Maverick
Game Engine Comparison Document

Table of Contents

1. Introduction

The following document outlines various game engines' technical specifications that will be taken into consideration to help determine which game engine to use for our FPS project.

The document has three main sections:

- Overview of Game Engines
  - This section gives an overview of the five game engines that were researched.
- Features
  - This section lists the general features of the five game engines.
- Requirements
  - This section lists game engine's requirements or features that are needed for this project.
  - Each 3D engine requirement is ranked from 1 to 3. A '3' means that the specification must exist, whereas a '1' indicates that if it is not available, the specification can still be fulfilled in another manner. If the specification does exist it will receive 1 point otherwise 0. Then each value (0 or 1) is multiplied with the corresponding rank and all the points are added up for the various 3D engines that were researched. The engine that receives the most points will be considered for purchase. If multiple engines are within a few points of one another, then we will inspect each engine individually and come up with a final decision.

2. Overview of Game Engines

2.1. 3DGameStudio

3DGameStudio is an authoring tool use to make 2D and 3D real-time applications. It combines the C-Script programming language with a high-end 3D engine, a 2D engine, a physics engine, a level, terrain and model editor, plus extended libraries of 3D objects, artwork and ready-assembled games. Simple games can be created with no programming knowledge by using pre-made scripts. For more advanced or commercial games, user can program using C-script or modify the source code directly. User can import 3D models from cheap tools such as Milkshape or high-end tools such as 3d Studio Max. There are four different editions that can be purchase. Each edition has different features and different prices. Features and prices can be found on their website, http://www.conitec.net/a4info.htm. There is also a team edition that can be use by ten different users as long it is on the same project. In the team edition, the game can be distributed freely as long as you pay the extra commercial license. They do have company support by using email. I email the company five times since March 30, 2005 with questions about their product, and finally got a response on June 10, 2005, so the company support might not be very good. They do have good community support. Most questions on their forums were answer in a decent amount of

time. Overall 3DGameStudio has a lot of features and seems very easy to use. The only major concern is maybe the poor company support.

2.2. Crystal Space

Crystal Space is described as a portable 3D Game Development Kit written in C++. I have to agree with many of the comments that I have read about CS being a feature rich engine and meet many of our requirements, but I am leery of using this engine for our project for the fact that it is still under development and seems to have steep learning curve. There is a Crystal Entity Layer or C.E.L. that is being designed for use with CS to help automate the process of making games, but it is still in beta development. Since CS is still under development and has a high learning curve I cannot recommend this engine for use with our game.

2.3. Ogre3D

Ogre3D is an object-oriented open source 3D graphics engine. The engine does not include typical built-in game engine components such as physics, handling of user input and AI. Instead the engine just focuses on 3D graphics and manipulation of a 3D scene. The engine is abstracted well enough so that a programmer does not specifically code for any particular 3D standard format, such as OpenGL or DirectX. From using a few of the tutorials the engine is logically laid out and easy to use. Ogre3D has great community support, which increases the amount of help that we would be able to receive. To use this engine on our project, we would need to integrate the physics library that is available, find a sound library and implement all the fps game code. Ogre3D would be a great engine choice because it's ease of use and features, but due the lack of time, I do not think that we can implement all the things we need to use this engine.

2.4. Reality Engine

The Reality Engine offers many of the tools that a 3D engine can offer to meet the demand of today's game developers. However, there are some disadvantages as I will discuss below that may stop us from purchasing this product. If Reality Engine is purchased, the license would grant full C++ engine source, example C++ and C# source to the game and tech demos, including demonstration media and scripts, and source to all the editing and compiling tools and utilities. Reality Engine can be fully integrated with .NET scripting languages, allowing programmers and artists to write fully debuggable, IDE-integrated code, from C#, C++/CLI, to VB.NET without the need for a compiler Reality Engine has fully integrated support for physics-based vehicles, including player control, AI, and networking. Reality Engine has an advanced high-performance physics engine. The AI has intelligent pathfinding and decision-making. After doing some more research on this engine and visiting a couple of forums to see what current users had to say about the system, I found that majority of the users gave this a product a 4 or 5 out of 5 (5 meaning excellent). However, I was not able to find much about the ease of use of this product. After reading the forums it came to my attention that some of the current users of Reality Engine are game developers in the real world. Reality Engine website does not publicize the cost of this product, so I thought I would give them a call to find out more details about their product and pricing.

Having spoken to Reality Engine representative on the phone I was asked what the product would be used for and I had described it to him as much in detail as I could. However, he would not tell me the price of the product and suggested that the Reality Engine would be too much and that I wouldn't find much support for it. In overall, this seems like a product for serious game developers. Although it does have the features we need for our project, it might be out of the question due to the fact that we do not have the money to purchase it.

2.5. Torque

The Torque Game Engine (TGE), created by Garage Games, is a professional, cross platform 3D Engine that is available to the general public for $100 per license. There is also an educational license that could potentially be even cheaper. Its primary use was for a first person shooter that came out a few years back called Tribes 2. With the purchased license, the user gets full access to the C++ source code to modify as they see fit, and if one decided to create a game using it, you pretty much have no restrictions if you wanted to sell it. The engine also contains a powerful scripting language that allows one to modify most game logic without rebuilding a new .EXE. The TGE also comes with other tools that are mainly used to convert various external assets to one that can be used from within the engine. The code for these tools is included also. I have emailed one of the representatives at the company, and received a reasonably fast response (within the same day). He seemed interested in working with our team/school. There is also an active community of independent developers that frequently post tips, resources, and code samples on the Garage Games site.

3. Features

	Game Engines				
Features	**3DGameStudio**	**Crystal Space**	**Ogre3D**	**Reality Engine**	**Torque**
Rendering	• Material properties for static and dynamic objects. • 2D renderer for still images, 2D sprites, panels, buttons, sliders, overlays, and movies. • 3D views and movies can be rendered to curved surfaces. • Programmable 2D and 3D effects like lens flares, bullet holes, cartoon rendering.	• Multi-texturing and Mipmapping • Supports GIF, TGA, PNG, BMP, JPG • Perspective correct texture mapping with interpolation every 16 pixels • Transparent and semi-transparent textures • Vertex and Pixel Shaders • CLOD for terrain • Landscape engine supports scattering and lod • 2D sprites and a particle system for terrain • Currently CS fonts and true type fonts are supported • GUI system • Per-vertex lighting and Light mapping • Special effects include Environment Mapping, Lens Flares, Billboarding, Particle System, Sky, and Mirror	• Direct3D • OpenGL • Material declaration language • Shader support • Complete range of fixed function operations • Multiple pass effects • Multiple material techniques • Material LOD support • PNG, JPEG, TGA, BMP, DDS, 1D, volumetric and DXT/S3TC texture support • Projective texturing support • Multiple shadow rendering techniques • Particle System • Support for skyboxes, skyplanes and skydomes • Billboarding	• Per-Pixel Lighting and shading • Shading Types: • Dynamic projection mapping, • Normal mapping, • Phong specularity, • Per-pixel reflection mapping, • Refractions, • Virtual displacement (parallax) mapping, • Animated textures, • Mix/detail shaders, • Fabric, • Anisotropic scattering, • Water, • Other configurable pixel & vertex shaders. • Dynamic world lighting and shadowing	• Terrain rendering for exteriors and BSP style renderer for interiors • Multi-pass texturing • Volumetric fog • Decals • Environment mapping • Light mapping • Particle engine • water engine • No shader support by default

Licensing	• Team/Instructor edition plus a single commercial edition license: $199 + $199 = $398. • Can be install on ten different machines. • All members of the team must use the software for the same project or for learning purposes. • The game can be publish or sell royalty free. • Has a 'watermark' or required 'made by A6' screen at startup for all develop games.	• LGPL license • Modifications must be released back to the community	• Modifications must be released back to the community • Ogre3D source must be passed on with all copyrights intact • Customizations must be clearly documented • Full source is included	• License includes full source to editor and full rights to distribute editor with your game to allow mod-makers and enthusiasts to breathe new life into your title.	• $100 per user (user is defined as anyone who views C++ source code) • 50% off for educational license • No royalties for most projects
Scripting	• Has built in script to have full control of game. • C-script is a simplified version of C++. Also very similar to JavaScript. • New users have pre-made script available. • Expert users can interface to the engine using C/C++ or Delphi.	• Many languages can be used for scripting if bindings are created with Swig. So far Python has the most bindings.		• .NET scripting languages, C#, C++/CLI, and VB.NET • Python	• Torquescript: C++ style runtime scripting language
Built-in Editors	• Level • Model • Script editors			• Reality Editor • World • GUI Designer	• GUI • World • Mission

Physics	• Polygon level collision detection. • Physics engine supports gravity, damping, elasticity, friction, and hinge, ball, wheel, and slider joints	• Basic Physics, Collision Detection, Rigid Body • Hierarchical bounding box collision detection system • physics library using ODE		• Supports multi-primitive, arbitrary joint linkages, stacking, breakage, and particle physics. • Integrated physics editing inside of Reality Builder, supporting creation of optimized collision primitives for models and skeletal animated meshes; constraint editing; and interactive physics simulation and tweaking in-editor.	• Bounding box collision detection, collision with world and other objects, vehicle physics

Scene Management	• BSP, Portals, Occlusion Culling, PVS, LOD: • Seamless indoor and outdoor support	• Visibility system based on a combination of portals, kd-tree, and coverage buffer • XML world file format allowing you to easily redefine the world • Levels can be stored in standard compressed ZIP archives so that you can easily make a bundle of one level • A converter to convert MAP files (from Quake/Half-Life) to CS is also included. Also, Blender scripts (Python) are included to export models and levels from within Blender • Sequence manager with triggers allowing the definition of object interactions from within the level itself	• Highly customizable, flexible and not tied any single scene type • Hierarchical scene graph • Querying features	• Full artist-driven and procedural Level-Of-Detail support. • Per-pixel Occlusion Culling. • Reload Scripts on the fly to see your changes without having to restart the application	• BSP/portals for interior objects • LOD for terrain, spatial database

Animation	• Keyframe Animation, • Skeletal Animation • Morphing • Animation Blending	• Keyframe Animation and Skeletal Animation • 3D animated skeletal meshes using Cal3D animation	• Skeletal animation	• Character Normal Mapping, Spherical Harmonics, Rag Doll Physics, skeleton-based multi-weighted. • Seamless transitioning between physics & keyframe animation. • Physics-based character bone influences also allow for procedural animation. • Ragdoll character animation, allowing you to mix physics with animations for dynamic effects such as character damage • Shader animation	• Blended animations • Skeletal-based

Mesh	• Mesh Loading, Skinning, Deformation • Exporters supplied for Milkshape, Worldcraft, gamespace, 3dStudio, Maya	• 3D triangle mesh sprites with frame animation. • Converters for Milkshape, Maya, Cal3d, 3DS, Quake MDL and Quake II MD2 formats to Crystal Space are included. • Importers for 3DS, MDL, MD2, OBJ, POV, and ASE are also included. The meshes are actually progressive meshes allowing for dynamic LOD (level of detail) changes. • There is also support for skeletal sprites	• Flexible formats accepted • Biquadric Bezier patches for curved surfaces • Progressive meshes • Static geometry batcher	• Character Normal Mapping & Spherical Harmonics, with skeleton-based, multi-weighted-bone vertex shader animation • Characters can contain any number of arbitrary pixel & vertex shaders on multiple materials • Precomputed Radiance Transfer data & lightmaps automatically mapped to Level-Of-Detail meshes	• Exporters supplied for Milkshape, 3dStudio, Maya.
Sound	• 3D sound sources (wav, ogg, CD, mp3) with Doppler effect	• 2D and 3D sound. • DS3D, EAX, A3D • WAV, Ogg/Vorbis, AU, AIFF, IFF, and MOD		• Environmental Audio with EAX and 5.1 Surround. • Ogg Vorbis sound streams	• OpenAL 3D sound support; loads .ogg and .wav files
AI	• Pathfinding, Decision Making • Finite State Machines • Also has pre-made AI script			• Intelligent pathfinding, both predetermined and dynamic with obstacle avoidance. • Decision-making based on adaptive state machines. • Reactions to stimuli such as sight and sound.	• Little official support, more features available with community resources

Input Handling	• Mouse and keyboard support • Has a pre-made player-input script that can be modified.			• Mouse and keyboard support	• Mouse and keyboard support
Tools	• Supports external 3D modelers mentioned earlier in 'Mesh' section • Terragen		• Export from many modeling tools including Milkshape3D, Maya and Blender	.NET support Python support Max Maya	• Supports external 3D modelers mentioned earlier in 'Mesh' section • Quark • Hammer
Platforms	• Windows	• Windows • Mac OS • Linux	• Windows • Mac OS • Linux	• Windows • Xbox	• Windows • Mac OS • Linux

Misc	• Slow motion/quick motion effect. • Save / Load system for resuming games at arbitrary positions. • Multi-player client/server mode for LAN and Internet (TCP/IP, UDP).		• Common resource infrastructure for memory management and loading from archives • Flexible plug-in architecture allows engine to be extended without recompilation • "Controllers" allow easy organization of derived values between objects • Debugging memory manager • Examples of combing Ogre with other libraries • XMLConverter	• Plug-ins for 3D Studio Max and Maya. • Includes Voice Communication • Optimized Client/Server-Authoritative networking	• Extensive networking • Large community • Extensive documentation

4. Requirements

	Game Engines				
Requirements (rank)	**3DGameStudio**	**Crystal Space**	**Ogre3D**	**Reality Engine**	**Torque**
Graphics					
2d overlays (3)	✓	✓	✓	✓	✓
Indoor multi-floored textured environments (3)	✓	✓	✓	✓	✓
Animation support (3)	✓	✓	✓	✓	✓
Outdoor environments (1)	✓	✓	✓	✓	✓
3d mesh support (3)	✓	✓	✓	✓	✓
Physics					
Collision detection (3)	✓	✓	×	✓	✓
AI					
Pathfinding (3)	✓	×	×	✓	×

Decision making (2)	✓	×	×	✓	✓
Input/Output					
Handle user input (3)	✓	✓	×	✓	✓
Sound support(3)	✓	✓	×	✓	✓
External Tools					
Mesh exporters for cheap programs (3)	✓	✓	✓	✓	✓
Importing levels from cheap editors (3)	✓	✓	✓	✓	✓
OS Support					
Windows (3)	✓	✓	✓	✓	✓
Mac (1)	×	✓	✓	×	✓
Linux (1)	×	✓	✓	×	✓
Programming					
Source included (3)	✓	✓	✓	✓	✓
Scripting (2)	✓	✓	×	✓	✓
Engine source in c/c++ (2)	✓	✓	✓	✓	✓
Object handling/manipulation (3)	✓	✓	✓	✓	✓
Other					
License within budget (3)	✓	✓	✓	×	✓
License supports unlimited distribution (3)	✓	✓	✓	×	✓
Comprehensive documentation (3)	✓	✓	✓	✓	✓
Free company support (2)	✓	✓	✓	✓	✓
Community support (2)	✓	✓	✓	×	✓
Ease of use (3)	✓	×	✓	×	✓
Built-in editors (2)	✓	×	×	✓	✓
Fps framework (2)	✓	×	×	✓	✓
TOTAL (68)	66	56	49	53	65

5. References

http://www.conitec.net/a4info.htm
http://www.crystalspace3d.org/tikiwiki/tiki-view_articles.php
http://www.ogre3d.org/
http://www.artificialstudios.com/
http://www.garagegames.com/pg/product/view.php?id=1