Computer Graphics for Games Video Game Technologies

<u>_____</u>

14475: MSc in Computer Science and Engineering 13828: MSc in Game Design and Development

Chap. 2 — Game Engines Game Engine

Overview

- What is a game engine?
- Game engines:
 - Commercial
 - Open source
- Game engine architecture
 - Physics, Al, Graphics, etc.

What is a game engine?

- A **game engine** is the core software component of a computer or video game or other interactive application with real-time graphics (taken from Wikipedia)
- The term "game engine" was coined in the mid-1990s due to the development of first person shooters such as *Doom*, *Wolfenstein 3D*...



Episode I: Knee-Deep in the Dead takes place in the facilities of the UAC and the military on Phobos.



The title screen showing the protagonist B.J. Blazkowicz waiting in ambush.

1010111010000

Game engine: main goals

- Provide for underlying technologies
 - Graphics Rendering
 - Physics engine
 - Sound
 - Scripting
 - Animation
 - Artificial Intelligence
 - Networking
 - ...
- Simplify development process
- Run on multiple platforms

Top 10 commercial engines

(http://www.develop-online.net) at Friday, 26th June 2009

- Unreal Engine 3
- Gamebryo Lightspeed
- CryEngine 3
- Unity 3D
- BlitzTech
- Infernal Engine
- Vision Engine 7.5
- Bigworld Technology Suite
- Vicious Engine 2
- Torque 3D







The Best Game Engine for Beginners in 2023 (https://cutt.ly/v8bnKIS) at Friday, 11th January 2023

- Unity
- Unreal Engine
- Gamemaker
- CryEngine
- Phaser
- Godot



Unity Engine



CryEngine

101010101011010000

Open-source engines in 2023 (https://cutt.ly/S8bQuNG)

- Armory
- Cry Engine
- Flax Engine
- Gdevelop
- Godot Engine
- jMonkey
- OGRE
- Open3D Engine
- OpenSceneGraph
- Panda3D



Armory

Godot



.01.01.01.01.01.101.0000

Game middleware

- Components in game engines can be based on **middleware** (Havok, SpeedTree, ...)
- Increasing popularity of MMOGs spawns new middlewares:
 - Gamebryo, HeroEngine, RealmCrafter, MultiverseNetwork, ...

• Advantages in using a game engine:

- Less development time required
- Less testing and debugging
- Many features directly available
- Better focus on the game design

Disadvantages in using a game engine:

- No control over the implementation of features
- Adding features not yet in the game engine might be cumbersome
- Dependent on other licensing scheme for release
- Other libraries/toolkits linked with the game engine (physics, Al...)

The game loop

- A game is a real-time interactive application
- Three tasks that run concurrently:
 - Recompute the state of the world
 - The player interacts with the world
 - The resulting state must be presented to the user (graphics, sound, etc.)
- Limitations of real-world technology
 - I-2 processors with limited memory and speed (10 years ago). And now?

The game loop: coupled approach

- Ist try: design update/render process in a single loop (coupled approach).
- Advantages of the coupled approach:
 - Both routines are given equal importance
 - Logic and presentation are fully coupled

• Disadvantages:

- Variation in complexity in one of the two routines influences the other one
- No control over how often a routine is updated



The game loop: multi-threaded approach

- 2nd try: design update process using two threads:
- **Advantages** of the multi-threaded approach:
 - Both update and render loops run at their own frame rate

• Disadvantages:

- Not all machines are that good at handling threads (precise timing problems)
- Synchronization issues (two threads accessing the same data)





Hardware layer

- Physical
 - Graphics card
 - Sound card
 - Physics card
 - Input devices (keyboard, mouse, joysticks, game pads, steering wheels, ...)
- Drivers
 - Low level interface

Hardware abstraction layer

- DirectX
 - HAL (hardware abstraction layer)
 - Components
 - DirectDraw, Direct3D
 - DirectSound, DirectMusic

rororqrororror

- DirectInput, DirectPlay
- (DirectSetup)
- Still low level routines
- OpenGL
- Others

User interface

- To develop a generic high level design for a simple (2D) game.
- Rather simple
- Monitors input devices and buffers any data received
- Displays menus and online help (can nowadays be pretty complex)
- Should be reusable, especially as a part of a game engine



- Higher level interface, tuned to a particular
 graphics and game type
 - Sprite-based
 - Isometric
 - Full 3D
- Can deal with higher level modeling concepts
 - Sprites
 - Solids
 - Characters (articulated) ...
 - Scene Manager
 - Each scene is represented by a scene graph
 - Contains everything that appears on the screen
 - There may be different scene managers for terrain (heightmap), exterior and interior scenes, ...



- Handles more complicated display aspects
 - Mini map
 - Multiple views
 - Overlays
 - Special effects ...
- Some of these engines are for sale or available on the web
- Often remade or heavily tuned for each game
 - Too much time and money is spent on this

rorojdrororrro<mark>rodoo 470 / Cararo, 10100</mark>

Sound engine

- Function of sound
 - Effects to enhance reality
 - Ambience
 - Clues about what to do
 - Clues about what is about to happen (but be careful)

Sound formats

- Wave (high quality, lots of memory, fast)
- MP3 (high quality, compressed, slower)
- Midi (lower quality, very low storage, limited, adaptable)
- CD (Very high quality, fast, limited to background music)



Game Engines

SONiVOX® Embedded Audio Synthesis (EAS™) technology is a multi-platform audio engine for embedded systems and devices

m/categories/audi

o-engine

- Simultaneous sounds
 - Mixers (hidden in the HAL)
 - Buffer management
 - Streaming sound
 - Special features
 - Positional 3D sound (possibly with Dolby surround)
 - Important for clues
 - Adaptive music (DirectMusic)
 - Some sound engines: https://www.g2.co
 - Wwise
 - FMOD
 - Allegro

OpenAL

Al engine

- Behaviour& interaction (dialogue) scripts
 - Especially in adventure games
- Flocking
- Obstacle avoidance
- Attack strategies
 - Hiding
 - Attacking player as a team of enemies
- Decision making
- Path planning
 - Search algorithms
 - Waypoint networks
- Crowd behaviours

Panic, riots



- Al-Implant
- DirectlA
- SimBionic
- AlSeek (dedicated Al card)

.o.o.d.o.o....o.o.oooo

AI.IMPLANT

https://www.g2.com/categories/physics-engine

Physics engine

- Handles the simulation of the world
 - Collisions
 - Terrain changes
 - Waves in the sea
 - Explosions
 - Object destruction
- Limited or non-existent in simple games
- Some commercial/open source engines:

hav

- PhysX
- Bullet
- Havok
- Tokamak
- JigLib

- Physics hardware:
 - Nvidia/Ageia PhysX
- Physics is more and more integrated into the gameplay and game subsystems
 - Physics-based animation
 - Interaction with objects using physics

Scripting engine

Advantages:

- Easy control of many (or all) features in the game engine
- Scripting language often provides full OO control (like Lua)
- Promotes data-driven design

Disadvantages:

- Performance
- Development support tools
- Learning curve
- Common scripting languages:





Implementing A Scripting Engine

http://www.flipcode.com/archives/Implementing A Scripting Engine-Part 1 Overview.shtml

Scripting engine (cont.)

• What belongs in the engine and what belongs in the script?

Graphics Rendering Shadows/lighting Occlusion culling

Physics

Raycasts

Dynamics Collision detection

engine

ΑΙ

Path-finding Fuzzy controllers Planning/A* search **Graphics** Tome-of-day Add/remove lights Load/moving objects

Physics

Object mass/friction Collision events Raycasts events

AI Path selection Decision making Goals/Objectives

script

rururqrororrorgooo

rororqrororror

Summary:

- What is a game engine?
- Why to build up a game engine?
- Game engines:
 - Commercial
 - Open source
- Game engine components and middleware
- The game loop
- Game Engine Architecture
 - Physics, Al, Graphics, etc.