PingCAP TiDB

变革下的金融业数据库转型之路

韩锋

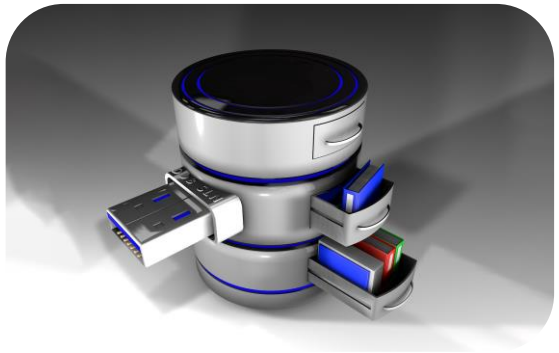
SphereEx 联合创始人

CCIA（中国计算机协会）常务理事，前Oracle ACE，腾讯TVP，阿里云MVP，dbaplus等多家社群创始人或专家团成员。有着丰富的一线数据库架构、软件研发、产品设计、团队管理经验。曾担任多家公司首席DBA、数据库架构师等职。在云、电商、金融、互联网等行业均有涉猎，精通多种关系型数据库，对NoSQL及大数据相关技术也有涉足，实践经验丰富。曾著有数据库相关著作《SQL优化最佳实践》、《数据库高效优化》。

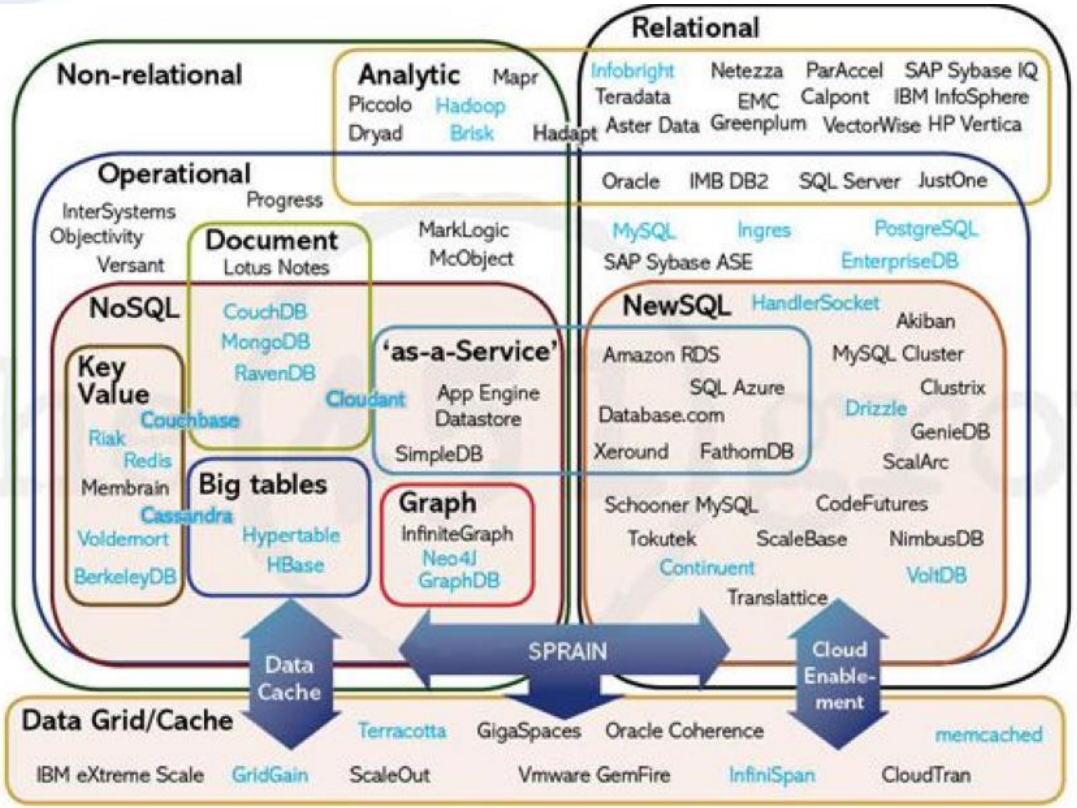
背景与趋势

背景

金融行业，作为数据使用的“高地”，长期以来非常重视数据技术发展。作为金融行业的重要组成部分，银行业一致致力于构建稳健的数据基础设施。作为数据的主要载体，数据库在其中扮演着非常重要的角色。随着近些年来，以分布式、云原生、多模异构等为代表的新型数据库产品出现，银行业正在经历新的一轮技术迭代更新周期。但因银行业务系统非常复杂，很难找到一种“完美”产品覆盖所有业务场景。如何根据金融业务场景，找到最适合的数据库成为广大金融从业者需面对的问题。



趋势



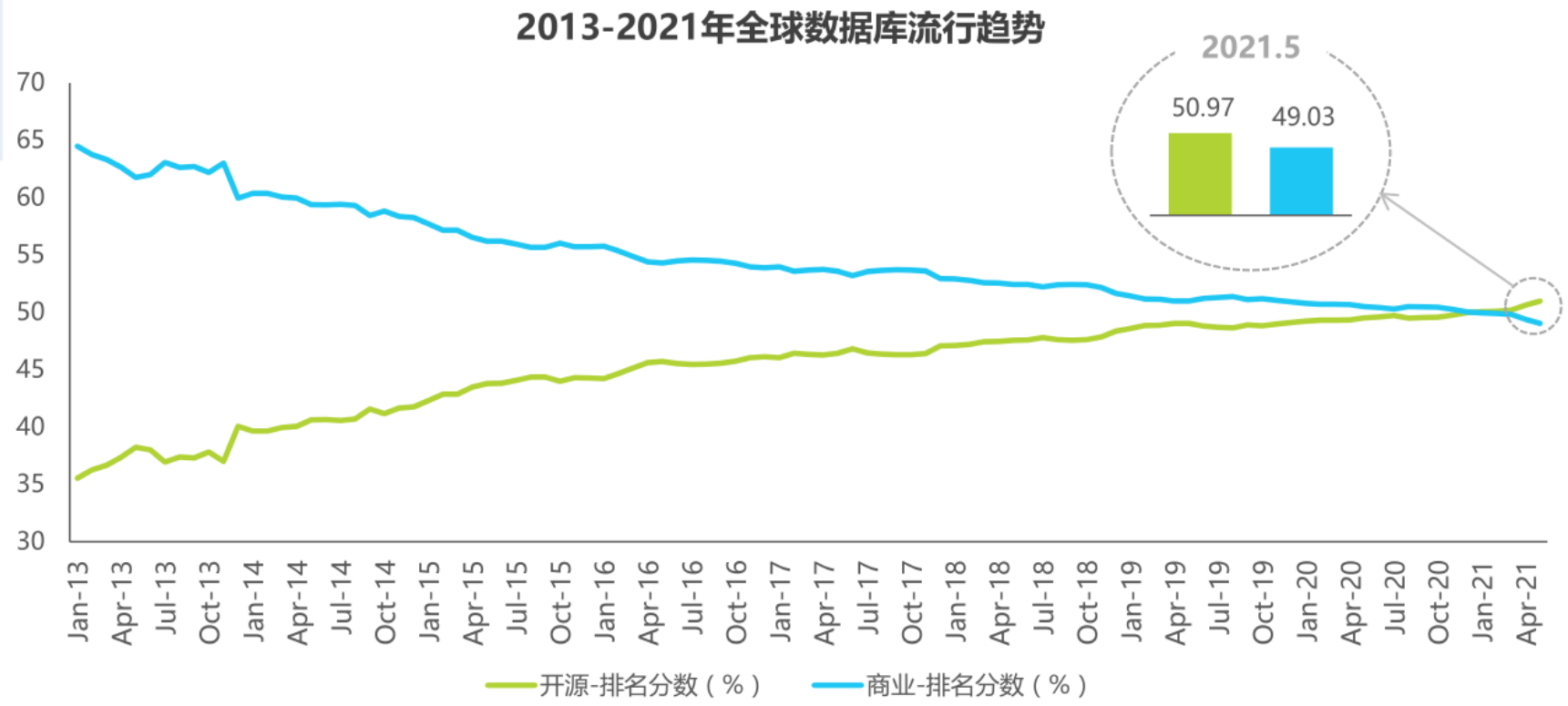
多场景

多形态

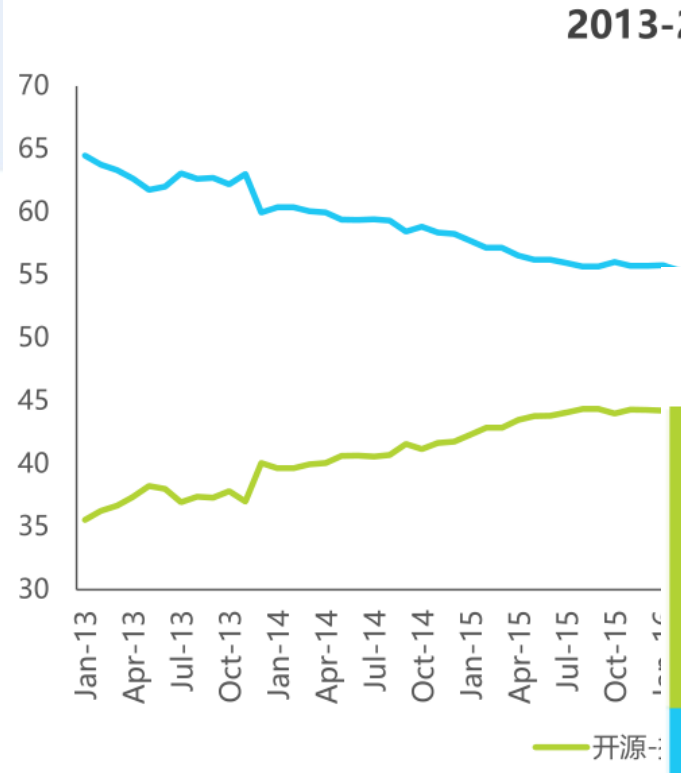
多技术栈

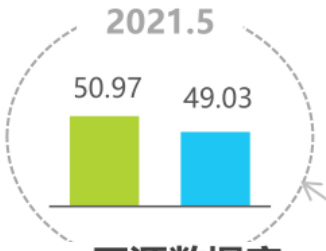
多元路线

趋势


2013-2021年全球数据库流行趋势
2021.5
50.97
49.03
70
65
60
55
50
45
40
35
30
Jan-13
Apr-13
Jul-13
Oct-13
Jan-14
Apr-14
Jul-14
Oct-14
Jan-15
Apr-15
Jul-15
Oct-15
Jan-16
Apr-16
Jul-16
Oct-16
Jan-17
Apr-17
Jul-17
Oct-17
Jan-18
Apr-18
Jul-18
Oct-18
Jan-19
Apr-19
Jul-19
Oct-19
Jan-20
Apr-20
Jul-20
Oct-20
Jan-21
Apr-21
开源-排名分数（%）
商业-排名分数（%）

趋势

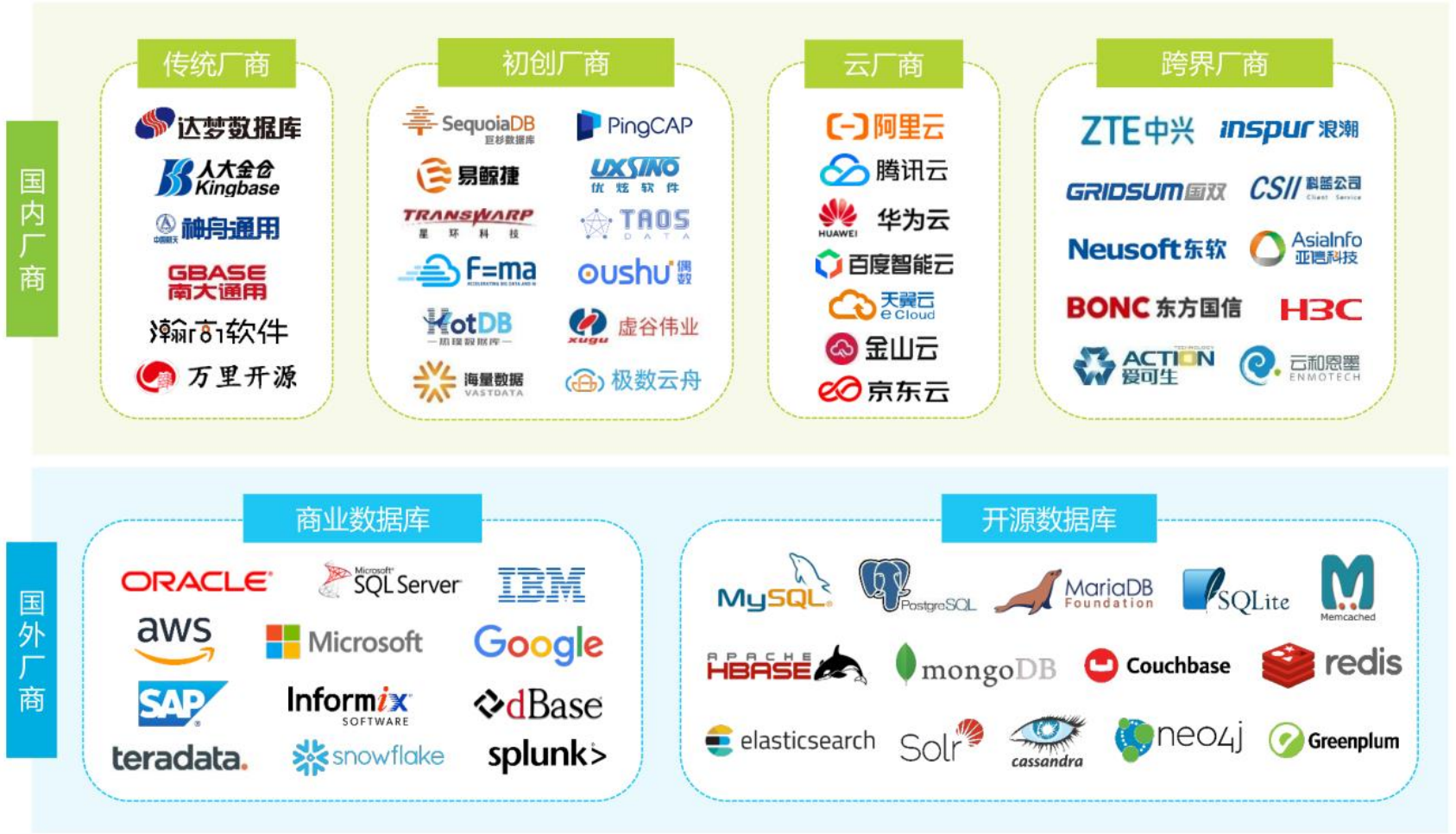




开源数据库 vs 商业数据库

	前期：选型采购	中期：开发部署	后期：运维使用
开源	**选型成本** 开源产品/社区繁多造成选型障碍	**开发成本** 自研案例缺失 **部署成本** 问题不确定性	**扩展限制** 企业级性能差 **Bug优化** 部分Bug较多
开源	**评测成本** 需要结合需求做大量的POC测试	**迁移成本** 源码、接口不兼容问题	**配套升级** 缺乏配套产品 **人力投入** 大量人力运维
商业	**产品费用** 商业license价格高昂	**服务费用** 包括个性化定制、部署、培训、后续运维的费用	
商业	**咨询费用** 前期咨询产生的费用		

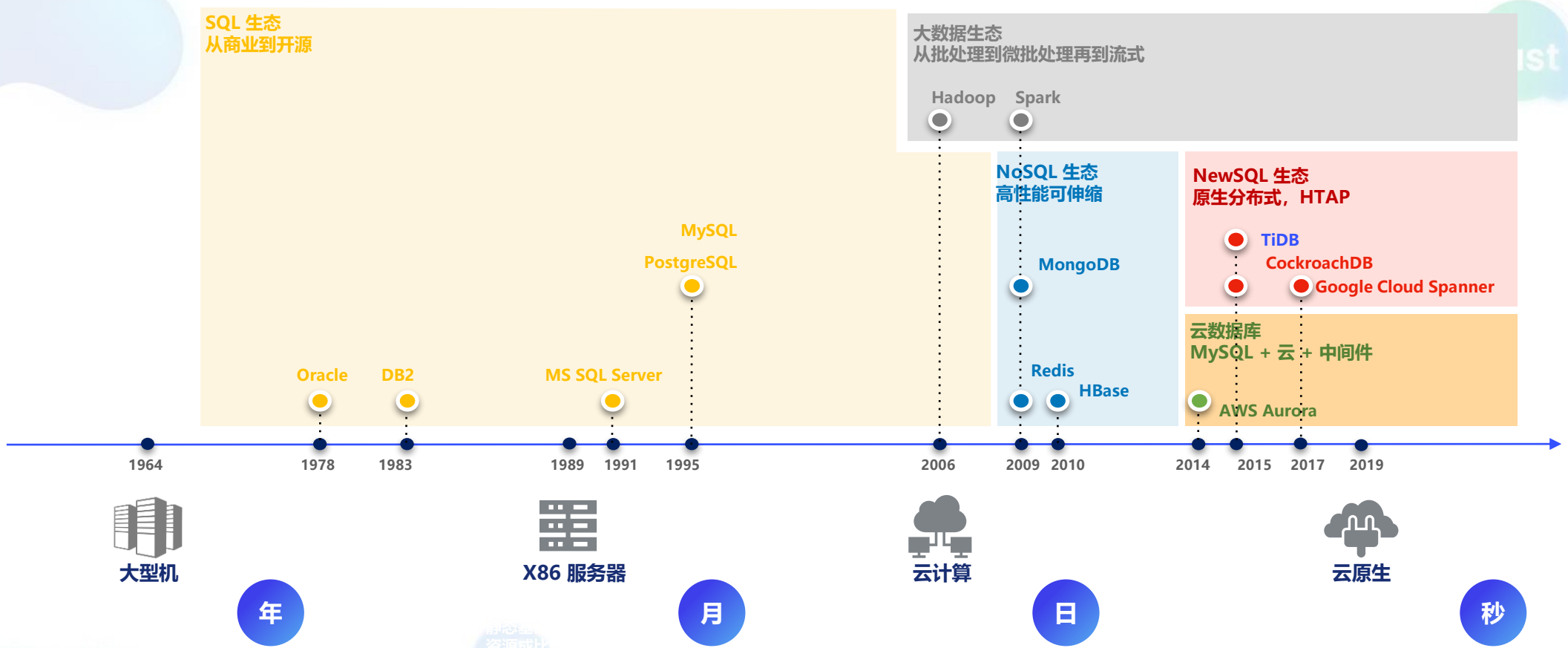
趋势


国内厂商
传统厂商
达梦数据库
人大金仓 Kingbase
神舟通用
GBASE 南大通用
瀚高软件
万里开源
初创厂商
SequoiaDB 巨杉数据库
PingCAP
易鲸捷
UXSINO 优炫软件
TRANSWARP 星环科技
TAOS DATA
Fema
oushu 偶数
HotDB 热璞数据库
虚谷伟业
海量数据 VASTDATA
极数云舟
云厂商
阿里云
腾讯云
华为云 HUAWEI
百度智能云
天翼云 e Cloud
金山云
京东云
跨界厂商
ZTE 中兴
inspur 浪潮
GRIDSUM 国双
CSII 科蓝公司
Neusoft 东软
AsiaInfo 亚信科技
BONC 东方国信
H3C
ACTION 爱可生
云和恩墨 ENMOTECH
国外厂商
商业数据库
ORACLE
Microsoft SQL Server
IBM
aws
Microsoft
Google
SAP
Informix SOFTWARE
dBase
teradata.
snowflake
splunk>
开源数据库
MySQL
PostgreSQL
MariaDB Foundation
SQLite
Memcached
APACHE HBASE
mongoDB
Couchbase
redis
elasticsearch
Solr
cassandra
neo4j
Greenplum

趋势

《金融科技（FinTech）发展规划（2019-2021）》中明确指出：“加强分布式数据库的研发应用。做好分布式数据库金融应用的长期规划，加大研发与应用投入力度。有计划、分步骤稳妥推动分布式数据产品先行先试，形成可借鉴、能推广的典型案例和解决方案，为分布式数据库在金融领域的全面应用探明路径。建立健全产学结合、校企协同的人才培养机制，持续加强分布式数据库底层和前沿技术研究，制定分布式数据库金融应用标准规范，从技术架构、安全防护、灾难恢复等方面明确管理要求，确保分布式数据库在金融领域的稳妥应用。”

趋势


SQL 生态
从商业到开源
大数据生态
从批处理到微批处理再到流式
Hadoop
Spark
NoSQL 生态
高性能可伸缩
NewSQL 生态
原生分布式，HTAP
TiDB
CockroachDB
Google Cloud Spanner
MongoDB
MySQL
PostgreSQL
云数据库
MySQL + 云 + 中间件
Oracle
DB2
MS SQL Server
Redis
HBase
AWS Aurora
1964
1978
1983
1989
1991
1995
2006
2009
2010
2014
2015
2017
2019
大型机
X86 服务器
云计算
云原生
年
月
日
秒
私有，专用，封闭，昂贵
CPU、内存、磁盘等
按需启动，弹性扩展
微服务化，容器化包装，动态管理

分布式数据库

PingCAP
DevCon 2022

概述

分布式数据库，作为一种新型数据库产品架构，正处于蓬勃发展阶段。其具备的数据分片管理、分布式事务、读写分离等关键分布式能力，能够很好地满足企业在高性能、大数据量等多种业务场景。近年来，各国产厂商都在积极推进分布式数据库产品的研发，技术已经逐步成熟。金融行业，作为数字化转型的先导性行业，对数据基础设施有着更高的要求。分布式数据库的出现，恰好可以满足金融企业迫切需求。随着近些年来，分布式数据库的成熟，并在金融行业已有成功案例投入生产系统使用。相信，两者的结合，必定在未来能擦出更多火花，促进金融行业在新时代转型发展。

分类

最佳实践：解耦路线依赖

通过数据库的技术标准化和轻量化工作，形成统一的数据库使用规范，解耦应用和底层数据库技术架构，在标准数据库协议及语义下，可以很轻松的更换数据库架构。通过构建异构间数据同步、流量接入控制（限流、灰度等）、全局数据服务（如事务、快照等），实现业务的无感切换和迁移回退，保证最大的灵活可控。

技术路线	路线优点	路线缺点	适用场景
分布式中间件+单机数据库	• 稳定性高 • 极致性能 • 超大规模	• 功能不完整 • 潜力不足	• 大规模OLTP • 高性能高并发场景
原生分布式数据库	• 高弹性 • 无侵入 • 功能完整	• 风险中高 • 稳定性待验	• 中小规模OLTP • 混合负载场景 • 高可用场景
业务自研+开源单机数据库	• 灵活可控 • 风险低	• 侵入强 • 成本高	• 小规模OLTP • 数量众多外围系统

选择

系统分类	典型系统	规模占比
业务类 （一类）	• **核心业务系统** 存储款、查询、对账、转账、信贷、代收付等 • **信贷管理系统** • **支付与清算系统** • **理财系统** • **中间业务系统**	42%
渠道前置类 （二类）	• **前置类系统** 渠道整合、中间业务、交易/数据交互 • **自助类系统** ATM、查询机 • **网银\手机银行** 网银、手机银行、支付宝、微信等 • **柜台交易系统** 网点渠道类系统，如柜面 • **呼叫中心电话银行**	20%
管理决策类 （三类）	• **业务管理类系统** 财务管理、信贷管理、资产管理、报表（商业智能）类、客户关系管理、企业资源管理、风险管理系统 • **人行管理类系统** 反洗钱、反假币、监管报送类、金融审计和稽查 • **门户网站** 宣传和信息发布	36%
其他支持类 （四类）	• **其他系统** 如办公OA系统	2%

需求分类	典型系统	适用场景						建议架构
		业务场景	数据规模	事务一致性	访问负载	分析能力	应用适配能力	
事务类	• 渠道类(单元化) • 核心业务类(单元化) • 其他类	• 联机交易 • 轻量数据分析	中	• 单库强一致 • 多库应用层解决	• 高并发 • 小数据量事务读写 • 小数据量分析	弱	高	• 单机集中式架构 • 垂直资源扩展 • 外置高可用及容灾
	• 渠道类 • 其他类	• 联机交易 • 简单事务 • 大并发	大	• 应用层最终一致	• 点查、点写 • 有限规模关联分析	弱	低 （应用逻辑入侵）	• 分布式架构(单机引+中间件计算) • 应用入侵,需改造 • 外置高可用及容灾
	• 核心业务类 • 管理决策类 • 其他类	• 联机交易 • 批量处理 • 实时分析 • 混合负载	大	• 强一致	• 点查、点写 • 有限规模关联分析	弱	高 （对应用透明）	• 分布式架构(分布式存储+计算引擎) • 透明分布式 • 内置高可用及容灾
分析类	• 管理决策类 • 其他类	• 批量处理 • 复杂分析 • 非实时查询	大	• 弱一致	• 复杂分析	强	低 （应用逻辑入侵）	• 分布式架构(MPP) • 透明分布式 • 外置高可用及容灾
混合类	• 渠道类(新业务) • 核心业务类 • 管理决策类 • 其他类	• 联机交易 • 批量处理 • 实时分析 • 混合负载 • 轻量级复杂分析	大	• 强一致	• 高并发 • 小数据量事务读写 • 复杂分析	强	高 （对应用透明）	• 分布式数据库 • 透明分布式 • 内置高可用及容灾

选择

指标名	指标说明	适配系统
业务类型	OLTP	核心业务、渠道类
	OLAP	管理决策类
	HTAP	管理决策类
事务处理数（TPS）	低：< 100	管理决策类
	中：100~2000	核心业务类
	高：2000~10000	渠道类
	超高：>10000	渠道类
事务一致性	强一致	核心业务、渠道类
	最终一致	管理决策类
并发数	低：<200	管理决策类
	中：200~2000	核心业务类
	高：>2000	渠道类
数据量（TB）	低：<10TB	渠道类
	中：10~100TB	核心业务、渠道类
	高：>100TB	渠道、管理决策类
操作特征	读多写少	渠道、核心业务类
	读少写多	管理决策类
	特殊（批量导入）	管理决策类
结构复杂度	低：简单对象（表、索引）	渠道类
	中：复杂对象（视图、序列等）	核心业务、渠道类
	高：自定义对象	渠道、管理决策、其他类
SQL复杂度	低：点查、小范围查询	核心业务类
	中：关联、子查询、聚合查询	渠道、核心业务类
	高：复杂查询（如多层嵌套）	管理决策类
	超高：库内计算（如存储过程）	管理决策类
数据耦合性	低：无交互，完美分片	渠道类
	中：少量交互	渠道、核心业务类
	高：大量交互	渠道、管理决策类

指标	适配画像	备注
数据规模	规模中等及以上	•小规模，可考虑单机库；此架构适合大规模 •大规模下，若单元化已可考虑，保留未来扩展性
查询并发数	中等及更高	•可扩展计算节点，适配高并发查询场景 •低延时场景，可考虑采用应用端分片计算逻辑
事务一致性	•本地强一致 •全局最终一致	分布式事务存在较大开销，建议优先用本地事务或最终一致性事务
操作特征	不限	不限操作特征，在分片下可实现读写能力扩展
结构复杂度	简单	•简单数据对象（如表、索引） •复杂对象考虑在应用侧解决
SQL复杂度	•简单（不限） •中等复杂（跨跨片，有限规模） •高度复杂（单片内）	•在单片下，复杂语句可下推到单机解决 •在跨分片下，复杂语句可通过上层计算引擎解决，但受限于规模）；在大规模下不合适
数据耦合性	•无（完美分片） •有（限部分场景）	•完美分片场景都适用 •需要跨片处理，限部分读场景或低频写入

转型之路

PingCAP
DevCon 2022

策略

《金融科技（FinTech）发展规划（2019-2021）》中明确指出：“加强分布式数据库的研发应用。做好分布式数据库金融应用的长期规划，加大研发与应用投入力度。有计划、分步骤稳妥推动分布式数据产品先行先试，形成可借鉴、能推广的典型案例和解决方案，为分布式数据库在金融领域的全面应用探明路径。建立健全产学结合、校企协同的人才培养机制，持续加强分布式数据库底层和前沿技术研究，制定分布式数据库金融应用标准规范，从技术架构、安全防护、灾难恢复等方面明确管理要求，确保分布式数据库在金融领域的稳妥应用。”

- **供应链安全**

无论企业选择商业产品还是开源项目，只有将核心技术掌握在手中，才能做到真正安全。核心技术掌握在谁手里，并不是指源代码。源代码不是技术，只是技术的载体。脱离了技术主导者，即便拥有源代码，也无法持续发展。只有做到从根本上解决技术供应链的安全和可持续发展问题，才能真正意义上实现。

- **金融合规要求**

金融业，作为数据密集行业，业务依赖数据的产生与流转。因而金融业对交易与数据的承载者—数据库，提出非常高的要求，诸如数据库的稳定性、可靠性等。此外，金融业的数据通常也具有高价值，因此对于数据安全方面也同样有着极高的要求。作为涉及国计民生的重要行业，国家也将金融业作为重点监管行业。

- **金融创新基础**

金融业务正在发生变化。以银行业为例，过去靠人工完成拉存款、放贷款，现在随着互联网变，催生如在线银行、手机银行、数字银行等多种业务形式。上述对底层基础设施提出新的要求，如何通过底层技术革命，促使金融服务的方式发生代际更替。作为代表，数据库也被赋予更高的要求，包括分布式、混合负载、多模、智能应用等能力受到更多关注。这些都对金融企业如何使用好数据库、乃至如何掌控这一技术栈的发展提出了更高要求。

阶段

在信创改造过程，大致将其划分为四个阶段。

- **选型评估阶段**

完成信创技术栈的选型，并开展相关技术栈培训，做好基础储备工作；同步完成资源、成本评估。

- **研发测试阶段**

完成业务系统针对信创技术栈的改造及测试，其中涉及到较大的成本（人力、时间）的投入。

- **验证并行阶段**

完成业务系统改造后，需针对新平台的功能、稳定性、可用性等方面进行验证。为保证替换平稳，推荐使用并行方式进行。

- **上线支持阶段**

在系统已经得到充分验证后，将业务系统从原技术栈完全迁移到新技术栈。此阶段需重点解决迁移及出现问题的保障维护方面。

选型评估阶段

迁移评估

对迁移工作做整体规划，制定原则，明确迁移范围、方式、是否停机、窗口期等。除技术外，其他如组织、管理、资源等，也在这一阶段一并考虑。迁移是个很复杂的过程，涉及的方方面面很多，尽量在项目之初就有个全面的掌握。

业务梳理

在迁移准备阶段，就对涉及的业务有个全面的梳理非常有必要。这里需要梳理的信息，非常宽泛。包括但不限于对业务系统涉及的软硬件环境、与数据库交互、业务系统间调用关系等。后续在做应用系统改造规划中，上述信息非常重要，其有助于评估工作难点、工作量等。

使用者	xx 部门/xx 系统
交互方式	□人机交互　□应用交互　□数据交互
使用终端	□PC　□APP　□其他
使用特征	使用时长：__时__分 ~ __时__分 高峰市场：__时__分 ~ __时__分 在线用户数：____ 并发用户数：____ 并发最大数：____
重要级别	□A+　□A　□B　□C
可用时长	□7x24　□5x8　□5x12　□_____
关联业务	业务 1：业务名称，关联简述 业务 2：业务名称，关联简述 业务 3：业务名称，关联简述 ...

系统来源	□自建　□外采　□合作　□云　□其他
基础技术栈	
开发语言	□Java　□C/C++ □Go　□Python　□其他　□混合
语言版本	
应用拓扑	
交互方式	连接方式：□IP　□VIP　□DNS　□OTHER 读写分离：_____
国产化诉求	芯片：_____ 操作系统：_____ 中间件：_____ 其他组件：_____

数据存量	#单机/集中式 单机数据规模：_____ 集群数据估摸：_____ #主从/分布式/云 单体/分片数据规模：_____ 数据副本数：_____ 总数据规模：_____
数据增量	_____/ 月
数据特征	冷热分层：_____ 数据压缩：_____ 数据转储：_____ 数据热点：_____
计算场景	□OLTP　□OLAP　□HTAP　□_____ 说明：
计算指标	QPS：_____ TPS：_____ RT：_____ R/W：_____
日志规模	_____GB/日
日志峰值	_____GB/分钟
数据库热点	
一致性要求	□强一致　□弱一致 说明：_____
扩展性要求	接入扩展：□水平　□无 计算扩展：□垂直　□水平　□混合　□无 存储扩展：□垂直　□水平　□混合　□无
高可用要求	□冷备　□热备　□双活　□多活　□无
高可用指标	同城：RPO=　RTO= 异地：RPO=　RTO=
高可用方案	
数据库特性	□特殊对象：_____ □库内计算(包、存储过程、函数、触发器等) □其他特性：_____

选型评估阶段

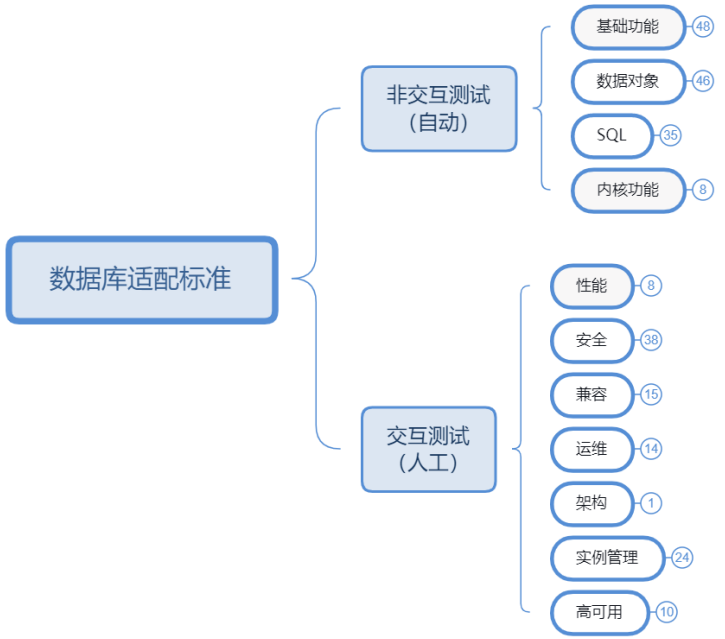
场景评估

收敛公司内对数据库场景需求，制定一个产品选型矩阵。根据要求对待选产品进行评估工作，包括初始调研、技术评估、数据库评测（功能、非功能、业务等）、适配性评估等。此项原则是在方案选型中保持自由度，不绑定厂商，随时可替换。

需求分类	适用场景					
	业务场景	数据规模	事务一致性	应用适配能力	典型负载	分析能力
事务类	联机交易 轻量数据分析	中	单库强一致 多库应用解决	高	高并发小数据量 事务性读写 / 小数据量分析	弱
	联机交易 简单事务 大并发	大	应用层最终一致性	低应用 逻辑有入侵	点查 / 点写 / 有限的关联和分析支持	弱
	联机交易 批量处理 实时分析 混合负载	大	强一致	高 对应用透明	点查 / 点写 / 有限的关联和分析支持	弱
分析类	批量处理 复杂分析 非实时查询	大	弱一致	低，应用逻辑入侵	复杂分析查询	强
事务/分析混合类	联机交易 批量处理 实时分析 混合负载 轻量级复杂分析	大	强一致	高 对应用透明	高并发小数据量 事务性读写 / 复杂分析	强

选型测试

对待选数据库产品进行全方位的评估，包括从功能、非功能、业务三个角度进行测试。需要建立企业自有的测试样板集。



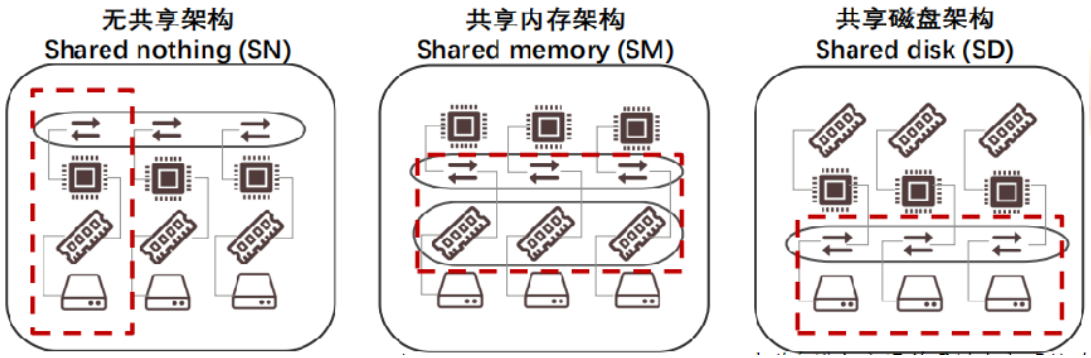
选型评估阶段

技术培训

对于新引入技术栈进行培训，包含对架构、研发、运维人员的培训。对研发人员，阐述架构设计、结构优化、SQL语句等与之前差异。对运维人员，侧重将这种新数据库融入到现有的运维体系中。特别是分布式架构数据库，与传统集中式数据库不同，其对于运维带来的挑战也更大。

方案制定

在明确选型后，需完成相关技术方案的制定。这里需结合业务发展、资源投入、基础环境、高可用等诸多因素进行考虑。此阶段输出成果，将直接影响后续的资源评估、改造及测试等。



选型评估阶段

资源评估

资源评估通常是前置的，因为很多企业是采用预算制，需要提前很长周期做好后续的采购规划。因此将资源评估工作做了前置。上述资源，特指硬件及基础设施部分。

研发评估

包含对数据结构、语句及应用自身的改造评估。在结构上各数据库能力层次不齐，需进行调整甚至重构。建议减少复杂对象使用，只考虑表索引；其余视图、序列、触发器、存储过程、函数、包等都通过外部的等价设计来完成。目的是减少对数据库依赖。此外，分布式下还需考虑分片策略等。对 SQL 语句的评估，建议从复杂度、方言等进行评估。

对象使用情况

对象类别	数量
表	15492
表(大表)	47
表(分区表)	0
字段(大对象)	18
索引(B树)	6303
索引(其他)	23
视图	5

SQL类别	数量
总SQL数	174
超长SQL	64
ANTI SQL	5
Oracle Syntax SQL	52
Join 3+ Table SQL	28
SubQuery SQL	40

研发测试阶段

工作排期

在研发开始前，需完成必要的排期工作。这也是很多信创项目的痛点之一，很难做到合理评估。建议采用“打样”方式，摸索内部研发能力，给出符合企业实际情况的排期。

工作量评估依据

1.梳理分析（M列）

关联系数

* 特殊SQL语句使用数量(E列)：特殊SQL改造量较大，建议按高成本核算。
* 使用特殊数据类型表的数量(F列)：使用特殊数据类型表改造成本较高，建议按高成本核算。
* 特殊对象数量(I列)：特殊对象是指目标库不兼容部分，需要进行对象级别改造适配，按对象数评估。对于存储过程、函数、触发器等按行数处理。
* 系统负载-DML次数(J列)：DML次数间接反映系统负载，影响整体改造成本，按100万、1000万、10000万分段来评估。
* SQL总数(K列)：SQL总数反映整理改造工作量。
* 选型(L列)：如目标库成本高，给出比例系数（对比MySQL）。

计算公式

工作量 = （特殊SQL语句使用数量 * 系数 + 使用特殊数据类型表的数量 * 系数 + 特殊对象数量 * 系数 + SQL总数 * 系数 ）* （1+系统负载系数）* 选型系数

2.代码修改（N列）

关联系数

* 平台相关(M列)：项目组需配合平台测试等带来的工作量。参照[梳理分析]部分得到的工作量，乘以此系数。

计算公式

工作量 = 梳理分析 * 系数

3.单元测试（O列）

关联系数

* 平台相关(M列)：项目组需配合改造测试等带来的工作量。参照[梳理分析]部分得到的工作量，乘以此系数。

计算公式

工作量 = 梳理分析 * 系数

4.迁移工作量（P列）

关联系数

* 数据量(H列)：整体迁移工作量，乘以此系数。
* 选型(L列)：目标库迁移成本高些，乘以此系数。

计算公式

工作量 = 数据量 * 数据量系数 * 选型系数

研发改造

对数据对象进行改造，有些简单映射即可，有些则需重构。有些对象（如复杂对象或重计算类对象），可从数据库端剥离，在上层等价实现。对语句修改，这项工作非常浩大。如用 ORM 工具还好，如有硬 code，改动较繁琐。有工具可辅助完成改造工作，但转换后仍需人工审核，要保证语义等价和执行效率等等。

研发测试阶段

应用改造

配合迁移工作，应用也存在改造的工作量。例如有些在数据库端实现的逻辑，是需要改在应用端实现的；有些应用本身在新数据库架构下就需要进行适配性改造等等。

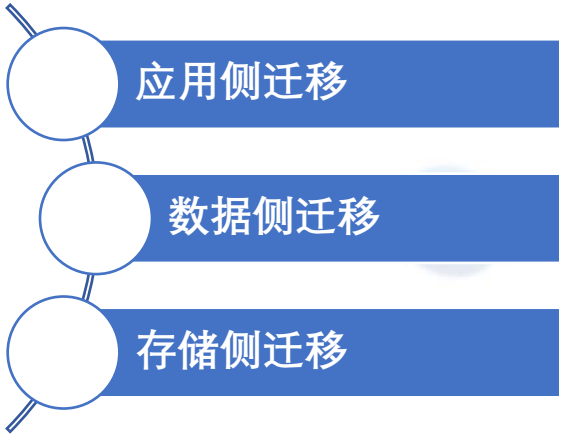
功能测试

数据库改造（对象+SQL）和应用改造均完成后，还需要一个关键步骤—功能测试。按业务功能拆分，针对每个独立功能，从应用侧角度进行测试验证，来保证上述改造的结果正确且性能符合预期。此处的功能测试，与后面的上线交割的功能测试不同，更强调是以小的应用单元为测试目标。这样便于随时修改，随时测试。

验证并行阶段

迁移方案

确定迁移方案，包括数据迁移与应用迁移。针对前者主要取决于源数据库、物理环境、迁移窗口、是否并行、是否回退等诸多因素。从原理上分，可分为应用侧迁移、数据库侧迁移、存储侧迁移三种，各有优劣。个人建议，优先应用侧，原因是与业务结合紧密、同步验证容易、方便并行回退等等。但缺点在于，需要应用侧有修改工作，无法形成统一标准的方案。



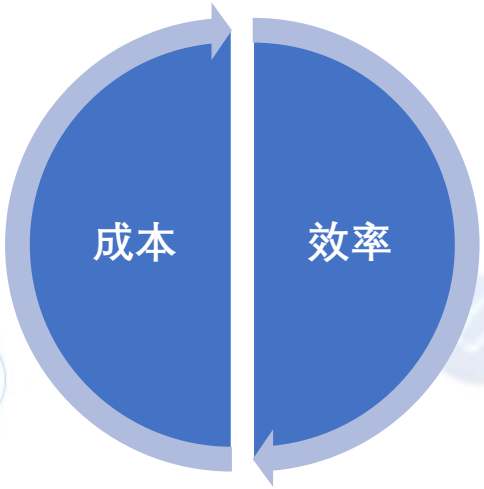
数据迁移

- **结构迁移**。一般可提前完成。结构确定后，即可完成。不与数据同步放在一起，为了便于问题排查分析。
- **全量迁移**。一般是可离线、静态去做，通过备份恢复、导入导出等方式，将静态数据迁移过去。可不在停机窗口进行，并记录好位点。
- **增量迁移**。从全量迁移位点开始，可采取停机或不停机的方式。一般通过追日志方式，追齐数据，并短时静默应用，完成数据最终达到一致的状态。

支持上线阶段

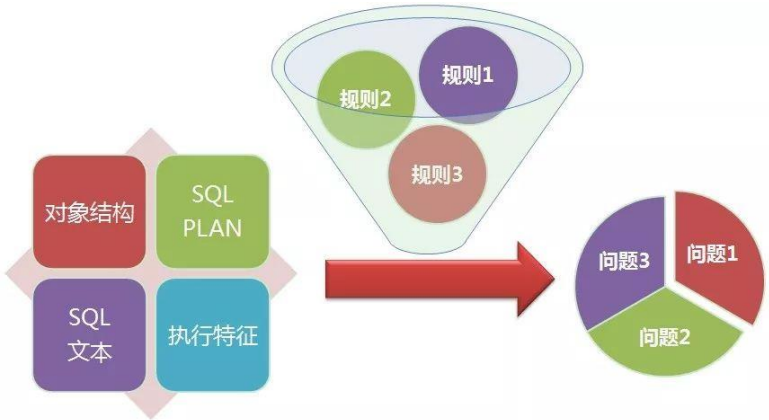
数据校验

在上线前需完成对数据迁移后的质量判断，这就引入数据校验的初衷。是对同步两边的数据是否一致做出判断，为是否正式切换的判断依据之一。难点在于：海量数据对比效率、异构数据比对及实时同步下同步下对比。



语句审核

在上线交割及日常保障中，还需支持 SQL 审核，是为了保证语句运行质量。SQL 审核细分，可分为事前、事中、事后审核，这里更多指事前审核部分。即在开发过程中，针对 SQL 运行情况给予评估判断，来保证上线后的质量可控。一般是通过预定义一组规则，完成对语句的审核。这一过程贯穿在整个开发过程中。



未来展望

展望…

作为数据应用高地，金融行业一直走在技术前沿。这些年数据库领域快速、蓬勃发展，新技术、新架构、新方向层出不穷。相信未来，将在金融领域得到更为广泛的实践。从具体技术来看，下面技术将成为热点…

- **分布式**

面对海量规模及高并发等场景，分布式数据库在基本功能、稳定性、性能等方面已趋于成熟。各家产品开始在易用性、可观察性、诊断能力、生态兼容等方向发力，并陆续开始有成果落地。相信分布式数据库的不断发展，会在未来得到更大范围的使用。

- **云原生**

在云数据库领域，云原生数据库成为焦点，各厂商已从传统数据库托管类产品的竞争，过渡到自有云原生数据库产品的较量，承载规模、弹性能力、极致性价比等成为了发展要点，特别是Serverless方向，已成为后续发展的主流方向之一。

- **HTAP**

HTAP，特别是分布式能力的引入，为AP方向的能力提供了更多算力，也为HTAP带来更多想象空间。从用户视角来看，HTAP简化了原有技术栈，统一访问方式，为用户带来更优质的体验。

- **NoSQL**

以图、时序为代表的产品成为了发展热点，一大批初创企业及产品受到了更多关注。在产品能力上，纷纷从兼容生态、突出核心、提升易用性等角度切入，并在更多场景上得到使用验证，大大扩展了原有的使用空间。

Thanks