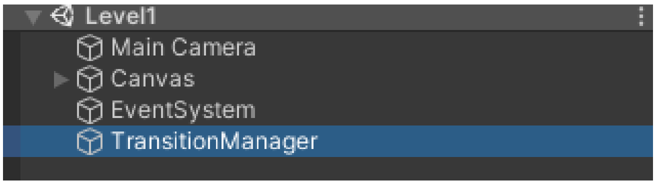
Easy Transitions

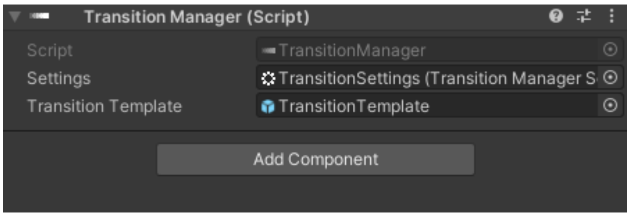
Documentation

Overview

Setting up the Transition Manager

First create a new GameObject. To make things more clear lets call Game Object „TransitionManager“.




Transition Manager (Script)

Script: TransitionManager

Settings: TransitionSettings (Transition Manager S

Transition Template: TransitionTemplate

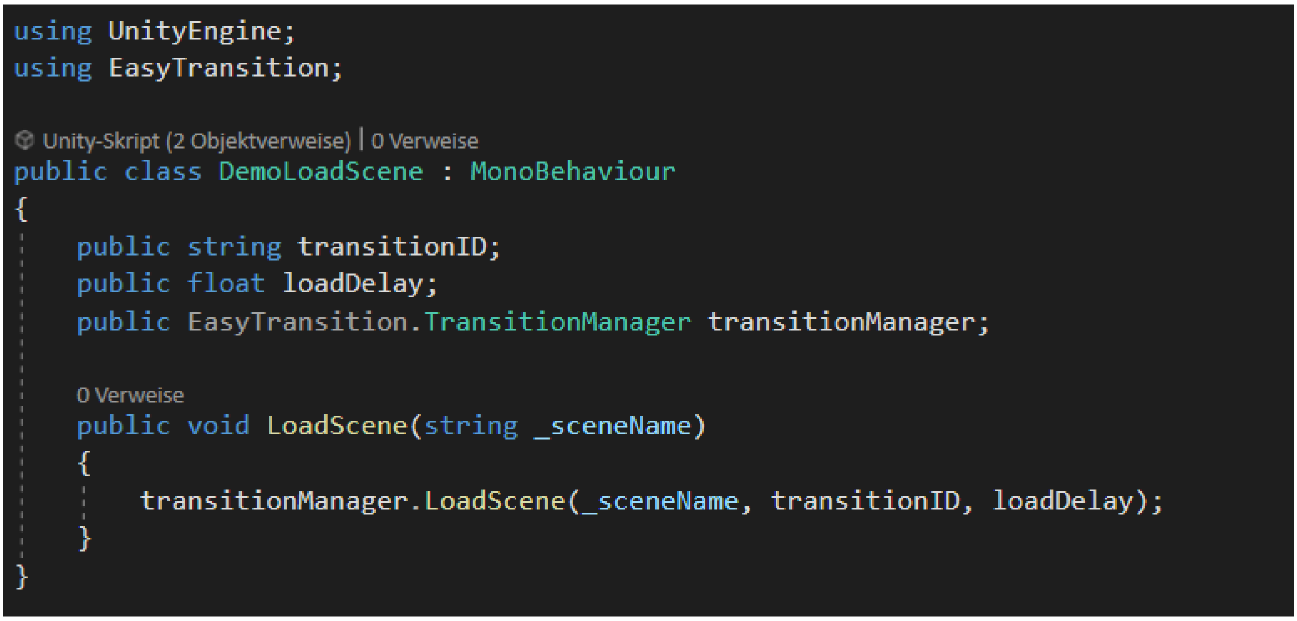
The second step is to add the **TransitionManager** component to your Game Object.

Changing the Scene

To change the scene with a premande or custom transition you can use the **TransitionManager.LoadScene(** *string SceneName, string TransitionID, float LoadDelay* **);**

function inside the TransitionManager class.

Just call the function from a button or other script and it will change the scene with the given transition.

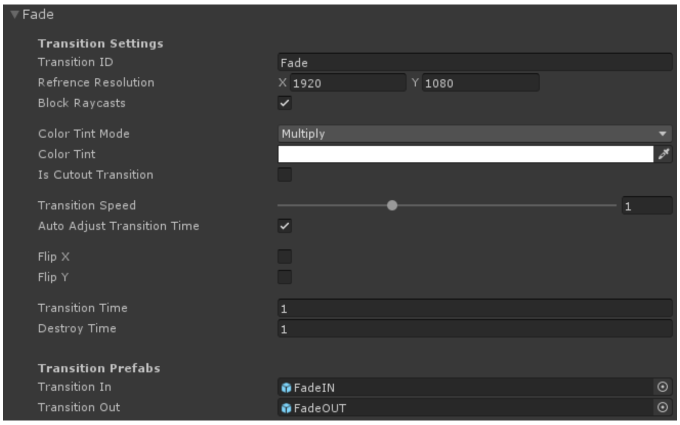

```
using UnityEngine;
using EasyTransition;

Unity-Skript (2 Objektverweise) | 0 Verweise
public class DemoLoadScene : MonoBehaviour
{
    public string transitionID;
    public float loadDelay;
    public EasyTransition.TransitionManager transitionManager;

    0 Verweise
    public void LoadScene(string _sceneName)
    {
        transitionManager.LoadScene(_sceneName, transitionID, loadDelay);
    }
}
```

Changing the Transition Manager Settings

To change the premade Transition Manager Settings locate the „Easy Transition“ folder in your project. Click on the „Transition Settings“ scriptable file. Now you have full control over the transitions in your game.



▼ Fade

Transition Settings

Transition ID	Fade
Refrence Resolution	X 1920 Y 1080
Block Raycasts	☑
Color Tint Mode	Multiply
Color Tint	⬜
Is Cutout Transition	□
Transition Speed	1
Auto Adjust Transition Time	☑
Flip X	□
Flip Y	□
Transition Time	1
Destroy Time	1

Transition Prefabs

Transition In	FadeIN
Transition Out	FadeOUT

Transition ID: Is the name of the transition. With it you can call it from the Transition Manager class.

Refrence Resolution: Is the original resolution of the canvas you made it in. For most transitions this would be **1080p** or **(1920, 1080)** but if you have made your transition with another screen resolution you can change that here.

Block Raycasts: Is for blocking the interaction with the UI on screen while the transition is running. If you set it to **false** you can still interact with the UI on screen, even without having to wait for the transition to end.

Color Tint Mode: Changes the way of the tint mode. **Multiply:** The tint color gets multiplied with the color of the transition. **Add:** The tint color gets added to the color of the transition.

Color Tint: This changes the color of the transition based on the **Color Tint Mode**. If you want to change the color of a black transitions use **Add** mode. If the transition is allready white use **Multiply** to change the color of the transition.

Is Cutout Transition: This must be set to true if the transition uses the **CutoutMaskUI** component. Changing the color is disabled when this option is set to true. If changing the color of the transition is needed, directly change it in the transition prefabs.

Transition Speed: Changes the speed of the animator components on the transition. To be able to change the transition speed with this value your transition must contain a **Animator** component.

Auto Adjust Transition Time: Automatically changes the **Transition Time** and **Destroy Time** value based on the set **Transition Speed** value.

Flip X: Flips the transition on the x axis.

Flip Y: Flips the transition on the y axis.

Transition Time: Is the time between the transition start and the scene switch in seconds. Depending on the type of transition you‘ll have to change this value.

Destroy Time: Is the time between the scene switch and the transition instance getting destroyed in seconds. Depending on the type of transition you‘ll have to change this value.

Transition In: Is the transition prefab for before the scene switch.

Transition Out: Is the transition prefab for after the scene switch.

Functions and Properties

Transition Manager

Function	Parameters	Description
LoadScene	string SceneName, string TransitionID, float LoadDelay	This function switches the scene with the given parameters. (Scene Name)
LoadScene	Int SceneIndex, string TransitionID, float LoadDelay	This function switches the scene with the given parameters. (Scene Index)
GetSceneIndex	string SceneName	Returns the index oft he given scene name

Property	Type	Description
transitionManagerSettings	**TransitionManagerSettings**	This variable stores all the transitions and settings. You can change them at any time.
transitionTemplate	**GameObject**	This function switches the scene with the given parameters. (Scene Index)
multiplyColorMaterial	**Material**	(**private**) The material for the multiply color mode.
addColorMaterial	**Material**	(**private**) The material for the add color mode.

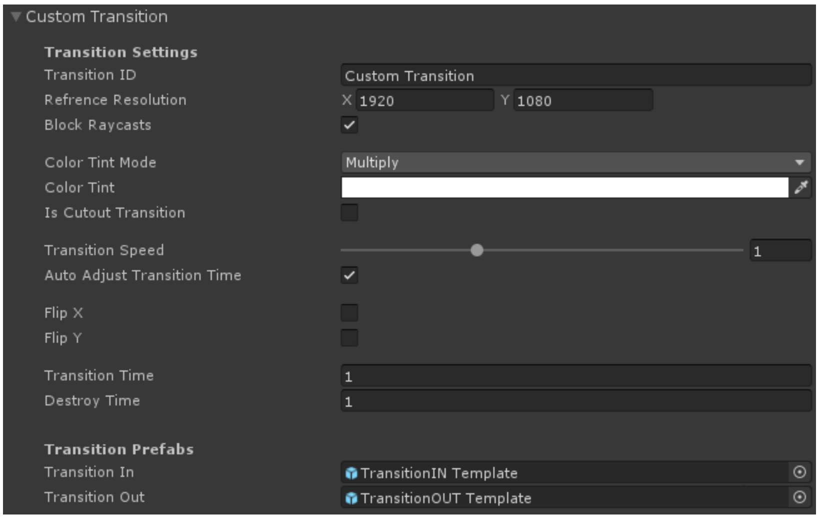
Transition Settings

Property	Type	Description
transitionID	string	This variable stores the name oft he transition. With it you can access it from the Transition Manager.
refrenceResolution	**Vector2**	This defines the resolution of the canvas the transition was made in. Normally this is (1920, 1080) but this could change for some transitions.
blockRaycasts	bool	If this bool iss et to true, any interactions with the UI on screen will be blocked until the transition is over.
colorTintMode	**ColorTintMode**	Sets the color mode to either multiply or add.
colorTint	**Color**	Changes the color of the transition based on **ColorTintMode**.
isCutoutTransition	bool	Set to true if there is a **CutoutMaskUI** component on the transition.
transitionSpeed	float	Changes the speed of the **Animator** components on the transition.
autoAdjustTransitionTime	bool	Automatically adjusts the **Transition Time** and **Destroy Time** values based on the **Transition Speed** value.
flipX	bool	Flips the transition instance on the x axis.
flipY	bool	Flips the transition instance on the y axis.
transitionTime	float	The time in seconds between the transition start and the scene switch.
destroyTime	float	The time when the transition object gets destroyed after the scene load.
transitionIn	**GameObject**	This variable stores a prefab oft the In animation of the transition.
transitionOut	**GameObject**	This variable stores a prefab oft the Out animation of the transition.

Included Transitions

Transition ID	Description
Fade	A simple black screen fade transition.
CircleWipe	A simple circle wipe transition.
LiniarWipe	A simple liniar wipe transition.
RectangleGrid	A more complex transition using many rectangles on a grid.
DoubleWipe	Similar to the liniar wipe transition but using two rectangles.
DiagonalRectangleGrid	Similar to the rectangle wipe transition but diagonal.
RectangleWipe	Similar to the circle wipe transition but using a rectangle instead of a circle.
VerticalCurtain	A simple vertical curtain transition.
HorizontalCurtain	A simple horizontal curtain transition.
Brush	A more complex transition where a brush paints across the scene.
PaintSplash	A more complex transition with many paint splashes on the screen.
Noise	A simple noise dissolve/resolve transition.

Custom Transitions



Custom Transition

Transition Settings

Transition ID: Custom Transition
Refrence Resolution: X 1920 Y 1080
Block Raycasts: ✓

Color Tint Mode: Multiply
Color Tint: □
Is Cutout Transition: □

Transition Speed: 1
Auto Adjust Transition Time: ✓

Flip X: □
Flip Y: □

Transition Time: 1
Destroy Time: 1

Transition Prefabs

Transition In: TransitionIN Template
Transition Out: TransitionOUT Template

To add a custom transition locate the **TransitionSettings** file in the **EasyTransitions** folder. Click on the plus symbol on the bottom of the list.

To name your transition change the **Transition ID**.

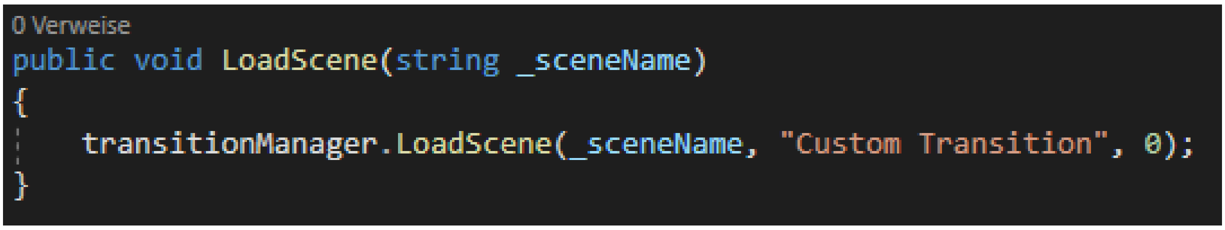
Change the **Refrence Resolution** to the resolution the canvas you made your transition in. If you made the transition with the HD (1920, 1080) resolution you can leave this setting at its default values.

Change the **Transition In** and **Transition Out** with a prefab of an animated image. Make sure to not have a canvas in your prefab but only the image.

Change the **Transition Time** to the time in seconds where the cut (scene change) of your transition should happen. And the **Destroy Time** to when the transition instance should get destroyed after switching the scene in seconds.

The other settings are not that important for the beginning. You can play around with them to perfectionate your transition.

To use your custom transition just call the **LoadScene();** function from your **TransitionManager** instance. Make sure to swap the **„Custom Transition“** with your transition id.



```
0 Verweise
public void LoadScene(string _sceneName)
{
    transitionManager.LoadScene(_sceneName, "Custom Transition", 0);
}
```