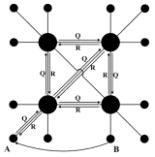
DATAGURU 炼数成金

Leader-us系列



大型分布式系统案例实战 第5周

法律声明

【声明】本视频和幻灯片为炼数成金网络课程的教学资料，所有资料只能在课程内使用，不得在课程以外范围散播，违者将可能被追究法律和经济责任。

课程详情访问炼数成金培训网站

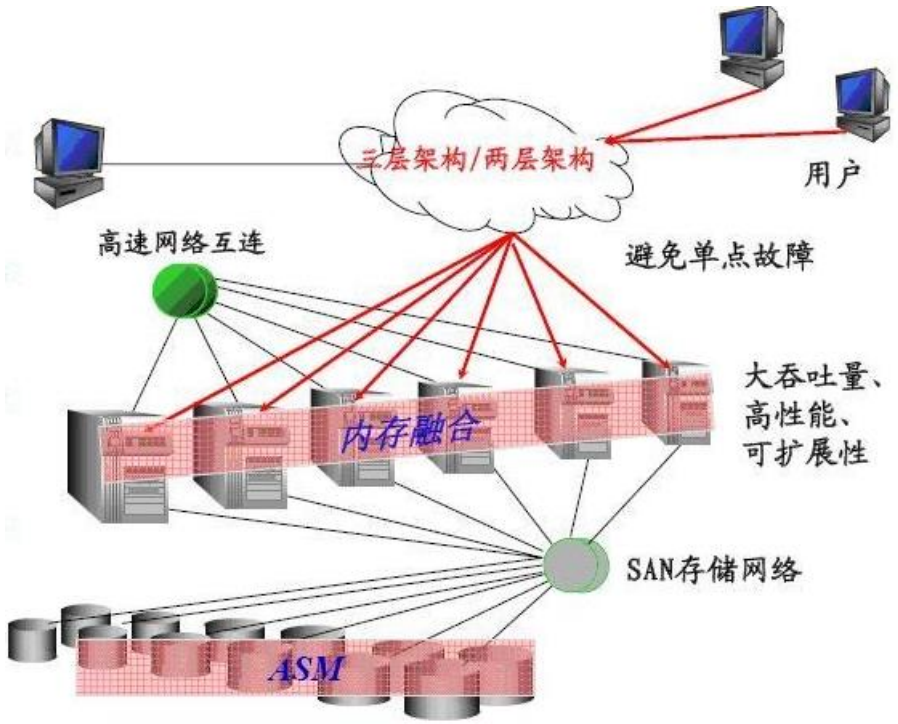
http://edu.dataguru.cn

分布式系统之分布式数据库篇

- 分布式数据库那点事
- 分布式数据库热点技术
- Mycat入门

分布式数据库那点事

Oracle怎么实现分布式——Oracle Rac



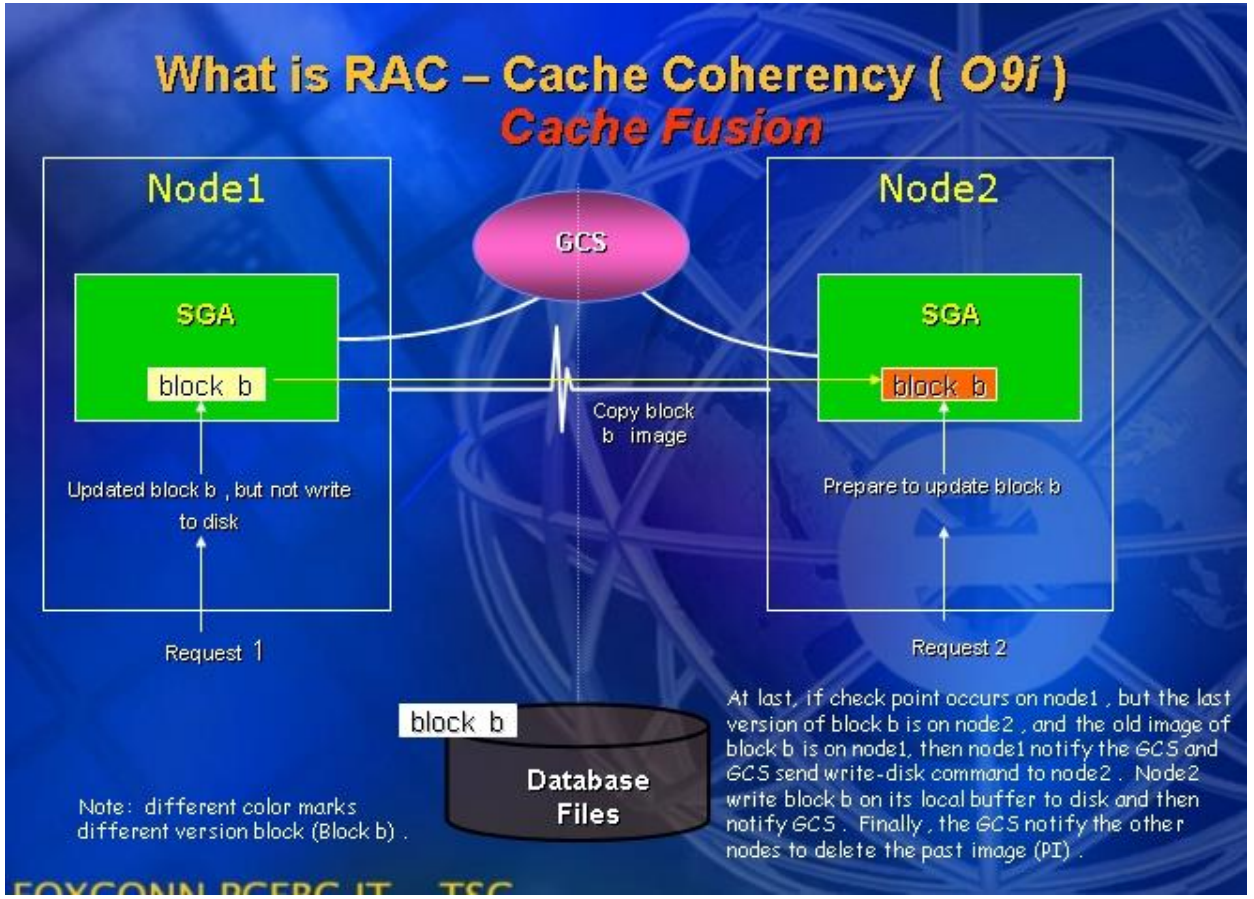
- 多节点负载均衡;
- 通过并行执行技术提高事务响应时间—-通常用于数据分析系统;
- 通过横向扩展提高每秒交易数和连接数 ;—-通常对于联机事务系统;
- 节约硬件成本，可以用多个廉价PC服务器代替昂贵的小型机或大型机，同时节约相应维护成本;
- 可扩展性好，可以方便添加删除节点，扩展硬件资源;

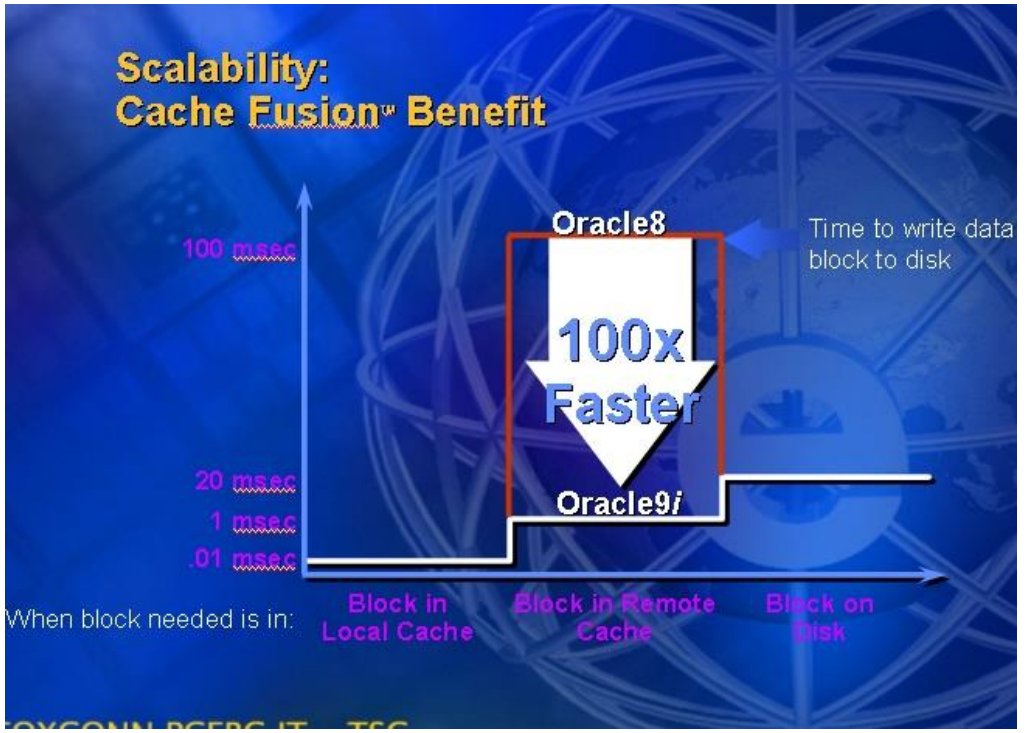
往往人们会认为RAC有2个节点性能就会提升2倍，这是一个误区，由于要保证数据的一致性往往性能会消耗在内存间的数据块相互拷贝和交叉上，因此不一定性能会好于单节点，而且节点越多性能曲线就会下降越快。

分布式数据库那点事

Oracle Rac的缓存融合技术（Cache fusion）

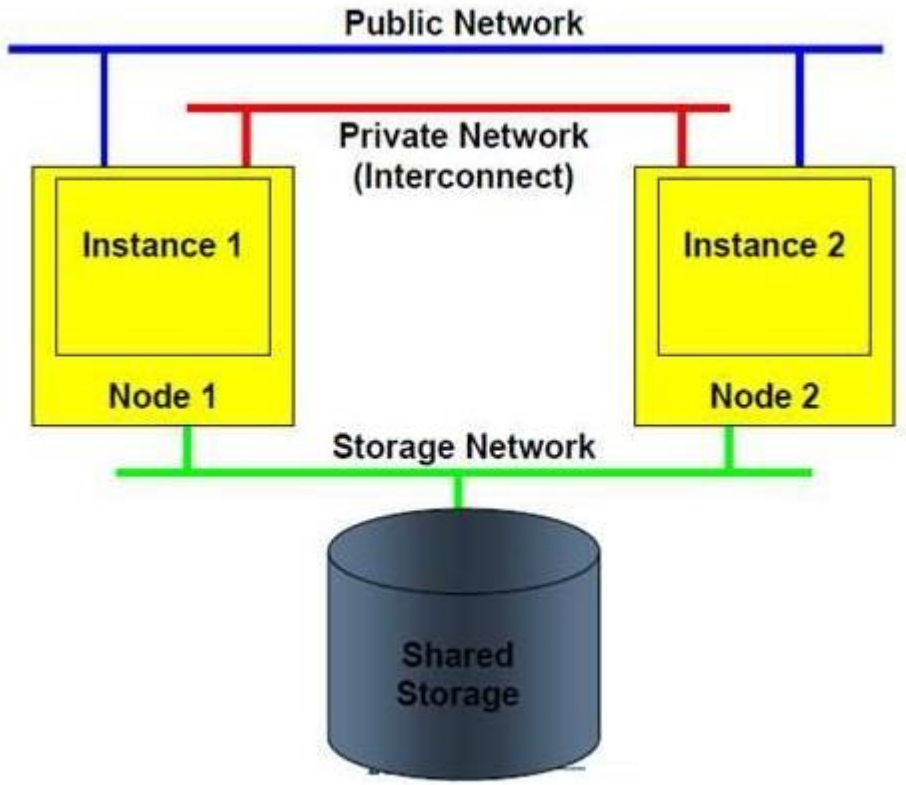
1.保证缓存的一致性
2.减少共享磁盘IO的消耗
3.在RAC环境中多个节点保留了同一份的DB CACHE





分布式数据库那点事

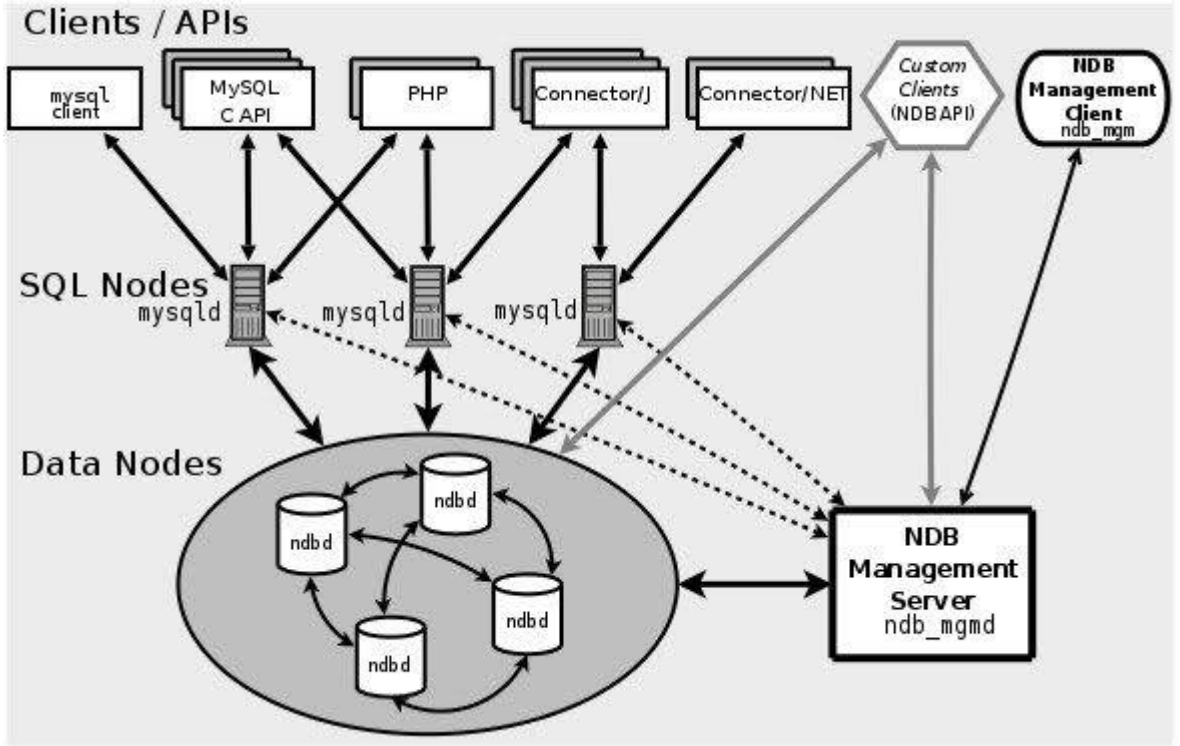
Oracle Rac双节点HA的原理和流程



在生产使用中，突然instance1 shutdown，那么在其上面没有完成的事物如何处理呢。
1）当实例1 crash后，实例2通过VIP就可以知道实例1已经down了。
2）此时需要处理的有2部分数据，一部分是commit的数据，一部分非commit数据
3）对于已经commit写入redo日志但是还没有来得及写入数据文件的记录，实例2可以访问实例1的redo log并从最后一次check point之后的信息开始实例恢复。把数据同步到最新状态。
4）对于没有commit的数据利用undo旧映像进行回滚事物。

分布式数据库那点事

MySQL Cluster



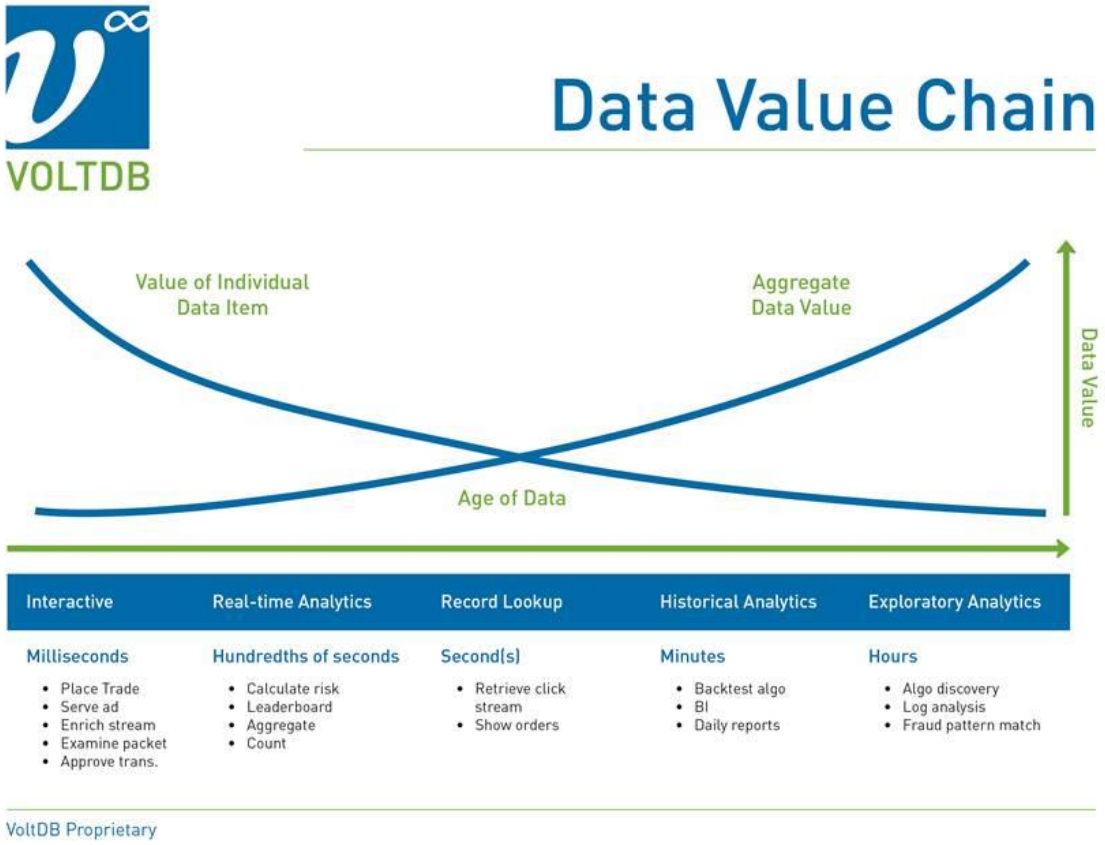
MySQL集群是一种在无共享架构（SNA，Share Nothing Architecture）系统里应用内存数据库集群的技术。这种无共享的架构可以使得系统使用低廉的硬件获取高的可扩展性。 MySQL集群是一种分布式设计，目标是要达到没有任何单点故障点。因此，任何组成部分都应该拥有自己的内存和磁盘。

优点：

1) 99.999 %的高可用性
2) 快速的自动失效切换
3) 灵活的分布式体系结构，没有单点故障
4) 高吞吐量和低延迟
5) 可扩展性强，支持在线扩容

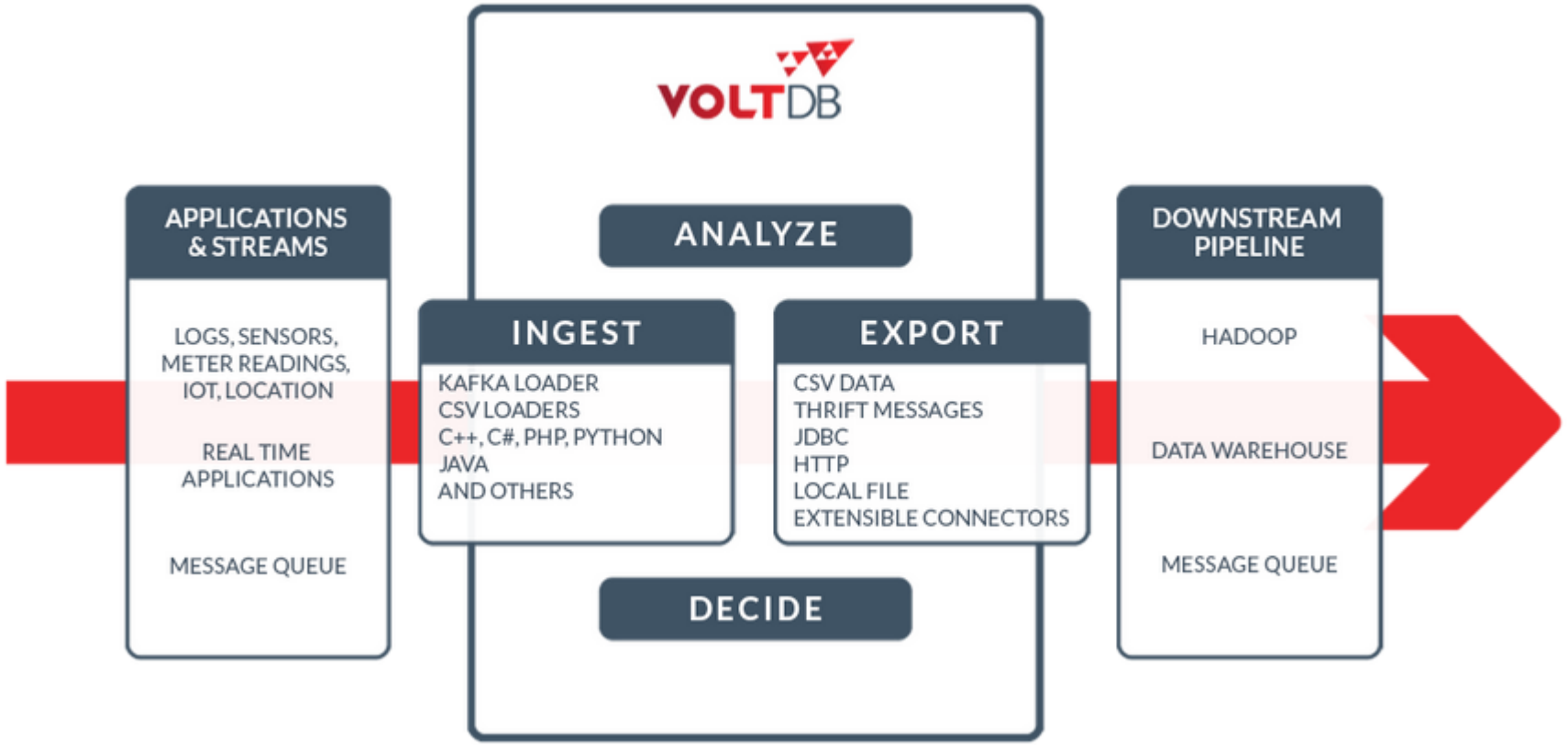
分布式数据库那点事

VoltDB



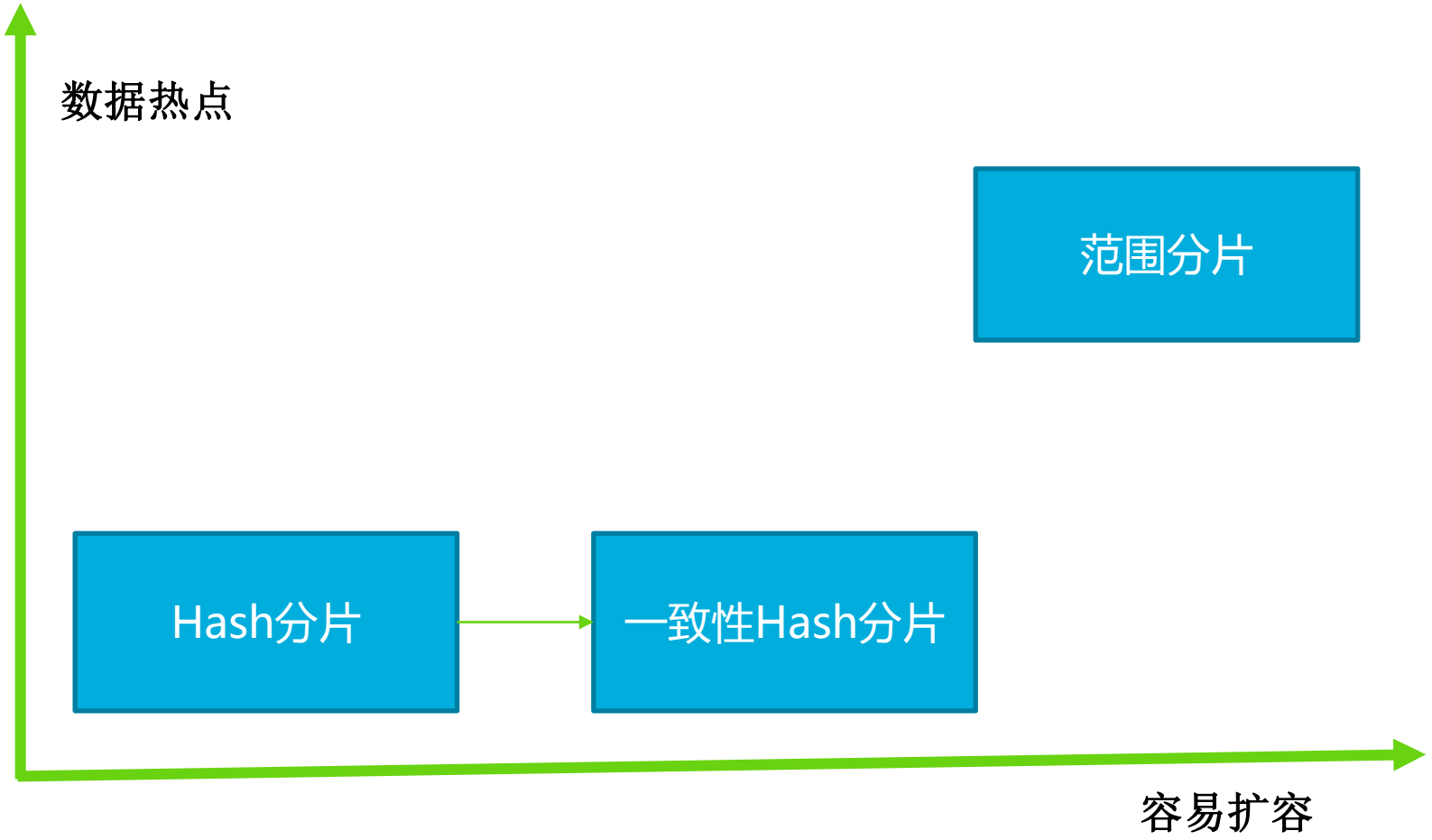
分布式数据库那点事

VoltDB Streaming Data Pipeline



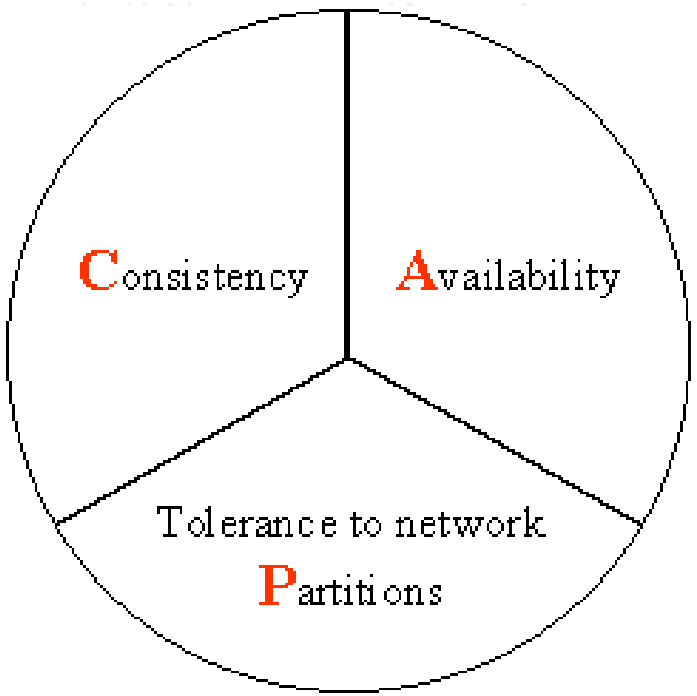
分布式数据库热点技术

数据分片与扩容



分布式数据库热点技术

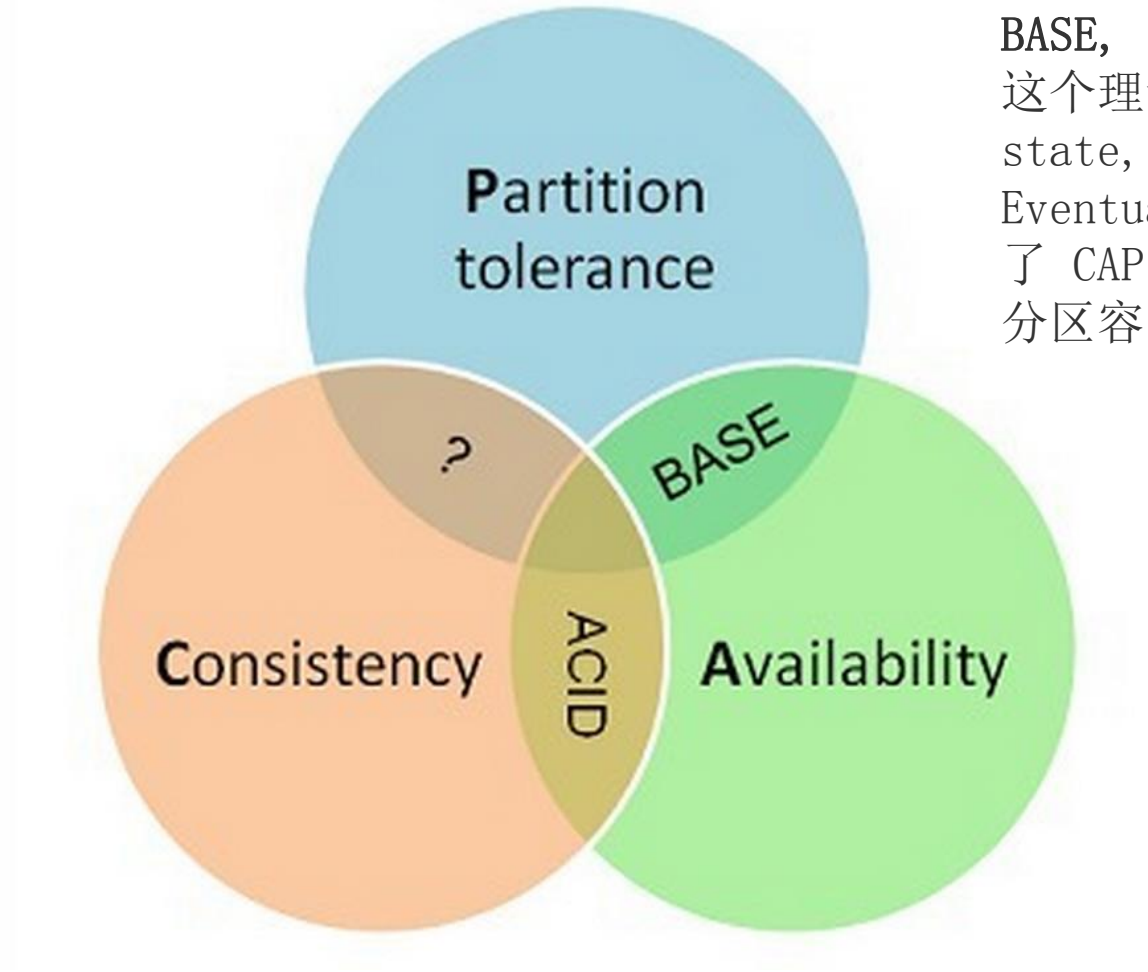
CAP理论



CA，没有扩展性，如MySQL、Oracle等
CP，不能忍受节点宕机，有Oracle RAC、Sybase集群等
AP，牺牲数据一致性，多数NoSQL系统采用

分布式数据库热点技术

ACID vs BASE



BASE，最终一致性

这个理论由 B asically A vailable， S oft state， E ventual consistency 组成。核心的概念是 Eventual consistency ——最终一致性。它局部的放弃了 CAP 理论中的“完全”一致性，提供了更好的可用性和分区容忍度。

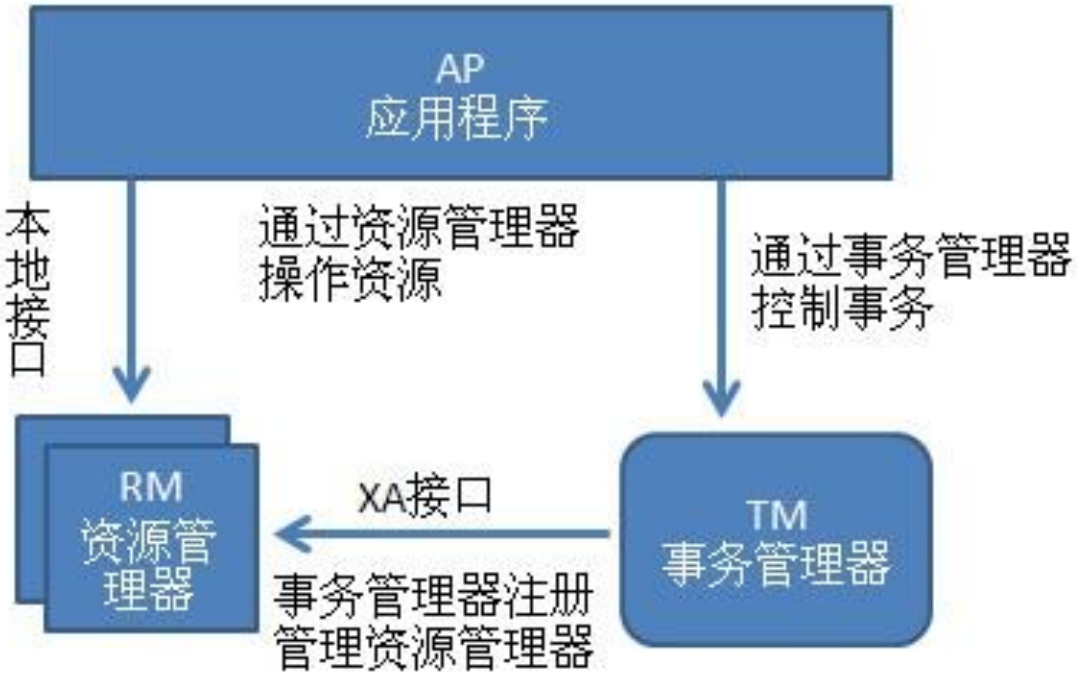
RYW（Read-Your-Writes）consistency

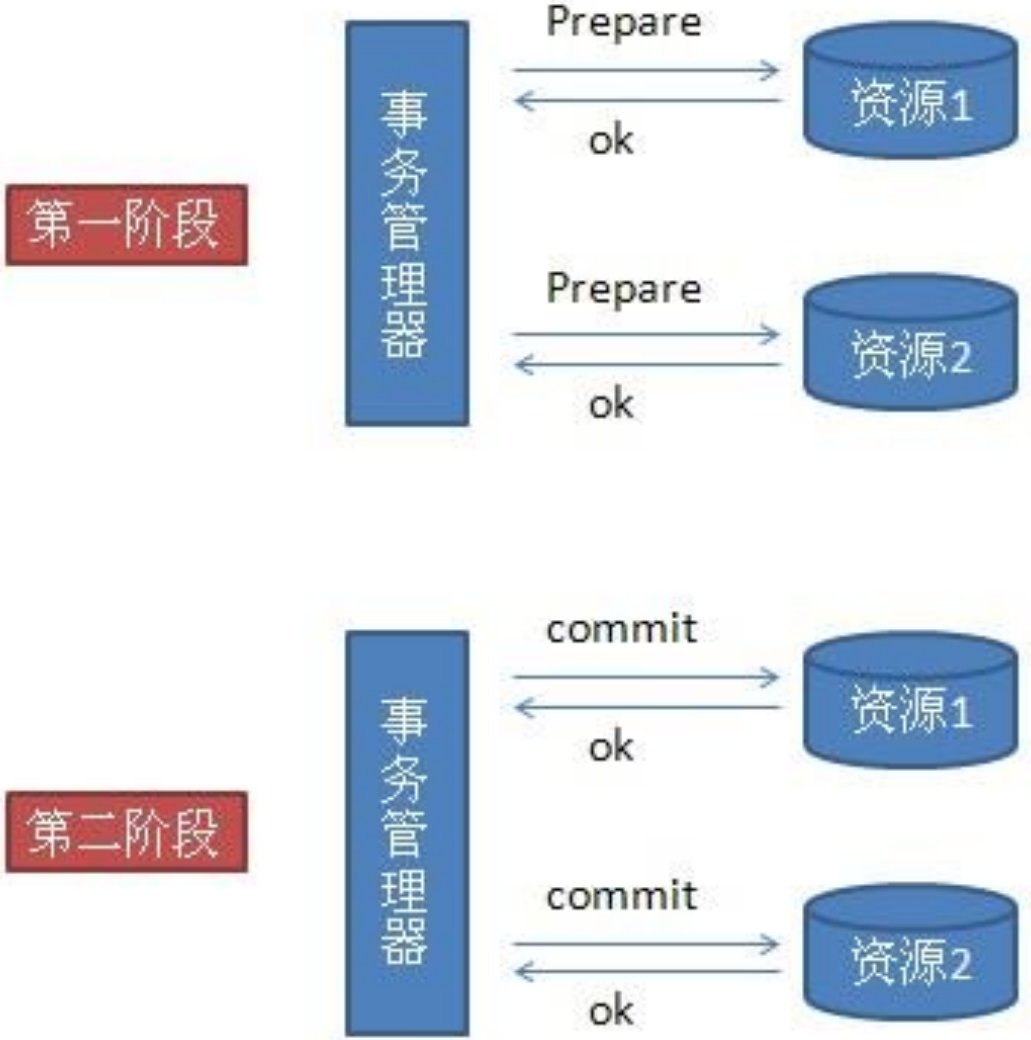
RYW consistency 是最终一致性的补充，它保证业务在会话中一定能读到上一次写入的数据。

分布式数据库热点技术

分布式事务——跨多个资源保证事务一致性

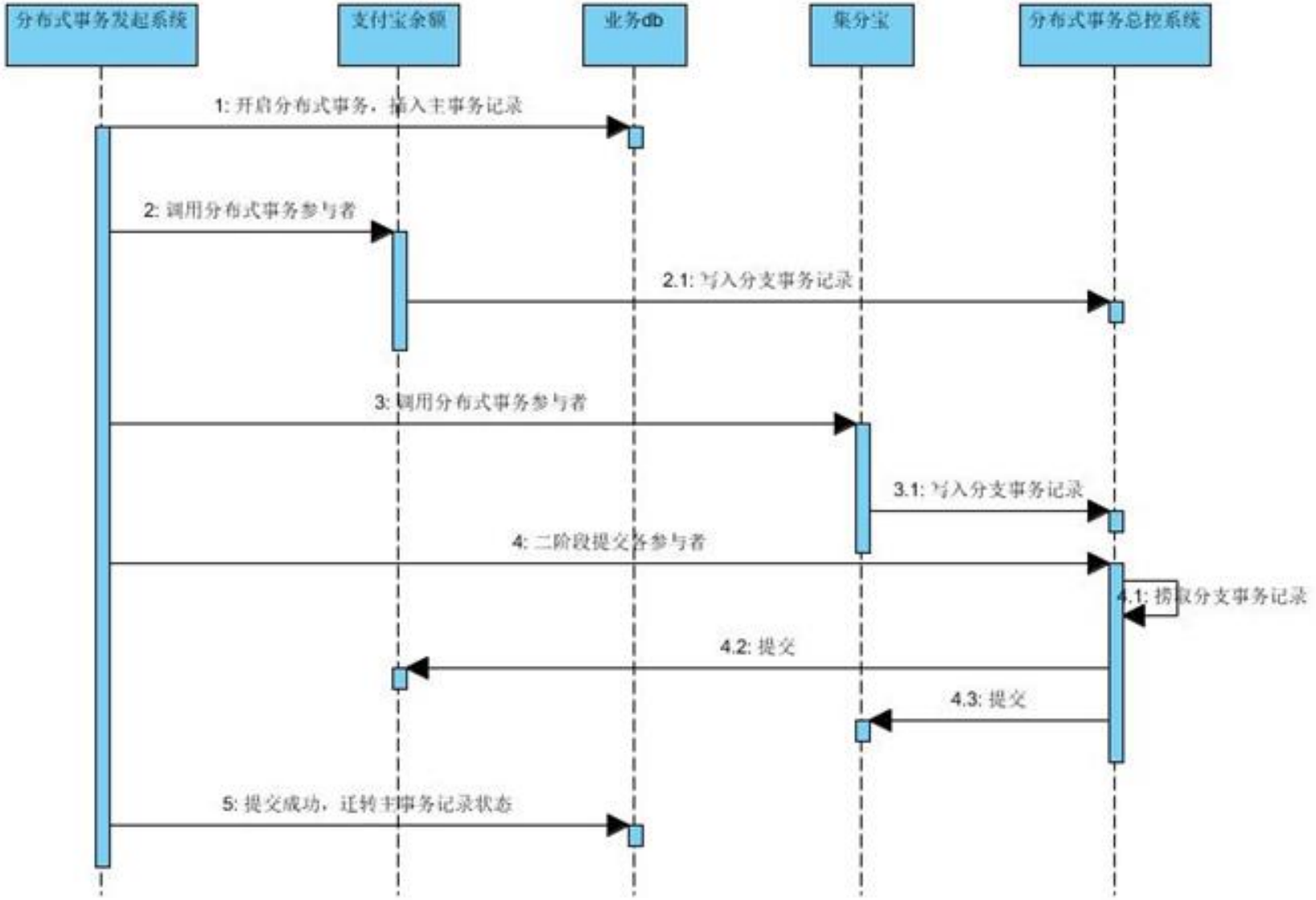
X/Open DTP模型

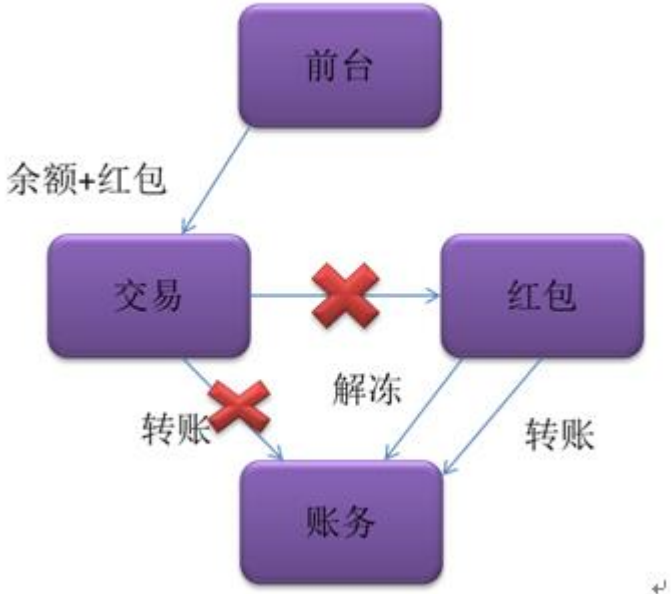




分布式数据库热点技术

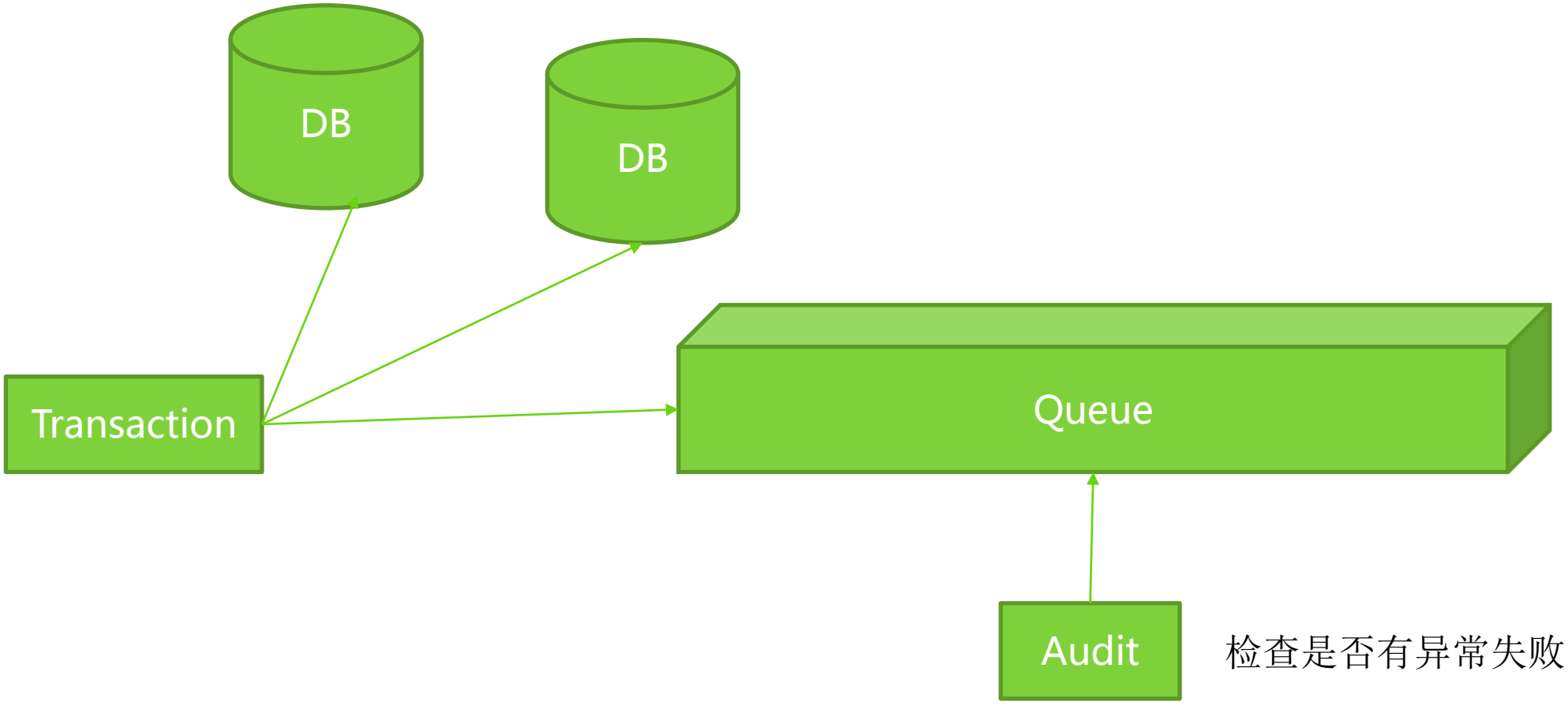
淘宝XA案例分析





分布式数据库热点技术

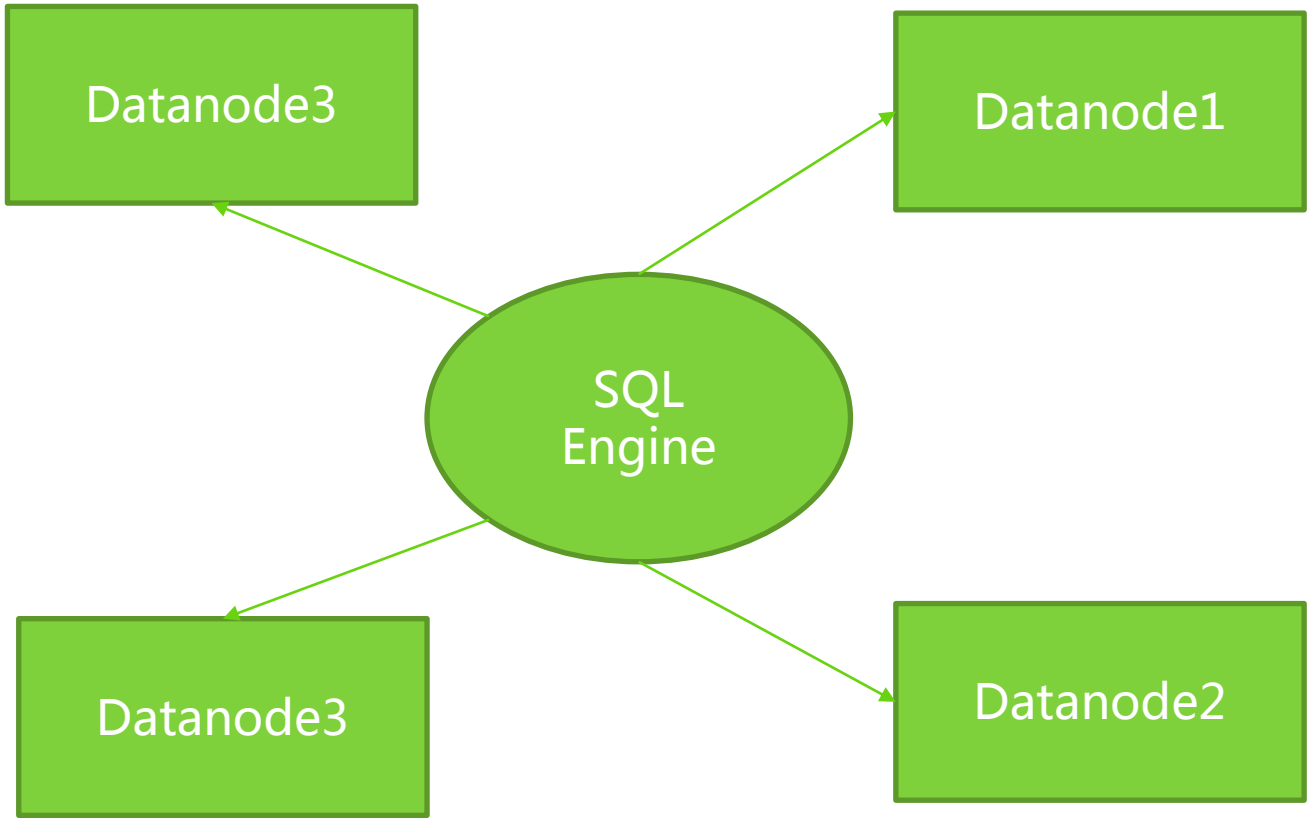
事务补偿机制——避免XA的低性能



检查是否有异常失败的事务，进行补充

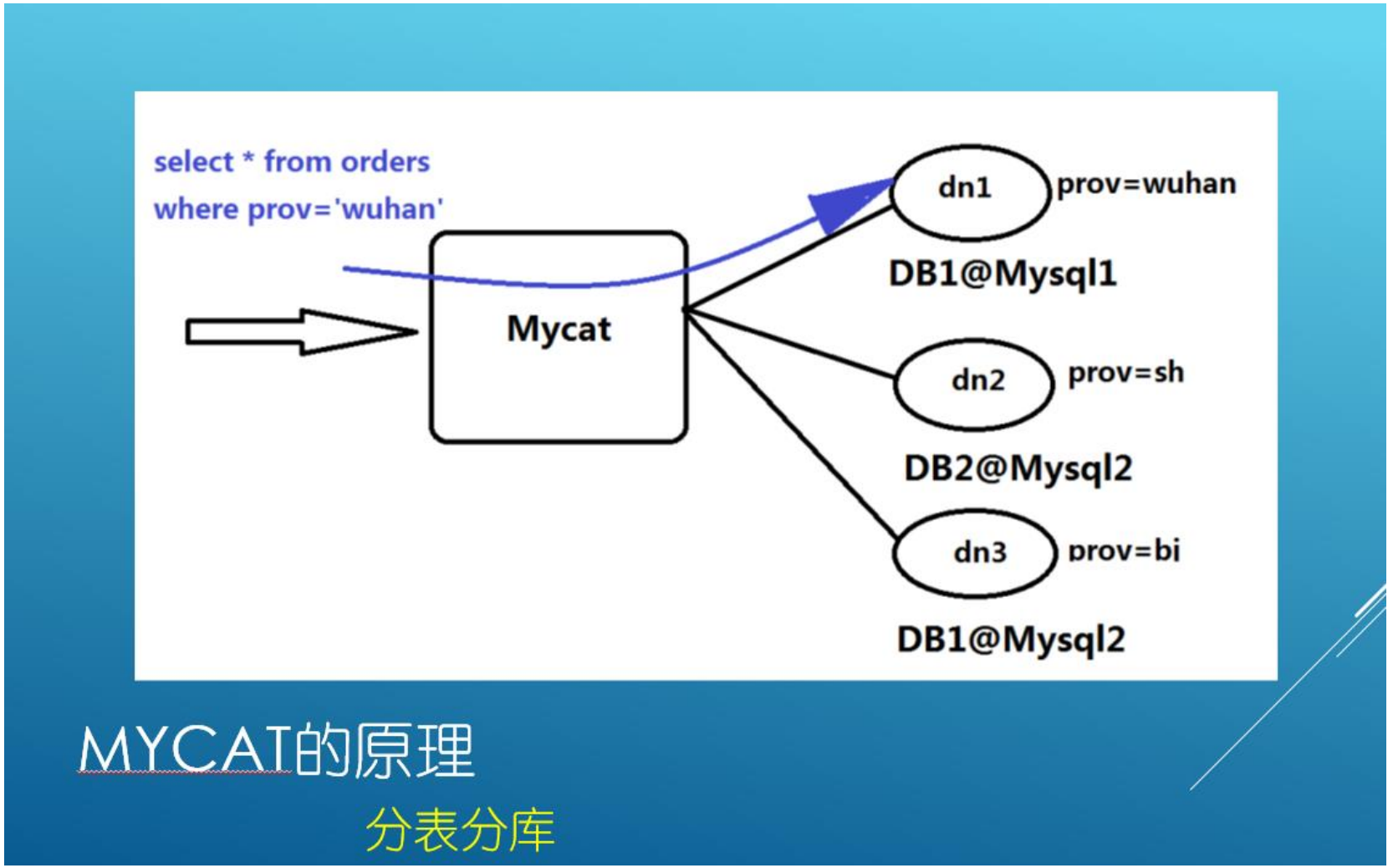
跨节点查询和Join

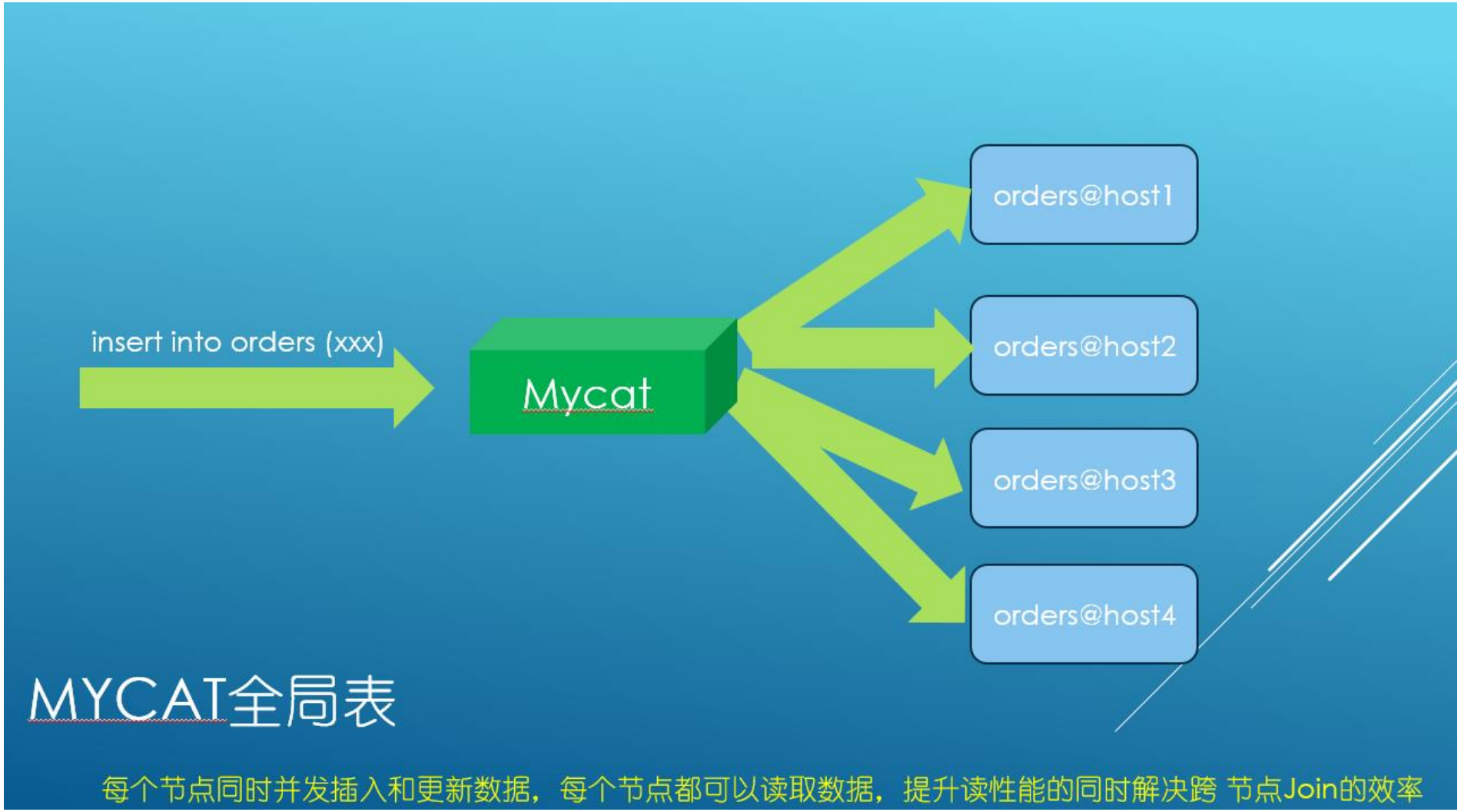
当两个表的数据分布在多个节点上时，这两个表之间的Join就是一个困难的事情

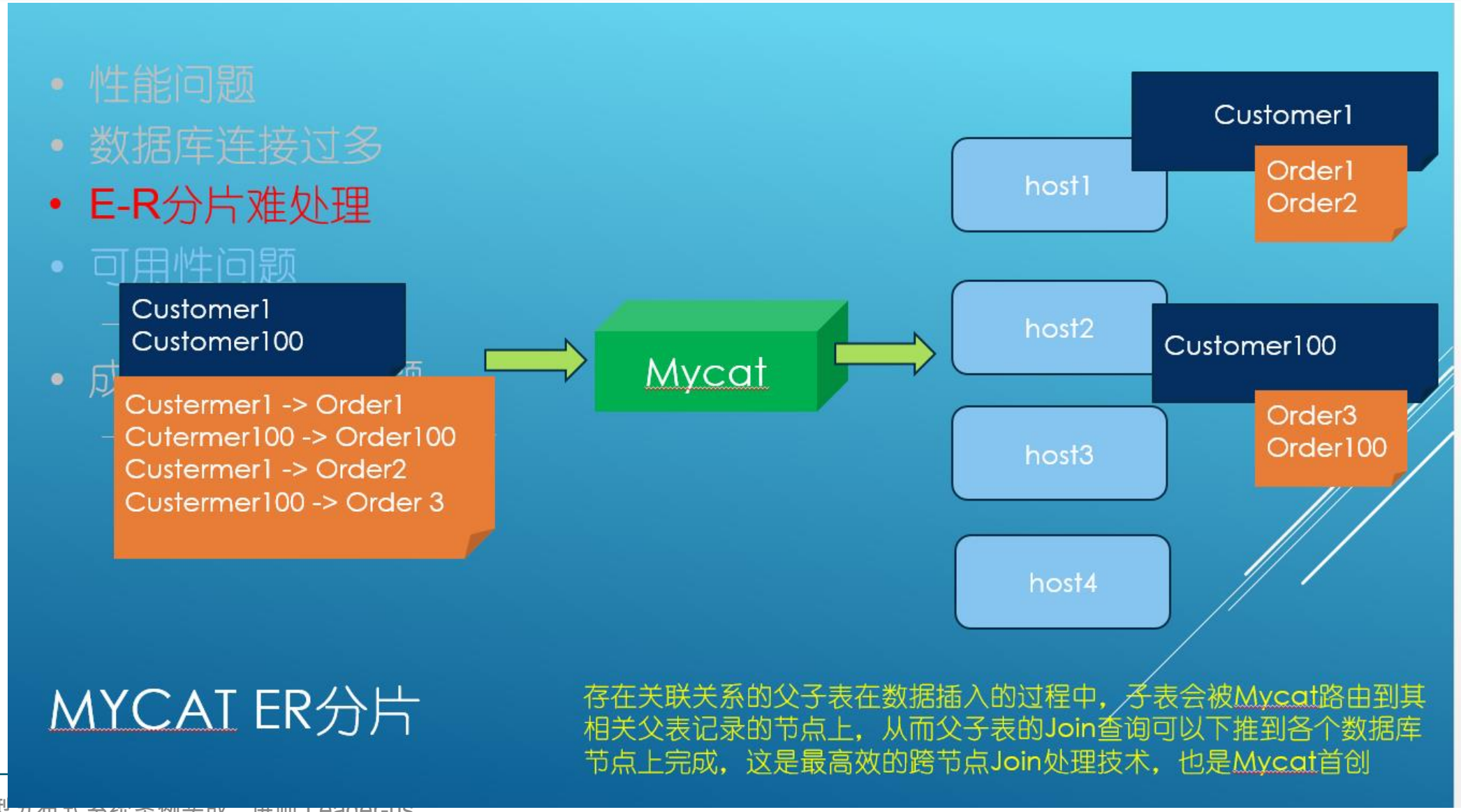


Mycat入门

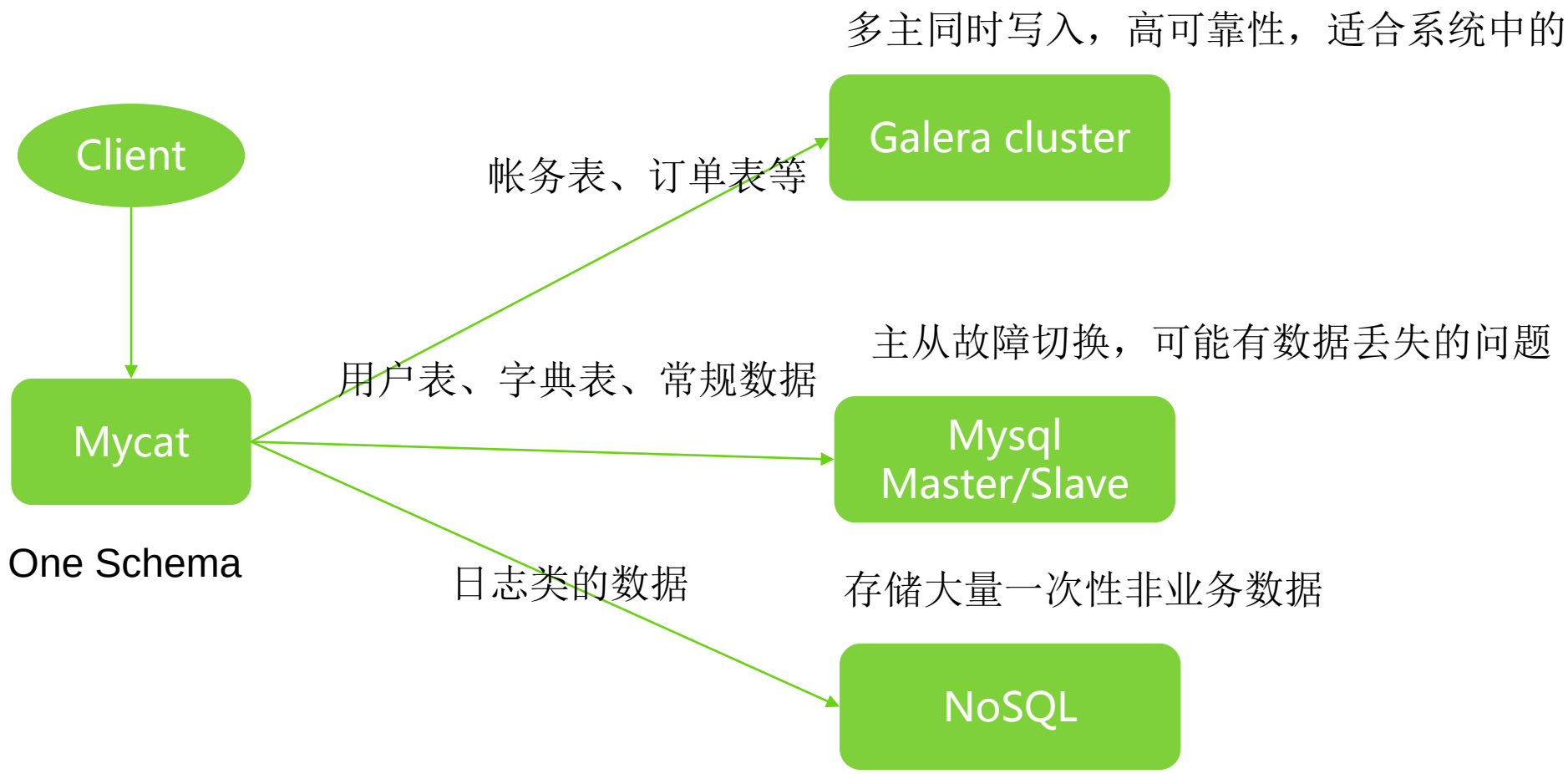
- 分布式数据库中间件
- 基于阿里开源的Cobar
- 被用于众多互联网项目
- 中国最活跃的Mycat社区


select * from orders
where prov='wuhan'
Mycat
dn1
prov=wuhan
DB1@Mysql1
dn2
prov=sh
DB2@Mysql2
dn3
prov=bi
DB1@Mysql2
MYCAT的原理
分表分库


insert into orders (xxx)
Mycat
orders@host1
orders@host2
orders@host3
orders@host4
MYCAT全局表
每个节点同时并发插入和更新数据，每个节点都可以读取数据，提升读性能的同时解决跨 节点Join的效率


性能问题
数据库连接过多
E-R分片难处理
可用性问题
Customer1
Customer100
Custermer1 -> Order1
Cutermer100 -> Order100
Custermer1 -> Order2
Custermer100 -> Order 3
Mycat
host1
Customer1
Order1
Order2
host2
Customer100
Order3
Order100
host3
host4
MYCAT ER分片
存在关联关系的父子表在数据插入的过程中，子表会被Mycat路由到其相关父表记录的节点上，从而父子表的Join查询可以下推到各个数据库节点上完成，这是最高效的跨节点Join处理技术，也是Mycat首创


全局表技术
独创的ER关系分片
基于Catlet的两表自动Join模块
复杂SQL可通过用户自定义的Catlet进行处理
未来引入Sorm/Spark Stream等技术来处理海量数据计算
Catlet是Java编写的一段程序，类似数据库中的存储过程，可以实现任意复杂SQL的Join、Group、Order等功能
MYCAT跨分片解决方案汇总


多主同时写入，高可靠性，适合系统中的关键表
Client
Galera cluster
帐务表、订单表等
主从故障切换，可能有数据丢失的问题
用户表、字典表、常规数据
Mycat
Mysql
Master/Slave
One Schema
日志类的数据
存储大量一次性非业务数据
NoSQL

Thanks

FAQ时间