

Game Engines

CMPM 164, F2019

Prof. Angus Forbes (instructor)

angus@ucsc.edu

Montana Fowler (TA)

mocfowle@ucsc.edu

Website: creativecoding.soe.ucsc.edu/courses/cmpm164

Slack: <https://ucsccmpm164.slack.com>

Class information

Class website:

<https://creativecoding.soe.ucsc.edu/courses/cmpm164>

Slack is the main form of class communication:

<https://ucsccmpm164.slack.com>

Our TA is Montana Fowler

They will lead the lab sections starting next week

This Class

What we will cover in this class:

- Overview of game engine architecture,
- Introduction to Unreal,
- How to write a ray tracer,
- Overview of real-time ray tracing,
- Investigation of contemporary rendering techniques,
- Mathematics for rendering

What needs to happen to make a game playable, innovative, engaging?

This Class

- Recognition that games curriculum is not currently providing sufficient exposure to Unreal Engine and C++, which the games industry has identified as important skills
- Game Engines are increasingly complex, makes more sense to work with an existing engine rather than build one. We will investigate the main components of game engines, focusing mainly on graphics.
- Advances in machine learning techniques enable new real-time techniques (e.g., real-time ray tracing)
- Aligns with other games and graphics courses at UCSC
 - Game AI, Game Audio, Capstone Sequence, Game Graphics, Computer Graphics (which use WebGL or Unity)

This Class

“Jobs requiring Unreal Engine skills offer an average salary that is \$14,354 above those that do not.”

“Real-time technology skills are the fastest rising career segment of 3D graphics—Growing 601% faster than the market.”

Unity vs. UE?

Built with Unreal Engine:

Fortnite, Gears of War, Bioshock, Mass Effect, Batman: Arkham Asylum, Deus Ex, Splinter Cell, Mortal Kombat X ...

Lots of debate about Unity vs. Unreal vs. other game engines (CryEngine, Godot). General consensus is that it's good to know both, as they each have their pros and cons.

- Unreal seen as better for "professional" games, easier to make multiplayer games, photorealistic graphics, etc.
- Unity seen as better for mobile platforms, VR prototypes, puzzle games, etc.

(But many features are available across both engines.)

Game Engines

What is a game engine? What are the different components of an engine? Why learn how to use one?

Game Engines

Include a wide range of interconnected components:

Rendering,
Physics and collision detection,
Resource/Asset Management,
Inputs/Controllers,
Character animation,
VFX/Particle systems,
Level design, World building, Authoring tools
Audio/Sound effects,
Artificial intelligence/NPCs,
VR/AR,
Networking/Multiplayer

Game Engines

Trade off between *generality* and *genericness*.

- Specialized games can modify some aspect of the game engine: How things look or sound, What interactions are possible, What is emphasized in the virtual space (physics, collisions), How players communicate, How agents behave, etc.
- All of this can be customized (and optimized), leading to specialized engines created for a specific game, or generic engines that are finely tuned for a particular genre of games.
- Game engines are complex, but their goal is to mitigate complexity. Game engines aim to make it easy to manage common aspects of creating games, and *possible* to customize various parts of the game (through scripting, modifying the source code, swapping out components)

Real-time Revolution?

As GPUs become increasingly powerful, game engines can be used to process visual effects and render entire films in real-time.

“Rebirth” (https://www.youtube.com/watch?v=offK9dz_huU)

“Troll” (<https://www.youtube.com/watch?v=3wUVLYKPB6g>)

“Adam” (<https://www.youtube.com/watch?v=D7Jrxj4Uc-o>)

“Book of the Dead” (https://www.youtube.com/watch?v=DDsRfbfnC_A)

“Reflections” (<https://www.youtube.com/watch?v=J3ue35ago3Y>)

Ray tracing

Ray tracing models the behavior of photons bouncing around a scene, which is computationally expensive.

- We will build a (non-realtime) ray tracer as one of the assignments
- Explore new techniques in realtime ray tracing.

Grading

Homeworks (30%):

Reading, Video tutorials

Ray-tracer in C++

Assignments in using Unreal and HLSL

Investigating different aspects of game engines

Project (50%):

You will implement a technique of your choice using the Unreal Game Engine, and describe in detail the how the game engine executes the technique.

For Monday

Download and explore Unreal:

- Make sure you download and test Unreal Engine on your laptop or a lab computer: <https://www.unrealengine.com/en-US/> (the stable version is "4.23"). UE should work on any recent laptop or desktop, but runs better with a fast GPU. You'll need to create an account on Epic in order to download UE and access documentation and video tutorials, etc.
- Follow the introductory tutorials at:
<https://learn.unrealengine.com/courses/library>
- Get familiar with the Unreal Docs:
<https://docs.unrealengine.com/en-US/index.html>

For Monday

Homework 1:

- Investigate interesting topics and features available in Unreal Engine by watching videos from SIGGRAPH, GDC, or other conferences
- Start the introductory Learning Paths on "Getting Started in Unreal Engine" and "Introducing the Principles of Real-Time Rendering"