

gcc,make,cmake傻傻分不清楚?

转载

libaineu2004

2017-08-12 16:38:26

27707

收藏 73

分类专栏:LinuxCMake文章标签:gccmakefilecmake

Linux

同时被 2 个专栏收录

4 订阅

64 篇文章

订阅专栏

CMake是一种跨平台编译工具，比make更为高级，使用起来要方便得多。CMake主要是编写CMakeLists.txt文件，然后用cmake命令将CMakeLists.txt文件转化为make所需要的makefile文件，最后用make命令编译源码生成可执行程序或共享库（so(shared object)）。它的作用和qt的qmake是相似的。

作者：辉常哥

链接：<https://www.zhihu.com/question/36609459/answer/89743845>
来源：知乎
著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。

- 1.gcc是GNU Compiler Collection（就是GNU编译器套件），也可以简单认为是编译器，它可以编译很多种编程语言（括C、C++、Objective-C、Fortran、Java等等）。
- 2.当你的程序只有一个源文件时，直接就可以用gcc命令编译它。
- 3.但是当你的程序包含很多个源文件时，用gcc命令逐个去编译时，你就很容易混乱而且工作量大
- 4.所以出现了make工具
make工具可以看成是一个智能的批处理工具，它本身并没有编译和链接的功能，而是用类似于批处理的方式—通过调用makefile文件中用户指定的命令来进行编译和链接的。
- 5.makefile是什么？简单的说就像一首歌的乐谱，make工具就像指挥家，指挥家根据乐谱指挥整个乐团怎么样演奏，make工具就根据makefile中的命令进行编译和链接的。
- 6.makefile命令中就包含了调用gcc（也可以是别的编译器）去编译某个源文件的命令。
- 7.makefile在一些简单的工程完全可以人工手下，但是当工程非常大的时候，手写makefile也是非常麻烦的，如果换了个平台makefile又要重新修改。
- 8.这时候就出现了Cmake这个工具，cmake就可以更加简单的生成makefile文件给上面那个make用。当然cmake还有其他功能，就是可以跨平台生成对应平台能用的makefile，你不用再自己去修改了。
- 9.可是cmake根据什么生成makefile呢？它又要根据一个叫CMakeLists.txt文件（学名：组态档）去生成makefile。
- 10.到最后CMakeLists.txt文件谁写啊？亲，是你自己手写的。
- 11.当然如果你用IDE，类似VS这些一般它都能帮你弄好了，你只需要按一下那个三角形
- 12.cmake是make maker，生成各种可以直接控制编译过程的控制器的配置文件，比如makefile、各种IDE的配置文件。
- 13.make是一个简单的通过文件时间戳控制自动过程、处理依赖关系的软件，这个自动过程可以是编译一个项目。

作者：任卫
链接：<https://www.zhihu.com/question/36609459/answer/68231785>
来源：知乎
著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。

gcc是编译器，其实后还有连接器等
make是另外一个程序，根据makefile的指示，尽可能协调

libaineu2004

关注

36

5

73







件可能包好多源代码，也可能生成若干二进制文件，还可能指示程序套件的安装程序制作和/或安装过程，那make就通过调用编译器连接器还有其他外部工具达成这一系列工作。make不光可以调用gcc编译器，也可以调用javac啊。makefile精力集中于定义工作流水线。

CMake 是另外一个更高层一点的工具，它的输出是makefile等。支持使用更简单的语言描述编译安装过程。支持多个makefile系列、VisualStudio等

- 下午补充下推荐使用场景：
- 1 gcc编译器套件：最简单啦，单文件工程比如helloworld直接使用就行了
 - 2 make&makefile：小中大型工程肯定要用的啦，大型工程的makefile系统不少也都是噩梦般的存在。适用于工具链【较为统一】的工程中，有一定跨平台能力。
 - 3 CMake：若需要更为广泛的支持多编译工具链更广泛地跨平台且还想构建复杂的跨平台makefile系统，那CMake确实是个很好的选择，由CMake来先编译出不同平台的makefile工程，而CMakeLists.txt集中声明描述下工程组件的关系即可。

从源文件

```
hello.c
```

生成（构建）可执行文件

```
hello
```

需要执行

```
gcc hello.c -ohello
```

将上面的命令写到一个文件中，这个文件就是Makefile文件，执行make会执行Makefile中的所有命令

cmake是用来生成Makefile文件的工具，生成Makefile文件的工具不止有cmake，还有autotools。Qt环境下还有qmake

作者：知乎用户
链接：<https://www.zhihu.com/question/36609459/answer/190939645>
来源：知乎
著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。

Cmake知识----编写CMakeLists.txt文件编译C/C++程序

1.CMake编译原理

CMake是一种跨平台编译工具，比make更为高级，使用起来要方便得多。CMake主要是编写CMakeLists.txt文件，然后用cmake命令将CMakeLists.txt文件转化为make所需要的makefile文件，最后用make命令编译源码生成可执行程序或共享库（so(shared object)）。因此CMake的编译基本就两个步骤：


1. cmake

2. make


cmake 指向CMakeLists.txt所在的目录，例如cmake .. 表示CMakeLists.txt在当前目录的上一级目录。cmake后会生成很多编译的中间文件以及makefile文件，所以一般建议新建一个新的目录，专门用来编译，例如


mkdir


build


libaineu2004


关注

36

5

73





(

```
cd build
cmake ..

make
```

make根据生成makefile文件，编译程序。

2.使用Cmake编译程序

我们编写一个关于开平方的C/C++程序项目，即 $b = \sqrt{a}$ ，以此理解整个CMake编译的过程。

a.准备程序文件

文件目录结构如下：

```
.
├── build
├── CMakeLists.txt
├── include
│   └── b.h
└── src
    ├── b.c
    └── main.c
```

头文件b.h，如下所示：

```
#ifndef B_FILE_HEADER_INC

#define
    B_FIEL_HEADER_INC

#include
<math.h>

double
    cal_sqrt(double
        value);

#endif
```

头文件b.c，如下所示：

```
#include "
../include/b.h
"

double
    cal_sqrt(double
        value)
{
    return sqrt(value);
}
```

main.c主函数，如下所示：

```
#include "
../include/b.h
"

#include
<stdio.h>
int
    main(int
        argc, char
        ** argv)
{

double
```



libaineu2004

关注

👍 36

💬 5

★ 73



```
a = 49.0;

double
b = 0.0;

    printf(
"
input a:%f\n
"
,a);
    b
= cal_sqrt(a);
    printf(
"
sqrt result:%f\n
"
,b);

return
0
;
}
```

b.编写CMakeLists.txt

接下来编写CMakeLists.txt文件，该文件放在和src，include的同级目录，实际方哪里都可以，只要里面编写的路径能够正确指向就好了。CMakeLists.txt文件，如下所示：

```
1
#1
.cmake version, 指定cmake版本

2
cmake_minimum_required(VERSION 3.2
)

3

4
#2
.project name, 指定项目的名称，一般和项目的文件夹名称对应

5
PROJECT(test_sqrt)

6

7
#3
.head
file
path, 头文件目录

8
INCLUDE_DIRECTORIES(

9
include

10
)

11

12
#4
.source directory, 源文件目录
```



libaineu2004

关注

👍 36

💬 5

★ 73



(

```
13
    AUX_SOURCE_DIRECTORY(src DIR_SRCS)

14

15
    #5
    .set environment variable, 设置环境变量, 编译用到的源文件全部都要放到这里, 否则编译能够通过, 但是
    是执行的时候会出现各种问题, 比如"
    symbol lookup error xxxxx , undefined symbol
    "

16
    SET(TEST_MATH

17
    ${DIR_SRCS}

18
    )

19

20
    #6
    .add executable file
    , 添加要编译的可执行文件

21
    ADD_EXECUTABLE(${PROJECT_NAME} ${TEST_MATH})

22

23
    #7
    .add link library, 添加可执行文件所需要的库, 比如我们用到了libm.so (命名规则:
    lib+name+.so) , 就添加该库的名称

24
    TARGET_LINK_LIBRARIES(${PROJECT_NAME} m)
```

CMakeLists.txt主要包含以上的7个步骤, 具体的意义, 请阅读相应的注释。

c.编译和运行程序

准备好了以上的所有材料, 接下来, 就可以编译了, 由于编译中出现许多中间的文件, 因此最好新建一个独立的目录build, 在该目录下进行编译, 编译步骤如下所示:

```
mkdir
  build
cd build
cmake ..

make
```

操作后, 在build下生成的目录结构如下:

```
├── build
│   ├── CMakeCache.txt
│   ├── CMakeFiles
│   │   └──
3.2
.2
│   │   │   ├── CMakeCCompiler.cmake
│   │   │   ├── CMakeCXXCompiler.cmake
│   │   │   ├── CMakeDetermineCompilerABI_C.bin
│   │   │   ├── CMakeDetermineCompilerABI_CXX.bin
│   │   │   ├── CMakeSystem.cmake
│   │   │   ├── CompilerIdC
│   │   │   │   ├── a.out
│   │   │   │   └── CMakeCCompilerId.c
```



libaineu2004

关注

36

5

73



(

```
| | | | | CompilerIdCXX
| | | | |   | a.out
| | | | |   | CMakeCXXCompilerId.
cpp
| | | | | cmake.check_cache
| | | | | CMakeDirectoryInformation.cmake
| | | | | CMakeOutput.log
| | | | | CMakeTmp
| | | | | feature_tests.bin
| | | | | feature_tests.c
| | | | | feature_tests.cxx
| | | | | Makefile2
| | | | | Makefile.cmake
| | | | | progress.marks
| | | | | TargetDirectories.txt
| | | | | test_sqrt.
dir
| | | | | build.
make
| | | | | C.includecache
| | | | | cmake_clean.cmake
| | | | | DependInfo.cmake
| | | | | depend.internal
| | | | | depend.
make
| | | | | flags.
make
| | | | | link.txt
| | | | | progress.
make
| | | | | src
| | | | |   | b.c.o
| | | | |   | main.c.o
| | | | | cmake_install.cmake
| | | | | Makefile
| | | | | test_sqrt
| | | | | CMakeLists.txt
| | | | | include
| | | | |   | b.h
| | | | | src
| | | | |   | b.c
| | | | |   | main.c
```

注意在build的目录下生成了一个可执行的文件test_sqrt,运行获取结果如下：

```
命令：
.
/test_sqrt
结果：
input a:
49.000000

sqrt result:
7.000000
```

d.源码

地址: [test_sqrt.tar.gz](#)

3.参考资料

[1]. [CMake 使用方法 & CMakeList.txt](#)



libaineu2004

关注

36

5

73



(

全面学习make工具,学写makefile,gcc全部的选项详解

05-05

全面学习make工具教程，使用MAKE。学写makefile,对项目工程进行管理；gcc全部的选项详解，绝对最全面的翻...

cmake 指定 gcc 版本

量化窝 7022

centos的gcc版本一般比较低，有时候需要用到高版本的gcc，所以会自己装一个。cmake执行编译的时候，默认使...

优质评论可以帮助作者获得更高权重

评论

来碗豆腐脑：小白被clangd折腾的死去活来，cmake怎么生成compilie_command.json?..... 3 月前

回复

waruqi：还有个 xmake 1 年前

回复

weixin_43409736：linux小白，学到了，谢谢老铁分享 1 年前

回复

登录 查看 5 条热评

相关推荐

更多相似内容

GCC,Cmake,make,makefile傻傻分不清楚_u010029134的博客

7-14

GCC,Cmake,make,makefile傻傻分不清楚 1. GCC:GNU编译器套件,可以编译多种语言(包括C,C++,Objective-C,Fo...

gcc、g++、gdb、make、cmake傻傻分不清_m0_51962318的...

8-10

makefile命令中就包含了调用gcc(也可以是别的编译器)去编译某个源文件的命令。 维基百科make CMake CMake(...

浅述gcc、Make、CMake的作用及关系

Knight-ZXW的博客 4792

gcc gcc 是GNU编译器套件，是Linux下默认的C/C++编译器.在windows环境下可以通过MinGw等GNU for Window...

gcc make makefile 用法

xueyul的博客 6219

一.gcc 编译指令: gcc t.c -o t生产可执行文件 此编译过程是分为四个阶段进行的： 1.预处理(也称预编译，Preproc...

linux编译器的下关于gcc、g++、make和CMake几个概念的...

8-28

CMake是一种跨平台编译工具,比make更为高级,使用起来要方便得多。CMake主要是编写CMakeLists.txt文件,然...

Linux下gcc/g++、make和cmake的区别_Enzo 想砸电脑的博客

8-13

但是随之而来也就是编写cmakelist文件,它是cmake所依据的规则。所以在编程的世界里没有捷径可走,还是要脚踏...

gcc与make的定义和区别

u010879745的博客 1328

作者：辉常哥 链接：https://www.zhihu.com/question/36609459/answer/89743845 来源：知乎 著作权归作者所有...

Cmake学习笔记（一）：认识gcc、make、cmake

qq_20123207的博客 3403

本人邮箱jinbo666888@qq.com，欢迎交流。我们写的程序语言机器并不能理解，机器语言是010101这样的二进...

编译器gcc、clang、make、cmake辨析_A big bliss的博客

8-5

&& make gcc,make,cmake傻傻分不清楚? \$_firecat全宏的代码足迹\$ 2万+ CMake是一种跨平台编译工具,比make更...

【Linux内核实验】gcc与make使用

w1020187000的博客 1494

1.实验任务 1) 熟悉并基本掌握GCC的基础知识 2) 为自己的工程编写makefile文件 2.实验内容 2.1 gcc 执行过...

gcc和make的区别

越努力，越幸运 707

1. gcc是编译器 而make不是 make是依赖于Makefile来编译多个源文件的工具 在Makefile里同样是用gcc(或者别的...

Linux下编程工具---gcc和make

c++小白的博客 435

文章目录c语言编译过程1、编译预处理2、编译阶段3、优化阶段4、汇编过程5、链接程序GCC1、预处理,生成.i ...

【工具】Cmake与gcc的关系

weixin_30498921的博客 97

1.gcc是GNU Compiler Collection（就是GNU编译器套件），也可以简单认为是编译器，它可以编译很多种编程语...

go_cmake:使用cmake 构建一个 go 项目-源码

06-30

go_cmake 假设您是一家商店。继续断言您将用于一两个项目。Go 有一个经过深思熟虑的构建系统。可以让...

关于gcc、make和CMake的区别

weixin_30352191的博客 489

CMake是一种跨平台编译工具，比make更为高级，使用起来要方便得多。CMake主要是编写CMakeLists.txt文件...

gcc、make、makefile、cmake、cmakelists区别

午后斜阳 7071

转自：http://www.zhihu.com/question/36609459 辉常哥1.gcc是GNU Compiler Collection（就是GNU编译器套...

(转) **gcc与cmake的关系** weixin_43257951的博客 16
gcc与cmake的关系 1.**gcc**是GNU Compiler Collection（就是GNU编译器套件），也可以简单认为是编译器，它可...

make太慢了，加快编译速度的方法 **make -j** 熊孩子的博客???

make -j 既然IO不是瓶颈，那CPU就应该是一个影响编译速度的重要因素了。 3万+

用**make -j**带一个参数，可以...

使用 CMake 打造 Golang 风格的单元测试框架

agresss的专栏 · 832

参考文档: CMake Wiki

公安备案号11010502030143 京ICP备19004658号 京网文〔2020〕1039-165号 经营性网站备案信息 北京互联网违法和不良信息举报中心
网络110报警服务 中国互联网举报中心 家长监护 Chrome商店下载 ©1999-2021北京创新乐知网络技术有限公司 版权与免责声明 版权申诉
出版物许可证 营业执照

libaineu2004 关注

36 5 73 (

Qt作品

10篇

最新评论

我的Qt作品（5）使用Qt+Halcon实现模...
libaineu2004: 加QQ私聊

我的Qt作品（5）使用Qt+Halcon实现模...
chuxuezhewei: 博主您好,看到您这份文章
很兴奋, 请问可以分享下源码吗? 1466i ...

LNK2019: 无法解析的外部符号 __imp_R...
ZhiShiiii: 输入了之后 还是会有无法解析的
外部符号 这怎么办

使用OpenCV实现Halcon算法（1）亚像...
m0_50840355: 图片和配置文件能提供一下
吗, 运行自己的图片和参数不一样

彻底解决Qt中文乱码以及汉字编码的问题...
弦乐四重奏: 感谢博主!

您愿意向朋友推荐“博客详情页”吗?

强烈不推荐 不推荐 一般般 推荐 强烈推荐

最新文章

LNK2019: 无法解析的外部符号
__imp_RegisterDeviceNotification

OpenCV图像增强算子

使用OpenCV实现Halcon算法（6）
dyn_threshold算子

2021

08月
12篇

07月
7篇

06月
4篇

05月
6篇

04月
9篇

03月
8篇

02月
11篇

01月
10篇

2020年 224篇

2019年 175篇

2018年 287篇

2017年 132篇

2016年 10篇

2015年 209篇

2014年 152篇

2013年 2篇

广告 X

New NDT Whitepapers Available

Learn more about the
benefits of CT inline
inspection for the
automotive industr

Waygate Technolog

Open



libaineu2004

关注

36

5

73



(