



(12) 发明专利申请

(10) 申请公布号 CN 112100152 A

(43) 申请公布日 2020.12.18

(21) 申请号 202010959886.8

(22) 申请日 2020.09.14

(71) 申请人 广州华多网络科技有限公司

地址 511400 广东省广州市番禺区南村镇
万博二路79号万博商务区万达商业广
场北区B-1栋24层

(72) 发明人 陈智铭 高翔 郭荣洁 林锦培

(74) 专利代理机构 广州华进联合专利商标代理
有限公司 44224

代理人 黄晓庆

(51) Int.Cl.

G06F 16/182 (2019.01)

G06F 16/11 (2019.01)

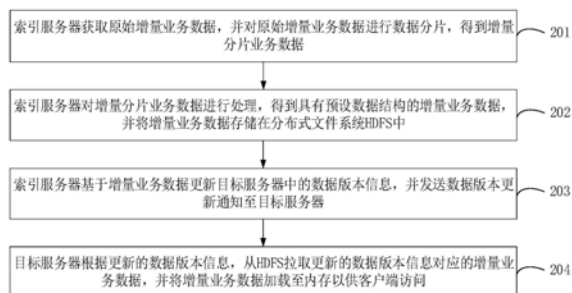
权利要求书3页 说明书23页 附图8页

(54) 发明名称

业务数据处理方法、系统、服务器和可读存
储介质

(57) 摘要

本申请涉及一种业务数据处理方法、系统、服务器和可读存储介质。方法包括：索引服务器获取原始增量业务数据，并对原始增量业务数据进行数据分片，得到增量分片业务数据；索引服务器对增量分片业务数据进行处理，得到具有预设数据结构的增量业务数据，并将增量业务数据存储在分布式文件系统HDFS中；索引服务器基于增量业务数据更新目标服务器中的数据版本信息，并发送数据版本更新通知至目标服务器；目标服务器根据更新的数据版本信息，从HDFS拉取更新的数据版本信息对应的增量业务数据，并将增量业务数据加载至内存以供客户端访问。采用本方法能够提升业务数据处理系统的自动化能力、可扩展性以及降低内存资源浪费。



1. 一种业务数据处理方法,其特征在于,应用于业务数据处理系统,所述业务数据处理系统包括索引服务器和多个业务服务器,所述方法包括:

所述索引服务器获取原始增量业务数据,并对所述原始增量业务数据进行数据分片,得到增量分片业务数据;

所述索引服务器对所述增量分片业务数据进行处理,得到具有预设数据结构的增量业务数据,并将所述增量业务数据存储于分布式文件系统HDFS中;

所述索引服务器基于所述增量业务数据更新目标服务器中的数据版本信息,并发送数据版本更新通知至所述目标服务器,所述目标服务器为所述多个业务服务器中的一个业务服务器;

所述目标服务器根据更新的数据版本信息,从所述HDFS拉取所述更新的数据版本信息对应的所述增量业务数据,并将所述增量业务数据加载至内存以供客户端访问。

2. 根据权利要求1所述的方法,其特征在于,所述方法还包括:

所述索引服务器根据所述增量业务数据的所述更新的数据版本信息以及所述增量业务数据在所述HDFS中的第一存储地址,生成第一数据基本信息;

对应地,所述目标服务器根据更新的数据版本信息,从所述HDFS拉取所述更新的数据版本信息对应的所述增量业务数据,包括:

所述目标服务器根据所述更新的数据版本信息,从所述索引服务器中获取所述第一数据基本信息;

所述目标服务器根据所述第一数据基本信息包括的所述第一存储地址,从所述第一存储地址指示的HDFS存储位置中拉取所述更新的数据版本信息对应的所述增量业务数据。

3. 根据权利要求1所述的方法,其特征在于,所述索引服务器对所述增量分片业务数据进行处理,得到具有预设数据结构的增量业务数据,包括:

所述索引服务器采用拉链法对所述增量分片业务数据进行处理,得到具有预设数据结构的所述增量业务数据。

4. 根据权利要求1所述的方法,其特征在于,所述业务数据处理系统还包括数据源服务器,所述方法还包括:

所述数据源服务器获取所述原始增量业务数据,将所述原始增量业务数据存储至所述HDFS,并将所述原始增量业务数据的第二数据基本信息发送至所述索引服务器,所述第二数据基本信息至少包括所述原始增量业务数据在所述HDFS中的第二存储地址;

对应地,所述索引服务器获取原始增量业务数据,包括:

所述索引服务器根据所述第二数据基本信息包括的所述第二存储地址,从所述第二存储地址指示的HDFS存储位置中拉取所述原始增量业务数据。

5. 根据权利要求1所述的方法,其特征在于,所述方法还包括:

所述索引服务器按照预设的数据格式,对所述增量分片业务数据进行解析,得到解析后的增量分片业务数据;

对应地,所述索引服务器对所述增量分片业务数据进行处理,得到具有预设数据结构的增量业务数据,包括:

所述索引服务器对所述解析后的增量分片业务数据进行处理,得到具有预设数据结构的所述增量业务数据。

6. 根据权利要求1所述的方法,其特征在于,所述方法还包括:

所述目标服务器获取全量数据版本信息,并根据所述全量数据版本信息,从所述HDFS拉取所述全量数据版本信息对应的全量业务数据,所述全量数据版本信息是所述索引服务器基于所述全量业务数据在所述目标服务器中更新的;

所述目标服务器将所述全量业务数据加载至内存以供客户端访问。

7. 根据权利要求6所述的方法,其特征在于,所述方法还包括:

所述目标服务器若检测到重启触发事件,则根据所述更新的数据版本信息从所述HDFS拉取所述增量业务数据,并根据所述全量数据版本信息从所述HDFS拉取所述全量业务数据;

所述目标服务器将拉取的所述增量业务数据和所述全量业务数据加载至内存以供客户端访问。

8. 根据权利要求6所述的方法,其特征在于,所述方法还包括:

所述索引服务器从所述HDFS获取全量分片业务数据,所述全量分片业务数据是对原始全量业务数据进行数据分片后得到的;

所述索引服务器采用开放定址法对所述全量分片业务数据进行处理,生成具有预设数据结构的所述全量业务数据;

所述索引服务器将所述全量业务数据存储至所述HDFS,并根据所述全量业务数据的全量数据版本信息和第三存储地址生成第三数据基本信息;

对应地,所述目标服务器根据所述全量数据版本信息,从所述HDFS拉取所述全量数据版本信息对应的全量业务数据,包括:

所述目标服务器根据所述全量数据版本信息,从所述索引服务器中获取所述第三数据基本信息;

所述目标服务器根据所述第三数据基本信息包括的所述第三存储地址,从所述第三存储地址指示的HDFS存储位置中拉取所述全量数据版本信息对应的所述全量业务数据。

9. 根据权利要求8所述的方法,其特征在于,所述方法还包括:

所述索引服务器获取所述原始全量业务数据的分片方式信息,并检测所述分片方式信息是否与本地分片方式信息相同,所述本地分片方式信息与所述多个业务服务器的数量有关;

若所述分片方式信息与所述本地分片方式信息相同,所述索引服务器则执行所述索引服务器从所述HDFS获取全量分片业务数据的步骤;

若所述分片方式信息与所述本地分片方式信息不同,所述索引服务器则根据所述本地分片方式信息对所述原始全量业务数据进行数据分片,得到所述全量分片业务数据,并执行所述索引服务器从所述HDFS获取全量分片业务数据的步骤。

10. 根据权利要求1所述的方法,其特征在于,所述目标服务器中包括第一分片数据副本和第二分片数据副本,所述第一分片数据副本和所述第二分片数据副本用于提供相同的数据访问服务,所述目标服务器将所述增量业务数据加载至内存,包括:

所述目标服务器根据所述增量业务数据对所述第一分片数据副本进行数据更新,在所述第一分片数据副本数据更新的过程中,所述目标服务器通过所述第二分片数据副本提供数据访问服务,并禁止数据更新过程中的第一分片数据副本提供数据访问服务;

在所述第一分片数据副本数据更新完成后,所述目标服务器根据所述增量业务数据对所述第二分片数据副本进行数据更新,在所述第二分片数据副本数据更新的过程中,所述目标服务器通过数据更新后的第一分片数据副本提供数据访问服务,并禁止数据更新过程中的第二分片数据副本提供数据访问服务。

11. 根据权利要求1所述的方法,其特征在于,所述方法还包括:

所述索引服务器若接收到所述客户端发送的数据地址获取请求,所述索引服务器则根据所述数据地址获取请求确定所述目标服务器的服务地址;

所述索引服务器将所述服务地址返回至所述客户端;

所述客户端根据所述服务地址,发送针对所述增量业务数据的访问请求至所述目标服务器;

所述目标服务器接收所述访问请求,并基于内存中加载的所述增量业务数据响应所述访问请求。

12. 根据权利要求11所述的方法,其特征在于,所述索引服务器则根据所述数据地址获取请求确定所述目标服务器的服务地址,包括:

所述索引服务器对所述数据地址获取请求进行解析,得到所述数据地址获取请求对应的关键码;

所述索引服务器根据所述关键码,从所述多个业务服务器中确定所述关键码对应的所述目标服务器的服务地址。

13. 一种业务数据处理系统,其特征在于,所述业务数据处理系统包括索引服务器和多个业务服务器,目标服务器为所述多个业务服务器中的一个业务服务器;

所述索引服务器,用于获取原始增量业务数据,并对所述原始增量业务数据进行数据分片,得到增量分片业务数据,对所述增量分片业务数据进行处理,得到具有预设数据结构的增量业务数据,并将所述增量业务数据存储在分布式文件系统HDFS中;

所述索引服务器,还用于基于所述增量业务数据更新目标服务器中的数据版本信息,并发送数据版本更新通知至所述目标服务器;

所述目标服务器,用于根据更新的数据版本信息,从所述HDFS拉取所述更新的数据版本信息对应的所述增量业务数据,并将所述增量业务数据加载至内存以供客户端访问。

14. 一种服务器,包括存储器和处理器,所述存储器存储有计算机程序,其特征在于,所述处理器执行所述计算机程序时实现权利要求1至12中任一项所述的方法的步骤。

15. 一种计算机可读存储介质,其上存储有计算机程序,其特征在于,所述计算机程序被处理器执行时实现权利要求1至12中任一项所述的方法的步骤。

业务数据处理方法、系统、服务器和可读存储介质

技术领域

[0001] 本申请涉及计算机技术领域,特别是涉及一种业务数据处理方法、系统、服务器和可读存储介质。

背景技术

[0002] 随着业务的不断发展,业务数据处理系统需要处理的业务数据也越来越多。现有的业务数据处理系统中,离线数据源的业务数据需要人工导入到Redis (Remote Dictionary server,远程字典服务)等存储引擎中才能提供线上服务。

[0003] 以Redis为例,Redis是基于内存的内存数据库,Redis的数据集中存储在内存中,为避免突然宕机导致的数据丢失,RDB (Redis DataBase,Redis默认的持久化方式)会周期性地把数据复制一份到内存,然后对复制的数据进行持久化。

[0004] 然而,上述业务数据处理系统的自动化能力有待提高,且内存数据库的可扩展性较差,而Redis存储引擎的持久化还会导致内存资源的浪费,不利于业务数据处理系统提供稳定的线上服务。

发明内容

[0005] 基于此,有必要针对上述技术问题,提供一种能够提升业务数据处理系统的自动化能力、可扩展性以及降低内存资源浪费的业务数据处理方法、系统、服务器和可读存储介质。

[0006] 第一方面,提供了一种业务数据处理方法,应用于业务数据处理系统,所述业务数据处理系统包括索引服务器和多个业务服务器,所述方法包括:

[0007] 所述索引服务器获取原始增量业务数据,并对所述原始增量业务数据进行数据分片,得到增量分片业务数据;

[0008] 所述索引服务器对所述增量分片业务数据进行处理,得到具有预设数据结构的增量业务数据,并将所述增量业务数据存储于分布式文件系统HDFS中;

[0009] 所述索引服务器基于所述增量业务数据更新目标服务器中的数据版本信息,并发送数据版本更新通知至所述目标服务器,所述目标服务器为所述多个业务服务器中的一个业务服务器;

[0010] 所述目标服务器根据更新的数据版本信息,从所述HDFS拉取所述更新的数据版本信息对应的所述增量业务数据,并将所述增量业务数据加载至内存以供客户端访问。

[0011] 在其中一个实施例中,所述方法还包括:

[0012] 所述索引服务器根据所述增量业务数据的所述更新的数据版本信息以及所述增量业务数据在所述HDFS中的第一存储地址,生成第一数据基本信息;

[0013] 对应地,所述目标服务器根据更新的数据版本信息,从所述HDFS拉取所述更新的数据版本信息对应的所述增量业务数据,包括:

[0014] 所述目标服务器根据所述更新的数据版本信息,从所述索引服务器中获取所述第

一数据基本信息；

[0015] 所述目标服务器根据所述第一数据基本信息包括的所述第一存储地址，从所述第一存储地址指示的HDFS存储位置中拉取所述更新的数据版本信息对应的所述增量业务数据。

[0016] 在其中一个实施例中，所述索引服务器对所述增量分片业务数据进行处理，得到具有预设数据结构的增量业务数据，包括：

[0017] 所述索引服务器采用拉链法对所述增量分片业务数据进行处理，得到具有预设数据结构的所述增量业务数据。

[0018] 在其中一个实施例中，所述业务数据处理系统还包括数据源服务器，所述方法还包括：

[0019] 所述数据源服务器获取所述原始增量业务数据，将所述原始增量业务数据存储至所述HDFS，并将所述原始增量业务数据的第二数据基本信息发送至所述索引服务器，所述第二数据基本信息至少包括所述原始增量业务数据在所述HDFS中的第二存储地址；

[0020] 对应地，所述索引服务器获取原始增量业务数据，包括：

[0021] 所述索引服务器根据所述第二数据基本信息包括的所述第二存储地址，从所述第二存储地址指示的HDFS存储位置中拉取所述原始增量业务数据。

[0022] 在其中一个实施例中，所述方法还包括：

[0023] 所述索引服务器按照预设的数据格式，对所述增量分片业务数据进行解析，得到解析后的增量分片业务数据；

[0024] 对应地，所述索引服务器对所述增量分片业务数据进行处理，得到具有预设数据结构的增量业务数据，包括：

[0025] 所述索引服务器对所述解析后的增量分片业务数据进行处理，得到具有预设数据结构的所述增量业务数据。

[0026] 在其中一个实施例中，所述方法还包括：

[0027] 所述目标服务器获取全量数据版本信息，并根据所述全量数据版本信息，从所述HDFS拉取所述全量数据版本信息对应的全量业务数据，所述全量数据版本信息是所述索引服务器基于所述全量业务数据在所述目标服务器中更新的；

[0028] 所述目标服务器将所述全量业务数据加载至内存以供客户端访问。

[0029] 在其中一个实施例中，所述方法还包括：

[0030] 所述目标服务器若检测到重启触发事件，则根据所述更新的数据版本信息从所述HDFS拉取所述增量业务数据，并根据所述全量数据版本信息从所述HDFS拉取所述全量业务数据；

[0031] 所述目标服务器将拉取的所述增量业务数据和所述全量业务数据加载至内存以供客户端访问。

[0032] 在其中一个实施例中，所述方法还包括：

[0033] 所述索引服务器从所述HDFS获取全量分片业务数据，所述全量分片业务数据是对原始全量业务数据进行数据分片后得到的；

[0034] 所述索引服务器采用开放定址法对所述全量分片业务数据进行处理，生成具有预设数据结构的所述全量业务数据；

[0035] 所述索引服务器将所述全量业务数据存储至所述HDFS,并根据所述全量业务数据的全量数据版本信息和第三存储地址生成第三数据基本信息;

[0036] 对应地,所述目标服务器根据所述全量数据版本信息,从所述HDFS拉取所述全量数据版本信息对应的全量业务数据,包括:

[0037] 所述目标服务器根据所述全量数据版本信息,从所述索引服务器中获取所述第三数据基本信息;

[0038] 所述目标服务器根据所述第三数据基本信息包括的所述第三存储地址,从所述第三存储地址指示的HDFS存储位置中拉取所述全量数据版本信息对应的所述全量业务数据。

[0039] 在其中一个实施例中,所述方法还包括:

[0040] 所述索引服务器获取所述原始全量业务数据的分片方式信息,并检测所述分片方式信息是否与本地分片方式信息相同,所述本地分片方式信息与所述多个业务服务器的数量有关;

[0041] 若所述分片方式信息与所述本地分片方式信息相同,所述索引服务器则执行所述索引服务器从所述HDFS获取全量分片业务数据的步骤;

[0042] 若所述分片方式信息与所述本地分片方式信息不同,所述索引服务器则根据所述本地分片方式信息对所述原始全量业务数据进行数据分片,得到所述全量分片业务数据,并执行所述索引服务器从所述HDFS获取全量分片业务数据的步骤。

[0043] 在其中一个实施例中,所述目标服务器中包括第一分片数据副本和第二分片数据副本,所述第一分片数据副本和所述第二分片数据副本用于提供相同的数据访问服务,所述目标服务器将所述增量业务数据加载至内存,包括:

[0044] 所述目标服务器根据所述增量业务数据对所述第一分片数据副本进行数据更新,在所述第一分片数据副本数据更新的过程中,所述目标服务器通过所述第二分片数据副本提供数据访问服务,并禁止数据更新过程中的第一分片数据副本提供数据访问服务;

[0045] 在所述第一分片数据副本数据更新完成后,所述目标服务器根据所述增量业务数据对所述第二分片数据副本进行数据更新,在所述第二分片数据副本数据更新的过程中,所述目标服务器通过数据更新后的第一分片数据副本提供数据访问服务,并禁止数据更新过程中的第二分片数据副本提供数据访问服务。

[0046] 在其中一个实施例中,所述方法还包括:

[0047] 所述索引服务器若接收到所述客户端发送的数据地址获取请求,所述索引服务器则根据所述数据地址获取请求确定所述目标服务器的服务地址;

[0048] 所述索引服务器将所述服务地址返回至所述客户端;

[0049] 所述客户端根据所述服务地址,发送针对所述增量业务数据的访问请求至所述目标服务器;

[0050] 所述目标服务器接收所述访问请求,并基于内存中加载的所述增量业务数据响应所述访问请求。

[0051] 在其中一个实施例中,所述索引服务器则根据所述数据地址获取请求确定所述目标服务器的服务地址,包括:

[0052] 所述索引服务器对所述数据地址获取请求进行解析,得到所述数据地址获取请求对应的关键词;

[0053] 所述索引服务器根据所述关键码,从所述多个业务服务器中确定所述关键码对应的所述目标服务器的服务地址。

[0054] 第二方面,提供了一种业务数据处理系统,所述业务数据处理系统包括索引服务器和多个业务服务器,目标服务器为所述多个业务服务器中的一个业务服务器;

[0055] 所述索引服务器,用于获取原始增量业务数据,并对所述原始增量业务数据进行数据分片,得到增量分片业务数据,对所述增量分片业务数据进行处理,得到具有预设数据结构的增量业务数据,并将所述增量业务数据存储在分布式文件系统HDFS中;

[0056] 所述索引服务器,还用于基于所述增量业务数据更新目标服务器中的数据版本信息,并发送数据版本更新通知至所述目标服务器;

[0057] 所述目标服务器,用于根据更新的数据版本信息,从所述HDFS拉取所述更新的数据版本信息对应的所述增量业务数据,并将所述增量业务数据加载至内存以供客户端访问。

[0058] 第三方面,提供了一种服务器,包括存储器和处理器,所述存储器存储有计算机程序,所述处理器执行所述计算机程序时实现如上述第一方面所述的业务数据处理方法。

[0059] 第四方面,提供了一种计算机可读存储介质,其上存储有计算机程序,所述计算机程序被处理器执行时如上述第一方面所述的业务数据处理方法。

[0060] 本申请实施例提供的技术方案带来的有益效果至少包括:

[0061] 本申请实施例业务数据处理系统中,索引服务器获取到原始增量业务数据后,对原始增量业务数据进行数据分片,得到增量分片业务数据,索引服务器对该增量分片业务数据进行处理,得到具有预设数据结构的增量业务数据,并将该增量业务数据存储在HDFS中,这样,实现了增量业务数据的自动存储更新,且通过数据分片,对分片后的增量分片业务数据进行处理后分布式存储在HDFS,可以实现对原始增量业务数据的分布式管理,即使业务数据急剧增加,通过数据分片也可以实现对业务数据的分布式管理,增强了业务数据处理系统的可扩展性;而后,索引服务器基于增量业务数据更新目标服务器中的数据版本信息,并发送数据版本更新通知至目标服务器,目标服务器根据更新的数据版本信息,从HDFS拉取更新的数据版本信息对应的增量业务数据,并将增量业务数据加载至内存以供客户端访问,这样,目标服务器根据更新的数据版本信息直接从HDFS自动拉取增量业务数据,并将拉取的增量业务数据加载至内存以供客户端访问,实现了增量业务数据的自动加载,提升了业务数据处理系统的自动化能力,这就避免了传统技术中,业务数据需要人工导入到内存中的Redis中才能提供线上访问导致的业务数据处理系统的自动化能力差的问题。另外,本申请实施例是将增量业务数据存储在HDFS中,由目标服务器自动拉取增量业务数据并加载至内存,而不需要将增量业务数据人工导入Redis中,因此也就不需要基于Redis的RDB实现持久化,从而避免了RDB需要将内存中的业务数据复制一份到内存,再对复制的数据进行持久化所造成的内存浪费,因此本申请实施例还降低了目标服务器的内存资源的浪费,从而有利于提供稳定的线上服务。

附图说明

[0062] 图1为一个实施例中业务数据处理方法的应用环境图;

[0063] 图2为一个实施例中业务数据处理方法的流程图;

- [0064] 图3为一种示例性的业务数据二分片过程示意图；
- [0065] 图4为一个实施例中索引服务器进行服务发现的流程图；
- [0066] 图5为一种示例性的索引服务器进行服务发现的示意图；
- [0067] 图6为一个实施例中业务数据处理方法的流程图；
- [0068] 图7为一个实施例中业务数据处理方法的流程图；
- [0069] 图8为一个实施例中索引服务器建库的流程图；
- [0070] 图9为一个实施例中服务扩缩容过程中索引服务器建库的流程图；
- [0071] 图10为一个实施例中目标服务器将增量业务数据加载至内存的流程图；
- [0072] 图11为一种示例性的分片数据示意图；
- [0073] 图12为一个实施例中业务数据处理方法的流程图；
- [0074] 图13为一种示例性的基于组件的业务数据处理系统示意图；
- [0075] 图14为一个实施例中服务器的内部结构图。

具体实施方式

[0076] 为了使本申请的目的、技术方案及优点更加清楚明白，以下结合附图及实施例，对本申请进行进一步详细说明。应当理解，此处描述的具体实施例仅仅用以解释本申请，并不用于限定本申请。

[0077] 下面，将对本申请实施例提供的业务数据处理方法所涉及到的实施环境进行简要说明。

[0078] 该实施环境包括业务数据处理系统，该业务数据处理系统可以包括如图1所示的数据源服务器101、索引服务器102、多个业务服务器和HDFS (Hadoop Distributed File System, Hadoop分布式文件系统) 104，需要说明的是，图1仅示例性地示出了目标服务器103，该目标服务器103为多个业务服务器中的一个。

[0079] 其中，数据源服务器101可以通过网络与索引服务器102以及与HDFS104进行通信，索引服务器102可以通过网络与数据源服务器101、与目标服务器103、与HDFS104以及与客户端105进行通信，目标服务器103可以通过网络与索引服务器102、与HDFS104以及与客户端105进行通信。

[0080] 其中，数据源服务器101和索引服务器102可以是一台服务器，也可以是由多台服务器组成的服务器集群，数据源服务器101、索引服务器102和目标服务器103可以是塔式服务器、机架服务器、刀片式服务器、高密度服务器、单路服务器、双路服务器或者多路服务器等，客户端105可以是个人计算机、笔记本电脑、媒体播放器、智能电视、智能手机、平板电脑和便携式可穿戴设备等，本申请实施例对此不作具体限定。

[0081] 在一个实施例中，如图2所示，提供了一种业务数据处理方法，以该方法应用于图1所示的业务数据处理系统为例进行说明，该业务数据处理系统可以包括索引服务器和多个业务服务器，目标服务器为多个业务服务器中的一个业务服务器。该方法包括步骤201、步骤202、步骤203以及步骤204：

[0082] 步骤201，索引服务器获取原始增量业务数据，并对原始增量业务数据进行数据分片，得到增量分片业务数据。

[0083] 在业务数据处理系统中，存在大量的离线业务数据，索引服务器对这些业务数据

进行相应处理,然后对处理后的业务数据进行建库,将处理后的业务数据分布式存储在HDFS(Hadoop Distributed File System,Hadoop分布式文件系统)中。目标服务器则可以直接从HDFS中拉取这些处理后的业务数据,并加载至内存以供客户端访问。

[0084] 随着时间的推移,业务数据处理系统会不断产生新增的业务数据,即原始增量业务数据,索引服务器获取原始增量业务数据,并对原始增量业务数据进行数据分片,得到增量分片业务数据。在一种可能的实施方式中,索引服务器可以通过map-reduce任务将原始增量业务数据分为多个分片,这样,各个分片经过索引服务器处理后可以分布式存储在HDFS中,实现了原始增量业务数据的分布式管理。

[0085] 参见图3,图3示出了一种示例性的业务数据二分片过程示意图。如图3所示,索引服务器通过map-reduce任务将原始增量业务数据分为两个分片,每个分片为一个增量分片业务数据。

[0086] 步骤202,索引服务器对增量分片业务数据进行处理,得到具有预设数据结构的增量业务数据,并将增量业务数据存储于分布式文件系统HDFS中。

[0087] 本申请实施例中,索引服务器对增量分片业务数据进行处理后再存储,将增量分片业务数据处理得到具有预设数据结构的增量业务数据,预设数据结构例如可以是散列表的形式,这样有利于增量业务数据的快速查找。

[0088] 在步骤202一种可能的实施方式中,索引服务器可以采用拉链法对增量分片业务数据进行处理,得到具有预设数据结构的增量业务数据。

[0089] 拉链法又叫链地址法,拉链法是把具有相同散列地址的关键字(同义词)值放在同一个单链表(同义词链表)中。索引服务器采用拉链法对增量分片业务数据进行哈希处理,生成增量分片业务数据对应的增量业务数据。采用拉链法生成的增量业务数据,增量业务数据支持更改插入,从而能够实现增量业务数据的数据自动更新。

[0090] 索引服务器对增量分片业务数据进行处理,得到具有预设数据结构的增量业务数据后,将该增量业务数据存储于分布式文件系统HDFS中,这样,实现了增量业务数据的自动存储更新,且通过数据分片,对分片后的增量分片业务数据进行处理后分布式存储在HDFS,可以实现对原始增量业务数据的分布式管理,即使业务数据急剧增加,通过数据分片也可以实现对业务数据的分布式管理,增强了业务数据处理系统的可扩展性。

[0091] 步骤203,索引服务器基于增量业务数据更新目标服务器中的数据版本信息,并发送数据版本更新通知至目标服务器。

[0092] 其中,目标服务器为多个业务服务器中的一个业务服务器。

[0093] 索引服务器将增量业务数据存储于HDFS中后,索引服务器将增量业务数据的数据版本信息更新至目标服务器,并发送数据版本更新通知至目标服务器,提示目标服务器根据该增量业务数据的数据版本信息从HDFS中拉取该增量业务数据。

[0094] 本申请实施例中,数据版本信息可以表征业务数据的版本信息,如增量业务数据的数据版本信息则可以是增量业务数据的版本号。

[0095] 步骤204,目标服务器根据更新的数据版本信息,从HDFS拉取更新的数据版本信息对应的增量业务数据,并将增量业务数据加载至内存以供客户端访问。

[0096] 目标服务器接收到索引服务器发送的数据版本更新通知后,则根据更新的数据版本信息(即索引服务器更新至目标服务器的增量业务数据的数据版本信息)从HDFS拉取该

更新的数据版本信息对应的增量业务数据。

[0097] 在一种可能的实施方式中,索引服务器将增量业务数据存储到HDFS中之后,索引服务器还可以根据增量业务数据的更新的数据版本信息以及增量业务数据在HDFS中的第一存储地址,生成第一数据基本信息,其中,该更新的数据版本信息用于目标服务器根据更新的数据版本信息获取第一数据基本信息,第一数据基本信息用于目标服务器根据第一存储地址指示的HDFS存储位置拉取增量业务数据。目标服务器可以通过执行如下步骤A11和步骤A12实现步骤204的过程:

[0098] 步骤A11,目标服务器根据更新的数据版本信息,从索引服务器中获取第一数据基本信息。

[0099] 步骤A12,目标服务器根据第一数据基本信息包括的第一存储地址,从第一存储地址指示的HDFS存储位置中拉取更新的数据版本信息对应的增量业务数据。

[0100] 这样,目标服务器根据更新的数据版本信息,从HDFS拉取该更新的数据版本信息对应的增量业务数据时,则可以根据该更新的数据版本信息,从索引服务器中获取第一数据基本信息,再从该第一数据基本信息包括的第一存储地址所指示的HDFS存储位置中,拉取更新的数据版本信息对应的增量业务数据。

[0101] 目标服务器从HDFS拉取到增量业务数据后,则自动将增量业务数据加载至目标服务器的内存,这样,实现了增量业务数据的自动加载,提升了业务数据处理系统的自动化能力。

[0102] 目标服务器若接收到客户端发送的针对该增量业务数据的访问请求,目标服务器则基于内存中的增量业务数据响应数据访问请求,供客户端访问加载至内存中的增量业务数据,这样,将离线的业务数据自动转化为线上服务,提升了业务数据处理的效率。

[0103] 可选地,本申请实施例目标服务器可以部署于kubernetes等容器集群中。以kubernetes为例,目标服务器作为kubernetes中的一个服务节点,通过启动目标服务器中部署于pod中的容器,向客户端提供数据访问服务,该容器在运行时,则可以向客户端提供增量业务数据的读、写等服务。

[0104] 可选地,索引服务器也可以部署于kubernetes中,索引服务器通过consul进行服务发现,consul是基于G0语言开发的开源工具,主要面向分布式,服务化的系统提供服务注册、服务发现和配置管理的功能。

[0105] 在一种可能的实施方式中,参见图4,索引服务器通过consul进行服务发现的过程可以包括图4所示的步骤401、步骤402、步骤403和步骤404:

[0106] 步骤401,索引服务器若接收到客户端发送的数据地址获取请求,索引服务器则根据数据地址获取请求确定目标服务器的服务地址。

[0107] 客户端向索引服务器发送数据地址获取请求以获取可以访问的业务服务器的地址,索引服务器通过consul从多个业务服务器中确定出用于向客户端提供服务的目标服务器。

[0108] 参见图5,图5为一种示例性的索引服务器进行服务发现的示意图。如图5所示,server可以是业务服务器,consul定期对各server进行健康检查,探测各server是否正常运行,将正常运行的server加入server list (服务器列表),将未正常运行的server从服务器列表中移除,确保服务器列表中的业务服务器可以正常提供服务。

[0109] 索引服务器从consul获取正常运行的server的服务地址,该服务地址可以是IP:Port的形式,索引服务器可以建立各个关键码和各个服务地址之间的映射关系,不同的关键码对应不同的服务地址,关键码可以是字符串的形式。

[0110] 在一种可能的实施方式中,索引服务器可以对数据地址获取请求进行解析,得到数据地址获取请求对应的关键码,索引服务器根据该关键码以及建立的映射关系,从多个业务服务器中确定该关键码对应的目标服务器的服务地址,以实现通过consul从多个业务服务器中确定出用于向客户端提供服务的目标服务器的过程。本申请实施例索引服务器通过consul进行服务发现,实现客户端和业务服务器的解耦,提升业务处理效率。

[0111] 步骤402,索引服务器将服务地址返回至客户端。

[0112] 步骤403,客户端根据服务地址,发送针对增量业务数据的访问请求至目标服务器。

[0113] 步骤404,目标服务器接收访问请求,并基于内存中加载的增量业务数据响应访问请求。

[0114] 索引服务器从多个业务服务器中确定出目标服务器的服务地址后,索引服务器将服务地址返回至客户端。客户端则向该服务地址发送针对增量业务数据的访问请求,目标服务器接收到该访问请求后,则基于内存中加载的增量业务数据响应数据访问请求,供客户端访问增量业务数据。作为一种实施方式,客户端可以通过http-proxy接口从目标服务器中访问业务数据。

[0115] 本申请实施例中,客户端可以发送包括关键码的数据地址获取请求,从索引服务器获取返回的目标服务器的服务地址,客户端即可基于该服务地址访问目标服务器中的增量业务数据,由此,实现了客户端的k-v (key-value) 查询访问。

[0116] 在其它实施例中,索引服务器在建库时也可以建立向量型数据库,从而客户端可以实现基于向量的业务数据查询访问,在此不做具体限制。

[0117] 本申请实施例业务数据处理系统中,索引服务器获取到原始增量业务数据后,对原始增量业务数据进行数据分片,得到增量分片业务数据,索引服务器对该增量分片业务数据进行处理,得到具有预设数据结构的增量业务数据,并将该增量业务数据存储于HDFS中,这样,实现了增量业务数据的自动存储,且通过数据分片,并对分片后的增量分片业务数据进行处理后分布式存储在HDFS,可以实现对原始增量业务数据的分布式管理,即使业务数据急剧增加,通过数据分片也可以实现对业务数据的分布式管理,增强了业务数据处理系统的可扩展性;而后,索引服务器基于增量业务数据更新目标服务器中的数据版本信息,并发送数据版本更新通知至目标服务器,目标服务器根据更新的数据版本信息,从HDFS拉取更新的数据版本信息对应的增量业务数据,并将增量业务数据加载至内存以供客户端访问,这样,目标服务器根据更新的数据版本信息直接从HDFS自动拉取增量业务数据,并将拉取的增量业务数据加载至内存以供客户端访问,实现了增量业务数据的自动加载,提升了业务数据处理系统的自动化能力,这就避免了传统技术中,业务数据需要人工导入到内存中的Redis中才能提供线上访问导致的业务数据处理系统的自动化能力差的问题。另外,本申请实施例是将增量业务数据存储于HDFS中,由目标服务器自动拉取增量业务数据并加载至内存,而不需要将增量业务数据人工导入Redis中,因此也就不需要基于Redis的RDB实现持久化,从而避免了RDB需要将内存中的业务数据复制一份到内存,再对复制的数据进行

持久化所造成的内存浪费,因此本申请实施例还降低了目标服务器的内存资源的浪费,从而有利于提供稳定的线上服务。

[0118] 在一个实施例中,基于上述图2所示的实施例,本实施例涉及的是索引服务器如何获取原始增量业务数据以及索引服务器如何对增量业务数据进行数据规范化的过程。本实施例业务数据处理系统还包括数据源服务器,如图6所示,本实施例提供的业务数据处理方法还包括步骤205:

[0119] 步骤205,数据源服务器获取原始增量业务数据,将原始增量业务数据存储至HDFS,并将原始增量业务数据的第二数据基本信息发送至索引服务器。

[0120] 其中,第二数据基本信息至少包括原始增量业务数据在HDFS中的第二存储地址。

[0121] 对应地,步骤201可以包括如图6所示的步骤2011:

[0122] 步骤2011,索引服务器根据第二数据基本信息包括的第二存储地址,从第二存储地址指示的HDFS存储位置中拉取原始增量业务数据,并对原始增量业务数据进行数据分片,得到增量分片业务数据。

[0123] 请继续参见图6,步骤2011之后,本实施例提供的业务数据处理方法还可以包括步骤206:

[0124] 步骤206,索引服务器按照预设的数据格式,对增量分片业务数据进行解析,得到解析后的增量分片业务数据。

[0125] 本申请实施例中,索引服务器对原始增量业务数据进行数据分片,得到增量分片业务数据之后,索引服务器可以按照预设的数据格式,对增量分片业务数据进行解析,得到解析后的增量分片业务数据。预设的数据格式例如可以是Protobuf格式。

[0126] 对应地,步骤202可以包括如图6所示的步骤2021:

[0127] 步骤2021,索引服务器对解析后的增量分片业务数据进行处理,得到具有预设数据结构的增量业务数据,并将增量业务数据存储于HDFS中。

[0128] 索引服务器可以采用拉链法对解析后的增量分片业务数据进行哈希处理以生成增量业务数据来实现步骤202的过程。这样,解析后的增量分片业务数据则均为统一的数据格式,使用方拿到的是结构化的数据,达到了规范化数据的目的。避免了传统技术中,由于数据源复杂,数据定义多种多样,造成的数据维护难度大的问题。本实施例降低了增量业务数据的维护难度。

[0129] 在一个实施例中,基于上述图2所示的实施例,本实施例涉及的是目标服务器如何基于全量业务数据提供客户端访问的过程,以及涉及在目标服务器重启的情况下,目标服务器如何恢复业务数据的过程。本申请实施例中,全量业务数据可以是历史业务数据,增量业务数据即为新增的业务数据。

[0130] 参见图7,目标服务器可以通过执行如图7所示的步骤701和步骤702实现基于全量业务数据提供客户端访问的过程:

[0131] 步骤701,目标服务器获取全量数据版本信息,并根据全量数据版本信息,从HDFS拉取全量数据版本信息对应的全量业务数据。

[0132] 其中,全量数据版本信息是索引服务器基于全量业务数据在目标服务器中更新的。

[0133] 本申请实施例中,全量业务数据可以是索引服务器对原始全量业务数据进行相应

的处理后得到的,如分片处理、哈希处理等。索引服务器对原始全量业务数据进行相应的处理后,将得到的全量业务数据存储至HDFS。

[0134] 索引服务器将全量业务数据的全量数据版本信息更新至目标服务器,并通知目标服务器拉取全量业务数据,目标服务器则根据该全量数据版本信息,从HDFS拉取全量数据版本信息对应的全量业务数据。

[0135] 步骤702,目标服务器将全量业务数据加载至内存以供客户端访问。

[0136] 目标服务器将全量业务数据加载至内存,客户端访问全量业务数据和/或增量业务数据时,目标服务器即可基于内存中加载的业务数据向客户端提供服务。由此,将离线的业务数据自动转化为线上服务,提升了业务处理的效率。

[0137] 请继续参见图7,目标服务器可以通过执行图7所示的步骤703和步骤704实现在目标服务器重启的情况下,恢复业务数据的过程:

[0138] 步骤703,目标服务器若检测到重启触发事件,则根据更新的数据版本信息从HDFS拉取增量业务数据,并根据全量数据版本信息从HDFS拉取全量业务数据。

[0139] 步骤704,目标服务器将拉取的增量业务数据和全量业务数据加载至内存以供客户端访问。

[0140] 本申请实施例中,若目标服务器发生意外宕机等状况导致目标服务器重启,目标服务器则重新从索引服务器获取全量数据版本信息和更新的数据版本信息,目标服务器根据更新的数据版本信息从HDFS拉取增量业务数据,并根据全量数据版本信息从HDFS拉取全量业务数据。

[0141] 目标服务器将拉取的增量业务数据和全量业务数据加载至内存,客户端访问全量业务数据和/或增量业务数据时,目标服务器即可基于内存中加载的业务数据向客户端提供服务。这样,即使目标服务器重启,也不会造成数据丢失。

[0142] 传统技术中,为了避免服务器重启时的业务数据丢失,一般是通过Redis的RDB进行数据持久化,在数据持久化过程中存在双倍内存占用的问题的,造成内存资源的浪费。而本申请实施例中,目标服务器重启后可以根据更新的数据版本信息和全量数据版本信息从HDFS拉取业务数据,从而不必占用目标服务器的额外内存进行数据持久化,节约了目标服务器的内存资源。

[0143] 在一个实施例中,基于上述图7所示的实施例,参见图8,本实施例涉及的是索引服务器对全量业务数据建库的过程。如图8所示,该过程可以包括步骤801、步骤802和步骤803:

[0144] 步骤801,索引服务器从HDFS获取全量分片业务数据。

[0145] 本申请实施例中,业务数据处理系统还包括数据源服务器,在一种可能的实施方式中,数据源服务器导出全量分片业务数据,该全量分片业务数据是对原始全量业务数据进行数据分片后得到的,数据源服务器将全量分片业务数据推到HDFS。

[0146] 在另一种可能的实施方式中,数据源服务器导出原始全量业务数据,数据源服务器可以将该原始全量业务数据推到HDFS,之后,由索引服务器对原始全量业务数据进行数据分片,得到全量分片业务数据,并将全量业务数据存储于HDFS。

[0147] 这样,索引服务器即可从HDFS获取全量分片业务数据。

[0148] 步骤802,索引服务器采用开放定址法对全量分片业务数据进行处理,生成具有预

设数据结构的全量业务数据。

[0149] 预设数据结构例如可以是散列表的形式,开放定址法即从发生冲突的那个单元开始,按照一定的次序,从散列表中查找出一个空闲的存储单元,把发生冲突的待插入元素存入到该单元中的一类处理冲突的方法。

[0150] 本申请实施例中,索引服务器采用开放定址法对全量分片业务数据进行哈希处理,生成全量分片业务数据对应的全量业务数据。采用开放定址法生成的全量业务数据紧凑,有利于减少全量业务数据的存储占用空间。

[0151] 步骤803,索引服务器将全量业务数据存储至HDFS,并根据全量业务数据的全量数据版本信息和第三存储地址生成第三数据基本信息。

[0152] 其中,全量数据版本信息用于目标服务器根据全量数据版本信息获取第三数据基本信息,第三数据基本信息用于目标服务器根据第三存储地址指示的HDFS存储位置拉取全量数据版本信息对应的全量业务数据,从而实现对全量业务数据的建库。

[0153] 在一种可能的实施方式中,步骤803之后,目标服务器可以通过执行如下步骤A21和步骤A22实现步骤701的过程:

[0154] 步骤A21,目标服务器根据全量数据版本信息,从索引服务器中获取第三数据基本信息。

[0155] 步骤A22,目标服务器根据第三数据基本信息包括的第三存储地址,从第三存储地址指示的HDFS存储位置中拉取全量数据版本信息对应的全量业务数据。

[0156] 本申请实施例中,索引服务器将全量业务数据存储至HDFS后,索引服务器根据全量业务数据的全量数据版本信息和第三存储地址生成第三数据基本信息,索引服务器更新目标服务器中的数据版本信息,通知目标服务器拉取数据。目标服务器获取全量数据版本信息,并根据全量数据版本信息,从索引服务器获取全量数据版本信息对应的第三数据基本信息,接着,目标服务器根据第三数据基本信息包括的第三存储地址,从第三存储地址指示的HDFS存储位置中拉取全量数据版本信息对应的全量业务数据。

[0157] 这样,通过将离线的全量业务数据在HDFS中建库,目标服务器从HDFS中自动拉取全量业务数据,并加载至内存提供数据访问服务,实现了离线的业务数据的自动化建库,达到了离线的业务数据自动转换为线上服务的效果。

[0158] 可选地,步骤801之后,本实施例业务数据处理方法还可以包括步骤A31:

[0159] 步骤A31,索引服务器按照预设的数据格式,对全量分片业务数据进行解析,得到解析后的全量分片业务数据。

[0160] 本申请实施例中,索引服务器从HDFS获取全量分片业务数据之后,索引服务器按照预设的数据格式,对全量分片业务数据进行解析,得到解析后的全量分片业务数据。预设的数据格式例如可以是Protobuf (Google Protocol Buffer)。

[0161] 对应地,索引服务器可以通过采用开放定址法对解析后的全量分片业务数据进行哈希处理,生成全量业务数据来实现步骤802的过程。这样,解析后的全量分片业务数据则均为统一的数据格式,使用方拿到的是结构化的数据,达到了规范化数据的目的。避免了传统技术中,由于数据源复杂,数据定义多种多样,造成的数据维护难度大的问题。本实施例降低了全量业务数据的维护难度。

[0162] 在一个实施例中,基于上述图8所示的实施例,参见图9,本实施例涉及的是在服务

扩缩容过程中,目标服务器如何对离线的业务数据建库的过程。如图9所示,该过程可以包括如下步骤:

[0163] 步骤901,索引服务器获取原始全量业务数据的分片方式信息,并检测分片方式信息是否与存储的本地分片方式信息相同。

[0164] 在实际的应用场景中,随着业务的变化,常常需要对业务服务器的数量进行扩容或缩容,发生扩容或者缩容后,业务数据处理系统中业务服务器的数量将会发生变化。

[0165] 考虑到上述扩缩容的情况,本申请实施例中,索引服务器在建库前,先获取原始全量业务数据的分片方式信息,检测分片方式信息是否与存储的本地分片方式信息相同,本地分片方式信息与多个业务服务器的数量有关,如本地分片方式信息可以是扩缩容后业务数据处理系统中的业务服务器的数量。

[0166] 以扩容场景为例,例如,索引服务器对原始全量业务数据是按2分片进行分片的,那么,索引服务器获取原始全量业务数据的分片方式信息即为2,扩容后业务数据处理系统中的业务服务器的数量为4,索引服务器则检测到2和4是否相同。

[0167] 若分片方式信息与本地分片方式信息相同,则表征原始全量业务数据的分片方式和扩缩容后的业务服务器的数量是匹配的,索引服务器则执行如上文的步骤801、步骤802和步骤803,对离线的业务数据进行建库。

[0168] 若分片方式信息与本地分片方式信息不同,则表征原始全量业务数据的分片方式是按未扩缩容之前的业务服务器的数量进行分片的,因此,需要对原始全量业务数据进行重新分片,以匹配扩缩容后的业务服务器的数量,索引服务器则执行步骤902:

[0169] 步骤902,索引服务器则根据本地分片方式信息对原始全量业务数据进行数据分片,得到全量分片业务数据。

[0170] 服务器按照本地分片方式信息对原始全量业务数据进行数据分片,即按照扩缩容后的机器数量对原始全量业务数据重新进行数据分片,得到全量分片业务数据,索引服务器将全量分片业务数据存储于HDFS。

[0171] 重新分片后,在步骤902的基础上,索引服务器则执行如上文的步骤801、步骤802和步骤803,对离线的业务数据进行建库。

[0172] 本申请实施例中,索引服务器在建库时能够根据结合扩缩容情况自动化建库,实现了离线的业务数据的自动化管理,更好满足实际业务对扩缩容的需求。

[0173] 在一个实施例中,在上述图2所示实施例的基础上,参见图10,本实施例涉及的是目标服务器将增量业务数据加载至内存的过程。本申请实施例中,目标服务器中包括第一分片数据副本和第二分片数据副本,第一分片数据副本和第二分片数据副本用于提供相同的数据访问服务。如图10所示,该过程可以包括步骤2041和步骤2042:

[0174] 步骤2041,目标服务器根据增量业务数据对第一分片数据副本进行数据更新,在第一分片数据副本数据更新的过程中,目标服务器通过第二分片数据副本提供数据访问服务,并禁止数据更新过程中的第一分片数据副本提供数据访问服务。

[0175] 本申请实施例中,每个节点多副本,节点即业务服务器。第一分片数据副本和第二分片数据副本可以是上一个版本的增量业务数据(假设称为历史增量业务数据)的两个副本,这两个副本均对外提供历史增量业务数据的访问服务。

[0176] 若历史增量业务数据更新,更新后的数据为本申请实施例的增量业务数据,目标

服务器获取增量业务数据后,则需要将增量业务数据加载至内存,以采用增量业务数据对历史增量业务数据进行更新。

[0177] 本申请实施例中,目标服务器会逐一更新分片数据副本,这样保证至少有一个分片数据副本能够提供服务。

[0178] 作为一种实施方式,目标服务器部署于Kubernetes,第一分片数据副本和第二分片数据副本通过不同的pod对外提供服务。目标服务器根据增量业务数据对第一分片数据副本进行数据更新时,目标服务器将第一分片数据副本对应的pod的状态从运行改为停止运行,以禁止数据更新过程中的第一分片数据副本提供数据访问服务。若有客户端的访问请求,目标服务器则通过第二分片数据副本对应的pod向客户端提供数据访问服务。

[0179] 步骤2042,在第一分片数据副本数据更新完成后,目标服务器根据增量业务数据对第二分片数据副本进行数据更新,在第二分片数据副本数据更新的过程中,目标服务器通过数据更新后的第一分片数据副本提供数据访问服务,并禁止数据更新过程中的第二分片数据副本提供数据访问服务。

[0180] 目标服务器根据增量业务数据对第一分片数据副本数据更新完成后,目标服务器根据增量业务数据对第二分片数据副本进行数据更新,

[0181] 同理,目标服务器将第二分片数据副本对应的pod的状态从运行改为停止运行,以禁止数据更新过程中的第二分片数据副本提供数据访问服务。若有客户端的访问请求,目标服务器则通过数据更新后的第一分片数据副本对应的pod向客户端提供数据访问服务。由此,在加载至内存的过程中不会对线上业务造成影响,确保了线上服务的服务质量,避免了传统技术中对业务数据进行更新的同时对外提供服务,导致的服务不稳定的问题。

[0182] 参见图11,图11为一种示例性的分片数据示意图。如图11所示,分片一和分片二可以是不同的业务分片数据,例如分片一和分片二可以是不同的增量业务数据。分片一被第一业务服务器加载至第一业务服务器的内存中,分片二被第二业务服务器加载至第二业务服务器的内存中,pod:0和pod:1为分片一的两个分片数据副本,pod:2和pod:3为分片二的两个分片数据副本。

[0183] 在数据更新过程中,第一业务服务器更新pod:0且通过pod:1提供服务,第二业务服务器更新pod:2且通过pod:3提供服务,pod:1和pod:3的时间戳一致,即数据版本信息一致,例如均为20200401。

[0184] pod:0和pod:2更新完成后,pod:0和pod:2的数据版本信息均更新为最新的版本信息,例如更新为20200402,第一业务服务器更新pod:1且第二业务服务器更新pod:3,这样,更新后的pod:0和pod:2提供服务。

[0185] 由此,对于同一批次的数据访问,业务数据处理系统基于版本控制,各个被访问的业务服务器均提供相同时间戳(数据版本信息)的分片数据副本供客户端访问,从而保证同一批次的数据访问,访问的是同一个版本的业务数据,确保数据访问的一致性。

[0186] 在一个实施例中,参见图12,提供一种业务数据处理方法的流程图,该方法可以应用于图1所示实施环境中。如图12所示,该方法可以包括以下步骤:

[0187] 步骤1001,数据源服务器导出全量分片业务数据,数据源服务器将全量分片业务数据推到HDFS。

[0188] 其中,该全量分片业务数据是对原始全量业务数据进行数据分片后得到的。

[0189] 步骤1002,数据源服务器获取原始增量业务数据,将原始增量业务数据存储至HDFS,并将原始增量业务数据的第二数据基本信息发送至索引服务器。

[0190] 其中,第二数据基本信息至少包括原始增量业务数据的第二存储地址;数据源服务器可以不断获取原始增量业务数据。

[0191] 步骤1003,索引服务器根据第二数据基本信息包括的第二存储地址,从第二存储地址指示的HDFS存储位置中拉取原始增量业务数据,并对原始增量业务数据进行数据分片,得到增量分片业务数据。

[0192] 步骤1004,索引服务器获取原始全量业务数据和原始增量业务数据的分片方式信息,并检测分片方式信息是否与本地分片方式信息相同。

[0193] 步骤1005,若相同,索引服务器则按照预设的数据格式,对全量分片业务数据进行解析,得到解析后的全量分片业务数据。

[0194] 步骤1006,索引服务器采用开放定址法对解析后的全量分片业务数据进行处理,生成具有预设数据结构的全量业务数据。

[0195] 步骤1007,索引服务器将全量业务数据存储至HDFS,并根据全量业务数据的全量数据版本信息和第三存储地址生成第三数据基本信息,并通知目标服务器拉数据。

[0196] 步骤1008,目标服务器获取全量数据版本信息,并根据全量数据版本信息,从索引服务器获取全量数据版本信息对应的第三数据基本信息。

[0197] 步骤1009,目标服务器根据第三数据基本信息包括的第三存储地址,从第三存储地址指示的HDFS存储位置中拉取全量数据版本信息对应的全量业务数据。

[0198] 步骤1010,索引服务器按照预设的数据格式,对增量分片业务数据进行解析,得到解析后的增量分片业务数据。

[0199] 步骤1011,索引服务器采用拉链法对解析后的增量分片业务数据进行处理,生成具有预设数据结构的全量业务数据。

[0200] 步骤1012,索引服务器将增量业务数据存储至HDFS,并根据增量业务数据更新的数据版本信息以及增量业务数据在HDFS中的第一存储地址,生成第一数据基本信息。

[0201] 步骤1013,索引服务器基于增量业务数据更新目标服务器中的数据版本信息,并发送数据版本更新通知至目标服务器。

[0202] 步骤1014,目标服务器若接收到索引服务器发送的数据版本更新通知,则根据更新的数据版本信息,从索引服务器中获取第一数据基本信息。

[0203] 步骤1015,目标服务器根据第一数据基本信息包括的第一存储地址,从第一存储地址指示的HDFS存储位置中拉取更新的数据版本信息对应的增量业务数据。

[0204] 步骤1016,目标服务器将全量业务数据和增量业务数据加载至内存以供客户端访问。

[0205] 目标服务器将增量业务数据加载至内存的过程中,目标服务器根据增量业务数据对第一分片数据副本进行数据更新,在第一分片数据副本数据更新的过程中,目标服务器通过第二分片数据副本提供数据访问服务,并禁止数据更新过程中的第一分片数据副本提供数据访问服务;目标服务器中包括第一分片数据副本和第二分片数据副本,第一分片数据副本和第二分片数据副本用于提供相同的数据访问服务。

[0206] 在第一分片数据副本数据更新完成后,目标服务器根据增量业务数据对第二分片

数据副本进行数据更新,在第二分片数据副本数据更新的过程中,目标服务器通过数据更新后的第一分片数据副本提供数据访问服务,并禁止数据更新过程中的第二分片数据副本提供数据访问服务。

[0207] 步骤1017,索引服务器若接收到客户端发送的数据地址获取请求,索引服务器对数据地址获取请求进行解析,得到数据地址获取请求对应的关键码,索引服务器根据关键码,从多个业务服务器中确定关键码对应的目标服务器的服务地址,索引服务器将服务地址返回至客户端。

[0208] 步骤1018,客户端根据服务地址,发送业务数据的访问请求至目标服务器。

[0209] 步骤1019,目标服务器若接收到客户端发送的业务数据的访问请求,则基于内存中的全量业务数据和/或增量业务数据响应数据访问请求。

[0210] 步骤1020,目标服务器若检测到重启触发事件,则根据更新的数据版本信息从HDFS拉取增量业务数据,并根据全量数据版本信息从HDFS拉取全量业务数据。

[0211] 步骤1021,目标服务器将拉取的增量业务数据和全量业务数据加载至内存以供客户端访问。

[0212] 本申请实施例中,作为一种实施方式,索引服务器index server可以包括增量数据分片delta sharding组件、更新peta update组件,索引服务器基于该多个组件执行索引服务器的步骤。目标服务器server可以包括consul组件以及controller组件,目标服务器基于该多个组件执行目标服务器的步骤。

[0213] 索引服务器、目标服务器、数据源服务器、HDFS以及各个组件之间可以通过接口访问。参见图13,图13为一种示例性的基于组件的业务数据处理系统示意图,客户端通过peta client访问server。索引服务器、目标服务器、数据源服务器、HDFS以及各个组件之间基于接口的数据交互过程如下:

[0214] 1) 数据源服务器ts导出全量分片业务数据base,ts将全量分片业务数据推到HDFS。

[0215] 2) ts获取原始增量业务数据delta,将原始增量业务数据存储至HDFS,并调用index server的add_delta_file接口将原始增量业务数据的第二数据基本信息发送至索引服务器。

[0216] 其中,第二数据基本信息至少包括原始增量业务数据的第二存储地址,ts可以不断获取原始增量业务数据。

[0217] 3) delta sharding定时访问index server,通过get_delta_sharding_file接口及set_delta_sharding_file_working接口获取第二数据基本信息,跑Map/Reduce任务对原始增量业务数据进行数据分片,分片后得到增量分片业务数据,delta sharding将增量分片业务数据存储至HDFS中,并调用add_delta_sharded_file将增量分片业务数据的文件信息推送至index server,增量分片业务数据的文件信息包括增量分片业务数据的数据版本信息和存储路径信息。

[0218] 4) peta update在建库前,调用index server的add_and_get_shard_count接口获取原始全量业务数据和原始增量业务数据的分片方式信息,并检测分片方式信息是否与本地分片方式信息相同。

[0219] 5) 若相同,peta update则按照预设的数据格式,对全量分片业务数据进行解析,

得到解析后的全量分片业务数据。peta update采用开放定址法对解析后的全量分片业务数据进行处理,生成具有预设数据结构的全量业务数据。peta update将全量业务数据存储至HDFS,并根据全量业务数据的全量数据版本信息和第三存储地址生成第三数据基本信息。

[0220] 6) peta update按照预设的数据格式,对增量分片业务数据进行解析,得到解析后的增量分片业务数据。peta update采用拉链法对解析后的增量分片业务数据进行处理,生成具有预设数据结构的增量业务数据。peta update将增量业务数据存储至HDFS,并根据增量业务数据的更新的数据版本信息和第一存储地址生成第一数据基本信息。

[0221] 7) peta update更新consul上的数据版本信息,并通知controller拉数据。

[0222] 8) controller调用index server的get_base_and_delta_file_list获取全量数据版本信息,并根据全量数据版本信息,从索引服务器获取全量数据版本信息对应的第三数据基本信息。controller根据第三数据基本信息包括的第三存储地址,从第三存储地址指示的HDFS存储位置中拉取全量数据版本信息对应的全量业务数据。

[0223] 9) controller调用index server的get_base_and_delta_file_list获取更新的数据版本信息。controller根据更新的数据版本信息,从索引服务器获取更新的数据版本信息对应的第一数据基本信息。controller根据第一数据基本信息包括的第一存储地址,从第一存储地址指示的HDFS存储位置中拉取更新的数据版本信息对应的增量业务数据。

[0224] 10) controller触发server加载数据,server将全量业务数据和增量业务数据加载至内存以供客户端访问。

[0225] server将增量业务数据加载至内存的过程中,server根据增量业务数据对第一分片数据副本进行数据更新,在第一分片数据副本数据更新的过程中,server通过第二分片数据副本提供数据访问服务,并禁止数据更新过程中的第一分片数据副本提供数据访问服务;server中包括第一分片数据副本和第二分片数据副本,第一分片数据副本和第二分片数据副本用于提供相同的数据访问服务。

[0226] 在第一分片数据副本数据更新完成后,server根据增量业务数据对第二分片数据副本进行数据更新,在第二分片数据副本数据更新的过程中,server通过数据更新后的第一分片数据副本提供数据访问服务,并禁止数据更新过程中的第二分片数据副本提供数据访问服务。

[0227] 11) index server若接收到客户端发送的数据地址获取请求,index server对数据地址获取请求进行解析,得到数据地址获取请求对应的关键码,index server根据关键码,从多个业务服务器中确定关键码对应的目标服务器的服务地址。

[0228] 12) index server将服务地址返回至客户端。

[0229] 13) 客户端根据服务地址,发送针对增量业务数据的访问请求至server。

[0230] 14) server若接收到客户端发送的业务数据的访问请求,则基于内存中的全量业务数据和/或增量业务数据响应数据访问请求。

[0231] 15) controller若检测到重启触发事件,则根据更新的数据版本信息从HDFS拉取增量业务数据,并根据全量数据版本信息从HDFS拉取全量业务数据。

[0232] 16) controller触发server重新加载数据,server将拉取的增量业务数据和全量业务数据加载至内存以供客户端访问。

[0233] 应该理解的是,虽然上述流程图中的各个步骤按照箭头的指示依次显示,但是这些步骤并不是必然按照箭头指示的顺序依次执行。除非本文中有明确的说明,这些步骤的执行并没有严格的顺序限制,这些步骤可以以其它的顺序执行。而且,上述流程图中的至少一部分步骤可以包括多个步骤或者多个阶段,这些步骤或者阶段并不必然是在同一时刻执行完成,而是可以在不同的时刻执行,这些步骤或者阶段的执行顺序也不必然是依次进行,而是可以与其它步骤或者其它步骤中的步骤或者阶段的至少一部分轮流或者交替地执行。

[0234] 在一个实施例中,提供了一种业务数据处理系统,所述业务数据处理系统包括索引服务器和多个业务服务器,目标服务器为所述多个业务服务器中的一个业务服务器;

[0235] 所述索引服务器,用于获取原始增量业务数据,并对所述原始增量业务数据进行数据分片,得到增量分片业务数据,对所述增量分片业务数据进行处理,得到具有预设数据结构的增量业务数据,并将所述增量业务数据存储于分布式文件系统HDFS中;

[0236] 所述索引服务器,还用于基于所述增量业务数据更新目标服务器中的数据版本信息,并发送数据版本更新通知至所述目标服务器;

[0237] 所述目标服务器,用于根据更新的数据版本信息,从所述HDFS拉取所述更新的数据版本信息对应的所述增量业务数据,并将所述增量业务数据加载至内存以供客户端访问。

[0238] 在一个实施例中,所述索引服务器还用于根据所述增量业务数据的所述更新的数据版本信息以及所述增量业务数据在所述HDFS中的第一存储地址,生成第一数据基本信息;

[0239] 对应地,所述目标服务器具体用于根据所述更新的数据版本信息,从所述索引服务器中获取所述第一数据基本信息;根据所述第一数据基本信息包括的所述第一存储地址,从所述第一存储地址指示的HDFS存储位置中拉取所述更新的数据版本信息对应的所述增量业务数据。

[0240] 在一个实施例中,所述索引服务器具体用于采用拉链法对所述增量分片业务数据进行处理,得到具有预设数据结构的所述增量业务数据。

[0241] 在一个实施例中,所述业务数据处理系统还包括数据源服务器,所述数据源服务器,用于获取所述原始增量业务数据,将所述原始增量业务数据存储至所述HDFS,并将所述原始增量业务数据的第二数据基本信息发送至所述索引服务器,所述第二数据基本信息至少包括所述原始增量业务数据在所述HDFS中的第二存储地址;

[0242] 对应地,所述索引服务器具体用于根据所述第二数据基本信息包括的所述第二存储地址,从所述第二存储地址指示的HDFS存储位置中拉取所述原始增量业务数据。

[0243] 在一个实施例中,所述索引服务器,具体用于按照预设的数据格式,对所述增量分片业务数据进行解析,得到解析后的增量分片业务数据;对所述解析后的增量分片业务数据进行处理,得到具有预设数据结构的所述增量业务数据。

[0244] 在一个实施例中,所述目标服务器,还用于获取全量数据版本信息,并根据所述全量数据版本信息,从所述HDFS拉取所述全量数据版本信息对应的全量业务数据,所述全量数据版本信息是所述索引服务器基于所述全量业务数据在所述目标服务器中更新的;所述目标服务器还用于将所述全量业务数据加载至内存以供客户端访问。

[0245] 在一个实施例中,所述目标服务器还用于若检测到重启触发事件,则根据所述更

新的数据版本信息从所述HDFS拉取所述增量业务数据,并根据所述全量数据版本信息从所述HDFS拉取所述全量业务数据;将拉取的所述增量业务数据和所述全量业务数据加载至内存以供客户端访问。

[0246] 在一个实施例中,所述索引服务器,还用于从所述HDFS获取全量分片业务数据,所述全量分片业务数据是对原始全量业务数据进行数据分片后得到的;采用开放定址法对所述全量分片业务数据进行处理,生成具有预设数据结构的所述全量业务数据;将所述全量业务数据存储至所述HDFS,并根据所述全量业务数据的全量数据版本信息和第三存储地址生成第三数据基本信息;

[0247] 对应地,所述目标服务器具体用于根据所述全量数据版本信息,从所述索引服务器中获取所述第三数据基本信息;根据所述第三数据基本信息包括的所述第三存储地址,从所述第三存储地址指示的HDFS存储位置中拉取所述全量数据版本信息对应的所述全量业务数据。

[0248] 在一个实施例中,所述索引服务器还用于获取所述原始全量业务数据的分片方式信息,并检测所述分片方式信息是否与本地分片方式信息相同,所述本地分片方式信息与所述多个业务服务器的数量有关;若所述分片方式信息与所述本地分片方式信息相同,所述索引服务器则执行所述索引服务器从所述HDFS获取全量分片业务数据的步骤;若所述分片方式信息与所述本地分片方式信息不同,所述索引服务器则根据所述本地分片方式信息对所述原始全量业务数据进行数据分片,得到所述全量分片业务数据,并执行所述索引服务器从所述HDFS获取全量分片业务数据的步骤。

[0249] 在一个实施例中,所述目标服务器中包括第一分片数据副本和第二分片数据副本,所述第一分片数据副本和所述第二分片数据副本用于提供相同的数据访问服务,所述目标服务器具体用于根据所述增量业务数据对所述第一分片数据副本进行数据更新,在所述第一分片数据副本数据更新的过程中,所述目标服务器通过所述第二分片数据副本提供数据访问服务,并禁止数据更新过程中的第一分片数据副本提供数据访问服务;在所述第一分片数据副本数据更新完成后,所述目标服务器根据所述增量业务数据对所述第二分片数据副本进行数据更新,在所述第二分片数据副本数据更新的过程中,所述目标服务器通过数据更新后的第一分片数据副本提供数据访问服务,并禁止数据更新过程中的第二分片数据副本提供数据访问服务。

[0250] 在一个实施例中,所述索引服务器还用于若接收到所述客户端发送的数据地址获取请求,所述索引服务器则根据所述数据地址获取请求确定所述目标服务器的服务地址;所述索引服务器将所述服务地址返回至所述客户端;所述客户端根据所述服务地址,发送针对所述增量业务数据的访问请求至所述目标服务器;

[0251] 所述目标服务器还用于接收所述访问请求,并基于内存中加载的所述增量业务数据响应所述访问请求。

[0252] 在一个实施例中,所述索引服务器具体用于对所述数据地址获取请求进行解析,得到所述数据地址获取请求对应的关键码;根据所述关键码,从所述多个业务服务器中确定所述关键码对应的所述目标服务器的服务地址。

[0253] 关于业务数据处理系统的具体限定可以参见上文中对于业务数据处理方法的限定,在此不再赘述。上述业务数据处理系统中的各个模块可全部或部分通过软件、硬件及其

组合来实现。上述各模块可以硬件形式内嵌于或独立于服务器中的处理器中,也可以以软件形式存储于服务器中的存储器中,以便于处理器调用执行以上各个模块对应的操作。

[0254] 在一个实施例中,提供了一种服务器,其内部结构图可以如图14所示。该服务器包括通过系统总线连接的处理器、存储器和网络接口。其中,该服务器的处理器用于提供计算和控制能力。该服务器的存储器包括非易失性存储介质、内存储器。该非易失性存储介质存储有操作系统、计算机程序和数据库。该内存储器为非易失性存储介质中的操作系统和计算机程序的运行提供环境。该服务器的数据库用于存储业务数据处理数据。该服务器的网络接口用于与外部的终端通过网络连接通信。该计算机程序被处理器执行时以实现一种业务数据处理方法。

[0255] 本领域技术人员可以理解,图14中示出的结构,仅仅是与本申请方案相关的部分结构的框图,并不构成对本申请方案所应用于其上的服务器的限定,具体的服务器可以包括比图中所示更多或更少的部件,或者组合某些部件,或者具有不同的部件布置。

[0256] 在一个实施例中,提供了一种服务器,包括存储器和处理器,存储器中存储有计算机程序,该处理器执行计算机程序时实现以下步骤:

[0257] 获取原始增量业务数据,并对所述原始增量业务数据进行数据分片,得到增量分片业务数据;

[0258] 对所述增量分片业务数据进行处理,得到具有预设数据结构的增量业务数据,并将所述增量业务数据存储在分布式文件系统HDFS中;

[0259] 基于所述增量业务数据更新目标服务器中的数据版本信息,并发送数据版本更新通知至所述目标服务器;

[0260] 其中,所述目标服务器为所述多个业务服务器中的一个业务服务器,所述数据版本更新通知用于通知所述目标服务器,根据更新的数据版本信息从所述HDFS拉取所述更新的数据版本信息对应的所述增量业务数据,并将所述增量业务数据加载至内存以供客户端访问。

[0261] 在一个实施例中,该处理器执行计算机程序时还实现以下步骤:

[0262] 根据所述增量业务数据的所述更新的数据版本信息以及所述增量业务数据在所述HDFS中的第一存储地址,生成第一数据基本信息;其中,所述第一数据基本信息用于所述目标服务器根据所述更新的数据版本信息,从所述索引服务器中获取所述第一数据基本信息;所述第一数据基本信息用于所述目标服务器根据所述第一数据基本信息包括的所述第一存储地址,从所述第一存储地址指示的HDFS存储位置中拉取所述更新的数据版本信息对应的所述增量业务数据。

[0263] 在一个实施例中,该处理器执行计算机程序时还实现以下步骤:

[0264] 采用拉链法对所述增量分片业务数据进行处理,得到具有预设数据结构的所述增量业务数据。

[0265] 在一个实施例中,所述业务数据处理系统还包括数据源服务器,所述数据源服务器用于获取所述原始增量业务数据,将所述原始增量业务数据存储至所述HDFS,并将所述原始增量业务数据的第二数据基本信息发送至所述索引服务器,所述第二数据基本信息至少包括所述原始增量业务数据在所述HDFS中的第二存储地址;

[0266] 对应地,该处理器执行计算机程序时还实现以下步骤:

[0267] 根据所述第二数据基本信息包括的所述第二存储地址,从所述第二存储地址指示的HDFS存储位置中拉取所述原始增量业务数据。

[0268] 在一个实施例中,该处理器执行计算机程序时还实现以下步骤:

[0269] 按照预设的数据格式,对所述增量分片业务数据进行解析,得到解析后的增量分片业务数据;

[0270] 对所述解析后的增量分片业务数据进行处理,得到具有预设数据结构的所述增量业务数据。

[0271] 在一个实施例中,所述目标服务器用于获取全量数据版本信息,并根据所述全量数据版本信息,从所述HDFS拉取所述全量数据版本信息对应的全量业务数据,所述全量数据版本信息是所述索引服务器基于所述全量业务数据在所述目标服务器中更新的;所述目标服务器将所述全量业务数据加载至内存以供客户端访问。

[0272] 在一个实施例中,所述目标服务器用于若检测到重启触发事件,则根据所述更新的数据版本信息从所述HDFS拉取所述增量业务数据,并根据所述全量数据版本信息从所述HDFS拉取所述全量业务数据;

[0273] 所述目标服务器将拉取的所述增量业务数据和所述全量业务数据加载至内存以供客户端访问。

[0274] 在一个实施例中,该处理器执行计算机程序时还实现以下步骤:

[0275] 从所述HDFS获取全量分片业务数据,所述全量分片业务数据是对原始全量业务数据进行数据分片后得到的;

[0276] 采用开放定址法对所述全量分片业务数据进行处理,生成具有预设数据结构的所述全量业务数据;

[0277] 将所述全量业务数据存储至所述HDFS,并根据所述全量业务数据的全量数据版本信息和第三存储地址生成第三数据基本信息;

[0278] 其中,所述第三数据基本信息用于所述目标服务器根据所述全量数据版本信息,从所述索引服务器中获取所述第三数据基本信息;所述目标服务器根据所述第三数据基本信息包括的所述第三存储地址,从所述第三存储地址指示的HDFS存储位置中拉取所述全量数据版本信息对应的所述全量业务数据。

[0279] 在一个实施例中,该处理器执行计算机程序时还实现以下步骤:

[0280] 获取所述原始全量业务数据的分片方式信息,并检测所述分片方式信息是否与本地分片方式信息相同,所述本地分片方式信息与所述多个业务服务器的数量有关;

[0281] 若所述分片方式信息与所述本地分片方式信息相同,则执行所述索引服务器从所述HDFS获取全量分片业务数据的步骤;

[0282] 若所述分片方式信息与所述本地分片方式信息不同,则根据所述本地分片方式信息对所述原始全量业务数据进行数据分片,得到所述全量分片业务数据,并执行所述索引服务器从所述HDFS获取全量分片业务数据的步骤。

[0283] 在一个实施例中,所述目标服务器中包括第一分片数据副本和第二分片数据副本,所述第一分片数据副本和所述第二分片数据副本用于提供相同的数据访问服务,所述目标服务器具体用于根据所述增量业务数据对所述第一分片数据副本进行数据更新,在所述第一分片数据副本数据更新的过程中,所述目标服务器通过所述第二分片数据副本提供

数据访问服务,并禁止数据更新过程中的第一分片数据副本提供数据访问服务;在所述第一分片数据副本数据更新完成后,所述目标服务器根据所述增量业务数据对所述第二分片数据副本进行数据更新,在所述第二分片数据副本数据更新的过程中,所述目标服务器通过数据更新后的第一分片数据副本提供数据访问服务,并禁止数据更新过程中的第二分片数据副本提供数据访问服务。

[0284] 在一个实施例中,该处理器执行计算机程序时还实现以下步骤:

[0285] 若接收到所述客户端发送的数据地址获取请求,则根据所述数据地址获取请求确定所述目标服务器的服务地址;

[0286] 将所述服务地址返回至所述客户端;所述服务地址用于所述客户端根据所述服务地址,发送针对所述增量业务数据的访问请求至所述目标服务器;所述目标服务器接收所述访问请求,并基于内存中加载的所述增量业务数据响应所述访问请求。

[0287] 在一个实施例中,该处理器执行计算机程序时还实现以下步骤:

[0288] 对所述数据地址获取请求进行解析,得到所述数据地址获取请求对应的关键码;

[0289] 根据所述关键码,从所述多个业务服务器中确定所述关键码对应的所述目标服务器的服务地址。

[0290] 在一个实施例中,提供了一种计算机可读存储介质,其上存储有计算机程序,计算机程序被处理器执行时实现以下步骤:

[0291] 获取原始增量业务数据,并对所述原始增量业务数据进行数据分片,得到增量分片业务数据;

[0292] 对所述增量分片业务数据进行处理,得到具有预设数据结构的增量业务数据,并将所述增量业务数据存储于分布式文件系统HDFS中;

[0293] 基于所述增量业务数据更新目标服务器中的数据版本信息,并发送数据版本更新通知至所述目标服务器;

[0294] 其中,所述目标服务器为所述多个业务服务器中的一个业务服务器,所述数据版本更新通知用于通知所述目标服务器,根据更新的数据版本信息从所述HDFS拉取所述更新的数据版本信息对应的所述增量业务数据,并将所述增量业务数据加载至内存以供客户端访问。

[0295] 在一个实施例中,该计算机程序被处理器执行时还实现以下步骤:

[0296] 根据所述增量业务数据的所述更新的数据版本信息以及所述增量业务数据在所述HDFS中的第一存储地址,生成第一数据基本信息;其中,所述第一数据基本信息用于所述目标服务器根据所述更新的数据版本信息,从所述索引服务器中获取所述第一数据基本信息;所述第一数据基本信息用于所述目标服务器根据所述第一数据基本信息包括的所述第一存储地址,从所述第一存储地址指示的HDFS存储位置中拉取所述更新的数据版本信息对应的所述增量业务数据。

[0297] 在一个实施例中,该计算机程序被处理器执行时还实现以下步骤:

[0298] 采用拉链法对所述增量分片业务数据进行处理,得到具有预设数据结构的所述增量业务数据。

[0299] 在一个实施例中,所述业务数据处理系统还包括数据源服务器,所述数据源服务器用于获取所述原始增量业务数据,将所述原始增量业务数据存储至所述HDFS,并将所述

原始增量业务数据的第二数据基本信息发送至所述索引服务器,所述第二数据基本信息至少包括所述原始增量业务数据在所述HDFS中的第二存储地址;

[0300] 对应地,该计算机程序被处理器执行时还实现以下步骤:

[0301] 根据所述第二数据基本信息包括的所述第二存储地址,从所述第二存储地址指示的HDFS存储位置中拉取所述原始增量业务数据。

[0302] 在一个实施例中,该计算机程序被处理器执行时还实现以下步骤:

[0303] 按照预设的数据格式,对所述增量分片业务数据进行解析,得到解析后的增量分片业务数据;

[0304] 对所述解析后的增量分片业务数据进行处理,得到具有预设数据结构的所述增量业务数据。

[0305] 在一个实施例中,所述目标服务器用于获取全量数据版本信息,并根据所述全量数据版本信息,从所述HDFS拉取所述全量数据版本信息对应的全量业务数据,所述全量数据版本信息是所述索引服务器基于所述全量业务数据在所述目标服务器中更新的;所述目标服务器将所述全量业务数据加载至内存以供客户端访问。

[0306] 在一个实施例中,所述目标服务器用于若检测到重启触发事件,则根据所述更新的数据版本信息从所述HDFS拉取所述增量业务数据,并根据所述全量数据版本信息从所述HDFS拉取所述全量业务数据;

[0307] 所述目标服务器将拉取的所述增量业务数据和所述全量业务数据加载至内存以供客户端访问。

[0308] 在一个实施例中,该计算机程序被处理器执行时还实现以下步骤:

[0309] 从所述HDFS获取全量分片业务数据,所述全量分片业务数据是对原始全量业务数据进行数据分片后得到的;

[0310] 采用开放定址法对所述全量分片业务数据进行处理,生成具有预设数据结构的所述全量业务数据;

[0311] 将所述全量业务数据存储至所述HDFS,并根据所述全量业务数据的全量数据版本信息和第三存储地址生成第三数据基本信息;

[0312] 其中,所述第三数据基本信息用于所述目标服务器根据所述全量数据版本信息,从所述索引服务器中获取所述第三数据基本信息;所述目标服务器根据所述第三数据基本信息包括的所述第三存储地址,从所述第三存储地址指示的HDFS存储位置中拉取所述全量数据版本信息对应的所述全量业务数据。

[0313] 在一个实施例中,该计算机程序被处理器执行时还实现以下步骤:

[0314] 获取所述原始全量业务数据的分片方式信息,并检测所述分片方式信息是否与本地分片方式信息相同,所述本地分片方式信息与所述多个业务服务器的数量有关;

[0315] 若所述分片方式信息与所述本地分片方式信息相同,则执行所述索引服务器从所述HDFS获取全量分片业务数据的步骤;

[0316] 若所述分片方式信息与所述本地分片方式信息不同,则根据所述本地分片方式信息对所述原始全量业务数据进行数据分片,得到所述全量分片业务数据,并执行所述索引服务器从所述HDFS获取全量分片业务数据的步骤。

[0317] 在一个实施例中,所述目标服务器中包括第一分片数据副本和第二分片数据副

本,所述第一分片数据副本和所述第二分片数据副本用于提供相同的数据访问服务,所述目标服务器具体用于根据所述增量业务数据对所述第一分片数据副本进行数据更新,在所述第一分片数据副本数据更新的过程中,所述目标服务器通过所述第二分片数据副本提供数据访问服务,并禁止数据更新过程中的第一分片数据副本提供数据访问服务;在所述第一分片数据副本数据更新完成后,所述目标服务器根据所述增量业务数据对所述第二分片数据副本进行数据更新,在所述第二分片数据副本数据更新的过程中,所述目标服务器通过数据更新后的第一分片数据副本提供数据访问服务,并禁止数据更新过程中的第二分片数据副本提供数据访问服务。

[0318] 在一个实施例中,该计算机程序被处理器执行时还实现以下步骤:

[0319] 若接收到所述客户端发送的数据地址获取请求,则根据所述数据地址获取请求确定所述目标服务器的服务地址;

[0320] 将所述服务地址返回至所述客户端;所述服务地址用于所述客户端根据所述服务地址,发送针对所述增量业务数据的访问请求至所述目标服务器;所述目标服务器接收所述访问请求,并基于内存中加载的所述增量业务数据响应所述访问请求。

[0321] 在一个实施例中,该计算机程序被处理器执行时还实现以下步骤:

[0322] 对所述数据地址获取请求进行解析,得到所述数据地址获取请求对应的关键码;

[0323] 根据所述关键码,从所述多个业务服务器中确定所述关键码对应的所述目标服务器的服务地址。

[0324] 本领域普通技术人员可以理解实现上述实施例方法中的全部或部分流程,是可以通过计算机程序来指令相关的硬件来完成,所述的计算机程序可存储于一非易失性计算机可读取存储介质中,该计算机程序在执行时,可包括如上述各方法的实施例的流程。其中,本申请所提供的各实施例中所使用的对存储器、存储、数据库或其它介质的任何引用,均可包括非易失性和易失性存储器中的至少一种。非易失性存储器可包括只读存储器(Read-Only Memory,ROM)、磁带、软盘、闪存或光存储器等。易失性存储器可包括随机存取存储器(Random Access Memory,RAM)或外部高速缓冲存储器。作为说明而非局限,RAM可以是多种形式,比如静态随机存取存储器(Static Random Access Memory,SRAM)或动态随机存取存储器(Dynamic Random Access Memory,DRAM)等。

[0325] 以上实施例的各技术特征可以进行任意的组合,为使描述简洁,未对上述实施例中的各个技术特征所有可能的组合都进行描述,然而,只要这些技术特征的组合不存在矛盾,都应当认为是本说明书记载的范围。

[0326] 以上所述实施例仅表达了本申请的几种实施方式,其描述较为具体和详细,但并不能因此而理解为对发明专利范围的限制。应当指出的是,对于本领域的普通技术人员来说,在不脱离本申请构思的前提下,还可以做出若干变形和改进,这些都属于本申请的保护范围。因此,本申请专利的保护范围应以所附权利要求为准。

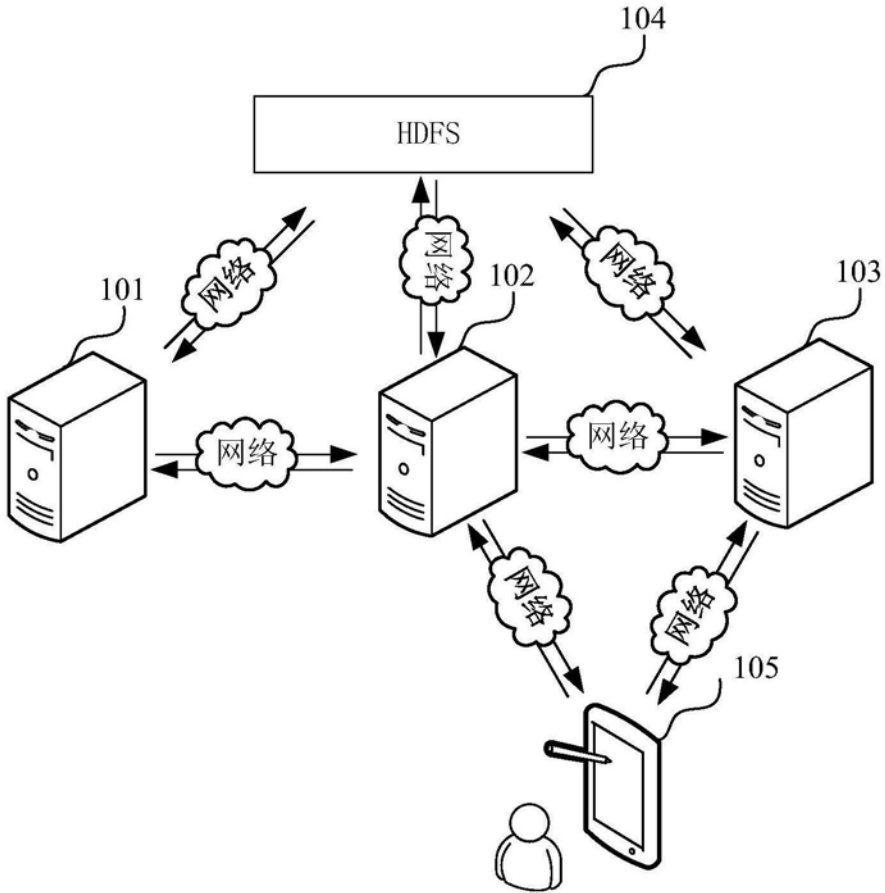


图1

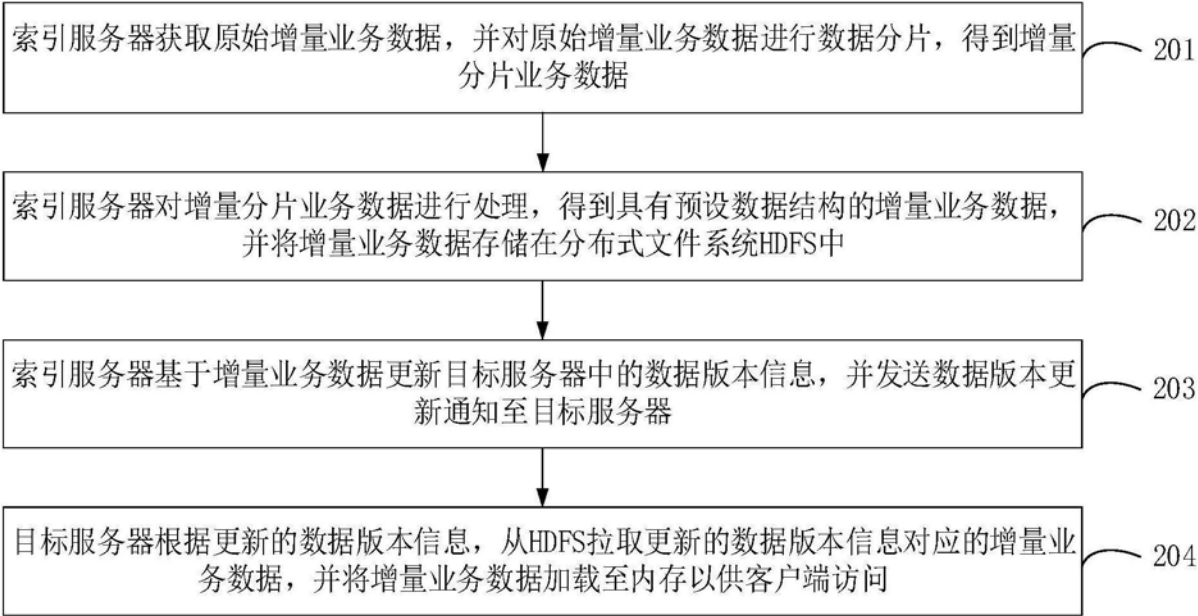


图2

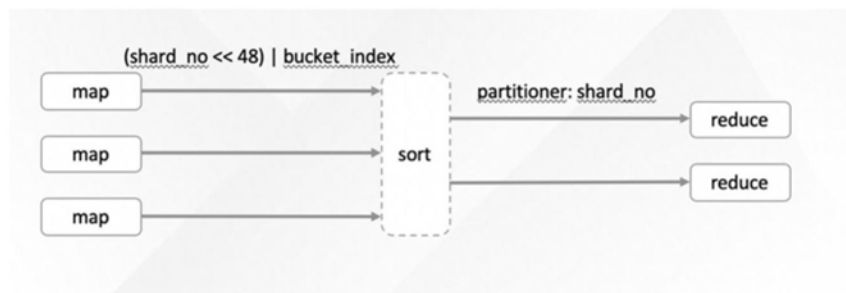


图3

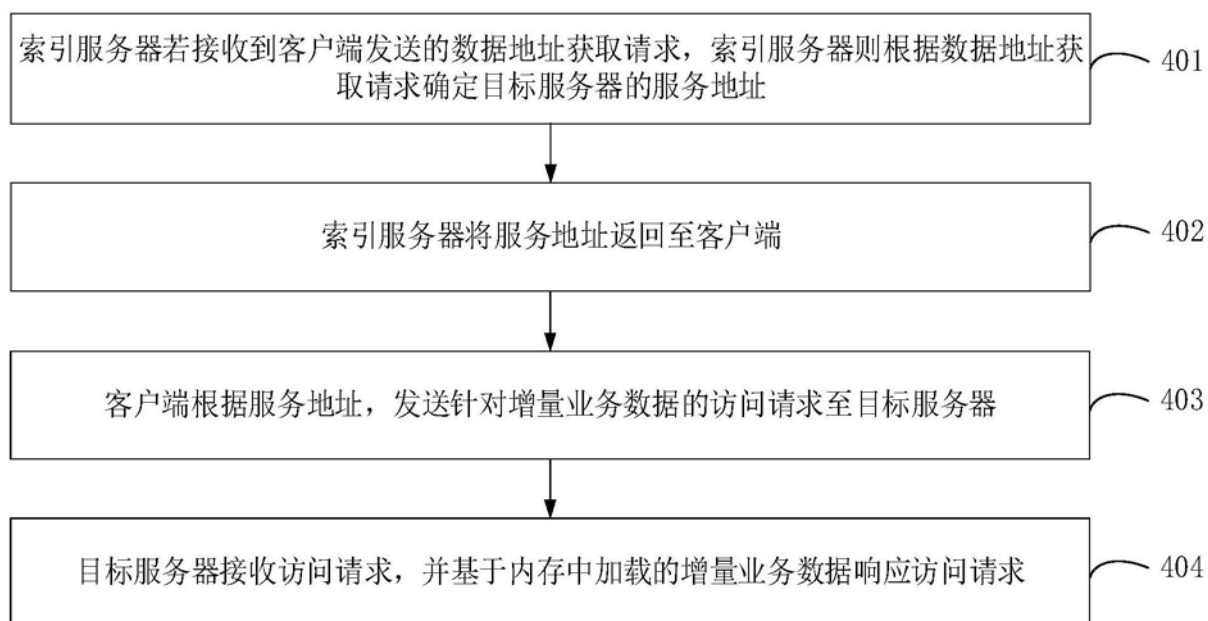


图4

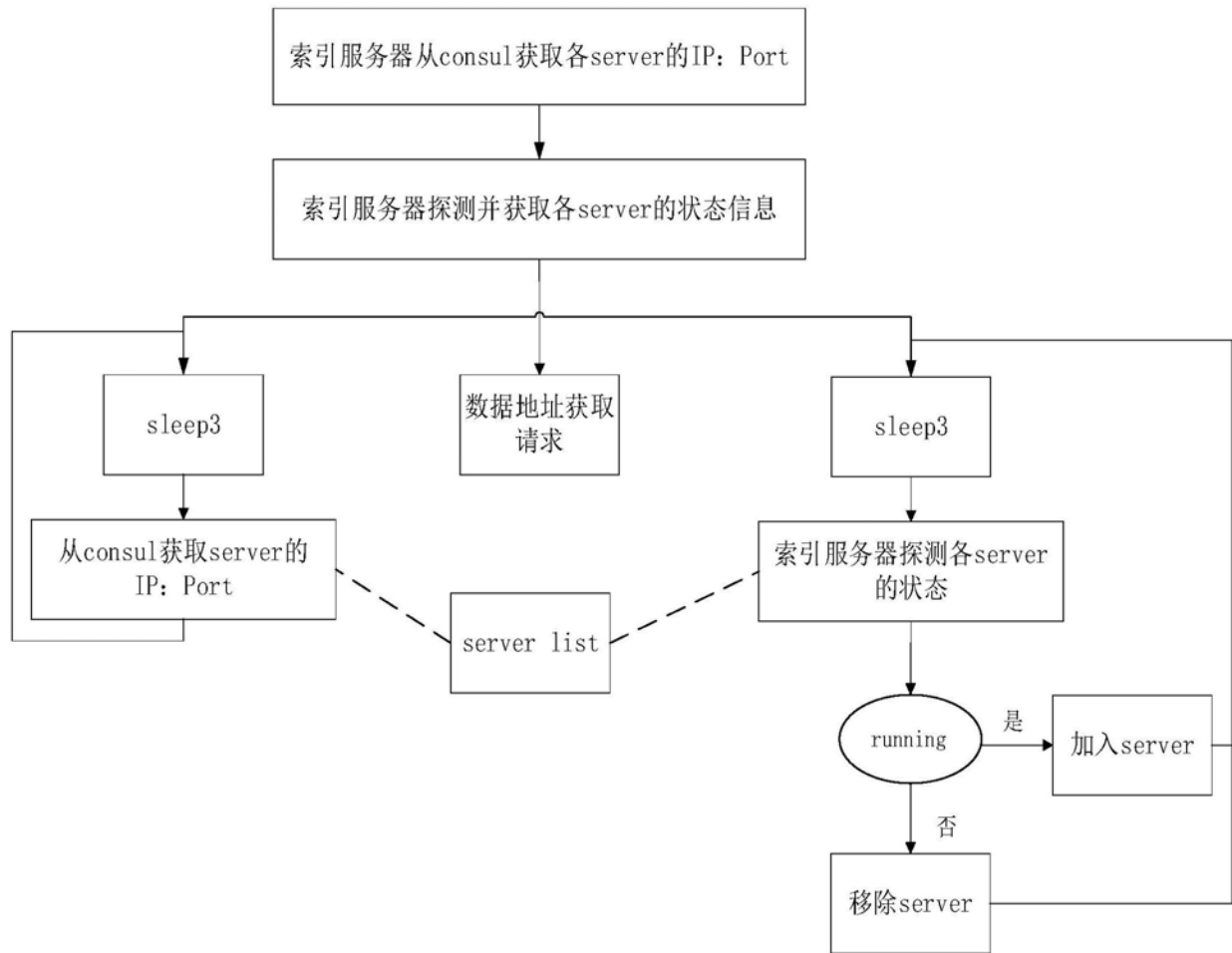


图5

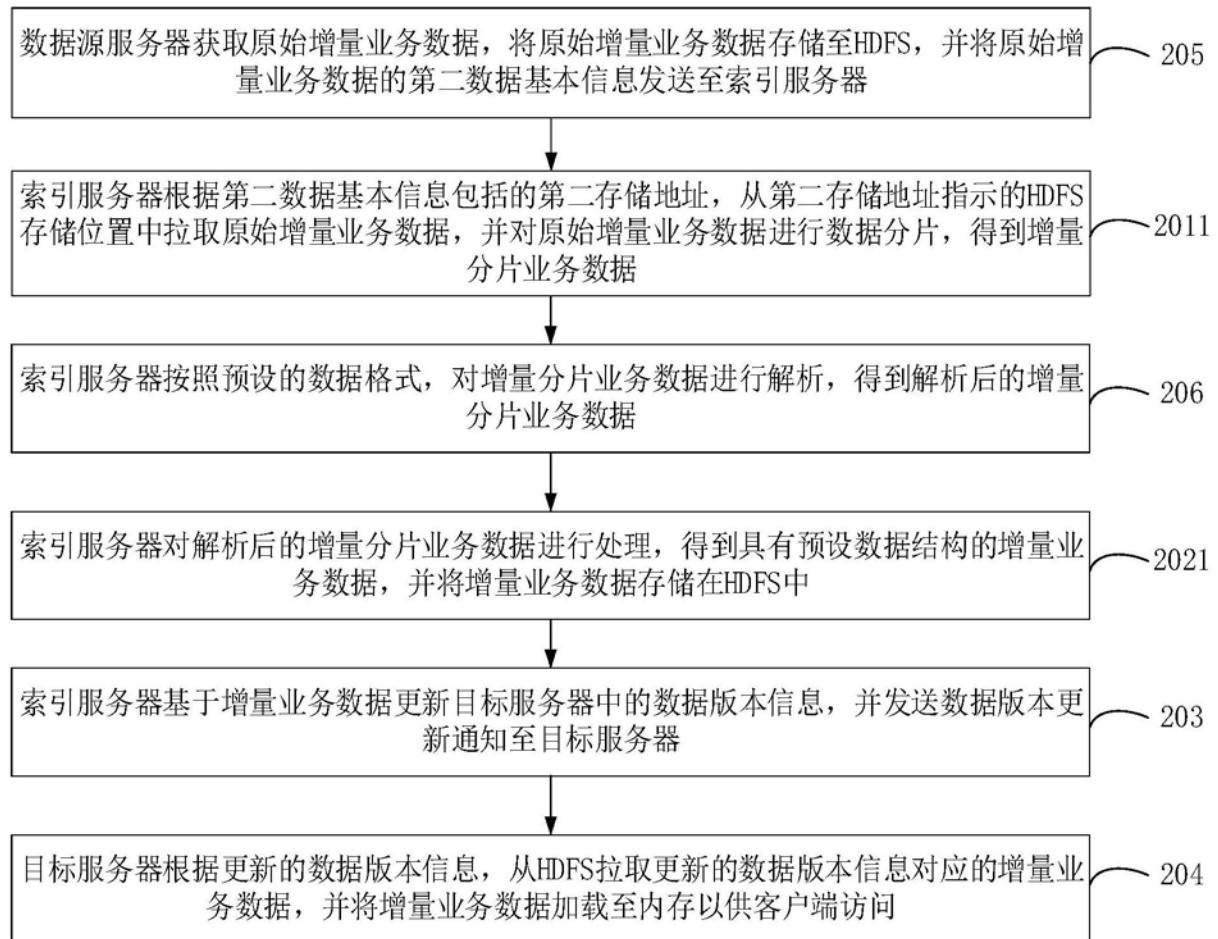


图6

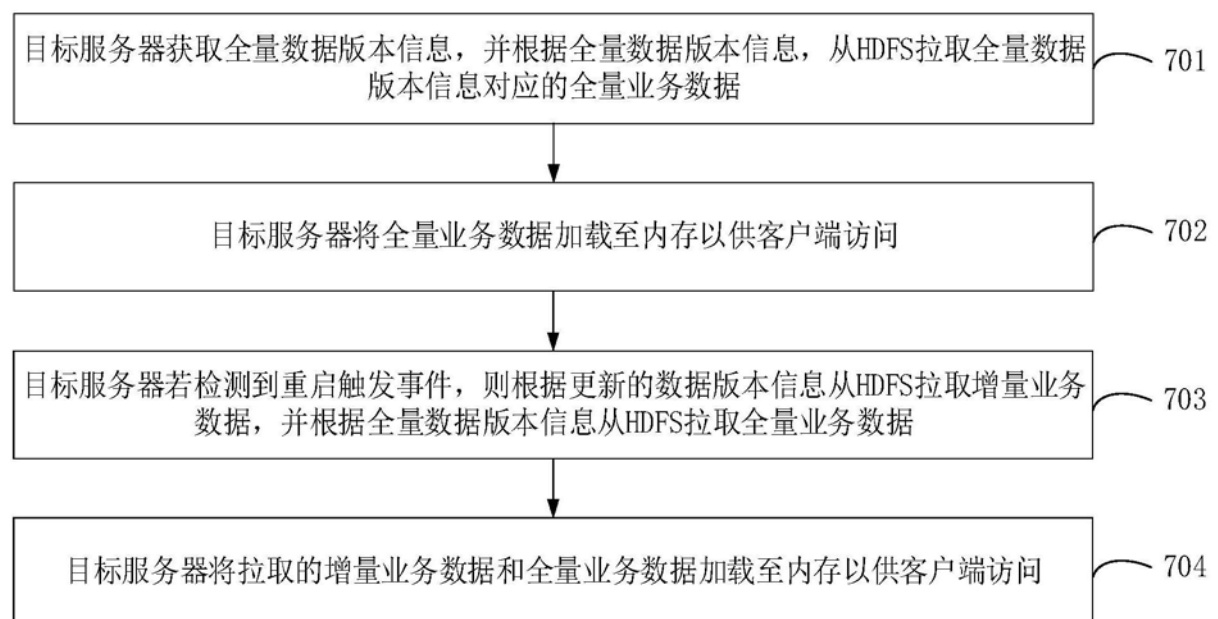


图7

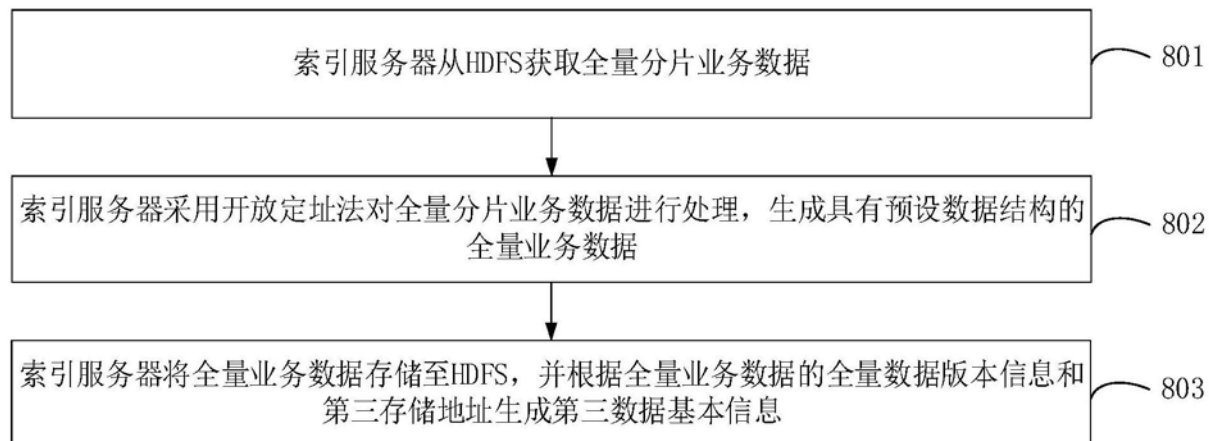


图8

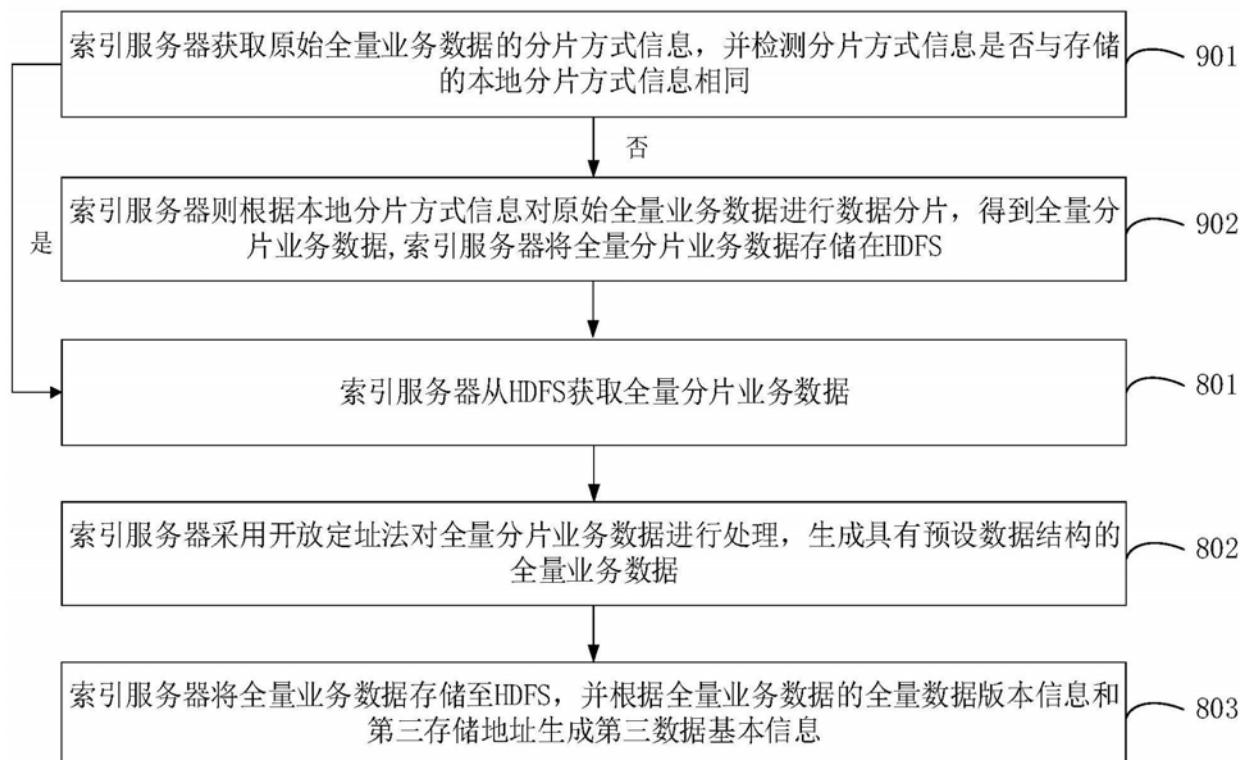


图9

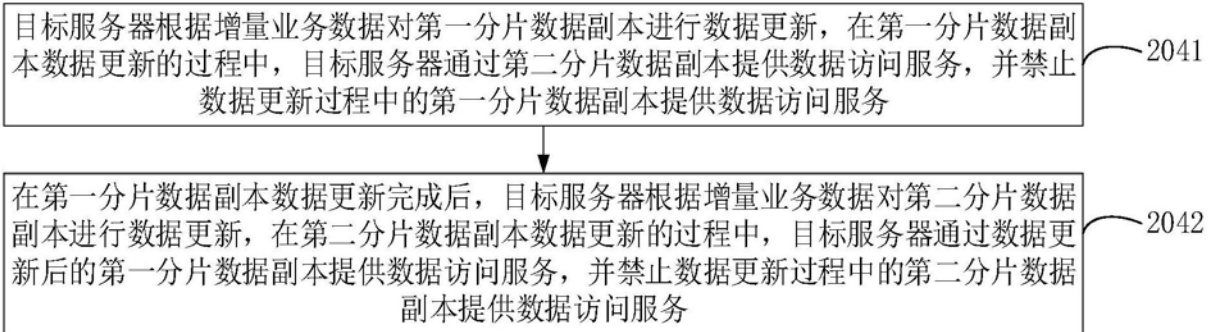


图10

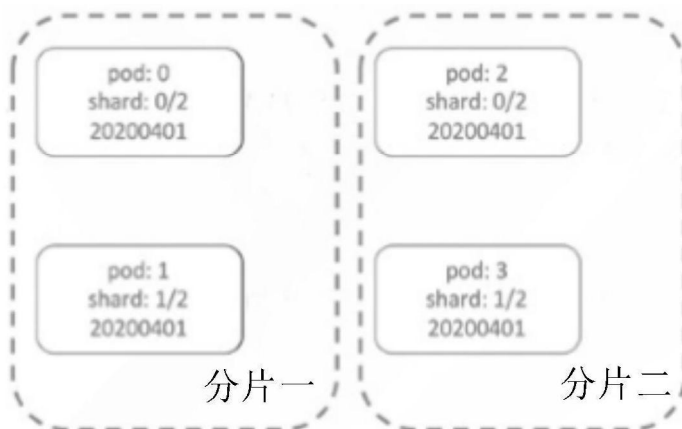


图11

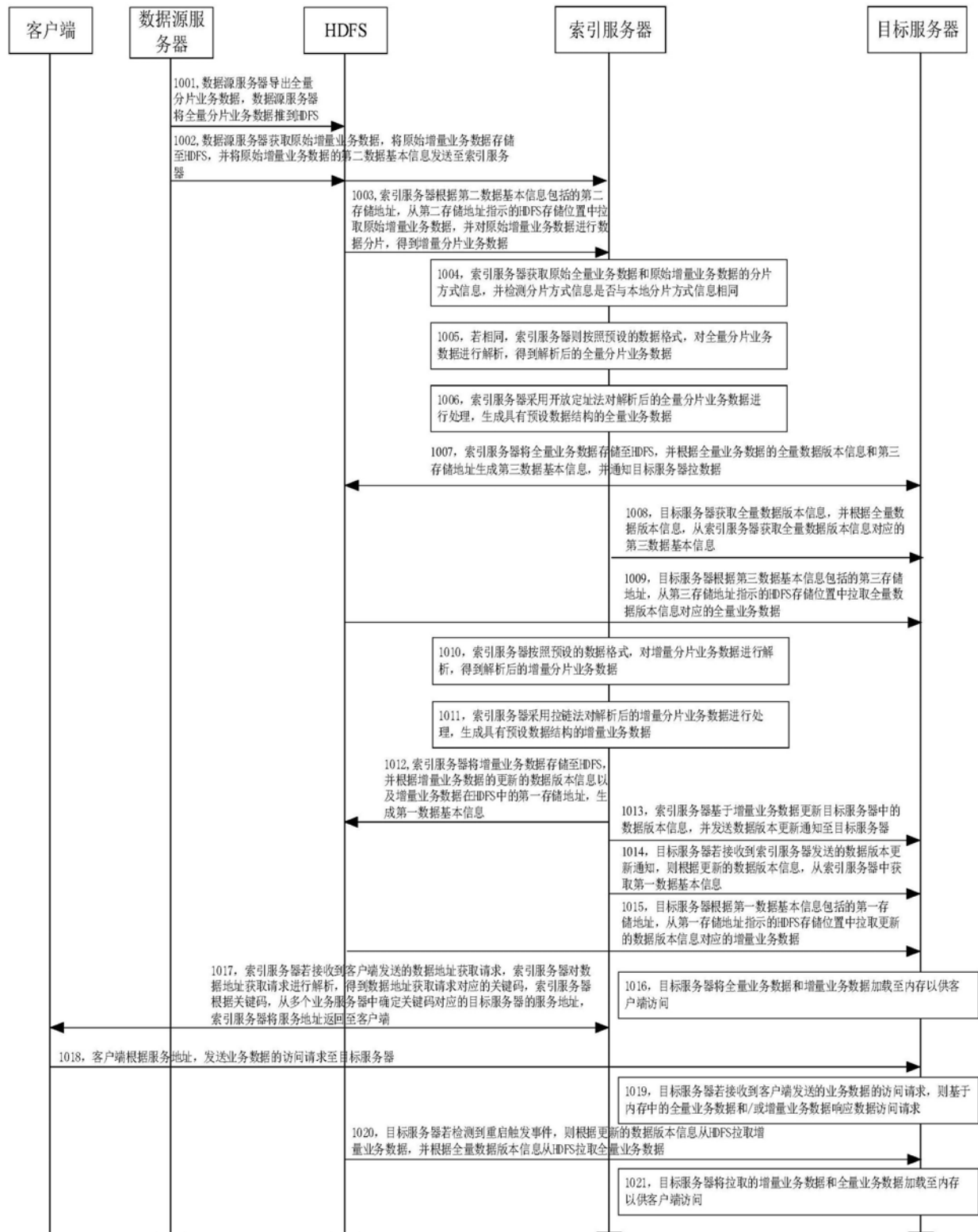


图12

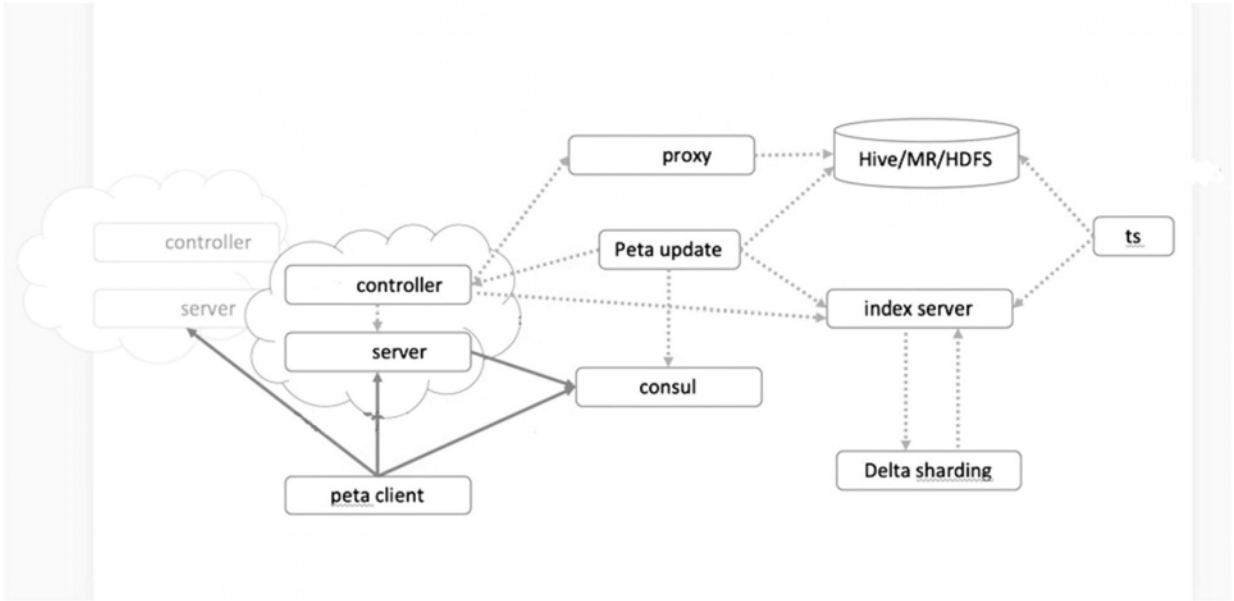


图13

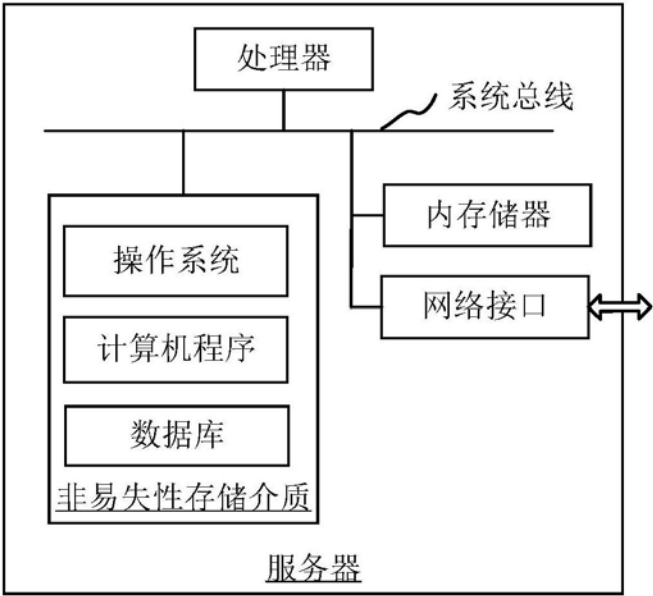


图14