

12306--如何实现查票、购票



菠菜谈天说地

百家号 | 04-23 12:36

[关注](#)

前言

记得几年前，一到过年用12306买票就成为了一大难题，各种服务不响应，后来就是各种显示排队中...当时还有同事跟菠菜开玩笑说，要不你跳槽过去给他们优化优化？也有同事开玩笑说12306是几个毕业生做的...开了个人公众账号后，菠菜就一直挂念着，如果我去设计12306，该如何支撑超高并发场景呢？不过呢，考虑到大家的阅读时间，菠菜会分为两篇文章来写，本文着重讲一下如何实现查车次，查余票，购票的，下一篇文章会着重写一下如何支撑春节的购票高峰。

设计难点

一个车次，会经途n个站，而n个站点都可以单独的拆开来售卖，比如：北京->廊坊->天津->德州->济南，用户可以买北京->天津的，也可以买天津->济南的，那如何根据我的起始点查询到相应车次，以及购票，是需要好好设计下的。

数据结构

先来定义几个名词：train表示列车，ticket表示票，site表示站（如北京站、廊坊站），order表示购票记录。它们的结构定义如下：

```
type Train struct { // 列车
    ID      int64    // 列车的id
    Type    int      // 列车类型, 1表示普快, 2表示动车...
    Num     string   // 列车的车次, 如 G331
    Sites   []*Site  // 列车所经过的站点, 按照顺序排列
    Tickets []*Ticket // 列车所包含的车票
}

type Ticket struct { // 车票
    ID      int64    // 车票ID
    Type    int      // 车票类型, 1表示硬座, 2表示硬卧, 3表示软卧, 4表示一等座, 5表示二等座...
    TrainID int64    // 从属的列车ID
    Sites   []*Site  // 车票所包含的站点, 按照顺序排列
    Lock    *lock    // 锁
}

type Site struct { // 站点
    GlobalID int64    // 站点ID, 全局统一
```

[去百度App听](#)

```

Type      int    // 站点类型, 1表示始发站、2表示过路站、3表示终点站
Name      string // 站点名称
DepartTime string // 离开时间, 如 +0,03:00 或 +1,03:00 前缀+0表示当日时间, +1表示次日依次类推
ArrivalTime string // 到达时间, 格式如上
Price     int64  // 上一站到这一站的售卖价格, 单位: 分, 仅仅ticket引用的时候使用
Status    int    // 状态, 0表示可以售卖, 1表示锁定, 2表示已经售卖, 仅仅ticket引用的时候使用
}

type Order struct { // 购买
    OrderID      int64    // 订单ID
    UserID       int64    // 用户ID
    PurchaseTrainID int64    // 购买的列车ID
    PurchaseTicketID int64    // 购买的车票ID
    PurchaseSites []*Site  // 购买的站点ID
    Status       int     // 状态, 0表示完成, 1表示待付款, 2表示取消
}

```

打开百度App, 看更多图片

注：这里值得注意的一点是，乘客购买的其实不是ticket，而是ticket中的一段，只不过有些乘客会购买整段的ticket而已。另外所有的结构定义仅仅给出了关键字段，不求全。对于这些结构如何在MySQL中存储，大概如下：

```

create table `train` (
  `id`      bigint
  `type`    int
  `num`     char
)

create table `site` (
  `id`      bigint
  `name`    varchar
)

create table `train_site` (
  `id`      bigint
  `site_id` bigint
  `train_id` bigint
  `depart_time` varchar
  `arrival_time` varchar
)

create table `ticket` (
  `id`      int
  `train_id` int
  `type`    int
)

create table `ticket_site` (
  `id`      bigint
  `site_id` bigint
  `train_id` bigint
  `price`   int
  `status`  int
)

create table `order` (
  `id`      bigint
  `user_id` bigint
  `train_id` bigint
  `ticket_id` bigint
  `sites`   varchar
  `price`   bigint
  `status`  int
)

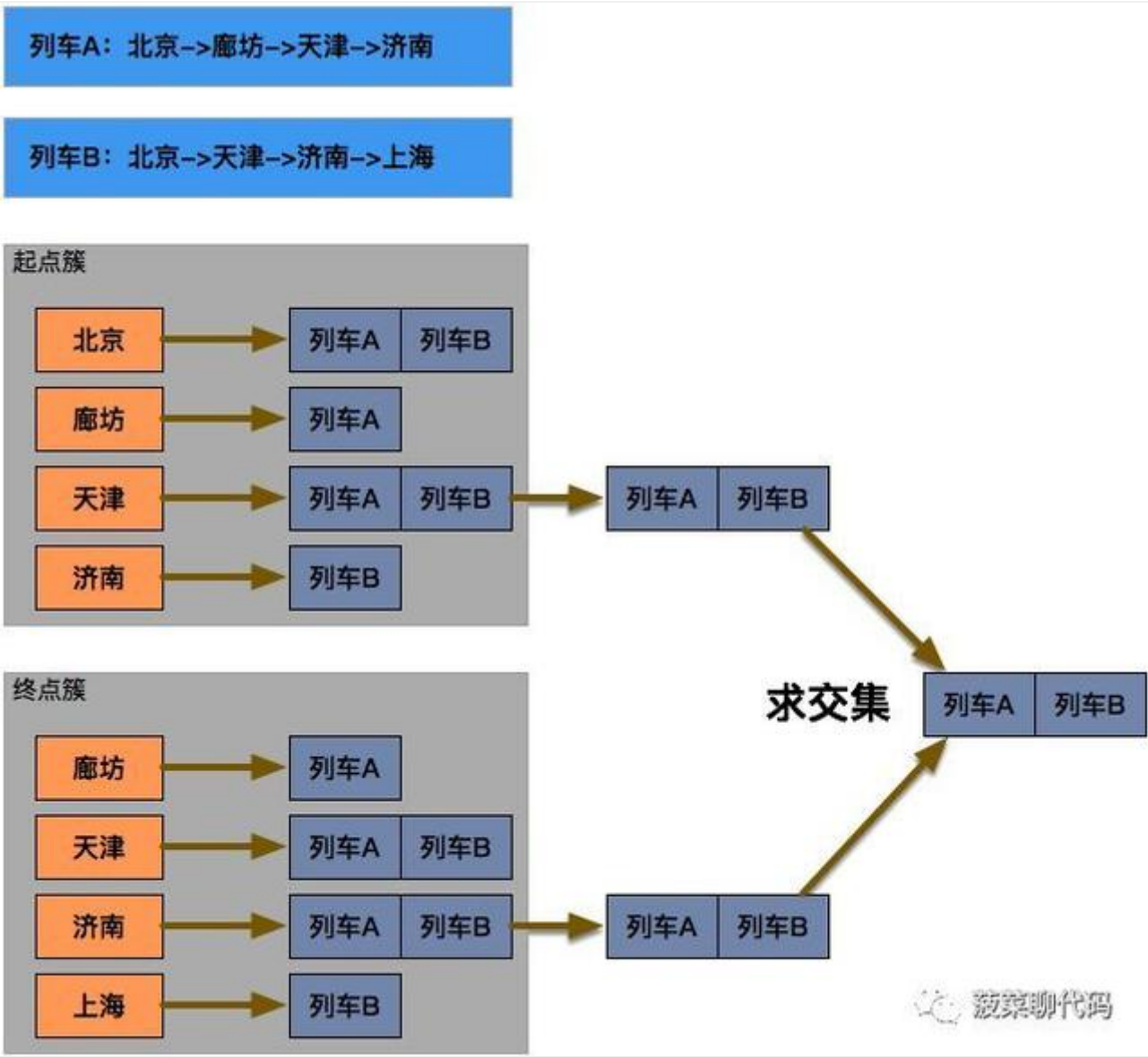
```

如何检索

如何检索车次

去百度App听

这里将用到倒排索引（菠菜之前在RTB-DSP中介绍过，有兴趣的小伙伴可以再看看）。首先按照天来建立索引（后边在介绍类似索引时默认去掉了天的维度，是为了方便），然后每天里会建立两个索引簇：起点、终点，最后遍历所有列车，根据列车所包含的站点建立相应索引，注意的点是，列车的起点只能在“起点簇”中建立索引，列车的终点只能在“终点簇”中建立索引，而其他站点在两个簇中都要建立索引。如下图：



假设有两列车，A、B，建立索引后，如果检索“天津->济南”的列车，最终会检索出A、B都可以到。

如何检索余票

查询出列车后，列车对应的票就自然而然的可以得到，这里余票检索到的办法是遍历，如果使用倒排的话，成本比较高，主要的问题在于下图：

假设有一张票是北京->济南的，而中途廊坊->天津的已售，那么参考车站的倒排索引模式，起点簇有北京，终点簇有济南，而如果用户查询北京->济南的余票，就会将这张实际不符合的车票查询出来。

而遍历会有性能问题吗？我们可是要支撑春节高并发场景的。菠菜做了一个测试，假设一列车有20(车厢) * 20(一节车厢的排数) * 5(一排有5座) = 2000张票，而每张票假设有20个站点，即40000万个站点要遍历，那经过菠菜测试，耗时不超过40微秒，假设用户查询余票一次查询50趟列车的余票，那总共耗时在**2毫秒**，性能足够了。

不过还有一个地方需要达成共识，就是在高并发下，**余票一定是不准的，在查询的时候是允许有误差的**。但是在购票的时候，这个误差是不允许存在的。导致误差的原因有很多，比如当查询出余票为3的时候，返回给客户端展示，但是在这个时间段内，已经有他人成功购买了一张票，这时余额应该是2。

如何购票

由于查询余票我们采取了看似暴力的遍历方式，然后这让购票变的简单。其实高并发下，写操作一定需要锁，而锁的粒度越大，整体的性能越低，而目前的数据结构，会让锁尽可能的小。

首先用户会选定某一车次，并且给出起终点，然后参考上边的查询逻辑，查询出车次的全部余票，**注意，这里是余票**，而不是全部车票，因为如果是拿全部车票去逐个遍历是否可以购买，会让锁的范围变大。

遍历余票，需先对ticket进行加锁，然后再次确认该票是否符合用户需求（查询跟操作是两个时间点的事儿，怕这两个时间点之间车票有变化），大体如下：

```
/*
 * 简易购买代码，返回参数为购买到的Ticket
 * tickets 表示余票
 * startSite、endSite 表示起终点
 */
func buy(tickets []*Ticket, startSite *Site, endSite *Site) *
```

去百度App听


```
one.Lock.Lock()
if isAllTicketSitesCanBuy(one, startSite, endSite) {
    canBuy = true
    updateTicketSitesAlreadyBuy(one, startSite, endSite)
}
one.Lock.Unlock()

if canBuy {
    err := saveOrder()
    if err != nil {
        recoverTicketSites(one, startSite, endSite)
        return nil
    }
}

return one
}

return nil
}
```

 菠菜聊代码

最后

本篇文章是在比较微观的角度去看如何实现基本的查票、购票过程，下篇文章会在比较宏观的角度去上思考，如何支撑春节超高并发的流量，而且还要保证高可靠性。

本文由百家号作者上传并发布，百家号仅提供信息发布平台。文章仅代表作者个人观点，不代表百度立场。未经作者许可，不得转载。

[举报](#)

相关阅读

8款菠菜的烹制菜肴，喜欢吃菠菜的朋友一定要收藏了！

[去百度App听](#)[打开百度App](#) 小咖美食



打开百度App 东方头条

压力传感器种类，压力传感器种类及误差分析



打开百度App 电动邦服务

菠菜怎么做好吃？菠菜的做法大全，这样的家常菜能抗衰老！



打开百度App 东方头条

打开百度App看更多资讯

精彩视频

春运抢票渐白热化 27日起试点川渝方向候补购票

02:28

打开百度App 生活欢乐汇

2019就要到了，2018的flag你实现了吗？到“武答案

去百度App听

02:12

[打开百度App](#) 楚天都市报

抢票不怕了！12306自动抢票“神器”上线，网上排队候补购票了

02:31

[打开百度App](#) 杭州24小时

官方抢票功能上线？12306“候补购票”了解一下 春运买票就靠它了

01:32

[打开百度App](#) 江西TV都市情缘

热门推荐

腰间盘突出严重影响生活！骨科专家力推的一个方法，一劳永逸！



[打开百度App](#) 东方头条



停机维护近17个小时还卡顿？玩家：这回该杀个策划祭旗了吧？

[打开百度App](#) 电竞加百利

大话降龙：伏虎变“断臂独眼侠”？田螺大仙的弟子“有医保”！



[打开百度App](#) 守候的书本

威马纯电动SUV将在今天上市销售 续航600km! 补贴后15万元起



打开百度App 甲壳虫开汽车

搞笑趣图

好有中国气质的名字，哈哈哈



打开百度App 熊孩子基地



搞笑图片：我画的是抽象的画，你们不懂

打开百度App 熊孩子基地

打出租车的就是比等巴士的拽



打开百度App 蠢萌萌宠

不知道这车开到60码坐着是什么感觉

去百度App听



[打开百度App](#) 搞笑奇趣汇

热门评论

[打开百度App,说说我的想法](#)

去百度App听