



(12) 发明专利

(10) 授权公告号 CN 102262626 B

(45) 授权公告日 2013. 08. 07

(21) 申请号 201010180365. 9

(22) 申请日 2010. 05. 24

(73) 专利权人 阿里巴巴集团控股有限公司

地址 英属开曼群岛大开曼资本大厦一座四
层 847 号邮箱

(72) 发明人 王晶昱

(74) 专利代理机构 北京鑫媛睿博知识产权代理
有限公司 11297

代理人 龚家骅

(51) Int. Cl.

G06F 17/30 (2006. 01)

(56) 对比文件

CN 101697152 A, 2010. 04. 21,

CN 101140594 A, 2008. 03. 12,

CN 1783063 A, 2006. 06. 07,

CN 101566986 A, 2009. 10. 28,

陆丽娜, 孟虹, 魏恒义, 杨麦顺. 并行数据库的改进 Hash 划分方法及并行 Join 算法. < 计算机研究与发展 >. 2000, (第 2 期),

审查员 田志刚

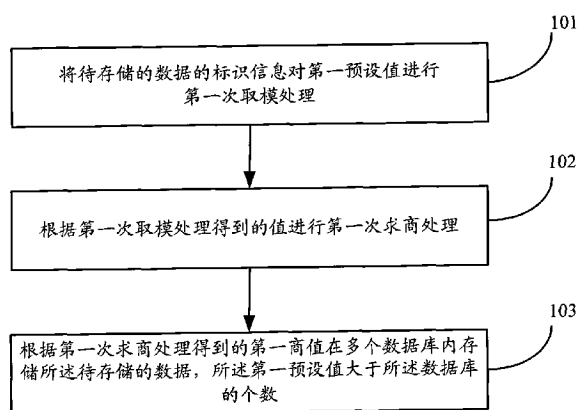
权利要求书3页 说明书9页 附图8页

(54) 发明名称

一种数据库存储数据的方法及装置

(57) 摘要

本申请公开了一种数据库存储数据的方法和装置。本申请中, 将待存储的数据的标识信息对预设值进行取模处理, 并对取模处理得到的值进行求商处理, 然后根据求商处理的结果将数据存储在不同数据库, 由于数据库的个数小于预设值, 通过这种方式可以使存储设备在取模处理时选择一个较大的预设值, 然后再根据数据库的个数选择较小值对取模结果进行求商, 根据求商的值将数据存储到多个数据库中, 这种存储方式, 在数据库的扩展过程中, 只需对需要切分的数据库再次进行取模等步骤, 根据计算出的值进行数据存储, 对其他数据库无影响, 避免了可能的 rehash 操作, 存储设备负担小、数据迁移简单并且迁移过程中数据不易丢失, 有效地实现了数据库的平滑扩展。



1. 一种数据库存储数据的方法,其特征在于,包括:

将待存储的数据的标识信息对第一预设值进行第一取模处理;

根据第一取模处理得到的值进行第一求商处理;

根据第一求商处理得到的第一商值,在多个数据库内存储所述待存储的数据,所述第一预设值大于所述数据库的个数;

其中,根据所述第一求商处理得到的第一商值在多个数据库内存储所述待存储的数据,包括:

获取所述第一求商处理得到的值的整数部分;

根据所述第一求商处理得到的值的整数部分,将待存储的数据存储于不同的数据库中。

2. 如权利要求1所述的方法,其特征在于,

所述第一求商处理的第一除数为所述数据库的个数;

所述第一预设值为所述数据库的个数的整数倍数。

3. 如权利要求1所述的方法,其特征在于,还包括:

将所述多个数据库中的任一数据库切分为多个子数据库的步骤,将由被切分的数据库存储的数据的标识信息对第二预设值进行第二取模处理;

根据第二取模处理得到的值,在多个子数据库内存储由被切分的数据库存储的数据。

4. 如权利要求2所述的方法,其特征在于,还包括:

将所述多个数据库中的任一数据库切分为多个子数据库的步骤,将由被切分的数据库存储的数据的标识信息对第三预设值进行第三取模处理;

对所述第三取模处理的结果进行第二求商处理,所述第二求商处理的第二除数为所述第一除数除自身及1以外的约数;

根据第二求商处理得到的第二商值,在多个子数据库内存储由被切分的数据库存储的数据。

5. 如权利要求3或4所述的方法,其特征在于,所述将所述多个数据库中的任一数据库切分为多个子数据库的步骤之后,还包括:

将待存储的数据的标识信息对所述第一预设值进行所述第一取模处理,并根据第一取模处理得到的值进行第一求商处理;

根据第一求商处理的结果判断该待存储的数据是否属于所述被切分的数据库。

6. 如权利要求1-4中任一项所述的方法,其特征在于,将待存储的数据的标识信息对第一预设值进行第一取模处理之前,还包括:判断所述待存储的数据的标识信息是否大于阈值;

当所述判断结果为否时,执行将待存储的数据的标识信息对所述第一预设值进行第一取模处理的步骤;

当所述判断结果为是时,将所述待存储的数据的标识信息对所述第一预设值的倍数进行第一取模处理。

7. 如权利要求4中所述的方法,其特征在于,所述在多个数据库内存储所述待存储的数据包括:

对于向任一数据库内存储的数据进行第四取模处理,根据所述第四取模处理得到的值

进行第三求商处理,根据所述第三求商处理的值将数据存储到不同的目标表。

8. 一种数据库存储数据的装置,其特征在于,包括:

取模模块,用于将待存储的数据的标识信息对第一预设值进行第一取模处理;

求商模块,用于根据所述第一取模处理得到的值进行第一求商处理;

存储模块,用于根据所述第一求商处理得到的第一商值,在多个数据库内存储所述待存储的数据,所述第一预设值大于所述数据库的个数;

其中,所述存储模块包括:

获取子模块,用于获取所述第一求商处理得到的值的整数部分;

存储子模块,用于根据所述第一求商处理得到的值的整数部分将待存储的数据存储到不同的数据库。

9. 如权利要求 8 所述的存储装置,其特征在于,

所述第一求商处理的第一除数为所述数据库的个数;

所述第一预设值为所述数据库的个数的整数倍数。

10. 如权利要求 8 所述的存储装置,其特征在于,

所述取模模块还用于:当对所述多个数据库中的任一数据库进行切分操作时,将由被切分的数据库存储的数据的标识信息对第二预设值进行第二取模处理;

所述存储模块还用于:根据第二取模处理得到的值,在多个子数据库内存储由被切分的数据存储的数据。

11. 如权利要求 9 所述的存储装置,其特征在于,

所述取模模块还用于:当对所述多个数据库中的任一数据库进行切分操作时,将由被切分的数据库存储的数据的标识信息对第三预设值进行第三取模处理;

所述求商模块还用于:对所述第三取模处理的结果进行第二求商处理,所述第二求商处理的第二除数为所述第一除数除自身及 1 以外的约数;

所述存储模块还用于:根据第二求商处理得到的第二商值,在多个子数据库内存储由被切分的数据库存储的数据。

12. 如权利要求 10 或 11 所述的存储装置,其特征在于,还包括:

判断模块,用于在对所述多个数据库中的任一数据库进行切分操作后,通过所述取模模块将待存储的数据的标识信息对所述第一预设值进行所述第一取模处理,通过所述求商模块根据第一取模处理得到的值进行第一求商处理,并根据第一求商处理的结果判断该待存储的数据是否属于由所述被切分的数据库存储的数据。

13. 如权利要求 8-11 所述的存储装置,其特征在于,还包括:

判断模块,用于将待存储的数据的标识信息对第一预设值进行第一取模处理之前,判断所述待存储的数据的标识信息是否大于阈值;

所述取模模块还用于:当所述判断结果是否为否时,将所述待存储的数据的标识信息对所述第一预设值进行第一取模;当所述判断结果为是时,将所述待存储的数据的标识信息对所述第一预设值的倍数进行第一取模处理。

14. 如权利要求 11 所述的存储装置,其特征在于,存储模块在多个数据库内存储所述待存储的数据时,

所述取模模块还用于:对于向任一数据库内存储的数据进行第四取模处理;

所述求商模块还用于 :根据所述第四取模处理得到的值进行第三求商处理 ;
所述存储模块还用于 :根据所述第三求商处理的值将数据存储到不同的目标表。

一种数据库存储数据的方法及装置

技术领域

[0001] 本发明涉及数据库领域,尤其涉及一种数据库存储数据的方法及装置。

背景技术

[0002] 数据库(Database)是按照数据结构来组织、存储和管理数据的仓库。数据库产生于距今五十年前,随着信息技术和市场的发展,数据管理不再仅仅是存储和管理数据,而转变成用户所需要的各种数据管理的方式。数据库有很多种类型,从最简单的存储有各种数据的表格到能够进行海量数据存储的大型数据库系统都在各个方面得到了广泛的应用。

[0003] 在互联网行业中,企业面对的是庞大的数据以及海量的数据访问,因此数据库在互联网企业中的作用越来越重要。传统的数据库系统基于单台数据库对数据进行管理,但是随着企业的高速发展,使得业务应用对数据库的访问压力成几何级增长,单台数据库所带来的数据量和访问量瓶颈问题日益严重,基于单台数据库系统对数据进行管理的方式已经不能够适应飞速发展中的互联网产业的需求。

[0004] 为了解决单台数据库所带来的数据量和访问量瓶颈问题,必须将原来构建于单台数据库的业务逻辑进行切分,构建包括多台数据库的业务逻辑。一般的切分方法是按照逻辑表进行一些垂直的切分,即将一些大的表切分出来放在单独的数据库存储设备中,例如原单台数据库中存放 10 类业务的数据,其中第 1 类业务的数据量非常大,则将该第 1 类业务对应的数据存放到其他数据库,并相应的切分逻辑数据表。但是当一张逻辑数据表的数据量继续增长,到达单个数据库的瓶颈时,这种简单的切分方法依然面临数据量和访问量瓶颈问题。

[0005] 将一张逻辑表分为多个子库和子表,即分库和分表的核心目的是减少业务系统对单台数据库的依赖。但在整个存储业务系统中,一般分为前端负载均衡、应用存储设备和后端存储等几个主要部分。在这种比较普遍的结构中,应用存储设备和负载均衡都有比较成熟的技术和理论可以实现比较平滑的扩展,但由于数据本身是有状态的,因此数据库的扩展相对比较困难。

[0006] 现有技术中提供的数据库分库解决方案,可以将单个数据库切分为多个数据库,在一定程度上解决了数据库扩展的问题。这种方案通常是按照一个查询量最高的字段进行取模操作。例如通常用户标识(user_id)为查询量最高的字段,根据 user_id 取模后的值,决定这条数据最终应该写到哪个数据库中。例如将单个数据库内的数据分到 16 个数据库时,根据其 user_id mod 16 后的值,决定这条数据最终应该写到哪个数据库中。在对数据库进一步扩展时,这种简单的数据库分库方案将导致大批量的数据的 rehash(重新哈希运算)。rehash 就是对一组数据按照另外的值进行取模并分配的过程,例如,对 1 至 100 这 100 个数,如果每个数都对 10 取模,那么 100 个数就可以根据计算的结果 0~9,平均的分配到 10 个虚拟的组里面去。但如果这个值要变大,如变到 20,那么从 1~100 的所有数字对 20 取模以后的值有很大一部分都和对 10 取模的结果不同,这 100 个数字也自然会被分配到和对 10 取模后不同的虚拟组里面去,这一过程就是 rehash。对应于实际场景,如果原来有 10

个数据库,那么可以对某个值(例如 user_id)对 10 取模,这样就可以将数据比较均衡的分配到 10 个数据库中。假设数据量继续增大,10 个数据库已经不能够满足业务发展的需要,需要再增加 10 个数据库,那么如果还希望平均的分配数据,就需要对以前已经分配到 10 个数据库中的数据根据当前数据库的数量 20,进行重新分配。

[0007] 经过 rehash 后,大批量的数据都需要重新分配数据库。但是,这种大范围的数据迁移,即大量数据在数据库中的位置更新会导致存储设备负担加重、数据迁移复杂,并且在迁移过程中容易导致数据丢失。

发明内容

[0008] 本申请提供了一种数据库存储数据的方法,以解决现有数据库扩展过程中,存储设备负担加重、数据迁移复杂,容易导致数据丢失的问题。

[0009] 本申请还提供了一种存储装置,以解决现有数据库扩展过程中,存储设备负担加重、数据迁移复杂,容易导致数据丢失的问题。

[0010] 本申请提供一种数据库存储数据的方法,包括:

[0011] 将待存储的数据的标识信息对第一预设值进行第一次取模处理;

[0012] 根据第一次取模处理得到的值进行第一次求商处理;

[0013] 根据第一次求商处理得到的第一商值,在多个数据库内存储所述待存储的数据,所述第一预设值大于所述数据库的个数。

[0014] 本申请还提供一种存储装置,包括:

[0015] 取模模块,用于将待存储的数据的标识信息对第一预设值进行第一次取模处理;

[0016] 求商模块,用于根据所述第一次取模处理得到的值进行第一次求商处理;

[0017] 存储模块,用于根据所述第一次求商处理得到的第一商值,在多个数据库内存储所述待存储的数据,所述第一预设值大于所述数据库的个数。

[0018] 与现有技术相比,本申请至少具有以下优点:

[0019] 本申请将待存储的数据的标识信息对预设值进行取模处理,并对取模处理得到的值进行求商处理,然后根据求商处理的结果将数据存储在不同数据库,由于数据库的个数小于预设值,通过这种方式可以使存储设备在取模处理时选择一个较大的预设值,然后再根据数据库的个数选择较小值对取模结果进行求商,根据求商的值将数据存储到多个数据库中,这种存储方式,在数据库的扩展过程中,只需对需要切分的数据库再次进行取模等步骤,根据计算出的值进行数据存储,对其他数据库无影响,避免了可能的 rehash 操作,存储设备负担小、数据迁移简单并且迁移过程中数据不易丢失,有效地实现了数据库的平滑扩展。

附图说明

[0020] 为了更清楚地说明本申请或现有技术中的技术方案,下面将对本申请或现有技术描述中所需要使用的附图作简单的介绍,显而易见地,下面描述中的附图仅仅是本申请的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其它的附图。

[0021] 图 1 是本申请实施例一提供的数据库存储数据的方法流程示意图;

- [0022] 图 2 是现有技术的分库方式示意图；
- [0023] 图 3 是本申请实施例一中具体实现场景下数据库存储数据的方法的流程示意图；
- [0024] 图 4 是本申请实施例一提供的分库方式示意图；
- [0025] 图 5 是本申请实施例二提供的数据库切分的方法的流程示意图；
- [0026] 图 6 是本申请实施例三提供的数据库切分的方法的流程示意图；
- [0027] 图 7 是本申请实施例三中分库方式示意图；
- [0028] 图 8 是本申请实施例五提供的数据库存储数据的方法的流程示意图；
- [0029] 图 9 是本申请实施例中分库分表方法的示意图；
- [0030] 图 10-13 是本申请实施例六提供的存储装置的结构示意图。

具体实施方式

[0031] 下面将结合本申请中的附图,对本申请中的技术方案进行清楚、完整的描述,显然,所描述的实施例是本申请的一部分实施例,而不是全部的实施例。基于本申请中的实施例,本领域普通技术人员在没有做出创造性劳动的前提下所获得的所有其它实施例,都属于本申请保护的范围。

[0032] 此处需要说明的是,本申请实施例中的各步骤可以由单独一个具备计算存储能力的设备完成,如服务器或存储设备等执行,也可以由多个设备共同执行完成,例如由服务器与存储设备组合执行完成,即本申请实施例中利用服务器或存储设备等相同或者不同的存储装置,进行数据存储的过程仅仅是一个实现发明目的的具体实例,并不用于限制本申请的范围,可以根据实际需要灵活采用相同或者不同的设备完成本发明实施例的发明目的。

[0033] 实施例一

[0034] 本申请实施例一提供了一种数据库存储数据的方法,如图 1 所示,包括:

[0035] 步骤 101,将待存储的数据的标识信息对第一预设值进行第一次取模处理;

[0036] 步骤 102,根据第一次取模处理得到的值进行第一次求商处理;

[0037] 步骤 103,根据第一次求商处理得到的第一商值在多个数据库内存储所述待存储的数据,所述第一预设值大于所述数据库的个数。

[0038] 优选地,所述第一求商处理的第一除数为所述数据库的个数。

[0039] 优选地,所述第一预设值为所述数据库的个数的整数倍数。

[0040] 优选地,根据所述第一求商处理得到的第一商值在多个数据库内存储所述待存储的数据,具体可包括:

[0041] 获取所述第一求商处理得到的值的整数部分;

[0042] 根据所述第一求商处理得到的值的整数部分,将待存储的数据存储于不同的数据库中。

[0043] 为了更清楚地阐述实施例一中数据库存储数据的方法,即数据库的分库方法,下面结合具体实现场景进行进一步详细介绍。

[0044] 首先介绍现有技术中的分库方式。

[0045] 现有技术的分库方式采用对标识(id)简单取模的分库方式。如图 2 所示,假设将数据划分到 4 个数据库中,则分库规则为 $\text{id} \bmod 4$,其中的 id 为数据的标识信息,例如 user_id(用户标识)。对于 0~20 这 21 个值来说,数据在数据库内的分布如图 2 所示。其

中,槽 0($\text{id mod } 4 = 0$ 对应的数据库)中存储的数据遵循规则 $\text{id mod } 4 = 0$,槽 1 中存储的数据遵循规则 $\text{id mod } 4 = 1$,槽 2 中存储的数据遵循规则 $\text{id mod } 4 = 2$,槽 3 中存储的数据遵循规则 $\text{id mod } 4 = 3$ 。

[0046] 其中,槽是 hash 以后的结果值中的一个,如 $\text{id mod } 4$,那么如果 id 为自然数,这个计算式最终只会有 4 个结果,也就是 0、1、2、3,这四个结果的每一个都是一个槽。槽 0 中存储 0、4、8、12、16、20 六个数值,槽 1 中存储 1、5、9、13、17 五个数值,槽 2 中存储 2、6、10、14、18 五个数值,槽 3 中存储 3、7、11、15、19 五个数值。

[0047] 在这种结构下,如果 4 个库变为 8 个库的时候,因为数据从对 4 取模变为对 8 取模,所以大批量的数据都需要重新分配数据库,这会使迁移成本成倍增加。

[0048] 本申请实施例一提供的数据库存储数据的方法,如图 3 所示,其中,执行数据存储的设备以服务器为例,包括以下步骤:

[0049] 步骤 301,服务器对待存储的数据的标识信息进行取模处理。

[0050] 具体的,本申请实施例中数据库以设置在服务器中为例进行说明,即存储设备以服务器为例。

[0051] 本申请实施例中,待存储的数据包括服务器中原数据库内的数据,该原数据库需要切分为多个数据库,即进行分库处理。需要说明的是,待存储的数据并不限于原数据库内的数据,也可以是服务器新接收的需要存储的数据。下面以及后续实施例中待存储的数据均以原数据库内的数据为例进行说明,当然如果替换为服务器新接收的需要存储的数据,则对应的实施方式仍然可以执行。

[0052] 服务器对原数据库内存储的数据的标识信息进行取模处理之前,需要设置取模所需的预设值,服务器对数据的标识信息进行取模处理具体为将数据的标识信息对预设值进行取模,该预设值大于分库后得到的新的数据库的个数,例如为分库后得到的数据库的个数的整数倍数。与上述图 2 对应,本申请实施例中以 0-32 这 33 个数划分到 4 个数据库内为例,则分库后得到的数据库的个数为 4,预设的值为 4 的整数倍,以 16 为例。需要说明的是,该预设值还可以是 8、32 或者其他值,此处仅以 16 作为一具体实现方式进行介绍。

[0053] 服务器将 0-32 这 33 个数进行取模处理后得到 0-15 这 16 个值。

[0054] 步骤 302,服务器根据取模处理得到的值进行求商处理。

[0055] 具体的,本申请实施例中,服务器对取模处理得到的值进行求商处理时所使用的除数,为分库后得到的数据库的个数。本申请实施例中该个数具体为 4。

[0056] 服务器根据取模处理得到的值进行求商处理具体为将 0-15 这 16 个值分别除以 4 得到整数部分为 0-3 的值,即图 4 中所示的 $(\text{id Mod } 16)/4$ 得到的值的整数部分为 0-3,为方便描述,在以下内容中直接写为 $(\text{id Mod } 16)/4 = 0-3$ 。

[0057] 步骤 303,服务器根据求商处理得到的值在分库后的数据库内存储数据。

[0058] 具体的,服务器存储步骤 302 中求商处理得到的值与原始数据的对应关系,本步骤中根据该对应关系在 4 个数据库内分别存储对应的 0-32 这 33 个数。如图 4 所示,其中槽 0 中存储的数据遵循规则 $(\text{id mod } 16)/4 = 0$,槽 1 中存储的数据遵循规则 $(\text{id mod } 16)/4 = 1$,槽 2 中存储的数据遵循规则 $(\text{id mod } 16)/4 = 2$,槽 3 中存储的数据遵循规则 $(\text{id mod } 16)/4 = 3$ 。其中,槽 0 中存储 0 ~ 3、16 ~ 20 共 9 个数值,槽 1 中存储 4 ~ 7、21 ~ 24 共 8 个数值,槽 2 中存储 8 ~ 11、25 ~ 28 共 8 个数值,槽 3 中存储 12 ~ 15、29 ~ 32 共 8 个数值。

值。

[0059] 本申请实施例中,将待存储的数据的标识信息对预设值进行取模处理,并对取模处理得到的值进行求商处理,然后根据求商处理的结果将数据存储在小于是预设值的个数的数据库中。由于数据库的个数小于预设值,通过这种方式可以使存储设备在取模处理时选择一个较大的预设值,然后再根据数据库的个数选择较小值对取模结果进行求商,从而将待存储的数据存储到多个数据库中,这种存储方式在进行数据库的扩展时,只需对由被切分的数据库存储的数据的标识信息进行计算处理即可,并根据计算出的值进行存储,对其他数据库并无影响,避免了对其他数据库中存储的数据的 rehash 操作,存储设备负担小、数据迁移简单并且迁移过程中数据不易丢失,有效地实现了数据库的平滑扩展。

[0060] 实施例二

[0061] 如图 5 所示,本申请实施例二是在实施例一的基础上,对实施例一中的其所述多个数据库中的任一数据库进行的一种切分操作,即将所述多个数据库中的任一数据库切分为多个子数据库,具体包括以下步骤:

[0062] 步骤 104,将被切分的数据库内存储的数据的标识信息对第二预设值进行第二次取模处理;该第二预设值可以与第一预设值可以相同或者不同。

[0063] 步骤 105,根据第二次取模处理得到的值,在多个子数据库内存储由被切分的数据库存储的数据。

[0064] 为了更清楚地阐述实施例二中数据库存储数据的方法,下面结合具体实现场景进行进一步详细介绍,其中,执行数据存储的设备以服务器为例。

[0065] 结合图 4 中所示的数据存储,假定此时槽 0 的数据容量不足,则本申请实施例中服务器可以只对该槽 0 进行分库处理。

[0066] 具体地,服务器对槽 0 中的数据的标识信息进行第二次取模处理,该第二次取模处理使用第二预设值,例如槽 0 中的数据的标识信息对 2 进行取模处理,根据取模结果将槽 0 中的数据分配到两个子数据库内,当然还可以根据实际需要,将槽 0 中的数据的标识信息对 3、4 或者其他数值取模,根据取模结果将槽 0 中的数据分配到多个子数据库内。

[0067] 本实施例二中,由被切分的数据库存储的数据包括由被切分的数据库已经存储的数据以及之前由实施例一计算出的由被切分的数据库存储的待存储的数据。即切分后,实施例一中确定的由被切分的数据库存储的待存储的数据,经过实施例二中的步骤后,被分别存储于对应的子数据库中。在此过程中,不需要对其他数据库中存储的数据进行处理,因此避免了对其他数据库中存储的数据的 rehash 操作。

[0068] 所述第二预设值可根据被切分的数据库所存储的数据的标识信息情况进行选择,优选地,选择的第二预设值能够使得多个子数据库能够平均分配由被切分的数据库存储的数据。

[0069] 实施例三

[0070] 本申请实施例三同样是在实施例一的基础上,对实施例一中的其所述多个数据库中的任一数据库进行的一种切分操作,即将所述多个数据库中的任一数据库切分为多个子数据库,如图 6 所示,具体包括以下步骤:

[0071] 步骤 106,将由被切分的数据库存储的数据的标识信息对第三预设值进行第三次取模处理;

[0072] 步骤 107, 对所述第三次取模处理的结果进行第二次求商处理;

[0073] 步骤 108, 根据第二次求商处理得到的第二商值, 在多个子数据库内存储由被切分的数据库存储的数据。

[0074] 本实施例三中, 由于实施例一在进行第一次取模处理时的第一预设值大于数据库个数, 当实施例一中的任一数据库根据本实施例三切分时, 通过第二次求商处理使该被切分的数据库内的数据重新存储, 而其他数据库内的数据则可以不变。

[0075] 优选地, 所述第三预设值与第一预设值相同。

[0076] 优选地, 所述第二次求商处理的第二除数为所述第一除数除自身及 1 以外的约数。

[0077] 为了更清楚地阐述实施例三中数据库存储数据的方法, 下面结合具体实现场景进行进一步详细介绍, 其中, 执行数据存储的设备以服务器为例。

[0078] 结合图 4 中所示的数据存储, 假定此时槽 0 的数据容量不足, 则本申请实施例中服务器可以只对该槽 0 进行分库处理。

[0079] 服务器将槽 0 中的数据的标识信息对第三预设值进行第三次取模处理, 优选地, 该第三次取模处理的第三预设值同步骤 301 中取模处理的预设值相同。然后, 对第三次取模的结果进行第二次求商处理, 此处第二次求商处理的第二除数为步骤 302 中求商处理的除数的除其自身及 1 之外的约数, 4 的约数包括 1、2 和 4, 除自身及 1 之外后为 2。即本实施例中该第二除数为 2, 根据该第二除数将槽 0 划分为 2 个子数据库。服务器进一步根据第二次求商处理得到的第二商值在 2 个子数据库内存储槽 0 内的数据。

[0080] 如图 7 所示, 对槽 0 进行划分后, 原槽 0 内的数据被分配到槽 0_0 和 0_1 中。

[0081] 本实施例三同样具备实施例二的优点, 即在此过程中, 不需要对其他数据库中存储的数据进行处理, 避免了对其他数据库中存储的数据的 rehash 操作。并且经过本实施例三切分后的各子数据库, 还可以根据实施例二或本实施例三中的方式进行进一步的切分, 采用本实施例三中的方式进行进一步的切分时, 只要该第二除数还具备除自身及 1 之外后约数即可, 并且在此过程中, 同样地不需要对其他子数据库中存储的数据进行处理, 避免了对其他子数据库中存储的数据的 rehash 操作, 依此类推, 可有效地实现数据库的平滑扩展。

[0082] 实施例四

[0083] 本实施例四是在实施例二和 / 或实施例三的基础上, 在对所述多个数据库中的任一数据库进行切分操作后, 增加一判断过程, 即判断待存储的数据是否为属于被切分的数据库存储的数据, 该判断过程具体包括:

[0084] 将待存储的数据的标识信息对所述第一预设值进行第一次取模处理, 根据第一次取模处理得到的值进行第一次求商处理;

[0085] 根据第一次求商处理的结果判断该待存储的数据是否属于所述被切分的数据库。如果是, 则执行实施例二或实施例三的步骤, 将该待存储的数据存入对应的子数据库中。

[0086] 实施例四中, 多个数据库内的某一个数据库需要进行切分时, 通过预设条件的设置获知该某一个数据库内存储的数据, 进而仅对该数据库内存储的数据进行迁移处理, 即迁移到该数据库切分得到的多个子数据库内。

[0087] 具体的, 以图 4 中槽 0 的扩展为例进行说明, 则其预设条件为 $(id \bmod 16)/4 = 0$,

即首先判断数据是否为槽 0 内的数据,然后根据判断结果进行数据存储。具体可以表示为:如果落到 $(id \bmod 16)/4 = 0$ 的槽内的这组数据被分配到多个子槽中,仅将槽 0 中存储的数据再进行切分,这样就可以解决 rehash 的问题。具体步骤如下:

[0088] 首先,判断 $(id \bmod 16)/4$ 的值是否等于 0,如果是,计算 $(id \bmod 16)/2$ 的值,根据 $(id \bmod 16)/2$ 计算出的值,将对应 id 的待存储数据存储于对应的子槽中。

[0089] 这样就等于将原来在第 0 个槽内的数据,重新进行了切分,其中,只是槽 0 内的数据进行 rehash,其余槽内的数据不受影响。在数据库内的数据持续的增长时,这种方法可以保证数据库的平滑扩展,并且在数据库切分时,只需要变动被由切分的数据库存储的数据即可。这种模式可以使用小型的服务器或存储设备来解决大型服务器或存储设备才能解决的问题。例如首先使用小型存储设备存储数据,随着存储数据的增多而需要进行数据库切分时,将由切分的数据库存储的数据存储于新增的服务器或存储设备中。这种方式,使得服务器或存储设备可以按当前需要设计,而不需要在一开始就预先采用大型服务器或存储设备,这大大降低了数据的存储成本。

[0090] 实施例五

[0091] 本申请实施例五中,根据数据库 id 生成的特点可以很清晰的得知,数据的增长是沿着时间线逐渐增长的,在上述实施例一至四的分库分表的基础上,还可以根据 id 的范围进行再一次的切分。如果 id 小于设定的阈值时,那么执行实施例一至四中任一实施例步骤;当 id 大于等于设定的阈值,可以执行新的存储方法。通过这样的模式,可以实现对旧数据存储模型的兼容,进一步增强了数据库的平滑扩展性。例如,假设实施例一至四中设定的第一预设值为 16,即最大可以在不需要进行 rehash 的情况下,将数据存储于 16 个数据库中,当需要对 16 个数据库中的任何一个进行切分时,就需要使用 rehash 的方式了。为解决该问题,本实施例五在实施例一至四的基础上,提供如下技术方案。

[0092] 如图 8 所示,在将待存储的数据的标识信息对第一预设值进行第一次取模处理之前,执行以下步骤:

[0093] 步骤 100,判断所述待存储的数据的标识信息是否大于设定阈值;如果否,执行实施例一中的所述步骤 101;如果是,执行步骤 109;

[0094] 步骤 109,存储该待存储的数据。

[0095] 所述步骤 109 中,对该待存储数据的存储可以采用现有技术实现。本实施例五中,优选地,采用本实施例一至实施例四中任一方法进行,只需要根据需要设定其相应的第一预设值、第一除数、第二预设值、第二除数等值即可。进一步优选地,所述相应的第一预设值为实施例一中所述第一预设值的倍数。。这样可保证数据存储方式的一致性,方便存储此部分标识信息大于设定阈值的数据的数据库的平滑扩展。

[0096] 例如,同样以数据的标识信息 user_id 为例,设定第一预设值为 16,第一预设值的倍数以 2 为例,当然还可以根据需要设置为其他倍数。第一预设值的 2 倍数为 32,假设 id 是逐渐自增的,即随着时间的变化,id 逐渐增大。设定阈值为 10000000000,需要说明的是,该第一预设值、阈值可以根据实际需要灵活设置,本实施例中设置的值仅为实现本实施例中的方法所使用的一种具体实现方式。具体步骤如下:

[0097] 步骤 1001,判断 user_id 的值是否小于等于 10000000000,如果是,执行步骤 1002;否则执行步骤 1003;

[0098] 步骤 1002, 将原数据库内存储的数据的标识信息对第一预设值进行第一次取模处理, 取模方式为 $\text{id mod } 16$, 即对应实施一中的步骤 101, 并执行其后的步骤;

[0099] 步骤 1003, 将原数据库内存储的数据的标识信息对预设值的倍数 32 进行取模处理, 此时, 取模方式为 $\text{id mod } 32$, 根据取模处理的结果进行数据存储。

[0100] 所述步骤 1003 中可以根据实施例一至实施例四中任一实施例描述的方式对取模后的结果进行处理, 并进行数据存储; 还可以直接根据取模处理的结果进行数据存储, 即将取模处理的结果与数据对应, 根据取模处理的结果将数据存储到不同的数据库内。例如当取模方式为 $\text{id mod } 32$ 时, 取模结果为 0-31, 服务器将数据存储到 32 个划分后的数据库内。

[0101] 实施例六

[0102] 本申请实施例中提供的分库方法同时还可以辅以分表方法, 该分表方法的具体实现与实施例一至实施例五中的分库方法相似, 即根据数据标识信息进行取模处理以及求商处理, 根据求商处理的结果将不同的数据标识信息存储到不同的目标表中, 如图 9 所示, 其中第一次计算得到的下标对应的标志为数据库的标志, 第二次计算得到的下标所对应的标志为数据库内的目标表。其中的键 (key) 和值 (val) 对应于本申请实施例六中的数据标识信息和求商处理后的结果, 分库规则和分表规则为实施例一至实施例五中任一实施例描述的方式, 即目标表可以采用与数据库切分相同的方式进行切分。例如, 实施例一中在多个数据库内存储待存储的数据时, 对于向任一数据库内存储的数据进行第四次取模处理, 根据第四次取模处理得到的值进行第三次求商处理, 根据第三次求商处理的值将数据存储到不同的目标表。优选的, 第四次取模处理和第三次求商处理分别与第一次取模处理和第一次求商处理相同。当目标表需要切分时, 则可以根据实施例二或三中所述的方法进行切分, 例如根据实施例二中所述的方法对目标表内存储的数据进行第二次取模处理后, 根据取模处理的结果切分目标表; 或者根据实施例三中所述的方法对目标表内存储的数据进行第三次取模处理和第二次求商处理, 根据第二次求商处理的结果切分目标表。

[0103] 实施例七

[0104] 本申请实施例六提供一种存储装置, 如图 10 所示, 包括:

[0105] 取模模块 11, 用于将待存储的数据的标识信息对第一预设值进行第一次取模处理;

[0106] 求商模块 12, 用于根据所述第一次取模处理得到的值进行第一次求商处理;

[0107] 存储模块 13, 用于根据所述第一次求商处理得到的第一商值, 在多个数据库内存储所述待存储的数据, 所述第一预设值大于所述数据库的个数。

[0108] 优选地, 所述第一次求商处理的第一除数为所述数据库的个数;

[0109] 所述第一预设值为所述数据库的个数的整数倍数。

[0110] 进一步地, 所述取模模块 11 还用于: 当对所述多个数据库中的任一数据库进行切分操作时, 将由被切分的数据库存储的数据的标识信息对第二预设值进行第二次取模处理;

[0111] 相应的, 所述存储模块 12 还用于: 根据第二次取模处理得到的值, 在多个子数据库内存储由被切分的数据库存储的数据。

[0112] 所述取模模块 11 还用于: 当对所述多个数据库中的任一数据库进行切分操作时, 将由被切分的数据库存储的数据的标识信息对第三预设值进行第三次取模处理;

[0113] 相应的,所述求商模块 12 还用于:对所述第三次取模处理的结果进行第二次求商处理,所述第二次求商处理的第二除数为所述第一除数除自身及 1 以外的约数;

[0114] 相应的,所述存储模块 13 还用于:根据第二次求商处理得到的第二商值,在多个子数据库内被切分的数据库存储的数据。

[0115] 如图 11 所示,该装置还包括:

[0116] 判断模块 14,用于在对所述多个数据库中的任一数据库进行切分操作后,通过所述取模模块将待存储的数据的标识信息对所述第一预设值进行所述第一次取模处理,通过所述求商模块根据第一次取模处理得到的值进行第一次求商处理,并根据第一次求商处理的结果判断该数据是否属于由所述被切分的数据库存储的数据。

[0117] 如图 12 所示,所述存储模块 13 包括:

[0118] 获取子模块 131,用于获取所述第一次求商处理得到的值的整数部分;

[0119] 存储子模块 132,用于根据所述第一次求商处理得到的值的整数部分将待存储的数据存储到不同的数据库。

[0120] 如图 13 所示,该设备还包括:

[0121] 判断模块 14,用于判断所述待存储的数据的标识信息是否大于阈值;

[0122] 所述取模模块 11 还用于:当所述判断结果为否时,将所述待存储的数据的标识信息对所述第一预设值进行第一次取模;当所述判断结果为是时,将所述待存储的数据的标识信息对所述第一预设值的倍数进行第一次取模处理。

[0123] 所述取模模块 11 还用于:对于向任一数据库内存储的数据进行第四次取模处理;相应的,所述求商模块 12 还用于:根据所述第四次取模处理得到的值进行第三次求商处理;所述存储模块 13 还用于:根据所述第三次求商处理的值将数据存储到不同的目标表。

[0124] 本申请实施例中,通过存储装置对待存储的数据进行取模处理,并将取模后的结果进行求商处理,根据求商处理得到的值在分库后的数据库内存储原数据库内存储的数据,实现了数据库的切分处理,解决数据库的平滑扩展问题。

[0125] 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本发明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。基于这样的理解,本发明的技术方案本质上或者说对现有技术做出贡献的部分可以以软件产品的形式体现出来,该计算机软件产品存储在一个存储介质中,包括若干指令用以使得一台计算机设备(可以是个人计算机,服务器,或者网络设备等)执行本发明各个实施例所述的方法。

[0126] 本领域技术人员可以理解附图只是一个优选实施例的示意图,附图中的模块或流程并不一定是实施本发明所必须的。

[0127] 本领域技术人员可以理解实施例中的装置中的模块可以按照实施例描述进行分布于实施例的装置中,也可以进行相应变化位于不同于本实施例的一个或多个装置中。上述实施例的模块可以合并为一个模块,也可以进一步拆分成多个子模块。

[0128] 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。

[0129] 以上公开的仅为本发明的几个具体实施例,但是,本发明并非局限于此,任何本领域的技术人员能思之的变化都应落入本发明的保护范围。

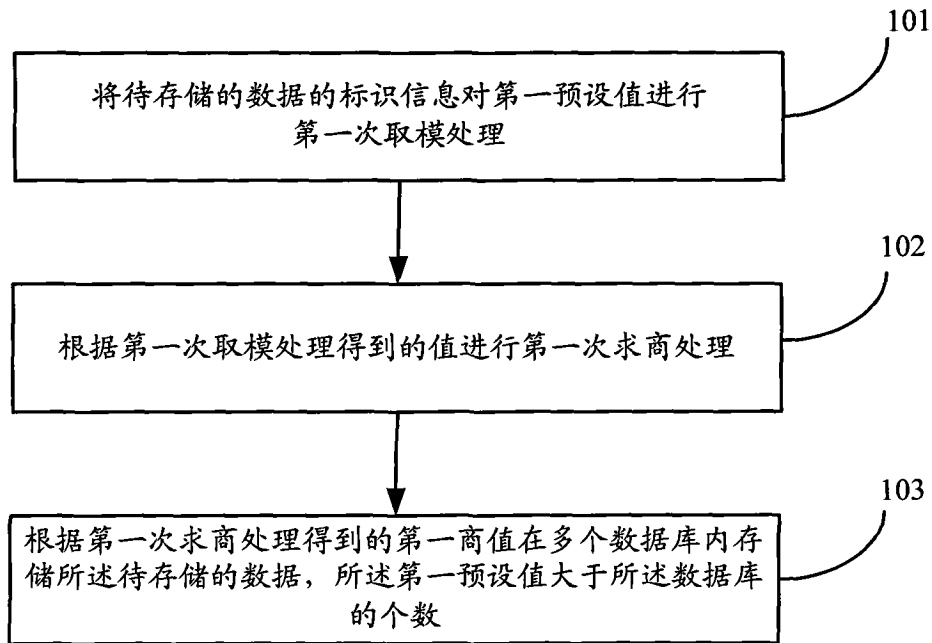


图 1

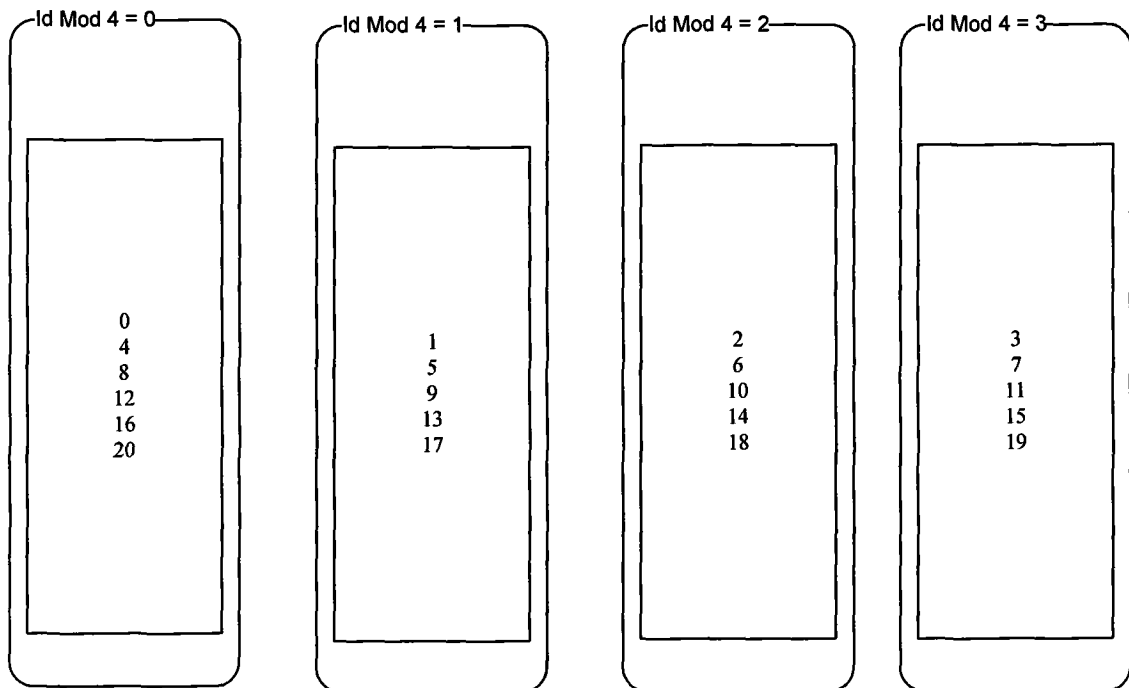


图 2

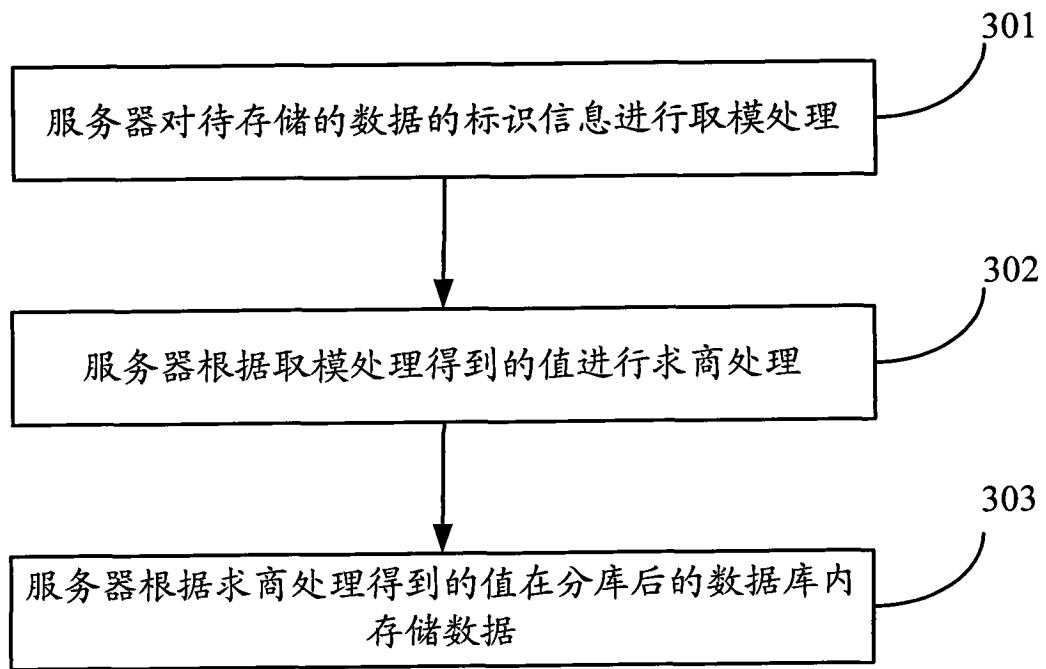


图 3

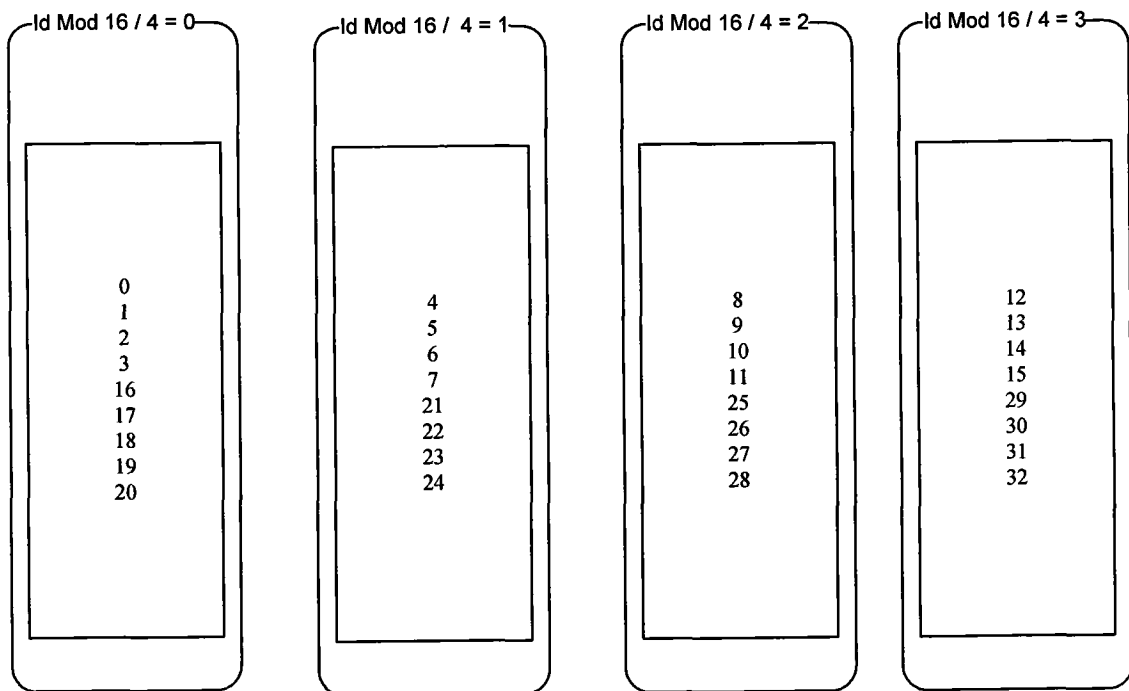


图 4

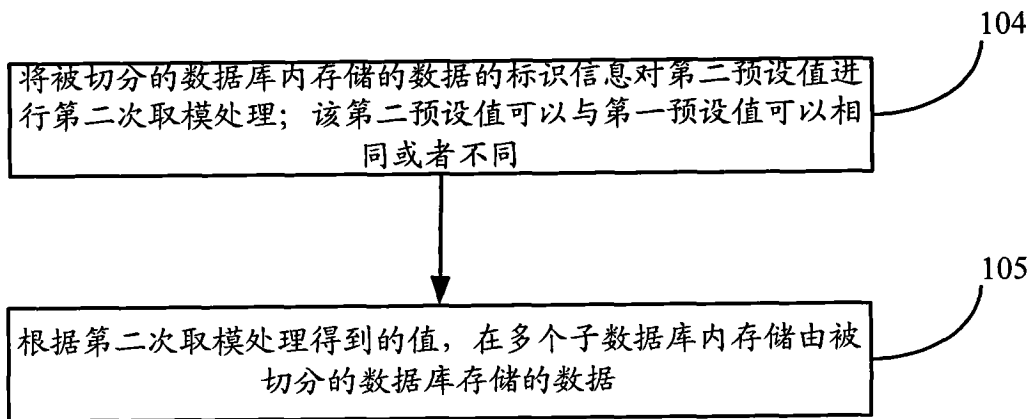


图 5

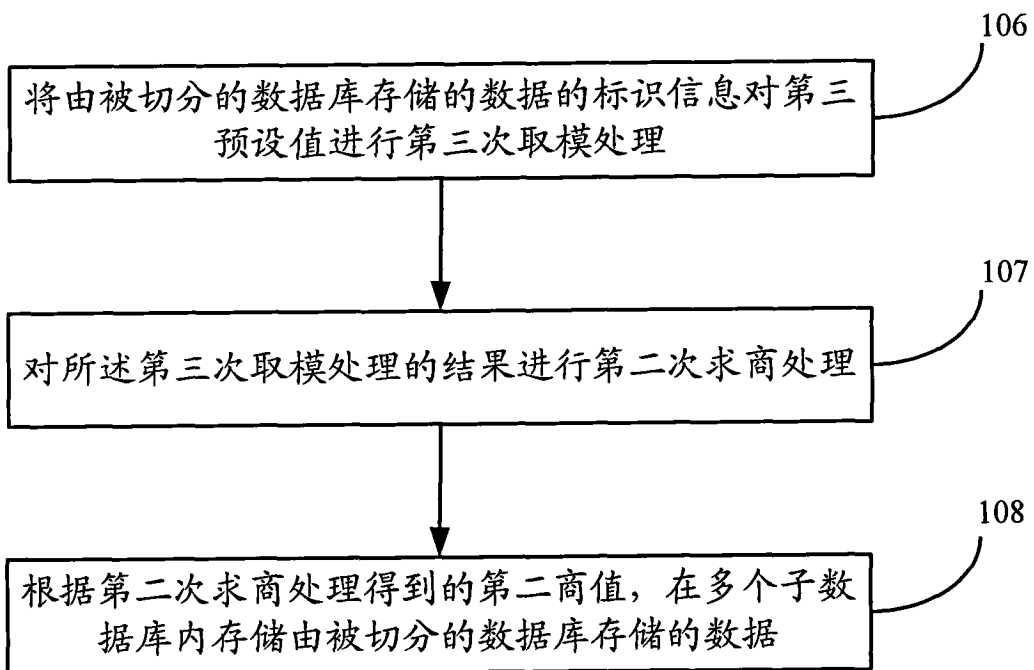


图 6

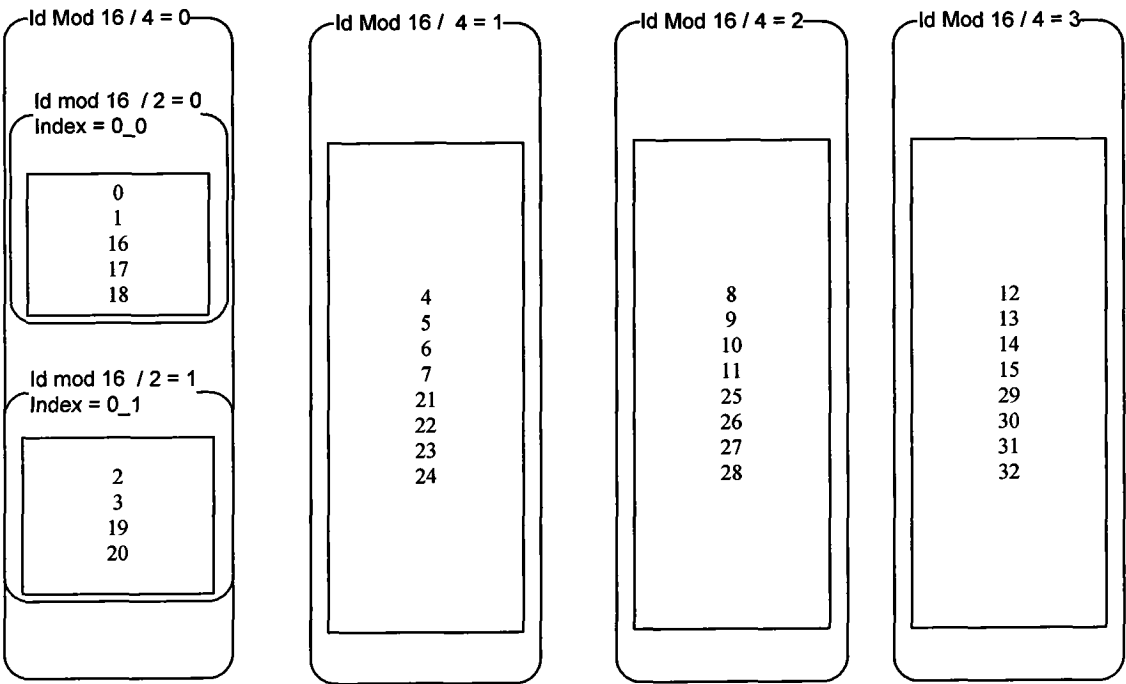


图 7

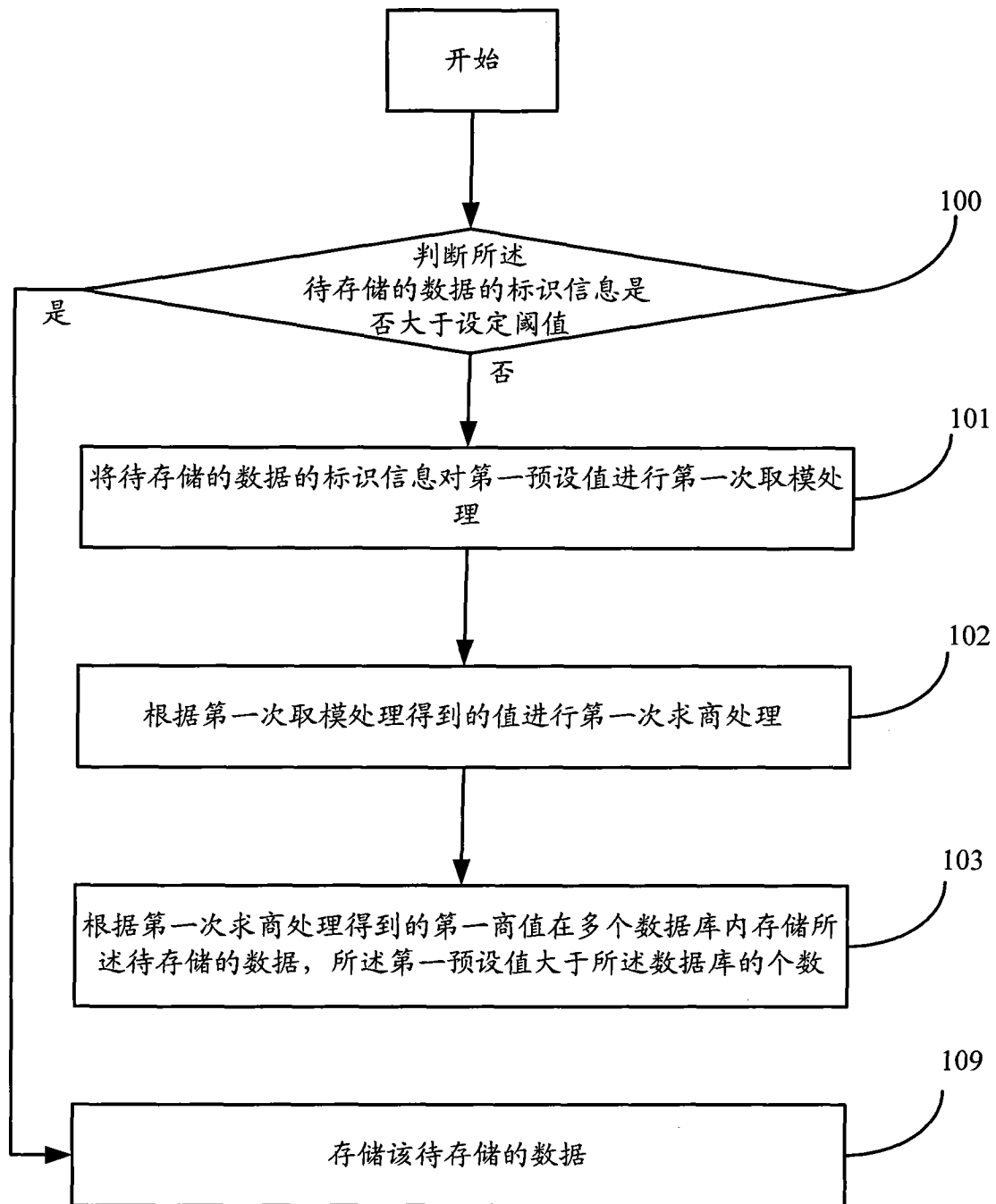


图 8

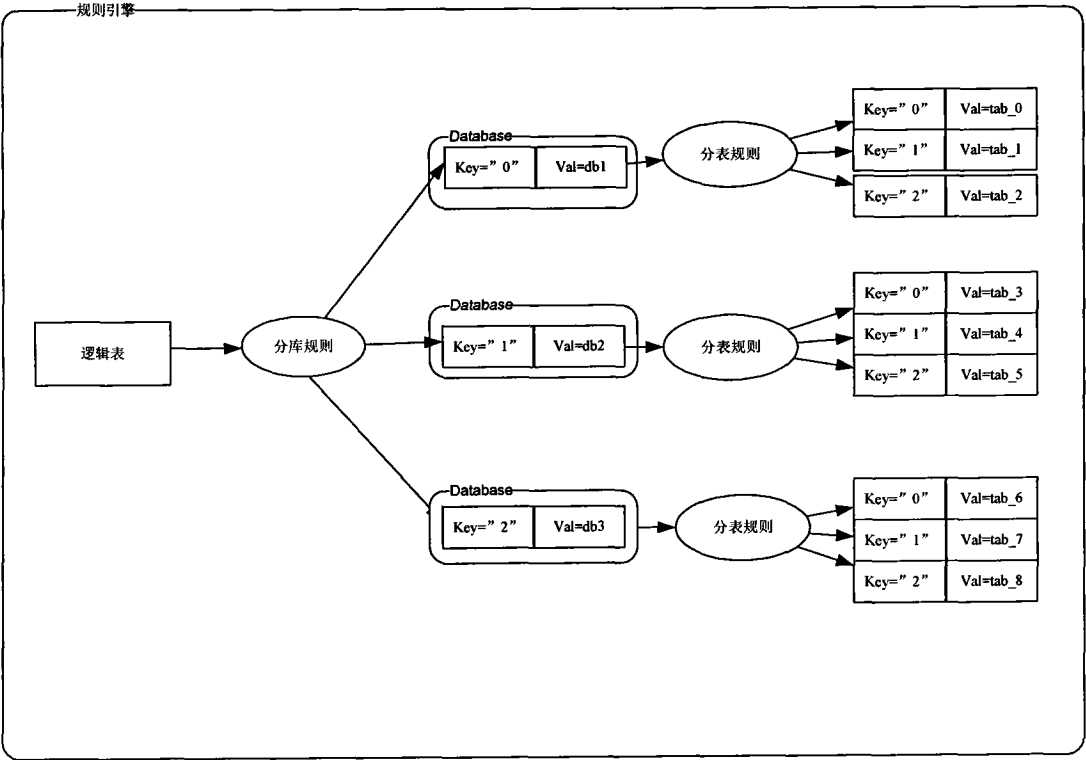


图 9

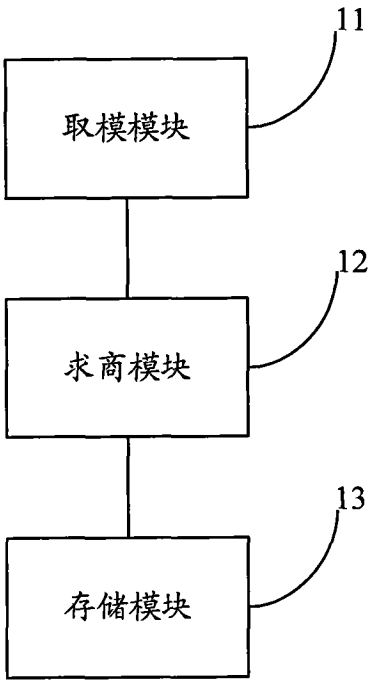


图 10

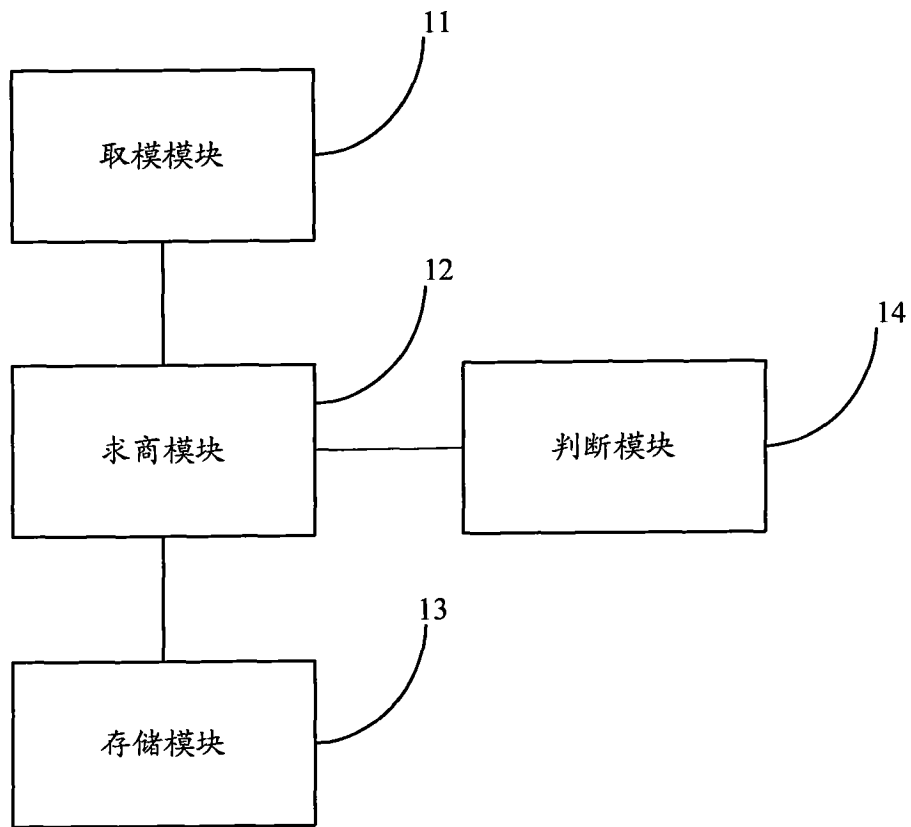


图 11

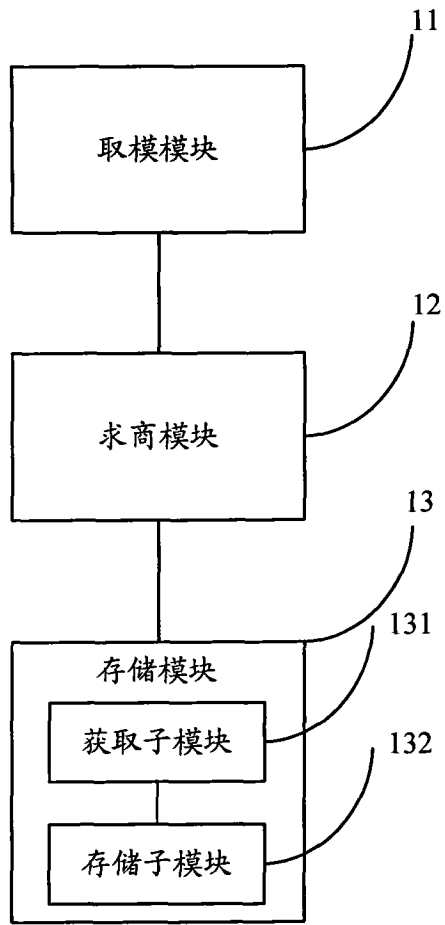


图 12

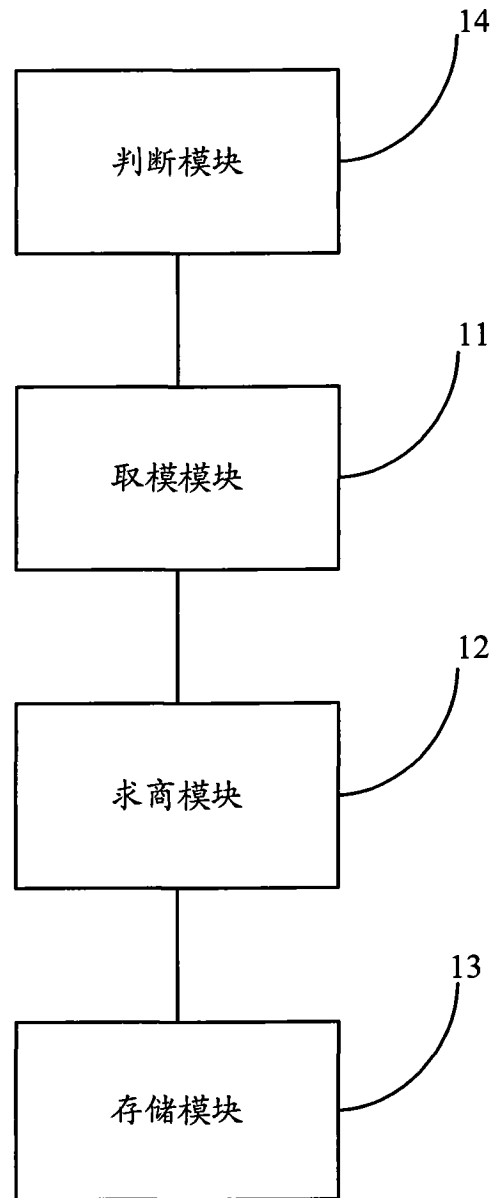


图 13