



# 第十一届中国数据库技术大会

DATABASE TECHNOLOGY CONFERENCE CHINA 2020

## 架构革新 高效可控



北京国际会议中心 | 2020/12/21-12/23

# 网易DDB的单元化架构演进之路

周劲松

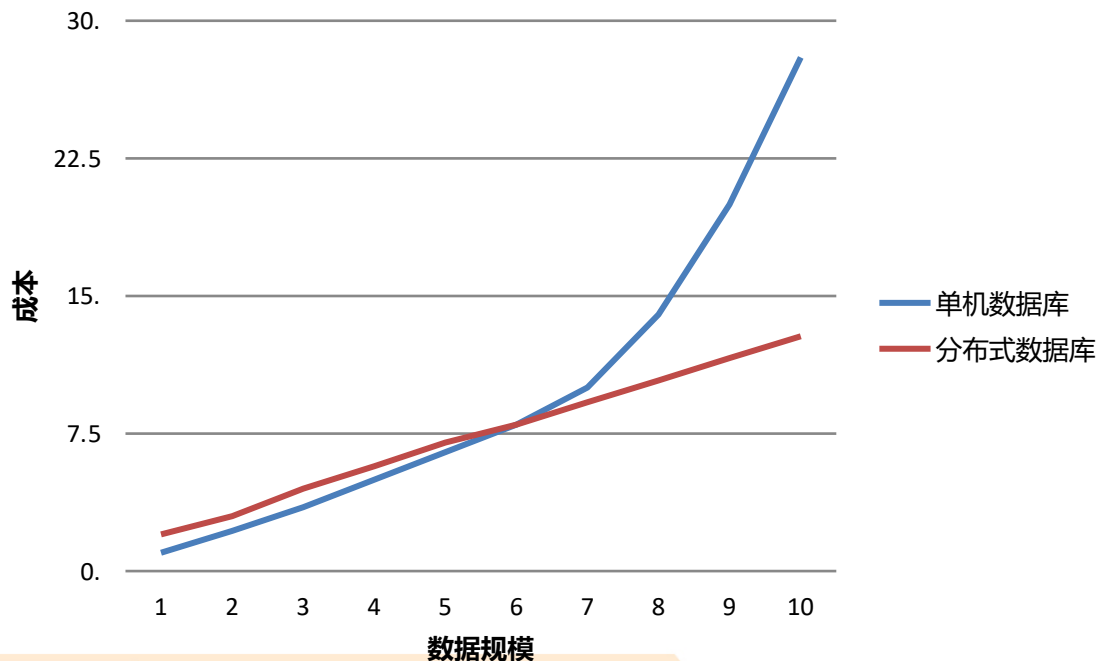


# 目录

- DDB简介
- 中间件到代理的转变
- 软件到平台的转变
- DDB的跨机房方案
- DDB的单元化方案
- 单元化的双向同步实现
- 单元化的数据一致性保障

## DDB：一步到位的分布式数据库

数据库成本曲线



PB级结构化数据存储

百万级qps

每日GB-TB数据增长

在线扩缩容

管理上千个数据节点

标准化的访问协议

## DDB发展历史

2006年开始，DDB为网易各大互联网产品提供透明分库分表服务。  
10年来不断完善，精益求精，是网易大体量互联网产品的立身之本。

2006年  
博客上线

简单SQL兼容  
部分管理功能

2008年  
V2.0发布

SQL兼容扩充  
在线扩容功能  
图形化管理工具

2010年  
V3.0发布

分布式事务  
在线修改表结构  
SQL兼容扩充  
管理功能完善  
集群规模上千

2012年  
V4.0发布

多语言支持  
SQL统计功能  
云计算DDB

2018年  
V5.0

架构简化  
服务拆分  
SQL兼容度进  
一步扩充  
多机房方案

# DDB核心功能

## 数据分布

两级映射  
自定义哈希函数

## 标准化

SQL92 高兼容  
全局自增ID  
支持explain  
数据导入导出  
MySQL通信协议

## 分布式事务

实现2PC协议  
数据高一致  
用户透明  
自动识别

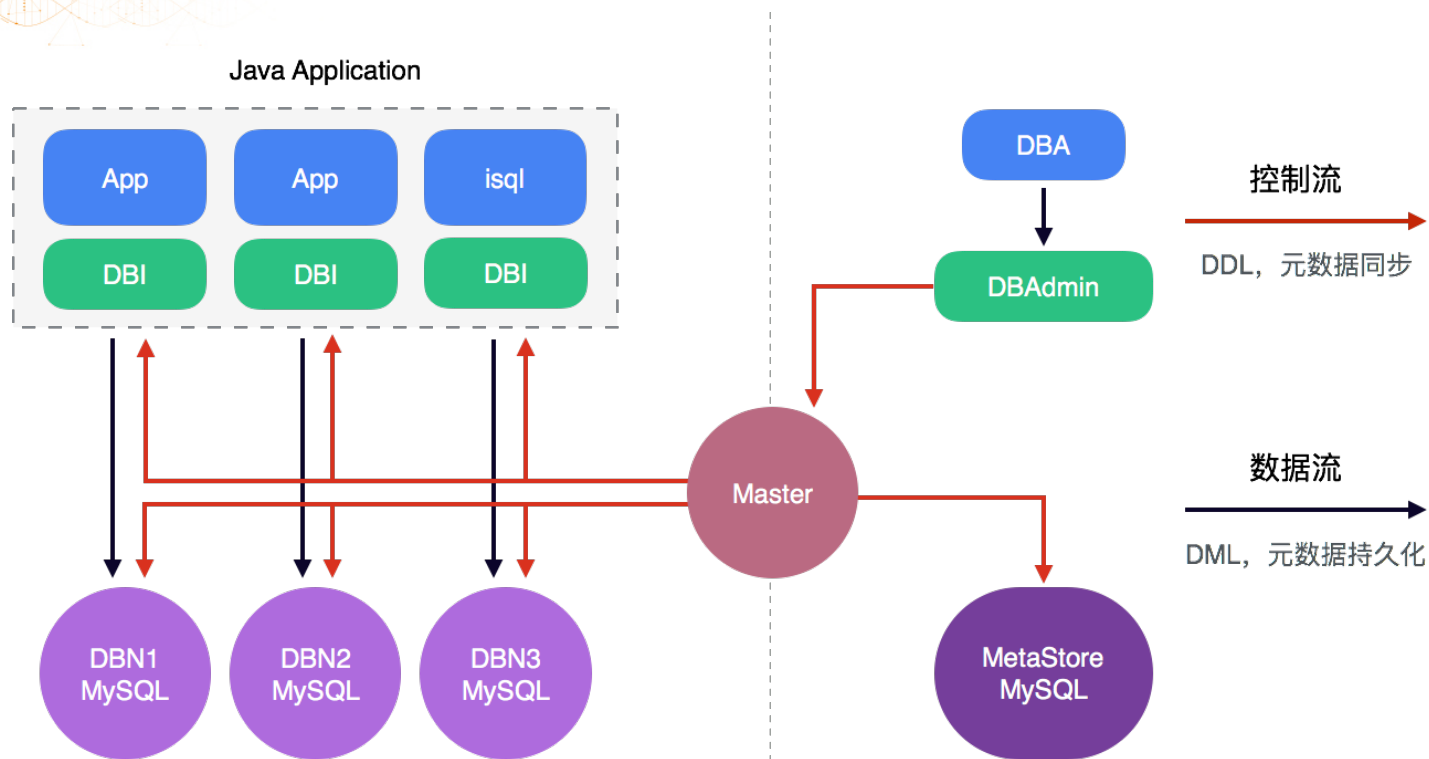
## 多机房方案

读写分离  
两地三中心  
单元化

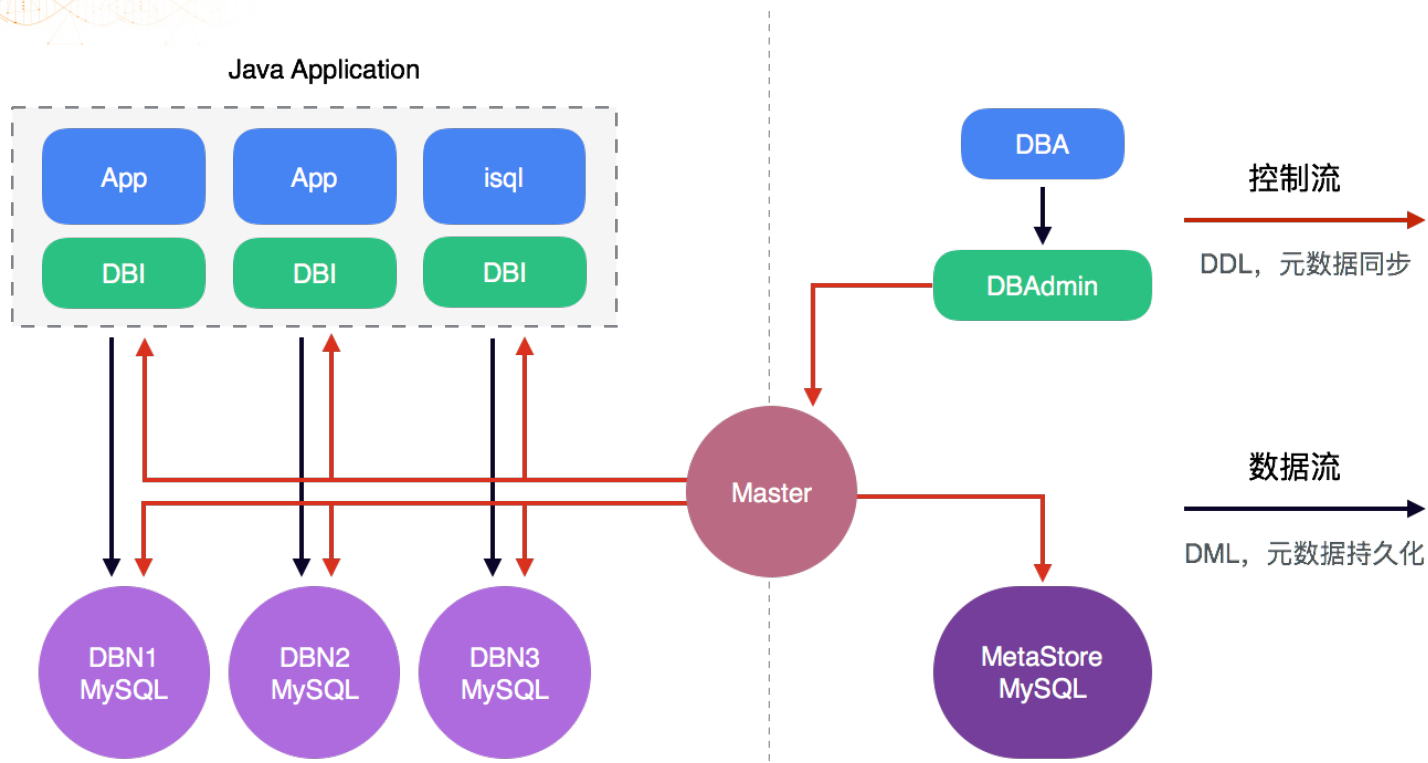
## SQL统计

SQL模式统计  
SQL频度统计  
慢SQL统计  
多维度QPS统计

## DDBV1-V3架构：DBI模式

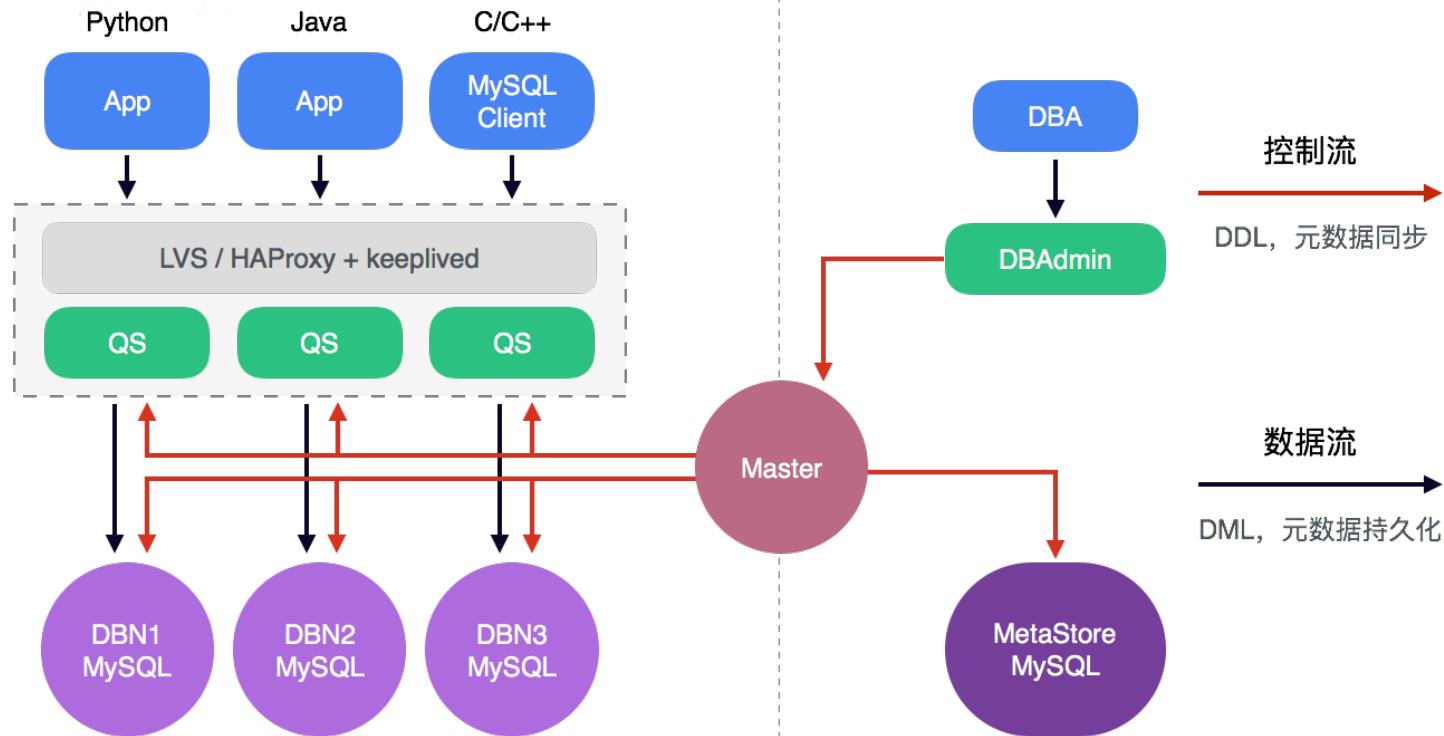


## DDBV1-V3架构：DBI模式

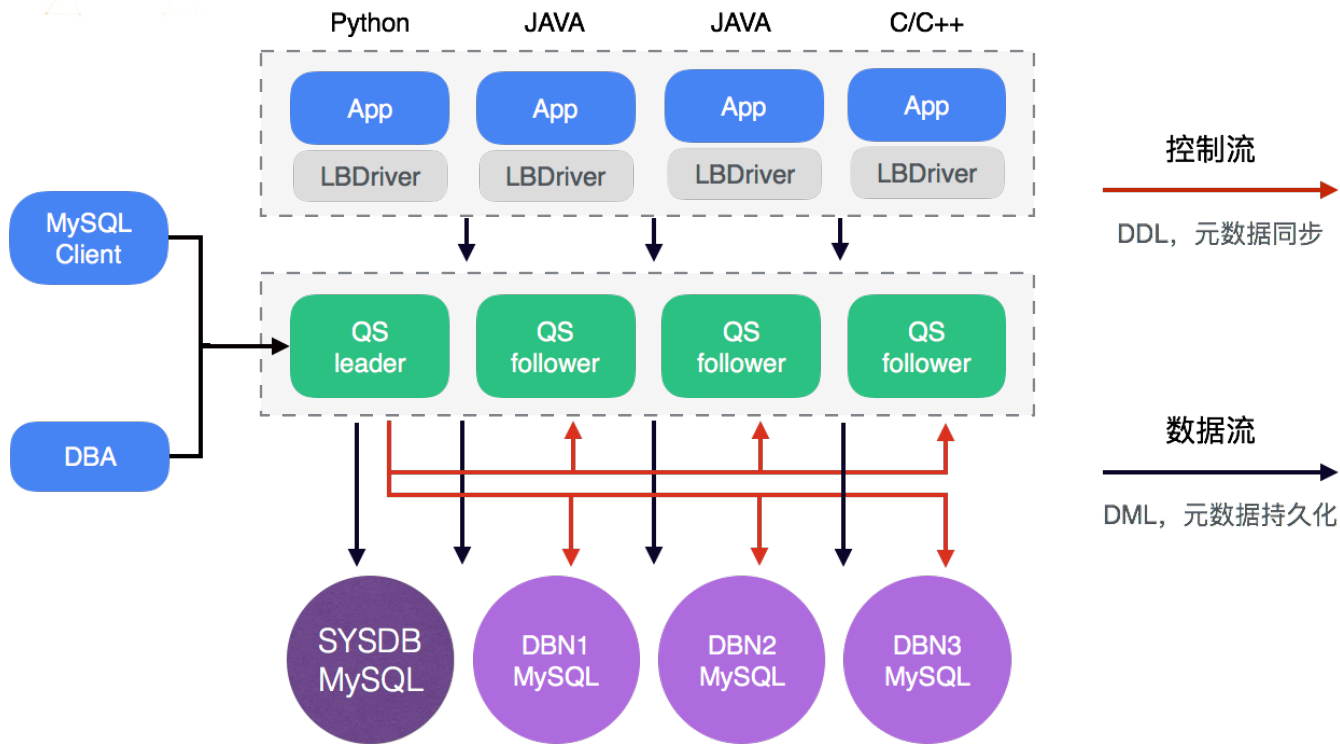


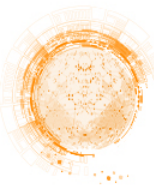


## DDBV4架构：QS模式

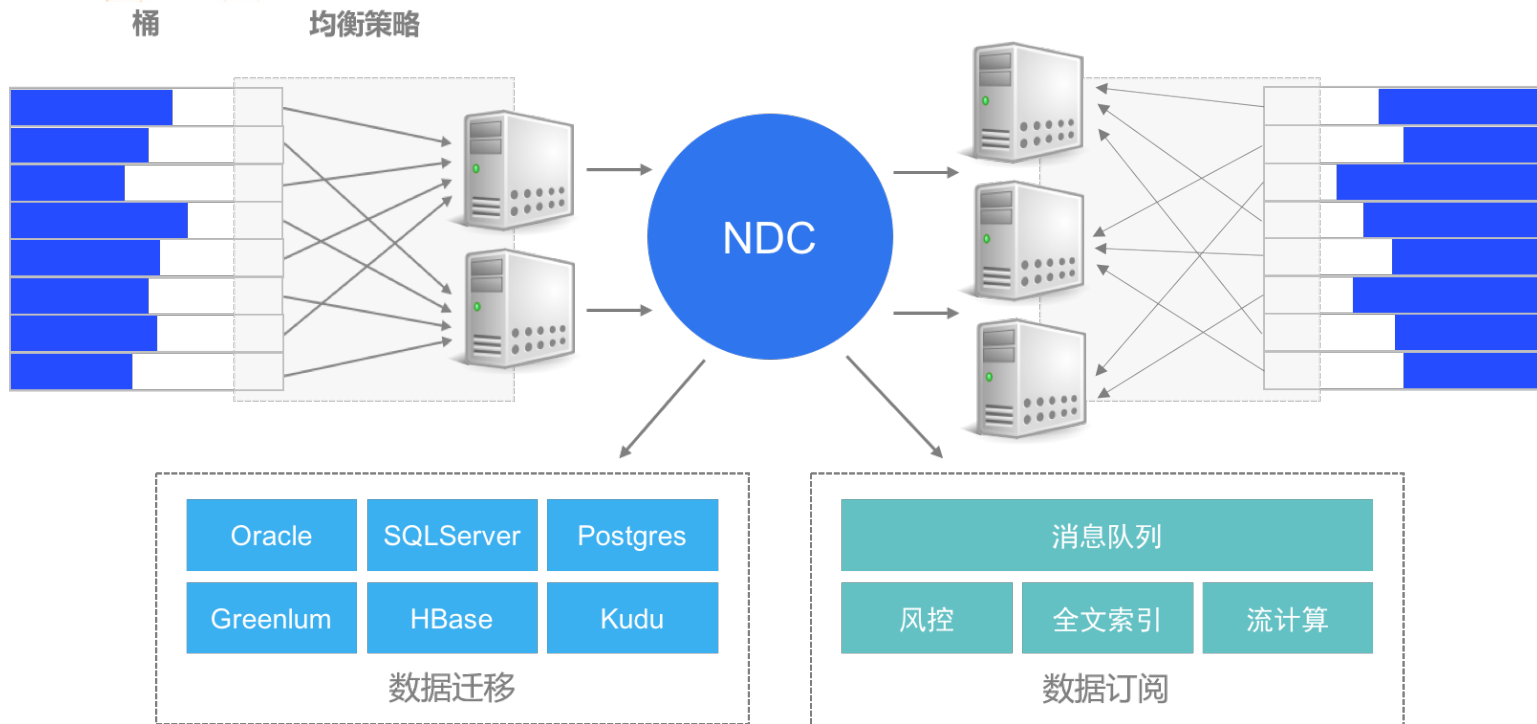


## DDBV5架构：去中心化





## DDBV5架构：NDC服务拆分





架构革新 高效可控  
第十一届中国数据库技术大会  
DATABASE TECHNOLOGY CONFERENCE CHINA 2020

# DDBV5架构：平台化

DDBAdmin

分布式数据库

单机数据库

admin

退出

DDBAdmin

cm-commons-... >

集群概览

命令行 ×



cm-commons-user-device

DDB 版本: 5.0.3

创建时间: 2020-02-07 17:11:56

集群状态: 可用

入口地址: 10.194.59.132:6000

集群设置

事件与报警

SQL 统计

集群信息

查询节点数 2

数据分库 8

总连接数 121

活跃连接数 0

后端连接数 280

活跃后端连接数 0

总连接数



吞吐率



响应时间



集群概览

库表管理

数据导入

数据导出

语句审核

查询节点

分库管理

数据镜像

分区策略

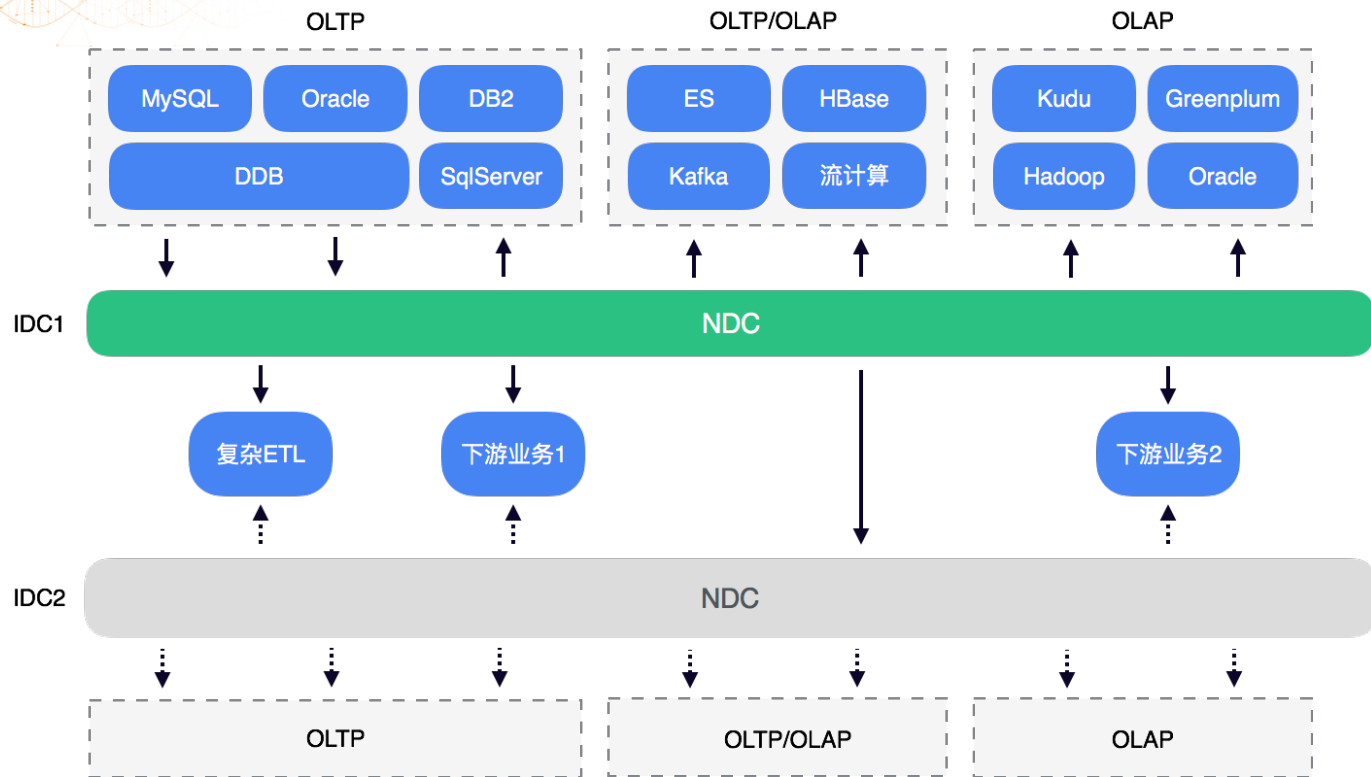
账号管理

数据迁移

定时任务

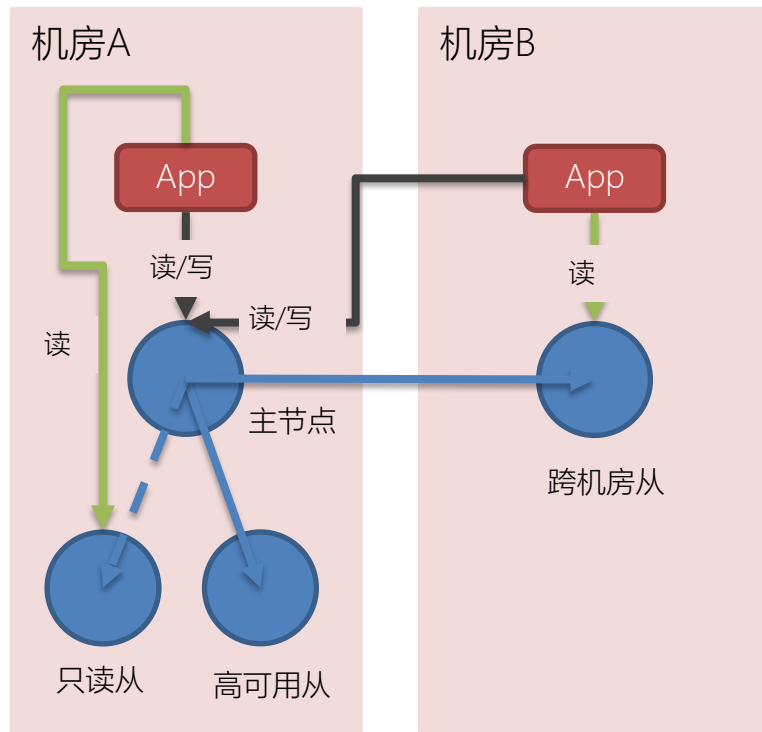
触发器

## DDBV5架构：跨机房



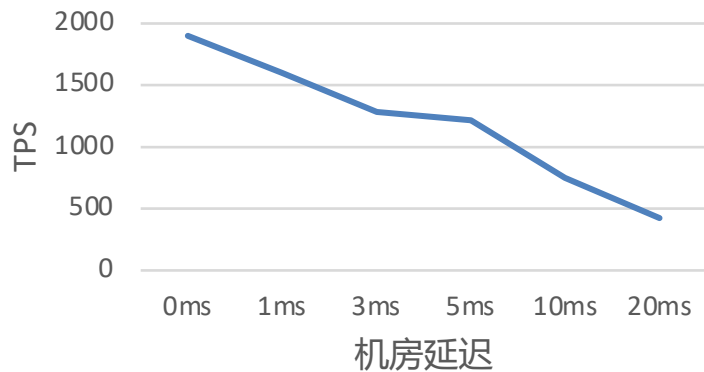
## DDBV5架构：跨机房

- 收益
  - 宿主机级别容灾
  - 读写分离
- 适用场景
  - 业务发展初期
- 问题
  - 容灾级别低

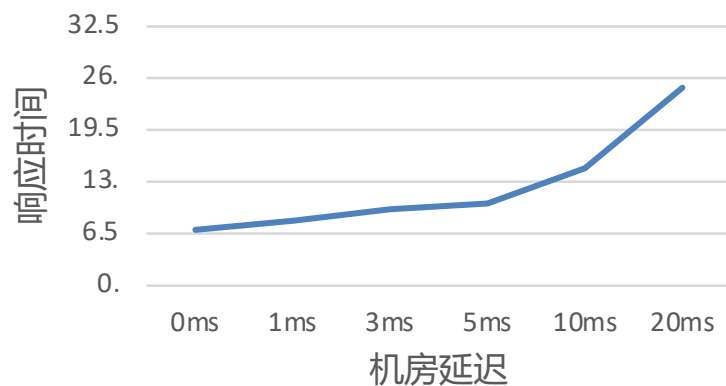


## 机房延迟影响

### 机房延迟影响TPS



### 机房延迟影响响应时间



## DDBV5架构：单元化

### ■ 收益

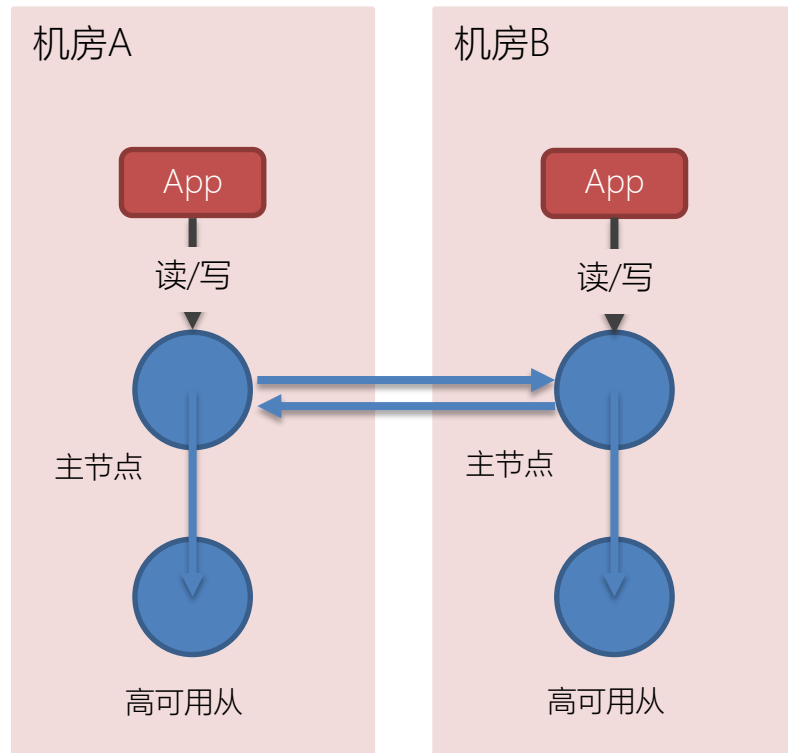
- 跨机房级别容灾
- 读写本机房

### ■ 适用场景

- 业务发展后期

### ■ 问题

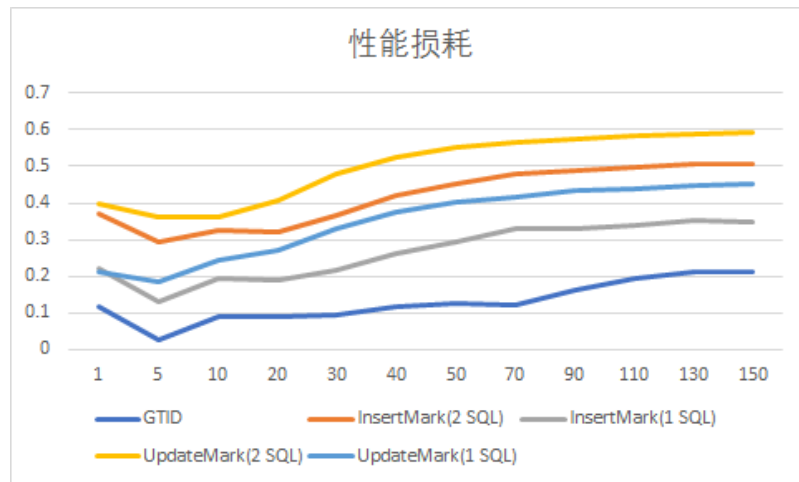
- 业务改造代价大





## DDB单元化：解决回环复制

方案	优点	缺点
引入额外字段	性能影响小 跨数据源	对业务表有侵入
GTID	性能影响小 无侵入	只试用MySQL
事务中引入额外SQL	跨数据源 侵入小	性能影响较大





## DDB单元化：解决数据冲突

### ■ 加入版本字段

#### ■ update\_time

### ■ 改写同步语句

#### ■ Where update\_time <= \${data\_update\_time}

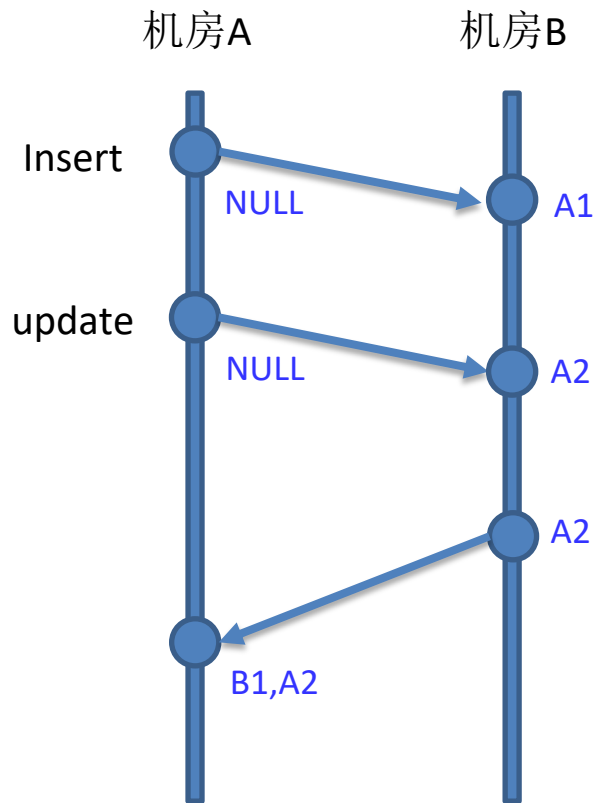
```
CREATE TABLE `user` (  
    `id` int(11) NOT NULL,  
    `name` varchar(256) DEFAULT NULL,  
    `update_time` timestamp(6) NOT NULL  
    DEFAULT CURRENT_TIMESTAMP(6)  
    ON UPDATE CURRENT_TIMESTAMP(6),  
    PRIMARY KEY (`id`),  
) ENGINE = InnoDB CHARSET = utf8mb4
```

```
UPDATE `user`  
    SET name = 'john', update_time = '2020-  
    06-09 13:27:32.237'  
WHERE id = 1 AND  
    update_time <= '2019-05-10 13:27:32.237'
```



## DDB单元化：发现数据双写

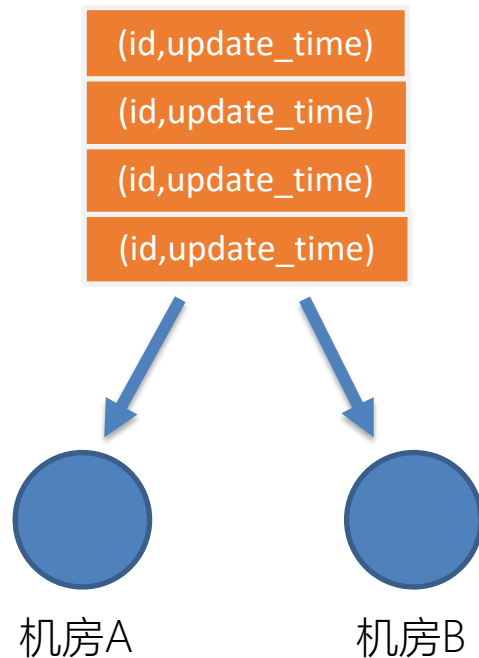
```
CREATE TABLE `user` (  
  `id` int(11) NOT NULL,  
  `name` varchar(256) DEFAULT NULL,  
  `ndc_version` varchar(256) DEFAULT NULL  
  `update_time` timestamp(6) NOT NULL  
  DEFAULT CURRENT_TIMESTAMP(6)  
  ON UPDATE CURRENT_TIMESTAMP(6),  
  PRIMARY KEY (`id`),  
) ENGINE = InnoDB CHARSET = utf8mb4
```



## DDB单元化：实时数据校验

### ■ 校验等级

- 双写数据校验
- 冲突数据校验
- 风险数据校验





# DDB单元化：管理平台

网易数据运河

平台管理

环境: 全部

项目: 全部

hzzhoujingsong

退出

源端管理

目标端管理

同步任务

订阅任务

双向同步

校验任务

执行节点

创建任务

搜索任务名称

任务名称

anti\_usersocial\_...

## 详细信息

任务ID	25	任务名称	anti_usersocial_multi_hz_jd	任务状态	● 运行中
创建时间	2020-11-23 14:07:07	全量同步	无	增量延迟	945 ms
项目	music				

刷新

启动

停止

删除

详情

配置

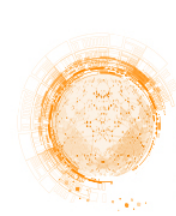
复制

屏蔽报警

## 单向同步

请输入地址

<input type="checkbox"/> 同步方向 ▾	源端地址	目标地址	任务状态 ▾	增量延迟	操作
<input type="checkbox"/> anti_ddb_music_multijd_7777 -> anti...	10.196.51.39:4332	10.200.133.185:...	● 运行中	945 ms	<a href="#">详情</a> <a href="#">回退</a>
<input type="checkbox"/> anti_ddb_music_multijd_7777 -> anti...	10.196.51.39:4331	10.200.133.185:...	● 运行中	731 ms	<a href="#">详情</a> <a href="#">回退</a>
<input type="checkbox"/> anti_ddb_music_multihz_7777 -> anti...	10.200.133.183:...	10.196.48.25:8888	● 运行中	735 ms	<a href="#">详情</a> <a href="#">回退</a>
<input type="checkbox"/> anti_ddb_music_multihz_7777 -> anti...	10.200.133.183:...	10.196.48.25:8888	● 运行中	301 ms	<a href="#">详情</a> <a href="#">回退</a>



## – DDB十年架构变迁

- V1-V3: 驱动模式，满足JAVA应用所需
- V4: 代理模式，多语言支持，增强可用性和运维能力
- V5: 平台模式，精简掉lvs和master，面向租户和IDC的解决方案

## – DDB单元化

- 版本字段发现数据双写
- 时间字段解决数据冲突
- 实时数据校验发现数据不一致

# THANKS

