

(12) 发明专利申请

(10) 申请公布号 CN 102902730 A

(43) 申请公布日 2013. 01. 30

(21) 申请号 201210333370. 8

(22) 申请日 2012. 09. 10

(71) 申请人 新浪网技术(中国)有限公司
地址 100080 北京市海淀区北四环西路 58 号理想国际大厦 20 层

(72) 发明人 全宗峰

(74) 专利代理机构 北京市京大律师事务所
11321

代理人 黄启行 方晓明

(51) Int. Cl.

G06F 17/30(2006. 01)

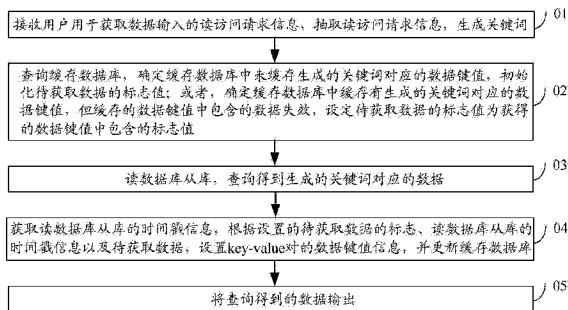
权利要求书 3 页 说明书 12 页 附图 3 页

(54) 发明名称

基于数据缓存的读数据方法及装置

(57) 摘要

本发明公开了一种基于数据缓存的读数据方法及装置。接收用户用于获取数据输入的读访问请求信息,抽取生成关键词;确定缓存数据库中未缓存生成的关键词对应的数据键值,初始化待获取数据的标志值;或者,确定缓存数据库中缓存有生成的关键词对应的数据键值,但缓存的数据键值中包含的数据失效,设定待获取数据的标志值为获得的数据键值中包含的标志值;读数据库从库,查询得到生成的关键词对应的数据;获取读数据库从库的时间戳信息,根据设置的待获取数据的标志、读数据库从库的时间戳信息以及待获取数据,设置键值对的数据键值信息,并更新缓存数据库;将查询得到的数据输出。应用本发明,可以提升数据的缓存效率,优化数据缓存的综合性能。



1. 一种基于数据缓存的读数据方法,该方法包括:
 - 接收用户用于获取数据输入的读访问请求信息,抽取读访问请求信息,生成关键词;
 - 查询缓存数据库,确定缓存数据库中未缓存生成的关键词对应的数据键值,初始化待获取数据的标志值;或者,确定缓存数据库中缓存有生成的关键词对应的数据键值,但缓存的数据键值中包含的数据失效,设定待获取数据的标志值为获得的数据键值中包含的标志值;
 - 读数据库从库,查询得到生成的关键词对应的数据;
 - 获取读数据库从库的时间戳信息,根据设置的待获取数据的标志、读数据库从库的时间戳信息以及待获取数据,设置键值对的数据键值信息,并更新缓存数据库;
 - 将查询得到的数据输出。
2. 根据权利要求 1 所述的方法,其中,在所述接收用户用于获取数据输入的读访问请求信息之前,进一步包括:
 - 获取用户写入的数据信息,抽取用户写入的数据信息,生成对应的关键词;
 - 查询缓存数据库,查找是否存在生成的关键词对应的数据键值,如果存在该数据键值,将缓存数据库中缓存的该数据删除;如果不存在该数据键值,根据生成的关键词以及用户写入的数据信息,生成关键词-数据键值对,并写入数据库主库。
3. 根据权利要求 1 或 2 所述的方法,其中,缓存数据库中,以关键词-数据键值存储数据,所述数据键值包含:时间戳、标志以及数据。
4. 根据权利要求 3 所述的方法,其中,所述初始化待获取数据的标志值包括:
 - 设定待获取数据的标志值为 0。
5. 根据权利要求 4 所述的方法,其中,所述缓存的数据键值中包含的数据失效包括:
 - 解析查询得到的数据键值,获得数据键值中包含的时间戳信息,判断解析时的当前时间戳与获得的时间戳,如果当前时间戳大于获得的时间戳,确定该数据键值中包含的数据失效。
6. 根据权利要求 5 所述的方法,其中,进一步包括:
 - 如果当前时间戳小于或等于获得的时间戳,将查询得到的数据键值中包含的数据作为待获取数据输出。
7. 根据权利要求 5 所述的方法,其中,所述根据设置的待获取数据的标志、读数据库从库的时间戳信息以及待获取数据,设置键值对的数据键值信息包括:
 - 将待获取数据作为数据键值信息中的数据;
 - 将设置的待获取数据的标志加 1,作为数据键值信息中的标志;
 - 根据数据键值信息中的标志、读数据库从库的时间戳信息以及预先设置的时间戳值计算公式,计算更新的缓存数据有效时间,作为数据键值信息中的时间戳。
8. 根据权利要求 7 所述的方法,其中,所述时间戳值计算公式为:

$$\text{time}=\text{time}2+\alpha \times \eta^{\text{flag}}$$
 式中,
 - time 为更新的缓存数据有效时间;
 - time2 为读数据库从库的时间戳;
 - α 为数据基准有效时间;

η 为基准扩展值；
flag' 为更新数据的标志；
flag' = flag+1；
flag 为待获取数据的标志。

9. 根据权利要求 1 或 2 所述的方法,其中,所述在将缓存数据库中缓存的该数据删除后,或,写入数据库主库后,进一步包括:向用户返回写入成功信息。

10. 根据权利要求 1 或 2 所述的方法,其中,进一步包括:

如果查询数据库从库,不存在与生成的关键词对应的数据,向用户返回读数据失败信息。

11. 一种基于数据缓存的读数据装置,其特征在于,该装置包括:读数据模块、缓存数据库模块以及数据库从库模块,其中,

读数据模块,用于接收用户用于获取数据输入的读访问请求信息,抽取读访问请求信息,生成关键词;

缓存数据库模块,用于根据读数据模块生成的关键词进行查询,确定缓存数据库中未缓存生成的关键词对应的数据键值,初始化待获取数据的标志值;或者,确定缓存数据库中缓存有生成的关键词对应的数据键值,但缓存的数据键值中包含的数据失效,设定待获取数据的标志值为获得的数据键值中包含的标志值;

数据库从库模块,用于读数据库从库,查询得到生成的关键词对应的数据,将查询得到的数据输出;

获取读数据库从库的时间戳信息,根据设置的待获取数据的标志、读数据库从库的时间戳信息以及待获取数据,设置键值对的数据键值信息,并更新缓存数据库。

12. 根据权利要求 11 所述的装置,其特征在于,所述缓存数据库模块包括:查询单元、判断单元、初始化单元以及数据键值处理单元,其中,

查询单元,用于根据读数据模块生成的关键词进行查询;

判断单元,判断查询单元是否查询到生成的关键词对应的数据键值,如果是,将查询得到的数据键值输出至数据键值处理单元;否则,向初始化单元输出触发信息;

初始化单元,用于根据接收的触发信息,初始化待获取数据的标志值,向数据库从库模块输出;

数据键值处理单元,用于解析查询得到的数据键值,获得数据键值中包含的时间戳信息,判断解析时的当前时间戳与获得的时间戳,如果当前时间戳小于或等于获得的时间戳,将查询得到的数据键值中包含的数据作为待获取数据输出;如果当前时间戳大于获得的时间戳,设定待获取数据的标志值为获得的数据键值中包含的标志值,向数据库从库模块输出。

13. 根据权利要求 12 所述的装置,其特征在于,进一步包括:写数据模块以及数据库主库模块,其中,

写数据模块,获取用户写入的数据信息,抽取用户写入的数据信息,生成对应的关键词;

缓存数据库模块,进一步用于根据写数据模块生成的关键词进行查询,查找是否存在生成的关键词对应的数据键值,如果存在该数据键值,将缓存数据库中缓存的该数据删除;

如果不存在该数据键值,根据生成的关键词以及用户写入的数据信息,生成关键词-数据键值对,并写入数据库主库模块;

数据库主库模块,用于接收缓存数据库模块写入的关键词-数据键值对,进行存储。

14. 根据权利要求 13 所述的装置,其特征在于,所述缓存数据库模块进一步包括:

写数据处理单元,用于判断查询单元是否根据写数据模块生成的关键词查询到对应的数据键值,如果是,将缓存该数据删除;如果不存在该数据键值,根据生成的关键词以及用户写入的数据信息,生成关键词-数据键值对,并写入数据库主库模块。

15. 根据权利要求 11 至 14 任一项所述的装置,其特征在于,所述根据设置的待获取数据的标志、读数据库从库的时间戳信息以及待获取数据,设置键值对的数据键值信息包括:

将待获取数据作为数据键值信息中的数据;

将设置的待获取数据的标志加 1,作为数据键值信息中的标志;

根据数据键值信息中的标志、读数据库从库的时间戳信息以及预先设置的时间戳值计算公式,计算更新的缓存数据有效时间,作为数据键值信息中的时间戳。

16. 根据权利要求 15 所述的装置,其特征在于,所述时间戳值计算公式为:

$$\text{time}=\text{time}2+\alpha \times \eta^{\text{flag}}$$

式中,

time 为更新的缓存数据有效时间;

time2 为读数据库从库的时间戳;

α 为数据基准有效时间;

η 为基准扩展值;

flag' 为更新数据的标志;

flag' =flag+1;

flag 为待获取数据的标志。

基于数据缓存的读数据方法及装置

技术领域

[0001] 本发明涉及缓存数据库技术,尤其涉及一种基于数据缓存的读数据方法。

背景技术

[0002] 在当今的信息社会,数据库技术及软件已广泛应用到各企业单位。例如,银行系统、税务系统、售票系统、财务系统、人事系统等,在 web 时代,管理诸如用户信息、商品信息、社交信息等浩如烟海的技术资料,缓存数据库技术是目前唯一可行的方案。

[0003] 然而,当包含技术资料的数据量以 T、P 为单位计算时,磁盘 I/O,即应用程序访问物理数据源成为数据库访问效率不可逾越的鸿沟。一方面,需要使用数据库来存储大数据量的技术资料;另一方面,低效的磁盘 I/O 操作限制了数据库的高效运行。

[0004] 为了提高数据库的运行效率,现有技术提出了数据缓存技术。缓存数据介于应用程序和物理数据源之间,其作用是为了降低应用程序对物理数据源访问的频次,由于缓存的介质一般是内存,因而,其读写速度较快,从而提高应用的运行性能。

[0005] 使用缓存不可避免地带来了两个技术问题,一是缓存中的数据与物理数据的一致性;二是缓存数据的时效性(有效期)。特别是目前大型数据库普遍采用集群方式存储数据,并且采用数据库主库以及数据库从库,对数据写操作和数据读操作进行分离,使得缓存数据与物理数据的交互,与单台主机上的内存与磁盘数据交互相比,变得更加复杂和多样,数据同步性和时效性问题也变得更加突出。

[0006] 目前,基于键值(key-value)对的缓存数据库技术有很多,例如,分布式缓存系统(Memcached)、key-value 存储系统(Redis, Remote Dictionary Server)等,实现了缓存数据的高效访问,然而,对于缓存数据的时效性和一致性,全部依赖于数据库运营者预先制定的策略。传统的策略,是在更新数据库主库的同时,使缓存数据库中缓存的数据无效化;接着从数据库从库读取数据,并将读取的数据写入缓存数据库,例如,在 Memcached 中更新缓存数据;在更新缓存数据的同时,根据服务器(Memcached)负载,固定设定缓存数据有效时间,如 24 小时等。

[0007] 然而,一旦数据库读写访问频繁,由于数据库主库与数据库从库不能进行实时同步更新,将导致数据库主库与数据库从库不一致现象会经常发生,并且不一致的时间由于高度依赖于服务器负载(设定的缓存数据有效时间)而不可测,使得缓存数据与数据库中物理数据不一致的现象经常发生;进一步地,在缓存数据的有效期内,或者下一次对数据库主库中不一致的数据进行写访问之前,用户通过缓存机制访问缓存数据库得到的数据与数据库主库中写入的数据是不一致的,一致性较差;而如果将缓存数据失效时间设定的较短,虽然可以提高数据一致性,但由于缓存数据库中数据频繁失效,从而带来缓存命中率降低,影响缓存的效率。因而,如何在数据一致性、时效性和缓存高效率之间取得较好的平衡,是一个亟待解决的问题。

发明内容

[0008] 本发明的实施例提供一种基于数据缓存的读数据方法,提升数据的缓存效率,优化数据缓存的综合性能。

[0009] 本发明的实施例还提供一种基于数据缓存的读数据装置,提升数据的缓存效率,优化数据缓存的综合性能。

[0010] 为达到上述目的,本发明实施例提供的一种基于数据缓存的读数据方法,包括:

[0011] 接收用户用于获取数据输入的读访问请求信息,抽取读访问请求信息,生成关键词;

[0012] 查询缓存数据库,确定缓存数据库中未缓存生成的关键词对应的数据键值,初始化待获取数据的标志值;或者,确定缓存数据库中缓存有生成的关键词对应的数据键值,但缓存的数据键值中包含的数据失效,设定待获取数据的标志值为获得的数据键值中包含的标志值;

[0013] 读数据库从库,查询得到生成的关键词对应的数据;

[0014] 获取读数据库从库的时间戳信息,根据设置的待获取数据的标志、读数据库从库的时间戳信息以及待获取数据,设置键值对的数据键值信息,并更新缓存数据库;

[0015] 将查询得到的数据输出。

[0016] 其中,在所述接收用户用于获取数据输入的读访问请求信息之前,进一步包括:

[0017] 获取用户写入的数据信息,抽取用户写入的数据信息,生成对应的关键词;

[0018] 查询缓存数据库,查找是否存在生成的关键词对应的数据键值,如果存在该数据键值,将缓存数据库中缓存的该数据删除;如果不存在该数据键值,根据生成的关键词以及用户写入的数据信息,生成关键词-数据键值对,并写入数据库主库。

[0019] 其中,缓存数据库中,以关键词-数据键值存储数据,所述数据键值包含:时间戳、标志以及数据。

[0020] 其中,所述初始化待获取数据的标志值包括:

[0021] 设定待获取数据的标志值为0。

[0022] 其中,所述缓存的数据键值中包含的数据失效包括:

[0023] 解析查询得到的数据键值,获得数据键值中包含的时间戳信息,判断解析时的当前时间戳与获得的时间戳,如果当前时间戳大于获得的时间戳,确定该数据键值中包含的数据失效。

[0024] 其中,进一步包括:

[0025] 如果当前时间戳小于或等于获得的时间戳,将查询得到的数据键值中包含的数据作为待获取数据输出。

[0026] 其中,所述根据设置的待获取数据的标志、读数据库从库的时间戳信息以及待获取数据,设置键值对的数据键值信息包括:

[0027] 将待获取数据作为数据键值信息中的数据;

[0028] 将设置的待获取数据的标志加1,作为数据键值信息中的标志;

[0029] 根据数据键值信息中的标志、读数据库从库的时间戳信息以及预先设置的时间戳值计算公式,计算更新的缓存数据有效时间,作为数据键值信息中的时间戳。

[0030] 其中,所述时间戳值计算公式为:

[0031] $time=time2+\alpha \times \eta^{flag}$

- [0032] 式中，
- [0033] time 为更新的缓存数据有效时间；
- [0034] time2 为读数据库从库的时间戳；
- [0035] α 为数据基准有效时间；
- [0036] η 为基准扩展值；
- [0037] flag' 为更新数据的标志；
- [0038] $\text{flag}' = \text{flag} + 1$ ；
- [0039] flag 为待获取数据的标志。
- [0040] 其中，所述在将缓存数据库中缓存的该数据删除后，或，写入数据库主库后，进一步包括：向用户返回写入成功信息。
- [0041] 其中，进一步包括：
- [0042] 如果查询数据库从库，不存在与生成的关键词对应的数据，向用户返回读数据失败信息。
- [0043] 一种基于数据缓存的读数据装置，该装置包括：读数据模块、缓存数据库模块以及数据库从库模块，其中，
- [0044] 读数据模块，用于接收用户用于获取数据输入的读访问请求信息，抽取读访问请求信息，生成关键词；
- [0045] 缓存数据库模块，用于根据读数据模块生成的关键词进行查询，确定缓存数据库中未缓存生成的关键词对应的数据键值，初始化待获取数据的标志值；或者，确定缓存数据库中缓存有生成的关键词对应的数据键值，但缓存的数据键值中包含的数据失效，设定待获取数据的标志值为获得的数据键值中包含的标志值；
- [0046] 数据库从库模块，用于读数据库从库，查询得到生成的关键词对应的数据，将查询得到的数据输出；
- [0047] 获取读数据库从库的时间戳信息，根据设置的待获取数据的标志、读数据库从库的时间戳信息以及待获取数据，设置键值对的数据键值信息，并更新缓存数据库。
- [0048] 所述缓存数据库模块包括：查询单元、判断单元、初始化单元以及数据键值处理单元，其中，
- [0049] 查询单元，用于根据读数据模块生成的关键词进行查询；
- [0050] 判断单元，判断查询单元是否查询到生成的关键词对应的数据键值，如果是，将查询得到的数据键值输出至数据键值处理单元；否则，向初始化单元输出触发信息；
- [0051] 初始化单元，用于根据接收的触发信息，初始化待获取数据的标志值，向数据库从库模块输出；
- [0052] 数据键值处理单元，用于解析查询得到的数据键值，获得数据键值中包含的时间戳信息，判断解析时的当前时间戳与获得的时间戳，如果当前时间戳小于或等于获得的时间戳，将查询得到的数据键值中包含的数据作为待获取数据输出；如果当前时间戳大于获得的时间戳，设定待获取数据的标志值为获得的数据键值中包含的标志值，向数据库从库模块输出。
- [0053] 进一步包括：写数据模块以及数据库主库模块，其中，
- [0054] 写数据模块，获取用户写入的数据信息，抽取用户写入的数据信息，生成对应的关

关键词；

[0055] 缓存数据库模块,进一步用于根据写数据模块生成的关键词进行查询,查找是否存在生成的关键词对应的数据键值,如果存在该数据键值,将缓存数据库中缓存的该数据删除;如果不存在该数据键值,根据生成的关键词以及用户写入的数据信息,生成关键词-数据键值对,并写入数据库主库模块;

[0056] 数据库主库模块,用于接收缓存数据库模块写入的关键词-数据键值对,进行存储。

[0057] 所述缓存数据库模块进一步包括:

[0058] 写数据处理单元,用于判断查询单元是否根据写数据模块生成的关键词查询到对应的数据键值,如果是,将缓存该数据删除;如果不存在该数据键值,根据生成的关键词以及用户写入的数据信息,生成关键词-数据键值对,并写入数据库主库模块。

[0059] 所述根据设置的待获取数据的标志、读数据库从库的时间戳信息以及待获取数据,设置键值对的数据键值信息包括:

[0060] 将待获取数据作为数据键值信息中的数据;

[0061] 将设置的待获取数据的标志加1,作为数据键值信息中的标志;

[0062] 根据数据键值信息中的标志、读数据库从库的时间戳信息以及预先设置的时间戳值计算公式,计算更新的缓存数据有效时间,作为数据键值信息中的时间戳。

[0063] 所述时间戳值计算公式为:

[0064] $time=time2+\alpha \times \eta^{flag'}$

[0065] 式中,

[0066] time 为更新的缓存数据有效时间;

[0067] time2 为读数据库从库的时间戳;

[0068] α 为数据基准有效时间;

[0069] η 为基准扩展值;

[0070] $flag'$ 为更新数据的标志;

[0071] $flag' = flag + 1$;

[0072] flag 为待获取数据的标志。

[0073] 由上述技术方案可见,本发明实施例提供了一种基于数据缓存的读数据方法及装置,接收用户用于获取数据输入的读访问请求信息,抽取读访问请求信息,生成关键词;查询缓存数据库,确定缓存数据库中未缓存生成的关键词对应的数据键值,初始化待获取数据的标志值;或者,确定缓存数据库中缓存有生成的关键词对应的数据键值,但缓存的数据键值中包含的数据失效,设定待获取数据的标志值为获得的数据键值中包含的标志值;读数据库从库,查询得到生成的关键词对应的数据;获取读数据库从库的时间戳信息,根据设置的待获取数据的标志、读数据库从库的时间戳信息以及待获取数据,设置键值对的数据键值信息,并更新缓存数据库;将查询得到的数据输出。这样,通过为缓存数据库中缓存的数据设置时间戳以及标志,在缓存的数据键值中包含的数据失效后,对数据键值中包含的标志进行更新,并相应更新数据键值中包含的时间戳,避免了频繁对缓存数据库进行更新,提升数据的缓存效率,使数据一致性、时效性和缓存高效率之间取得较好的平衡,优化了数据缓存的综合性能。

附图说明

[0074] 为了更清楚地说明本发明实施例或现有技术中的技术方案,以下将对实施例或现有技术描述中所需要使用的附图作简单地介绍。显而易见地,以下描述中的附图仅仅是本发明的一些实施例,对于本领域普通技术人员而言,还可以根据这些附图所示实施例得到其它的实施例及其附图。

[0075] 图 1 为本发明实施例基于数据缓存的读数据方法流程示意图。

[0076] 图 2 为本发明实施例基于数据缓存的读数据方法具体流程示意图。

[0077] 图 3 为本发明实施例基于数据缓存的写数据方法具体流程示意图。

[0078] 图 4 为本发明实施例基于数据缓存的读数据装置结构示意图。

具体实施方式

[0079] 以下将结合附图对本发明各实施例的技术方案进行清楚、完整的描述,显然,所描述的实施例仅仅是本发明的一部分实施例,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的前提下所得到的所有其它实施例,都属于本发明所保护的范围。

[0080] 现有基于数据缓存的读数据方法,根据用户输入的关键词,查询缓存数据库,如果缓存数据库中没有该关键词对应的数据键值,则从数据库从库读取该关键词对应的数据,并将读取的数据写入缓存数据库,例如,在 Memcached 中更新缓存数据,在更新缓存数据的同时,根据服务器(Memcached)负载,固定设定缓存数据有效时间,如 24 小时,以作为用户后续从缓存数据库中读取该数据的有效期。但该固定设置缓存数据有效时间的方式,如果缓存数据有效时间值设置较长,容易导致用户通过缓存机制访问得到的数据与数据库主库中写入的数据的一致性较差;而如果将缓存数据有效时间值设定的较短,虽然可以提高用户通过缓存机制访问得到的数据与数据库主库中写入的数据的一致性,但由于缓存数据库中数据需要频繁更新,根据关键词查询缓存数据库,从而带来缓存命中率降低,在查询失败后,需要再从数据库从库中读取数据,并将读取的数据写入缓存数据库中,影响缓存的性能,也降低了读数据的效率。

[0081] 本发明实施例中,提供了一种缓存数据库中读数据的技术,通过增加一个单时间戳,可有效解决现有缓存技术中数据时效性和一致性的问题。具体来说,在缓存数据库缓存的数据中设置时间戳(time)信息以及标志(flag)信息,通过获取读访问请求信息;抽取读访问请求信息,形成关键词(key);根据关键词查询缓存数据库,如果存在该关键词对应的数据键值(value),则根据键值判断该关键词对应的数据是否失效,如果没有失效,则将该关键词对应的数据输出给用户;如果该关键词对应的数据已经失效、或者,缓存数据库中不存在该关键词对应的数据键值,则从数据库从库中读取该关键词对应的数据,并设置该关键词对应的数据键值,即生成 key-value 对,根据设置的数据键值对更新缓存数据库。这样,可以充分考虑数据时效性和缓存访问的高效性,在数据一致性、时效性和访问高效性之间取得了较好的平衡。

[0082] 以下以 Memcached 作为缓存数据库,来说明本发明的读写过程。Memcached 是一个高性能的分布式内存对象缓存系统,用于动态 Web 应用以减轻数据库负载。通过在内存中

缓存数据和对象来减少读取数据库的次数,从而提供动态、数据库驱动网站的速度。

[0083] 所应说明的是,本发明实施例不仅仅只应用于以 Memcached 作为缓存数据库,还可以应用于采用其它缓存技术作为缓存数据库的情形,例如,Redis 等。

[0084] 图 1 为本发明实施例基于数据缓存的读数据方法流程示意图。参见图 1,该流程包括:

[0085] 步骤 01,接收用户用于获取数据输入的读访问请求信息,抽取读访问请求信息,生成关键词;

[0086] 步骤 02,查询缓存数据库,确定缓存数据库中未缓存生成的关键词对应的数据键值,初始化待获取数据的标志值;或者,确定缓存数据库中缓存有生成的关键词对应的数据键值,但缓存的数据键值中包含的数据失效,设定待获取数据的标志值为获得的数据键值中包含的标志值;

[0087] 步骤 03,读数据库从库,查询得到生成的关键词对应的数据;

[0088] 步骤 04,获取读数据库从库的时间戳信息,根据设置的待获取数据的标志、读数据库从库的时间戳信息以及待获取数据,设置 key-value 对的数据键值信息,并更新缓存数据库;

[0089] 步骤 05,将查询得到的数据输出。

[0090] 较佳地,本发明实施例中,在步骤 01 之前,还可以进一步包括:

[0091] A01,获取用户写入的数据信息,抽取用户写入的数据信息,生成对应的关键词;

[0092] A02,查询缓存数据库,查找是否存在生成的关键词对应的数据键值,如果存在该数据键值,将缓存数据库中缓存的该数据删除;如果不存在该数据键值,根据生成的关键词以及用户写入的数据信息,生成关键词-数据键值对,并写入数据库主库。

[0093] 实际应用中,在将缓存数据库中缓存的该数据删除后,以及写入数据库主库后,还可以进一步向用户返回写入成功信息。

[0094] 下面基于图 1,对本发明实施例作进一步详细描述。

[0095] 图 2 为本发明实施例基于数据缓存的读数据方法具体流程示意图。本发明实施例提供了用户通过缓存机制读取数据库的方法,参见图 2,该流程包括:

[0096] 步骤 11,接收用户用于获取数据输入的读访问请求信息;

[0097] 本步骤中,当用户需要获取数据,即从数据库中查询数据时,也就是从缓存数据库或数据库从库中,读取相应数据,输入查询数据的查询语句,即读访问请求信息,例如,如果用户需要搜索数据库以获取某种技术资料,举例来说,需要从数据库中读取与 technologyinformation 相关的数据,则可以在数据库的读数据窗口界面中输入读访问请求信息:technologyinformation。

[0098] 本发明实施例中,设要获取的数据为变量 data。

[0099] 步骤 12,抽取读访问请求信息,生成关键词;

[0100] 本步骤中,根据用户输入的读访问请求信息:technology information,进行抽取操作,生成关键词(key)为 tech_info。例如,可以根据预先设置的关键词生成软件,对用户输入的信息进行处理,生成关键词,关于抽取读访问请求信息,生成关键词的流程,具体可参见相关技术文献,在此不再赘述。

[0101] 步骤 13,查询缓存数据库,判断缓存数据库中是否存在生成的关键词对应的数据

键值,如果是,执行步骤 14,否则,初始化待获取数据的标志值,执行步骤 15;

[0102] 本步骤中,在需要查询待获取数据时,首先从缓存数据库中进行查询,如果缓存数据库中存储有关键词对应的数据键值,则提取数据键值中包含的数据,直接向用户返回查询得到的数据,则可以避免从数据库从库中查询待获取数据,从而有效提高读数据的速率和效率。

[0103] 本发明实施例中,在缓存数据库中查找是否存在key为tach_info的数据键值,如果存在该数据键值,执行步骤 14,如果不存在该数据键值,初始化待获取数据的标志值,例如,设定待获取数据的标志值(flag)为0为某一预先设置的常数,执行步骤 15。

[0104] 本发明实施例中,缓存数据库中,以关键词-数据键值存储数据,其中,关键词可以根据实际应用中的需求,利用关键词提取技术,对大量的查询语句进行提取得到,当然,实际应用中,也可以由数据库维护者根据用户需求设定;数据键值(value)包含:时间戳(time)、标志(flag)以及数据(data),即存储的数据中,携带有关键词信息、时间戳信息以及标志信息,与现有采用关键词-数据键值存储数据不同的是,本发明实施例中,其时间戳值可动态调整,此外,还增加了用于作为动态调整缓存数据有效时间的计算因子,即标志,其中,

[0105] 时间戳,用于标识缓存数据有效时间;

[0106] 标志,用于作为动态调整缓存数据有效时间的计算因子。

[0107] 步骤 14,解析查询得到的数据键值,获得数据键值中包含的时间戳信息,判断解析时的当前时间戳与获得的时间戳,如果当前时间戳小于或等于获得的时间戳,将查询得到的数据键值中包含的数据作为待获取数据,执行步骤 17;如果当前时间戳大于获得的时间戳,设定待获取数据的标志值为获得的数据键值中包含的标志值,执行步骤 15;

[0108] 本步骤中,解析查询得到的数据键值,得到时间戳(time0)、标志(flag0)和数据(data0),设当前时间戳为time1,将得到的时间戳与当前时间戳进行比较,即比较time1与time0的大小:

[0109] 如果time1小于或等于time0,即小于或等于缓存数据有效时间,说明缓存数据库中当前缓存的数据data0在有效期内,将data0赋值给data,即将该缓存的数据作为待获取数据,执行步骤 17;如果time1大于time0,即大于缓存数据有效时间,说明缓存数据库中当前缓存的数据data0已超出有效期,数据无效,因而,查询得到的该数据不能作为待获取数据,设定待获取数据的标志值为获得的数据键值中包含的标志值,即将待获取数据的标志值flag设定为flag0,执行步骤 15;

[0110] 步骤 15,读数据库从库,查询得到生成的关键词对应的数据,将该数据作为待获取数据;

[0111] 本步骤中,从数据库从库中查询是否存在与生成的关键词相匹配的关键词,如果存在相匹配的关键词,获取该匹配的关键词对应的数据data1,将data1赋值给data;如果不存在,则结束读数据流程,向用户返回读数据失败信息。

[0112] 本发明实施例中,数据库从库中缓存的数据格式与现有技术相同,采用关键词-数据作为存储格式。

[0113] 步骤 16,获取读数据库从库的时间戳信息,根据设置的待获取数据的标志、读数据库从库的时间戳信息以及待获取数据,设置key-value对的数据键值信息,并更新缓存数

数据库；

[0114] 本步骤中,通过设定缓存 key-value 对的 value 信息,将 key-value 对存储到缓存数据库中,从而实现对缓存数据库中该数据缓存数据有效时间的更新。

[0115] 设读数据库从库的时间戳为 time2,其中,设定 value 中的时间戳值计算公式为：

$$[0116] \quad \text{time} = \text{time2} + \alpha \times \eta^{\text{flag}'}$$

[0117] 式中,

[0118] time 为更新的缓存数据有效时间；

[0119] time2 为读数据库从库的时间戳；

[0120] α 为数据基准有效时间；

[0121] η 为基准扩展值；

[0122] flag' 为更新数据的标志；

$$[0123] \quad \text{flag}' = \text{flag} + 1$$

[0124] flag 为待获取数据的标志。

[0125] 其中,

[0126] $\alpha \times \eta^{\text{flag}'}$ 也成为有效时间间隔,设置时,主要考虑对于读写改变频率快的数据,其有效时间间隔较短;对于读写改变频率慢的数据,其有效时间间隔较长。本发明实施例中,控制有效时间间隔长短的量是标志 flag 的值,如果缓存中数据经常处于无效状态,表明读写改变频率较快,通过将该数据对应的标志 flag 的值设置为 0,从而可以使得其有效时间间隔较长。

[0127] 数据基准有效时间,由缓存管理者根据实际情况设定,可以是经验时间,例如,10s、30s 等,只要满足对于单位时间内访问次数较多的数据,其对应的数据基准有效时间较小,而对于单位时间内访问次数较少的数据,其对应的数据基准有效时间较大即可,当然,实际应用中,也可以根据动态统计的数据访问频率,适时调整对应的数据基准有效时间；

[0128] 基准扩展值是每次设定数据基准有效时间间隔的倍数,可以由缓存管理者根据实际情况设定,为正值,一般大于 1,例如,取为 2 或其他常数值；

[0129] 本发明实施例中,在每次从缓存数据库中读取的缓存数据失效后,将待获取数据的标志值 flag 设定为 flag0,在设置 key-value 对的数据键值信息时,更新待获取数据的标志值 flag,即将 flag0 执行加 1 作为更新的待获取数据的标志值 flag。

[0130] 实际应用中,设定的时间戳值公式也可以采用其他公式,例如,改变上式中的指数值, $\text{time} = \text{time2} + \alpha \times \eta^{1+\text{flag}'}$;再例如,还可以设置为: $\text{time} = \text{time2} + \alpha \times \eta \times \text{flag}'$ 等。

[0131] 这样,缓存的数据的 value 由时间戳 time、标志 flag 和数据 data 拼接而成,键值为生成的关键词。

[0132] 举例来说,假设有一客户信息数据库,包含缓存数据库、数据库主库以及数据库从库,用户需要查找客户名为 angle 的信息,假设客户信息数据库使用 Memcached 做数据缓存库,用户名 angle 对应的关键词 key 为客户名自身,即 angle,用户输入读访问请求信息: angle,客户信息数据库先在缓存数据库中查找关键词 angle 对应的数据键值。

[0133] 如果在缓存数据库中存在 key 为 angle 的数据键值,进行解析,其 value 为 2012 : 06 : 1200 : 00 : 00 ; 1 ; tel : xxxx,其中,时间戳 time0 为 2012 : 06 : 1200 : 00 : 00、标志 flag0 为 1、data0 为 tel : xxxx。

[0134] 假设解析时的当前时间戳 $time1$ 为 2012:08:2100:00:00, 则由于 $time1$ 大于 $time0$, 说明数据 $data0$ 已超出有效期, 数据无效, 则读数据库从库, 查找关键词为 $angle$ 的信息, 获取 $angle$ 对应的数据 $data$ 为 $tel:aaxx$, 将 $data$ 返回给用户, 并重设 Memcached 的缓存数据。

[0135] 获取读数据库从库的时间戳(也可以是重设 Memcached 的缓存数据的时间戳) $time2$, 例如, $time2$ 为 2012:08:2100:01:00。由于重设 Memcached 的缓存数据的时间戳与解析时的当前时间戳(也可以是访问数据库从库获得数据的时间戳)不一致, 因而, $time2$ 与 $time1$ 不同。设定待获取数据(待更新的缓存数据键值) $flag=flag0+1=2$, 取数据基准有效时间为 30s, 则可计算得到 Memcached 中缓存数据有效期(缓存数据有效时间)为:

[0136] $time=time2+\alpha \times \eta^{flag}=2012:08:2100:03:00$

[0137] 如此, 可设定 key 为 $angle$ 的数据在 Memcached 中保存的 $value$ 为 2012:08:2100:03:00;2;tel:aaxx。

[0138] 如果解析时的当前时间戳 $time1$ 为 2012:06:1100:00:00, 由于 $time1$ 小于 $time0$, Memcached 中数据仍然有效, 此时, 直接将 $data$ 赋值为 $tel:xxxx$, 并返回给用户即可。

[0139] 步骤 17, 将待获取数据返回给用户。

[0140] 图 3 为本发明实施例基于数据缓存的写数据方法具体流程示意图。本实施例提供了用户通过缓存机制写数据库的方法, 写完数据库主库时, 更新缓存 Memcached 中的数据, 参见图 3, 该流程包括:

[0141] 步骤 21, 获取用户写入的数据信息;

[0142] 本步骤中, 可以在数据库的写数据窗口界面中输入数据信息。

[0143] 步骤 22, 抽取用户写入的数据信息, 生成对应的关键词;

[0144] 本步骤中, 如果用户需要往数据库中写入数据, 例如, 需要写入某种技术资料 $technology\ information$, 则可以输入 $technology\ information$ 及其具体内容, 数据库抽取用户写入的数据信息, 生成关键词为 $tech_info$, 设定缓存 key - $value$ 对的 key 为 $tech_info$ 。

[0145] 本发明实施例中, key 值是根据用户访问数据库的关键词获取的, 例如, 访问 Mysql 中的数据时需要主键, 可以根据主键设定 Memcached 中 key 。而如何设定 key , 可以由 Memcached 管理员自定, 可以将主键作为 key , 也可以将主键与其他信息作为 key 。本发明主要是对 $value$ 值做设定, 与 key 无关。

[0146] 步骤 23, 查询缓存数据库, 查找是否存在 key 为 $tech_info$ 对应的数据键值, 如果存在该数据键值, 执行步骤 24, 如果不存在该数据键值, 执行步骤 25;

[0147] 步骤 24, 将缓存数据库中缓存的该数据删除, 执行步骤 26;

[0148] 步骤 25, 根据生成的关键词以及用户写入的数据信息, 生成关键词-数据键值对, 并写入数据库主库;

[0149] 步骤 26, 向用户返回写入成功信息。

[0150] 由上述可见, 本发明实施例基于数据缓存的读写数据方法, 在原始数据的基础上, 通过额外增加时间戳和标志 $flag$, 时间戳用来标志数据的有效期, 即缓存数据有效时间, 标志用来计算有效时间, 即作为动态调整缓存数据有效时间的计算因子; 并设定更新数据的

时间戳为当前时间与有效时间间隔之和,其中,有效时间间隔与数据基准有效时间、基准扩展值以及标志有关,在设定有效时间间隔时,利用了如下思想:读写改变频率快的数据,其有效时间间隔较短;读写改变频率慢的数据,其有效时间间隔较长,控制有效时间间隔长短的量是标志 flag 的值。这样,对于读写改变频率快的数据,更新频率较快,对于读写改变频率慢的数据,更新频率慢,从而在保证数据读写一致性的基础上,降低了由于缓存数据库中数据频繁失效导致的缓存命中率降低,影响缓存的效率的问题,较好地达到了数据一致性和访问缓存的高效性之间的平衡。

[0151] 举例来说,假如有 key0 及其对应的值 value0,如果对 key0,长时间没有进行读操作,即不改变 key0 的值,随着时间的增加,其 value0 中对应的 flag 值会随着 value0 的失效而逐渐增加,value0 每失效一次,flag 增加 1,即有效时间间隔变为前一次的基准扩展值的倍数。本发明实施例中,由于有效时间间隔呈指数增长,因而,只需要很少次数的读取数据库,即可将有效时间变的很长。设预定时间为 t,则 value 的失效次数,即在预定的时间内,需要对缓存数据进行读访问以更新的次数为:

[0152] $\log(t/\text{数据基准有效时间})/\log(\text{基准扩展值})$ 。

[0153] 因而,与传统的策略相比,需要对缓存数据进行读访问以更新的次数要少,少的次数为:

[0154] $\log(t/\text{数据基准有效时间})/\log(\text{基准扩展值})-1$ 。

[0155] 如果 key0 对应的数据写操作频繁,则其有效时间间隔约为数据基准有效时间,与传统的策略相比,当基准有效时间设定较小时,数据不一致的持续时间也很小。

[0156] 本发明实施例中,在每次数据失效后,Memcached 中如果仍然有该数据,则 flag 并没有设定为 0。只有在写数据库主库时,将数据删除或者在 Memcached 自动清理过期数据时,将数据删除后,Memcached 中不存在关键字为 key 的数据键值时,将 flag 设定为 0,若不存在该数据项,将 flag 设定为 0。

[0157] 再举一具体例子,设用于统计的时间为 1 小时,传统策略中设定有效时间分别为 100s 和 10s,其中,失效时间相当于本发明实施例的数据基准有效时间,相应地,设本策略中数据基准有效时间也分别为 100s 和 10s,基准扩展值为 2,则在 1 小时内,传统策略与本发明策略的数据对比分别如表 1 和表 2 所示。其中,表 1 设定的有效时间为 100s,表 2 设定的有效时间为 10s。

[0158] 表 1

[0159]

	传统策略	本发明策略
对数据库更新次数	36	6
数据不一致时间上限	100	1600
数据不一致时间下限	100	100
数据不一致平均时间	100	

[0160] 表 2

[0161]

	传统策略	本发明策略
对数据库更新次数	360	11
数据不一致时间上限 (s)	10	1052
数据不一致时间下限 (s)	10	10
数据不一致平均时间 (s)	10	327

[0162] 由表 1 和表 2 可见,对于有效时间为 100s 的情形,在 1 小时内,传统策略需要对数据库进行 36 次更新,而本发明策略只需对数据库进行 6 次更新;对于有效时间为 10s 的情形,在 1 小时内,传统策略需要对数据库进行 360 次更新,而本发明策略只需对数据库进行 11 次更新。这样,有效降低了数据库频繁更新的次数,从而降低了由于缓存数据库中数据频繁失效导致的缓存命中率降低的问题,有利于降低数据库所在服务器的负载;同时,也能够较好地保证数据的一致性,在数据一致性和访问缓存的高效性之间,较好地达到了平衡。

[0163] 图 4 为本发明实施例基于数据缓存的读数据装置结构示意图。参见图 4,该装置包括:读数据模块、缓存数据库模块以及数据库从库模块,其中,

[0164] 读数据模块,用于接收用户用于获取数据输入的读访问请求信息,抽取读访问请求信息,生成关键词;

[0165] 缓存数据库模块,用于根据读数据模块生成的关键词进行查询,确定缓存数据库中未缓存生成的关键词对应的数据键值,初始化待获取数据的标志值;或者,确定缓存数据库中缓存有生成的关键词对应的数据键值,但缓存的数据键值中包含的数据失效,设定待获取数据的标志值为获得的数据键值中包含的标志值;

[0166] 数据库从库模块,用于读数据库从库,查询得到生成的关键词对应的数据,将查询得到的数据输出;

[0167] 获取读数据库从库的时间戳信息,根据设置的待获取数据的标志、读数据库从库的时间戳信息以及待获取数据,设置 key-value 对的数据键值信息,并更新缓存数据库。

[0168] 其中,缓存数据库模块包括:查询单元、判断单元、初始化单元以及数据键值处理单元(图中未示出),其中,

[0169] 查询单元,用于根据读数据模块生成的关键词进行查询;

[0170] 判断单元,判断查询单元是否查询到生成的关键词对应的数据键值,如果是,将查询得到的数据键值输出至数据键值处理单元;否则,向初始化单元输出触发信息;

[0171] 初始化单元,用于根据接收的触发信息,初始化待获取数据的标志值,向数据库从库模块输出;

[0172] 数据键值处理单元,用于解析查询得到的数据键值,获得数据键值中包含的时间戳信息,判断解析时的当前时间戳与获得的时间戳,如果当前时间戳小于或等于获得的时间戳,将查询得到的数据键值中包含的数据作为待获取数据输出;如果当前时间戳大于获得的时间戳,设定待获取数据的标志值为获得的数据键值中包含的标志值,向数据库从库模块输出。

- [0173] 较佳地,该装置还可以进一步包括:写数据模块以及数据库主库模块,其中,
- [0174] 写数据模块,获取用户写入的数据信息,抽取用户写入的数据信息,生成对应的关键词;
- [0175] 缓存数据库模块,进一步用于根据写数据模块生成的关键词进行查询,查找是否存在生成的关键词对应的数据键值,如果存在该数据键值,将缓存数据库中缓存的该数据删除;如果不存在该数据键值,根据生成的关键词以及用户写入的数据信息,生成关键词-数据键值对,并写入数据库主库模块;
- [0176] 数据库主库模块,用于接收缓存数据库模块写入的关键词-数据键值对,进行存储。
- [0177] 较佳地,缓存数据库模块进一步包括:
- [0178] 写数据处理单元,用于判断查询单元是否根据写数据模块生成的关键词查询到对应的数据键值,如果是,将缓存该数据删除;如果不存在该数据键值,根据生成的关键词以及用户写入的数据信息,生成关键词-数据键值对,并写入数据库主库模块。
- [0179] 其中,
- [0180] 根据设置的待获取数据的标志、读数据库从库的时间戳信息以及待获取数据,设置键值对的数据键值信息包括:
- [0181] 将待获取数据作为数据键值信息中的数据;
- [0182] 将设置的待获取数据的标志加1,作为数据键值信息中的标志;
- [0183] 根据数据键值信息中的标志、读数据库从库的时间戳信息以及预先设置的时间戳值计算公式,计算更新的缓存数据有效时间,作为数据键值信息中的时间戳。
- [0184] 本发明实施例中,时间戳值计算公式为:
- [0185] $time=time2+\alpha \times \eta^{flag'}$
- [0186] 式中,
- [0187] time 为更新的缓存数据有效时间;
- [0188] time2 为读数据库从库的时间戳;
- [0189] α 为数据基准有效时间;
- [0190] η 为基准扩展值;
- [0191] flag' 为更新数据的标志;
- [0192] $flag' = flag+1$;
- [0193] flag 为待获取数据的标志。
- [0194] 显然,本领域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,倘若对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,则本发明也包含这些改动和变型在内。

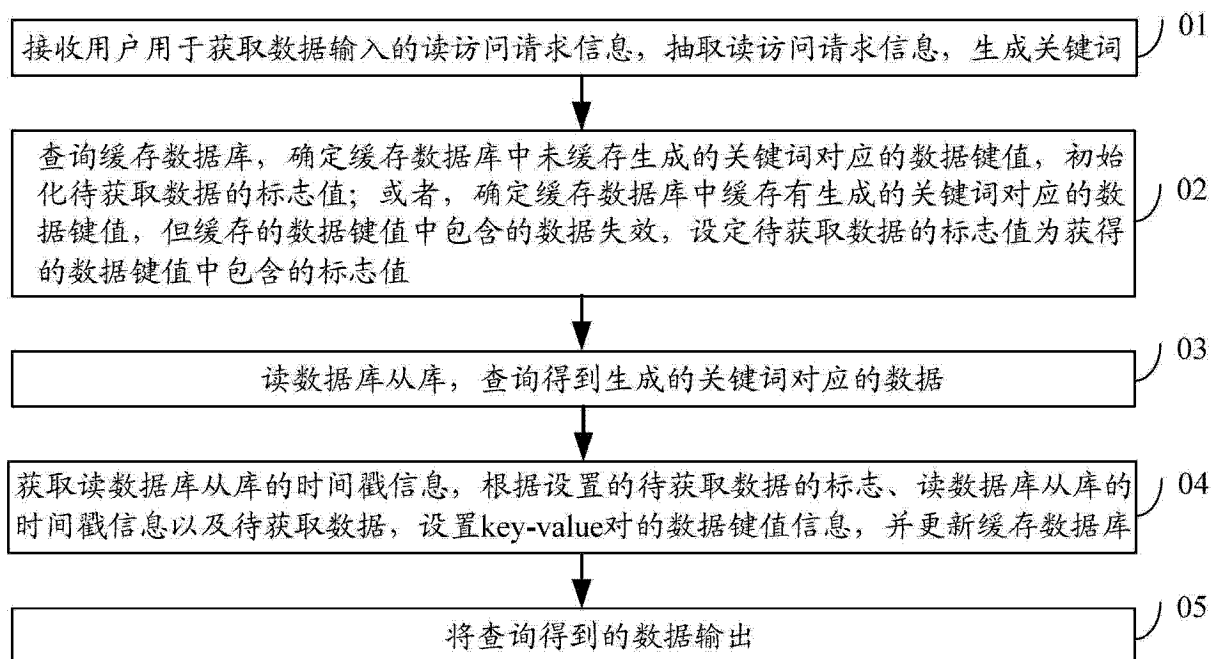


图 1

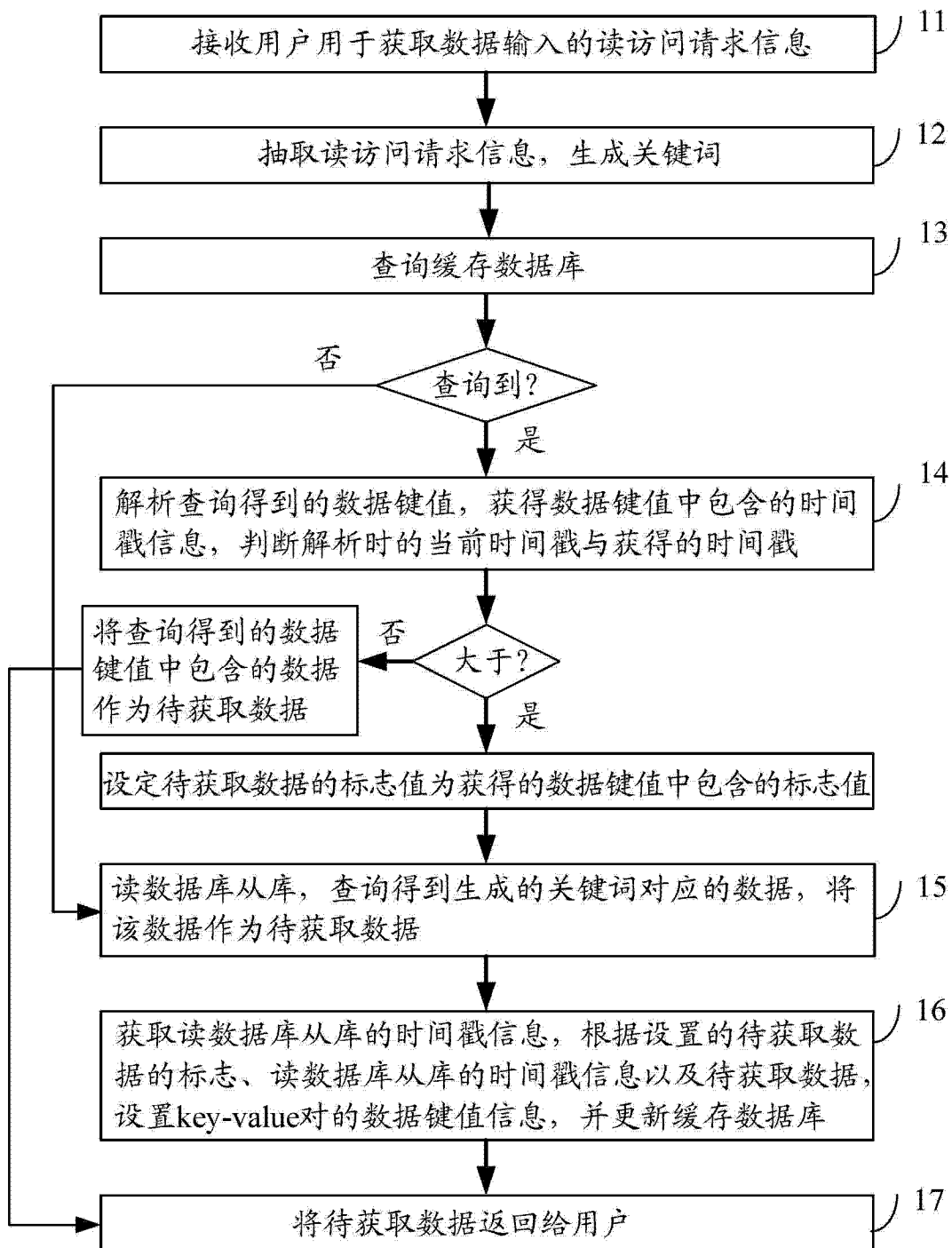


图 2

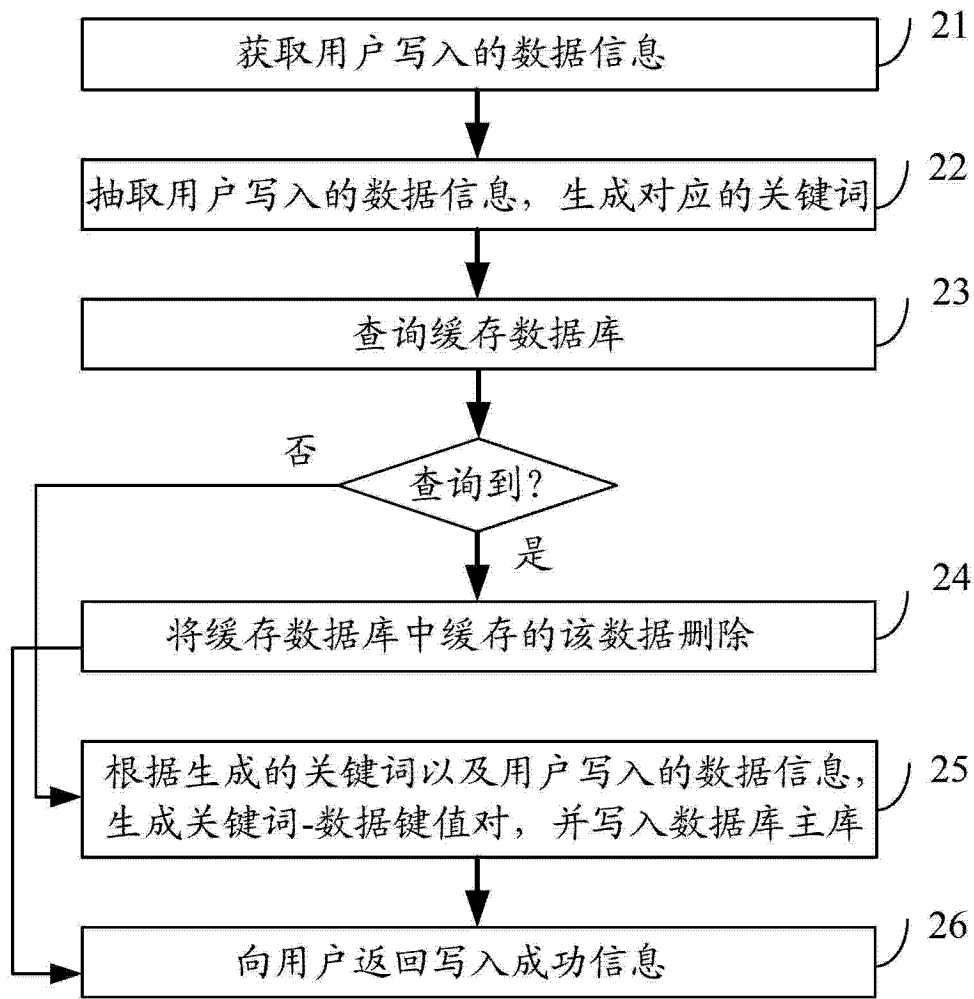


图 3

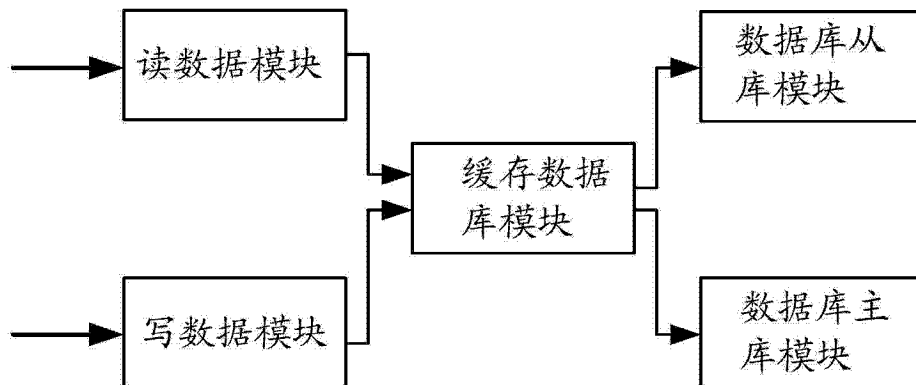


图 4