

联系方式

- 手机：15915874328
- Email: xzghfat@163.com

个人信息

- 徐自刚/男/1988
- 本科/武汉工程大学信息与计算科学
- 工作年限：11年
- Github: <https://github.com/xzghfat>
- 期望职位：后台 (golang python) 高级工程师/架构师
- 期望薪资：税前月薪30k~35k，特别喜欢的公司可例外
- 期望城市：广州

华为（广州分公司）（2017年3月 ~ 2021年11月）

IoT 物联网，广东移动BOSS（业务支撑系统）

我在华为先是在广东移动boss下一个子系统营帐系统做组长（16级），三年后就换到IoT物联网部门。主要工作就是改造广东移动的后台系统支持5G业务（前一代是4G，也就是进华为是版本）；后台主要用c++, golang, python混合编程。数据库是oracle，缓存用的是华为自研的内存数据库（名字忘了，用得少）；移动整个服务接口众多有1300多个，我负责的模块实现的就有400多了，而且相互调用，逻辑复杂。很难管理，技术总监决定各子模块整理出所有接口调用，然后公共接口用kubernetes微服务框架技术栈来改造现有的模型。陆续改造一年多，公共接口最终圆满完成。管理团队，协调任务，整理资料，划分模块，分配任务和学习新的框架架构是我主要工作内容。期间改造模块一波三折，到最后顺利完成也是有很大的收获的（虽然很累）。在换到IoT物联网部门的主要工作就是通过华为的云端技术把广东移动boss系统服务配置到云端，主要是kubernates, docker和华为云等框架的配置管理部署工作。在工作当中的遇到的困难和解决办法：

1. 管理团队的问题：由于第一次管人，没有经验，很多事情没有处理好，比如分配工作啊，即时掌握进度了。分清重要和紧急的事；
2. 新技术的学习和改造：之前没有熟悉微服务架构这一当要用新的东西心里还是没有底的。部署了微服务框架后进行调试，很多东西需要自己去摸索，看技术文档。学习的过程中有些吃力，效率很是低下。后来发现高手都是在数据社区里，而且和人交流学习得较快。于是在社区提问，搜贴发现很多问题很好解决，自己问题解决了框架就清楚了。
3. 新的技术层出不穷，好多rpc, kubernates等等；
4. 最后在物联网工作的几个月，就是容器配置部署。

总结：在华为的4年多，感觉自己是有收获很多的：

1. 微服务技术从不懂到熟悉，很多东西都是新出来。golang语言的学习，框架技术栈都有在扩充；
2. 管理能力的提升，刚进来不知到怎么管理，到最后和同事合作愉快，相处融洽；
3. 用到了很多技术，学了很多东西；最后发现没有方向，而K8s让我感到是真正我想学的东西，想研究的方向；

广东邮政信息技术有限公司（2012年8月～2017年3月）

邮政代收代缴交易系统

我在这家公司负责后台新业务功能的设计和开发工作，平台的流程处理优化和统计系统架构的设计。用到的技术有：c/c++, python, oracle, redis和kafka等。在这近5年的工作中，我逐步提高了自己的编程和设计能力。调整了交易系统模块架构，把原本混乱无章的交易系统拆分成了两个大的模块：在线交易系统和统计报表系统。改进交易系统的网络IO处理速度，提高了QPS，降低了latency；报表系统弃用原先c/c++处理方法，用python更快地开发出更丰富的模块功能；数据库启用primary/secondary/backup加redis存储来改造臃肿混乱的原方案。通过读写分离和redis读写能力快的特点加速了数据库服务器的处理速度。一系列的改造，从业务模块，处理流程到数据库，使整个系统变得高效能，稳定，情绪，可扩展的和容易维护的。由于这些改进，我第三年就成为后台组的组长和不错的奖励；

其实在改造的过程中也是遇到很多挑战：

第一业务模块管理混乱，各种业务混在一在，没有一个清晰的分类；报刊，保险，交通违规等等业务和代付系统，第三方支付系统串行代码串行在一起组成一个流程。没有写成接口。即过程式的编程模式，通过细分代码抽象成接口，提供API调用，实现接口调用风格的代码；

第二业务接口改成后接着就遇到数据库io瓶颈了，有个年终充值话费活动

很吸引人流量很大，每条请求响应时间很长，我们测试后发现数据库的

读写慢，每次查询要读写表，流程处理繁琐导致数据库更新很慢。通过讨论和查资料可以对数据库进行优化，大致是对常用的不会更改的数据存到redis中，如电话号，读取和品牌信息等，不用再取数据库的了。充值之后成功先记录（号码，金额）到临时表（不加索引）中，然后等流量下去的时候再入到正式表中。改动之后就在模拟测试，发现比之间快了63%左右，等到下次活动时就没有数据库io慢的原因了；

第三每月月底或月初各下游或第三方公司都需要报表统计数据，大大小小的各式各样的报表很是心烦：需要报表的格式啊（excel，txt），内容，统计粒度啊等等真实五花八门。而且是来一个直接无脑过程式的实现，之后要改统计字段，就要改代码。这种做法既效率低下且很蠢。于是就提出把报表对账业务独立出来，做一个独立且通用的报表系统：可以自由配置字段，可以导入导出，保留查询历史记录。上网查寻发现python开源库就有这样的库而且非常简单，跟领导沟通后就采用了开源库来改的方案。于是就开始熟悉python库，之后就编写一套接口，通过配置来实现各种报表的样式了。完成了那一刻由衷的满足，再也不用担心对接人员的催促了；

其实还有一些小的优化细节略去不提；在公司中我获得的经验有：

1. 有好的编码习惯即接口是编程，复用代码，虽然大部分是c来写，但是用c也可以写接口式的代码；
2. 业务优化：异步调用，消息队列和scaling;
3. 数据库优化：读写分离，分表/分库，分布式架构；

联系方式

- 手机：15915874328
- Email：xzghfat@gmail.com

技能清单

以下均为我熟练使用的技能

- 编程语言：golang/python/c
- 通用组件：redis/Kafka/mysql/oracle
- 系统架构：高并发/分布式系统
- 容器相关：docker/Kubernetes/minikube/skaffold
- 研究方向：Kubernetes/DynamoDB

熟悉的技能：

c++/rpc/PaaS/

致谢

感谢您花时间阅读我的简历，期待能有机会和您共事。