Graphics Programming

Evolution of Graphics

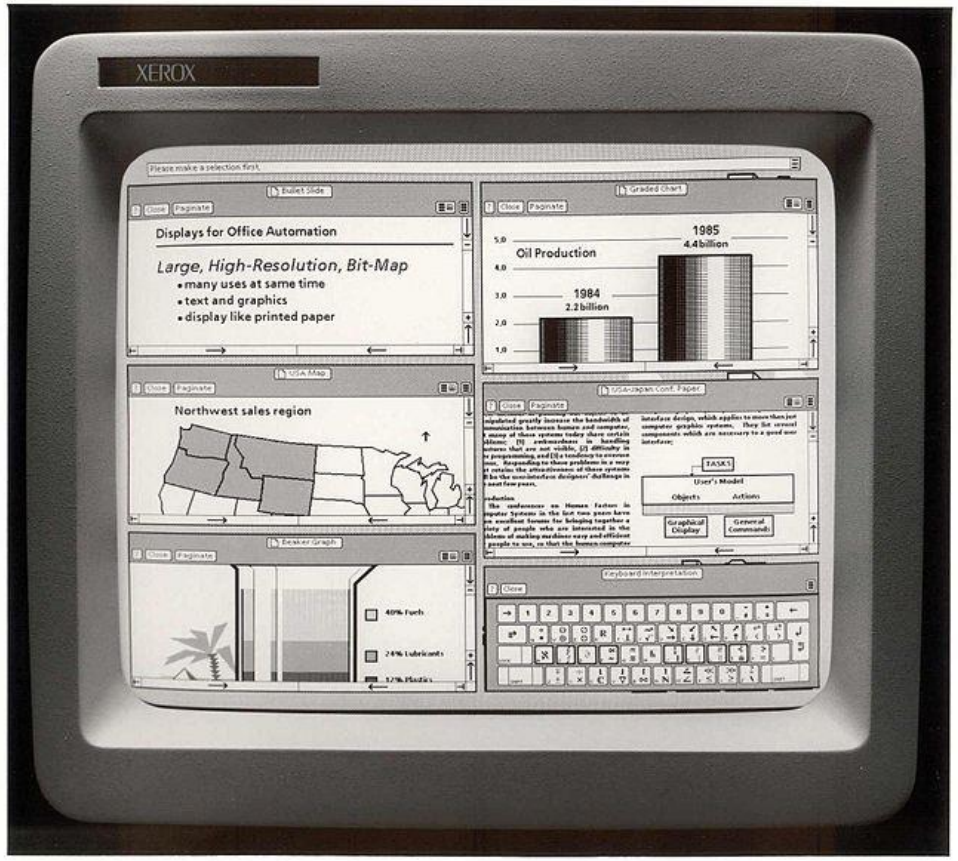
Early GUIs

Draw data to a computer terminal

Foundation of visual operating systems

Color complexity has grown over time:

- 1-bit (black and white)
- 2-bit (4 colors)
- 4-bit (16 colors)
- 8-bit (256 colors)
- 24-bit (16,777,216 colors)



Color Depth Visualized

24-bit - 16M Colors

8-bit - 256 Colors

4-bit - 16 Colors

The Start of Computer Gaming

With the advent of GUIs came the start of video games

Spacewar! (pictured) is credited as being one of the earliest video games

Converts user input into a behavior and changes the program accordingly

Updates the display continuously

Video Game Graphics

Graphics improvements allow for rendering objects in video games at near photo-realistic qualities

Improvements in modern GPUs allow for processing and rendering of graphics to be done in parallel, resulting in more detailed imagery

For VR, photorealism is one part of immersion experience

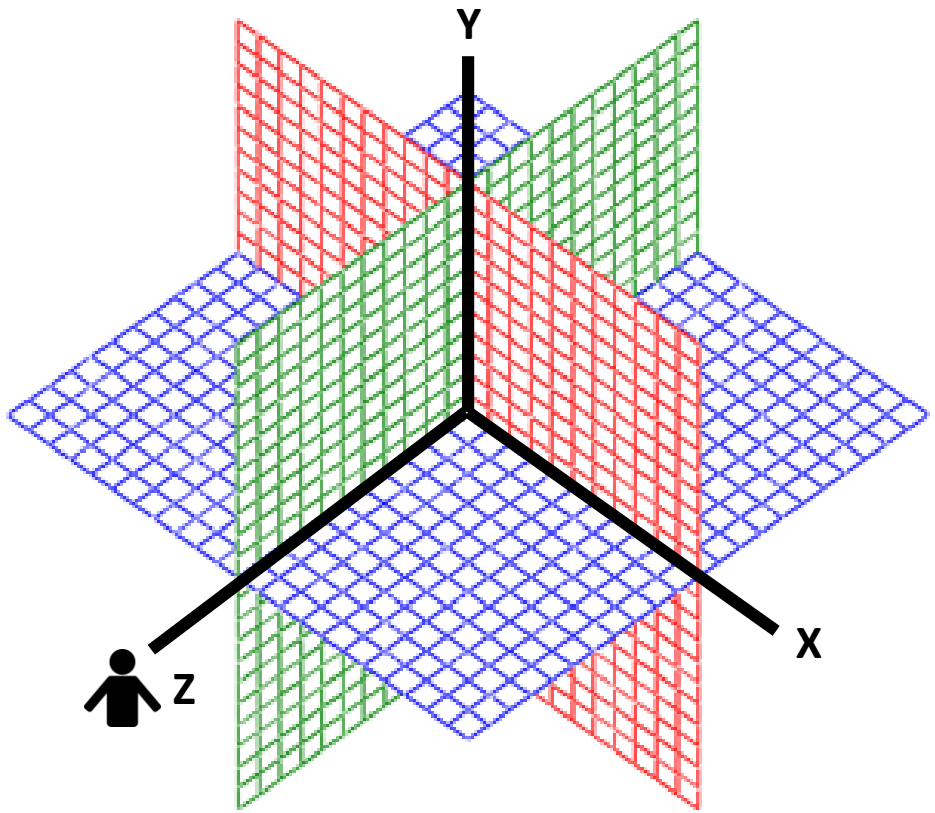
3D Scene Design

The 3D Coordinate System

Objects within a scene are placed a specific location with an X, Y, and Z value

Each object also has X, Y, and Z properties related to the size of the object: width, height, and depth

For this course, we will talk about Y being the positive upward direction, X being to a viewer's left or right, and Z being distance from the viewer

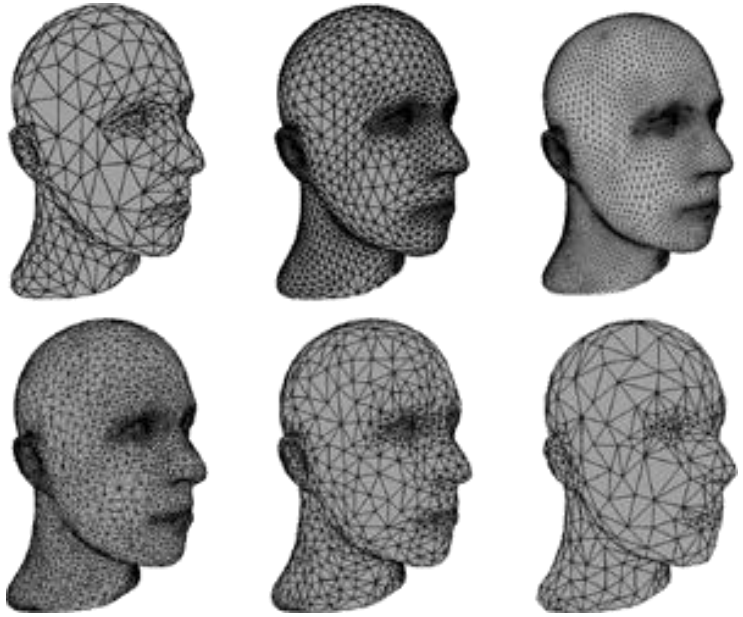


Scene Objects

Types of potential objects in a scene:

- Character / Camera
- Lights
- Props
- Enemies / Non-Player Characters
- Environment (Terrain) objects
- User interfaces
- Whatever you can imagine!

Detailed Mesh Objects



Graphics cards generally render meshes as a series of triangles

More triangles = more detail = more processing power needed

Lighting & Shading

Lighting is a great way to add realism to your scene and give objects more depth to their appearance

More lights = more processing and less performance

Different types of lights give different effects

Reflections, shading, and shadows are generally calculated automatically with modern game engines

Different materials on objects will have different shading effects

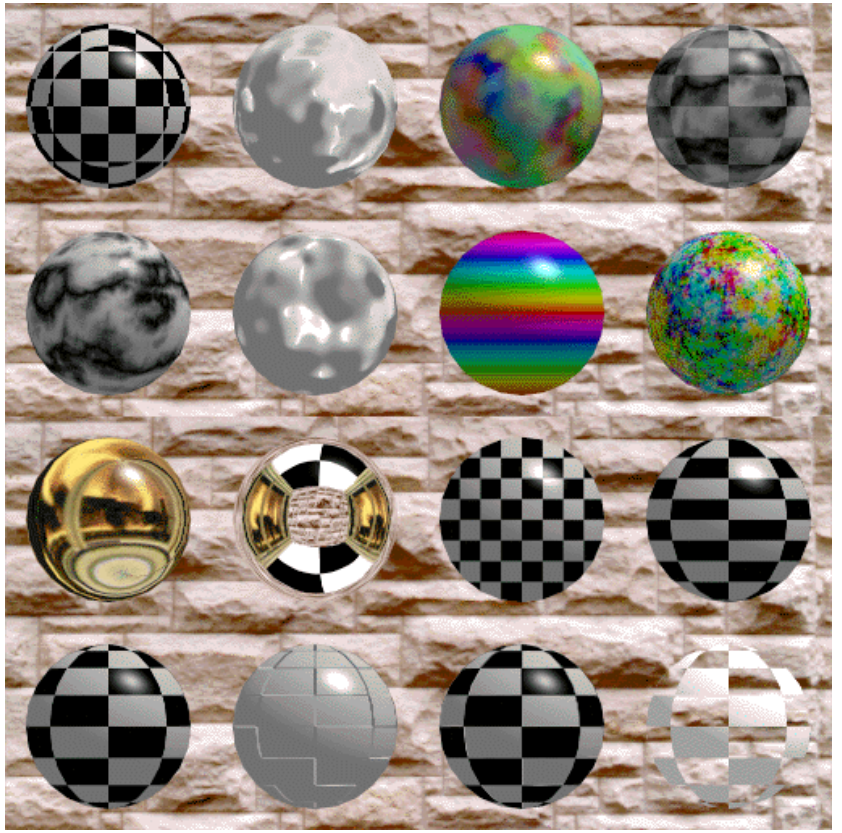

Flat
Gouraud
Phong

Materials & Textures

Applying different materials to objects can allow you them to take on more detailed appearances without using complex mesh geometry

Textures are materials that can include information about how to render the object

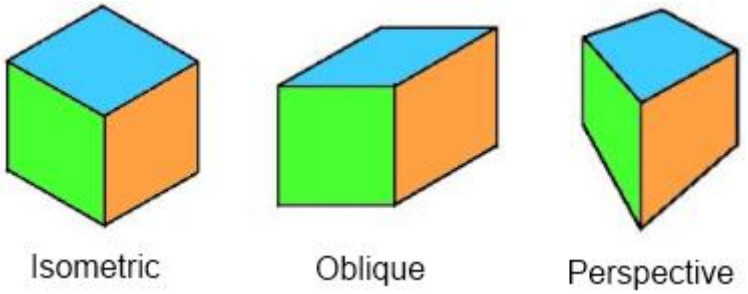
- Example: Rocks may have a bumpy surface applied to them to give the appearance of not being smooth

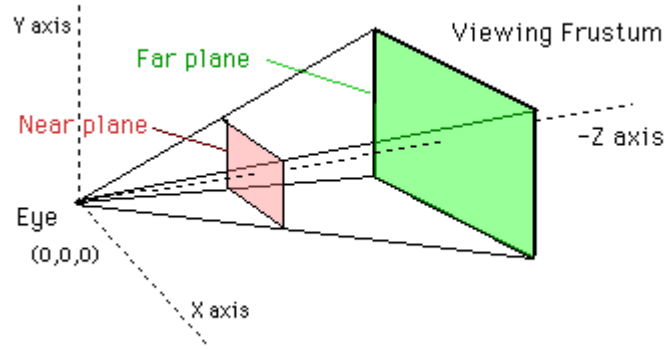


Cameras

Determine what part of the scene is in view

Can change effects based on the type of *projection* that is set up





Characters

PLAYER CHARACTER

Controlled by the person in the game

Camera follows

First Person: Camera is through the eyes of the character

Third Person: Camera usually sits above looking down on the character

NON-PLAYER CHARACTER

Enemies or Allies

Program behavior through scripts

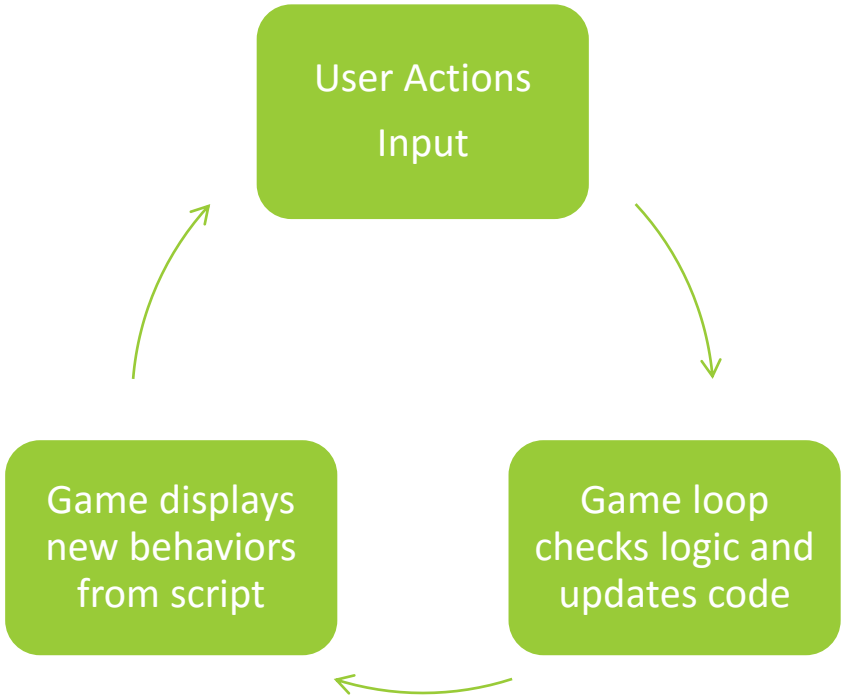
Behaviors may change based on user input

Scripts

Code that tells objects how to act in the game

Game Loop: updates every “clock tick” (frame) and checks to see if the behavior of anything in the game scene should change

Commonly written in C#, UnityScript, Boo (Unity), C++ (Unreal), JavaScript (WebVR) for virtual and augmented reality



User Interfaces

Present the user with interactions and information

Can be part of the in-game experience (such as with a dialog box) or separate from it (such as with a menu or level selection option)

UI Considerations in Virtual Reality

Text is harder to read when it's rendered stereoscopically*

Placement in space might not be convenient depending on where the player is facing

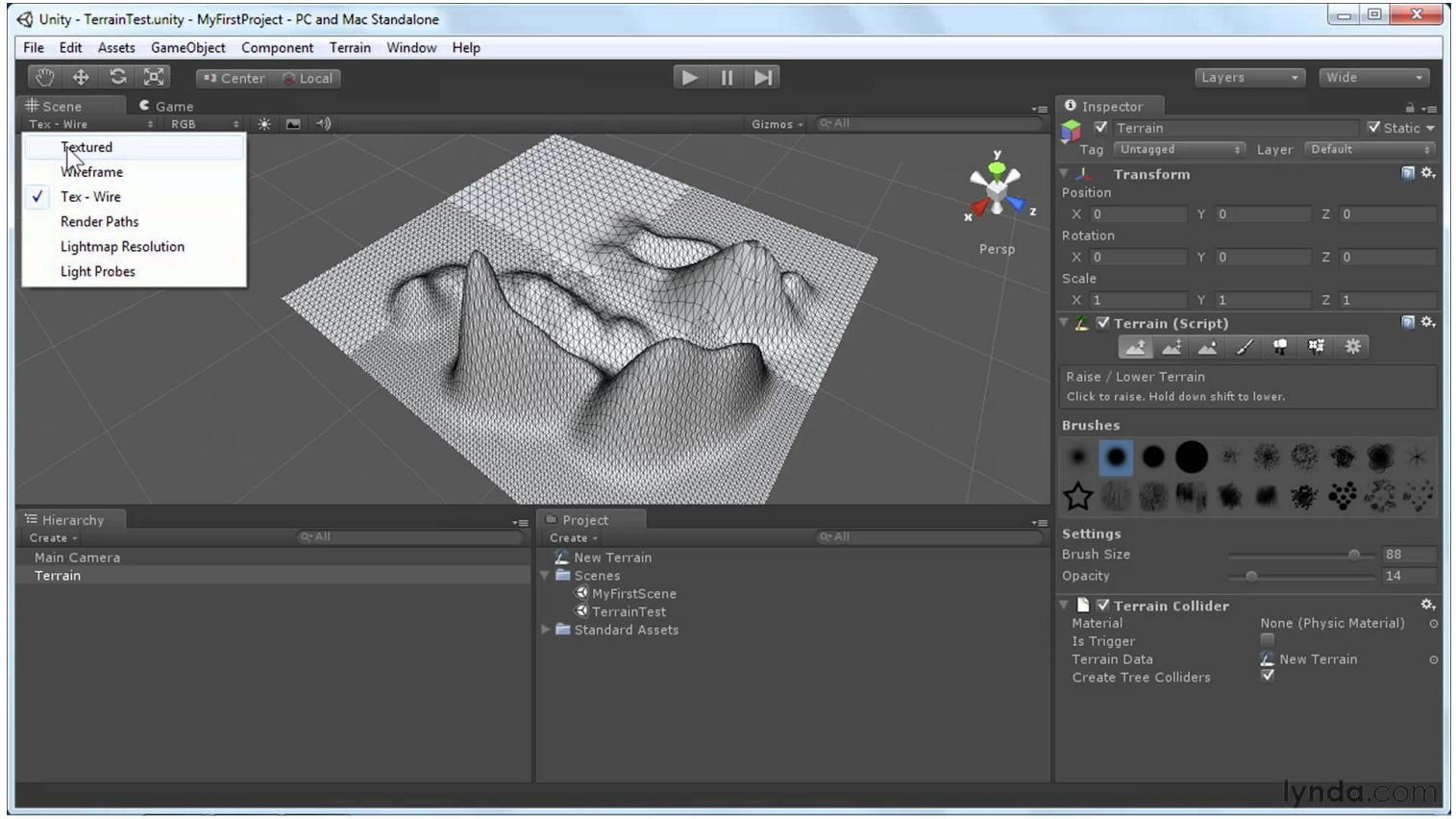
Reading may be uncomfortable depending on what is being displayed

Interactive elements (e.g. buttons) need to support various inputs

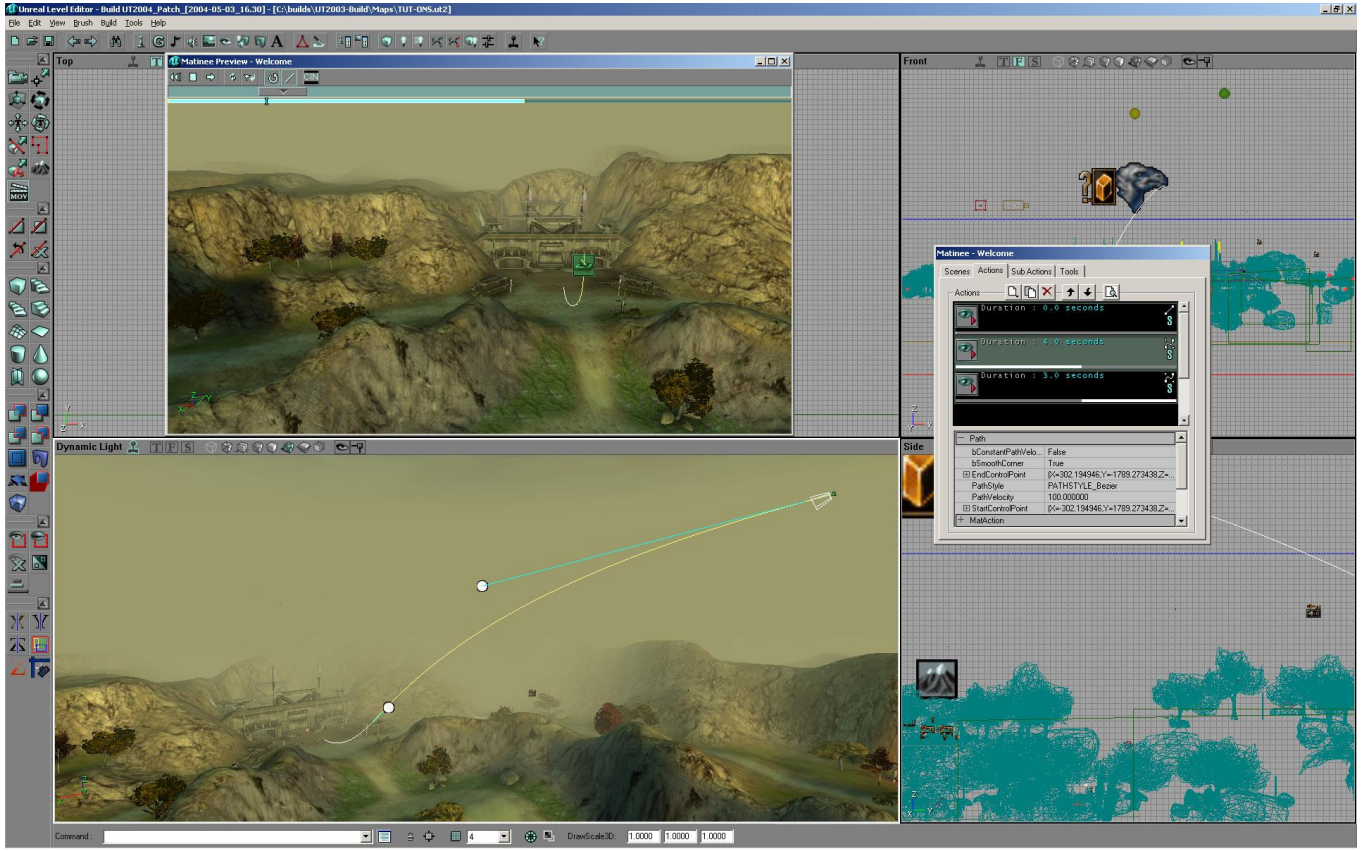
* Green is the easiest color to read in VR

3D Development Tools

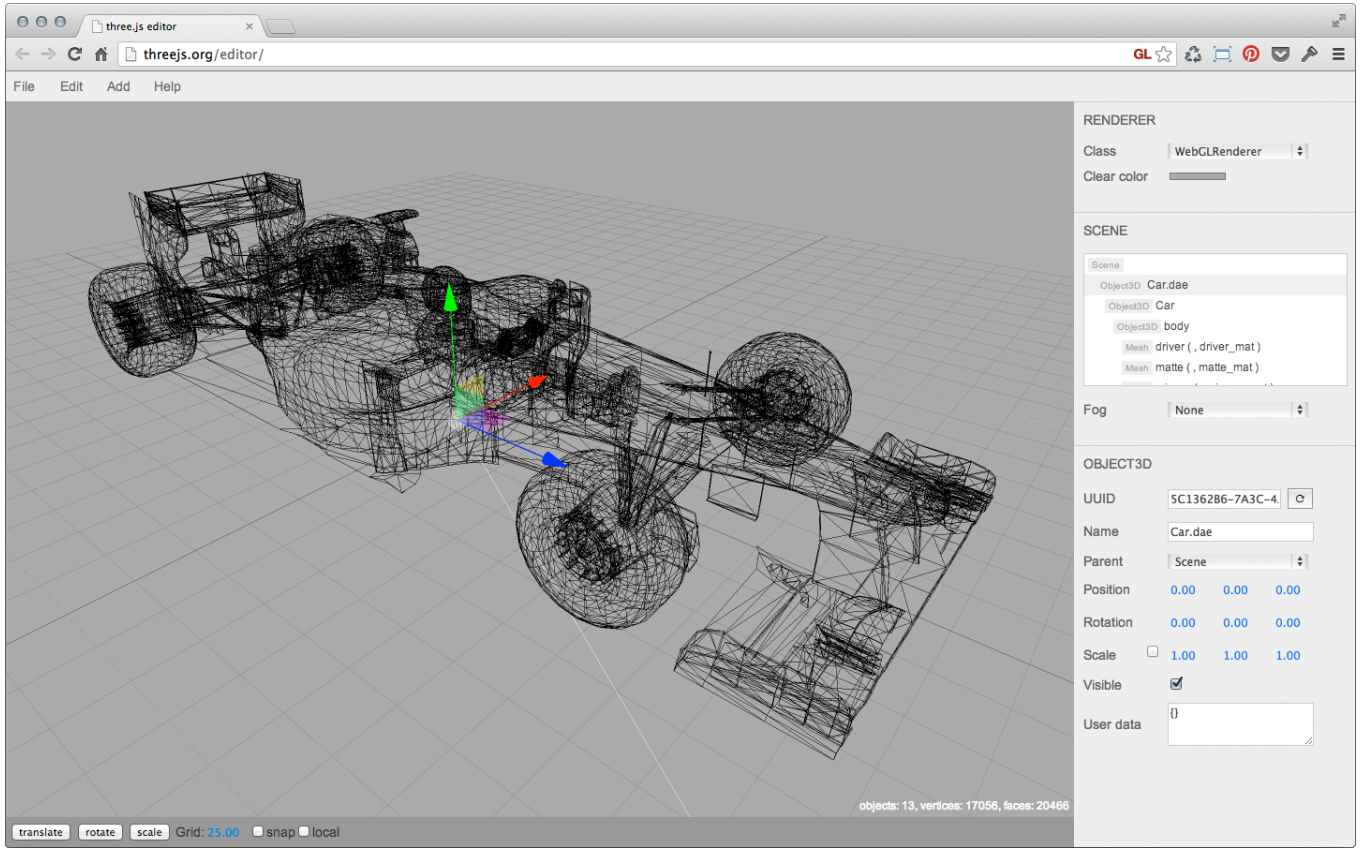
Unity 3D


Unity - TerrainTest.unity - MyFirstProject - PC and Mac Standalone
File Edit Assets GameObject Component Terrain Window Help
Center Local
Layers Wide
Scene Game
Tex - Wire RGB Gizmos All
Textured
Wireframe
Tex - Wire
Render Paths
Lightmap Resolution
Light Probes
Persp
Inspector
Terrain Static
Tag Untagged Layer Default
Transform
Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
Terrain (Script)
Raise / Lower Terrain
Click to raise. Hold down shift to lower.
Brushes
Settings
Brush Size 88
Opacity 14
Terrain Collider
Material None (Physic Material)
Is Trigger
Terrain Data New Terrain
Create Tree Colliders
Hierarchy
Create All
Main Camera
Terrain
Project
Create All
New Terrain
Scenes
MyFirstScene
TerrainTest
Standard Assets
lynda.com

Unreal Engine



Three.JS Editor



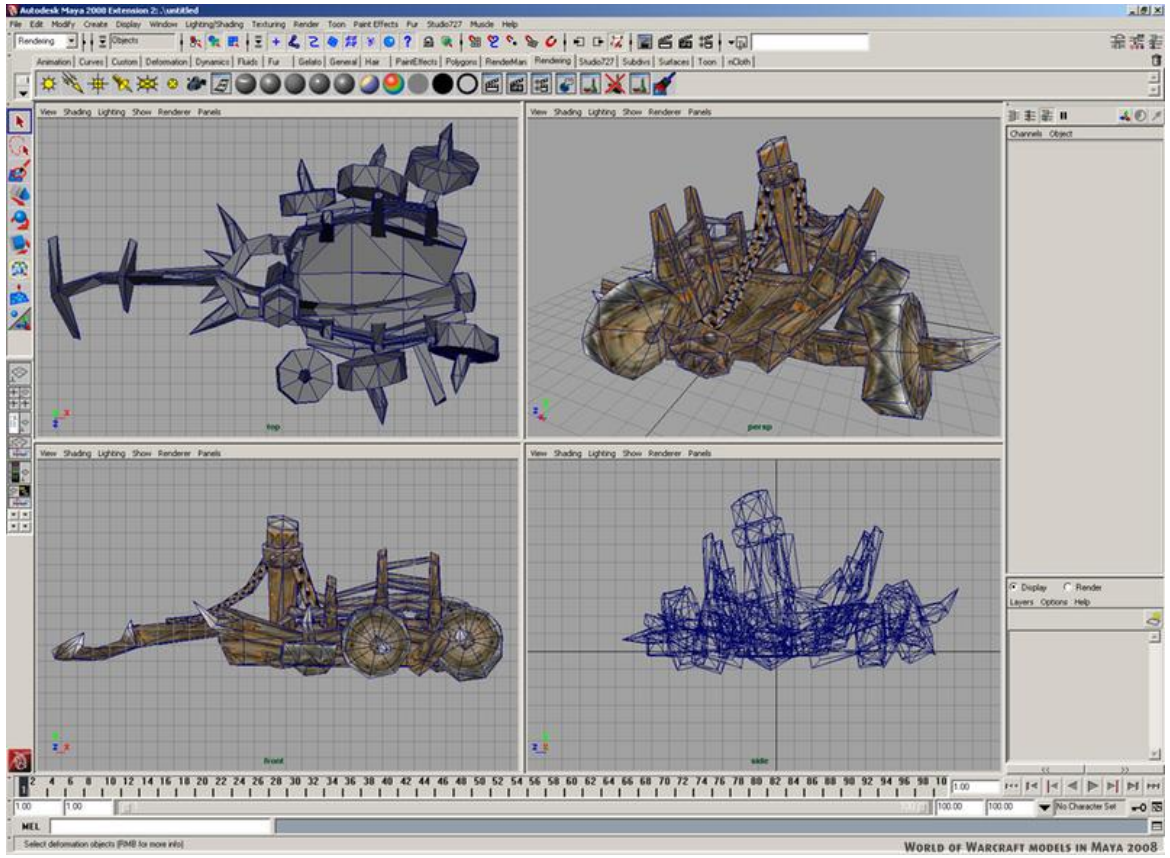
3D Modeling

Creating Models

3D models can be built and textured with specialized software:

- Maya 3D
- Blender
- Z-Brush
- SketchUp

Models are converted into a file format that can be processed by graphics programs (such as a .obj) file



Finding Models

There are many resources for finding 3D models to use in a software application online

Unity and Unreal both have marketplaces where assets for game and app development can be purchased