Introduction to Unity

CAP 6121 – Spring 2013

Salman Cheema

Monday, January 14th, 2013

What is Unity3D

- Game Development Tool
  - www.unity3d.com
- Useful Features
  - Imports 3D models easily*
  - Terrain Modeling Tool
  - Integrated Physics Engine (NVidia PhysX)
  - Audio
  - Networking
  - Highly scriptable (C#, Boo and Javascript)
  - Very easy to prototype games
- Free and Pro versions
  - Pro version has more features, e.g. ability to play videos

Why use Unity?

Unity

- High Level
- Little programming
- Components already available
- Highly visual

OpenGL, DirectX, XNA

- Low level
- Lots of Programming
- Start from scratch
- Result isn't immediately visible

Outline

- Unity's Editor
- Prefabs
- Scripting
- Creating a small-scale FPS
  - Camera Setup
  - Terrain Modeling and Lighting
  - Importing/using Assets (Models, Textures, Sounds, particles)
  - Basic Player State Management
  - Collision Detection
  - Simple Enemy AI
- Using the Kinect with Unity3D

The Unity Editor

- Scene View
  - Positioning/manipulation of objects in the environment
- Inspector
  - Alter properties of game objects
- Project
  - Shows everything in current project
- Hierarchy
  - Contains objects in currently loaded scene

Components of a Unity Project

- Scenes
  - Initial Menu, Game Level(s), High Scores, …
- Game Objects
  - Geometry, Particles, Camera(s), …
- Scripts
  - Behavior for Player, Enemies, Collisions, …
- Other Resources
  - Sounds, fonts, images, prefabs …

Example :

- Create a Project
- Model Terrain
- First Person Controller
- Mess around with Lights

GameObjects

- Everything in a scene = GameObject
  - Characters, Power Ups, Explosions
- Container
  - Can be empty
- Can have custom components
  - Scripts, Colliders, Rigid Bodies
- Parenting: Can be arranged in a hierarchy
  - Useful for constructing complicated objects

Prefabs

- Prefabs = “Templates”
- Example: Health Powerup
  - Load 3D model for Heart
  - Position point lights to light the model properly
- What if you want a bunch of powerups
  - Solution: Create a template (prefab) and clone it
- Benefit: Changes to template applied to all instances

Physics

- RigidBody Component
  - Forces, velocity, …
- Collider Component
  - Box
  - Sphere
  - Capsule
  - Mesh
- Trigger
  - Ignored by the physics engine
  - Can be used to trigger game events, cut scenes, etc

Scripting

- Change behavior of Game objects
  - Only form of programming required in unity
  - Most important aspect of a game
  - Can be written in C#, JavaScript, or Boo
- Important functions
  - Start — Called when a script is instantiated
  - Update — Called once every frame
  - FixedUpdate — Physics update
  - OnGUI — Used to display GUI (score, health,…)
  - OnCollisionEnter — Collision Detection
  - OnTriggerEnter — Collision with a Trigger
- Online Manual
  http://docs.unity3d.com/Documentation/ScriptReference/index.html

Scripting : Important Classes

- Mathematics
  - Vector3, Quaternion, Mathf, Ray, …
- Audio Related
  - AudioClip, audio, …
- Physics Related
  - Rigidbody, Collider, Physics, …
- GUI Related
  - Texture2D, GUI, …
- Others
  - GameObject, Input, Application, …

Scripting Example

- Animate Power Up
- Show Health Bar
- Pickup Script
- Enemy AI script
- Combat

Scripting : Fine Print

- A script can be applied to multiple game objects
  - Each game object gets own copy
  - Public variables visible in Inspector
    - modifiable at runtime
    - Can drap and drop
- Be careful with parenting and tags
  - Components referenced in script may be within children
- Make use of Debug.Log for debugging
- Be Cautious: Build incrementally

Important Links for Unity

- Models
  - Google Sketchup warehouse
- Documentation
  - http://unity3d.com/company/support/documentation/
- Script Reference
  - http://docs.unity3d.com/Documentation/ScriptReference/index.html

Setting up the Kinect on a PC

- Use the Kinect-Unity Interface Plugin
  - http://eecs.ucf.edu/isuelab/unity.php
- Pre-requisites
  - Microsoft Kinect SDK
  - http://www.microsoft.com/en-us/kinectforwindows/develop/developer-downloads.aspx
- NOTE: Plugin May not be compatible with Unity4.0
  - Tested against Unity 2.6.1
    - with SDK 1.0
  - Having problems with Unity 3.5.7
    - But supposed to work with latest SDK

Interfacing Kinect with Unity

- Instructions to get DLL.
  - http://eecs.ucf.edu/isuelab/unity.php
- Write unity script to interface with DLL
- Key Components
  - KUInterfaceCPP.dll
  - Put it above the "Assets" folder in your project
- Example

Questions?