OSG编译

**编译版本：osg3. 4.0+VS2013+win10**

### 一．需要的源码及一些库

1．OSG源码

此次使用OSG版本为：OpenSceneGraph-3.4.0.ZIP

下载链接：

<http://www.osgchina.org/index.php?option=com_content&view=article&id=222:openscenegraph-3-4-0&catid=106&Itemid=478>

2．第三方库

此次使用版本：3rdParty\_VS2013\_v120\_x86\_x64\_V9\_full.ZIP

（本次下载的第三库相对完整，对于gdal来说不需要编译，但是类似于geos等没有进行编译，所以要自己再配一下，见osgearth章节的配置第三方库）

下载链接：

<http://www.openscenegraph.org/index.php/download-section/dependencies>

3．数据包

此次使用数据版本为：[OpenSceneGraph-Data-3.0.0.zip](http://www.openscenegraph.org/downloads/stable_releases/OpenSceneGraph-3.0/data/OpenSceneGraph-Data-3.0.0.zip)

下载链接：

<http://www.openscenegraph.org/index.php/download-section/data>

4.CMake编译器：

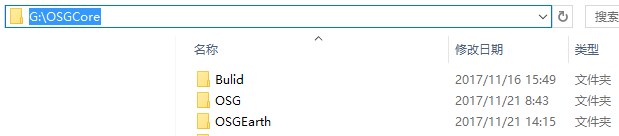
cmake可以使用安装版也可以使用绿色版的，这里我使用的是3.8.2绿色版本（直接运行bin里面的.exe文件即可）

下载链接：

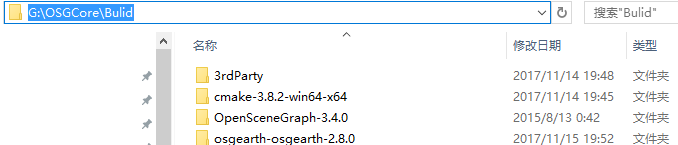
<https://cmake.org/download/>

### 二、编译准备

1．建立一个OSGCore文件夹，便于管理，文件结构如下：



G:\OSGCore\Bulid:OSG以及OSGEarth源码放在这个位置，cmake-3.8.2-win64-x64绿色版本文件也放与此，以及3rdParty第三方库也放在这里；结构如图所示



说明：

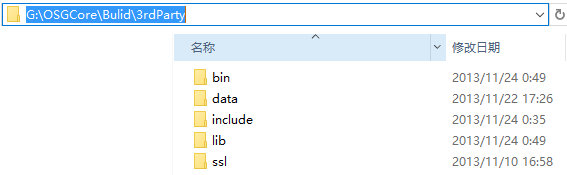
G:\OSGCore\Bulid\3rdParty：将第三方依赖库解压后放于此

G:\OSGCore\Bulid\OpenScenGraph-Data-3.0.0：数据包解压后放于此。

G:\OSGCore\OSG:将OSG编译后的工程文件放置在此。

G:\OSGCore\OSGEarth:将OSGEarth编译后的工程文件放置在此。

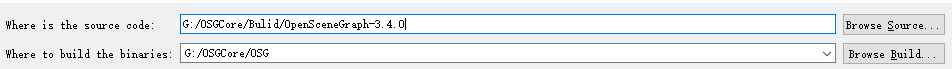
2. 重新组织3rdParty，把x64文件夹中的内容拷贝到3rdParty中，拷贝完成后把x86,x64删除。拷贝完的结构如下图所示：



### 三、cmake配置，并编译源码

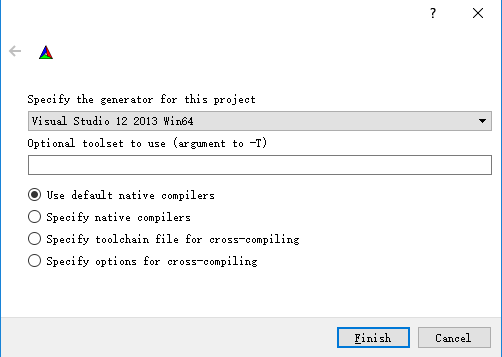
1.打开源码OpenSceneGraph-3.4.0.zip解压后的文件,将里面的CmakeLists.txt,拖到Cmake界面内或者在打开的cmake 3.8.2中点击，通过定位文件位置来打开OpenSceneGraph-3.4.0源码文件夹，点击选择生成工程的存放位置。

2. 配置cmake:



3. 配置2013 WIN64编译模式：

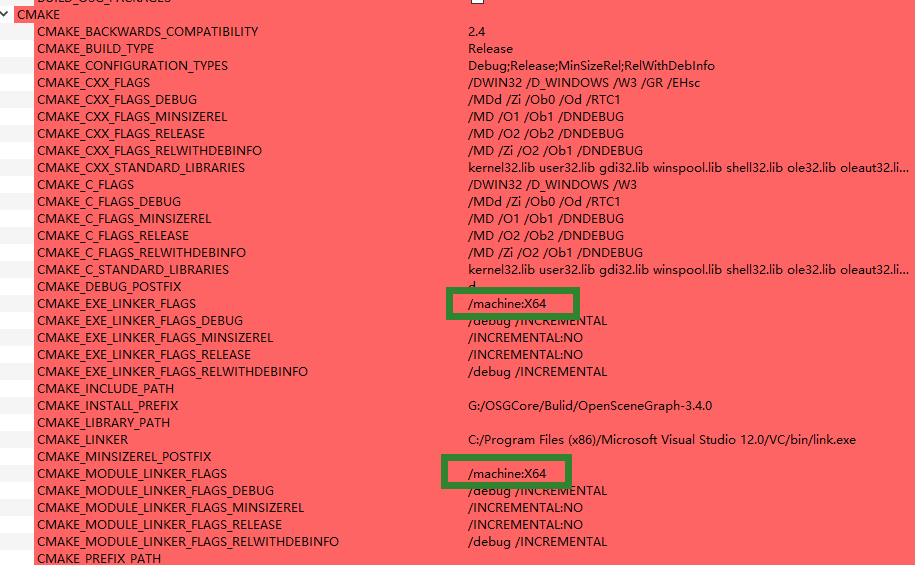
在完成上面的步骤2后，点击，打开vs选择工具，选择2013 WIN64，

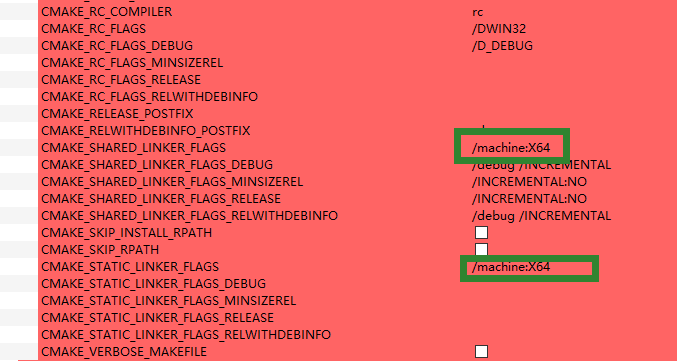


4. 点击finish结束，等待生成，勾上Advanced 和group,结果如下图：



注意：CMAKE中的解决方案平台需要制定为64位，要不然在编译的时候可以出现问题。在完成上述的步骤后，绿色的框标识的位置表示解决方案平台使用的是64位。如果结果如图则不需要进行改变，接着进行下一步即可，如果不是请手动更改。





5. 对cmake中的一些错误进行修改，首先修改ACTUAL\_3RDPARTY\_DIR值为D:/OSG/3rdParty(一直定位到bin、lib、include所在目录);



6．BUILD\_OSG\_EXAMPLES: on



7．CMAKE\_INSTALL\_PREFIX：G:/OSGCore/Bulid/OpenSceneGraph-3.4.0



点击Configure



8. BUILD\_MFC\_ EXAMPLE：on

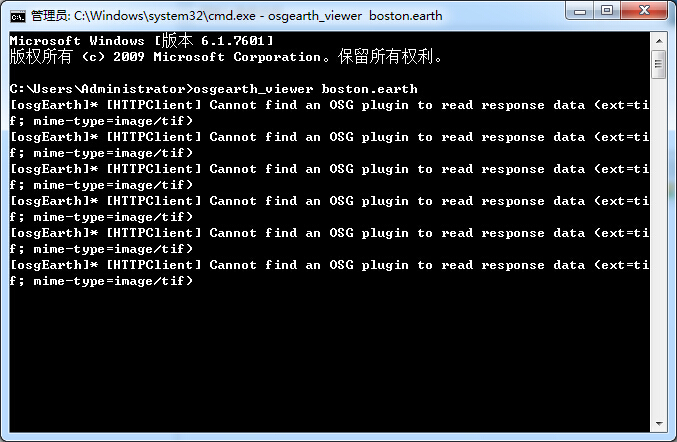


9. 检查TIFF、JPEG等一些插件的库有没有配好。配置结果如下





如果配置不正确，后面在跑osgearth的时候会出现以下问题：



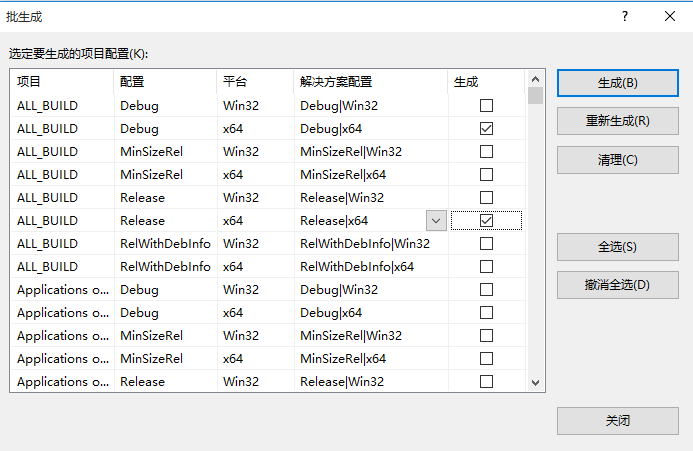
10. 编译源码

用VS2013打开OpenSceneGraph.sln（此文件在G:/OSGCore/OSG下），

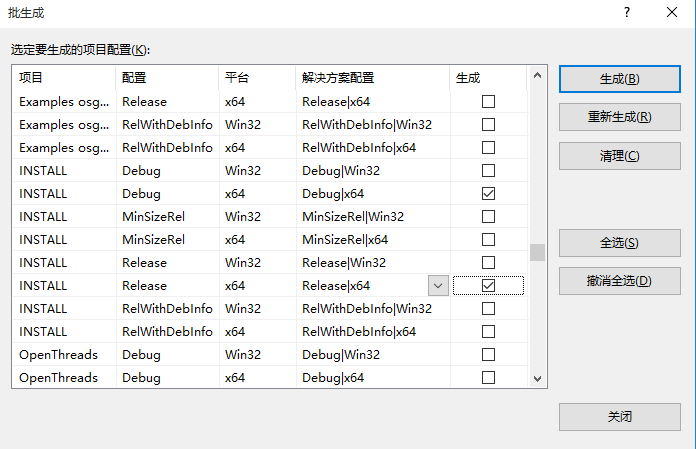
注意这里我们编译的是64位文件



然后点击生成-批生成，对All\_BUILD进行生成，选择（Debug和Release两个版本），这段编译时间是很长的，估计2个小时。耐心等待吧



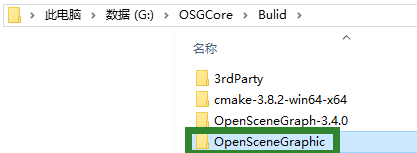
11. 完成后，再对INSTALL也进行相应生成编译。这段时间相应会短一点。



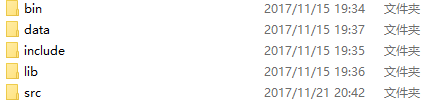
### 四、环境变量的设置及测试

#### 软件部署（只是列出了debug，release于debug相同）

1. 在G:\OSGCore\Bulid新建一个名为OpenSceneGraphic的文件夹，用来存放部署文件。

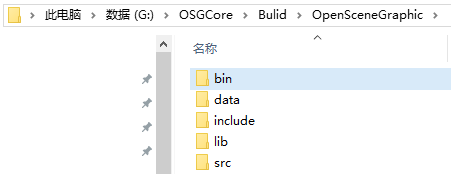


2. 在OpenSceneGraphic下面建立5个文件夹



3. 下面是拷贝工作，首先把debug版本拷贝过去：

|  |  |  |
| --- | --- | --- |
| G:\OSGCore\Bulid\OpenScenGraph-Data-3.0.0下所有 | 拷贝到 | G:\OSGCore\Bulid\OpenSceneGraphic\data下 |
| 1.搜索G:\OSGCore\OSG\下的.exe文件  2.搜索G:\OSGCore\OSG\下的.dll文件  3.G:\OSGCore\OSG\bin文件夹下如果有dll或者exe一起考到1和2 所说的位置下 | 1.G:\OSGCore\Bulid\OpenSceneGraphic \bin  2.G:\OSGCore\Bulid\OpenSceneGraphic h\bin  3.G:\OSGCore\OSG\bin下有osgPlugins-3.0.1文件，在G:\OSGCore\Bulid\OpenSceneGraphic\bin下新建文件夹：文件命名osgPlugins-3.0.1，把.dll文件拷贝到此位置。 |
| 1.把G:\OSGCore\OSG\include  2.G:\OSGCore\Bulid\OpenSceneGraph-3.4.0\include | 1.G:\OSGCore\Bulid\OpenSceneGraphic \include  2.G:\OSGCore\Bulid\OpenSceneGraphic include选择合并 |
| 1.G:\OSGCore\OSG\lib下文件夹osgPlugins-3.0.1  2.G:\OSGCore\OSG\lib下的lib文件 | 1.G:\OSGCore\Bulid\OpenSceneGraphic \lib（直接拷贝文件于此就可以了）  2.G:\OSGCore\Bulid\OpenSceneGraphic \lib |
| G:\OSGCore\Bulid\OpenSceneGraph-3.4.0\src下所有 | G:\OSGCore\Bulid\OpenSceneGraphi\src下 |



4.然后设置环境变量

**OSG\_FILE\_PATH: G:\OSGCore\Bulid\OpenSceneGraphic\data**

**PATH：G:\OSGCore\Bulid\OpenSceneGraphic\bin**

5．测试处理：打开cmd命令，输入osgviewer viewer cow.osg看看是否有头牛,如图



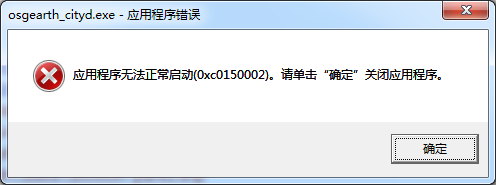
OSGEarth编译

**编译版本：**osg3.4.0+VS2013+win10+osgearth2.8

### 一、编译OSGEarth的注意事项：

1. 第三方库里编译的一些库，要和你编OSGEarth时用的是同一个编译器；

如第三方库里的gdal是用vs2010，你编译OSGEarth时用vs2013，并使用那个第三方库，在后边跑的时候就会出现像下边这种问题。

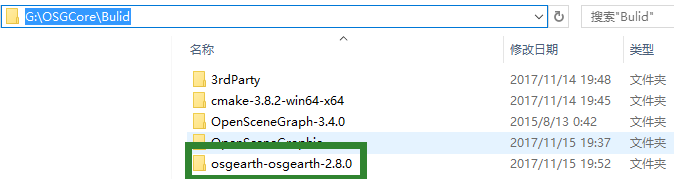


2. 编译OSG时用的第三方库(3rdParty)和编译OSGEarth时用的第三方库尽量保持一致，论坛里有人说有关系，以免出现意外，又要重编，所以建议第三方库使用为同一个。

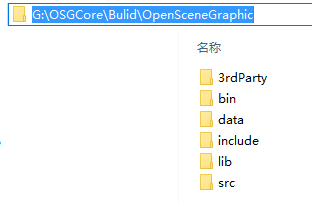
3. 后面我们使用到的cmake都是使用同一个文件下的exe文件，后面我们会对cmake的文件夹进行一定的修改，所以需要保证使用的cmake文件不变，否则会出问题。

#### 文件目录构建

1. 我们在G:\OSGCore文件夹下编译osgEarth，首先建立一个文件夹：G:\OSGCore\osgEarth（在本文osg编译的第二部分编译准备中已经说明了文件的结构构建）,用来存放编译后的工程，把源代码放到G:\OSGCore\Bulid文件夹中,重命名为osgearth-osgearth-2.8.0：



2. 将文件结构构建好之后我们开始进行数据拷贝，首先把G:\OSGCore\Bulid\文件夹下的3rdParty文件夹拷贝到G:\OSGCore\Bulid\OpenSceneGraphic\文件夹下。



### 二、配置第三方库：

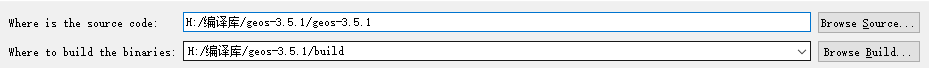
**第一编译GEOS：**（几何引擎——开源）是一个Java拓扑套件（JTS）的C++移植。因此，它的目的是在C++中包含完整的JTS移植。这包括了所有的OpenGIS“简单SQL特征”：空间预测函数和空间操作员，以及具体的JTS拓扑功能

下载链接：<http://trac.osgeo.org/geos/>

使用cmake编译geos ：

编译步骤为：

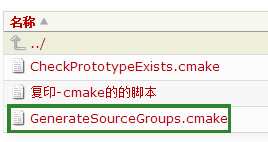
1. 打开cmake，加载geos源码和定位geos的工程存放位置；



2. 点击configure，会报错，首先设置CMAKE\_INSTALL-PREFIX定位到geos源码文件目录；

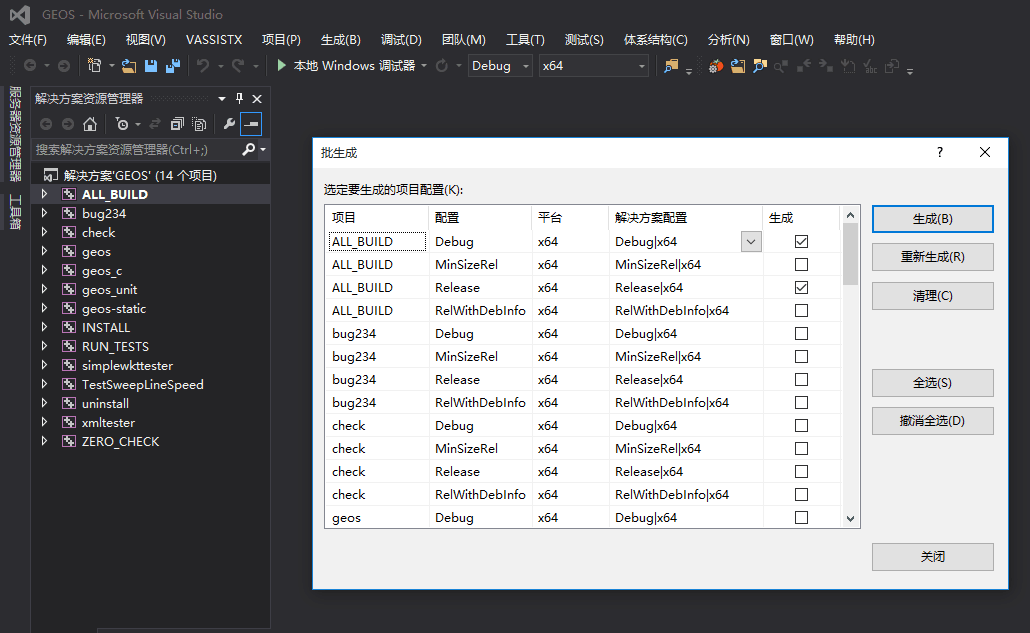


3. 打开链接<https://trac.osgeo.org/geos/browser/trunk/cmake/modules> 下载GenerateSourceGroups.cmake文件



4. 将下载后的文件放到使用的cmake3.8.2中，我使用cmake的文件路径为G:\OSGCore\Bulid\cmake-3.8.2-win64-x64，则将刚才下载的文件拷贝到G:\OSGCore\Bulid\cmake-3.8.2-win64-x64\share\cmake-3.8\Modules文件下。

然后点击configure，不报错了，点击generate，生成编译工程文件，然后打开vs2013工程文件，点击批生成，如图

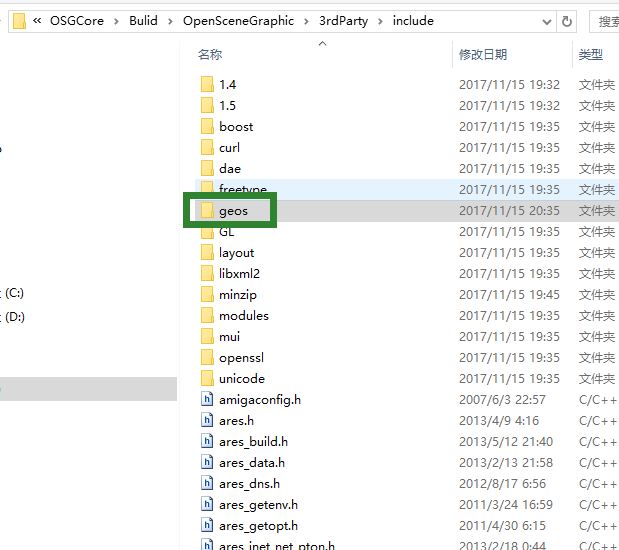


然后点击生成

5编译后需要转移以下文件到G:\OSGCore\Bulid\OpenSceneGraphic\3rdParty\下的对应位置：（第一个是将geos源码目录下的include文件夹下的geos文件夹拷贝到G:\OSGCore\Bulid\OpenSceneGraphic\3rdParty\include文件夹中，第二个是将编译工程文件夹下的include文件下的geos里面的两个.h文件拷贝到前面的:G\OSGCore\Bulid\OpenSceneGraphic\3rdParty\include\geos中，将编译后生成的lib文件拷贝到G:\OSGCore\Bulid\OpenSceneGraphic\3rdParty\lib下，将编译后生成的dll拷贝到G:\OSGCore\Bulid\OpenSceneGraphic\3rdParty\bin下）

如果找到的两个dll文件名称相同，则将debug下的dll的文件的名称后加上d（文件可以在geos编译的工程文件里面找）

拷贝结束后的文件夹目录结构：



**第二下载拷贝MiNizip:**就用3rdParty/lib下面的minizip.lib和minizipD.lib，但头文件需要下载，把解压后的头文件放到G\OSGCore\Bulid\OpenSceneGraphic\3rdParty\include\minizip下

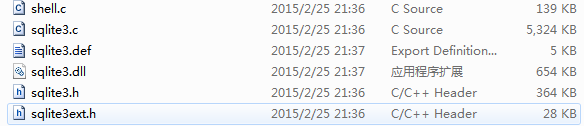
下载链接：<http://www.winimage.com/zLibDll/minizip.html>

**第三编译SQLite:** 是一款轻型的数据库，是遵守ACID的关联式数据库管理系统，它的设计目标是嵌入式的，而且目前已经在很多嵌入式产品中使用了它，它占用资源非常的低，在嵌入式设备中，可能只需要几百K的内存就够了。它能够支持Windows/Linux/Unix等等主流的操作系统，同时能够跟很多程序语言相结合，比如 Tcl、C#、PHP、Java等，还有ODBC接口，同样比起Mysql、PostgreSQL这两款开源世界著名的数据库管理系统来讲，它的处理速度比他们都快。

下载链接：<http://www.sqlite.org/download.html>

编译步骤：

1. 首先下载两个压缩包：解压后，把sqlite-dll-win64-x64-3210000下的两个文件放到sqlite-amalgamation-3210000解压后的目录下，解压后文件夹重命名为sqlite3。组合成下面所示：



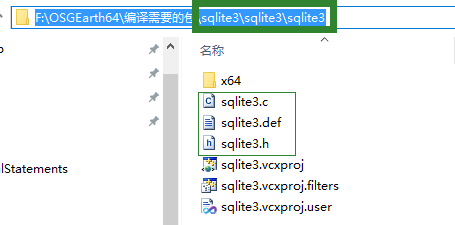
2．打开vs2013后新建动态连接库项目，将工程放置到前面重新命名的sqlite3文件夹中，工程名称命名为sqlite3；win32控制台dll，空项目。



点击确定后将解决方案平台设置为64位



3．完成上述步骤后在重命名的sqlite3文件下会出现一个sqlite3文件夹，里面存放的刚才创建的工程文件，拷贝sqlite3文件夹下的sqlite3.h, sqlite3.c, sqlite3.def 3个文件到之前新建的sqlite3工程下的对应位置中，对应的位置如图所示：

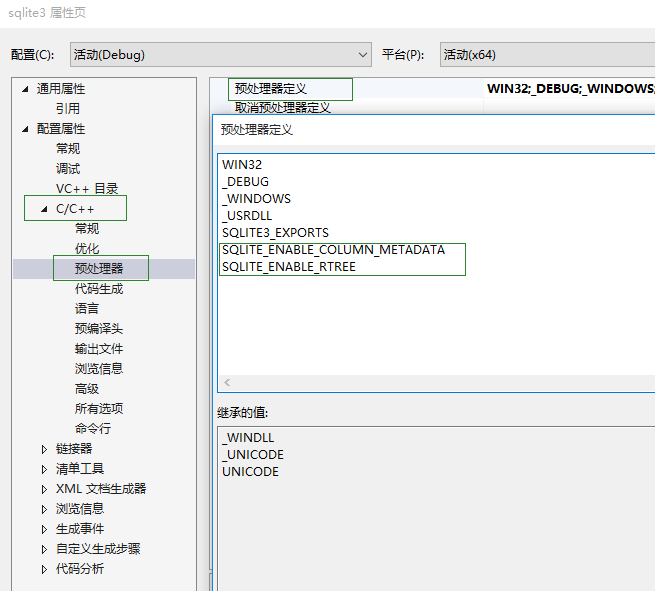


4．打开vs工程后，添加头文件sqlite3.h到头文件； 添加源文件sqlite3.c，sqlite3.def到源文件。

5．设置项目属性

SQLITE\_ENABLE\_COLUMN\_METADATA

SQLITE\_ENABLE\_RTREE



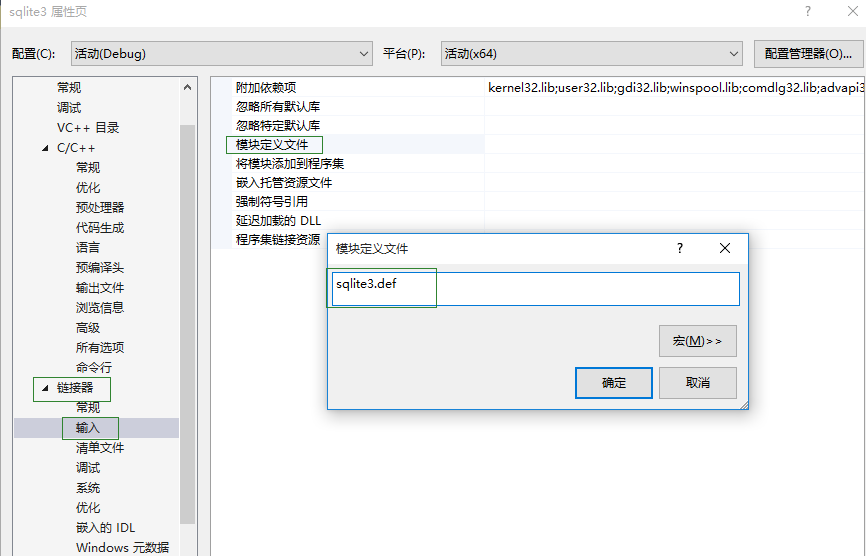
需要注意的是：在完成上述的步骤后，点击批生成后，发现此时只生成了sqlite\_win32\_dll.dll文件，没有生成对应的引入库，即.lib文件，生成lib文件的具体方法如下：

**第一种方法**：修改sqlite3.c文件中的SQLITE\_API宏定义（当前版本在sqlite3.c的第26行），将

把 # define SQLITE\_API 修改成 # define SQLITE\_API \_\_declspec(dllexport)

重新编译即可生成sqlite\_win32\_dll.dll和sqlite\_win32\_dll.lib。

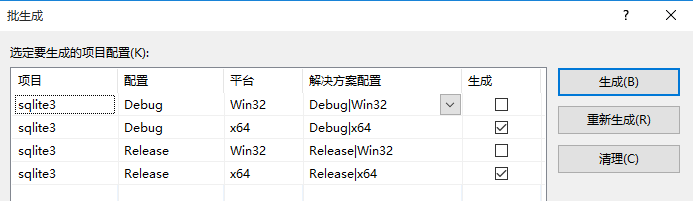
**第二种方法：**直接把sqlite3.def添加在如图所示位置：



问题解决方案链接：

<http://www.cnblogs.com/xFreedom/archive/2012/04/04/2432386.html>

点击批生成：



6．生成debug 和release版本，分别命名为sqlite3d.lib|sqlite3d.dll, sqlite3.lib|sqlite3.dll

7．把sqlite3.lib 和sqlite3.dll拷贝到

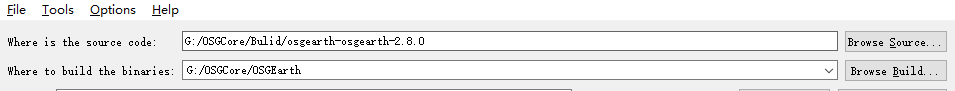
G\OSGCore\Bulid\OpenSceneGraphic\3rdParty\bin、 G\OSGCore\Bulid\OpenSceneGraphic\3rdParty \lib

### 三、编译准备

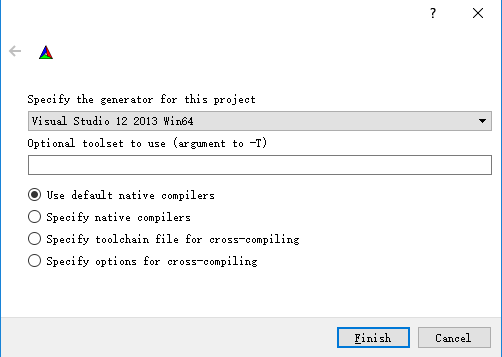
G:\OSGCore\Bulid\osgearth-osgearth-2.8.0：osgearth源码存放位置

### 四、cmake配置，并编译源码

1. 配cmake



2. 同时依然需要注意的是64位的问题。具体和OSG编译文档的设置方法相似，在设置cmake的时候点击2013 WIN64即可

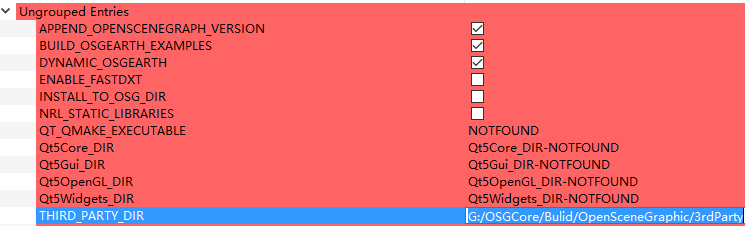




3. 指定OSG路径：

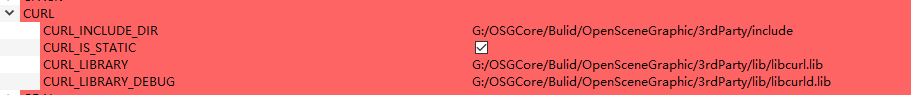


4．指定3rdParty路径：



5. 点击Configure，勾上Advanced

CURL：



GDAL：



GEOS:



ZLIB:



其余的见机行事

注意：这里需要说的是，ZLIB文件在我们使用的cmake版本中不是上面的截屏那样的，会出现上图所示四个或者三个，上面那个图是四个，还有一个对应的ZLIB\_LIBRARY\_RELEASE，那就需要按照下图的修改方法来修改，修改后再重新点击一下Configure，会出现红色错误，但是不用管。

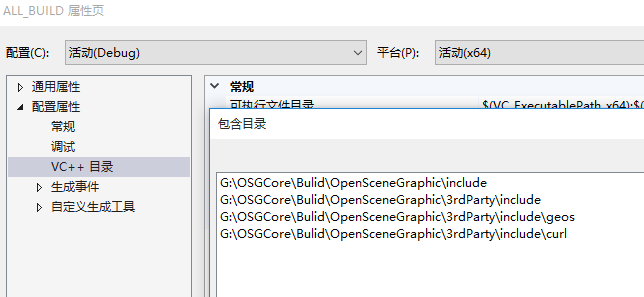


6. 编译OSGEarth

如果配置完成后，打开G:\OSGCore\OSGEarth文件夹，打开OSGEARTH工程在，注意是否是64位程序。



设置ALL\_BUILD的include 包含目录：



编译osgearth的过程中，有几个项目需要设置include包含目录，主要有：

1）osgEarthSymbology

2）osgEarthFeatures

3）osgearth\_arcgis

4）Sample osgearth\_featureinfo

5) osgEarthUtil

6) osgearth\_feature\_ogr

7) osgearth\_feature\_wfs

8) osgearth\_feature\_tfs



生成-批生成，对All\_BUILD进行生成，选择（Debug和Release两个版本），这段编译时间是很长的，估计2个小时。

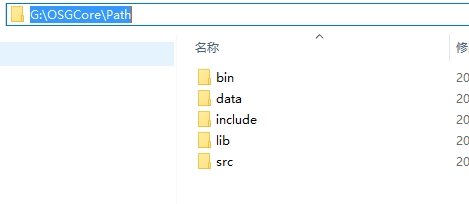
注意上述的配置include包含目录文件只是列出了debug的，release的配置方式和debug相同的，重复上述的操作即可

### 五、环境变量设置及测试

#### 软件部署（只是列出了debug，release于debug相同）

osgEarth软件包主要包括五个目录：bin,data,include,lib,src。

**1.** G:\OSGCore文件夹下建立一个文件夹path；G:\OSGCore\Path,在path下建立上述五个子目录，如下图所示：



**2.** 组织bin目录，bin主要包含dll和exe,如下表所示：

|  |  |  |
| --- | --- | --- |
| bin | osg | 第一项：.dll |
| 第二项：.exe |
| osgEarth | 第三项：.dll |
| 第四项：.exe |
| 3rdParty | 第五项：.dll |

拷贝第一项：定位G:\OSGCore\Bulid\OpenSceneGraphic\bin，搜索\*.dll，拷贝到bin

拷贝第二项：定位G:\OSGCore\OSGEarth，，搜索\*.dll，拷贝到bin

拷贝第三项：定位G:\OSGCore\OSGEarth，，搜索\*.exe，拷贝到bin

拷贝第四项：定位G:\OSGCore\Bulid\OpenSceneGraphic\3rdParty，搜索\*.dll，拷贝到bin

**3.** 组织data目录：

首先把G:\OSGCore\Bulid\OpenSceneGraphic\data下的所有拷贝到data

然后把G:\OSGCore\Bulid\osgearth-osgearth-2.8.0\data下的所有拷贝到data

首先把G:\OSGCore\Bulid\osgearth-osgearth-2.8.0\tests下的所有拷贝到data

**4.** 组织include

首先把G:\OSGCore\Bulid\OpenSceneGraphic\include下的所有拷贝到include

然后把G:\OSGCore\Bulid\osgearth-osgearth-2.8.0\src下的所有拷贝到include，可以把该目录下的.cpp文件删除，保留.h头文件。

**5.** 组织Lib同dll

|  |  |  |
| --- | --- | --- |
| lib | osg | 第一项：lib |
|  |
| osgEarth | 第二项：lib |
|  |
| 第三方库 | 第五项：lib |

拷贝第一项：定位G:\OSGCore\Bulid\OpenSceneGraphic\lib，搜索\*.lib，拷贝到lib

拷贝第二项：定位G:\OSGCore\OSGEarth\x64\Debug，搜索\*.lib，拷贝到lib

拷贝第三项：定位G:\OSGCore\Bulid\OpenSceneGraphic\3rdParty，搜索\*.lib，拷贝到lib

把osg环境变量进行一定的修改：

**OSG\_FILE\_PATH: G:\OSGCore\Path\data**

**PATH：G:\OSGCore\Path\bin**

#### 软件测试

打开cmd，输入osgearth\_viewer gdal\_tiff.earth 观察会不会出现下面的图

