

Wave One

Thank you for considering this asset. The goal of this asset is to allow developers for quick configuration of enemy waves whether you are making a CoD Zombies clone or a tower defense game. It contains various variations of start points, spawners and end points to suit your needs. Also the way this asset was developed, it is quite straightforward if this asset has been used but you want to include your own variations in the system. In this PDF I will go over the different features and explain what they do so you can start building your game.

If you are more interested in the scripting reference please refer to the GitHub wiki.

Contents

Note	3
Editor	3
Game events.....	3
Scripts	3
WaveManager	3
WaveConfigurator	3
Start points	3
Start point pickers	4
Spawners	4
Spawner pickers	5
End points.....	5
Formations	5
Demo	5

Note

Before adding your own variations to the asset or updating to a new version, please make a backup of your project so your configurations are not lost.

Editor

This asset contains custom editor scripts. If you have other custom editors in your project you can merge the contents of this Editor folder with the one you currently have. If you want to keep things a bit organized you could make a folder for this asset and put the editor contents in that.

Game events

This assets uses scriptable objects to create game events. These event are used to bypass hard references within code and can be customized to pass any data we want. For this project I only needed to pass 'bool' and 'int' events, however you can easily create another data type if needed. You can refer to the scripting reference if you need help setting up your own data types.

Scripts

These management scripts are by no means required in the scene for this asset to work correctly. However they do add quality of life improvements and better workflow. Also by using the management scripts you are less prone to error by missing scripts.

WaveManager

The WaveManager keeps a refence to all the WaveConfigurator scripts in the scene. This can also be turned off in the editor to select specific WaveConfigurator scripts instead. This is particularly useful in an open world settings where there may be multiple different events going on which all have their own WaveManager. This manager also keeps track of the following important things:

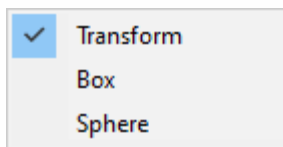
- Have the spawners for this manager started?
- How many spawners have finished their wave?
- Have all spawners completed their wave?

Along with that it also has a couple public methods. One of them is used to start the waves for each of the given WaveConfigurator scripts. The other is used to update the states of the previous 3 points.

WaveConfigurator

The WaveConfigurator allow users to quickly and easily put together wave components. There are multiple enums the user can choose from to design how their wave is going to function. The following chapters will explain what each enum actually does.

Start points

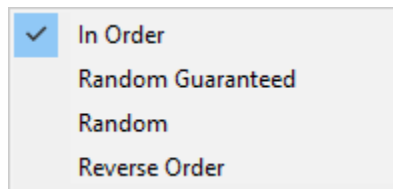


Start points indicate where enemies are able to spawn.

- Transform: *allows the user to specify certain Vector3 points in world space as transforms.*

- **Box:** allows the user to specify certain Vector3 points in world space as transforms. Also has a Vector3 which indicates what size the shape should be. When spawning a random point inside the open space is chosen. Also allows for a minimum distance from the center and the option to max all axis (the point moves towards corners) or only one of the axis (the point moves towards edges).
- **Sphere:** allows the user to specify certain Vector3 points in world space as transforms. Also has a float which indicates what size the shape should be and a float which indicates the minimum distance from the center. When spawning a random point inside the open space is chosen.

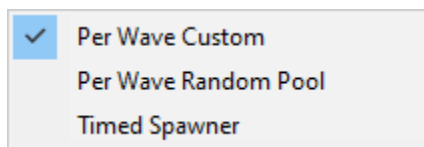
Start point pickers



Start point pickers indicate how the start points should be picked.

- **In Order:** the start points will be chosen from top to bottom as displayed in the editor.
- **Random Guaranteed:** a random start point will be chosen each time, however guaranteeing that every start point in the list will be chosen at least once before a second time.
- **Random:** a random start point will be chosen each time.
- **Reverse Order:** the start points will be chosen from bottom to top as displayed in the editor.

Spawners

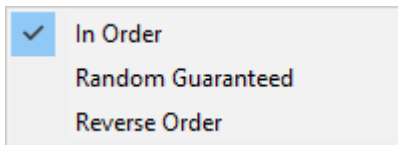


Spawners indicate how the outlook of the wave will be.

- **Per Wave Custom:** with this spawner you can specify how many waves you want. Per wave you can specify how many enemies you want to spawn and the amount of deployments. For each enemy there are field to specify how many of this enemy you want to spawn and the group size.
- **Per Wave Random Pool:** with this spawner you can specify how many waves you want. Per wave you can specify which enemies you want to spawn, the size of the pool for this wave and the amount of deployments. In the enemies list other than indicating enemies, you can also specify the group size for each enemy. A random enemy will be chosen each time.
- **Timed Spawner:** this spawner does not contain waves. Instead you can specify for how long this spawner will be running (in seconds). It is also possible to use this spawner as an endless spawner by setting the time to 0 seconds. In the enemies list you can again specify the enemies and group sizes for each with this time a weight addition. These weights are relative to each other meaning a weight of 2 will theoretically spawn twice as much a weight of 1.

Spawner pickers

Spawner pickers indicate how the enemies will be chosen. This option is only available with the



PerWaveCustom spawner.

- In Order: *the enemies will be chosen from top to bottom per wave as displayed in the editor.*
- Random Guaranteed: *the enemies will be chosen randomly per wave, however guaranteeing that each enemy will be chosen before the wave/deployment ends.*
- Reverse Order: *the enemies will be chosen per from bottom to top per wave as displayed in the editor.*

End points

End points can be either disabled or enabled. End points assume that you are using the Unity nav mesh components for the path finding. If this is not the case you will have to make some changes to the asset. For more information please refer to the scripting reference.

Formations

When spawning enemies they are always spawned in formation (even for solo enemies, which will just be the center). For now only the square formation is available but you can easily add your own. For more information please refer to the scripting reference.

Demo

The demo folder contains a couple different things. It contains prefabs for some enemies and the materials for them. It also contains the NavMeshComponents used for this project because the enemies in this demo use the pathfinding provided by Unity. Furthermore, it contains some demo scenes and scripts. DemoScene_Spawners has a WaveManager with 3 different WaveConfigurators showcasing the different spawners along with a canvas showing how to link that up to the WaveManager.

DemoScene_MultipleWaveManagers is a scene which shows how to WaveManagers to the different WaveConfigurators and how to link up the buttons.