```
CMake使用protobuf生成c++代码
```

```
明霁天明
            ● 于 2021-02-06 18:32:03 发布 ● 7003 🏚 收藏 39
                                                                                                     版权
    分类专栏: CMake 文章标签: cmake c++ protobuf
      CMake 专栏收录该内容
                                                                                                订阅专栏
                                                                                       2篇文章
                                                                                 1订阅
目录标题
    CMake编译protobuf生成c++代码
         1. protobuf_generate_cpp生成源码
         2.使用execute process命令生成源码
         3.使用add_custom_target与add_custom_command生成源码
```

## 4.总结

本文为个人使用cmake编译protobuf生成源码的经验总结,下面将介绍三种生成protobuf源码的cmake编写方式 1. protobuf\_generate\_cpp生成源码

cmake提供了FindProtobuf模块,可以通过find\_package命令查找Protobuf进行使用,官网给的使用示例如下:

CMake编译protobuf生成c++代码

find\_package(Protobuf REQUIRED) include\_directories(\${Protobuf\_INCLUDE\_DIRS})

```
include_directories(${CMAKE_CURRENT_BINARY_DIR})
      protobuf_generate_cpp(PROTO_SRCS PROTO_HDRS foo.proto)
      protobuf_generate_cpp(PROTO_SRCS PROTO_HDRS EXPORT_MACRO DLL_EXPORT foo.proto)
      protobuf_generate_cpp(PR0T0_SRCS PR0T0_HDRS DESCRIPTORS PR0T0_DESCS foo.proto)
      protobuf_generate_python(PROTO_PY foo.proto)
      add_executable(bar bar.cc ${PROTO_SRCS} ${PROTO_HDRS})
      target_link_libraries(bar ${Protobuf_LIBRARIES})
这里使用protobuf_generate_cpp命令将foo.proto文件生成源码,使用PROTO_SRC,PROTO_HARS变量分别指代生成的cpp和h文件并可用于连接到
target和设置include
不过这种方法有两个<mark>缺点</mark>:
```

● 要求protobuf\_generate\_cpp命令和生成add\_executable() 或 add\_library() 的命令必须在同一个CMakeList中. ● 该方法(当前3.18)仍无法设置源码的生成路径,只能默认在相应的build-tree中生成

- 2.使用execute\_process命令生成源码 为解决方法一中的缺点,可以使用cmake中的execute\_process命令调用protoc程序来自定义生成源码的方法,示例如下:

## #设置输出路径 (MESSAGE\_DIR \${CMAKE\_BINARY\_DIR}/message)

find\_package(Protobuf 3 REQUIRED)

if(EXISTS "\${CMAKE\_BINARY\_DIR}/message" AND IS\_DIRECTORY "\${CMAKE\_BINARY\_DIR}/message")

```
SET(DST_DIR ${MESSAGE_DIR})
    else()
 8
            file(MAKE_DIRECTORY ${MESSAGE_DIR})
            SET(DST_DIR ${MESSAGE_DIR})
    endif()
10
11
    #设置protoc的搜索路径
13
    LIST(APPEND PROTO_FLAGS -I${CMAKE_SOURCE_DIR}/msg/message)
14
   #获取需要编译的proto文件
   file(GLOB_RECURSE MSG_PROTOS ${CMAKE_SOURCE_DIR}/msg/message/*.proto)
17
    set(MESSAGE_SRC "")
    set(MESSAGE_HDRS "")
19
    foreach(msg ${MSG_PROTOS})
20
            get_filename_component(FIL_WE ${msg} NAME_WE)
21
22
            list(APPEND MESSAGE_SRC "${PROJECT_BINARY_DIR}/message/${FIL_WE}.pb.cc")
23
            list(APPEND MESSAGE_HDRS "${PROJECT_BINARY_DIR}/message/${FIL_WE}.pb.h")
24
25
            # 生成源码
26
            execute_process(
               COMMAND ${PROTOBUF_PROTOC_EXECUTABLE} ${PROTO_FLAGS} --cpp_out=${DST_DIR} ${msg}
                                                          ~
```

● \$SRC\_DIR/addressbook.proto 为需要编译的proto文件

● MESSAGE\_SRC和MESSAGE\_HDRS可以用于连接target和设置include\_directories

SET(MESSAGE\_DIR \${CMAKE\_BINARY\_DIR}/message)

file(MAKE\_DIRECTORY \${MESSAGE\_DIR})

SET(PROTO\_META\_BASE\_DIR \${MESSAGE\_DIR})

LIST(APPEND PROTO\_FLAGS -I\${CMAKE\_SOURCE\_DIR}/msg/message)

1 | protoc -I=\$SRC\_DIR --cpp\_out=\$DST\_DIR \$SRC\_DIR/addressbook.proto

在示例中PROTOBUF\_PROTOC\_EXECUTABLE指代protobuf生成源码的可执行程序protoc,其使用格式为

## 在示例代码中

其中

• 首先设置了源码输出路径DST\_DIR并且当DST\_DIR不存在时生成该目录;

-I=\$SRC\_DIR 为编译时搜索proto的根目录

−cpp\_out=\$DST\_DIR为源码输出路径

之后通过\*LIST(APPEND PROTO\_FLAGS -I\${CMAKE\_SOURCE\_DIR}/msg/message)\*对protoc的搜索路径进行设置;

add\_custom\_command命令

#设置输出路径

endif()

#设置protoc的搜索路径

8

10

11

12

4.总结

- 这种方法仍然存在<mark>缺点</mark>:每次执行cmake后,都会重新生成proto源码,导致make时会因为源码变动(内容未变,只是重新生成)而重新编译程序 3.使用add\_custom\_target与add\_custom\_command生成源码
- find\_package(Protobuf 3 REQUIRED)

为解决方法二中重新编译的问题,在该方法中引入add\_custom\_target命令生成一个自定义target,并令该target依赖于生成源码的

● 最后execute\_process命令遍历中的每个proto文件执行protoc命令,并将生成的源码分别追加到变量MESSAGE\_SRC和MESSAGE\_HDRS中;

if(EXISTS "\${CMAKE\_BINARY\_DIR}/message" AND IS\_DIRECTORY "\${CMAKE\_BINARY\_DIR}/message") SET(PROTO\_META\_BASE\_DIR \${MESSAGE\_DIR}) else()

```
14
     #获取需要编译的proto文件
     file(GLOB_RECURSE MSG_PROTOS ${CMAKE_SOURCE_DIR}/msg/message/*.proto)
     set(MESSAGE_SRC "")
     set(MESSAGE_HDRS "")
  17
  18
  19
     foreach(msg ${MSG_PR0T0S})
  20
            get_filename_component(FIL_WE ${msg} NAME_WE)
 21
  22
            list(APPEND MESSAGE_SRC "${PROJECT_BINARY_DIR}/message/${FIL_WE}.pb.cc")
  23
            list(APPEND MESSAGE_HDRS "${PROJECT_BINARY_DIR}/message/${FIL_WE}.pb.h")
  24
  25
            # 使用自定义命令
  26
            add_custom_command(
 27
              OUTPUT "${PROJECT_BINARY_DIR}/message/${FIL_WE}.pb.cc"
  28
                    "${PROJECT_BINARY_DIR}/message/${FIL_WE}.pb.h"
  29
              COMMAND ${PROTOBUF_PROTOC_EXECUTABLE}
  30
              ARGS --cpp_out ${PROTO_META_BASE_DIR}
 31
                -I ${CMAKE_SOURCE_DIR}/msg/message
  32
               ${msg}
  33
              DEPENDS ${msg}
  34
              COMMENT "Running C++ protocol buffer compiler on ${msg}"
  35
              VERBATIM
  36
 37
     endforeach()
  38
  39
     # 设置文件属性为 GENERATED
     set_source_files_properties(${MESSAGE_SRC} ${MESSAGE_HDRS} PROPERTIES GENERATED TRUE)
  41
     # 添加自定义target
     add_custom_target(generate_message ALL
                                                      ~
可以看到本方法的前一部分与方法二相似,不同点只在于使用了add_custom_command替换了execute_process命令,将add_custom_command的
OUTPUT与add_custom_target的DEPENDS绑定,即可依据绑定的target是否变动来决定源码生成命令是否执行(绑定方法:add_custom_command
的OUTPUT为MESSAGE_SRC和MESSAGE_HDRS,同时MESSAGE_SRC和MESSAGE_HDRS为add_custom_target的DENPENDS)
其中值得主要的有两点:
 ● 设置生成的源码文件属性GENERATED为TRUE,否则cmake时会因找不到源码而报错
 • 使用add_custom_target添加目标时要设置ALL关键字,否则target将不在默认编译列表中
这样就能实现proto生成源码配置定制化并避免不必要的重新编译了
```

protobufv3.2.0rc2 win linux均可编译 03-10 linux编译:cd cmake 然后 cmake . 然后 make 然后可选 make install windows编译:启动cmake界面,源代码路径指向cmake文件夹,生成路径可新建一个文件建比如...

三、cmake下使用protobuf 3.1 安装cmake sudoapt-getinstallcmake cmake-qt-qui 1 3.2 编写CMakeLists文件 mkdir-p CSDN\_ws/protobuf/cmake/cpCSDN\_ws/protob...

1.先下载Cmakehttps://cmake.org/download/根据版本下载对应的安装程序即可 2.VS 2017安装https://www.microsoft.com/zh-cn/ 3.最新版的protobuf下载地址:https://...

Protobuf是google开发的一个序列化和反序列化的协议库,我们可以自己设计传递数据的格式,通过.proto文件定义我们的要传递的数据格式。例如,在深度学习中常...

07-12

1-16

1-15

gcs\_20210916的博客 **①** 5863

fengfengdiandia的专栏 0 4075

《好好先生》专栏 ① 1258

举报

专栏目录

如果只有少量proto文件且在同一文件夹下,可使用方法一的protobuf\_generate\_cpp生成源码,若proto文件较多且层次复杂,建议使用方法三.

CMake Protobuf 示例 使用 CMake 通过 protobuf 管理 C++ 项目的最小示例。 建造 \$ mkdir build \$ cd build \$ cmake .. \$ make 跑步 \$ ./main

具体使用可以参考该项目https://github.com/mingjitianming/transmit\_asio中CMakeLists的使用方式

cmake-protobuf-example:使用 CMake 通过 protobuf 管理 C++ 项目的最小示例

ubuntu下protobuf使用案例[命令行->cmake->VS code->bazel]\_windSeS的...

使用VS2017 cmake编译protobuf3.10.0\_popcorn、的博客\_vs...

Protobuf在Cmake中的正确使用

【CMake/Protobuf】 CMake 下使用 protobuf

cmake protobuf

CMake编译protobuf

protobuf

```
c++使用protobuf文件生成pb.h和pb.cc 热门推荐
简介 protobuf可以用于描述数据结构,可以用于序列化和反序列化,并且适用于多平台的扩展,是比较好的数据描述平台。 操作 CMakeLists.txt cmake_minimum_req...
Cmake编译protobuf_weixin_30466953的博客
                                                                                                          1-11
Cmake编译protobuf 编译指令,在powershell中执行:.\protoc.exe.\ive.proto --cpp_out..\protoc.exe.\ive.proto --csharp_out.注意:运行库为 多线程DLL(/MD),protob...
cmake 生成 protobuf_学之之博未若知之之要知之之要未若行之之实的博 ...
cmake 生成 protobuf 平台:ubuntu 14.04 X64 简介 protobuf可以用于描述据结构,可以用于序列化和反序列化,并且适用于多平台的扩展,是比较好的数据描述平台。 操...
C++使用protobuf 作为网络消息协议
                                                                                                        03-25
```

```
一个c++使用protobuf作为消息协议的一个小demo,从这个demo里你可以很好地理解进行socket编程中的数据包的设计以及数据的打包和解包。
CMakeList中自动编译protobuf文件
                                                                                                     HuaShao 0 1285
可以使用下面的命令: protobuf_generate_cpp protobuf_generate_python 例如: find_package(Protobuf REQUIRED) include_directories(${Protobuf_INCLUDE_DI...
CMake动态编译protobuff_林临工作室的博客
                                                                                                               1-17
CMake对protobuff支持不是很好,想要动态编译.proto文件,不是那么容易的。 本文适用CMake3.14.6 protobuf3.9.1centos7.4 网上搜索,CMake提供如下命令支持PB 1 fin...
protobuf C++ 使用Cmake编译protobuf-3.19.1 专题 Java C#示例 VS201...
一、使用CMake编译 官方教程:protobuf/README.md at master · protocolbuffers/protobuf · GitHub 1.下载源码protobuf-cpp-3.19.1.zip 2.编译 //开始之前要下载proto...
Protobuf基本用法
                                                                                                 春江水暖鸭先知 ① 7981
本文主要讲解protobuf的基本用法(caffe使用protobuf所用到的语法) protobuf简介 protobuf功能是把某种数据结构的信息以某种格式保存起来。它主要用于文件存储以...
cmake 中使用 protobuf 生成文件
                                                                                               protobuf 提供了自动<mark>生成</mark>文件的功能,我直接看的cmake文件,里面没说函数说明,我没有全部弄明白,但可以正常<mark>使用</mark>: add_library(myTarget ${src}) protobuf_ge...
```

```
偷得浮生半日闲 ② 325
ProtoBuf
ProtoBuf简述 protocol buffers 是一种语言无关、平台无关、可扩展的序列化结构数据的方法,可用于通信协议、数据存储等。 使用流程: 创建 .proto 文件,定义数据...
带cmake支持的protobuf2.6.1源码
                                                                                                    11-19
protobuf2.6.1源码,加入了cmake支持,Linux下和Windows下都能构建,生成Debug/Release版本静态库,Debug版本带d作为后缀。
Protocol-Buffer-Examples:带有CMake的Google协议缓冲区3.0初学者的C++示例
                                                                                                    05-25
```

两个或多个proto文件不在一个目录,如果直接<mark>使用protobuf\_</mark>generate\_cpp来<mark>生成</mark>,直接会报错。 若将多个proto文件放到同一个目录...然后import proto文件名即可,...

C++Protobuf的生成与使用 编写.proto文件 syntax:表明protobuf的版本号(末尾不要忘了;) package:表明<mark>生成</mark>的类对象位于哪一个命名空间(末尾不要忘了;) 每一个类属性后面的必须跟一个编号,... cmakeList编译protobuf.pb.h时报错fatal error: google/protobuf/port\_def.inc: No such file or directory usstmiracle的博客 @ 8320 写一个protobuf文件,编译后会生成两个文件一个.cc和一个.h文件。 export LD\_LIBRARY\_PATH=/usr/local/lib protoc --cpp\_out=./ forward\_protobuf.proto 单独的编... \$firecat利白的代码足迹\$ **②** 2211 Could NOT find Protobuf (missing: Protobuf\_PROTOC\_EXECUTABLE)

出现如题所示的问题,怎么办?解决办法:添加这个变量,指向 D:\vcpkg\packages\protobuf\_x86-windows-static\tools\protobuf\protoc.exe所在的路径

协议缓冲区示例 雷茂 介绍 协议缓冲区,也称为Protobuf,是Google的语言无关,平台无关的可扩展机制,用于序列化结构化数据。 协议缓冲区版本3(Proto3)于201...

一. 简介 **Pro**tocol Buffers 是一种数据序列化的格式。特点是语言无关,平台无关。 相比于 XML 的优势是更小,更快和更简单。适合数据存储或 RPC 数据交换,可用… protobuf生成的cpp库相关的API接口 protobuf语法: syntax = "proto2" 定义一个测试文件: test.proto 内容如下 syntax = "proto2"; message People { required int32 id = 1; optional string name = 2; optio...

CMake编译protobuf 1.从网址https://github.com/protocolbuffers/protobuf/tree/v3.15.6下载protobuf; 下载后得到压缩文件: protobuf-all-3.15.6.zip; 2. 将protobuf-... protobuf的c++生成简明过程示例 zmgarenas的专栏 @ 921 1、 定义.proto文件: 消息体中: required 必须的 optional 可选择的 repeated 不定项的(0个或多个) 枚举:略 2、编译: 生成cpp文件:执行(proto... 一文读懂——Protobuf 最新发布 

© 2022 CSDN 皮肤主题: 大白 设计师: CSDN官方博客 返回首页

关于我们 招贤纳士 商务合作 寻求报道 🕿 400-660-0108 ☑ kefu@csdn.net 💟 在线客服 工作时间 8:30-22:00 北京互联网违 明霁天明(