1 使用前先在 handler\_config.properties 里配置好对应的路径。

2 .proto 文件的命名规范为 模块名\_其他（功能描述等,可省略）.proto。首字母大写。

如：Friend.proto。

3 .proto编写规范。

首行必须是@module开头，如下：

*//@module{"name":"common","desc":"公共模块","codeRange":"[1,500]"}*

module里的内容为json格式，解释如下：

name：模块名

desc：本文件功能模块说明

codeRange：本模块使用的消息号的范围。格式为[from,to]。

第二行为java\_package开头，java\_package路径最后字段应该与模块名一致。

如：**option** java\_package = "com.troy.protobuf.friend";

第三行为java\_outer\_classname开头，格式为 .proto的文件名+ proto。

如：**option** java\_outer\_classname = **"**Friend**Proto"**;

.proto 里的**message** 的命名格式统一如下：

请求消息：PB+功能名+Req 例如 PBFriendApplyAgreeReq 同意好友申请  
 响应消息：PB+功能名+Resp 例如 PBFriendListResp 好友列表信息  
 复用Bean: PB+功能名称+Bean 例如 PBFriendBean 好友信息

*//@msg* 是用标记有什么协议，其中一些会生成Handler类，格式为json 格式，不换行。例子如下：

*//@msg{"code":1, "name":"Ping", "desc":"心跳客户端请求", "msgBody":"PBPingReq", "template":"GameHandler"}***message** PBPingReq {  
 **required int64** clientTick = 1; *//客户端时间* **required int32** delay = 2; *//延迟*}

*//@msg{"code":608, "name":"GameStarted", "desc":"游戏正式开始"}*

没有*msgBody，*可以没有对应的**message**。

有5个参数，

code ：消息号，必填。注意：消息号使用规范   
 1.消息号长度为short，大小范围 0，32767 的十进制。  
 2.每个模块使用一个区间段的协议号 例如好友模块指令号范围[501,600]。  
 3.请求跟响应指令号不复用,比如，请求使用奇数501，响应使用偶数 502 。  
 4.不使用二级指令，每个指令号需定义为 唯一的。

name ：生成的handler类的名字格式是 name+Handler。MsgCode常量名字格式为name+Req/Resp,尾缀取决于code的奇偶，通常为对应**message**的名称的功能名。没有另起。必填。

desc ：handler功能描述，必须填。

template ：生成的handler继承的父类。 code为偶数的可不填。有本参数才会生成handler类。

msgBody：消息体，表示该协议通讯时传输的内容。若填写，必须有一个对应的**message**。

对于**optional、 required、 repeated 开头的行要加注释，不然程序会报错。如**

**required int32** friendId = 1; *// 好友id*

注意：注释要在同一行才有效。且用 “//” 注释的

**运行后生成的文件有：MsgCode.java，ProtoConfig.xml 和各种handler.java文件**

完整例子如下：

Game.proto文件

*//@module{"name":"game","desc":"游戏战斗模块" ,"codeRange":"[601,900]"}***option** java\_package = **"com.game.protobuf.game"**;  
**option** java\_outer\_classname = **"GameProto"**;  
  
*//目标信息***message** PBTargetBean {  
 **required int32** targetID = 1; *//目标ID* **required int32** targetX = 2; *//目标x* **required int32** targetY = 3; *//目标Y* **required int32** targetZ = 4; *//目标Y* **required bool** isMiss = 5; *//是否miss*}  
  
*//目标伤害包***message** PBTargetHpBean {  
 **required int32** targetID = 1; *//技能目标ID* **required bool** isMiss = 2; *//是否丢失* **required bool** isCrit = 3; *//暴击* **required int32** type = 4; *//1物理伤害,2治疗* **required int32** curValue = 5; *//具体值* **required int32** curHp = 6; *//当前血量*}  
  
*//@msg{"code":601, "name":"StartGame", "desc":"请求加入游戏战斗", "msgBody":"PBStartGameReq", "template":"PlayerHandler"}***message** PBStartGameReq {  
 **required int32** heroId = 1; *//英雄id* **required int32** mapId = 2; *//地图id*}  
  
*//@msg{"code":602, "name":"StartGame", "desc":"准备撮合", "msgBody":"PBReadyMatchResp"}***message** PBReadyMatchResp{  
 **required int32** matchTime = 1; *//匹配一轮的时长（单位毫秒）*}  
  
*//游戏开始倒计时  
//@msg{"code":606, "name":"StartLeaveTime", "desc":"游戏开始倒计时","msgBody":"PBStartLeaveTimeResp"}***message** PBStartLeaveTimeResp{  
 **required int32** leaveTime =1;*//倒计时（单位毫秒）*}  
  
*//@msg{"code":608, "name":"GameStarted", "desc":"游戏正式开始"}  
  
//@msg{"code":618, "name":"SkillLockTarget", "desc":"锁定目标", "msgBody":"PBSkillLockTargetResp"}***message** PBSkillLockTargetResp {  
 **required bool** isLeft = 4; *//朝向 true为左，false为右* **required int32** skillID = 1; *//技能ID* **required int32** releaserID = 2; *//释放者ID* **repeated** PBTargetBean targets = 3; *//目标列表* **optional int32** angle = 5;*//释放角度*}  
  
*//@msg{"code":620, "name":"SkillRelease", "desc":"技能释放", "msgBody":"PBSkillReleaseResp"}***message** PBSkillReleaseResp {  
 **required int32** skillID = 1; *//技能ID* **required int32** releaserID = 2; *//释放者ID* **repeated** PBTargetHpBean targets = 3; *//伤害列表* **required int32** hurtValue = 4; *//技能释放者扣血（自残）* **optional int32** masterId = 5;*//所属英雄主人id* **optional int32** angle = 6;*//释放角度*}