

Project Name: **Spawner Free**
Modified Date: **15 Nov. 13**
Author: **Corrupted Smile Studio**
Version: **2.0**
Source: **C#**
[More info Link](#)



Contents

Overview	3
Issues	3
Short Tutorial	3
Class Overview	4

Overview

Provides an easy to use spawn system for spawning enemy units, contains a few useful native spawn modes (continual, once, wave, time wave and time split wave).

Issues

No known issues. No testing has been made with Mac computers but no errors are anticipated for Macs.

Short Tutorial

1. Move the **Gizmos** folder to the root of the **Assets** folder
2. Place the Spawner.cs MonoBehaviour on an empty GameObject.
3. Assign unit prefabs in the Assign Units foldout.

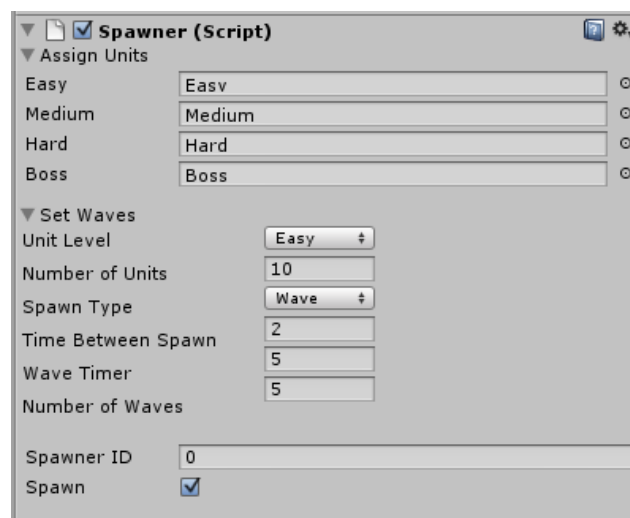


Figure 1 Inspector

4. To any units you want to spawn, you will need to add the **SpawnAI.cs** component. You can subclass it to better suit your requirements and use that component instead.
5. Set the **Unit Level** for the waves.
6. Set the **Number of Units** for each wave
7. Set the **Spawn Type**
8. Set the **Time Between Spawn**
9. If spawn type is set to **normal, once or just wave** skip to step **11**
10. Set the **Wave Timer**, which is the time between waves (with Timed Wave, it is the time between the start of waves. Whereas with Time Split Wave, it is the time since the last unit dies).
11. Set the **Number of Waves**. This is how many waves you want to eventually have spawned.
12. You will need to set **Spawner ID** manually.
13. If **Spawn** is set to true, it will start spawning as soon as the game is loaded.

- 14.If you have [Pool Manager by Path-o-Logical](#) package (a GameObject pooling system). Continue on with the steps, if not stop here.
- 15.Within the **InstanceManager.cs** file simply uncomment the first line within the file. (**#define POOLMANAGER_INSTALLED**) this will allow the system to make use of Pool Manager.

Class Overview

Within this release there is an included zip file called Documentation. Unzip this file **OUTSIDE** your Unity project to view a [Doxygen](#) generated class diagrams and method overviews.