

天异云同学面经

- 1、自我介绍
- 2、学习内容
- 3、学习笔记
- 4、面试部分
- 6、简历指导
- 7、面试总结
- 8、面试共性题总结
- 9、彩蛋：天异云同学的采访

1、自我介绍

儒猿-石杉架构课

大家好，我是天异云，有10年多的工作经验，今年31岁，大专毕业，后来自考了本科，个人技术很菜。

<https://docs.qq.com/pdf/DY3dHR2RDTG9CUU91>

2、学习内容

学习时间：周一到周五午休和下班后的整块时间，以及地铁上的零碎时间。周末全天学习。

学习内容：架构班主JAVA+辅大数据，部分课程重复重修，儒猿技术窝全部课程（吸收的不是很好，后面要重修）

JAVA部分和大数据部分如下：

推荐的学习顺序

4	《互联网Java架构师训练营》	第一阶段：从0到1：开发10万行代码量的大型电商系统v1.0版本 02_站在架构师的角度为团队设计软件开发流程和规范
5	《互联网Java架构师训练营》	第一阶段：从0到1：开发10万行代码量的大型电商系统v1.0版本 03_10万行代码的大型电商系统v1.0版本开发实战
6	《互联网Java架构师训练营》	第二阶段：电商系统的分布式架构实战 01_Spring Cloud从0基础入门到精通核心组件源码
7	《互联网Java架构师训练营》	第二阶段：电商系统的分布式架构实战 02_基于Spring Cloud的大型电商系统分布式架构重构实战
8	《互联网Java架构师训练营》	第二阶段：电商系统的分布式架构实战 03_多场景下的企业级分布式事务技术深入底层剖析
9	《互联网Java架构师训练营》	第二阶段：电商系统的分布式架构实战 04_分布式电商系统的核心链路数据一致性实战
10	《互联网Java架构师训练营》	第二阶段：电商系统的分布式架构实战 05_多种开源技术实现的企业级分布式锁源码剖析
11	《互联网Java架构师训练营》	第二阶段：电商系统的分布式架构实战 06_大型电商系统之核心交易接口的一致性保障实战
12	《互联网Java架构师训练营》	第五阶段：复杂业务场景下的系统设计实践 01_精通基础开发技术之JDK集合包源码剖析
13	《互联网Java架构师训练营》	第五阶段：复杂业务场景下的系统设计实践 01_精通基础开发技术之JDK并发包源码剖析以及真实生产案例实战
14	《互联网Java架构师训练营》	第五阶段：复杂业务场景下的系统设计实践 01_精通基础开发技术之讲给Java工程师听的大白话网络课程
15	《互联网Java架构师训练营》	第五阶段：复杂业务场景下的系统设计实践 01_精通基础开发技术之Java NIO技术精讲以及原理剖析
16	《互联网中间件架构师实战训练营》	第三阶段：基础架构体系中的技术中台设计与研发 01_自研分布式海量小文件存储系统
17	《互联网Java架构师训练营》	第五阶段：复杂业务场景下的系统设计实践 01_精通基础开发技术之Netty技术精讲以及源码剖析
19	《互联网中间件架构师实战训练营》	第二阶段：企业级微服务技术体系的自主研发 01_自研企业级微服务技术栈框架
20	《互联网Java架构师训练营》	第八阶段：每秒数十万高并发的系统架构设计 06_互联网顶级大厂的超高并发秒杀系统架构设计
21	《互联网Java架构师训练营》	第五阶段：复杂业务场景下的系统设计实践 3、专治复杂业务场景的领域驱动模型设计
22	《互联网Java架构师训练营》	第六阶段：支撑数十亿量级的海量数据架构设计 01_海量数据存储技术选型以及架构设计
23	《互联网Java架构师训练营》	第六阶段：支撑数十亿量级的海量数据架构设计 02_自研分布式唯一ID生成中间件
24	《互联网Java架构师训练营》	第六阶段：支撑数十亿量级的海量数据架构设计 03_10亿级订单量的订单系统分库分表架构设计
25	《互联网Java架构师训练营》	第五阶段：复杂业务场景下的系统设计实践 04_基于DDD设计交易核心之复杂订单系统
26	《互联网Java架构师训练营》	第五阶段：复杂业务场景下的系统设计实践 02_精通核心开源开发框架

<https://docs.qq.com/pdf/DY3dHR2RDTG9CUU91>

序号（大数据）

课程目录

1	第一阶段：大规模分布式文件存储系统的架构剖析 01_hadoop hdfs从0基础入门到精通源码
2	第一阶段：大规模分布式文件存储系统的架构剖析 02_hadoop源码二次开发、系统优化以及问题修复
3	第二阶段：每秒百万吞吐的消息中间件架构剖析 01_支撑每日10亿数据量的Kafka架构实战
4	第二阶段：每秒百万吞吐的消息中间件架构剖析 02_探索世界上最复杂的消息系统：Kafka内核源码分析
5	选修课：大数据工程师的职位分析 01_大数据架构师能力模型、职业规划以及年薪展望
6	选修课：电商行业数据分析需求介绍 02_大型电商数据化运营平台项目需求分析
7	选修课：企业级数据采集系统的架构设计与实践 01_每日TB量级的数据采集平台架构实战
8	第三阶段：企业级分布式协调系统的架构剖析 01_ZooKeeper顶尖高手课程：从实战到源码
9	选修课：数据仓库基础知识入门突击 03_大数据面试突击课程设计说明
10	选修课：数据仓库基础知识入门突击 04_大数据面试突击之离线数仓
11	第四阶段：分布式海量数据存储系统内核剖析 01_HBase从入门到精通
12	
13	
14	

3、学习笔记

1、《从 0 开始带你成为JVM实战高手》（下面链接是我做的笔记汇总）

有道云链接：<http://note.youdao.com/noteshare?id=30d822dad76304f5fda97fe8caf43f20>

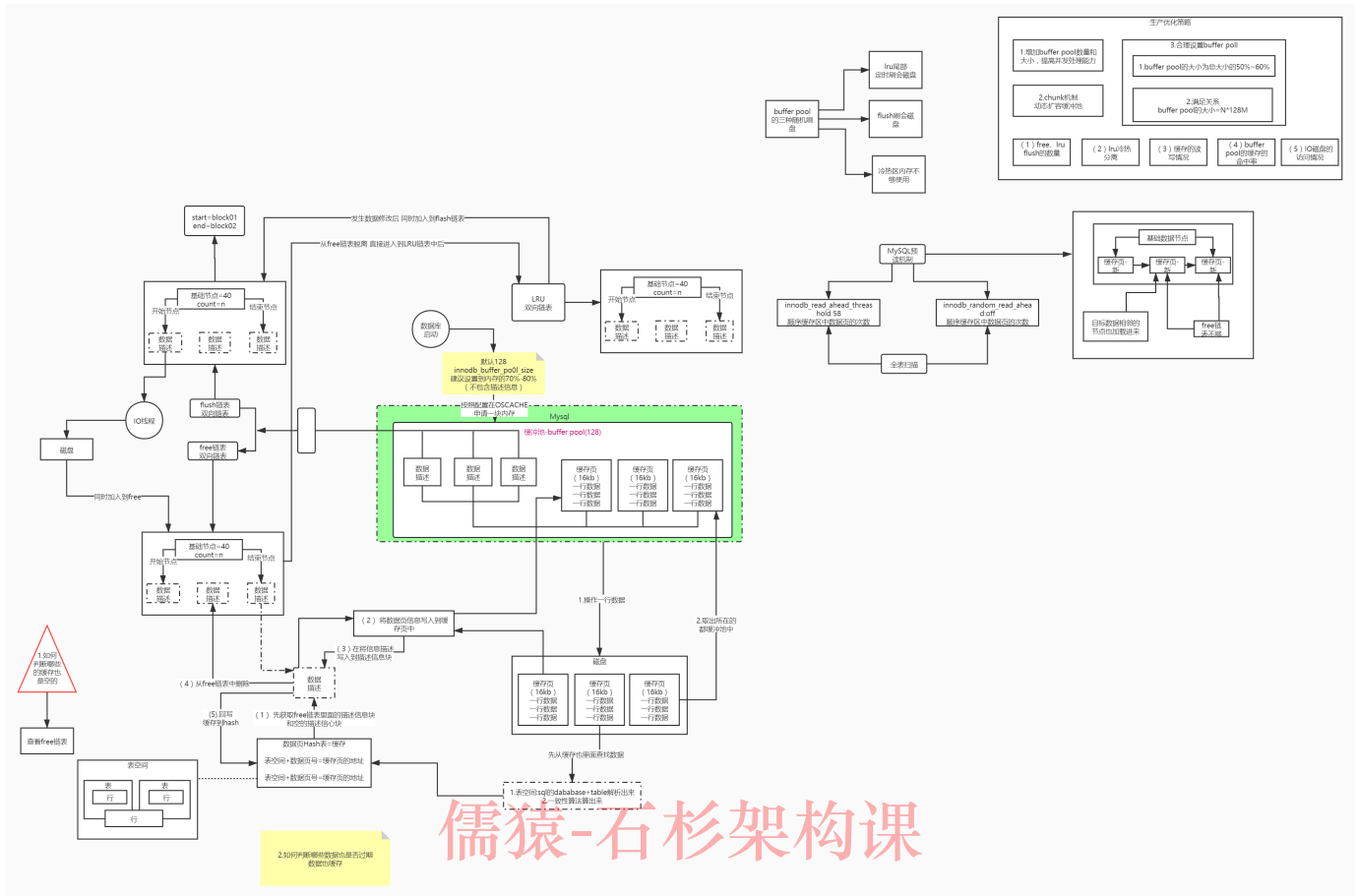


儒猿-石杉架构课

2、《从零开始带你成为MySQL实战优化高手》专栏部分流程图：（原图链接）

<https://www.processon.com/view/link/60ad1fe707912906fbd3c402>

<https://docs.qq.com/pdf/DY3dHR2RDTG9CUU91>



<https://docs.qq.com/pdf/DY3dHR2RDTG9CUU91>

3、架构班并发JUC部分流程图：(原图链接)

<https://www.processon.com/view/link/5f13f8c8e401fd06f3e26eb5>



4、架构班分布式事务笔记：（下面链接是我的笔记汇总）

笔记：<http://note.youdao.com/noteshare?id=d4b563faeb73d1baea957aa177a194ae>

分布式事物

源码分析

6-最大努力通知

5-最终一致性

4-接口幂等性

3-Saga事务

2-TCC事务

0-基础理论

2PC - 3PC - Paxos - Raft(Kafka - eureka - AP) - ZAB(ZK - CP)

7-总结-常用的分布式事务-和-场景

自己设计一个分布式事务框架

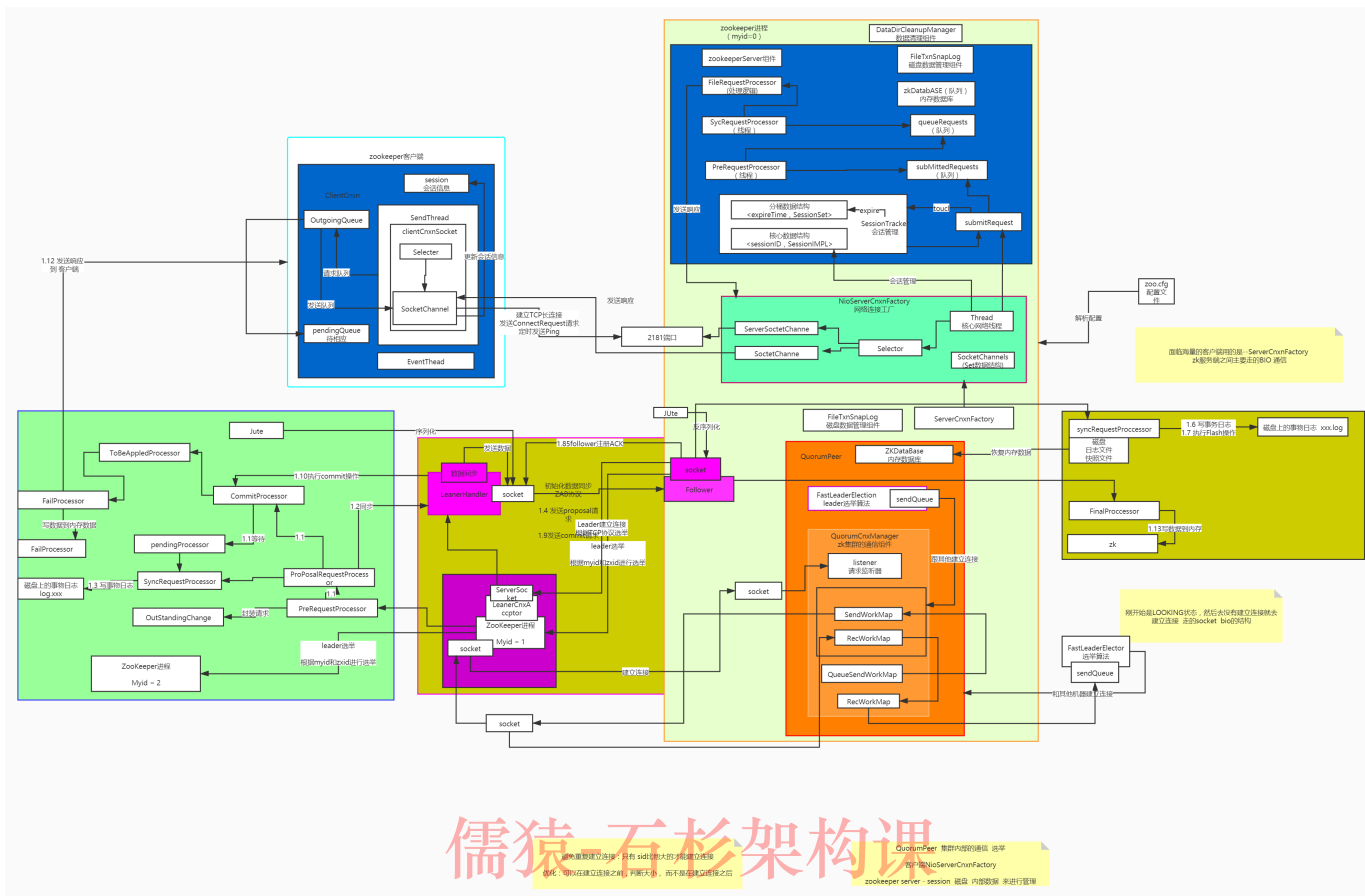
儒猿-石杉架构课

5、架构班Kafka源码部分：(原图链接)

<https://www.processon.com/view/link/5f340b0ae401fd51f06ce835>



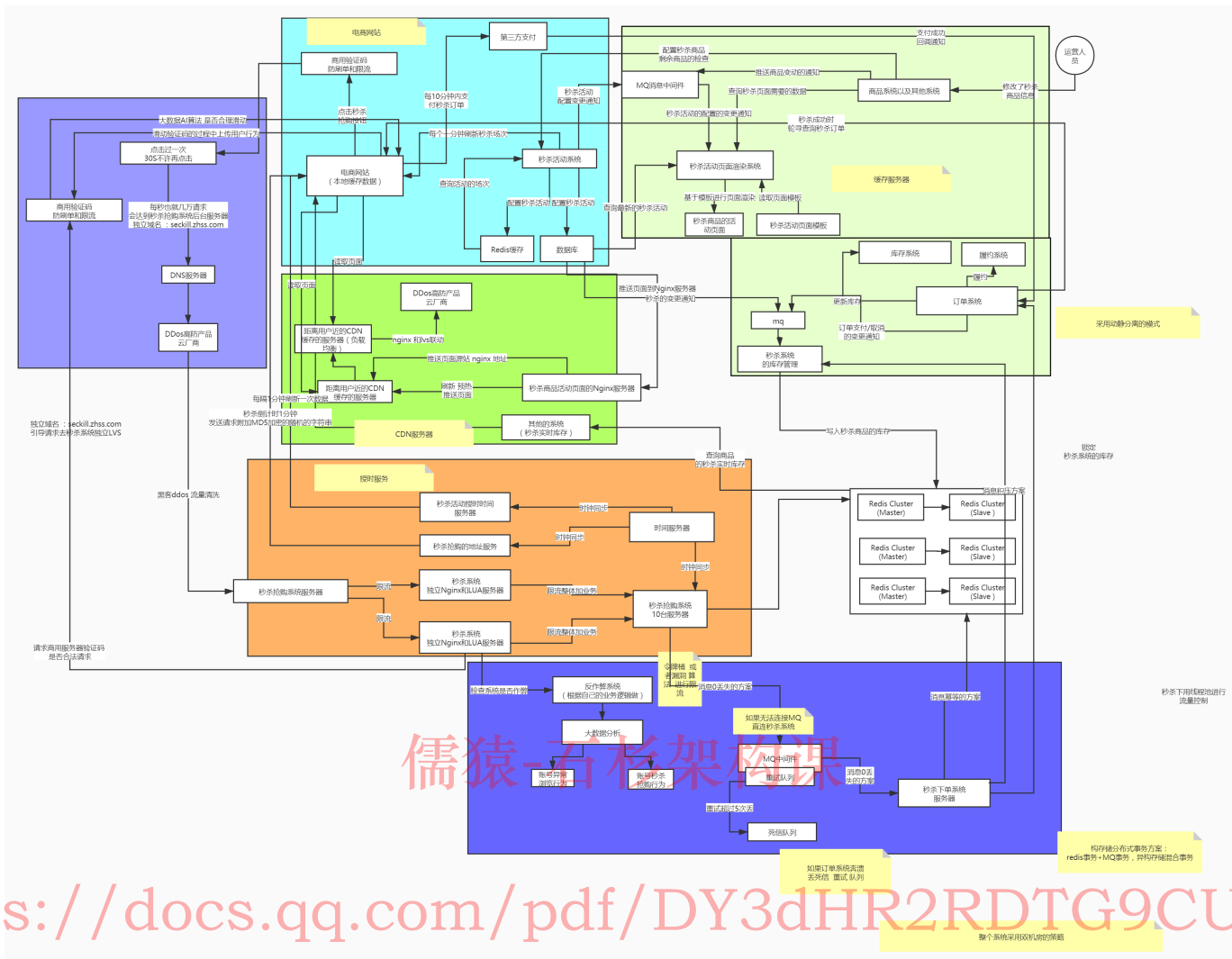
<https://www.processon.com/view/link/5f13a2da1e08537d50b2bb42>



<https://docs.qq.com/pdf/DY3dHR2RDTG9CUU91>

7、秒杀流程图：（原图链接）

<https://www.processon.com/view/link/5f0b28ab5653bb6858b70f4e>



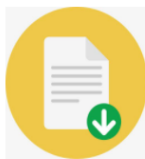
4、面试部分

1、首先去学习系统里，申请跳槽指导，整体流程：



请下载简历投递文档后，按照指导文档去进行投递。每天可对面试过程进行反馈，老师会对反馈进行回复。可以查看内推职位信息，通过学员去进行职位内推。

如果要进行offer选择指导，请加石杉老师的QQ号码（[1861313135](#)）



简历投递文档

上一步

查看申请详情

面试答疑

内推职位列表

2、学习系统里也有其他同学发布的内推，架构班同学内推是很好的方式，可以更好的提前了解面试公司的情况和部门情况：

批量审核通过					
<input type="checkbox"/>	职位名称	发布人手机号	发布人QQ号码	发布人用户名	审核状态
<input type="checkbox"/>	Java开发工程师	1861313135	521112006	森	审核通过
<input type="checkbox"/>	Java高级开发工程师/开发专家	13975111317	961112275	Spurs	审核通过
<input type="checkbox"/>	高级/资深Java工程师	1397511131	351117092	杰克叉烧包	审核通过
<input type="checkbox"/>	软件开发工程师	1761111190	651118534	养喵少年	审核通过
<input type="checkbox"/>	Java开发工程师	1345011112	301115256	bighong0404	审核通过
<input type="checkbox"/>	java工程师	1570711171	741117013	mccken	审核通过
<input type="checkbox"/>	Java开发工程师	1766111117	471117923	狗十七	审核通过
<input type="checkbox"/>	Java高级/资深/技术专家	15811131164	4471110948	云鹤	审核通过
<input type="checkbox"/>	资深Java工程师/架构师	1811111176	641117874	Pz	审核通过
<input type="checkbox"/>	Java高级开发工程师	1397511131	511114092	bing'wen'wu'hen	审核通过
<div><div>< 1 2 3 ... 11 > 到第 1 页 确定 共 107 条 10 条/页</div></div>					

3、准备时长：

4个月左右，前3月需要用来复习，差缺补漏，面一些不想去的公司或者更高端的职业来看自己的不足

试水情况：刚开始10家直接面资深/架构，直接挂，业务聊不透，后来咨询架构班学员修改简历，有了很大改善，复试和终面的机会多了，再后来咨询老师，先上岸，找独角兽或者大厂。

4、面试指导：（跟老师的部分聊天）



老师我现在在物联网的小公司 做的都是一些 导入导出工作，流程相关，30岁了这次跳槽是进入互联网公司吗？电子商务一类？创业型和中小都可以吧？之前只关注工资这一块，业务啥的没考虑，这次注重有并发亮的公司，那直播行业算吗？



石杉

同学，你现在这阶段，还不到找一个业务深扎的时候



石杉

你先要进大厂或者独角兽，刷履历，刷两年，然后跳出来，进大厂或者独角兽当架构师



石杉

先上岸



石杉

这个时候你是不考虑业务的



石杉

当上大厂或者独角兽架构师了，接着再找一个有深度的业务领域去深扎，成为业务专家

5、面试情况：

庆幸最终拿到了独角兽offer，在没有报架构班以前，基本只有一面就挂，在架构班学习之后基本能拿到复试，或者拿到终面试的机会

7、作为老学员总结架构课：

架构班课程的学习是为了更好的铺垫自己的职业生涯，如果认为报了班就能出去随便拿offer，可能会背道而驰，课程学习需要你付出时间和行动，不断的总结复习自省，到底是哪里出了问题，多和身边优秀的人学习

6、简历指导

1、改造之前的简历：

儒猿-石杉架构课

<https://docs.qq.com/pdf/DY3dHR2RDTG9CUU91>

项目经历

项目一 云

项目简介：

ProudThink工业服务云，从用户端的服务价值出发，聚焦智能设备的网络价值，充分考量运营商、制造商、服务商三类服务对象的业务发展需求，为其提供设备接入、监控诊断、资产管理、远程维护、预测预警等一站式 SaaS 服务平台，实现云端的智能设备运营生态。

项目用到的技术：

Springboot , Spring, SpringMVC, Redis, Kafka, Mysql

项目问题一：

由于在项目内部告警调用接口比较频繁，会大量卡主接口，导致项目无法接口挂起

解决方案：

监测系统告警：系统内接口某些时刻调用频繁，接口大量卡住，有些调用可以走异步消费，很明显，要用 mq 来解耦接口调用，用 mq 对于高并发做消峰，同时 mq 走异步消息发送提升接口性能。

参考一下因素，最终选择 Kafka 来解决上述问题。

考虑因素 1：特殊场景下，会产生大量高并发。可能达到几万甚至十万级并发量。

考虑因素 2：实现高可用且后期方便扩展-分布式。

项目问题二：

根据现场的人员反馈，在正常的情况下，会存在数据丢失情况的发生，对于预防生产停产会造成损失，报警信息缺失，更可能造成安全的生产事故

解决方案：

排查过程发现，在发送数据和接收端都不存在数据丢失的问题，然后和运维工程师，对 Kafka 排查发现是 Kafka 的 ack 设置有问题，将 Kafka 的 ack 参数设置成 1，防止数据丢失

2、改造之后的简历：

在没有修改简历之前的面试一团糟，能收到的面试邀请不是很多，在修改后整体面试邀请的成功率提升不少，基本能达到投递10个，就有一半的邀请，所以建议大家严格按照老师的要求是修改简历和准备面试



个人总结

- 6年研发经验
- 负责过系统的核心业务的架构设计与研发，系统有千万日增数据，深入阅读过多个开源项目的源码，对Spring Cloud、MySQL、JVM、并发编程、Redis等技术有源码级研究

工作经历

- [模糊] [模糊] 至今
- 工业处理上亿规模数据的海量存储，每日百万消息数据
- [模糊] [模糊]
- 参与需求调研，讨论，设计系统，开发代码，维护系统
- [模糊] [模糊] 月
- 开发主R，项目成员五人，负责多个项目的对接开发工作，负责核心模块的开发，指导多个项目的子系统的运营与开发

教育经历

海口经济学院
软件工程 本科

儒猿-石杉架构课

2016年07月 - 2020年12月
海口

专业技能

1. 7年JAVA开发经验，高并发，高可用有过项目实践
2. 熟悉网络编程（Netty）、并发编程、JVM内存管理
3. 熟练使用SpringMVC、SpringBoot、SpringCloud 能快速搭建微服务
4. 深入理解SpringCloud常用组件源码，研究过Eureka、Ribbon、Feign、Zuul核心源码
5. 熟悉分布式事务，2PC、3PC、可靠消息最终一致性方案、最大努力通知方案
6. 深入理解MySQL、SQL优化、ShardingSphere分库分表、索引优化、性能调优、数据库灾备 等实战经验
7. 消息中间件：掌握RocketMQ、Kafka、Zookeeper原理及集群布，对Zookeeper、Kafka有过源码级研究
8. 熟悉Redis的数据备份恢复方案、高可用架构，对Nginx + Redis + Ehcache 多层缓存架构方案、缓存雪崩解决方案、缓存重建并发冲突解决方案
9. 熟悉DDD领域驱动设计，在项目中有过应用
10. 其他：了解并使用过Hadoop、Spark、Hive、Hbase等大数据相关技术
11. 熟练使用IDEA、Eclipse等开发工具，GIT、SVN等辅助工具

项目经历

项目一、智维工业服务云

项目简介:

ProudThink工业服务云,为其提供设备接入、监控诊断、资产管理、远程维护、预测预警等一站式SaaS服务平台,实现云端的智能设备运营生态。

项目用到的技术:

SpringCloud, Redis, Kafka, Mysql

主要工作内容:

1. 参与设备告警上报的需求分析,搜集告警的常用数据项,进行技术选型和指导技术方案核心的业务流程
2. 告警数据上报之后,需要对告警进行分类分发

3. 制定技术方案

1)Mq技术选型方案:

由于在项目告警调用接口比较频繁,会大量卡主接口,导致项目无法接口挂起

解决方案:

监测系统告警:系统内接口某些时刻调用频繁,接口大量卡住,有些调用可以走异步消费,很明显,要用mq来解耦接口调用,用mq对于高并发做消峰,同时mq走异步消息发送提升接口性能。

参考一下因素,最终选择Kafka来解决上述问题。

因素1:特殊场景下,会产生大量高并发,可能达到几万甚至十万级并发量。

因素2:实现高可用且后期方便扩展-分布式。

2) kafka出现百万消息积压。

技术挑战:

hbase日增每天100G数据,告警数据上报 -> Kafka-> 接收存入Hbase,生产上出现了百万消息积压,导致外挂系统数据统计错误。系统部署4台4核32G服务器,服务器内存、CPU都在合理范围内,出现消息积压的原因是对账流程过于复杂,响应比较慢。

解决方案:

首先,增加消费者的线程数量,提升CPU的利用率。其次,每次拉取消息的时候,可以选择批量处理

实现效果:

极大的增加了消费消息的速率,提升系统资源利用率。其实出现消息积压的时候,首先就是要考虑增加消费能力,不能只是一味的增加消费者。

3、个人履历提升:

在架构班助教的2次助力的情况下,个人的履历有了明显了,从最初的外包->没有知名的A轮融资公司->现在估值10亿美金

下面分类中的公司仅仅作参考，每个公司都会因为各自的情况在不同分类中浮动。

1. 互联网大厂（知名度耳熟能详）
 - 百度，阿里巴巴，腾讯，京东，美团，今日头条，滴滴，小米，拼多多等。
2. 一线互联网公司（知名度很高，市值在几十亿美元规模以上）
 - 快手，好未来，爱奇艺，哔哩哔哩，众安保险，跟谁学，微博，58，360，网易等。
3. 中型互联网公司（有知名度，市值较低，一般在10亿美金以内）
 - 凤凰网，世纪佳缘，荔枝FM，尚德机构等。
4. 互联网独角兽（还没上市，知名度很高，估值也很高）
 - Keep，boss直聘，水滴，好大夫在线，作业帮，猿辅导，每日优鲜，vipkid，掌门教育，叮咚买菜，小红书，哈罗出行，货拉拉，喜马拉雅等。
5. 传统大型公司
 - 永辉超市，平安科技，顺丰科技，龙湖地产，华润地产等。
6. 小型互联网公司（还没上市，处于ABCD轮融资，没什么知名度，但也是互联网行业）
7. 非外包小公司（有自己的产品，公司规模比较小）
8. 外包公司（专门做别人的项目，包人头、包项目）



激活 Windows
转到“设置”以激活 Windows。

儒猿-石杉架构课

7、面试总结

面试之前原理性的课程必看，要能自己讲出来原理，突击1-3必备，JVM，MQ(其中一个必备)，mysql 必备，面试突击一针对普遍性问题，突击二，可以帮助你修改简历，突击三：基础性问题，另外多刷题，多总结，面试前可以尝试录音，回来路上多听，问题出在哪里

8、面试共性问题总结

1、索引的底层数据结构 为什么用的是B+ 不用B- 还了解哪些树

根据磁盘查找存取次数往往由树的高度所决定，所以，只要我们通过某种较好的树结构减少树的结构 尽量减少树的高度

首先你要回答出mysql中索引采用的是B+树结构，然后把B+树的结构描述下，延伸到索引下推，索引覆盖，聚簇索引、辅助索引等等。

1)、Hash哈希，只适合等值查询，不适合范围查询。

2)、一般二叉树，可能会特殊化为一个链表，相当于全表扫描。

3)、红黑树，是一种特化的平衡二叉树，MySQL 数据量很大的时候，索引的体积也会很大，内存放不下的

而从磁盘读取，树的层次太高的话，读取磁盘的次数就多了。

4)、B-Tree，叶子节点和非叶子节点都保存数据，相同的数据量，B+树更矮壮，也就是说，相同的数据

量，B+树数据结构，查询磁盘的次数会更少。

然后关于explain你要回答出一下几个核心字段，序号、type、possible_keys、Key、rows、extra，及每

个字段有哪些常见的项

儒猿-石杉架构课

2、如何进行分库分表

分库分表的原理：将一个库里面分配到多个库里面

垂直切分：垂直拆分一般是按照业务和功能的维度进行拆分，把数据分别放到不同的数据库中。

水平切分：水平拆分是把相同的表结构分散到不同的数据库和不同的数据表中，避免访问集中的单个数据库或者单张数据表，具体的分库和分表规则，一般是通过业务主键，进行哈希取模操作。

3、redis 的数据结构，redis 过期缓存的侧率

五种数据结构：

字符串：key-value

列表(List)：是链表-双向链表，列表类型

哈希(hash)：多个键值的映射，

集合(Set)：可以用作某些去重功能，

有序集合(zset)：有序集合，可以使用范围查找，排行榜功能或者topN功能

4、es itf 算法 倒排索引

通常是通过检索文章，逐个找到对应的关键字的为

倒排索引 通过分词，形成词和文章的关联 词典+映射表即为倒排索引

TF-IDF算法 评估一个词语在一个文件中的重要程度 如果某个词或短语在一篇文章中出现的频率TF高，并且在其他文章中很少出现，则认为此词或者短语具有很好的类别区分能力，适合用来分类

TF-IDF TF： 词语在该文件中出现的次数

IDF：可以由总文件数目除以包含该词语之文件的数目，再将得到的商取对数得到

5、jvm g1和cms 的执行过程 和区别

区别一： 使用范围不一样

CMS收集器是老年代的收集器，可以配合新生代的Serial和ParNew收集器一起使用

G1收集器收集范围是老年代和新生代。不需要结合其他收集器使用

区别二： STW的时间

CMS收集器以最小的停顿时间为目标的收集器。

G1收集器可预测垃圾回收的停顿时间（建立可预测的停顿时间模型）

区别三： 垃圾碎片

CMS收集器是使用“标记-清除”算法进行的垃圾回收，容易产生内存碎片

G1收集器使用的是“标记-整理”算法，进行了空间整合，降低了内存空间碎片

初始标记阶段：会让线程全部停止，也就是 Stop the World 状态

并发标记阶段：对所有的对象进行追踪，这个阶段最耗时。但这个阶段是和系统并发运行的，所以不会对系统运行造成影响

重新标记阶段：由于第二阶段是并发执行的，一边标记垃圾对象，一边创建新对象，老对象会变成垃圾对象。所以第三阶段也会进入 Stop the World 状态，并且重新标记，标记的是第二阶段中变动过的少数对象，所以运行速度很快

并发清理阶段： 这个阶段也是会耗费很多时间，但因为是并发运行的，所以对系统不会造成很大的影响

初始标记： 标记GC Roots 可以直接关联的对象，该阶段需要线程停顿但是耗时短

并发标记：寻找存活的对象，可以与其他程序并发执行，耗时较长

最终标记：并发标记期间用户程序会导致标记记录产生变动（好比一个阿姨一边清理垃圾，另一个人一边扔垃圾）虚拟机会将这段时间的变化记录在Remembered Set Logs 中。最终标记阶段会向Remembered Set合并并发标记阶段的变化。这个阶段需要线程停顿，也可以并发执行

筛选回收：对每个Region的回收成本进行排序，按照用户自定义的回收时间来制定回收计划

6、什么情况下应该考虑使用G1

参考官方文档：

实时数据占用超过一半的堆空间

对象分配或者晋升的速度变化大

希望消除长时间的GC停顿（超过0.5-1秒）

儒猿-石杉架构课

7、LRU 算法和缓存过期机制

<https://docs.qq.com/pdf/DY3dHR2RDTG9CUU91>

1.FIFO（First In First out）：先进先出，淘汰最先近来的页面，新进来的页面最迟被淘汰，完全符合队列。

2.LRU（Least recently used）：最近最少使用，淘汰最近不使用的页面

3.LFU（Least frequently used）：最近使用次数最少，淘汰使用次数最少的页面

8、缓存雪崩缓存穿透解决方案

缓存雪崩：和数据库不一致

缓存穿透：数据库没有，判断是否为空直接返回

缓存击穿：缓存击穿实际上是缓存雪崩的一个特例，缓存击穿是指缓存中没有但数据库中有的数据（一般是缓存时间到期），这时由于并发用户特别多，同时读缓存没读到数据，又同时去数据库去取数据，引起数据库压力瞬间增大，造成过大压力。击穿与雪崩的区别即在于击穿是对于某一特定的热点数据来说，而雪崩是全部数据。

解决方案：

1、设置key永远不过期，或者快过期时，通过另一个异步线程重新设置key

2、双重校验（Dubbo Check）类似线程安全的懒汉单例模式实现，保证只会有一个线程去访问数据库缓存

9、和数据库双写一致性方案，大促场景下的 redis 缓存超卖方案

10、RocketMQ 基本架构原理，Broker 高可用、高性能机制，raft 协议选举与 Dledger 主从同步机制

RocketMQ 的事务消息机制，能够基于具体业务场景灵活运用 RocketMQ，确保消息 0 丢失，顺序性，幂等性等

儒猿-石杉架构课

11、redis 和 zookeeper 分布式锁原理

12、ZAB 和 Paxos 算法的联系与区别？

相同点：

- (1) 两者都存在一个类似于 Leader 进程的角色，由其负责协调多个 Follower 进程的运行
- (2) Leader 进程都会等待超过半数的 Follower 做出正确的反馈后，才会将一个提案进行提交
- (3) ZAB 协议中，每个 Proposal 中都包含一个 epoch 值来代表当前的 Leader 周期，Paxos 中名字为 Ballot

不同点：

ZAB 用来构建高可用的分布式数据主备系统（Zookeeper），Paxos 是用来构建分布式一致性状态机系统。

9、彩蛋：天异云同学的采访

1、今年多大？

30，还有几月就31岁了

2、有什么兴趣爱好？为什么喜欢做这些事？

打游戏，学习，学习为了更好的突破，更好的去工作中快速上手

3、家乡是哪里？为什么决定来北京发展？怎么看待北京这座城市。

山西临汾，二线城市漂泊近3载，遍地外包，机会渺茫，来北京试水。
北京，如果是IT人士，只要肯努力，机会还是有，一座包容性很强的城市

4、以后有回家发展的规划吗？

目前没有

5、决定从之前公司离职的原因？

30岁了，试水，尝试破局，这只是开始

6、这次跳槽成功后，心里的感受是怎么样？

一个很大的进步，可以说是成功，但只是破局，但并未结束，只是开始，后边 演绎还要靠自己继续努力

7、以后还会坚持学习吗？未来的计划是怎样的？

会坚持，如果你发现老师在提问区11点，还在回答作业，或者1，2，5点，你会知道老师还在努力，你更亦如此，未来计划，重修架构课，修正自己的学习方法，不要着急，可以慢一点，必须吃透为主

8、结语

话外音：老师是一名极度低调，不在乎任何Title，并且在职业生涯有一个很好的指导方向，在架构班3年，硬生生看到课程大纲刚出，各大培训机构抄袭大纲，以最初的原始高并发为例

以下为粗略大纲，最后的实际内容预计将会达到 200-300 讲，几乎每个技术点都会使用案例展示，同时核心源码会图解剖析

- 01、多线程基础（一）：线程、进程以及上下文切换
- 02、多线程基础（一）：同步与异步、并发与并行、阻塞与非阻塞
- 03、多线程基础（二）：饥饿、饿死、死锁、资源不足、竞争
- 04、多线程基础（三）：5 种并发原则
- 05、多线程基础（四）：Amdahl 定律和 Gustafson 定律
- 06、Java 内存模型（一）：cpu 的内存模型图解
- 07、Java 内存模型（二）：缓存一致性协议的图解
- 08、Java 内存模型（三）：硬件缓存区的图解
- 09、Java 内存模型（二）：Java 的内存模型图解
- 10、Java 内存模型（二）：原子性、可见性、有序性
- 11、Java 内存模型（二）：指令重排与 happens-before 原则
- 12、Java 内存模型（三）：深入剖析指令重排的原理
- 13、Java 内存模型（四）：深入剖析顺序一致性的原理
- 14、volatile 结合 Java 内存模型的原理剖析
- 18、double-check 结合 Java 内存模型的深入分析
- 19、Thread 深入源码剖析（一）：核心原理
- 20、Thread 深入源码剖析（二）：线程状态转换
- 21、Thread 深入源码剖析（二）：sleep、join、yield、interrupt、top、wait、notify
- 22、Thread 深入源码剖析（四）：线程组、守护线程、优先级
- 23、Thread 深入源码剖析（五）：线程异常处理
- 24、案例实战：基于多线程的餐厅排队系统
- 25、案例实战：仿 windows 文件系统的搜索框
- 26、案例实战：手写支持请求超时的接口
- 27、案例实战：手写数据库连接池
- 28、案例实战：手写线程池
- 29、案例实战：手写 C/S 架构服务器
- 30、案例实战：手写千万量级数据清洗程序
- 31、案例实战：手写 TB 级大文件分析程序
- 32、多线程并发安全问题
- 33、synchronized 同步机制深入原理剖析
- 34、案例实战：手写实现分布式文件系统的 edits Log 机制

具体讲解内容以架构班课程为准 比这个更好

儒猿-石杉架构课

<https://docs.qq.com/pdf/DY3dHR2RDTG9CUU91>