GameCore开发·

记得更新一下数据库设计，我有一点改动，在Docs/fastma\_G123.sql。更改后转储数据与结构，命名为G123W1。

涉及的数据库：



麻烦做一下之前提到的关于外键的更改，聊天记录如下：

**同时会有多个game吗？**

**GeoWat 19:06:34**

**不会**

**GeoWat 19:06:38**

**就一个**

**GeoWat 19:07:06**

**Game表存的是网站总体的状态**

**王景楠 19:07:39**

**你这第一步是怎么根据iteration**

**王景楠 19:07:54**

**或者说你解释一下game的字段都是些啥**

**GeoWat 19:15:20**

**iteration:racetrack里不是有一个属性是iteration么，我是用它来标记每一个赛道的迭代，不然赛道这个表里面的字段每8个不就重复了嘛**

**GeoWat 19:15:48**

**total\_volume:整个游戏的投注代币量**

**GeoWat 19:16:00**

**user\_count:用户数量计数**

**GeoWat 19:16:19**

**racetrack\_row:有多少个赛道，现在设计的是8个**

**GeoWat 19:17:00**

**is\_betting:是否正在下注，1就是可以投注，0就是锁定筹码，进入赛跑阶段**

**王景楠 19:17:59**

**我懂了**

**王景楠 19:18:05**

**但你这个game表的设计就很**

**王景楠 19:18:24**

**不把他当数据库用 而是把它当作了内存在用**

**王景楠 19:18:54**

**你这样岂不是game表永远只有一条记录**

**王景楠 19:19:59**

**你应该把iteration改为game\_id之类的 每当一个新游戏开始就插入一条新记录 这样还可以把游戏的开启历史记录下来**

**GeoWat 19:20:13**

**soge**

**王景楠 19:20:23**

**然后racetrack的一个字段外键关联到game表**

**王景楠 19:20:34**

**racetrack和game是多对一关系**

**GeoWat 19:21:37**

**betorder表里也有一个属性是iteration**

**王景楠 19:21:43**

**racetrack里面的stake\_token是啥**

**王景楠 19:21:55**

**那就把iteration当作game的主键**

**王景楠 19:22:06**

**别的表外键关联到game**

**GeoWat 19:22:21**

**用来标记订单下在哪个迭代里**

**王景楠 19:23:00**

**那一样的逻辑 改为标记订单是对应到哪场游戏**

**GeoWat 19:23:20**

**知道了**

**GeoWat 19:23:41**

**stake\_token就是目前有多少个token下注在了这一个赛道上**

**GeoWat 19:23:52**

**方便结算**

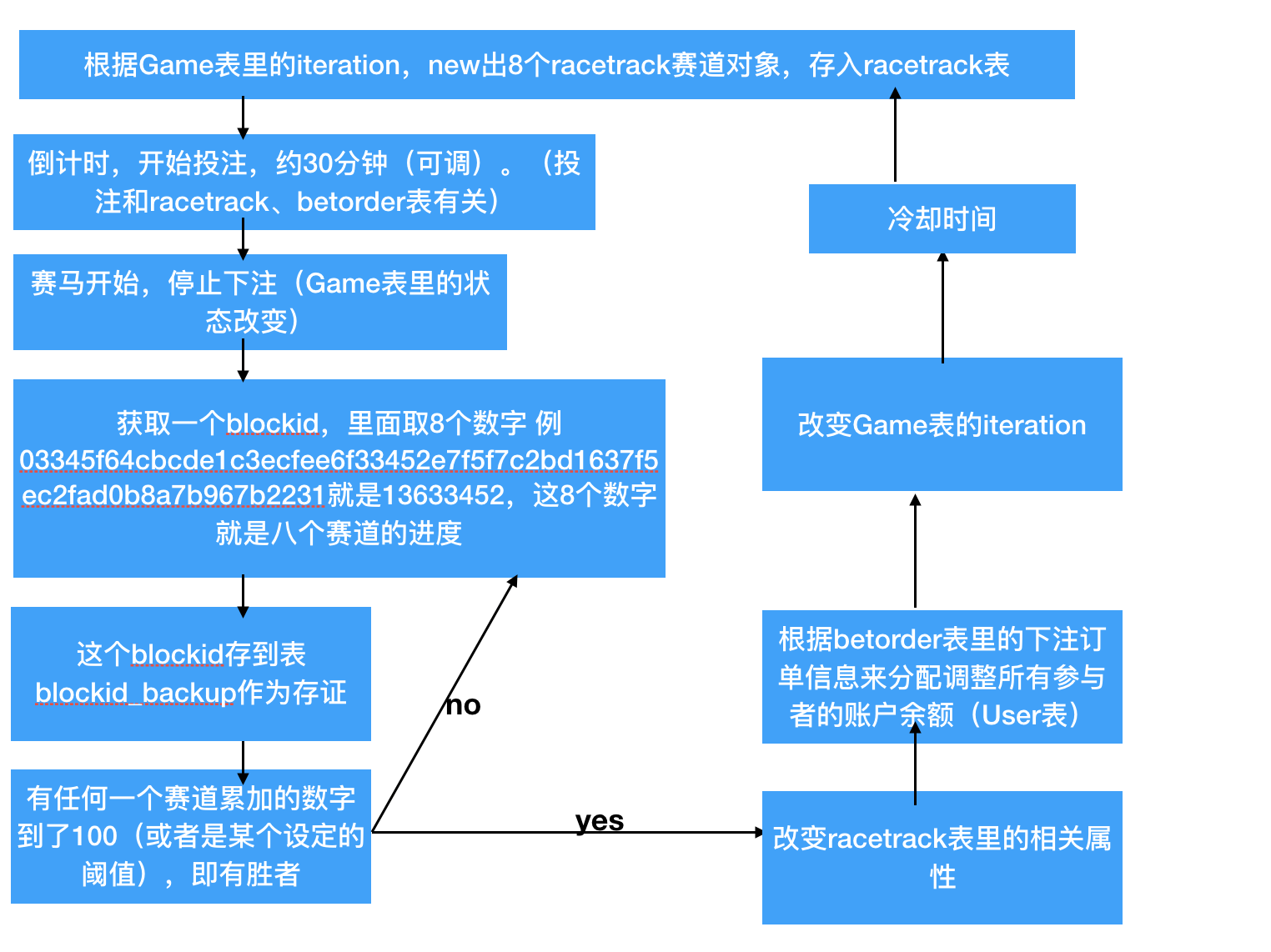
**王景楠 19:23:57**

**噢**

**GeoWat 19:24:54**

**因为结算应该是此用户下注token数\*（输家总token数-我们的抽成）/ 赢家总token数**

流程图如下：



对外的接口如下：

投注：

post，/order 验证当前游戏状态，如果还没有开始比赛（可以投注的状态，未在结算冷却时间），就进行投注，投注订单记录到betorder表，每个赛马（在数据库里是赛道）获得的投注量增加，投注后要在user表减去相应余额。

Get，/order 获取所有的投注订单

开放信息：

Get /racetrack，获取racetrack表里当前迭代（iteration）的所有信息。最重要的是每个赛道的进度值distance，全部的用户会周期性的获取这个值以便前端马的跑动展示。

Get /game 获取game 表里的全局游戏信息

Get /blockidbackup 获取blockid\_backup表里的所有存证信息

备注：

关于赛马进度算法：

由于现在是初期，第一版能用就行，所以用http请求的方式隔一段时间获取一个最新的blockid，获取blockid之后取八个数字，这八个数字就视为这个时间间隔里8个马跑出的距离，然后累加，谁先达到100谁就赢了

关于结算算法：

每次有胜者之后，游戏进入冷却状态，开始结算所有的订单，目前使用最简单的结算算法，就是赢了的那条赛道的所有人，依据投注数量平分输了的其他赛道的所有投注的token

某人赢得的token=（所有输了的赛道的token之和）\*（某人投注的token数/所有赢了的人押的token数）

如果赢了的赛道没有人投注，那就把所有的token计入一个叫做winpool的user名下，等于归入奖池，日后在其他的活动中使用

你写的后端文件放在backend/GameCore里，可以和我写的nodejs后端区分开（用别的方式也可以）