ISpatialIndex< T >

+ virtual void insert(const T &object, float x, float y)=0
+ virtual std::vector< T > query(float x, float y, float range)=0
+ virtual void update(const T &object, float newX, float newY)=0
+ virtual void remove(const T &object)=0
+ virtual void clear()=0
+ virtual ~ISpatialIndex ()=default

OptimizedSpatialIndex< T >

+ std::vector< std::shared _ptr< SpatialObject< T > > > spatialObjects
- float size
- bool isSubdivided
- std::unique_ptr< OptimizedSpatial Index< T > > children
- std::pair< float, float > offset
- static const int MAX_OBJECTS
- static const int MIN_SIZE

+ OptimizedSpatialIndex (float size)
+ void insert(const T &object, float x, float y) override
+ std::vector< T > query (float x, float y, float range) override
+ void update(const T &object, float newX, float newY) override
+ void remove(const T &object) override
+ void clear() override
+ ~OptimizedSpatialIndex () override=default
- void _query(float x, float y, float range, std::vector < T > &result)
- bool inBounds(const std ::pair< float, float > &pos) const
- void setOffset(float offsetX, float offsetY)
- void subdivide()
- bool canMerge() const
- void merge()
- bool isEmpty() const
- bool intersectsRange(float min_x, float max_x, float min_y, float max_y) const
- int getChildIndex(float x, float y) const
- float getDistance(float x1, float y1, float x2, float y2) const

DefaultSpatialIndex< T >

- std::vector< SpatialObject < T > > spatialObjects

+ DefaultSpatialIndex()
+ void insert(const T &object, float x, float y) override
+ std::vector< T > query (float x, float y, float range) override
+ void update(const T &object, float newX, float newY) override
+ void remove(const T &object) override
+ void clear() override
- std::vector< SpatialObject < T > >::iterator findObject (const T &object)