

点赞再看，养成习惯，微信搜索【三太子敖丙】关注这个互联网苟且偷生的工具人。

本文 **GitHub** <https://github.com/JavaFamily> 已收录，有一线大厂面试完整考点、资料以及我的系列文章。

## 前言

这又是一期读者的面经分享，很巧的是，他在蘑菇街面了三轮，最后虽然没过，但是也希望分享出来大家瞅瞅。

我这周可能会单独做个大厂面试流程的视频，涉及每一轮的考察点注意事项，如果觉得有必要，可以留言让我知道你们想看啥。

## 面试经历

### 一. 11.20 字节跳动一面Java开发，直接挂（耻辱开头.....）

1. 介绍主要项目，怎么做的；
  - 本来想简要介绍做的业务，但面试官要求详细介绍，所以二十分钟都在介绍业务；
  - **送命问题：数据量多少？**说了实话，实际使用单表1000—10000级别。导致后面基本上面试官根本不想问问题了.....
2. 对 Spring 的理解？
3. 对 AOP 的理解？
4. 讲一下 Java 的静态代理和动态代理

然后就没了，进入问问题环节..... 当然知道已经凉了。面试官说的问题主要在于，在研究所的技术栈还是太落后，说互联网的思路和我们这种人不一样。在提了并没有什么实际性的建议之后，结束面试。

收获：

1. 第一次面试，终于踏出了这一步；
2. 认识到了自己与一线大互联网公司的差距，待继续努力；
3. 由于面试时的准备方向错误，在数据量方面的成果无心之失直接判了死刑，并没有表达出来真正的性能和准备的内容。以后准备，一定要向**大数据量、优化**等方向考虑！
4. 表达能力太差。【对..... 的理解？】这种问题一抛出来，尝试用浅显易懂的方式

### 二. 12.04 蘑菇街一面高级 Java 开发

1. 介绍主要项目（大概二十五分钟左右）；
  - 经过上一次失败，这次梳理了业务的主线，比较熟练的介绍了业务，依旧花了十五分钟，不过中间穿插了询问项目方面的问题；

- 询问问题：
  - (1) 动态模板的数据源从哪儿来？答：资源目录注册，通过各个数据服务获取；
  - (2) 增量数据比较多的情况下，数据源怎么处理？答：一方面，数据源通常都是数量比较固定的内容，由各个业务来封装处理，对于增量数据比较多的情况，使用动态模板的关联表模式；另一方面，数据源数据比较多的时候，通常会使用树状结构，数据服务有通用的懒加载处理方式，不会出现性能瓶颈；
  - (3) 动态模板怎么做数据的逻辑处理？答：动态模板只负责数据的记录，关于逻辑处理是另外一个业务，在简历中也有介绍。然后介绍了动态汇总的设计逻辑，大概十分钟。（说明面试官对做的项目有兴趣，并且我的回答已经有了引导性）
  - (4) 如果逻辑处理过程中并没有抛出技术性的异常，而是计算错误的异常（类似于 1+1，结果算出来是 3 的错误），有没有什么处理？（问的是有没有计算监控功能）答：没有，只是较为严密的关注了技术上的异常，计算上的错误没有处理。如果真的处理了这种问题，需要通过运维人员在配置页面中清除 Redis 缓存。

## 2. 线程池基础

- 对于一个普通的线程池，coreSize = 5, maxSize = 10，阻塞队列长度 20，且插入线程是永久执行的，那么不断插入线程，线程池中的数量以及对应的反应如何？答：该问题只需了解线程池的运作原理即可回答。

## 3. Spring AOP

- 类内部调用 AOP 问题：a() 方法被 @Around 注解用来输出日志，b() 方法没有 AOP 注解，但 b() 方法内部调用 a() 方法，那么调用 b() 方法会有怎么样的输出？答：不会有日志输出。因为这是类内部调用，而 AOP 是通过代理进行的，类内部调用不会调用代理类，不走代理所以不会有日志输出。如果想要有日志输出，需要在该类中通过 AopProxy 将代理对象（命名为 proxy）获取，然后在 b() 方法中通过 proxy 调用 a() 方法；

4. 为什么临时决定找工作？答：因为比较熟悉这里的工作，想找挑战。

5. 向面试官提问问题。问了问业务内容，技术栈。

## 三. 12.13 蘑菇街二面高级 Java 开发

整体就是介绍主要项目，时间总共四十分钟左右。

### 1. 动态模板

- 动态模板的主线流程已经比较熟练了，该方面流程介绍没有什么问题。
- 询问问题：
  - (1) 动态模板的数据源从哪儿来？答：资源目录注册，通过各个数据服务获取；
  - (2) 动态模板里面的数据怎么来？答：用户添加了模板之后，往里面录入数据；

2. 有堆 Dump 和数据库连接池优化经验，怎么做的？

### 3. 动态汇总统计

- 此次面试的绝对雷区。在叙述该部分的时候，我说明了该项目是我设计的。但在描述项目的设计理念时，并没有像上次一样描述的很清楚，反而描述卡顿，中间一度难以进行下去。该部分的设计一定要形成稿子，并且熟练表达，否则适得其反。

4. 简历中写了统计的效率优化，具体是怎么做的呢？

- 两个出发点进行优化。

- 第一是粒度，我们有的汇总结果是需要统计多个表的。原本在实现的时候，由于要求赶时间上线，又有性能要求，所以当时同事写的时候，在增删改数据时，我们的数据服务触发了订阅，通知集群中所有服务数据更新的事件，所有服务又分别发起了五个表的查询请求，对查询结果进行处理后存入缓存。

也就是说，改一次数据，会导致一个服务的五次查询，如果集群里有十个服务，那么改一次数据就会触发五十次查询。但是从业务的角度来看，改了一个表的数据只需要清除该表的缓存即可，其他四个表的缓存并不需要修改，也就是粒度太粗，所以我第一步是将粒度细化到表的级别。

- 第二是历史数据。因为我们的业务场景是允许查询时间段内的信息，而业务场景又有限制：今天之前的历史数据是不能修改的。也就是说我们可以把历史数据作为长时间缓存，今天及其以后的数据作为短期缓存，修改数据只触发短期缓存的更新。

#### 5. 简历上写了使用 ZK 的分布式锁，为什么要选用这种实现方式？使用场景是怎么样子的？

- 业界分布式锁通常情况下是用 Redis 的 setnx 来实现的，但当时我们的 Redis 还没有搭建起来，缓存用的是 Memcache；此外，我们的注册中心基础是 ZK，而 ZK 的瞬时顺序节点是可以实现分布式锁的功能，所以我就使用了。
- 使用场景：我们的汇总比较麻烦一些，所以进行一次汇总时间相对较长。虽然通常汇总结果都会放到缓存里，但如果某一时间缓存中没有数据，又有大量并发进行汇总结果的查询，那么就会引发穿透。所以一方面我和同事搭建了布隆过滤器，防止缓存穿透的情景。

另一方面，大量并发同时查询汇总结果时，由于服务做了负载均衡，所以集群中的服务都会接收到请求，缓存中没有汇总结果，会所有大量服务进行计算。但其实只需要一个服务进行缓存的计算即可，其他服务的计算属于资源浪费行为。所以我对一个需要进行汇总的业务使用了分布式锁，用查询条件作为锁，获取到该锁的服务进行计算，并负责将计算结果置入缓存，其他服务在尝试获取该锁的时候失败，返回一个默认的结果，这样便避免了多余的计算。

#### 6. 有什么需要问面试官的问题？问了问业务内容，技术栈。

### 四. 12.20 蘑菇街三面高级 Java 开发（失败）

共 40 分钟左右；

#### 1. 自我介绍

#### 2. 简历上写有 JVM 优化经历，怎么做的？

#### 3. 如果设计一个自动监控系统，如何设计？如何实现？

- 由于在刚才的 JVM 优化过程，目的是监测某个压测问题，我介绍了一个通过 JVM 检查问题原因的流程。所以面试官提出了该问题。
- 关于自动监控系统的设计，我了解有 ES, Logstash, Kibana 的开源套件，有助于日志收集和监控，但具体我没有过多了解；
- 如果是基于我们的系统进行实现的话，应该会基于我们的注册中心开发一个新的新的监控服务。我们的注册中心以 WAR 包的粒度，将对应服务注册到 ZK 集群上。所以新的监控服务设计的基本思想，应该是对所有注册到 ZK 上的服务发送心跳包，收集各服务所在服务器的系统指标、服务运行相关信息；
- 面试官追问：监控哪些服务器和服务的哪些指标？怎么监控？

- 答：服务器指标应该监控服务器的 CPU 占用率、内存使用量、网络各状态的连接数、网络带宽；服务的指标应该监控 Heap 堆的容量、单位时间内 GC 次数、各状态线程数量；
- 具体监控的方法我认为 Java 中应该给出了 API 接口可以获取这些信息，因为之前我之做的业务操作日志记录系统中，有 IP 信息的获取记录；当时就是调的 Java 的 API，所以我感觉应该有相似的调用方法；

#### 4. 索引如何设计？

- 是由于我说我们目前业务的日志操作监控系统是把数据存到了数据库中，所以提到了该问题。
- 我的回答：
  - 我们构建索引时主要针对几种情况：
    - 对查询使用较多的字段考虑构建索引；
    - 可能会比较多使用到筛选、排序的字段；如比较常用的时间类型；
    - 有下拉树唯一标识性质的 ID 字段；

#### 5. ZK 怎么用的？选举机制？

- 提到 ZK 是因为我提到我们的服务注册到了注册中心上，注册中心是用 ZK 为基础的；
- 我的项目中接触到 ZK，主要是在服务集群计算的时候为了节省计算资源，所以使用 ZK 作分布式锁；此外 ZK 也用来我们注册中心的注册；

#### 6. 互联网的架构了解哪些？

- 回答了秒杀系统，具体有点忘了。（可以看看丙丙的历史文章，秒杀是我的爆文）

#### 7. Spring 的思想？代理如何实现？

- Spring 的思想主要是按照之前写的博文进行描述的

#### 8. 为什么选择使用 SSM 框架？为什么使用 Spring boot？

- 使用 SSM 框架的原因主要是这是整个业界比较通用的框架，而且也是部门的通用框架；
- 使用 Spring Boot 的原因是，Spring 和 Spring MVC 有很多配置文件，比较繁琐，而 Spring Boot 将很多配置给集成了进来，使用基于注解的配置方式比较简单方便。

#### 9. 了解的互联网常用的技术栈还有哪些？

- 除了简历上写的，还提到了 Nginx 用于反向代理和负载均衡；如果到达了 Nginx 的反向代理极限，可以使用 F5, LVS 进一步扩充并发量的支持，但我们的并发量没有到达这种程度，所以没有再向上了解。

#### 10. 秒杀系统中，缓存与数据库的一致性如何保证？库存，网络波动。

## 絮叨

可以看出每一面都不容易哈，大家真的要好好学习呀。

Tips：我有个读者已经面进我们公司做我的同事了，丙丙还和他约了一楼咖啡厅喝茶，到时候也会分享出来。





我是敖丙，一个在互联网苟且偷生的工具人。

创作不易，不想被白嫖，各位的「三连」就是丙丙创作的最大动力，我们下次见！

---

文章持续更新，可以微信搜索「三太子敖丙」第一时间阅读，回复【资料】【面试】【简历】有我准备的一线大厂面试资料和简历模板，本文 **GitHub** <https://github.com/JavaFamily> 已经收录，有大厂面试完整考点，欢迎Star。



你知道的越多，你不知道的越多