Contents

[Physics Engine 3](#_Toc507493256)

[PhysicsScene 3](#_Toc507493257)

[IFixedUpdater 3](#_Toc507493258)

[PhysicsObject 3](#_Toc507493259)

# Introduction

\*\*\*\*\*\*TO DO\*\*\*\*\*\*

# Dependencies

\*\*\*TODO say how it used Bootstrap, glm\*\*\*

# Physics Engine

Classes in the PhysicsEngine library are in the physics namespace.

## PhysicsScene

A physics scene contains physics objects which interact with each other and manages the simulation.

During each fixed duration physics step the following actions are taken:

* FixedUpdater objects observing the scene have their fixedUpdate method called
* earlyUpdate is called on all physics objects. This is used to apply forces from springs.
* fixedUpdate is called on all physics objects. This is where rigidbodies move.
* Collisions are checked between physics objects. On a hit, the objects inform observers about it and resolve the collision

After all physics steps in an update, the physics objects are drawn. This interpolates their current and previous position by the amount of time left in the update. This means the object is consistently drawn one timestep before the current time, preventing temporal aliasing.

PhysicsObjects and FixedUpdaters are stored as shared pointers, since the scene will probably share responsibility for them with some other object. While FixedUpdaters can be removed at any time, PhysicsObjects should not be removed during a collision. Instead, the object’s kill() method should be called. This will flag it for removal at the end of a fixed update. Also, Clear should never be called within fixedUpdate or OnCollision functions.

## IFixedUpdater

This purely abstract class is an interface for objects to update during PhysicsScene fixed timestep updates. When added to a PhysicsScene, the fixedUpdate method will be called each timestep.

## PhysicsObject