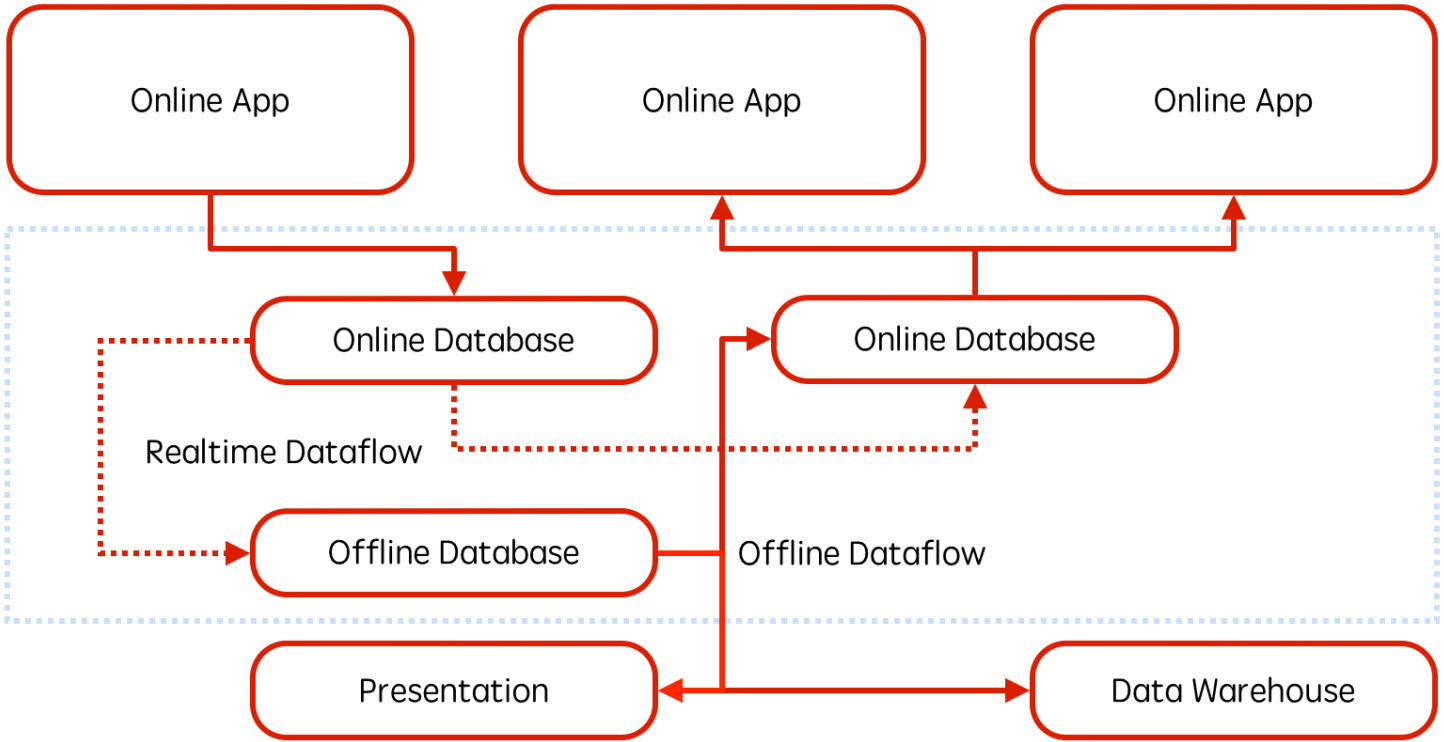
TiDB TechDay 2019

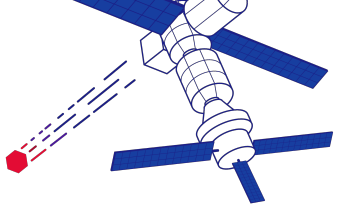
TiDB在小红书多场景应用

张俊骏
jzhang4@xiaohongshu.com
2019-06-30

小红书数据服务整体架构

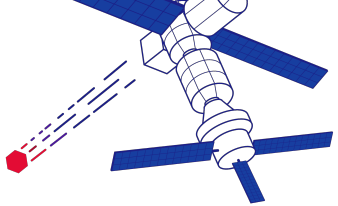


数据服务组件RoadMap



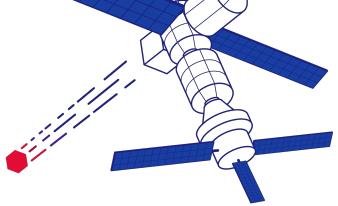
- 基本功能
  - 读写场景（get/batch get/scan/filter query/agg query）
  - 响应延迟（均线/95线/99线/999线/…）
  - 带宽
  - 可扩展性
- etl
  - 数据同步（同构或异构场景）
  - 离线导出（dump/scan）
  - 实时导出（binlog）
  - 离线导入（一般为mr或spark job）

数据服务组件RoadMap



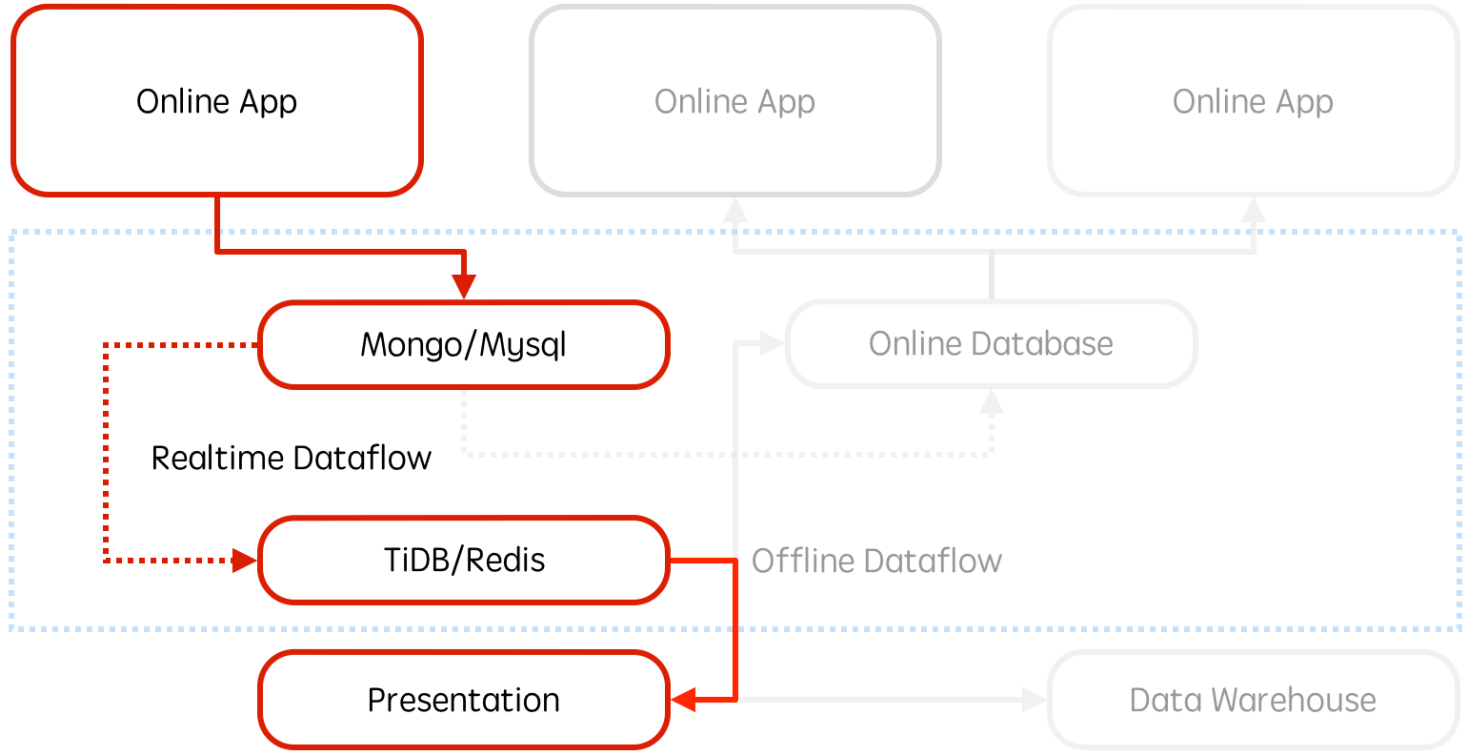
- 部署
  - 组件管理界面
  - 版本、机型选择（云主机、裸金属、自建机房）
  - 监控、报警、日志收集（机器级、应用级、业务级）
  - 跨区/跨云部署
  - 附属组件（如zk、lb、jmx_exporter等）部署
- 运维
  - 扩容、缩容、迁移（跨区迁移与机型升级）
  - 版本升级
  - 问题排查（日志分析）

数据服务组件RoadMap

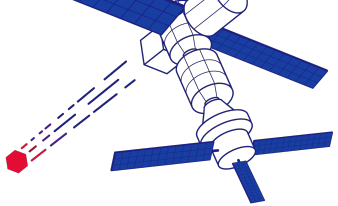


- 优化
  - 配置调优
  - 客户端代码调优
  - 二次开发
  - 三次开发|- -
- 其他
  - 安全
  - 审计
  - 服务化
  - 容器化

场景1：展示类业务

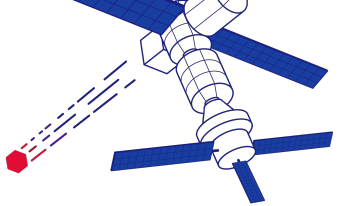

Online App
Online App
Online App
Mongo/Mysql
Online Database
Realtime Dataflow
TiDB/Redis
Offline Dataflow
Presentation
Data Warehouse

项目1：大促实时看板



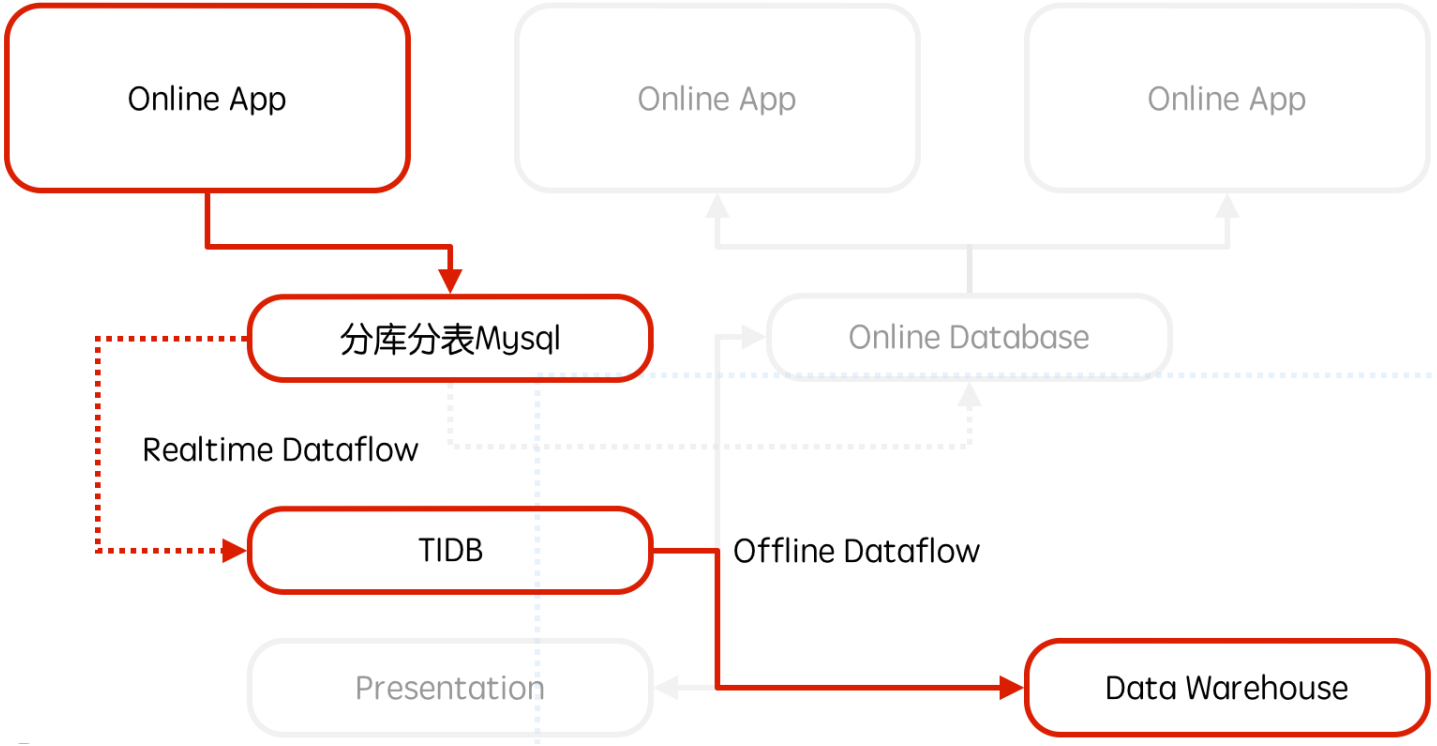
- 业务场景
  - 共需支持8个实时报表，QPS最高的报表写入均值5K，峰值接近200K
  - 每2s会有一个较大的聚合查询query
  - 部分聚合结果先写入redis, 再pop到tidb
- 集群规模
  - 10 tikv + 3 pd
  - 每个节点挂载3.5T*4块NVME SSD（事实证明，这个选型有缺陷）
- 实现细节
  - 采用多线程保证最终一致性的方式写入
  - batch insert size = 100时大致能达到吞吐和延迟综合最优
- 全程写入和查询稳定，写入时延<20ms，查询时延<1s

项目2：实时业务数据展示

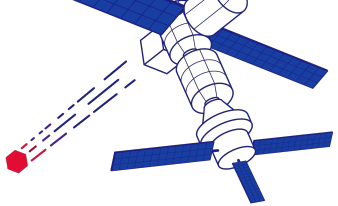


- 业务场景
  - 主要针对业务方实时问题分析的需求，其次需要作为离线ETL任务的数据源，同时预备改为线上服务
  - 共需支持超过100个Mysql/mongoDB数据库的实时展示，峰值总读写QPS超过500K
  - 按业务线拆分集群，但有部分核心表一式多份
  - 绝大多数为点查，亦有少数单表聚合查询和跨表join查询
- 集群规模
  - 现有10 tikv + 3 pd的集群3个，后续按需扩增
- 实现细节
  - 基于Canal进行oplog/binlog实时同步
  - 部分DDL场景仍需要手工介入
  - 多租户问题在展示层解决
- 逐步上线中，问题尚未完全暴露

场景2：分析类业务

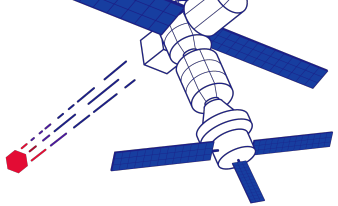

Online App
Online App
Online App
分库分表Mysql
Online Database
Realtime Dataflow
TIDB
Offline Dataflow
Presentation
Data Warehouse

项目3：分库分表Mysql ETL



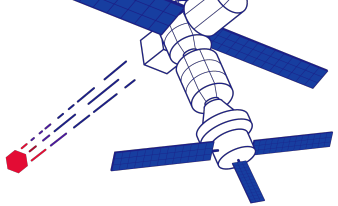
- 业务场景（以最大的表为例）
  - 上游10节点的MySQL，共计10000个分表，存量数据1000亿条左右，每日增量10亿+
  - QPS写入均值3000条/s，峰值接近10000条/s
  - 增量ETL任务，故离线库只需保留约30天的dtm表
  - 每日夜间（白天偶尔）会有基于sqoop的抽数任务触发
- 集群规模
  - 10 tikv + 3 pd
- 实现细节
  - 对mysql自增ID进行了处理
  - 对sqoop进行了部分对于tiDB的适配
  - 调整了tidb的max transaction size，以优化抽取效率

项目4：Mysql归档



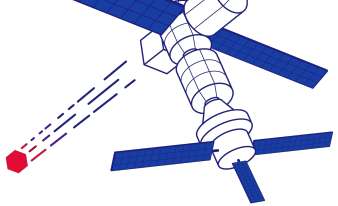
- 业务场景（以最大的表为例）
  - 主要为物流仓储部门的订单及衍生信息，上游Mysql按仓库分表，归档字段大致一致
  - 每月进行归档到TiDB的任务，数据量在数十亿
  - 对QPS没有太高要求
  - 与业务方暂定，过去一年半的记录存放在TiDB，更久远的记录会进一步归档至S3/Cos
- 集群规模
  - 25 tikv + 3 pd，未来还有扩容要求
- 实现细节
  - 对历史库数据进行了梳理
  - 对DDL操作进行了特殊逻辑处理
  - 定期进行数据校验

场景3：线上服务



- 已有一些业务接入hive -> TiDB的T+1级别数据服务
- 新上线业务的关系型数据库选型已经开始倾向于TiDB
- 接入实时读写数据服务的业务目前较少
  - 代码更改成本
  - 数据迁移成本
  - 运维成本
  - 技术栈成本

未来接入计划



- ETL方向
  - 支持更多事务隔离性要求低的场景
- 跨数据中心部署
- 自动化运维方向
  - TiDB on k8s
- TiFlash/Tidis/ClickHouse

Thank You !