



## C# API

version 0.1

### class Animation

float GetDuration()

### class Attachment

void SetIdentifier(System::String ^id)  
void SetParameter(System::String ^id, System::String ^value)  
bool Defined()

class DiagnosticsHandler  
void Log(System::String ^id)

## class Material\_Part

Mixer\_Input ^GetInput(uint32 index)

## class Material

void SetColour(Vector3 ^c)  
void GetColour(Vector3 ^c)  
Material\_Part ^GetPart(uint32 part\_index)

## class Mixer\_Input

void SetTextureCoordinateScale(Vector2 ^s)  
void SetTextureCoordinateOffset(Vector2 ^s)

## class Node

void Disconnect()  
void GetPosition(Vector3 ^\_v)  
void SetVisibility(bool visible)  
Attachment ^GetAttachment();  
void SetRight(float x, float y, float z);  
void SetUp(float x, float y, float z);  
void SetFront(float x, float y, float z);  
void SetPosition(float x, float y, float z);  
void SetPositionRelative(float x, float y, float z);  
void SetVelocity(float x, float y, float z);  
void AddChild(Node ^node)

## class Object

Node ^GetNode();  
SceneGraph ^GetOwner();  
SceneGraph ^GetGeometry();  
void SetGeometry(SceneGraph ^geom);  
void SetAnimationPlaybackOffset(float time);  
bool Intersection(Object ^other);

## class Program

```
void Terminate();  
bool Initialize(DiagnosticsHandler ^dh, Sound_Manager ^sm, Zone ^zone, SceneGraph ^scene,  
String ^path);  
bool OnMove(const Vector3 &delta);  
bool PerformProcessing(float current_time);  
bool OnButtonDown(uint32 button, float current_time);
```

## class SceneGraph

```
void Save(System::String ^local_filesystem_path)  
void SetIdentifier(System::String ^id);  
  
Material ^FindMaterial(System::String ^id);  
Animation ^GetAnimation();  
Node ^FindNode(System::String ^id);  
Node ^GetRoot();  
Node ^CreateNode(Node ^parent); // Must have a matching Dispose.  
Attachment ^CreateAttachment(Node ^dest, System::String ^class_id);  
void Disconnect(Attachment ^%att);  
Object ^CreateObject(Node ^dest);  
void Disconnect(Object ^%obj);
```

## class Sound\_Manager

```
void GetSamplerResolution(unsigned int ^dest);  
float Sample(uint32 channel, uint32 index);
```

## struct Vector2

```
Vector2() {}  
Vector2(float _x, float _y)  
static Vector2 ^ operator + (Vector2 ^c1, Vector2 ^c2)  
static Vector2 ^ operator - (Vector2 ^c1, Vector2 ^c2)  
static Vector2 ^ operator * (Vector2 ^c1, float s)  
static Vector2 ^ operator * (float s, Vector2 ^c1)  
static Vector2 ^ operator -(Vector2 ^c1)
```

## struct Vector3

```
Vector3() {}  
Vector3(float _x, float _y, float _z)  
static Vector3 ^ operator + (Vector3 ^c1, Vector3 ^c2)  
static Vector3 ^ operator - (Vector3 ^c1, Vector3 ^c2)  
static Vector3 ^ operator * (Vector3 ^c1, float s)  
static Vector3 ^ operator * (float s, Vector3 ^c1)  
static Vector3 ^ operator -(Vector3 ^c1)
```

## class Zone

```
SceneGraph ^ GetScene();  
SceneGraph ^ Load(System::String ^geometry_id, System::String ^search_path);  
Sound_Manager ^ GetSoundManager();  
bool Loading();
```