

资料

更新记录

日期	作者	版本	变更内容	类型	批准人	批准日期
2019.12.9	孟津平 张雪	draft	创建文档	N		

[类型：修改Modified，新建New，增加Add，删除Deleted]

一、前言

第三代客运数据库业务模块结合OLTP和OLAP的特性，既需要事务的并发处理，还需要大数据量的分析汇总，公司决定采用安全性更强的Oracle 19c。系统模块则是采用PostgreSQL 12实现。

二、实例拆分

在考虑第三代客运系统业务相关度和上线后产生的数据量，具体的实例及拆分情况如下：

- ① 销售组
 - ② 交付组
 - ③ 联运开账组
 - ④ 税务管理组
 - ⑤ 分摊组
 - ⑥ 销售审核组
 - ⑦ 空白票与通知单组
 - ⑧ 主文件
 - ⑨ 系统管理
 - ⑩ 全系统账务
- 应收管理

以上11个系统中1、2、3、10、11是每个系统单独一个实例，共五个实例。其余4-7一个实例，8-9一个实例，全系统共七个实例组成。

三、分库及分表

3.1. 分库方式及原则

分库原则：如果没有性能问题尽量不进行分库，因为分库需要处理跨库 join、跨库分页排序、额外的逻辑运算、分布式事务一致等情况。

分库方式：

- 1. 垂直(纵向)拆分/拆字段

大字段的垂直切分。

按照用途垂直切分。

按照访问频率垂直切分。

1. 水平(横向)拆分/拆记录行：数据量一直增长的业务数据，需要水平切分，才能保证数据库的磁盘IO, CPU等资源的稳定可用。

按照时间水平切分。

按照用户ID范围切分。

分类拆分。

3.2. 分表及分区

1. 分表是把一张大表分成多个小表适用如下场景：

2. 一张表的查询速度已经慢到影响使用的时候。

3. 当频繁插入或者联合查询时，速度变慢。

4. 理论上数据库中单表大小超过3G或数据量超过1000万就需执行分表操作，比如每年创建一个表，数据存储到对应的年表中，必定会减少很多数据量。（如果分成成年表数据量还是过大，可以细分到月表，天表...）。

5. 分区是把一张表的数据分成N多个区块，仍然是同一张表跨存储不跨DB，适用如下景：

6. 一张表的查询速度已经慢到影响使用的时候。

7. 对数据的操作往往只涉及一部分数据，而不是所有的数据

3.3. 分库分表的负担

1. 增删改操作：

sql改写：改写得到每个库执行的sql

sql路由：改写的sql需要在对应库执行

结果集合并：多个库的结果集需要最好合并处理

2. 全局分布式id：

取代分库分表之前的sequence，防止数据冲突

3. 分布式事务：

分库分表需要保证不同数据库上的事务一致性

3.4. 分库分表技术选择

选用中间件原因：

1. 通过定义分片规则，由中间件进行SQL拆分，SQL重写，SQL路由和结果集合并，对应用透明。

2. 有对应的分布式事务处理架构，处理分布式事务一致性问题。

分库分表中间件：smart-client和proxy两类，

Smart-client: tddl, zebbra, zdal, sharding-jdbc

Proxy: cobar, mycat, atlas, drds, sharding-sphere

Proxy特点是多语言支持，但实现复杂，需要在应用层和数据库层提供中间层。

Smart-client特点是实现简单，在应用层实现，但仅支持java语句，版本需要跟从应用升级（依赖jar包）。

3.5. 分库分表调查研究：

1. 数据量

销售组：1年6千万，10个大表，最大200字段；数据量不大，无分库分表必要。

全系统账务：12个大表，一个月2亿数据量，其中一个大表一个月6千万，一个5千万，一个3千万，业务复杂度不高，数据量大，每年新增1.2TB，数据量大，有分库分表必要，分库分表的识别字段应确定是账务月。

交付组：运输客票、运输客票实时数据，数据量每月大概7百万条数据；大表PNR数据，每一张票的数据变化过程都要记录，每月大概7千万条数据。

2. 分库分表问题：

业务只限于业务月（每个系统不一样），可以用业务月作为分库分表的识别字段。但是其他系统需要查询此数据库信息时，除了票号，还需要对应的业务月作为查询条件，因此其他系统的表可能需要冗余此系统的业务月字段；或者定义能够识别出业务月字段的唯一code。

四、读写分离技术及对比

- 1. Data Guard：提供管理，监视和自动化软件，用于创建和维护生产数据库的一个或多个同步副本，以保护Oracle数据免受故障，灾难，人为错误和数据损坏，同时为关键任务应用程序提供高可用性。 Data Guard包含在Oracle数据库企业版中。
- 2. Active Data Guard：是Oracle数据库企业版的选件许可证。Active Data Guard支持扩展基本Data Guard功能的高级功能。这些包括：

实时查询 - 负责将主库最新的数据刷新到物理备库

自动块修复 - 自动修复对坏的数据块进行修复

远程同步 - 任何情况下的主备同步，实现零数据丢失保护。

备库开启块跟踪 - 在备库开启块跟踪，实现备用数据库上启用增量备份。

Active Data Guard滚动升级 - 简化计划停机时间

全局数据库服务 - 跨复制数据库的负载平衡和服务管理。

应用程序连续性 - 使中断对用户透明。

- 1. Oracle GoldenGate（OGG）：是一个实现异构IT环境间数据实时数据集成和复制的综合软件包。该产品集支持高可用性解决方案，实时数据集成，事务更改数据捕获，运营和分析企业系统之间的数据复制，转换和验证。Oracle GoldenGate 12 c通过简化配置和管理，加强与Oracle数据库的集成，支持云环境，扩展异构性以及增强安全性，实现了高性能。除了用于实时数据移动的Oracle GoldenGate核心平台，Oracle还提供了适用于Oracle GoldenGate的管理包（用于Oracle GoldenGate部署的可视化的管理和监视解决方案）和Oracle GoldenGate Veridata（允许在两个正在使用的数据库之间进行高速，海量的比较）。

表格对比总结：

	Oracle data guard	Oracle goldengate
原理	复制归档日志或在线日志	抽取在线日志中数据变化，转换为GGS自定义格式存储在本地队列或者远程队列
稳定性	作为灾备稳定性高	稳定性不如ADG
维护	维护简单，极少出问题	命令行方式，维护较复杂
对象支持	完全支持	部分对象需手工创建维护
备份端可用性	备份端处于恢复或只读状态，只读状态下不可恢复	备份端可以做读库，可以做写库，实现双中心
接管时间	DG:数据库工作在mount下，接管业务要open ADG:可实现立即接管	可实现立即接管
复制方式	DG:备库mount下应用redo log，无法同步复制 ADG:在备库open应用redo log，近似同步复制	提供秒级别大数据量捕捉和恢复，异步复制
资源占用	复制通过LGWR或者ARCn实现，占用数据资源少	业务高峰占用资源较多，可采用downstream模式，拆分出单独抽取服务器
异构数据库支持	仅支持oracle，且操作系统相同，版本号可不同	两端Oracle版本可以不同，操作系统也可不同
带宽占用	Oracle net传输日志，可通过高级压缩压缩2-3倍	TCP/IP传输数据，集成数据压缩可达到9:1
拓扑结构	可实现一对多	可实现一对一，一对多，多对一，双向复制等多种拓扑结构

ADG主要解决了DG时代读写不能并行的问题，ADG还是能做到近似同步复制，而OGG的数据复制在亚秒级，还是只能算作异步复制。

应用场景分析

DG可以用在容灾测试上，在金融、电力、能源行业，生产上常见的容灾架构为ADG，尤其是异地灾备。也有部分较高要求的采用 RAC + ADG，这里的RAC有的是基于存储集群虚拟出来的分布式卷之上做的RAC，有的是通过ASM冗余设计本身实现的。OGG在重大变更需要异构数据库同步数据的场合下或者是数据集中平台上采用。

ADG，最常用的同城，异地灾备解决方案，物理级备份，备机不可写，传输数据为所有redo日志的更改，数据量稍大，不过从以往的使用经验来看，也不太会影响网络，除非应用对网络有很苛刻的要求，即使有，也可以通过vlan或者路由或者多网卡的方法特别建立网络通道，主备库完全一致，缺点是必须全库备份。OGG，DSG这两个是一个类型的，逻辑备份，主要采用特有的技术从联机日志中抽取更改项应用到备库，主备库为两个库，可以全库同步也可以同步单张表或数张表，同步速度较快，传输数据量很少，DML操作和DDL操作均支持。

ADG 同构平台数据同步，OGG可以异构平台数据同步。

ADG 可以通过快照方式保留当前时刻点数据，OGG不能做到。

ADG 直接通过日志重做实现数据复制，OGG是通过对日志加工之后的模式进行数据分析实现复制。

分schema，恢复数据不受影响。

oracle虚拟机，备份回复，sla的要求。

1、数据库选型，版本

oracle 19C 业务模块

postgreSQL 12 系统管理，主文件

2、实例拆分

7实例

3、分库分表

表记录数：

销售： 1年6kw，10个表，最大200字段，

税： 票号分表，

账务：1年10亿，30个大表，最大100字段，按月份，feedback code ， 账务月

表大小：