



云数据库中间件的前世今生

曹宗南

LEADING NEW ICT

内容

- 单机数据库的扩展之路
- 数据库中间件演进
- 实践经验

某银行生活类APP

每天2小时集中写入数据量**1到2亿**

性能瓶颈：写入耗时超过**1小时**

容量瓶颈：单表数据大于**20亿**

解决方案：

硬件升级？成本高、有天花板

单机数据库遇到的瓶颈

数据分片

4个数据库中间件

16个MySQL实例

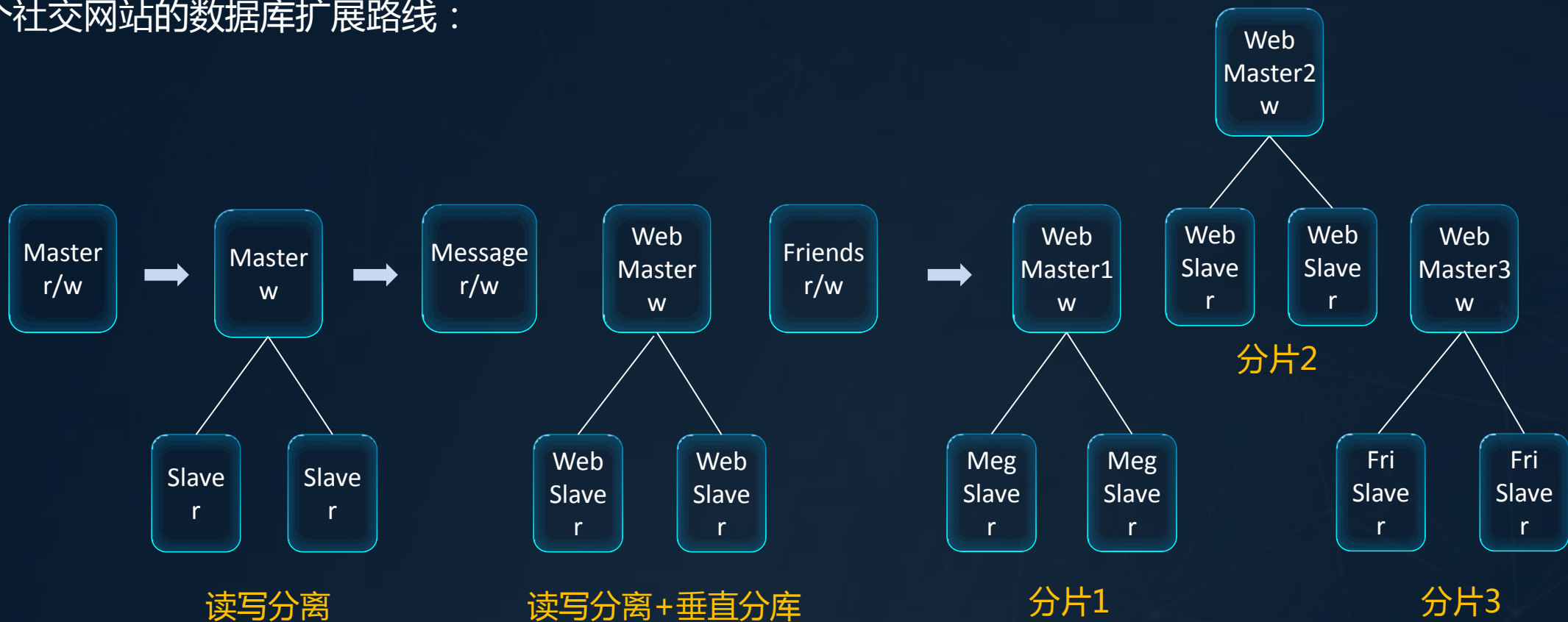
128个分片

解决效果：数据写入~~1小时~~？

No! 只要**5分钟**

数据库分布式扩展思路

一个社交网站的数据库扩展路线：

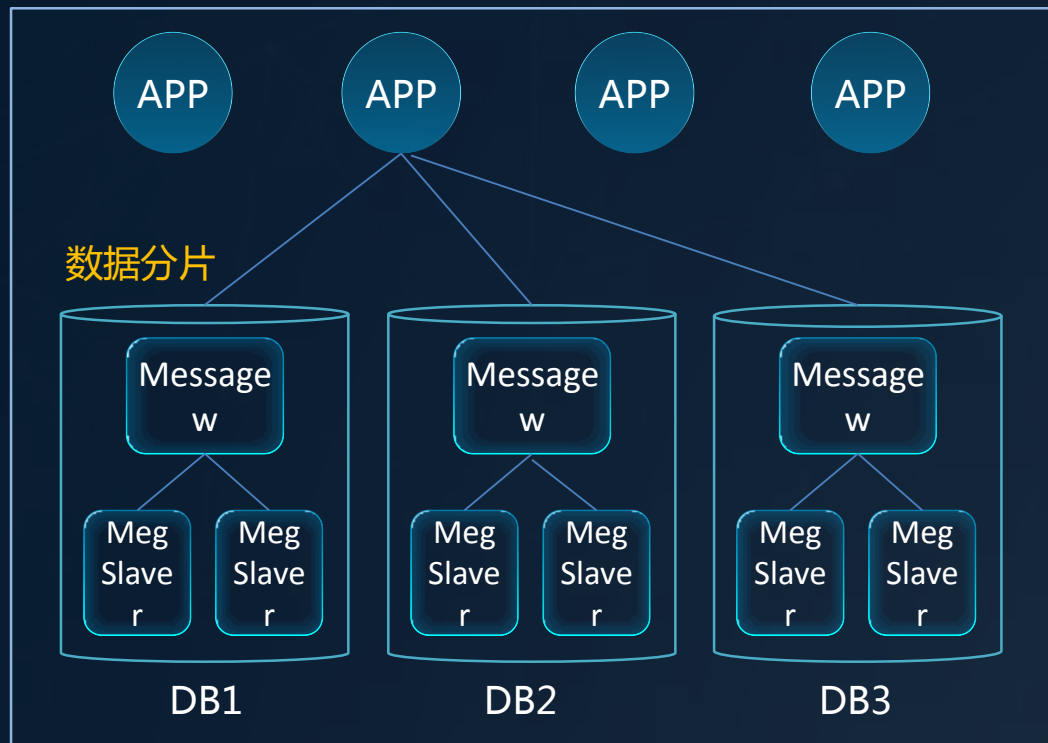


用数据分片来解决数据库扩展的瓶颈



常见数据分片方案

应用层分片方案



如当当的Sharding-JDBC，淘宝的TDDL等

- 优势：直连数据库，额外开销少
- 劣势：1) 无法实现连接数共享
2) 应用侵入方式，后续升级更新数量庞大，运维成本高
3) 多数只支持Java语言

中间件分片方案

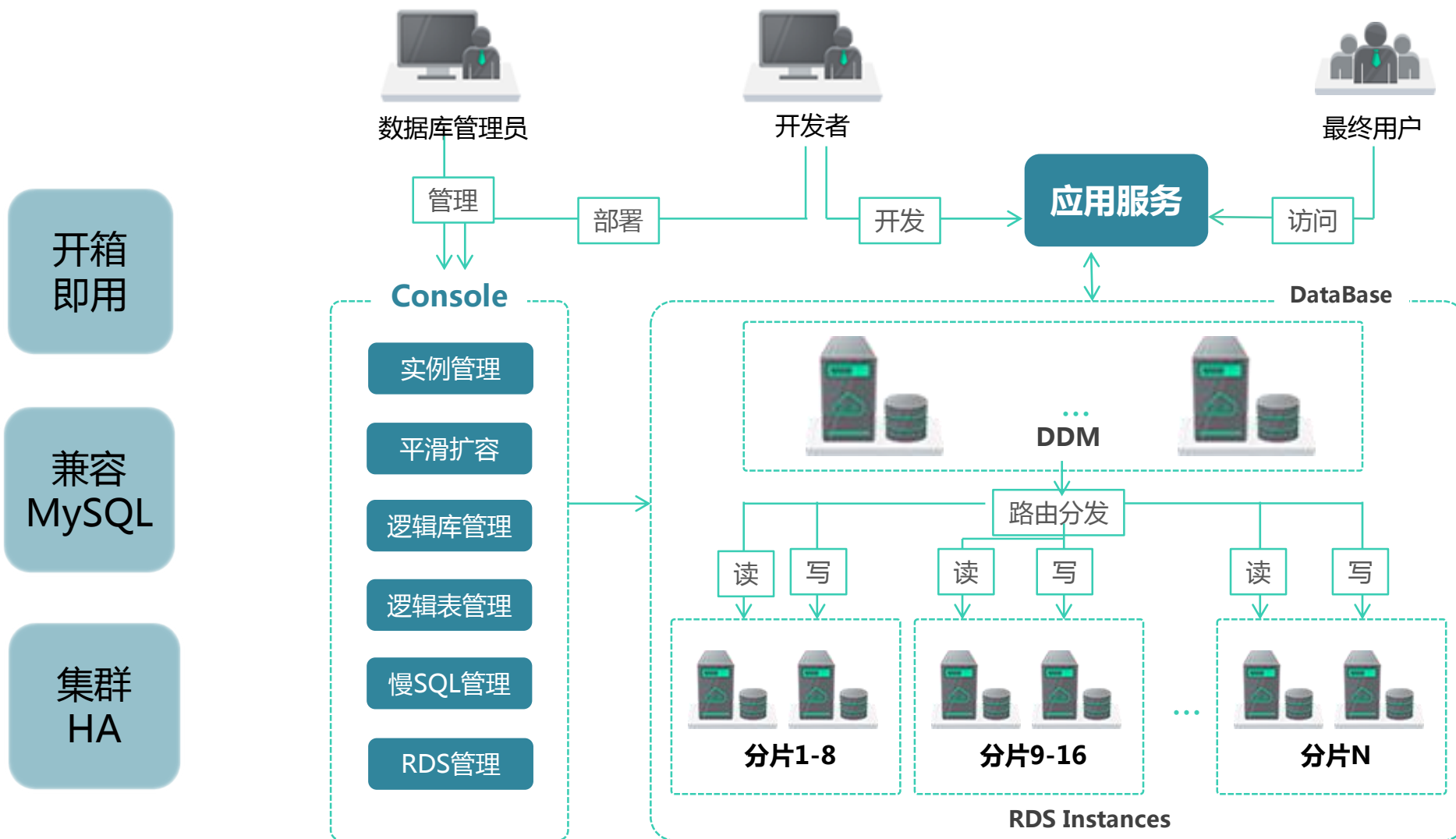


如开源的MyCAT，Cobar等

- 优势：1) 标准MySQL协议，**应用0改动**，与语言无关
2) 对应用**完全透明**地进行数据库扩展
3) 通过后端数据库连接**共享复用**
- 劣势：可能存在额外的时延

广告时间：DDM（叮当猫）

华为云上的分布式数据库中间件



内容

- 单机数据库的扩展之路
- 数据库中间件演进
- 实践经验

开源的那只猫  已经很nice

但是总有不完美的地方：

无法平滑扩容

分片算法太多，选择困难

数据库连接数问题

join和复杂sql问题

Buffer Pool不够用



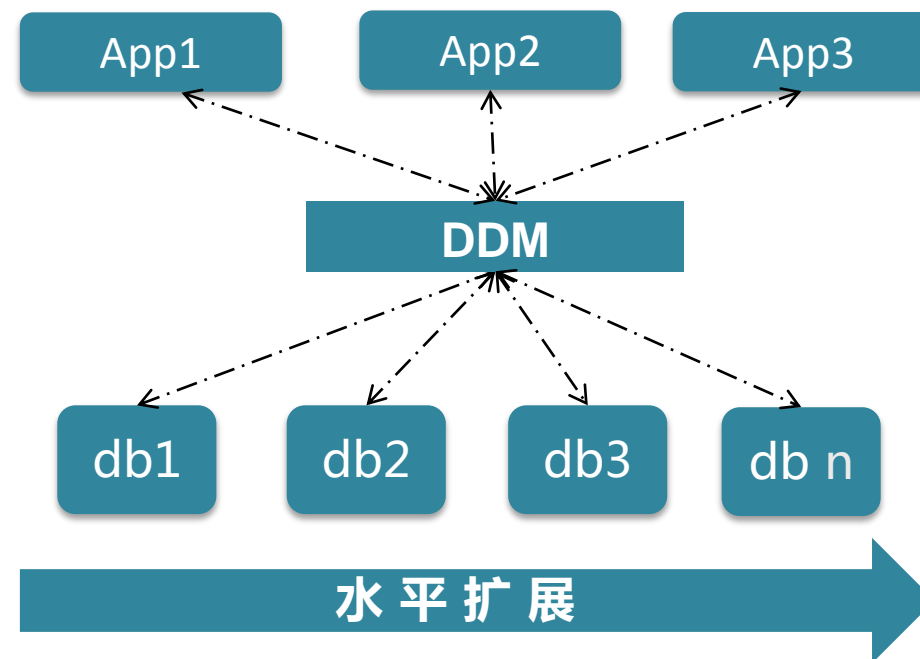
平滑扩容设计

问题：

业务发展导致原有分片达到瓶颈，增加RDS（MySQL）
需要迁移数据

传统解法：

1. 停机迁移：对业务影响大，不支持增量
2. 应用双写：开销大
3. 整库迁移：预留好分片个数，将整个分片库迁移到新的RDS上，分片个数无法扩展

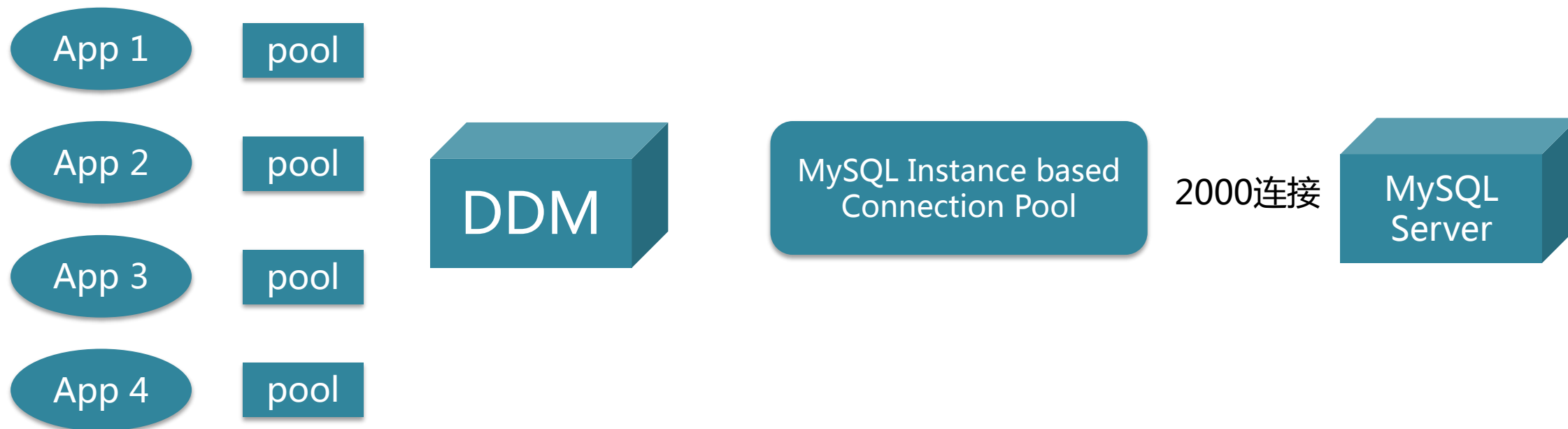


DDM实现数据库水平扩展，**数据自动重分布**：

- 无限扩展
- 全自动化
- 业务影响小

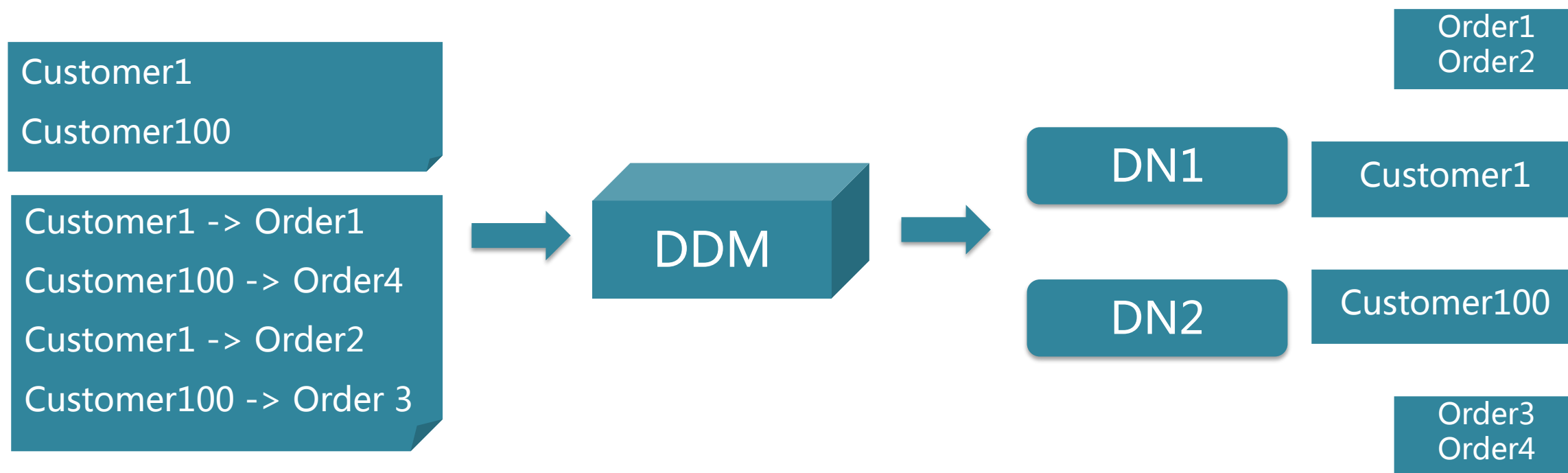
连接数收敛设计

问题： 1.业务微服务拆分后，每个微服务分到连接数很少。2000个连接100个微服务，每个微服务20个连接。
2.前端连接10000远大于后端2000，并发多取不到可用连接
3.针对不带拆分条件sql，需要路由到所有分片，一次需要申请分片个数连接，经常几十并发就报错。



join路由优化

问题：遇到不少join sql返回错误返回结果



内容

- 单机数据库的扩展之路
- 数据库中间件演进
- 实践经验

数据拆分标准如何判断

并发很多

系统很慢

备份慢、DDL慢

存储量上T

单表记录超千万

拆分后是否能提升性能？拆分标准

真实案例：

慢sql过多拖垮MySQL、数据量不大

单表拆分标准由来

判断标准是否需要与时俱进

MySQL最新版测试单表上5千万查询性能好像
没有明显下降

某银行案例.....



sql不带分片条件会怎样

响应时间？

整体吞吐量？

**update orders
set name= '123'**



orders@host1

orders@host2

.....

orders@host16

什么样的sql才是可线性扩展

鼠标点点加个实例就能线性加性能
想想就有点小激动



假设：

orders以order_id作为拆分键

user以user_id为拆分键

1. update orders set name= '123'
2. select * from orders where **order_id=100**
3. update orders set name= '123' where **order_id in (100,200,300)**
4. update orders set name= '123' where **order_id between 100 and 1000**
5. select... from user join orders on **user.user_id = orders.user_id** where **user.user_id=100**
6. select... from user join orders on **user.user_name = orders.user_name** where **user.user_id=100**

不建议使用SQL：不具备扩展性



那么拆分键如何选择

拆分键关系是否能线性扩展
选**最频繁的或者最重要的查询条件**

ER分片、Table Group：**有关联关系，选用相同拆分键。**

好处多多：join、分布式事务

全局表：变身多份

单表：数据量太少，就不拆了吧



想要多个拆分键怎么办

字段拼接 ？

一个分片算法支持多个字段？

Sql中查询条件带1个、多个怎么办？

空间换时间？

性能极致优化

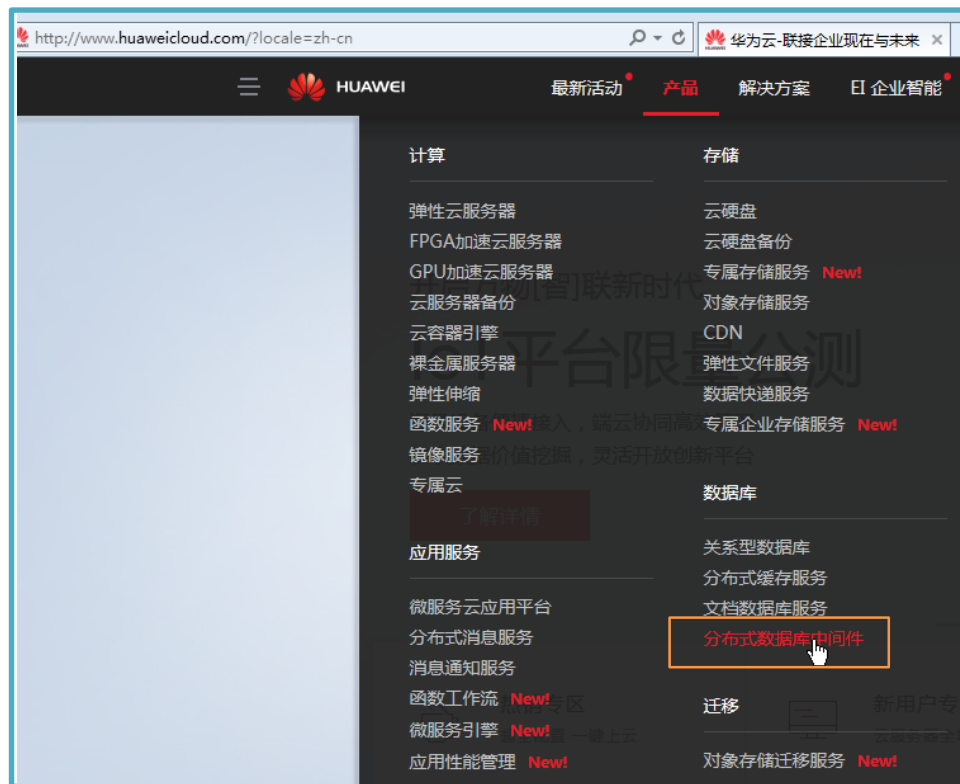
分布式事务

SQL兼容性

OLAP



DDM目前免费，欢迎大家体验



分布式数据库中间件 DDM

DDM (Distributed Database Middleware) 专注于解决数据库分布式扩展问题，突破了传统数据库的容量和性能瓶颈，实现海量数据高并发访问。

免费 [了解详情 >](#)

立即使用

产品优势

- 弹性**
 - 自动水平拆分；业务不中断平滑扩容；一键线性扩展。
- 性能**
 - PB级数据量访问；十倍于单机数据库连接数；百万级高并发。
- 可靠**
 - 集群高可用，秒级故障自动恢复；强一致性与最终一致性事务支持；多种安全策略保护数据库和用户隐私。
- 易用**
 - 兼容MySQL协议；业务代码零改动，透明读写分离；控制台轻松运维。



THANK YOU

LEADING NEW ICT