chapter3.2 cmake使用Protobuf的代码生成

本示例展示如何使用 protobuf 生成源码文件。

Protocol Buffers 是一种由谷歌提出的数据序列化的格式。用户可以提供一个描述了数据的 `.proto` 格式的文件，通过protobuf编译器，这一文件可以被编译成包括C++在内的一系列编程语言的源码文件。你可以理解为泛式编程！

1. 文件结构：

```unix
.
├── AddressBook.proto
├── CMakeLists.txt
├── README.adoc
├── build
└── main.cpp

2 directories, 4 files
```

- AddressBook.proto - 使用protocol buffer定义的proto文件 example
  - AddressBook.proto - proto file from main protocol buffer example
- CMakeLists.txt - 描述你希望能够运行的CMake命令
  - CMakeLists.txt - Contains the CMake commands you wish to run
- main.cpp - 包含主函数的源文件
  - main.cpp - The source file from the protobuf example.

2. 安装前置：

1）进入你的linux虚拟机
2）要求预安装protocol buffers的二进制文件和库文件，在Ubuntu系统中，可以使用下述命令安装：

```
sudo apt-get install protobuf-compiler libprotobuf-dev
```

```
huluobo@huluobodeMacBook-Pro ~/cmake-examples/myCmake/chapter3.2  main ± orb
huluobo@ubuntu:/Users/huluobo/cmake-examples/myCmake/chapter3.2$ sudo apt-get install protobuf-compiler libprotobuf-dev
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
libprotobuf-dev is already the newest version (3.21.12-1ubuntu7).
protobuf-compiler is already the newest version (3.21.12-1ubuntu7).
The following packages were automatically installed and are no longer required:
  libperl5.34 libtiff5 perl-modules-5.34
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 284 not upgraded.
huluobo@ubuntu:/Users/huluobo/cmake-examples/myCmake/chapter3.2$
```

3. 文件详解：

3.1 导出变量

在本示例中，通过CMake protobuf包到处并使用的变量如下：

- `PROTOBUF_FOUND` - 标志Protocol Buffers是否安装
  - `PROTOBUF_FOUND` - If Protocol Buffers is installed
- `PROTOBUF_INCLUDE_DIRS` - 描述protobuf头文件路径
  - `PROTOBUF_INCLUDE_DIRS` - The protobuf header files
- `PROTOBUF_LIBRARIES` - 描述protobuf库文件路径
  - `PROTOBUF_LIBRARIES` - The protobuf library

关于Protobuf变量更详细的描述可以查阅 `FindProtobuf.cmake` 文件获取

3.2 源文件生成

CMake protobuf包中包含了一些列的帮助函数来简化源代码生成流程，在本示例中，我们是用下面的语句来生成C++源码：

```
PROTOBUF_GENERATE_CPP(PROTO_SRCS PROTO_HDRS AddressBook.proto)
```

参数含义：

- PROTO_SRCS - 将被存储在.pb.cc文件里的变量名
  - PROTO_SRCS - Name of the variable that will store the .pb.cc files.
- PROTO_HDRS- 将被存储在.pb.h文件里的变量名
  - PROTO_HDRS- Name of the variable that will store the .pb.h files.
- AddressBook.proto - 用于编译的原始.proto文件
  - AddressBook.proto - The .proto file to generate code from.

3.3 文件生成

在调用过 `PROTOBUF_GENERATE_CPP` 函数之后，上述变量便可以被使用了，他们将被视为protobuf编译器执行特定命令生成的输出结果。

为了进一步完成文件的生成，你需要将他们添加进库函数或者可执行文件中，例如：

```
add_executable(protobuf_example
    main.cpp
    ${PROTO_SRCS}
    ${PROTO_HDRS})
```

这使得 `make` 这一命令被调用时，protobuf编译器也将随之被调用。

当.proto文件被改变时，与其相关联的源代码文件也将被自动重新生成；不过，如果.proto文件没有发生修改，重新执行 `make` 命令，并不会发生任何变化。

3.4 过程解读：

```
cmake_minimum_required(VERSION 3.5)

# （1）设定项目名
project (protobuf_example)

# （2）查包
find_package(Protobuf REQUIRED)

# （3）检查是否搜索成功
if(PROTOBUF_FOUND)
    message ("protobuf found")
else()
    message (FATAL_ERROR "Cannot find Protobuf")
endif()

# （4）生成 .h and .cxx files
PROTOBUF_GENERATE_CPP(PROTO_SRCS PROTO_HDRS AddressBook.proto)

# （5）打印生成文件的路径（详解看：上述3.1和3.2）
message ("PROTO_SRCS = ${PROTO_SRCS}")
message ("PROTO_HDRS = ${PROTO_HDRS}")

# （6）利用main.cpp制作可执行文件，called：protobuf_example
add_executable(protobuf_example
    main.cpp
    ${PROTO_SRCS}
```

```
    ${PROTO_HDRS})

# （7）为 可执行文件protobuf_example 链接子目录
target_include_directories(protobuf_example
    PUBLIC
    ${PROTOBUF_INCLUDE_DIRS}
    ${CMAKE_CURRENT_BINARY_DIR}
)

# （8）为 可执行文件protobuf_example 链接 库函数${PROTOBUF_LIBRARIES}
target_link_libraries(protobuf_example
    PUBLIC
    ${PROTOBUF_LIBRARIES}
)
```

4. 总览：

```
huluobo@ubuntu:/Users/huluobo/cmake-examples/myCmake/chapter3.2/build$ cmake ..
-- The C compiler identification is GNU 12.3.0
-- The CXX compiler identification is GNU 12.3.0
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Check for working C compiler: /usr/bin/cc - skipped
-- Detecting C compile features
-- Detecting C compile features - done
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Check for working CXX compiler: /usr/bin/c++ - skipped
-- Detecting CXX compile features
-- Detecting CXX compile features - done
-- Found Protobuf: /usr/lib/aarch64-linux-gnu/libprotobuf.so (found version "3.21.12")
protobuf found
PROTO_SRCS = /Users/huluobo/cmake-examples/myCmake/chapter3.2/build/AddressBook.pb.cc
PROTO_HDRS = /Users/huluobo/cmake-examples/myCmake/chapter3.2/build/AddressBook.pb.h
-- Configuring done
-- Generating done
-- Build files have been written to: /Users/huluobo/cmake-examples/myCmake/chapter3.2/build
huluobo@ubuntu:/Users/huluobo/cmake-examples/myCmake/chapter3.2/build$ ls
CMakeCache.txt  CMakeFiles  Makefile  cmake_install.cmake
huluobo@ubuntu:/Users/huluobo/cmake-examples/myCmake/chapter3.2/build$ make VERBOSE=1
/usr/bin/cmake -S/Users/huluobo/cmake-examples/myCmake/chapter3.2 -B/Users/huluobo/cmake-
examples/myCmake/chapter3.2/build --check-build-system CMakeFiles/Makefile.cmake 0
/usr/bin/cmake -E cmake_progress_start /Users/huluobo/cmake-examples/myCmake/chapter3.2/build/CMakeFiles
/Users/huluobo/cmake-examples/myCmake/chapter3.2/build//CMakeFiles/progress.marks
make  -f CMakeFiles/Makefile2 all
make[1]: Entering directory '/Users/huluobo/cmake-examples/myCmake/chapter3.2/build'
make  -f CMakeFiles/protobuf_example.dir/build.make CMakeFiles/protobuf_example.dir/depend
make[2]: Entering directory '/Users/huluobo/cmake-examples/myCmake/chapter3.2/build'
[ 25%] Running cpp protocol buffer compiler on AddressBook.proto
/usr/bin/protoc --cpp_out /Users/huluobo/cmake-examples/myCmake/chapter3.2/build -I /Users/huluobo/cmake-
examples/myCmake/chapter3.2 /Users/huluobo/cmake-examples/myCmake/chapter3.2/AddressBook.proto
[libprotobuf WARNING google/protobuf/compiler/parser.cc:646] No syntax specified for the proto file:
AddressBook.proto. Please use 'syntax = "proto2";' or 'syntax = "proto3";' to specify a syntax version.
(Defaulted to proto2 syntax.)
cd /Users/huluobo/cmake-examples/myCmake/chapter3.2/build && /usr/bin/cmake -E cmake_depends "Unix
Makefiles" /Users/huluobo/cmake-examples/myCmake/chapter3.2 /Users/huluobo/cmake-
examples/myCmake/chapter3.2 /Users/huluobo/cmake-examples/myCmake/chapter3.2/build /Users/huluobo/cmake-
examples/myCmake/chapter3.2/build /Users/huluobo/cmake-
examples/myCmake/chapter3.2/build/CMakeFiles/protobuf_example.dir/DependInfo.cmake --color=
make[2]: Leaving directory '/Users/huluobo/cmake-examples/myCmake/chapter3.2/build'
make  -f CMakeFiles/protobuf_example.dir/build.make CMakeFiles/protobuf_example.dir/build
make[2]: Entering directory '/Users/huluobo/cmake-examples/myCmake/chapter3.2/build'
[ 50%] Building CXX object CMakeFiles/protobuf_example.dir/main.cpp.o
```

```
/usr/bin/c++  -I/Users/huluobo/cmake-examples/myCmake/chapter3.2/build  -MD -MT
CMakeFiles/protobuf_example.dir/main.cpp.o -MF CMakeFiles/protobuf_example.dir/main.cpp.o.d -o
CMakeFiles/protobuf_example.dir/main.cpp.o -c /Users/huluobo/cmake-examples/myCmake/chapter3.2/main.cpp
[ 75%] Building CXX object CMakeFiles/protobuf_example.dir/AddressBook.pb.cc.o
/usr/bin/c++  -I/Users/huluobo/cmake-examples/myCmake/chapter3.2/build  -MD -MT
CMakeFiles/protobuf_example.dir/AddressBook.pb.cc.o -MF
CMakeFiles/protobuf_example.dir/AddressBook.pb.cc.o.d -o
CMakeFiles/protobuf_example.dir/AddressBook.pb.cc.o -c /Users/huluobo/cmake-
examples/myCmake/chapter3.2/build/AddressBook.pb.cc
[100%] Linking CXX executable protobuf_example
/usr/bin/cmake -E cmake_link_script CMakeFiles/protobuf_example.dir/link.txt --verbose=1
/usr/bin/c++ CMakeFiles/protobuf_example.dir/main.cpp.o CMakeFiles/protobuf_example.dir/AddressBook.pb.cc.o
-o protobuf_example  /usr/lib/aarch64-linux-gnu/libprotobuf.so
make[2]: Leaving directory '/Users/huluobo/cmake-examples/myCmake/chapter3.2/build'
[100%] Built target protobuf_example
make[1]: Leaving directory '/Users/huluobo/cmake-examples/myCmake/chapter3.2/build'
/usr/bin/cmake -E cmake_progress_start /Users/huluobo/cmake-examples/myCmake/chapter3.2/build/CMakeFiles 0
huluobo@ubuntu:/Users/huluobo/cmake-examples/myCmake/chapter3.2/build$ ls
AddressBook.pb.cc  AddressBook.pb.h  CMakeCache.txt  CMakeFiles  Makefile  cmake_install.cmake
protobuf_example
huluobo@ubuntu:/Users/huluobo/cmake-examples/myCmake/chapter3.2/build$ ./protobuf_example test.db
test.db: File not found.  Creating a new file.
Enter person ID number: 11
Enter name: Boxuan Hu
Enter email address (blank for none): 3157726199@qq.com
Enter a phone number (or leave blank to finish):
huluobo@ubuntu:/Users/huluobo/cmake-examples/myCmake/chapter3.2/build$ ls
AddressBook.pb.cc  AddressBook.pb.h  CMakeCache.txt  CMakeFiles  Makefile  cmake_install.cmake
protobuf_example  test.db
huluobo@ubuntu:/Users/huluobo/cmake-examples/myCmake/chapter3.2/build$ cat test.db
        Boxuan Hu
                        3157726199@qq.com
```

此时整体结构：

```
.
├── AddressBook.pb.cc
├── AddressBook.pb.h
├── CMakeCache.txt
├── CMakeFiles
│   ├── 3.25.1
│   │   ├── CMakeCCompiler.cmake
│   │   ├── CMakeCXXCompiler.cmake
│   │   ├── CMakeDetermineCompilerABI_C.bin
│   │   ├── CMakeDetermineCompilerABI_CXX.bin
│   │   ├── CMakeSystem.cmake
│   │   ├── CompilerIdC
│   │   │   ├── CMakeCCompilerId.c
│   │   │   ├── a.out
│   │   │   └── tmp
│   │   └── CompilerIdCXX
│   │       ├── CMakeCXXCompilerId.cpp
│   │       ├── a.out
│   │       └── tmp
│   ├── CMakeDirectoryInformation.cmake
│   ├── CMakeOutput.log
│   ├── CMakeRuleHashes.txt
│   ├── CMakeScratch
│   ├── Makefile.cmake
│   ├── Makefile2
│   ├── TargetDirectories.txt
│   ├── cmake.check_cache
│   ├── pkgRedirects
```

```
│   ├── progress.marks
│   └── protobuf_example.dir
│       ├── AddressBook.pb.cc.o
│       ├── AddressBook.pb.cc.o.d
│       ├── DependInfo.cmake
│       ├── build.make
│       ├── cmake_clean.cmake
│       ├── compiler_depend.make
│       ├── compiler_depend.ts
│       ├── depend.make
│       ├── flags.make
│       ├── link.txt
│       ├── main.cpp.o
│       ├── main.cpp.o.d
│       └── progress.make
├── Makefile
├── cmake_install.cmake
├── protobuf_example
└── test.db

10 directories, 37 files
```