(19)国家知识产权局

(12)发明专利

(10)授权公告号 CN 116126872 B
(45)授权公告日 2023.06.23

(21)申请号 202310410695.X

(22)申请日 2023.04.18

(65)同一申请的已公布的文献号
申请公布号 CN 116126872 A

(43)申请公布日 2023.05.16

(73)专利权人 紫金诚征信有限公司
地址 610095 四川省成都市自由贸易试验区成都高新区天府大道北段1677号B座1层

(72)发明人 王锦胤 马绍桐 刘海涛

(74)专利代理机构 北京慧加伦知识产权代理有限公司 16035
专利代理师 兰海叶

(51)Int.Cl.
G06F 16/22(2019.01)
G06F 16/23(2019.01)
G06F 16/2455(2019.01)
G06F 16/27(2019.01)

(56)对比文件
CN 112306700 A,2021.02.02
CN 112765166 A,2021.05.07
CN 113609374 A,2021.11.05
CN 113760988 A,2021.12.07
CN 113761018 A,2021.12.07
CN 114510486 A,2022.05.17
CN 115048372 A,2022.09.13
CN 115185967 A,2022.10.14
CN 115328958 A,2022.11.11
CN 115587118 A,2023.01.10
US 2015169661 A1,2015.06.18
WO 2009004231 A2,2009.01.08
屈美娟;付良廷;.大数据文件存储策略探索.科技创新与应用.2019,(第12期),第146-147页.
张东;亓开元;吴楠;辛国茂;刘正伟;颜秉珩;郭锋;.云海大数据一体机体系结构和关键技术.计算机研究与发展.2016,(第02期),第148-163页.

审查员 田志方

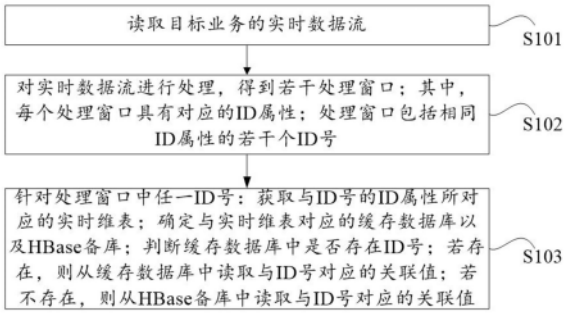
权利要求书2页 说明书9页 附图3页

(54)发明名称

一种针对实时维表的关联方法、装置及计算机可读介质

(57)摘要

本发明实施例公开了一种针对实时维表的关联方法、装置及计算机可读介质，该方法应用于第一设备；一具体实施方式至少包括：首先读取目标业务的实时数据流；并对实时数据流进行处理，得到若干处理窗口；其次针对处理窗口中任一ID号：获取与ID号的ID属性所对应的实时维表；确定与实时维表对应的缓存数据库以及HBase备库；判断缓存数据库中是否存在ID号；若存在，则从缓存数据库中读取与ID号对应的关联值；若不存在，则从HBase备库中读取与ID号对应的关联值。由此通过设置缓存数据库，能够将百万级/s的实时数据流关联维表的时间从分钟级优化为秒级，从而减少了数据延迟，提高了实时数据流关联维表的时效性。



CN 116126872 B

1.一种针对实时维表的关联方法,其特征在于,应用于第一设备;所述方法包括:

读取目标业务的实时数据流;

对所述实时数据流进行处理,得到若干处理窗口;其中,每个所述处理窗口具有对应的ID属性;所述处理窗口包括相同ID属性的若干个ID号;

针对所述处理窗口中任一ID号:获取与所述ID号的ID属性所对应的实时维表;确定与所述实时维表对应的缓存数据库以及HBase备库;判断所述缓存数据库中是否存在所述ID号;若存在,则从所述缓存数据库中读取与所述ID号对应的关联值;若不存在,则从所述HBase备库中读取与所述ID号对应的关联值。

2.根据权利要求1所述的方法,其特征在于,所述对所述实时数据流进行处理,得到若干处理窗口;包括:

针对所述实时数据流中任一实时数据:对所述实时数据进行行转列,得到若干ID号;每个所述ID号具有对应的ID属性;

基于预设时间窗口,将实时数据流中的所有ID按照ID属性进行窗口聚合,生成若干个处理窗口。

3.根据权利要求1所述的方法,其特征在于,所述确定与所述实时维表对应的缓存数据库以及HBase备库,包括:

从所述实时维表中读取数据,并基于预设条件对所读取数据进行过滤,得到目标数据;

将所述目标数据写入本地对应的缓存区域,生成第一触发指令;

基于所述第一触发指令,生成与所述实时维表对应的缓存数据库;

基于所述缓存数据库更新HBase备库,得到与所述实时维表对应的HBase备库。

4.根据权利要求3所述的方法,其特征在于,所述基于所述缓存数据库更新HBase备库,得到与所述实时维表对应的HBase备库,包括:

基于HBase主库,对所述缓存数据库进行监听;

若监听结果表征所述缓存数据库中存在与所述HBase主库不同的数据;则将所述缓存数据库中更新数据写入所述HBase主库;

基于所述HBase主库的数据更新结果,同步更新HBase备库,得到与所述实时维表对应的HBase备库。

5.根据权利要求3所述的方法,其特征在于,还包括:

基于所述第一触发指令,对所述缓存数据库中更新数据进行监控;

若监控结果表征所述缓存数据库中更新数据的存储时间大于预设时间,则从所述缓存数据库中清除所述更新数据。

6.根据权利要求5所述的方法,其特征在于,还包括:

基于所述缓存数据库中更新数据的清除操作,生成第二触发指令;

基于所述第二触发指令,从当前实时维表读取数据,并基于读取结果对所述缓存数据库进行数据更新。

7.根据权利要求1所述的方法,其特征在于,所述缓存数据库包括预设时间存储的更新数据以及常用数据表;

当常用数据表中包含更新数据时,常用数据表是基于对应更新数据的更新而更新。

8.根据权利要求1所述的方法,其特征在于,还包括:

将所述关联值写入分布式数据库中。

9.一种针对实时维表的关联装置,其特征在于,应用于第一设备;所述装置包括:

读取模块,用于读取目标业务的实时数据流;

处理模块,用于对所述实时数据流进行处理,得到若干处理窗口;其中,每个所述处理窗口具有对应的ID属性;所述处理窗口包括相同ID属性的若干个ID号;

关联模块,用于针对所述处理窗口中任一ID号:获取与所述ID号的ID属性所对应的实时维表;确定与所述实时维表对应的缓存数据库以及HBase备库;判断所述缓存数据库中是否存在所述ID号;若存在,则从所述缓存数据库中读取与所述ID号对应的关联值;若不存在,则从所述HBase备库中读取与所述ID号对应的关联值。

10.一种计算机可读介质,其特征在于,其上存储有计算机程序,所述程序被处理器执行时实现如权利要求1-8中任一项所述的方法。

一种针对实时维表的关联方法、装置及计算机可读介质

技术领域

[0001] 本发明属于计算机技术领域，尤其涉及一种针对实时维表的关联方法、装置及计算机可读介质。

背景技术

[0002] 各企业在构建自己的实时数仓以及实时指标的过程中，需要进行多维度指标的整合，指标对应的数据往往存在于关系型业务库的多个表里或者消息队列中。全量实时场景往往只能作为统计，而不能提供实时的服务。流跟流进行数据关联，因关联实时维表的实时数据流流量太大导致实时维表数据关联延迟，从而无法满足业务要求的时效性问题。

[0003] 例如：在实际应用中，实时维表的数据一般是放在HBase主库中，实时数据关联实时维表通常是直接关联HBase数据库里的数据。若将百万级/s的实时数据流直接关联HBase主库，那么对1min的实时数据流执行完关联操作通常要消耗小时级的时间；由此导致数据关联延迟严重，从而影响业务要求的时效性。

[0004] 为此，针对流量比较大的实时数据流进行实时维表关联时，急需要提供一种有效且快速的关联方法以解决现有技术中数据关联延迟的问题。

发明内容

[0005] 针对现有技术存在的上述问题，本发明实施例提供了一种针对实时维表的关联方法、装置及计算机可读介质，能够实现快速且准确地将百万级/s的实时数据流关联到实时维表，提高了百万级/s的实时数据流关联维表的时效性。

[0006] 根据本发明实施例第一方面，提供一种针对实时维表的关联方法，应用于第一设备；所述方法包括：读取目标业务的实时数据流；对所述实时数据流进行处理，得到若干处理窗口；其中，每个所述处理窗口具有对应的ID属性；所述处理窗口包括相同ID属性的若干个ID号；针对所述处理窗口中任一ID号：获取与所述ID号的ID属性所对应的实时维表；确定与所述实时维表对应的缓存数据库以及HBase备库；判断所述缓存数据库中是否存在所述ID号；若存在，则从所述缓存数据库中读取与所述ID号对应的关联值；若不存在，则从所述HBase备库中读取与所述ID号对应的关联值。

[0007] 可选的，所述对所述实时数据流进行处理，得到若干处理窗口；包括：针对所述实时数据流中任一实时数据：对所述实时数据进行行转列，得到若干ID号；每个所述ID号具有对应的ID属性；基于预设时间窗口，将实时数据流中的所有ID按照ID属性进行窗口聚合，生成若干个处理窗口。

[0008] 可选的，所述确定与所述实时维表对应的缓存数据库以及HBase备库，包括：从所述实时维表中读取数据，并基于预设条件对所读取数据进行过滤，得到目标数据；将所述目标数据写入本地对应的缓存区域，生成第一触发指令；基于所述第一触发指令，生成与所述实时维表对应的缓存数据库；基于所述缓存数据库更新HBase备库，得到与所述实时维表对应的HBase备库。

[0009] 可选的，所述基于所述缓存数据库更新HBase备库，得到与所述实时维表对应的HBase备库，包括：基于HBase主库，对所述缓存数据库进行监听；若监听结果表征所述缓存数据库中存在与所述HBase主库不同的数据；则将所述缓存数据库中更新数据写入所述HBase主库；基于所述HBase主库的数据更新结果，同步更新HBase备库，得到与所述实时维表对应的HBase备库。

[0010] 可选的，所述的方法还包括：基于所述第一触发指令，对所述缓存数据库中更新数据进行监控；若监控结果表征所述缓存数据库中更新数据的存储时间大于预设时间，则从所述缓存数据库中清除所述更新数据。

[0011] 可选的，所述的方法还包括：基于所述缓存数据库中更新数据的清除操作，生成第二触发指令；基于所述第二触发指令，从当前实时维表读取数据，并基于读取结果对所述缓存数据库进行数据更新。

[0012] 可选的，所述缓存数据库包括预设时间存储的更新数据以及常用数据表。

[0013] 可选的，所述的方法还包括：将所述关联值写入分布式数据库中。

[0014] 根据本发明实施例第二方面，还提供一种针对实时维表的关联装置；所述装置包括：读取模块，用于读取目标业务的实时数据流；处理模块，用于对所述实时数据流进行处理，得到若干处理窗口；其中，每个所述处理窗口具有对应的ID属性；所述处理窗口包括相同ID属性的若干个ID号；关联模块，用于针对所述处理窗口中任一ID号：获取与所述ID号的ID属性所对应的实时维表；确定与所述实时维表对应的缓存数据库以及HBase备库；判断所述缓存数据库中是否存在所述ID号；若存在，则从所述缓存数据库中读取与所述ID号对应的关联值；若不存在，则从所述HBase备库中读取与所述ID号对应的关联值。

[0015] 根据本发明实施例第三方面，还提供一种计算机可读介质，其上存储有计算机程序，所述程序被处理器执行时实现如第一方面所述的方法。

[0016] 本发明实施例提供一种针对实时维表的关联方法，应用于第一设备；所述方法包括：首先，读取目标业务的实时数据流；并对所述实时数据流进行处理，得到若干处理窗口；其中，每个所述处理窗口具有对应的ID属性；所述处理窗口包括相同ID属性的若干个ID号；其次，针对所述处理窗口中任一ID号：获取与所述ID号的ID属性所对应的实时维表；确定与所述实时维表对应的缓存数据库以及HBase备库；判断所述缓存数据库中是否存在所述ID号；若存在，则从所述缓存数据库中读取与所述ID号对应的关联值；若不存在，则从所述HBase备库中读取与所述ID号对应的关联值。由此，通过设置缓存数据库，能够将百万级/s的实时数据流关联维表的时间从分钟级优化为秒级，从而减少了数据延迟，提高了百万级/s的实时数据流关联维表的时效性。

附图说明

[0017] 后文将参照附图以示例性而非限制性的方式详细描述本发明的一些具体实施例。附图中相同的附图标记标示了相同或类似的部件或部分。本领域技术人员应该理解，这些附图未必是按比例绘制的。附图中：

[0018] 图1为本发明一实施例提供的针对实时维表的关联方法的流程示意图；

[0019] 图2为本发明一实施例中对实时数据流进行处理的流程示意图；

[0020] 图3为本发明一实施例中确定与实时维表对应的缓存数据库以及HBase备库的流

程示意图；

[0021] 图4为本发明一应用例提供的针对实时维表的关联方法的流程示意图；

[0022] 图5为本发明一实施例提供的针对实时维表的关联装置的结构示意图。

具体实施方式

[0023] 为使本发明的目的、特征、优点能够更加的明显和易懂，下面将结合本发明实施例中的附图，对本发明实施例中的技术方案进行清楚、完整地描述，显然，所描述的实施例仅仅是本发明一部分实施例，而非全部实施例。基于本发明中的实施例，本领域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。

[0024] 如图1所示，为本发明一实施例提供的针对实时维表的关联方法的流程示意图。

[0025] 一种针对实时维表的关联方法，应用于第一设备；所述方法至少包括如下步骤：

[0026] S101，读取目标业务的实时数据流；

[0027] S102，对实时数据流进行处理，得到若干处理窗口；其中，每个处理窗口具有对应的ID属性；处理窗口包括相同ID属性的若干个ID号；

[0028] S103，针对处理窗口中任一ID号：获取与ID号的ID属性所对应的实时维表；确定与实时维表对应的缓存数据库以及HBase备库；判断缓存数据库中是否存在ID号；若存在，则从缓存数据库中读取与ID号对应的关联值；若不存在，则从HBase备库中读取与ID号对应的关联值。

[0029] 在S101至S102中，首先通过Flink(Apache Flink，开源流处理框架)读取目标业务的实时数据流；其中，实时数据流包括若干实时数据；例如：这里的“若干”用于指示百万级以上的数据量。之后针对实时数据流中的任一实时数据：对实时数据流进行语义解析，并基于语义解析结果提取与用户请求对应的若干ID号；获取每个ID号对应的ID属；由此，每一实时数据包括若干个ID属性；最后，基于ID属性，按照预设时间窗口将实时数据流划分为若干处理窗口。

[0030] 例如：针对百万级/s的实时数据流，对实时数据流进行语义解析，得两个ID号，分别为“姓名ID号”和“订单ID号”；之后分别获取两个ID号对应的ID属性，其中，姓名ID号对应ID属性为用户ID，订单ID号对应的ID属性是内容ID；最后，基于用户ID和内容ID，将实时数据流中ID号按照ID属性窗口聚合；并将窗口聚合后的若干ID号按照1min的时间窗口划分处理窗口。由于每个ID属性具有对应的处理窗口，因此对实时数据流按时间窗口划分后能获得两个不同ID属性的处理窗口。

[0031] 在S103中，Flink从实时维表读取的数据不是存储在缓存数据库就是存储在HBase主库中；其中，HBase备库是对HBase主库的备份。

[0032] 需要说明的是，缓存数据库存储的数据基于预设时间的触发更新至HBase主库中。

[0033] 例如：获取用户ID对应的第一实时维表，以及与内容ID对应的第二实时维表；其中，第一实时维表存储用户ID与身份证的对应关系，第二实时维表存储内容ID与购买清单的对应关系。Flink从第一实时维表读取的数据和从第二实时维表读取的数据分别存储于缓存的不同区域，从而得到第一缓存数据库和第二缓存数据库；基于第一缓存数据库的更新数据对HBase主库进行数据更新，同时基于第二缓存数据库的更新数据对HBase主库进行更新；由此，第一缓存数据库的更新数据和第二缓存数据库的更新数据最终都存储于HBase

主库中。

[0034] 针对第一处理窗口中任一姓名ID号:Flink首先判断缓存数据库中是否存在该姓名ID号;若存在,则此时从第一缓存数据库中读取与该姓名ID号对应的身份证号;若不存在,则说明存储于第一缓存数据库中实时维表数据已经更新至HBase主库中,此时从HBase备库中读取与该姓名ID号对应的身份证号。

[0035] 针对第二处理窗口中任一订单ID号:Flink首先判断缓存数据库中是否存在该订单ID号;若存在,则此时从第二缓存数据库中读取与该订单ID号对应的购买清单;若不存在,则说明存储于第二缓存数据库中实时维表数据已经更新至HBase主库中,此时从HBase备库中读取与该订单ID号对应的购买清单。

[0036] 在这里,实时数据流用于指示实时更新的百万级/s的数据流。对实时数据流的处理过程不做任何限定,可以基于上述方法进行处理,还可以利用Flink进行处理。

[0037] 本实施例在对实时数据流进行时关联时先从缓存数据库中关联,若缓存数据库不存在则从HBase备库中关联;由此通过在HBase主库前设置缓存数据库,不仅分担了实时数据流的关联量,而且提高了数据关联的速度;由此能够将百万级/s的实时数据流关联维表的时间从分钟级优化为秒级,从而减少了数据延迟,提高了实时数据流关联维表的时效性。

[0038] 如图2所示,为本发明一实施例中对实时数据流进行处理的流程示意图。

[0039] 在优选的实施方式中,对实时数据流进行处理,得到若干处理窗口;至少包括如下步骤:

[0040] S201,针对实时数据流中任一实时数据:对实时数据进行行转列,得到若干ID号;每个ID号具有对应的ID属性;

[0041] S202,基于预设时间窗口,将实时数据流中的所有ID按照ID属性进行窗口聚合,生成若干个处理窗口。

[0042] 具体地,利用Flink对实时数据进行行转列,得到若干ID号;针对若干ID属性中任一种ID属性:按照预设时间窗口将实时数据流中该ID属性对应的所有ID号进行窗口聚合,生成处理窗口;基于若干ID属性,生成若干处理窗口。

[0043] 例如:利用Flink对实时数据进行行转列,得到“姓名ID号”和“订单ID号”;其中,“姓名ID号”对应用户ID,“订单ID号”对应内容ID;按照预设时间窗口将实时数据流中所有ID号基于ID属性进行窗口聚合分类,生成用户ID对应的第一处理窗口,以及内容ID对应的第二处理窗口;其中,第一处理窗口都是用户ID对应的所有ID号,第二处理窗口都是内容ID对应的所有ID。

[0044] 本实施例通过Flink将读取的实时数据先进行行专列,得到若干ID号;之后基于预设时间窗口,将实时数据流中所有ID按照ID属性进行窗口聚合,生成ID属性对应的处理窗口;由此,将实时数据流中所有ID号按照ID属性以及时间窗口划分成不同ID属性的处理窗口;从而能够基于ID属性将ID号关联对应的实时维表,提高了实时数据流关联维表的时效性。

[0045] 如图3所示,为本发明一实施例中确定与实时维表对应的缓存数据库以及HBase备库的流程示意图。

[0046] 在优选的实施方式中,确定与实时维表对应的缓存数据库以及HBase备库,至少包括如下步骤:

[0047] S301,从实时维表中读取数据,并基于预设条件对所读取数据进行过滤,得到目标数据;

[0048] S302,将目标数据写入本地对应的缓存区域,生成第一触发指令;

[0049] S303,基于第一触发指令,生成与实时维表对应的缓存数据库;

[0050] S304,基于缓存数据库更新HBase备库,得到与实时维表对应的HBase备库。

[0051] 具体地,基于缓存数据库更新HBase备库,得到与实时维表对应的HBase备库,包括:基于HBase主库,对缓存数据库进行监听;若监听结果表征缓存数据库中存在与HBase主库不同的数据;则将缓存数据库中更新数据写入HBase主库;基于HBase主库的数据更新结果,同步更新HBase备库,得到与实时维表对应的HBase备库。

[0052] 例如:首先Flink从实时维表中读取数据,并根据预设条件对读取数据进行过滤,从而将读取数据中不合规数据删除,得到第一目标数据和第二目标数据;其次,Flink将第一目标数据和第二目标数据分别存储于本地对应的第一缓存区域和对应的第二缓存区域,从而得到与第一目标数据对应的第一缓存数据库以及与第二目标数据对应的第二缓存数据库;之后,基于HBase主库,HBase Proxy对第一缓存数据库的更新数据进行监听;若监听结果表征第一缓存数据库中存在与HBase主库不同的数据,则将第一缓存数据库中更新数据写入HBase主库;若监听结果表征第一缓存数据库中不存在与HBase主库不同的数据,则说明第一缓存数据库中更新数据已经写入HBase主库中。同时,HBase Proxy对第二缓存数据库的更新数据进行监听;若监听结果表征第二缓存数据库中存在与HBase主库不同的数据,则将第二缓存数据库中更新数据写入HBase主库;若监听结果表征第二缓存数据库中不存在与HBase主库不同的数据,则说明第二缓存数据库中更新数据已经写入HBase主库中;最后,通过BingLog日志保证HBase主库与HBase备库的数据一致性。

[0053] 本实施例通过HBase Proxy对缓存数据库与HBase主库的数据一致性进行监控,从而使得缓存数据库中更新数据能够及时更新至HBase主库中,进而保证了缓存数据库与HBase主库的数据一致性。本实施例通过设置HBase备库不仅能够保证实时维表数据存储的安全性,而且能够使得HBase主库与HBase备库的数据保持一致,从而有利于实时数据流的关联。

[0054] 另外,本实施例将更新数据写入HBase主库中,而在进行数据关联时是从HBase备库中读取数据;由此将读数据和写数据两个过程分开,减少服务器处理数据的压力,提高了实时数据流关联维表的效率。

[0055] 在优选的实施方式中,所述的方法还包括:基于所述第一触发指令,对所述缓存数据库中更新数据进行监控;若监控结果表征所述缓存数据库中更新数据的存储时间大于预设时间,则从所述缓存数据库中清除所述更新数据。

[0056] 例如:Flink将目标数据写入本地对应的缓存区域时,Flink就开始记录目标数据在缓存区域的存储时间;当存储时间大于预设时间时,则清除缓存数据库中的更新数据。

[0057] 在这里,预设时间是根据实际业务场景确定的;例如:预设时间为24h。

[0058] 由此,本实施例通过对缓存数据库中更新数据的存储时间进行设置,并在更新数据的存储时间大于预设时间时删除缓存数据库中更新数据,由此防止过期的更新数据浪费缓存区域,提高了缓存数据库的利用率,从而提高了实时数据流关联维表的时效性。

[0059] 在优选的实施方式中,所述的方法还包括:基于所述缓存数据库中更新数据的清

除操作,生成第二触发指令;基于所述第二触发指令,从当前实时维表中读取数据,并基于读取结果对所述缓存数据库进行数据更新。

[0060] 具体地,当第一设备接收到缓存数据库中更新数据已经清除完的指令时,第一设备控制Flink从当前实时维表中读取数据,并在对读取数据进行过滤处理后写入缓存数据库中。例如:若缓存数据库中存储的更新数据24h清除一次,那么Flink 24h对实时数据进行一次读取操作。由此,缓存数据库能够按照预设时间对所存储的实时维表的数据进行更新,提高了实时数据关联的时效性。

[0061] 在优选的实施方式中,所述缓存数据库包括预设时间存储的更新数据以及常用数据表。

[0062] 具体地,若常用数据表中包含更新数据时,则基于更新数据对常用数据表中数据进行更新;若常用数据表中不包含更新数据时,则不需要对常用数据表进行更新。在对缓存数据库中更新数据进行删除时,是基于预设时间进行删除,而对于常用数据表中更新数据则是基于常用数据表的更新操作进行删除,而不是基于预设时间进行删除。也就是说,常用数据表中更新数据的删除是基于其对应的更新数据触发,而非预设时间触发。

[0063] 在优选的实施方式中,所述的方法还包括:将所述关联值写入分布式数据库中。

[0064] Flink将所述关联值写入分布式数据库中。在这里,分布式数据库用于指示存储实时维表的分布式数据库之外的其他分布式数据库。

[0065] 下面结合具体应用对本实施例上述方法进行详细说明。

[0066] 读取目标业务的实时数据流;针对实时数据流中任一实时数据:对实时数据进行行转列,得到若干ID号;其中,每个ID号具有对应的ID属性;基于预设时间窗口,将实时数据流中的所有ID按照ID属性进行窗口聚合,生成若干个处理窗口。

[0067] 针对所述处理窗口中任一ID号:获取与该ID号的ID属性所对应的实时维表;确定与该实时维表对应的缓存数据库以及HBase备库;判断所述缓存数据库中是否存在所述ID号;若存在,则从所述缓存数据库中读取与所述ID号对应的关联值;若不存在,则从所述HBase备库中读取与所述ID号对应的关联值;其中,所述缓存数据库包括预设时间存储的更新数据以及常用数据表;当常用数据表中包含更新数据时,常用数据表是基于对应更新数据的更新而更新。

[0068] 将所述关联值写入分布式数据库中。

[0069] 确定与所述实时维表对应的缓存数据库以及HBase备库,包括:从所述实时维表中读取数据,并基于预设条件对所读取数据进行过滤,得到目标数据;将所述目标数据写入本地对应的缓存区域,生成第一触发指令;基于所述第一触发指令,生成与所述实时维表对应的缓存数据库;基于HBase主库,对所述缓存数据库进行监听;若监听结果表征所述缓存数据库中存在与所述HBase主库不同的数据,则将所述缓存数据库中更新数据写入所述HBase主库;基于所述HBase主库的数据更新结果,同步更新HBase备库,得到与所述实时维表对应的HBase备库。

[0070] 基于第一触发指令,对所述缓存数据库中更新数据进行监控;若监控结果表征所述缓存数据库中更新数据的存储时间大于预设时间,则从所述缓存数据库中清除所述更新数据。基于所述缓存数据库中更新数据的清除操作,生成第二触发指令;基于所述第二触发指令,从当前实时维表读取数据,并基于读取结果对所述缓存数据库进行数据更新。

[0071] 如图4所示,为本发明一应用例提供的针对实时维表的关联方法的流程示意图。

[0072] Flink读取实时数据,并对实时数据进行行专列,得到两个ID号,分别为“姓名ID号”和“订单ID号”;然后Flink获取“姓名ID号”的ID属性为用户ID,并获取“订单ID号”的ID属性为内容ID;之后Flink基于用户ID,按照1分钟时间窗口对实时数据流进行窗口聚合,得到用户ID对应的第一处理窗口;同时Flink基于内容ID,按照1分钟时间窗口对实时数据流进行窗口聚合,得到内容ID对应的第二处理窗口。

[0073] 获取与用户ID对应的第一实时维表,以及与内容ID对应的第二实时维表;其中,第一实时维表存储姓名ID号与身份证的对应关系,第二实时维表存储订单ID号与购买清单的对应关系。Flink读取第一实时维表的数据,并在对读取数据进行数据处理后写入用户Redis缓存;同时Flink读取第二实时维表的数据,并在对读取数据进行数据处理后写入内容Redis缓存中。同时Flink分别对用户Redis缓存和/或内容Redis缓存中更新数据的存储时间进行监控,若监控结果表征存储时间达到预设时间,则清除用户Redis缓存和/或内容Redis缓存的更新数据。

[0074] HBase Proxy分别对用户Redis缓存以及内容Redis缓存进行监控;若监控结果表征用户Redis缓存存在HBase主库不同的数据,则基于用户Redis缓存的更新数据对HBase主库进行更新,同步更新HBase备库;若监控结果表征内容Redis缓存存在HBase主库不同的数据,则基于内容Redis缓存的更新数据对HBase主库进行更新,同步更新HBase备库。

[0075] 针对第一处理窗口中任一姓名ID号:判断用户Redis缓存中是否存在该姓名ID号;若存在,则从用户Redis缓存中读取与该姓名ID号对应的身份证号;若不存在,则从HBase备库中读取与姓名ID号对应的身份证号;针对第二处理窗口中任一订单ID号:判断内容Redis缓存中是否存在订单ID号;若存在,则从内容Redis缓存中读取与该订单ID号对应的购买清单;若不存在,则从HBase备库中读取与该订单ID号对应的购买清单。最后,将若干身份证号和/或购买清单通过Sink输出,并写入分布式数据库中。

[0076] 本实施例在实时计算中进行了削峰填谷的操作,例如:基于实时数据流从缓存数据库读数据或者从HBase备库中读数据的操作,以及 Flink向缓存数据库写数据的操作;将上述两操作时间错开。由此,能够防止第一设备发生雪崩,提高了实时数据流关联维表的时效性。

[0077] 如图5所示,为本发明一实施例提供的针对实时维表的关联装置的结构示意图。

[0078] 一种针对实时维表的关联装置,应用于第一设备;所述装置500包括:读取模块501,用于读取目标业务的实时数据流;处理模块502,用于对所述实时数据流进行处理,得到若干处理窗口;其中,每个所述处理窗口具有对应的ID属性;所述处理窗口包括相同ID属性的若干个ID号;关联模块503,用于针对所述处理窗口中任一ID号:获取与所述ID号的ID属性所对应的实时维表;确定与所述实时维表对应的缓存数据库以及HBase备库;判断所述缓存数据库中是否存在所述ID号;若存在,则从所述缓存数据库中读取与所述ID号对应的关联值;若不存在,则从所述HBase备库中读取与所述ID号对应的关联值。

[0079] 在优选的实施方式中,处理模块包括:行专列单元,用于针对所述实时数据流中任一实时数据:对所述实时数据进行行转列,得到若干ID号;每个所述ID号具有对应的ID属性;窗口聚合单元,用于基于预设时间窗口,将实时数据流中的所有ID按照ID属性进行窗口聚合,生成若干个处理窗口。

[0080] 在优选的实施方式中，关联模块包括：过滤单元，用于从所述实时维表中读取数据，并基于预设条件对所读取数据进行过滤，得到目标数据；第一生成单元，用于将所述目标数据写入本地对应的缓存区域，生成第一触发指令；第二生成单元，用于基于所述第一触发指令，生成与所述实时维表对应的缓存数据库；获得单元，用于基于所述缓存数据库更新HBase备库，得到与所述实时维表对应的HBase备库。

[0081] 在优选的实施方式中，获得单元包括：监听子单元，用于基于HBase主库，对所述缓存数据库进行监听；写入子单元，用于若监听结果表征所述缓存数据库中存在与所述HBase主库不同的数据；则将所述缓存数据库中更新数据写入所述HBase主库；更新子单元，用于基于所述HBase主库的数据更新结果，同步更新HBase备库，得到与所述实时维表对应的HBase备库。

[0082] 在优选的实施方式中，所述装置还包括：监控模块，用于基于所述第一触发指令，对所述缓存数据库中更新数据进行监控；清除模块，用于若监控结果表征所述缓存数据库中更新数据的存储时间大于预设时间，则从所述缓存数据库中清除所述更新数据。

[0083] 在优选的实施方式中，所述装置还包括：生成模块，用于基于所述缓存数据库中更新数据的清除操作，生成第二触发指令；更新模块，用于基于所述第二触发指令，从当前实时维表读取数据，并基于读取结果对所述缓存数据库进行数据更新。

[0084] 在优选的实施方式中，所述缓存数据库包括预设时间存储的更新数据以及常用数据表；当常用数据表中包含更新数据时，常用数据表是基于对应更新数据的更新而更新。

[0085] 在优选的实施方式中，所述装置还包括：写入模块，用于将所述关联值写入分布式数据库中。

[0086] 上述装置可执行本发明一实施例所提供的针对实时维表的关联方法，具备执行针对实时维表的关联方法相应的功能模块和有益效果。未在本实施例中详尽描述的技术细节，可参见本发明一实施例所提供的针对实时维表的关联方法。

[0087] 本发明还提供一种电子设备，包括：处理器；用于存储所述处理器可执行指令的存储器；所述处理器，用于从所述存储器中读取所述可执行指令，并执行所述指令以实现本发明所述的针对实时维表的关联方法。

[0088] 除了上述方法和设备以外，本申请的实施例还可以是计算机程序产品，其包括计算机程序指令，所述计算机程序指令在被处理器运行时使得所述处理器执行本说明书上述“示例性方法”部分中描述的根据本申请各种实施例的方法中的步骤。

[0089] 所述计算机程序产品可以以一种或多种程序设计语言的任意组合来编写用于执行本申请实施例操作的程序代码，所述程序设计语言包括面向对象的程序设计语言，诸如Java、C++等，还包括常规的过程式程序设计语言，诸如“C”语言或类似的程序设计语言。程序代码可以完全地在用户计算设备上执行、部分地在用户设备上执行、作为一个独立的软件包执行、部分在用户计算设备上部分在远程计算设备上执行、或者完全在远程计算设备或服务器上执行。

[0090] 此外，本申请的实施例还可以是计算机可读存储介质，其上存储有计算机程序指令，所述计算机程序指令在被处理器运行时使得所述处理器执行本说明书上述“示例性方法”部分中描述的根据本申请如下各实施例的方法中的步骤。

[0091] 所述计算机可读存储介质可以采用一个或多个可读介质的任意组合。可读介质可

以是可读信号介质或者可读存储介质。可读存储介质例如可以包括但不限于电、磁、光、电磁、红外线、或半导体的系统、装置或器件，或者任意以上的组合。可读存储介质的更具体的例子(非穷举的列表)包括：具有一个或多个导线的电连接、便携式盘、硬盘、随机存取存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPROM或闪存)、光纤、便携式紧凑盘只读存储器(CD-ROM)、光存储器件、磁存储器件、或者上述的任意合适的组合。

[0092] 以上结合具体实施例描述了本申请的基本原理，但是，需要指出的是，在本申请中提及的优点、优势、效果等仅是示例而非限制，不能认为这些优点、优势、效果等是本申请的各个实施例必须具备的。另外，上述公开的具体细节仅是为了示例的作用和便于理解的作用，而非限制，上述细节并不限制本申请为必须采用上述具体的细节来实现。

[0093] 本申请中涉及的器件、装置、设备、系统的方框图仅作为例示性的例子并且不意图要求或暗示必须按照方框图示出的方式进行连接、布置、配置。如本领域技术人员将认识到的，可以按任意方式连接、布置、配置这些器件、装置、设备、系统。诸如“包括”、“包含”、“具有”等等的词语是开放性词汇，指“包括但不限于”，且可与其互换使用。这里所使用的词汇“或”和“和”指词汇“和/或”，且可与其互换使用，除非上下文明确指示不是如此。这里所使用的词汇“诸如”指词组“如但不限于”，且可与其互换使用。

[0094] 还需要指出的是，在本申请的装置、设备和方法中，各部件或各步骤是可以分解和/或重新组合的。这些分解和/或重新组合应视为本申请的等效方案。

[0095] 提供所公开的方面的以上描述以使本领域的任何技术人员能够做出或者使用本申请。对这些方面的各种修改对于本领域技术人员而言是非常显而易见的，并且在此定义的一般原理可以应用于其他方面而不脱离本申请的范围。因此，本申请不意图被限制到在此示出的方面，而是按照与在此公开的原理和新颖的特征一致的最宽范围。

[0096] 为了例示和描述的目的已经给出了以上描述。此外，此描述不意图将本申请的实施例限制到在此公开的形式。尽管以上已经讨论了多个示例方面和实施例，但是本领域技术人员将认识到其某些变型、修改、改变、添加和子组合。

[0097] 在本说明书的描述中，参考术语“一个实施例”、“一些实施例”、“示例”、“具体示例”、或“一些示例”等的描述意指结合该实施例或示例描述的具体特征、结构、材料或者特点包含于本发明的至少一个实施例或示例中。而且，描述的具体特征、结构、材料或者特点可以在任一个或多个实施例或示例中以合适的方式结合。此外，在不相互矛盾的情况下，本领域的技术人员可以将本说明书中描述的不同实施例或示例以及不同实施例或示例的特征进行结合和组合。

[0098] 此外，术语“第一”、“第二”仅用于描述目的，而不能理解为指示或暗示相对重要性或者隐含指明所指示的技术特征的数量。由此，限定有“第一”、“第二”的特征可以明示或隐含地包括至少一个该特征。在本发明的描述中，“多个”的含义是两个或两个以上，除非另有明确具体的限定。

[0099] 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技术人员在本发明揭露的技术范围内，可轻易想到变化或替换，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所述权利要求的保护范围为准。

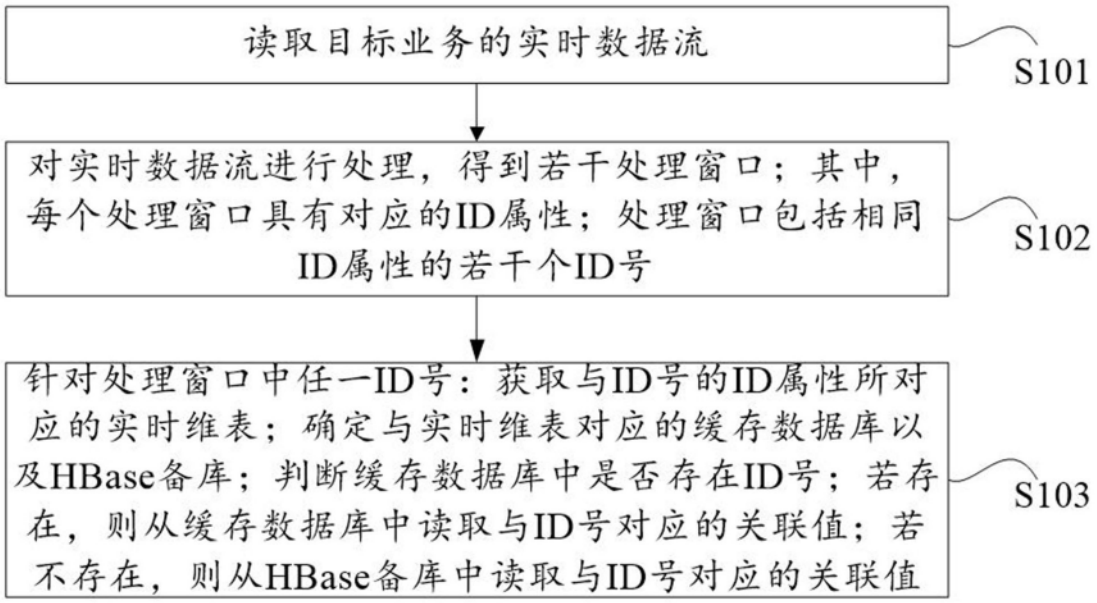

读取目标业务的实时数据流
S101
对实时数据流进行处理，得到若干处理窗口；其中，每个处理窗口具有对应的ID属性；处理窗口包括相同ID属性的若干个ID号
S102
针对处理窗口中任一ID号：获取与ID号的ID属性所对应的实时维表；确定与实时维表对应的缓存数据库以及HBase备库；判断缓存数据库中是否存在ID号；若存在，则从缓存数据库中读取与ID号对应的关联值；若不存在，则从HBase备库中读取与ID号对应的关联值
S103

图 1

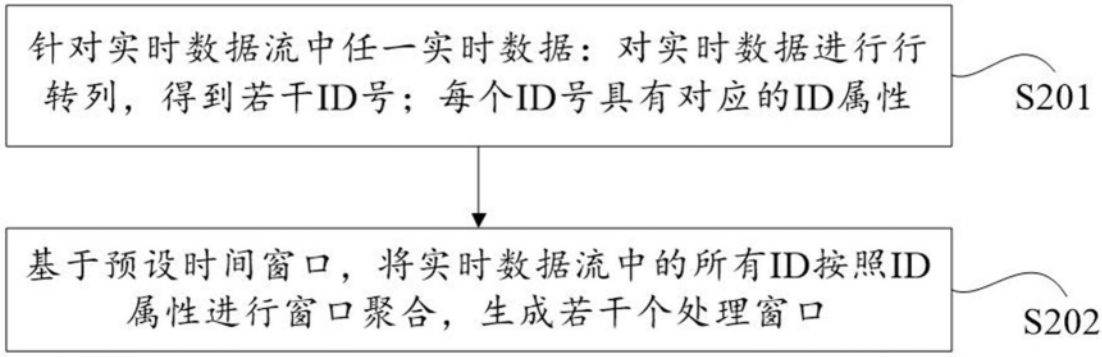

针对实时数据流中任一实时数据：对实时数据进行行转列，得到若干ID号；每个ID号具有对应的ID属性
S201
基于预设时间窗口，将实时数据流中的所有ID按照ID属性进行窗口聚合，生成若干个处理窗口
S202

图 2

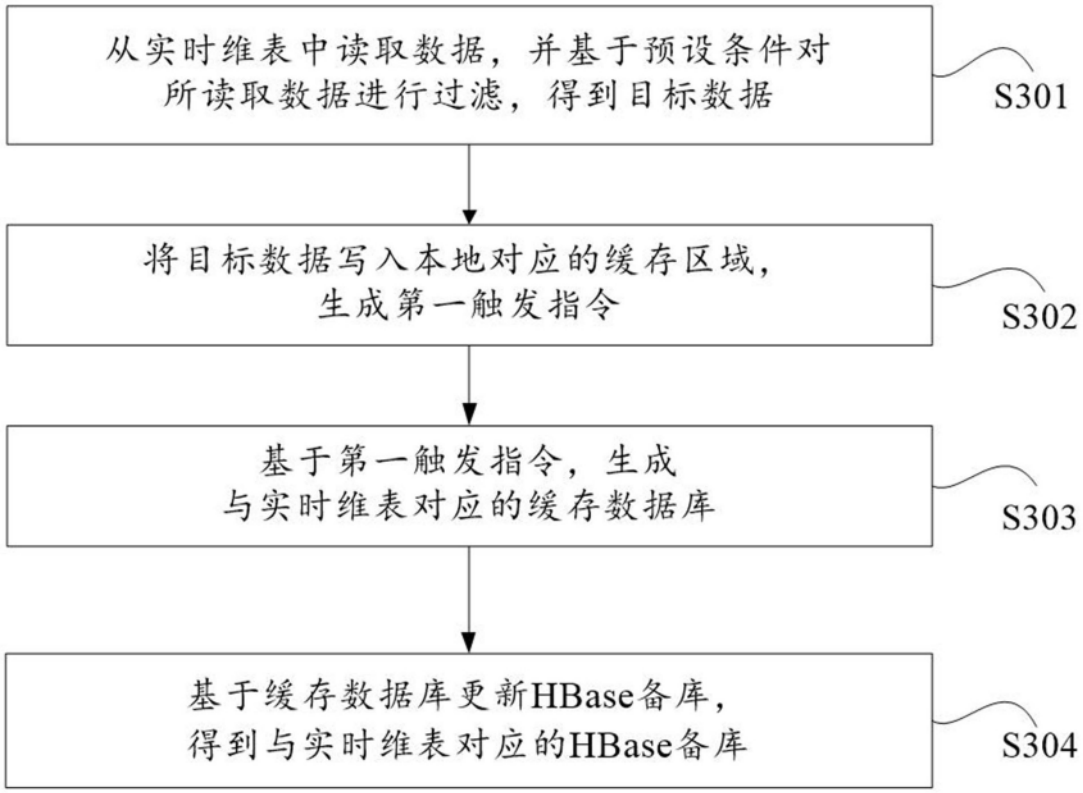

从实时维表中读取数据，并基于预设条件对所读取数据进行过滤，得到目标数据
S301
将目标数据写入本地对应的缓存区域，生成第一触发指令
S302
基于第一触发指令，生成与实时维表对应的缓存数据库
S303
基于缓存数据库更新HBase备库，得到与实时维表对应的HBase备库
S304

图 3

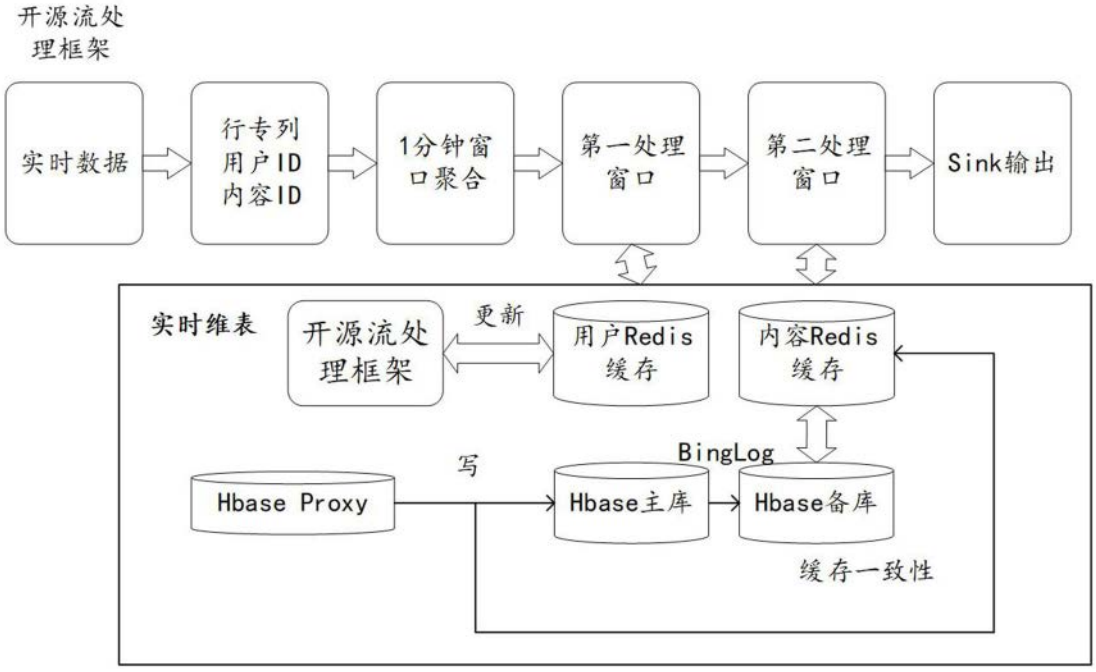

开源流处理框架
实时数据
行专列
用户ID
内容ID
1分钟窗口聚合
第一处理窗口
第二处理窗口
Sink输出
实时维表
开源流处理框架
更新
用户Redis缓存
内容Redis缓存
BingLog
写
Hbase Proxy
Hbase主库
Hbase备库
缓存一致性

图 4

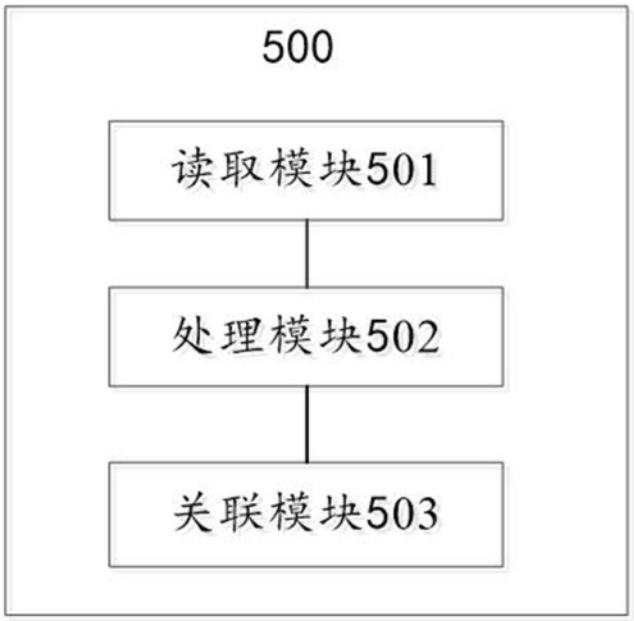


图 5