v1.1 10/27/2016

Introduction

Thanks for the purchase and support! We are a community of VR devs, working together to create games, experiences, development tools, and tutorials in an effort to empower emerging VR developers worldwide. Join us here: https://www.youtube.com/nurfacegames/

Video Tutorials

https://www.youtube.com/watch?v=dbDAmuTH5sw

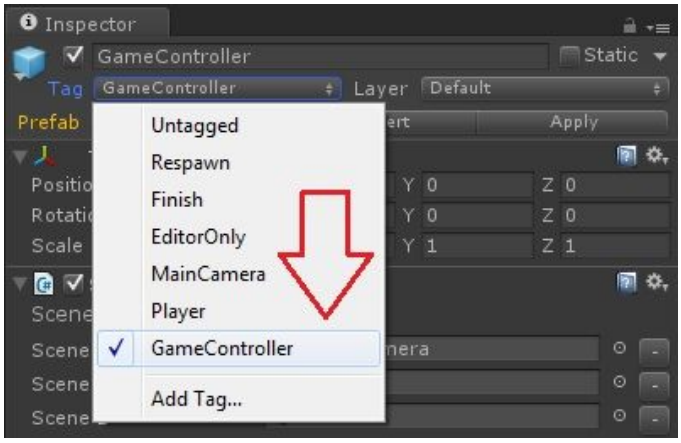
What is Additive Scene Manager?

Dealing with additive scenes and asynchronous scene loading can be difficult and confusing. This asset provides a set of functions that simplifies the asynchronous loading and unloading of additive scenes. The SceneController script has a custom inspector to display which scenes are loading/loaded so it's easy to understand what is happening and what loading stage each scene is at.

How To Use

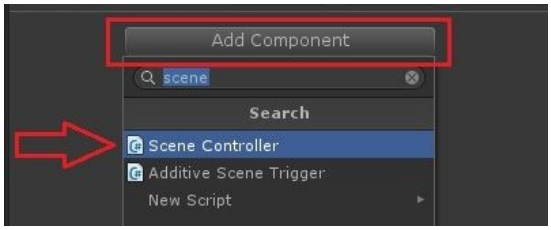
The player/camera/VRsetup is the object that needs to move between the additive scenes. Your main scene with **Main Camera** should have a Game Controller. More info can be found here: https://unity3d.com/learn/tutorials/topics/scripting/persistence-saving-and-loading-data

If you do not have a game controller, add a new empty GameObject, name it appropriately, and add Unity's default tag *GameController*:



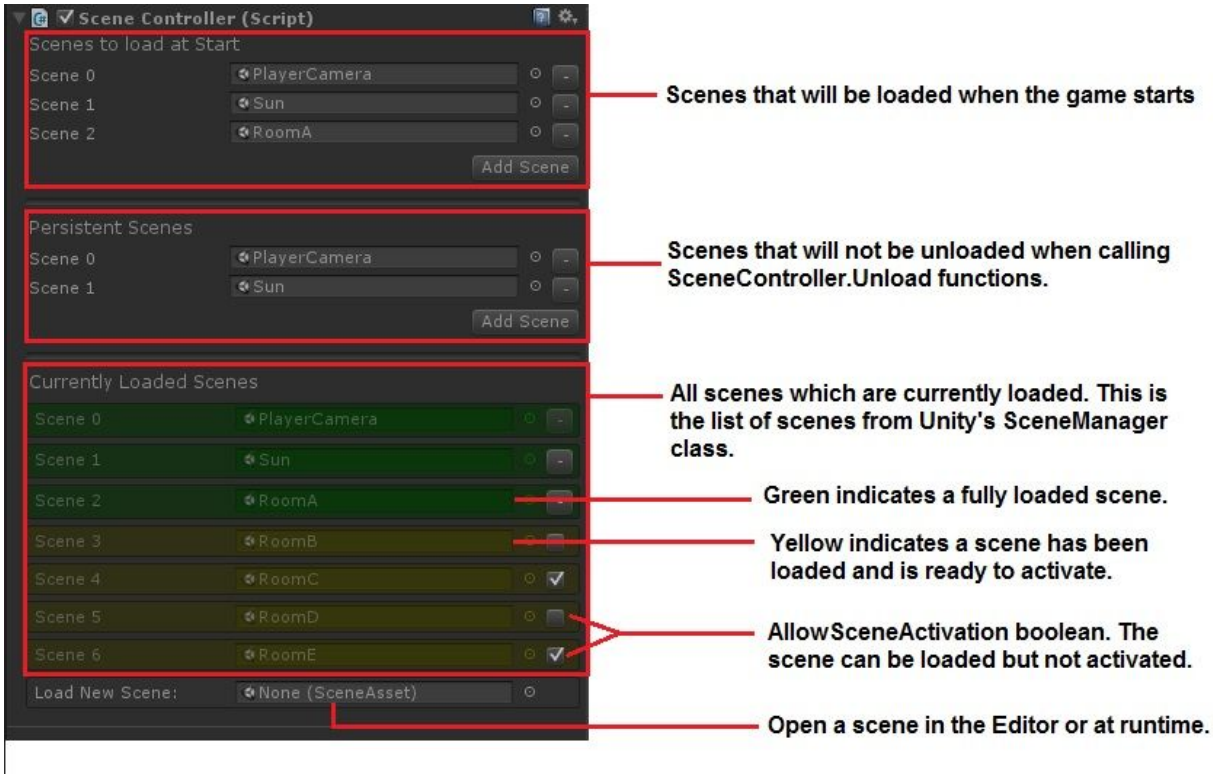
SceneController.cs

The SceneContoller script contains functions for loading and unloading scenes. Add this script to your GameController:



The SceneController class gives the ability to:

- Add scenes that will be loaded when the game starts.
- Add 'persistent' scenes that will not be unloaded by SceneController's Unload functions.
- View all scenes which are currently loaded by Unity's SceneManager class (https://docs.unity3d.com/ScriptReference/SceneManagement.SceneManager.html)
- View what loading stage each scene is at:
  Green = Fully Loaded. Async progress 1.0 (Async operation is completed/null)
  Yellow = Loaded but not activated. Async progress 0.9 (allowSceneActivation if false)
  Red = Loading in progress. Async progress 0.0 - 0.8. (Async operation is loading data)
- Set the *allowSceneActivation* boolean via a checkbox for scenes that are loading.
- Load new scenes into the editor or at runtime.



SceneController.cs Functions

public bool SceneExists(string name**)**
Check if a scene exists in the Unity project.

public void LoadLevel(string sceneName**)**
Load a scene additively and asynchronously. allowSceneActivation will be set to true.

public void LoadLevel(string sceneName, bool allowSceneActivation**)**
Load a scene additively and asynchronously, with control of allowSceneActivation boolean.

public void UnloadLevels()
Unload all scenes. Persistent Scenes (set via inspector) will not be unloaded.

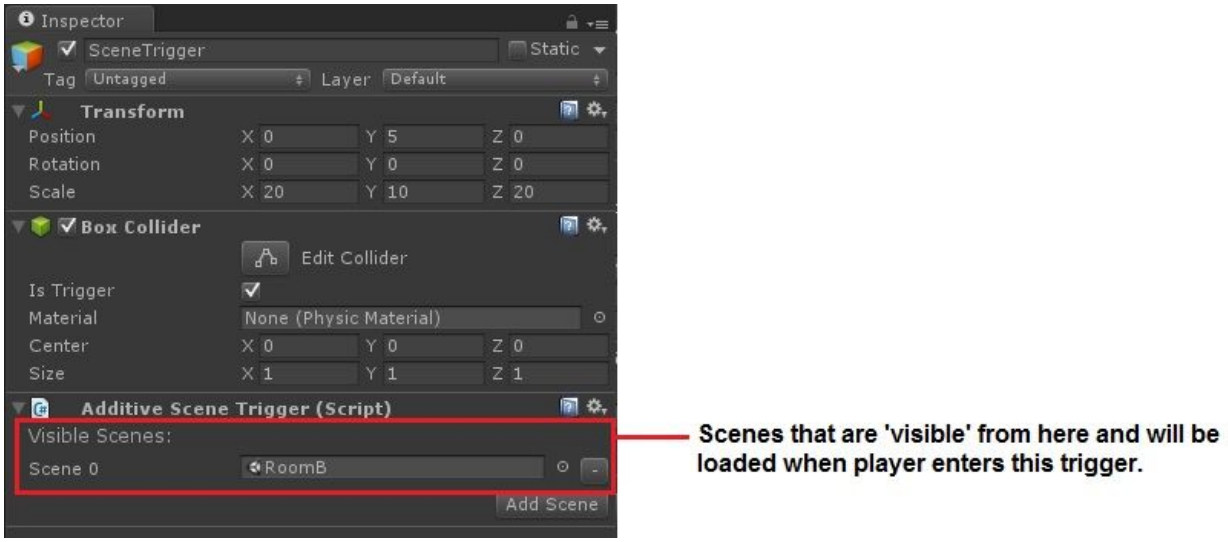
public void UnloadLevels(string exception**)**
Unload all scenes except '*exception*'. Persistent Scenes will not be unloaded.

public void UnloadLevels(string[] exceptions**)**
Unload all scenes except the list of *'exceptions'*. Persistent Scenes will not be unloaded.

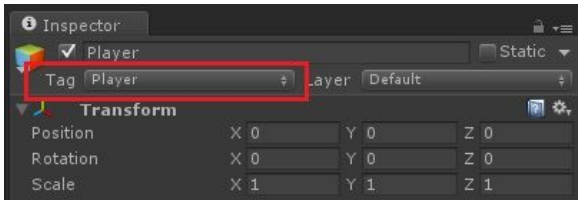
(reserved for future functions...)

AdditiveSceneTrigger.cs

This script utilizes the SceneController functions so you can get additive scenes working in your project without using any code. AdditiveSceneTrigger requires a Unity trigger, so add a collider and set *Is Trigger:*



- **Visible Scenes** is a list of scenes that are visible from this trigger. Any scene that is set here will be additively loaded when the player enters this trigger. All other scenes will be unloaded, except *Persistent Scenes*.
- Tag "**Player**" must be assigned to the Player/Main Camera object:

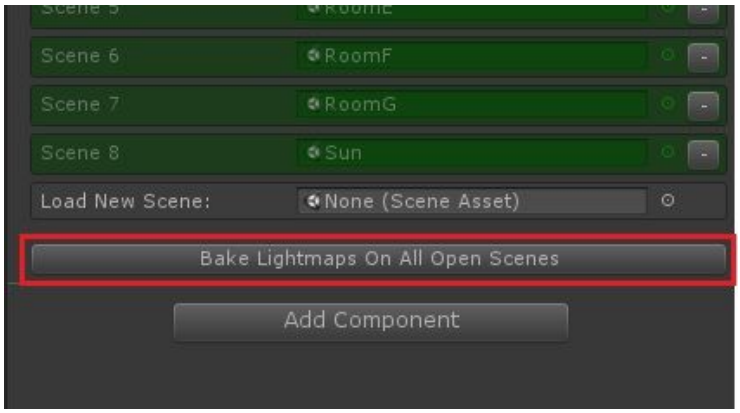


(see AdditiveSceneTrigger.cs line 15: *if (!other.gameObject.CompareTag("Player")) return;*)

Lightmapping Multiple Scenes

It is possible to bake lighting on multiple opened scenes at the same time. This uses the Unity Scripting API function, *Lightmapping.BakeMultipleScenes:*
https://docs.unity3d.com/ScriptReference/Lightmapping.BakeMultipleScenes.html

Open all of the scenes that you want to bake within the Hierarchy Window and then click the "Bake Lightmaps On All Open Scenes" button on the Scene Controller Inspector:



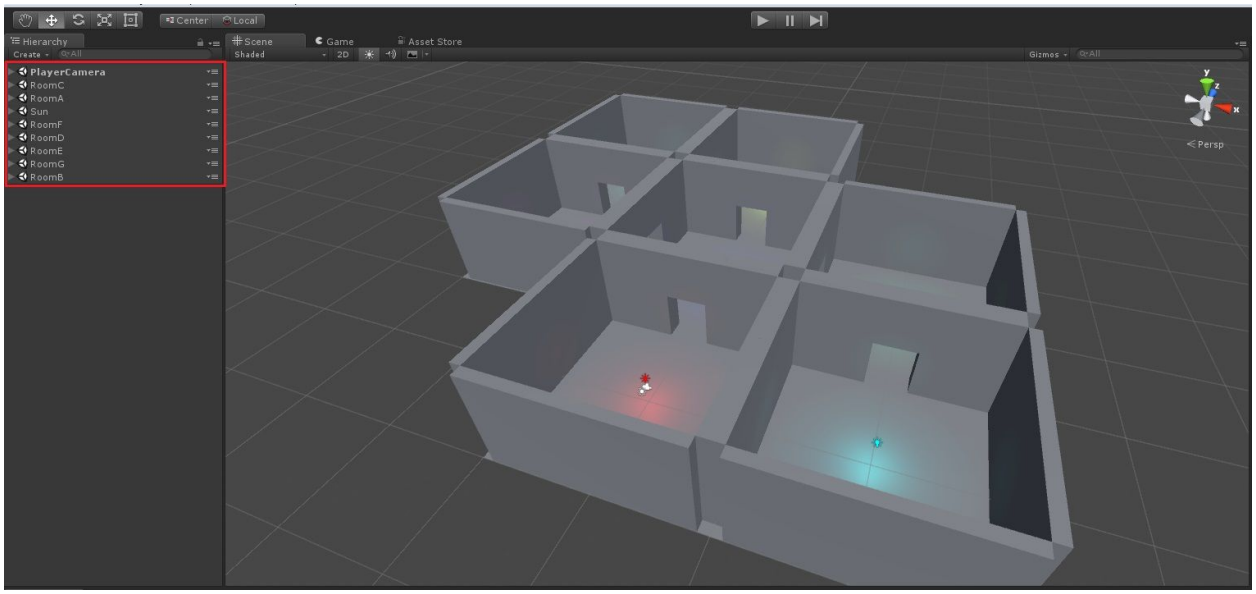
Unity will be unresponsive until the bake has finished on all scenes. This operation is currently not possible to run additively like a normal lightmap bake. I've opened a feature request for this, so that Unity Editor can be used while a multi-scene bake is running:
https://feedback.unity3d.com/suggestions/bakemultiplescenes-asynchronously-bakemultiplescenesasync-string-paths

The forum post can be found here:
https://forum.unity3d.com/threads/bakemultiplescenes-asynchronously.436790/

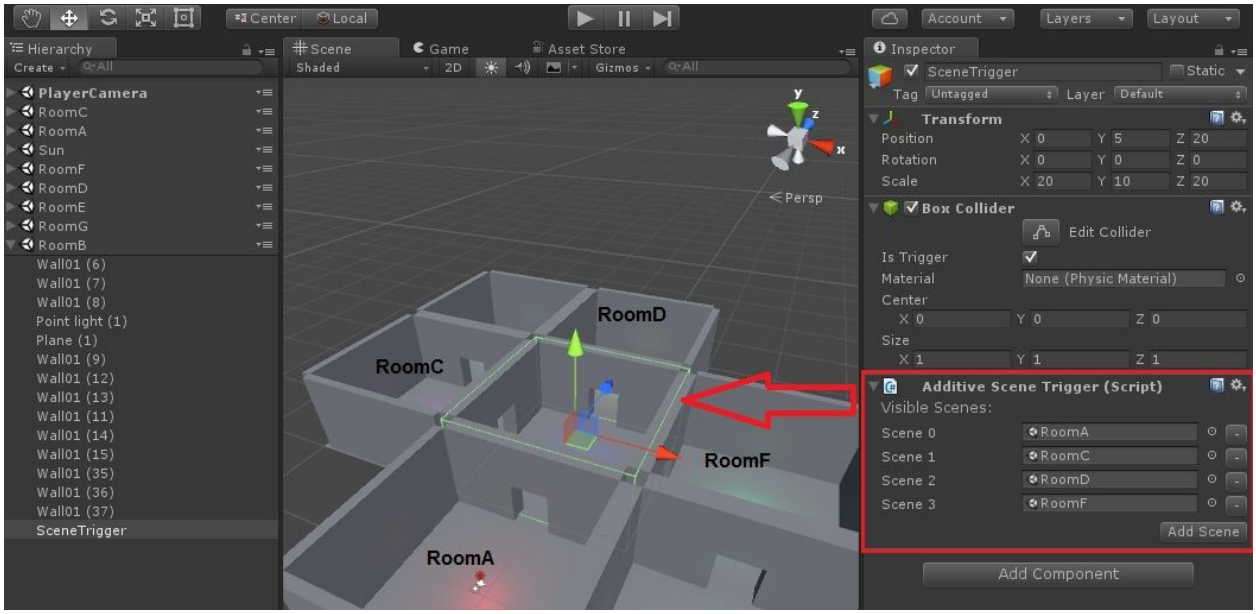
For now, until Unity adds a *BakeMultipleScenesAsync()* function to the Lightmapping API, be prepared to wait until the bake completely finished before you can use Unity Editor again.

Demo Project

The included project located in */AdditiveSceneManager/Demo/* has 9 scenes for how an example game may be set up. To see the complete project, load all 9 scenes in Unity:

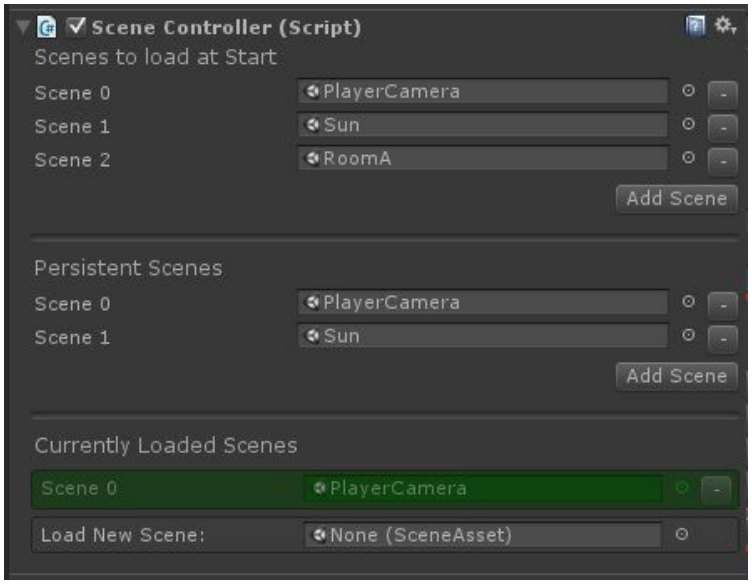


The idea is that the **PlayerCamera** scene and the **Sun** scene are to be loaded at all times, and each room will be loaded additively. Each room has an **AdditiveSceneTrigger** script with the visible scenes configured. In this example, we have only set the “Visible Scenes” for each room to be the room directly next to it. The following image shows how the middle room, RoomB, is configured:

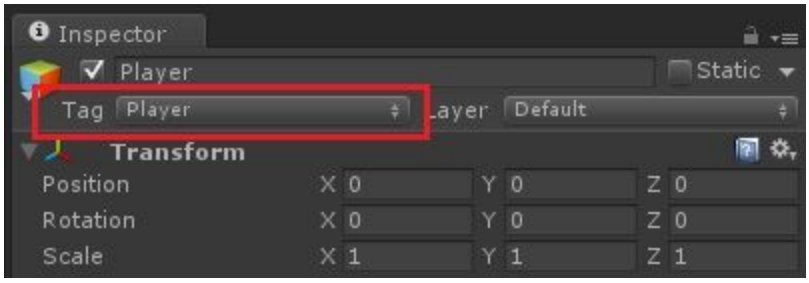


Open the **PlayerCamera** scene and select the **GameController** gameobject.

- **Scenes To Load At Start** should be set to: PlayerCamera, Sun, RoomA
- **Persistent Scenes** should be set to: PlayerCamera, Sun



- **Important!** Ensure the Player gameobject has the **Player** tag set:



You are now ready to play the game, and each room will be loaded additively and asynchronously and any scene that is not persistent nor visible will be automatically unloaded.

- In the demo, hold “Alt” to rotate the camera and click to start or stop movement.

Additional Support

For a video tutorials related to this asset, please click here:
Intro: *https://www.youtube.com/watch?v=dbDAmuTH5sw*

Join our VR community here for VR tutorials and videos:
https://www.youtube.com/nurfacegames/

For any questions or support, please email:
nurfacegames@gmail.com