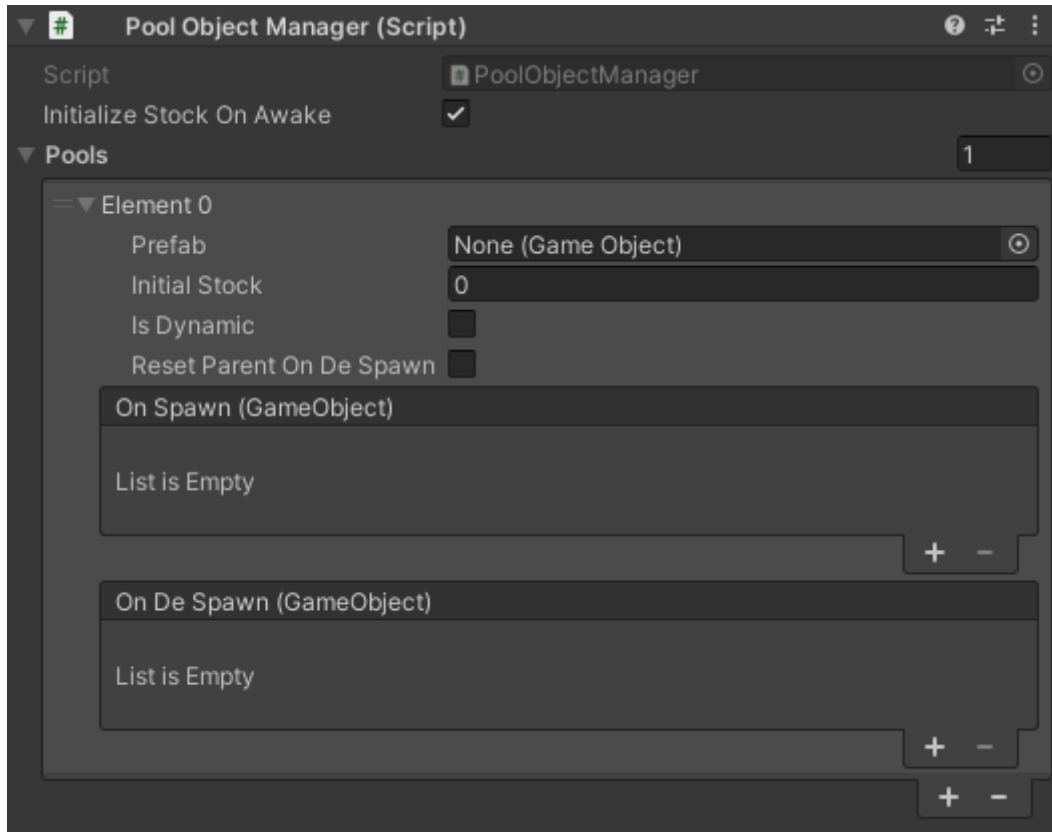


# POOL OBJECT MANAGER



## OVERVIEW

It uses a set of initialized objects ready for use. As long as you have objects, you don't create on demand. The idea is to reuse objects, instead of deleting and creating them again. Improves performance when creating bulk instances.

## FEATURES

- We create x quantity of necessary objects.
- We add them to the pool and deactivate them.
- We ask for them one by one every time we need them.
- If the pool is dynamic and there are no objects available in the pool, we create new ones and add them.
- If we are done using the object, we deactivate it and return it to the pool.
- We can add notify events when an object gets out and returns to the pool.

## CONFIGURATION

### Instance

Select an object and add the PoolObjectManager component. Only by doing this is it ready to be used.

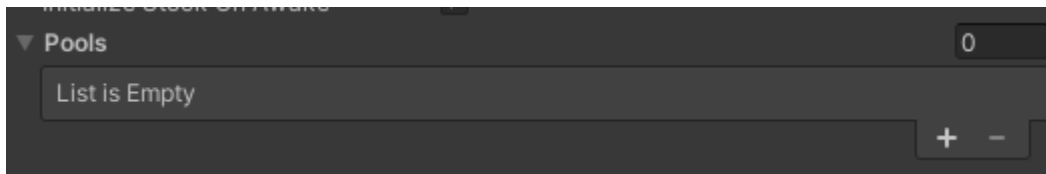
### Manager Settings

#### Initialize stock on awake



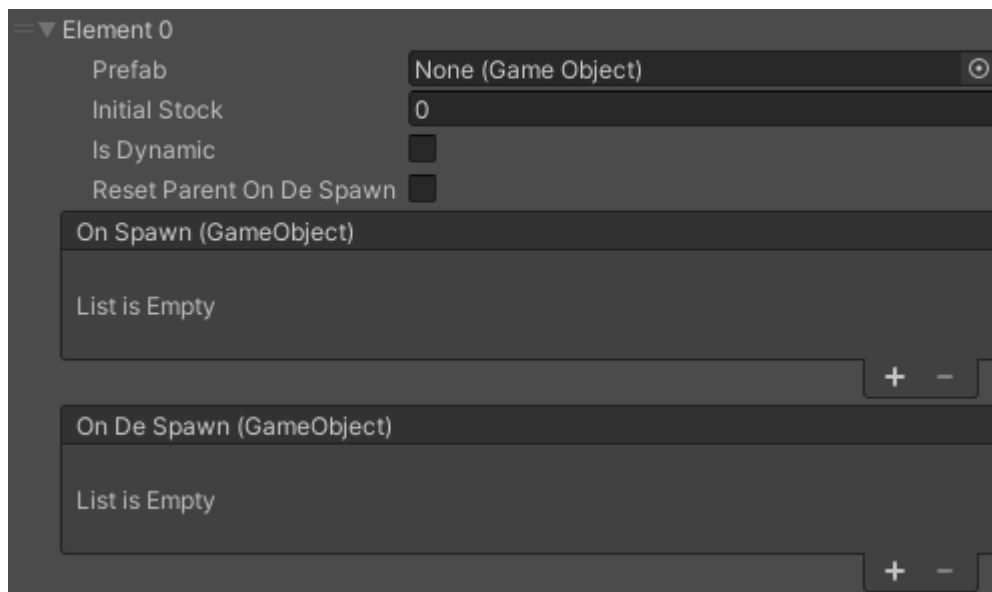
We can choose if the pools will be initialized on Awake or when we try to get an object.

#### Pool list



We can add multiple pools as you like. Each group has its own setting.

## Pool Setting



### Prefab



The object that will be used as a reference to generate the pool.

### Initial Stock



The initial stock indicates how many copies of the prefab it will create at its initialization.

### Is Dynamic



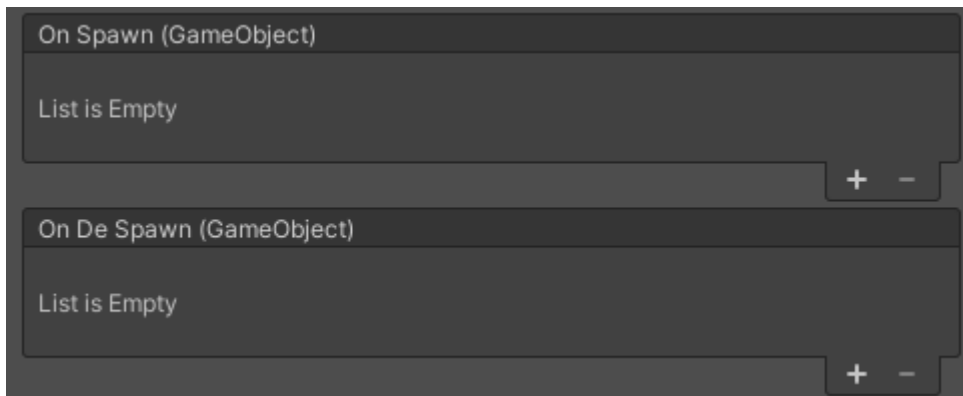
Indicate if the pool can generate a new instance of the prefab if there isn't an object available.

### Reset Parent On De Spawn



When objects are initialized, they are created in a new game object as parent. With this option, you can indicate whether the object, when it returns from the group, returns as a child of that object.

## Notify Events



You have 2 unity events:

-**OnSpawn(GameObject)**: This event is called when an object leaves the pool. The output object is passed by parameter.

-**OnDeSpawn(GameObject)**: This event is called when an object returns to the pool. The output object is passed by parameter.

## Additional API

Also, there is a set of public properties/methods available to clients of the PoolObjectManager class.

Method name	Description
<b>GameObject SpawObject(GameObject prefab, Vector3 position, Quaternion rotation)</b>	Get an object from the pool.
<b>void DeSpawnObject(GameObject obj)</b>	Return an object to the pool.
<b>void ReturnAllPool(GameObject prefab)</b>	Returns all the objects in a pool.

