 2002.

[Lua05] LUA, http://www.lua.org, 05/07/18

[Ode05] OPEN DYNAMICS ENGINE ODE, http://opende.sourceforge.net/, 05/07/18

[Ope05] OPENAL, http://www.openal.org, 05/07/18

[Osg05] OPENSCENEGRAPH, http://openscenegraph.sourceforge.net, 05/07/18

[OAL05] osgAL http://www.sourceforge.net/projects/osgal, 05/07/18

[Sun02] SUNNA P., "Real-time Character Animation", *Master´s thesis UMNAD 397*, Department of Computing Science, Umeå University, 2002

[Tol05] TOLUA++, http://www.codenix.com/~tolua/, 05/07/18