3D Rendering Engine for Simulation

MTP First Stage

by Lt Col Chetan Dewan

Roll No 06305403

under guidance of

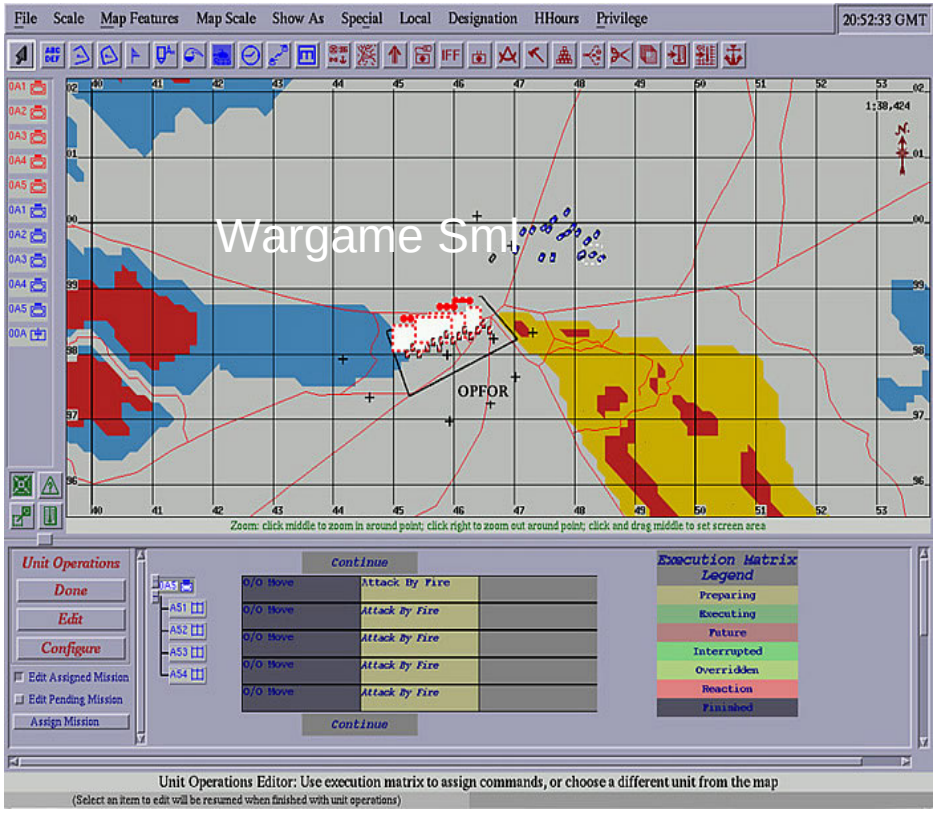
Prof Sharat Chandran

Motivation

Customize any open source 3D rendering engine for developing Military Simulations.

Aim

Creation of gaming toolkit for near real-time visualisation of Entity Simulators and Wargames.



Overview

- Brief: Wargaming and simulation
- Problems with Commercial systems
- Design Philosophy—Our Credo
- Base Rendering Engine
- Advanced Features (Wish list)
- Delta 3D
- Work Plan

Wargaming and Entity Sml

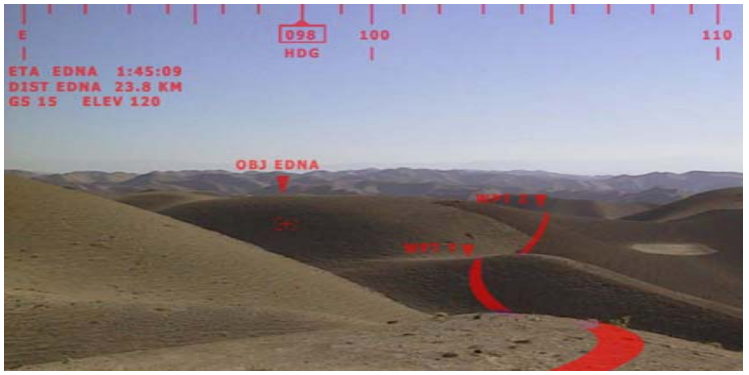
- Historical war gaming and sml used by Germans in WW-II.
  - Manual games
  - Predict outcome of war
  - Tactical planning
  - Low cost training

- US Army's (Advanced Research Projects Agency (ARPA)).
  - Project for creating networked simulations
  - War-games and entity simulators could interact with each other in a "virtual world."

Entity Sml

- Computer systems which model operator control.
- Graphical displays on a monitor or large screens.
- Use full-scale models of the system of concern.
- Operator inputs in real time.
- Input devices are mock-up controls which are true replicas of the original controls.
- DOF Robots used for motion.

War Games Sml



- Computer programs model the interaction between forces such as army divisions. (Groups)
- Forces modeled are controlled by operators representing force command staffs
- Operators follow orders given by actual commanders at various different levels.
- Played over a wide area network.

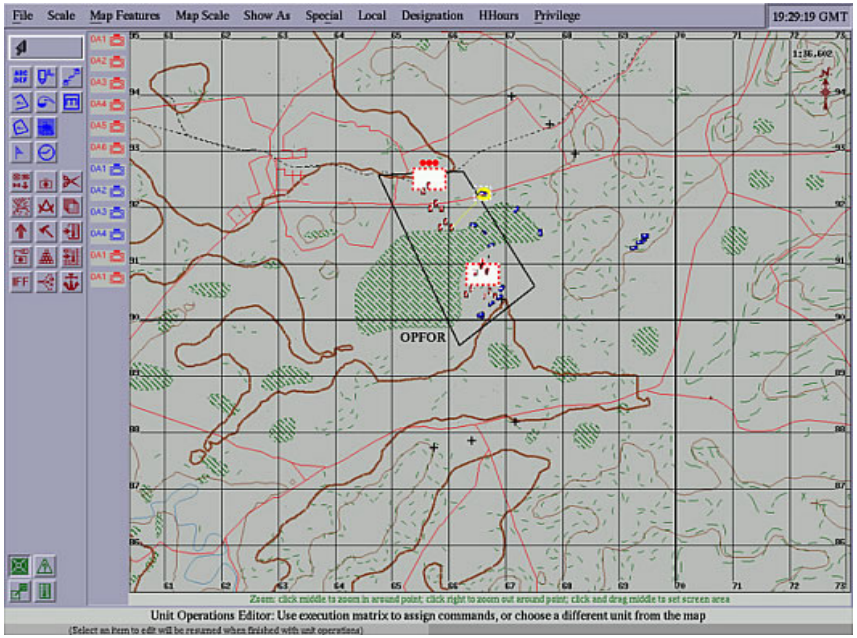
Contd…

War Games Sml

- Control instructions, which may or may not be in real time, are input by key board commands.

- Results of interaction between entities are determined by programmed probabilistic relationships.

Well known US war-game ModSAF (Modular Semi-Automated Forces)



DIS

- Both types of simulations model a virtual world in which the forces or entities operate.
- Distributed Interactive simulations use a network to bring entities represented by different computers into the same virtual world.
- DARPA project attempt to merge the entity and war-game simulators in the same virtual space.

The Combined Arms Tactical Trainer (CATT) facility is one such simulator which is a networked suite of simulators.

Collective trainers

These don’t fall under the category of war-games but are a no of entity simulators networked together in a virtual world to give collective training to a bunch of soldiers at once.

- limited tactical training ability
- LIC training
- CI operations
- Guerrilla warfare

Problems with commercial Simulation Systems

- Simulation systems priced with costs running into six figures for each application built.
- Projects locked into the proprietary technologies.
- No way to modify the underlying engine.
- Developers have to build “work-arounds” to overcome problems with proprietary tools.
- For off the shelf systems there are never any free upgrades.

Design philosophy

- Keep everything open to avoid vendor lock-ins and increase flexibility.
- Use a modular design so that anything can be swapped out as technologies mature at different rates and anything can be added.
- Make it as generic as possible since it is not known what application it's going to have to support next.
- Open source community as indirect developers.
- Reusability of existing code and models

Our Approach

- The basic idea is to get an engine at minimal cost yet be very good.
- Developing our own ruled out.
- Modifying an existing source code also ruled out.
- Use open source projects as Building Blocks to get desired functionality and quality.
- Windows DLL approach.

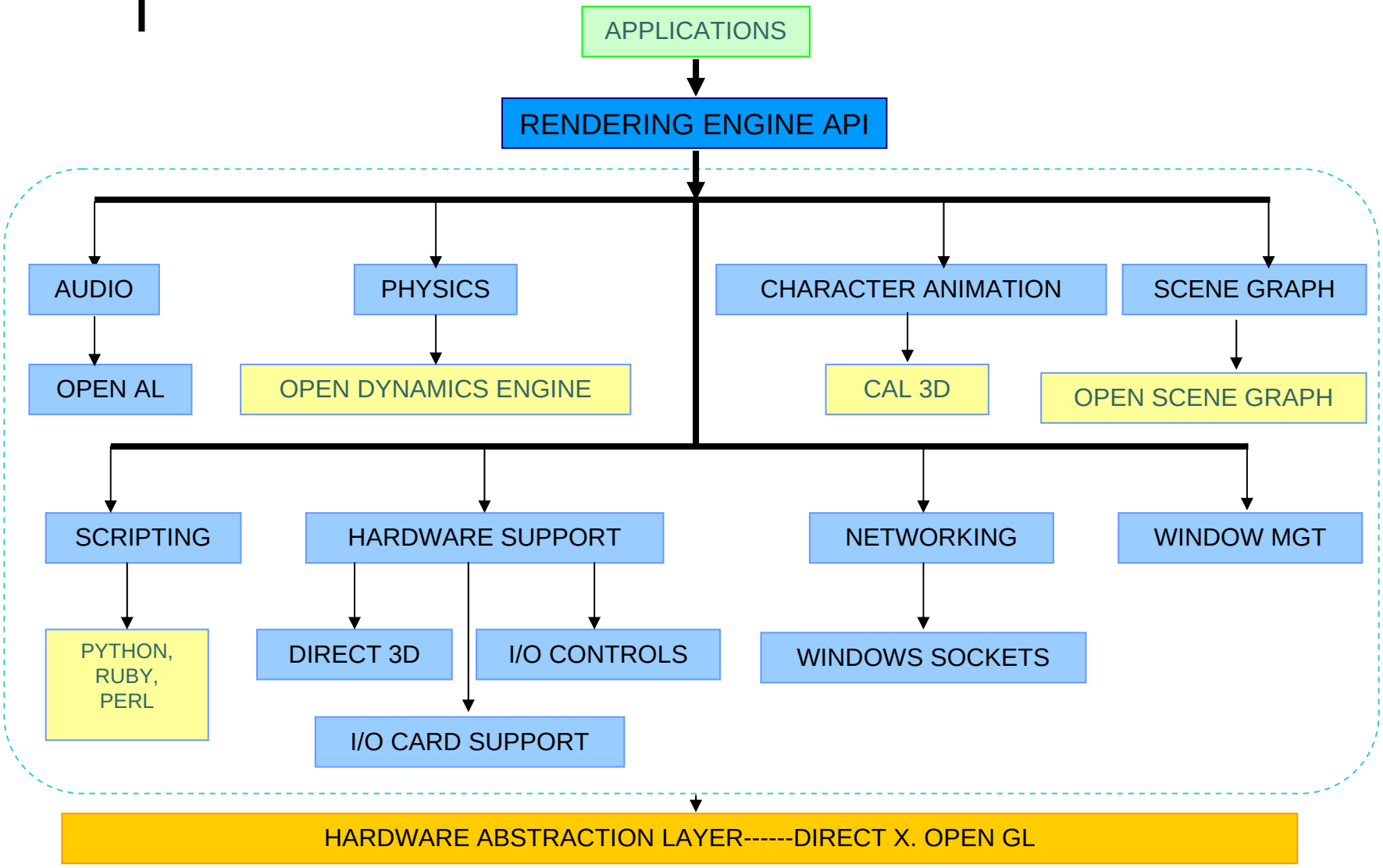
Benefits

- Low Cost development
- Less time Consuming
- Comparatively little code needs to be written.
- Our engine improves as the Open Source project is improved.

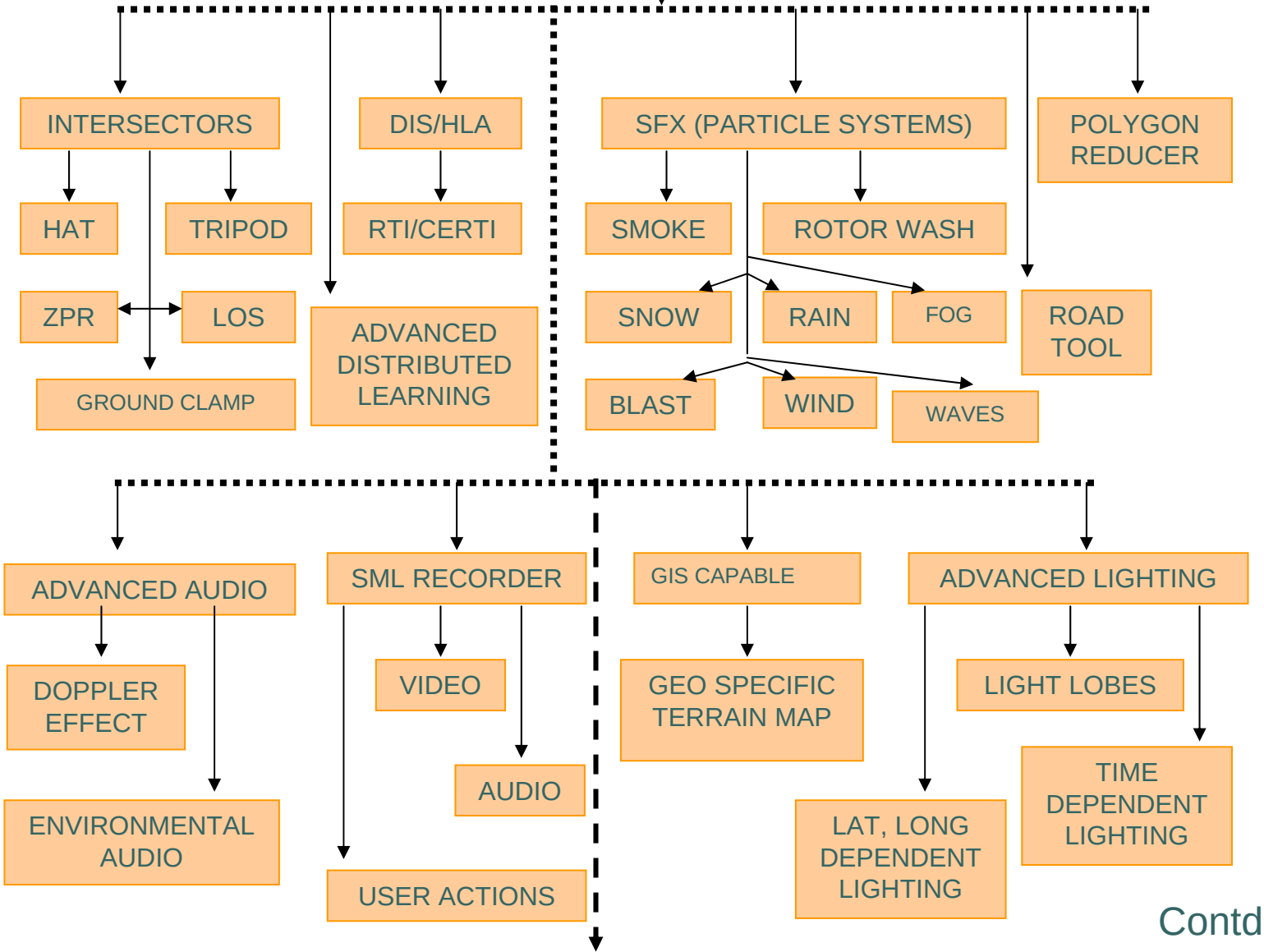
Challenges to our approach

- Visualising base engine architecture.
- Selecting the Base rendering engine from the hundreds on the net.
- Determining exactly what extra features to add to the engine and How?
- Creating and implementing the add-on architecture.
- Identify existing open source projects for our add-on's

Base Engine Design

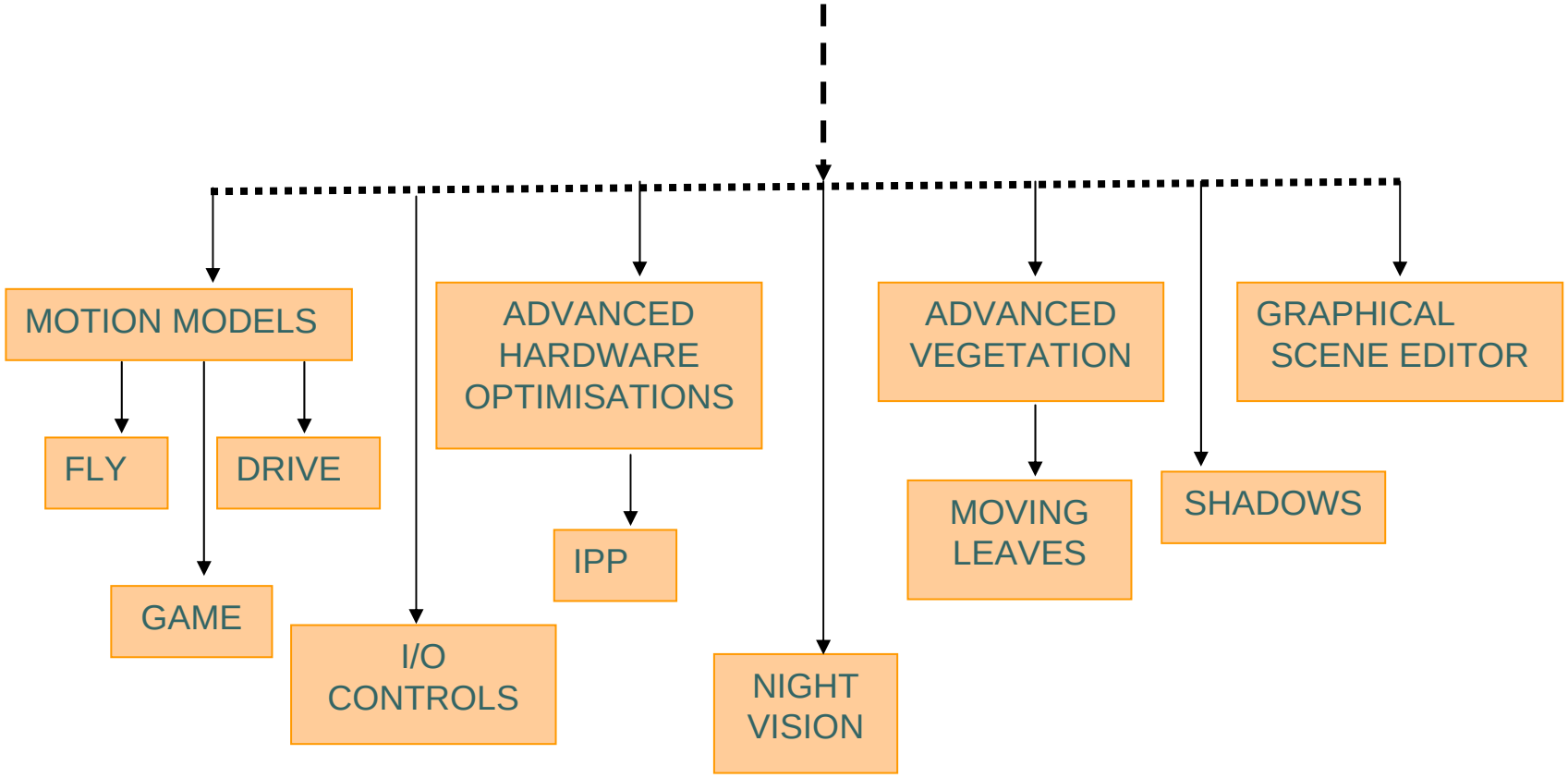


Add-on enhancements



Contd…

Add-on enhancements



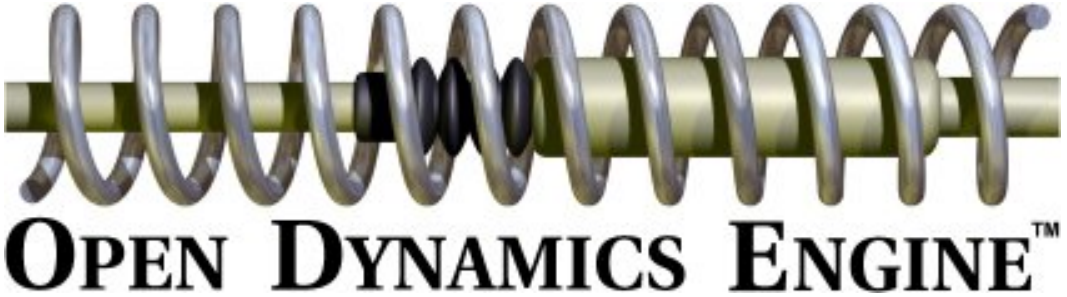
Selecting the Base Engine

- Support for popular 3D Visual database formats particularly the *OpenFlight* format.
- Basic engine components such as:-
  - Audio support.
  - Basic Physics support.
  - Ease of manipulation of 3D objects.
  - Character animation.
- Popularity with gaming community.
- Visual studio.net support. (C++)

Our choice:

- Delta3D uses OpenSceneGraph (OSG) for rendering.
- High performance 3D graphics toolkit.
- Used in visual simulation, games, virtual reality, scientific visualization and modeling.
- Written in Standard C++ .
- OpenGL as its underlying rendering API.

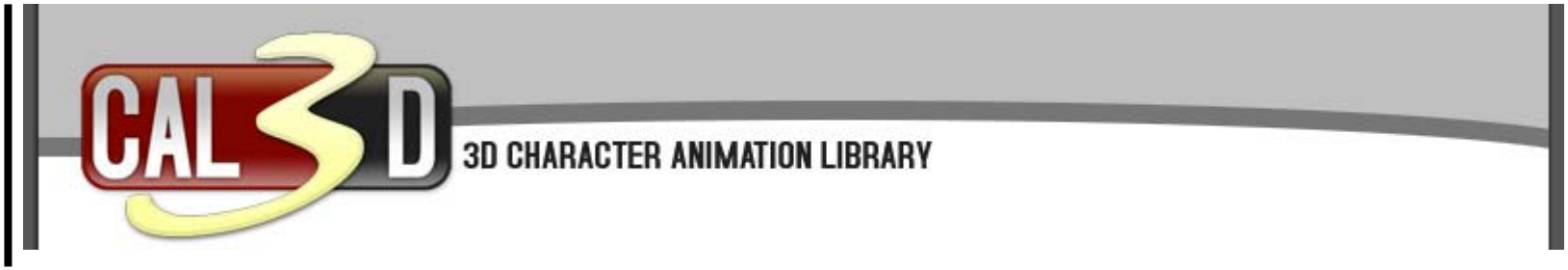
Physics



- Physics performed by the Open Dynamics Engine (ODE) library.
- ODE is a high performance library for simulating rigid body dynamics.
- C/C++ API.
- Used in several computer games, 3D authoring tools and simulation tools.
- Can realistically model several devices/physical phenomena, such as joints, springs, damping devices (e.g., shock absorbers), friction, gears, motors, and collisions.
- Useful for simulating vehicles, objects in virtual reality environments and virtual creatures

Audio

- Open Audio Library (OpenAL) is a software interface to the audio hardware.
- Resembles the OpenGL API in coding style and conventions
- Produces high-quality audio output, specifically multichannel output of 3D arrangements of sound sources around a listener.
- Includes extensions to handle sound-source directivity and distance-related attenuation.
- Effects --- Doppler, environmental effects such as reflection, obstruction, transmission, reverberation etc.



- Cal3D animates characters.
- Cal3D is a skeletal based 3D character animation library
- Written in C++.
- Exporters for most popular (both open source and proprietary) 3D database formats.
- The Cal3D C++ library loads exported files, build characters and run animations.
- Cal3D allows character animations, such as walking and running etc.

Scripting

- Scripting language to allow advanced behaviors to be added to a game with a minimum of C++ programming on the developers' part.
- Delta3D uses the Python.

GUI

- FLTK is a cross-platform C++ GUI toolkit for UNIX®/Linux® (X11), Microsoft® Windows®, and MacOS® X.
- FLTK provides modern GUI functionality without the bloat and supports 3D graphics via OpenGL® and its built-in GLUT emulation.

Ce GUI

- Crazy Eddie's GUI System is a free library providing windowing and widgets for graphics APIs / engines where such functionality is not natively available, or severely lacking.

Open producer

- Open Producer is a cross-platform C++/OpenGL library that is focused on Camera control.
- Producer's Camera provides projection, field of view, viewpoint control, and frame control.
- Producer can be used in a multi-tasking environment to allow multiple Camera's
- Support for multiple display subsystems.

XML

- Xerces-C++ is a validating XML parser written in a portable subset of C++.
- Xerces-C++ makes it easy to give your application the ability to read and write XML data.

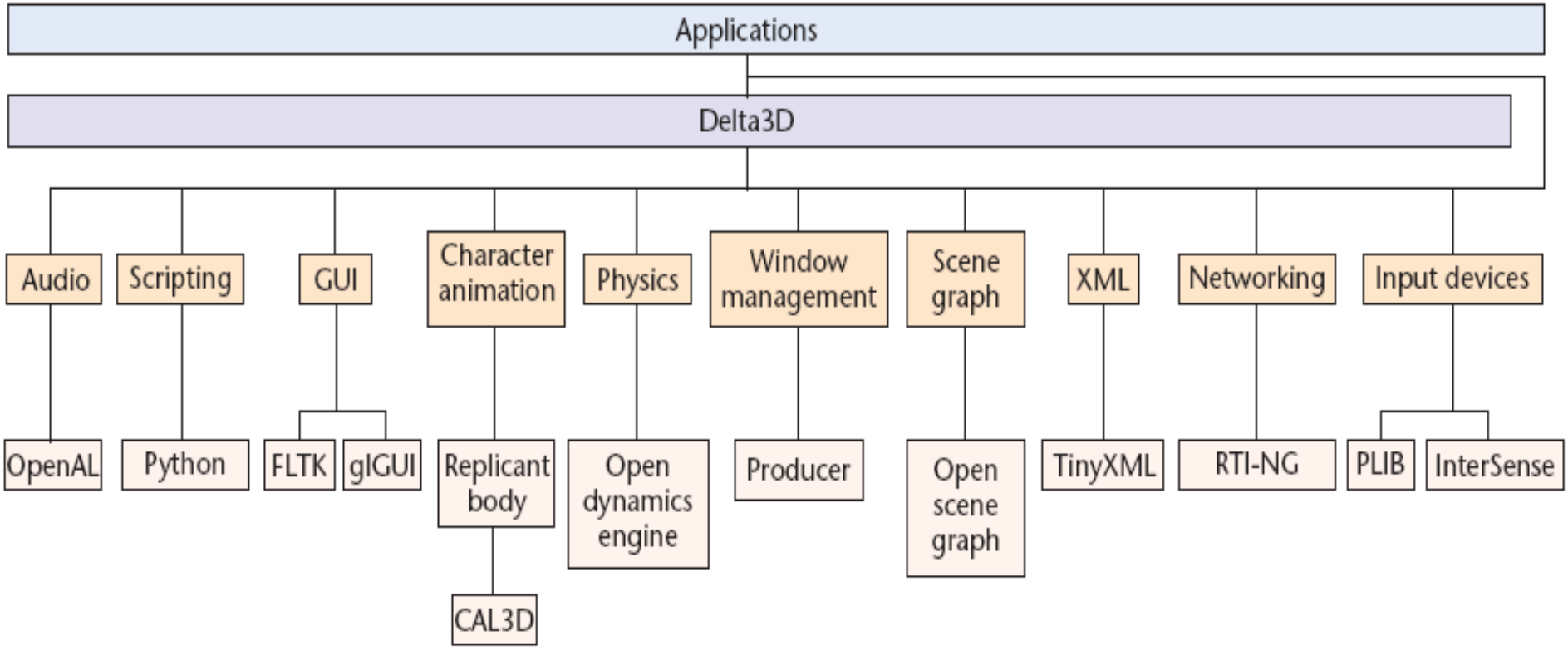
I/O

STEVE'S PORTABLE GAME LIBRARY.

- **PLIB: A Suite of Portable Game Libraries.**

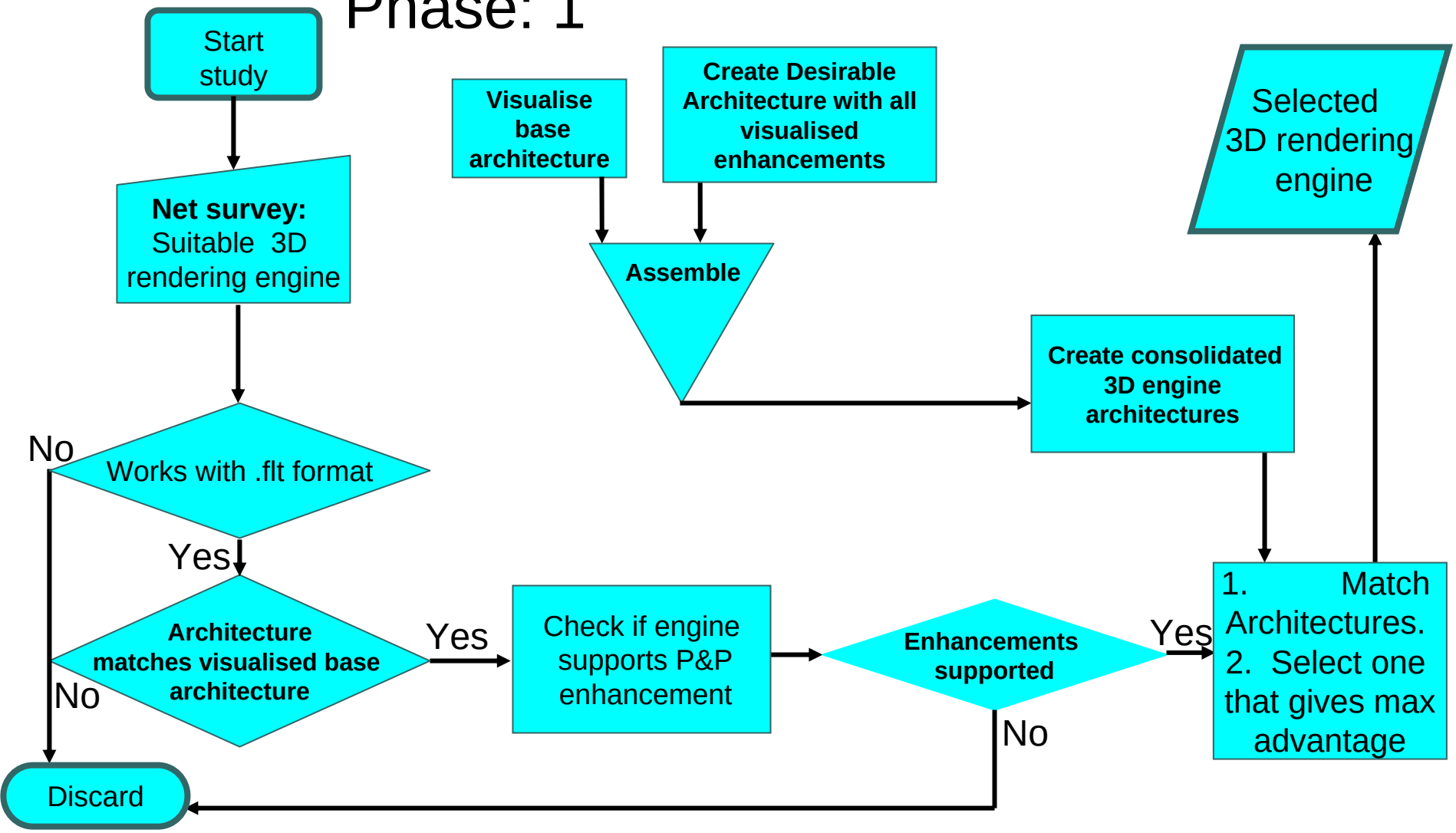
- **Intersense provides a tracking library which can be used for i/o of generic trackers.**

Block schematic Delta 3D



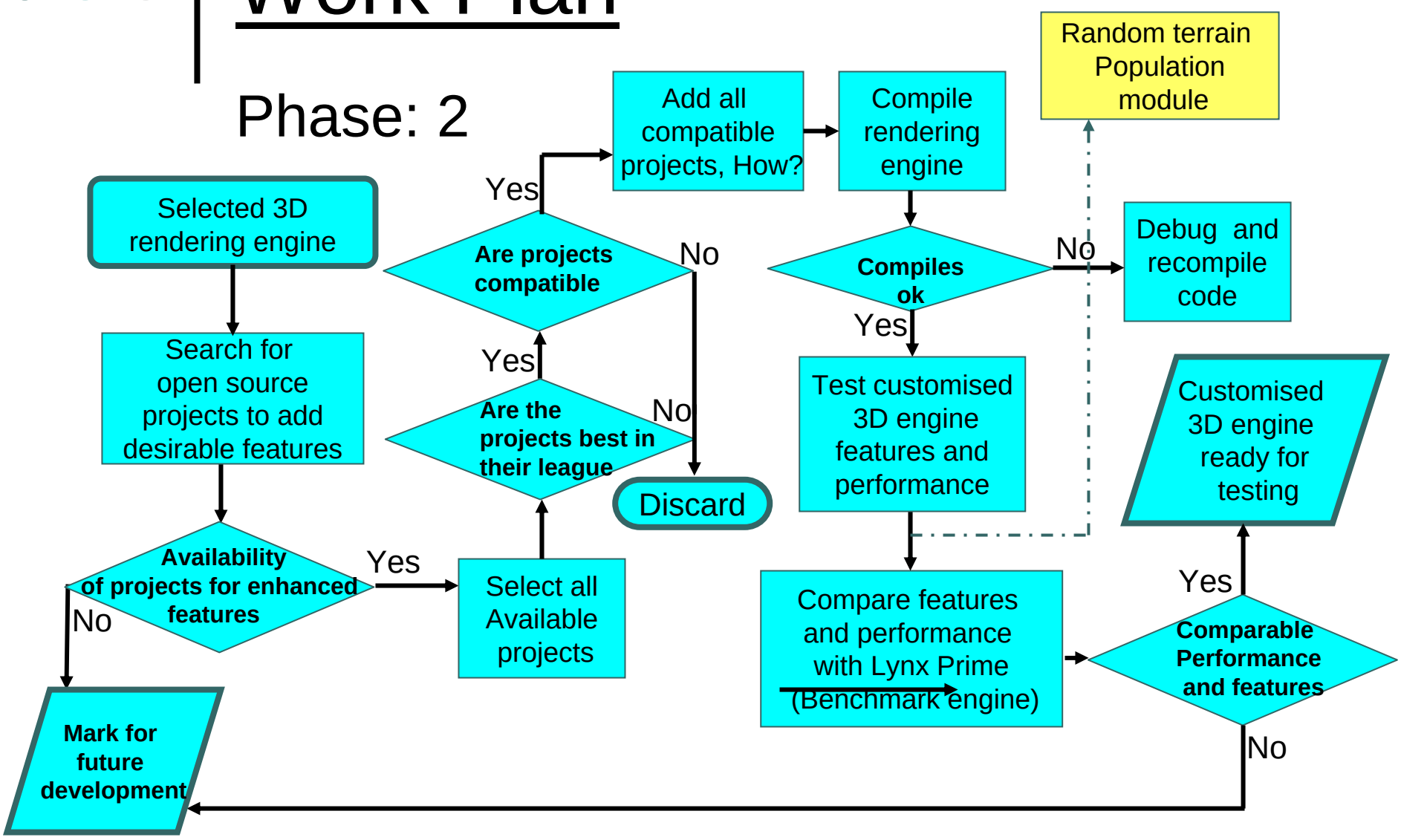
Work Plan

Phase: 1



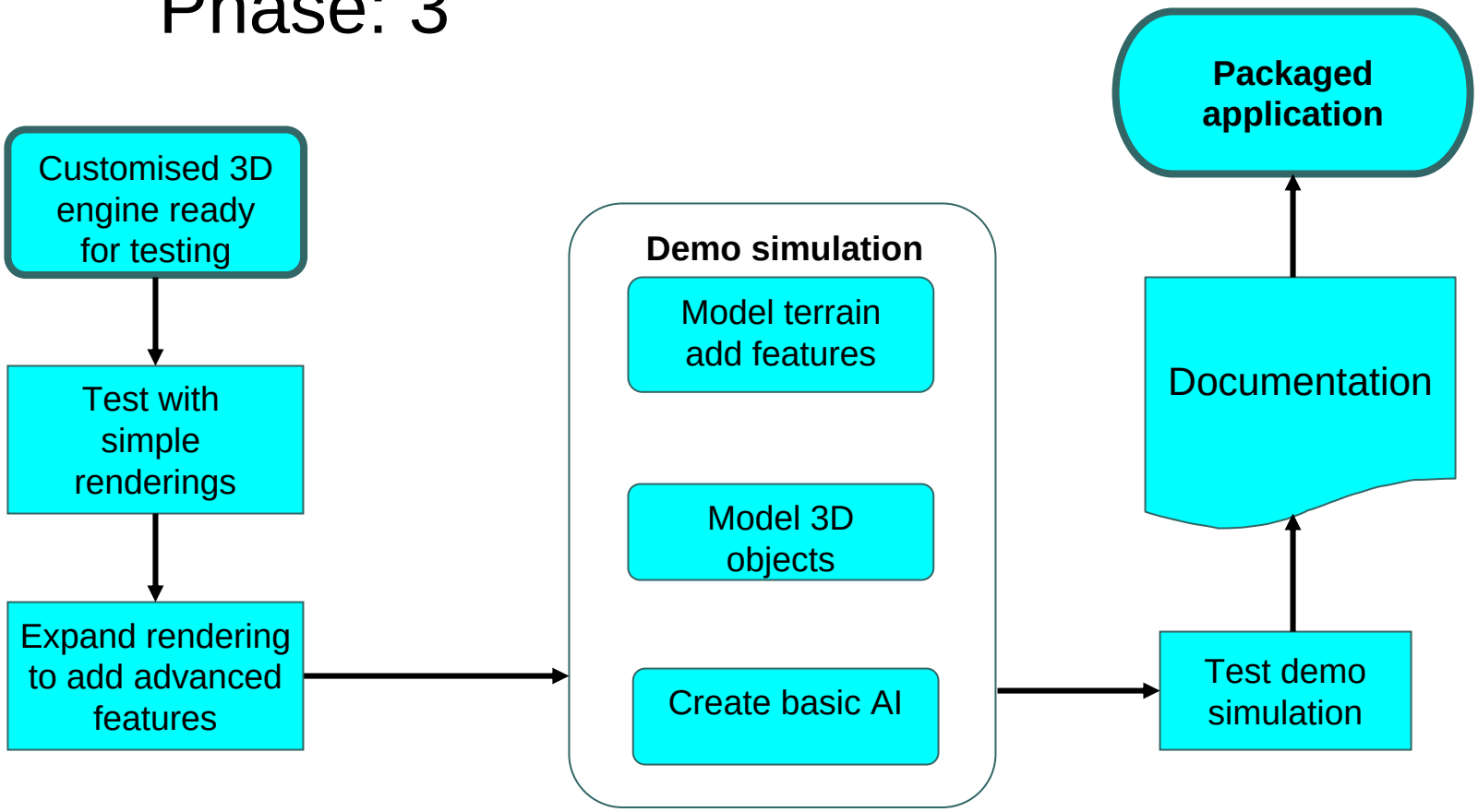
Work Plan

Phase: 2



Work Plan

Phase: 3



References

[1] James F Dunnigan. The Complete Wargames Handbook, Chapter 1, Retrieved June 2007.
URL http://www.hyw.com/Books/WargamesHandbook/Contents.htm

[2] Steven M. Alexander, David O. Ross and William J. Lloyd. Developing Intelligently Interactive Computerized Wargames. Pentagon reports, Final report Feb 2000-01

[3] Delta 3D features, Retrieved Mar 2007.
URL http://www.delta3d.org/article.php?story=20051209133127695&topic=docs

[4] Perry McDowell , Rudolph Darken, Joe Sullivan, Erik Johnson; Delta3D: A Complete Open Source Game and Simulation Engine for Building Military Training Systems.

[5] Multigen Corp. OpenFlight Visual database format, Retrieved June 2007.
URL http://www.multigen.com/products/standards/openflight/index.shtml

[6] Soren Hannibal. 3d engines for games, a broader perspective, Retrieved June 2007.
URL http://www.gamasutra.com/features/20001013/hannibal_pfv.htm.

[7] OpenSceneGraph Home Page, Retrieved
May 17, 2007 from URL http://www.openscenegraph.com/index.php

[8] Open Dynamics Engine Home Page, Retrieved May 20, 2007 URL http://ode.org/

[9] Open Audio Library Specifications Page,
Retrieved May 20, 2007 URL http://www.openal.org/oalspecs-specs/x44.html.

[10] Character Animation Library 3D FAQ Page, Retrieved May 20, 2007
URL http://download.gna.org/cal3d/documentation/api/html/

[11] Lawrence Rosenblum and Michael Macedonia Projects in VR. The Delta3D Open Source Game Engine, Volume 25, Issue 3, May-June 2005 Page(s): 10 – 12, Retrieved May 2007.

Additional web links

- Comprehensive graphics toolkit URL http://www.quantum3d.com/products/software/archive-do%20not%20post/vtree.html
- Intel performance primitives (IPP); Retrieved May, 2007 URL http://www.intel.com/cd/software/products/asmo-na/eng/302910.htm
- Delta 3D ; Retrieved May, 2007 URL http://www.movesinstitute.org
- Wargames ; Retrieved June, 2007 URL http://www.strategypage.com/military_photos/200522323.aspx
- Defence Advanced Research Projects Agency ; Retrieved June, 2007 URL http://en.wikipedia.org/wiki/DARPA.
- Aviation Reconfigurable Manned Simulator (AVCATT); Retrieved June, 2007 URL http://www.peostri.army.mil/PRODUCTS/AVCATT/.
- Game; Retrieved May, 2007 URL http://en.wikipedia.org/wiki/Half-Life_2
- game ; Retrieved May, 2007 URL http://www.quake4game.com/

Thank you

?????