

MFC 与 OSG 结合说明文档

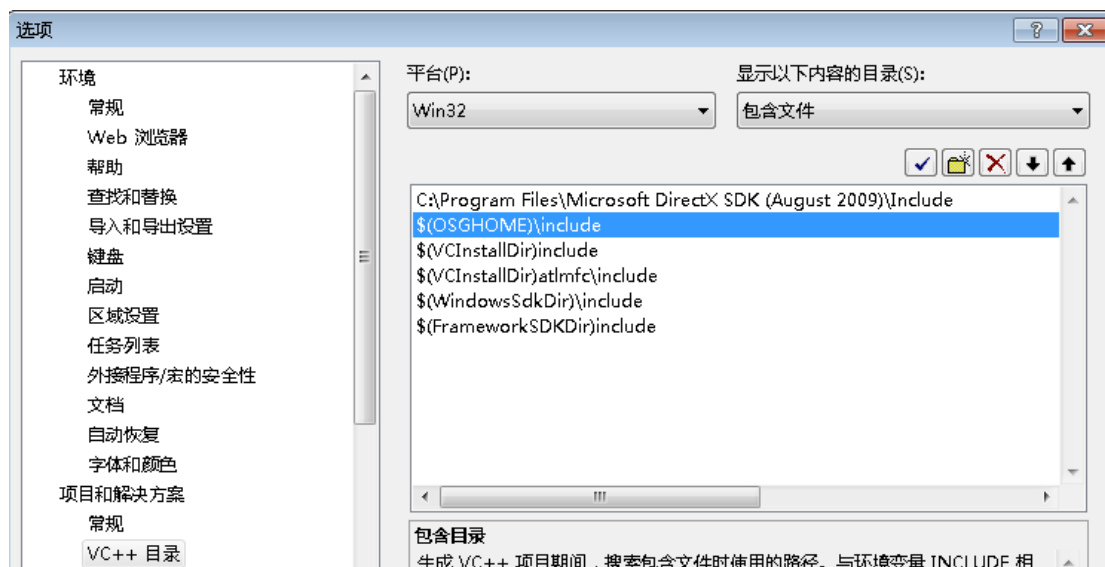
1. 环境配置

OSG 版本问题说明 现在我所提供的预编译好的 OSG 的版本是 2.8.2 和 2.9.9 , 其中 2.8.2 是基于 VS2005(SP1)编译的 ,2.9.9 版本是基于 VS2008(SP1) 编译的 , 所以 , 用哪个预编译的版本需要相应的 VS。如果是自己的 VS 不是以上版本 , 可以通过下载 OSG 的源代码自己编译(相关工具 , Cmake 在 svn 里面有)。

建议步骤如下 : 先安装 2.8.2 版本 , 那个会把环境变量设置好 , 然后会有 OSG 的源代码 , 还有示例程序。如果是 VS2005 版本 , 到此配置即可。如果是 VS2008 的 , 那么可以修改环境变量中的 OSG_HOME 变量 , 指定到 2.9.9 的位置。**【注意 OSG 中不能存在中文路径 , 稍后版本解决 , 所以 , osg 的位置 , 包括模型的位置 , 都不要存在中文的东西】**

2. 配置 VS

上图说明问题 : 首先在【工具】->【选项】->【项目和解决方案】->【VC++ 目录】设置一下包含文件和库文件 , 如下图设置。



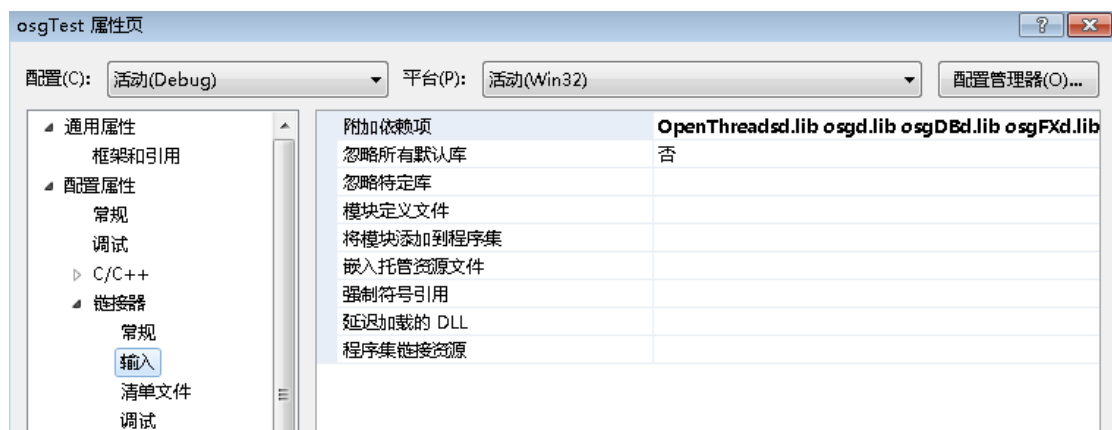
3. 开始新建 MFC 工程。一般是单文档。

然后我一般采用多字符集模式，要不然用 unicode 的话，会经常有一个_T 的问题。。之后就是在工程的依赖里面设置依赖库。【项目】->【属性】->【链接器】->【输入】->【附加依赖库】如下图：

一般是添加如下的依赖：

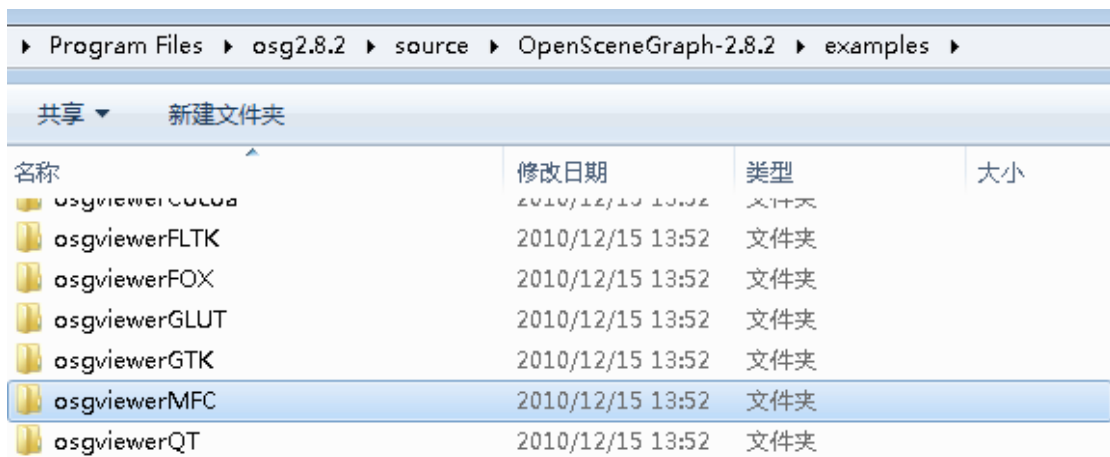
release: OpenThreads.lib osg.lib osgDB.lib osgFX.lib osgGA.lib
osgManipulator.lib osgParticle.lib osgShadow.lib osgSim.lib
osgTerrain.lib osgText.lib osgUtil.lib osgViewer.lib

Debug: OpenThreadsd.lib osgd.lib osgDBd.lib osgFXd.lib osgGAd.lib
osgManipulator.lib osgParticled.lib osgShadowd.lib osgSimd.lib
osgTerrain.lib osgTextd.lib osgUtild.lib osgViewerd.lib

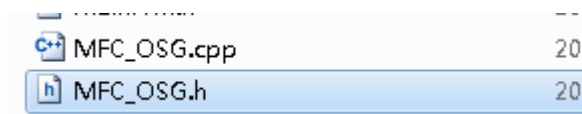


到此为止，配置的已经完成。需要开始 osg 的代码了。

到程序的源码里面找到如下的一个程序：



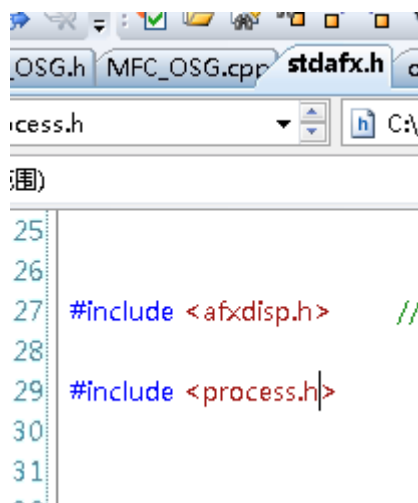
然后把里面的



这两个文件拷贝到你新建的工程下面，然后添加到工程中。

之后，需要开始修改一些东西。

首先在 stdafx.h 中添加#include<process.h>这一句代码



然后在工程的 view 类的头文件中添加如下代码：

【注意，此处的 OSG 的类名被我改了，这个看着更舒服】

//核心osg对象

CCoreOSG * mOSG ;

//线程句柄

HANDLE mThreadHandle ;

```

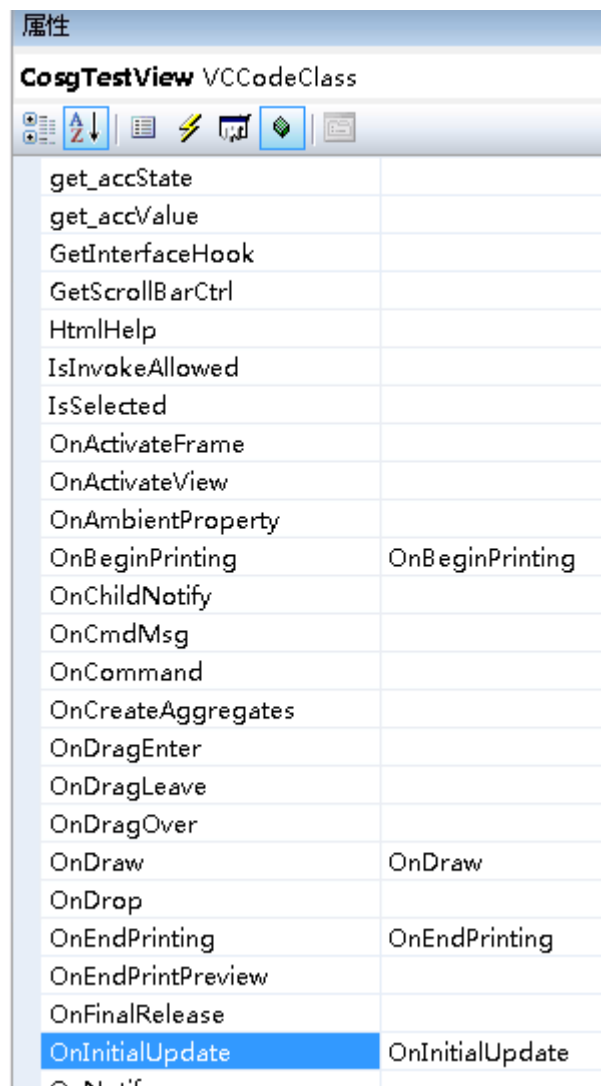
MFC_OSG.cpp  stdafx.h  osgTestView.h
view
ew

CosgTestDoc* GetDocument() const;

操作
public:
//核心osg对象
CCoreOSG * mOSG ;
//线程句柄
HANDLE mThreadHandle ;

```

然后，就是在 view 类中添加 WM_CREATE 的消息响应，以及重写 OnInitialUpdate 函数，全部可以在属性中添加



然后，在 oncreate 和 onInitialUpdate 中添加如下代码：

```

// CosgTestView 消息处理程序

void CosgTestView::OnInitialUpdate()
{
    CView::OnInitialUpdate();

    // TODO: 在此添加专用代码和/或调用基类
    mOSG->InitOSG("cow.osg");

    mThreadHandle = (HANDLE)_beginthread(&CCoreOSG::Render, 0, mOSG);
}

int CosgTestView::OnCreate(LPCREATESTRUCT lpCreateStruct)
{
    if (CView::OnCreate(lpCreateStruct) == -1)
        return -1;

    // TODO: 在此添加您专用的创建代码
    mOSG = new CCoreOSG(m_hWnd);

    return 0;
}

```

至此，一个 OSG 和 MFC 结合的 helloWorld 程序就完成了。

最后希望和预祝各位 OSG 路途快乐。。。。