

cmake整理：在编译时拷贝文件之add_custom_comand 和 add_custom_target

原创

2016年04月08日 14:59:59

标签：cmake / cmake拷贝 / cmake命令

6491

第一种通用形式：

add_custom_command：增加定制化的构建规则到生成的构建系统中。对于add_custom_command，有两种使用形式，一种形式是增加一个定制命令用来产生一个输出。

```
add_custom_command(OUTPUT output1 [output2 ...]
                    COMMAND command1[ARGS] [args1...]
                    [COMMAND command2 [ARGS] [args2...] ...]
                    [MAIN_DEPENDENCYdepend]
                    [DEPENDS[depends...]]
                    [IMPLICIT_DEPENDS<lang1> depend1 ...]
                    [WORKING_DIRECTORYdir]
                    [COMMENT comment] [VERBATIM] [APPEND])
```

不要同时在多个相互独立的目标中执行上述命令产生相同的文件，主要是为了防止冲突产生。如果有多条命令，它们将会按顺序执行。ARGS是为了向后兼容，使用过程中可以忽略。MAIN_DEPENDENCY完全是可选的，它主要是针对Visual Studio给出的一个建议。在Makefile中，它会产生一个这样的新目标：

加入CSDN，享受更精准的内容推荐，与500万程序员共同成长！



放牛娃不吃草

原创

207

粉丝

303

喜欢

405

评论

176



等级：博客 7

访问量：118万+

积分：1万+


排名：885



世界豪华车排名



OUTPUT: MAIN_DEPENDENCY DEPENDS
COMMAND

第二种形式是为某个目标如库或可执行程序添加一个定制命令。这对于要在构建一个目标之前或之后执行一些操作非常有用。该命令本身会成为目标的一部分，仅在目标本身被构建时才会执行。如果该目标已经构建， 将不会执行。

第二种：标记为在什么时候执行命令：编译前，编译后，链接前

```
add_custom_command(TARGET target
    PRE_BUILD | PRE_LINK | POST_BUILD
    COMMAND command1[ARGS] [args1...]
    [COMMENT comment2[ARGS] [args2...] ...]
    [WORKING_DIRECTORY dir]
    [COMMENT comment][VERBATIM])
```

命令执行的时机由如下数字决定：

PRE_BUILD - 命令将会在其他依赖项执行前执行
PRE_LINK - 命令将会在其他依赖项执行完后执行
POST_BUILD - 命令将会在其他依赖项构建完后执行。

其中，PRE_BUILD只被Visual Studio 7及之后的版本支持，其他所有的构建文件产生器将视PRE_BUILD为PRE_LINK。如果指定了WORKING_DIRECTORY，那么命令将会在指定的目录下执行。如果是相对路径，那么该路径将被解释为与当前源码目录对应的构建目录相对的路径。如果设置了COMMENT，那么在编译时，命令执行前会将COMMENT的内容当做信息输出。如果指定了APPEND

Scrapy笔记

Coursera吴恩达《神经网络与深度学习》
课程笔记（5）-- 深层神经网络

Coursera吴恩达《神经网络与深度学习》
课程笔记（4）-- 浅层神经网络

Coursera吴恩达《神经网络与深度学习》
课程笔记（3）-- 神经网络基础之Python与向量化

Coursera吴恩达《神经网络与深度学习》
课程笔记（2）-- 神经网络基础之逻辑回归

文章分类

AI	8篇
boost	53篇
linux系统常用	52篇
openwrt内核	18篇
交叉编译	7篇
C/C++/.NET/汇编	43篇

[展开](#)

博主专栏

加入CSDN，享受更精准的内容推荐，与500万程序员共同成长！

[登录](#)

[注册](#)

CY选项将会忽略。但是将来可能会使用。如果指定了VERBATIM选项，那么，所有传递到命令的参数将会被适当地转义，这样命令接受到的参数将不会改变。建议使用VERBATIM选项，如果客制命令的输出不是创建一个存储在磁盘上的文件，需要使用命令SET_SOURCE_FILES_PROPERTIES把它标记为SYMBOLIC。

IMPLICIT_DEPENDS选项请求扫描一个输入文件的隐含依赖项。特定的语言会指明对应编程语言，它会使用相应的依赖项扫描器。目前仅支持C和CXX语言依赖项扫描器。目前IMPLICIT_DEPENDS选项仅被Makefile生成器支持，其它构建文件的产生器将会忽略该选项。

如果COMMAND指定了一个可执行的目标（由ADD_EXECUTABLE创建），那么它会自动地被在构建时创建的可执行目标路径替换。另外，也会添加一个目标级的依赖，使得可执行目标总会在使用了该客制命令的任何目标之前构建。然而，它不会增加一个文件级的依赖，这种依赖会使得只要该可执行程序被重新编译，该客制命令也会重新运行。

DEPENDS选项指定命令所依赖的文件。如果任何依赖项是同一目录中其他另一个客制命令 OUTPUT(CMakeLists.txt)。那么CMake会自动地将其引入到执行该客制命令的目标中来。如果没有指定DEPENDS，那么只要OUTPUT不见了，该命令就会运行。如果该命令并没有实际去创建OUTPUT，那么该规则总是执行。如果DEPENDS指定了任何一个目标（由ADD_*系列命令创建），那么就会创建一个目标级的依赖以确保该目标比任何使用该客制命令的目标要先构建。另外，如果该目标是一个可执行文件或是一个库，那么就会创建一个文件级的依赖，这样会使得只要该目标重新编译，该客制命令就会重新运行。

重要：使用 target 标记为总是过期，这样每次编译时都会执行上面规定的 command

- **add_custom_target**: 增加一个没有输出的目标，使得它总是被构建。 `add_custom_target(Na`

加入CSDN，享受更精准的内容推荐，与500万程序员共同成长！



11 篇

嵌入式系统学习

75415

文章存档

2018年3月	1篇
2018年2月	7篇
2017年9月	2篇
2017年8月	1篇
2017年6月	1篇
2017年4月	7篇

[展开](#)

他的热门文章

nginx+ffmpeg搭建rtmp直播rtsp流的flash服务器

19686

OpenWRT 教程 之 设置网络和无线

16428

Live555接收h264使用ffmpeg解码为YUV420

14586

```
[DEPENDS depend depend depend ... ]
[WORKING_DIRECTORY dir]
[COMMENT comment] [VERBATIM]
[SOURCES src1 [src2...]]
```

增加一个指定名字的目标，并执行指定的命令。该目标没有输出文件，总是被认为是过期的，即使是在试图用目标的名称来构建一个文件。使用ADD_CUSTOM_COMMAND命令来创建一个具有依赖项的文件。默认情况下，没有任何目标会依赖该定制目标。使用ADD_DEPENDENCIES来添加依赖项或成为别的目标的依赖项。如果指定了ALL选项，那就表明该目标会被添加到默认的构建目标，使得它每次都构建。(该命令的名称不能命名为ALL)。命令和参数都是可选的，如果没有指定，将会创建一个空目标。如果设置了WORKING_DIRECTORY，那么该命令将会在指定的目录中运行。如果它是个路径，那它会被解析为相对于当前源码目录对应的构建目录。如果设置了COMMENT，在构建的时候，该值会被当成信息在执行该命令之前显示。DEPENDS参数可以是文件和同一目录中的其他命令的输出。

如果指定了VERBATIM，所有传递给命令的参数将会被适当地转义。建议使用该选项。

SOURCES选项指定了包含进该定制目标的额外的源文件。即使这些源文件没有构建规则，但是它们会被增加到IDE的工程文件中以方便编辑。

例子：

ONVIF客户端搜索设备获取rtsp地址开发笔记(精华篇)

12153

解决openwrt ipk missing dependencies libpthread librt

11102

Openwrt挂载移动硬盘，U盘

10837

macOS上安装opencv3笔记

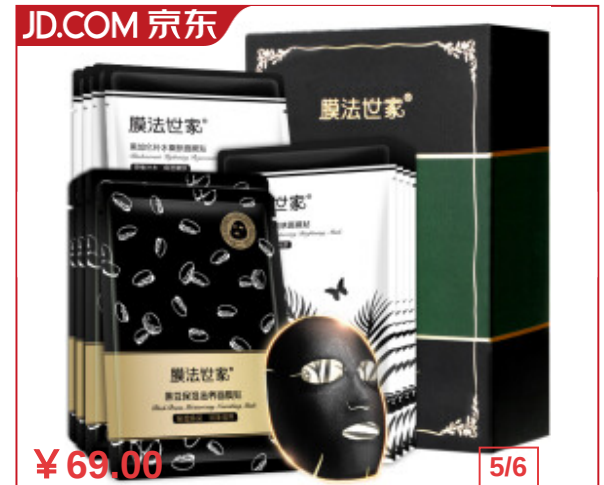
10722

C++哈希表使用教程（STL）

10564

cmake 常用变量和常用环境变量查表手册--整理

10236



加入CSDN，享受更精准的内容推荐，与500万程序员共同成长！

登录

注册

```
COMMAND echo "Generating log.txt file..."
COMMAND ${CMAKE_COMMAND} -E copy ${CMAKE_CURRENT_LIST_FILE} ${TEST_FILE}
COMMENT "This is a test"
)
add_custom_target(Test1 ALL DEPENDS ${TEST_FILE})
```

```
add_custom_command(TARGET Test1
PRE_BUILD
COMMAND echo "Executing a fake command"
COMMENT "This command will be executed before building target Test1"
)
```

结果：

```
[100%] This is a test
Generating log.txt file...
This command will be executed before building target Test1
executing a fake command
[100%] Built target Test1
```

(这是 make 时候进行执行)



请扫描二维码联系客服

✉ webmaster@csdn.net

☎ 400-660-0108

🗨 QQ客服 🗨 客服论坛

关于 招聘 广告服务 百度

©1999-2018 CSDN版权所有

京ICP证09002463号

经营性网站备案信息

网络110报警服务

中国互联网举报中心

北京互联网违法和不良信息举报中心

```

3. add_custom_command(OUTPUT COPY_RES
4.   COMMAND ${CMAKE_COMMAND} -E copy_directory ${CMAKE_CURRENT_SOURCE_DIR}/${COPYITE
M} ${EXECUTABLE_OUTPUT_PATH}/${COPYITEM}
5.   )
6.
7. add_custom_target(CopyTask ALL DEPENDS COPY_RES)

```

另外一种用法：直接输入命令：

```

[plain]
1. execute_pro COMMAND ${CMAKE_COMMAND} -E chdir ${CMAKE_CURRENT_SOURCE_DIR} $ENV{PROT
OC} --cpp_out=. proto/Echo.proto conti/beagent/msg_catalog.proto
2.   COMM [CMAKE_COMMAND] -E chdir ${CMAKE_CURRENT_SOURCE_DIR} $ENV{PROTOC} --cpp
_out=. prot o.proto conti/beagent/connmgr/topic_parameterize.proto
3.   COMMAND ${CMAKE_COMMAND} -E chdir ${CMAKE_CURRENT_SOURCE_DIR} $ENV{PROTOC} --cpp
_out=. proto/Echo.proto conti/beagent/connmgr/subscribe.proto
4.   COMMAND ${CMAKE_COMMAND} -E chdir ${CMAKE_CURRENT_SOURCE_DIR} $ENV{PROTOC} --cpp
_out=. proto/Echo.proto conti/beagent/connmgr/service_connection_manager.proto
5.   COMMAND ${CMAKE_COMMAND} -E chdir ${CMAKE_CURRENT_SOURCE_DIR} $ENV{PROTOC} --cpp
_out=. proto/Echo.proto conti/beagent/connmgr/out_data.proto
6.   COMMAND ${CMAKE_COMMAND} -E chdir ${CMAKE_CURRENT_SOURCE_DIR} $ENV{PROTOC} --cpp
_out=. proto/Echo.proto conti/beagent/connmgr/in_data.proto
7.   COMMAND ${CMAKE_COMMAND} -E chdir ${CMAKE_CURRENT_SOURCE_DIR} $ENV{PROTOC} --cpp
_out=. proto/Echo.proto conti/beagent/connmgr/app_status.proto
8.   COMMAND ${CMAKE_COMMAND} -E chdir ${CMAKE_CURRENT_SOURCE_DIR} $ENV{PROTOC} --cpp
_out=. proto/Echo.proto conti/beagent/connmgr/advertise.proto
9.   COMMAND ${CMAKE_COMMAND} -E chdir ${CMAKE_CURRENT_SOURCE_DIR} $ENV{PROTOC} --cpp
_out=. proto/mqtt_diag_message.proto
10.  COMMAND ${CMAKE_COMMAND} -E chdir ${CMAKE_CURRENT_SOURCE_DIR} $ENV{PROTOC} --cpp
_out=. proto/carriers.proto
11.   )

```

加入CSDN，享受更精准的内容推荐，与500万程序员共同成长！

登录

注册

(这是 cmake 时候进行执行)



版权声明：本文为博主原创文章，未经博主允许不得转载。

<http://blog.csdn.net/luopeiyuan1990/article/details/51096777>



目前您尚未登录，请 [登录](#) 或 [注册](#) 后进行评论

CMake命令之add_custom_comand 和 add_custom_target

加入CSDN，享受更精准的内容推荐，与500万程序员共同成长！

[登录](#)

[注册](#)



fuyajun01 2013年05月09日 19:34 14265

CMake指令集锦



wuzhou185 2015年05月21日 15:02 1909

整合了CMake的常用指令，并给出了每条指令的简述及用法，最后还给出CMake中相关的信息变量的含义...



恭喜！领取您的阿里云服务免费套餐>>

阿里云，阿里巴巴集团云计算品牌，全球卓越的云计算技术和服务提供商。



CMake如何执行 shell 命令



crazyhacking 2014年07月03日 18:07 9287

摘要：我在cmake编译后想执行一些特定的shell命令（执行、lcof收集代码覆盖报告等），我又不想写到XX.sh的shell脚本中，如何直接通过CMake执行shell命令呢？在网上翻...

cmake使用



kl222 2012年12月05日 10:26 15826

【语法】 注释 #： #我是注释 命令语法 COMMAND： COMMAND(参数1 参数2 ...) 字符串列 A;B...

CMake初步（2）



ZXGIS 2011年12月14日 11:03 8958

转自：《你所不知的OSG》第一章：CMake初步（2）
<http://bbs.osgchina.org/forum.php?mod=viewthread&tid=1229&fromuid=3434> ...



Cmake用到的语句记录



Zzhouzhou237 2017年07月17日 09:28 907

1、如题：Implant 下有文件夹Nails、Plates，希望用Cmake命令将每个文件夹中的文件拷贝到Bin目录下，CMakeLists.txt与Implant放到同一目录下，可在CMake...

CMakeFile命令



bytxl 2015年09月08日 10:25 3693

file:文件操作命令. file(WRITE filename "message towrite"...) WRITE 将一则信息写入文件'filename'中，如果该文件存在...

cmake使用示例与整理总结



wzzfeitian 2014年11月10日 00:37 104663

本文源码托管于github上cmake_demo cmake中一些预定义变量 PROJECT_SOURCE_DIR 工程的根目录PROJECT_BINARY_DIR 运行cmake命令的目录,通...

自己创建XXConfig.cmake文件



historycomputer 2017年03月01日 19:42 2575

创建自定义的XXConfig.cmake文件：我们在使用cmake进行代码构建的时候，当需要通过find_package(XXX)来配置相关库的路径（头文件路径、链接库路径）的时候，如果没有在系统环境...

ros catkin_make 出现add_custom_target cannot create target 错误

今天用catkin_make编译ros包出现如下错误: CMake Error at /home/linus/work/catkin_ws/src/build/my_package/CMakeLists.txt:10 (add_custom_target):

加入CSDN，享受更精准的内容推荐，与500万程序员共同成长！

[登录](#)[注册](#)



qq_29573053 2017年06月29日 11:01 415

cmake 那些事儿 ~



willability 2015年08月12日 11:25 1232

程序引入cmake

嘿！阿里云服务，今日起可免费体验>>

阿里巴巴集团旗下云计算“牌”，全球卓越的云计算技术和服务提供商。



CMakeFile命令之file



fuyajun01 2013年05月03日 14:38 5928

file:文件操作命令。file(filename "message towrite"...) WRITE 将一则信息写入文件'filename'中，如果该文件存在，它会覆...

CMake 常用命令解析



tiemaxiaosu 2016年10月21日 11:42 2917

参考： http://www.cnblogs.com/coderfenghc/archive/2012/06/16/CMake_ch_01.html 一、常用命令解析 1、CMAKE_MINIM...

cmake的使用笔记



cartzhang 2014年03月19日 11:04 3384

1. cmake使用网上还蛮多的。可是当你自己去用的时候，才觉得，都不是你想要的。 自己做个记录，用法全是基本用法。 2. 我在里面直接写注释说明了，直接贴了。 #cmake #1. # c...

加入CSDN，享受更精准的内容推荐，与500万程序员共同成长！

登录

注册

本博文的大概框架：内里有重复，待有时间重新整理 1，cmake 的介绍，下载，安装和使用 2，cmake 的手册详解，我关注了 -C和-G 的使用 3，在Linux中构建cmake 的工程...

cmake使用方法详解



tianmo2010

2013年06月14日 17:35

8093

cmake 简介 CMake是一个跨平台的安装(编译)工具,可以用简单的语句来描述所有平台的安装(编译过程)。他能够输出各种各样的makefile或project文件,能测试编译器所支持...



1

CMake入门指南



GarfieldEr007

2016年05月10日 20:54

3423

CMake是一个比make更高级的编译配置工具，它可以根据不同平台、不同的编译器，生成相应的Makefile或者vcproj项目。通过编写CMake.txt，可以控制生成的Makefil...



燕山二手房出售



个人房源房屋出售

百度广告

CMake常见指令总结，修改的更加容易理解

PROJECT(工程名字) 这条指令会自动创建两个变量：_BINARY_DIR(二进制文件保存路径) _SOURCE_DIR (源代码路径) cmake系统也帮助我们预定义了PR...



u012839187

2015年05月17日 17:28

7622

Cmake使用心得



jonathan321

2016年07月21日 22:27

2226

加入CSDN，享受更精准的内容推荐，与500万程序员共同成长！

登录

注册

Cmake使用指南



fan_hai_ping

2012年11月21日 17:10

11994

1.Cmake简介来自于：<http://zh.wikipedia.org/wiki/CMake> CMake是一个跨平台的安装(编译)工具，可以用简单的语句来描述所有平台的安装(编译过程)。他...



1



加入CSDN，享受更精准的内容推荐，与500万程序员共同成长！

[登录](#)[注册](#)