# Actor代码生成相关内容

## 简介

描述文件使用xml格式

xml文件分为两种：

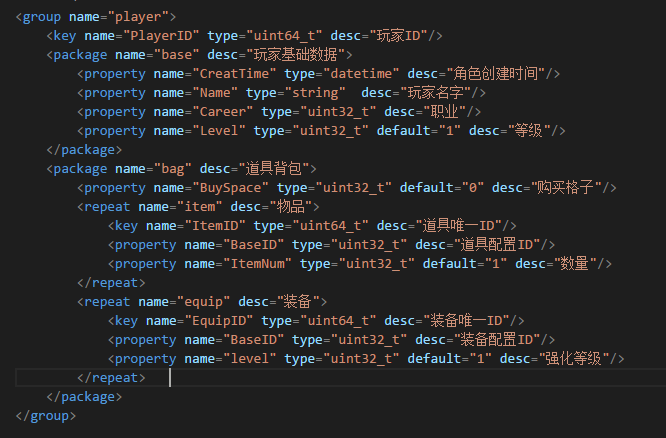
描述数据字段、类型和层次结构的Data描述文件

描述Actor管理的数据的Actor描述文件

xml文件约定格式：

节点名称以及属性名称都要用小写。数据类型也要用小写，因为在存储上没有什么差异，以及管理的便捷性，整数类型只支持64位和32的有符号和无符号类型，所有索引（玩家ID等）都使用64位无符号整数表示。desc是对当前节点的描述，方便阅读，后面有时间可能会把它作为代码注释。

## Data描述文件：



根节点为group 属性只有一个name

group有一个key节点（现在的key节点没有使用，只是为了方便阅读，不过解析的时候用到，所以还是要老老实实写上去）若干以key为第一索引的package。如果没有key节点，代表这个group是个全局唯一的。

key 有name和type、desc的属性

package 有name、desc 属性

若干property子节点、若干以group的key作为第一索引的repeat。

property 有name、type、desc属性

repeat有name、desc属性

一个key节点、若干property节点

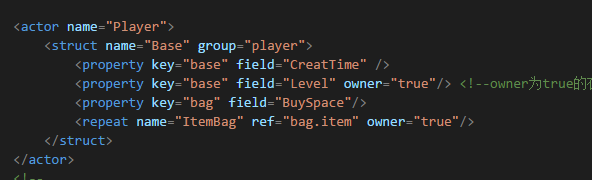
所有的数据都是利用redis的hash存储的

对于一个没有repeat的package，它在redis的存储是一个hash，在redis的hash key是groupName.packageName:groupKey的形式。Package内的所有property都是这个hash的field。

package中的repeat在redis中的存储也是一个单独的hash，有点类似mysql里面使用组合主键的表，repeat本质上和package没啥区别，也是一个hash，只不过在redis里面的命名使用 groupName.packageName.repeatName:groupKey:repeatKey的格式。repeat

在逻辑上属于package，在存储上没啥关系。

## Actor描述文件：





Actor就是每个Actor在逻辑上要管理或者查询的数据，每个Actor最后会生成一个结构体，里面的所有元素都在Data描述文件里面定义，这里只是引用那里的对象

actor 有name属性

若干struct节点和map节点

struct 有name 、group属性，group指的是在Data描述文件里面定义的group

若干property节点和若干repeat节点，里面property和repeat都是引用的group里面定义的元素，struct会生成一个class，并作为一个对象存在于actor的结构体中。

map 有name属性和group属性，map内部结构和struct其实完全一样，只不过在actor中是用map的形式保存。

property 有key、field 、owner属性，key指的是Data描述文件中的package的名字，field指的是package中的property。Owner是可选的，如果owner是true，代表这里能够修改这条属性的值，一条属性只有一个地方可以是owner，非owner只有在owner修改值的时候接收订阅然后修改。

repeat 有name、ref、owner属性，name的定义和代码内的生成有关，只要在struct或者map中不重复就行了。ref是用点分割的第一个指的是group中的package的name第二个是package中repeat的名字。Owner是可选项，如果为true，将忽略内部的所有property项，使用data描述文件里面的所有项，并对所有项有修改权限。同一个repeat也是只有一个actor可以修改，别的都是只读。

为了降低不必要的复杂度，repeat里面不能嵌套repeat。Map和struct本质一样。

同一层级的节点不能有重名，无论actor还是data的描述文件。

## 关于生成工具的使用：

启动参数需要两个 --xml= 和--cpp= ，xml对应的是data描述文件的位置，里面可以嵌套文件夹，工具会扫描这个文件夹以及子文件夹内所有的xml文件，另外xml的文件后缀名请使用小写。--cpp是生成的C++头文件存放的位置，文件名和xml的文件名一样，目录结构和xml的也一样。因为windows权限的问题，第一次生成的时候，可能子文件夹不存在，需要手动创建或者使用管理员权限运行。

关于生成工具的build相关内容：

工程文件使用premake创建的，可以修改premake来修改工程相关属性

运行make.bat文件会直接生成工程文件和目录，clear会删除生成的所有目录，解决方案生成在Build/vs2015 里面。