Game Development In JavaScript

First Impressions

Javascript

- Designed for web application
- Had a rough childhood
- Very misunderstood
- Performance difference across browsers

Benchmark

- Pros/Cons

JavaScript: WebGL & Canvas

Canvas API	WebGL equivalent	Three.js framework equivalent
scale	texImage2D + modify vertices	Camera
rotate	texImage2D + modify vertices	Camera
translate	texImage2D + modify vertices	Camera
transform	texImage2D + modify vertices	Camera
clearRect	clear+viewport, or teximage2d	Textures
fillRect	clear+viewport, or teximage2d	Textures
strokeRect	teximage2d	Textures
path	teximage2d+stencil + vertex paths	Camera + objects
fillText	teximage2d+stencil + vertex paths	Camera + objects + textures
strokeText	teximage2d+stencil + vertex paths	Camera + objects + textures
drawImage	teximage2d	textures
createImageData	fill buffer of teximage2d in CPU application	CPU application code
getImageData	readpixels	readpixels
putImageData	fill buffer of teximage2d in CPU application	fill buffer of teximage2d in CPU application
	+ programmable shaders (vertex, fragment)	
	+ offscreen buffers	
	+ efficient 3D representation of depth	
	+ efficient POINTS rendering	
		+ Convenience methods - Materials (sprite ..)
		+ Convenience methods - Lights
		+ Convenience methods - Scenes

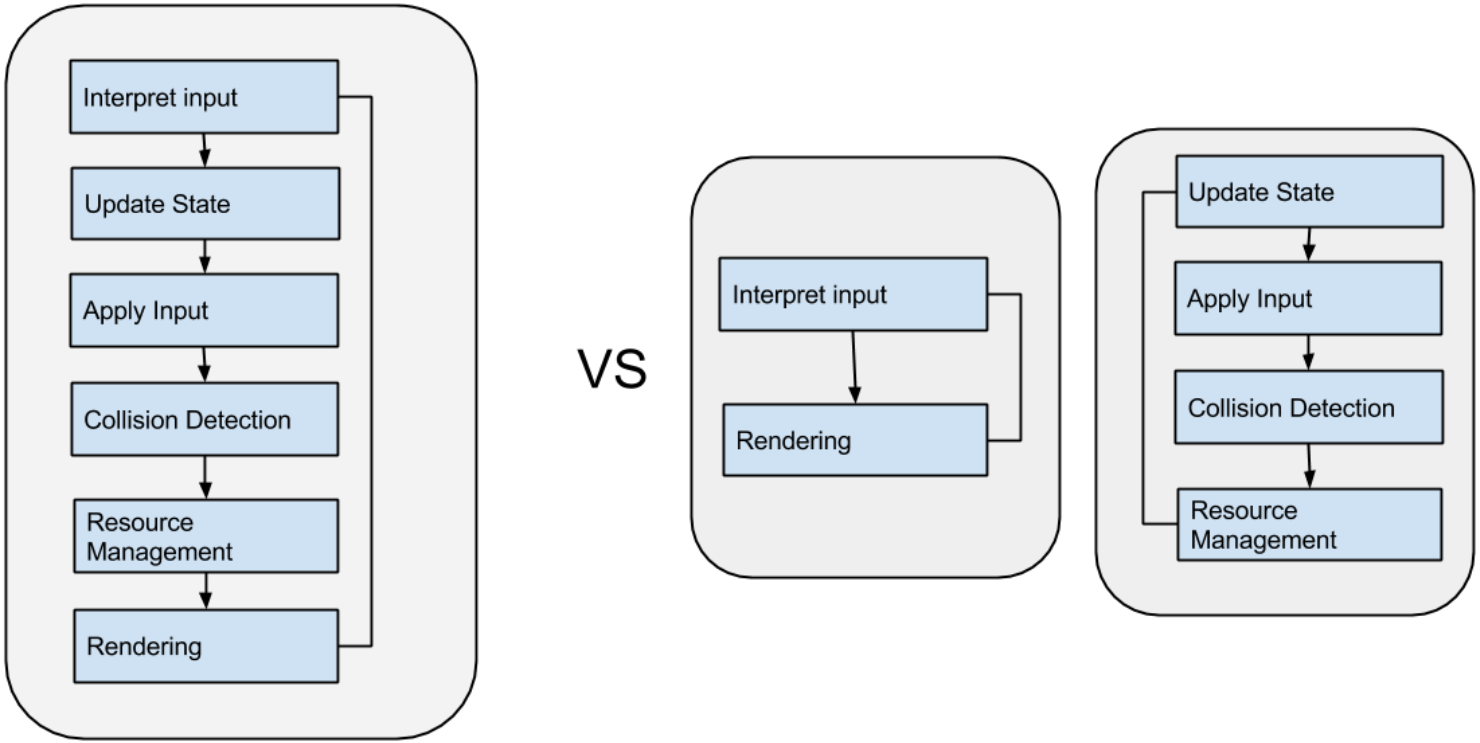
Javacript Graphics libraries

- THREE.js
  - 3D only
  - Performance test
- PIXI.js
  - 2D only
  - Performance test

Google is your friend

- StackOverflow
- Shadertoy
- CGLearn ;)
- Gamedev.net
- TigSource

Lets Talk Game cycles



Worker

- Great if you want to utilize more than 1 core on your CPU
- Completely **seperated** from main context!
- Send data as messages.
- Benchmark

Memory management in Javascript

- Scopes and vars

a = “such Waste”

var a = “such Efficient”

// Much wow!

- Delete Keyword
- Require JS + Object Oriented approach = 

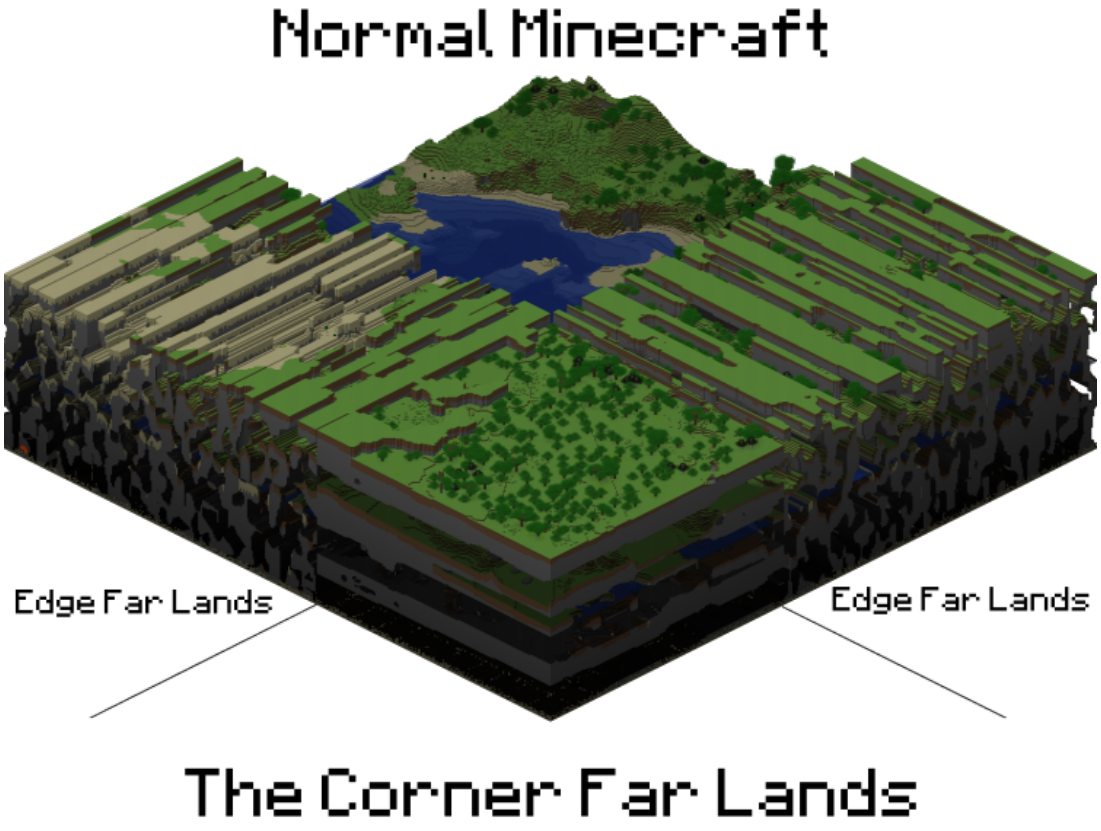
Environments

- Linear Environments (Half Life 2, Duke Nukem)
- Zoned Environments (Guild Wars 2, Age of Conan)
- Streamed Environments (Skyrim, World of Warcraft, Gothic 3, GTA 5, Minecraft)

Example: Minecraft World

- World is defined by Chunks.
- Seed is used to generate Chunks.
- Chunks are loaded and unloaded from the “scene“ as player travels around.
- Unloaded Chunks are not deleted

Example: Minecraft World - Farlands



Dynamics

Games are more interesting if objects within the game can interact.

- Interactions
- Physics

All such activities require us to know something about proximity

Physics Engines/Libraries

- Cannon
- Ammo
- Box2D
- PhysiJS

Collision Detection

Separated into two phases:

- Broad phase
- Narrow phase

Collision Detection: Broad Phase

Find potential collision candidates in space.
Problem?

Methods to partition space

- None
- Grid
- Tree structures (example)

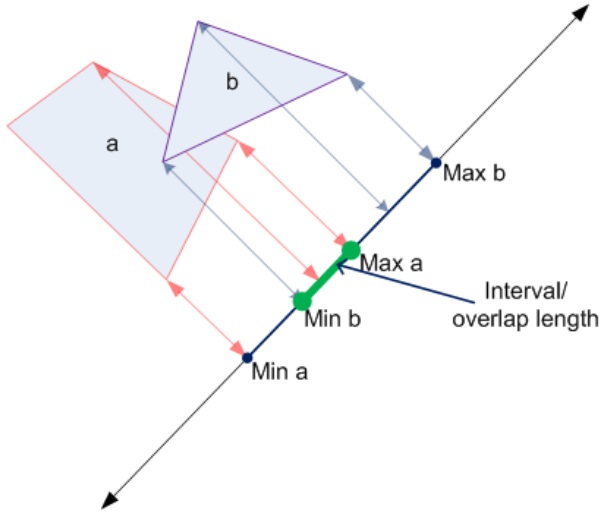
Collision Detection: Narrow Phase

Using Mesh for collision testing is expensive, lets use simplification instead.

Collision Detection: Narrow Phase

Object to Object collisions testing

- Separated Axis Theorem
- Line segment intersection



Collision Detection: False Negatives/Positives