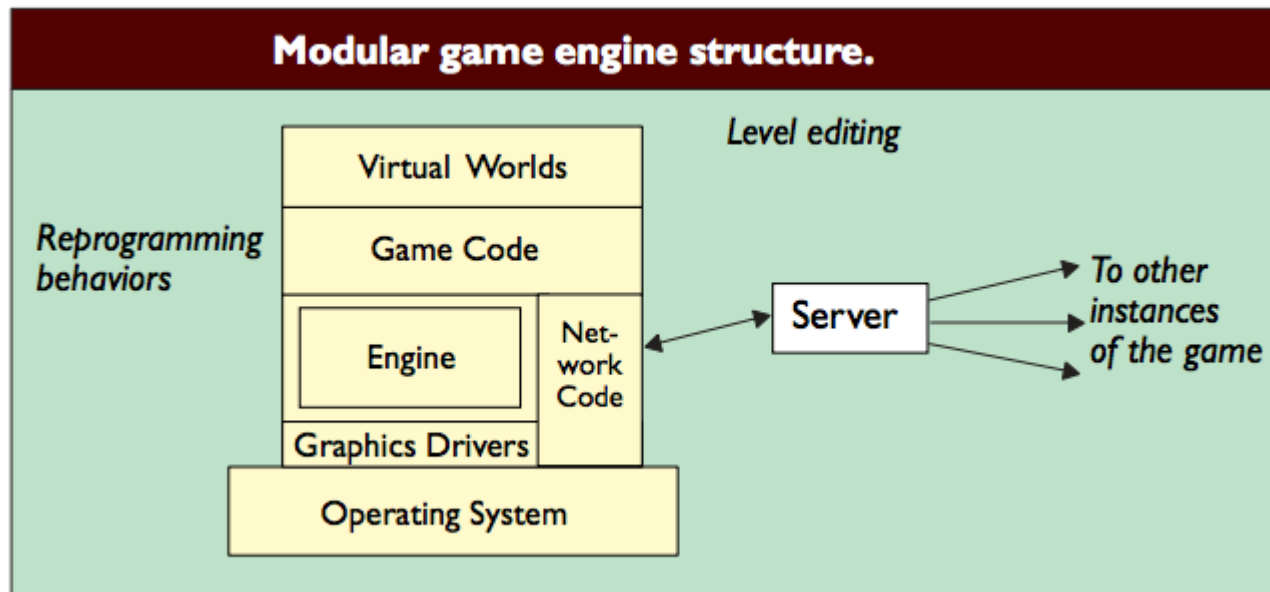


# **Game Engines Overview**

**Ernesto Araiza & Saber Adavi**  
**Natural User Interaction in Cars**

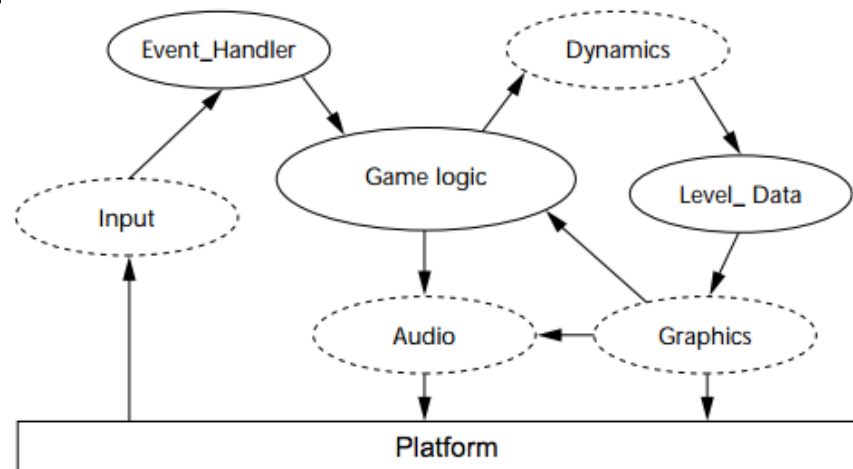
# Definition

A game engine is a system designed for the creation and development of video games. "...that collection of modules of simulation code that do not directly specify the game's behavior."



# Some functions of a Game Engine

- Efficient display
- Culling
- Dynamic collision detection
- Drawing exteriors and terrain
- Interact with external d
- etc...



# Write our own Game Engine using an existing Rendering Engine?

Disadvantages of using an existing one:

- Learning curve of how it works
- Learning a new programming language?
- Less room for customization?

# Write our own Game Engine using an existing Rendering Engine?

Probably not, there many of them.

Advantages of using an existing one:

- Modularity
- Time saving
- Getting support from community

# What are we looking into a Game Engine for our project?

- Simple connectivity to 3rd party devices such as Kinect
- Find one that has been used for car games?
- Choose one that uses the programming language that we are more experienced
- Find one with a strong community of developers
- Robust, mature, maybe commercial games have used it

# Game Engines (or middleware) most used in the academic world

- UDK ( Unreal Engine 2,3)
- Panda 3D
- CryEngine
- Quake (id Tech)
- Source Engine (Half-life)
- Ogre3D
- Delta 3D
- OpenSceneGraph
- Unity

# UDK



License: Free/EULA. (Not open source).

Programming language: UnrealScript (Bytecode-compiled scripting language similar to Java), DLL.

Rendering: DX and OpenGL

Third Party Devices Support: Good



# Panda3D

From Disney Company



Features:

- particle systems,
- physics and collisions,
- soft- and hard-skin character animation,
- 2D and 3D nonuniform rational B-spline support,
- 2D graphical user interface support,
- multipass rendering, and
- a real-time shader framework.

# Panda3D

- Panda employs a scene graph
- Written in C++
- developers can use Python's Tkinter/Pmw wrappers around the Tcl/Tk API to generate graphical user interfaces
- To integrate with the variety of input devices that virtual reality applications require, Panda supports the Virtual Reality Peripheral Network API ([www.cs.unc.edu/research/vrpn](http://www.cs.unc.edu/research/vrpn)).
- It can import models and animations from common programs such as Maya, MultiGen, 3Dstudio Max, Softimage, and Lightwav
- Panda benefits from having a thriving open source community for documentation, testing, and development of new features.
- Panda uses the Python scripting language ([www.python.org](http://www.python.org)) as an interface to C++ libraries

[More features...](#)

# CryEngine



- Free non-Commercial use / EULA
- But by the information they put in their website, if you want to use it for Serious Games/Simulation, you may need to contact them in order to have access for some features (and yes! it's not free)
- It is well known for it's excellent visual performance

# CryEngine

- Python Scripting Supported
- It is possible to parse data between APIs and Devices using XML
- Supports Kinect
  - [video](#)

# VRPN : Virtual Reality Peripheral Network

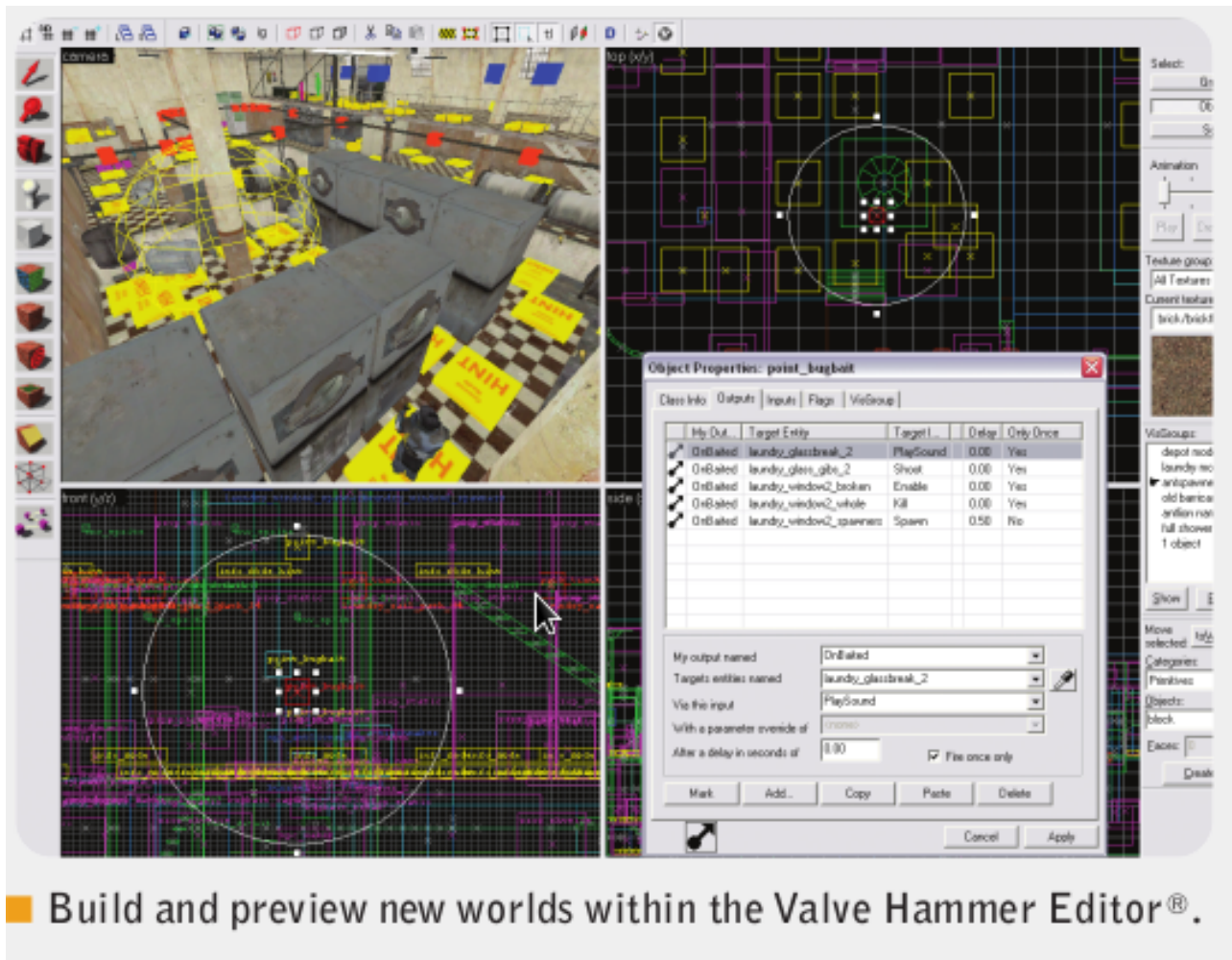
- designed to implement a network-transparent interface between application programs and the set of physical devices
- It supports a long list of devices, included:
  - [Viewpoint](#) Eye tracker, Kinect ...
- Cross Platform
  - [Video](#)

# Source Engine



- All code written in C/C++ using Visual Studio 7.1. Easily and quickly derive new entities from existing base classes.
- Internal context sensitive performance monitoring system
- Graphics performance measurement tools built into the engine
- Dx9 shaders all written in High Level Shading Language (HLSL)
- For full features you need to pay
- Supports Kinect through openNI
  - [Video](#)

# Source Engine



# OGRE 3D



- Open Source
- Cross Platform
- You can use kinect with openNI library
  - [Video](#)
- Builds on Visual C++ and Code::Blocks on Windows



# Delta 3D



- Used in training, education, visualization, and entertainment.
- It's open source
- They put OpenSceneGraph and CAL3D together
- It has a high-level, cross-platform (Win32 and Linux) C++ API designed
- It supports Kinect through its linux drivers
- Integrates with: Open Scene Graph, Open Dynamics Engine, Character Animation Library, and OpenAL.

# id Tech (Quake)



- Newer versions (4,5) lack good online community
- id Tech 5 is open source
- Good Visual Effects (Rage)
- Really hard to master
- Information about the game engine features could not be easily found on the web

# Torque 3D Open Source 3.0

- MIT Licensed Open Source ( written C++)
- TorqueScript is a fast, easy to use C++ like scripting language that ties the various elements of a project together
- TorqueScript features:
  - Object-oriented programming
  - Transparent interconnection with external C++ objects
- Built-in fast 3D math (vectors, matrices, and quaternions with all corresponding functions)

# Torque 3D Open Source 3.0

- Torque is built with an abstracted C++ Physics API with the ability to be integrated with major 3rd party physics libraries.
- Torque also has a C++ Sound API

# Game Engines In Research

- We reviewed 37 Papers and Dissertation that used a game engine for their research, or discussed game engines.

# Game Engines In Research

Reviewed Game Engine:

- UDK (Unreal Engine 2, 3, 3.5)
- Id Tech ( Quake, 3, 4)
- Source Engine (Half life)
- Ogre
- Panda 3D
- Delta 3D
- OpenSceneGraph
- Jupiter EX
- Torque Engine

# Game Engines In Research

- Papers before 2009, mostly didn't use UDK because of license.
- in November 2009, Epic released a free version of UE3's [SDK](#), called the Unreal Development Kit (UDK)
- In the papers after this time UDK and Unity3D gained a lot of attention,

# Game Engines In Research

- UDK free versions offer all the functionalities of Game Engine
- In some researches they used Open Source engines to access the codes



# FROM ABSTRACT DATA MAPPING TO 3D PHOTOREALISM: UNDERSTANDING EMERGING INTERSECTIONS IN VISUALISATION PRACTICES AND TECHNIQUES

## Game engines vs VR engines

- Game engines have more visual effects and an easier workflow than the VR-engines
- Game engines costs are generally much lower than VR-engines.
- The quality and usability of game engines exceeds that of the best Virtual Reality tools.
- The rapid development of game engines exceeds the virtual reality development.
- With game optimized multiplayer code it is now possible to create a presentation, and host a server online. Therefore, many people can then log on and experience not only the e.g. architectural presentation of a new building, but also the interaction of lots of people within this world.
- As the game engines and their tools mature, it is becoming easier and easier to make presentations.

# Developing a Low-Cost Driving Simulator for the Evaluation of In-Vehicle Technologies

- After evaluating several open-source and commercial alternatives, the commercial driving game rFactor [8] was chosen as the software platform for our driving simulator.
- It offers a convincing, realistic driving experience thanks to richly detailed graphics, accurate vehicle physics, and full support of force feedback steering wheels.
- And while it does not offer the complete flexibility of an open-source product, the game does allow for a deep degree of modification and customization.
- They used an eye tracker in their research

## **Chronic Pain Rehabilitation with a Serious Game using Multimodal Input**

- They connected Unity 3D to kinect using TCP sockets and XML message.

## **Development and Evaluation of Low Cost Game-Based Balance Rehabilitation Tool Using the Microsoft Kinect Sensor**

- They used kinect with Unity 3D

# An Evaluation of Visual Gesture Based Controls for Exploring Three Dimensional Environments (2012)

Unreal Engine, CryEngine, unity and Ogre  
He wanted to use kinect for gesture recognition

Ogre Provides:

- Rendering Engine
- scene graph engine

Lacks:

- Physics engine for collision detection
- You can add them with additional libraries

He used Xinput library for interacting with kinect.

# Human - Virtual Agent Interaction

He picked UDK

He used FUBI and C++ for gesture recognition.

# FUBI : Full Body Interaction Framework

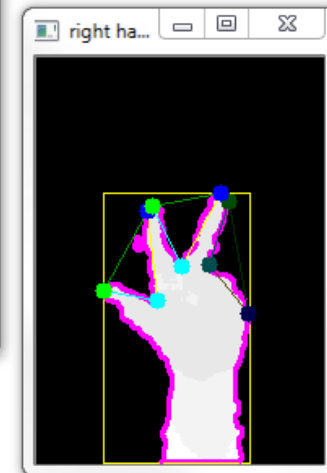
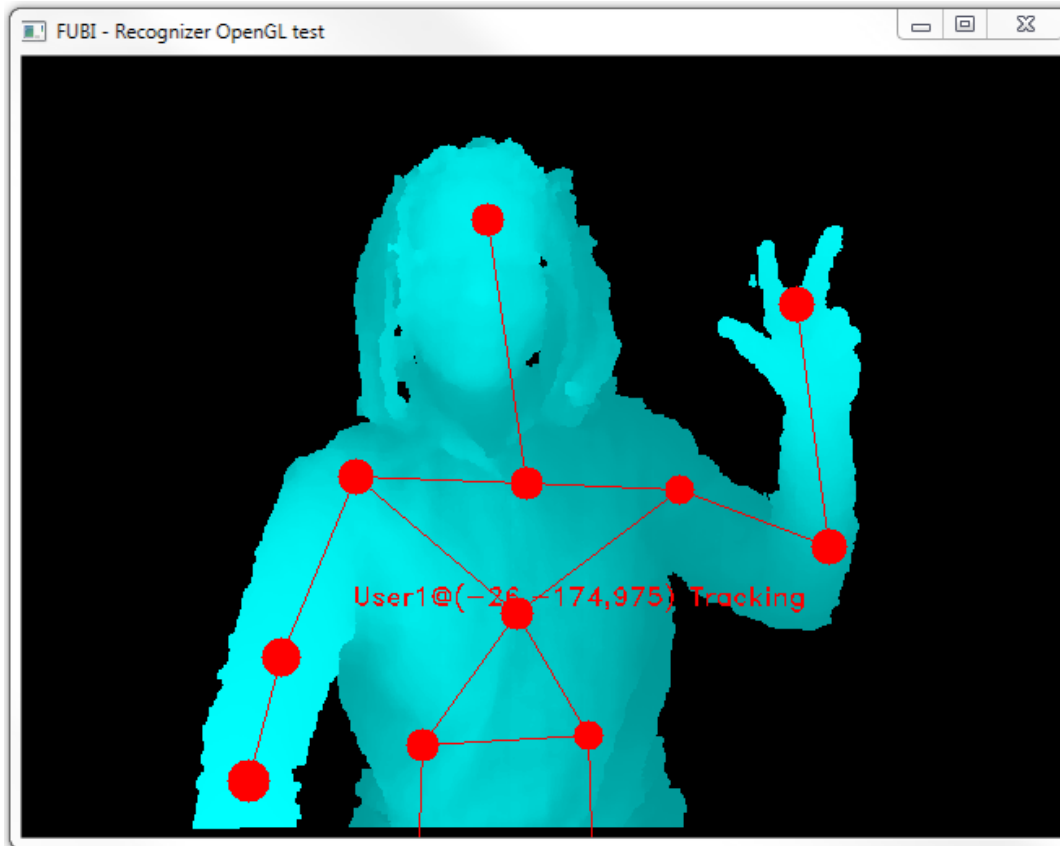
- Full Body Interaction Framework (FUBI) is a framework for recognizing full body gestures and postures in real time from the data of a depth sensor integrated using OpenNI or the Kinect SDK.



# FUBI : Full Body Interaction Framework

1. **Static postures:** Configuration of several joints, no movement (e.g. figure 1: "arms crossed").
2. **Gestures with linear movement:** Linear movement of several joints with specific direction and speed (e.g. figure 2: "right hand moves right").
3. **Combination of postures and linear movement:** Combination of sets of 1 and 2 with specific time constraints (e.g. figure 3: "waving right hand").
4. **Complex gestures:** Detailed observation of one (or more) joints over a certain amount of time and recognition of specific patterns/paths (e.g. symbolic gestures like hand writing shapes).

# FUBI : Full Body Interaction Framework





# FUBI : Full Body Interaction Framework

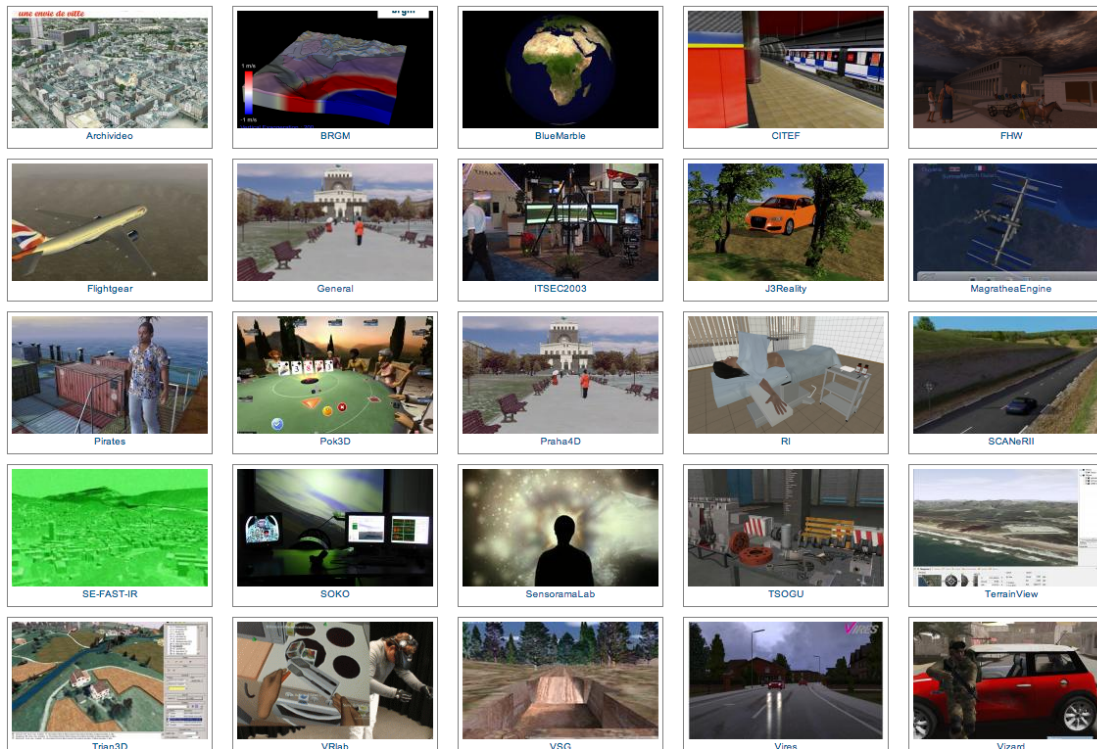
- FUBI is written in C++ and currently offers a C++-API.
- It has been only tested on Windows 7, but at least the C++-part should contain no platform dependent code.
- FUBI is freely available under the terms of the Eclipse Public License - v 1.0.
- [Video](#)

# Rapid prototyping a virtual fire drill environment using computer game technology

- The Source Engine was chosen for the fire drill simulator as it provides everything required to produce a high quality virtual environment and has extensive support included with the Source software developers kit (SDK).
- The technical features of the Source Engine were also more than adequate for the requirements of the problem domain, e.g. support for particle effects such as fire and smoke and additional properties that are particularly relevant to a fire drill simulator including allowing fire entities to ignite other fire entities, giving the impression of a spreading fire, which will help to increase the realism of the environment

# OpenSceneGraph

Open source high performance 3D graphics toolkit



# OpenSceneGraph Features

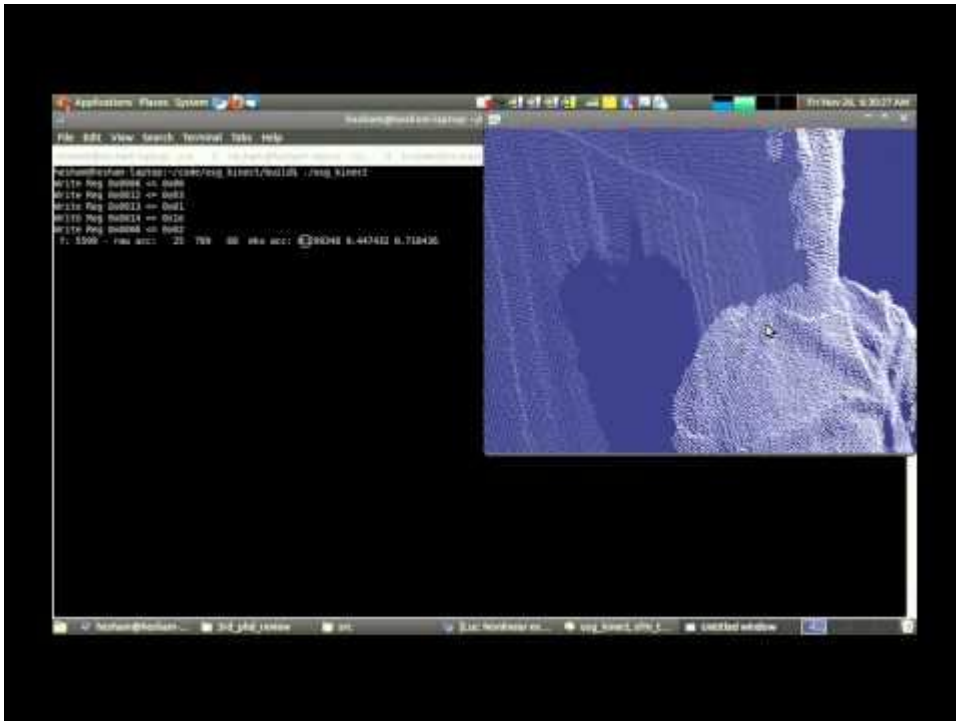
- The OpenSceneGraph is open source, real-time graphics middle-ware used by application developers in fields that range from visual simulation (flight, marine, vehicle, space simulator) to virtual and augmented reality, to medical and scientific visualisation, to education and games.
- Cross-platform, it uses OpenSceneGraph Public License, which is a relaxation of the Less GNU Public License (LGPL)

# OpenSceneGraph Features

- Standard C++, but with Java, Lua and Python bindings for the OpenSceneGraph are available as Community projects.
- Uses database loaders and different node kits to manage particles, terrain rendering, character animation, etc.
- Portability: requires OpenGL and C++

[List of features...](#)

# OpenSceneGraph Examples with Kinect



<http://www.youtube.com/watch?v=11Dp7Zilvwl>