Getting Started With ScriptEase II and Unity

This series of tutorials will get you started with ScriptEase II and the Unity translator. A basic knowledge of Unity is assumed for these tutorials. A good starting point is Unity's own tutorials at http://unity3d.com/learn. Of course, since you'll be using ScriptEase II, you won't need to go through the scripting tutorials! Unity also has a great community forum with lots of resources at http://forum.unity3d.com. Please also make sure you have read the ScriptEase II Unity Instruction Guide, as it will mention if there are any known issues.

This tutorial teaches you how to use ScriptEase II with the Park Unity project as an example. Note that the Park project is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License. Please visit http://creativecommons.org/licenses/by-nc-sa/3.0/ to view a copy of this license before using the project to create your game. If you would like to use a project created using the Park scene commercially, or if you do not have the Park project, please contact the ScriptEase II team at script@cs.ualberta.ca.

In this first tutorial, we will be learning what ScriptEase II is and using it to make an apple fall out of the tree when you go near it. By the end of this tutorial, you will know how to generate scripts for your game with ScriptEase II, and be able to move onto the more advanced tutorials.

Getting Started:

1. Save the Park project somewhere you can find it again later.

2. Open the Park project with Unity and take a look around. Try playing it. You'll notice that the Player character already has a third person controller attached that allows you to walk around with the WASD keys, and a mouse look controller that allows you to look with the mouse.

3. Now open ScriptEase II from the ScriptEase II desktop shortcut.

> *Tip:* Unity allows you to use ScriptEase II while Unity is open. However, you will have to reopen ScriptEase II if you change anything in the Unity editor, such as adding a new Game Object or attaching a light.

4. Create a new Story from the File menu. Name your story "Park", enter your name in the Author field, and write "An Open World Adventure in a Park" in the description. Select "Unity" in the "Game" box and then find the "Park" project folder. Click on it, and then on finish when you're done.

> *Tip:* The Description and Author are only used by developers; they will not be visible to players.

5. ScriptEase II will now load your project. Let's take a quick look around first.

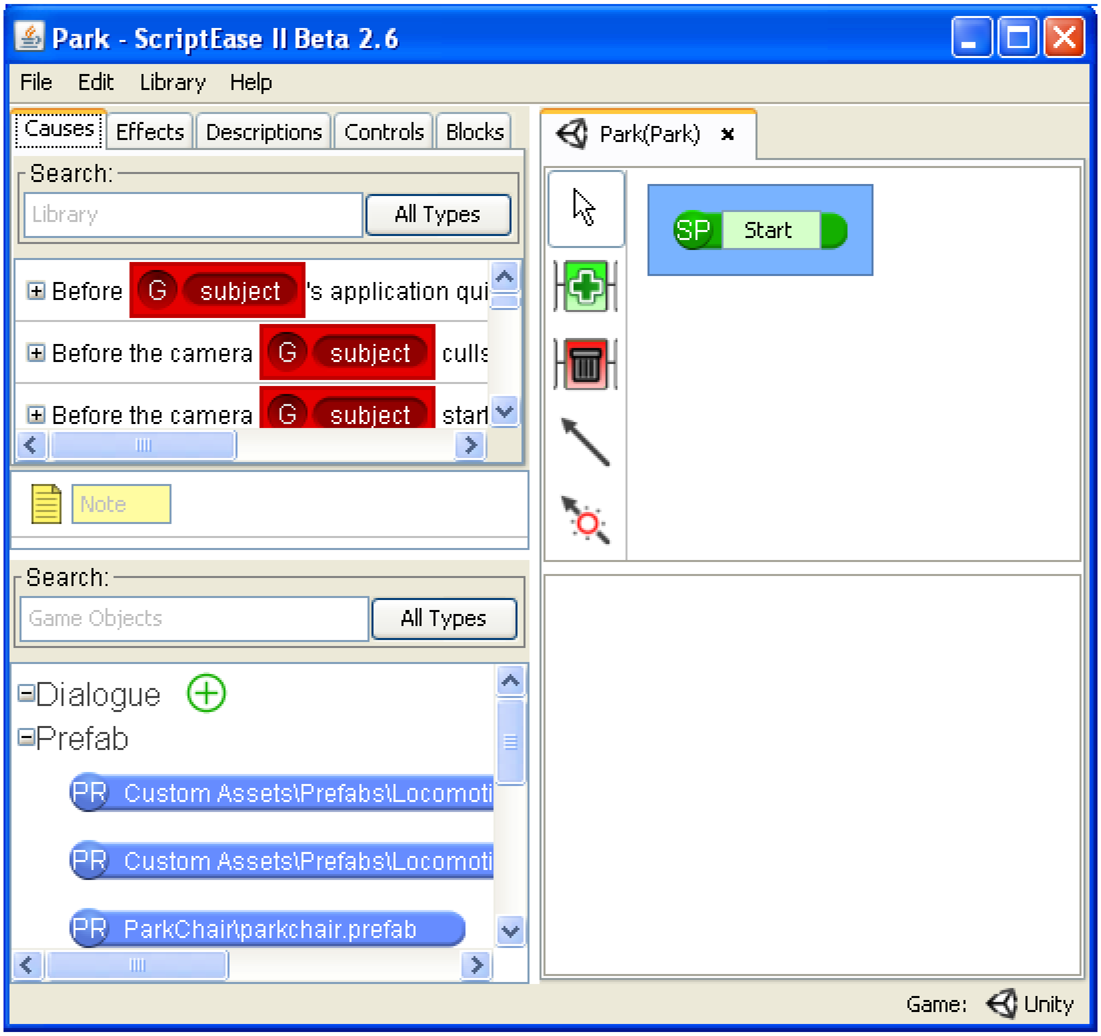

Park - ScriptEase II Beta 2.6
File
Edit
Library
Help
Causes
Effects
Descriptions
Controls
Blocks
Search:
Library
All Types
Before G subject 's application qui
Before the camera G subject culls
Before the camera G subject start
Note
Search:
Game Objects
All Types
Dialogue
Prefab
PR Custom Assets\Prefabs\Locomoti
PR Custom Assets\Prefabs\Locomoti
PR ParkChair\parkchair.prefab
Park(Park)
SP Start
Game: Unity

Figure 1: ScriptEase II when you first load your project. This may look different depending on your operating system.

6. ScriptEase II should look similar to Figure 1. The left side contains all assets that you can add to a story. The top part is the Library pane, which contains story components, such as causes and effects, which are used to make things happen. Underneath the Library is the Game Object pane, which contains the assets in your project that can be used with story components. In between the Library and Game Object panes is a special “Note” component. Notes do not add any functionality; they are used to add comments.

Caution: If the Game Object pane isn’t there, you can open it by dragging up from under the Library Pane. This is a known bug.

7. The right side represents the game’s story. You use the story graph to create story points to organize the story. Underneath it is the Story Point pane, which represents the contents of the selected story point. Story components are dragged into it from the library.

Tip: The story is in a tab, which lets you open multiple stories at once.

8. Before we start creating our story, we need to add a special library to it. The Park project has its own custom library with Effects to play the animations. We need to import it by going to “Library → Add Library” and clicking on “ParkGameAnimationLibrary”. This will add a few more effects to ScriptEase II. It needs to be added manually because other projects do not support this library.

Tip: Effects from this library have an ANI tag on the left side.

Scripting Game Objects with Causes and Effects:

9. We should give the Player character a walking animation so it looks a bit more natural. We need the cause called “When subject is created”. Make sure you are in the Causes tab before using the search box to search for “created”.

10. Drag the “When subject is created” cause from the Library pane into the Story Point pane below the Story Graph to match Figure 2. We only have one story point, Start, when we create a story, so the cause we drag in will be placed in this story point. Since “When subject is created” only happens once, when the game starts, it is best to always put it in the Start story point.

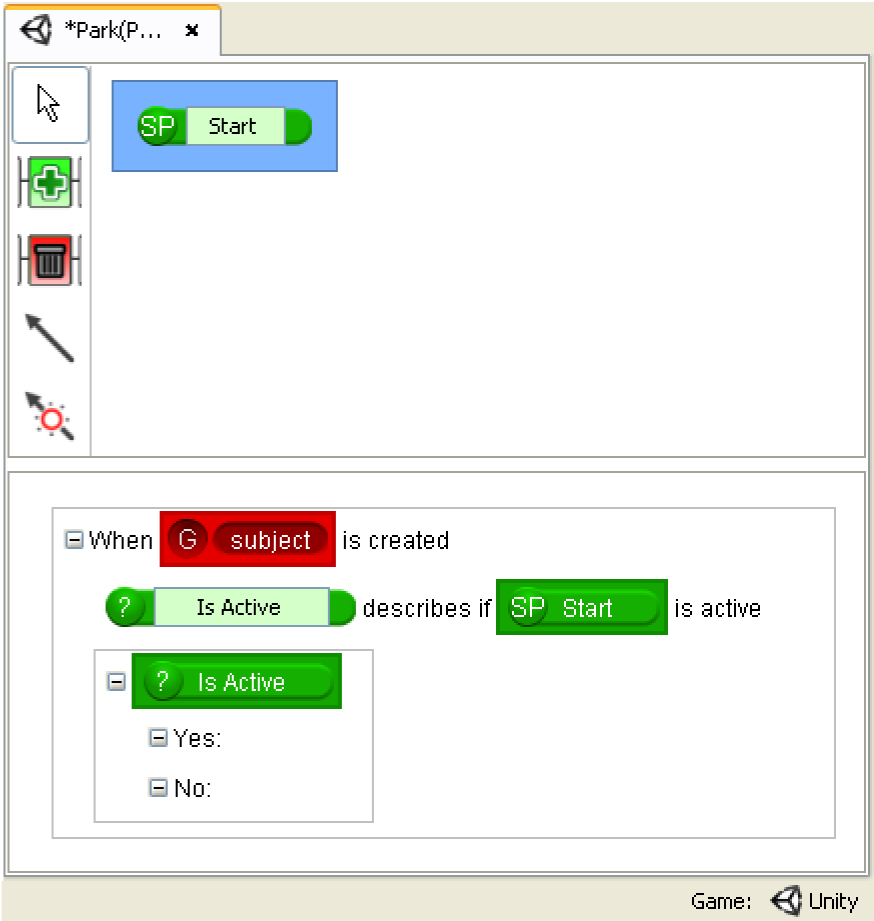
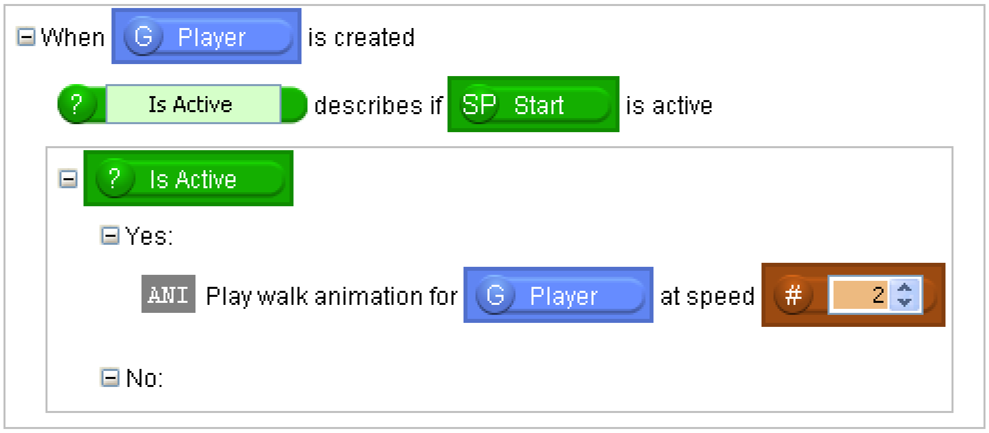

*Park(P...
SP Start
When G subject is created
? Is Active describes if SP Start is active
? Is Active
Yes:
No:
Game: Unity

Figure 2: A Cause has been added to the Start Story Point.

11. Every cause needs a subject. Let’s find the Player Unity GameObject in the Game Object pane below the Library. GameObjects are all contained in Scene files, so we need to expand the ParkScene scene object in the Game Object pane to find the Player object. Drag the blue Player object into the subject slot in the cause.

Tip: You can search in the Game Object pane.

12. We now have a complete cause, but the code it generates will not do anything yet! We still need to add effects. Click the “Effects” tab in the Library pane and search for “walk”. Drag the “Play walk animation for Character at speed” effect into the “Yes” part of the “When Player is created” cause to match Figure 3, although the slot in the effect will still be empty.



⊟ When G Player is created

? Is Active describes if SP Start is active

⊟ ? Is Active

⊟ Yes:

ANI Play walk animation for G Player at speed # 2

⊟ No:

Figure 3: The first Cause we are adding to our Story.

13. Drag the Player game object from the subject of the cause to the Character slot of the effect. We also want to set the speed to 2. This is a number you can experiment with later to control the speed of the walking animation. Your Cause should look like Figure 3.

14. The effect is in the “Yes” part of the Is Active question, which means it will only occur when the Start story point is active. Since the Start Story Point is active when the game starts, this will always and only happen when the game begins. We don’t need to worry about this right now as we will be going over the story system in detail later.

15. Save your story by going to “File” → “Save” in the menu or pressing Ctrl + S. You can save this SE2 story file anywhere. Saving will add the scripts to your project. Once it finishes saving, we can test it out by switching back to Unity. It will ask you to reload the project, which you should. In the Project browser, navigate to the ScriptEase Scripts folder, right click it, and click on “Refresh” to make sure the scripts are imported correctly.

Tip: If you encounter any problems with saving or testing the story, please refer to the ScriptEase II instructions.

16. Start playing the game. The player character should now have a walking animation! You have added your first script to the game. If you look in the Assets folder of the Unity project, you should see a new folder called "ScriptEase Scripts", and another one in "Plugins/ScriptEase C Scripts." You could make changes to these scripts, but ScriptEase II will discard them the next time you save the story. If you want to create your own scripts, make sure you do not save them to those folders, as everything in them is deleted whenever ScriptEase II is saved.

Descriptions:

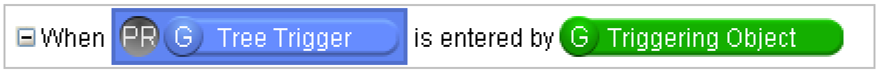
17. Let's add more scripts to liven up the Park. Fruit could drop when the player is by a tree. Leave Unity open and go back to ScriptEase II. First we need to find the "When subject is entered by Triggering Object" cause. You can search for "entered" to find it.

18. Fruit should drop regardless of our place in the story, so we drag the cause into the Start story point, either above "When Player is created", or at the very bottom. We will not have to add more story points until the next tutorial.

19. We can delete the two "Is Active" story components inside of the cause because we won't be using them; as mentioned before, this cause will not use the story system.

20. The Cause uses a Game Object that has a trigger component attached to it. One trigger in the Park scene is called Tree Trigger. Find it and drag it into the slot to as in Figure 4.

> *Tip:* If you're unfamiliar with Unity triggers, find the Tree Trigger Game Object in the Unity Editor. Notice that the "Is Trigger" checkbox is checked in its attached Collider.



⊟ When PR G Tree Trigger is entered by G Triggering Object

Figure 4: The temporarily empty second cause.

21. Now we want to create an Apple close to the Tree Trigger. However, there is no effect that can do this. Instead, we need to use a description. Descriptions are used to access game objects that can only be known during gameplay. The "Is Active describes if the Story Point is active" story component that is added by default to Causes is an example of a description. We need a different one to create our Apple.

22. Switch to the Descriptions tab in the Library and find the "Clone describes a clone of Object created at Game Object's current position" description.

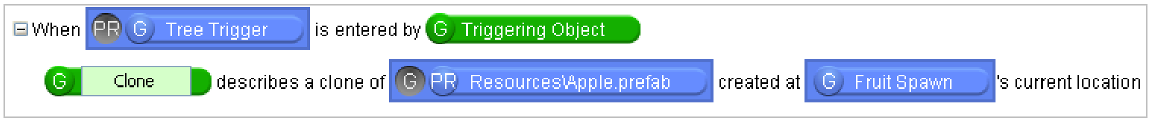
23. Drag the description into the Cause. Since we removed the story components that check whether the story point is active, the Tree Trigger checks if it is entered by an object at any point in the story.

Tip: Order matters! Objects created from descriptions can only be dragged down and only within the same Story Component.

24. We now need to find the Apple in the Game Object panel. However, it isn't in any of the Scenes because we haven't added it to one. Instead, the Apple is a Unity Prefab. We can drag it from the Prefabs category into the Object slot of the description.

Tip: We could drag the green Clone object created by the description into other effects to make things happen to the new Apple.

25. We need a place for the Apple to be created. This already exists in the Park scene as the Fruit Spawn, located inside the Tree Trigger we used earlier. Expand the Tree trigger and drag the Fruit Spawn into the Game Object slot in the description to match Figure 5.



⊟ When PR G Tree Trigger is entered by G Triggering Object

G Clone describes a clone of G PR Resources\Apple.prefab created at G Fruit Spawn 's current location

Figure 5: The second Cause of the story, now dropping apples on people's heads.

26. You should save the game now before you lose any work. Don't try it out in Unity yet! The apples will fall into the Tree Trigger, causing another Apple to be created. Then another and another. This will slow down your computer a lot, so we need to control when the Apples fall by checking to see what entered the trigger.

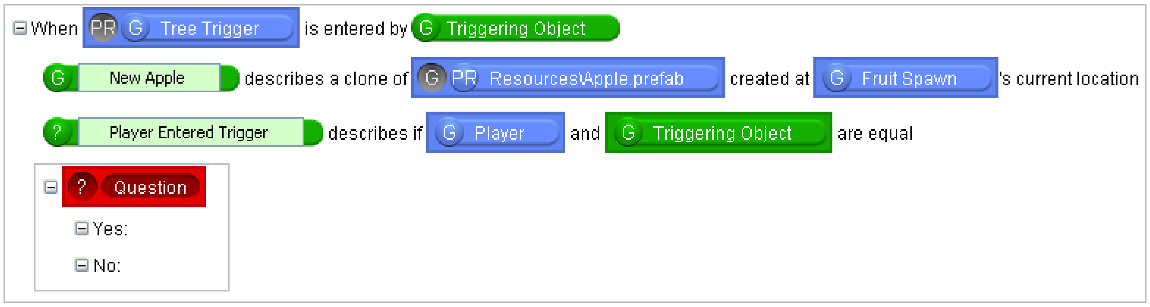
27. We can check if the Triggering Object is the Player by using a description called "Objects Equal describes if Object 1 equals Object 2." Drag this anywhere into the Cause.

28. Drag the Player object into one of its slots. Then drag the Triggering Object from the Cause into the other slot. The order of the slots does not matter.

29. We should also rename the object created by the description so it makes more sense. Change the name from "Objects Equal to" to "Player Entered Trigger". While we're at it, let's rename "Clone" to "New Apple."

Controls:

30. To use the Player Entered Trigger object created by the description, we need a special story component called a question. This is in the Controls category in the Library. Drag the question under the Player Entered Trigger description in the Cause as in Figure 6.



⊟ When PR G Tree Trigger is entered by G Triggering Object

G New Apple describes a clone of G PR Resources\Apple.prefab created at G Fruit Spawn 's current location

? Player Entered Trigger describes if G Player and G Triggering Object are equal

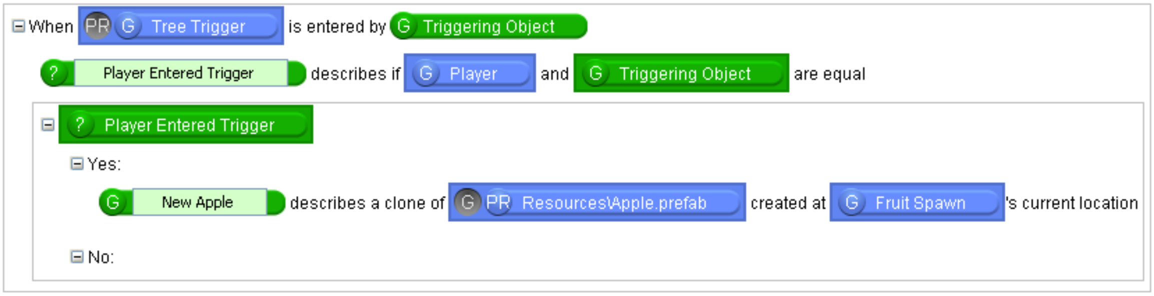
⊟ ? Question

⊟ Yes:

⊟ No:

Figure 6: The cause so far, with two descriptions and a question.

31. Drag the Player Entered Trigger object into the Question's slot. Then drag the entire New Apple description into the Question's Yes part. The Apple will now only happen if it was the Player who entered the trigger. Your final Cause should match Figure 7.



⊟ When PR G Tree Trigger is entered by G Triggering Object

? Player Entered Trigger describes if G Player and G Triggering Object are equal

⊟ ? Player Entered Trigger

⊟ Yes:

G New Apple describes a clone of G PR Resources\Apple.prefab created at G Fruit Spawn 's current location

⊟ No:

Figure 7: The completed trigger cause.

32. Every effect in a Question's Yes part will only occur if the object in the slot's question answers "Yes". Otherwise, everything in No occurs. You can drag story components into both parts. Questions can even be dragged into other questions.

33. Save the story and switch to Unity. Refresh your scripts and find the Tree Trigger by double clicking it in the Hierarchy view. Start the game and walk into the trigger!

Good job, you have completed the first tutorial! The Player's walk is animated now and fruits fall from trees. The Park still needs to feel more alive, so we will work on that in the next Tutorial.

ScriptEase II saves and reads from a *.ses story file. This file saves all of the story components you have added to the story. It is also linked to the game, making this file just as important as your game to work on a story in ScriptEase II. The story file isn't needed to play the game, so don't include it in the game's gold release. And always remember to back up regularly!

Caution: Since ScriptEase II is still in Beta, things can go wrong! Save copies of your .ses file elsewhere on your computer.

In the next tutorial, we will learn how to continue our story using Story Points in the story graph.