

# 林义清

男 | 26岁 18061877017 18061877017@163.com  
5年工作经验 | Java | 期望城市: 杭州



## 教育经历

南昌航空大学科技学院 本科 软件工程

2017-2021

## 个人优势

拥有5年Java开发经验，对IT技术充满热情，乐于钻研新技术及底层原理，并积极进行技术分享（全网粉丝2万+）。

**项目经验丰富：** 深入参与跨境电商、国际物流、新零售、社交内容及用户增长等多个领域，主导或参与TMS、WMS、OMS、国际物流平台、社交内容系统、用户增长策略、精准人群圈选及AI创新应用等项目开发。熟悉订单、商品、库存、优惠券、内容、人群画像等核心业务模块。

**技术能力扎实：** 具备大型互联网项目（亿级注册用户，百万级日活）架构设计与研发经验，深入理解大型互联网后端架构。熟练掌握Java技术栈，同时具备大型C端、B端中台开发能力。

**AI领域实践：** 紧跟AI前沿技术，熟悉AIGC，掌握大模型微调(如unsloth、MLX、Llamafactory)、Langchain、Ollama、MCP、RAG等框架及多种LLM应用，具备RAG系统、Agent系统的实际落地开发经验。

**综合素质：** 具备出色的问题解决能力和快速学习能力，能够应对复杂业务场景和技术挑战，驱动业务增长和技术创新。

## 专业技能

深入理解volatile、synchronized、Lock、ConcurrentHashMap、线程池等并发编程技术；

深入理解JVM、JMM，熟悉内存模型、类加载、垃圾收集器、GC算法等、具有生产JVM调优经验；

熟练使用Redis，熟悉其数据类型、持久化、主从复制、集群、缓存一致性等，对分布式锁实现有一定的理解；

熟练使用MySQL，熟悉底层数据结构、索引、事务、日志等，具有生产慢SQL调优经验；

熟悉Zookeeper，熟悉节点类型、ZAB协议以及分布式锁实现等；

熟悉Netty，对Reactor模型、零拷贝、粘/拆包具有一定的理解；

熟悉RabbitMQ、Kafka、RocketMQ，熟悉基础架构、副本同步机制、持久化机制、零拷贝以及常见问题解决方案；

熟悉Spring，对IOC、AOP、Bean生命周期、循环依赖等有一定的理解；

熟悉SpringBoot，对常用注解、自动装配原理、Jar启动流程、自定义Starter有一定的理解；

熟悉SpringCloud，对常用组件、分布式事务具有一定的理解；

熟悉常规的设计模式，熟悉责任链、工厂、策略等，具有项目落地实践经历；

熟悉SpringBoot3、SpringAi、SpringAi Alibaba，具有大模型应用项目落地经验；

熟悉Python以及具备AIGC相关知识，熟悉unsloth、langchain、langchain4j、MCP、ollama、RAG框架(Dify、MaxKb、Ragflow、Fastgpt等)以及各LLM大语言模型应用(Deepseek、Qwen、Gpt、Minimax、Claude等)，具有RAG系统、Agent系统落地开发经验，并且实时关注AI前沿技术；

熟练使用Windsurf、Cursor、Trae、Cline等AI工具；

## 工作经历

阿里巴巴-淘天集团-天猫校园 Java高级开发

2024.04-至今

极兔速递 Java开发

2022.07-2024.02

## 项目经历

### 校园AI大模型应用    Java高级开发

2025.03-至今

**背景:** 为在核心业务稳定基础上探索新增长机遇,并提升校园用户体验,我们积极拥抱大模型技术浪潮,启动面向校园场景的AI开放平台项目“青芒AI”,致力于深度创新并落地Agent应用。

**技术:** JDK21、DDD、Pandora Boot、Spring Boot 3、ASR、HSF(RPC)、RDB(Tair)、Redisson、Diamond/Switch、MetaQ(RocketMQ)、RAG、Opensearch、MCP、DashScope, Spring Ai、Spring Ai Alibaba、unsloth等。

**职责:**

参与公司级AI创新项目“青芒AI”的规划、设计与实施;

前期基于unsloth高效微调Qwen2.5模型,结合RAG框架搭建校园本地知识库,赋能智能客服,显著提升校园客诉处理效率;目前利用阿里云百炼平台(DashScope),集成并优化多种大模型(Qwen, Minimax, Deepseek等)及微调模型,成功将AI能力应用于校园全场景,打造出创新的校园Agent应用;

主导负责SpringAi、MCP、Agent、Ai搜索等相关场景落地;

**成果:**

- 1.成功打造校园智能客服系统,有效分担客服压力,客诉处理效率实现倍数级提升;
- 2.落地AI简历优化、AI职业规划、AI拍题答疑、AI搜疑、AI论文辅导等提效工具,以及AI校内资讯搜索、同校好友推荐等功能、工具落地,驱动商业化增长点提升5%;
- 3.推出AI写真、AI发帖、AI评论、AI算命等AIGC功能,丰富“找搭子”社交场景内容,促进社区用户活跃度提升10%;
- 4.基于RAG进行检索增强(向量模型 + Opensearch向量检索),打造社区AI搜索功能;
- 5.基于MCP协议,开放学生认证、校园消费活跃度等核心校园能力,拓展生态合作;

### 校园多渠道消息触达中心    Java高级开发

2025.03-至今

**背景:** 完善特定圈选人群发放制定权益触达推送,实现精准触达用户,节省运营成本。

**技术:** JDK21、DDD、Pandora Boot、Spring Boot 2、ASR、HSF(RPC)、RDB(Tair)、Diamond/Switch、MetaQ(RocketMQ)、ODPS(离线数仓、实时计算等)、TDDL(分库分表)、Schedulerx、TOC等

**职责:**

作为多渠道消息触达中心的核心技术开发人员,主导并从零到一构建了服务于公司核心C端千万级用户的消息触达系统以及不同圈选人群数据处理等;

率先采用JDK 21进行生产环境落地实践,负责解决线上版本兼容性挑战及JVM性能调优问题;

设计并实现了系统核心模块,包括超时任务中心(TOC)、分布式调度、多渠道消息适配与推送、以及高性能消息消费等关键组件;

**成果:**

- 1.超时任务中心(TOC)设计与实现: 针对运营特定延时任务需求,自主设计并实现超时任务中心(TOC);采用分治法管理任务,结合定时扫描器与 **分布式延时内存队列** (基于集群节点感知与任务均摊策略,支持节点宕机自愈),保障了大规模延时任务的精确、稳定处理;
- 2.千万级消息调度效率提升: 针对千万级消息触达的效率瓶颈,引入Schedulerx并应用分布式调度策略(**MapReduce**)进行主子任务分发与处理,将单机调度处理能力提升N倍,显著缩短整体任务耗时(线上千万触达数据,4台4C8G机器, **链路耗时10分钟以内**);
- 3.多渠道消息推送稳定性保障: 为解决消息触达的多渠道来源(如短信、AppPush等)及异构性问题,运用适配器及责任链设计模式,构建了灵活、可扩展的消息推送框架,确保消息在不同渠道的稳定、可靠分发,将推送成功率提升至接近100%;
- 4.消息消费性能优化与可靠性增强针对MetaQ消费性能瓶颈,通过增加分区(线上增至64个)将消费TPS从2000提升至峰值6000+;并结合Sentinel对特定渠道进行精细化限流,有效解决了下游依赖的处理瓶颈,将消息消费成功率**从70%提升至100%**;

- 5.消息触达状态一致性保障：针对MapReduce分发处理中，原先批量预存触达消息记录(发送中状态)在高并发下更新易导致数据库连接不足的问题，优化存储策略，改为消息实际触达后置单条记录最终状态（成功/失败），规避了并发更新瓶颈；同时对处理链路中的异常情况(catch块)引入MQ进行异步补偿兜底，确保了消息发送状态的最终100%一致性与准确性，并且增加触达消息过期策略，优化无效数据占用存储；
- 6.生产环境JVM性能调优与内存管控：面对容器内存占用从**40%飙升至80%**的问题，利用线上监控、AppInsight及Arthas等工具进行深度JVM内存分析与调优（G1 GC），成功将千万级消息处理场景下的容器内存使用率稳定控制在80%以内（线上4台4C8G机器，处理1000万消息时内存约70%），保障了服务的持续稳定运行；

校园社交找搭子社区

Java高级开发

2024.04-至今

**背景:** 针对App用户使用粘性问题，为了刺激用户和增强用户活跃度建设的社区互动场景-找搭子。

**技术:** JDK17、DDD、Pandora Boot、Spring Boot 2、HSF(RPC)、R/LDB(Tair)、Diamond/Switch、MetaQ(RocketMQ)、ODPS(离线数仓、实时计算等)、VVP(Flink)、Opensearch(Es)、Tisplus(检索)、TDDL(分库分表)等

**职责:**

作为校园社交内容领域的核心技术人员，主导并完成了从0到1的“找搭子”核心社交玩法开发，覆盖近数十万DAU。负责底层帖子、评论、话题、Feed流等核心模型的设计与实现，以及站内营销投放模块的开发。系统在校园热门活动期间，稳定支撑日均数十万用户访问；

多次担任项目PM，负责需求协调、进度推动，在项目时间紧迫的情况下，组织团队按时、高质量完成交付；

具备丰富的线上疑难问题排查经验，成功解决过高并发场景下的数据库死锁、慢SQL查询等性能瓶颈问题；

参与以及负责AI大模型在社区提效工具及娱乐化场景的应用开发；

**成果:**

- 1.提升查询框架易用性: 针对集团Tisplus组件使用缺乏扩展性、可读性痛点，参照MyBatis-Plus语法设计并实现了Lambda风格的查询Starter组件，显著增强了代码可读性与开发效率。
- 2.高并发计数模型设计: 主导设计了校园社区帖子、评论、个人主页等多维度、大数据量场景的计数模型；运用最终一致性思想，创新性采用 **实时缓存计数 + 离线数据校准** 方案，有效提升了系统在高并发下的稳定性与用户操作流畅度；
- 3.保障缓存操作原子性: 针对Redis的多次缓存操作场景，引入Lua脚本(eval/evalsha)执行，确保了复杂操作的局部原子性，防止数据不一致；
- 4.化评论删除性能: 针对多级评论数据删除操作耗时长的问题，利用最终一致性思想，通过MetaQ进行异步化处理，并结合自旋重试机制确保子评论数据的可靠删除，将操作RT控制在100ms以内；
- 5.构建可扩展Feed流体系: 主导设计了校园社区Feed流系统。借鉴Twitter Feed流设计理念，采用**Timeline**时间轴模型及**推拉结合**模式。初期使用TDDL进行写流量预备，并规划后期迁移至集团Tablestore以应对海量数据读写挑战。
- 6.解决Feed流乱序问题: 针对关注Feed流（涉及关注、取关/拉黑、发帖、删帖等操作）的复杂时序依赖，采用MetaQ**顺序消息与顺序消费**机制，有效处理了业务逻辑的消费乱序问题，并避免了锁竞争；
- 7.实现高效PV/UV统计: 为热议话题设计PV/UV统计功能。初期采用 **实时缓存累加 + 阈值批量入库** 方案(评估并排除了HyperLogLog、BloomFilter)，后期通过预留日志接入VVP(Flink)实现更精准的实时数据统计；
- 8.构建用户推荐流: 初期利用ODPS离线数仓进行用户画像分析与挖掘，结合Opensearch实现高效的业务查询与推荐结果透出，有效提升了用户社交发现与互动活跃度；
- 9.设计实时排行榜组件: 针对帖子维度的积分(小红花)礼物排行榜业务，设计并实现了基于 RDB(zet) + 数据备份兜底的实时、多维度排行榜组件；
- 10.保障数据版本兼容性: 采用责任链设计模式处理App数据版本兼容问题，构建了灵活可扩展的数据转化层，保证了代码的可读性与可维护性；

校园会员权益、互动、营销、商业化体系

Java高级开发

2024.04-至今

**背景:** 为了提升app用户留存，通过各种游戏、互动、优质的奖励等营销手段来吸引用户参与，提升互动的转化率来达到相应的活动指标，并且更好的打造小红花体系，另外配合开学季以及活动促销场景，在玩法上需要有更多的品牌曝光，有更多的商业化诉求，基于主要流量曝光和资源位等，以及为了在app完成消费行为后可获取小红花并兑换相应的权益，小红花作为校园积分串联线上线下场景，同时培养用户对小红花的价值感知；

**技术:** JDK11、DDD、 Pandora Boot、Spring Boot 2、HSF(RPC)、R/LDB(Tair)、Redisson、Diamond/Switch、MetaQ(RocketMQ)、ODPS(离线数仓、TDDL(分库分表)、Schedulerx等;

**职责:**

参与负责小红花积签到、暴走吧同学3D游戏、福利社拼团抽奖、小红花转盘抽奖、广告/奖品任务中心、年度报告等;

参与负责广告投放、小红花**充值交易提现**、分润看板、广告任务等;

参与小红花积分体系搭建, 如小红花权益兑换(各种券、各种优惠卡、奖品等)、花花公益等;

**成果:**

- 1.针对月度签到采用RDB(**bitmap**)进行存储, 通过位运算进行签到信息的处理, 显著减少存储空间占用并提升签到操作性能;
- 2.根据实际秒杀领取峰值场景 **QPS 1000-2000**, 支持12w用户并发领取的挑战)的需求重构满签领奖逻辑, 将领取缓存记录进行持久化处理, 避免覆盖操作以及独占锁转为分段锁并且采用双重检查以及幂等唯一处理避免并发操作时重复领取奖品问题, 另外对于不同奖品类型采用策略, 使用MQ进行下游兜底处理, 提升系统稳定性以及并发度;
- 3.针对拼团抽奖利用一次性延迟任务精确触发开奖; 应用**策略、模板**方法及工厂模式, 标准化抽奖流程并灵活支持多品牌、多规则的抽奖活动配置, 提高业务迭代效率;
- 4.针对小红花抽奖采用高并发抽奖库存扣减方案, 基于RDB(decr)分段库存(多key)策略, 有效 **分散热点**, 解决单key 性能瓶颈; 结合分布式锁进行兜底校验, 确保库存扣减精确无误, 防止超卖;通过MetaQ异步更新数据库库存, 显著降低数据库写压力; 集成 Sentinel实现**精细化限流**, 保障系统在高并发冲击下的稳定性;
- 5.参与并构建多维度实时排行榜(如分润收益、活跃度、游戏排行等)利用RDB(**zset**)高效实现实时排名计算与查询; 前置维护进行**DB数据备份**, 出现问题采用定时任务恢复兜底, 确保排行榜数据的实时性、高性能与持久化稳定性;为解决随机奖励发放后的追踪与核销难题, 设计并实现了后端奖品记录系统, 通过精细化的数据模型与事件日志, 精确追踪用户在游戏内获取的各类随机奖励(小红花、虚拟/实物), 确保数据准确无误, 为运营后续核对与兑换流程提供了可靠的数据支撑, 有效闭环了游戏激励机制;
- 6.主导设计并优化多元化权益兑换系统, 解决流程复杂、对接多下游(券、发货等)及高并发积分扣减难题: 运用策略与模板模式统一处理各类奖品校验与发放逻辑; 通过乐观锁保障积分扣减原子性, 并结合唯一请求ID实现幂等性防重; 引入MQ实现异步发放与失败重试, 辅以定时任务监控与补偿机制, 确保了兑换操作的最终一致性与系统高可用性;
- 7.主导小红花积分充值提现系统设计(含支付宝绑定、**规则决策树**校验), 核心提现流程落地 **最终一致性+业务顺序消息顺序消费与TCC** 双方案, 兼顾支付安全与时效, 并提供**对账兜底**能力;

**极兔ISV直播电商平台**

2023.02-2024.02

**背景:** 对接Instagram、Facebook等在线直播平台打造的电商平台, 扩展东南亚、海外物流业务。

**技术:** Spring Boot、Spring Cloud、Apollo、Oracle、Redis、Kafka、Netty、WebSocket、XxlJob等。

**职责:**

对接直播评论下单、帖子下单、商家后台下单、支付发货。

对接下游取号生成面单, 进行任务调度实现库内操作, 对接下游进行运单轨迹生成以及日常监控、优化工作。

**成果:**

1. 针对频繁读的问题, 采用Redis缓存用于直播相关信息临时处理, 提升并发读取能力。
2. 针对服务调用处理耗时过大的问题, 采用并行处理服务调用, 调用从10s压缩到1s, 考虑到io爆炸的问题, 部分接口进行缓存、异步处理。
3. 针对消息处理量的问题, 采用Kafka异步处理消息私聊, 提升消息处理吞吐量。
4. 针对要面向多客户通讯, 采用Netty处理后台内部即时通讯。

**极兔国内外端应用**

2022.11-2024.02

**背景:** 面向客户的物流订单下单平台。

**技术:** Spring Boot、Spring Cloud、Apollo、Oracle、Ogg、Redis、Kafka/RocketMQ、XxlJob、Zookeeper、ElasticSearch等。

**职责:**

涉及各大平台小程序、官网、Web、App端应用系统维护、监控、优化工作。

参与对接拼多多、小红书、快手等电商平台下单流程，端应用日常维护。

参与东南亚项目迁移工作，以及海外端应用日常维护工作。

**成果：**

1. 针对基础数据接口频繁调用超时的问题，采用缓存优化基础数据接口调用超时，从有到无。
2. 针对消息通知存储数据量过大的问题，采用反向思维进行业务接口优化，减少DB存储压力。
3. 针对老项目下单频繁io爆炸的问题，采用缓存、定时任务进行调整，从频繁io爆炸优化到几乎无延迟。
4. 针对老项目用户表结构以及相关信息的偏差，采用Kafka、Ogg进行用户信息重处理。
5. 针对RabbitMQ无法消息堆积的问题，采用RocketMQ进行渠道轨迹改造。

## 极兔国内TMS系统

2022.07-2022.11

**背景：**主要处理国内网点集散货、司机打卡、报表等的平台。

**技术：**Spring Boot、Spring Cloud、Apollo、Oracle、MySQL、Ogg、Canal、Redis、MQ、XxlJob、Zookeeper、ElasticSearch等。

**职责：**主要参与TMS系统网点支干线相关等业务迭代、报表、监控、优化工作

**成果：**

1. 针对复杂维度业务，采用设计模式处理复杂业务场景，如责任链处理多维度数据等。
2. 针对大数据的亿级数据，采用Kafka多分区多消费端处理数据推送。

## 亿品跨境电商系统

2021.06-2022.02

**背景：**对接各大跨境电商平台进行自动化卖商品的平台。

**技术：**Spring Boot、Spring Cloud Alibaba、MySQL、Canal、Redis、MQ、XxlJob、ElasticSearch等。

**职责：**

参与TMS、WMS、OMS系统维护。

参与ERP、OMS平台项目重构工作，进行系统架构升级。

主导1688、Lazada、Shopee等跨境电商平台类目处理以及参与商品订单数据同步任务。

参与日常慢SQL、接口性能调优。