



(12)发明专利申请

(10)申请公布号 CN 111427898 A
(43)申请公布日 2020.07.17

(21)申请号 202010182675.8

(22)申请日 2020.03.16

(71)申请人 上海爱数信息技术股份有限公司
地址 201112 上海市闵行区联航路1188号8
幢第2层A-1单元

(72)发明人 黄开德 苟露

(74)专利代理机构 上海科盛知识产权代理有限公司 31225
代理人 叶敏华

(51)Int.Cl.

- G06F 16/23(2019.01)
- G06F 16/242(2019.01)
- G06F 16/2455(2019.01)
- G06F 16/27(2019.01)
- G06F 11/14(2006.01)

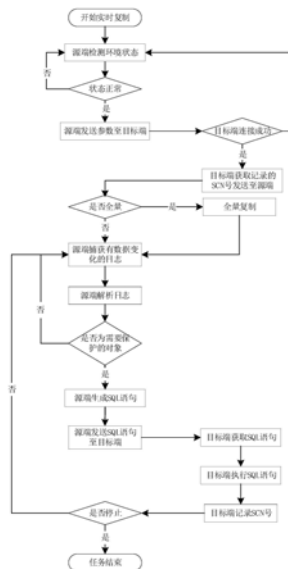
权利要求书3页 说明书8页 附图2页

(54)发明名称

一种基于解析Oracle日志的持续数据保护系统及方法

(57)摘要

本发明涉及一种基于解析Oracle日志的持续数据保护系统及方法,用于进行Oracle数据的保护,同时满足客户低RPO及数据库一致的需求场景,系统包括服务端Server,作为实时复制软件的控制台;客户端Agent,作为源端和目标端在实时复制软件的代理客户端;源端Source,作为用户的生产环境端;目标端Target,作为用户的灾备环境端;Database,作为源端Source和目标端Target数据库的实际数据存储;Redo日志文件或归档日志,用于负责存放所有对数据库更改的操作记录及更改的数据。与现有技术相比,本发明具有使RPO和RTO趋近于0,保证Oracle数据库数据的实时一致性与可用性等优点。



1. 一种基于解析Oracle日志的持续数据保护系统,用于进行Oracle数据的保护,同时满足客户低RPO及数据库一致的需求场景,其特征在于,该系统包括:

服务端Server,作为实时复制软件的控制台,用于担负与用户交互的任务,同时统筹管理包括客户端Agent在内的所有的资源;

客户端Agent,作为源端和目标端在实时复制软件的代理客户端,用于负责与Server进行交互,接收Server下达的实时复制任务的命令,并针对命令做出相应的处理,并将执行结果返回给Server;

源端Source,作为用户的生产环境端,用于在实时复制任务运行时,实时读取其Database的Redo日志文件或归档日志,通过处理获取有数据变化的日志后,解析日志并生成SQL语句;

目标端Target,作为用户的灾备环境端,用于接收来自源端Source的SQL语句并生成SQL文件,通过调用OCI在目标端Target的Database中执行SQL语句;

数据库的实际数据存储Database,作为源端Source和目标端Target数据库的实际数据存储,用于在第一次发起任务时进行全量复制,并通过数据泵直接将源端Source的Database中的数据同步到目标端Target的Database中;

Redo日志文件或归档日志,用于负责存放所有对数据库更改的操作记录及更改的数据。

2. 根据权利要求1所述的一种基于解析Oracle日志的持续数据保护系统,其特征在于,所述的源端Source所获取的有数据变化的日志为以事务为基本单位的日志。

3. 根据权利要求1所述的一种基于解析Oracle日志的持续数据保护系统,其特征在于,所述的源端Source产生的SQL语句通过TCP/IP网络协议传送到目标端Target,目标端Target接收到SQL语句后在目标端Target缓存目录中生成SQL文件。

4. 根据权利要求1所述的一种基于解析Oracle日志的持续数据保护系统,其特征在于,所述的服务端Server,负责发送实时复制的任务的指令给对应的客户端Agent,同时负责接收Agent返回的执行结果,监控任务的执行情况。

5. 根据权利要求1所述的一种基于解析Oracle日志的持续数据保护系统,其特征在于,Redo日志文件的个数和大小由数据库指定,采用循环复写的方式写入数据,当数据库开启归档模式时,在一个Redo日志文件写满时则进行归档,产生一个归档日志文件。

6. 根据权利要求3所述的一种基于解析Oracle日志的持续数据保护系统,其特征在于,当发起任务时,源端Source捕获日志,解析日志,生成SQL语句为多线程并发进行,目标端Target接收SQL语句及执行SQL语句为并发执行。

7. 一种实现如权利要求1~6任一项所述的基于解析Oracle日志的持续数据保护系统的持续数据保护方法,其特征在于,当保护的生产环境需要达成RPO趋近于零的效果时,部署实时复制任务,设置一个生产环境端作为源端Source,设置一个灾备环境端作为目标端Target。

8. 根据权利要求7所述的持续数据保护方法,其特征在于,所述实时复制任务按照如下步骤执行备份操作:

- 1) 选择需要保护的数据源及对象,建立实时复制任务,并发起任务,执行下一步;
- 2) 任务启动后,源端Source根据创建任务时配置的任务选项检测是否满足运行条件,

若满足,则执行下一步,否则,重复执行本步骤;

3) 源端Source将目标端Target所需的参数发送至目标端Target,并连接目标端Target,若目标端Target连接成功,且环境符合运行要求,则执行步骤4),否则重新开始执行步骤2);

4) 目标端Target获取上一次任务退出前记录的序列号及SCN号,并将获取的信息发送至源端Source,若在目标端Target缓存目录中找不到记录序列号和SCN号的配置文件,则目标端Target向源端Source返回结果0,源端Source根据获取的序列号判断是否进行全量,若源端Source接收到目标端Target返回的数据不为0,且能够利用序列号查询到日志,则任务不需要进行全量复制,执行步骤6),否则执行步骤5);

5) 源端Source通过全量复制将数据同步至目标端Target,在目标端Target创建目标端Target到源端Source的数据库链接后,目标端Target接收源端Source数据库中的数据并装载至目标端Target数据库,迁移完成后,执行步骤6);

6) 源端Source获取数据库的Redo日志文件或归档日志,处理数据获取日志的部分结构,并判断该日志是否有数据变化,若有,则执行步骤7),否则继续获取日志;

7) 源端Source获取以事务为基本单位的有数据变化的日志后进行日志解析,获取变化数据的信息,判断其是否是需要保护的對象,若是,则执行步骤8),否则继续执行步骤6),获取新的日志进行解析;

8) 源端Source获取到变化的数据,生成对应的DDL语句和DML语句,以事务为单位将SQL语句写入缓存中,并记录事务提交时的SCN号,执行步骤9);

9) 源端Source从缓存数据中获取SQL语句及SCN号,将SQL语句和SCN号通过网络发送至目标端Target,执行步骤10);

10) 目标端Target通过网络接收到源端Source发送的SQL语句以及SCN号,并将SQL语句写入缓存目录中,执行步骤11);

11) 目标端Target通过OCI接口执行SQL语句,将数据同步到目标端Target的数据库中,执行步骤12);

12) 目标端Target在执行完SQL语句后,将SCN号写入配置文件中,当任务异常停止或手动停止后,下一次重新发起任务时,源端Source若还存在该事务所在的日志时,则直接从该事务的下一个事务处进行解析,无需进行全量复制,当完成将SCN号写入配置文件的操作后,源端Source检测是否停止任务的指令是否发出,若未接收到停止指令,则执行步骤6),若检测出停止指令,执行步骤13);

13) 源端Source检测到结束任务的指令后发送至目标端Target,目标端Target接收消息后结束任务,并向源端Source发出目标端Target任务结束的消息,源端Source结束任务。

9. 根据权利要求8所述的持续数据保护方法,其特征在于,步骤2)中,源端Source根据创建任务时配置的任务选项检测是否满足运行条件具体为:

检测需要保护的数据库状态是否为open,若否,则重试;否则,进行下一步;

检测需要保护的数据库的模式是否为归档模式,若不是,则重试;否则,进行下一步;

检测需要保护的数据库中的日志模式是否为强制记录模式,并且是否开启日志补全,若不是,则重试;否则,进行下一步;

检测源端Source缓存目录是否有读写文件权限,若没有,则重试;否则,进行下一步;

获取源端Source当前序列号及SCN号,若获取失败,则重试;否则,进行下一步。

10. 根据权利要求8所述的持续数据保护方法,其特征在于,步骤3)中,需要检测环境是否符合运行要求具体为:

源端Source向目标端Target发送参数,并与目标端Target连接,若源端Source与目标端Target连接失败,则返回步骤2);否则,进行下一步;

目标端Target检查选择的数据库状态是否为open,若状态为非open状态,则返回步骤2),否则,进行下一步;

目标端Target检测目标端Target缓存目录是否有读写文件权限,若没有,则返回步骤2),否则,进行下一步。

一种基于解析Oracle日志的持续数据保护系统及方法

技术领域

[0001] 本发明涉及一种数据保护方法,尤其是涉及一种基于解析Oracle日志的持续数据保护系统及方法。

背景技术

[0002] 最早的Oracle数据库备份为脱机备份,又被称为冷备份。在备份开始前,数据库必须彻底关闭,并且需要将数据库使用的每个文件都备份下来。Oracle数据库第6版推出了热备份的概念。这意味着数据库可以在执行备份的同时保持打开状态并处理事务,但必须将每个表空间放在特殊的备份模式中,然后才能复制OS文件。因此,DBA修改其脚本以连接到数据库,获取表空间列表,将每个表空间放在备份模式中,执行OS文件副本备份,最后从备份模式中取出表空间。

[0003] Oracle数据库的第8版推出了Oracle Recovery Manager,即RMAN,作为Oracle数据库的推荐备份方式。RMAN是随Oracle服务器一同安装的,可以用来备份和恢复数据库文件、归档日志和控制文件的工具,用来执行完全或不完整的数据库恢复。RMAN采用备份级别实现增量备份,在一个完整的备份基础上采用增量备份减少备份的数量。

[0004] 目前,常见的Oracle数据库的数据保护模式为定时备份,即周期性地将Oracle数据库中的数据备份到存储介质中,大部分采用RMAN备份。对于定时备份,需要周期性的全量备份,增量备份,归档日志备份,形成对应的时间点用于数据恢复,可以将数据恢复到历史的某个状态。该备份方式仅适用于对RPO(Recovery Point Object,复原点目标)和RTO(Recovery Time Objective,复原时间目标)要求不高的数据保护场景,对于其他场景则会导致Oracle定时备份的RPO和RTO偏大问题,进而无法保证Oracle数据库数据的实时一致性与可用性。此外,部分对Oracle数据库的数据保护模式采用磁盘卷复制的方式,然而这种方式是对整个数据卷保护,只复制落盘数据,无法处理缓存数据,会出现数据库一致性差等问题。

发明内容

[0005] 本发明的目的就是为了解决上述现有技术存在的缺陷而提供一种使RPO和RTO趋近于零,保证Oracle数据库数据的实时一致性与可用性的基于解析Oracle日志的持续数据保护系统及方法。

[0006] 本发明的目的可以通过以下技术方案来实现:

[0007] 一种基于解析Oracle日志的持续数据保护系统,该方法用于实现Oracle数据的保护,同时满足客户低RPO及数据库一致的需求场景,具体包括:

[0008] 服务端Server,作为实时复制软件的控制台,用于担负与用户交互的任务,同时统筹管理包括客户端Agent在内的所有的资源。所述的服务端Server主要负责发送实时复制的任务的指令给对应的客户端Agent,同时负责接收Agent返回的执行结果,监控任务的执行情况。

[0009] 客户端Agent,作为源端和目标端在实时复制软件的代理客户端,用于负责与Server进行交互,接收Server下达的实时复制任务的命令,并针对命令做出相应的处理,并将执行结果返回给Server。

[0010] 源端Source,作为用户的生产环境端,用于在实时复制任务运行时,实时读取其Database的Redo日志文件或归档日志,通过处理获取有数据变化的日志后,解析日志并生成SQL语句;所述的源端Source所获取的有数据变化的日志为以事务为基本单位的日志。源端Source产生的SQL语句通过TCP/IP网络协议传送到目标端Target,目标端Target接收到SQL语句后在目标端Target缓存目录中生成SQL文件。

[0011] 目标端Target,作为用户的灾备环境端,用于接收来自源端Source的SQL语句并生成SQL文件,通过调用OCI在目标端Target的Database中执行SQL语句。

[0012] 数据库的实际数据存储Database,作为源端Source和目标端Target数据库的实际数据存储,用于在第一次发起任务时进行全量复制,并通过数据泵直接将源端Source的Database中的数据同步到目标端Target的Database中。

[0013] Redo日志文件或归档日志,用于负责存放所有对数据库更改的操作记录及更改的数据。Redo日志文件的个数和大小由数据库指定,采用循环复写的方式写入数据,当数据库开启归档模式时,在一个Redo日志文件写满时则进行归档,产生一个归档日志文件。

[0014] 当发起任务时,源端Source捕获日志,解析日志,生成SQL语句为多线程并发进行,目标端Target接收SQL语句及执行SQL语句为并发执行。

[0015] 一种持续数据保护方法,该方法的主要内容为:

[0016] 当保护的生产环境需要达成RPO趋近于0的效果时,部署实时复制任务,设置一个生产环境端作为源端Source,设置一个灾备环境端作为目标端Target。

[0017] 所述实时复制任务按照如下步骤执行备份操作:

[0018] S1、选择需要保护的数据源及对象,建立实时复制任务,并发起任务,执行下一步;

[0019] S2、任务启动后,源端Source根据创建任务时配置的任务选项检测是否满足运行条件,若满足,则执行下一步,否则,重复执行本步骤;

[0020] S3、源端Source将目标端Target所需的参数发送至目标端Target,并连接目标端Target,若目标端Target连接成功,且环境符合运行要求,则执行步骤S4,否则重新开始执行步骤S2;

[0021] S4、目标端Target获取上一次任务退出前记录的序列号及SCN号,并将获取的信息发送至源端Source,若在目标端Target缓存目录中找不到记录序列号和SCN号的配置文件,则目标端Target向源端Source返回结果0,源端Source根据获取的序列号判断是否进行全量,若源端Source接收到目标端Target返回的数据不为0,且能够利用序列号查询到日志,则任务不需要进行全量复制,执行步骤S6,否则执行步骤S5;

[0022] S5、源端Source通过全量复制将数据同步至目标端Target,在目标端Target创建目标端Target到源端Source的数据库链接后,目标端Target接收源端Source数据库中的数据并装载至目标端Target数据库,迁移完成后,执行步骤S6;

[0023] S6、源端Source获取数据库的Redo日志文件或归档日志,处理数据获取日志的部分结构,并判断该日志是否有数据变化,若有,则执行步骤S7,否则继续获取日志;

[0024] S7、源端Source获取以事务为基本单位的有数据变化的日志后进行日志解析,获

取变化数据的信息,判断其是否需要保护的對象,若是,则执行步骤S8,否则继续执行步骤S6,获取新的日志进行解析;

[0025] S8、源端Source获取到变化的数据,生成对应的DDL语句和DML语句,以事务为单位将SQL语句写入缓存中,并记录事务提交时的SCN号,执行步骤S9;

[0026] S9、源端Source从缓存数据中获取SQL语句及SCN号,将SQL语句和SCN号通过网络发送至目标端Target,执行步骤S10;

[0027] S10、目标端Target通过网络接收到源端Source发送的SQL语句以及SCN号,并将SQL语句写入缓存目录中,执行步骤S11;

[0028] S11、目标端Target通过OCI接口执行SQL语句,将数据同步到目标端Target的数据库中,执行步骤S12;

[0029] S12、目标端Target在执行完SQL语句后,将SCN号写入配置文件中,当任务异常停止或手动停止后,下一次重新发起任务时,源端Source若还存在该事务所在的日志时,则直接从该事务的下一个事务处进行解析,无需进行全量复制,当完成将SCN号写入配置文件的操作后,源端Source检测是否停止任务的指令是否发出,若未接收到停止指令,则执行步骤S6,若检测出停止指令,执行步骤S13;

[0030] S13、源端Source检测到结束任务的指令后发送至目标端Target,目标端Target接收消息后结束任务,并向源端Source发出目标端Target任务结束的消息,源端Source结束任务。

[0031] 进一步地,步骤S2中,源端Source根据创建任务时配置的任务选项检测是否满足运行条件具体为:

[0032] 检测需要保护的数据库状态是否为open,若否,则重试;否则,进行下一步;

[0033] 检测需要保护的数据库的模式是否为归档模式,若不是,则重试;否则,进行下一步;

[0034] 检测需要保护的数据库中的日志模式是否为强制记录模式,并且是否开启日志补全,若不是,则重试;否则,进行下一步;

[0035] 检测源端Source缓存目录是否有读写文件权限,若没有,则重试;否则,进行下一步;

[0036] 获取源端Source当前序列号及SCN号,若获取失败,则重试;否则,进行下一步。

[0037] 进一步地,步骤S3中,需要检测环境是否符合运行要求具体为:

[0038] 源端Source向目标端Target发送参数,并与目标端Target连接,若源端Source与目标端Target连接失败,则返回步骤S2;否则,进行下一步;

[0039] 目标端Target检查选择的数据库状态是否为open,若状态为非open状态,则返回步骤S2,否则,进行下一步;

[0040] 目标端Target检测目标端Target缓存目录是否有读写文件权限,若没有,则返回步骤S2,否则,进行下一步。

[0041] 进一步地,日志的解析过程主要包括如下步骤:

[0042] 以事务为基本单位获取有数据变化的日志,获取时使用XID参数区分。事务中需存在OP5.2,OP5.1,OP5.4,OP11.*或OP24.1。以update操作为例分析解析过程:

[0043] a. 在change参数OP=5.1的日志中获取OBJ参数,根据OBJ在数据字典中查询表所

属的用户,表名以及表的列信息。

[0044] b.在change参数OP=5.1的日志中获取修改之前的列号以及二进制数据。数据在日志中按照顺序存储,可通过列号判断改数据所属的列。

[0045] c.在change参数OP=11.5的日志中获取修改之后的列号以及二进制数据。

[0046] d.将获取到的二进制数据根据列类型解析为该类型的数据。

[0047] e.将解析出的数据,用户名,表名等信息拼接成SQL语句。

[0048] 与现有技术相比,本发明具有以下优点:

[0049] 1) 本发明通过实时监测生产环境中日志的数据变化情况,捕获有数据变化的日志并解析生成SQL语句,通过网络传输到灾备环境中,即实时同步至目标端,灾备环境在接收到SQL语句后装载执行,由于对日志的实时捕获解析以及同步,使得RPO和RTO趋近于0,保证Oracle数据库数据的实时一致性与可用性,有效降低RPO至趋近于0的效果。

[0050] 2) 本发明在实时复制时,对日志的整合和传输以事务为单位,无需处理复杂的回滚机制,在拥有高性能的同时还能够更好的保证数据复制的一致性和完整性;此外,实时复制不需要等待Redo Log文件归档后再进行日志读取等操作,而是直接读取日志的数据块内容,捕获日志和Oracle数据库写日志是并行操作而不互相影响。

[0051] 3) 在实时复制任务运行期间,生产端与灾备端数据库均处于open状态,灾备端数据能够实现查询,即时验证数据的有效性;此外,在正常情况下,实时复制都是实时的捕获Redo Log,在目标端执行完SQL语句后,在缓存目录中记录日志序列号及SCN号,在日常运行过程中,实时复制停止或其它原因需要读取归档日志时,可根据记录的序列号及SCN号获取归档日志,通过日志序列号和SCN可进一步保证数据的准确性。

[0052] 4) 对于本发明的实时复制任务,用户可以从界面配置中设置实时复制任务运行的时间段,实时复制软件会在指定时间段内将生产端数据同步至灾备端,用户自主设计同步时段,达到避开业务高峰期;实时复制任务是长期监控的任务,当任务出现异常时,无需人为干预,可自动尝试将任务恢复至正常状态;此外,控制台管理生产环境和灾备环境上的客户端,界面化操作客户端的方式,使用户操作实时复制任务更加便捷;同时,控制台可以实时监控任务的运行状态,用户可通过控制台了解任务当前执行情况。

[0053] 5) 本发明在增量复制过程中,源端占用CPU资源及内存资源低,不会影响源端生产环境的业务;而实时复制软件每天大约可分析处理1T左右日志,可满足业务需求,不会造成日志的累积。

[0054] 6) 本发明系统设置了生产环境的源端,以及作为灾备环境的目标端,源端生产环境出现宕机时,可以使用灾备环境替代生产环境,能够最大限度地减少数据丢失、经济损失以及生产力的降低。

[0055] 7) 本发明在实时复制过程中,可过滤掉生产环境删除用户或者删除表的操作,当遇到这些操作时,目标端不会执行删除操作,而是将表或者用户进行重命名操作,进而防止当用户误操作删除用户或者表时,用户和表无法恢复,有利于保证数据的完整性和安全性。

[0056] 8) 本发明的实时复制支持用户级复制和表级复制两种复制方式,用户级复制指源端数据库指定用户所包含的表、视图、索引、过程、函数、包、序列等数据对象全部复制到目标端数据库指定的用户下,同时,用户级复制支持源端用户名和目标端用户名不同;表级复制指源端数据库指定用户下的单个表复制到目标端数据库指定用户下的单个表,适用于落

盘数据、缓存数据等各种数据。

附图说明

[0057] 图1为本发明在数据保护时各个角色及角色间关系的结构图；

[0058] 图2为本发明发起实时复制任务及任务执行的流程图。

具体实施方式

[0059] 下面结合附图和具体实施例对本发明进行详细说明。显然，所描述的实施例是本发明的一部分实施例，而不是全部实施例。基于本发明中的实施例，本领域普通技术人员在没有做出创造性劳动的前提下所获得的所有其他实施例，都应属于本发明保护的范围。

[0060] 本发明提出了一种基于解析Oracle日志的持续数据保护技术，通过实时监测生产环境中日志的数据变化情况，捕获有数据变化的日志并解析生成SQL语句，通过网络传输到灾备环境中，灾备环境在接收到SQL语句后装载执行。由于对日志的实时捕获解析以及同步，使得RPO和RT0趋近于0，保证Oracle数据库数据的实时一致性与可用性。

[0061] 如图1所示，本发明涉及一种基于解析Oracle日志的持续数据保护系统，该系统将生产环境上的数据实时复制到灾备环境中，从而达到Oracle数据库中数据的保护。其中，Source是实时复制过程中的源端，Target是实时复制过程中的目标端。

[0062] 在本发明中，需要在源端和目标端分别安装Agent代理客户端。Agent作为实时复制软件在源端和目标端的代理，负责与Server进行交互，接收Server下达的命令，并针对命令做出相应的处理，并将执行结果返回给Server。

[0063] 本发明中的Server作为实时复制软件的控制台，是实时复制软件的管理单元，担负和用户交互的任务。在Server中管理所有的资源，包括对Agent的管理。Server负责发送实时复制的任务的指令给对应的Agent，同时，接收Agent返回的执行结果，监控任务的执行情况。

[0064] 本发明中的Source代表用户的生产环境。在实时复制任务运行时，源端实时读取Database的redo log (Redo日志文件) 或archive log (归档日志)，通过分析得到有数据变化的日志，该日志是以事务为基本单位。获取到有数据变化的日志后，解析日志并生成SQL语句。

[0065] 本发明中的Target代表用户的灾备环境。目标端接收来自源端的SQL语句并生成SQL文件，通过调用OCI在目标端Database中执行SQL语句。

[0066] 在本发明中，Database是源端和目标端数据库的实际数据存储。在第一次发起任务时，首先会进行全量复制，通过数据泵技术直接将源端Database中的数据同步到目标端的Database中。

[0067] 在本发明中，redo log和archive log为数据库中的重做日志和归档日志，负责存放所有对数据库更改的操作和更改的数据。Redo log的个数和大小由数据库指定，采用循环复写的方式写入数据，当数据库开启归档模式时，在一个redo log写满时，会进行归档，产生一个archive log。

[0068] 在本发明中，源端产生的SQL语句会经过TCP/IP网络协议传送到目标端，目标端接收到SQL语句后在目标端缓存目录中生成SQL文件。

[0069] 在本发明中,当保护的生产环境需要达到RPO趋近于0的效果时,操作员可部署实时复制任务。该任务中需要一个生产环境,即为实例中的源端,还需要一个灾备环境,即为实例中的目标端。操作员发起任务时,按照以下步骤执行备份操作:

[0070] 图2为发起数据保护任务时的任务流程示意图,实际当发起任务时,源端捕获日志,解析日志,生成SQL语句是多线程并发进行的,目标端接收SQL语句和执行SQL语句也是并发执行。图2中的流程具体为:

[0071] 步骤1、操作员选择需要保护的数据源及对象,建立实时复制任务,并发起任务,执行步骤2。需要注意的是:操作员创建任务时可选择对用户或者表进行保护。且操作员创建任务时需选择源端和目标端都需要选择缓存目录,用来存放需要记录的文件。

[0072] 步骤2、任务启动后,源端根据创建任务时配置的任务选项检测是否满足运行条件。如果满足,即状态正常,则执行步骤3,如果状态异常,重复执行步骤2。

[0073] 需要检测的选项包括:

[0074] 检测需要保护实例的数据库状态是否为open。如果状态为非open状态,则重试;否则,进行下一步。

[0075] 检测需要保护实例的数据库的模式是否为归档模式。如果状态为非归档模式,则重试;否则,进行下一步。

[0076] 检测需要保护实例的数据库中的日志模式是否为强制记录模式,并且是否开启日志补全。如果不是,则重试;否则,进行下一步。

[0077] 检测源端缓存目录是否有读写文件权限。如果没有,则重试;否则,进行下一步。

[0078] 获取源端当前序列号及SCN号。如果获取失败,则重试;否则,进行下一步。

[0079] 步骤3、源端将目标端所需要的参数发送至目标端,并且连接目标端,如果目标端连接成功,并且环境符合运行要求,则执行步骤4,否则重新开始执行步骤2。

[0080] 具体地,需要检测环境是否符合运行要求的选项包括:

[0081] 源端向目标端发送参数,并与目标端连接。如果源端与目标端连接失败,则返回步骤2;否则,进行下一步。

[0082] 目标端检查选择实例的数据库状态是否为open。如果状态为非open状态,则返回步骤2;否则,进行下一步。

[0083] 目标端检测目标端缓存目录是否有读写文件权限。如果没有,则返回步骤2;否则,进行下一步。

[0084] 步骤4、目标端获取上一次任务退出前记录的序列号及SCN号,且目标端将获取到的信息发送至源端,如果在目标端缓存目录中找不到记录序列号和SCN号的配置文件,那么目标端就向源端返回0。源端根据获取到的序列号判断是否进行全量。如果源端接收到目标端返回的数据不为0,并且能够使用序列号查询到日志,那么任务不需要进行全量复制,执行步骤6,否则执行步骤5。

[0085] 步骤5、源端通过全量复制将数据同步至目标端。在目标端创建目标端到源端的数据库链接,然后在目标端接收源端数据库中的数据并装载至目标端数据库。迁移完成后,执行步骤6。

[0086] 步骤6、源端获取数据库的redo日志或archive日志,通过分析日志的部分结构,判断该日志是否有数据变化,如果有数据变化则执行步骤7,否则继续获取日志。

[0087] 步骤7、源端获取到有数据变化的日志,该日志以事务为基本单位。解析日志,获取变化数据的信息,根据获取到变化信息中的obj号在数据字典中查询表名及用户名,将查询到的表名和用户名与创建任务时选择的对象进行对比,判断是否需要保护的对象。如果变化的数据属于需要被保护的对象,执行步骤8,否则继续执行步骤6,获取新的日志进行解析。

[0088] 步骤8、源端获取到变化的数据,生成对应的DDL语句和DML语句,以事务为单位将SQL语句写入缓存中,并且记录事务提交时的SCN号,执行步骤9。

[0089] 步骤9、源端从缓存中获取到SQL语句以及SCN号,将SQL语句和SCN号通过网络发送到目标端,执行步骤10。

[0090] 步骤10、目标端通过网络接收到源端发送的SQL语句以及SCN号,并将SQL语句写入缓存目录中,执行步骤11。

[0091] 步骤11、目标端通过OCI接口执行SQL语句,将数据同步到目标端的数据库中,执行步骤12。

[0092] 步骤12、1目标端在执行完SQL语句后,将SCN号写入配置文件中,当任务异常停止或者手动停止后,下一次重新发起任务的时候,源端如果还存在该事务所在的日志时,解析可以直接从该事务的下一个事务处进行解析,无需进行全量复制。当完成将SCN号写入配置文件的操作后,源端检测操作员是否发出停止任务的指令。如果未接收到停止指令,执行步骤6,如果操作员发出停止指令,执行步骤13。

[0093] 步骤13、源端检测到操作员发起的结束任务的指令,源端向目标端发送结束任务的消息,目标端接收到结束任务的消息后,结束任务,并向源端发出目标端任务结束的消息,源端结束任务。

[0094] 需要说明的是,本发明中的日志的解析技术主要的内容如下:

[0095] 1、众所周知,日志的逻辑结构包括以下三个部分:

[0096] a.改变向量(change vector):表示对数据库内某一个数据块所做的一次变更。包含了事务操作代码、变更从属数据块的地址以及更新后的数据。

[0097] b.重做记录(redo record):由一组改变向量组成,是一个改变向量的集合,代表一个数据库的变更(insert、update、delete等操作),构成数据库变更的最小恢复单位。

[0098] c.事务:是一组包含一条或多条语句的逻辑单元,每个事务都是一个原子单元。一个事务中包含多个重做记录。

[0099] Oracle日志文件是二进制格式文件,是由若干个指定大小的数据块组成的。在不同的操作系统上,块的大小是不同的,可以是512bytes,也可以是1K、2K等。日志文件的第一块是文件头,文件头中存储日志文件的数据块大小以及数据块的数量。日志文件的第二块重做日志头,该块中存储的是数据库相关信息,其中包括数据库ID,数据库名称,线程号等信息。

[0100] 从第二块起,每个数据块都有16字节数据头,包括当前数据块的编号,当前日志的序列号及偏移量,是当前数据块中,第一个redo record开始的位置。

[0101] 从第三个数据块开始,使用record结构记录Oracle的操作信息。Record包括record header和若干个change组成。

[0102] Record header中包括了LEN,VLD,RBA等参数。LEN是record结构的长度。VLD参数

决定了record header的长度,可以通过已有的VLD含义来解析出对应的长度。RBA是重做记录地址,由日志序列号,数据块号,在数据块内的偏移量组成,通过RBA可以快速定位recode在日志内的位置。

[0103] Change又分为change header,change length list和change body。

[0104] Change header的长度为24字节。Change header中包括OP,SCN等参数。OP是change的操作码,表示Oracle不同的数据操作。SCN是系统改变号,是在某个时间点定义数据库已提交版本的时间戳标记。

[0105] Change body根据change header中OP的不同拥有不同的结构。常见的操作码OP5.2是修改回滚段的头,是一个事务的开始。OP5.1为存储数据库操作前的旧数据。OP11.2记录插入(insert)数据的详细信息。OP11.3记录删除(delete)数据的详细信息。OP11.5记录更改(update)数据的详细信息。OP24.1记录DDL语句。OP5.4记录事务提交信息,是一个事务的结束。每个Change body中均存在XID字段,XID是事务的标识符。

[0106] 2、对于日志的解析,本发明采用以下方法实现:

[0107] 以事务为基本单位获取有数据变化的日志,获取时使用XID参数区分。事务中需存在OP5.2,OP5.1,OP5.4,OP11.*或OP24.1。以update操作为例分析解析过程:

[0108] a.在change参数OP=5.1的日志中获取OBJ参数,根据OBJ在数据字典中查询表所属的用户,表名以及表的列信息。

[0109] b.在change参数OP=5.1的日志中获取修改之前的列号以及二进制数据。数据在日志中按照顺序存储,可通过列号判断改数据所属的列。

[0110] c.在change参数OP=11.5的日志中获取修改之后的列号以及二进制数据。

[0111] d.将获取到的二进制数据根据列类型解析为该类型的数据。

[0112] e.将解析出的数据,用户名,表名等信息拼接成SQL语句。

[0113] 本发明通过实时监测生产环境中日志的数据变化情况,捕获有数据变化的日志并解析生成SQL语句,通过网络传输到灾备环境中,即实时同步至目标端,灾备环境在接收到SQL语句后装载执行,由于对日志的实时捕获解析以及同步,使得RPO和RTO趋近于0,保证Oracle数据库数据的实时一致性与可用性,有效降低RPO至趋近于0的效果。

[0114] 以上所述,仅为本发明的具体实施方式,但本发明的保护范围并不局限于此,任何熟悉本技术领域的工作人员在本发明揭露的技术范围内,可轻易想到各种等效的修改或替换,这些修改或替换都应涵盖在本发明的保护范围之内。因此,本发明的保护范围应以权利要求的保护范围为准。

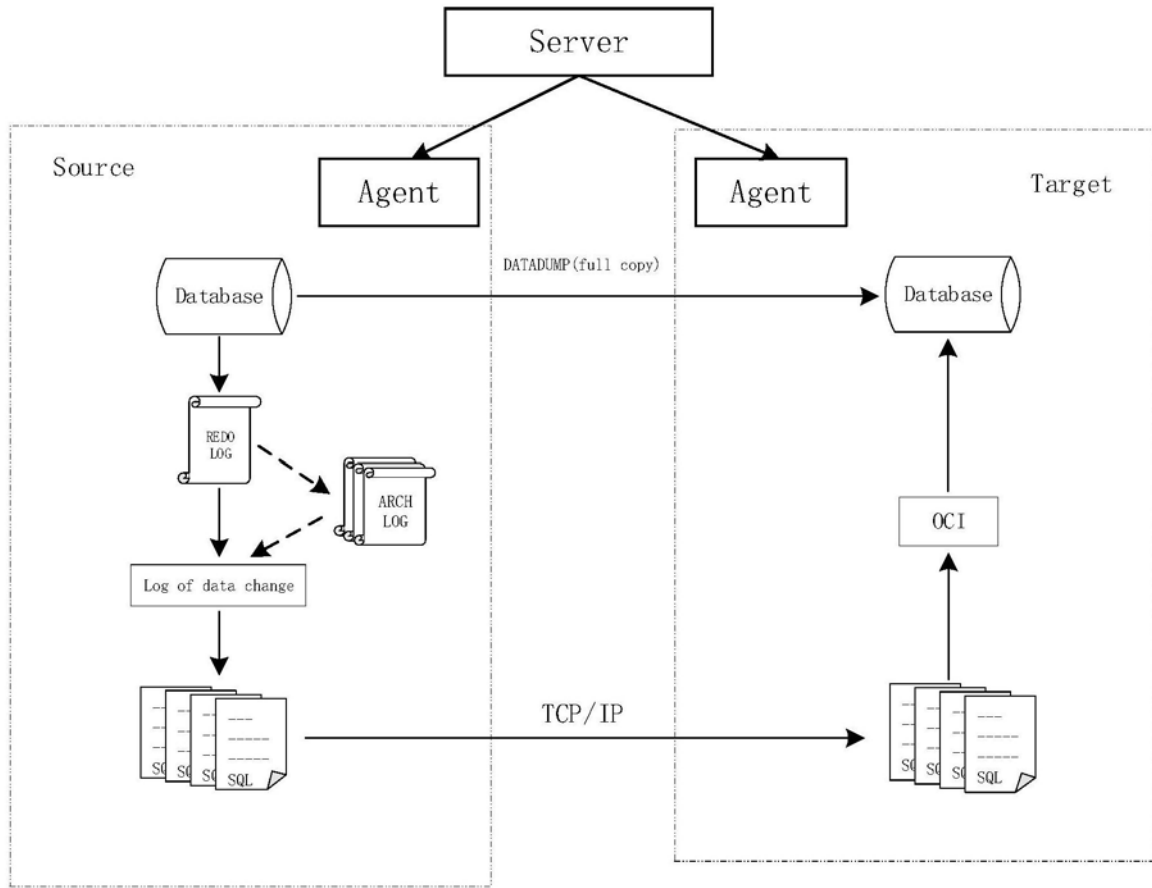


图1

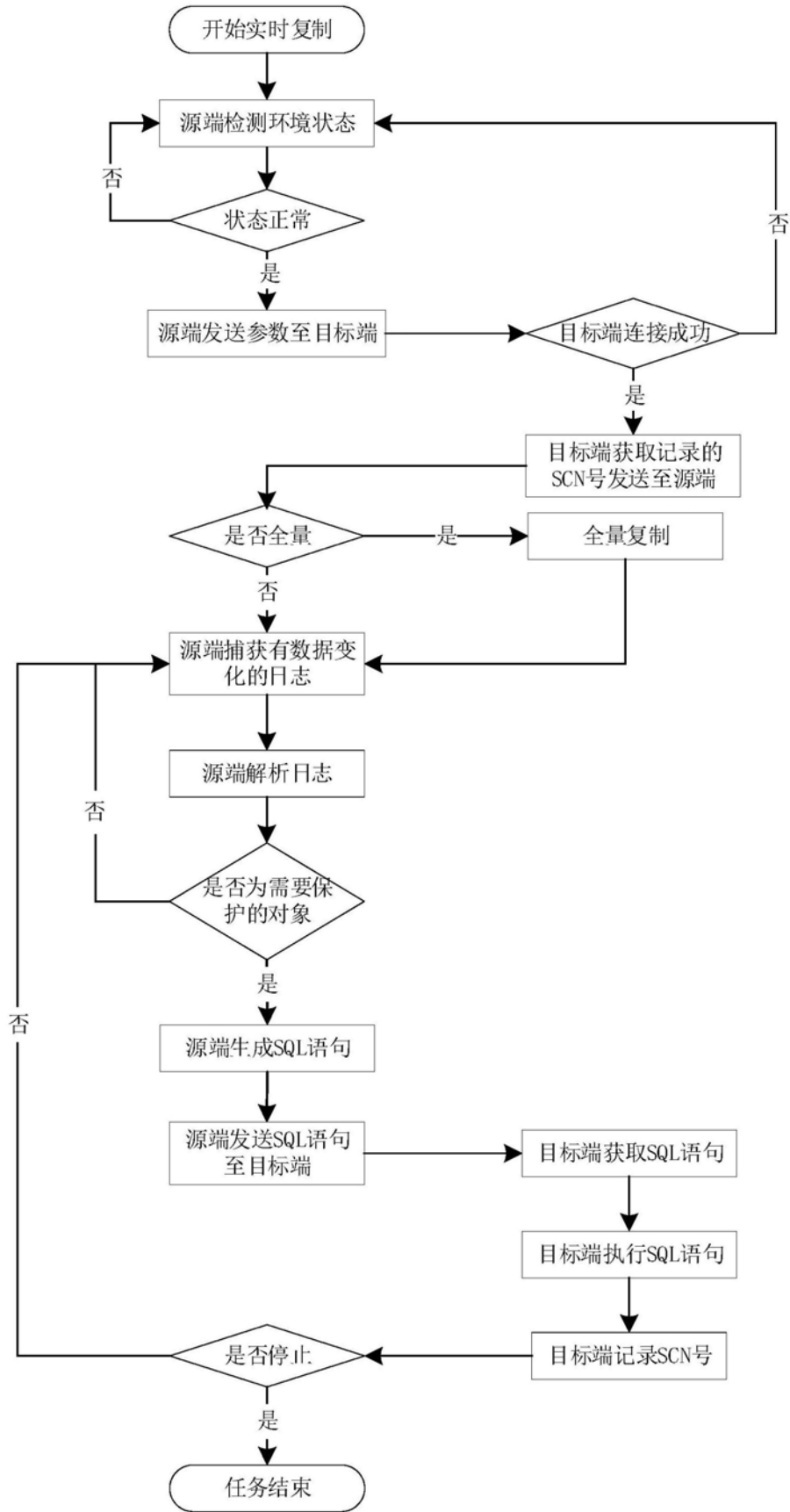


图2