Overview

Unity3D - is a commercial game development environment used by many studios who publish on iPhone, Android, PC/Mac and the consoles (i.e. Wii, Xbox). The cool thing about it is: there is a fully featured - free edition available for anyone to download. The main restriction is that you can only deploy games on PC/Mac and the web – but otherwise it is a fully workable commercial platform.

It is very powerful and is designed to be used by everyone on the development chain - from artists to programmers. It has proven to be so usable that Macquarie University will be using it as the main platform for a new course in 2012: "COMP260 Game Design”.

For more information, please visit:
http://www.comp.mq.edu.au/undergrad/info/units/
http://www.unity3d.com

The goal of this workshop is to give students a quick introduction to Unity so they are able to gain a working knowledge of the software. This can then be used as a basis for further specialisation depending on the student’s interest. At the end of the workshop students should be able to:

1. Start a new Unity3D project and create basic scenes.
   1.1. Understand the basic interface.
   1.2. Effectively navigate the scene view.
   1.3. Manipulate objects within the scene view.
   1.4. Manipulate objects using the inspector pane.
   1.5. Utilise the hierarchy and project panes.
   1.6. Understand basic lighting concepts.

2. Understand the role of GameObjects and Components.
   2.1. Add and remove components to GameObjects
   2.2. Manipulate Component parameters on GameObjects
   2.3. Understand the role of Collider components
   2.4. Understand the role of the RigidBody component

3. Create compound GameObjects
   3.1. Construct GameObject hierarchies
   3.2. Understand relative transformations within hierarchies.
   3.3. Create and use Prefabs

4. Understand how externally generated resources can be used in Unity3D projects.
   4.1. Create and use basic textures on GameObjects
   4.2. Understand the role of shaders on GameObjects
   4.3. Use externally generated 3d models in Unity3D scenes.

5. Understand the role of custom scripts as Components
   5.1. Effectively utilise pre-written custom scripts.
   5.2. Write a very basic custom script that is usable by others.

Note: Unity is designed to be the convergence platform for all the different crafts people involved in the game development process. It does not pretend to be a 3D modelling program, a photo editing package or a sound studio. However, it is designed to work well with specialist software in those fields.

Author(s): Matt Cabanag, [add your name here if you contribute to this document]
Activity 1: First Look at Unity3D

!! Pre Requisite: Unity3D
If you do not already have Unity3D installed, please download your free copy from the Unity3D website and install it on your computer. (http://www.unity3d.com)

!! Pre Requisite: Macquarie Unity3D Workshop Package
Assuming Unity is happily running on your PC, download the Mac Uni Unity Workshop package. http://web.science.mq.edu.au/~mattc/Unity3D/MQUnity3DWkshp_20110831.unitypackage

This will have a bunch of pre-made objects that you will be able to play with from the get-go. You will later learn how to make things from scratch, but at this stage let’s start with this.

Let’s Get Going:

1. Start Unity3D
2. Create a new project:
   2.1. If no project had previously been loaded into Unity, you will be met by the “Unity Project Wizard”. This is where you want to be.
   2.2. If there was a project previously loaded into Unity, it will load that automatically. In this case, go to File -> New Project.
   2.3. At this point you should now be at the “Unity Project Wizard”. Click on the “Create Project” tab. Do not select any packages. If you would like to modify the project location, do so now. Click on the “Create” button at the bottom right to proceed.
   2.4. Unity should now show you an empty project.
3. Import the Macquarie Unity3D Workshop package
   3.1. You should have somehow obtained the Macquarie Unity3D workshop package. It should look something like “MQUnity3DWkshp__.unitypackage”.
   3.2. Find this file on your system and double click it. You will be prompted to import the package contents. Import everything.
4. Load the workshop scene:
   4.1. After the contents have been imported, you should see new folders appear on the Project pane. Namely: “MQUnityWorkshop” and “StandardAssets”.
   4.2. By default, the project pane is at the bottom, middle section of the Window. If you refer to Figure 1 (see the figures section), it is labelled “E”.
   4.3. Expand the MQUnityWorkshop folder, then look for another folder called “Scenes”
   4.4. Expand the “Scenes” folder
   4.5. Double click on “mq_unity_workshop_lvl1”
5. Once you’ve loaded this scene, you should be seeing something like what is shown in Figure 1 (see the figures section). Each section of the diagram is labelled with coloured letters.
   5.1. “(A)” is the Scene View. This is where you will be spending most of your time editing your world. This is where you will place new items and manipulate them to suit your needs.
   5.2. “(B)” is the Game View. This is what the player of your game will actually be seeing. When you play-test your levels, you will be looking at this screen.
   5.3. “(C)” is the Inspector pane. If you select an object within the Scene View, all the attributes of that object will be displayed in here. Modifications these attributes can be made here too.
   5.4. “(D)” is the Hierarchy pane. This is a list of all the objects that exists in the current scene. In the same way that you can select objects in the Scene View, you can also select them here.
   5.5. “(E)” is the Project pane. This gives you a view of all the various resources that your project uses. In here you will see scripts, textures, (as you saw earlier) scenes. This is actually just a view of your computer’s file system. You can navigate the project folders outside of Unity.
6. Play around and get a feel for the engine.
   6.1. Click the “Play” button. Use WASD controls. Left click to shoot the laser, right click to shoot the grenade.
   6.1.1. Take note of the physics simulated objects.
   6.1.2. The floating MQ logos spin and move in different ways depending on where you hit them.
   6.1.3. The columns fall to the ground when you shoot them.
   6.1.4. The little dominos fall, one after another.
   6.1.5. All these behaviours come built in with Unity. No special programming had to be done.
   6.2. If you think the Game View is a bit small right now, you can maximise it by clicking on the pane options button. (See Figure 2). You can also automatically maximise the Game View when you press play by clicking on “Maximise on Play”.
   6.3. Notice that while you move things around in the Game View, the same changes are occurring on the Scene View – just not from the same perspective. In fact, you can still edit the scene while the “Play” button is depressed – but the changes will not be kept once you stop playing.
   6.4. You can pause the simulation by clicking the “Pause” button. It looks like “| |”
   6.5. If you want the simulation to only progress one frame at a time, use the “Step” button. It is right next to the “Pause” button and it looks like “>|”.
   6.6. Once you’ve finished playing around, click on the “Play” button again to stop the simulation.

7. Navigate the Scene View.
   7.1. Now that we’ve got some feel for the engine from the player’s perspective, we can start looking at it from the creator’s perspective.
   7.2. Click the “Scene” tab on the Scene View.
   7.3. You can also click within the view itself, but you may inadvertently select an object. We do not want to do that just yet.
   7.4. Activate and use FPS controls.
   7.4.1. Hold the right mouse button and move the mouse around. You will notice that you’re rotating the camera within the scene view.
   7.4.2. While holding the right mouse button, press the WASD keys. You should now be floating around the scene.
   7.5. Quickly zoom in and out. Quickly move side to side, up and down.
   7.5.1. Roll the scroll wheel forward and backward. You should be zooming in and out of the scene very quickly.
   7.5.2. Hold the middle mouse button and move the mouse around. You should move laterally – sideways and upwards.
   7.6. Focus on a particular object.
   7.6.1. Now click on an object in the scene to select it.
   7.6.2. You can use the FPS and other controls to manually get a good view of this particular object, but you can also just press the “F” key.
   7.6.3. You can select objects by clicking on them on the hierarchy pane. You don’t have to click on a visible object in the scene view.
   7.6.4. Note that the “F” key focus method will only work if your mouse is hovered over the Scene View.
   7.7. Orbit around a particular object.
   7.7.1. Select an object somehow; either through the Hierarchy pane or by clicking on them in the Scene view.
   7.7.2. Hold ALT, hold the left mouse button and move the mouse around. The scene view camera should now be orbiting around the selected object.
   7.8. Keep playing around with these controls until you are comfortable with them. Being able to move around your scenes effectively is important for editing your scenes effectively.
8. Manipulate objects within the scene view.
8.1. Now that we’re comfortable moving around the scene, we can now start moving the objects within the scene.
8.2. Move to a point where you have a good view of one of the rectangular columns.
8.3. Once there, click on one of the columns. Your screen should look something like Figure 3.
8.4. Move the object around.
8.4.1. You’ll notice you can move the object around if you drag the mouse.
8.4.2. Sometimes, this kind of movement is too imprecise for our purposes. You can get more precise movement by dragging the temporary “gizmo” arrows that is now sticking out from the object.
8.4.3. The blue arrow moves the object on the X axis only. The red arrow moves the object on the Z axis. The green arrow moves the object on Y axis
8.5. Rotate the object.
8.5.1. Click on the rotation mode button to activate rotation. See Figure 4.
8.5.2. You will notice that the three arrows have now become a bunch of circles.
8.5.3. Click and drag the object to for freeform rotation.
8.5.4. Click and drag a circle to rotate on that particular axis.
8.6. Scale the object.
8.6.1. Click on the scale mode button to activate. See Figure 5.
8.6.2. You will notice that the circles have now become a set of 4 boxes.
8.6.3. Click and drag the middle square to scale uniformly. You’ll scale the entire object.
8.6.4. Click on any of the outer boxes to scale on that particular axis. You’ll be able to stretch the object sideways, or up and down or in and out.
8.7. You can quickly switch between translation, rotation and scale modes by using the keyboard shortcuts. This is a lot faster than clicking the icons.
8.7.1. “W”: for translate
8.7.2. “E”: for rotate
8.7.3. “R”: for scale
8.7.4. As a side note, “Q” will activate lateral movement mode. This is the same thing as clicking the middle mouse button. This is more of a scene navigation tool than an object manipulation tool.
8.8. You might have noticed that while you have been manipulating objects, the values in the Inspector pane have been changing too. As a reminder, the Inspector pane is the one labelled “(C)” in Figure 1. You can manipulate objects by changing the values within the Inspector pane. This can sometimes be handy when you have something very specific in mind. If you don’t want any rotation in your object, just set all the rotation values to 0.
8.9. Practice re-arranging and re-scaling objects into various formations. As you can already imagine, a lot of the time you’ll spend with Unity will be spent doing this sort of thing.
9. Add objects into the scene.
   9.1. Up to this point, we've only been playing around with objects that already exist in the scene. Now we'll go ahead and add new ones.
   9.2. There a few ways to do this. The first method we'll look at is duplication.
         9.2.1.1. Select an object within the scene view.
         9.2.1.2. Now press CTRL+D
         9.2.1.3. You'll notice that the Hierarchy pane may have flashed a bit. This is because a new object has actually been added to the scene. It is not so obvious in the scene view because the new copy is actually overlapped with the original.
         9.2.1.4. Press “W” to go to translate mode, click and drag one of the arrows to move the new copy away from the original.
      9.2.2. Multiple object duplication.
         9.2.2.1. Sometimes, you might want to duplicate more than one object a time. It will very quickly become tedious if you could only do this one at a time.
         9.2.2.2. You can left-click drag on the scene view to create a selection box. Any objects that are completely inside this box will get selected. See Figure 6.
         9.2.2.3. Now press CTRL+D.
         9.2.2.4. You have now just duplicated all the selected objects. But the copies are still overlapping the originals. Move the copies.
      9.2.3. Be careful about overlapping duplicates. If you don't separate overlapping objects, they will **EXPLODE** when you press the play button.
   9.3. You can also add objects by dragging them in from the Project pane. As a reminder, the Project pane is the one labelled “(E)” in Figure 1.
      9.3.1. Go to the project pane and expand the “MQUnityWorkshop” folder
      9.3.2. Now find and expand the “Doodads” folder. You should now see a list of objects that you can add into your scene. These are called Prefabs, but we'll get back to that later.
      9.3.3. Drag one of the doodads into the Scene view to add them to the scene.
      9.3.4. You can also drag the doodads into the Hierarchy pane, but you don't have much control over where they appear on the scene.

**Exercise 1: Rube Goldberg Dominos**

Wikipedia Quote: "A **Rube Goldberg machine**, *contraption*, *device*, or *apparatus* is a deliberately over-engineered or overdone machine that performs a very simple task in a very complex fashion, usually including a chain reaction."


If you’ve gone through all the steps in Activity 1, you should now have enough skills and knowledge to create a very simple Rube Goldberg machine. For example, you should be able to set up some dangly legs, to be able to kick down the first of a series of dominos that will eventually cause the collapse of a simple tower. Use the pre-made objects within the scene and the project to create something like this. Be creative! Whoever comes up with the most interesting one (usually) gets a prize.
Activity 2: Starting Scenes from Scratch

[Create new scene, Add floor, Add directional light, Add completely new GameObjects (not a prefab), Add FPS script to viewer object, Add box colliders, Add RigidBody components, Create a compound GameObject, Create joints, Save to Prefab]

Exercise 2: Interactive Sculpture

[Get students to create a virtual interactive sculpture in the same vein as the dangly legs or the spinny thing doodads.]

Activity 3: Working With External Apps

[Create new texture with MS paint. Apply to GameObject. Demonstrate dynamic resource loading – edit in Paint, save, see it updated in Unity. Import 3D model created with Blender. Edit and save in Blender, see it updated in Unity.

Good time to segue into Blender or GIMP specific activities.]

Exercise 3: External Art

[Get students to create externally generated assets themselves; use said assets in their Unity projects]

Activity 4: Working With Scripts


Exercise 4: Hello Unity World

[Get students to make a HelloWorld script. Maybe a simple OnClickMove thingie..]
Figures:

Figure 1

Figure 2