REMOTE REPLAY

For Unity

V0.2, 2024-01-28

By RealitySims

INTRO

Remote Replay for Unity is a simple, lightweight system to record & analyze replays from Unity 2D games. You can integrate the system into your mobile games in just a few steps, and watch how your players play your game almost in real time.

HOW IT WORKS

Remote Replay records visual positions of sprites and camera for a period you can define (onboarding, challenging level or boss etc.). ***It does not record videos in the traditional sense (such as mp4).*** Once finished, it either caches the replay(if your player is offline) or connects to Firebase and sends the replay to Firebase storage. If the player is offline it simply tries to send their replay later.

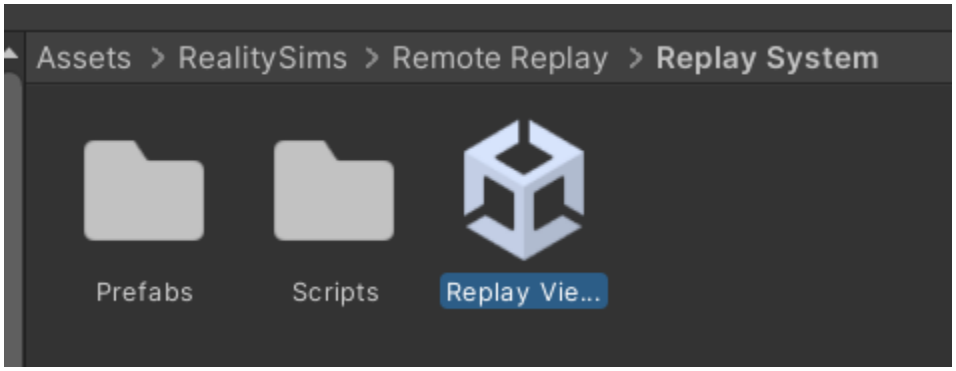
The replays are very lightweight (optimized data structure & gzip compression) - though it also depends on how you configure Remote Replay. For example, you can change the fps of recording or which data you record (rotation, Z axis etc.) which might affect the replay size considerably.

Later, when you want to watch your replays, check out Firebase storage for newly arrived replay files, copy & paste the filename into our replay player scene, and you're ready to watch.

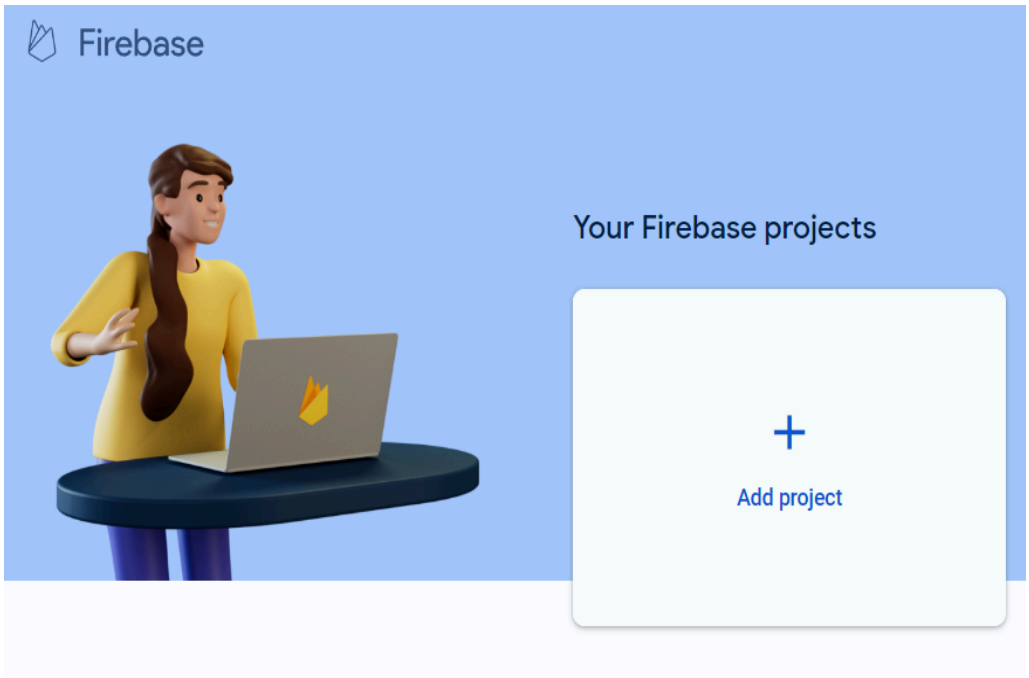
QUICKSTART

1) Add the package to your Unity Project, it will contain 2 folders:

- One called "*Examples*" with a couple of games which includes the replay recorder.
- Another folder called "*Replay System*" which also contains the replay player scene called "*Replay Viewer*"



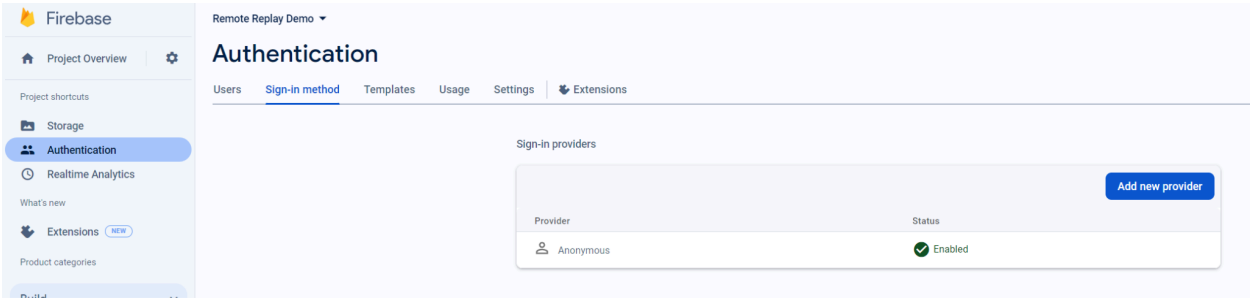
2) Create a new Firebase project at https://console.firebase.google.com



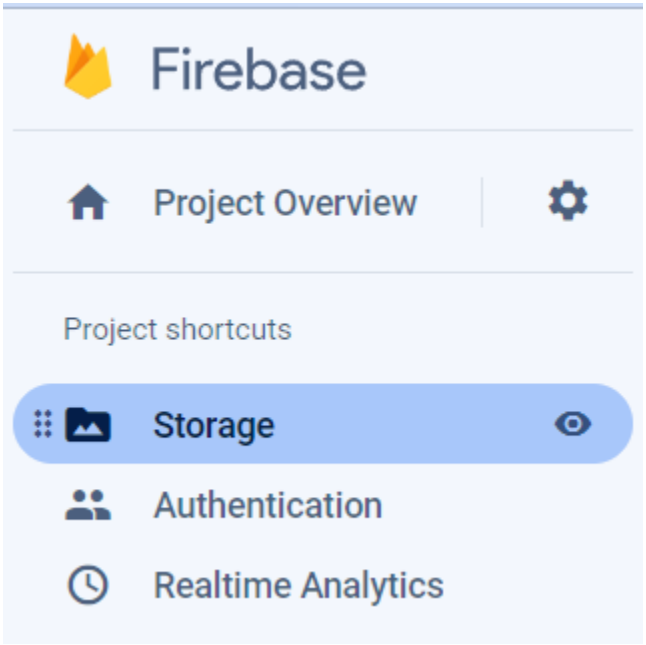
Please follow Google Manual (https://firebase.google.com/docs/unity/setup) if unsure how you finish setting up a Firebase project. This usually takes only 5 to 10 minutes. Don't forget to download google-services.json and put it into your Assets folder.

You might need to restart Unity, especially in case of subsequent Firebase errors.

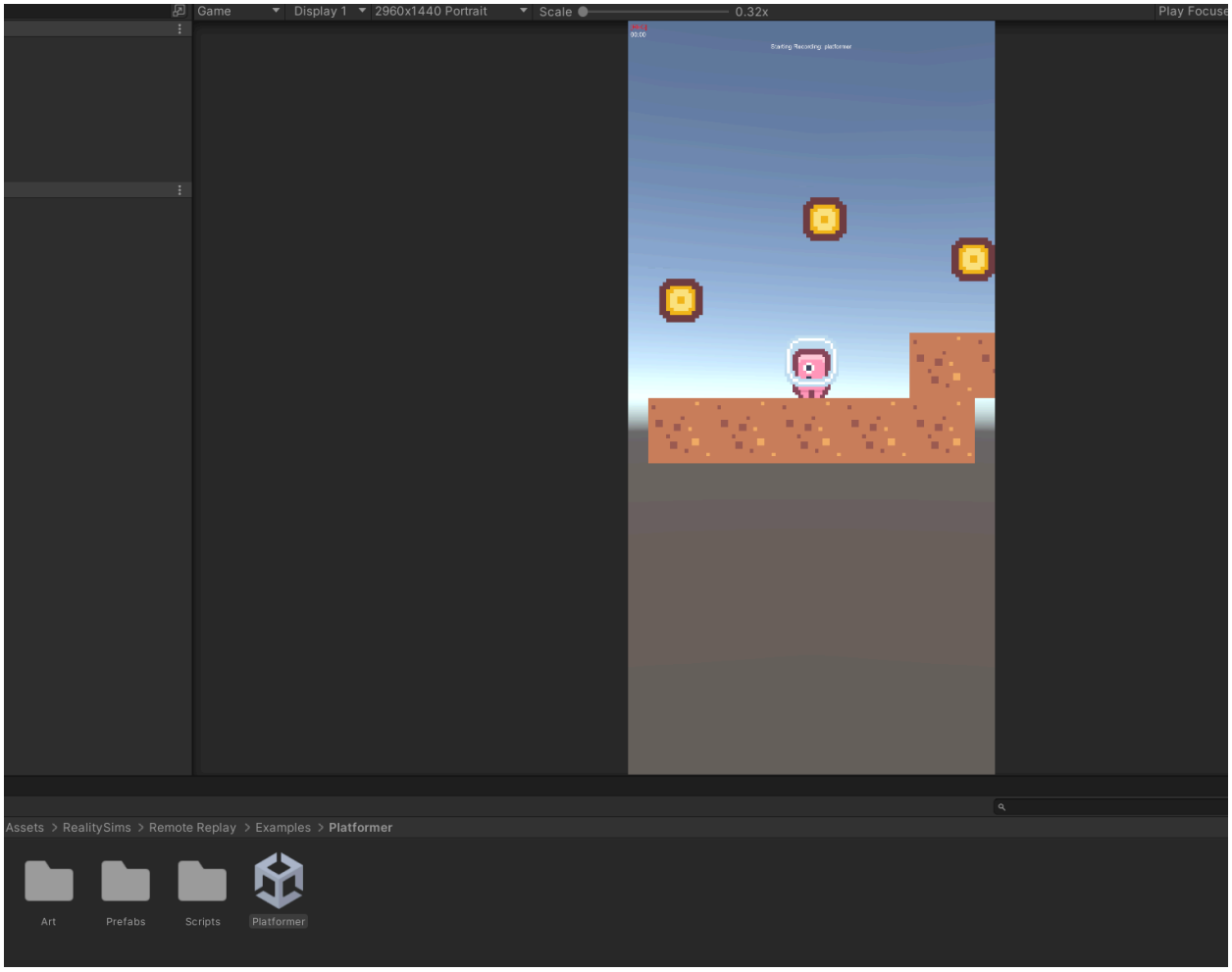
3) Enable Anonymous login in Firebase, if not enabled yet

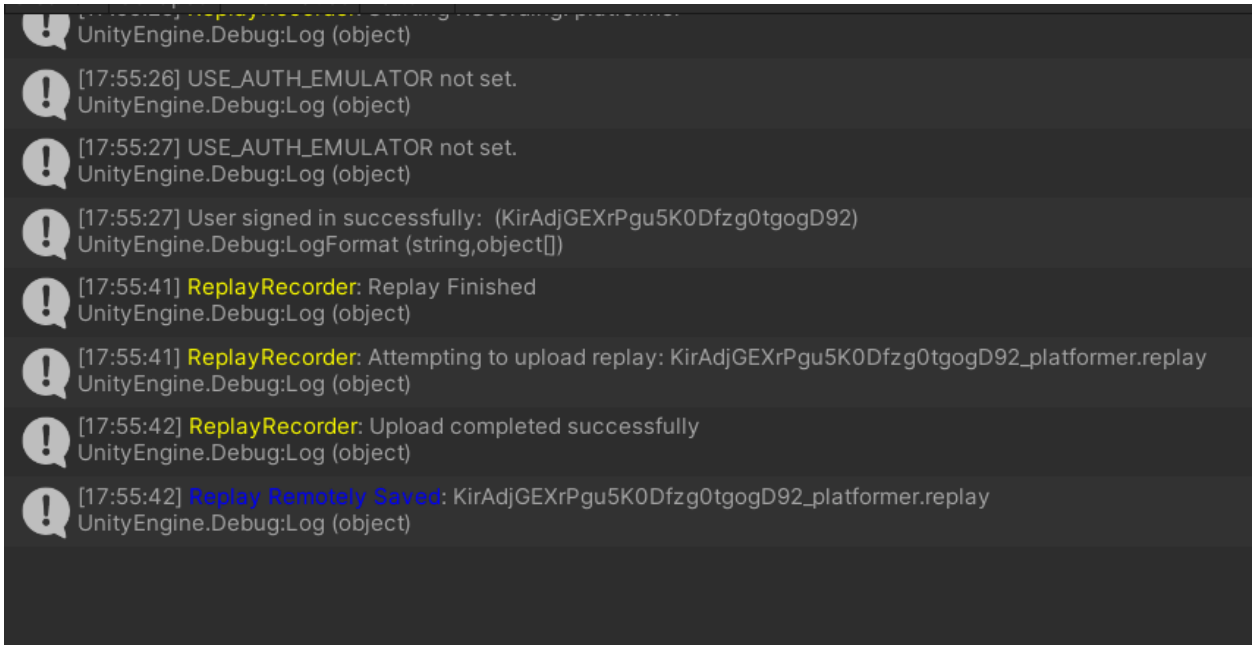


4) Enable Storage in Firebase, if not enabled yet. This is where your replays will be uploaded.

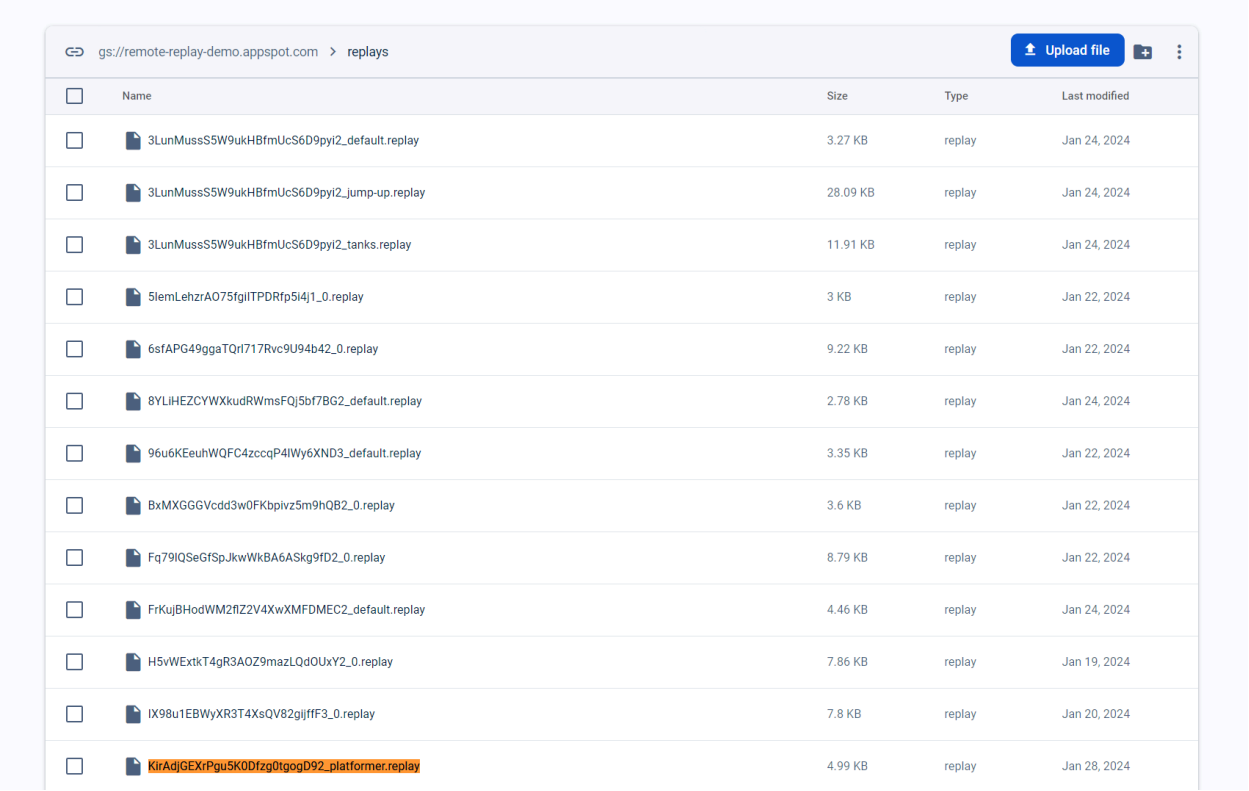


5) Open a sample game scene (*Examples/Platformer*), play for a couple seconds. You should see a red indicator in the top left corner, confirming a replay is being recorded for several seconds. After the recording finishes, you should see a message in the console once the video has been recorded & uploaded to Firebase.


Game
Display 1
2960x1440 Portrait
Scale
0.32x
Play Focuse
Assets > RealitySims > Remote Replay > Examples > Platformer
Art
Prefabs
Scripts
Platformer


UnityEngine.Debug:Log (object)
[17:55:26] USE_AUTH_EMULATOR not set.
UnityEngine.Debug:Log (object)
[17:55:27] USE_AUTH_EMULATOR not set.
UnityEngine.Debug:Log (object)
[17:55:27] User signed in successfully: (KirAdjGEXrPgu5K0Dfzg0tgogD92)
UnityEngine.Debug:LogFormat (string,object[])
[17:55:41] ReplayRecorder: Replay Finished
UnityEngine.Debug:Log (object)
[17:55:41] ReplayRecorder: Attempting to upload replay: KirAdjGEXrPgu5K0Dfzg0tgogD92_platformer.replay
UnityEngine.Debug:Log (object)
[17:55:42] ReplayRecorder: Upload completed successfully
UnityEngine.Debug:Log (object)
[17:55:42] Replay Remotely Saved: KirAdjGEXrPgu5K0Dfzg0tgogD92_platformer.replay
UnityEngine.Debug:Log (object)

6) Copy & paste the replay filename either from Unity console, or from Firebase storage, which at this point should contain the new replay filename, in the “replays” folder.

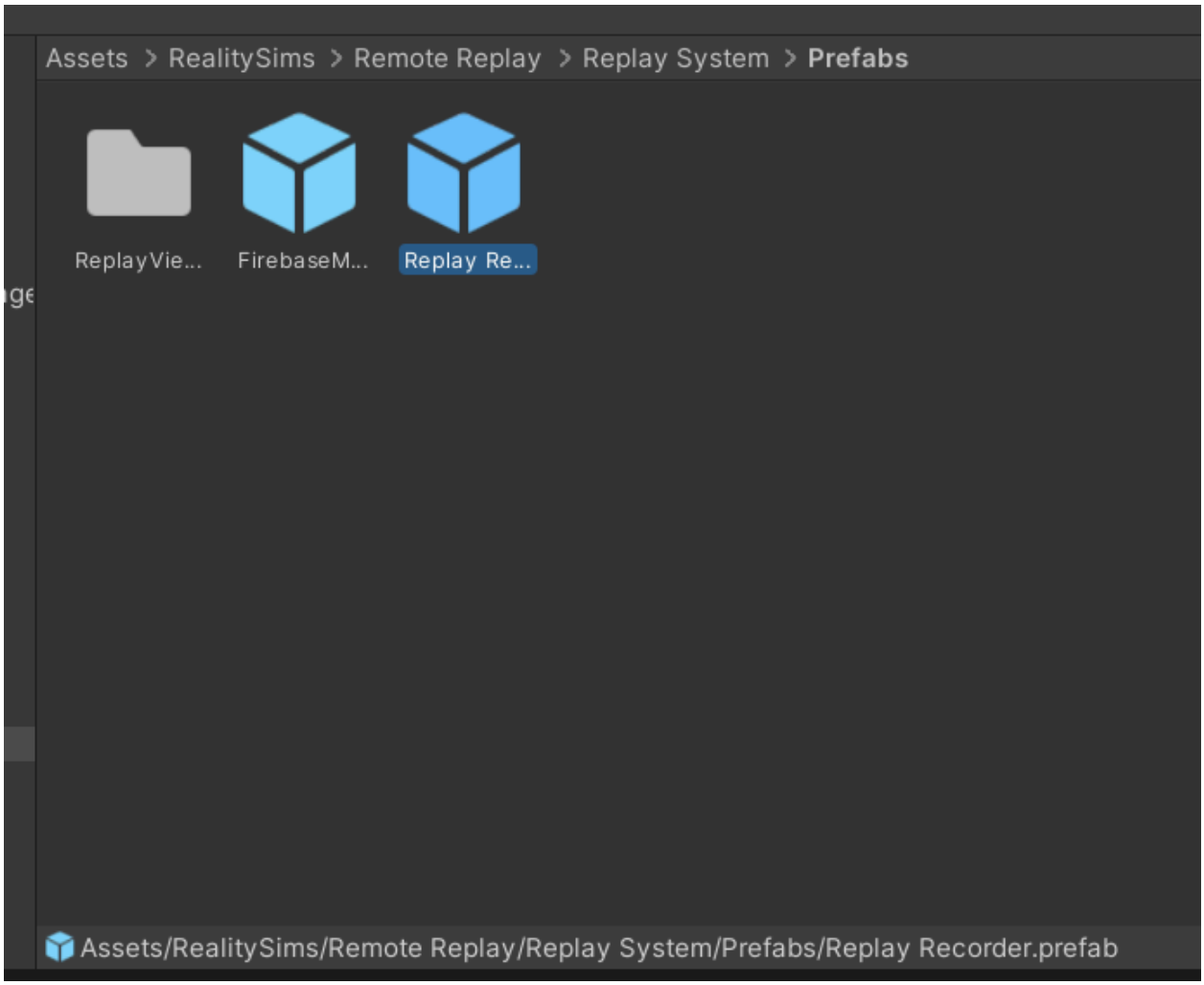


7) Launch the replay recorder scene in *“Replay System/Replay Viewer.scene”*, copy & paste the filename and watch the replay!

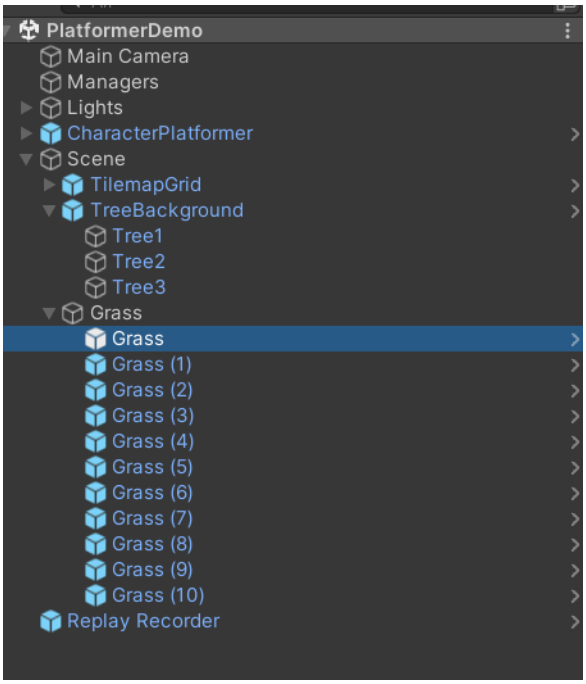
INTEGRATING REMOTE REPLAY INTO YOUR GAME

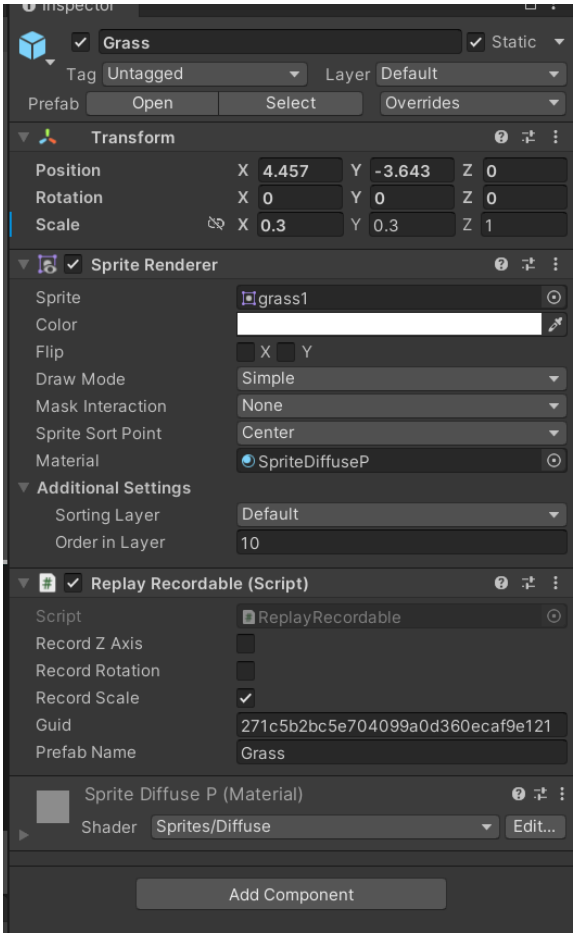
In order to integrate the replay system into your game, you need to do the following:

1) Drag the “Replay Recorder” from the Prefabs folder into your scene’s hierarchy.



2) Anything you want to record needs to be a prefab, **and** it needs to have a “ReplayRecordable” script attached to it. The script needs to be attached “inside” of the prefab, not from the scene view, and it can reside on top of the prefab’s hierarchy, in case you’re using nested game objects inside the prefab.

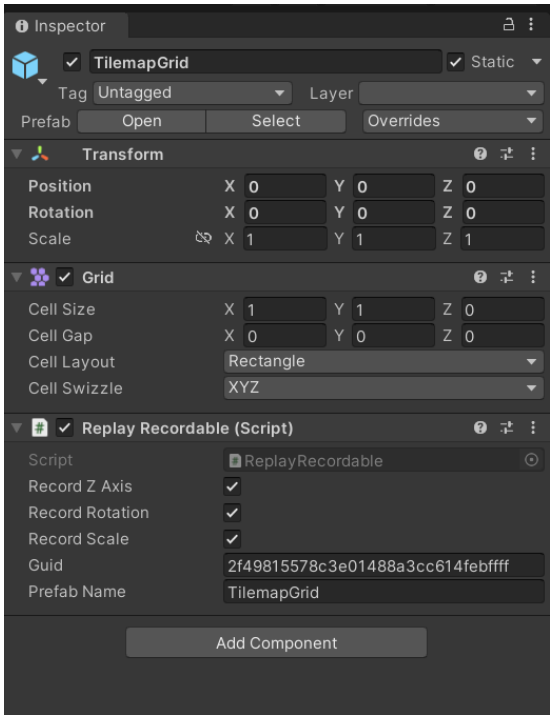

PlatformerDemo
Main Camera
Managers
Lights
CharacterPlatformer
Scene
TilemapGrid
TreeBackground
Tree1
Tree2
Tree3
Grass
Grass
Grass (1)
Grass (2)
Grass (3)
Grass (4)
Grass (5)
Grass (6)
Grass (7)
Grass (8)
Grass (9)
Grass (10)
Replay Recorder


Grass
Static
Tag Untagged
Layer Default
Prefab Open Select Overrides
Transform
Position X 4.457 Y -3.643 Z 0
Rotation X 0 Y 0 Z 0
Scale X 0.3 Y 0.3 Z 1
Sprite Renderer
Sprite grass1
Color
Flip X Y
Draw Mode Simple
Mask Interaction None
Sprite Sort Point Center
Material SpriteDiffuseP
Additional Settings
Sorting Layer Default
Order in Layer 10
Replay Recordable (Script)
Script ReplayRecordable
Record Z Axis
Record Rotation
Record Scale
Guid 271c5b2bc5e704099a0d360ecaf9e121
Prefab Name Grass
Sprite Diffuse P (Material)
Shader Sprites/Diffuse Edit...
Add Component

You can tick / untick which properties you want to record, depending on your game. Currently we support recording Z axis, rotation, scale. Position is recorded automatically.

SUPPORTED RECORDABLES

Currently, we record anything with a transform in the replay. However, we only support viewing *Sprites* with the SpriteRenderer component, as well as *Tilemaps*. To record a tilemap, you can attach the recordable to the gameObject with Grid component, like this:



If something is not supported to be viewed but you're still recording it, you will see a filled circle with the name of the prefab as a stand in.

Happy replaying!

***Support link*: <TBD, Unity link>**