达梦数据库

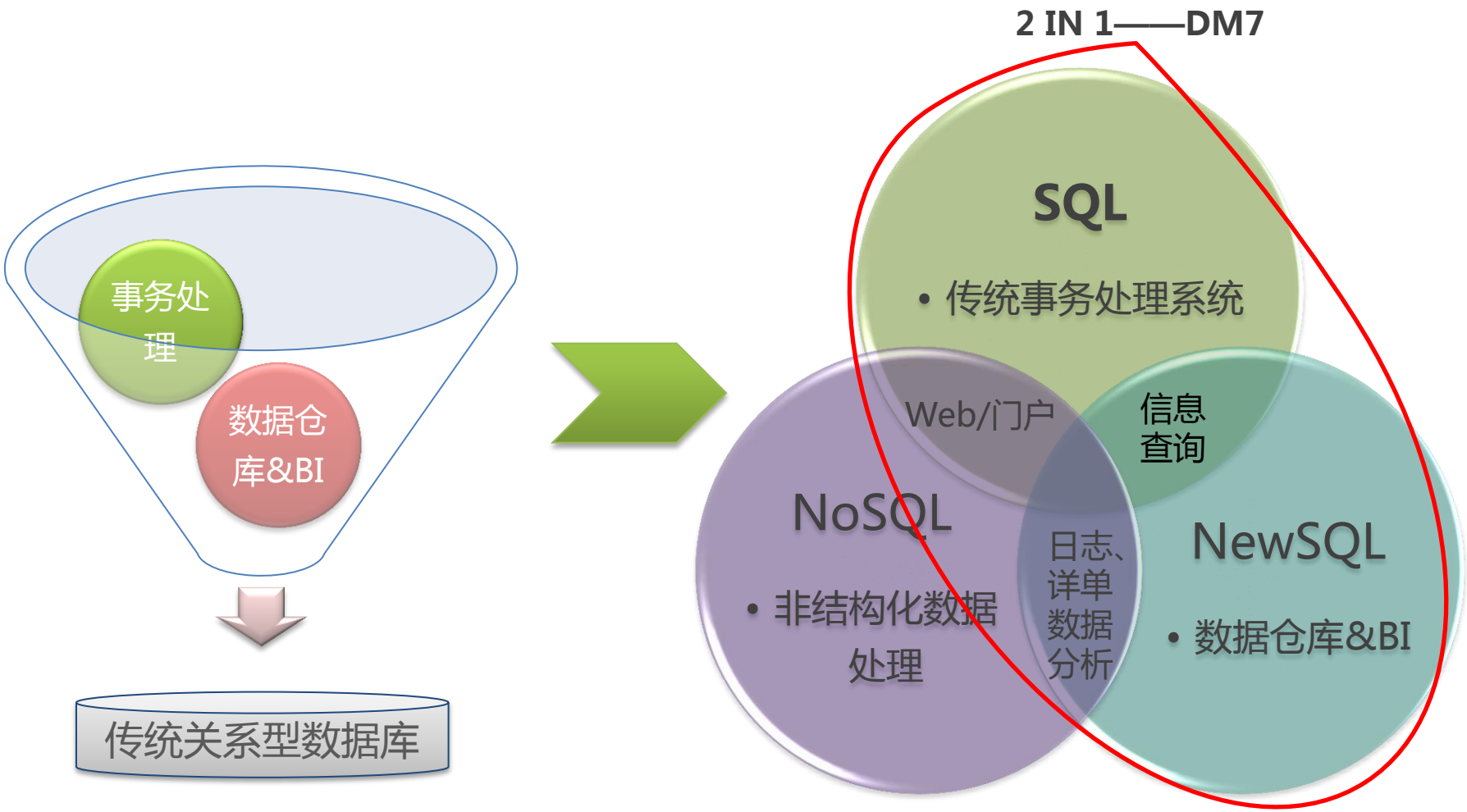
不只是事务处理

——DM7的跨界应用与改进实战

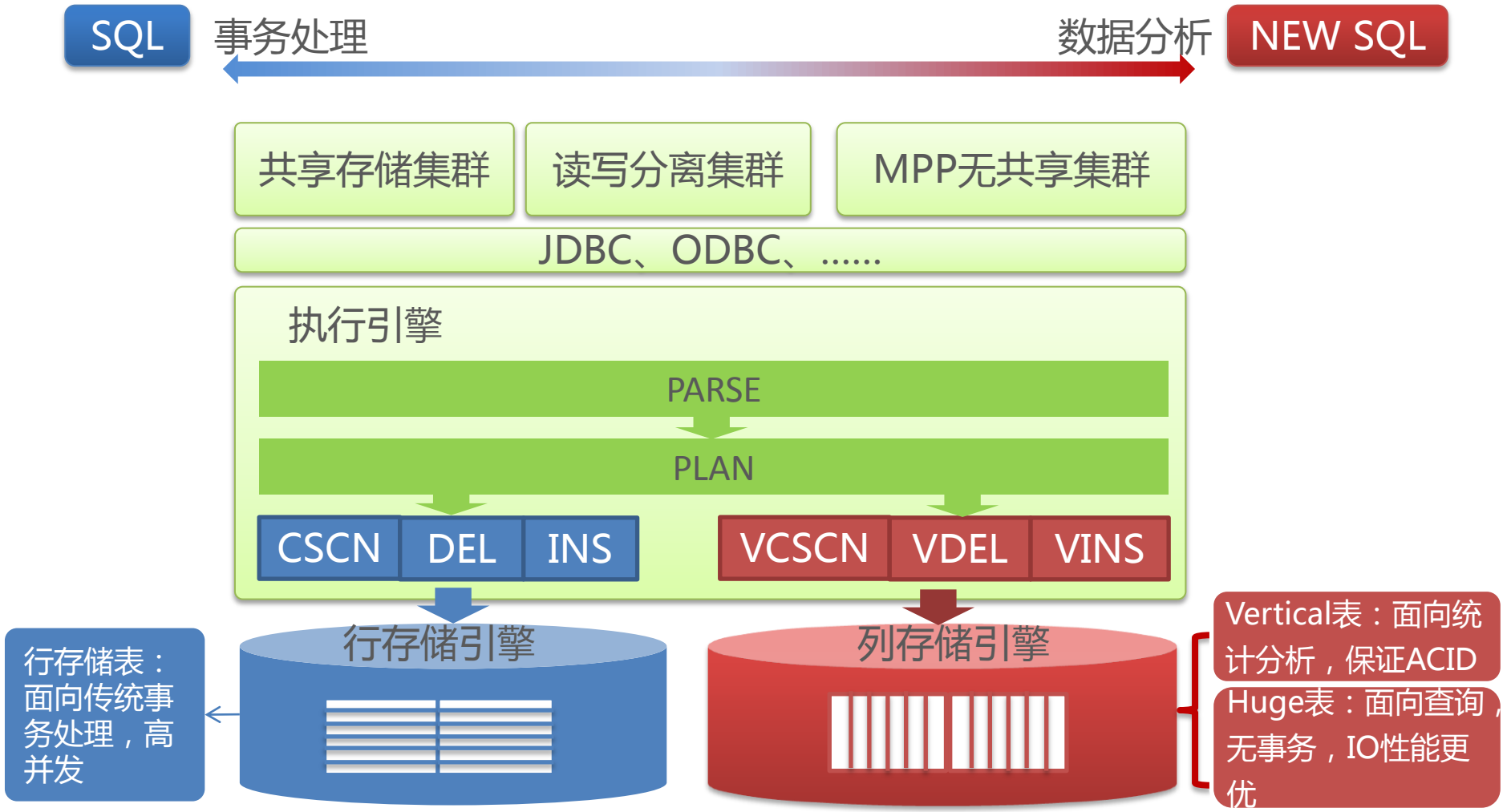
演讲人：周淳

www.dameng.com

数据库技术发展


事务处理
数据仓库&BI
传统关系型数据库
2 IN 1——DM7
SQL
• 传统事务处理系统
Web/门户
信息查询
NoSQL
• 非结构化数据处理
日志、详单数据分析
NewSQL
• 数据仓库&BI

DM7支持“跨界”的体系架构



达梦数据库

案例分享

事务处理

- 一体化调度运行管理系统——国家电网某省电力公司
- 财务共享服务平台——中国铁建

分析应用

- 话单综合分析系统——某运营商

混合负载

- 数字证书综合统计查询系统——公安部

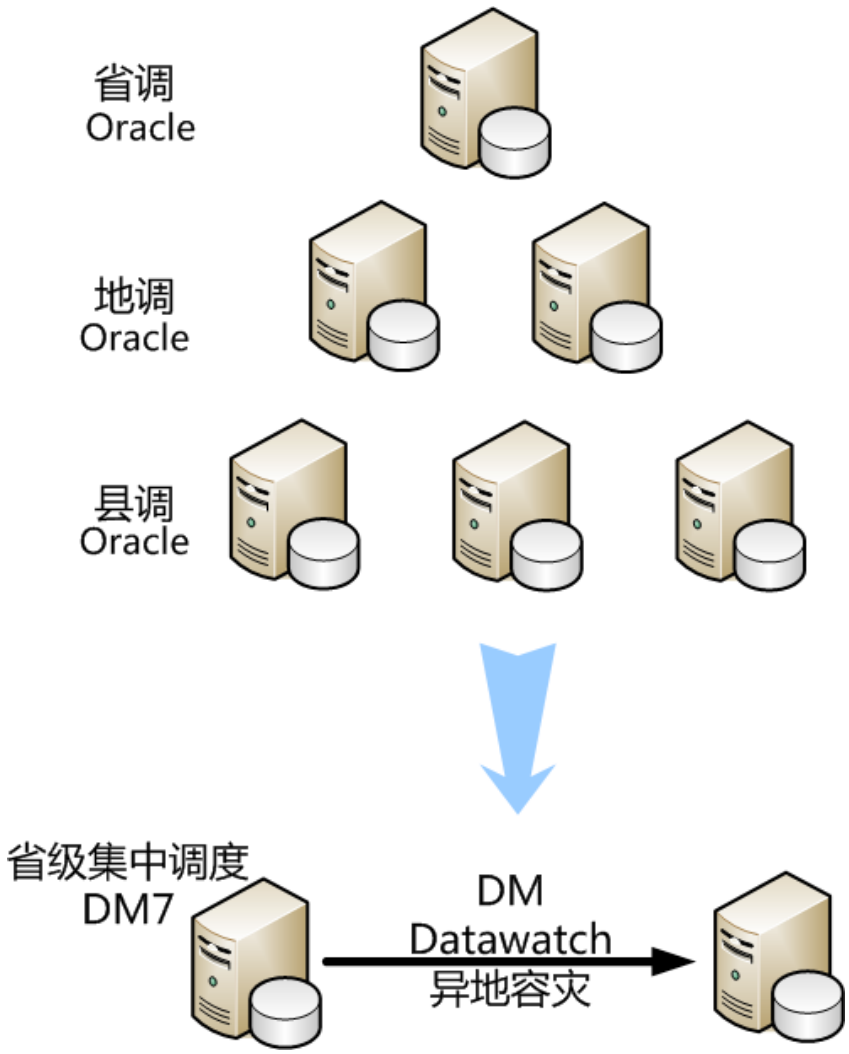
事务处理案例

省地县一体化调度运行管理系统

- 高并发：5000用户，峰值1159个数据库连接
- 大量更新+汇总信息查询

财务共享服务平台

省地县一体化调度运行管理系统



江苏OMS

项目简介：基于“大运行”体系的省地县一体化OMS系统遵循智能电网调度技术支持系统采用大集中的思路，将**原有的27套系统，整合为一套系统**，全省的省、地、县（配）三级调度机构共用一个数据库。

- 日SQL量：1.4亿
- 基于达梦数据守护方案实现异地容灾

达梦数据库

挑战

问题

- 采用集中式部署，并发较高。DM6面临高并发情况下性能响应时间较长问题
- 系统存在大量行级更新，及统计性查询，DM6基于行级锁的并发控制存在读写冲突、锁升级问题

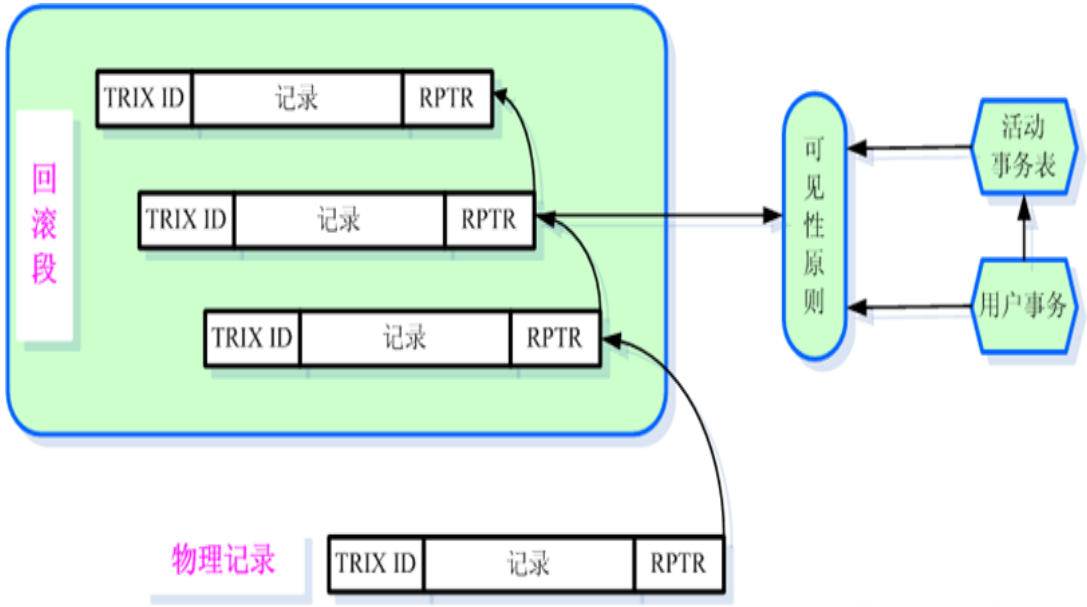
方案

- 实现了多版本并发控制，解决了读写冲突问题
- 避免锁升级

DM7并发处理

多版本并发控制

- 很多数据库实现了MVCC：Oracle、MySQL/InnoDB、PostGreSQL
- PostGreSQL：未引入回滚段
  - 各版本数据都存在数据文件，数据膨胀问题
  - 导致大量更新后的扫描性下降
- Oracle：基于块的MVCC
- DM7：基于行的MVCC
  - 更多并发更新、查询集中在一个Page的情况下
  - 提供更好的并发能力

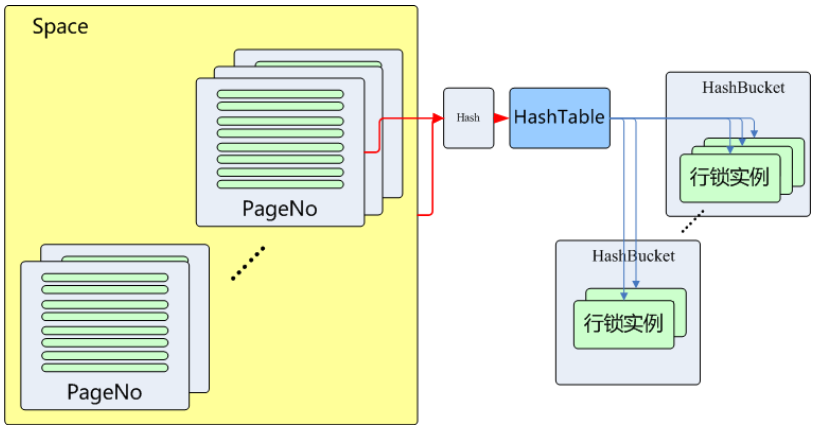


DM7并发处理

封锁机制

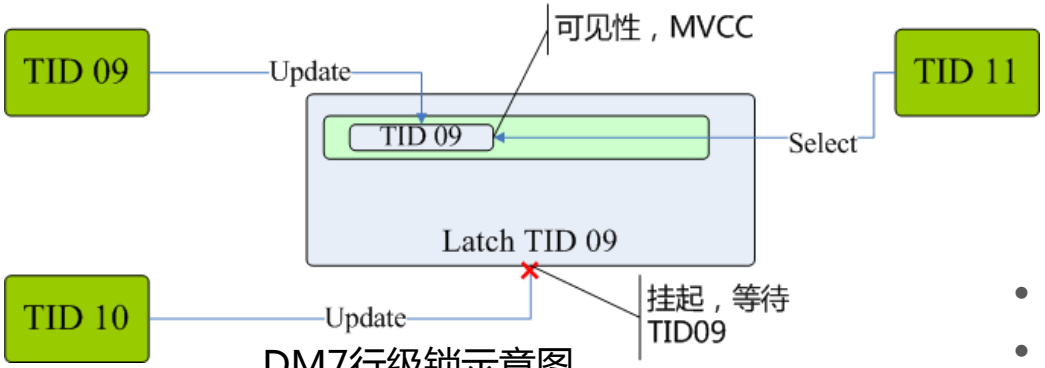
- Mysql/Innodb
  - 使用一个hash表总体管理行锁
  - 每个page的行锁使用一个bitmap表示
- 更新page越多，维护锁的内存开销越大
- 锁越多，需要进临界区的次数就越多

物理实现的锁代价是高昂的



Mysql/Innodb行锁示结构意图

- Oracle每个记录的LockBit指向Block的ITL，描述了锁信息
- 实现锁的消耗要小得多



DM7行级锁示意图

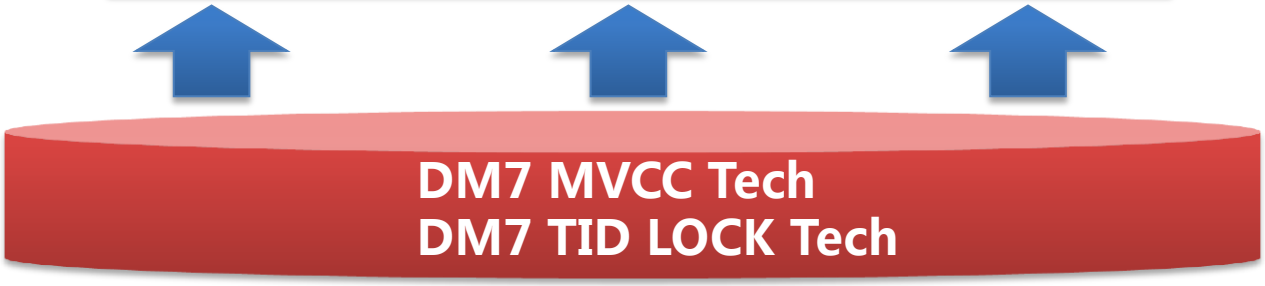
DM7更进一步，彻底消除行锁

- 修改过的行标识对应事务的编号(唯一递增， TID)
  - 读-写：根据本事务ID与记录TID大小判断可见性
  - 写-写：基于页面闩+TID可见性，实现锁定效果
- 减少资源消耗
- 没有锁升级问题

达梦数据库

效果

- DM7 MVCC技术避免了读写冲突问题
- DM7 TID锁机制避免了升级问题
- 系统在高并发环境下的资源占用降低了50%
- 核心业务并发查询性能提升了4倍
- 用户典型场景页面响应时间由10秒缩短至3秒



事务处理案例

省地县一体化调度运行管理系统

财务共享服务平台

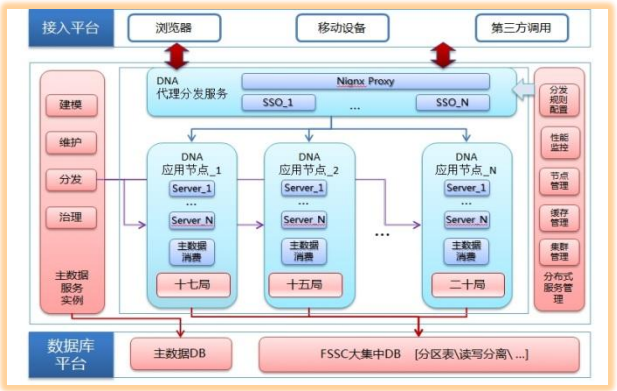
- 高并发：在线6000用户
- 读多写少：60%读，40%写
- 日常财务费控业务+周期性集中分析业务

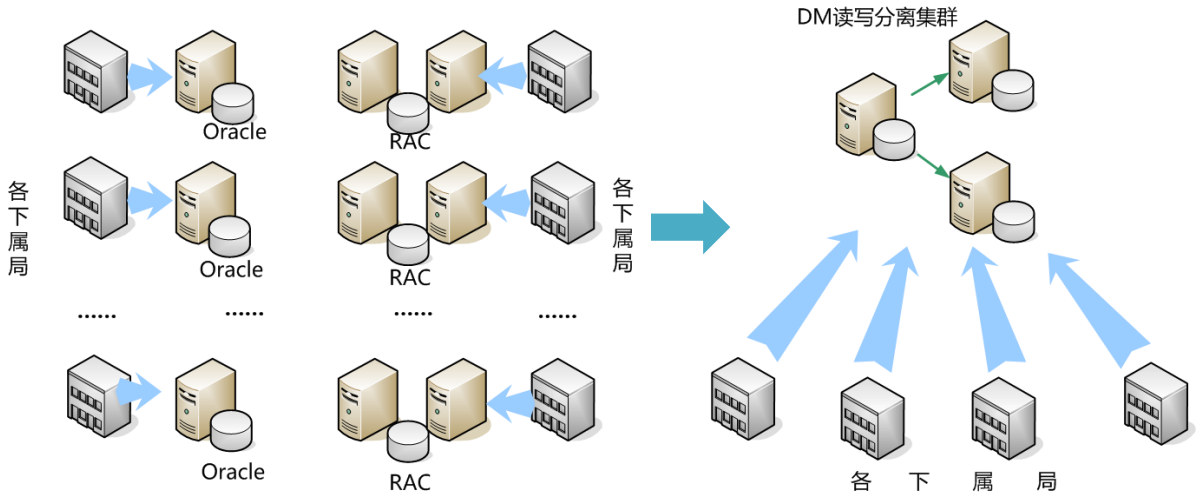
中铁建

中铁建财务共享服务平台

项目简介：下属20多个工程局的财务部门信息系统整合，形成集中式管理。

基于DM7.0读写分离集群，替换上一代系统的Oracle（部分使用Oracle单节点，部分使用Oracle RAC）





挑战

问题

高并发	典型业务
• 并发预期：目标是覆盖公司下属20多个工程局，完全覆盖后在线人数预期达到16000人 • 并发特点：读多写少，6:4	• 日常业务包括凭证录入等操作，进行并发增删改查 • 核算业务在月底生成报表，执行统计查询 • 月底、季度末、年底，核算业务形成长事务，大量占用CPU、内存资源， 导致日常业务响应时间延长

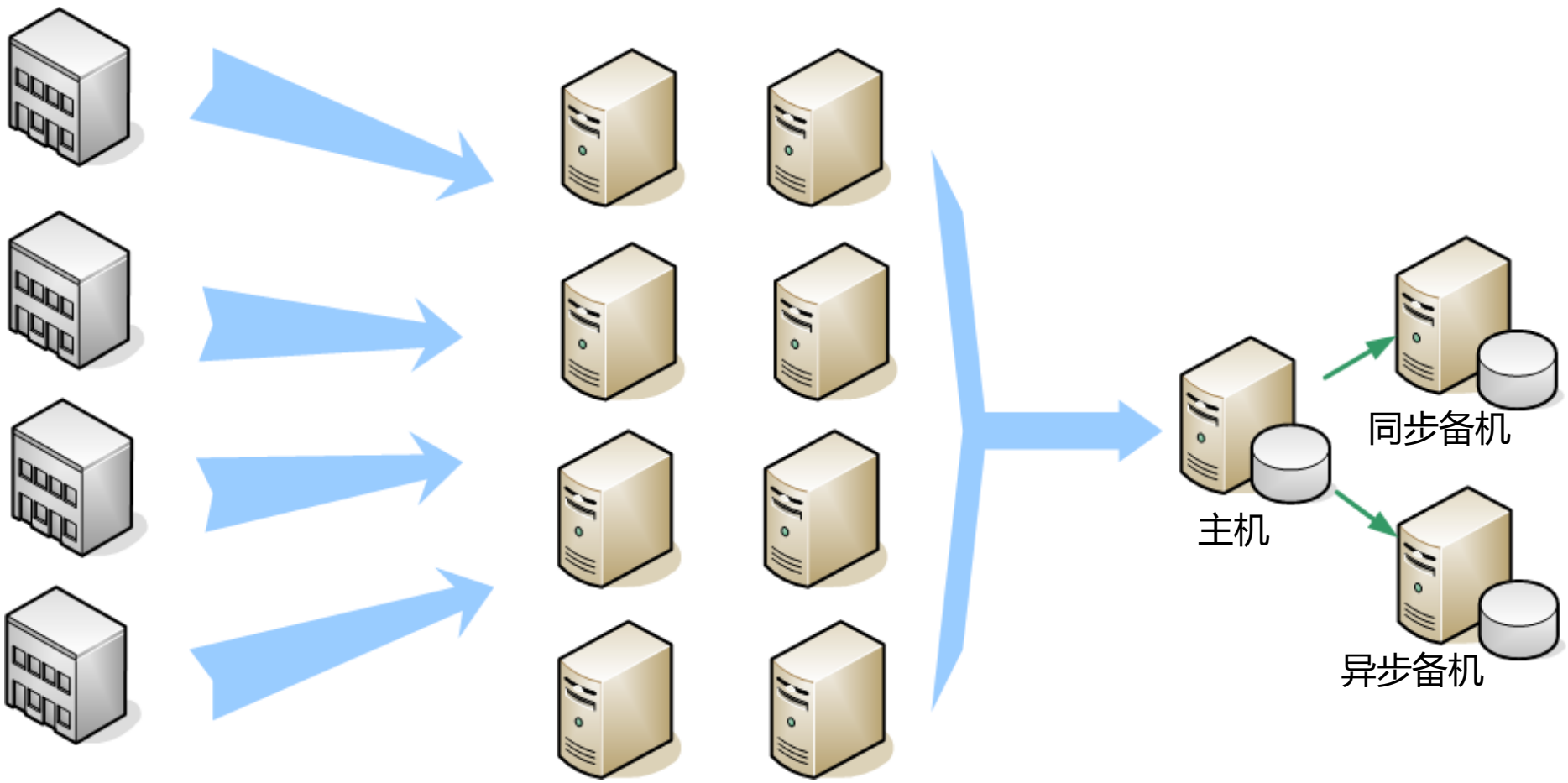
DM7通过具有负载均衡效果的读写分解集群降低每节点负载，提升并发处理能力，并降低核算业务对日常业务的影响

部署方案

目前已上线4个局

应用服务器：
8台应用服务器集群

数据库：一主两备



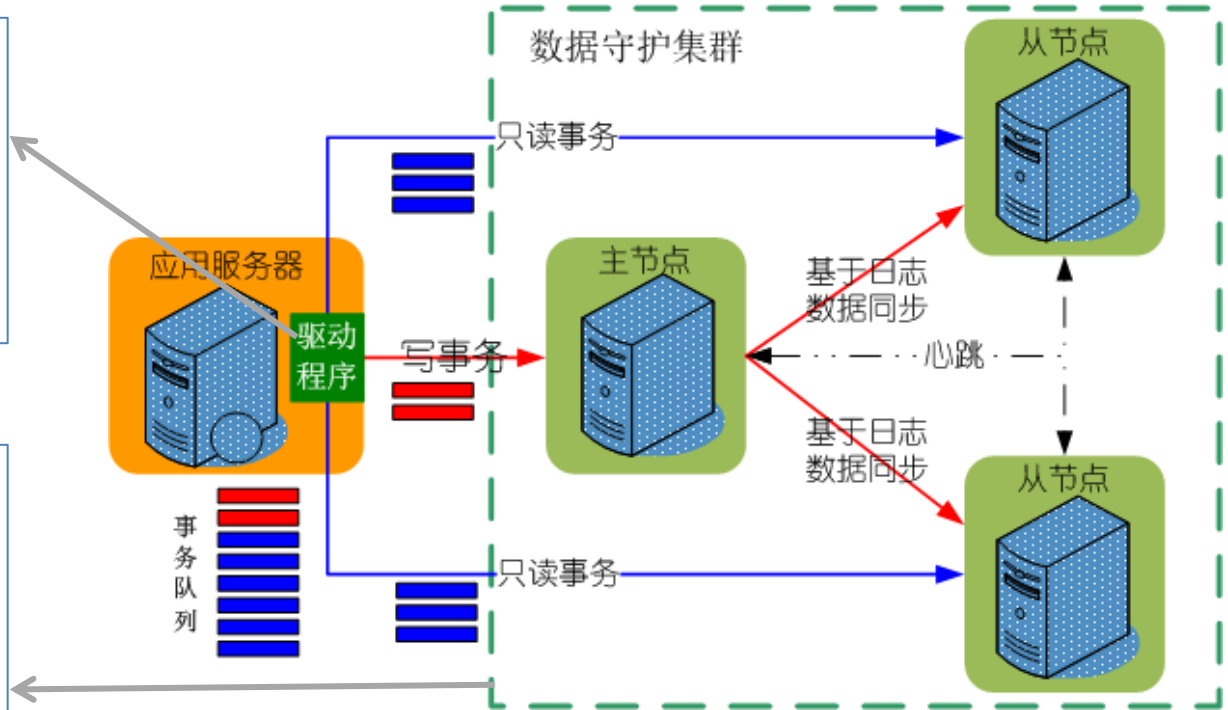
读写分离集群原理

驱动程序改造

- 写事务分发给主节点
- 读事务分发给从节点
- 主备数据冗余，基于日志同步

负载均衡

- 分担主节点读负载
- 实时同步节点&异步同步节点
- 最大可支持1主8备



读写分离集群的事务保证

- 类比：Mysql Proxy能够支持读写分离特性，但其设计有较大限制

> The read/write splitting is now following a simple rule:
>
> - send all non-transactional SELECTs to a slave
> - everything else goes to the master

来源：<<MySQL Proxy learns R/W Splitting>>
Jan Kneschke，mysqlproxy作者

- 结果就是：
  - 对事务的支持变弱，企业级应用受限
  - 存储过程无法实现读写分离

问题：事务A，包含三条SQL，类似：

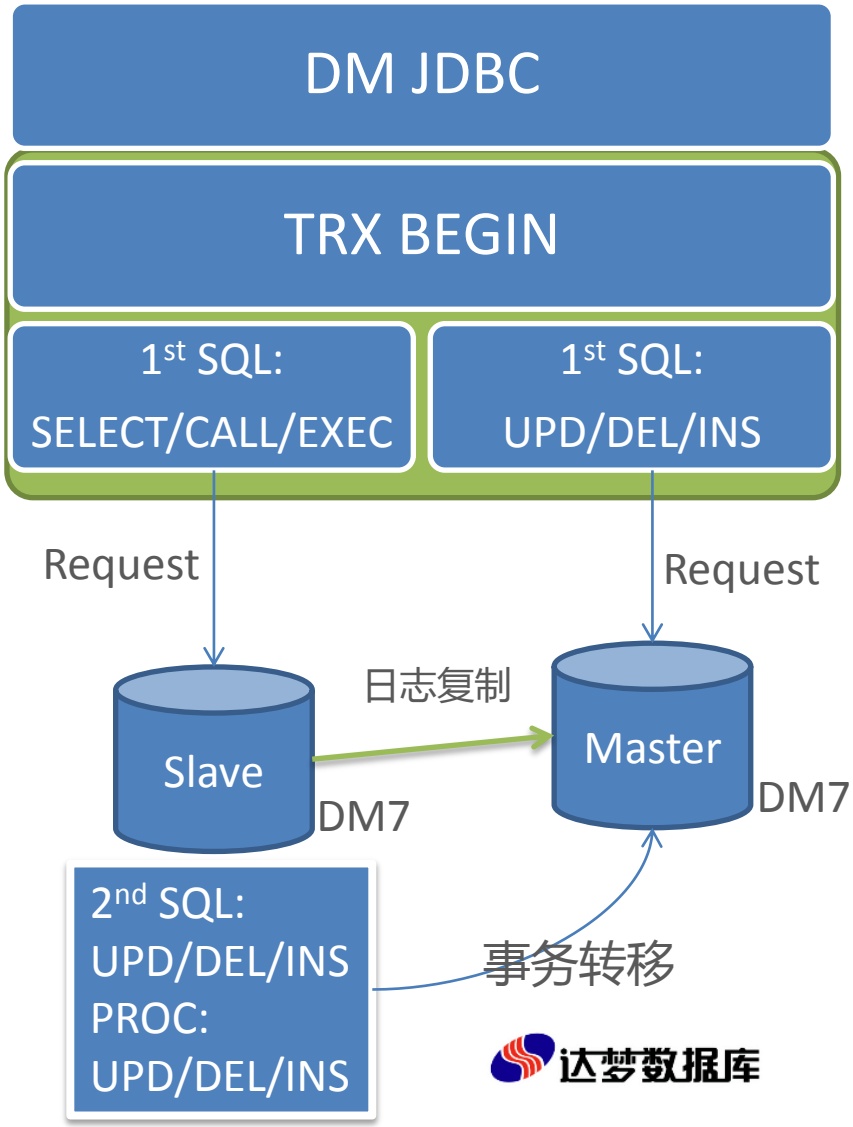
一个事务

SELECT C1 FROM T1 WHERE C2=XX;

UPDATE C1 SET C1=C1+10 WHERE C2=XX;

SELECT C1 FROM T1 WHERE C2=XX;

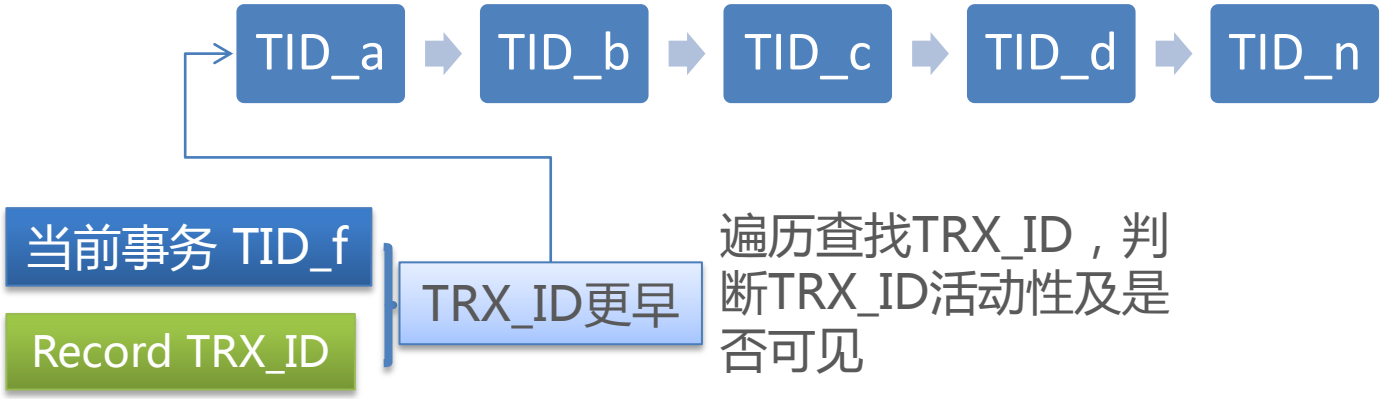
DM读写分离如何保证事务一致性？



读写分离集群的并发优化设计

问题：读写分离环境下，备机的并发能力可否进一步优化？

传统方式：单机环境并发查询，如何判断不同事务间的可见性关系？



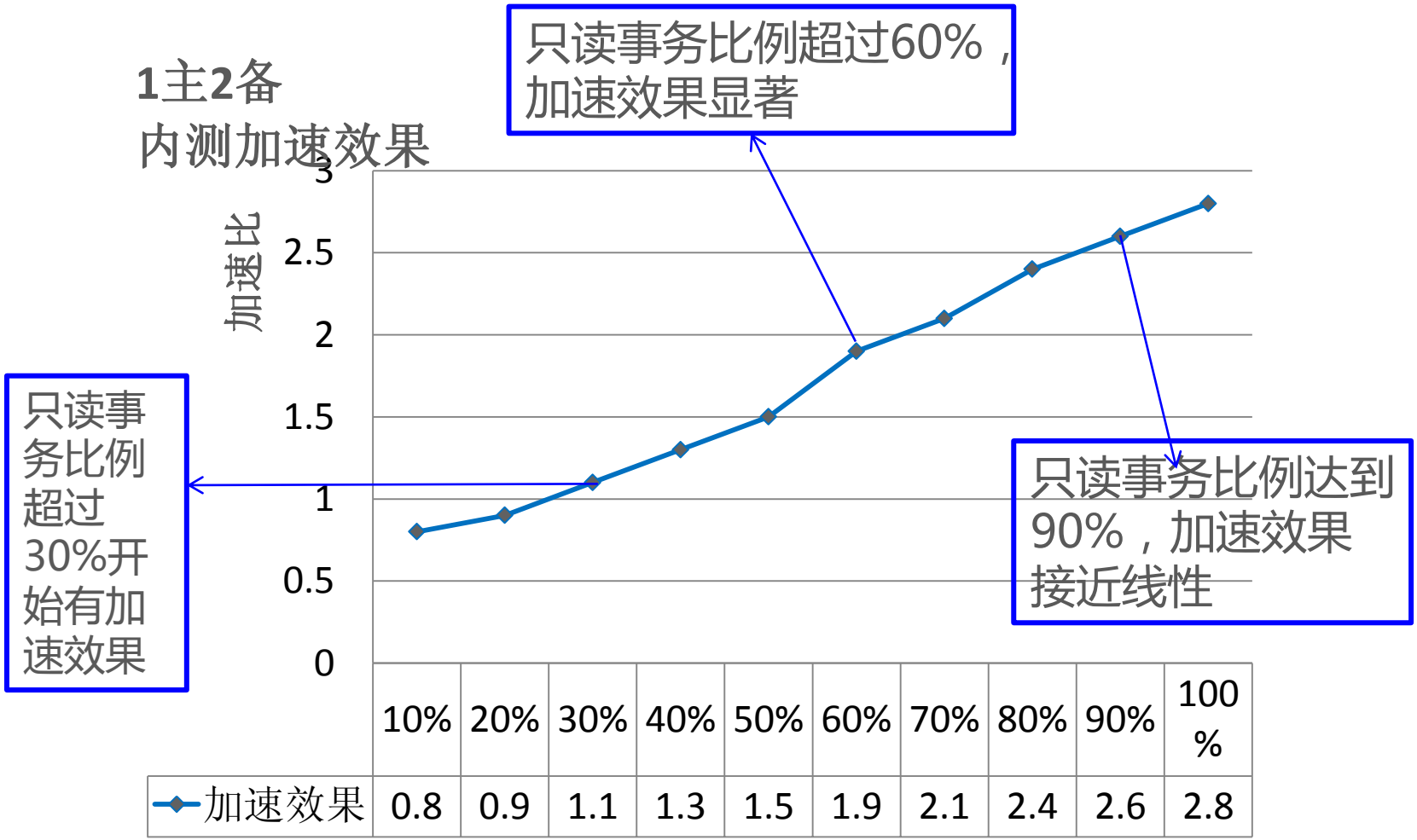
备机的现实环境是：读写分离从节点的记录事务号来源仅为主机REDO

简化备机的并发事务可见性判断逻辑，进一步提升备机并发性能

达梦数据库

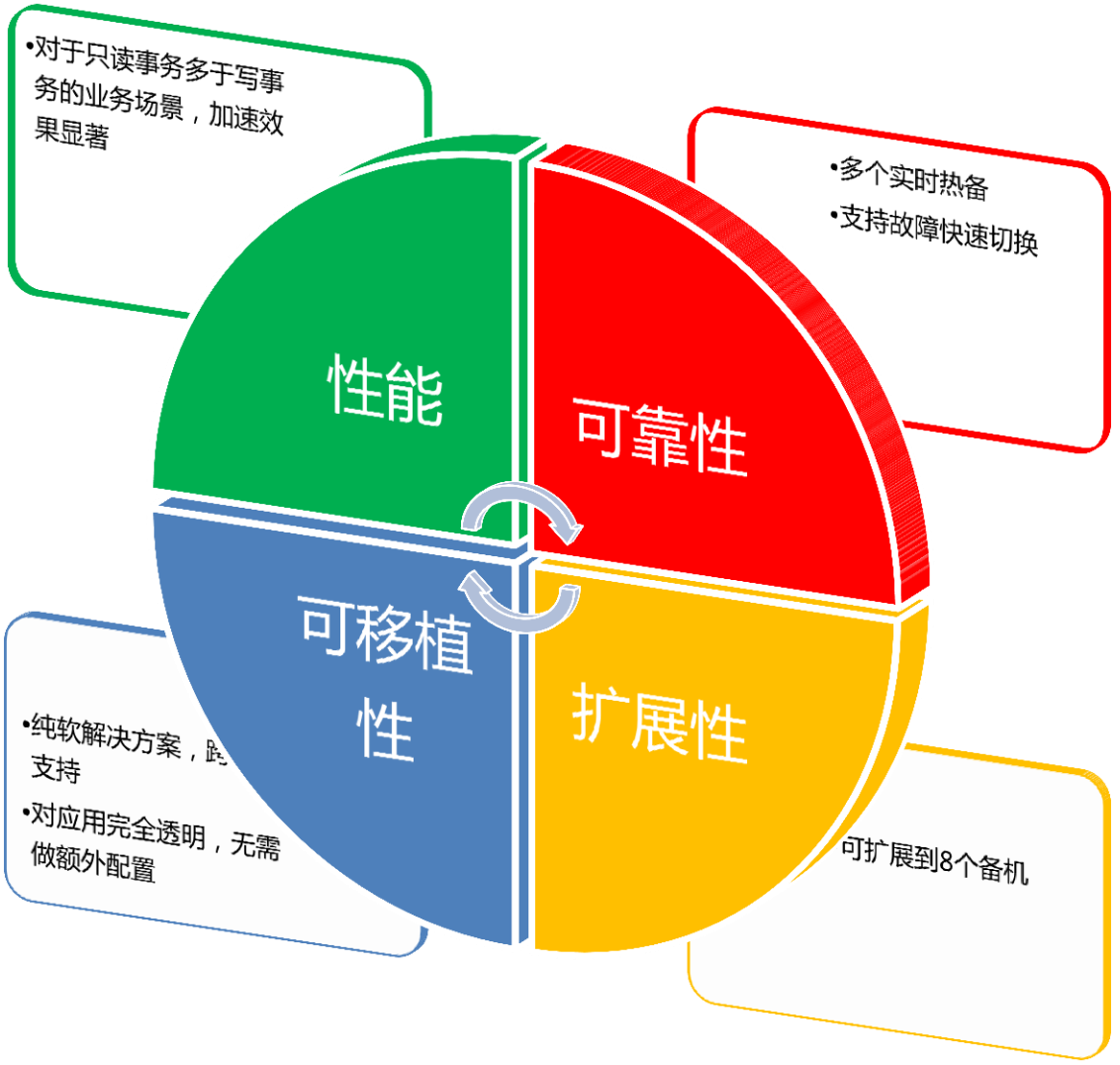
读写分离集群

1主2备
内测加速效果



	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
加速效果	0.8	0.9	1.1	1.3	1.5	1.9	2.1	2.4	2.6	2.8

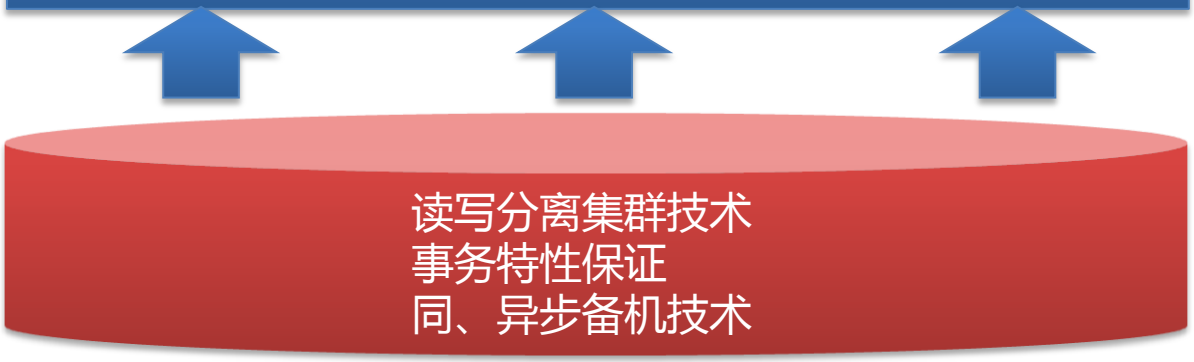
读写分离集群特点



达梦数据库

效果

- 日常业务响应时间基本不变的情况下，从单机支撑2000用户，提升到支撑了6000+用户
- 通过同步、异步备机混搭，保证了业务的性能
- 随着业务增加，1主2备还可持续扩展，直至1主8备
- 降低了用户基础设施成本

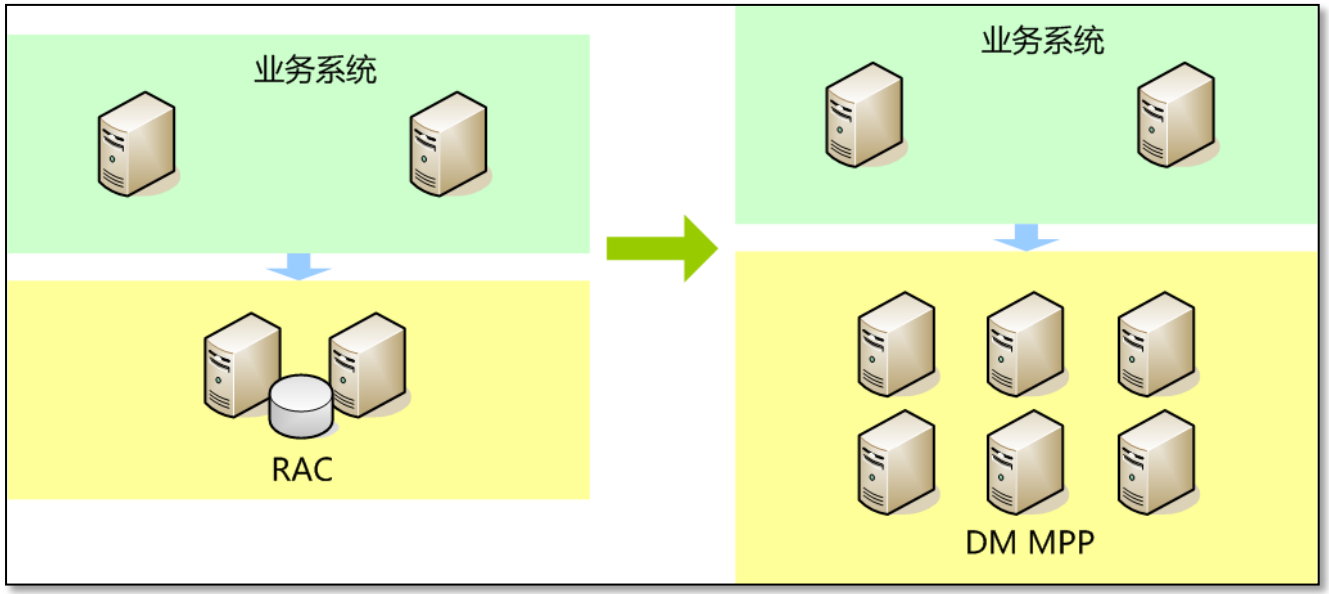


分析应用案例

项目简介

某运营商话单综合分析系统原有基于小型机+Oracle RAC，性能逐渐无法满足业务需求

- 大数据量，持续快速增加（100TB，千亿级，月增4TB）
- 并发分析请求
- 大量即席查询需求



- PC Server
  - 2X E5-2690
  - 128GB
  - 10K RPM SAS RAID5
- 万兆网络

达梦数据库

挑战

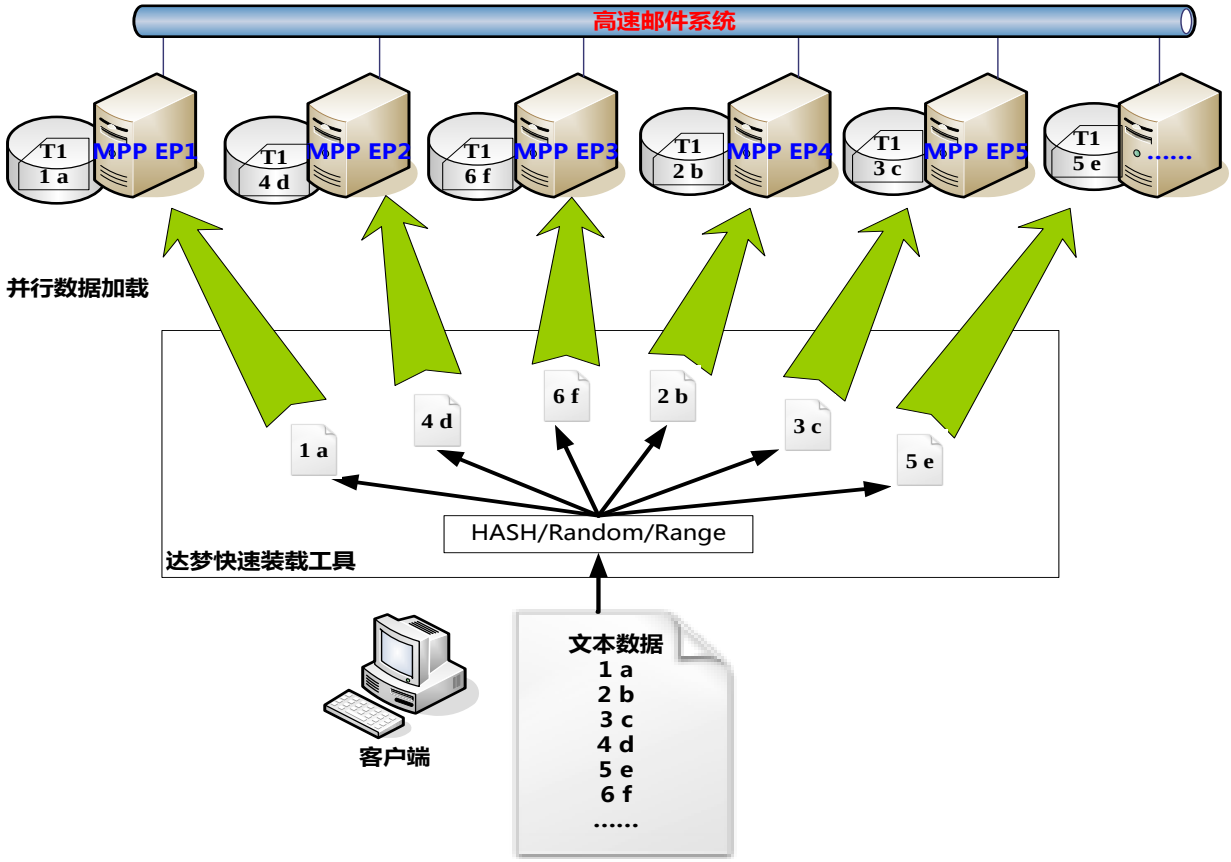
问题

- 数据装载速度要求
  - 每月增量数据装载窗口4小时
- 分析性能
  - 不断提升查询响应时间

DM7通过MPP+LPQ并行及“平坦化”等优化技术，成功提高了分析查询性能

并行加载

- 工具端并行，支持同时向多个MPP节点加载数据
  - 客户端文件分割
  - 客户端数据Hash分发寻径
  - 减少服务器CPU、网络资源消耗
- 服务器端并行，提供多个工作线程执行数据读取与写入
  - 快速装载技术：非常规Insert方式，Undo、Redo日志生成优化



达梦数据库

SQL查询优化技术

子查询平坦化技术

- “平坦化”概念：
  - 将子查询展开为与上层表的连接查询
  - Oracle中采取了类似的UNNEST_SUBQUERY技术
- 有何意义？不平坦化将导致：
  - 相关子查询情况下，子查询执行次数取决于外层查询与子查询关联条件的命中次数，形成类似Nest Loop Join的高代价操作
  - CBO对整体的代价估算难度增加

DM优化器自动实现平坦化改写，对应用透明

```
select count(*)
from   c_order
where  exists (
  select *
  from c_order_line
  where        o_id = ol_o_id
    and ol_quantity > 10
  )
```

- 相关子查询的“子计划”代价估算是一个难题
- 代价估算必须经过多次迭代
- 子查询内的表将被多次逻辑IO，次数取决于外层表数据量

```
select count(*)
from   c_order A semi join (
  select B.ROWID
  from c_order_line, c_order B
  where B.o_id = ol_o_id and ol_quantity > 10
  group by B.ROWID
  ) C on A.ROWID = C.ROWID
```

- 形成“平坦”的查询计划
- 大幅度降低内层表的IO次数
- 便于更精确的估算代价
- 更易于生成并行查询计划

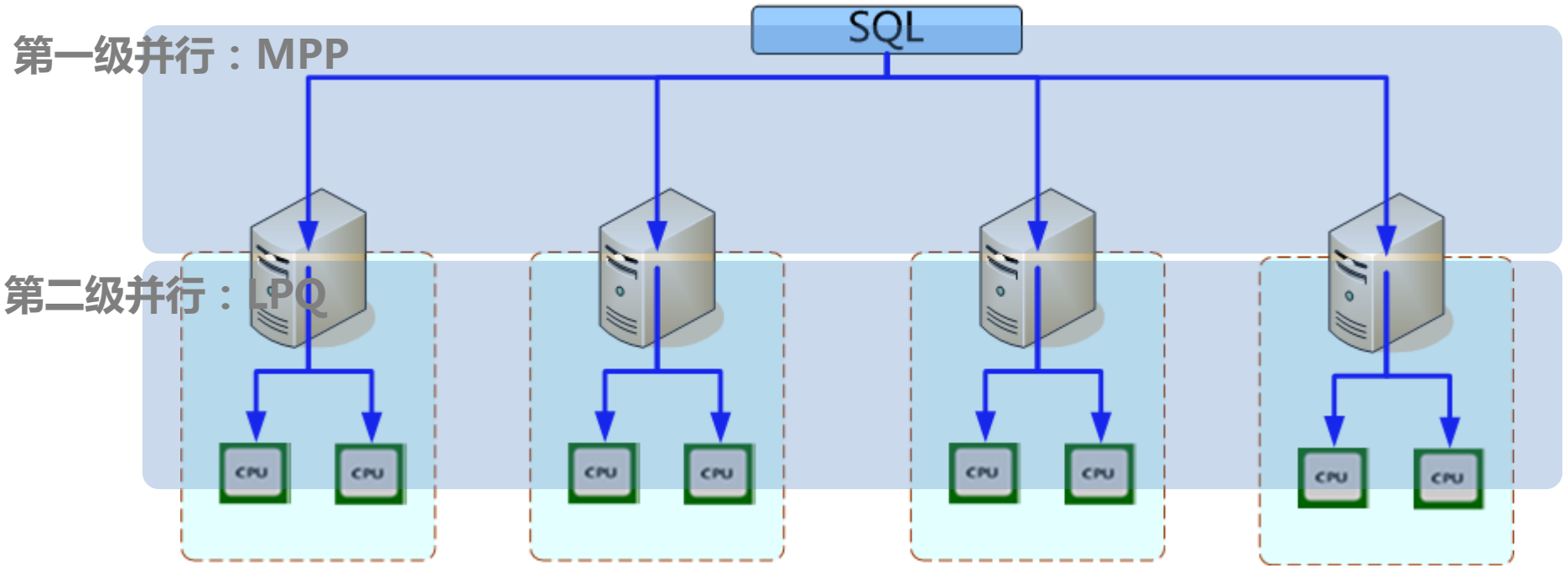
达梦数据库

查询并行化

- DM实现了两个级别的并行查询
  - 多节点间并行（MPP）
  - 节点内本地并行（LPQ）

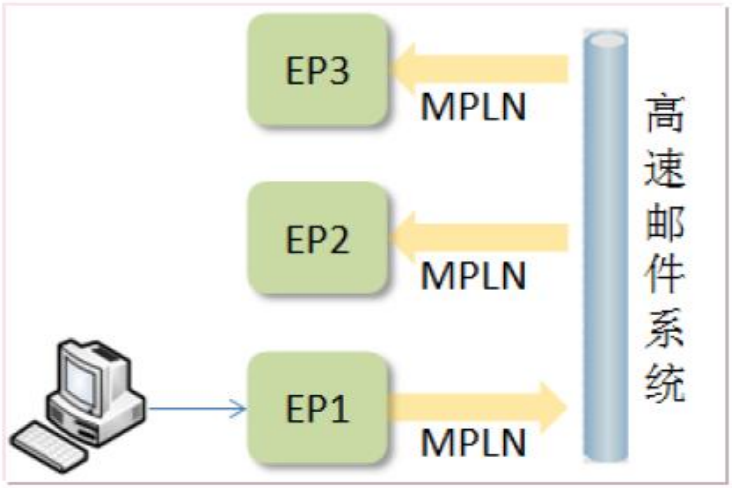
对称并行模型：

每个执行节点执行相同的计划

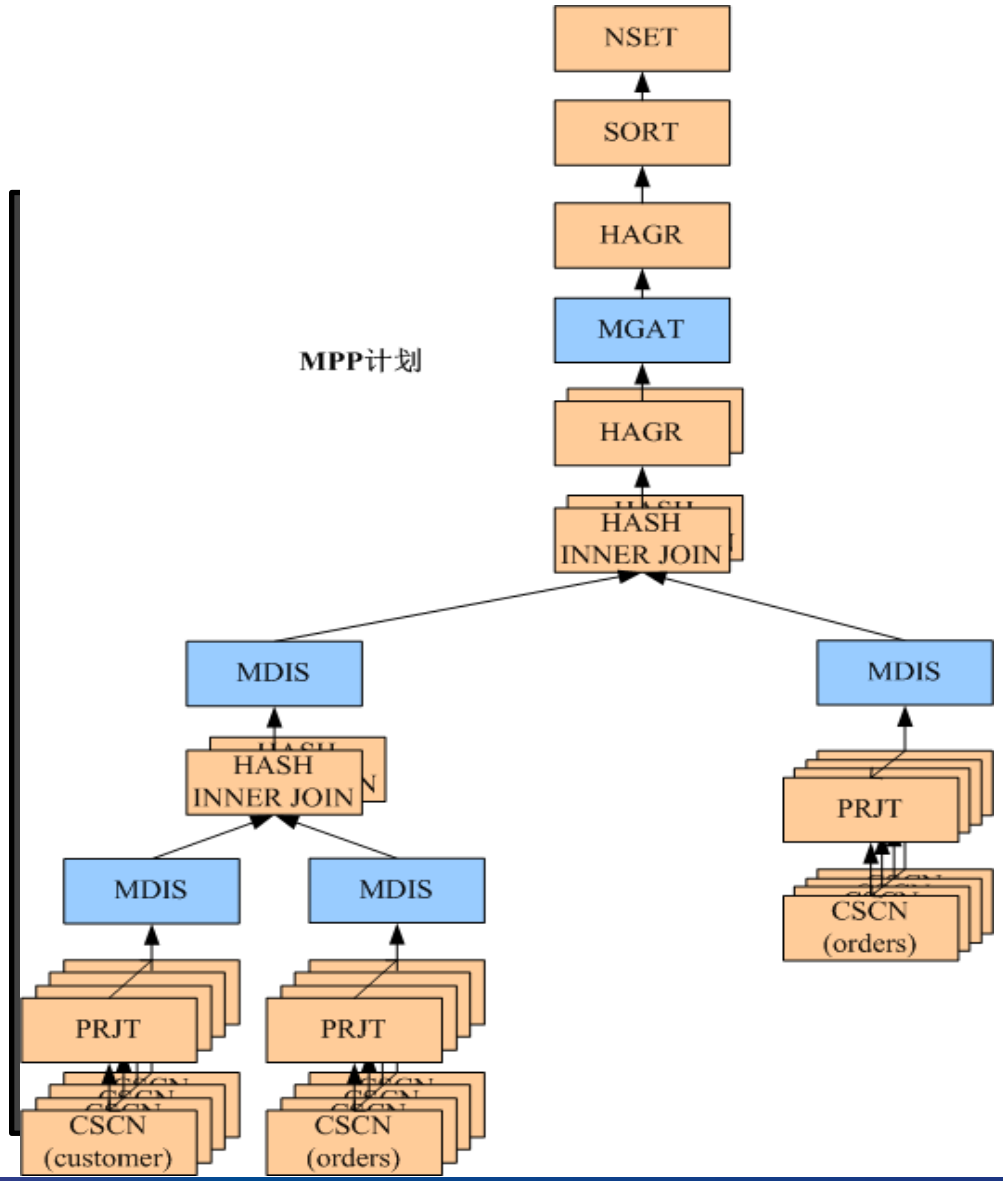


查询并行化

多机并行集群（MPP）



- 通过MDIS、MGAT等操作符，实现多机并行操作。
- 支持随机分布、HASH分布等多种数据划分方式
- 支持连接、分组操作的跨节点并行（hash分布）

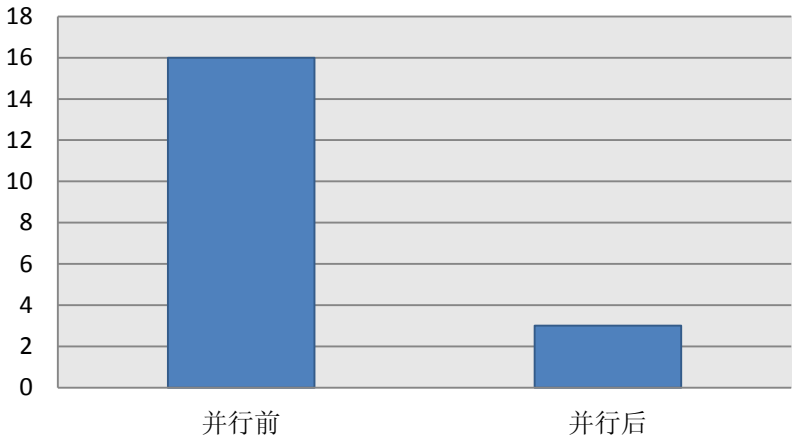


查询并行化

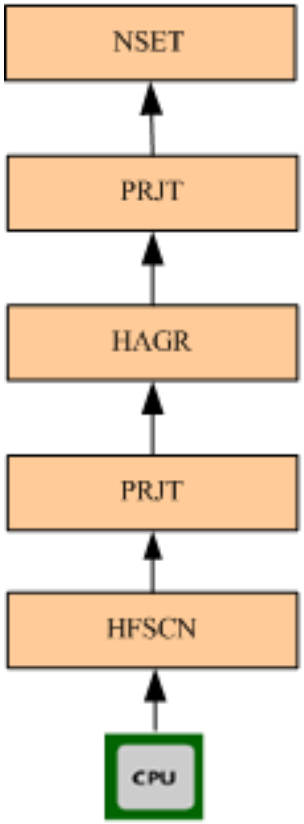
本地并行（LPQ）

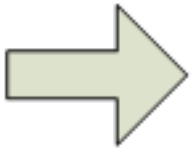
```
select
substr(ID,1,2) ID, RVAL, count(*) cnt,
TVAL from RX
group by substr(ID,1,2), RVAL,TVAL ;
```

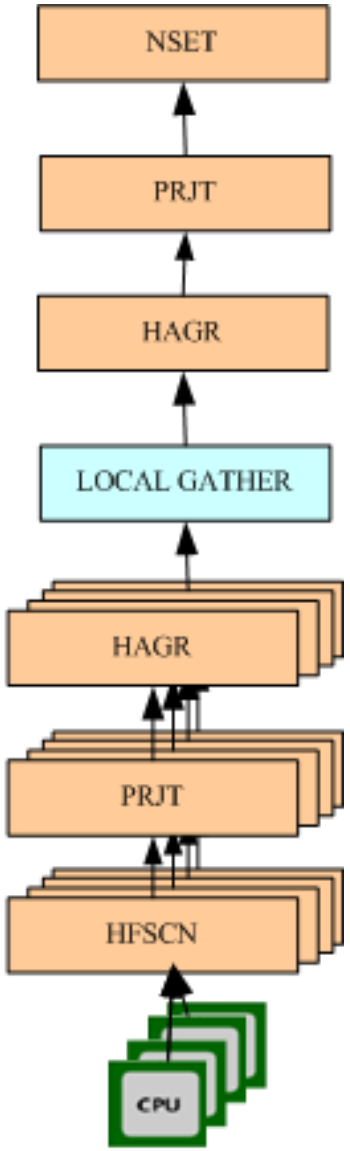
并行查询耗时比例



13亿条记录
CPU 2*4 2.13Ghz/mem 32G /
HDD 2*600G sas





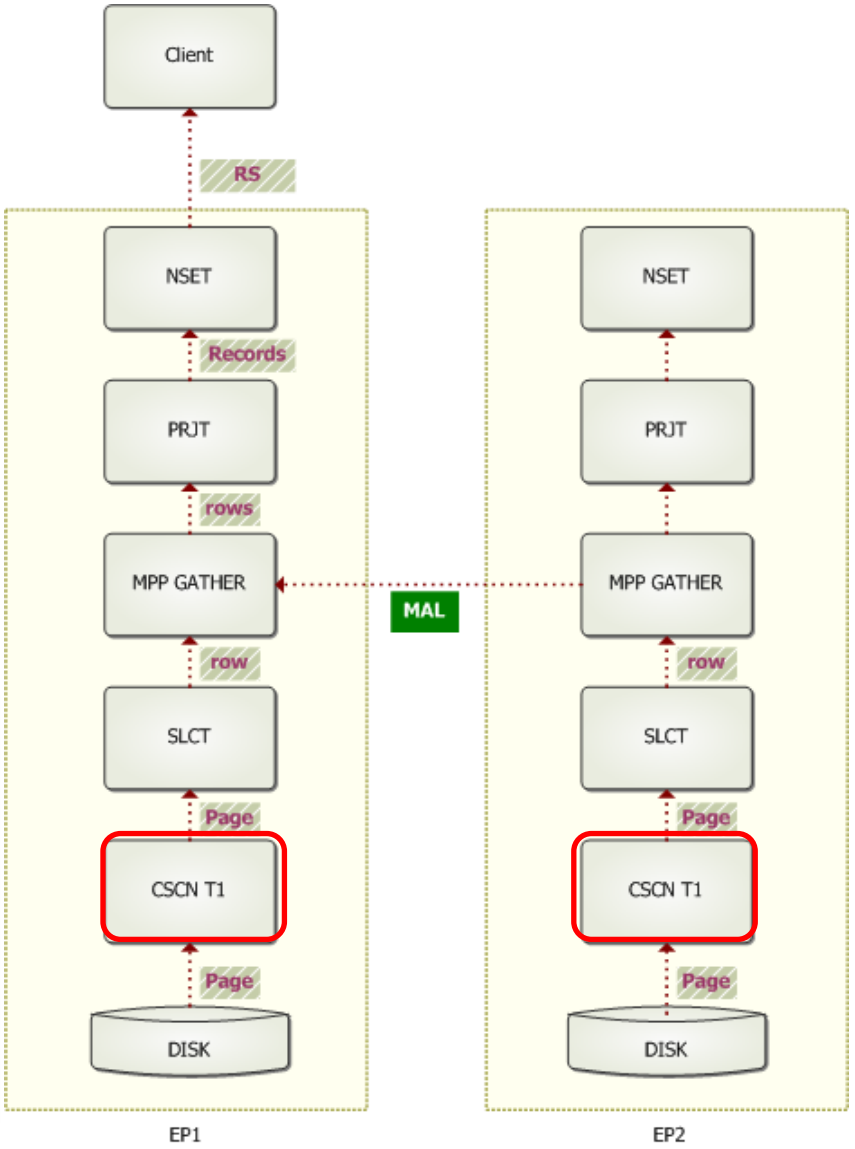


操作符跳转技术

实地使用过程中，发现了主节点处理滞后于从节点数据发送的问题，影响查询效率

- 结对的MPP GATHER操作符
  - EP2 数据通过MAL流向EP1
  - 每次MPP GATHER处理收到的MAL
- 大量MAL流向EP1时，造成MAL堆积
  - EP2硬件配置相对更好
  - EP2负载相对较轻
  - EP2数据碎片更少
  - EP1 SLCT屏蔽更多Rows

TIPS:
- DM7操作符
  - CSCN：聚集索引扫描
  - SLCT：选择运算
  - MPP GATHER：消息收集到主站点
  - PRJT：投影运算
  - NSET：结果集收集



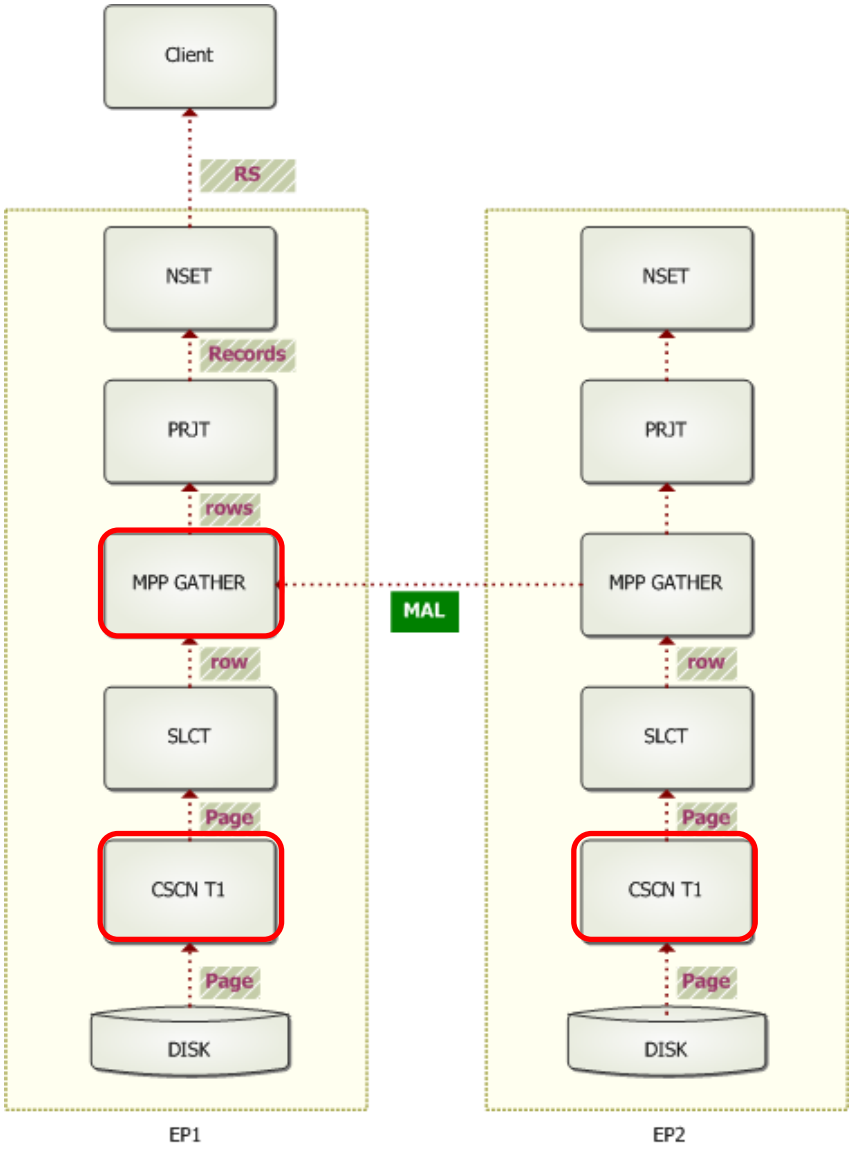
操作符跳转技术

思路：

- 数据来自EP1还是EP2不重要
- 对于数据的MPP操作处理，没必要顺序执行
- 可以适当的“乱序”

优化：

- 提升了邮件处理的优先级
- 由下到上顺序→跳转到祖先Motion节点先执行(根据收邮件情况)
- 祖先节点完成后跳回原有节点



效果

- 并行高速加载技术
- 子查询平坦化技术
- 多级并行处理MPP+LPQ
  - 操作符跳转技术

DM7 MPP(6节点)性能表现：

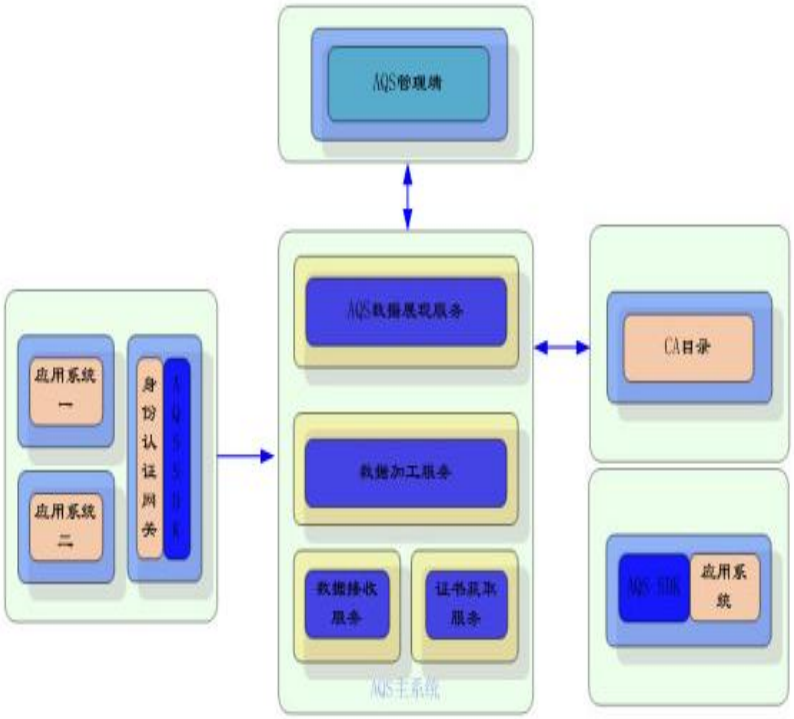
- 数据装载：均速1.25TB/小时
- 22个典型场景总耗时：800S
- 某国外MPP产品：2000S以上(18个部分场景)

典型场景一例：单表42亿(EH)+单表11.9亿(EL)+单表4.5亿(EW)关联统计查询

```
select h.C1  as CC1, h.C2  as CC2, h.C3  as CC3, h.C4  as CC4, l.D1  as DD1, l.D2  as
DD2, l.D3  as DD3, l.D4  as DD4, l.D5  as DD5, l.D6  as DD6, l.D7  as DD7
from EH h, EL l
where (h.C1 = l.C1 and h.C5 like to_char('2328%') and h.C6 between '2011-03-06
00:00:00' and '2011-03-07 23:59:59' and (1 = 1) and exists  (
select 1 from EW w  where w.E1 = h.C1 and E2 = '10000000‘ and upper(substr(w.E3,
1, 1))  >=  'a'  and upper(substr(w.E3, 1, 1)) <= 'z'));
```

混合负载案例

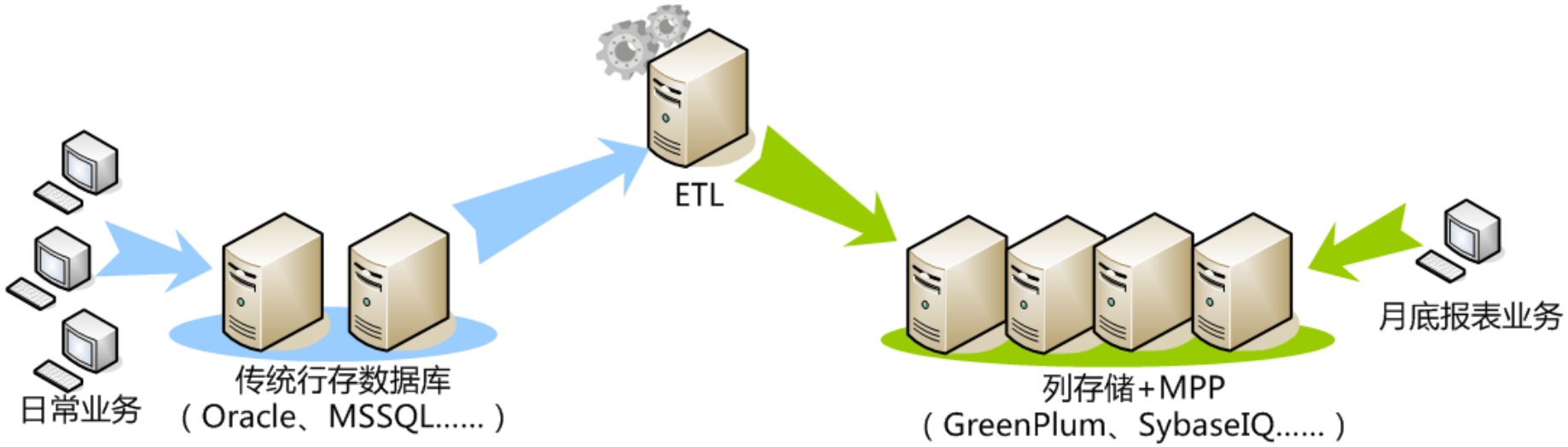
数字证书综合统计查询系统



- 对数字证书信息进行管理
- 日常提供证书日志审计存储、证书详细查询、访问信息查询等业务
  - 日志表——基于行存，按月、按地区分表
  - 日常业务并发用户300左右
- 每月底对证书状态进行汇总分析，形成报表
  - 统计表——基于列存，70亿记录
- 混合负载需求

方案分析

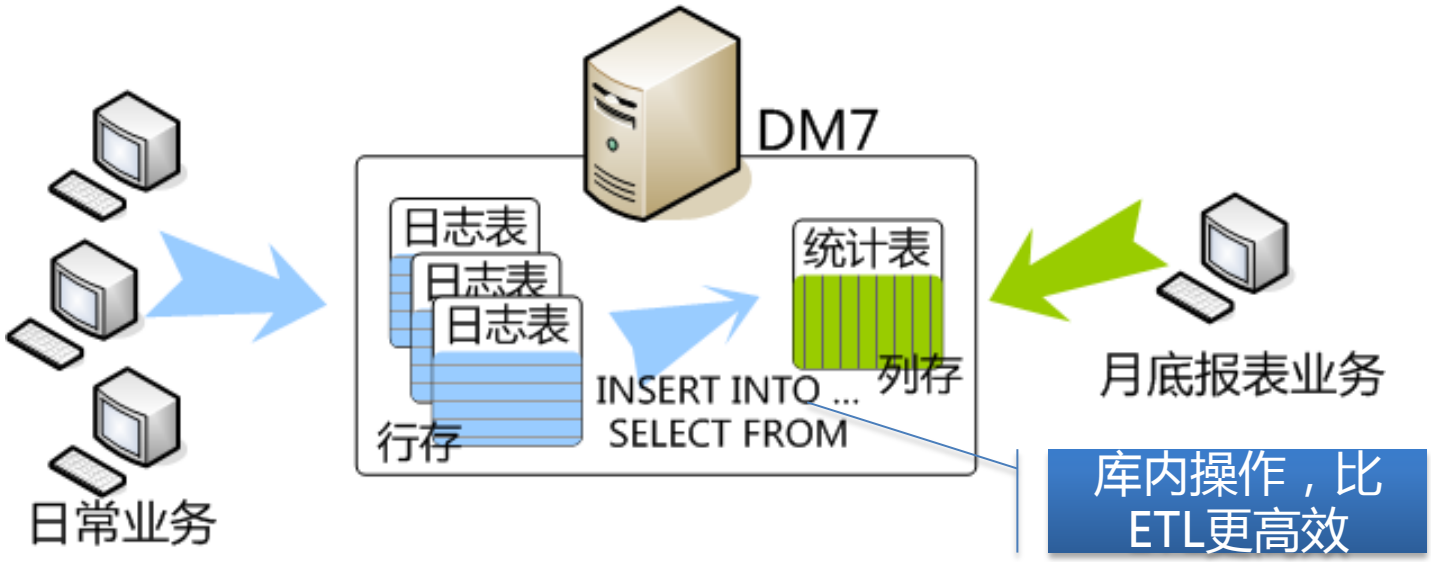
	行存表	列存表
优势	并发读写 定位查询	全表扫描 分析型查询
劣势	全表扫描	并发相对较差 查询列较多



传统方案

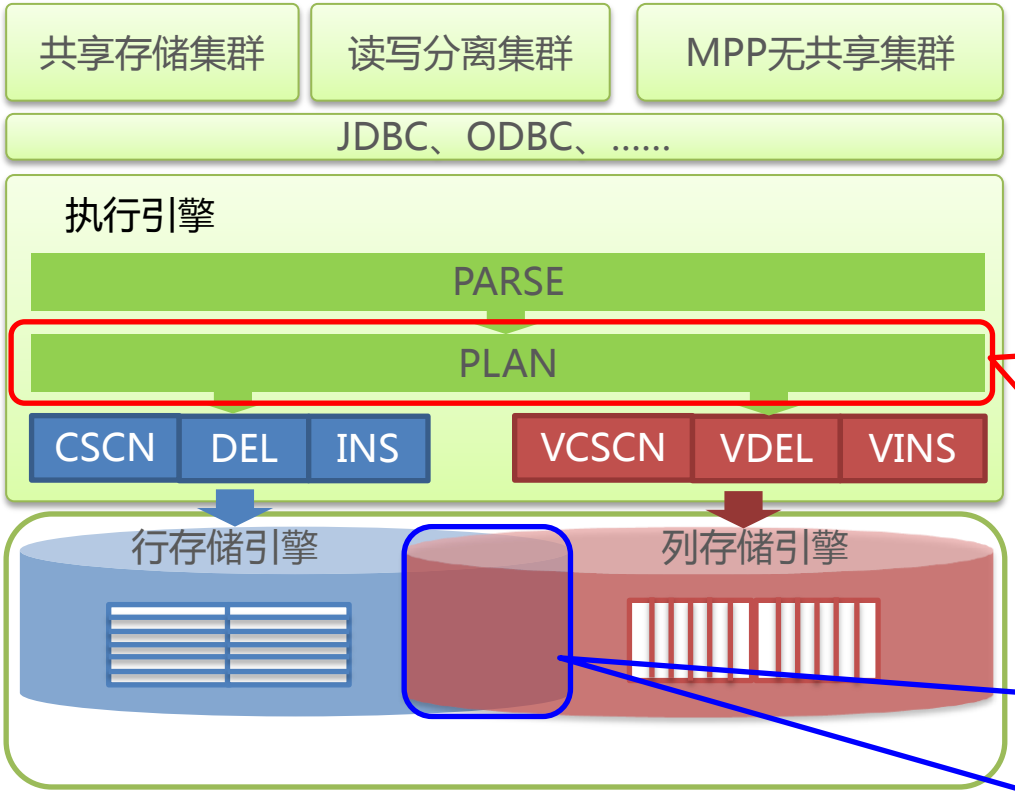
方案分析

	行存表	列存表
优势	并发读写 定位查询	全表扫描 分析型查询
劣势	全表扫描	并发相对较差 查询列较多

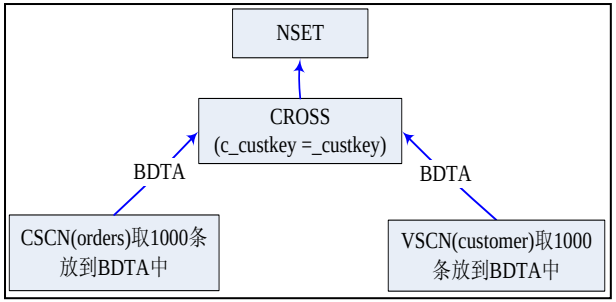


DM行列融合方案

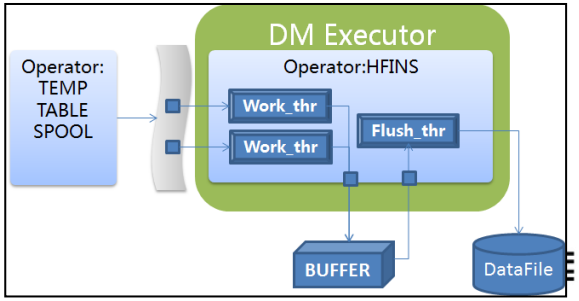
我们的方法：行列融合架构



- 支持行存表与列存表的联合查询
  - 在执行计划层，实现了行存操作符与列存操作符的对接



- 支持行存表与列存表之间的Insert into select等互操作

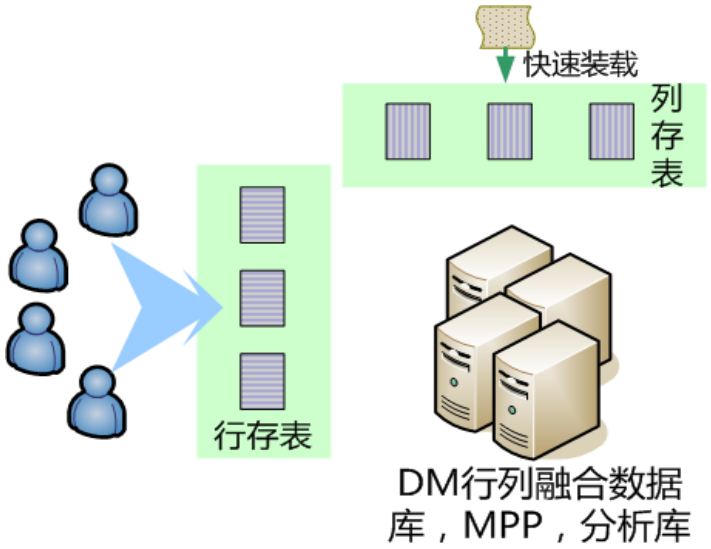


DM7行列深度融合不仅仅是提供两种存储引擎

几类“跨界”型应用

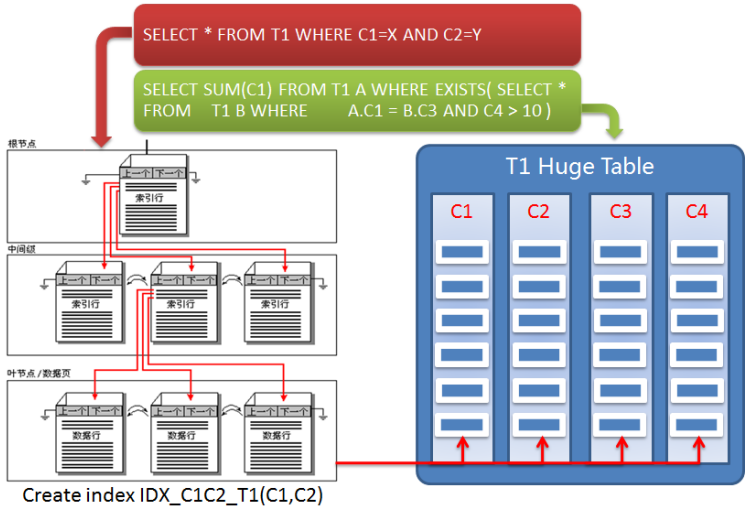
数据来源“跨界”

- 海量分析系统，但部分数据来源为集中的批量加载，部分数据来源为非集中的，并发的实时写入。
- 传统行存储表更适合并发的实施写操作
- 列存储表更适合集中式的批量入库



业务上“跨界”

- 事务为主，兼有部分有限范围内的汇总查询
- 分析为主，但也提供一些并发的、短小的、精确的明细查询服务



达梦数据库

DM7提供新的选择

DM7行列融合新的方案优势：

- 同构数据库系统——不再需要两个不同品牌的数据库
- 减少数据冗余——减少硬件投资
- 不同表之间数据同步——不再需要中间环节
- 降低用户应用难度——架构简单，部署简单、开发更简单

“一个”产品

我们所讨论的DM7，是“一个”产品

- **纯事务：**
  - DM7现在提供：读写分离集群；
  - &不久的将来，DM7能提供：共享存储式集群
- **纯分析：**
  - DM7提供：MPP+列存储+智能索引+数据压缩+多级分区
- **传统DW：**
  - DM7提供物化视图 on MPP、位图连接索引 on MPP，通过并行技术加速传统架构
- **混合负载：**
  - DM7提供：读写分离集群架构
  - DM7还能提供：深度的行列融合（e.g：Btree Index on Vertical Table；Row Table Join With Vertical Table in ONE SQL)，提升应用架构设计的灵活性

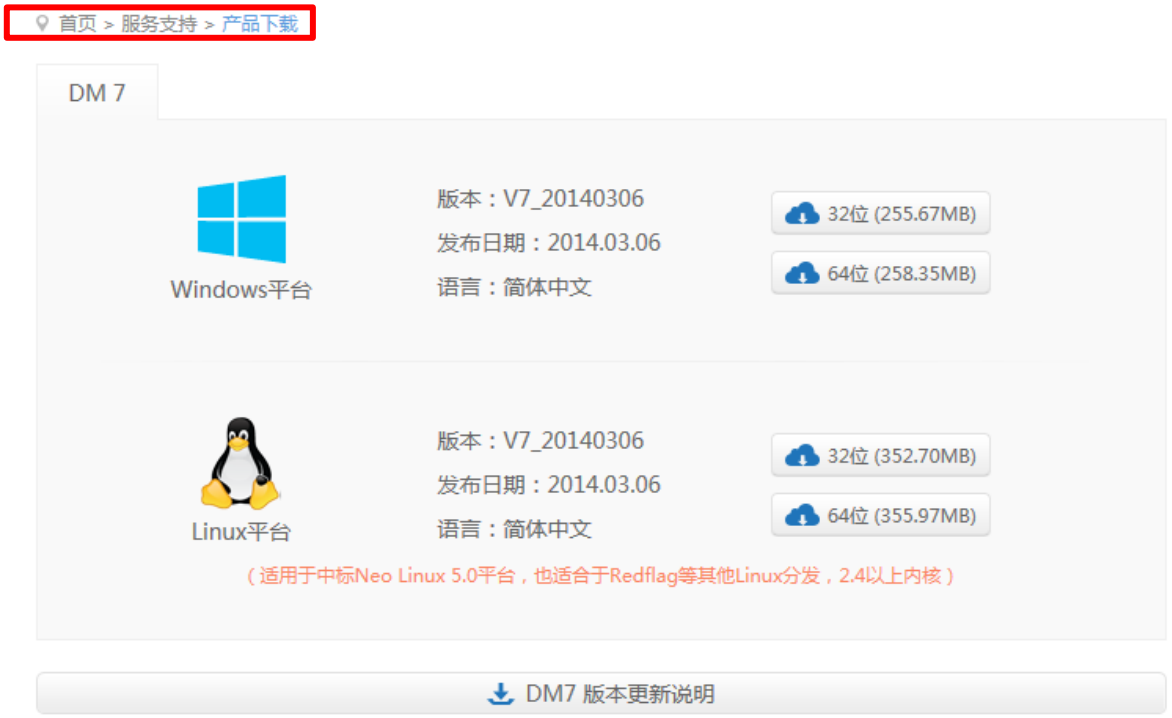
“一个”数据库，简化IT架构

达梦数据库

欢迎试用反馈

无论您是个人、企业、学校、政府机构……

- 欢迎下载并用于评估、教育、学习，并反馈您的意见
- DM7开发版下载
  - www.dameng.com
  - 无功能限制
  - 无时间限制
- 每月更新一个新版本，包含：
  - 集成全量补丁
  - 不断增加的新特性



DM是一个充满活力的产品，您的反馈意见能得到最快的响应

达梦数据库

致谢

THANK YOU!

达梦数据库