



Physics

+ std::vector<Rigidbody*>: rigidbodies

+ OctreeNode: root

+ UpdatePhysics()

Quad

Plane

Sphere

+ vec3: center

+ float: radius

+ computeNextPosition(): vec3 + updatePosition(): void

OctreeNode

+ get_potential_colliders(const Collider& collider, std::vector<Collider*>& colliders): void

- speedFromNewton(vec3 curSpeed, vec3 acceleration): vec3 - positionFromNewton(vec3 curPos, vec3 curSpd): vec3

+ addForce(vec3): void

+ AABB: Box

+ std::vector<TreeNode> children + std::vector<Collider*> colliders

+ generate(float depth): void

+ draw_node(): void

+ scan colliders(std::vector<Collider*>& colliders): void

+ vec3: center

+ vec3: normal

+ float: distance

+ vec3: extensions



