WOTA

51CTO

World Of Tech 2017

全球架构与运维技术峰会

2017年4月14日-15日　北京富力万丽酒店

ARCHITECTURE

WOTA

51CTO

出品人及主持人：

吕　毅

链家网架构师

大数据平台团队负责人

大数据系统架构

HBase in Alibaba Search

WOTA

51CTO

绝顶

阿里巴巴
搜索事业部高级技术专家

分享主题：

HBase in Alibaba Search

内容提要

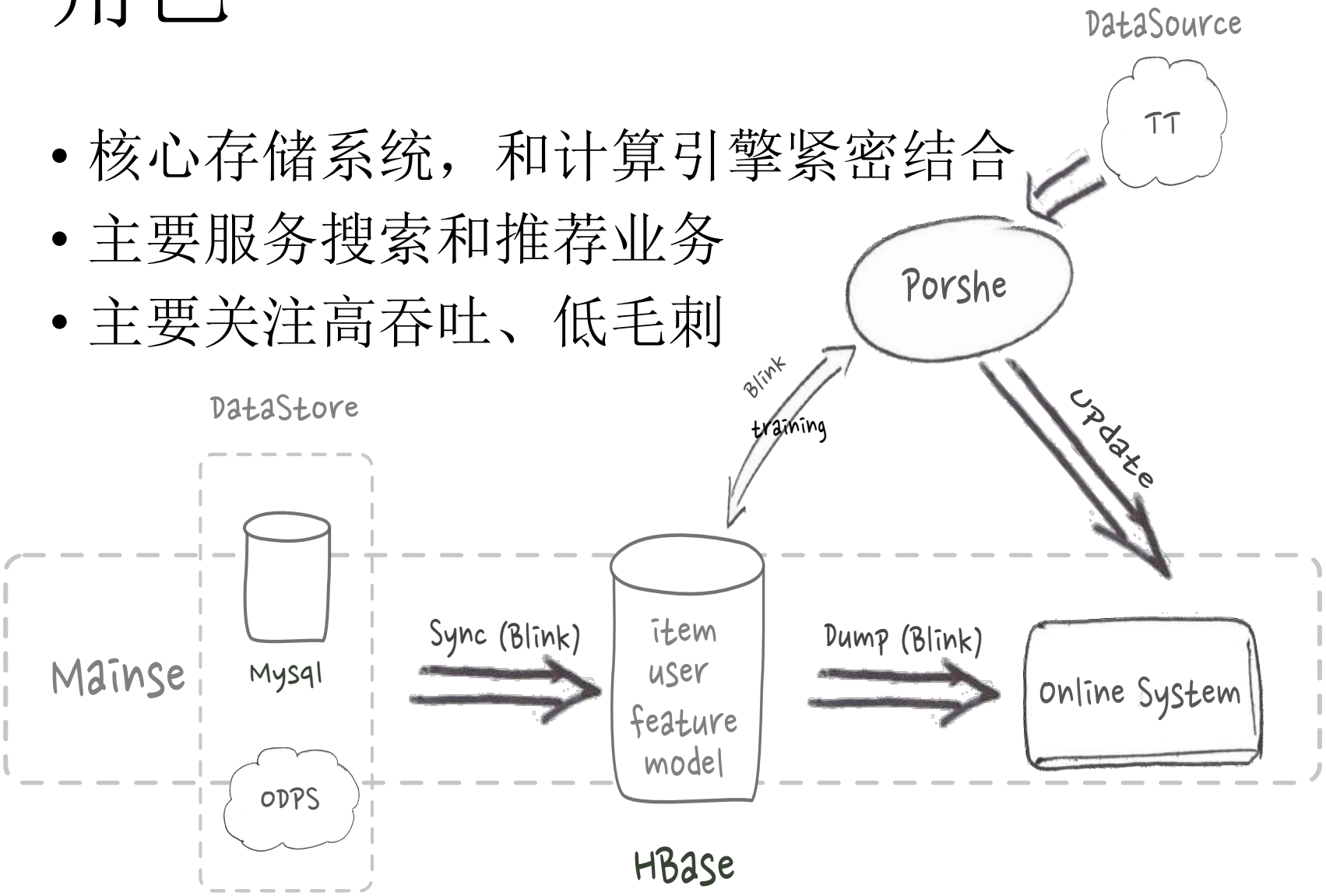
- HBase在阿里搜索的历史和规模
- HBase在阿里搜索的角色和主要应用场景
- 问题和优化
  - RPC的瓶颈和优化
  - 异步与吞吐
  - GC与毛刺
  - IO隔离与优化
- 开源&未来
- QA

历史和规模

- 历史
  - 2010年至今，历经10+个版本
    - 2010~2014: 0.20.6->0.90.3->0.92.1->0.94.1->0.94.2->0.94.5
    - 2014~2015: 0.94->0.98.1->0.98.4->0.98.8->0.98.12
    - 2016: 0.98.12->1.1.2
- 集群规模
  - 总节点数3000+，最大集群节点数1500+
- 服务能力
  - 双十一吞吐峰值
    - 集群超过4000万次 / 秒，单机达到10万次 / 秒
  - 单cpu core可支撑8000+ QPS

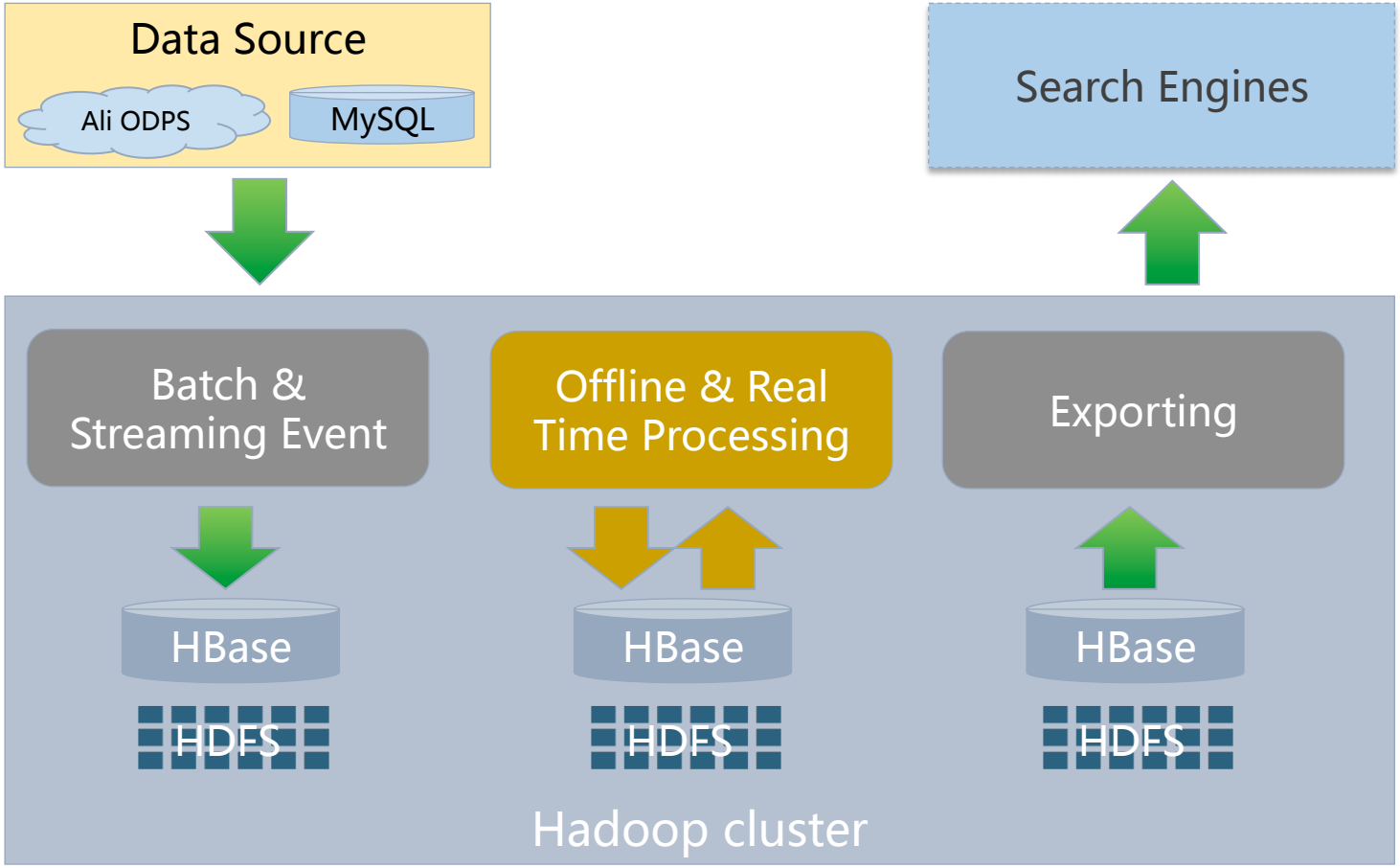
角色

- 核心存储系统，和计算引擎紧密结合
- 主要服务搜索和推荐业务
- 主要关注高吞吐、低毛刺



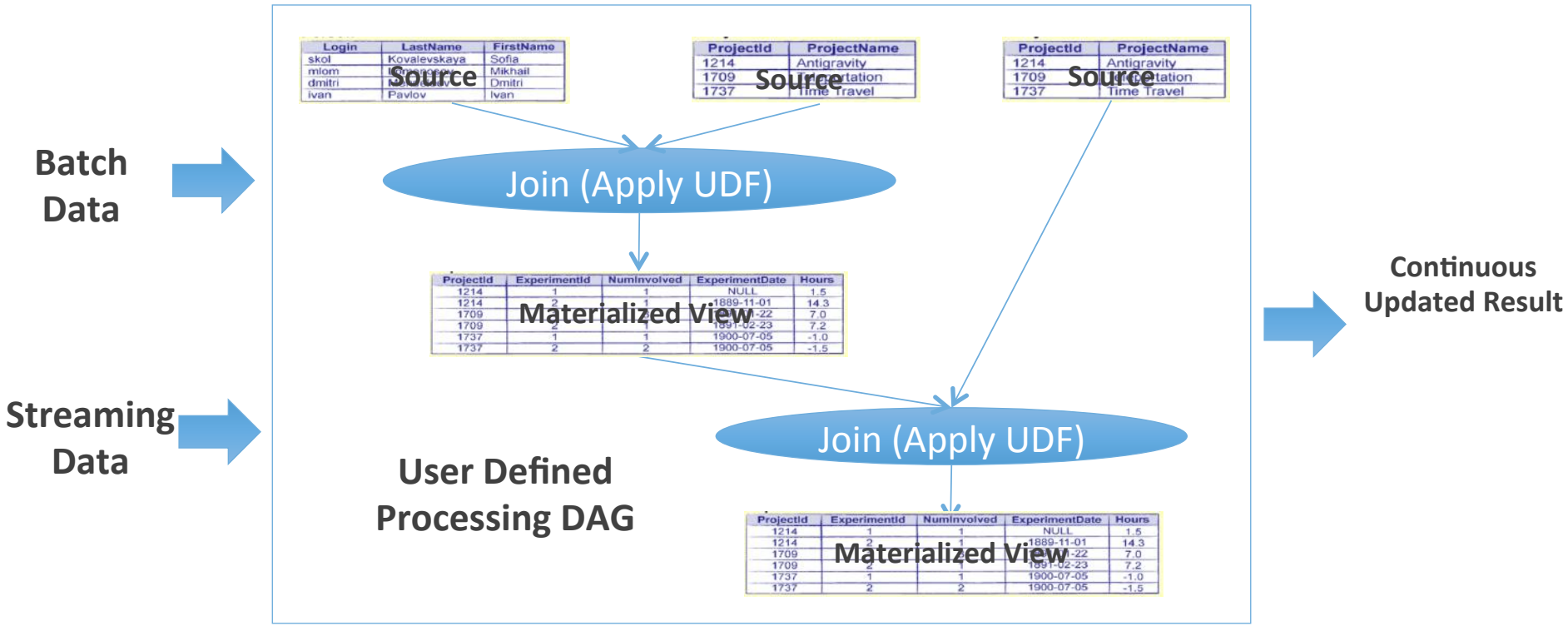
应用场景-索引构建

- Data Storage for Batch and Streaming Processing



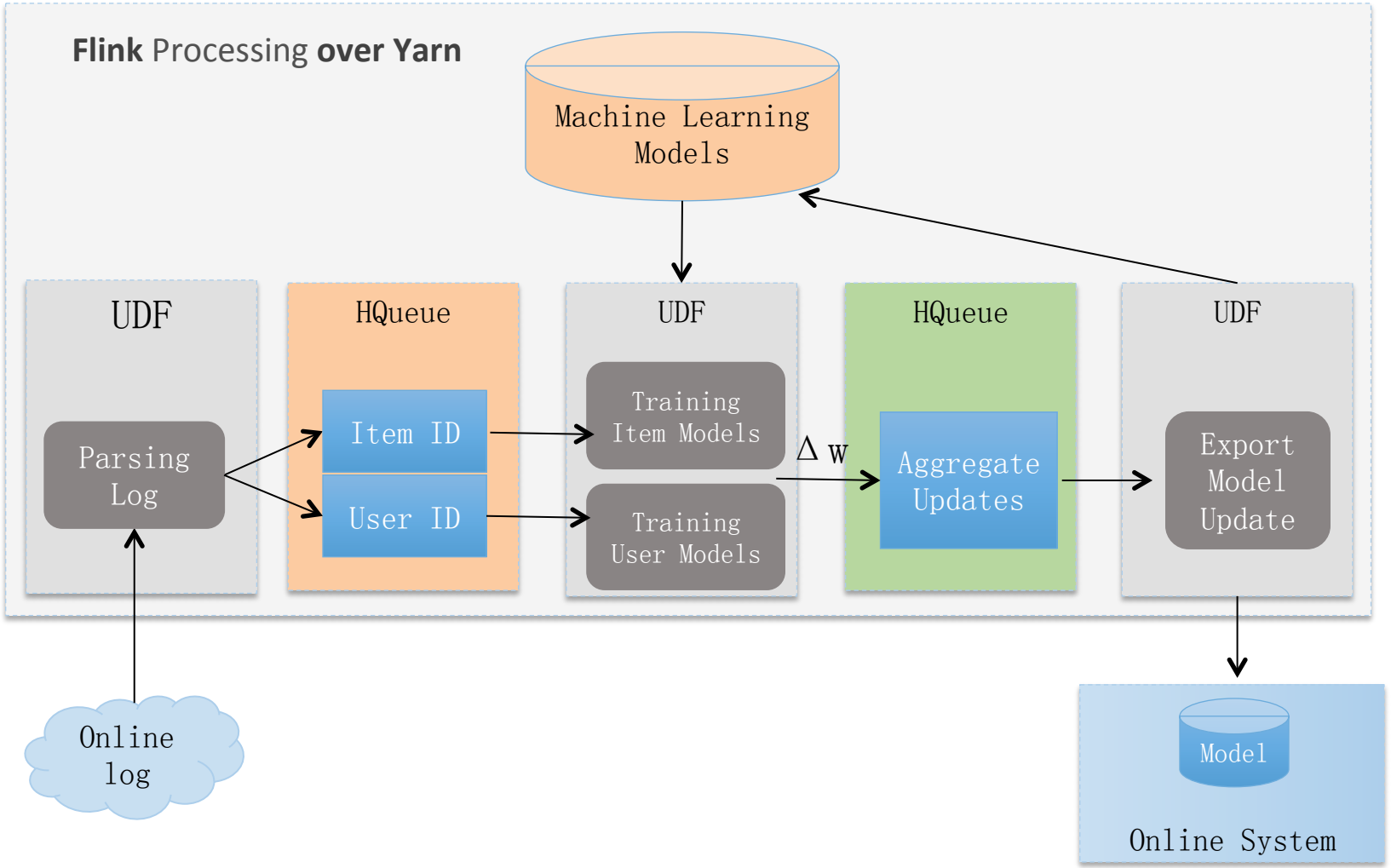
应用场景-索引构建

- Continuous Updated Materialized View on HBase



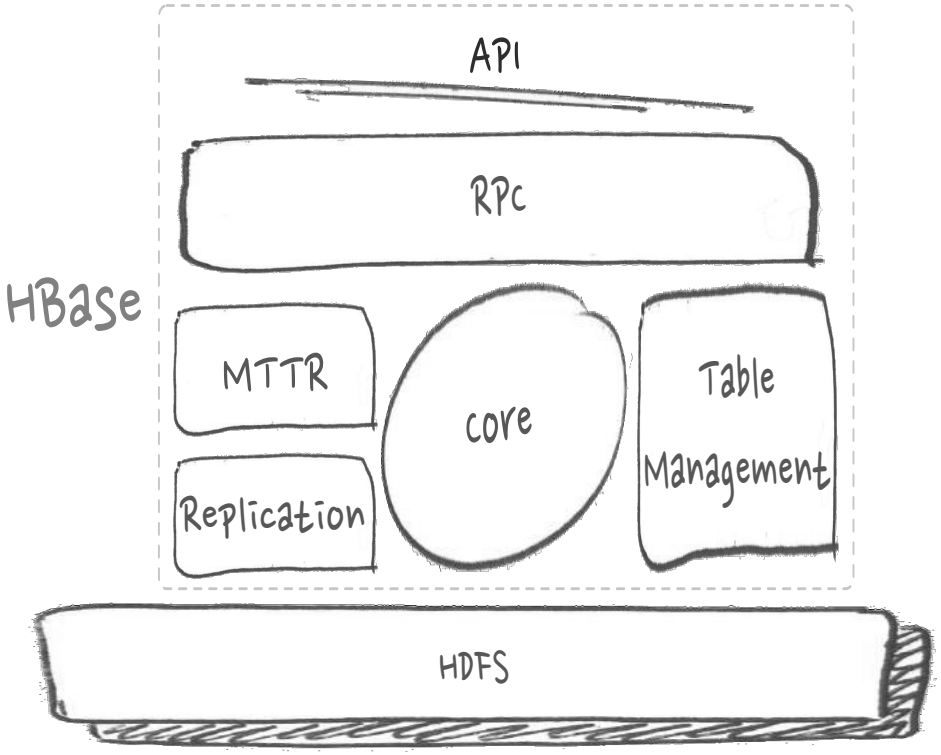
应用场景-机器学习

- Database and queue service for ML

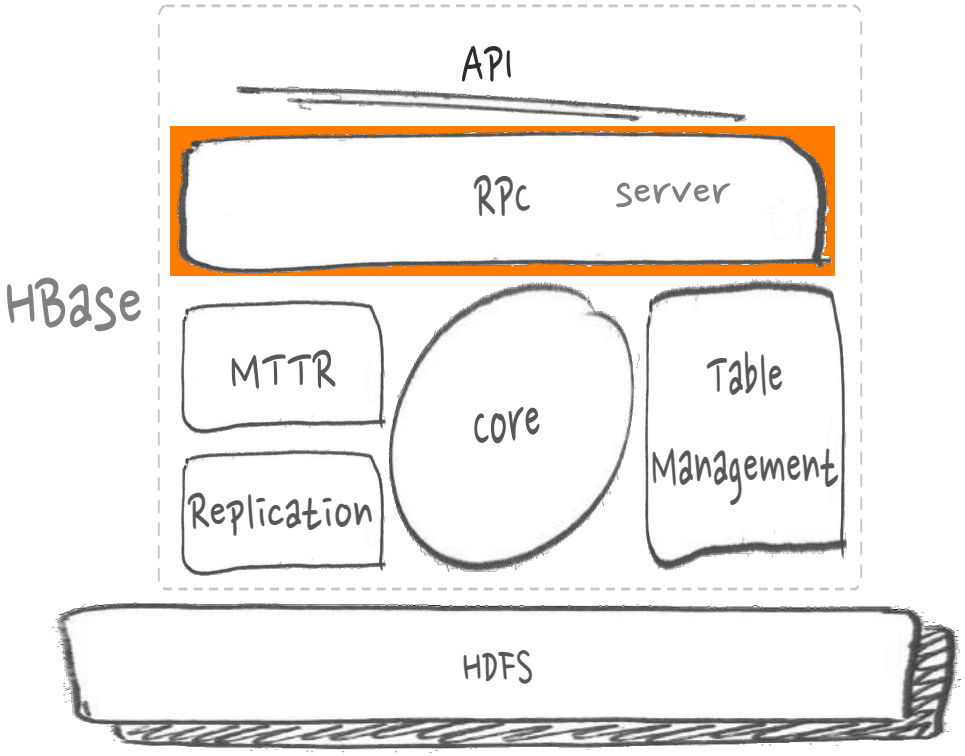


问题和优化:Overview

- HBase架构分层

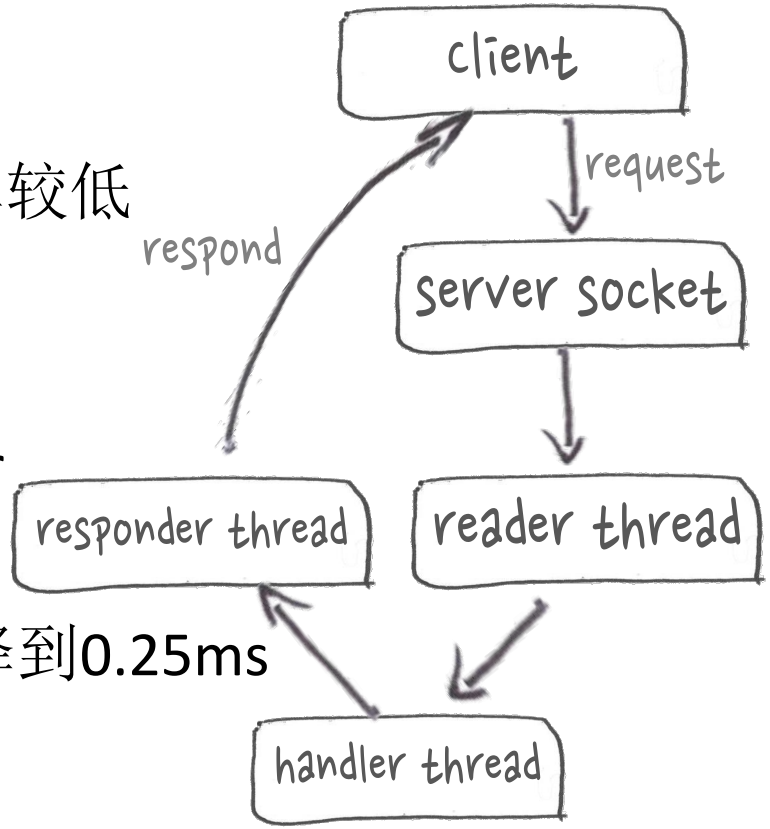


问题和优化:RPC的瓶颈和优化



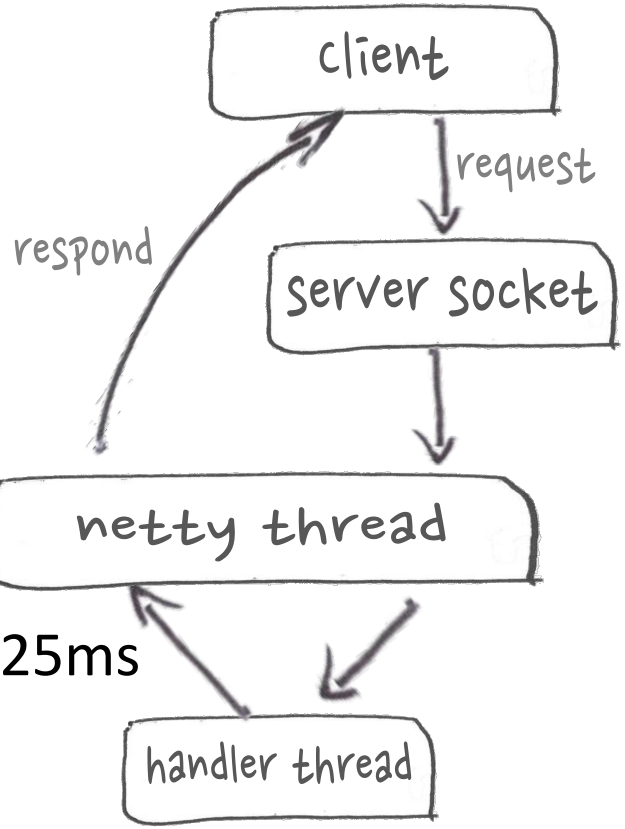
问题和优化:RPC的瓶颈和优化

- 实际问题
  - 原有RpcServer的线程模型效率较低
- 优化手段
  - Netty可以更高效的复用线程
  - 基于Netty实现HBase RpcServer
- 线上效果
  - rpc平均响应时间从0.92ms下降到0.25ms
  - Rpc吞吐能力提高接近2倍

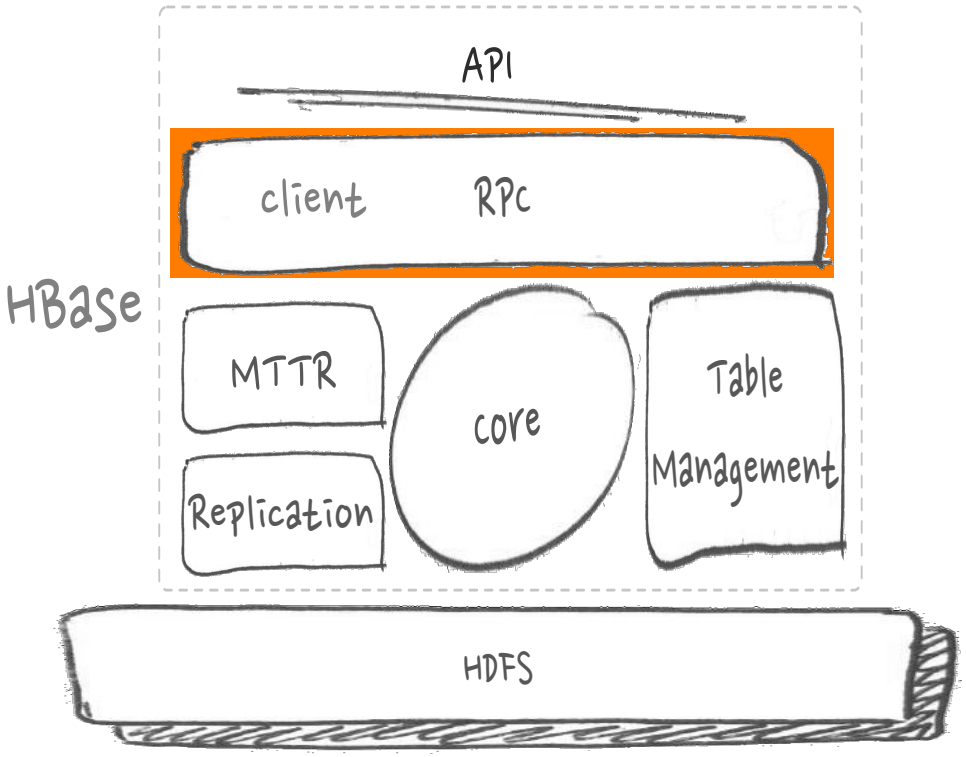


问题和优化:RPC的瓶颈和优化

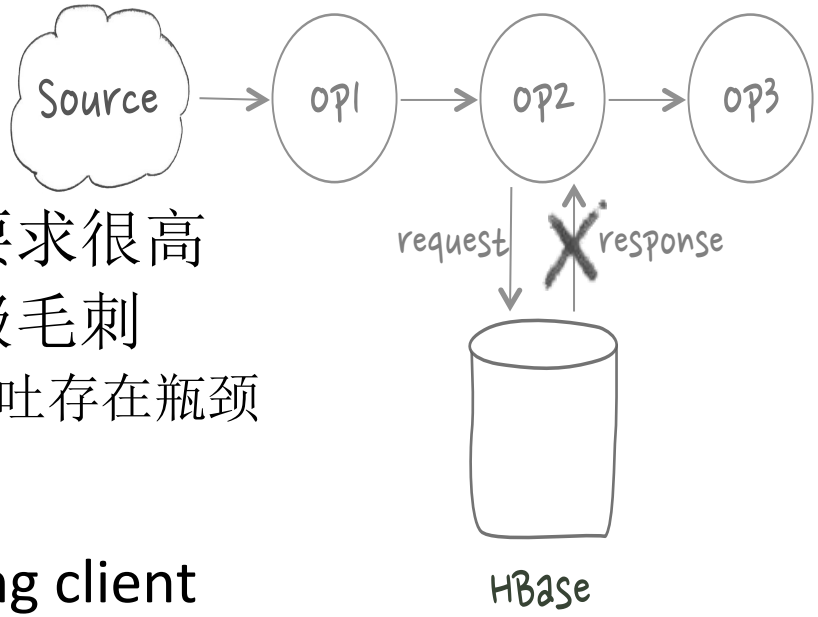
- 实际问题
  - 原有RpcServer的线程模型效率较低
- 优化手段
  - Netty可以更高效的复用线程
  - 基于Netty实现HBase RpcServer
- 线上效果
  - rpc平均响应时间从0.92ms下降到0.25ms
  - Rpc吞吐能力提高接近2倍



问题和优化:异步与吞吐


API
client
RPc
HBase
MTTR
core
Table
Management
Replication
HDFS

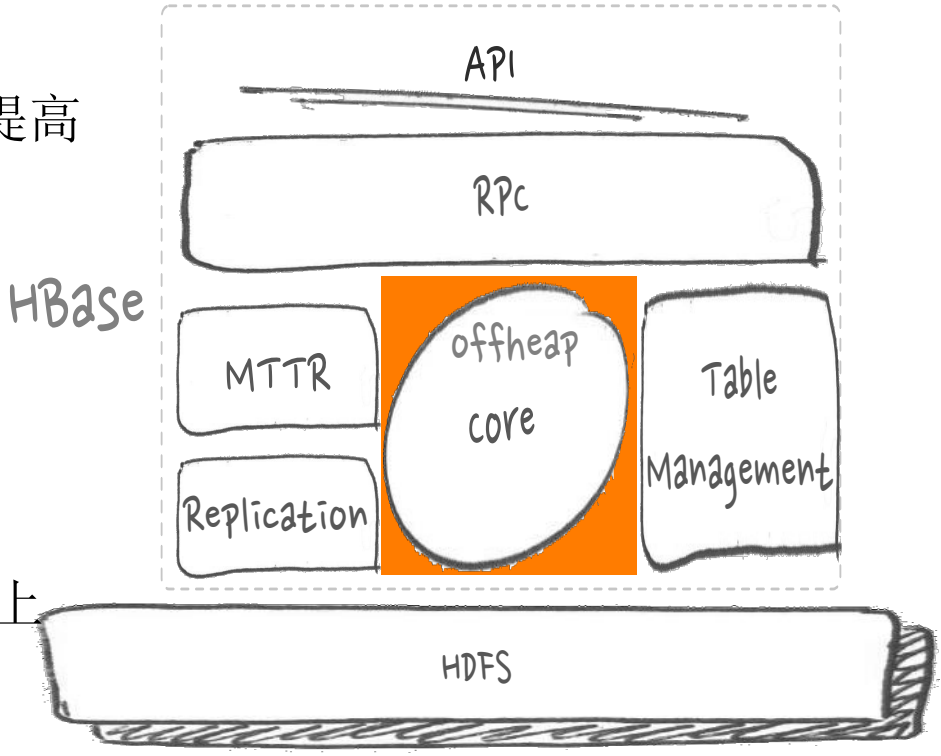
问题和优化:异步与吞吐



- 实际问题
  - 流式计算对于实时性的要求很高
  - 分布式系统无法避免秒级毛刺
    - 同步模式对毛刺敏感，吞吐存在瓶颈
- 优化手段
  - 基于netty实现non-blocking client
  - 基于protobuf的non-blocking Stub/RpcCallback实现callback回调
- 线上效果
  - 和flink集成后实测吞吐较同步模式提高2倍

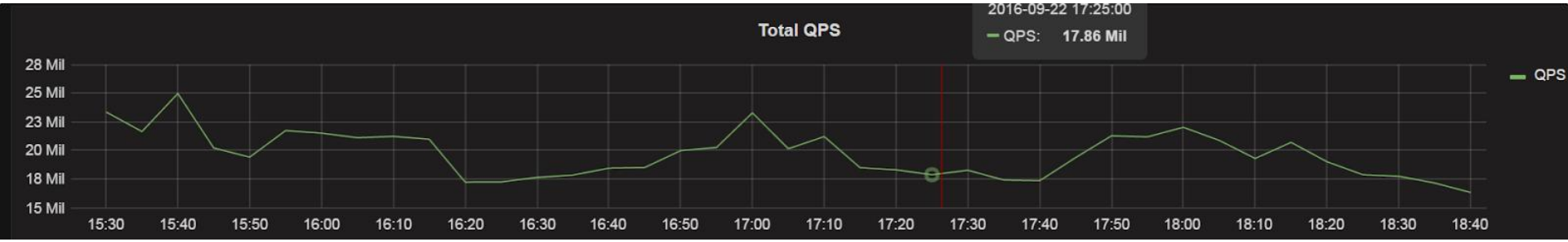
问题和优化:GC与毛刺

- 实际问题
  - PCIe-SSD的高IO吞吐能力下，读cache的换入换出速率大幅提高
  - 堆上的cache内存回收不及时，导致频繁的CMS gc甚至fullGC
- 优化手段
  - 实现读路径E2E的offheap
- 线上效果
  - Full和CMS gc频率降低200%以上
  - 读吞吐提高20%以上

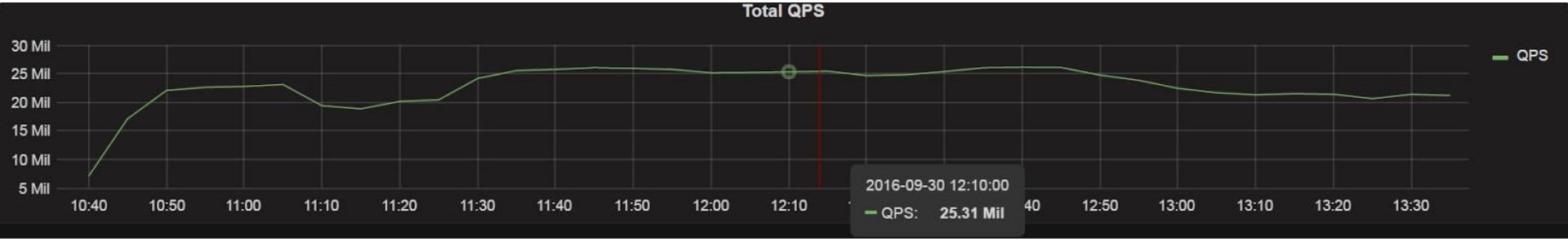


问题和优化:GC与毛刺

- Before



- After



- https://blogs.apache.org/hbase/entry/offheap-read-path-in-production

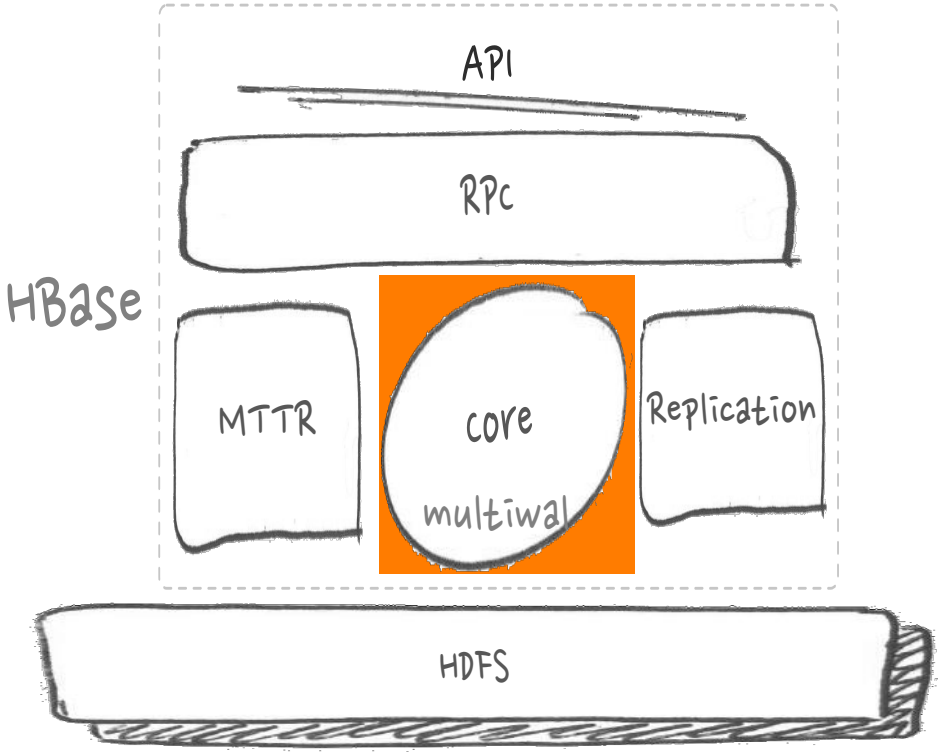
问题和优化: IO隔离和优化

- 实际问题
  - HBase对IO敏感，磁盘打满会造成大量毛刺
  - 大IO来源
    - 计算存储混布，batch作业产生大量的IO
    - HBase自身：Flush/Compaction
- 优化手段
  - 利用HDFS的Heterogeneous Storage功能
    - ALL_SSD for WALs
    - ONE_SSD for HFile
    - Bulkload支持指定storage policy
    - MR临时数据目录(mapreduce.cluster.local.dir)只使用SATA盘

问题和优化: IO隔离和优化

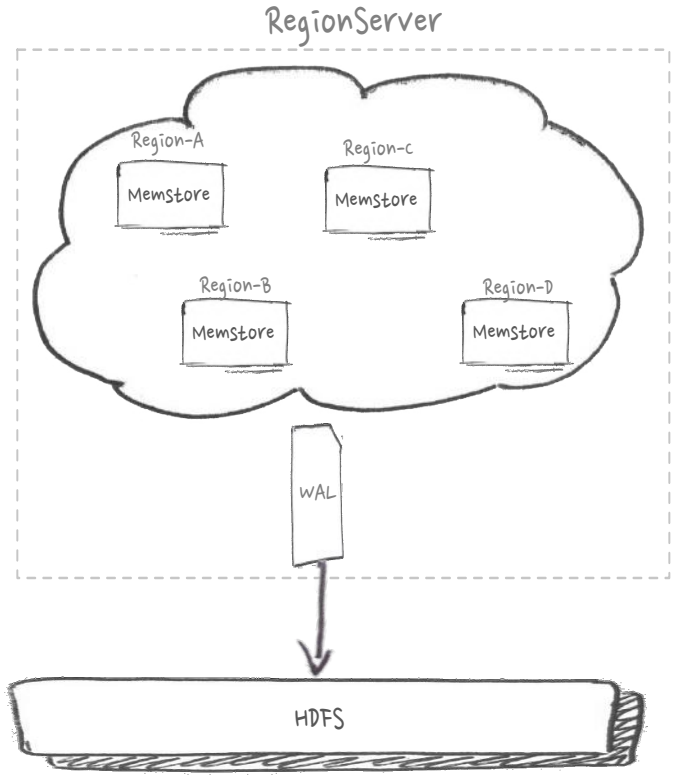
- 优化手段
  - Compaction限流
  - Flush限流
  - Per-CF flush
- 线上效果

问题和优化:IO利用



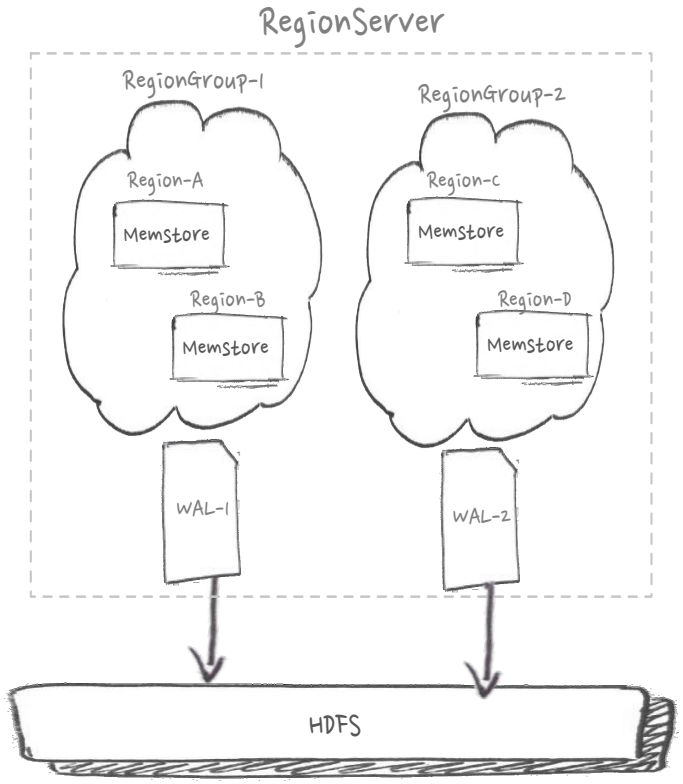
问题和优化:IO利用

- 实际问题
  - 单WAL无法充分使用磁盘IO
    - HDFS写3份副本
    - 通用机型有12块HDD盘
    - SSD的IO能力远超HDD
- 优化手段
  - 支持多WAL
    - 对region分组并进行合理映射
  - 支持app间IO隔离
    - 基于Namespace的WAL分组
- 上线效果
  - 全HDD盘下写吞吐提高20%，全SSD盘下写吞吐提高40%
  - 线上写入平均响应延时从0.5ms下降到0.3ms



问题和优化:IO利用

- 实际问题
  - 单WAL无法充分使用磁盘IO
    - HDFS写3份副本
    - 通用机型有12块HDD盘
    - SSD的IO能力远超HDD
- 优化手段
  - 支持多WAL
    - 对region分组并进行合理映射
  - 支持app间IO隔离
    - 基于Namespace的WAL分组
- 上线效果
  - 全HDD盘下写吞吐提高20%，全SSD盘下写吞吐提高40%
  - 线上写入平均响应延时从0.5ms下降到0.3ms



开源&未来

- 拥抱开源
  - HBASE-17263: Netty based rpc server impl
  - HBASE-16833: Implement asynchronous hbase client (*)
  - HBASE-11425: Cell/DBB end-to-end on the read-path (*)
  - HBASE-17138: Backport read-path offheap (HBASE-11425) to branch-1
  - HBASE-8329: Limit compaction speed (*)
  - HBASE-14969: Add throughput controller for flush
  - HBASE-14906: Improvements on FlushLargeStoresPolicy
  - HBASE-14457: Improve Multiple WAL for production usage
- 未来

About Us

- 团队现状
  - 2个社区committer，若干contributor
  - 回馈社区patch数超过100个
- We are recruiting
  - jueding.ly@alibaba-inc.com

Q & A


WOTA
51CTO
Thank you !