

OpenSceneGraph2.8.2 的编译

本文主要介绍了 Windows 下如何从源代码编译 OSG 的过程,不涉及使用安装包.

要编译 OSG, 首先要有可以工作的代码, 这些代码可以从 OSG 的官网下载, 地址如下

1.普通下载:

<http://www.openscenegraph.org/projects/osg/wiki/Downloads>

Stable Release 2.8.2 (bug/build fix release), released 28th July 2009. [Further information...](#)

Source Code:

source package : [OpenSceneGraph-2.8.2.zip](#)

svn tag: svn co <http://www.openscenegraph.org/svn/osg/OpenSceneGraph/tags/OpenSceneGraph-2.8.2> OpenSceneGraph

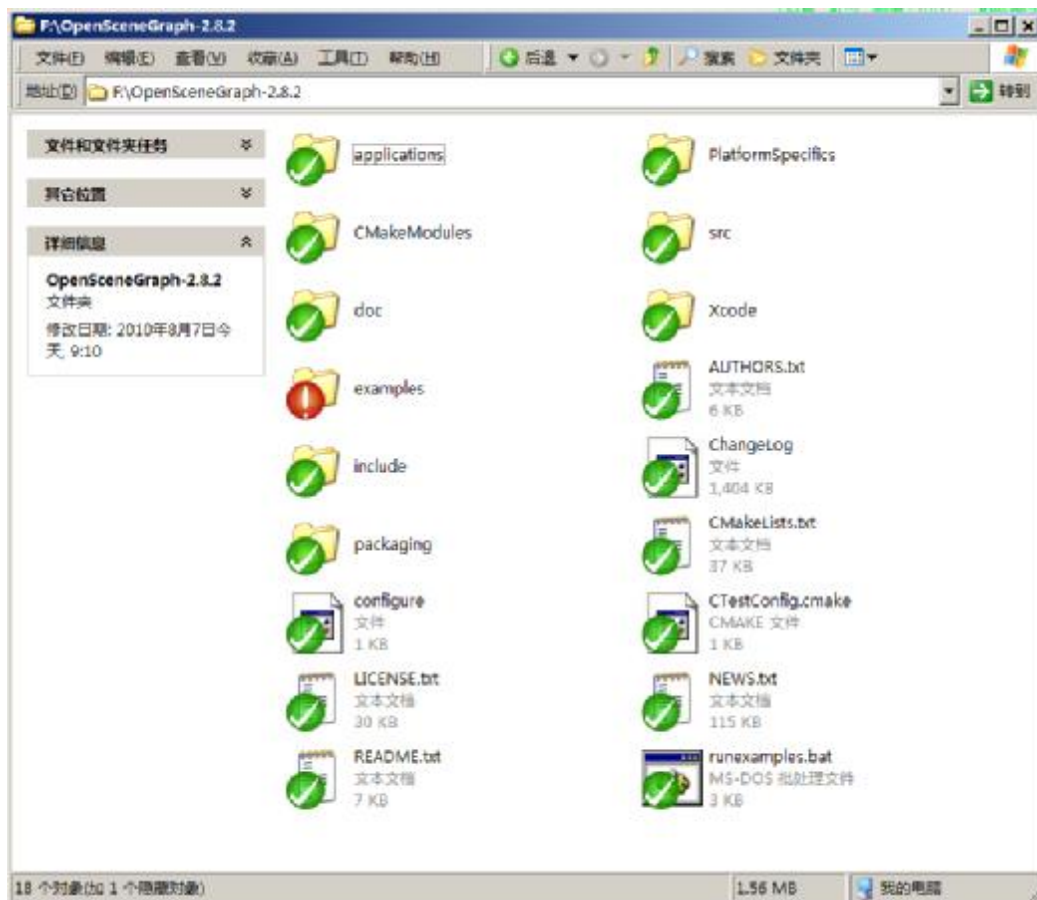
Binaries:

- [Windows Visual Studio 8 packages \(requirements\)](#)
- [Windows Visual Studio 9 packages \(requirements\)](#)
- [RedHat 5 / CentOS 5 64bit rpm packages](#)

2.svn 下载

地址: <http://www.openscenegraph.org/svn/osg/OpenSceneGraph/tags/OpenSceneGraph-2.8.2>

完成后,可以看到 OpenSceneGraph-2.8.2, 进入目录, 如下结构



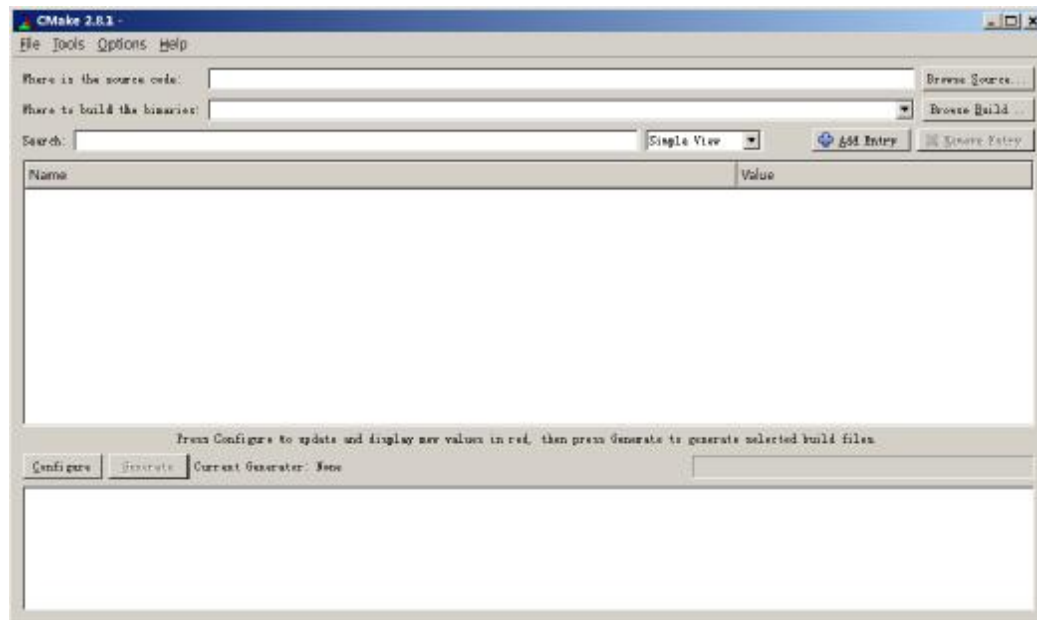
此时，说明已经得到了 OSG 的代码。

3.Cmake

下一步，就是准备编译工作 Cmake，以生成 OSG 工程，如果还没有安装这个工具，就需要先去如下地址下载一个

<http://www.cmake.org/cmake/resources/software.html>

安装完成后，打开，可以看到如下界面

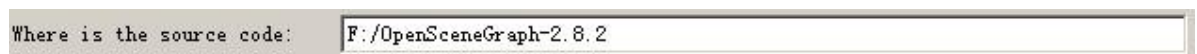


这时，是还没有打开任何 Cmake 文件时的样子

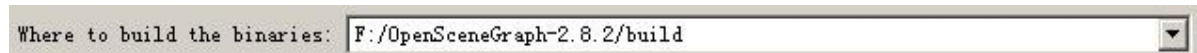
4.配置 OSG 的编译环境

进入刚才的 OSG 代码目录，里面应该有一个 CMakeLists.txt 文件，把此文件拖动进入 Cmake

的界面内，或是通过点击上面的 **Browse Source...** 按钮找到 OSG 的代码目录

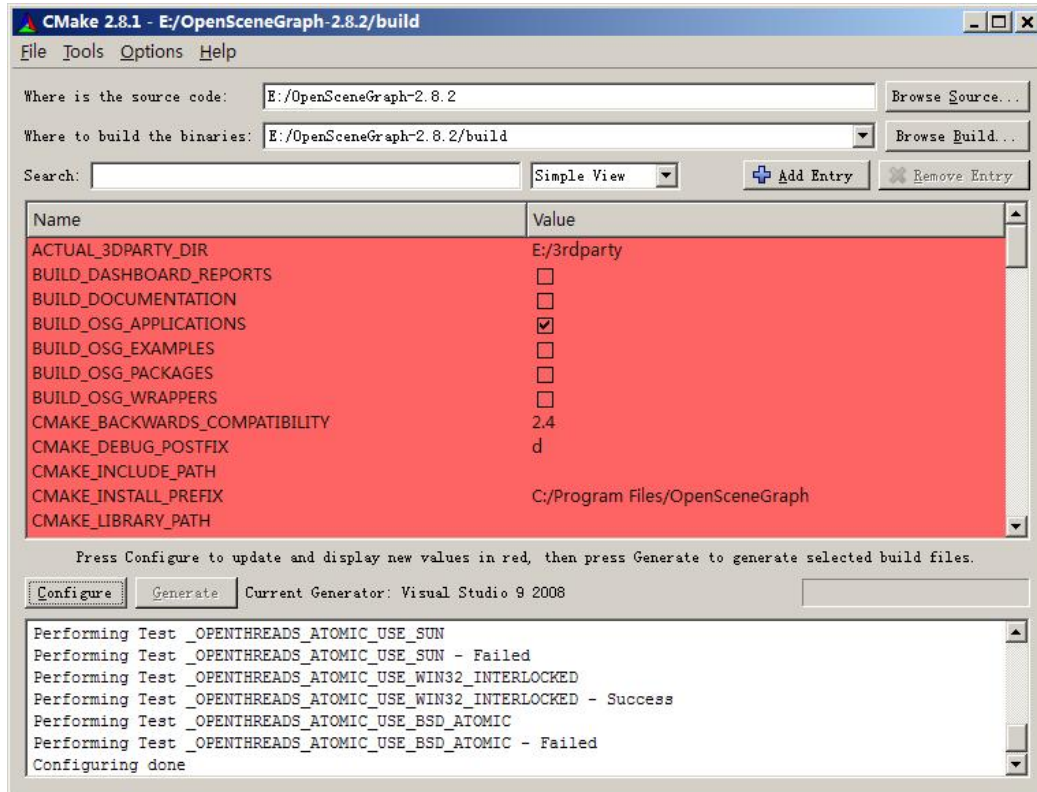


然后点击 **Browse Build...**，设定一个 OSG 的编译目录



一般会选择一个独立的目录，以使 OSG 的代码和编译中间文件分开，以后方便管理

此时，点击 **Configure** 中的 Configure，可以看到很多信息在 Cmake 的输出窗口出现，一定会有很多错误。



图中,红色部分就是没有正确配置的项目.先修改下面提到的设置,其它默认就可以(不是不解释,是本人也不知道这些设置的作用J).

有一个 3rdparty,这是 OSG 需要的依赖,没有正确设置,因为还没有下载这些库.

下载第三方库

第三方库也可以从源代码开始编译,不过,本文主要是介绍怎么编译 OSG,就直接从 OSG 上下载了.

SVN 地址:

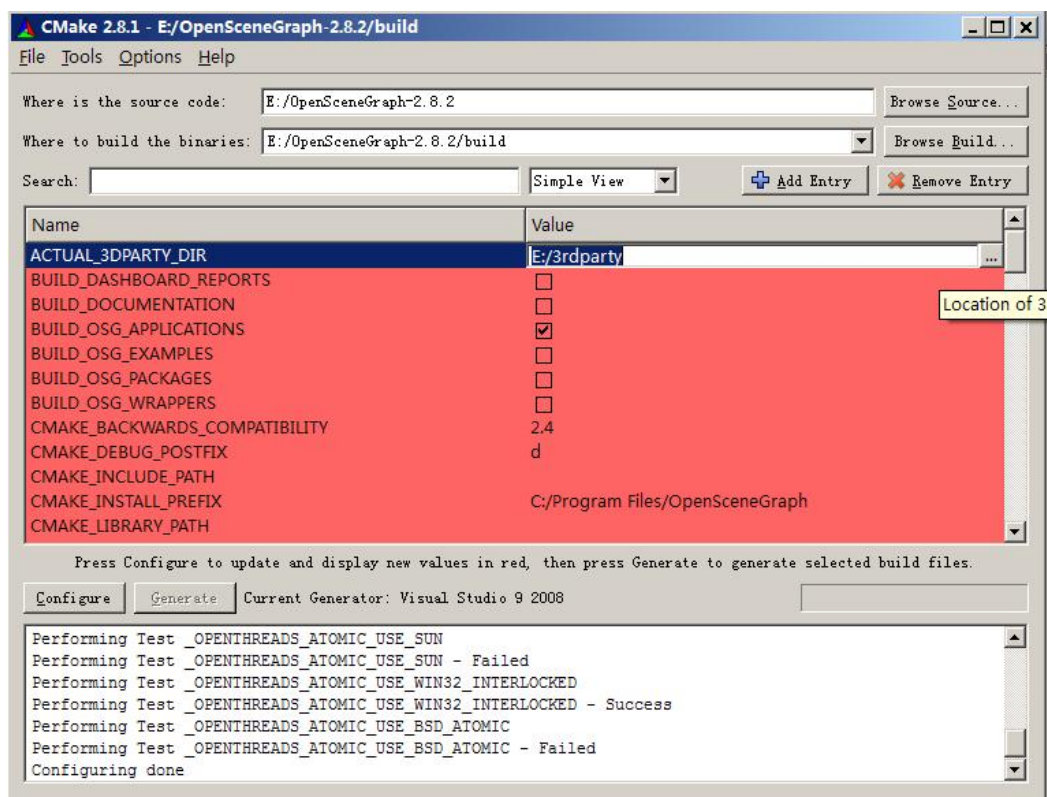
<https://osgtoy.svn.sourceforge.net/svnroot/osgtoy/3rdParty/branches/>

可以找一个和当前使用的 VS 相匹配的分支下载.

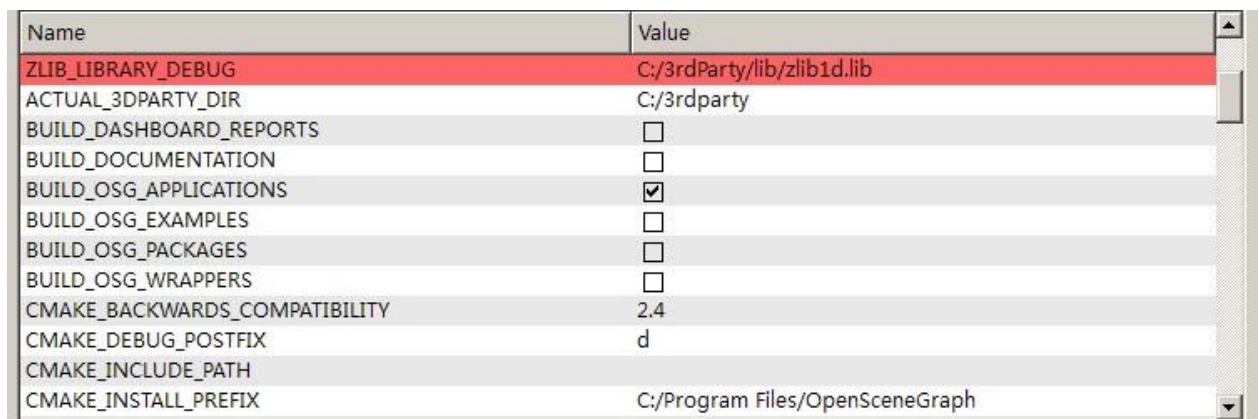
此时,可以继续 OSG 的配置了.第一个红色错误,在上面点一下,可以发现,后面出了一个可以选择路径的按钮,再点一下,找到刚才下载的依赖库的路径,重新点 **Configure** 按钮,可以发现多了下面这些配置变量

CURL_LIBRARY_DEBUG	C:/3rdParty/lib/curllibD.lib
FREETYPE_LIBRARY_DEBUG	C:/3rdParty/lib/freetype237_D.lib
GIFLIB_LIBRARY_DEBUG	C:/3rdParty/lib/libungifD.lib
GLUT_LIBRARY_DEBUG	C:/3rdParty/lib/glut32D.lib
JPEG_LIBRARY_DEBUG	C:/3rdParty/lib/libjpegD.lib
PNG_LIBRARY_DEBUG	C:/3rdParty/lib/libpng13d.lib
TIFF_LIBRARY_DEBUG	C:/3rdParty/lib/libtiffD.lib
ZLIB_LIBRARY_DEBUG	C:/3rdParty/lib/zlib1d.lib

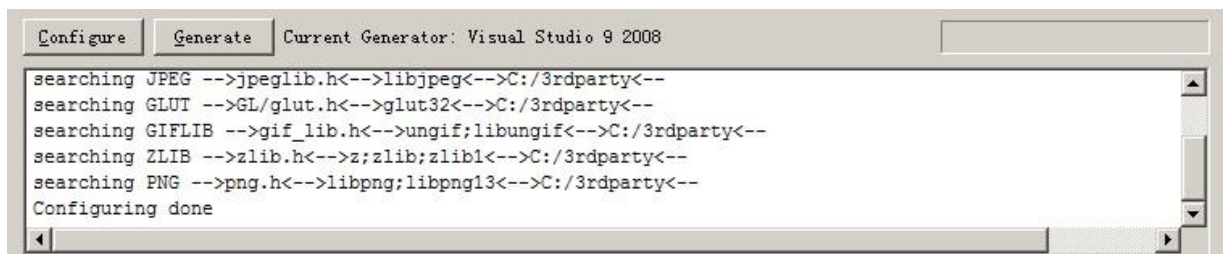
往下拉滚动条,可以发现还是有很多 COLLADA_DYNAMIC_LIBRARY-NOTFOUND 这样的条目,不过是白色,这些可以不加,对于现在编译 OSG 没有影响,不过,会使 OSG 不编译这些文件相关的插件,比如 QUICKTIME_INCLUDE_DIR-NOTFOUND,这个如果没有找到,OSG 则不会编译 QuickTime 相关的插件.



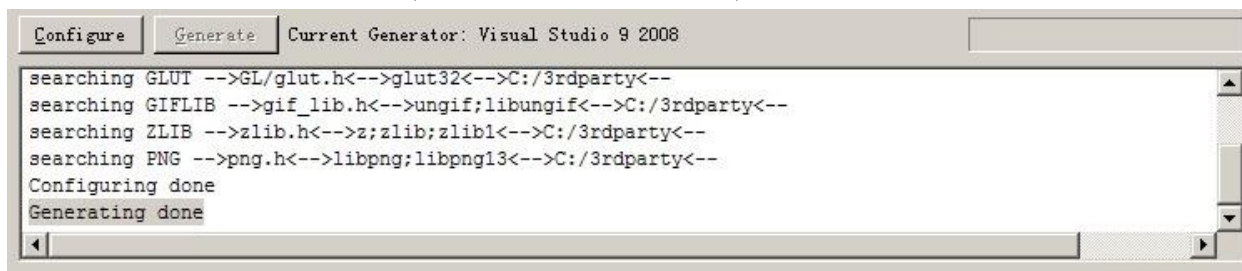
其它相关配置还有 OSG 的安装目录:CMAKE_INSTALL_PREFIX,可以调整一下,设定为自己相要的值.同时可以修改一下 BUILD_OSG_EXAMPLES,这样,生成的工程中,人包含 OSG 的例子,对于新手来说,这个是必要的,通过看 OSG 的例子,学习 OSG 的使用.



这时 Configure 旁边的 Generate 还是不能用, 再次点击 Configure 按钮,这一次的结果就是



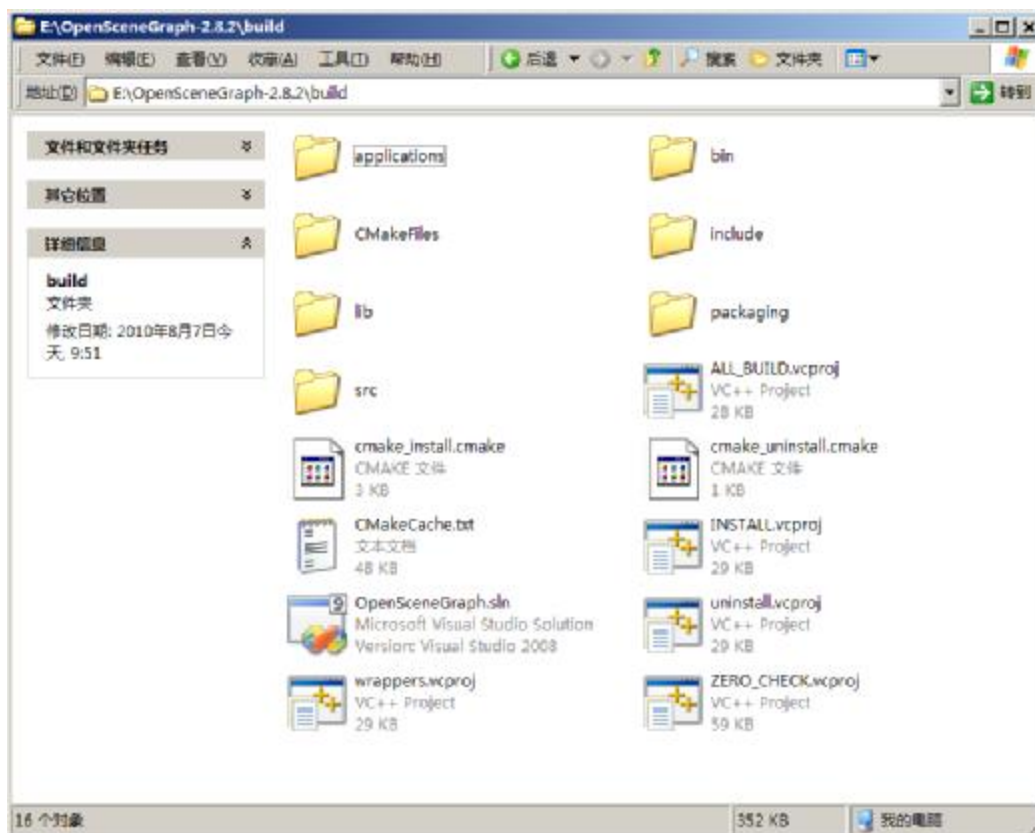
可以点击 **Generate** 了.等待一段时间,可以发现在这个输出窗口中,多了一行文字



表明现在正确的生成了 OSG 的工程.

5.编译

在 **Build** 目录下面,可以看到如下结构



如果用的是 VC9,直接双击 **OpenSceneGraph.sln**,就可以打开 OSG,进行编译.

可以看到有很多工程,需要注意的是,有一个 **ALL_BUILD** 和 **INSTALL** 工程,这两个都可以编译 OSG,不过,ALL_BUILD 只是进行了编译,INSTALL 在编译完成后,会把生成的 Bin,Lib,Include 等拷贝到刚才设置的 OSG 安装目录中.



选择 ALL_BUILD 编译,这个过程可能会需要很长时间,在等待过程中,可以考虑做点别的事情.

会在 build\bin 下面出现很多 EXE,大部分都是 OSg 的一些例子

osg2cpp.exe	18 KB	应用
osg2cppd.exe	54 KB	应用
osganimate.exe	47 KB	应用
osganimated.exe	142 KB	应用
osganimationmakepath.exe	52 KB	应用
osganimationmakepathd.exe	142 KB	应用
osganimationnode.exe	63 KB	应用
osganimationnoded.exe	200 KB	应用
osganimationskinning.exe	76 KB	应用
osganimationskinningd.exe	244 KB	应用
osganimationsolid.exe	52 KB	应用
osganimationsolidd.exe	168 KB	应用
osganimationtimeline.exe	45 KB	应用
osganimationtimelined.exe	167 KB	应用
osganimationviewer.exe	92 KB	应用
osganimationviewerd.exe	282 KB	应用
osgarchive.exe	24 KB	应用
osgarchived.exe	81 KB	应用
osgautocapture.exe	34 KB	应用
osgautocaptured.exe	81 KB	应用
osgautotransform.exe	38 KB	应用
osgautotransformd.exe	127 KB	应用
osgbillboard.exe	42 KB	应用
osgbillboardd.exe	142 KB	应用
osgblendequation.exe	22 KB	应用
osgblendequationd.exe	61 KB	应用
osgbrowser.exe	15 KB	应用
osgbrowserd.exe	49 KB	应用
osgcallback.exe	27 KB	应用
osgcallbackd.exe	73 KB	应用
osgcamera.exe	37 KB	应用
osgcamerad.exe	100 KB	应用

6.运行

这时,还需要去下载一个包,放到这里,才能看到例子的效果.OSG 运行时,需要的一些模型.在

刚才下载的文件中,除了源代码,就是链接库,还没提到模型呢.

<http://www.openscenegraph.org/projects/osg/wiki/Downloads/SampleDatasets>

在[这里](#)下载

同样,也可以下载压缩包,或是 SVN 方式下载.

解压后,放到 build\bin 下面,此时这个目录下面的结构为:

Catch		文件夹
Configuration		文件夹
Cubemap_axis		文件夹
Cubemap_snow		文件夹
ExampleThumbnails		文件夹
fonts		文件夹
Images		文件夹
osgPlugins-2.8.2		文件夹
osgWidget		文件夹
shaders		文件夹
SolarSystem		文件夹
osg.conf	1 KB	CONF 文件
osg2cppd.ilc	479 KB	Incremental
osg55-osgAnimationd.ilc	3,976 KB	Incremental
osg55-osgd.ilc	111,752 KB	Incremental
osg55-osgDBd.ilc	9,496 KB	Incremental
osg55-osgFXd.ilc	3,491 KB	Incremental
osg55-osgGAd.ilc	2,738 KB	Incremental
osg55-osgIntrospectiond.ilc	7,249 KB	Incremental
osg55-osgManipulatord.ilc	6.158 KB	Incremental

现在可以找一个例子运行了

