

# Spawn Factory Documentation

## Site and Socials

For support or to hang out, join us on [Discord!](#)

External documentation and more on our [Website!](#)

For updates and more follow us on [Twitter!](#)

## Getting Started

### Adding an Enemy Spawner

- Use the provided prefab or create an empty GameObject on your scene and add the **EnemySpawner** script to it.
- Set the ground layer(s) in the Spawner Settings tab. This is required to add nodes to the spawn area and for the enemies to spawn on.
- In the Player Detection tab, use Spawn on Load or set the player layer(s). This is required to detect when the player is within range to begin spawning.

How to keep the spawner working correctly if your enemy doesn't disable or destroy right away (not needed with Timer spawn type)

- Attach the **SpawnerCommunicator** script to your enemy. This is necessary if your enemy doesn't disable or destroy right away when dying (and only then)
- You will need to call the EnemyKilled() function from your ai script when the enemy is killed. **Example usage:**

```
public void CheckDied() {  
    if(health <= 0) {  
        isDead = true;  
        GetComponent<SpawnerCommunicator>().EnemyKilled();  
    }  
}
```

### Adding Pooling

- Use the provided prefab or create an empty GameObject on your scene and the PoolingManager script to it.
- In order for an enemy spawner to use pooling you must set the Spawning Method in the Spawner Settings tab to pooling on the spawner. The spawner will automatically add all enemies to the pool.
- Be sure your enemies disable when killed instead of being destroyed.

## How to Pool Objects NOT in a Spawner

- If you want the pool to persist between scenes add the object onto the **PoolingManager**. If you otherwise only want to pool an object in a single scene, create an empty GameObject and add the **PoolObjectsInScene** script to it (or use the prefab).
- Create a category in the Categories tab and add items to your categories in the Pooling tab (these are only for better organization).
- Each pool must have a unique tag or you will receive mixed objects.
- You will need to call the function [GetPooledObject](#) when it's being spawned and add the PooledObject script to your object (this requires you disable your object when you want to return it to the pool). Alternatively you may call [ReturnObjToPool](#) in your script to return it when you're ready.

## Using Pooled Object

- Add the **PooledObject** script to your object in the pool.
- Make sure the pool tag on **PooledObject** matches the unique tag in **PoolManager** or **PoolObjectsInScene** to reference the correct pool.
- Objects in enemy spawners do not need this component.

## Enemy Spawner Settings Details

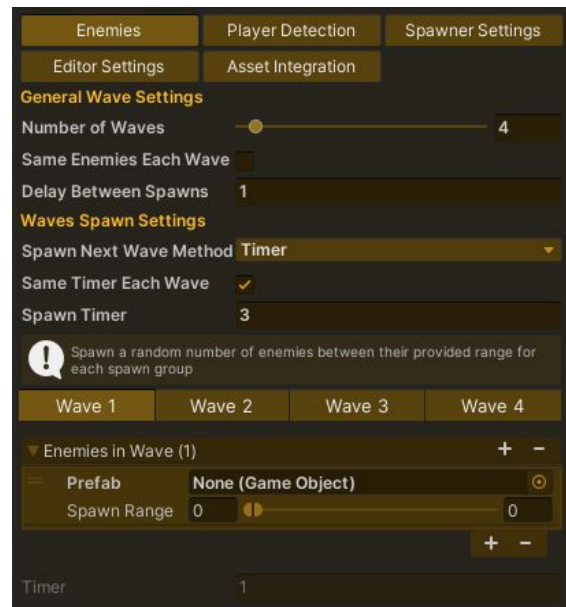
### Spawn Area in Scene View

- **Insert New Node:** Ctrl+Left Mouse Button in scene view
- **Delete Node:** Ctrl+Shift+Left Mouse Button on a node in scene view



## Enemies Tab

- **Number of Waves:** the amount of groups of enemies this spawner will spawn
- **Same Enemies Each Wave:** if this is marked true, every wave will have the same group of enemies.
- **Delay Between Spawns:** How long (in seconds) to wait between each enemy spawn.
- **Enemies in Wave:** the group of enemies to spawn
  - **Prefab:** the enemy prefab to spawn
  - **Spawn Range:** a random amount of enemies between min-max range will spawn every group.



## Spawn Next Wave Method



**Timer:** the next wave will spawn based on a timer (and nothing else)

- **Same Timer Each Wave:** If this is marked true, every wave will spawn after the same amount of time (set by spawn timer).
- **Spawn Timer:** How long (in seconds) each wave will wait before spawning the next wave (only usable if Same Timer Each Wave selected).

**After Wave Killed:** the next wave will spawn when all of the current wave is dead

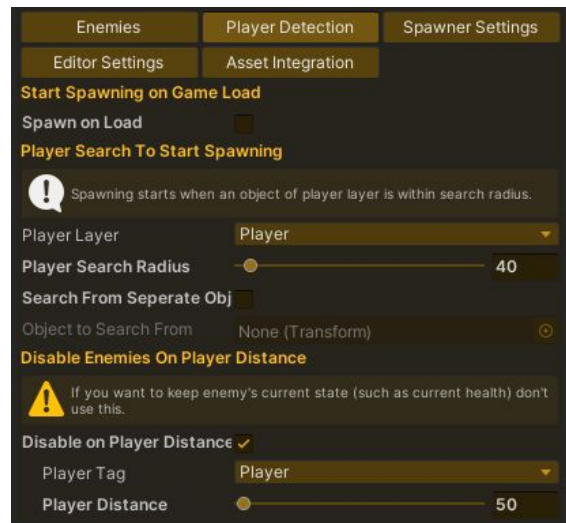
- **Delay Between Waves:** How long (in seconds) each wave will wait after all enemies are dead before spawning the next wave.

**Continuous Kills:** Enemies will spawn to a certain point, then after enough are killed more will spawn up to that point.

- **Max Enemies Out:** The maximum amount of enemies that may be out at one time.
- **Kills Before Spawn More:** How many enemies must die before more are spawned.
- **Delay Between Waves:** How long (in seconds) the spawning will pause between each wave (if this is 0, then waves aren't noticeable other than spawn order).

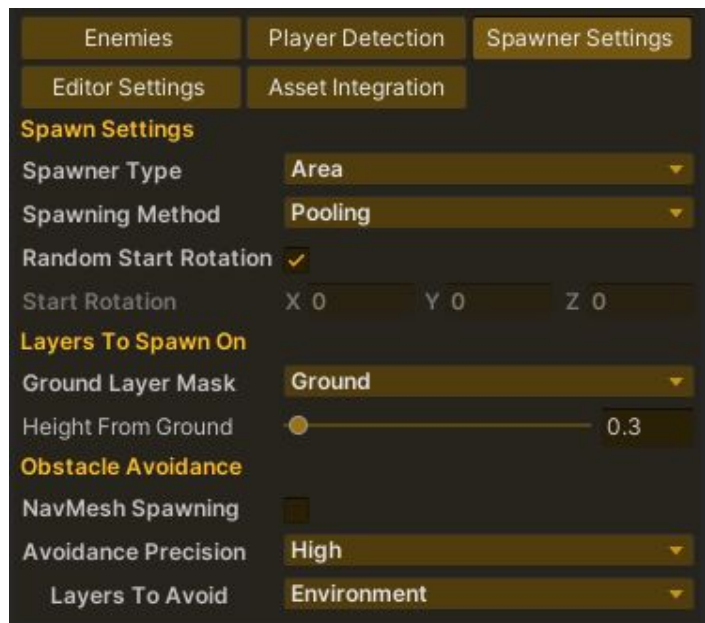
### Player Detection Tab

- **Spawn on Load:** When the game loads the spawner will start spawning (overrides all player search spawning).
- **Player Layer:** the layer used to search for players and start spawning.
- **Player Search Radius:** the radius around the spawner to search for players
- **Search From Separate Object:** This allows the spawner to detect the player from an entirely different object.
- **Object to Search From:** The object that will be used other than the spawner to detect if the player is near (near that object only).
- **Disable on Player Distance:** if this is marked true the enemies will disable automatically when players are more than "player distance" away and returned when a player is within distance again (does not return the same enemy).
- **Player Tag:** the tag used to find players on load to calculate their distance.
- **Player Distance:** the distance used to determine how far away players need to be for this to disable or re-enable.



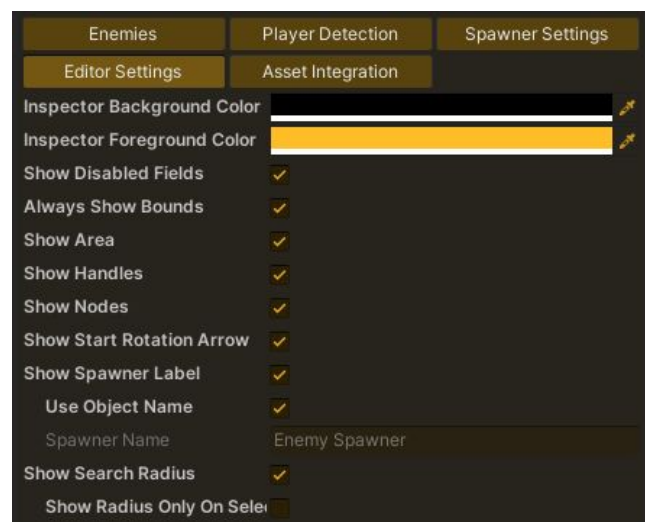
## Spawner Settings Tab

- **Spawner Type:** this specifies how enemies will spawn.
- **Spawning Method:** whether enemies are pooled or instantiated. Pooling is better for performance and just needs a pooling manager. Instantiated is much worse, but left in if you don't wanna pool.
- **Random Start Rotation:** Every enemy will spawn with a random starting rotation.
- **Start Rotation:** the rotation all enemies spawned by this spawner start at (this rotation is added to the spawner's rotation and has an arrow showing you).
- **Ground Layer Mask:** the layer nodes can be added to and enemies can spawn on (cannot spawn or add nodes without something of this layer underneath).
- **Height From Ground:** the height from the ground each new node will be added.
- **NavMesh Spawning:** if this is marked true enemies will only spawn on the NavMesh (may be combined with obstacle avoidance).
- **Avoidance Precision:** how many passes will be done to avoid obstacles (higher precision may use more performance) and if it can't avoid obstacles it won't spawn the enemy.
- **Layers To Avoid:** the layers with obstacles to avoid when spawning.



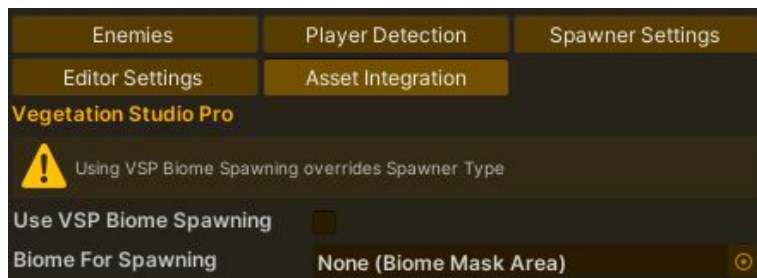
## Editor Settings Tab

- **Show Disabled Fields:** if marked true, disabled fields in the inspector will show as grayed out fields. If false they won't show at all.
- **Inspector Background Color:** the background color of the spawner's inspector
- **Inspector Foreground Color:** the foreground color of the spawner's inspector (affects no text other than labels).



- **Always Show Bounds:** if marked true, the boundaries of the spawner will show in scene view even when the spawner isn't selected, otherwise it only shows when spawner is selected.
- **Show Area:** if marked true the spawner outline will show in scene view.
- **Show Handles:** if marked true, the handles to move nodes will show in scene view.
- **Show Nodes:** if marked true, the sphere caps will show over nodes in scene view.
- **Show Start Rotation Arrow:** if marked true, the arrow indicating start rotation will appear in scene view.
- **Show Spawner Label:** if marked true, the spawner's name will appear in the middle of the spawner.
- **Use Object Name:** the spawner's name will be used as the label in scene view.
- **Spawner Name:** set your own custom spawner name to show as the label in scene view.
- **Show Search Radius:** if marked true, show wire sphere in scene view indicating player search radius.
- **Show Radius Only On Selected:** if marked true, show player search radius only when the spawner object is selected.

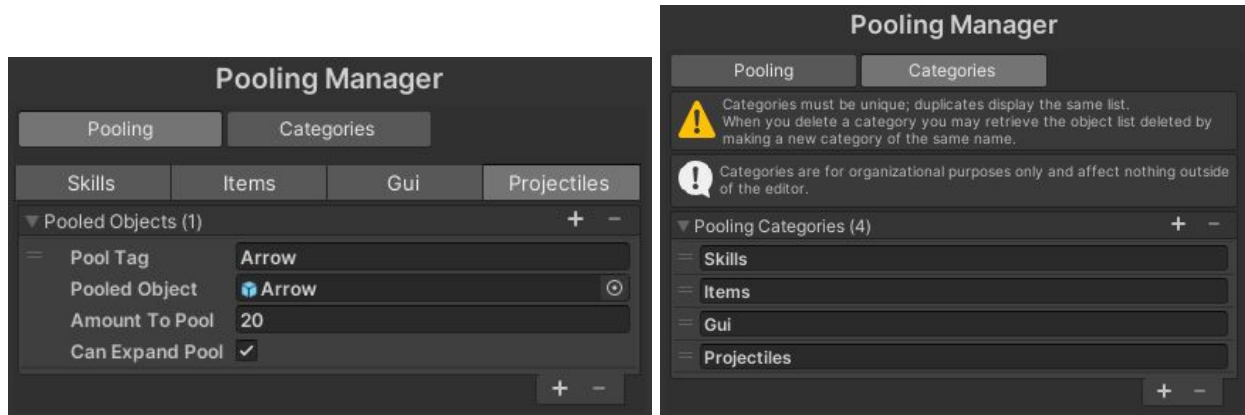
### Integration Tab (Only shown if you have an asset integrated with it)



### Vegetation Studio Pro

- **Use VSP Biome Spawning:** if marked true the enemy spawner's area will match VSP's biome exactly.
- **Biome For Spawning:** the VSP biome to match the spawner area to.

## Pooling Manager Settings Details



- **Categories:** Only for better organization of your pooled objects. Affects nothing else. Each category will have its own tab in Pooling with a list of pooled objects.
- **Pooled Objects:** the objects that will be pooled in the pooling manager.
  - **Pool Tag:** the unique tag used to determine which object to retrieve/return to pool.
  - **Pooled Object:** the object prefab to be pooled.
  - **Amount To Pool:** the initial amount of the object added to the pool.
  - **Can Expand Pool:** whether the pool is allowed to grow larger if all objects are currently in use.

## Custom Pooling in Scripts

### Important Property

public static PoolingManager instance - this holds the unique instance to your PoolingManager. There will only ever be 1 and it may be referenced from any script using PoolingManager.instance to call functions in the PoolingManager.

### Functions

- **Get an object from pool:** GetPooledObject(poolTag: string) | Returns a GameObject
- **Return an object to pool:** ReturnObjectToPool(poolTag: string, obj: GameObject) or ReturnObjectToPool(poolTag: string, obj: GameObject, timer: float)
- **Determine if tag is in pool:** CheckIfTagInPool(poolTag: string) | Returns a bool
- **Example:**  
GameObject obj = PoolingManager.instance.GetPooledObject("Arrows");  
This will return an object from the pool tagged Arrows.



## Pooled Object Settings Details

Pool Tag	<input type="text"/>
Destroy After Duration	<input type="checkbox"/>
Duration	<input type="text" value="0"/>

- **Pool Tag:** the unique tag used to reference the pool you want to return the object to when disabled.
- **Destroy After Duration:** if marked true the object will automatically disable and return to the pool after set Duration.
- **Duration:** the duration to wait before automatically returning the object to pool (only works if “destroy after duration” is true).

## Join the Community and Thank the Developer!

If you liked this project please write a review on the Asset Store!

Join us on discord: <https://discord.gg/pZMrHuZ> and share your feedback!

Thank You!