* 创建者：履霜坚冰
* 说明：简单地英译汉，更多细节还望大家补充，如果愿意，请留下你的论坛ID，方便咨询
* Irrlicht是一个用C++写成的高性能、多平台的实时3D图形引擎。它是一个强大的高级API用于建立一个完整的3D或2D程序，如：游戏，科学模拟！伴随它的还有一个优秀的文档和艺术级的表现特性。如：动态阴影，粒子系统，角色动画，室内室外技术，碰撞检测。所有这些都可以通过设计良好的C++接口获得，极其好用！

它主要的特性如下：

使用D3D和OPENGL进行高性能的实时3D渲染，Irrlicht支持6种渲染API

* + Direct3D 8.1
  + Direct3D 9.0
  + OpenGL 1.2-3.0
  + Irrlicht引擎的软件渲染系统
  + Burningsvideo 的软件渲染系统
  + 空设备，不使用任何渲染系统（程序依然正常跑，只是不显示）
* 当使用irrlicht的时候，程序员需要知道引擎当前使用的是哪种API，当然，这已经是一个抽象化的东西，你只需要告诉引擎你选择使用的API各类即可。
* 引擎支持多种图形API的原因：
  + 性能：有些图形显示卡对OPENGL进行了优化，而有一些对D3D进行了优化。
  + 平台：D3D不能在Mac和Linux下运行，但是OPENGL可以。当OPENGL也不合适的时候，引擎便会采用软件渲染方式。
  + 驱动问题：当用户在使用3D图形程序的时候，总是会遇到大大小小的问题导致应用程序崩溃，因为他们安装了一个旧的驱动。但也许切换一下图形驱动的类型，就能解决这个问题。

平台无关，能运行在 Windows95, 98, NT, 2000, XP, Linux, OSX, Solaris或其它一些系统上

* 下面是当前所支持的平台
  + Windows 98, ME, NT 4, 2000, XP, XP64, Vista, CE
  + Linux
  + OSX
  + Sun Solaris/SPARC
  + 所有使用SDL的平台

当然，计划远远比这些要多，但是并没有完全实现。引擎以同样的方式支持所有的平台。程序员在编程的时候只需要写一次代码便可以运行于所有的平台，而不需要更改一行代码。

强大的内建材质库，支持VS和PS

因为irrlicht是一个平台无关的引擎，为了快速地建立真实的场景效果，于是它提供了一些通用的材质。有一些材质是基于固定管线的（比如光照贴图）。而有一些是依赖3D硬件设备提供的可编程管线（比如法线贴图，视差贴图，逐像素光照等）。你可以在你的场景中混合使用这些材质而不会出任何问题。当你使用的材质硬件不支持时，引擎会选择一个相应的退化材质（fall back materials）。但是，如果这些材质满足不了你的要求，你就需要在运行时添加你的材质，但不需要你修改或重新编译引擎。当前支持的SHADER语言如下：

* + Pixel and Vertex Shaders 1.1 to 3.0
  + ARB Fragment and Vertex Programs
  + HLSL
  + GLSL

场景管理

非常灵活的场景管理实现准确的室内外混合

* irrlicht中的场景渲染采用了分层场景图（hierarchical scene graph）。场景结点可以相互挂接并跟随运动，视锥体裁剪掉它的子节点，做碰撞检测等。一个场景节点可以是一个摄相机，一个室内室外分界线（indoor or outdoor level），一个骨骼动画角色，一个运动的水，一个几何地形或是其它一些东西。。
* 这样，irrlicht引擎可以准确地将室内外场景混合，使程序员能够对场景进行完全的控制。它也很容易扩展，因为程序员可以增加自已的场景结点，模型，纹理加载器，GUI元素等等。
* 几何创建器（ geometry creator ）可以轻松地创建简单的几何体，如：圆柱体，立方体等。使用三角形，线，点，点精灵等基本图元可以使几何体有多边形，线框，点三种渲染方式。

角色动画系统，支持骨骼动画和Morph动画

* 目前实现了两种角色动画方式：
  + 渐变动画（Morph target animation）：模型关键帧之间进行线性插值。irrlicht中导入md2,md3文件的时候使用这种动画。其实现方式和Qake系列一样。
  + 骨骼动画：骨骼关节操纵皮肤，irrlicht中当导入.ms3d, .x, and .b3d 文件的时候，会使用骨骼动画。采用这种动画，可以很容易将一个物品挂接到运动模型的某个部位上。比如：只需要一行代码，就可以将一把武器挂到某个人物的手上，当手运动的时候，武器就会跟着动。
* 如果你不想了解，那你可以完全不用关心这个，因为所有的加载、控制和绘制工作都让irrlicht做完了。

粒子特效，公告板，光照贴图，环境贴图，模版阴影，等等一些高级特效

* 许多通用的特效都可以在irrlicht引擎中见到，并且它们不难运用，通常情况下程序员只需选择并启动即可。
* 引擎也经常扩展新的特效，下面是目前已经实现了的特效：
  + 真实的水面效果
  + 动态光照
  + 基于模版的动态阴影
  + 公告板
  + 凹凸贴图
  + 视差贴图
  + 透明物件
  + 光照贴图
  + 灵活的粒子系统，可用于火焰，雪，烟...
  + 球形贴图
  + 纹理动画
  + 天空盒和穹顶
  + 雾
  + 体积光

.Net语言绑定，使它可以被所有.Net平台的语言使用，如C#, VisualBasic,Delphi.NET.

包含了两种高效的平台和图形驱动无关的软渲染器。

* 它们包含了几乎所有需要的特性（加速VS质量）和属性：
  + 透视矫正纹理（ perspective correct texture mapping）
  + 双线性过滤（bilinear filtering）
  + 子像素正确性（sub pixel correctness）
  + 深度缓冲（z-buffer）
  + 平滑渐变着色（gouraud shading）
  + ALPHA混合和透明处理，快速的2D绘制等等。

强大，灵活，易用的UI系统

* 按钮
* 列表框
* 编辑框
* 标签
* 滚动条
* 等等

2D绘制功能，如ALPHA混合，关键色剔除，字体绘制，2D，3D混合图形等。

清析，易懂，很好的API文档，大量的例子和教程

使用纯C++编写，完全面向对象

可直接导入多种常见的模型文件和纹理格式

* 模型如：Maya (.obj), 3DStudio (.3ds), COLLADA (.dae), Blitz3D (.b3d), Milkshape (.ms3d), Quake 3 levels (.bsp), Quake2 models (.md2), Microsoft DirectX (.X)...
* 纹理如：Windows Bitmap (.bmp), Portable Network Graphics (.png), Adobe Photoshop (.psd), JPEG File Interchange Format (.jpg), Truevision Targa (.tga), ZSoft Painbrush (.pcx)...
* 引擎支持多种常见的文件格式，可以使用引擎直接导入使用。这样，当使用irrlicht引擎的时候，这些数据文件就不需要转换，节约开发时间。irrlicht内置的资源管理提供了对已经加载的模型文件或是数据文件进行简单的控制，和使用自身的缓存进行维护。
* 下面列表中所支持的文件格式经常在扩展，如果你想要的格式irrlicht不支持，你只需向我们提出要求即可。
* 当前支持的纹理文件格式：
  + JPEG File Interchange Format (.jpg, r/w)
  + Portable Network Graphics (.png, r/w)
  + Truevision Targa (.tga, r/w)
  + Windows Bitmap (.bmp, r/w)
  + Zsoft Paintbrush (.pcx, r/w)
  + Portable Pixmaps (.ppm, r/w)
  + Adobe Photoshop (.psd, r)
  + Quake 2 textures (.wal, r)
  + SGI truecolor textures (.rgb, r)
* 当前支持的模型文件格式
  + 动画模型
    - B3D files (.b3d, r, skeleton)
    - Microsoft DirectX (.x, r) (binary & text, skeleton)
    - Milkshape (.ms3d, r, skeleton)
    - Quake 3 models (.md3, r, morph)
    - Quake 2 models (.md2, r, morph)
  + 静态模型
    - Irrlicht scenes (.irr, r/w)
    - Irrlicht static meshes (.irrmesh, r/w)
    - 3D Studio meshes (.3ds, r)
    - Alias Wavefront Maya (.obj, r/w)
    - Lightwave Objects (.lwo, r)
    - COLLADA 1.4 (.xml, .dae, r/w)
    - OGRE meshes (.mesh, r)
    - My3DTools 3 (.my3D, r)
    - Pulsar LMTools (.lmts, r)
    - Quake 3 levels (.bsp, r)
    - DeleD (.dmf, r)
    - FSRad oct (.oct, r)
    - Cartography shop 4 (.csm, r)
    - STL 3D files (.stl, r/w)
    - PLY 3D files (.ply, r/w)

快速的，容易的碰撞检测和反应

使用优化的3D数学和模版容器库

直接读取压缩文件

(.zip, .pak, .pk3, .npk)

集成快速的XML解析器

Unicode支持,便于本地化

可以工作在Microsoft VisualStudio, Metrowerks Codewarrior, Bloodshed Dev-C++, Code

:Blocks, XCode, and gcc 3.x-4.x. 等环境下.

irrlicht引擎是开源并完全免费的.你可以调试,修正BUG,或者修改你不喜欢的部分.并且你不需要发布你所修改的内容.此引擎是基于[zlib licence](http://irrlicht3d.cn/wiki/index.php?title=Irrlicht%E6%8E%88%E6%9D%83&variant=zh-cn)的,并不是GPL/LGPL.

支持的渲染特性

* Irrlicht能够实现所有常见的高质量的材质渲染和效果,除开一些小小的例外,所有的特性支持硬件API加速,并且它的软渲染器也同样支持.
* 下面的列表列出了irrlicht引擎目前所支持的所有特性,对于某些特性需求,你可以提出扩展请求.
* 目前支持的渲染特性
  + 预定义材质
    - 实体(Solid)
    - 实体下的第二层纹理ALPHA混合 (Solid with alpha blending 2nd texture)
    - 可配置的光照贴图,并增加动态光支持 (Light maps with configurable pre-multiplication and additional dynamic light support)
    - 细节图(Detail map)
    - 球形贴图(Sphere map)
    - 环境映射(Environment reflection)
    - 使用纹理并用ADD运算的透明(Transparency by adding the texture)
    - 使用纹理ALPHA通道混合的透明(Transparency by using the texture alpha )
    - 使用纹理ALPHA但不混合的透明(Transparency by using the texture alpha without blending )
    - 使用顶点ALPHA的透明(Transparency by using the vertex alpha)
    - 法线图(Normal maps)
    - 视差图(Parallax maps)
    - 灵活的混合模式(Flexible blend mode rendering )
  + 材质颜色 Material Colors (ambient, diffuse, emissive, specular)
  + 发光 Shininess
  + 线宽(仅用于OPENGL) Line thickness (only OpenGL)
  + ZBuffer读/测试模式 ZBuffer write/test modes
  + 逐模型抗钜齿设置 Per mesh anti-aliasing settings
  + Alpha to Coverage
  + 颜色掩码 Color masking
  + 可配置的顶点色插值 Vertex colors with configurable interpretation
  + 线框/点云的绘制 Wireframe/Point cloud rendering
  + Gouraud/Flat填充 Gouraud/Flat shading
  + 可配置的光照模型 Lighting mode configurable
  + 前面,背面裁剪 Backface/Frontface culling
  + 逐模型雾 Fog enabling per mesh
  + 自动法线单位化 Automatic normals normalization
  + 纹理坐标重复/环绕寻址模式 Texture coordinates repeat/clamp modes
  + 逐纹理过滤(双线性过滤,三线性过滤,异向性过滤) Per texture filtering (bilinear, trilinear, anisotropic)
  + 纹理LOD矫正(Mipmap矫正) Texture LOD Bias
  + 纹理矩阵 Texture matrices
  + 随意的多层纹理数量 Arbitrary number of multi-textures