OLAP神器之  ClickHouse

ClickHouse在京东的应用

目录

ClickHouse简介

- ✦列式OLAP数据库
- ✦不依赖于HDFS存储
- ✦扩展SQL接口
- ✦查询快

ClickHouse简介

✦MPP

✦SIMD

✦Code generate

✦Choose algorithm by case

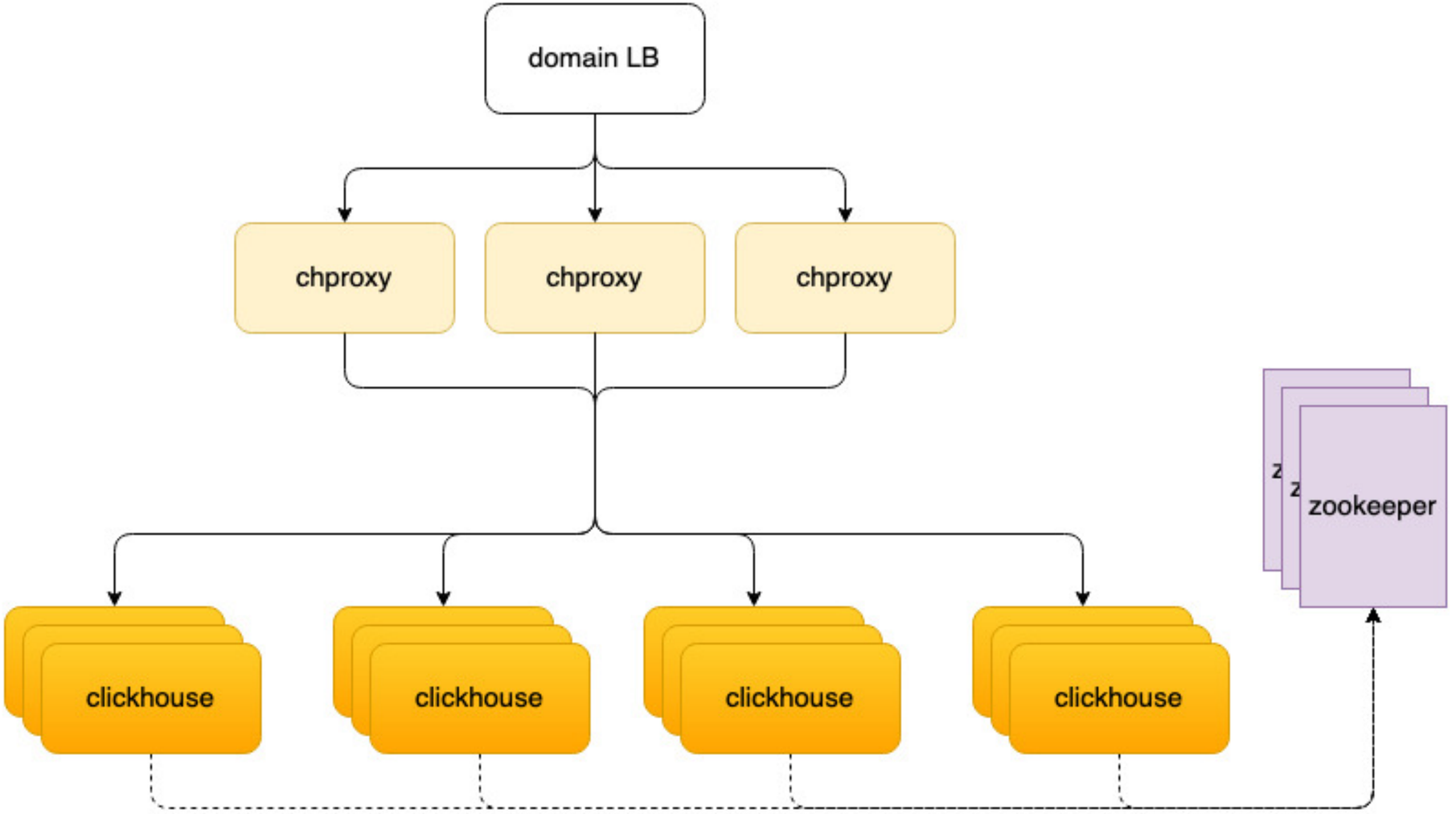
✦…

ClickHouse简介

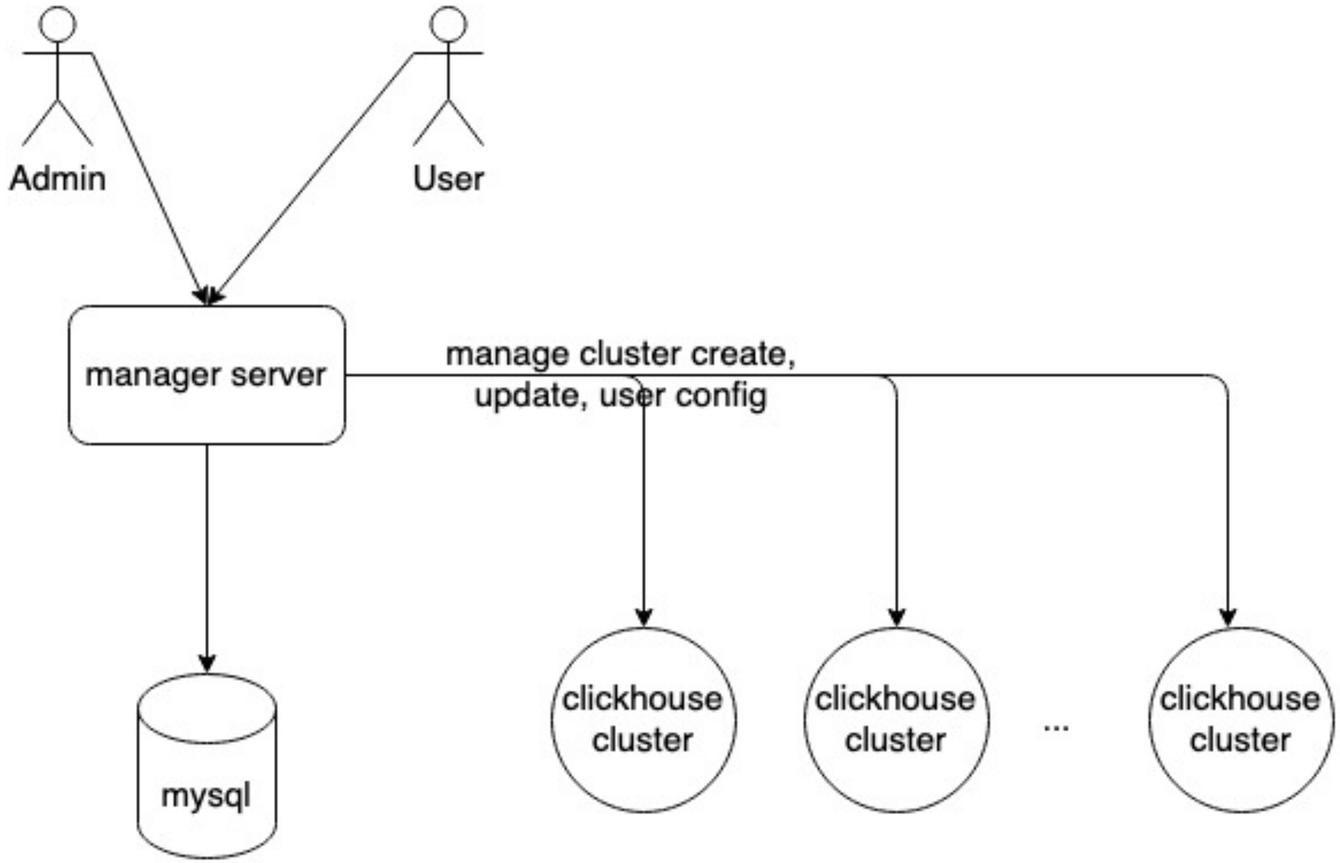
- No transaction
- Join performance
- Update/delete support
- Low concurrency

JD ClickHouse的应用

JD ClickHouse的应用



JD ClickHouse的应用



JD ClickHouse的应用

1.30+集群，1300+机器，2000+CK实例

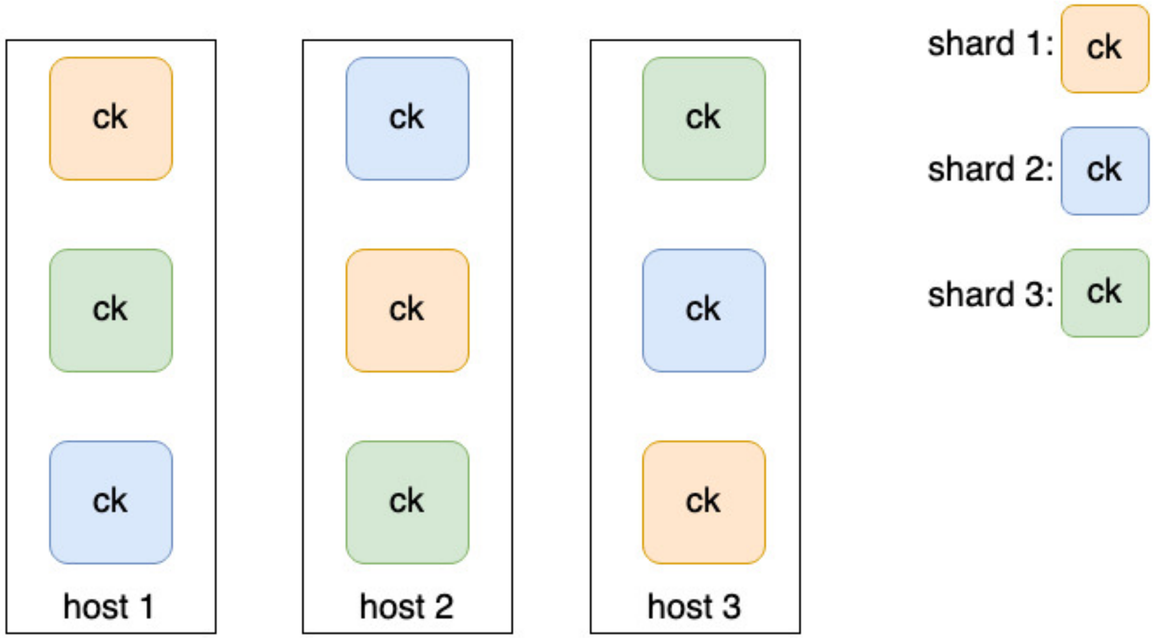
2.赋能100+业务

3.PB级数据

4.双十一峰值写入1300w/s，查询QPS 1400

- 商智
- 黄金眼
- 京东健康
- 7fresh
- 广告
- 物流
- 客服
- …

问题与方案



如果每台机器部署n个实例，那么n台机器就是一个Group，扩缩容时以Group为单位进行。

好处:
1. 扩缩容时需要迁移数据的CK实例数最少；
2. 数据交换基本只发生在Group内，减少相互影响；

问题与方案

1. 运维工具：
    1. 一键下线
    2. 节点替换
    3. 元数据一致性检查
    4. 集群迁移；
2. 管理平台：
    1. 集群申请与新建；
    2. 账号管理；
    3. 集群配置更新；

问题与方案

制定规范：

1. 根据数据量、查询QPS范围等确定集群规格；
2. 建库、建表规范；
3. 查询配额控制；
4. 大促演练；
5. CK使用的培训；

问题与方案

- 用户管理

```
    CREATE ROLE IF NOT EXISTS biz_admin ON CLUSTER X;
    GRANT ON CLUSTER X CREATE TEMPORARY TABLE, SOURCES ON *.* TO biz_admin;

    CREATE USER IF NOT EXISTS test ON CLUSTER X IDENTIFIED WITH double_sha1_password BY ‘1234’;

    GRANT ON CLUSTER X
        SHOW,
        SELECT,
        INSERT,
        ALTER,
        CREATE DATABASE, CREATE TABLE, CREATE VIEW,
        DROP,
        TRUNCATE,
        OPTIMIZE,
        SYSTEM MERGES, SYSTEM TTL MERGES, SYSTEM FETCHES, SYSTEM MOVES, SYSTEM SENDS, SYSTEM
REPLICATION QUEUES, SYSTEM SYNC REPLICA, SYSTEM RESTART REPLICA, SYSTEM FLUSH DISTRIBUTED,
    ON db.* TO test;

    GRANT ON CLUSTER X biz_admin TO test;

    -- add dictionary privilege
    GRANT ON CLUSTER X CREATE DICTIONARY, dictGet, DROP DICTIONARY ON db.* TO test;
    GRANT ON CLUSTER X SYSTEM RELOAD DICTIONARIES TO test;
```

问题与方案

- 磁盘故障、机器hang死时，所有的on cluster DDL都会阻塞然后失败
  - a. 一键下线工具
  - b. 通过SQL下线CK实例: SYSTEM PAUSE NODE x.x.x.x
- 读写未隔离，部分配额会相互影响
  - a. 细化配额管理
  - b. 读写账号分离

问题与方案

- 大查询导致CPU/Mem暴涨，服务不可用：
  - 配额限制，配置max_concurrent_queries_for_user，max_memory_usage_for_user，max_execution_time；
  - 督促部分业务添加缓存机制，降低大查询的并发

问题与方案

- 海量数据时zookeeper成为瓶颈，原因是承担的功能太多：

  1.一致性协调服务

  2.log_entry存储

  3.metadata存储

➡把log_entry从zk里剥离，RosksDB on raft

问题与方案

- TODO：
  - 慢查询诊断与隔离
  - 小查询加速
  - 存储冷热分离
  - SQL执行优化
  - …

展望

- 高可用优化
- 多租户隔离
- Join性能优化
- 秒级写入
- 事务支持
- HTAP
- …

Thanks!