

你知道的越多，你不知道的越多

点赞再看，养成习惯

本文 **GitHub** <https://github.com/JavaFamily> 已收录，有一线大厂面试点思维导图，也整理了很多我的文档，欢迎Star和完善，大家面试可以参照考点复习，希望我们一起有点东西。

## 前前言

为啥今天有个前前言呢？

因为你们的丙丙啊，昨天有牌面了哟，直接被微信官方推荐，知乎推荐，也就仅仅是还行吧（心里乐开花）。

也有好几个号主发来问候，直接实名羡慕哈哈，丙丙以后也是被微信推荐过的男人了，你们也是有牌面的读者了。



## 前言

你们是不是以为丙丙这么忙了，肯定不会写技术文章给你们看了？

丙丙暖男来的，不过说实话年前估计是没啥时间写技术文章了，上周末两篇你们也看到了，不是年会就是首脑峰会，实在很忙。

这周也是一样，我们年底了项目也赶，我今天一列下个Q的计划，我的个乖乖，平均三天一个需求，什么鬼啊。

而且周四又有技术部的年会，我还是视频的导演（打杂仔哈哈），肯定需要确认很多细节，到时候我应该还可以周末水一篇给你们看（剧透下那一期会有我女神入镜）。

下周周二我就坐车回家了，打工仔回家过年了，我也叫我妈把我要回家的消息放出去了，估计媒人这会在路上了，到时候相亲我又可以水文章了，过年素材也好多呀，开心。

这一篇主要说一下我之前的一个很愚蠢的bug，本来只打算让他呆我笔记里面的，但是还是忍不住想要分享出来，让大家避免这种低级错误（其实我想水一篇多少有点技术内容的文章，免得写N篇全是水日常的文章，你们估计又要Diss我了）。

## 正文

我先描述一下bug的现象哈：



业务方组员: 鸡蛋,敖丙 数据方组员: 鸡蛋,敖丙

注:填写组员中文名,用逗号隔开 (第一次相关人员需要TT订阅:广告投放系统通知)

这两个输入框的值都是我从KV（Redis之类的存储中间件）里面获取到的，也是可以实时修改的，我自作聪明，想着KV里面如果没值，就默认取一个静态变量，这样有个兜底，在类加载的时候加载静态变量，我取的时候也效率高了。

这就也为我后面的Bug埋下了伏笔，问题是这个Bug烦就烦在他在预发环境是好的，线上却是坏的。

先看看代码怎么写的：

```

public static String SEND_PEOPLE = "敖丙,鸡蛋";
public static String BUSINESS_USERS = "敖丙,鸡蛋";
public static String DATA_USERS = "敖丙,鸡蛋";

@Autowired
private MoguCacheFactory moguCacheFactory;
@Autowired
private OppoGoodItemsJob oppoGoodItemsJob;

static {
    String name = DspAdminMBContext.get("UI_Users");
    String bsUsers = DspAdminMBContext.get("business_users");
    String dataUsers = DspAdminMBContext.get("data_users");
    if (StringUtils.isNotBlank(name)) {
        SEND_PEOPLE = name;
    }
    if (StringUtils.isNotBlank(bsUsers)) {
        BUSINESS_USERS = bsUsers;
    }
    if (StringUtils.isNotBlank(dataUsers)) {
        DATA_USERS = dataUsers;
    }
}

```

可以看到代码里面，我是在静态代码块去KV取值，如果有值就用KV的做初始值，没取到我也有默认值，我当时还在想自己的构思真巧妙，用KV比DB效率高，用常量去做兜底，不至于没配置的情况没有值，报空指针啥的。

当时一个劲给自己加油打气，一个劲的妙啊，不知道自己写了多蠢的代码。

这样写看似没什么问题，但是我这个值是可以修改的这就有问题了，而且我有几个地方还是取的变量，不是一直取的KV。

而且常量被我当变量用，也都违背了这玩意的初衷了。

我上线后，三歪就说：敖丙出大问题了，为啥我刷新一下值就不对了，再刷新值就回来了。

我惊恐万分，深知我写了BUG，我不着痕迹的擦去我鬓角的汗水，深呼吸了一下。

回答道：我代码还没写好呢，你怎么乱点，我告诉过你不要乱点的，现在你点坏了吧。

三歪好像真的以为他错了，委屈巴巴的啥也没说就默默离开了。



这个时候我马上打开电脑，也顾不上歪歪的感受了，思绪转的飞快，开始在闹海中构建整个值传递的链路。

我改了页面的值，之后我也改了这个静态变量和KV的值，但是我在线上发现我刷新页面一会是修改之后的，一会是修改之前的值。

这就奇怪了呀，主要是在预发还不能复现，这就很坑爹了，难受呀歪歪。

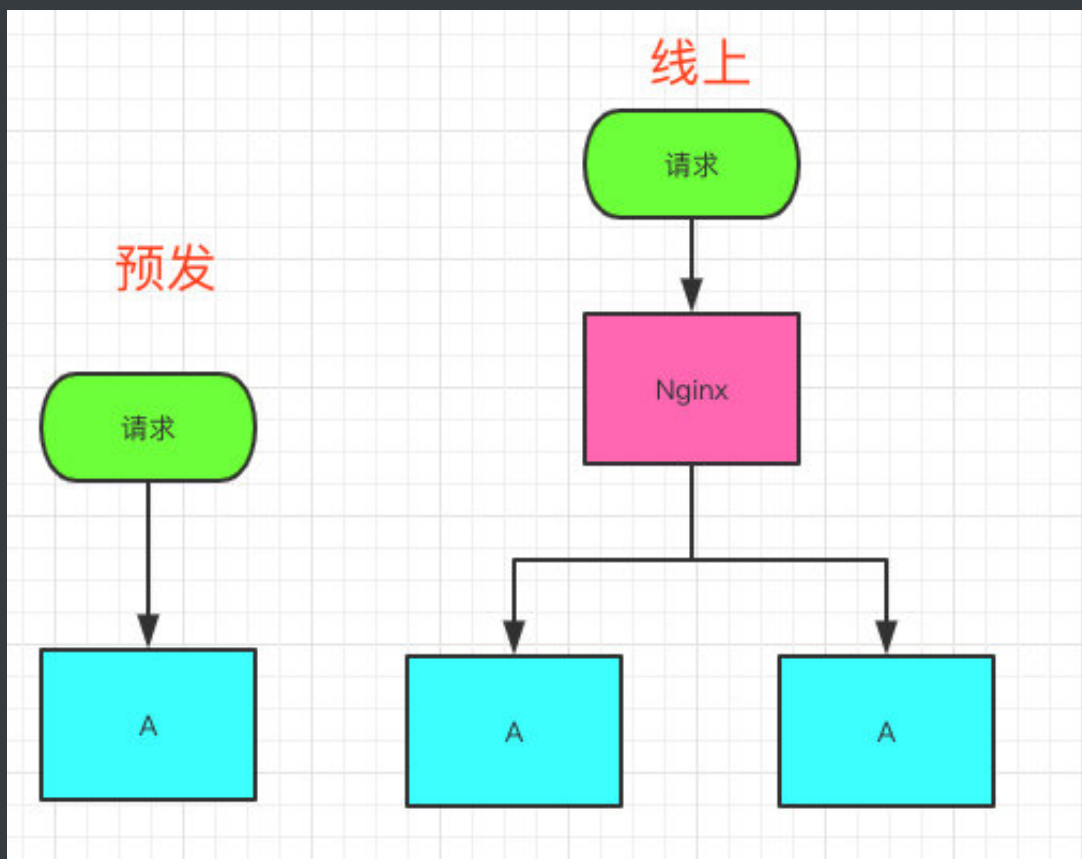
Tip：预发指的是，代码跟线上一样，数据库一样，环境配置不一样。

我把代码看了一遍又一遍，还是没发现问题。

在当天那个饱暖思淫欲的下午饭之后我，突然灵光一闪发现了问题的核心，预发是好的因为他是一台机器。

而我们线上的机器是负载均衡的，有两台机器，我修改一次只改了一台机器的值，另外一台没修改到。

那么问题就简单了，我们看下图：



预发情况，是单机，不管我怎么读怎么修改，每次数据都是准确的（这里我还没意识到其实每次去取变量是不对的）。

但是线上不一样，服务器启动的时候给每台机器赋值了，这样的情况如果只是读是没问题的（怪不得静态常量都是private的，能修改的话基本上都有这个问题）。

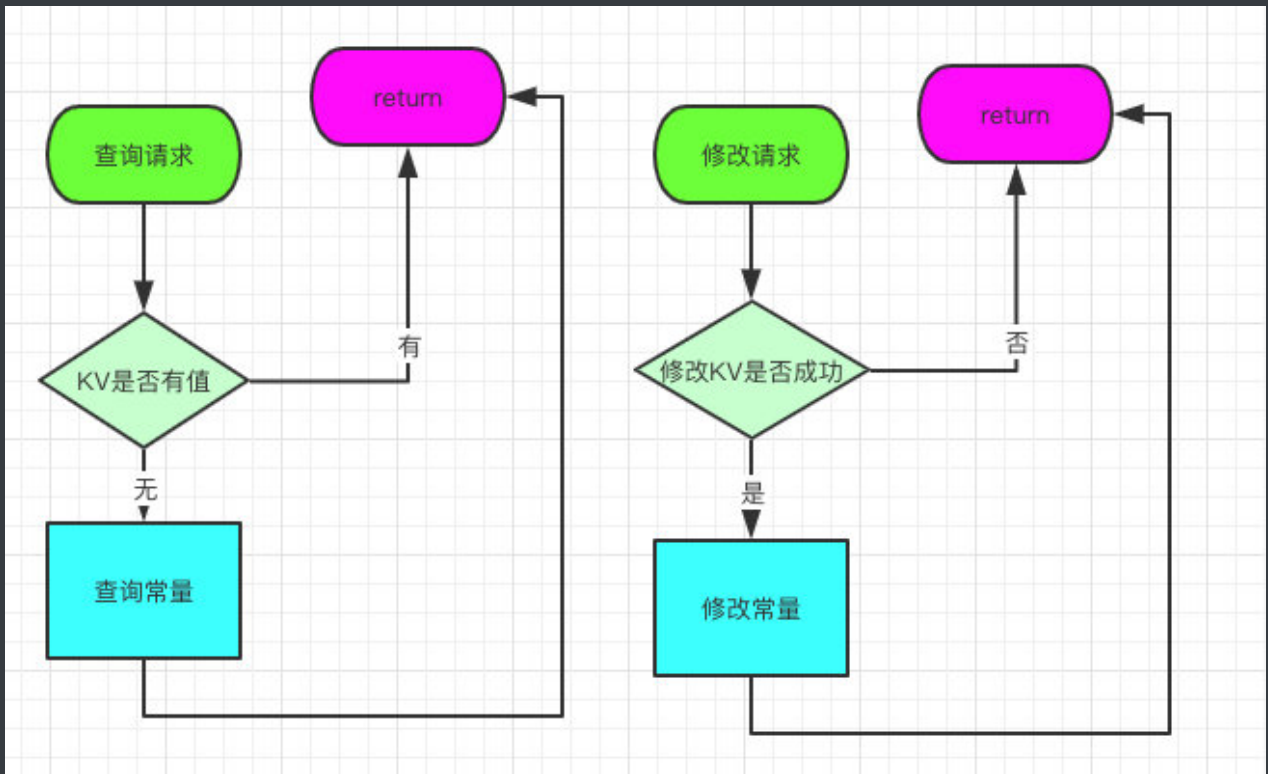
但是如果修改，你修改的请求只是打到了一台机器上，如果下次负载均衡请求到了这个机器，那么你很幸运，结果是对的，但是往往负载均衡算法就是那么公平，雨露均沾这样的情况 1/2 的请求，都会拿到错误的结果。

而且这个地方的问题还有一个就是，应该是修改KV之后，修改变量，我有的地方取的是变量，因为我想每次反正都改了，取也没事。

其实正确的做法是我每次去都读KV就好了，只有KV为空的时候才去读静态变量，作为一种兜底方案，不应该去改变常量的值。

而且大家要知道，我修改KV成功万一修改常量的时候出错了呢？你取常量的值就不对了。

所以查询，修改应该这样做：



主要是想提醒大家，谨慎操作静态常量，能不变就不变，不要写这么骚的操作了，我这算少的很多40、50台线上机器的服务，这种问题是真的难发现。

我是万万没想到，我居然写出这种代码，下次再写出来，只能让歪歪不要乱点了，免得又点坏了哈哈。

## 点关注，不迷路

好了各位，以上就是这篇文章的全部内容了，我是敖丙，励志做一名让大家都记得住的博主，能看到这里的人呀，都是人才。

我后面会每周都更新几篇一线互联网大厂面试和常用技术栈相关的文章，非常感谢人才们能看到这里，如果这个文章写得还不错，觉得「敖丙」我有点东西的话 求点赞👍 求关注❤️ 求分享👥 对暖男我来说真的 非常有用!!!

白嫖不好，创作不易，各位的支持和认可，就是我创作的最大动力，我们下篇文章见！

敖丙 | 文 【原创】

如果本篇博客有任何错误，请批评指教，不胜感激！

---

文章每周持续更新，可以微信搜索「三太子敖丙」第一时间阅读和催更（比博客早一到两篇哟），本文 **GitHub** <https://github.com/JavaFamily> 已经收录，有一线大厂面试点思维导图，也整理了很多我的文档，欢迎Star和完善，大家面试可以参照考点复习，希望我们一起有点东西。





## 联系我/公众号

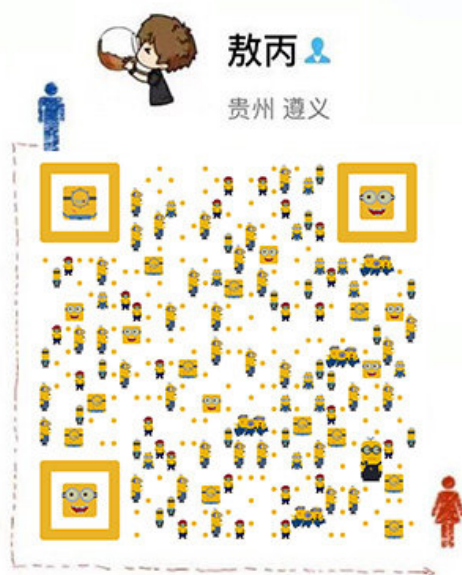
关注公众号回复“加群”我会拉你进技术交流群，说实话在这个群，哪怕您不说话，光看聊天记录，就能学到很多东西。

(阿里技术专家、Java3y作者，拼多多的技术大牛都在)

拉你的就是我个人微信我会交流每期写作思路，平时10点以后或者周末可以问我一些职场问题，工作日比较忙周末都在。

每个月我也会抽一些小伙伴线下喝咖啡交流心得，也会抽小伙伴帮忙看简历，广告的收入也会拿一部分买书送读者。

微信搜索「三太子敖丙」第一时间阅读或者扫描二维码↓。



微信搜一搜



三太子敖丙

别爱我，没结果