Contents

Introduction

Pure Pool is a professional and extensive object pooling solution, designed to improve the performance of your game. Simply replace Instantiate and Destroy calls with Acquire and Release, and enjoy a smoother gameplay experience.

What Is Object Pooling?

Object pooling is a design pattern that keeps initialised objects in a "pool" until they are needed, rather than creating them on demand. When they are finished with, they are placed back in the pool for re-use, instead of destroying them.

Why Use Object Pooling?

Object pooling is a simple and effective optimisation, that is especially useful in game programming and other high performance situations.

The main purpose of object pooling is to remove the constant instantiation and destruction of objects during the lifetime of the game. Instantiation has an up-front cost of allocating memory and initialising scripts. Destruction has an indirect cost due to the Garbage Collector (GC).

The Garbage Collector is responsible for cleaning up managed memory in a .NET application. Periodically it will scan all objects that are allocated, and will check if they are still referenced from anywhere in your code. If the object is not referenced, it can be safely deleted by the GC.

When the GC runs, your game will pause for a short period of time, usually no more than a few milliseconds. However, the more you destroy objects in Unity, the more objects there will be for the GC to scan and delete, and the more times the GC will run.

Additionally, with many objects being deleted, fragmentation of the heap (managed memory allocated to your application) can occur, and the GC will have to move objects around to help free up larger chunks of memory. This is yet more time that your game will be paused while the GC performs its work.

Instead of instantiating and destroying objects in your game, you should recycle and re-use them through object pooling. This keeps the number and duration of garbage collections to a minimum, and **ensures a smooth and responsive game**.

Additional Reading

Object Pool - Game Programming Patterns / Optimization Patterns