

# Intro to Unity Game Engine

by John Twycross ( [jtwycross@brookes.ac.uk](mailto:jtwycross@brookes.ac.uk) )

This session will acquaint participants with the foundations of application development using the Unity game engine. It will introduce techniques for working with gameobjects, User Interfaces and C# scripts.

**Concept:** Scenes, Game Objects and Scripts.

**Concept:** Project Files, Scene Hierarchy, Inspector

## **Rapid Application Development in Unity3d**

**Concept:** Camera, Lights and Action

Standard Assets

- Prototyping
- Characters
- Environments
- Vehicles

What is a package?

**Demo:**

RAD in action...

1. Assets > Import Package.
2. Build an example scene
3. Save the scene
4. File > Build Settings
5. Add scenes to Build
6. Select Platform
7. Build & Run

## Game Objects

**Concept:** Prefabs, Components, Rigid Objects

( <https://docs.unity3d.com/Manual/Prefabs.html> )

( <https://docs.unity3d.com/Manual/Components.html> )

( <https://docs.unity3d.com/ScriptReference/Component.html> )

( <https://docs.unity3d.com/Manual/class-Rigidbody.html> )

### Demo:

Importing 3D models

- Props, items & pickups.
- Environments
- Vehicles
- Characters

Importing 2D artwork

- Sprites
- UI graphics
- Textures

## Programming

### Concept: Design Patterns

#### Design Pattern - MVC

( <https://en.wikipedia.org/wiki/Model-view-controller> )

- Model class - (Data)
- Controller class - (Input & Transformations)
- Viewer - (Unity engine handles this)

#### Design Pattern - Observer ( [https://en.wikipedia.org/wiki/Observer\\_pattern](https://en.wikipedia.org/wiki/Observer_pattern) )

- Event handling
- Avoids use of Update()
- Triggers scripts when needed.

#### Research:

<http://gameprogrammingpatterns.com/contents.html>

<http://gameprogrammingpatterns.com/state.html>

## Application States

- Splash Screen
- Intro
- Main Menu
- Options
- Game Loop
- Pause
- Game Over

Finite State Machines ( <https://docs.unity3d.com/Manual/StateMachineBasics.html> )

## **Programming using**

### **Scripts as Components**

Reading variables in other scripts

Referencing game objects in scripts

Public variables in the inspector

### **Parent Child**

Broadcast message

( <https://docs.unity3d.com/ScriptReference/Component.BroadcastMessage.html> )

Get Component

( <https://docs.unity3d.com/ScriptReference/Component.html> )

## **Working with Game Objects**

Instantiate

( <https://docs.unity3d.com/Manual/InstantiatingPrefabs.html> )

Collision Events

( <https://docs.unity3d.com/Manual/CollidersOverview.html> )

Tags

( <https://docs.unity3d.com/Manual/Tags.html> )

Destroy

( <https://docs.unity3d.com/Manual/CreateDestroyObjects.html> )

## **Transformations**

**Concept:** Translate, Rotate, Scale

iTween

( [http://www.pixelplacement.com/site/portfolio/itween-httpyoutu-beqe5hpp4yah4listuue\\_21dtrvmiy3if0sayabla/](http://www.pixelplacement.com/site/portfolio/itween-httpyoutu-beqe5hpp4yah4listuue_21dtrvmiy3if0sayabla/) )

Animator

( <https://docs.unity3d.com/Manual/animator-UsingAnimationEditor.html> )

Code

( <https://docs.unity3d.com/ScriptReference/Transform.Translate.html> )