

## 联系方式

- 手机: 15868759135
- Email: zhujingdi1998@gmail.com
- QQ/微信号: 986494553/b986494553
- Github: [xiantang](#)

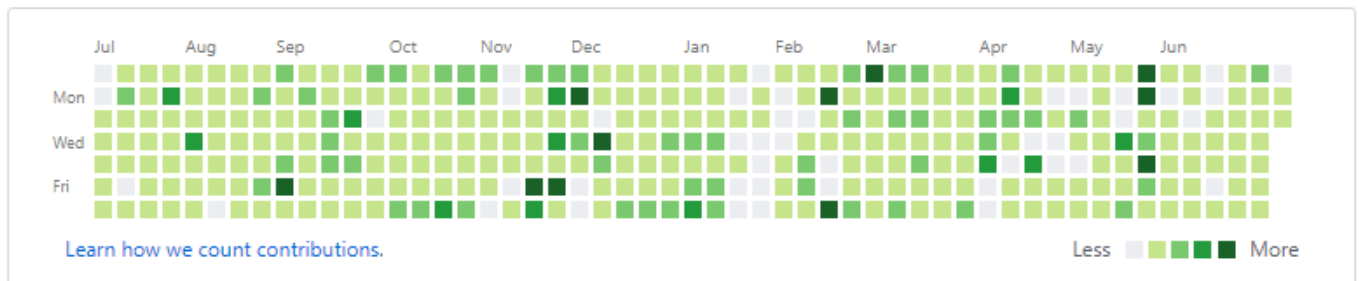
## 个人信息

- 朱静迪/男/1998
- 本科: 温州大学城市学院 2016/9 - 2020/6
- 专业: 计算科学与技术专业
- 技术博客: <http://xiantang.info>
- 期望职位: Java后端工程师/数据采集工程师
- 期望城市: 杭州/上海

## 开源项目和作品

1,381 contributions in the last year

Contribution settings ▾



## Java项目

### NIO WebServer(轮子)

- 简介:
  - 基于Java NIO 多线程、socket网络编程、XML 解析、log4j 日志的 HTTP 服务器和 Servlet 容器，并实现 热加载 功能。
- 技术细节:
  - 完成了对 HTTP 协议的部分支持，解析 POST GET 请求并且封装为自定义的 request 对象解析 url 返回对应的servlet 并封装为 HttpServletResponse，写入浏览器。
  - 使用 NIO Reactor 模型实现复用 socket 连接，并且通过反注册实现 keep-alive 长连接。
  - 实现 Context 容器以及 Wrapper 容器，并使用 Pipeline 机制使开发者能够对 HttpRequest 对象进行一系列逻辑操作。
  - 对于 NIO 中的 selector 的注册行为，使用并发注册与非阻塞轮询的方式提高性能，相较于串行方式提高了 10% 的吞吐量。
  - 使用自定义类加载器重新装载 class 文件实现热加载机制。

- 总结:
  - 使用JMeter进行压力测试: connection:close 以下测试总请求次数都为 20000 次2个线程, 每个线程循环访问10000次, 吞吐量为 630 个请求/sec。

## 京东全站商品监控系统

- 简介:
  - 基于 scrapy-redis 和 Spring Boot 的分布式价格监控平台。
- 技术细节:
  - 使用 RabbitMQ 作为消息队列, 通过 Quartz 定时扫描数据库, 发送邮件提醒用户价格更新。提高性能用redis进行缓存减少数据库访问压力。
  - 使用Redis作为缓存数据库, 加快了 50% 的加载速度。
  - 重构爬虫的 pipeline, 使用有限状态机省去了繁杂的 if-else 代码。通过阅读和修改 scrapy-redis 源码, 将原来基于 Redis set 的去重对列, 改为基于布隆过滤器的实现, 提高了去重的效率。
  - 使用 scrapyd 的api编写脚本控制爬虫的状态, 达到增量爬取的效果。
  - 使用 docker 进行爬虫的批量部署。
- 总结:
  - 爬虫部署在三台 Ubuntu 云服务器上, 并稳定爬取。
  - 只需三天就可以完成对京东的 1100w 商品数据的价格监控, 每日能够记录 3W+ 条价格更新。

## 工作以及开发经历

---

### 2019.05-至今 北京爱奇艺科技有限公司上海分公司

- 简介:
  - 使用 git 工作流参与QLB(QIYI Load Balance)的后端日常开发。
- 技术细节:
  - 在部署集群时, 使用 分布式锁 对集群进行加锁, 维持数据在分布式场景中的一致性。
  - 积极参与项目中的 code review 环节。
  - 将项目中部分串行的操作, 使用线程池并发完成, 提高了接口至少 50% 的查询速度。

## 技术文章

- [爬虫遭遇状态码521陷阱 破解js加密cookie](#) 收获 73 个赞同, 330 个收藏。

## 获得奖项

- **2018**第一届全国大数据竞赛二等奖
- **2018**年学业三等奖学金

## 致谢

---

感谢您花时间阅读我的简历, 期待能有机会和您共事。