(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 112052141 B
(45)授权公告日 2022.04.01

(21)申请号 202010908571.0

(22)申请日 2020.09.02

(65)同一申请的已公布的文献号
申请公布号 CN 112052141 A

(43)申请公布日 2020.12.08

(73)专利权人 平安科技(深圳)有限公司
地址 518000 广东省深圳市福田区福田街道福安社区益田路5033号平安金融中心23楼

(72)发明人 何自兴 张德顺 张汉文 张煜清 冯宇 冯玉娜

(74)专利代理机构 北京英特普罗知识产权代理有限公司 11015
代理人 程超

(51)Int.Cl.
G06F 11/30(2006.01)
G06F 16/13(2019.01)
G06F 16/14(2019.01)
G06F 16/182(2019.01)

审查员 薛聪帆

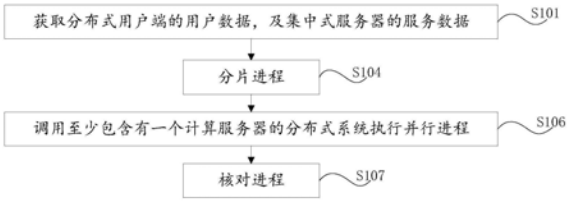
权利要求书3页 说明书12页 附图5页

(54)发明名称

数据分片校验方法、装置、计算机设备及可读存储介质

(57)摘要

本发明涉及云服务技术领域，公开了一种数据分片校验方法、装置、计算机设备及可读存储介质，包括：获取用户数据及服务数据；执行分片进程，以分别对用户数据和服务数据进行分片得到用户分片数据及服务分片数据；执行并行进程，以获取值相同的用户哈希码与服务哈希码，以及与用户哈希码和服务哈希码对应的用户目标数据和服务目标数据；执行核对进程，以对用户目标数据和服务目标数据进行校验以识别其中的异常数据。本发明不仅提高了数据核对效率的技术效果，还避免了占用集中式服务器的内存和算力，进而避免了当前用户端无法在比对操作执行过程中调用集中式服务器的数据库中其他信息的问题，极大的提升了用户端与服务器之间数据交互与调用的效率。



CN 112052141 B

1.一种数据分片校验方法,用于对分布式用户端和集中式服务器之间的同步数据进行校验,包括:

获取分布式用户端的用户数据,及集中式服务器的服务数据;

执行分片进程,以分别对所述用户数据和服务数据进行分片,汇总所述用户数据中包含有同一用户哈希码的用户数据元素得到用户分片数据,及汇总所述服务数据中包含有同一服务哈希码的服务数据元素得到服务分片数据;其中,所述用户数据元素是构成所述用户数据的最小单元,所述服务数据元素是构成所述服务数据的最小单元;其中,所述用户哈希码是通过hashmap方法中的第一哈希算法将用户数据中的用户键值进行hash运算后得到的哈希码,所述服务哈希码是通过hashmap方法中的第一哈希算法将服务数据中的服务键值进行hash运算后得到的哈希码;

调用至少包含有一个计算服务器的分布式系统执行并行进程,以从用户分片数据和服务分片数据中,获取值相同的用户哈希码与服务哈希码,以及与所述用户哈希码和服务哈希码对应的用户目标数据和服务目标数据;

执行核对进程,以对所述用户目标数据和服务目标数据进行校验以识别其中的异常数据;

所述核对进程包括:

从所述用户目标数据和服务目标数据中提取值一致的用户键值和服务键值,判断所述用户键值对应的目标文件与所述服务键值对应的目标文件是否一致;

若是,则判定所述用户键值及其目标文件与所述服务键值及其目标文件一致,并将所述用户键值对应的用户分片数据的用户状态标签修改为已核对,及将由所述服务键值对应的服务分片数据的服务状态标签修改为已核对;

若否,则判定所述用户键值及其目标文件,与所述服务键值及其目标文件为非匹配数据;

从所述用户目标数据中提取所述服务目标数据中不存在的用户键值,并将所述用户键值及其目标文件设为用户单边数据;从所述服务目标数据中提取所述用户目标数据中不存在的服务键值,并将所述服务键值及其目标文件设为服务单边数据;

汇总所述非匹配数据、用户单边数据和服务单边数据得到异常数据。

2.根据权利要求1所述的数据分片校验方法,其特征在于,执行分片进程之前,包括:

验证所述用户数据中的用户键值是否具有与其对应的目标文件;若是,则保留所述用户键值;若否,则判定所述用户键值为用户无效键值并将其保存在预设的异常数据库中,及将所述用户无效键值从所述用户数据中删除;

验证所述服务数据中的服务键值是否具有与其对应的目标文件;若是,则保留所述服务键值;若否,则判定所述服务键值为服务无效键值并将其保存在预设的异常数据库中,及将所述服务无效键值从所述服务数据中删除。

3.根据权利要求1所述的数据分片校验方法,其特征在于,所述分片进程包括:

获取用户数据的用户哈希表,及服务数据的服务哈希表;其中,所述用户哈希表反映了所述用户数据中各用户数据元素在分布式用户端中的保存位置,所述服务哈希表反映了所述服务数据中各服务数据元素在集中式服务器中的保存位置;

识别所述用户哈希表中哈希码对应的哈希桶,提取所述哈希桶中的用户数据元素并汇

总，得到用户分片数据；及识别所述服务哈希表中哈希码对应的哈希桶，提取所述哈希桶中的服务数据元素并汇总得到服务分片数据。

4.根据权利要求1所述的数据分片校验方法，其特征在于，执行分片进程之后，包括：

将所述用户分片数据录入预设的用户记录表中，并在所述用户记录表中的用户分片数据上载入用户状态标签；及将所述服务分片数据录入预设的服务记录表中，并在所述服务记录表中的服务分片数据上载入服务状态标签。

5.根据权利要求1所述的数据分片校验方法，其特征在于，所述并行进程包括：

调用所述分布式系统中的计算服务器，识别内容为待核对的用户状态标签并获取与所述用户状态标签对应的用户分片数据，及识别内容为待核对的服务状态标签并获取与所述服务状态标签对应的服务分片数据；

识别所述用户分片数据和服务分片数据中值相同的用户哈希码和服务哈希码，将分别与所述用户哈希码和服务哈希码对应的用户分片数据和服务分片数据，设为用户目标数据和服务目标数据；

将由所述计算服务器获取的用户分片数据的用户状态标签修改为核对中，及将由所述计算服务器获取的服务分片数据的服务状态标签修改为核对中，以避免分布式系统其它计算服务器获取所述用户分片数据和服务分片数据。

6.根据权利要求1所述的数据分片校验方法，其特征在于，汇总所述非匹配数据、用户单边数据和服务单边数据得到异常数据之后，还包括：

将所述异常数据上传至区块链中。

7.根据权利要求1所述的数据分片校验方法，其特征在于，判断所述用户键值对应的目标文件与所述服务键值对应的目标文件是否一致的步骤，包括：

通过预设的第一哈希算法计算所述用户键值和服务键值，得到用户哈希码和服务哈希码；

判断所述用户哈希码和服务哈希码是否一致；

若所述用户哈希码和服务哈希码一致，则通过预设的第二哈希算法计算所述用户哈希码及所述用户键值的目标文件长度，得到能够反映所述用户键值对应目标文件的存储位置的用户位置码；以及通过所述第二哈希算法计算所述服务哈希码及所述服务键值的目标文件长度，得到能够反映所述服务键值对应目标文件的存储位置的服务位置码；

若所述用户哈希码和服务哈希码不一致，则判定所述用户键值和服务键值的目标文件不一致；

根据得到的用户位置码和服务位置码，判断所述用户位置码和服务位置码是否一致；

若所述用户位置码和服务位置码一致，则判定所述用户键值和服务键值的目标文件一致；

若所述用户位置码和服务位置码不一致，则判定所述用户键值和服务键值的目标文件不一致。

8.一种数据分片校验装置，其特征在于，包括：

数据获取模块，用于获取分布式用户端的用户数据，及集中式服务器的服务数据；

分片业务模块，用于执行分片进程，以分别对所述用户数据和服务数据进行分片，汇总所述用户数据中包含有同一用户哈希码的用户数据元素得到用户分片数据，及汇总所述服

务数据中包含有同一服务哈希码的服务数据元素得到服务分片数据;其中,所述用户数据元素是构成所述用户数据的最小单元,所述服务数据元素是构成所述服务数据的最小单元;

目标识别模块,用于调用至少包含有一个计算服务器的分布式系统执行并行进程,以从用户分片数据和服务分片数据中,获取值相同的用户哈希码与服务哈希码,以及与所述用户哈希码和服务哈希码对应的用户目标数据和服务目标数据;其中,所述用户哈希码是通过hashmap方法中的第一哈希算法将用户数据中的用户键值进行hash运算后得到的哈希码,所述服务哈希码是通过hashmap方法中的第一哈希算法将服务数据中的服务键值进行hash运算后得到的哈希码;

核对业务模块,用于执行核对进程,以对所述用户目标数据和服务目标数据进行校验以识别其中的异常数据;

所述核对进程包括:

从所述用户目标数据和服务目标数据中提取值一致的用户键值和服务键值,判断所述用户键值对应的目标文件与所述服务键值对应的目标文件是否一致;

若是,则判定所述用户键值及其目标文件与所述服务键值及其目标文件一致,并将所述用户键值对应的用户分片数据的用户状态标签修改为已核对,及将由所述服务键值对应的服务分片数据的服务状态标签修改为已核对;

若否,则判定所述用户键值及其目标文件,与所述服务键值及其目标文件为非匹配数据;

从所述用户目标数据中提取所述服务目标数据中不存在的用户键值,并将所述用户键值及其目标文件设为用户单边数据;从所述服务目标数据中提取所述用户目标数据中不存在的服务键值,并将所述服务键值及其目标文件设为服务单边数据;

汇总所述非匹配数据、用户单边数据和服务单边数据得到异常数据。

9.一种计算机设备,其包括存储器、处理器以及存储在存储器上并可在处理器上运行的计算机程序,其特征在于,所述计算机设备的处理器执行所述计算机程序时实现权利要求1至7任一项所述数据分片校验方法的步骤。

10.一种计算机可读存储介质,所述可读存储介质上存储有计算机程序,其特征在于,所述可读存储介质存储的所述计算机程序被处理器执行时实现权利要求1至7任一项所述数据分片校验方法的步骤。

数据分片校验方法、装置、计算机设备及可读存储介质

技术领域

[0001] 本发明涉及云服务技术领域，尤其涉及一种数据分片校验方法、装置、计算机设备及可读存储介质。

背景技术

[0002] 所述分布式用户端与集中式服务器之间产生数据交互，例如：交易，金融管理，理财等虚拟货币交易场景，并最终完成数据同步，如：经过所述虚拟货币交易后，所述用户端中保存的金融信息与集中式服务器中保存的金融信息应为一致。然而，由于分布式用户端的数量庞大，因此，集中式服务器在与分布式用户端进行数据同步时可能会出现同步失败等问题，而同步失败后所得到的同步数据往往会仍然保存在所述用户端和服务器中，导致分布式用户端与集中式服务器之间的数据产生差异，且很难被识别出来。

[0003] 为保证所述服务器和用户端之间的数据是同步的，当前做法方法是，提取分布式用户端中的用户数据，并将其录入集中式服务器的数据库中，在所述数据库内将用户数据与服务数据进行一一比对。但是，发明人意识到，这种做法的比对过程不仅会调用集中式服务器的大量资源，而且由于数据量的庞大，其比对过程将十分缓慢；又由于所述比对过程是在数据库中进行的，因此，所述用户端无法在所述比对操作执行过程中调用集中式服务器的数据库中其他信息，极大的降低了用户端与服务器之间数据交互与调用的效率。

发明内容

[0004] 本发明的目的是提供一种数据分片校验方法、装置、计算机设备及可读存储介质，用于解决现有技术存在的比对过程将十分缓慢，及用户端无法在所述比对操作执行过程中调用集中式服务器的数据库中其他信息，极大的降低了用户端与服务器之间数据交互与调用的效率的问题。

[0005] 为实现上述目的，本发明提供一种数据分片校验方法，用于对分布式用户端和集中式服务器之间的同步数据进行校验，包括：

[0006] 获取分布式用户端的用户数据，及集中式服务器的服务数据；

[0007] 执行分片进程，以分别对所述用户数据和服务数据进行分片，汇总所述用户数据中包含有同一用户哈希码的用户数据元素得到用户分片数据，及汇总所述服务数据中包含有同一服务哈希码的服务数据元素得到服务分片数据；其中，所述用户数据元素是构成所述用户数据的最小单元，所述服务数据元素是构成所述服务数据的最小单元；

[0008] 调用至少包含有一个计算服务器的分布式系统执行并行进程，以从用户分片数据和服务分片数据中，获取值相同的用户哈希码与服务哈希码，以及与所述用户哈希码和服务哈希码对应的用户目标数据和服务目标数据；

[0009] 执行核对进程，以对所述用户目标数据和服务目标数据进行校验以识别其中的异常数据。

[0010] 上述方案中，执行分片进程之前，包括：

[0011] 验证所述用户数据中的用户键值是否具有与其对应的目标文件;若是,则保留所述用户键值;若否,则判定所述用户键值为用户无效键值并将其保存在预设的异常数据库中,及将所述用户无效键值从所述用户数据中删除;

[0012] 验证所述服务数据中的服务键值是否具有与其对应的目标文件;若是,则保留所述服务键值;若否,则判定所述服务键值为服务无效键值并将其保存在预设的异常数据库中,及将所述服务无效键值从所述服务数据中删除。

[0013] 上述方案中,所述分片进程包括:

[0014] 获取用户数据的用户哈希表,及服务数据的服务哈希表;其中,所述用户哈希表反映了所述用户数据中各用户数据元素在分布式用户端中的保存位置,所述服务哈希表反映了所述服务数据中各服务数据元素在集中式服务器中的保存位置;

[0015] 识别所述用户哈希表中哈希码对应的哈希桶,提取所述哈希桶中的用户数据元素并汇总,得到用户分片数据;及识别所述服务哈希表中哈希码对应的哈希桶,提取所述哈希桶中的服务数据元素并汇总得到服务分片数据。

[0016] 上述方案中,执行分片进程之后,包括:

[0017] 将所述用户分片数据录入预设的用户记录表中,并在所述用户记录表中的用户分片数据上载入用户状态标签;及将所述服务分片数据录入预设的服务记录表中,并在所述服务记录表中的服务分片数据上载入服务状态标签。

[0018] 上述方案中,所述并行进程包括:

[0019] 调用所述分布式系统中的计算服务器,识别内容为待核对的用户状态标签并获取与所述用户状态标签对应的用户分片数据,及识别内容为待核对的服务状态标签并获取与所述服务状态标签对应的服务分片数据;

[0020] 识别所述用户分片数据和服务分片数据中值相同的用户哈希码和服务哈希码,将分别与所述用户哈希码和服务哈希码对应的用户分片数据和服务分片数据,设为用户目标数据和服务目标数据;

[0021] 将由所述计算机服务器获取的用户分片数据的用户状态标签修改为核对中,及将由所述计算机服务器获取的服务分片数据的服务状态标签修改为核对中,以避免分布式系统其它计算服务器获取所述用户分片数据和服务分片数据。

[0022] 上述方案中,所述核对进程包括:

[0023] 从所述用户目标数据和服务目标数据中提取值一致的用户键值和服务键值,判断所述用户键值对应的目标文件与所述服务键值对应的目标文件是否一致,若是,则判定所述用户键值及其目标文件与所述服务键值及其目标文件一致,并将所述用户键值对应的用户分片数据的用户状态标签修改为已核对,及将由所述服务键值对应的服务分片数据的服务状态标签修改为已核对;若否,则判定所述用户键值及其目标文件,与所述服务键值及其目标文件为非匹配数据;

[0024] 从所述用户目标数据中提取所述服务目标数据中不存在的用户键值,并将所述用户键值及其目标文件设为用户单边数据;

[0025] 从所述服务目标数据中提取所述用户目标数据中不存在的服务键值,并将所述服务键值及其目标文件设为服务单边数据;

[0026] 汇总所述非匹配数据、用户单边数据和服务单边数据得到异常数据;

[0027] 汇总所述非匹配数据、用户单边数据和服务单边数据得到异常数据之后，还包括：

[0028] 将所述异常数据上传至区块链中。

[0029] 上述方案中，判断所述用户键值对应的目标文件与所述服务键值对应的目标文件是否一致的步骤，包括：

[0030] 通过预设的第一哈希算法计算所述用户键值和服务键值，得到用户哈希码和服务哈希码；

[0031] 判断所述用户哈希码和服务哈希码是否一致；

[0032] 若所述用户哈希码和服务哈希码一致，则通过预设的第二哈希算法计算所述用户哈希码及所述用户键值的目标文件长度，得到能够反映所述用户键值对应目标文件的存储位置的用户位置码；以及通过所述第二哈希算法计算所述服务哈希码及所述服务键值的目标文件长度，得到能够反映所述服务键值对应目标文件的存储位置的服务位置码；

[0033] 若所述用户哈希码和服务哈希码不一致，则判定所述用户键值和服务键值的目标文件不一致；

[0034] 根据得到的用户位置码和服务位置码，判断所述用户位置码和服务位置码是否一致；

[0035] 若所述用户位置码和服务位置码一致，则判定所述述用户键值和服务键值的目标文件一致；

[0036] 若所述用户位置码和服务位置码不一致，则判定所述用户键值和服务键值的目标文件不一致。

[0037] 为实现上述目的，本发明还提供一种数据分片校验装置，包括：

[0038] 数据获取模块，用于获取分布式用户端的用户数据，及集中式服务器的服务数据；

[0039] 分片业务模块，用于执行分片进程，以分别对所述用户数据和服务数据进行分片，汇总所述用户数据中包含有同一用户哈希码的用户数据元素得到用户分片数据，及汇总所述服务数据中包含有同一服务哈希码的服务数据元素得到服务分片数据；其中，所述用户数据元素是构成所述用户数据的最小单元，所述服务数据元素是构成所述服务数据的最小单元；

[0040] 目标识别模块，用于调用至少包含有一个计算服务器的分布式系统执行并行进程，以从用户分片数据和服务分片数据中，获取值相同的用户哈希码与服务哈希码，以及与所述用户哈希码和服务哈希码对应的用户目标数据和服务目标数据；

[0041] 核对业务模块，用于执行核对进程，以对所述用户目标数据和服务目标数据进行校验以识别其中的异常数据。

[0042] 为实现上述目的，本发明还提供一种计算机设备，其包括存储器、处理器以及存储在存储器上并可在处理器上运行的计算机程序，所述计算机设备的处理器执行所述计算机程序时实现上述数据分片校验方法的步骤。

[0043] 为实现上述目的，本发明还提供一种计算机可读存储介质，所述可读存储介质上存储有计算机程序，所述可读存储介质存储的所述计算机程序被处理器执行时实现上述数据分片校验方法的步骤。

[0044] 本发明提供的数据分片校验方法、装置、计算机设备及可读存储介质，通过获取分布式用户端的用户数据及集中式服务器的服务数据，以避免在集中式服务器的数据库中进

行数据核对，导致该服务器需要消耗大量资源进行数据核对，而无法为用户端提供数据调用服务，导致用户端与服务器之间数据交互与调用的效率降低的问题。

[0045] 通过执行分片进程对用户数据和服务数据进行分片，有助于提高数据核对效率；通过分布式系统执行并行进程，有助于提高所述用户目标数据和用户服务数据的识别效率；通过分布式系统中各计算服务器分别对其获得的用户目标数据和服务目标数据执行核对进程，不仅提高了数据核对效率的技术效果，还因调用外部的分布式系统对所述用户目标数据和服务目标数据进行核对，避免了占用集中式服务器的内存和算力，进而避免了当前用户端无法在所述比对操作执行过程中调用集中式服务器的数据库中其他信息的问题，极大的提升了用户端与服务器之间数据交互与调用的效率。

附图说明

[0046] 图1为本发明数据分片校验方法实施例一的流程图；

[0047] 图2为本发明数据分片校验方法实施例二中数据分片校验方法的环境应用示意图；

[0048] 图3是本发明数据分片校验方法实施例二中数据分片校验方法的具体方法流程图；

[0049] 图4是本发明数据分片校验方法实施例二中分片进程的流程图；

[0050] 图5是本发明数据分片校验方法实施例二中并行进程的流程图；

[0051] 图6是本发明数据分片校验方法实施例二中核对进程的流程图；

[0052] 图7是本发明数据分片校验方法实施例二中判断所述用户键值对应的目标文件与所述服务键值对应的目标文件是否一致的流程图；

[0053] 图8为本发明数据分片校验装置实施例三的程序模块示意图；

[0054] 图9为本发明计算机设备实施例四中计算机设备的硬件结构示意图。

具体实施方式

[0055] 为了使本发明的目的、技术方案及优点更加清楚明白，以下结合附图及实施例，对本发明进行进一步详细说明。应当理解，此处所描述的具体实施例仅用以解释本发明，并不用于限定本发明。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他实施例，都属于本发明保护的范围。

[0056] 现提供以下实施例：

[0057] 实施例一：

[0058] 请参阅图1，本实施例的一种数据分片校验方法，用于对分布式用户端和集中式服务器之间的同步数据进行校验，包括：

[0059] S101：获取分布式用户端的用户数据，及集中式服务器的服务数据。

[0060] S104：执行分片进程，以分别对所述用户数据和服务数据进行分片，汇总所述用户数据中包含有同一用户哈希码的用户数据元素得到用户分片数据，及汇总所述服务数据中包含有同一服务哈希码的服务数据元素得到服务分片数据；其中，所述用户数据元素是构成所述用户数据的最小单元，所述服务数据元素是构成所述服务数据的最小单元。

[0061] S106：调用至少包含有一个计算服务器的分布式系统执行并行进程，以从用户分

片数据和服务分片数据中，获取值相同的用户哈希码与服务哈希码，以及与所述用户哈希码和服务哈希码对应的用户目标数据和服务目标数据。

[0062] S107：执行核对进程，以对所述用户目标数据和服务目标数据进行校验以识别其中的异常数据。

[0063] 在示例性的实施例中，通过获取分布式用户端的用户数据，及集中式服务器的服务数据，以避免在集中式服务器的数据库中进行数据核对，导致该服务器需要消耗大量资源进行数据核对，而无法为用户端提供数据调用服务，导致用户端与服务器之间数据交互与调用的效率降低的问题。

[0064] 需要说明的是，所述分布式用户端与集中式服务器之间产生数据交互，例如：交易，金融管理，理财等虚拟货币交易场景，并最终完成数据同步，如：经过所述虚拟货币交易后，所述用户端中保存的金融信息(如本申请中的用户数据)与集中式服务器中保存的金融信息(如本申请中的服务数据)应为一致。示例性地，分布式用户端(如：投资者A)在集中式服务器(如：基金公司B)中购买了一份价值100万的基金(所述数据交互)，通过所述数据同步使分布式用户端和集中式服务器中分别保存有“投资者A在基金公司B购买了价值100万基金”的金融信息，以分别作为所述分布式用户端的用户数据和集中式服务器的服务数据。所述用户数据是经所述数据交互过程并完成所述数据同步后，保存在所述分布式用户端中的同步数据；所述服务数据是经所述数据交互过程并完成所述数据同步后，保存在所述集中式服务器中的同步数据。

[0065] 通过执行分片进程对用户数据和服务数据进行分片，以便于使用分布式系统中各计算服务器分别获取分片后的数据，有助于提高数据核对效率；通过分布式系统执行并行进程，以获取值相同的用户哈希码和服务哈希码，获得分别与所述用户哈希码和服务哈希码对应的用户目标数据和服务目标数据，以提高所述用户目标数据和用户服务数据的识别效率；通过分布式系统中各计算服务器分别对其获得的用户目标数据和服务目标数据执行核对进程，以识别出其中的异常数据，不仅实现了提高数据核对效率的技术效果，还因调用外部的分布式系统对所述用户目标数据和服务目标数据进行核对，避免了占用集中式服务器的内存和算力，进而避免了当前用户端无法在所述比对操作执行过程中调用集中式服务器的数据库中其他信息的问题，极大的提升了用户端与服务器之间数据交互与调用的效率。

[0066] 需要说明的是，本实施例中的所述用户哈希码是通过hashmap方法中的第一哈希算法将获得的数据的用户键值key进行hash运算后得到的哈希码hashcode；所述服务哈希码是通过hashmap方法中的第一哈希算法将获得的服务键值key进行hash运算后得到的哈希码hashcode。由于通过哈希算法对key进行hash运算属于现有技术，因此，关于哈算算法及其运算原理在此不做赘述。

[0067] 实施例二：

[0068] 本实施例为上述实施例一的一种具体应用场景，通过本实施例，能够更加清楚、具体地阐述本发明所提供的方法。

[0069] 下面，以在运行有数据分片校验方法的服务器中，对获取用户数据及服务数据，并执行分片进程及调用至少包含有一个计算服务器的分布式系统执行并行进程，获取值相同的用户哈希码与服务哈希码，以及与所述用户哈希码和服务哈希码对应的用户目标数据和

服务目标数据，最后执行核对进程识别异常数据为例，来对本实施例提供的方法进行具体说明。需要说明的是，本实施例只是示例性的，并不限制本发明实施例所保护的范围。

[0070] 图2示意性示出了根据本申请实施例二的数据分片校验方法的环境应用示意图。

[0071] 在示例性的实施例中，数据分片校验方法所在的认证服务器2通过网络分别连接分布式用户端3、集中式服务器4和分布式系统5；所述服务器2可以通过一个或多个网络提供服务，网络可以包括各种网络设备，例如路由器，交换机，多路复用器，集线器，调制解调器，网桥，中继器，防火墙，代理设备和/或等等。网络可以包括物理链路，例如同轴电缆链路，双绞线电缆链路，光纤链路，它们的组合和/或类似物。网络可以包括无线链路，例如蜂窝链路，卫星链路，Wi-Fi链路和/或类似物；所述分布式用户端3中的用户端可为保存有用户数据的智能手机、平板电脑、笔记本电脑、台式电脑等计算机设备，所述集中式服务器可为保存有服务数据的服务器或服务器系统集群，所述分布式系统中的计算服务器可为预存有核对进程的服务系统。

[0072] 图3是本发明一个实施例提供的一种数据分片校验方法的具体方法流程图，该方法具体包括步骤S201至S208。

[0073] S201：获取分布式用户端的用户数据，及集中式服务器的服务数据。

[0074] 本步骤中，所述分布式用户端与集中式服务器之间产生数据交互，例如：交易，金融管理，理财等虚拟货币交易场景，并最终完成数据同步，如：经过所述虚拟货币交易后，所述用户端中保存的金融信息(如本申请中的用户数据)与集中式服务器中保存的金融信息(如本申请中的服务数据)应为一致。示例性地，分布式用户端(如：投资者A)在集中式服务器(如：基金公司B)中购买了一份价值100万的基金(所述数据交互)，通过所述数据同步使分布式用户端和集中式服务器中分别保存有“投资者A在基金公司B购买了价值100万基金”的金融信息，以分别作为所述分布式用户端的用户数据和集中式服务器的服务数据。

[0075] 所述用户数据是经所述数据交互过程并完成所述数据同步后，保存在所述分布式用户端中的同步数据；所述服务数据是经所述数据交互过程并完成所述数据同步后，保存在所述集中式服务器中的同步数据。

[0076] 于本实施例中，获取的用户数据和服务数据可为资金变化表。

[0077] 因此，为避免在集中式服务器的数据库中进行数据核对，导致该服务器需要消耗大量资源进行数据核对，而无法为用户端提供数据调用服务，导致用户端与服务器之间数据交互与调用的效率降低，本步骤通过获取分布式用户端的用户数据，及集中式服务器的服务数据以避免上述问题发生。

[0078] S202：验证所述用户数据中的用户键值是否具有与其对应的目标文件。

[0079] 若是，则保留所述用户键值；

[0080] 若否，则判定所述用户键值为用户无效键值并将其保存在预设的异常数据库中，及将所述用户无效键值从所述用户数据中删除。

[0081] 通常的，用户数据在分布式用户端中通常以键值对(key-value)的形式保存，其中，所述用户键值是该键值对的键，所述目标文件是该键值对的值，因此，对于没有目标文件(value)的用户键值(即：key)来说，该用户键值是无效的，这种用户键值的形成必然是分布式用户端中出现了运行异常，因此，将所述用户键值为用户无效键值并将其保存在预设的异常数据库中，以便于提醒管理者异常的发生，通过将所述用户无效键值从所述用户数

据中删除，以保证用于进行核对的用户数据中所有的信息都是有效的；于本实施例中，可以通过轮询机制依次识别用户数据中的用户键值，并调取与所述用户键值对应的目标文件，如果能够调取到目标文件，则说明该用户键值具有与其对应的目标文件，如果无法调取到目标文件，则说明该用户键值不具有与其对应的目标文件。

[0082] 其中，于图3中，所述S202以以下标注展示：

[0083] S21：验证所述用户数据中的用户键值是否具有与其对应的目标文件。

[0084] S22：若是，则保留所述用户键值.

[0085] S23：若否，则判定所述用户键值为用户无效键值并将其保存在预设的异常数据库中，及将所述用户无效键值从所述用户数据中删除。

[0086] S203：验证所述服务数据中的服务键值是否具有与其对应的目标文件。

[0087] 若是，则保留所述服务键值；

[0088] 若否，则判定所述服务键值为服务无效键值并将其保存在预设的异常数据库中，及将所述服务无效键值从所述服务数据中删除。

[0089] 通常的，服务数据在集中式服务器中通常以键值对(key-value)的形式保存，因此，对于没有目标文件(value)的服务键值(即：key)来说，该服务键值是无效的，这种服务键值的形成必然是集中式服务器中出现了运行异常，因此，将所述服务键值为服务无效键值并将其保存在预设的异常数据库中，以便于提醒管理者异常的发生，通过将所述服务无效键值从所述服务数据中删除，以保证用于进行核对的服务数据中所有的信息都是有效的。

[0090] 其中，于图3中，所述S203以以下标注展示：

[0091] S31：验证所述服务数据中的服务键值是否具有与其对应的目标文件。

[0092] S32：若是，则保留所述服务键值。

[0093] S33：若否，则判定所述服务键值为服务无效键值并将其保存在预设的异常数据库中，及将所述服务无效键值从所述服务数据中删除。

[0094] S204：执行分片进程，以分别对所述用户数据和服务数据进行分片，汇总所述用户数据中包含有同一哈希码的用户数据元素得到用户分片数据，及汇总所述服务数据中包含有同一哈希码的服务数据元素得到服务分片数据；其中，所述用户数据元素是构成所述用户数据的最小单元，所述服务数据元素是构成所述服务数据的最小单元。

[0095] 为便于提高数据核对效率，本步骤通过执行分片进程对用户数据和服务数据进行分片，以便于使用分布式系统中各计算服务器分别获取分片后的数据，有助于提高数据核对效率。

[0096] 于本实施例中，采用哈希一致性分片规则对所述用户数据进行分片得到至少一个用户分片数据，将所述用户分片数据录入预设的用户记录表中，及采用哈希一致性分片规则对所述服务数据进行分片得到至少一个服务分片数据。

[0097] 在一个可选的实施例中，请参阅图4，所述分片进程包括：

[0098] S41：获取用户数据的用户哈希表，及服务数据的服务哈希表；其中，所述用户哈希表反映了所述用户数据中各用户数据元素在分布式用户端中的保存位置，所述服务哈希表反映了所述服务数据中各服务数据元素在集中式服务器中的保存位置。

[0099] 本步骤中，所述用户哈希表和服务哈希表，即：hashmap，是一种将用户数据以数组

方式对数据进行存储的数据结构，每一个用户数据元素或服务数据元素将以key-value对的形式作为一个entry保存在用户哈希表或服务哈希表中，而该entry则是所述哈希表中被视为最小整体的数据单元，其中，Entry是hashmap中保存一个键值对的单元模块。

[0100] S42：识别所述用户哈希表中哈希码对应的哈希桶，提取所述哈希桶中的用户数据元素并汇总得到用户分片数据；及识别所述服务哈希表中哈希码对应的哈希桶，提取所述哈希桶中的服务数据元素并汇总，得到服务分片数据。

[0101] 本步骤，采用hashmap中的get函数获取用户哈希表及服务哈希表中的哈希码hashcode，其中，hashcode是保存用户数据元素及服务数据元素的哈希桶编号。汇总保存在同一哈希桶中的用户数据元素形成用户分片数据，及汇总保存在同一哈希桶中的服务数据元素形成服务分片数据，使得用户分片数据与服务分片数据可以根据哈希码相互对应，以便于通过并行执行核对进程，对同一哈希码的用户分片数据和服务分片数据进行比对，以便于提高数据核对效率。

[0102] S205：将所述用户分片数据录入预设的用户记录表中，并在所述用户记录表中的用户分片数据上载入用户状态标签；及将所述服务分片数据录入预设的服务记录表中，并在所述服务记录表中的服务分片数据上载入服务状态标签。

[0103] 为避免分布式系统重复调用已核对的用户分片数据和服务分片数据，导致数据核对作业效率低下的问题，本步骤通过设置用户记录表和服务记录表，并载入所述用户分片数据和服务分片数据的状态标签，所述状态标签可包括：已核对，未核对，核对中，以明晰各分片数据的当前状态，进而避免分布式系统获取已核对和核对中的用户分片数据和服务分片数据，提高了数据核对作业的执行效率。

[0104] S206：调用至少包含有一个计算服务器的分布式系统执行并行进程，以从用户分片数据和服务分片数据中，获取值相同的用户哈希码与服务哈希码，以及与所述用户哈希码和服务哈希码对应的用户目标数据和服务目标数据。

[0105] 本步骤通过分布式系统执行并行进程，以获取值相同的用户哈希码和服务哈希码，获得分别与所述用户哈希码和服务哈希码对应的用户目标数据和服务目标数据，以提高所述用户目标数据和用户服务数据的识别效率。

[0106] 在一个可选的实施例中，请参阅图5，所述并行进程包括：

[0107] S61：调用所述分布式系统中的计算服务器，识别内容为待核对的用户状态标签并获取与所述用户状态标签对应的用户分片数据，及识别内容为待核对的服务状态标签并获取与所述服务状态标签对应的服务分片数据。

[0108] S62：识别所述用户分片数据和服务分片数据中值相同的用户哈希码和服务哈希码，将分别与所述用户哈希码和服务哈希码对应的用户分片数据和服务分片数据，设为用户目标数据和服务目标数据；

[0109] S63：将由所述计算机服务器获取的用户分片数据的用户状态标签修改为核对中，及将由所述计算机服务器获取的服务分片数据的服务状态标签修改为核对中，以避免分布式系统其它计算服务器获取所述用户分片数据和服务分片数据。

[0110] 本步骤中，所述用户状态标签和服务状态标签是可编辑的，对于被计算机服务器获取的用户分片数据和服务分片数据，编辑其用户状态标签和服务状态标签的内容，使其均由“待核对”改为“核对中。”

[0111] S207:执行核对进程,以对所述用户目标数据和服务目标数据进行校验以识别其中的异常数据。

[0112] 本步骤通过核对进程对用户目标数据和服务目标数据的异常进行识别,以识别出其中的异常数据,实现提高数据核对效率的技术效果。

[0113] 在一个可选的实施例中,请参阅图6,所述核对进程包括:

[0114] S71:从所述用户目标数据和服务目标数据中提取值一致的用户键值和服务键值,判断所述用户键值对应的目标文件与所述服务键值对应的目标文件是否一致;

[0115] S72:若是,则判定所述用户键值及其目标文件与所述服务键值及其目标文件一致,并将所述用户键值对应的用户分片数据的用户状态标签修改为已核对,及将由所述服务键值对应的服务分片数据的服务状态标签修改为已核对。

[0116] S73:若否,则判定所述用户键值及其目标文件,与所述服务键值及其目标文件为非匹配数据。

[0117] S74:从所述用户目标数据中提取所述服务目标数据中不存在的用户键值,并将所述用户键值及其目标文件设为用户单边数据。

[0118] S75:从所述服务目标数据中提取所述用户目标数据中不存在的服务键值,并将所述服务键值及其目标文件设为服务单边数据。

[0119] 例如:用户目标数据包括“数据A,数据B和数据C”,而服务目标数据包括“数据A、数据B和数据D”。那么,对于S74,由于“数据C”是存在于用户目标数据而不存在于服务目标数据的,因此,“数据C”则是用户单边数据。对于S75,由于“数据D”是存在于服务目标数据中而不存在与用户目标数据中的,因此“数据D”则是服务单边数据。

[0120] S76:汇总所述非匹配数据、用户单边数据和服务单边数据得到异常数据。

[0121] 具体地,构建用于存放所述非匹配数据的第一异常哈希表,构建用户存放所述用户单边数据的第二异常哈希表,删除服务单边数据中的服务键值,并构建用于存放所述服务单边数据的第三异常哈希表,汇总所述第一异常哈希表、第二异常哈希表和第三异常哈希表得到异常数据,并将该异常数据保存至异常数据库。

[0122] 本实施例中,通过hashmap算法对所述非匹配数据、用户单边数据和服务单边数据进行hash处理,得到异常哈希表。由于hashmap算法和hash处理属于现有技术,而本申请所要解决的技术问题是如何识别异常数据并对其进行存放,因此,hashmap算法和hash处理的技术原理在此不做赘述。

[0123] 可选的,汇总所述非匹配数据、用户单边数据和服务单边数据得到异常数据之后,还包括:

[0124] 将所述异常数据上传至区块链中。

[0125] 需要说明的是,基于异常数据得到对应的摘要信息,具体来说,摘要信息由异常数据进行散列处理得到,比如利用sha256s算法处理得到。将摘要信息上传至区块链可保证其安全性和对用户的公正透明性。用户设备可以从区块链中下载得该摘要信息,以便查证异常数据是否被篡改。本示例所指区块链是分布式数据存储、点对点传输、共识机制、加密算法等计算机技术的新型应用模式。区块链(Blockchain),本质上是一个去中心化的数据库,是一串使用密码学方法相关联产生的数据块,每一个数据块中包含了一批次网络交易的信息,用于验证其信息的有效性(防伪)和生成下一个区块。区块链可以包括区块链底层平台、

平台产品服务层以及应用服务层等。

[0126] 在一个可选的实施例中，请参阅图7，判断所述用户键值对应的目标文件与所述服务键值对应的目标文件是否一致的步骤，包括：

[0127] S71-1：通过预设的第一哈希算法计算所述用户键值和服务键值，得到用户哈希码和服务哈希码。

[0128] 本步骤中，所述用户哈希码是通过hashmap方法中的第一哈希算法将获得的数据的用户键值key进行hash运算后得到的哈希码hashcode；所述服务哈希码是通过hashmap方法中的第一哈希算法将获得的服务键值key进行hash运算后得到的哈希码hashcode。

[0129] S71-2：判断所述用户哈希码和服务哈希码是否一致。

[0130] S71-3：若所述用户哈希码和服务哈希码一致，则通过预设的第二哈希算法计算所述用户哈希码及所述用户键值的目标文件长度，得到能够反映所述用户键值对应目标文件的存储位置的用户位置码；以及通过所述第二哈希算法计算所述服务哈希码及所述服务键值的目标文件长度，得到能够反映所述服务键值对应目标文件的存储位置的服务位置码，得到所述用户位置码和服务位置码后执行S71-5。

[0131] 本步骤中，所述用户位置码是通过hashmap的第二哈希算法，对所述用户哈希码hashcode以及所述用户键值key对应的目标文件value的长度进行哈希运算得到数组下标；所述服务位置码是通过hashmap的第二哈希算法，对所述服务哈希码hashcode以及所述服务键值key对应的目标文件value的长度进行哈希运算得到数组下标。

[0132] S71-4：若所述用户哈希码和服务哈希码不一致，则判定所述用户键值和服务键值的目标文件不一致。

[0133] S71-5：根据得到的用户位置码和服务位置码，判断所述用户位置码和服务位置码是否一致。

[0134] S71-6：若所述用户位置码和服务位置码一致，则判定所述述用户键值和服务键值的目标文件一致；

[0135] 并将所述用户键值对应的用户分片数据的用户状态标签修改为已核对，及将由所述服务键值对应的服务分片数据的服务状态标签修改为已核对，以避免分布式系统中其它计算服务器获取所述用户分片数据和服务分片数据。

[0136] 本步骤中，所述用户状态标签和服务状态标签是可编辑的，对于被计算机服务器获取的用户分片数据和服务分片数据，编辑其用户状态标签和服务状态标签的内容，使其均由“核对中”改为“已核对。”

[0137] S71-7：若所述用户位置码和服务位置码不一致，则判定所述用户键值和服务键值的目标文件不一致。

[0138] 本步骤通过将用户分片数据和服务分片数据中的key，得到用户哈希码和服务哈希码，并对其进行比对，再根据用户哈希码和服务哈希码得到用户位置码和服务位置码，并对其进行比对，使核对进程无需对用户分片数据和服务分片数据进行逐一比对，即可准确的获得核对结果，提高了数据核对效率。

[0139] S208：将异常数据保存在预设的异常数据表中。

[0140] 为便于数据控制者得到异常数据并对异常情况进行处理，本步骤将所述异常数据保存在异常数据表中，具体地，将所述第一异常哈希表、第二异常哈希表和第三异常哈希表

发送预设的异常数据库中保存。

[0141] 实施例三：

[0142] 请参阅图8，本实施例的一种数据分片校验装置1，包括：

[0143] 数据获取模块11，用于获取分布式用户端的用户数据，及集中式服务器的服务数据；

[0144] 分片业务模块14，用于执行分片进程，以分别对所述用户数据和服务数据进行分片，汇总所述用户数据中包含有同一用户哈希码的用户数据元素得到用户分片数据，及汇总所述服务数据中包含有同一服务哈希码的服务数据元素得到服务分片数据；其中，所述用户数据元素是构成所述用户数据的最小单元，所述服务数据元素是构成所述服务数据的最小单元；

[0145] 目标识别模块16，用于调用至少包含有一个计算服务器的分布式系统执行并行进程，以从用户分片数据和服务分片数据中，获取值相同的用户哈希码与服务哈希码，以及与所述用户哈希码和服务哈希码对应的用户目标数据和服务目标数据；

[0146] 核对业务模块17，用于执行核对进程，以对所述用户目标数据和服务目标数据进行校验以识别其中的异常数据。

[0147] 可选的，所述数据分片校验装置1还包括：

[0148] 用户验证模块12，用于验证所述用户数据中的用户键值是否具有与其对应的目标文件；若是，则保留所述用户键值；若否，则判定所述用户键值为用户无效键值并将其保存在预设的异常数据库中，及将所述用户无效键值从所述用户数据中删除。

[0149] 可选的，所述数据分片校验装置1还包括：

[0150] 服务验证模块13，用于验证所述服务数据中的服务键值是否具有与其对应的目标文件；若是，则保留所述服务键值；若否，则判定所述服务键值为服务无效键值并将其保存在预设的异常数据库中，及将所述服务无效键值从所述服务数据中删除。

[0151] 可选的，所述数据分片校验装置1还包括：

[0152] 状态标签模块15，用于将所述用户分片数据录入预设的用户记录表中，并在所述用户记录表中的用户分片数据上载入用户状态标签；及将所述服务分片数据录入预设的服务记录表中，并在所述服务记录表中的服务分片数据上载入服务状态标签。

[0153] 可选的，所述数据分片校验装置1还包括：

[0154] 异常保存模块18，用于将异常数据保存在预设的异常数据表中。

[0155] 本技术方案应用于云服务的云计算技术领域，通过对获取到的用户数据及服务数据执行分片进程，及调用至少包含有一个计算服务器的分布式系统执行并行进程，以开展分布式系统的集群计算，并获取值相同的用户哈希码与服务哈希码，以及与所述用户哈希码和服务哈希码对应的用户目标数据和服务目标数据，最后执行核对进程识别异常数据。

[0156] 实施例四：

[0157] 为实现上述目的，本发明还提供一种计算机设备6，实施例三的数据分片校验装置1的组成部分可分散于不同的计算机设备中，计算机设备6可以是执行程序的智能手机、平板电脑、笔记本电脑、台式计算机、机架式服务器、刀片式服务器、塔式服务器或机柜式服务器(包括独立的服务器，或者多个应用服务器所组成的服务器集群)等。本实施例的计算机设备至少包括但不限于：可通过系统总线相互通信连接的存储器61、处理器62，如图9所示。

需要指出的是,图9仅示出了具有组件-的计算机设备,但是应理解的是,并不要求实施所有示出的组件,可以替代的实施更多或者更少的组件。

[0158] 本实施例中,存储器61(即可读存储介质)包括闪存、硬盘、多媒体卡、卡型存储器(例如,SD或DX存储器等)、随机访问存储器(RAM)、静态随机访问存储器(SRAM)、只读存储器(ROM)、电可擦除可编程只读存储器(EEPROM)、可编程只读存储器(PROM)、磁性存储器、磁盘、光盘等。在一些实施例中,存储器61可以是计算机设备的内部存储单元,例如该计算机设备的硬盘或内存。在另一些实施例中,存储器61也可以是计算机设备的外部存储设备,例如该计算机设备上配备的插接式硬盘,智能存储卡(Smart Media Card,SMC),安全数字(Secure Digital,SD)卡,闪存卡(Flash Card)等。当然,存储器61还可以既包括计算机设备的内部存储单元也包括其外部存储设备。本实施例中,存储器61通常用于存储安装于计算机设备的操作系统和各类应用软件,例如实施例三的数据分片校验装置的程序代码等。此外,存储器61还可以用于暂时地存储已经输出或者将要输出的各类数据。

[0159] 处理器62在一些实施例中可以是中央处理器(Central Processing Unit,CPU)、控制器、微控制器、微处理器、或其他数据处理芯片。该处理器62通常用于控制计算机设备的总体操作。本实施例中,处理器62用于运行存储器61中存储的程序代码或者处理数据,例如运行数据分片校验装置,以实现实施例一和实施例二的数据分片校验方法。

[0160] 实施例五:

[0161] 为实现上述目的,本发明还提供一种计算机可读存储介质,本实施例中该计算机可读存储介质可以是易失性的,也可以是非易失性的。如闪存、硬盘、多媒体卡、卡型存储器(例如,SD或DX存储器等)、随机访问存储器(RAM)、静态随机访问存储器(SRAM)、只读存储器(ROM)、电可擦除可编程只读存储器(EEPROM)、可编程只读存储器(PROM)、磁性存储器、磁盘、光盘、服务器、App应用商城等等,其上存储有计算机程序,程序被处理器62执行时实现相应功能。本实施例的计算机可读存储介质用于存储数据分片校验装置,被处理器62执行时实现实施例一和实施例二的数据分片校验方法。

[0162] 上述本发明实施例序号仅仅为了描述,不代表实施例的优劣。

[0163] 通过以上的实施方式的描述,本领域的技术人员可以清楚地了解到上述实施例方法可借助软件加必需的通用硬件平台的方式来实现,当然也可以通过硬件,但很多情况下前者是更佳的实施方式。

[0164] 以上仅为本发明的优选实施例,并非因此限制本发明的专利范围,凡是利用本发明说明书及附图内容所作的等效结构或等效流程变换,或直接或间接运用在其他相关的技术领域,均同理包括在本发明的专利保护范围内。

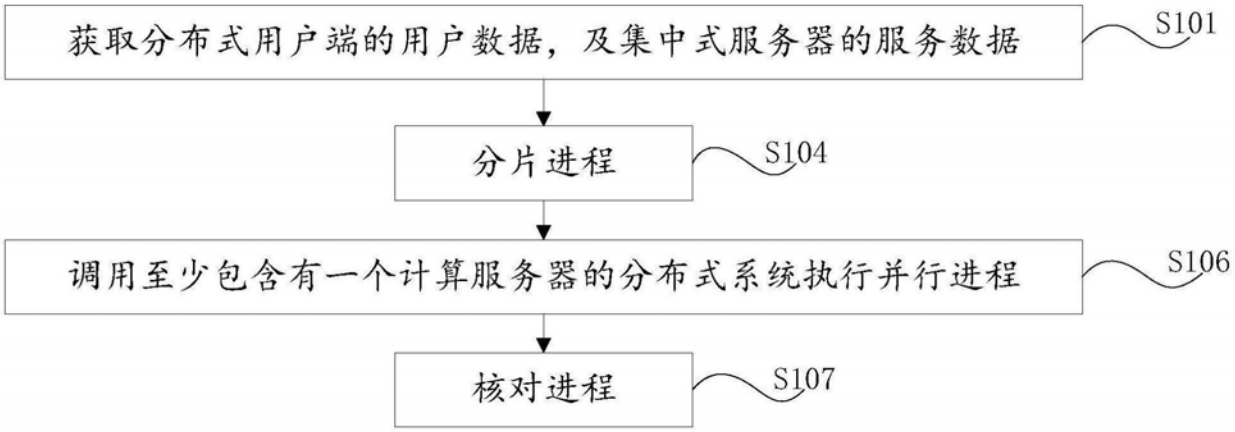

获取分布式用户端的用户数据，及集中式服务器的服务数据
S101
分片进程
S104
调用至少包含有一个计算服务器的分布式系统执行并行进程
S106
核对进程
S107

图1

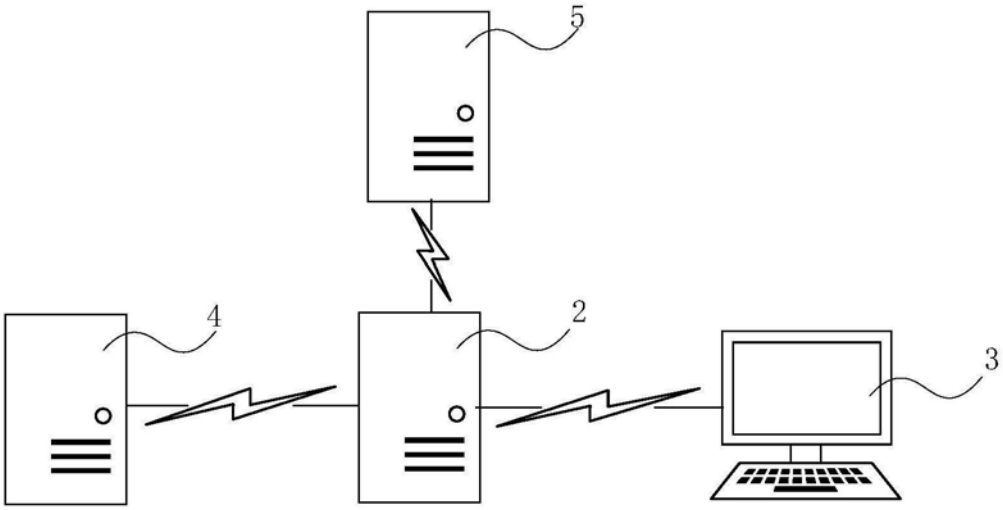

5
4
2
3

图2

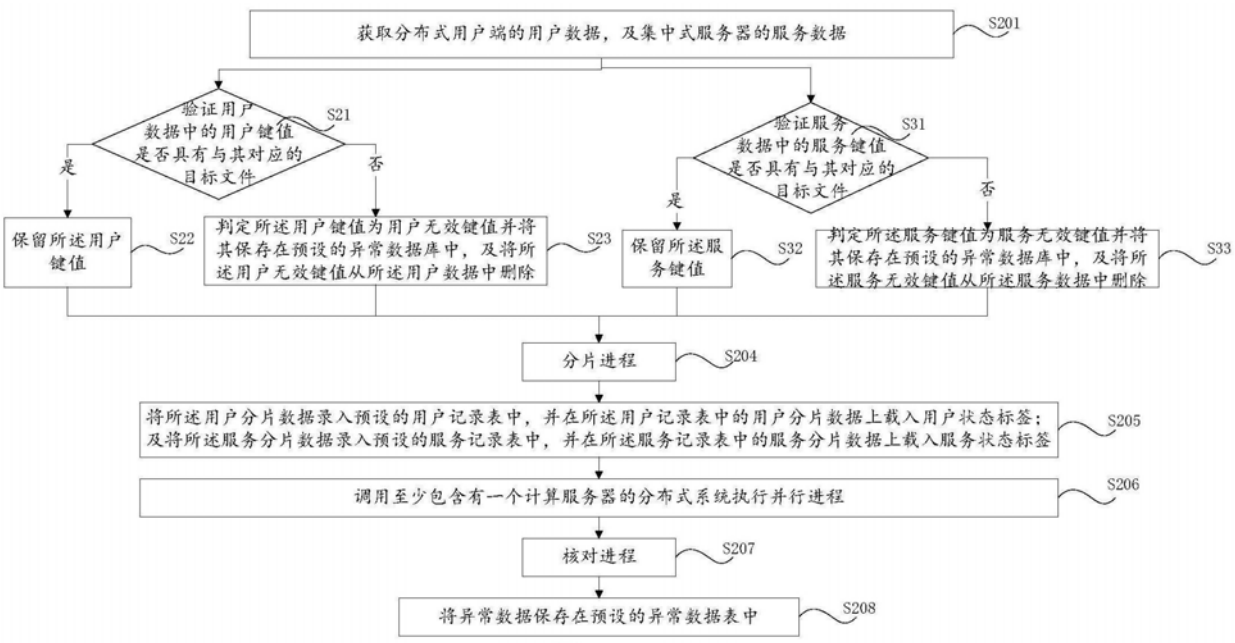

获取分布式用户端的用户数据，及集中式服务器的服务数据
S201
验证用户数据中的用户键值是否具有与其对应的目标文件
S21
是
否
保留所述用户键值
S22
判定所述用户键值为用户无效键值并将其保存在预设的异常数据库中，及将所述用户无效键值从所述用户数据中删除
S23
验证服务数据中的服务键值是否具有与其对应的目标文件
S31
是
否
保留所述服务键值
S32
判定所述服务键值为服务无效键值并将其保存在预设的异常数据库中，及将所述服务无效键值从所述服务数据中删除
S33
分片进程
S204
将所述用户分片数据录入预设的用户记录表中，并在所述用户记录表中的用户分片数据上载入用户状态标签；及将所述服务分片数据录入预设的服务记录表中，并在所述服务记录表中的服务分片数据上载入服务状态标签
S205
调用至少包含有一个计算服务器的分布式系统执行并行进程
S206
核对进程
S207
将异常数据保存在预设的异常数据表中
S208

图3

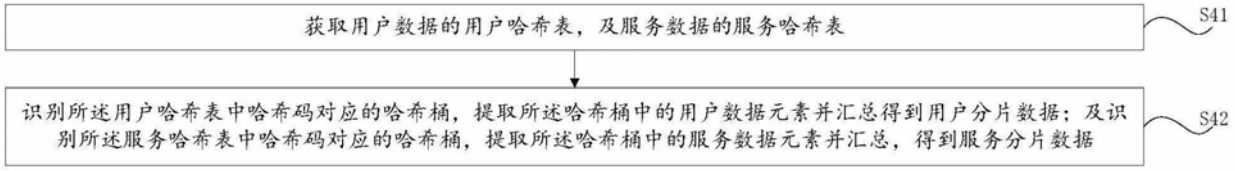

获取用户数据的用户哈希表，及服务数据的服务哈希表
S41
识别所述用户哈希表中哈希码对应的哈希桶，提取所述哈希桶中的用户数据元素并汇总得到用户分片数据；及识别所述服务哈希表中哈希码对应的哈希桶，提取所述哈希桶中的服务数据元素并汇总，得到服务分片数据
S42

图4

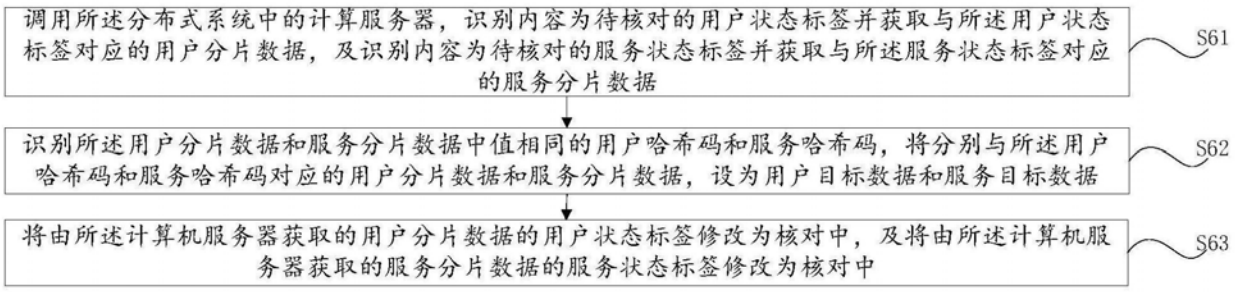

调用所述分布式系统中的计算服务器，识别内容为待核对的用户状态标签并获取与所述用户状态标签对应的用户分片数据，及识别内容为待核对的服务状态标签并获取与所述服务状态标签对应的服务分片数据
S61
识别所述用户分片数据和服务分片数据中值相同的用户哈希码和服务哈希码，将分别与所述用户哈希码和服务哈希码对应的用户分片数据和服务分片数据，设为用户目标数据和服务目标数据
S62
将由所述计算机服务器获取的用户分片数据的用户状态标签修改为核对中，及将由所述计算机服务器获取的服务分片数据的服务状态标签修改为核对中
S63

图5

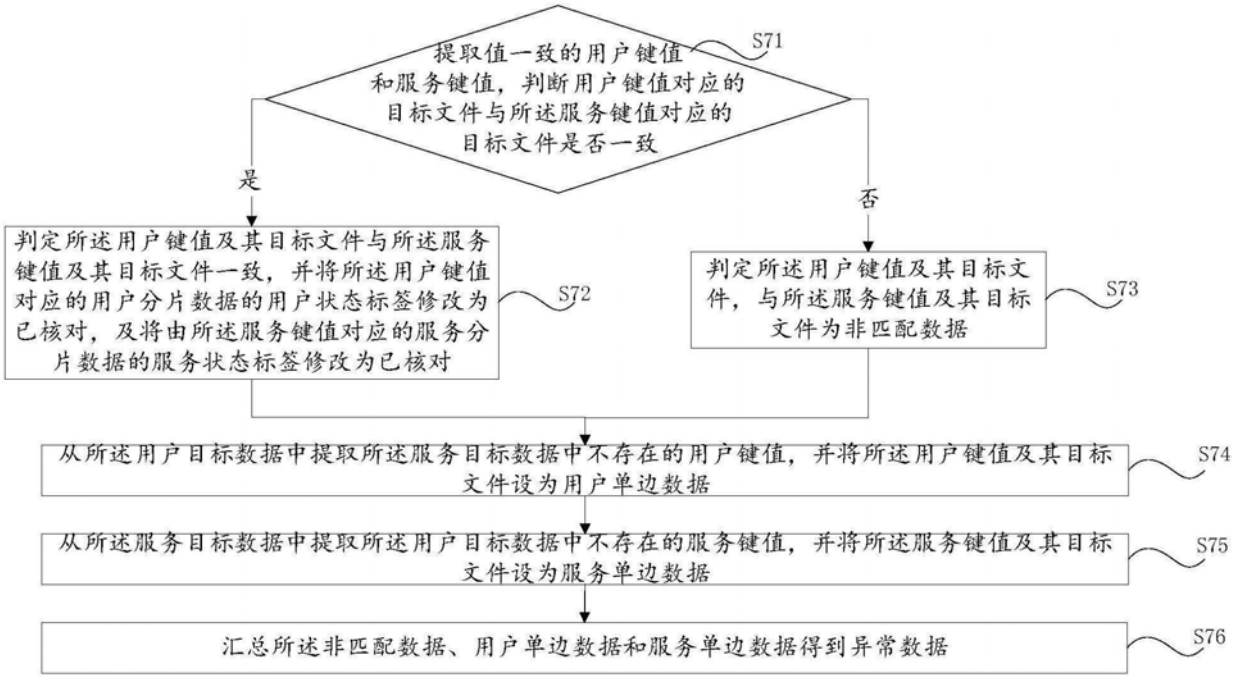

提取值一致的用户键值和服务键值，判断用户键值对应的目标文件与所述服务键值对应的目标文件是否一致
S71
是
否
判定所述用户键值及其目标文件与所述服务键值及其目标文件一致，并将所述用户键值对应的用户分片数据的用户状态标签修改为已核对，及将由所述服务键值对应的服务分片数据的服务状态标签修改为已核对
S72
判定所述用户键值及其目标文件，与所述服务键值及其目标文件为非匹配数据
S73
从所述用户目标数据中提取所述服务目标数据中不存在的用户键值，并将所述用户键值及其目标文件设为用户单边数据
S74
从所述服务目标数据中提取所述用户目标数据中不存在的服务键值，并将所述服务键值及其目标文件设为服务单边数据
S75
汇总所述非匹配数据、用户单边数据和服务单边数据得到异常数据
S76

图6

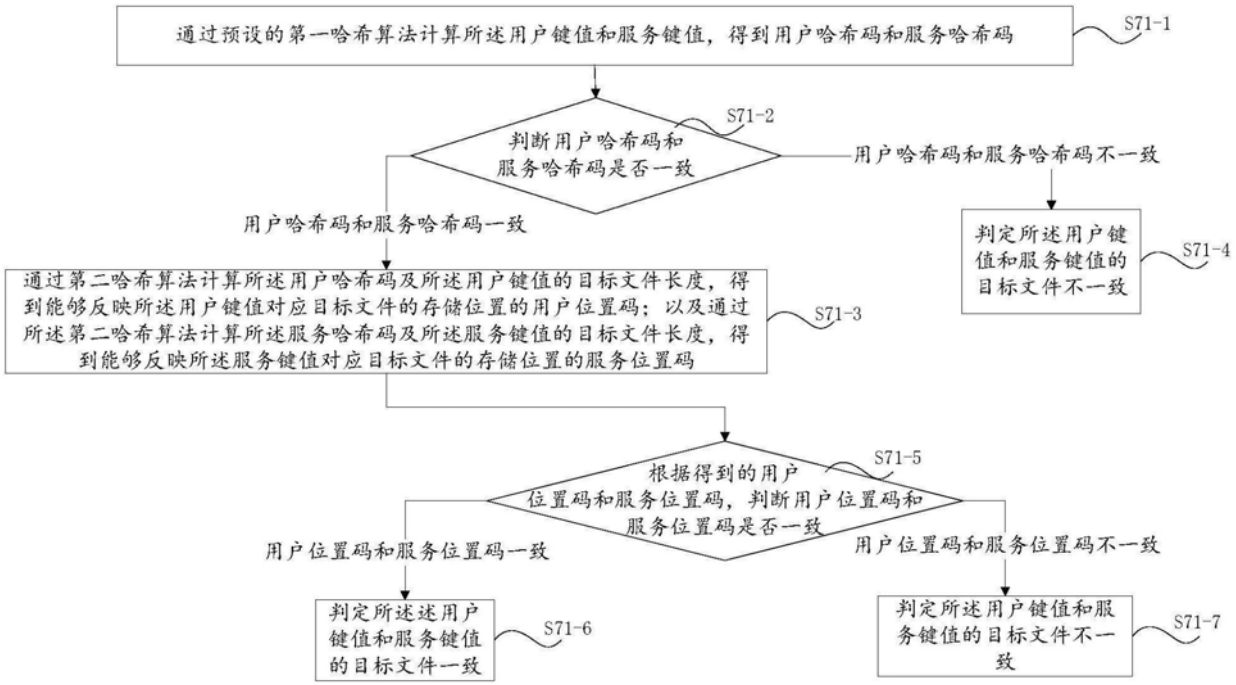

通过预设的第一哈希算法计算所述用户键值和服务键值，得到用户哈希码和服务哈希码
S71-1
判断用户哈希码和服务哈希码是否一致
S71-2
用户哈希码和服务哈希码不一致
用户哈希码和服务哈希码一致
判定所述用户键值和服务键值的目标文件不一致
S71-4
通过第二哈希算法计算所述用户哈希码及所述用户键值的目标文件长度，得到能够反映所述用户键值对应目标文件的存储位置的用户位置码；以及通过所述第二哈希算法计算所述服务哈希码及所述服务键值的目标文件长度，得到能够反映所述服务键值对应目标文件的存储位置的服务位置码
S71-3
根据得到的用户位置码和服务位置码，判断用户位置码和服务位置码是否一致
S71-5
用户位置码和服务位置码一致
用户位置码和服务位置码不一致
判定所述述用户键值和服务键值的目标文件一致
S71-6
判定所述用户键值和服务键值的目标文件不一致
S71-7

图7


数据获取模块11
用户验证模块12
服务验证模块13
分片业务模块14
状态标签模块15
目标识别模块16
核对业务模块17
异常保存模块18
数据分片校验装置1

图8

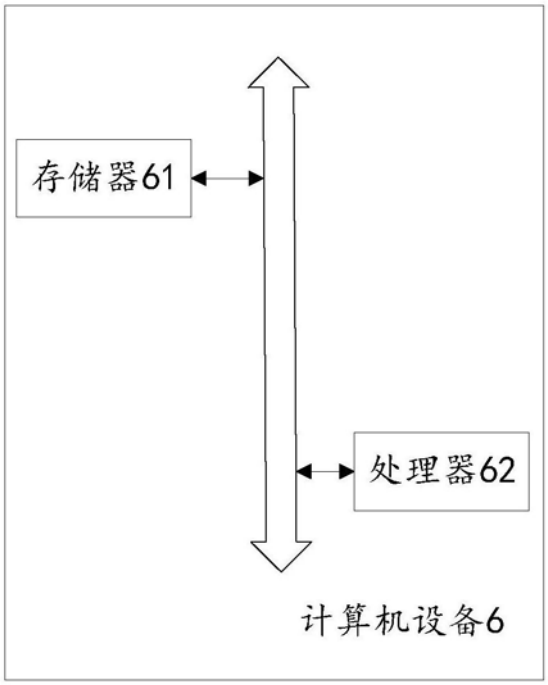


图9