Copy No.________

Defence Research and Development Canada
Recherche et développement pour la défense Canada

DEFENCE R&D DÉFENSE

Assessment of 1st Person Shooter Game Engines for Force Protection Applications

J. Frim and M. H. Thomson
Humansystems Incorporated

Humansystems Incorporated
111 Farquhar St., 2nd floor
Guelph, Ontario N1H 3N4

Project Manager: Mark Hazen 902 426-3100 x176

Contract Number: W7707-4-2677/A

Contract Scientific Authority: Allan Gillis 902 426-3100 x203

The scientific or technical validity of this Contract Report is entirely the responsibility of the contractor and the contents do not necessarily have the approval or endorsement of Defence R&D Canada.

__Terms of Release:__ The information contained herein is proprietary to Her Majesty and is provided to the recipient on the understanding that it will be used for information and evaluation purposes only. Any commercial use including use for manufacture is prohibited. Release to third parties of this publication or information contained herein is prohibited without the prior written consent of Defence R&D Canada.

Defence R&D Canada – Atlantic

Contract Report
DRDC Atlantic CR 2005-088
January 2006

Canada

This page intentionally left blank.

Assessment of 1st Person Shooter Game Engines for Force Protection Applications

J. Frim and M. H. Thomson
Humansystems Incorporated

Prepared by:

HumanSystems Incorporated
111 Farquhar St., 2nd Floor
Guelph, ON N1H 2N4

Project Manager: M. G. Hazen, 902 426-3100 x176
Contract number: W7707-4-2677/A
Contract Scientific Authority: Allan Gillis, 902 426-3100 x203

The scientific or technical validity of this Contract Report is entirely the responsibility of the contractor and the contents do not necessarily have the approval or endorsement of Defence R&D Canada.

Terms of release: The information contained herein is proprietary to Her Majesty and is provided to the recipient on the understanding that it will be used for information and evaluation purposes only. Any commercial use including use for manufacture is prohibited. Release to third parties of this publication or information contained herein is prohibited without the prior written consent of Defence R&D Canada.

Defence R&D Canada – Atlantic

Contract Report
DRDC Atlantic CR 2005-088
January 2006

Principal Author

Original signed by J. Frim

J. Frim

Approved by

Allan Gillis

Scientific Authority

Approved for release by

Kirk Foster

DRP Chair

Work conducted under ARP 11BK

Abstract

First Person shooter games allow humans to be immersed in a desktop virtual world. There are a variety of commercial off-the-shelf (COTS) multi-player game engines available that provide the infrastructure for building these worlds. Five game engines are assessed against 22 criteria to determine the their usefulness in providing a virtual world for naval Force Protection scenarios. The aim of this work is to use the resulting virtual world for concept-development and experimentation for Force Protection operations.

Résumé

Les jeux de tir permettent aux humains de s’immerger dans un monde virtuel avec un ordinateur de bureau. Toute une gamme de moteurs de jeux à joueurs multiples disponibles sur le marché (COTS) offrent une infrastructure pour créer ces mondes. Cinq moteurs de jeu ont été comparés selon 22 critères d’évaluation afin de juger de la faisabilité de les utiliser pour élaborer un monde virtuel pour les scénarios de protection de force navale. L’objectif de cette recherche est d’utiliser le monde virtuel qui a été élaboré pour expérimenter et développer un concept d’opérations pour la protection de la Force.

This page intentionally left blank.

Executive summary

Assessment of 1st Person Shooter Game Engines for Force Protection Applications

Frim, J. and Thomson, M. H.; DRDC Atlantic CR 2005-088; Defence R&D Canada – Atlantic; January 2006.

Introduction

First person shooter (FPS) games provide immersive environments that have become increasingly realistic. Recent applications of these games to military training and tactical development (e.g., America's Army) have shown that they can be used productively for more than just entertainment. A naval operation that has many characteristics in common with the FPS genre is the case of in-harbour force protection. There are however, few naval FPS games in existence.

While there are many FPS games (the majority of which are multi-player) they are built upon a much smaller number of game engines. These game engines provide the structure and game development infrastructure for building a wide variety of games.

Results

Five multiplayer FPS game engines are compared using 22 evaluation criteria to assess the feasibility of using them to build a multiplayer game which incorporates a Canadian Patrol Frigate (CPF). The intent was to build a game that would allow red and blue forces to attack/defend the CPF in order to conduct force protection concept development and experimentation. The complete evaluation is given in the report.

Significance

It was determined that there were at least two game engines (Unreal and Cry) that would do the job. The Unreal engine came out on top due mostly to its larger established user base.

Future plans

A 3D model of a CPF is being developed and imported into a FPS game that uses the Unreal engine and modified to provide an example immersive environment. This "game" will then be used to conduct force protection command and control experimentation.

Sommaire

Assessment of 1st Person Shooter Game Engines for Force Protection Applications

Frim, J. and Thomson, M. H.; DRDC Atlantic CR 2005-088; R & D pour la défense Canada – Atlantique; January 2006.

Introduction ou contexte

Les jeux de tir offrent des environnements immersifs de plus en plus réalistes. Les applications récentes de ces jeux dans les domaines de l'instruction militaire et du développement tactique (par exemple America's Army) ont démontré qu'ils peuvent être utilisés de façon productive pour plus que le simple divertissement. Plusieurs jeux de tir partagent en effet de nombreuses caractéristiques avec certaines opérations navales, notamment celles qui mettent en cause les unités de protection de la Force des ports. Mais il existe présentement peu de jeux de tir de type naval.

Bien qu'il y ait beaucoup de jeux de tir (la majorité sont à joueurs multiples), ils sont construits à partir de moteur de jeux beaucoup plus petits. Ces jeux offrent une structure et une infrastructure de développement de jeux pour élaborer une vaste gamme de jeux.

Résultats

Cinq moteurs de jeux de tir à joueurs multiples ont été comparés selon 22 critères d'évaluation afin de juger de la faisabilité de les utiliser pour élaborer un jeu à joueur multiple incorporant une Frégate canadienne de patrouille (FPC). L'intention était de construire un jeu permettant aux forces bleues et rouges d'attaquer et de défendre une FPC afin d'expérimenter et de développer un concept de protection de la Force. L'évaluation complète est présentée dans le rapport.

Importance

Il a été déterminé que deux moteurs de jeux (Unreal et Cry) pouvaient convenir. Le moteur d'Unreal s'est classé premier principalement à cause de sa vaste base d'utilisateurs mieux établie.

Perspectives

La modélisation 3D d'une FPC est en développement pour être importée dans un jeu de tir basé sur le moteur de jeu d'Unreal qui sera modifié afin d'offrir un exemple d'environnement immersif. Ce « jeu » sera ensuite utilisé pour effectuer l'expérimentation de contrôle et de commande de protection de la Force.

Table of contents

List of tables

1. Introduction

Recent incidents, such as the attack on the USS Cole, demonstrate the need to think about force protection for defending warships while in port and/or otherwise. As part of developing Maritime protection procedures, Defence Research and Development Canada (DRDC) – Atlantic requires an environment to simulate assaults on, and defence of, warships while in port. This simulation environment needs to be real-time, human-in-the-loop, in the style of current 3D games like America's Army. It is believed that a commercial first-person shooter game will provide sufficient realism for initial investigations and may allow the Canadian Forces (CF) to develop tactics and policies for force protection. For example, computer simulated scenarios may allow the CF to:

- Determine the number of sentries required while docked;
- Determine the most effective location of surveillance equipment (such as cameras and motion detectors);
- Develop defensive tactics for attacks from the land side;
- Choose the safest mooring point in an unfriendly harbour; and
- Choose the best mixture of light weapons to defend against suicide attacks.

The following study, therefore, examined the suitability of the latest commercial 3D first person shooter games and 3D game engines for the development of an upper deck model of a Halifax class Canadian Patrol Frigate (CPF) moored at port, including upper decks, some interior spaces and model surveillance cameras for use by the defending forces. The 3D model also required red and blue force personnel with standard issued weapons and equipment. Because the CPF model will be used in multi-player scenarios for developing tactics and policies for ship defence in port, it was important for the ship model to be able to be used by a variety of game engines.

The following report is broken down into four sections: Aim, Evaluation Criteria, Assessment, and Discussion and Suggestions.

2. Aim

The following aims were pursued in this study:

- Review relevant information regarding the availability, cost, capabilities, and quality of commercial 3-D game engines; and
- Obtain and assess 5 commercial 3-D game engines that are available at a low cost and compare the pros and cons of each engine according to the requirements for the CPF model.

3. Evaluation Criteria

The following criteria were used to evaluate the commercial 3D game engines:

1. Audio Processing – the 3D game engines ability to generate environmental sounds, special effects, etc.
2. EAX Support - Environmental Audio Extension; audio special effects
3. Sound Shaders – allows fine adjustments of audio properties
4. Video Interface – degree of video card requirements
5. Stencil Shadows (or Shadow Volumes) – the ability to generate geometrically accurate shadows
6. Vertex Lighting - determines the amount of light from light sources falling on objects or characters in game environment, creating realistic lighting and shading effects.
7. Geometric Mapping – it does fine variations to the geometry, using a texture like system (e.g. bump-mapping, normal-mapping, diffused specular bump-mapping)
8. Pixel Shading - creates surface detail on characters or objects (e.g. semi-translucent skin, shiny, reflective and irregular surfaces); it is not limited to surface detail, it also generates distortion and fine visual adjustments
9. Lip-synch – matching lip motion to speech
10. Software Developer Kit (SDK) – collection of source code and tools to modify the game
11. Modification Capacity – the ability to alter the game and its operations
12. Physics – how objects behave in the game environment, such as falling, walking, etc.
13. Particle Rendering – the capacity to generate small particles, such as sparks, water drops, in the game environment
14. Primitive Properties – how the game treats different polygons, such as breakable objects, moveable objects, etc.
15. Network Protocol – communication between computers
16. Expansive Outdoor Scenes – how large a terrain the engine can handle
17. Refraction - the quality of light when it moves from air to another substance (e.g. objects appear to bend when inserted into water).
18. AI - artificial intelligence; algorithms applied to characters to simulate actions of intelligent beings; how characters react to other characters and objects within the game environment
19. Multi-player – how many players are capable in a given map
20. Ballistics – how bullets behave in the game, i.e., what they do when they hit objects
21. Volumetric Fog – the ability to depict realistic fog
22. 3D Skybox - non-interactive portion of game map outside bounds of game play; creates the illusion game environment is larger than is modelled

4. Assessment

Following a general search, researchers chose four new cutting edge commercial 3D game engines and one older, proven 3D game engine to examine in detail. These include Texas-based developer id Software's *Doom 3*, Washington-based Valve's *Source*, Coburg-based (Germany) CryTek's *CryEngine*, Raleigh-based Epic's *UnrealEngine 2.5*, and Valve's original *Half-Life* engine.

Table 1 below is an assessment, identifying the pros and cons of each, followed by our summary ranking.

Table 1: Assessment of Unreal Engine 2.5

Developer Engine Flagship game (Reviewer Rank)	**Evaluation Categories**	**Pro**	**Con**
Epic Unreal Engine 2.5 Unreal Tournament 2004 (Reviewer Rank = 1)	**Audio Processing**	Real-time voice chat built-in	
	EAX Support	Audio processing supports A3D, patch also available to support EAX; DSP can calculate Doppler shift	
	Sound Shaders		
	Video Interface	Video interface does not indicate PCI Express is required	
	Stencil Shadows (or Shadow Volumes)	Good shadow technology – likely stencil shadows	
	Vertex Lighting	yes	
	Geometric Mapping	Normal mapping and bump mapping	Does not have a materials database – limited in kind of properties can give objects
	Pixel Shading	Pixel shaders	
	Lip-synch		No, must be manually coded in model animations
	Software Developer Kit (SDK)	SDK code (U-script) can create referee abilities	Extent of SDK modifiability is unknown; unsure if it will provide enough flexibility and versatility
	Modification Capacity	U-Script can provide realistic weapons effects on players Strike Force (realistic tactical simulator) suggests that Unreal Tournament 2004 has decent mod code	

	Physics	Physics (player and non-player) is done with Karma engine; supports character inverse-kinematics	
	Particle Rendering	Yes	
	Primitive Properties	Allows for texture and effect grouping (such as water) through it's "materials browser"	Not as versatile as Half-Life 2 with physics properties specific to materials
	Network Protocol		
	Expansive Outdoor Scenes	Very good expansive outdoor scenes	
	Refraction		
	Artificial Intelligence	UnrealEd provides path nodes for AI, with various types for different deviation tolerances	
	Multi-player	Players 32, maybe more	
	Ballistics		
	Volumetric Fog	yes	
	3D Skybox	Has isolated 3D skybox; renders accurate visual representation of dockyard environments, personal equipment, ships, weapons, etc. Skybox isolation allows for distant "infinite parallax" animated objects (ie. Clouds, aircraft, etc.)	
	General		Can generate commonly used surveillance equipment, though parallax might be an issue – the original Unreal Tournament allowed cameras to be dropped in the world, but the view angle panned around as the player moved w.r.t. the projection surface; the effect was more like looking through a window than looking at a video screen; it is unknown if this same limitation still exists with *Unreal Engine 2.5*

Table 2: Assessment of Cry engine

Developer Engine Flagship game (Reviewer Rank)	Evaluation Categories	Pro	Con
Crytek **CryEngine** **Far Cry** **(Reviewer Rank = 3)**	**Audio Processing**	Sound effects are top notch (e.g. insects buzz in ear, vehicles in distance); supports 5.1 surround sound	
	EAX Support	EAX support, some CPU-based DSP	
	Sound Shaders		Could not determine
	Video Interface	No indication PCI Express video interface is required	
	Stencil Shadows (or Shadow Volumes)	Stencil shadows, reflections on walls and weapons; excellent graphics and textures; excellent shadow effects; waves; reflections of terrain in water; Volumetric smooth-shadow implementations for realistic, dramatic indoor shadowing	
	Vertex Lighting	Yes	
	Geometric Mapping	Normal mapping and bump-mapping; lots of polygons Developed PolybumpTM; allows users to create and render extremely low poly model using an ultra high poly model; rendering time greatly reduced; increases image quality	Excellent graphics require processing power
	Pixel Shading	Excellent pixel shading Water ripples in the sunlight	
	Lip-synch	Yes unsure if it is automatic or manually coded	
	Software Developer Kit (SDK)	SDK allows weapons to be modelled with realistic effects on players; SDK can most likely create referee abilities	Though SDK will allow players to create total conversions, its effectiveness is unknown; full extent of modification with Sandbox and SDK unknown
	Modification Capacity	CryEngine Sandbox editor allows modders to create levels in real-time and play them (quick basic editing)	Terrain and structural deformation increases poly count; vehicles can most likely be modelled
	Physics	Advanced physics engine that makes most things interactive (e.g., wooden crates knocked into the water and bob, cups	

		falling from tables, etc.); character-inverse-kinematics; use of land/water vehicles	
	Particle Rendering	Yes	
	Primitive Properties		Advanced physics engine has limits; some things not as destructible as expected; might be able to produce wind effects on boats
	Network Protocol		Do not know
	Expansive Outdoor Scenes	Expansive outdoor scenes are excellent; up to 2 km	
	Refraction		Though water looks impressive, unsure if it truly refracts
	Artificial Intelligence	Dynamic “revolutionary” AI; enemy respond to players – know player behaviour and protect themselves; customizing enemies and behaviours without touching the core C++ code	Because the AI in Cry Engine is so complicated, it increases the drain on the computer resources in comparison to other engines
	Multi-player	Multiplayer 32	
	Ballistics	Ballistics code features target location specific damage and gravity effects	
	Volumetric Fog		
	3D Skybox	Has 3D skybox; renders accurate visual representation of dockyard environments, personal equipment, ships, weapons, etc.	
	General	Smooth frame rate Most likely can generate commonly used surveillance equipment	Difficulty of implementing hand signals unknown Might be able to have referee abilities Very new; risks are unknown

Table 3: Assessment of Doom 3 Engine

Developer Engine Flagship game (Reviewer Rank)	Evaluation Categories	Pro	Con
id **Doom 3** **Doom 3** **(Reviewer Rank = 2)**	**Audio Processing**	3D sound calculations on CPU is not limited by DirectSound; developers can programmatically modify in-game sounds (such as positional sound of voices from other players); do not have to upgrade sound card; 5.1 surround sound provides good sound effects	Unsure about doppler shifting
	EAX Support		No
	Sound Shaders		No details could be located
	Video Interface	Runs equally well on AGP and PCI Express	
	Stencil Shadows (or Shadow Volumes)	Stencil Shadows are impressive and enhance lighting effects; Doom 3 draws invisible shadows for every light source and every object in the game; multiple polygons	
	Vertex Lighting	Yes	
	Geometric Mapping	Normal mapping contains the height of each pixel on the base surface, and encodes the direction each pixel faces; adds realism without increasing the polygon count	
	Pixel Shading	Pixel shaders (good for special effects like distortion in water, waves, etc.)	Pixel shaders require processing power
	Lip-synch		Not automatic.
	Software Developer Kit (SDK)	SDK allows weapons to be modelled with realistic effects on players (for example, it depicts the amount of damage when shot, recoil effect during fire, etc.) SDK allows for referee abilities to be added	
	Modification Capacity		Full extent of modification capacity unknown (i.e. reviewers have not seen any "total conversion" multiplayer mods yet); common personal equipment can be added; weapon remodelling will be

			required Difficulty of modelling vehicles unknown
	Physics	Character-inverse-kinematics	May offer less interactivity than other games Unsure if physics can provide wind effects on light boats
	Particle Rendering	Yes	
	Primitive Properties		Primitive Properties handling method are unknown (i.e. Database, or basic polygon properties)
	Network Protocol		It might use more bandwidth than the other games
	Expansive Outdoor Scenes		Terrain and structural deformation increases poly count
	Refraction		Unknown
	Artificial Intelligence (AI)	AI scripted	Difficulty of scripting AI for non-players and hand signals is unknown
	Multi-player		5 maps; possible to add up to 16 players/map , but will have to compromise shadow effects, rag dolls and network will have to be fast; bugs joining on line servers as well; to run 16 players, will need a “beefy” server
	Ballistics	Ballistics control allows for target specific damage and gravity effects	Ballistics likely not affected by wind
	Volumetric Fog	Yes	
	3D Skybox	Has animated skybox; renders accurate visual representation of dockyard environments, personal equipment, ships, weapons, etc.	Skybox animation somewhat limited
	General	Runs on a wide range of hardware, maintaining decent frame rates Good expansive outdoor scenes; BSP and occlusion Four rendering paths (NV10, NV20, R200, ARB2) Can generate commonly used surveillance equipment	Loses visually impressive features if run on older hardware; may require video card upgrade Higher quality textures only used if high end video card installed. Multiple polygons require processing power; stress on GPU resources

Table 4: Assessment of Half-Life engine

Valve **Half-Life** **Half-Life** **(Reviewer Rank = 4)**	**Audio Processing**	CPU-based audio processing	
	EAX Support	Support for EAX	
	Sound Shaders		No
	Video Interface		Developed before PCI Express was invented
	Stencil Shadows (or Shadow Volumes)		No stencil shadows; only pre-computed shadows possible; shadow resolution somewhat limited by QBSP
	Vertex Lighting		Yes, but not dynamic. (ie. Pre-rendered, not in real-time)
	Geometric Mapping		No geometric mapping
	Pixel Shading		No pixel shaders
	Lip-synch		No lip-synch
	Software Developer Kit (SDK)	SDK covers client and server game code – complete, except for AI	
	Modification Capacity	Lots of mods for it; development time short; lots of tutorials and examples; lots of support on web forums and mod community	
	Physics		Internal physics, sloppy; inertia and momentum were non-existent, for example, if a player stood on a moving platform and then jumped, the player velocity w.r.t. the world coordinates would instantly decelerate to 0 and the moving platform would slip away underneath; as well, when firing RPGs and other weapons from a moving frame of reference, the bullets would follow a trajectory from a static frame of reference
	Particle Rendering	Yes	
	Primitive Properties	Primitive properties include basic types of solids, invisible, liquids, animations, and skybox/parallax scrolling	
	Network Protocol	Network protocol is limited packet structure, basic prediction	
	Expansive Outdoor Scenes		Poor expansive outdoor scenes
	Refraction		No
	Artificial Intelligence	Some AI mods have been released by mod community	AI codes have not been released for free

	Multi-player	32 players	
	Ballistics	Ballistics are basic location-specific damage (head, torso, appendages, and direction of attack); basic visual effects for bullets	
	Volumetric Fog		
	3D Skybox		No
	General	Low system requirements Around long time, patches and bug fixes released Low cost	Old game, with limitations; does not have the image quality of new, cutting edge 3D game engines

Table 5: Assessment of Halflife 2 Engine

Developer **Engine** **Flagship game** **(Reviewer Rank)**	**Evaluation Categories**	**Pro**	**Con**
Valve **Source** **Half-Life 2** **(Reviewer Rank = 5)**	**Audio Processing**	• Audio processing is CPU-based DSP; doppler shifting present for ballistics	•
	EAX Support	• Worked with Creative Labs to support EAX	•
	Sound Shaders	•	•
	Video Interface	• PCI Express is not required	•
	Stencil Shadows (or Shadow Volumes)	•	• Some use of stencil shadows, but rendering technology unknown
	Vertex Lighting	•	•
	Geometric Mapping	• Bump mapping; normal mapping contains the height of each pixel on the base surface, and encodes the direction each pixel faces; maintains the polygon count • Diffuse and specular bump mapping render things that are shiny and bumpy, and rough bumped surfaces, which will be most obvious when using a flashlight	•
	Pixel Shading	• Pixel shaders for water	• Pixel shaders and processing power
	Lip-synch	• Lip-synching – lips and face programmed automatically recognize what is said and animate correctly (Source engine handles a lot)	•
	Software Developer Kit (SDK)	• SDK allows weapons to be modelled with realistic effects on players • SDK can create referee abilities	•
	Modification Capacity	• More modable than *Half Life* engine; "complete control over rendering loop, networking system for multiplayer, detailed control over most other game aspects"	•
	Physics	• Physics system provides good, realistic musculature beneath character clothing; kinematic animated bone followers; character-inverse-kinematics; AI	•

		characters can interact with physically simulated objects vehicles are either raycast or have accurate “wheel physics”, i.e., the engine provides two modes of handling vehicles - one basically draws the vehicle at correct location, the other performs more realistic physics on the vehicle (wheel physics)	
	Particle Rendering	• Particle and lighting system (e.g. dust specks, real-time lighting, mirror effects, etc); flickering and pulsing effects; water with refraction and fresnel (i.e. shiny) effects	•
	Primitive Properties	• Primitive properties - materials database specifying fracture, mass, buoyancy	•
	Network Protocol	• Network protocol includes predictive analysis with interpolation and extrapolation for collision/hit detection and custom packet sizes / data	•
	Expansive Outdoor Scenes	• Bump mapping renders expansive outdoor areas; BSP and occlusion	•
	Refraction	•	•
	Artificial Intelligence	• AI scripted with intelligence, through SDK code	• Difficulty of scripting AI for non-players, and hand signals is unknown
	Multi-player	• Maximum players 32	•
	Ballistics	• Ballistics code renders individual Bullets	• Ballistics likely not affected by wind, definitely not affected by coriolis force
	Volumetric Fog	• Yes	•
	3D Skybox	• Has 3D skybox; renders accurate visual representation of dockyard environments, personal equipment, ships, weapons, etc.	•
	General	• It can run on various systems • Structures for future scalability • Can generate commonly used surveillance equipment	• All Valve products use Steam, which is a major disadvantage for protected networks • Terrain and structural deformation increases poly count

4.1 Requirements

The following provides a break down of the requirements needed to run each 3D game engine.

4.1.1 Doom 3:

- 3D Hardware Accelerator Card Required - 100% DirectX® 9.0b compatible 64MB Hardware Accelerated video card and the latest drivers*.
- Microsoft® Windows® 2000/XP
- Pentium® IV 1.5 GHz or Athlon® XP 1500+ processor or higher
- 384MB RAM
- 8x Speed CD-ROM drive (1200KB/sec sustained transfer rate) and latest drivers
- 2.2GB of uncompressed free hard disk space (plus 400MB for Windows® swap file)
- 100% DirectX® 9.0b compatible 16-bit sound card and latest drivers
- 100% Windows® 2000/XP compatible mouse, keyboard and latest drivers
- DirectX® 9.0b (included)

Multiplayer requirements:

- Internet (TCP/IP) and LAN (TCP/IP) play supported
- Internet play requires broadband connection and latest drivers
- LAN play requires network interface card and latest drivers[1]

Supported chipsets:

- ATI® Radeon(tm) 8500
- ATI® Radeon(tm) 9000
- ATI® Radeon(tm) 9200
- ATI® Radeon(tm) 9500
- ATI® Radeon(tm) 9600
- ATI® Radeon(tm) 9700
- ATI® Radeon(tm) 9800
- All nVidia® GeForce(tm) 3/Ti series
- All nVidia® GeForce(tm) 4MX series
- All nVidia® GeForce(tm) 4/Ti series
- All nVidia® GeForce(tm) FX series

[1] *Important Note: *Some 3D accelerator cards with the chipset listed here may not be compatible with the 3D accelerator features utilized by Doom 3. Please refer to your hardware manufacturer for 100% DirectX 9.0b compatibility. This product does not support Microsoft® Windows® 95/98/ME or NT.*

- nVidia® GeForce(tm) 6800

4.1.2 Unreal Engine 2.5:

- Operating System: Windows® 98/Me/2000/XP
- Processor: Pentium® III or AMD Athlon 1.0 GHz processor or faster (1.5 GHz or faster processor recommended)
- Memory: 128 MB RAM minimum (256 MB recommended)
- Hard Disk Space: 5.5 GB free
- Video: Any Windows-compatible video card (NVIDIA GeForce 2 or ATI Radeon with at least 64 megs of video memory recommended)
- Sound: Windows®-compatible sound card. NVIDIA® nForce(tm) or other motherboards/soundcards containing the Dolby® Digital Interactive Content Encoder required for Dolby Digital audio
- DirectX®: DirectX® version 8.1(included)or higher
- Multiplayer: Internet (TCP/IP) and LAN (TCP/IP)play supported | 33.6K baud modem or broadband Internet connection recommended

4.1.3 CryEngine:

- Computer: 1 GHz CPU
- Memory: 256 MB RAM
- Operating System: Windows® 98/ME/2000/XP
- Video Card: Geforce 4 MX, Geforce 3 or higher
- Drive Space: 582Mb hard drive space
- DirectX 9.0b must be installed
- Nvidia based cards must have : ForceWare drivers 53.03
- ATI based cards must have: Catalyst drivers 3.9

Recommended requirements:

- Computer: 2 GHz CPU
- Memory: 512 MB RAM
- DirectX 9.0b must be installed
- Nvidia based cards must have : ForceWare drivers 53.03
- ATI based cards must have: Catalyst drivers 3.9

Required operating systems:

- Windows® 98, Windows® ME, Windows® 2000, and Windows® XP

4.1.4 *Half*-Life:

- Windows® 95, Windows 98 or Windows NT 4.0

- Pentium® 133
- 24 MB RAM
- 2X CD-ROM drive
- Mouse and Keyboard
- 640x480 SVGA high color (16-bit) display
- Windows-compatible sound device

Recommended requirements:

- Pentium® 166+
- 32 MB RAM
- 3D accelerator card (OpenGL or Direct 3D)

4.1.5 *Source* (Half-Life 2):

Minimum system configuration

- 1.2 GHz Processor
- 256MB RAM
- DirectX 7 capable graphics card
- Windows 2000/XP/ME/98
- Mouse
- Keyboard
- Internet Connection

Recommended system configuration

- 2.4 GHz Processor
- 512MB RAM
- DirectX 9 capable graphics card 256mb
- Windows 2000/XP/ME/98
- Mouse
- Keyboard
- Internet Connection

5. Discussion and Suggestions

All of the commercial 3D game engines reviewed have impressive graphics and capabilities. They all can be used to produce a model Halifax class CPF for computer scenarios, which includes a complete model of the interior of the ship. Moreover, they also have the capacity, at varying degrees, to accomplish the following:

- Present a correct visual representation of dockyard environments, personal equipment, ships, weapons, etc.;
- Include the upper decks and interiors of major compartments accessible from the deck modelled for the CPF;
- Provide enough multi-player capacity to allow complete red and blue forces as well as "spectators" to act as video surveillance cameras, referees, and observers (with the exception of *Doom 3*);
- Script AI for non-player vehicles and bystanders, and events;
- Include explosive devices, grenades, rockets with realistic damage effects on people and vehicles (cratering, terrain deformation, and other blast damage effects can come later);
- Add hand signals (with the exception of *CryEngine*);
- Generate smoke, fog, rain, etc.;
- Add common personal equipment (binoculars, night vision, helmets, protective vests, etc), common small arms and other infantry weapons (assault rifles, machine guns, RPGs, etc), and commonly used surveillance equipment;
- Include realistic weapon effects on people (For example, a bullet in the head should kill you, not subtract 70% of your "Life force")
- Provide realistic, though not necessarily real, sound effects, foot-steps, 3D positional sound of voices from players speaking/yelling to each other, and other involuntary sounds;
- Make objects in the environment interactive (i.e., impede movement) and realistic (for example, players should not be able to walk along the top of an iron balcony railing while firing an assault rifle or run across a clothesline);
- Include referee abilities like declaring a player a casualty or determining the end of the scenario (for example, the attackers make it aboard the ship);
- Provide underwater gear (scuba tanks, underwater weapons) and environmental effects;
- Create ocean swells, waves, wind effects on light boats;
- Include good ballistic modelling (like windage, gravity, spin, etc);
- Have blast effects, including terrain deformation and damage to steel structures on the ship;
- Permit drivable land vehicles including crews (for example, a pickup-truck with a driver, a mounted weapon operator, and soldiers who can fire from the back or jump out at will);
- Allow players to build entrenchments before or during a game (for example, dig a fox hole or pile up debris for a barricade or cover); free-moving objects (such as crates) can be pushed around by simple collision. It is also possible to make a hole in the terrain covered by a breakable "ground" material that can be destroyed to "dig the hole".
- Record and play-back of the entire scenario, including frame-by-frame advance, go to a particular position, fast-forward/rewind and so on.

With respect to creating voice chat between team members (e.g. using radio channels as opposed to yelling), it may be possible to accomplish this within the 3D game engines. However, a better alternative might be to create a separate program. It was assumed that since the requirements specification document indicated a need for “radio chatter” and not “yelling”, the voice chat from other players would not require any of the game engine’s environmental / 3D audio processing (i.e. audio from a soldier’s radio will always be emitted from the same location relative to the soldier, and will always sound the same regardless of the environment). Most sound card drivers and audio APIs will allow multiple programs to simultaneously access the sound card. This would allow a completely separate voice chat application to run concurrently with the game. However, there are a few issues.

If the game calls for a different PCM format than the voice chat program (i.e. sample rate, bits per sample, number of channels), there could be a conflict. Also, if the game uses hardware environmental effects such as EAX, there is the possibility that when the game engine selects various EAX parameters (i.e. echo, reverberation, frequency response, etc.) those same parameters might affect the playback of the voice chat. An alternative is to use a hardware solution, where game players are linked by an analog "intercom" system. Mixing the sound would be done on the analog signal chain. A hardware based system does not tax the computers or network, but will require some development time (either purchasing a simple wired intercom and modifying it, or building an intercom system from scratch). Note that a software system will also require development time if a custom program is to be written or if the game engine audio system needs to be modified to provide suitable voice chat. If it is possible to use the game engine audio system for voice chat, it may also be possible to link game play with the audio control (i.e. automatically disable a player's comms when the player dies, or add noise to simulate a fading transmission on an analog radio between distant players. If a hardware system is implemented, killing a player's comms on death can be implemented fairly easily, assuming the SDK provides for I/O interfacing. But complex audio processing and control, such as noisy fading transmissions, would require significant development time.

After assessing each 3D game engine, it became apparent that some were more suitable than others. For example, though *Source* is a good engine, it requires an account with Steam, *Valve’s* new online distribution and anti-piracy requirement. Steam is a centralized service where subscribers can download games and other data. Steam’s centralized service also allows multiplayer games to connect when searching for other game servers. Many new products from *Valve*, including Half-Life 2, are encrypted and/or incomplete until the installer successfully connects to Steam and determines that a valid license has been purchased. Following this, the installer downloads either the decryption key or the remaining content. In essence, Steam prevents unauthorized “pirate” use of *Valve’s* products. However, it becomes a major obstacle for some legitimate uses of *Valve’s* products.

For example, Half-Life 2 will not run until it has been registered (over the Internet) and validated on Steam. Normally, when the game starts up, it connects to Steam over the Internet. Without this connection, the game will not play. It is possible to play offline without violating licence agreements, but it is a very messy and, more importantly, unreliable procedure.

The procedure for playing offline is as follows:

- The game must first be installed on the computer while connected to the Internet.

- Once the game has been unlocked and activated on Steam, the Internet connection must physically be disconnected without logging off of Steam. In other words, pull the plug, but do <u>not</u> hit the logoff button.

- The next time the computer is booted and Half-Life 2 is executed, it will attempt to connect to Steam. After approximately 90 seconds, it will time out and revert to “offline” game play.

- At this time, the game is functional. However, if at any point in time the logoff button is pressed, “offline” game play is disabled until the computer is reconnected to the Internet, where it can connect to Steam the next time the game is started.

On a protected network, such as the DRDC DREnet, firewalls block Internet traffic for custom protocols, including those used by Steam. Further, it is against the end user license agreement declared by *Valve* to tunnel the Steam protocol through a different protocol. Unfortunately, this prevents Steam from operating within a DRDC compound. Even if the computers were temporarily removed from the DRDC compound and moved to an unblocked Internet connection to install Half-Life 2, if the logoff button is ever accidentally pressed, Half-Life 2 would disappear until it was reinstalled again, following the same procedures.

Moreover, once a Steam powered product, such as Half-Life 2, is installed on a computer, it can not be transferred to another computer without reinstallation. This means that that the very same computer which is used to play the game must be the same computer used to activate it over the Internet. If Half-Life 2 is to be run on large, heavy computer systems (i.e., rack mounted cases), or computers buried deep in cabinets or cables, the set up must be disassembled for activation and any re-activation. It is impossible to create a backup of the installed game to use as a rescue should it ever fail to connect to Steam again.

Despite the fact that Half-Life 2 is an impressive game and the SDK for *Source* is excellent (i.e. the engine becomes quite versatile), it is locked up in a content control and validation system, Steam. If a license is purchased for the *Source* engine (i.e. entire source code), then the Steam requirements can more than likely be pulled. However, this is likely an expensive route. The original *Half-Life* is an option because it is proven and reliable. But it does not have the same capabilities of the new commercial 3D game engines (see Table 4).

On the other hand, *CryEngine* is very good. *CryEngine* is a relatively new engine and two different interactive game demos failed to install. A non-interactive demo video was downloaded and ran at a very slow pace, so much so that the audio and video lost synchronization by a couple of seconds. After purchasing the software, reviewers were impressed with the level of graphical detail. The AI seems to anticipate the player’s moves, thereby changing each time a mission is run. The AI also tries to avoid being killed, employing stealth, etc. It has incredible expansive outdoor scenes (e.g. up to 2 km). Though *CryEngine* is a promising commercial 3D game engine, it still remains “buggy” and the risks are truly unknown. Program patches are still being implemented by CryTek to correct software malfunctions to improve overall game execution. Because it is such a new 3D game engine and given the strict timeline for the second part of this technical demonstration, it might prove to be too time consuming mastering its code for modification.

Doom 3 is a very high quality commercial 3D game engine, and a good choice for the Force Protection Simulation Demonstration. The sound system is great. The lighting is excellent. Character bone movements are grouped so that multiple animations can run simultaneously (i.e. head and torso separate from the feet). The interactive program demo ran extremely well on the HSI computer. The SDK documentation appears to cover a lot, and the entire game code is distributed for free non-commercial use (i.e. non-profit). It was built with Microsoft Visual

Studio.NET 2002. Some information is available to compile with Visual Studio.NET 2003/2005. There are exporters available for Maya and 3D Studio Max as well as model viewers. The DoomEdit map editor is similar to Q3Radiant, a long time popular map editor, but tailored for *Doom 3*.

One potential drawback to *Doom 3*, however, is the number of players it can include. When first run, the reviewers found that it was not possible to have more than four players / map. However, follow up enquiry to id Software revealed that it is possible to add more than four, a possible 16, but this will compromise shadow effects, rag dolls (i.e., death animation), and possibly force some entities (i.e. automatic doors, sound emitters, etc.) to be rendered client-side only. Moreover, the network will have to be fast with an extremely fast, dedicated game server. At this time, it is impossible to determine how many players can be added and what the overall consequences of adding more will be without modifying the code on the engine. However, reviewers were able to join a game of Doom 3 online, which had more than 8 players.

The *Unreal Engine 2.5* demo program ran extremely well on the HSI□ computer. It was also interactive, so reviewers were able to play. The SDK released for free, non-commercial use only provides the U-Scripts, which may or may not provide enough control over the game for a Force Protection simulation. In other words, there is no C++ code made available as a free, non-commercial SDK for Unreal Tournament. However, the "Unwheel" mod and a few others look like there has been substantial changes. But to maintain human realism, the access to the physics in this engine is critical because currently, players have the ability to leap to great heights and run at unprecedented speeds. One interesting feature in the Unreal Tournament 2004 is the built-in real-time voice chat. But reviewers are unable to determine both how much control the SDK would provide over this feature or whether this feature would require modification if used.

UnrealEd, the map editor for Unreal Engine 2.5, is currently the only editor which successfully imported the CPF static mesh model with solid surfaces. The tools released for CryEngine caused 3D Studio Max to crash. Doom Edit had several undocumented quirks, and thus, the reveiwers were unable to import the CPF ship without "black hole" surface textures. It is believed that UnrealEd can also convert between CSG brushes and static meshes. However, current attempts have so far been unsuccessful. Nevertheless, it is our opinion that *Unreal Engine 2.5* is the best commercial 3D game engine for the purposes of Force Protection Simulation Demonstration, followed by *Doom 3* and *CryEngine.*

Bibliography

"Neowin.net > Problem loading Far Cry Demo -- HELP!" _Neowin.net_ [Online] Available http://www.neowin.net/forum/lofiversion/index.php/t269009.html, Jan 9, 2005

"Unreal Engine 2" _Unreal Technology_ [Online] Available http://www.unrealtechnology.com/flash/technology/ue2.shtml, Jan 2, 2005

"Marvelous Mods" _Maximum PC_ Fall 2004, p.101

"Jamlander" [Online] Available http://www.unrealtournament.com/ut2003/ct_winners_bt_2.php

Maximum PC Fall 2004, p.80

Maximum PC Fall 2004, p.18

"Valve Software's Source Engine" _Maximum PC_ Fall 2004, p.53

"Half-Life 2 - Steam Fooled/Patched" _Half Life Source_ [Online] Available http://www.halflifesource.com/forums/showthread.php?threadid=22602, Nov 16, 2004

"How To Make Steam Offline Mode Work" _Steam_ [Online] Available http://steampowered.custhelp.com/cgi-bin/steampowered.cfg/php/enduser/std_adp.php?p<URL truncated>, Jan 14, 2005

"Half-Life 2 / Source Mod FAQ" _VERC Collective_ [Online] Available http://collective.valve-erc.com/index.php?faq=source_mod_faq&section=106159033732<URL truncated>, Aug 23, 2003

"Doom 3" _Maximum PC_ Fall 2004, p.93

"9 Things You Need To Know About Doom 3" _Maximum PC_ Fall 2004, p.94

"id Software's Doom 3 Engine" _Maximum PC_ Fall 2004, p.50

Kasavin, Greg "Doom 3 Review" _GameSpot_ [Online] Available http://www.gamespot.com/pc/action/doom3/printable_6104145.html, Aug 4, 2004

"Far Cry" _Maximum PC_ Fall 2004, p.79

"Crytek's CryEngine" _Maximum PC_ Fall 2004, p.55

"Crytek Technology: Specifications v 1.21" _Crytek_ [Online] Available http://www.crytek.com/polybump/index.php?lang=eng&sx=polybump, Jan 2, 2005

'Pistol Pete' "PC Games - Far Cry Review" _newegg.com_ [Online] Available http://www.gamers-depot.com/games/action/far_cry/001.htm, Jan 2, 2005

Crecente, Brian "Far Cry for PC video game review and cheats" _Gamezilla_ [Online] Available http://www.gamezilla.com/review.aspx?review=8835, Apr 3, 2004

Bosworth, Tracy "Far Cry PC Review" _real GAMER_ [Online] Available http://www.realgamer.co.uk/pcgames/review/far_cry.html, Apr 17, 2004

Distribution list

Document No.: DRDC Atlantic CR 2005-088

LIST PART 1: Internal Distribution by Centre:

6	DRDC Atlantic Library
1	Allan Gillis
1	Mark Hazen
8	TOTAL LIST PART 1

LIST PART 2: External Distribution by DRDKIM

1	DRDKIM-3
1	TOTAL LIST PART 2

9 TOTAL COPIES REQUIRED

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (The name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Centre sponsoring a contractor's report, or tasking agency, are entered in section 8.)

Humansystems Incorporated
111 Farquhar St., 2nd Floor
Guelph, ON N1H 2N4

2. SECURITY CLASSIFICATION (Overall security classification of the document including special warning terms if applicable.)

Unclassified

3. TITLE (The complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S, C, R or U) in parentheses after the title.)

Assessment of 1st Person Shooter Game Engines for Force Protection Applications

4. AUTHORS (last name, followed by initials – ranks, titles, etc. not to be used)

Frim, J. and Thomson, M. H.

5. DATE OF PUBLICATION (Month and year of publication of document.)

January 2006

6a. NO. OF PAGES (Total containing information, including Annexes, Appendices, etc.)

21

6b. NO. OF REFS (Total cited in document.)

0

7. DESCRIPTIVE NOTES (The category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

Contract Report

8. SPONSORING ACTIVITY (The name of the department project office or laboratory sponsoring the research and development – include address.)

DRDC Atlantic

9a. PROJECT OR GRANT NO. (If appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant.)

11BK

9b. CONTRACT NO. (If appropriate, the applicable number under which the document was written.)

W7707-4-2677/A

10a. ORIGINATOR'S DOCUMENT NUMBER (The official document number by which the document is identified by the originating activity. This number must be unique to this document.)

10b. OTHER DOCUMENT NO(s). (Any other numbers which may be assigned this document either by the originator or by the sponsor.)

DRDC Atlantic CR 2005-088

11. DOCUMENT AVAILABILITY (Any limitations on further dissemination of the document, other than those imposed by security classification.)

(X) Unlimited distribution
(   ) Defence departments and defence contractors; further distribution only as approved
(   ) Defence departments and Canadian defence contractors; further distribution only as approved
(   ) Government departments and agencies; further distribution only as approved
(   ) Defence departments; further distribution only as approved
(   ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (Any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). However, where further distribution (beyond the audience specified in (11) is possible, a wider announcement audience may be selected.))

13. ABSTRACT (A brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), (R), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual.)

First Person shooter games allow humans to be immersed in a desktop virtual world. There are a variety of commercial off-the-shelf (COTS) multi-player game engines available that provide the infrastructure for building these worlds. Five game engines are assessed against 22 criteria to determine the their usefulness in providing a virtual world for naval Force Protection scenarios. The aim of this work is to use the resulting virtual world for concept-development and experimentation for Force Protection operations.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (Technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus, e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

First Person Shooter
Games
Game Engine
Assessment

This page intentionally left blank.