**RandomChest&Reward的逻辑规则**

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 修改时间 | 修订人 | 备注 |
| V1.00 | 2022.02.18 | 李钊 | 创建文档 |
|  |  |  |  |

# 1. 概述

●本文为纯后端逻辑说明

# 2. 数据

●相关XML如下：

|  |  |
| --- | --- |
| 配置表 | 说明 |
| Item | 道具信息 |
| Random\_Chest | 使用道具时的奖励数据 |
| Enu\_Reward | 奖励信息的配置表 |
|  |  |

# 3. Enu\_Reward

●配置表数据范例如图所示：



【表2-1 Enu\_Reward表】

●数据id与type

101 = 【type=1】 = 代表无私有属性的英雄 = 填写方法 [101,heroid,0]

102 = 【type=2】 = 代表装备 = 填写方法 [102,equipid,种族标签]

103 = 【type=3】 = 代表道具 = 填写方法 [103,itemid,数量]

104 = 【type=4】 = 有私有属性的英雄 = 填写方法 [104,cardid,等级]

201~2xx = 【type=5】 = 数值类资产 = 填写方法 [2xx,0,数量]

301~3xx = 【type=6】 = 按时间和进度产速计算的数值类资产 = 填写方法 [3xx,0,秒数]

## 3.1 具体的填写逻辑

●101 = 【type=1】 = 代表无私有属性的英雄 = 填写方法 [101,heroid,0]

HeroId 填写的是 Hero\_Detail.xlsx中的ID

后端发给玩家这个奖励时，是按照HeroDetail中的初始化状态发的

0的占位符没有任何意义

●102 = 【type=2】 = 代表装备 = 填写方法 [102,equipid,种族标签]

EquipId填写的是Equip\_Detail.xlsx的ID

种族标签的填写

0=正常随机

-1=绝对没有标签

-2=肯定有标签

1=1族标签

2=2族标签

3=3族标签

4=4族标签

5=5族标签

6=6族标签

7=7族标签

=0=正常随机，要先随有无，有标签后随是什么种族标签

随有无，按照Equip\_Detail中的raceRate【raceRate=25=25%几率有标签】

如果有了标签，要随机是什么标签时：

读取DataSetting中的EquipRaceRateParam=[10,10,10,10,5,5,0]

其中7个整数依次代表7个种族标签的出现权重

如果是洗练标签时，权重读取EquipRefrashRaceRateParam，逻辑相似

●103 = 【type=3】 = 代表道具 = 填写方法 [103,itemid,数量]

ItemId填写的是Item.xlsx的ID

数值资产的映射道具是无效的，给予数值资产不通过配道具的形式给奖励

●104 = 【type=4】 = 有私有属性的英雄 = 填写方法 [104,cardid,等级]

CardId填写的是Gacha\_Card的ID

等级填写的是，英雄生成时的等级，这个等级如果超过了Quality的合法值，则取Quality合法值的上限

Quality的合法值在Hero\_Quality.xlsx的【合法值level】

●201~2xx = 【type=5】 = 数值类资产 = 填写方法 [2xx,0,数量]

●301~3xx = 【type=6】 = 按时间和进度产速计算的数值类资产 = 填写方法 [3xx,0,秒数]

用【 秒数 / 20 \* 对应产速 】= 数字资产的数量

## 3.2 关于粉尘的道具化映射显示

●因为【205粉尘】的【displayItem = 1】，所以这个数值资产要在道具背包中以道具化的方式显示，那么：

1. 用【205粉尘】的itemId值【映射道具id】去【Item表】中取这个“道具”的数据，用于显示

2. 所有道具相关逻辑都是一样的，与Item表中的其他道具逻辑一致

●与【205粉尘】的逻辑一样，如果【displayItem = 1】则一并处理

## 3.3 关于数值型资产在奖励预览或获得奖励时的道具化映射显示

●在奖励预览或者获得奖励界面中要显示数值资产时，要用道具化显示，那么：

1. 不用去考虑【displayItem = ?】的问题，统一到【Item表】中去寻找对应映射道具的数值

2. 用道具一样的逻辑去显示这些 数值资产

# 4. Item的Reward

●在【Item表】中，如果【usdType=1】时会直接按照Reward逻辑获得奖励，规则如下：

1. useEff则是Reward的值，填写方法=[int,int,int]

2. 读取的逻辑与之前讲的Enu\_Reward一致

3. 填写的多个Reward要并联处理，一并获得

4. 显示时填写在前的显示在前，填写在后的显示在后

5. 如果填写了多个相同的枚举项，比如2个10000金币，则用两个道具化图标显示奖励

# 5. Random\_Chest

●在【Item表】中，如果【usdType=2】时会直接按照Random\_Chest逻辑获得奖励，规则如下：

1. useEff则是Random\_Chest的ID，填写方法=int

2. 用Random\_Chest的ID去Random\_Chest.xlsx中去取对应索引的【reward】值

3. Random\_Chest的reward填写方法=[103,1001,1,1000],[103,1002,1,9000]

4. [a,b,c,d]中的[a,b,c]就是Reward填写方法，规则详见Enu\_Reward

5. [a,b,c,d]中的d,则是这一个奖励项的权重

6. 随机一遍会在reward填写的若干个奖励项中随机出一个作为结果

7. 一个Random\_Chest可以随机多次【time=要随机几次】，也就是每一次都可以有一个奖励，随机多次后，把奖励的集合，一并发给玩家

8. 显示的时候，先随机出的显示在前

9. 如果填写了多个相同的枚举项，比如2个10000金币，则用两个道具化图标显示奖励