www.qconferences.com
www.qconbeijing.com
www.qconshanghai.com

伦敦 | 北京 | 东京 | 纽约 | 圣保罗 | 上海 | 旧金山

London · Beijing · Tokyo · New York · Sao Paulo · Shanghai · San Francisco

QCon全球软件开发大会

International Software Development Conference

www.infoq.com/cn

InfoQueue

@InfoQ infoqchina

软件
正在改变世界！

京东大规模内存存储平台

刘海锋

JD.京东 COM

京东自研分布式存储产品体系

一年里所做的努力

- ✓ 京东文件系统
- ✓ 对象存储
- ✓ 弹性块存储
- ✓ 高速NoSQL服务

自主研发

- 需求驱动，分期开展，高度定制，互相复用

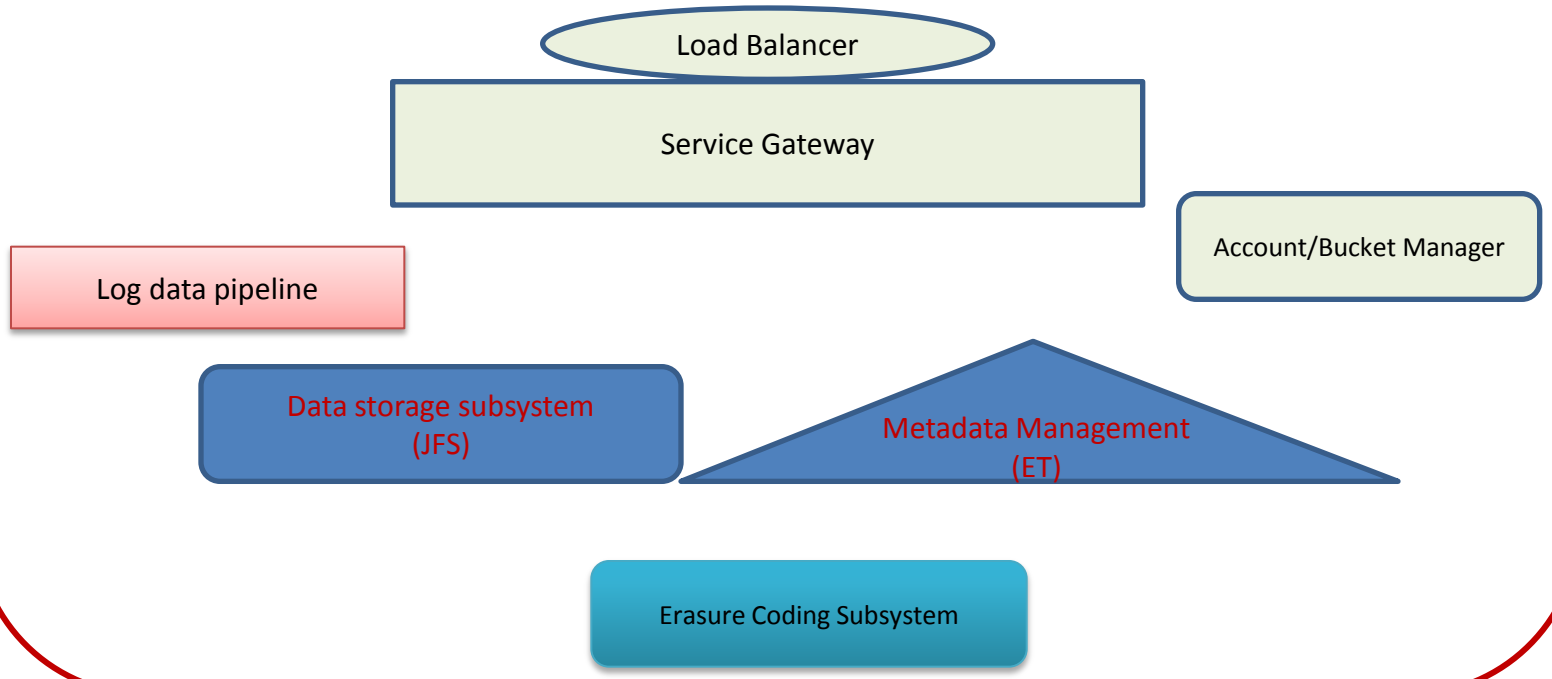
Jingdong File System

- 海量小文件
  - ✓ Paxos变体复制协议
- 大文件
  - ✓ Chain Replication
- 分布式名字空间
  - ✓ 基于RDBMS，平滑扩展

负责交易订单、商品图片、仓库流水等核心数据存储

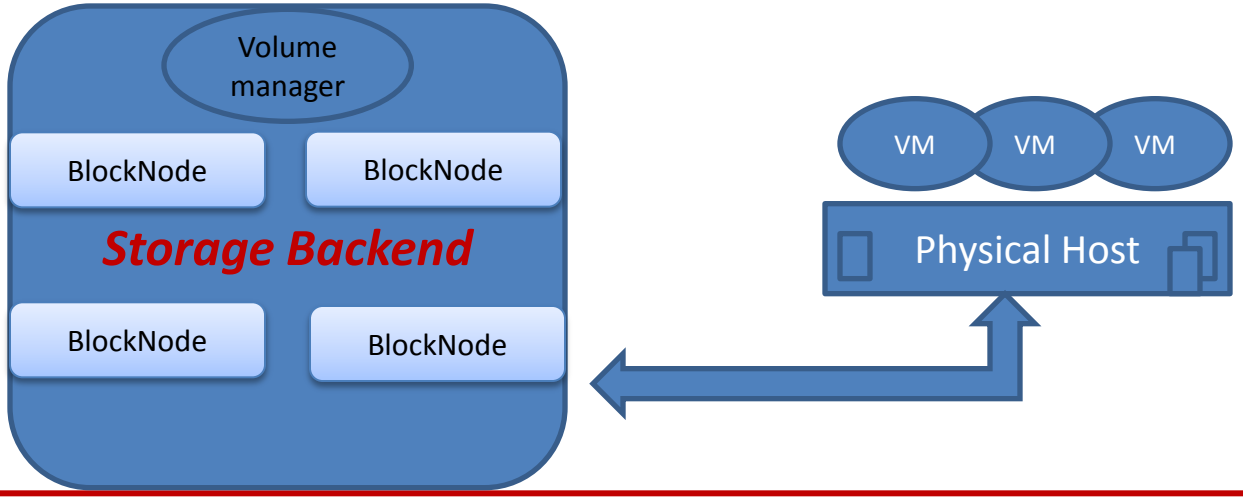
Jingdong Object Storage

S3兼容，服务公司众多业务，PB规模



Jingdong Block Storage

- 持久、无故障的“云硬盘”抽象
  - ✓ 后端实现复制和高可用
  - ✓ 主要应用于虚拟化平台
- 技术挑战
  - ✓ 故障切换与恢复、性能调优



Jingdong In-Memory Store

What is Jimstore

- ✓ 在之前Redis缓存平台基础上进一步创新
- ✓ 兼容Redis协议的高速NoSQL服务
- ✓ 综合利用RAM、SSD、HD多级存储层次
- ✓ 性能*规模/成本 = 软件研发*硬件投入*运维手段

“Memory is the new disk.”

– Jim Gray

在线数据常驻内存

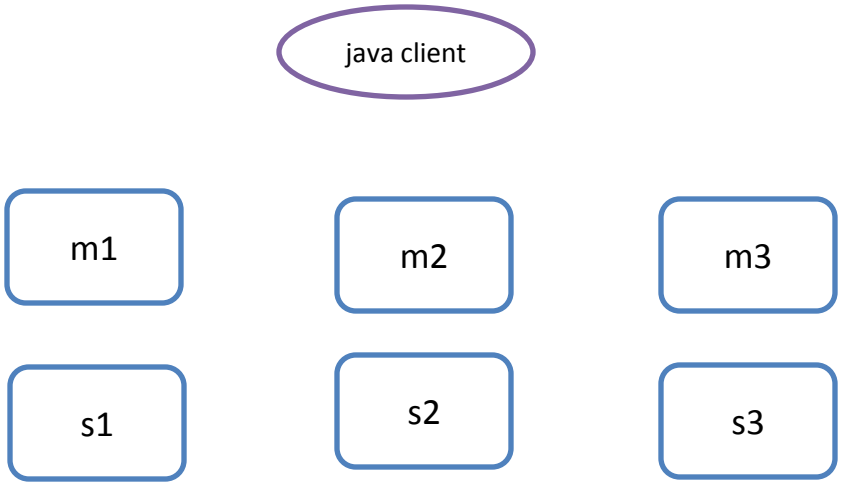
- 商品价格
- 购物车
- 配送
- 推荐
- Social
- …

演进历史

- 短平快
  - ✓ 分散管理的小集群
- 统一平台
  - ✓ 服务化、自动化
- 自主研发
  - ✓ 痛点驱动，持续开展

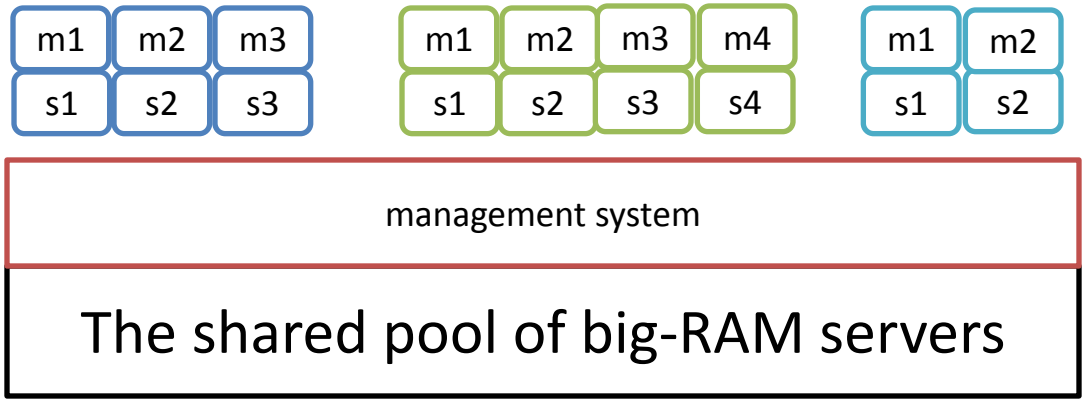
演进历史 – 短平快

- Redis单实例
- Redis主从复制
- 客户端pre-sharding



演进历史 – 集中服务化

- 统一缓存平台
  - ✓ 几百个大小不一的集群，几千个Redis实例，几百台大内存机器
  - ✓ 日益完善的监控管理系统与服务流程



演进历史 – 痛点

- ✓ 内存超标
- ✓ 数据持久性不够
- ✓ 启动慢
- ✓ 故障检测不准
- ✓ 手动主从切换
- ✓ 集群难以扩展
- ✓ …

重新思考

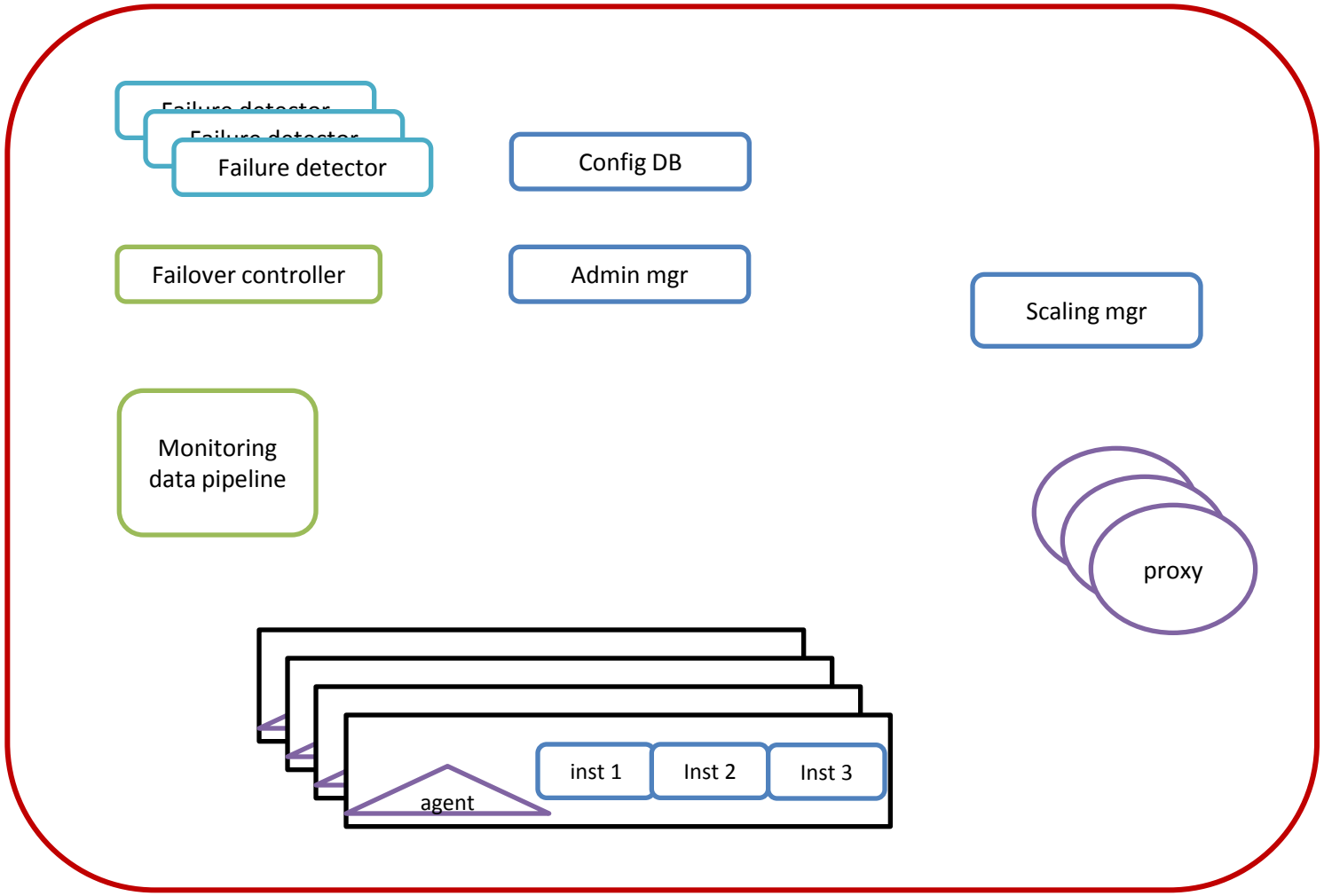
业务需要更高质量的NoSQL服务

- ✓ 使用方便、数据类型丰富
- ✓ 不同的服务质量需求
- ✓ 全自动管理与故障处理
- ✓ 平滑扩容
- ✓ 引入SSD降低整体成本

JimStore - 核心特性

- Redis协议兼容
- 自动的故障检测与切换
  - ✓ 分布式选举，auto failover
- 多租户公平性保证
  - ✓ 流量控制
- 定制存储引擎
  - ✓ 综合利用内存、SSD、磁盘
- 数据无损的平滑扩容
  - ✓ Our very own ‘Redis Cluster’

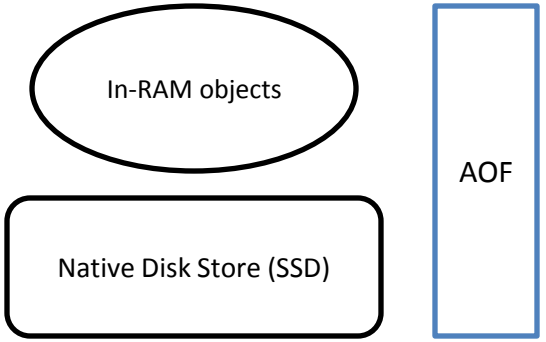
JimStore Components



JimStore – 默认存储引擎

- RAM-SSD二级存储，读优化
  - ✓ 热数据驻留内存，SSD存储全量
  - ✓ mmaped copy-on-write B+Tree
  - ✓ 延迟与QPS接近原生Redis

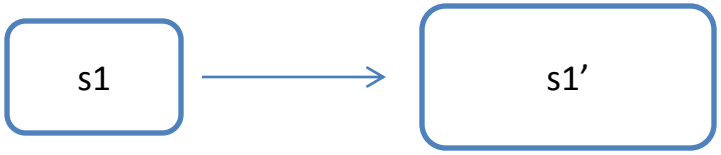
- 收益
  - ✓ 单实例容量
  - ✓ 数据持久性
  - ✓ 启动速度



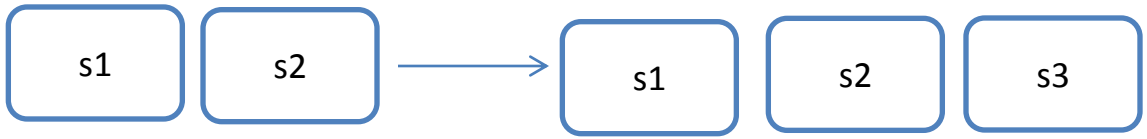
JimStore – 容量扩展

两类扩容模式

- 纵向扩容：分片数不变，增加单分片容量



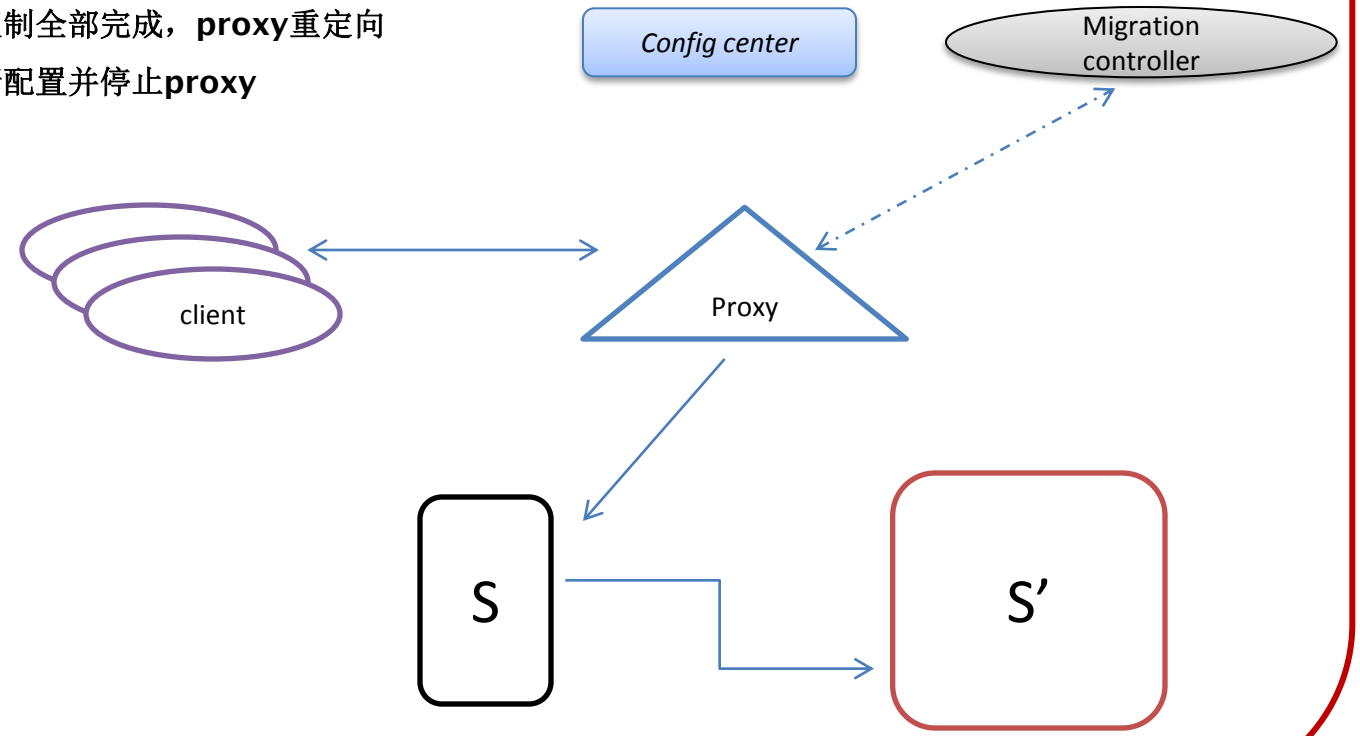
- 横向扩容：灵活增加分片数目
  - ✓ 并不要求数目翻倍



JimStore – 纵向扩容

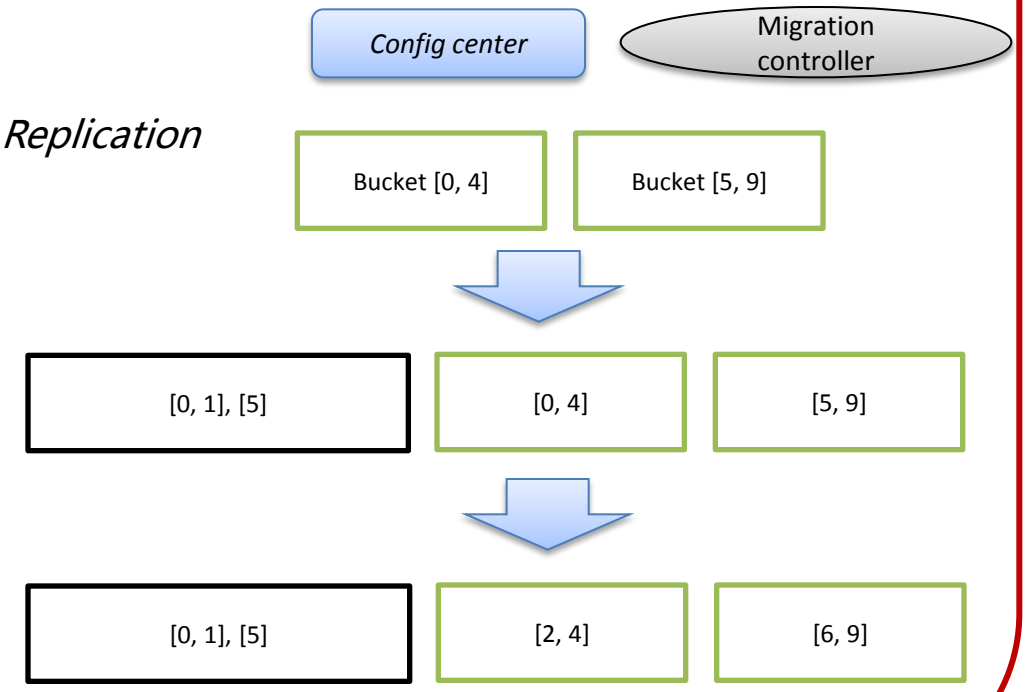
Steps

- ✓ 启动**proxy**并更新配置
- ✓ 待复制全部完成，**proxy**重定向
- ✓ 更新配置并停止**proxy**



JimStore – 横向扩容

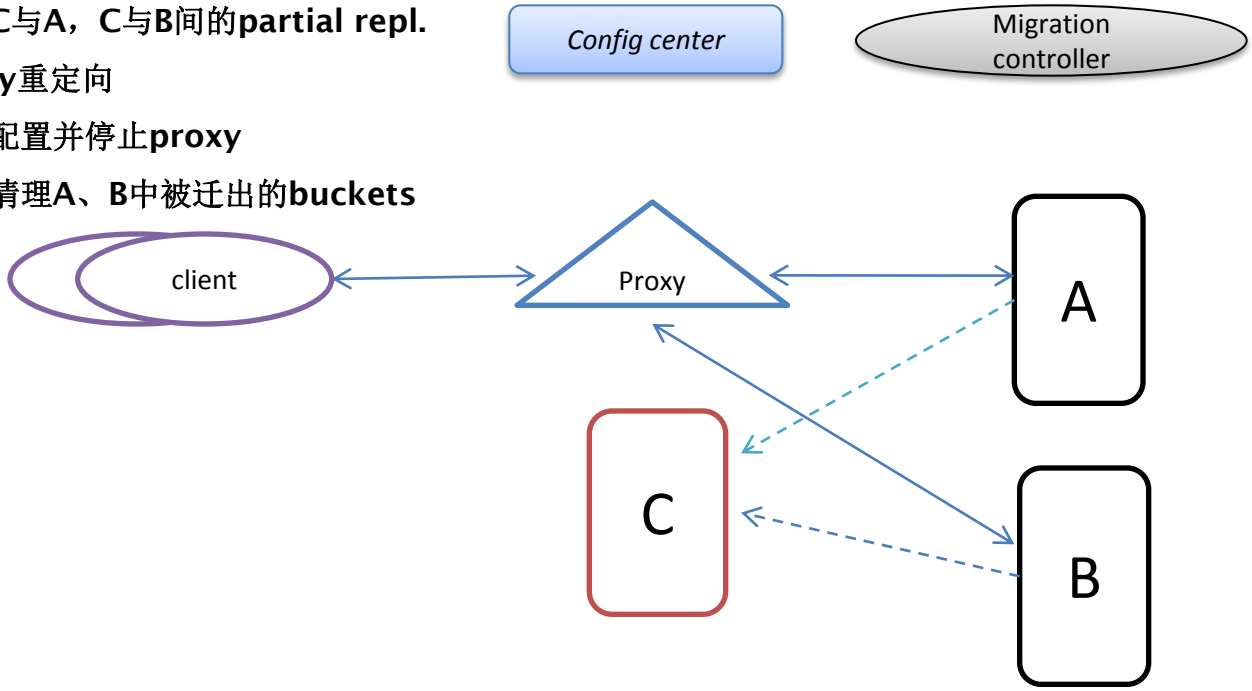
- 新增抽象
  - ✓ db, shards, *buckets*, keys
- 针对扩容的复制协议
  - ✓ *Bucket-level Selective Replication*



JimStore – 横向扩容

Steps

- ✓ 启动**proxy**并更新配置
- ✓ 并行**C**与**A**，**C**与**B**间的**partial repl.**
- ✓ **proxy**重定向
- ✓ 更新配置并停止**proxy**
- ✓ 离线清理**A**、**B**中被迁出的**buckets**



总结

愿景

- ✓ Leveraging RAM and SSD, we are building a fully-managed, infinitely scalable, highly available, fast NoSQL service compatible with Redis

策略

- ✓ 围绕业务需求，分期开展

挑战

- ✓ 软件质量保证与运维管理

Thank You.

And we are hiring☺

JD.京东 COM

InfoQ
Enterprise Software Development Community

特别感谢合作伙伴

InfoQ
Enterprise Software Development Community

特别感谢媒体伙伴（部分）