Bada 构建主从/去中心混合架构的NoSQL

DTCC

2015中国数据库技术大会

DATABASE TECHNOLOGY CONFERENCE CHINA 2015

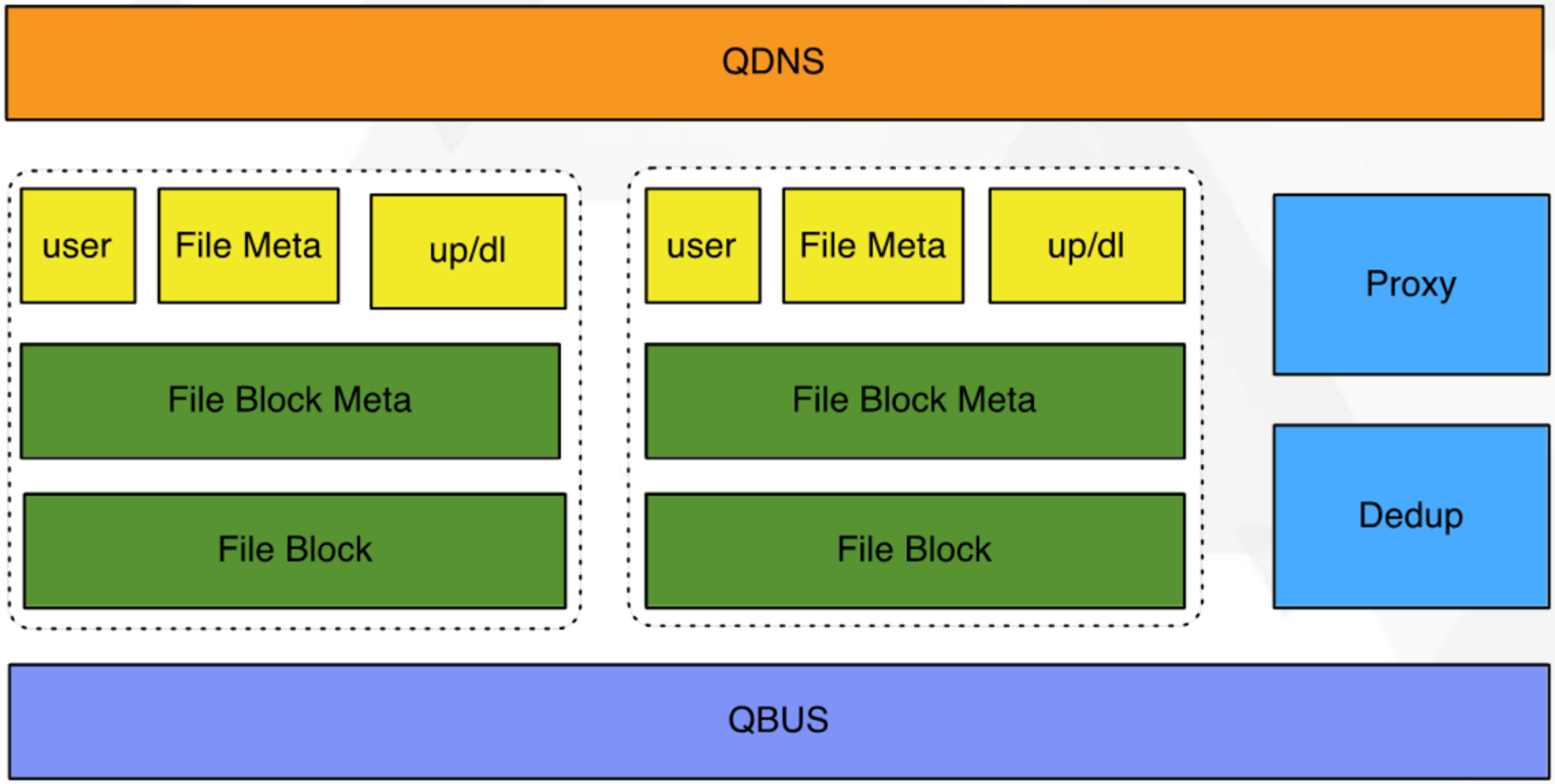
大数据技术探索和价值发现

IT168 ChinaUnix ITPUB

概要

- 背景介绍 - 百亿数据的困惑
- 混合架构设计的考虑
- 新架构？新挑战！
- 经验总结

背景介绍 - 业务架构


QDNS
user
File Meta
up/dl
File Block Meta
File Block
user
File Meta
up/dl
File Block Meta
File Block
Proxy
Dedup
QBUS

背景介绍 – 原有方案

- (File / File Block) Deduplication
  - MongoDB v2.0/2.2/2.4
  - Mongos + Mongod(Primary 1 + Secondary 2)
  - 144G RAM + 300G SSD * 5
  - 12 servers / IDC, 20+ IDC

背景介绍 – 原有方案

- 百亿数据面临的问题
  - Bson 数据膨胀率高
  - 扩展节点周期长
  - 延迟不稳定：Thread -> DB锁
  - 数据可靠性低, 主从切换数据丢失

混合架构设计的考虑

- 新的方案？
  - 延迟低、稳定
  - 膨胀率低
  - 节点伸缩效率高

混合架构设计的考虑

- 延迟低、稳定
  - 磁盘介质：SSD
  - 存储引擎：LevelDB / RocksDB
  - 网络模型：Erlang OTP
  - 副本机制：Primary & Secondary

混合架构设计的考虑

- 膨胀率
  - LevelDB： snappy
  - RocksDB： snappy，zlib, bzip2, lz4, lz4_hc

混合架构设计的考虑

- 节点伸缩
  - 迁移最小粒度partition
  - LevelDB Instance/partition

混合架构设计的考虑

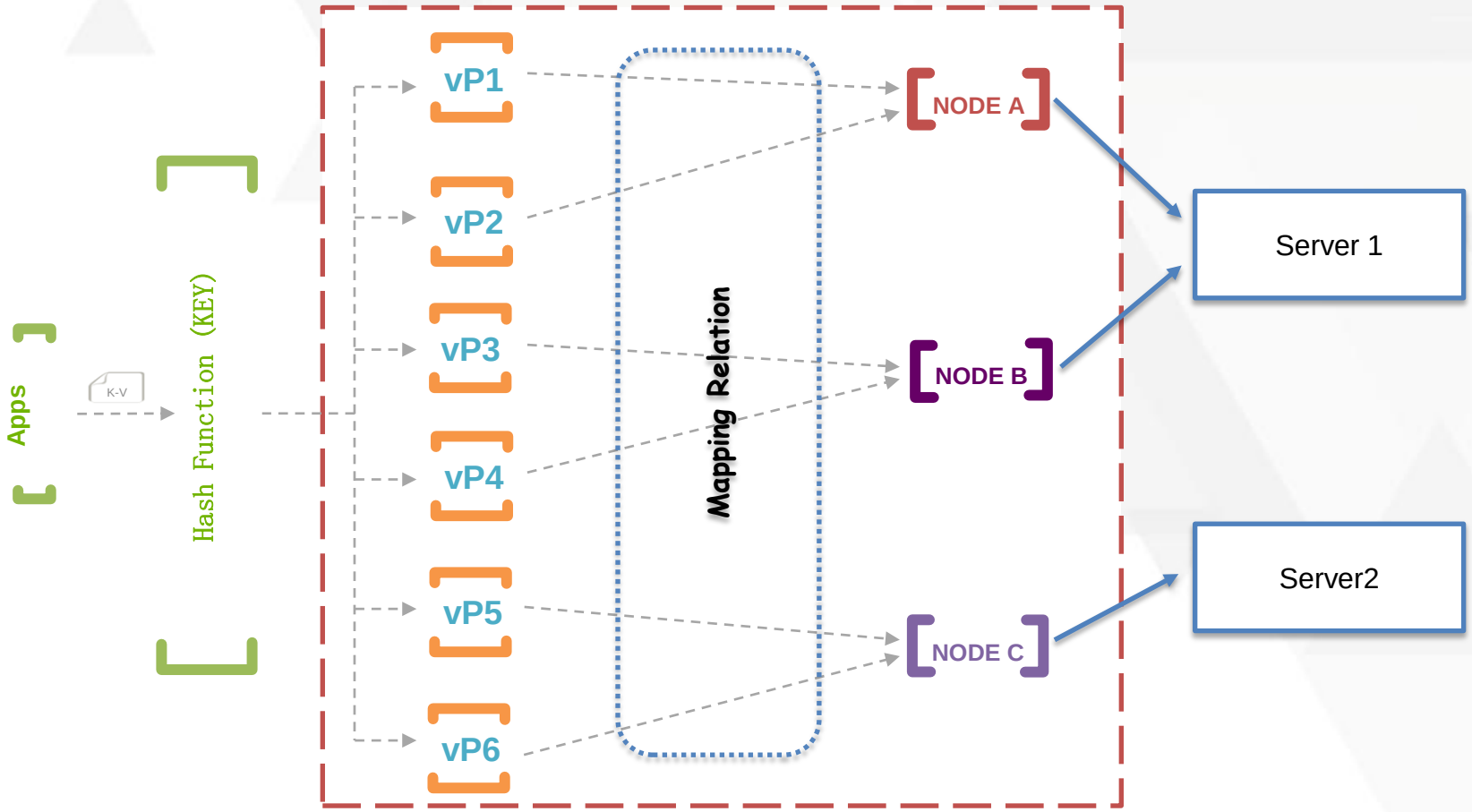
- 分布式系统关键要素
  - 路由策略
  - 副本策略
  - 一致性
  - 容错

混合架构设计的考虑

- 路由策略
  - 两级映射

- 优点：
  - 算法简单
  - 节点伸缩负载均衡

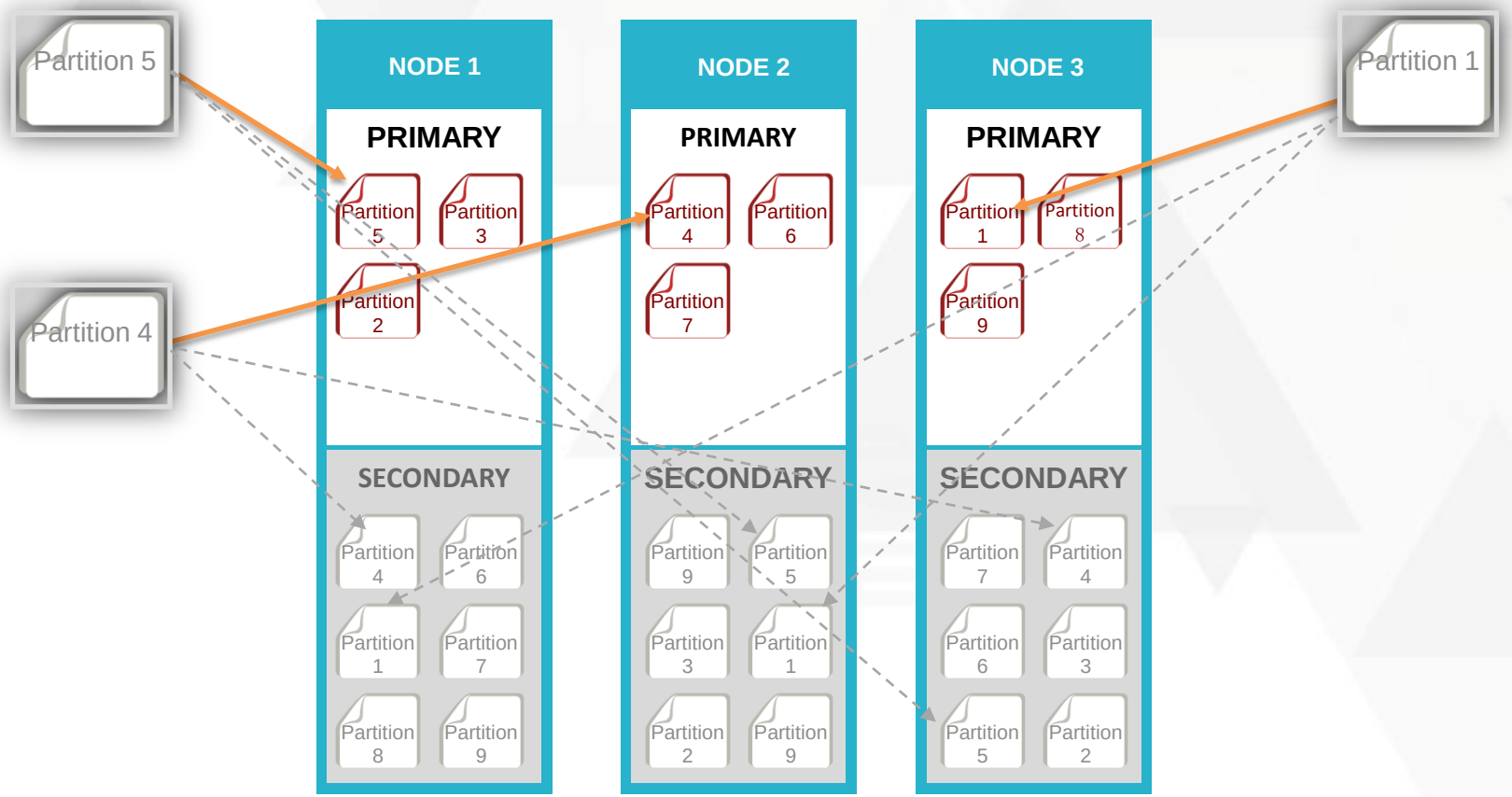
混合架构设计的考虑


Apps
K-V
Hash Function (KEY)
vP1
vP2
vP3
vP4
vP5
vP6
Mapping Relation
NODE A
NODE B
NODE C
Server 1
Server2

混合架构设计的考虑

- 副本策略：
  - Primary + Secondary * 2
  - 扩容、Binlog复制以Partition为单位
  - partition、主副本均匀分布在所有Node
- 优点
  - 读写单副本，延迟低
  - 节点伸缩效率高
  - Binlog并发复制
  - 数据节点负载均衡

混合架构设计的考虑

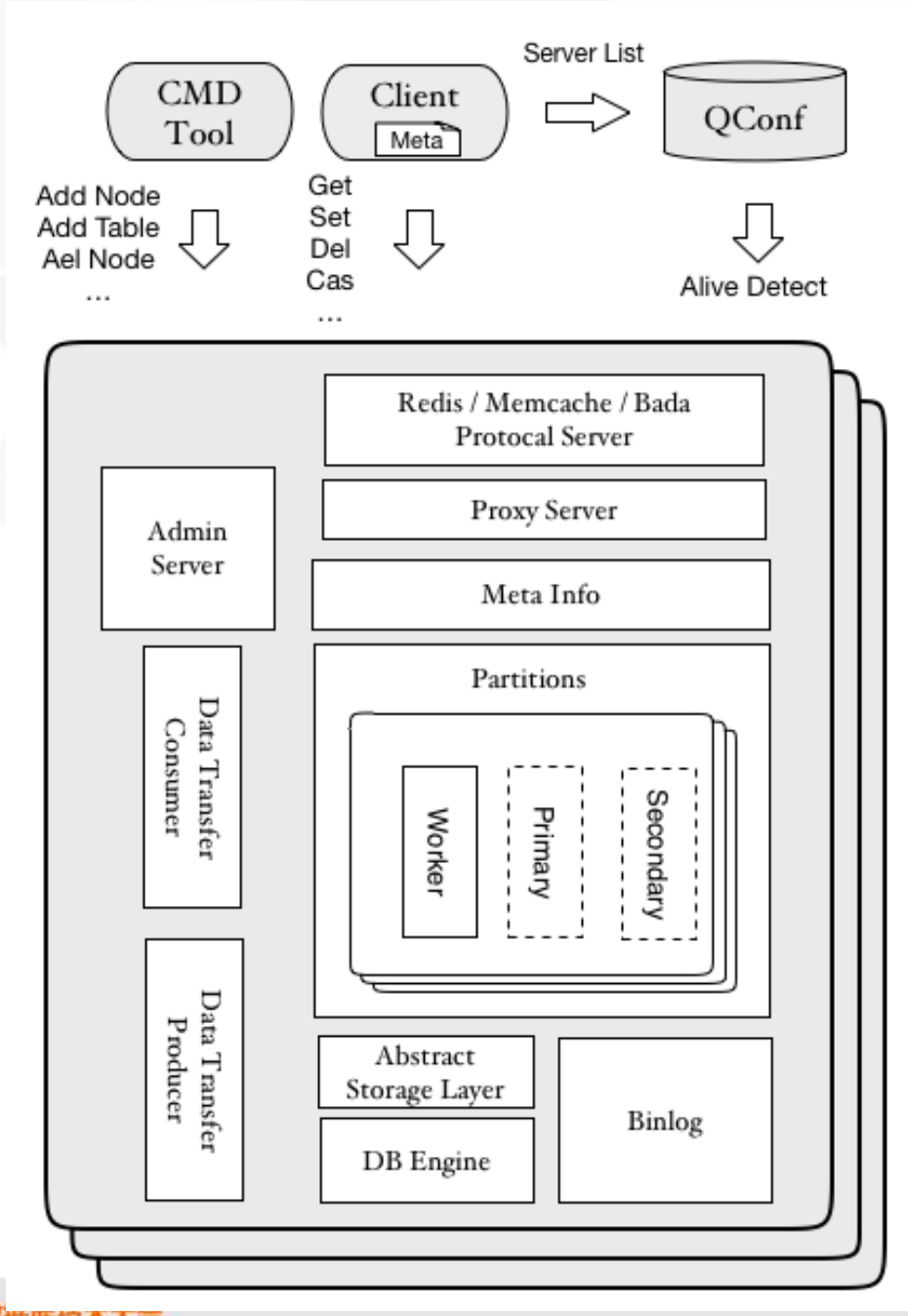

Partition 5
Partition 4
NODE 1
PRIMARY
Partition 5
Partition 3
Partition 2
SECONDARY
Partition 4
Partition 6
Partition 1
Partition 7
Partition 8
Partition 9
NODE 2
PRIMARY
Partition 4
Partition 6
Partition 7
SECONDARY
Partition 9
Partition 5
Partition 3
Partition 1
Partition 2
Partition 9
NODE 3
PRIMARY
Partition 1
Partition 8
Partition 9
SECONDARY
Partition 7
Partition 4
Partition 6
Partition 3
Partition 5
Partition 2
Partition 1

混合架构设计的考虑

- 一致性
  - 最终一致
  - 读写主副本，多数情况下一致性强

混合架构设计的考虑

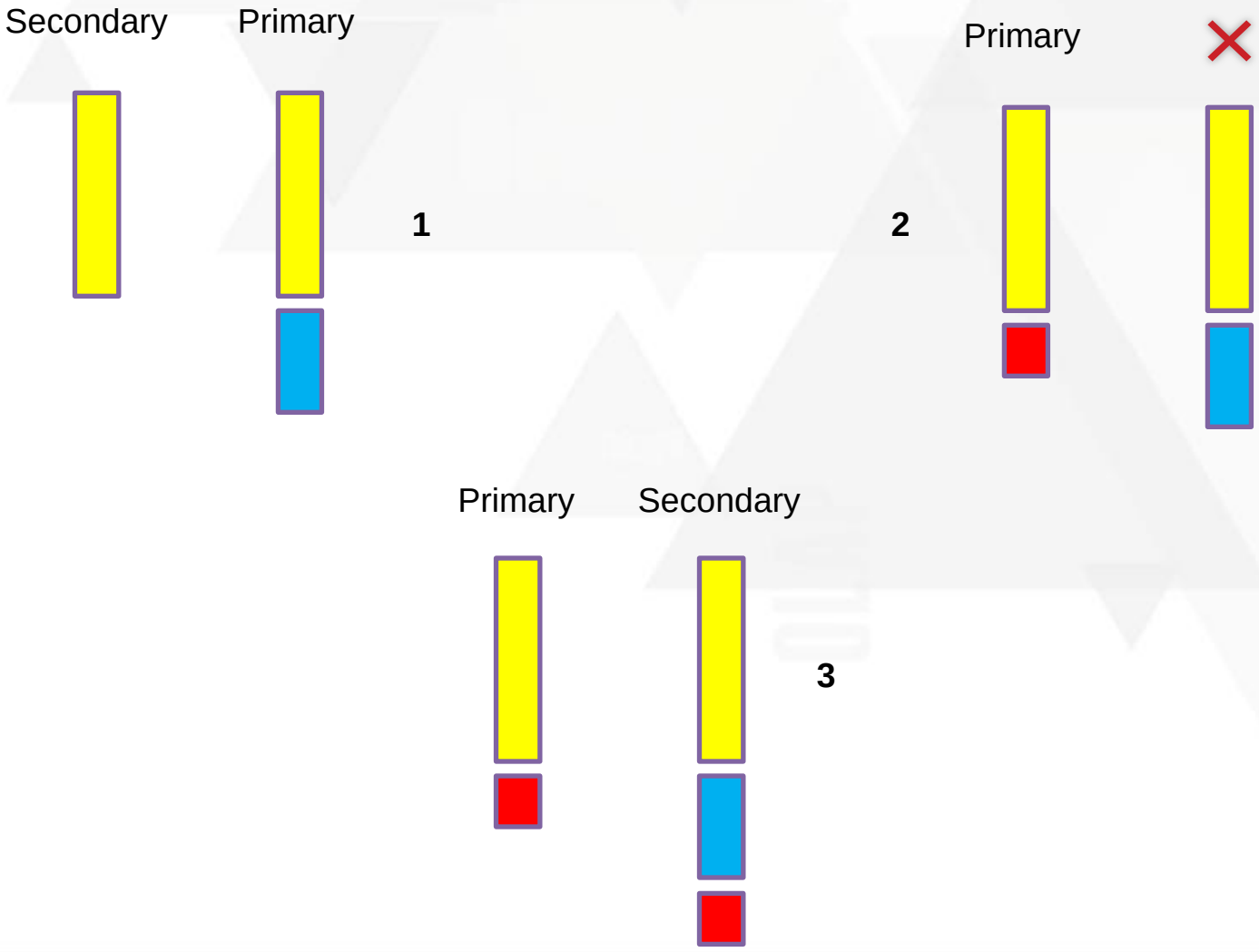
- 容错：
  - 选举 N/2+1
  - 最大 gopid(serverID, opid, timestamp)
  - 每组partition之间投票


CMD Tool
Client
Meta
Server List
QConf
Add Node
Add Table
Ael Node
...
Get
Set
Del
Cas
...
Alive Detect
Redis / Memcache / Bada Protocal Server
Proxy Server
Admin Server
Meta Info
Partitions
Worker
Primary
Secondary
Data Transfer Consumer
Data Transfer Producer
Abstract Storage Layer
DB Engine
Binlog

新架构？新挑战！

- 问题：主从架构异步复制存在数据丢失
  - 网络故障
  - 服务器、机柜维护

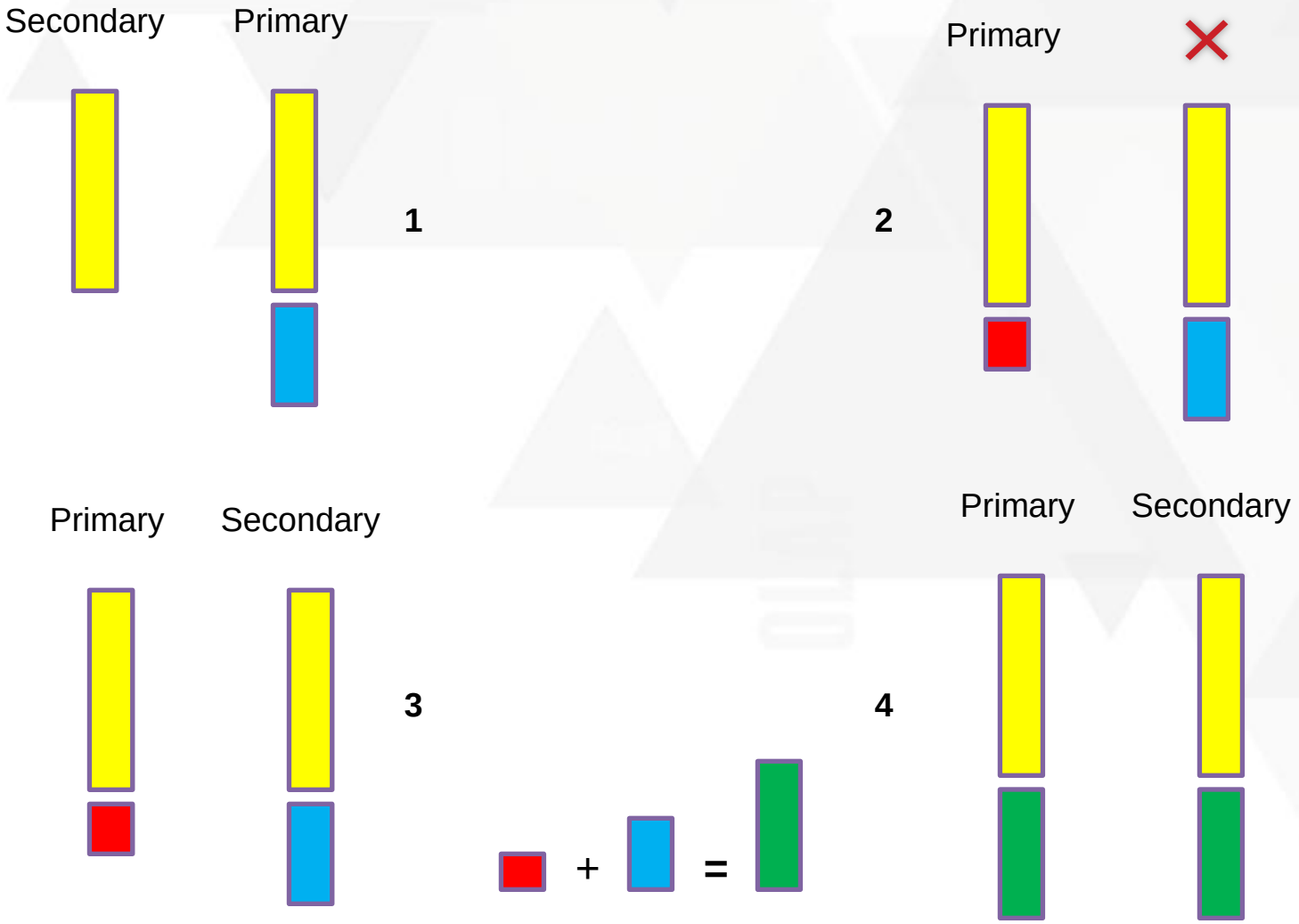
新架构？新挑战！



新架构？新挑战！

- 解决
  - Binlog Merge

新架构？新挑战！


Secondary
Primary
1
Primary
2
Primary
Secondary
3
+
=
Primary
Secondary
4

新架构？新挑战！

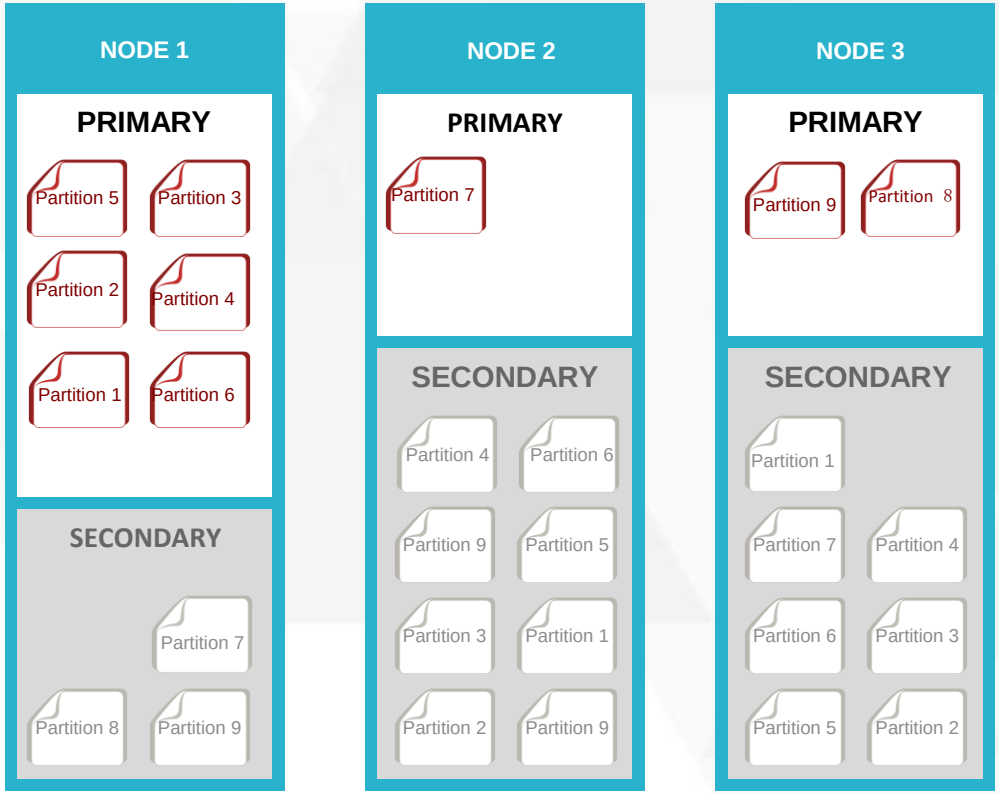
- 优化
  - 高效的查找同步点
    - 批量发送，二分查找
  - 如何最快恢复服务
    - 对key排序，只回放最后一次操作
  - 保持多副本binlog文件一致
    - Secondary binlog 删除重做

新架构？新挑战！

- 问题：多partition引发的选举风暴
  - 无谓的网络、CPU消耗
- 解决：
  - 广播 -> 多播
  - 优化选举时机

新架构？新挑战！

- 问题：主副本不均衡
  - CPU、网络、I/O倾斜
- 解决：
  - 自动主平衡算法
  - 要求最小步数，changeprimary 最小影响



经验总结

- 业务需求出发
- 选择开源方案
- 设计尽可能简单
- 小步快跑
- 单元、集成、沙盒测试
- 灰度上线，预案就绪

WEIBO: @Chancey
MAIL: chanceycn@gmail.com
GITHUB: https://github.com/Qihoo360