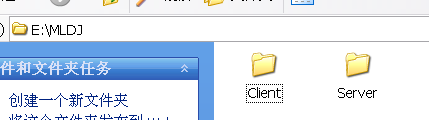
消息包生成工具使用说明

# 使用说明

1. **编写PBMessage.proto放到指定目录下（目前是CreatePacket根目录），需要注意的PBMessage.proto文件格式是ANSI格式，最好不要用记事本编辑，保存为其他格式会报错，生成不了C++下面的PBMessage.pb.cc和PBMessage.pb.h文件，用UE比较好。**
2. Winodws7用户运行ZCreatePacket.bat，Windows8用户运行ZCreatePacket\_WP8.bat
3. 生成的文件为在 \tmp 目录下
4. ZCreatePacket.bat中的DstTableFolder和SrcTableFolder路径的说明：

基本的目录结构是：



其中，

Client是：

svn://10.6.34.93/MOBILERPG\_RES/Branches/Version/Main

Server是：

svn://10.6.35.101/MOBILERPG/MLDJ/Branches/Version/Main

基于上面的目录结构设置了：从客户端表格生成表格代码并拷贝到客户端的路径，以及从公共表格或者服务器表格生成表格代码并拷贝到服务器的路径。

1. 目前默认生成的C++消息包和处理函数，全部自动拷贝到相关目录，C#的只拷贝消息包，处理函数不拷贝，需要自己写。
2. 所有新加消息包文件，都必须手动添加到SVN中，不要忘记Add了。

# 注意点

1. PBMessage.proto的编写需要符合规范：

目前规范为：

//#Client Connect to GameServer

//@CG

message CG\_LOGIN\_001

{

optional string AccountName = 1;//账户名

optional string password = 2;//密码

optional string mechineid = 3;

optional string version = 4;//版本

optional int64 userguid =5;//重新连接验证用户guid

}

其中前2行不能省略；

//#Client Connect to GameServer 该行为消息包注释

//@CG 该行为标示信息从C到G

开始标示不要省略，如下：

//#

//@

1. PBMessage.proto的message部分的规范参考Google官方规范。
2. PBMessage.proto中的Enum规范如下：

例子：

//#test message

//@CG

message CG\_Test

{

required string name = 1; //名字

enum TYPE

{

TYPE\_INVALID = 0;

TYPE\_1 = 1;

TYPE\_2 = 2;

TYPE\_3 = 3;

}

required int32 type = 2; //TYPE类型的数据

required string testData = 3; //数据

}

官方规范为：

enum 中不能出现负数，结尾符号必须是;

MLDJ规范：

对enum的调用不能直接写enum类型，

而应该写required int32 type = 2; 这样的代码；

即不能写为：required TYPE type = 2;

另外也不支持官方enum的默认类型，即不支持下面写法：

required TYPE type = 2[default = TYPE\_INVALID];

1. **MLDJ支持的数值类型如下：**

下面是MLD中Protobuf支持的一套基本数据类型，以及映射到C++\C#等语言的基础数据类型：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| protobuf 数据类型 | 描述 | 打包 | C++语言映射 | C# |
| int32 | 32位整数、 | N | int32 | int32 |
| uin32 | 无符号32位整数 | N | uin32 | uin32 |
| int64 | 64位整数 | N | int64 | int64 |
| uint64 | 64为无符号整 | N | uint64 | uint64 |
| string | 只能处理 ASCII字符 | N | std::string | string |
| bytes | 用于处理多字节的语言字符、如中文 | N | std::string | bytes |
| enum | 可以包含一个用户自定义的枚举类型uint32 | N(uint32) | enum | enum |

N 表示打包的字节并不是固定的，而是根据数据的大小或者长度。例如int32，如果数值比较小，在0~127时，使用一个字节打包。

注意：

1. 不支持float和double数据类型，如果有需要使用float和double的，请使用int32或者int64模拟，比如一端发包时候，把float数值乘以10或者更多，转成int32等发送，另一端收包时候再除以10或者更多，注意数值范围。
2. 不支持sint32，sint64，fixed32，fixed64，sfixed32，sfixed64等。
3. To be continue …