<https://blog.csdn.net/gua_mass/article/details/61620481>

**一、写本文的目的**

CMake有很多复杂且与程序结构密切相关的功能，诸如跨平台编译，生成安装包，输出标准构建文档等等，如果同学们有兴趣的话可以自行学习和了解，能够很好的加深对程序架构的理解与认识。本文面向的对象是只有《c语言程序设计》这一门先修课作为基础的，参与年度计划的同学们，考虑到同学们的接受能力以及笔者自身水平有限，故只介绍cmake-gui的部分用法。以帮助大家完成年度计划任务。

**二、CMake是什么**

--CMake是一个跨平台的编译(Build)工具,可以用简单的语句来描述所有平台的编译过程。

对参加年度计划的同学们来说，我们对CMake作用的认识可以到此为止:“CMake是一个帮助我们实现编译过程的软件。”还记得在计划开始的第一周给大家留的作业吗？让大家从新建工程开始，实现一个控制台输出Hello World!的程序。

大家需要做的操作大概是如下几步：

n  新建项目

n  新建c++文件

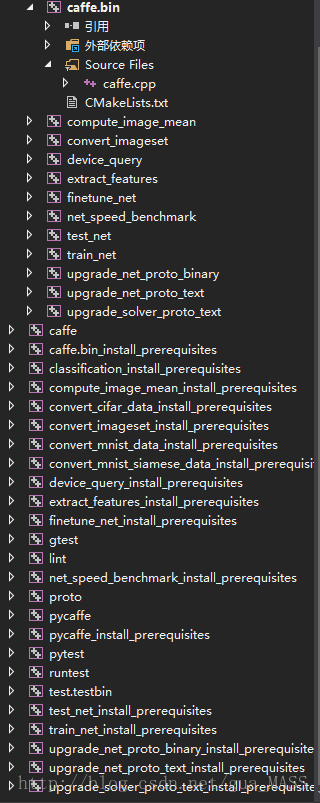
n  在c++文件内编写程序

n  添加源/头文件进工程

这是完成一个工程建立的最基本步骤。

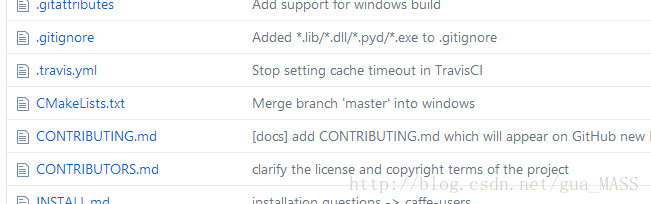
现在考虑一种更好的情况，当我们通过一些手段已经得到了我们想要的源码.cpp，这个时候我们再建立工程的步骤就更加简单了，只需要进行第一步和第四步即可。That`s the point，CMake的作用就是帮助你在拥有源码的情况下帮助你实现第一步和第四步的。

现在你应该要问“为何如此简单的操作需要使用软件来实现？”，当你们深入学习编程后，你们会意识到这是一个很蠢的问题，不过现在同学们还没有接触过真正的工程代码，所以问出这样的问题也无可厚非。



如图所示，是一个深度学习框架的部分工程列表。里面有超过40个互相调用的工程共同组成，一些用于生成库文件，一些用于实现逻辑功能。他们之间的调用关系复杂而严格，如果我想在这样复杂的框架下进行二次开发，显然只拥有它的源码是远远不够的，还需要清楚的明白这几十个项目之间的复杂关系，在没有原作者的帮助下进行这项工作几乎是不可能的。

即使是原作者给出了相关的结构文档，对新手来说建立工程的过程依旧是漫长而艰辛的，因此CMake的作用就凸显出来了。原作者只需要生成一份CMakeLists.txt文档，框架的使用者们只需要在下载源码的同时下载作者提供的CMakeLists.txt，就可以利用CMake，在”原作者的帮助下“进行工程的搭建。

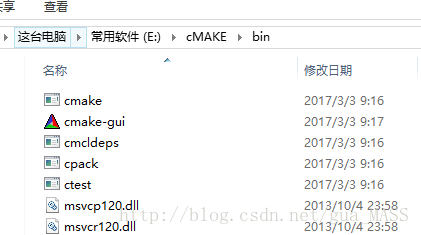


**三、CMake - gui的使用**

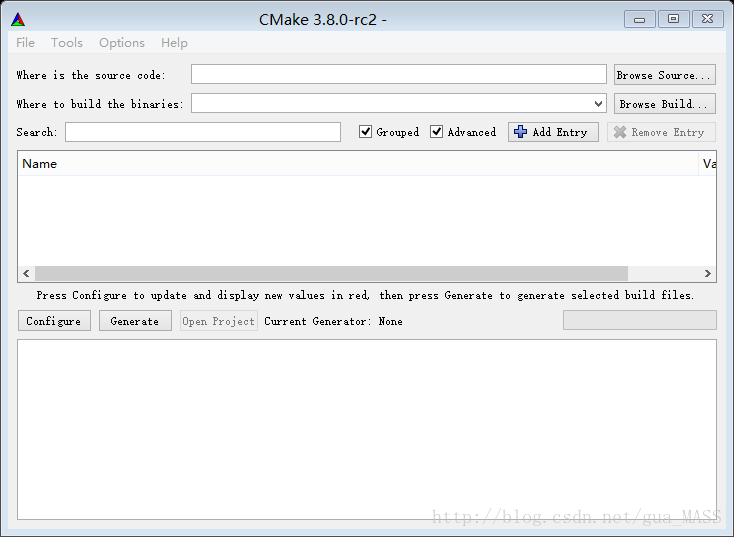


首先通过我在群里发布的安装包安装CMake。在安装时要确保系统成功的添加了CMake的环境变量。因为只有这样你才能够在cmd中启动和使用CMake。当然本文中不会介绍如何在cmd中使用CMake。

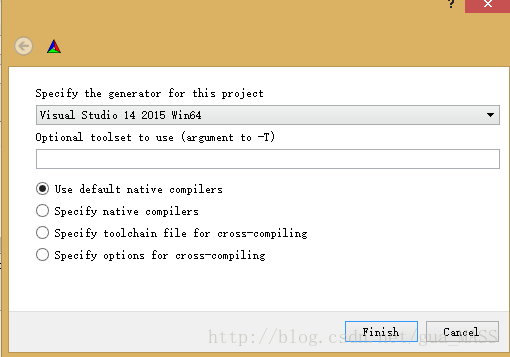
我们讨论一种更加”所见即所得“的使用方法：CMake –gui。你可以在CMake的安装路径下的bin文件夹中找到CMake – gui，如图所示。



打开后就可以进入gui。



Gui界面如图所示，如果你英文水平可以的话你应该已经不用我教的了。如果你和笔者一样从没学习过英语(XD)，那么接着往下看。在Whereis the source code后面的小白框中添加你所下载的源码路径，并且确保你将CMakeLists.txt放置其中。Where to build the binaries后的小白框是你希望将编译好的工程存放在哪个路径当中。当你完成上述工作后，点击配置按钮，用于配置你编译器类型。你可以选择32位或者64位VS或者MinGW等等，看你的需要决定。哦，我忘了你没接触过英文，配置的英文是Configure。



之后点击Finish，完成配置，如果你的配置参数（在配置和生成按钮上有个表框，里面可以修改生成工程的相关参数，如使用什么样的运算库，是否给其他语言留出接口等等，参考你所构建工程的原作者写给你的文档进行修改）没有出错的话，再点击Generate生成对应的工程文件。如果你选择VS作为IDE那么你应该会在Where to build the binaries这个目录下找到生成的.sln文件，打开工程，设置ALL\_BUILD为活动项目，编译即可。

**四、结语**

至此CMake的讲解全部完毕，当然事情肯定不会这么简单，因为大部分初学者都不会做我在括号里用红色标注的重要步骤:参考你所构建工程的原作者写给你的文档进行参数修改。这会导致你在配置时出现很多错误，你可以在这篇BLOG下留言或者给我发邮件求助。当然这都是权宜最好的解决办法就是：学！好！英！语！认真的看一下原作者的文档，才不会导致框架出现严重的错误。希望这篇BLOG可以帮助到你。参加激光雷达项目的同学们本周的作业是用CMake编译上周提供给你们的源码，并提交笔记。ANN字符识别组和人脸识别组的同学继续参考书上的内容进行学习，并提交笔记。海上目标跟踪组的同学请用OpenCV库显示一张图片，源码我会发在群里。

版权声明：本文为博主原创文章，未经博主允许不得转载。 https://blog.csdn.net/gua\_MASS/article/details/61620481