百度分布式数据库实践

肖智文/尹博学
dba@baidu.com

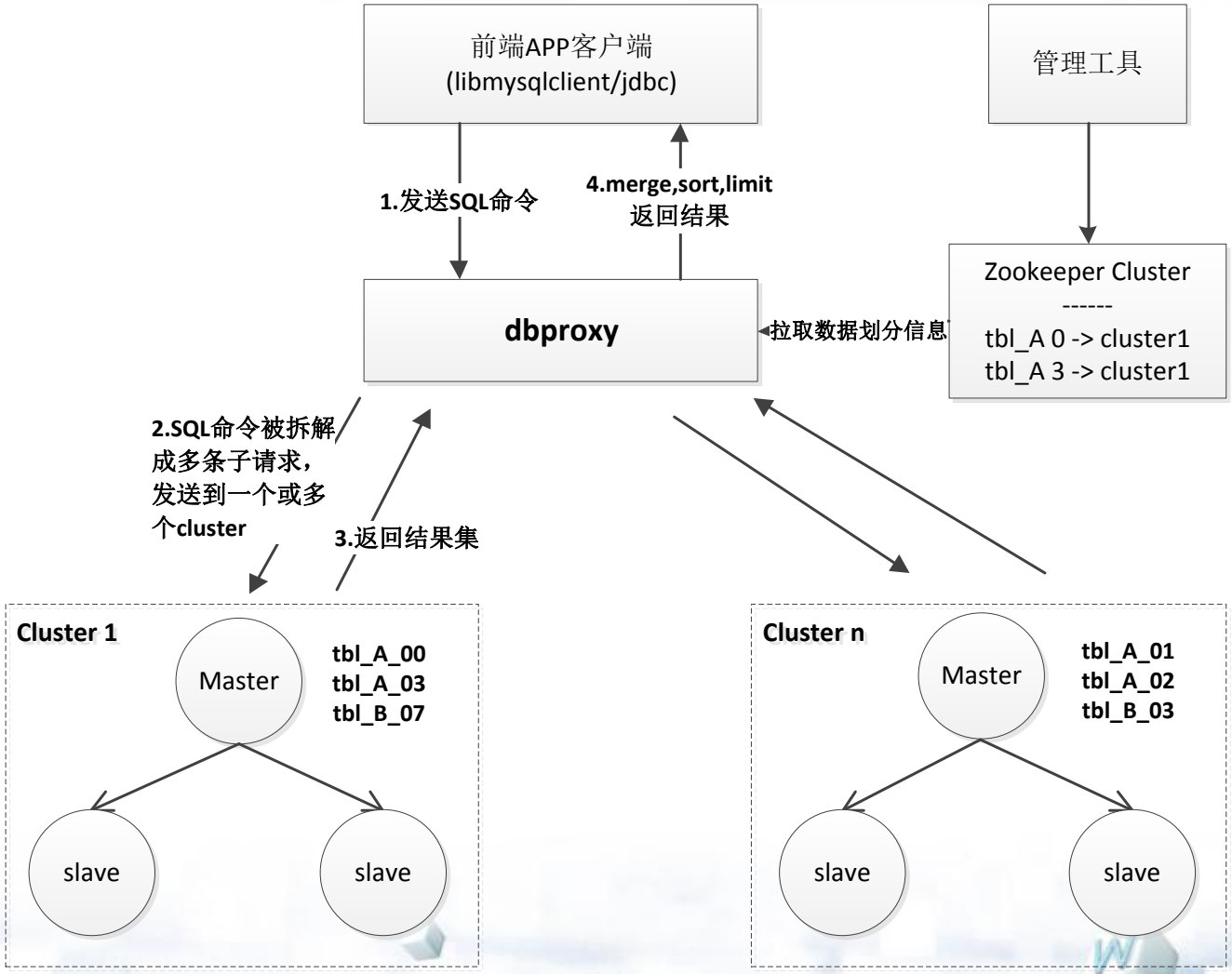
Baidu 百度

DTCC2012

分布式数据库

- ✓ 产品定位
  - 尽量保证数据库特性，提升数据规模
  - 线上低延迟的访问
  - 满足具有一定复杂关系的数据操作
- ✓ 设计原则
  - 应用访问方式不变
  - 应用知道数据逻辑分布
  - 不同访问模式提供的功能不同
  - 自动发现/人工决定/自动处理

分布式架构



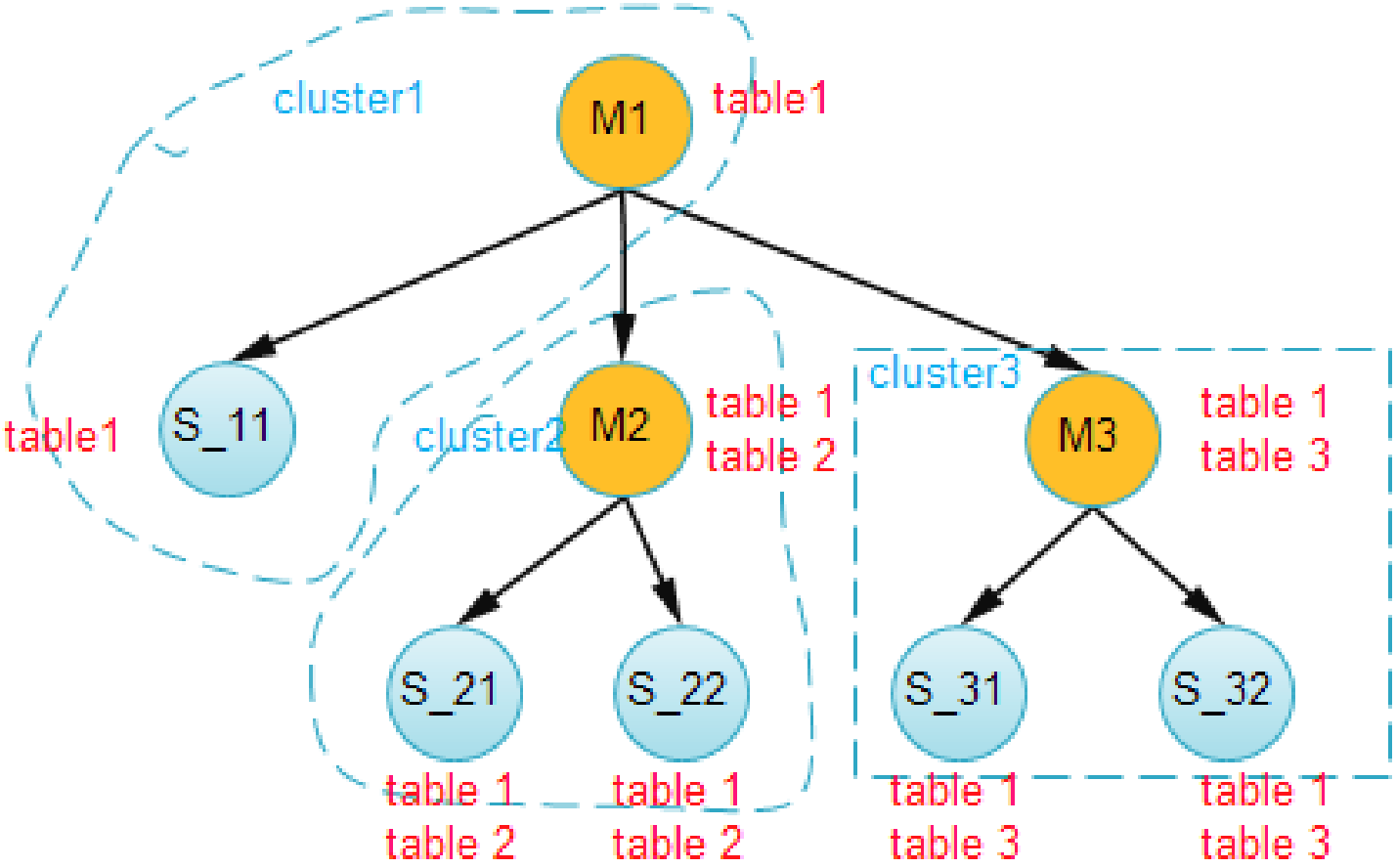
数据访问模式

- ✓ Scan & Search
- ✓ 基于Partition Key
  - 单表单机
  - 单表多机
  - 多表单机
  - 多表多机
- ✓ 不基于Partition Key
  - 单表
  - 多表

数据划分

- ✓ 数据划分列
  - 划分和定位数据，不能更新该字段
  - 与索引没有关系
  - 只支持一维
- ✓ 实现的划分方式
  - 范围分断（range）
  - 散列取模（hash_mod）
  - 枚举/枚举（list）
  - 组合模式
- ✓ 数据关系
  - 继承
  - 绑定

数据拓扑

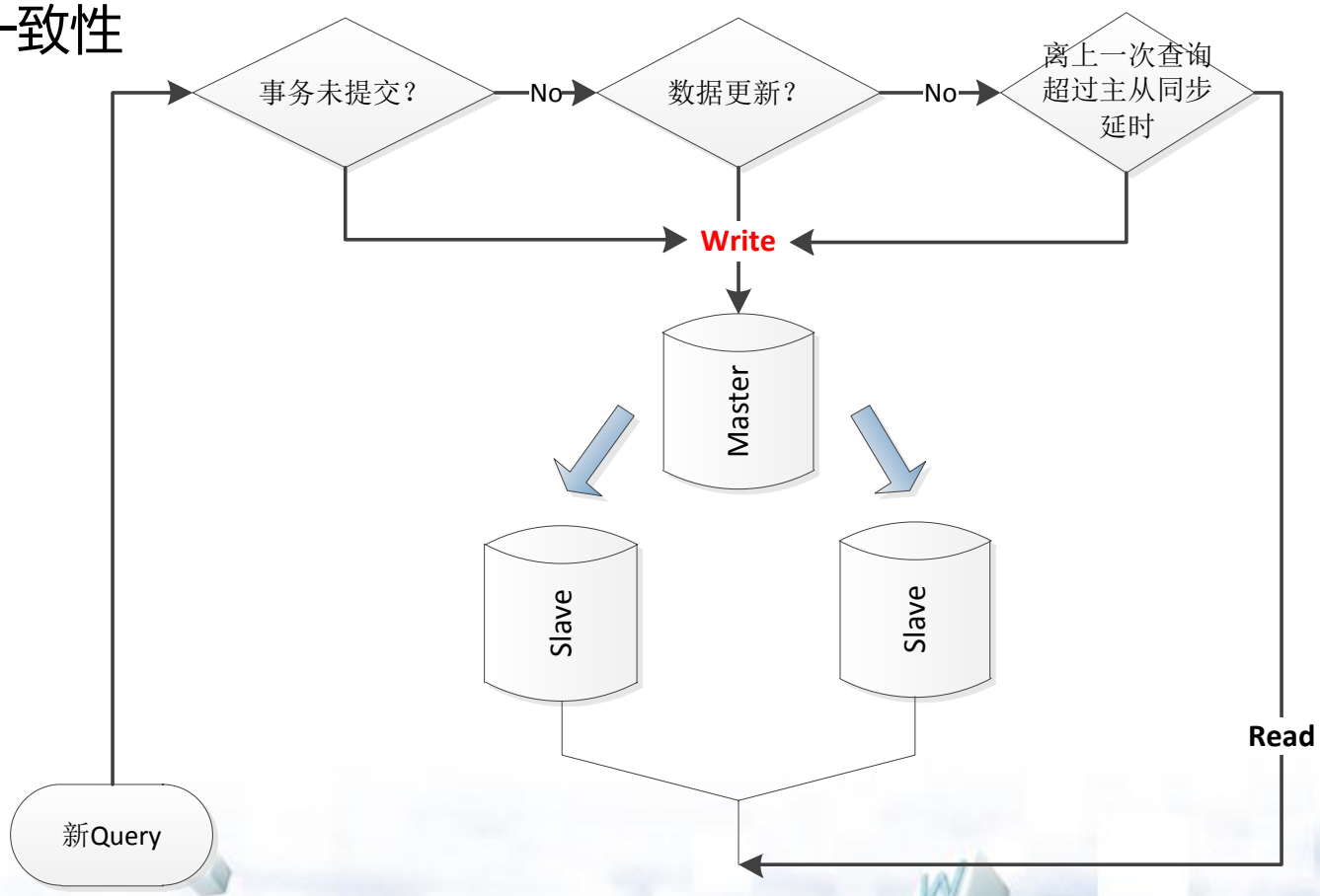

cluster1
M1
table1
table1
S_11
cluster2
M2
table 1
table 2
cluster3
M3
table 1
table 3
S_21
S_22
S_31
S_32
table 1
table 2
table 1
table 2
table 1
table 3
table 1
table 3

事务实现

- ✓ 支持的功能
  - 访问落在同一分片上的请求，由mysql保证
  - 数据绑定
  - 小表通过继承
- ✓ 不支持
  - 分布式事务

读写分离

- ✓ 事务内的所有语句发往主库执行
- ✓ 保证会话一致性



数据一致性

- ✓ metadata使用zookeeper
- ✓ 同一tablet的不同副本之间
  - 最终一致性
  - 会话一致性
- ✓ 不同tablet之间
- ✓ 不支持分布式事务

系统可用性 - 1

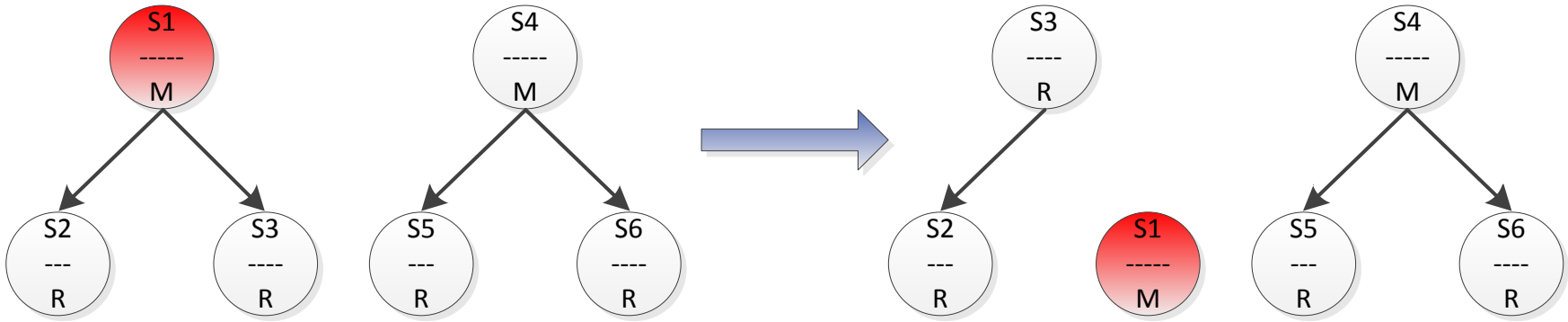
- ✓ 元信息（zookeeper）
  - 数据拓扑不能变更
  - 读写功能不受影响
- ✓ 接入层（dbproxy）
  - 应用端重试
- ✓ 数据层（tablet）
  - 多副本
  - 健康检查
  - 主从切换

系统可用性 - 2

- ✓ 心跳
  - 对每个sharding，在主库上每秒写入当前unix_timestamp
  - 在副本节点上读取该记录，判断同步延迟和主从同步关系
- ✓ 节点健康度
  - 机器负载
  - 数据不同步
  - 数据同步延时超过阀值
  - 同步恢复后提供读服务
- ✓ 副本节点
  - 通知人工处理
  - 根据配置的副本数自动添加副本

系统可用性 - 3

- ✓ 主节点故障
  - 副本节点提升
  - 副本数保全



数据扩容

- ✓ 预防扩容
  - 单机多实例部署
  - 拆分成较多在数据
- ✓ tablet 分裂
  - 基于mysql主从复制
- ✓ tablet 迁移
- ✓ 数据均衡
  - 二维划分
  - 动态路由

管理工具

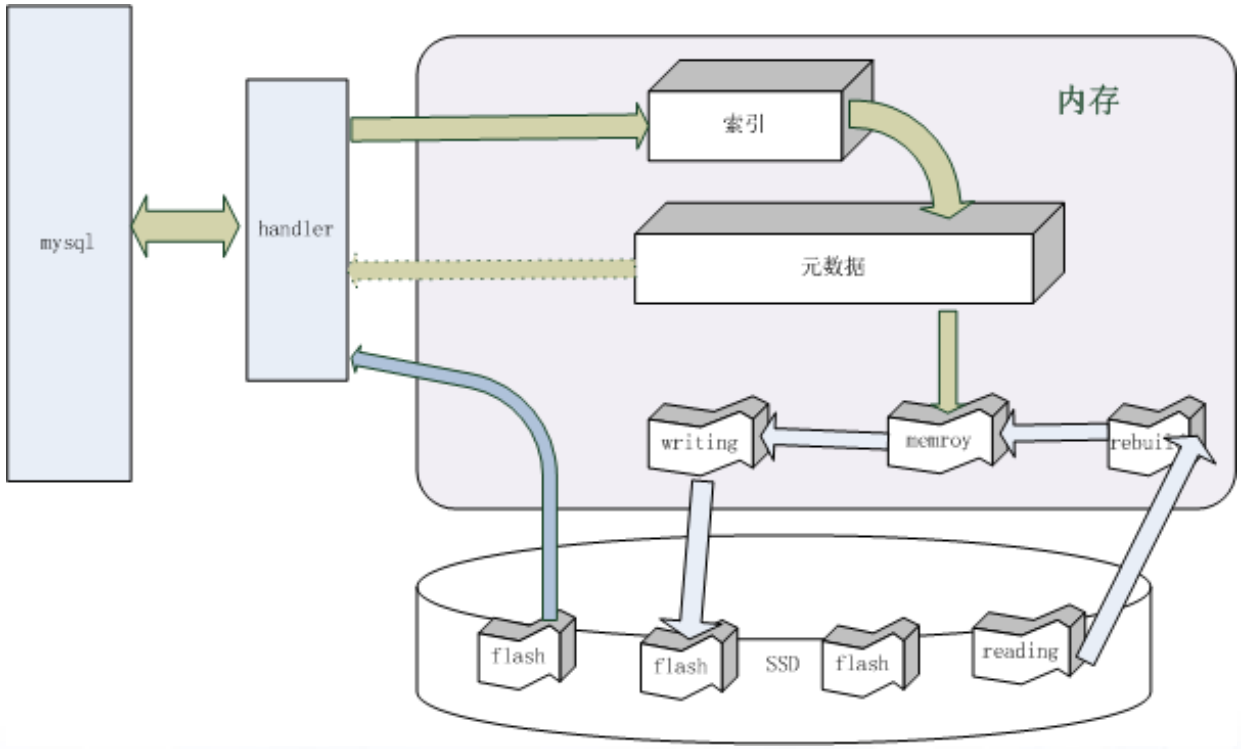
- ✓ Schema管理
  - 扩容
  - DDL
- ✓ 配置管理
- ✓ 用户管理
  - 增删用户
  - 属性修改
- ✓ 查询接口

MySQL功能扩展 - 1

- ✓ online schema change
  - 只支持innodb

MySQL功能扩展 - 2

- ✓ 支持主键查询、更新、删除的mysql引擎
  - 内存索引
  - 优化随机写
- ✓ 功能
  - 支持schema
  - snapshot
  - KV类应用
  - 原子计数
- ✓ 性能
  - QPS：6万
  - TPS：4万



应用示例 - 1

- ✓ 应用场景
  - 数据量上百T
  - 每天更新量几十T
- ✓ 问题
  - 数据导入效率
  - 吞吐

应用示例 - 2

- ✓ 应用场景
  - 数据量　数T
  - TPS　　百万/秒
- ✓ 问题
  - 并发性能

问题与挑战

- ✓ 集群化
  - 不同应用间的资源隔离
  - 数据迁移、扩容的
  - 运维的自动化程度
- ✓ 功能
  - 自动扩容
  - 自运维
  - 管理工具

谢谢！