ICS 35.240.40
CCS A 11

团　体　标　准

T/BFIA 010—2022

商业银行分布式联机交易系统技术规范

Technical specification of distributed online transaction system for commercial banks

2022－04－07 发布　　2022－04－07 实施

北京金融科技产业联盟　发布

目　　次

前　言

本文件按照GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规则起草。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别专利的责任。

本文件由北京金融科技产业联盟归口。

本文件起草单位：神州数码信息服务股份有限公司、中国工商银行股份有限公司、北京金融科技产业联盟、中国人民银行清算总中心、赞同科技股份有限公司、杭州趣链科技有限公司、恒生电子股份有限公司、广东华兴银行股份有限公司、北京国家金融科技认证中心有限公司、深圳市腾讯计算机系统有限公司、中国光大股份有限公司、成都虚谷伟业科技有限公司、北京东方通科技股份有限公司、广发银行股份有限公司、晋商银行股份有限公司、中电金信软件有限公司、阜新银行股份有限公司、交通银行股份有限公司、北京银行股份有限公司、北京华胜信泰数据技术有限公司、宁波银行股份有限公司、广西北部湾银行股份有限公司、湖南三湘银行股份有限公司、江西裕民银行股份有限公司、宁夏银行股份有限公司、平安银行股份有限公司、中国银联股份有限公司、杭州银行股份有限公司、华为技术有限公司、武汉达梦数据库股份有限公司。

本文件主要起草人：薛春雨、夏龙飞、滕达、沈伟、施媛、聂丽琴、黄本涛、李明艳、张蕾、李璐、张睿、张弦、王嘉琪、刘戈、林智、张璐、杜静漪、刘智慧、熊钊隆、李振、叶强林、苏强、陈明、刘浩然、杜蓉、明玉琢、李彦清、李志鹏、赵磊、李钢、顾志鹏、李欢、赵平、李向军、周勇汐、田立斌、郭志军、马锋、王子健、刘顺华、刘淼、范小东、郑永顺、刘家模、崔汉新、唐文刚、陈佳霖、丁伟、袁晟、耿道武、王立军、王小勇、白玫、苟凯俞、徐建芳、厉华、白阳、董里、李金亮、黄海明。

商业银行分布式联机交易系统技术规范

1 范围

本文件规定了商业银行分布式联机交易系统中分布式技术覆盖的范围及具备的基础功能。

本文件适用于商业银行分布式联机交易系统的建设。

2 规范性引用文件

本文件没有规范性引用文件。

3 术语和定义

下列术语和定义适用于本文件。

3.1

信息 information

关于客体（如事实、事件、事物、过程或思想，包括概念）的知识，在一定的场合中具有特定的意义。

［来源：GB/T 5271.1—2000,01.01.01］

3.2

数据 data

信息的可再解释的形式化表示，以适用于通信、解释或处理。

注：可以通过人工或自动手段处理数据。

［来源：GB/T 5271.1—2000,01.01.02］

3.3

分布式 distributed

〈计算机〉物理上由多个计算机参与执行，但在逻辑上完成的是同一个任务。

注：对使用者来说，就像一台计算机在执行一样。

3.4

微服务 micro service

〈计算机〉单一职责的、轻量化的服务单元。

3.5

数据分片 data sharding

将数据表中的数据，按照一定的分片规则分散存储到多个数据存储节点，以均衡节点间数据容量及访问负载。

［来源：JR/T 0203—2020,3.7］

3.6

分片键 sharding key

参与数据分片规则计算的业务属性。

3.7

事务 transaction

1组以原子性、一致性、持久性、隔离性为特征的相关操作。

［来源：JR/T 0203—2020,3.2］

3.8

联机交易系统 online transaction system

<计算机>立即响应客户请求并返回结果的系统。

注：商业银行一般将跟联机交易系统相关的批处理业务也包括在内。

4 缩略语

下列缩略语适用于本文件。

API：应用程序接口（Application Programming Interface）

SQL：结构化查询语言（Structured Query Language）

5 概述

商业银行采用分布式技术建设联机交易系统时，不仅需要考虑具体的业务功能，还需要提供完整的分布式技术体系，为系统提供全方位的分布式能力，当性能遇到瓶颈时可通过横向扩展的策略予以解决，并且对于行业重点关注的问题需提供可落地的解决方案。另外，分布式应覆盖商业银行联机交易系统的所有维度，并且还需为商业银行的一些特殊需求提供针对性的能力。应用框架基于对分布式技术的封装，并结合具体业务场景提供框架级的支持，以满足上层业务的快速开发。参考架构见图1，其中分布式技术体系应包括：

a) 微服务平台：服务层面分布式的具体体现，将一个复杂系统拆分为多个微服务，不仅可以增加服务的运行实例，还可以快速灵活地应对业务变化；

b) 分布式数据：将大数据量表中的数据进行水平切分，用多个物理单元来承载，不仅降低了单表的数据量，还增加了可用的物理资源；

c) 分布式缓存访问：对于大量的数据查询请求，采用分布式缓存进行存储，降低与数据库的交互次数，缩短服务的响应时间；

d) 分布式消息访问：为分布式系统之间提供了一种异步的通讯方式，主要利用消息中间件对系统进行解耦，以及控制系统的流量等；

e) 分布式调度：提供统一的分布式调度机制，协调多个微服务节点及数据库节点共同参与运行，充分利用物理资源，将批量处理的时间控制在一定范围内；

f) 分布式事务：由于金融业务的特殊性，对事务一致性的要求比其他行业要高，所以，在上述的分布式体系下，还要提供跨数据库、跨微服务的事务一致性保证机制。

图 1　商业银行分布式联机交易系统参考架构

上述六个部分，从不同的维度为商业银行联机交易系统提供了完整的分布式解决方案，但在具体的落地过程中，应根据交易系统的实际需要进行合理选择。具体参考原则如下：

——“微服务平台”是分布式联机交易系统的基础，均需具备；

——如果交易系统的数据量比较大（通常以单表数据量超过千万为标准），需具备“分布式数据”的能力；

——如果要获取更高的系统性能，需引入缓存机制，并具备“分布式缓存访问”的能力；

——如果涉及分布式系统间的解耦或异步通信的场景，需引入“消息中间件”，并具备对其的访问能力；

——如果涉及比较复杂的批处理业务场景，需具备“分布式调度”的能力；

——如果涉及跨微服务或者跨数据库的事务场景，需具备“分布式事务”的能力。

6　微服务平台

6.1　概述

微服务平台为联机交易系统提供服务分布式的运行机制，以及配套的管理体系，包括网关、运行框架和运维监控三部分。

6.2　网关

网关为外部系统访问提供统一地接入及控制，具体要求如下：

a) 协议转换：应负责接收外部的服务请求，并转换为微服务内部的访问。

b) 服务鉴权：应对服务的访问进行控制，只有在权限范围内的才可以访问，否则将被拒绝。具体的权限设定应根据交易系统的实际情况来定，常规情况可只做身份的合法性判断。如果要

控制得更加精准，宜通过对用户权限进行多维度的管理来实现。

c) 流量控制：应对请求的总量进行控制，如果超过限制，后续的请求将被拒绝或者等待。还可按具体的业务维度进行更细粒度的控制，例如按交易渠道进行控制。

d) 熔断降级：如果某个服务的失败率比较高，或者发现一些明确的异常情况，有可能影响到其他服务的正常运行，应自动切断该服务的所有请求或者切换为本地的备用服务，以保护系统整体的可用性。

6.3 运行框架

运行框架为微服务平台提供最内核的运行保障，具体要求如下：

a) 服务注册/发现：服务提供者在启动时应将服务访问地址注册到“注册中心”，服务消费者应通过“注册中心”获取到最新的服务列表，当服务提供者下线时，注册信息应从“注册中心”删除，服务消费者应更新服务提供者列表。

b) 负载均衡：服务消费者调用提供者时，应根据本地的列表并依据一定的算法选择具体的提供者进行调用，整体应保证到多个提供者上的负载相对均衡。

c) 自动隔离/恢复：当服务提供者出现异常情况（例如宕机），应自动从“注册中心”删除，当其恢复后应自动加入，期间消费者应同步获取最新的服务提供者列表。

d) 集群容错：当服务调用发生异常时，应提供多种容错机制，例如重试（可以设定重试次数）、直接抛错等机制。

6.4 运维监控

运维监控为微服务平台提供运维管理能力，具体要求如下：

a) 灰度发布：服务有新版本发布时，可通过灰度发布只对部分用户开放，运行一段时间如果达到预期，再对所有用户开放。

b) 统一配置：应支持通过统一的配置中心调整微服务实例的相关参数，并支持配置信息的版本管理，以便配置出现问题后进行统一回退操作。

c) 服务治理对服务运行态的情况进行动态调控，具体要求如下：

1）负载策略调整：应支持对多个服务提供者的负载均衡策略进行调整，防止某个提供者的负载过高；

2）流控策略调整：应支持根据实际需要对流量控制的策略进行动态调整；

3）熔断降级控制：应支持对服务的熔断及降级的策略进行调整；

4）路由控制：应支持通过配置对服务的路由策略进行调整，例如暂时把某一个服务提供者屏蔽。

d) 调用链跟踪：应支持查看服务调用过程的完整链路信息，通过主动发现和追踪业务系统的调用关系，快速定位系统瓶颈。

e) 分布式日志：应把分布式系统的相关日志信息进行采集、汇总，并把跨多个系统的日志串接起来，提供统一的页面按不同维度进行日志信息查询。

f) 监控告警：应支持对监控对象制定告警策略，当达到告警条件时产生告警信息，并通过多种

方式（例如邮件、短信）及时通知运维人员进行处理。

7 分布式数据

7.1 概述

分布式数据的目的是将大数据量的表通过水平拆分，使数据存储到多个不同的物理节点，查询时可以到具体的节点快速访问。行业内主要包括“技术架构层实现”和“在分布式数据库实现”两种模式，商业银行可根据自身实际情况进行选择。本文件仅规范“技术架构层实现”模式，包括数据路由、SQL兼容性、数据库关联性及运维监控四部分。

7.2 数据路由

数据路由按照分片规则将SQL发送到具体的分片执行，具体要求如下：

a) 应提供成熟的分片规则，包括取模、日期、范围、枚举、哈希等。
b) 应支持自定义分片规则。
c) 应同时支持分库和分表。
d) 对某些特殊 SQL 应支持指定分片路由。
e) 对一些使用率较高的 SQL，但查询条件不包含分片键的情况（例如客户信息表按客户号分片，用手机号查询），应支持间接通过分片键路由。
f) 应支持同时按多个维度进行分片及路由。

7.3 SQL 兼容性

应支持商业银行常用的SQL语句，具体要求如下：

a) 应支持查询条件包含分片键的新增、修改（被修改的字段不含分片键）、删除及查询操作。
b) 应支持跨库的聚合函数、关联查询、排序、分页、分组、联合查询等常用组合。
c) 应支持字符串、数学、日期、格式化、聚合等常用函数。

7.4 数据库关联性

应支持常用的关系型数据库。

7.5 运维监控

运维监控为分布式数据提供配套的运维管理能力，具体要求如下：

a) 应支持可视化地对数据分片规则进行调整。
b) 应支持对后端数据库进行管理。
c) 应为数据管理员提供数据的操作及管理能力。
d) 应提供在线扩容机制。

8 分布式缓存访问

分布式缓存可以为分布式系统提供应用级的缓存能力，大幅提升系统的响应速度，但在使用过程中应尽可能降低对业务的侵入，同时兼顾金融场景对数据准确性的要求。具体要求如下：

a) 基础的分布式缓存访问的 API：应提供缓存分布式访问的基础 API，为业务框架及特殊情况使用提供支持。

b) 应提供与数据库配合使用的能力，具体要求如下：

1) 与数据库配合的查询，应支持优先从缓存查询；

2) 与数据库配合的写操作，应保证缓存与数据库的数据一致性；

3) 应降低对业务的侵入，如果是基于 Java 语言实现宜采用注解方式。

c) 防止脏读：应提供完善的机制，控制缓存跟数据库配合使用时，脏数据对交易系统造成的影响，主要涉及与数据库配合的读写交叉的情况、并行写的情况，以及缓存服务端物理故障的情况。

9 分布式消息访问

消息中间件为分布式系统提供异步通讯机制，从而应对各种系统间的解耦、削峰填谷等业务场景，从分布式联机交易系统视角来看，具体要求如下：

a) 应提供常用消息中间件的访问能力。

b) 应提供框架级的消息可靠发送机制。

c) 消息的消费者处理逻辑应具备幂等特性。

10 分布式调度

分布式调度主要用于分布式系统各种任务的协同工作，最大限度发挥分布式系统的特性，以更高效的方式完成待处理任务。具体要求如下：

a) 应支持可视化地定义任务的执行顺序及依赖关系。

b) 应将任务分配到多个节点执行，如果某节点在执行过程中宕机应自动切换到其他节点执行。

c) 如果执行过程出现业务异常时，应支持查看详细错误信息，并支持断点续跑。

d) 应提供在任何时间至少有一个调度节点可用的保证机制。

e) 对涉及大数据量表的任务，应支持对数据进行分段，并在多个节点同时执行。

f) 应至少支持手动、定时和事件驱动三种执行方式。

g) 应支持可视化地查看任务的执行情况，并支持相关运维操作。

11 分布式事务

11.1 概述

分布式事务主要包括跨微服务和跨数据库两种类型，并且不同的业务场景对事务的关注点也有一定差异，所以应综合考虑并选择合适的处理机制，以适配不同的业务场景。应考虑的维度包括：

——隔离性：一个事务内部操作的资源对并发的其他事务的隔离情况。

——一致性保证度：一个事务所有相关操作的一致性保证情况。

——风险：处理机制自身是否存在一定的风险，例如脑裂、雪崩等。

——性能：处理机制对系统性能的影响。

——复杂度：处理机制对系统开发复杂度的影响。

11.2 处理机制

分布式事务的处理机制应至少包括如下一种：

a) 冲正模式：该模式在框架级提供统一的机制，避免每个开发人员重复实现，规避技术风险。其主要应用于一个事务跨多个服务的场景，要求每个服务都提供对应的冲正服务，当后续的服务执行失败时，框架将自动执行前面服务的冲正服务，将之前已经执行成功的服务回退到执行前的状态。该模式是分布式系统中最基本的一种模式，隔离性、一致性保证度较弱，对性能的影响相对较小。

b) 基于业务逻辑的两阶段模式：该模式由框架提供基于两阶段的运行机制，具体每个阶段要执行的业务逻辑由开发人员按照规范实现，在第一阶段应进行业务的准备工作及资源的预留，在第二阶段应进行真正的提交/取消操作。该模式主要提供分布式事务的控制机制，具体的逻辑由业务实现，所以开发的复杂度及对开发人员的能力要求较高，隔离性、一致性保证度依赖业务的具体实现，在性能上有一定优势。

c) 基于 SQL 的两阶段模式：该模式也是基于两阶段的实现机制，但不需要开发人员参与，而是通过框架对 SQL 进行拦截处理来实现两阶段的一种方式。该模式隔离性及一致性保证度高，对业务系统无侵入，性能损耗相比前两种稍大。

11.3 运维监控

分布式事务出现问题对系统的影响较大，系统应提供对应的运维管理能力，具体要求如下：

a) 应支持对分布式事务的整体运行情况进行管理，例如查看分布式事务的总数、成功及失败的笔数等。

b) 应对异常事务提供对应的告警机制。

c) 应支持查看异常事务的详细信息，以便于人工干预。

12 异常处理

对如下异常情况应提供应对机制，具体要求如下：

a) 系统运行资源不足：在系统运行过程中，当发现系统运行资源不足（例如 CPU 长时间大于 80%）时，宜自动增加运行实例，以提升系统整体的处理能力。

b) 数据量超出系统承载能力：当系统运行一段时间后，由于数据量的快速增长，已有的分库数已经不能满足业务需要时，应对数据库进行扩容，在扩容过程中应尽可能地减少对业务的影响。

c) 物理故障造成的事务异常：由于分布式事务涉及多个独立的事务资源，所以不管是哪种模式，

在部分事务资源出现物理故障的情况下，都面临不一致的风险，针对于这种情况，应提供如下的应对措施：

1) 自动恢复：当物理故障解除后，系统应自动检测，并对受影响的事务提供自动处理机制，尽最大可能保证事务的一致性，避免相关事务由人工处理；
2) 人工干预：当出现极端异常情况，自动恢复机制也无法恢复的情况下，应支持人工干预，并提供必要的事务信息（例如全局事务状态、出现问题的分支信息、事务的上下文等）以供分析。

参 考 文 献

[1] GB/T 5271.1—2000 信息技术 词汇 第1部分：基本术语
[2] GB/T 10113—2003 分类与编码通用术语
[3] GB/T 29246—2017 信息技术 安全技术 信息安全管理体系 概述和词汇
[4] JR/T 0203—2020 分布式数据库技术金融应用规范 技术架构