

目录

第 1 章 高可用架构案例精选	1
郭斯杰 / 1.1 Twitter 高性能分布式日志系统架构解析	1
1.1.1 为什么需要分布式日志	1
1.1.2 Twitter 如何考虑这个问题	4
1.1.3 基于 Apache BookKeeper 构建 DistributeLog	5
1.1.4 DistributeLog 案例分享	13
1.1.5 疑问与解惑	13
颜国平 / 1.2 腾讯基于用户画像大数据的电商防刷架构	16
1.2.1 背景介绍	16
1.2.2 黑产现状介绍	16
1.2.3 腾讯内部防刷架构	18
1.2.4 腾讯大数据收集维度	20
1.2.5 腾讯大数据处理平台——魔方	21
1.2.6 疑问与解惑	24
王渊命 / 1.3 如何设计类似微信的多终端数据同步协议：Grouk 实践分享	26
1.3.1 移动互联网时代多终端数据同步面临的挑战	26
1.3.2 多终端数据同步与传统消息投递协议的差异	27
1.3.3 Grouk 在多终端数据同步协议上的探索实践	28
1.3.4 疑问与解惑	32
周 洋 / 1.4 如何实现支持数亿用户的长连消息系统：Golang 高并发案例	33
1.4.1 关于 push 系统对比与性能指标的讨论	33
1.4.2 消息系统架构介绍	35
1.4.3 哪些因素决定推送系统的效果	37

1.4.4	GO 语言开发问题与解决方案	38
1.4.5	消息系统的运维及测试	41
1.4.6	疑问与解惑	42
唐福林 / 1.5	雪球在股市风暴下的高可用架构改造分享	46
1.5.1	雪球公司的介绍	46
1.5.2	雪球当前总体架构	47
1.5.3	雪球架构优化历程	48
1.5.4	关于架构优化的总结和感想	53
1.5.5	疑问与解惑	54
麦俊生 / 1.6	亿级短视频社交美拍架构实战	59
1.6.1	短视频市场的发展	59
1.6.2	美拍的发展	60
1.6.3	短视频所面临的架构问题	61
1.6.4	为支持亿级用户，美拍架构所做的一些改进	62
1.6.5	后续发展	68
刘道儒 / 1.7	微博“异地多活”部署经验谈	69
1.7.1	微博异地多活建设历程	69
1.7.2	微博异地多活面临的挑战	70
1.7.3	异地多活的最佳实践	73
1.7.4	异地多活的新方向	74
孙宇聪 / 1.8	来自 Google 的高可用架构理念与实践	75
1.8.1	决定可用性的两大因素	76
1.8.2	高可用性方案	77
1.8.3	可用性 7 级图表	80
1.8.4	疑问与解惑	81
那 谁 / 1.9	深入理解同步 / 异步与阻塞 / 非阻塞区别	84
1.9.1	同步与异步	84
1.9.2	阻塞与非阻塞	85
1.9.3	与多路复用 I/O 的联系	86
第 2 章	高可用架构原理与分布式实践	88
黄东旭 / 2.1	Codis 作者细说分布式 Redis 架构设计	88
2.1.1	Redis、Redis Cluster 和 Codis	88

2.1.2	我们更爱一致性	90
2.1.3	Codis 在生产环境中的使用经验和坑	91
2.1.4	分布式数据库和分布式架构	94
2.1.5	疑问与解惑	95
霍泰稳 / 2.2	给你介绍一个不一样的硅谷	98
2.2.1	Uber	98
2.2.2	Coursera	99
2.2.3	Airbnb	102
2.2.4	硅谷行带给我的一些影响	106
2.2.5	疑问与解惑	106
金自翔 / 2.3	解耦的艺术——大型互联网业务系统的插件化改造	110
2.3.1	插件化	110
2.3.2	如何处理用户交互	115
2.3.3	如何处理数据	115
2.3.4	总结	116
沈 剑 / 2.4	从零开始搭建高可用 IM 系统	117
2.4.1	什么是 IM	117
2.4.2	协议设计	118
2.4.3	WEB 聊天室	122
2.4.4	IM 典型业务场景	126
2.4.5	疑问与解惑	126
陈宗志 / 2.5	360 分布式存储系统 Bada 的架构设计和应用	129
2.5.1	主要应用场景	129
2.5.2	整体架构	130
2.5.3	主要模块	131
2.5.4	数据分布策略	132
2.5.5	请求流程	133
2.5.6	多机房架构	134
2.5.7	FAQ	138
2.5.8	疑问与解惑	139
张 亮 / 2.6	新一代分布式任务调度框架：当当 Elastic-Job 开源项目 的 10 项特性	143

2.6.1 为什么需要作业（定时任务）	143
2.6.2 当当之前使用的作业系统	144
2.6.3 Elastic-Job 的来历	144
2.6.4 Elastic-Job 包含的功能	145
2.6.5 Elastic-Job 的部署和使用	146
2.6.6 对开源产品的开发理念	147
2.6.7 未来展望	148
2.6.8 疑问与解惑	149
付海军 / 2.7 互联网 DSP 广告系统架构及关键技术解析	152
2.7.1 优秀 DSP 系统的特点	152
2.7.2 程序化购买的特点	153
2.7.3 在线广告的核心问题	156
2.7.4 在线广告的挑战	156
2.7.5 DSP 系统架构	157
2.7.6 RTB 投放引擎的架构	158
2.7.7 DMP	160
2.7.8 广告系统 DMP 数据处理的架构	160
2.7.9 用户画像的方法	162
2.7.10 广告行业的反作弊	165
2.7.11 P2P 流量互刷	166
2.7.12 CPS 引流作弊	167
2.7.13 疑问与解惑	168
王卫华 / 2.8 亿级规模的 Elasticsearch 优化实战	170
2.8.1 索引性能（Index Performance）	170
2.8.2 查询性能（Query Performance）	171
2.8.3 其他	173
2.8.4 疑问与解惑	174
杨卫华 / 2.9 微博分布式存储考试题：案例讲解及作业精选	179
2.9.1 访问场景	179
2.9.2 设计	180
2.9.3 sharding 策略	180
2.9.4 案例精选	181

李 凯 / 2.10 架构师需要了解的 Paxos 原理、历程及实战	184
2.10.1 数据库高可用性难题	184
2.10.2 Paxos 协议简单回顾	185
2.10.3 Basic Paxos 同步日志的理论模型	186
2.10.4 Multi Paxos 的实际应用	187
2.10.5 依赖时钟误差的变种 Paxos 选主协议简单分析	190
2.10.6 疑问与解惑	191
温 铭 / 2.11 OpenResty 的现在和未来	193
2.11.1 OpenResty 是什么，适合什么场景下使用	193
2.11.2 某安全公司服务端技术选型的标准	194
2.11.3 如何在项目中引入新技术	196
2.11.4 如何入门以及学习的正确方法	197
2.11.5 OpenResty 中的测试和调试	199
2.11.6 NginScript 是否会替代 OpenResty	201
2.11.7 未来重点解决的问题和新增特性	202
2.11.8 开源社区建设	203
2.11.9 疑问与解惑	203
第 3 章 电商架构热点专题	205
张升涛 / 3.1 亿级商品详情页架构演进技术解密	205
3.1.1 商品详情页	205
3.1.2 商品详情页发展史	209
3.1.3 遇到的一些问题和解决方案	220
3.1.4 总结	228
3.1.5 疑问与解惑	229
杨 超 / 3.2 大促系统全流量压测及稳定性保证——京东交易架构	232
3.2.1 交易系统的三个阶段	232
3.2.2 交易系统的三层结构	233
3.2.3 交易系统的访问特征	234
3.2.4 应对大促的第 1 步：全链路全流量线上压测	234
3.2.5 应对大促的第 2 步：根据压力表现进行调优	237
3.2.6 异步和异构	240

3.2.7	应对大促的第3步：分流与限流	242
3.2.8	应对大促的第4步：容灾降级	244
3.2.9	应对大促的第5步：完善监控	245
3.2.10	疑问与解惑	246
吕毅 / 3.3	秒杀系统架构解密与防刷设计	248
3.3.1	抢购业务介绍	248
3.3.2	具体抢购项目中的设计	249
3.3.3	如何解耦前后端压力	250
3.3.4	如何保证商品库的库存可靠	252
3.3.5	如何与第三方多方对账	254
3.3.6	项目总结	255
3.3.7	疑问与解惑	255
王富平 / 3.4	Lambda 架构与推荐在电商网站实践	257
3.4.1	Lambda 架构	257
3.4.2	1号店推荐系统实践	260
3.4.3	Lambda 的未来	262
3.4.4	思考	263
3.4.5	疑问与解惑	263
杨硕 / 3.5	某公司线上真实流量压测工具构建	265
3.5.1	为什么要开发一个通用的压测工具	265
3.5.2	常见的压测工具	266
3.5.3	构建自己的压测工具	266
3.5.4	疑问与解惑	271
第4章	容器与云计算	273
陈飞 / 4.1	微博基于 Docker 容器的混合云迁移实战	273
4.1.1	为什么要采用混合云的架构	273
4.1.2	跨云的资源管理与调度	275
4.1.3	容器的编排与服务发现	278
4.1.4	混合云监控体系	284
4.1.5	前进路上遇到的那些坑	286
4.1.6	疑问与解惑	286

高 磊 / 4.2 互联网金融创业公司 Docker 实践	287
4.2.1 背景介绍	287
4.2.2 容器选型	287
4.2.3 应用迁移	288
4.2.4 弹性扩容	291
4.2.5 未来规划	295
4.2.6 疑问与解惑	295
高永超 / 4.3 使用开源 Calico 构建 Docker 多租户网络	297
4.3.1 PaaS 平台的网络需求	297
4.3.2 使用 Calico 实现 Docker 的跨服务器通讯	298
4.3.3 利用 Profile 实现 ACL	301
4.3.4 性能测试	306
4.3.5 Calico 的发展	308
4.3.6 疑问与解惑	309
彭哲夫 / 4.4 解析 Docker 在芒果 TV 的实践之路	310
4.4.1 豆瓣时期	310
4.4.2 芒果 TV 的 Nebulium Engine	311
4.4.3 Project Eru	312
4.4.4 细节	313
4.4.5 网络	314
4.4.6 存储	315
4.4.7 Scale	316
4.4.8 资源分配和集群调度	316
4.4.9 服务发现和安全	317
4.4.10 实例	317
4.4.11 总结	318
4.4.12 疑问与解惑	318
王关胜 / 4.5 微博基于 Docker 的混合云平台设计与实践	323
4.5.1 微博的业务场景及混合云背景	323
4.5.2 三大基础设施助力微博混合云	326
4.5.3 微博混合云 DCP 系统设计核心：自动化、弹性调度	328
4.5.4 引入阿里云作为第 3 机房，实现弹性调度架构	330

4.5.5	大规模集群操作自动化	331
4.5.6	不怕峰值事件	332
第 5 章	运维保障	333
王 康 / 5.1	360 如何用 QConf 搞定两万以上服务器的配置管理	333
5.1.1	设计初衷	333
5.1.2	整体认识	334
5.1.3	架构介绍	335
5.1.4	QConf 服务端	336
5.1.5	QConf 客户端	336
5.1.6	QConf 管理端	340
5.1.7	其他	341
5.1.8	疑问与解惑	343
尤 勇 / 5.2	深度剖析开源分布式监控 CAT	347
5.2.1	背景介绍	347
5.2.2	整体设计	348
5.2.3	客户端设计	349
5.2.4	服务端设计	352
5.2.5	总结感悟	357
杨尚刚 / 5.3	单表 60 亿记录等大数据场景的 MySQL 优化和运维之道	359
5.3.1	前言	359
5.3.2	数据库开发规范	360
5.3.3	数据库运维规范	363
5.3.4	性能优化	368
5.3.5	疑问与解惑	375
秦 迪 / 5.4	微博在大规模、高负载系统问题排查方法	379
5.4.1	背景	379
5.4.2	排查方法及线索	379
5.4.3	总结	384
5.4.4	疑问与解惑	385
秦 迪 / 5.5	系统运维之为什么每个团队存在大量烂代码	387
5.5.1	写烂代码很容易	387

5.5.2	烂代码终究是烂代码	388
5.5.3	重构不是万能药	392
5.5.4	写好代码很难	393
5.5.5	悲观的结语	394
秦迪 / 5.6	系统运维之评价代码优劣的方法	395
5.6.1	什么是好代码	395
5.6.2	结语	403
5.6.3	参考阅读	403
秦迪 / 5.7	系统运维之如何应对烂代码	404
5.7.1	改善可维护性	404
5.7.2	改善性能与健壮性	409
5.7.3	改善生存环境	412
5.7.4	个人感想	414
第 6 章	大数据与数据库	415
王劲 / 6.1	某音乐公司的大数据实践	415
6.1.1	什么是大数据	415
6.1.2	某音乐公司大数据技术架构	418
6.1.3	在大数据平台重构过程中踩过的坑	425
6.1.4	后续的持续改进	430
王新春 / 6.2	实时计算在点评	431
6.2.1	实时计算在点评的使用场景	431
6.2.2	实时计算在业界的使用场景	432
6.2.3	点评如何构建实时计算平台	433
6.2.4	Storm 基础知识简单介绍	434
6.2.5	如何保证业务运行的可靠性	436
6.2.6	Storm 使用经验分享	438
6.2.7	关于计算框架的后续想法	442
6.2.8	疑问与解惑	442
王卫华 / 6.3	百姓网 Elasticsearch 2.x 升级之路	446
6.3.1	Elasticsearch 2.x 变化	446
6.3.2	升级之路	448

6.3.3	优化或建议	451
6.3.4	百姓之道	452
6.3.5	后话: Elasticsearch 5.0	453
6.3.6	升级 2.x 版本成功, 5.x 版本还会远吗	454
董西成 张虔熙 / 6.4	Hadoop、HBase 年度回顾	457
6.4.1	Hadoop 2015 技术发展	457
6.4.2	HBase 2015 年技术发展	460
6.4.3	疑问与解惑	466
常 雷 / 6.5	解密 Apache HAWQ——功能强大的 SQL-on-Hadoop 引擎	469
6.5.1	HAWQ 基本介绍	469
6.5.2	Apache HAWQ 系统架构	472
6.5.3	HAWQ 中短期规划	479
6.5.4	贡献到 Apache HAWQ 社区	479
6.5.5	疑问与解惑	480
萧少聪 / 6.6	PostgreSQL HA 高可用架构实战	482
6.6.1	PostgreSQL 背景介绍	482
6.6.2	在 PostgreSQL 下如何实现数据复制技术的 HA 高可用集群	483
6.6.3	Corosync+Pacemaker MS 模式介绍	484
6.6.4	Corosync+Pacemaker M/S 环境配置	485
6.6.5	Corosync+Pacemaker HA 基础配置	488
6.6.5	PostgreSQL Sync 模式当前的问题	492
6.6.6	疑问与解惑	492
王晶昱 / 6.7	从 NoSQL 历史看未来	495
6.7.1	前言	495
6.7.2	1970 年: We have no SQL	496
6.7.3	1980 年: Know SQL	497
6.7.4	2000 年: No SQL	502
6.7.5	2005 年: 不仅仅是 SQL	504
6.7.6	2013 年: No, SQL	505
6.7.7	阿里的技术选择	505
6.7.8	疑问与解惑	506
杨尚刚 / 6.8	MySQL 5.7 新特性大全和未来展望	508

6.8.1	提高运维效率的特性	508
6.8.2	优化器 Server 层改进	511
6.8.3	InnoDB 层优化	513
6.8.4	未来发展	517
6.8.5	运维经验总结	518
6.8.6	疑问与解惑	519
谭政 / 6.9	大数据盘点之 Spark 篇	521
6.9.1	Spark 的特性以及功能	521
6.9.2	Spark 在 Hulu 的实践	525
6.9.3	Spark 未来的发展趋势	528
6.9.4	参考文章	530
6.9.5	疑问与解惑	530
萧少聪 / 6.10	从 Postgres 95 到 PostgreSQL 9.5: 新版亮眼特性	532
6.10.1	Postgres 95 介绍	532
6.10.2	PostgreSQL 版本发展历史	533
6.10.3	PostgreSQL 9.5 的亮眼特性	534
6.10.4	PostgreSQL 还可以做什么	544
6.10.5	疑问与解惑	547
毕洪宇 / 6.11	MongoDB 2015 回顾: 全新里程碑式的 WiredTiger 存储引擎	551
6.11.1	存储引擎的发展	551
6.11.2	复制集改进	555
6.11.3	自动分片机制	556
6.11.4	其他新特性介绍	556
6.11.5	疑问与解惑	558
王晓伟 / 6.12	基于 Xapian 的垂直搜索引擎的构建分析	561
6.12.1	垂直搜索的应用场景	561
6.12.2	技术选型	563
6.12.3	垂直搜索的引擎架构	564
6.12.4	垂直搜索技术和业务细节	566
6.12.5	疑问与解惑	568

第 7 章 安全与网工	572
郭 伟 / 7.1 揭秘 DDoS 防护——腾讯云大禹系统	572
7.1.1 有关 DDoS 简介的问答	574
7.1.2 有关大禹系统简介的问答	575
7.1.3 有关大禹系统硬件防护能力的问答	576
7.1.4 有关算法设计的问答	577
7.1.5 大禹和其他产品、技术的区别	578
冯 磊 赵星宇 / 7.2 App 域名劫持之 DNS 高可用——开源版 HttpDNS 方案详解	580
7.2.1 HttpDNSLib 库组成	581
7.2.2 HttpDNS 交互流程	582
7.2.3 代码结构	583
7.2.4 开发过程中的一些问题及应对	586
7.2.5 疑问与解惑	593
马 涛 / 7.3 CDN 对流媒体和应用分发的支持及优化	595
7.3.1 CDN 系统工作原理	595
7.3.2 网络分发过程中 ISP 的影响	602
7.3.3 防盗链	603
7.3.4 内容分发系统的问题和应对思路	604
7.3.5 P2P 穿墙打洞	607
7.3.6 疑问与解惑	609
马 涛 / 7.4 HTTPS 环境使用第三方 CDN 的证书难题与最佳实践	611
蒋海滔 / 7.5 互联网主要安全威胁分析及应对方案	613
7.5.1 互联网 Web 应用面临的主要威胁	613
7.5.2 威胁应对方案	616
7.5.3 疑问与解惑	624