Welcome!

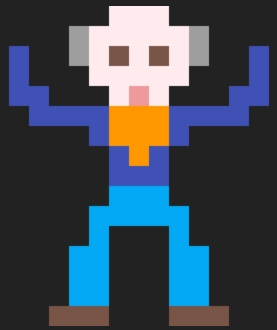
While you watch this video...

Windows: Install Visual Studio with C++

Mac: Install Xcode

Carmine T. Guida

cguida@nyu.edu



It’s in the…

Syllabus

What is a game?

“a game is the voluntary attempt to overcome unnecessary obstacles”

- Bernard Suits (The Grasshopper)

“a game is a series of interesting decisions”

- Sid Meier (Creator of Civilization)

What is Game Programming?

The types of games we are making:

What are some common things video games have?

Graphics
Input
Game Logic
Audio

Physics, AI, UI

The heart of Game Programming:

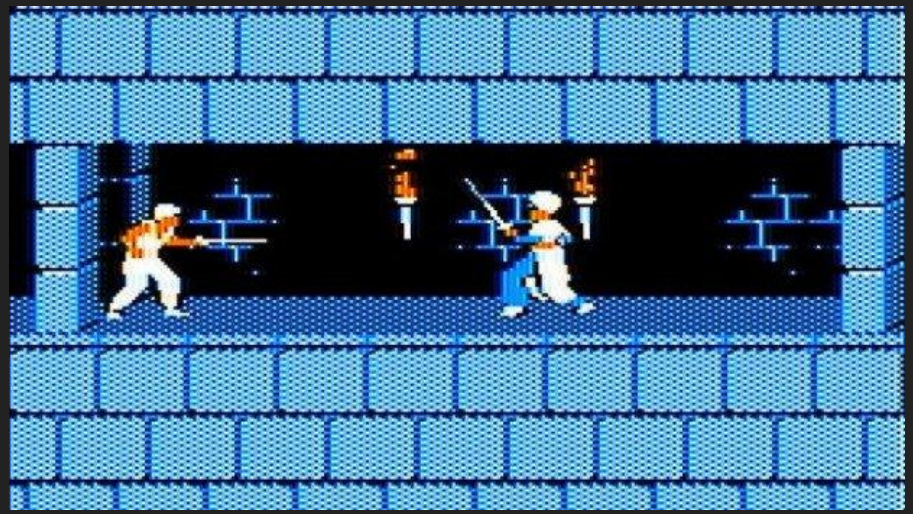
```
Startup();

while (gameIsRunning) {
    ProcessInput();
    Update();
    Render();
}

Shutdown();
```

A brief history of Video Game Programming

“This archive contains the source code for the original Prince of Persia game that I wrote on the Apple II, in 6502 assembly language, between 1985-89.

The game was first released by Broderbund Software in 1989, and is part of the ongoing Ubisoft game franchise.”

- Jordan Mechner



https://github.com/jmechner/Prince-of-Persia-Apple-II

Back in the day...

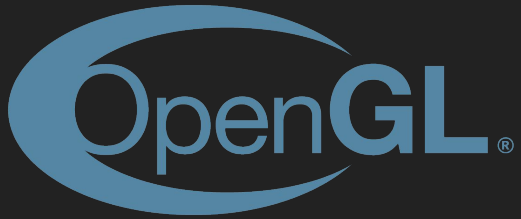
OpenGL and DirectX

(1992) (1995)

Game Engines

(We're not using these.)

This course:



What is SDL?

Layer on top of OpenGL

Window Management, Input, Events, Audio, Game Controllers and more.
(OpenGL is a Graphics Library and does not provide these things!)

Mac OS, Windows, Linux, iOS, Android

SDL in the real world!

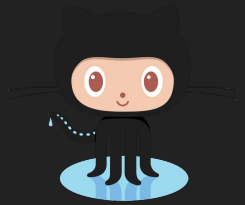
https://en.wikipedia.org/wiki/Source_(game_engine)

“The first of Valve's games to support Linux was Team Fortress 2, the port released in October 2012 along with the closed beta of the Linux version of Steam. Both the OS X and Linux ports of the engine take advantage of OpenGL and are powered by SDL.”

Dota 2, Half-Life, FTL: Faster Than Light, VVVVVV
Linux version of Unreal Tournament, also popular with Emulators

Let’s get you ready to code!

git and github



For each project, you will submit a link to your github repository.

If you have not already done so…

Setup an account at github: https://github.com
Install git: https://git-scm.com
Learn how to use git (Create a repository, push code to github): https://try.github.io

Course Resources

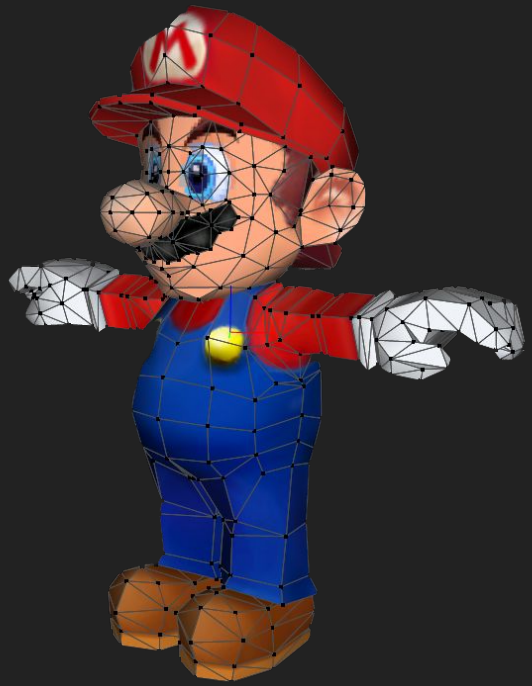
Resources, libraries, assets and lecture slides are available in the following GitHub repository. Note that lecture slides and project requirements may be delivered as we go:

https://github.com/carmineguida/CS3113

Let’s Review the Windows and Mac Setup Instructions!

How did we get to where we are today?


BAMF



Hardware (Specialized) → Software Rendering → Hardware Rendering

Hardware (Specialized)

The 2600 did not use a frame buffer. Instead the video device used two 8-pixel bitmapped sprites, two 1-pixel "missile" sprites, a 1-pixel "ball", and a 40-pixel "playfield" that is drawn by writing a bit pattern for each line into a register just before the television scans that line. As each line is scanned, a game must identify the non-sprite objects that overlap the next line, assemble the appropriate bit patterns to draw for those objects, and write the pattern into the register.

Racing the Beam
The Atari Video Computer System is a book by Ian Bogost and Nick Montfort

https://en.wikipedia.org/wiki/Atari_2600

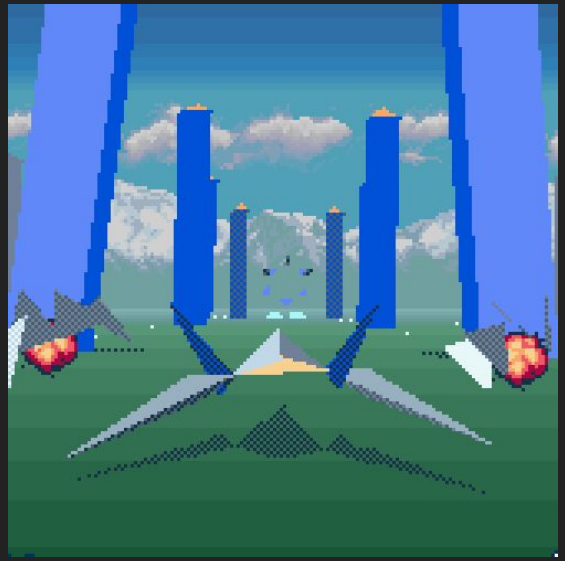
The NES uses a custom-made Picture Processing Unit (PPU) developed by Ricoh. All variations of the PPU feature 2 kB of video RAM, 256 bytes of on-die "object attribute memory" (OAM) to store the positions, colors, and tile indices of up to 64 sprites on the screen, and 28 bytes of on-die palette RAM to allow selection of background and sprite colors. The console's 2 kB of onboard RAM may be used for tile maps and attributes on the NES board and 8 kB of tile pattern ROM or RAM may be included on a cartridge.

https://en.wikipedia.org/wiki/Nintendo_Entertainment_System


SUPER NINTENDO
SELECT
START
POWER
RESET
EJECT

Mode 7

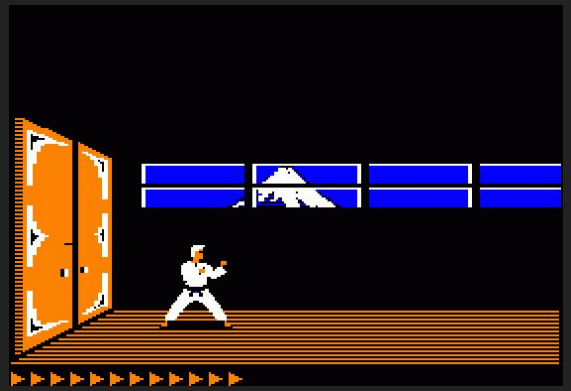

SCORE 000051640
PLAYER
ENEMY
BLOCK 4-3
♥=27 P=05
TIME 386

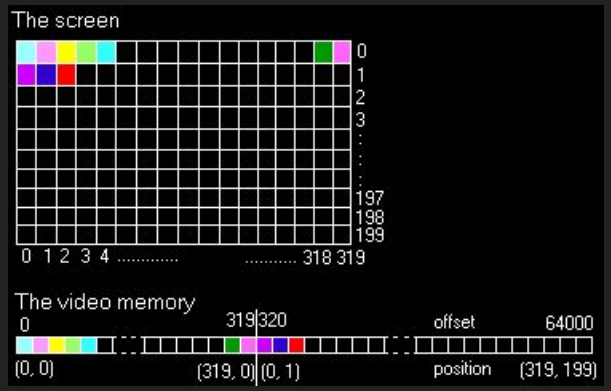


The Super FX Chip (Star Fox and more)
https://www.youtube.com/watch?v=Opzomu6mgYk&start=313&end=462

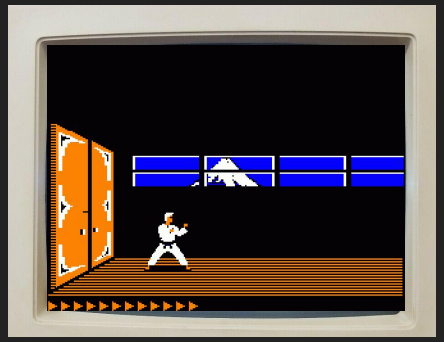
Software Rendering

Software Rendering





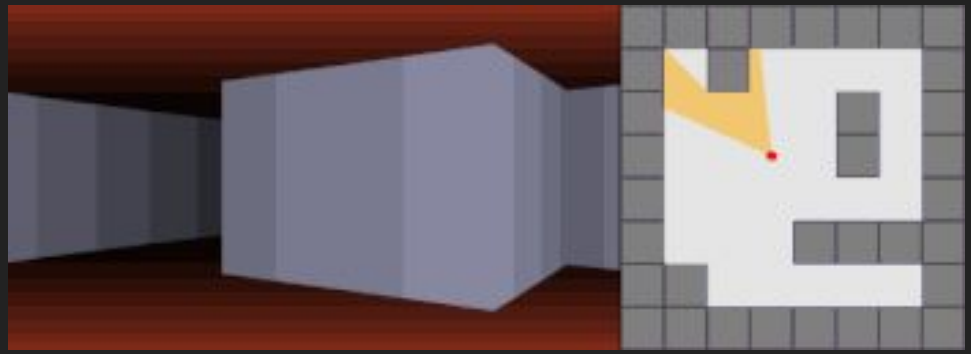
Scene built in RAM



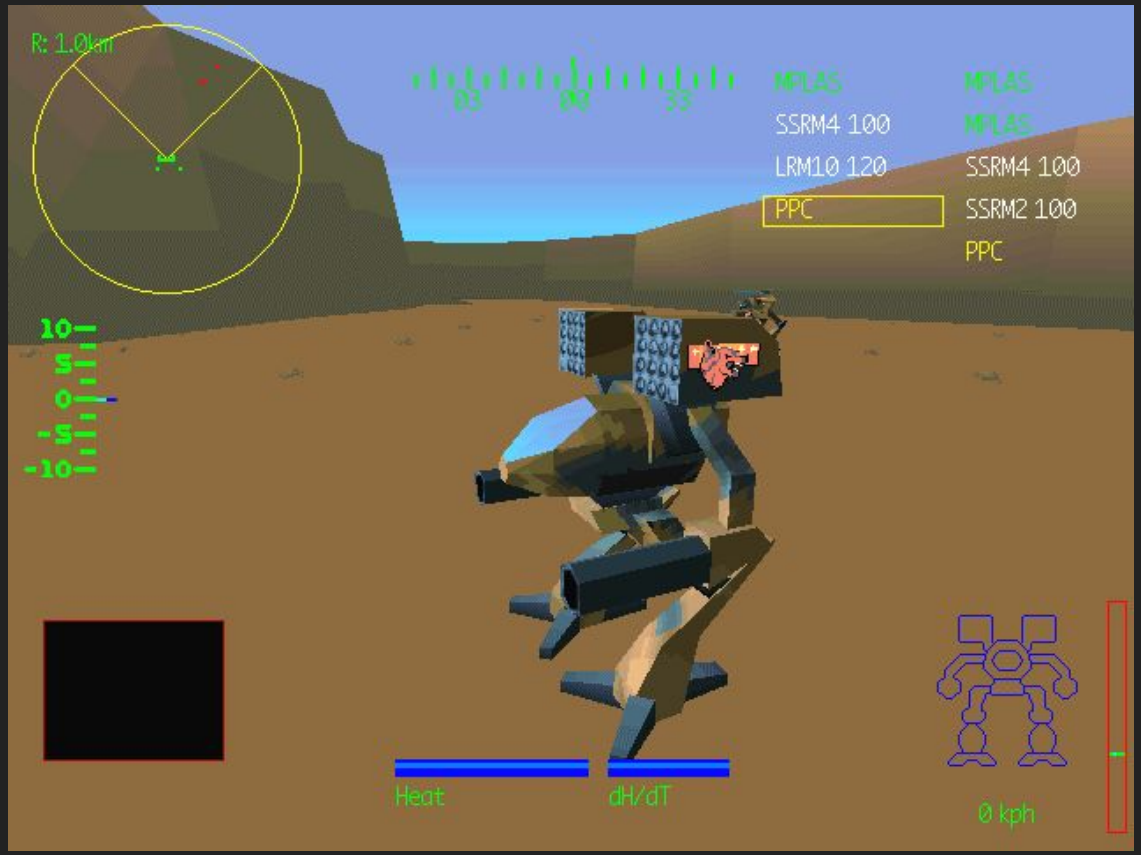
Video Card

Raycasting

Wolfenstein 3D
Doom
Duke Nukem 3D
(and more)



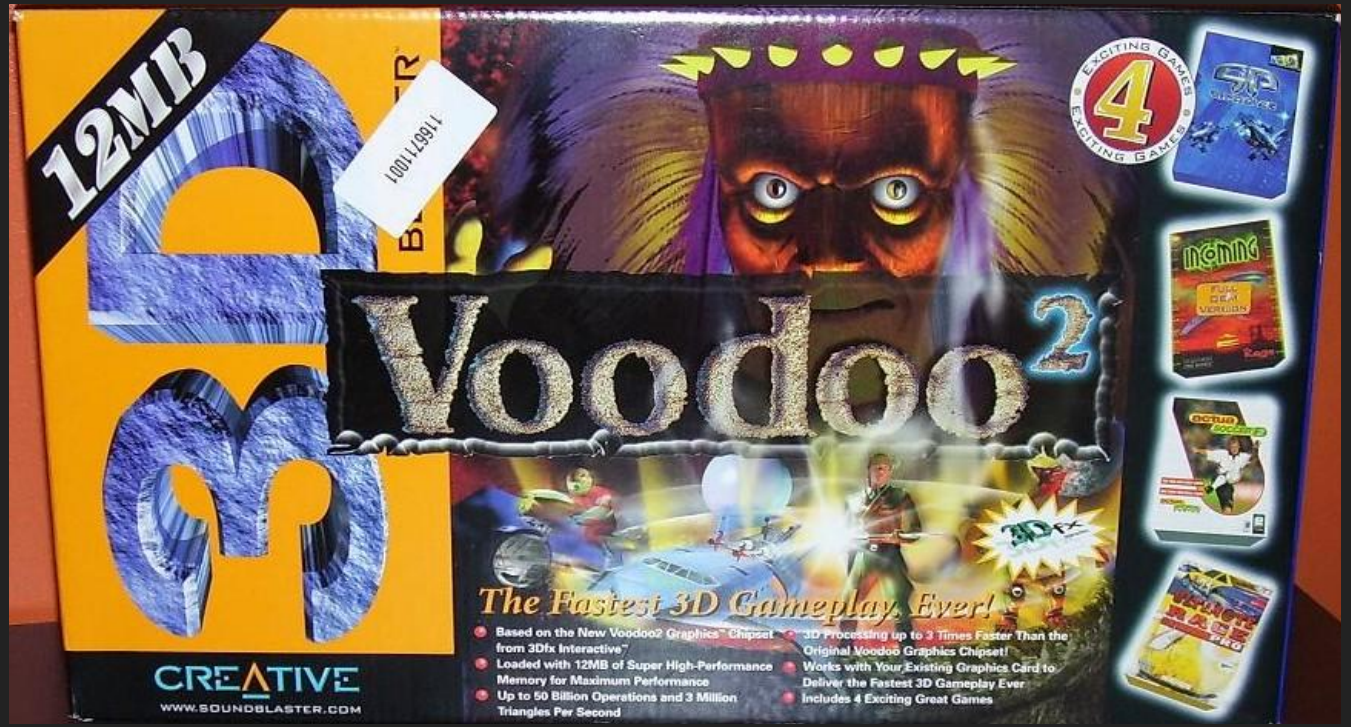
(The predecessor to Elite Dangerous)



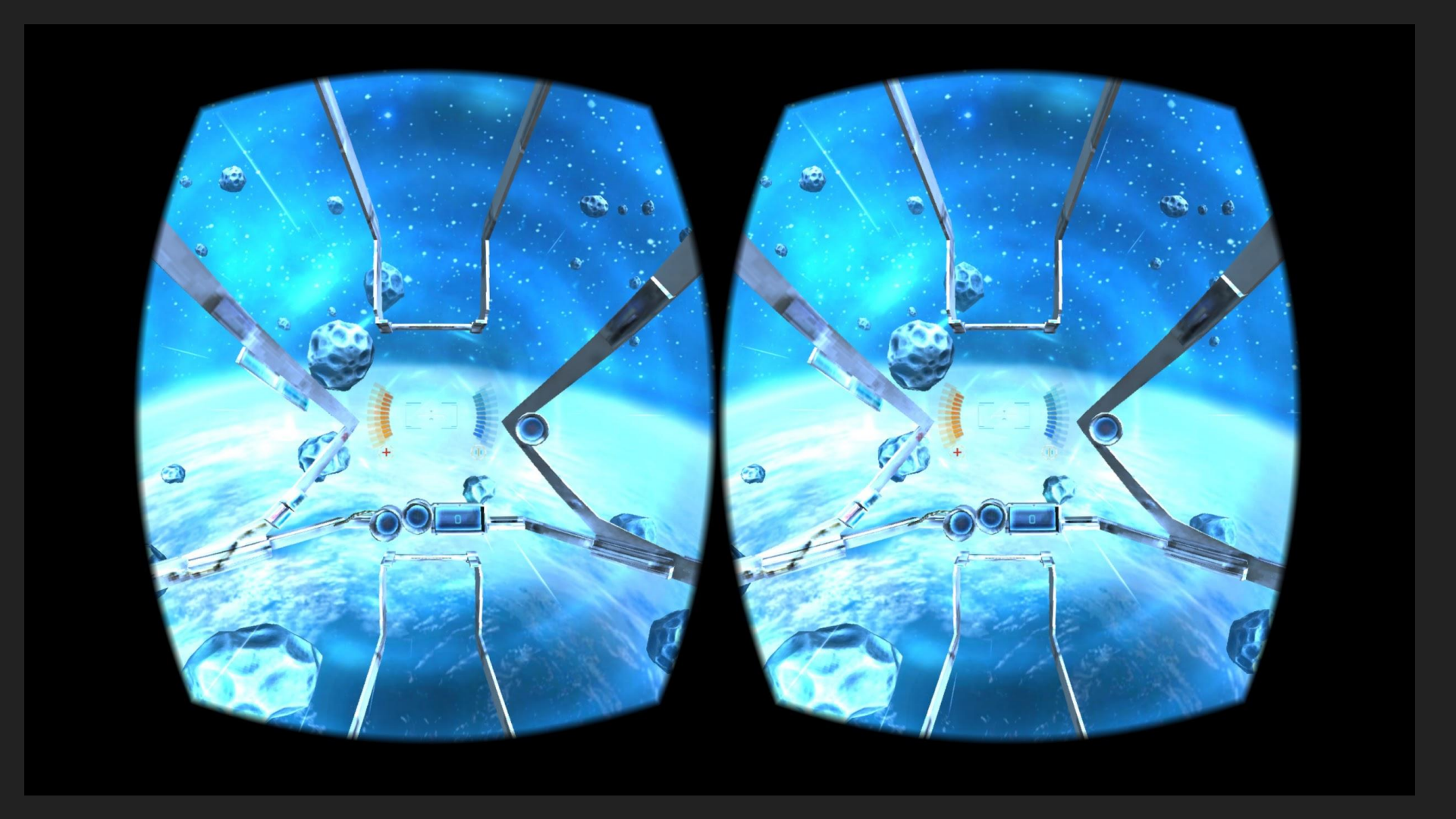
(MechWarrior 2: I lost my mind when I first played this game.)

Hardware Rendering

Graphics Cards!

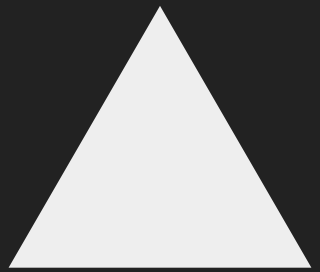



Ordinary VGA Quake
Resolution: 320x200
Colors: 256
Frame-rate: 30fps
OpenGL Quake on 3Dfx
Resolution: 640x480
Colors: 65,536
Frame-rate: 30fps
Created by Mark D. Rejhon - www.marky.com

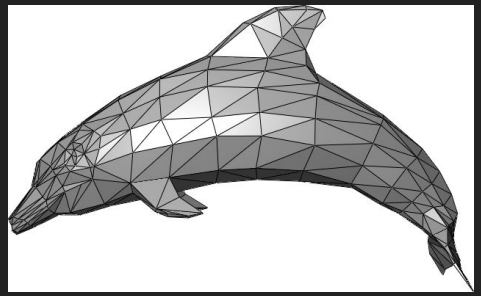


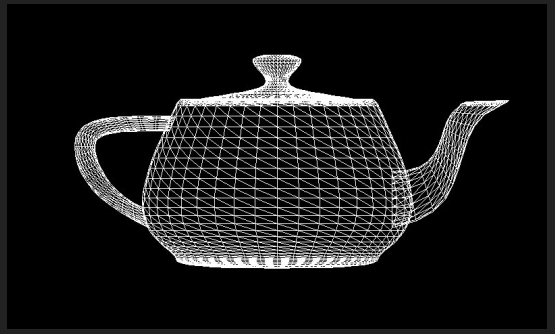
How does this all work?

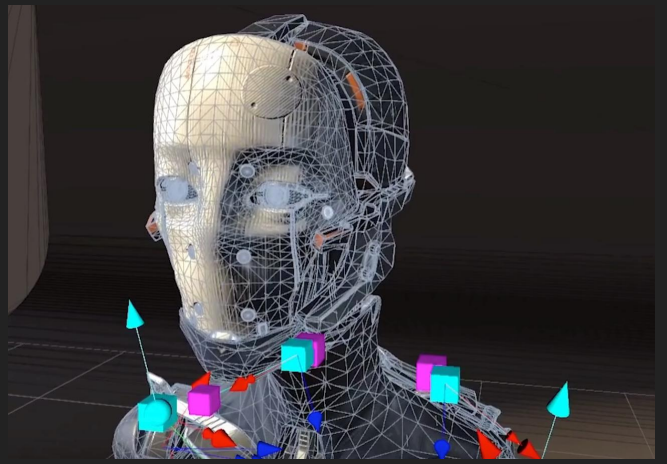
Triangles!

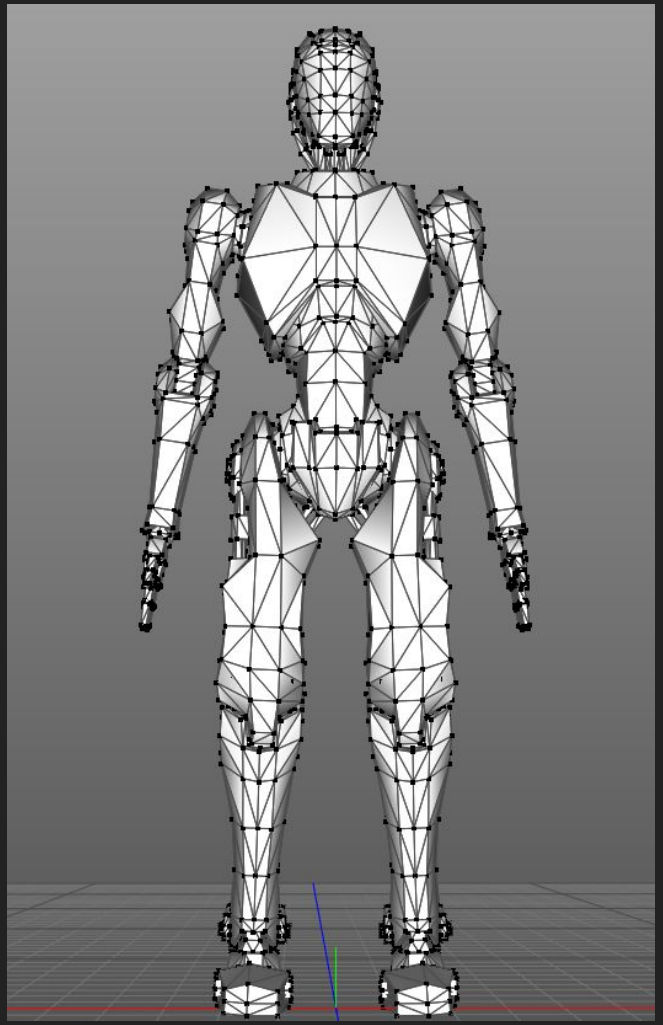


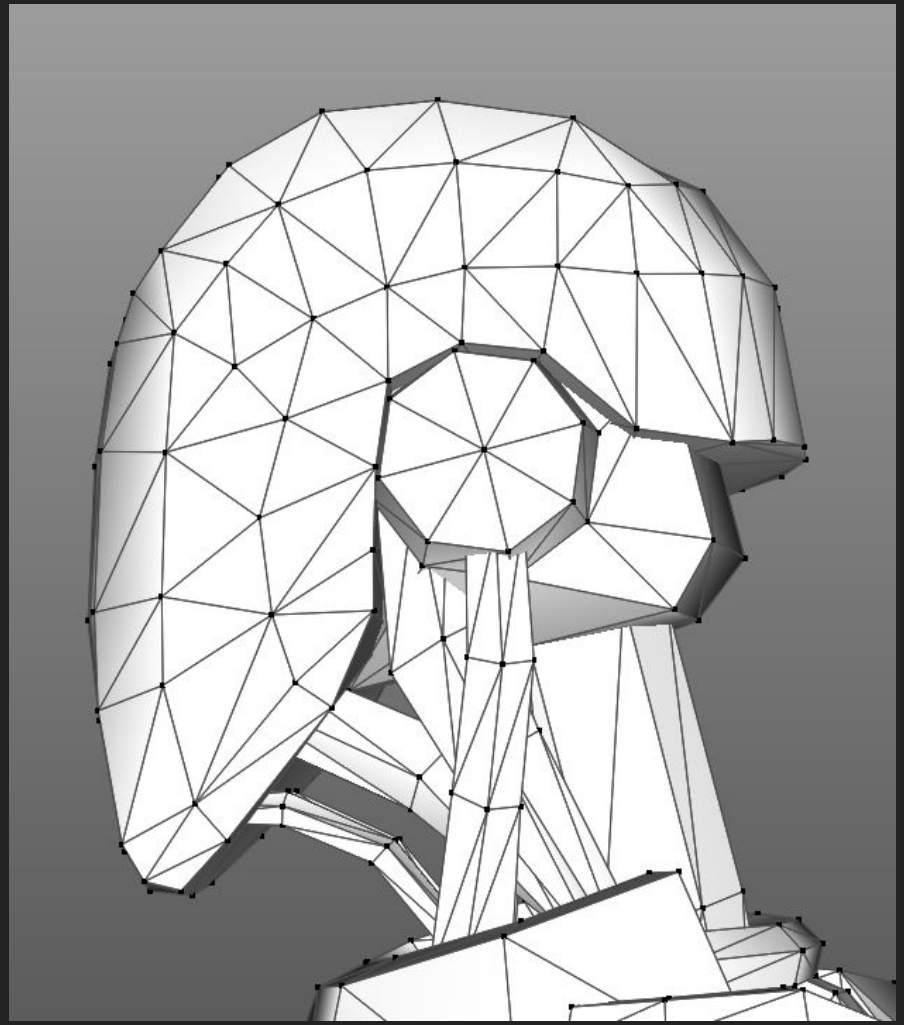
Lots of Triangles!



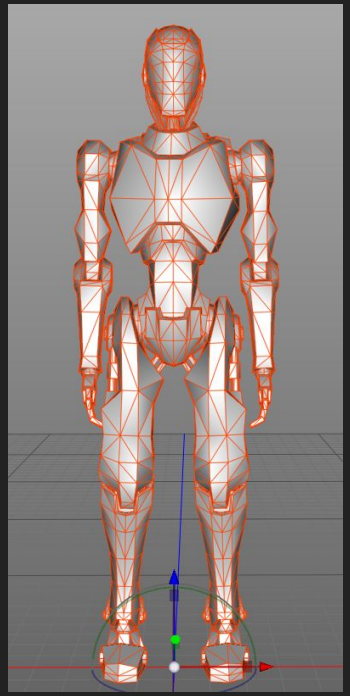


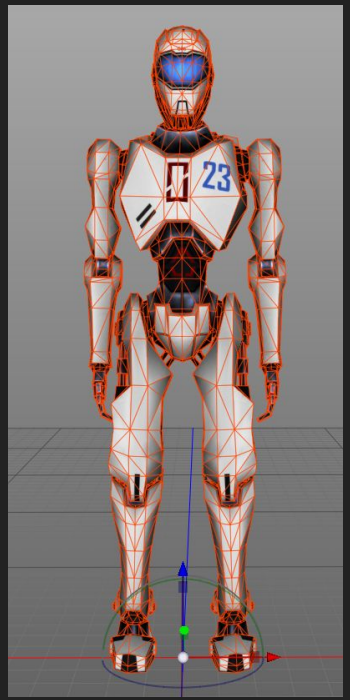


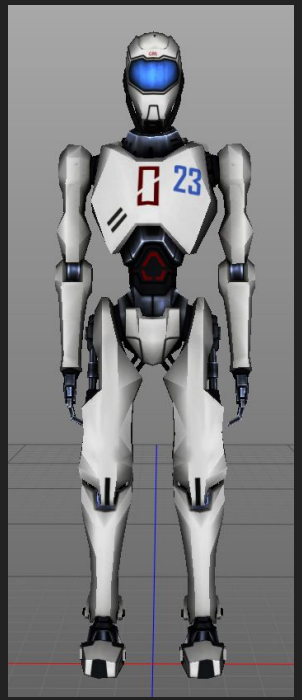




...and Textures



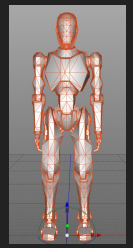


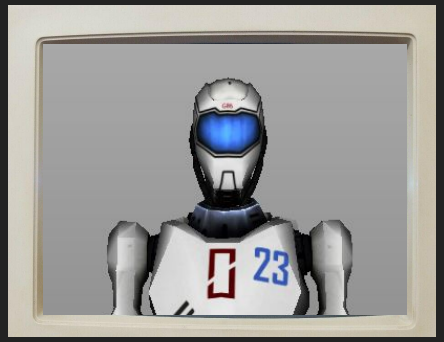


Hardware Rendering

Load assets to Graphics Card RAM

Run Game Logic
Tell Graphics Card what to draw



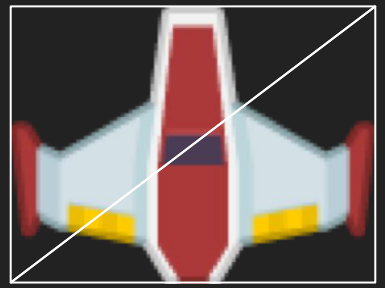


CPU + RAM → Graphics Card → Display

2D is really 3D

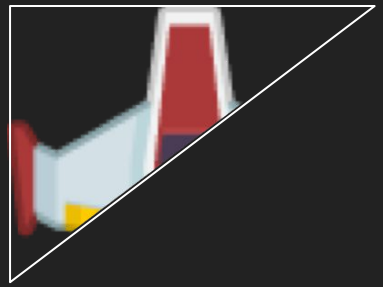
(in modern times)

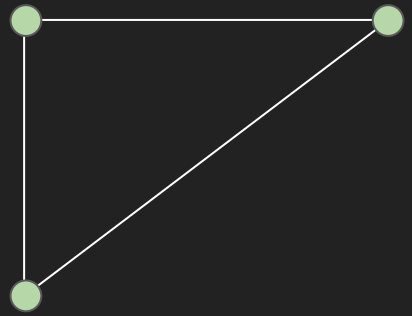
2D Sprite Made of 2 Triangles



A Triangle Has 3 Vertices

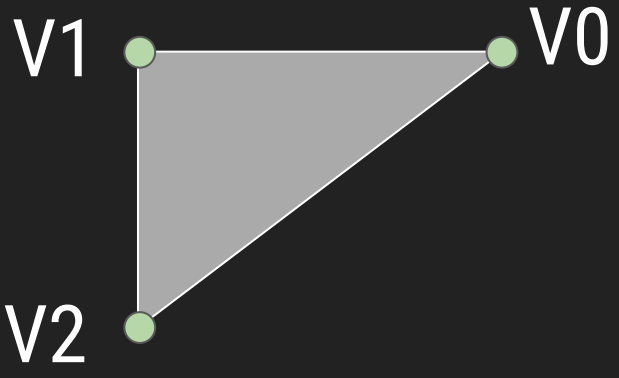
(each point is a vertex)



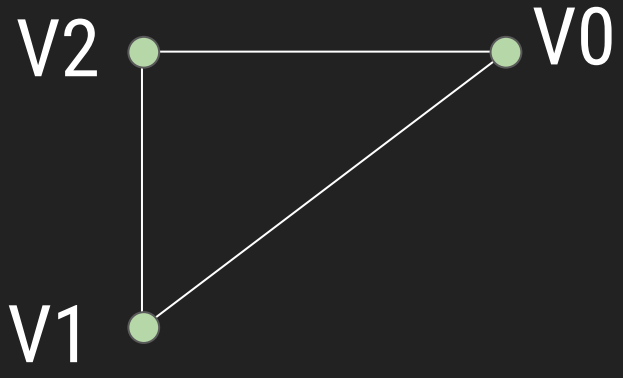


Triangles are 1 sided!

(the side is determined by order of the vertices)



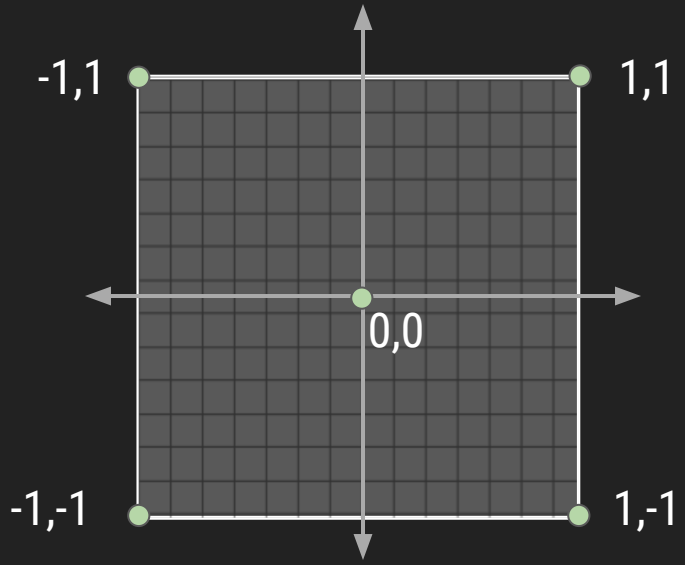
Counter-Clockwise
(Front Facing)



Clockwise
(Back Facing)

Texture Coordinates

Model Coordinates vs Texture Coordinates



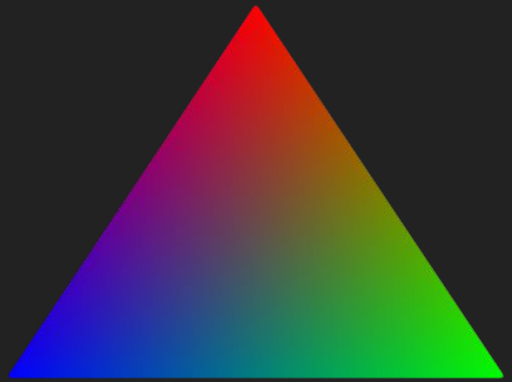
Shaders!

Vertex Shader

Translates vertices to screen positions.

Fragment Shader

For each pixel, determines what color to draw.
Interpolated by distance from vertices.
(might also grab a pixel/color from a texture)



Let’s draw a triangle!