



(12) 发明专利申请

(10) 申请公布号 CN 113407378 A

(43) 申请公布日 2021.09.17

(21) 申请号 202110700134.4

G06F 16/27 (2019.01)

(22) 申请日 2021.06.23

(71) 申请人 深圳前海微众银行股份有限公司
地址 518027 广东省深圳市前海深港合作
区前湾一路1号A栋201室

(72) 发明人 罗鹏

(74) 专利代理机构 北京同达信恒知识产权代理
有限公司 11291

代理人 雷航

(51) Int. Cl.

G06F 11/14 (2006.01)

G06F 16/22 (2019.01)

G06F 16/242 (2019.01)

G06F 16/2458 (2019.01)

G06F 16/25 (2019.01)

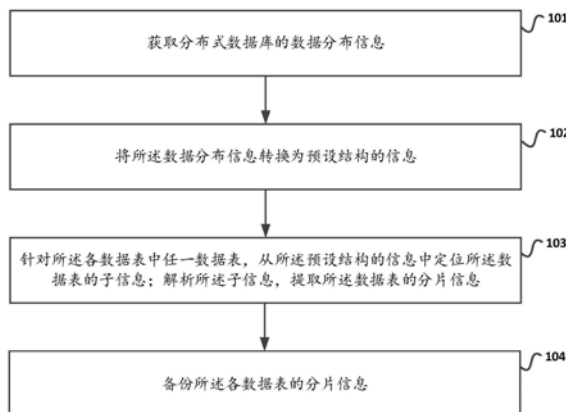
权利要求书2页 说明书12页 附图2页

(54) 发明名称

一种分布式数据库的分片信息备份方法及装置

(57) 摘要

本发明公开了一种分布式数据库的分片信息备份方法及装置,其中方法为:获取分布式数据库的数据分布信息;所述数据分布信息指示了所述分布式数据库的各数据表的数据记录在多个数据库实例上的分布;将所述数据分布信息转换为预设结构的信息;针对所述各数据表中任一数据表,从所述预设结构的信息中定位所述数据表的子信息;解析所述子信息,提取所述数据表的分片信息;并备份所述各数据表的分片信息。上述方法应用于金融科技(F i ntech)时,能够备份分布式数据库在多个数据库实例上的分片信息。



1. 一种分布式数据库的分片信息备份方法,其特征在于,包括:

获取分布式数据库的数据分布信息;所述数据分布信息指示了所述分布式数据库的各数据表的数据记录在多个数据库实例上的分布;

将所述数据分布信息转换为预设结构的信息;

针对所述各数据表中任一数据表,从所述预设结构的信息中定位所述数据表的子信息;解析所述子信息,提取所述数据表的分片信息;

备份所述各数据表的分片信息。

2. 如权利要求1所述的方法,其特征在于,所述将所述数据分布信息转换为预设结构的信息,包括:

将所述数据分布信息转换为二进制的数据库输入流,并将所述数据库输入流按预设格式转换为至少一个字符串;

将所述至少一个字符串作为所述预设结构的信息。

3. 如权利要求2所述的方法,其特征在于,所述从所述预设结构的信息中定位所述数据表的子信息,包括:

针对第一字符串,所述第一字符串为所述至少一个字符串中任一字符串,若所述第一字符串为数据库属性的字符串,且所述第一字符串中的数据库变量指示了所述分布式数据库,且所述第一字符串中的数据表变量指示了所述数据表,则确定所述第一字符串对应的预设位置范围内的字符串为所述数据表的子信息,所述预设位置范围内的字符串包括所述第一字符串。

4. 如权利要求3所述的方法,其特征在于,所述解析所述子信息,提取所述数据表的分片信息,包括:

针对所述预设位置范围内的字符串中任一字符串,若该字符串为所述第一字符串,则根据该字符串解析出所述分布式数据库的库名称以及所述数据表的表名称;

若该字符串中包括分片标识,则根据该字符串中所述分片标识对应的列名解析出分片列名;

将所述库名称、所述表名称和所述分片列名,作为所述数据表的分片信息。

5. 如权利要求1所述的方法,其特征在于,所述获取分布式数据库的数据分布信息,包括:

查询所述分布式数据库的系统表,获得所述多个数据库实例中的数据块存储信息;

根据所述多个数据库实例中的数据块存储信息,确定所述数据分布信息。

6. 如权利要求1-5任一项所述的方法,其特征在于,还包括:

获取所述分布式数据库中所述各数据表的表索引对象;

针对所述各数据表中任一数据表,根据所述数据表的表索引对象,确定所述数据表的索引字段和所述数据表的索引属性;

根据所述数据表的索引字段和所述数据表的索引属性,生成所述数据表的索引创建命令;所述索引创建命令用于备份所述数据表的索引信息。

7. 如权利要求6所述的方法,其特征在于,所述根据所述数据表的表索引对象,确定所述数据表的索引字段和所述数据表的索引属性,包括:

将所述索引对象中字段名与索引标识符匹配的字段,作为所述数据表的索引字段;

根据所述索引对象中第一字段的取值,确定所述数据表的索引是否为唯一索引;
根据所述索引对象中第二字段的取值,确定所述数据表的索引的过期时间。

8. 如权利要求1-5任一项所述的方法,其特征在于,还包括:

针对所述各数据表中任一数据表,读取所述数据表的数据记录;

针对所述数据表的数据记录的任一数据记录,获取所述数据记录中的默认主键,若所述默认主键为预设数据类型的主键,则将所述数据记录转换为json字符串;

将所述json字符串按照结构化查询语言SQL的格式,生成所述数据记录的SQL命令;所述SQL命令用于备份所述数据记录。

9. 一种分布式数据库的分片信息备份装置,其特征在于,包括:

备份模块,用于获取分布式数据库的数据分布信息;所述数据分布信息指示了所述分布式数据库的各数据表的数据记录在多个数据库实例上的分布;将所述数据分布信息转换为预设结构的信息;

解析模块,用于针对所述各数据表中任一数据表,从所述预设结构的信息中定位所述数据表的子信息;解析所述子信息,提取所述数据表的分片信息;

所述备份模块,还用于备份所述各数据表的分片信息。

10. 一种计算机设备,其特征在于,包括程序或指令,当所述程序或指令被执行时,如权利要求1至8中任意一项所述的方法被执行。

一种分布式数据库的分片信息备份方法及装置

技术领域

[0001] 本发明涉及金融科技(Fintech)领域中的数据库领域,尤其涉及一种分布式数据库的分片信息备份方法及装置。

背景技术

[0002] 随着计算机技术的发展,越来越多的技术应用在金融领域,传统金融业正在逐步向金融科技(Fintech)转变,但由于金融行业的安全性、实时性要求,也对技术提出的更高的要求。目前金融数据对容错性要求较高,因此,常采用分布式数据库(如mongodb)的方式存储金融数据,即分布式数据库的数据是分布在多个数据库实例(如每个主机运行一个数据库实例)上存储的,每个数据库实例只存储分布式数据库中的一部分数据。在一些情形下,需要对分布式数据库进行备份。

[0003] 然而,目前对分布式数据库的备份是针对其中数据记录的备份,但每个数据表的数据记录具体存储在哪个数据库实例中,依据的是数据表的分片信息(数据表的一个或多个列名),而且分布式数据库中并没有提供获取数据表的分片信息的接口或命令,所以目前还不能备份分布式数据库中数据表的分片信息。这是一个亟待解决的问题。

发明内容

[0004] 本发明提供一种分布式数据库的分片信息备份方法及装置,解决了现有技术中不能备份分布式数据库中数据表的分片信息的问题。

[0005] 第一方面,本发明提供一种分布式数据库的分片信息备份方法,包括:

[0006] 获取分布式数据库的数据分布信息;所述数据分布信息指示了所述分布式数据库的各数据表的数据记录在多个数据库实例上的分布;

[0007] 将所述数据分布信息转换为预设结构的信息;

[0008] 针对所述各数据表中任一数据表,从所述预设结构的信息中定位所述数据表的子信息;解析所述子信息,提取所述数据表的分片信息;

[0009] 备份所述各数据表的分片信息。

[0010] 上述方法中,将所述数据分布信息转换为预设结构的信息后,可以从预设结构的信息中定位到数据表的子信息,从而解析子信息,提取出该数据表的分片信息,从而基于数据分布信息,通过数据的处理、转换,直接提取到数据表的分片信息,从而能够备份数据表的分片信息。

[0011] 可选的,所述将所述数据分布信息转换为预设结构的信息,包括:

[0012] 将所述数据分布信息转换为二进制的数据库输入流,并将所述数据库输入流按预设格式转换为至少一个字符串;

[0013] 将所述至少一个字符串作为所述预设结构的信息。

[0014] 上述方法中,将所述数据分布信息转换为二进制的数据库输入流,并将所述数据库输入流按预设格式转换为至少一个字符串,从而使得所述数据分布信息标准化、结构化,从而

获得的所述预设结构的信息可以更便于数据处理,便于提取出各数据表的分片信息。

[0015] 可选的,所述从所述预设结构的信息中定位所述数据表的子信息,包括:针对第一字符串,所述第一字符串为所述至少一个字符串中任一字符串,若所述第一字符串为数据库属性的字符串,且所述第一字符串中的数据库变量指示了所述分布式数据库,且所述第一字符串中的数据表变量指示了所述数据表,则确定所述第一字符串对应的预设位置范围内的字符串为所述数据表的子信息,所述预设位置范围内的字符串包括所述第一字符串。

[0016] 上述方法中,通过数据库属性和数据库变量、数据表变量等内容定位出了所述数据表的子信息,由于可以通过数据库属性判断是所述分布式数据库,且数据库和数据表在创建时均是唯一的,从而可以更高效、准确地从所述预设结构的信息中定位需要解析的数据表的子信息。

[0017] 可选的,所述解析所述子信息,提取所述数据表的分片信息,包括:

[0018] 针对所述预设位置范围内的字符串中任一字符串,若该字符串为所述第一字符串,则根据该字符串解析出所述分布式数据库的库名称以及所述数据表的表名称;若该字符串中包括分片标识,则根据该字符串中所述分片标识对应的列名解析出分片列名;将所述库名称、所述表名称和所述分片列名,作为所述数据表的分片信息。

[0019] 上述方式下,第一字符串解析出所述分布式数据库的库名称以及所述数据表的表名称,并对其中包括分片标识的字符串解析,得到分片列名,从而将所述库名称、所述表名称和所述分片列名都得到,因此数据表的分片信息可以清晰地表示出数据库数据表信息以及相应的数据库实例上分布的依据。

[0020] 可选的,所述获取分布式数据库的数据分布信息,包括:查询所述分布式数据库的系统表,获得所述多个数据库实例中的数据块存储信息;根据所述多个数据库实例中的数据块存储信息,确定所述数据分布信息。

[0021] 上述方式下,通过查询所述分布式数据库的系统表,获得所述多个数据库实例中的数据块存储信息,可以从物理分布上直接获取所述数据分布信息,从而更为快捷、直接。

[0022] 可选的,还包括:

[0023] 获取所述分布式数据库中所述各数据表的表索引对象;

[0024] 针对所述各数据表中任一数据表,根据所述数据表的表索引对象,确定所述数据表的索引字段和所述数据表的索引属性;

[0025] 根据所述数据表的索引字段和所述数据表的索引属性,生成所述数据表的索引创建命令;所述索引创建命令用于备份所述数据表的索引信息。

[0026] 上述方式下,确定所述数据表的索引字段和所述数据表的索引属性后,根据所述数据表的索引字段和所述数据表的索引属性,生成所述数据表的索引创建命令,从而将所述数据表的索引通过索引创建命令的形式保存下来,从而方便分布式数据库的恢复。

[0027] 可选的,所述根据所述数据表的表索引对象,确定所述数据表的索引字段和所述数据表的索引属性,包括:

[0028] 将所述索引对象中字段名与索引标识符匹配的字段,作为所述数据表的索引字段;

[0029] 根据所述索引对象中第一字段的取值,确定所述数据表的索引是否为唯一索引;

[0030] 根据所述索引对象中第二字段的取值,确定所述数据表的索引的过期时间。

[0031] 上述方式下,通过索引标识符匹配以及第一字段、第二字段的取值,从而更准确地确定所述数据表的索引字段和所述数据表的索引属性。

[0032] 可选的,还包括:

[0033] 针对所述各数据表中任一数据表,读取所述数据表的数据记录;

[0034] 针对所述数据表的数据记录的任一数据记录,获取所述数据记录中的默认主键,若所述默认主键为预设数据类型的主键,则将所述数据记录转换为json字符串;

[0035] 将所述json字符串按照结构化查询语言SQL的格式,生成所述数据记录的SQL命令;所述SQL命令用于备份所述数据记录。

[0036] 上述方式下,针对任一数据记录,若所述默认主键为预设数据类型的主键,都可以将所述数据记录转换为json字符串,进而生成数据记录的SQL命令,由于主键在数据表中是唯一的,所以能准确地保留数据记录的基本信息,且json字符串占用的存储空间较小,可以更直接、准确地保存数据记录,且生成可以直接执行的SQL命令,从而方便数据记录的恢复。

[0037] 第二方面,本发明提供一种分布式数据库的分片信息备份装置,包括:备份模块,用于获取分布式数据库的数据分布信息;所述数据分布信息指示了所述分布式数据库的各数据表的数据记录在多个数据库实例上的分布;将所述数据分布信息转换为预设结构的信息;

[0038] 解析模块,用于针对所述各数据表中任一数据表,从所述预设结构的信息中定位所述数据表的子信息;解析所述子信息,提取所述数据表的分片信息;

[0039] 所述备份模块,还用于备份所述各数据表的分片信息。

[0040] 可选的,所述解析模块具体用于:

[0041] 将所述数据分布信息转换为二进制的数据库输入流,并将所述数据库输入流按预设格式转换为至少一个字符串;

[0042] 将所述至少一个字符串作为所述预设结构的信息。

[0043] 可选的,所述解析模块具体用于:

[0044] 针对第一字符串,所述第一字符串为所述至少一个字符串中任一字符串,若所述第一字符串为数据库属性的字符串,且所述第一字符串中的数据库变量指示了所述分布式数据库,且所述第一字符串中的数据表变量指示了所述数据表,则确定所述第一字符串对应的预设位置范围内的字符串为所述数据表的子信息,所述预设位置范围内的字符串包括所述第一字符串。

[0045] 可选的,所述解析模块具体用于:

[0046] 针对所述预设位置范围内的字符串中任一字符串,若该字符串为所述第一字符串,则根据该字符串解析出所述分布式数据库的库名称以及所述数据表的表名称;

[0047] 若该字符串中包括分片标识,则根据该字符串中所述分片标识对应的列名解析出分片列名;

[0048] 将所述库名称、所述表名称和所述分片列名,作为所述数据表的分片信息。

[0049] 可选的,所述解析模块具体用于:查询所述分布式数据库的系统表,获得所述多个数据库实例中的数据块存储信息;

[0050] 根据所述多个数据库实例中的数据块存储信息,确定所述数据分布信息查询所述分布式数据库的系统表,获得所述多个数据库实例中的数据块存储信息;

- [0051] 根据所述多个数据库实例中的数据块存储信息,确定所述数据分布信息。
- [0052] 可选的,所述备份模块还用于:
- [0053] 获取所述分布式数据库中所述各数据表的表索引对象;
- [0054] 针对所述各数据表中任一数据表,根据所述数据表的表索引对象,确定所述数据表的索引字段和所述数据表的索引属性;
- [0055] 根据所述数据表的索引字段和所述数据表的索引属性,生成所述数据表的索引创建命令;所述索引创建命令用于备份所述数据表的索引信息。
- [0056] 可选的,所述备份模块具体用于:
- [0057] 将所述索引对象中字段名与索引标识符匹配的字段,作为所述数据表的索引字段;
- [0058] 根据所述索引对象中第一字段的取值,确定所述数据表的索引是否为唯一索引;
- [0059] 根据所述索引对象中第二字段的取值,确定所述数据表的索引的过期时间。
- [0060] 可选的,所述备份模块还用于:
- [0061] 针对所述各数据表中任一数据表,读取所述数据表的数据记录;
- [0062] 针对所述数据表的数据记录的任一数据记录,获取所述数据记录中的默认主键,若所述默认主键为预设数据类型的主键,则将所述数据记录转换为json字符串;
- [0063] 将所述json字符串按照结构化查询语言SQL的格式,生成所述数据记录的SQL命令;所述SQL命令用于备份所述数据记录。
- [0064] 上述第二方面及第二方面各个可选装置的有益效果,可以参考上述第一方面及第一方面各个可选方法的有益效果,这里不再赘述。
- [0065] 第三方面,本发明提供一种计算机设备,包括程序或指令,当所述程序或指令被执行时,用以执行上述第一方面及第一方面各个可选的方法。
- [0066] 第四方面,本发明提供一种存储介质,包括程序或指令,当所述程序或指令被执行时,用以执行上述第一方面及第一方面各个可选的方法。
- [0067] 本发明的这些方面或其他方面在以下实施例的描述中会更加简明易懂。

附图说明

- [0068] 为了更清楚地说明本发明实施例中的技术方案,下面将对实施例描述中所需要使用的附图作简要介绍,显而易见地,下面描述中的附图仅仅是本发明的一些实施例,对于本领域的普通技术人员来讲,在不付出创造性劳动性的前提下,还可以根据这些附图获得其他的附图。
- [0069] 图1为本发明实施例提供的一种分布式数据库的分片信息备份方法对应的流程示意图;
- [0070] 图2为本发明实施例提供的一种分布式数据库的分片信息备份方法中预设结构的信息的示意图;
- [0071] 图3为本发明实施例提供的一种分布式数据库的分片信息备份方法对应的时序示意图;
- [0072] 图4为本发明实施例提供的一种分布式数据库的分片信息备份装置的结构示意图。

具体实施方式

[0073] 为了使本发明的目的、技术方案和优点更加清楚,下面将结合附图对本发明作进一步地详细描述,显然,所描述的实施例仅仅是本发明一部分实施例,而不是全部的实施例。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前提下所获得的所有其它实施例,都属于本发明保护的范围。

[0074] 在金融机构(银行机构、保险机构或证券机构)在进行业务(如银行的贷款业务、存款业务等)运转过程中,分布式数据库的数据是分布在多个数据库实例(如每个主机运行一个数据库实例)上存储的,在一些情形下,需要对分布式数据库进行备份。目前对分布式数据库的备份是针对整体数据的备份,但不能备份分布式数据库在多个数据库实例上的分片信息。这种情况不符合银行等金融机构的需求,无法保证金融机构各项业务的高效运转。

[0075] 如图1所示,本发明提供一种分布式数据库的分片信息备份方法。

[0076] 步骤101:获取分布式数据库的数据分布信息。

[0077] 步骤102:将所述数据分布信息转换为预设结构的信息。

[0078] 步骤103:针对所述各数据表中任一数据表,从所述预设结构的信息中定位所述数据表的子信息;解析所述子信息,提取所述数据表的分片信息。

[0079] 步骤104:备份所述各数据表的分片信息。

[0080] 举例来说,上述分布式数据库可以为mongodb,mongodb是一个基于分布式文件存储的数据库,是非关系型数据库,同时也是一个NOSQL数据库产品。分片指分布式数据库分布在每个实例上的数据。举例来说,mongodb可以将数据打散分布在不同的mongodb实例,这个实例就是一个分片。例如A表有3千万条数据,数据库有3个分片(3台机器),每个分片可以存储1千万条数据。当在某个分片查询时,只需要从原数据范围的1/3就能查到期望的数据,极大提高了查询效率。分片信息则是指,将数据分散到不同机器所需要指定的列名,这些列名也可以称为片键。

[0081] 需要说明的是,目前的实际应用场景中,备份完整的分布式数据库还需要备份分布式数据库的各数据表的待备份索引信息,待备份索引信息用于恢复分布式数据库中各数据表的索引,那么基于恢复的分布式数据库中各数据表的索引,所述各数据表的待备份数据记录可以按照索引进行恢复,得到各数据表的数据记录,再基于通过数据分布信息,便可以解析得到各数据表的分片信息,便可以基于分布式数据库、各数据表以及各数据表的分片信息进行备份。

[0082] 需要说明的是,在实际软件实现时,可以在步骤101之前修改数据库备份配置文件,指定需要连接的分布式数据库,同时指定分布式数据库中的哪些表需要导出数据记录,可以选择分布式数据库中的部分数据进行备份,各数据表的数据既可以是分布式数据库中全部的数据,也可以是部分数据,各个步骤可以封装在一个Shell脚本中。

[0083] 具体来说,Shell脚本中可以封装backup函数和restore函数.backup函数用于执行步骤101~步骤104,具体可以运行jar包命令的形式封装进backup函数,从而简化用户操作,降低出错概率,具体命令可以为backup.sh backup.restore函数用于连接数据库,并依次执行各数据表的索引恢复、各数据表的数据记录恢复、各数据表的分片信息恢复,具体命令可以为backup.sh restore。

[0084] 一种可能的实现方式中,备份分布式数据库的各数据表的待备份索引信息的步骤

具体可以为：

[0085] 获取所述分布式数据库中所述各数据表的表索引对象；

[0086] 针对所述各数据表中任一数据表，根据所述数据表的表索引对象，确定所述数据表的索引字段和所述数据表的索引属性；

[0087] 根据所述数据表的索引字段和所述数据表的索引属性，生成所述数据表的索引创建命令；所述索引创建命令用于备份所述数据表的索引信息。

[0088] 需要说明的是，上述每个数据表的表索引对象中记录了这个数据表的索引字段和所述数据表的索引属性，具体如何根据所述数据表的表索引对象，确定所述数据表的索引字段和所述数据表的索引属性的方式有多种，比如，在表索引对象中定义变量，通过给变量赋值记录索引字段和索引属性，也可以通过定义函数，由函数的运行结果来记录索引字段和索引属性。

[0089] 举例来说，索引创建命令具体为：`db.collection.createIndex(keys,options)`。`db.collection.createIndex`为索引创建命令的名称，`keys,options`为参数。

[0090] 其中，`keys`参数定义了索引由哪些字段组成，还可以包含其他的信息，如索引字段的排序方式，样例为`{acctNo:1,mntSysTime:1}`。`options`参数定义了索引属性，以及是否在后台创建索引，是否唯一，索引名称，数据过期时间等。

[0091] 上述方式下，确定所述数据表的索引字段和所述数据表的索引属性后，根据所述数据表的索引字段和所述数据表的索引属性，生成所述数据表的索引创建命令，从而将所述数据表的索引通过索引创建命令的形式保存下来，从而方便分布式数据库的恢复。

[0092] 上述实现方式中，根据所述数据表的表索引对象，确定所述数据表的索引字段和所述数据表的索引属性的具体过程可以如下：

[0093] 将所述索引对象中字段名与索引标识符匹配的字段，作为所述数据表的索引字段；

[0094] 根据所述索引对象中第一字段的取值，确定所述数据表的索引是否为唯一索引；

[0095] 根据所述索引对象中第二字段的取值，确定所述数据表的索引的过期时间。

[0096] 举例来说，具体步骤如下：

[0097] 读取各数据表的名称，便于根据表名获取数据表的索引以及备份数据；遍历数据表，获取数据表的表索引对象；并根据表索引对象解析索引字段和索引属性。同时根据解析出来的索引字段和索引属性生成索引创建命令，方便后续索引恢复，具体步骤举例如下：

[0098] 组装`keys`参数：获取表索引对象`key`字段，得到索引字段对象，将对象转化为字符串。

[0099] 组装`options`参数：

[0100] 从表索引对象获取`background`字段，得到索引是否需要后台创建，该字段是否可以为空，需要兼容为空的情况，具体可由`background`字段的取值表征。

[0101] 从表索引对象获取`name`字段，得到索引名称。

[0102] 从表索引对象获取`unique`字段（第一字段），得到索引是否是唯一索引，该字段可能为空，需要兼容为空的情况。

[0103] 从表索引对象获取`expireAfterSeconds`字段（第二字段），得到数据过期时间，该字段可能为空，为空则表示数据永不过期。需要判断过期时间为整数还是浮点数据，然后根

据数据类型得到不同数据类型的值。

[0104] 根据字符串拼接得到索引可执行命令,将命令写入到文件。样例:db.表名.createIndex({acctNo:1,mntSysTime:1},{name:idx_xxxx,backgroud:xxx,expireAfterSeconds:xxx,unique:xxxx}),createIndex表示索引执行命令,{acctNo:1,mntSysTime:1}表示索引字段,{name:idx_xxxx,backgroud:xxx,expireAfterSeconds:xxx,unique:xxxx})表示索引属性,name,backgroud,expireAfterSeconds,unique均为索引属性。

[0105] 上述方式下,通过索引标识符匹配以及第一字段、第二字段的取值,从而更准确地确定所述数据表的索引字段和所述数据表的索引属性。

[0106] 一种可能的实现方式中,备份所述各数据表的待备份数据记录的具体步骤可以为:

[0107] 针对所述各数据表中任一数据表,读取所述数据表的数据记录;针对所述数据表的数据记录的任一数据记录,获取所述数据记录中的默认主键,若所述默认主键为预设数据类型的主键,则将所述数据记录转换为json字符串;将所述json字符串按照结构化查询语言SQL的格式,生成所述数据记录的SQL命令;所述SQL命令用于备份所述数据记录。具体步骤可以如下:

[0108] 根据配置文件配置需要备份的各数据表的数据;将读取到的各数据表的数据生成可执行命令,方便后续数据恢复,具体步骤如下:

[0109] 根据各数据表的表名,然后检查当前数据表是否需要备份数据,如果不需要则跳过数据备份。读取表所有数据,并遍历;还可以获取默认主键的字段如“_id”字段。该字段为默认主键,数据插入时不填则自动生成主键。这一步需要检查返回的主键是分布式数据库的系统自行生成带有业务含义的主键(预设数据类型的主键),还是数据库自动生无实际意义的主键。如果是有业务含义的主键则需要保留,不具有业务含义的主键在生成可执行命令时需忽略,通过检查返回“_id”的数据类型判断。

[0110] 可以通过toJson方法将各数据表的待备份数据记录(即各数据表的数据)转化为可以命令行执行的字符串,拼接数据插入命令,并将命令写入文件。最终拼接完成的命令样例:db.表名.insert({字段A:字段A的值,字段B:字段B值}),从而拼接为SQL格式的SQL命令,需要说明的是,SQL命令可以是SQL语句,也可以不是SQL语句,SQL命令和SQL语句可以通过预设规则建立映射关系,从而能够减小SQL命令的存储空间大小。

[0111] 上述方式下,针对任一数据记录,若所述默认主键为预设数据类型的主键,都可以将所述数据记录转换为json字符串,进而生成数据记录的SQL命令,由于主键在数据表中是唯一的,所以能准确地保留数据记录的基本信息,且json字符串占用的存储空间较小,可以更直接、准确地保存数据记录,且生成可以直接执行的SQL命令,从而方便数据记录的恢复。

[0112] 一种可能的实现方式中,步骤101中获取所述分布式数据库在多个数据库实例上的数据分布信息的具体步骤可以为:

[0113] 执行状态查询命令,获得分布式数据库的数据分布信息。

[0114] 步骤102具体可以为:

[0115] 将所述数据分布信息转换为二进制的的数据输入流,并将所述数据输入流按预设格式转换为至少一个字符串;将所述至少一个字符串作为所述预设结构的信息。

[0116] 具体来说,可以如下:

[0117] 执行sh.status命令(状态查询命令)。一个分布式数据库(如mongodb)集群可以有多个数据库实例,该命令返回数据库集群所有数据库的数据分布信息。例如每个片的块信息等,但该命令并不是直接的数据查询命令,该命令返回的信息为非结构化、杂乱的文本信息,可读性较差。数据查询命令执行完成后直接返回结果,sh.status命令执行完成后并不能直接得到执行结果,只是简单的把结果输出到控制台(多行),命令的返回值为空。所以可以通过调用java底层方法Runtime.getRuntime().exec(命令)得到一个输入流,然后读取数据流中的二进制数据,按行读取,然后将每行读取的结果转换为字符串存入集合,作为预设结构的信息,具体代码样例可以如图2所示,以便后续解析。

[0118] 上述方法中,将所述数据分布信息转换为二进制的数据库输入流,并将所述数据库输入流按预设格式转换为至少一个字符串,从而使得所述数据分布信息标准化、结构化,从而获得的所述预设结构的信息可以更便于数据处理,便于提取出各数据表的分片信息。

[0119] 具体来说,所述解析所述数据分布信息,获得所述各数据表的分片信息具体过程可以如下:

[0120] 针对第一字符串,所述第一字符串为所述至少一个字符串中任一字符串,若所述第一字符串为数据库属性的字符串,且所述第一字符串中的数据库变量指示了所述分布式数据库,且所述第一字符串中的数据表变量指示了所述数据表,则确定所述第一字符串对应的预设位置范围内的字符串为所述数据表的子信息,所述预设位置范围内的字符串包括所述第一字符串。

[0121] 上述解析所述子信息,提取所述数据表的分片信息的具体步骤如下:

[0122] 针对所述预设位置范围内的字符串中任一字符串,若该字符串为所述第一字符串,则根据该字符串解析出所述分布式数据库的库名称以及所述数据表的表名称;若该字符串中包括分片标识,则根据该字符串中所述分片标识对应的列名解析出分片列名;将所述库名称、所述表名称和所述分片列名,作为所述数据表的分片信息。

[0123] 上述方式下,第一字符串解析出所述分布式数据库的库名称以及所述数据表的表名称,并对其中包括分片标识的字符串解析,得到分片列名,从而将所述库名称、所述表名称和所述分片列名都得到,因此数据表的分片信息可以清晰地表示出数据库数据表信息以及相应的数据库实例上分布的依据。

[0124] 具体步骤可以如下:

[0125] 检查所述字符串是否以“databases:”开头(是否为数据库属性的字符串),如果是则将gotDatabaseNode变量的取值设为true。如果gotDatabaseNode(数据库变量)为true(预设取值),并且该字符串解析为“{\“_id\“: \“”+分布式数据库的数据库名称+ \“”,”开头,将gotDbnameNode设为true。

[0126] 如果gotDbnameNode为true,并且该字符串以分布式数据库的数据库名称开头,那么该字符串可以解析出所述分布式数据库的库名称以及所述数据表的表名称,同时该行的后续一行(预设位置范围内的字符串)则是需要包括分片标识的字符串,可以解析出分片列名。

[0127] 上述解析分片列名的过程如下:

[0128] 获取第一字符串的下一行字符串。如果该行字符串以“shardkey:”开头,则截取

“shardkey:”以后的字符串,截取后的字符串为分片列名。样例:shard key: {“custacctNo”:1,“entityCode”:1}。

[0129] 那么将两步骤截取到的结果拼接为数据库可执行的分片命令。样例:db.runCommand({shardCollection:“数据库名称.表名”,key: {“字段A”:1.0,“字段B”:1.0,“字段C”:1.0}})。同时将结果写入文件。

[0130] 每个字符串都按照上述步骤遍历,直到遍历完所有字符串,需要说明的是,为列方便解析,可以设置每个字符串为一行。

[0131] 以上步骤已完成的表结构、数据以及分片信息备份。如果不需要恢复,以下步骤可以不操作。这一步修改需要将备份的数据恢复表哪个数据库。

[0132] 另一种可能的实现方式中,步骤101中获取所述分布式数据库在多个数据库实例上的数据分布信息的具体步骤可以为:

[0133] 查询所述分布式数据库的系统表,获得所述多个数据库实例中的数据块存储信息;

[0134] 根据所述多个数据库实例中的数据块存储信息,确定所述数据分布信息。

[0135] 由于在关系型数据库中,所有表信息,包括索引、字段、数据类型、主键、分区等所有信息都能从元数据库找到。那么类比关系型数据库,在分布式数据库中(如mongodb)就会找到chunks表,这是一张mongodb的系统表,保存着数据块在mongodb集群中的数据块存储信息。使用数据块存储信息可以通过进一步解析获取分片信息,生成分片命令。

[0136] 上述方式下,通过查询所述分布式数据库的系统表,获得所述多个数据库实例中的数据块存储信息,可以从物理分布上直接获取所述数据分布信息,从而更为快捷、直接。

[0137] 步骤101~步骤104的方法中,将所述数据分布信息转换为预设结构的信息后,可以从预设结构的信息中定位到数据表的子信息,从而解析子信息,提取出该数据表的分片信息,从而基于数据分布信息,通过数据的处理、转换,直接提取到数据表的分片信息,从而能够备份数据表的分片信息。

[0138] 下面结合图3,进一步详细描述本发明提供一种分布式数据库的分片信息备份方法。

[0139] 步骤1:修改数据库连接信息以及指定哪些表需要备份数据。

[0140] 步骤2:执行backup.sh backup命令。

[0141] 步骤3:获取分布式数据库的各数据表(图3中举例为所有数据表)。

[0142] 步骤4:遍历分布式数据库的各数据表,获取各数据表的索引对象(索引信息对象)。

[0143] 步骤5:解析返回的索引对象,筛选索引相关信息(索引字段和索引属性),并生成可执行命令(索引创建命令)。

[0144] 步骤6:根据配置文件读取各数据表的数据记录(基础数据)。

[0145] 步骤7:将各数据表的数据记录生成可执行命令。

[0146] 步骤8:执行sh.status。

[0147] 步骤9:解析返回文本(预设结构的信息),生成分片信息可执行命令。

[0148] 步骤10:执行需要恢复的数据库信息。

[0149] 步骤11:执行backup.sh restore恢复各数据表的索引与分片信息。

- [0150] 步骤12:执行步骤5的执行创建索引命令。
- [0151] 步骤13:执行步骤7的基础数据恢复命令。
- [0152] 步骤14:执行步骤10的命令创建分片命令。
- [0153] 需要说明的是,基于图3示出的方法,每次版本上线后执行backup.sh backup命令备份分布式数据库的索引,数据记录以及分片信息,所以在需要恢复时可以选择恢复到任意版本。这样就不需要手动整合按顺序整理历史的每个版本的数据库脚本。并且,上述中会过滤掉不需要备份数据的数据表,只备份指定的各数据表,效率高。数据恢复只需要执行backup.sh restore,命令简单,可操作性高。
- [0154] 如图4所示,本发明提供一种分布式数据库的分片信息备份装置,包括:备份模块401,用于获取分布式数据库的数据分布信息;所述数据分布信息指示了所述分布式数据库的各数据表的数据记录在多个数据库实例上的分布;将所述数据分布信息转换为预设结构的信息;
- [0155] 解析模块402,用于针对所述各数据表中任一数据表,从所述预设结构的信息中定位所述数据表的子信息;解析所述子信息,提取所述数据表的分片信息;
- [0156] 所述备份模块401,还用于备份所述各数据表的分片信息。
- [0157] 可选的,所述解析模块402具体用于:
- [0158] 将所述数据分布信息转换为二进制的数据库输入流,并将所述数据库输入流按预设格式转换为至少一个字符串;
- [0159] 将所述至少一个字符串作为所述预设结构的信息。
- [0160] 可选的,所述解析模块402具体用于:
- [0161] 针对第一字符串,所述第一字符串为所述至少一个字符串中任一字符串,若所述第一字符串为数据库属性的字符串,且所述第一字符串中的数据库变量指示了所述分布式数据库,且所述第一字符串中的数据表变量指示了所述数据表,则确定所述第一字符串对应的预设位置范围内的字符串为所述数据表的子信息,所述预设位置范围内的字符串包括所述第一字符串。
- [0162] 可选的,所述解析模块402具体用于:
- [0163] 针对所述预设位置范围内的字符串中任一字符串,若该字符串为所述第一字符串,则根据该字符串解析出所述分布式数据库的库名称以及所述数据表的表名称;
- [0164] 若该字符串中包括分片标识,则根据该字符串中所述分片标识对应的列名解析出分片列名;
- [0165] 将所述库名称、所述表名称和所述分片列名,作为所述数据表的分片信息。
- [0166] 可选的,所述解析模块402具体用于:查询所述分布式数据库的系统表,获得所述多个数据库实例中的数据块存储信息;
- [0167] 根据所述多个数据库实例中的数据块存储信息,确定所述数据分布信息查询所述分布式数据库的系统表,获得所述多个数据库实例中的数据块存储信息;
- [0168] 根据所述多个数据库实例中的数据块存储信息,确定所述数据分布信息。
- [0169] 可选的,所述备份模块401还用于:
- [0170] 获取所述分布式数据库中所述各数据表的表索引对象;
- [0171] 针对所述各数据表中任一数据表,根据所述数据表的表索引对象,确定所述数据

表的索引字段和所述数据表的索引属性;

[0172] 根据所述数据表的索引字段和所述数据表的索引属性,生成所述数据表的索引创建命令;所述索引创建命令用于备份所述数据表的索引信息。

[0173] 可选的,所述备份模块401具体用于:

[0174] 将所述索引对象中字段名与索引标识符匹配的字段,作为所述数据表的索引字段;

[0175] 根据所述索引对象中第一字段的取值,确定所述数据表的索引是否为唯一索引;

[0176] 根据所述索引对象中第二字段的取值,确定所述数据表的索引的过期时间。

[0177] 可选的,所述备份模块401还用于:

[0178] 针对所述各数据表中任一数据表,读取所述数据表的数据记录;

[0179] 针对所述数据表的数据记录的任一数据记录,获取所述数据记录中的默认主键,若所述默认主键为预设数据类型的主键,则将所述数据记录转换为json字符串;

[0180] 将所述json字符串按照结构化查询语言SQL的格式,生成所述数据记录的SQL命令;所述SQL命令用于备份所述数据记录。

[0181] 基于同一发明构思,本发明实施例还提供了一种计算机设备,包括程序或指令,当所述程序或指令被执行时,如本发明实施例提供的分布式数据库的分片信息备份方法及任一可选方法被执行。

[0182] 基于同一发明构思,本发明实施例还提供了一种计算机可读存储介质,包括程序或指令,当所述程序或指令被执行时,如本发明实施例提供的分布式数据库的分片信息备份方法及任一可选方法被执行。

[0183] 本领域内的技术人员应明白,本发明的实施例可提供为方法、或计算机程序产品。因此,本发明可采用完全硬件实施例、完全软件实施例、或结合软件和硬件方面的实施例的形式。而且,本发明可采用在一个或多个其中包含有计算机可用程序代码的计算机可用存储介质(包括但不限于磁盘存储器、CD-ROM、光学存储器等)上实施的计算机程序产品的形式。

[0184] 本发明是参照根据本发明实施例的方法、设备(系统)、和计算机程序产品的流程图和/或方框图来描述的。应理解可由计算机程序指令实现流程图和/或方框图中的每一流程和/或方框、以及流程图和/或方框图中的流程和/或方框的结合。可提供这些计算机程序指令到通用计算机、专用计算机、嵌入式处理机或其他可编程数据处理设备的处理器以产生一个机器,使得通过计算机或其他可编程数据处理设备的处理器执行的指令产生用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的装置。

[0185] 这些计算机程序指令也可存储在能引导计算机或其他可编程数据处理设备以特定方式工作的计算机可读存储器中,使得存储在该计算机可读存储器中的指令产生包括指令装置的制品,该指令装置实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能。

[0186] 这些计算机程序指令也可装载到计算机或其他可编程数据处理设备上,使得在计算机或其他可编程设备上执行一系列操作步骤以产生计算机实现的处理,从而在计算机或其他可编程设备上执行的指令提供用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的步骤。

[0187] 尽管已描述了本发明的优选实施例,但本领域内的技术人员一旦得知了基本创造性概念,则可对这些实施例作出另外的变更和修改。所以,所附权利要求意欲解释为包括优选实施例以及落入本发明范围的所有变更和修改。

[0188] 显然,本领域的技术人员可以对本发明进行各种改动和变型而不脱离本发明的精神和范围。这样,倘若本发明的这些修改和变型属于本发明权利要求及其等同技术的范围之内,则本发明也意图包含这些改动和变型在内。

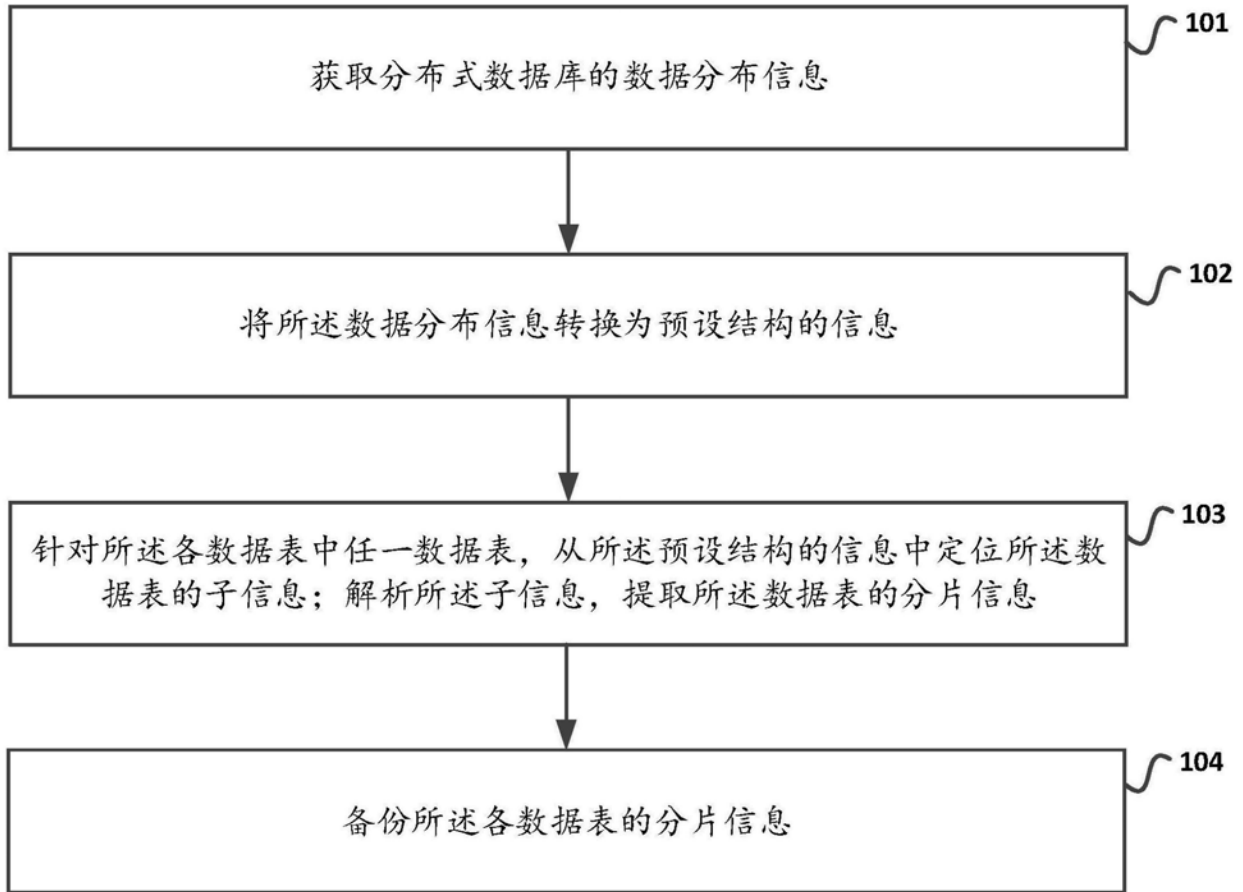


图1

```

1 #script 4.343s
2 --- Sharding Status ---
3- sharding version: {
4   -_id: NumberInt(1),
5   "minCompatibleVersion": NumberInt(5),
6   "currentVersion": NumberInt(6),
7   "clusterId": ObjectId("5fbc78883808aad352f1239c")
8 }
9 shards:
10 { "_id": "shard1_4_2", "host": "shard1_4_2/172.22.0.44:37821", "state": NumberInt(1) }
11 { "_id": "shard2_4_2", "host": "shard2_4_2/172.22.0.44:37822", "state": NumberInt(1) }
12 active mongoses:
13   "4.2.10": 2
14 autosplit:
15   Currently enabled: yes
16 balancer:
17   Currently enabled: yes
18   Currently running: no
19   Failed balancer rounds in last 5 attempts: 4
20   Last reported error: Could not find host matching read preference { mode: "primary" } for set shard1_4_2
21   Time of Reported error: Thu Nov 26 20:20 21:57:41 GMT+0800 (中国标准时间)
22   Migration Results for the last 24 hours:
23     No recent migrations
24 databases:
25 { "_id": "dpvcs", "primary": "shard1_4_2", "partitioned": true, "version": { "uuid": UUID("5147356c-8618-4e9f-95fa-611e994791b5"), "lastMod": NumberInt(1) } }
26   dpvcs.baCardCtrlMast
27     shard key: { "cardNo": 1, "ctrlCode": 1, "entityCode": 1 }
28     unique: false
29     balancing: true
30     chunks:
31       | shard1_4_2 1
32       | { "cardNo": MinKey, "ctrlCode": MinKey, "entityCode": MinKey } --> { "cardNo": MaxKey, "ctrlCode": MaxKey, "entityCode": MaxKey } on: shard1_4_2 Timestamp(1, 0)
33   dpvcs.baCardCustacct
34     shard key: { "cardNo": 1, "entityCode": 1 }
35     unique: false
36     balancing: true
37     chunks:
38       | shard1_4_2 1
39       | { "cardNo": MinKey, "entityCode": MinKey } --> { "cardNo": MaxKey, "entityCode": MaxKey } on: shard1_4_2 Timestamp(1, 0)
  
```

图2

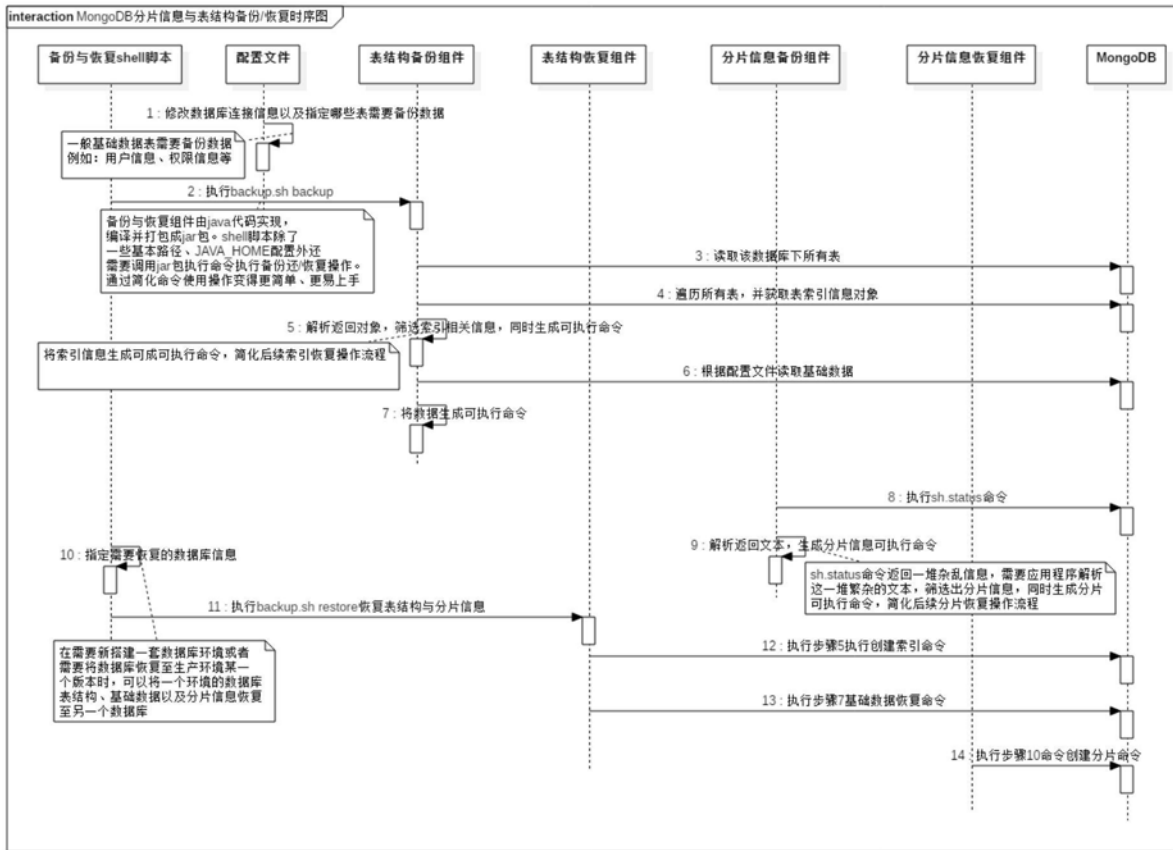


图3

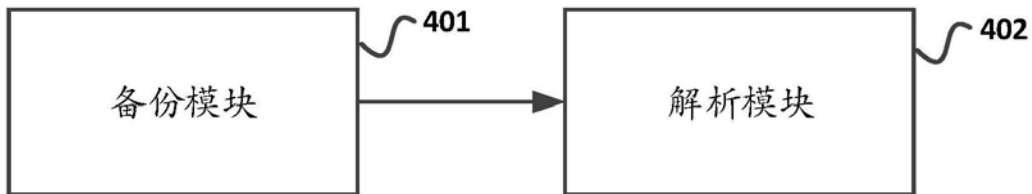


图4