# 说明文档

## 概述

这是一个劣质单机的开源游戏，主要是给刚入门的游戏开发者一些基本的参考。游戏的制作分为lua部分和excel配置表部分。对普通人来讲编辑配置表就可以实现大部分内容，但是天赋，技能高阶特效，特技的特定效果，更多高度自定义的内容，必须通过编辑lua实现。

游戏是金庸群侠传X的重制版，我保留了金Xmod制作的最主要的习惯，也就是配置表编辑，但是我对配置表进行了大规模的修改，配置表语法也进行了大规模的更新，对金x原版的配置表语法也只保留了最主要的部分。所以之前有过金Xmod编辑体验的人必须重新学习配置表结构和配置表语法。

游戏引擎是cocos2dx3.17.2，一个流行的开源免费的引擎，但是引擎代码太多，而我实际上创建的lua项目没有多少对C++部分的改动，所以游戏只开源lua部分。游戏99.99%的代码都是lua写的，所以这相当于完全开源。

为了使开发更容易，我重新设计了excel到lua的转化工具，这部分是完全python实现；加密方案是luajit字节码，为了更容易的加密，我也写了一个简单的python脚本来批量执行这部分内容。和金X有所不同的是，python使用python3.7.3作为开发版本。

## Lua部分

尽管游戏使用了简单易学的脚本语言lua语言作为主要开发语言，但是这并不代表着游戏代码结构是混乱的，也不代表二次开发的难度非常低。因为为了开发这种规模的游戏，代码必须拥有足够的抽象层次，面向对象是一个比较容易接受的方式，lua没有原生的class 关键词，所以lua的面向对象实现一直是各凭本事，cocos2dx-lua提供了一种非常简单的class实现，虽然这种实现对初学者来讲依然非常难以理解。但是这种简单的class实现并不能满足我对项目的要求，在游戏的第一轮开发时我完整的采用了cocos-lua 的class方案，开始重构的时候我弃用这种方式，开始独立重新设计class体系。整个class我重新设计了10个版本，每一个版本的实现，都意味着对原始工程代码的一定规模的修改。规模最大的修改是第一个版本和当前版本，对所有代码都进行了大规模修改。要理解我写的核心代码，就必须学习我写的开源lua类库 luaclass：<https://github.com/CppCXY/luaClass>当然如果对lua元编程感兴趣，可以考虑看我的另一个开源项目<https://github.com/CppCXY/luaSyntax>

游戏分为核心代码，资源脚本以及mod代码，核心代码不表，资源代码主要是excel配置表转化而来，mod代码是在src/modInterfaceLua目录下的代码，为了提供良好的示范性，我将许多内容，也写在了modInterfaceLua目录的文件中，所以这部分可以根据注释和名称自己学习。

## 配置表部分

和金X相比，配置表有过大规模的修改，许多field被弃用，新增了许多配置表，配置表语法高度自定义，并且可以实现内联lua代码。下面逐一介绍配置表和配置表语法。

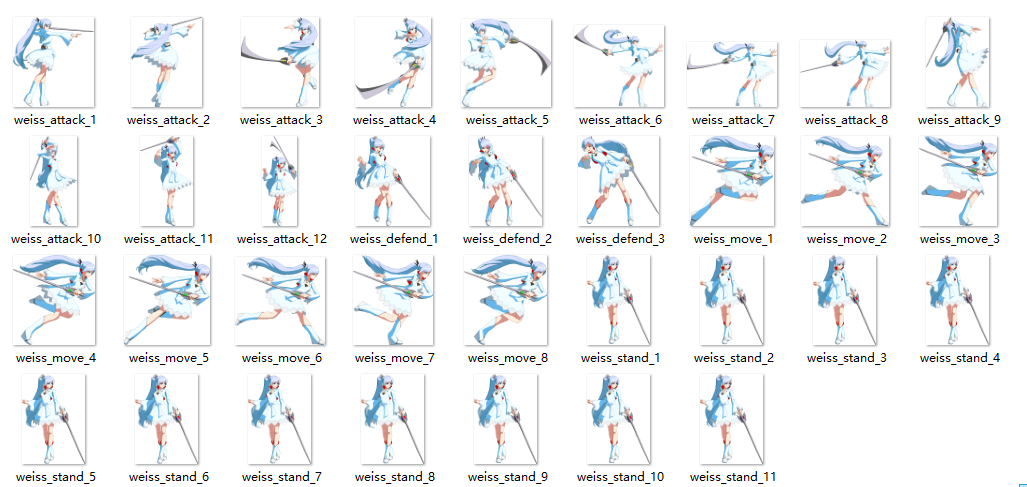
### A模型设置

金X提供了基本的模型使用方式。但是，是通过手动编辑xml实现的，xml编辑对普通人来说并不便利，所以我把模型的编辑交给了配置表。

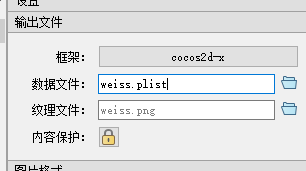
1. 与金X不同的是，对模型的要求不再是一张张的图片。模型必须是规范的，只接受合并导出的plist文件和同名的png文件。
2. 那么如何制作这两个文件呢？首先去下载texturePacker，这个软件是付费的，但是有免费模式，并且免费模式功能足够制作模型文件。
3. 然后一个模型一个独立的文件夹，模型在文件夹内的命名必须是规范的，遵守这样的规范：
4. 攻击模型的格式=模型名+”\_attack\_”+编号+”.png”
5. 被攻击模型的格式=模型名+”\_defend\_”+编号+”.png”
6. 站立模型的格式=模型名+”\_stand\_”+编号+”.png”
7. 移动模型的格式=模型名+”\_move\_”+编号+”.png”

模型名指的是模型的名称，这里建议使用拼音或者缩写，编号指的是1，2，3，4，5这样连续递增的数字。所有模型的最低编号是从1开始的，”.png”指的是文件后缀只能是png

一切尽在图中



1. 然后将这个文件拖拽到texturepacker，在右边命名，这个命名务必和上文提到的模型名一样。



1. 然后点上面的发布精灵表，会得到两个输出文件，把这两个文件放入相关的目录内之后，开始编辑配置表。
2. 配置表编辑就很简单，自学吧。
3. 技能特效格式和之前的差不多，完整的格式是 模型名+”\_skill\_”+编号+”.png”
4. 技能特效所放的目录是skillAnimation，其配置表的编辑同样是，Animation，不过是第二个sheet。编辑非常简单，自学。

### D地图

地图Excel的编辑分为两个部分

1. 在地图sheet，写上地图的基本信息，包含地图ID，地图名，背景图，音乐。描述和音效这两项弃用。
2. 第二部分是地图事件，这部分主要是布置地图的事件，事件出现的坐标，事件出现的条件和事件内容。
3. 语法跟金X基本一致，但是事件不能重复的field，除了可以写once，也可以写具体的数字，表达可以触发几次。
4. 我基本上使用的是TAG类语法进行事件控制，tag语法更为灵活，可以添加，延时添加和删除TAG，其中延时添加用于实现和时间有关的事件。这一点需要参考J剧情CJ.xlsx。
5. TAG语法简介：
6. HAS.TAG 表示有什么标志
7. NOT.HAS 表示没有什么标志

### J剧情

剧情配置表语法和金X基本一直，但是我大量使用的是自定义语法。这一点没法具体介绍，去琢磨一下 配置表就很容易知道了。

剧情语法支持内联lua，内联lua语法可以调用任何游戏内资源，除非有足够的实力，否则不要使用这种语法。

### 其他配置表

1. 对skill外功有范围的修改，增加了耗蓝参数。奥义，内功，绝技暂时没有改动。
2. 成长模板可以设置每一级的属性点增加量
3. 开场问答合并到story实现，具体为J剧情开场。
4. 引入全新的任务系统，具体去看R任务，其中涉及到的语法我忘了，暂时就使用我展示的语法吧。
5. 删掉了之前的商店系统，设计了全新的商店系统，可以限制购买次数，和每一次购买的获得。可以设计花式购买条件，但是购买条件不等于最终消费，必须再定义最终消耗。值得一提的是，取消了银子，元宝这种设定，所有交易合并为物品，或者好感，或者其他可自定义的条件。
6. 头像选择，合并到配置表，可以分别定义男女的头像配置表。
7. 针对连续的提问，单独设计了配置表W问题，具体使用语法参考武林试部分。
8. 修改了配置表Z战斗，每一场战斗可以指定掉落
9. 重新设计了配置表Z资源，所有资源必须显式提供文件后缀，将不同类别的资源分割到不同的sheet，但是最终他们会合并到一个lua文件中。
10. 值得一提的是，如果有足够的能力，可以自己编辑转化规则，转化规则使用python脚本标记。
11. 加密去tools/encrypt.py会自动导出到gameRelease/src/的相关文件夹里面

## 结语

这个说明文档等于没有说，但是对总体的编辑方式有一定的提示作用。