



(12)发明专利申请

(10)申请公布号 CN 110895524 A
(43)申请公布日 2020.03.20

(21)申请号 201810972812.0

(22)申请日 2018.08.24

(71)申请人 武汉斗鱼网络科技有限公司
地址 430073 湖北省武汉市洪山区光谷软件园F3栋11楼

(72)发明人 张振铎

(74)专利代理机构 武汉河山金堂专利事务所
(普通合伙) 42212
代理人 胡清堂 陈懿

(51) Int. Cl.
G06F 16/215(2019.01)

权利要求书2页 说明书7页 附图2页

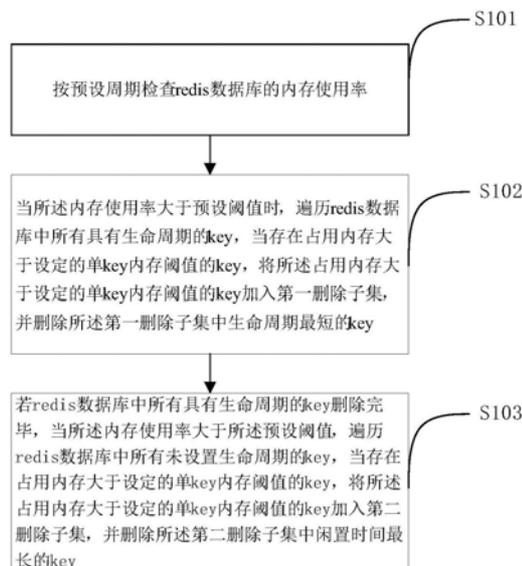
(54)发明名称

redis满载时键的复合过期方法、装置、服务器及存储介质

(57)摘要

本发明提出一种redis满载时键的复合过期方法,包括如下步骤:按预设周期检查redis的内存使用率;当内存使用率大于预设阈值时,从具有生命周期的key中,删除占用内存大于设定的单key内存阈值且生命周期最短的key;从未设置生命周期的key中,删除占用内存大于设定的单key内存阈值且闲置时间最长的key,直到内存使用率达到要求。本发明还提出一种键的复合过期装置,包括内存检测模块、过期删除模块、闲置删除模块。本发明提出一种服务器,其处理器执行所述方法的步骤。本发明还提供一种计算机可读存储介质,其计算机程序执行所述方法的步骤。本发明全面、复合的采用两种算法结合的策略删除redis的key,保证redis稳健运行。本发明属于软件工程领域。

CN 110895524 A



1. 一种redis满载时键的复合过期方法,其特征在于,包括如下步骤:

S101、按预设周期检查redis数据库的内存使用率;

S102、当所述内存使用率大于预设阈值时,遍历redis数据库中所有具有生命周期的key,当存在占用内存大于设定的单key内存阈值的key,将所述占用内存大于设定的单key内存阈值的key加入第一删除子集,并删除所述第一删除子集中生命周期最短的key;

S103、若redis数据库中所有具有生命周期的key删除完毕,当所述内存使用率大于所述预设阈值,遍历redis数据库中所有未设置生命周期的key,当存在占用内存大于设定的单key内存阈值的key,将所述占用内存大于设定的单key内存阈值的key加入第二删除子集,并删除所述第二删除子集中闲置时间最长的key。

2. 根据权利要求1所述的一种redis满载时键的复合过期方法,其特征在于,所述步骤S102中当所述内存使用率大于预设阈值时,使用scan命令扫描redis数据库中所有的key,并使用ttl命令判断各个key是否设置生命周期,之后遍历redis数据库中所有具有生命周期的key。

3. 根据权利要求1所述的一种redis满载时键的复合过期方法,其特征在于,所述步骤S102还包括如下内容:

每删除一个key,检查redis数据库的内存使用率,若所述内存使用率小于或等于所述预设阈值,结束删除,若所述内存使用率大于所述预设阈值,继续删除,直到所述第一删除子集中所有具有生命周期的key删除完毕。

4. 根据权利要求3所述的一种redis满载时键的复合过期方法,其特征在于,若所述第一删除子集中所有具有生命周期的key删除完毕,当所述内存占用率大于所述预设阈值时,若redis中存在具有生命周期的key,计算当前所有具有生命周期的各个key占用内存的平均值,设定单key内存阈值为小于所述平均值的任意值,并执行步骤S102。

5. 根据权利要求1所述的一种redis满载时键的复合过期方法,其特征在于,所述单key内存阈值为单个key所占用内存的阈值,根据需求自主设置,若所述单key内存阈值设置为动态阈值,则每删除一个key,所述单key内存阈值相应减小。

6. 根据权利要求1所述的一种redis满载时键的复合过期方法,其特征在于,步骤S103中将所述占用内存大于设定的单key内存阈值的key加入第二删除子集之后,并删除所述第二删除子集中闲置时间最长的key之前还包括:

使用redis内存对象的idelttime命令获取所述第二删除子集中各个key的闲置时间。

7. 根据权利要求1所述的一种redis满载时键的复合过期方法,其特征在于,所步骤S103还包括如下内容:

每删除一个key,检查redis数据库的内存使用率,若所述内存使用率小于或等于所述预设阈值,结束删除,若所述内存使用率大于所述预设阈值,继续删除,直到所述内存使用率小于或等于所述预设阈值。

8. 一种redis满载时键的复合过期装置,其特征在于,包括如下内容:

内存检测模块:按预设周期检查redis数据库的内存使用率;

过期删除模块:当所述内存使用率大于预设阈值时,遍历redis数据库中所有具有生命周期的key,当存在占用内存大于设定的单key内存阈值的key,将所述占用内存大于设定的单key内存阈值的key加入第一删除子集,并删除所述第一删除子集中生命周期最短的key;

闲置删除模块:若redis数据库中所有具有生命周期的key删除完毕,当所述内存使用率大于所述预设阈值,遍历redis数据库中所有未设置生命周期的key,当存在占用内存大于设定的单key内存阈值的key,将所述占用内存大于设定的单key内存阈值的key加入第二删除子集,并删除所述第二删除子集中闲置时间最长的key。

9.一种服务器,包括存储器、处理器以及存储在所述存储器中并可在所述处理器上运行的计算机程序,其特征在于,所述处理器执行所述计算机程序时实现如权利要求1至7中任一项所述redis满载时键的复合过期方法的步骤。

10.一种计算机可读存储介质,所述计算机可读存储介质存储有计算机程序,其特征在于,所述计算机程序被处理器执行时实现如权利要求1至7中任一项所述redis满载时键的复合过期方法的步骤。

redis满载时键的复合过期方法、装置、服务器及存储介质

技术领域

[0001] 本发明公开一种redis满载时键的复合过期方法、装置、服务器及存储介质,属于软件工程领域。

背景技术

[0002] 软件程序经常使用redis缓存一些数据,redis是基于内存的高性能的key-value数据库,其中,所述key为redis数据库中数据标识。内存是稀缺资源,当redis中的key较多,而程序开发人员又不能主动及时的对redis进行维护,就会造成redis内存打满,可能造成数据丢失。面对这个问题,redis数据库自身提供了当redis内存打满时,淘汰key的策略,比如在代码中设置redis的数据的生命周期,数据过期后自动清除缓存,以保证redis数据库能够正常健康的运行。

[0003] 然而在实际使用中,当数据量暴涨时,生存期的key仍旧会打满redis的内存,单key过大时可能造成redis阻塞,且内存耗费严重,造成redis数据库无法正常使用。

发明内容

[0004] 鉴于此,本发明提出一种redis满载时键的复合过期方法、装置、服务器及存储介质,用于redis数据库满载时有效的释放内存。

[0005] 本发明实施例的第一方面,提供了一种redis满载时键的复合过期方法,包括如下步骤:

[0006] S101、按预设周期检查redis数据库的内存使用率;

[0007] S102、当所述内存使用率大于预设阈值时,遍历redis数据库中所有具有生命周期的key,当存在占用内存大于设定的单key内存阈值的key,将所述占用内存大于设定的单key内存阈值的key加入第一删除子集,并删除所述第一删除子集中生命周期最短的key;

[0008] S103、若redis数据库中所有具有生命周期的key删除完毕,当所述内存使用率大于所述预设阈值,遍历redis数据库中所有未设置生命周期的key,当存在占用内存大于设定的单key内存阈值的key,将所述占用内存大于设定的单key内存阈值的key加入第二删除子集,并删除所述第二删除子集中闲置时间最长的key。

[0009] 本发明实施例的第二方面,提供了一种redis满载时键的复合过期装置,包括以下内容:

[0010] 内存检测模块:按预设周期检查redis数据库的内存使用率;

[0011] 过期删除模块:当所述内存使用率大于所述预设阈值时,遍历redis数据库中所有具有生命周期的key,当存在占用内存大于设定的单key内存阈值的key,将所述占用内存大于设定的单key内存阈值的key加入第一删除子集,并删除所述第一删除子集中生命周期最短的key;

[0012] 闲置删除模块:若redis数据库中所有具有生命周期的key删除完毕,当所述内存使用率大于所述预设阈值,遍历redis数据库中所有未设置生命周期的key,当存在占用内

存大于设定的单key内存阈值的key,将所述占用内存大于设定的单key内存阈值的key加入第二删除子集,并删除所述第二删除子集中闲置时间最长的key。

[0013] 本发明实施例的第三方面,提供了一种服务器,包括存储器、处理器以及存储在所述存储器中并可在所述处理器上运行的计算机程序,所述处理器执行所述计算机程序时实现如本发明实施例第一方面所述方法的步骤。

[0014] 本发明实施例的第四方面,提供了一种计算机可读存储介质,所述计算机可读存储介质存储有计算机程序,所述计算机程序被处理器执行时实现本发明实施例第一方面提供的所述方法的步骤。

[0015] 本发明提供的技术方案带来的有益效果是:

[0016] 本发明提出一种redis满载时键的复合过期方法、装置、服务器及存储介质,提出了两种算法结合的策略删除key。根据redis数据库的对象模型的内存占用、生命周期以及限制时长,结合redis数据库健康检查定时任务,自动在redis数据库将要满载时,过期设置了有生命周期的大内存key,以及未设置生命周期的大内存key,这两种算法结合,每删除1个key则检查内存空间是否达到要求,达到要求立即终止删除。经过两个算法的执行,redis数据库的内存空间已经经过有效的释放。根据系统的实际需求,全面、复合的采用两种算法结合的策略删除redis数据库的key,使得redis数据库的内存空间得到合理、有效的释放,保证redis数据库稳健运行。

附图说明

[0017] 为了更清楚地说明本发明的技术方案,下面将对本发明技术描述中所需要使用的附图作简单地介绍,显而易见地,下面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来讲,在不付出创造性劳动性的前提下,还可以根据这些附图获得其他的附图。

[0018] 图1为本发明实施例一提供的方法流程示意图;

[0019] 图2为本发明实施例二提供的装置结构示意图;

[0020] 图3为本发明实施例三提供的服务器结构示意图。

具体实施方式

[0021] 本发明提出一种redis满载时键的复合过期方法、装置、服务器及存储介质,用于redis数据库满载时清理释放内存。

[0022] 以下描述中,为了说明而不是为了限定,提出了诸如特定系统结构、技术之类的具体细节,以便透彻理解本申请实施例。然而,本领域的技术人员应当清楚,在没有这些具体细节的其它实施例中也可以实现本申请。在其它情况中,省略对众所周知的系统、装置、电路、以及方法的详细说明,以免不必要的细节妨碍本申请的描述。

[0023] 应当理解,当在本说明书和所附权利要求书中使用时,术语“包括”指示所描述特征、整体、步骤、操作、元素和/或组件的存在,但并不排除一个或多个其它特征、整体、步骤、操作、元素、组件和/或其集合的存在或添加。

[0024] 还应当理解,在此本申请说明书中所使用的术语仅仅是出于描述特定实施例的目的而并不意在限制本申请。

[0025] 下面对本发明实施例中的技术方案进行清楚、完整地描述,显然,下面所描述的实施例仅仅是本发明一部分实施例,而非全部的实施例。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前提下所获得的所有其它实施例,都属于本发明保护的范围。

[0026] 本发明实施例一,提供了一种redis满载时键的复合过期方法,参见图1,本发明提出的技术方案包括以下步骤:

[0027] S101、按预设周期检查redis数据库的内存使用率;

[0028] 按预设周期检查redis数据库的内存使用率,当redis的内存使用率大于预设阈值时,开始执行key的删除程序,清理释放内存,当所述内存使用率小于或等于所述预设阈值时,不执行key的删除程序或者停止删除程序。所述内存使用率阈值,为已用内存占总内存的百分比的允许最大值,该阈值为百分比,可根据实际需求设置,比如阈值设为95%,则:当已用内存大于redis数据库总内存的95%时,执行删除程序,清理释放redis内存,若已用内存小于或等于redis数据库总内存的95%时,不执行删除程序或停止删除程序。

[0029] S102、当所述内存使用率大于预设阈值时,遍历redis数据库中所有具有生命周期的key,当存在占用内存大于设定的单key内存阈值的key,将所述占用内存大于设定的单key内存阈值的key加入第一删除子集,并删除所述第一删除子集中生命周期最短的key;

[0030] 可选的,所述步骤S102中当内存使用率大于预设阈值时,使用scan命令扫描redis中所有的key,并使用ttl命令判断各个key是否设置生命周期,之后遍历redis数据库中所有具有生命周期的key。所述scan命令是一个基于游标的迭代器,该命令每次被调用都需要使用上一次这个调用返回的游标作为该次调用的游标参数,以此来延续之前的迭代过程,ttl(time to live)命令秒为单位返回key的剩余过期时间。

[0031] 可选的,步骤S102还包括以下内容:每删除一个key,检查redis的内存使用率,若所述内存使用率小于或等于所述预设阈值,结束删除,若所述内存使用率大于所述预设阈值,继续删除,直到所述第一删除子集中所有具有生命周期的key删除完毕。

[0032] 可选的,若所述第一删除子集中所有具有生命周期的key删除完毕,当所述内存占用率大于所述预设阈值时,若redis中存在具有生命周期的key,计算当前所有具有生命周期的各个key占用内存的平均值,设定单key内存阈值为小于所述平均值的任意值,并执行步骤S102。

[0033] 可选的,该技术方案中所述单key内存阈值为单个key所占用内存的阈值,根据需求自主设置,若所述单key内存阈值设为动态阈值,每删除一个key,所述单key内存阈值相应减小。比如本实施例中将单key内存阈值设定为总内存除以key总量,即各个key所占用的平均内存。

[0034] S103、若redis数据库中所有具有生命周期的key删除完毕,当所述内存使用率大于所述预设阈值,遍历redis数据库中所有未设置生命周期的key,当存在占用内存大于设定的单key内存阈值的key,将所述占用内存大于设定的单key内存阈值的key加入第二删除子集,并删除所述第二删除子集中闲置时间最长的key。

[0035] 可选的,步骤S103中将所述占用内存大于设定的单key内存阈值的key加入第二删除子集之后,并删除所述第二删除子集中闲置时间最长的key之前还包括:使用redis内存对象的idelttime命令获取所述第二删除子集中各个key的闲置时间。

[0036] 可选的,所述步骤S103还包括如下内容:每删除一个key,检查redis的内存使用率,若所述内存使用率小于或等于所述预设阈值,结束删除;若所述内存使用率大于所述预设阈值,继续删除,直到所述内存使用率小于或等于所述预设阈值。

[0037] 此外redis数据库自身也提供了当redis内存打满时,淘汰key的策略,程序根据自身情况,选择并在redis初始化时作为参数设置。当所述步骤S103执行完毕后,可触发redis自身提供的淘汰策略,保证redis内存打满时,能够持续开辟空间供新key使用。

[0038] 本发明实施例二,参见图2,介绍了一种redis满载时键的复合过期装置,具体包括如下模块:

[0039] 内存检测模块201:按预设周期检查redis数据库的内存使用率;

[0040] 过期删除模块202:当所述内存使用率大于预设阈值时,遍历redis数据库中所有具有生命周期的key,当存在占用内存大于设定的单key内存阈值的key,将所述占用内存大于设定的单key内存阈值的key加入第一删除子集,并删除所述第一删除子集中生命周期最短的key;

[0041] 闲置删除模块203:若redis数据库中所有具有生命周期的key删除完毕,当所述内存使用率大于所述预设阈值,遍历redis数据库中所有未设置生命周期的key,当存在占用内存大于设定的单key内存阈值的key,将所述占用内存大于设定的单key内存阈值的key加入第二删除子集,并删除所述第二删除子集中闲置时间最长的key。

[0042] 本发明实施例三:图3是本发明实施例提供的服务器结构的示意图。所述服务器,为提供计算服务的设备,通常指具有较高计算能力,通过网络提供给多个用户使用的计算机。如图3所示,该实施例的服务器3包括:存储器310、处理器320以及系统总线330,所述存储器310包括存储其上的可运行的程序3101,本领域技术人员可以理解,图3中示出的终端设备结构并不构成对终端设备的限定,可以包括比图示更多或更少的部件,或者组合某些部件,或者不同的部件布置。

[0043] 下面结合图3对服务器的各个构成部件进行具体的介绍:

[0044] 存储器310可用于存储软件程序以及模块,处理器320通过运行存储在存储器310的软件程序以及模块,从而执行终端的各种功能应用以及数据处理。存储器310可主要包括存储程序区和存储数据区,其中,存储程序区可存储操作系统、至少一个功能所需的应用程序(比如声音播放功能、图像播放功能等)等;存储数据区可存储根据终端的使用所创建的数据(比如音频数据、电话本等)等。此外,存储器310可以包括高速随机存取存储器,还可以包括非易失性存储器,例如至少一个磁盘存储器件、闪存器件、或其他易失性固态存储器件。

[0045] 在存储器310上包含数据同步方法的可运行程序3101,所述可运行程序3101可以被分割成一个或多个模块/单元,所述一个或多个模块/单元被存储在所述存储器310中,并由处理器320执行,以完成通知的传递并获取通知实现过程,所述一个或多个模块/单元可以是能够完成特定功能的一系列计算机程序指令段,该指令段用于描述所述计算机程序3101在所述服务器3中的执行过程。例如,所述计算机程序3101可以被分割为内存检测模块、过期删除模块、闲置删除模块。

[0046] 处理器320是服务器的控制中心,利用各种接口和线路连接整个终端设备的各个部分,通过运行或执行存储在存储器310内的软件程序和/或模块,以及调用存储在存储器

310内的数据,执行终端的各种功能和处理数据,从而对终端进行整体监控。可选的,处理器320可包括一个或多个处理单元;优选的,处理器320可集成应用处理器和调制解调处理器,其中,应用处理器主要处理操作系统、应用程序等,调制解调处理器主要处理无线通信。可以理解的是,上述调制解调处理器也可以不集成到处理器320中。

[0047] 系统总线320用来连接计算机内部各功能部件,可以传送数据信息、地址信息、控制信息,其种类可以是例如PCI总线、ISA总线、VESA总线等。处理器320的指令通过总线传递至存储器310,存储器310反馈数据给处理器320,系统总线330负责处理器320与存储器310之间的数据、指令交互。当然系统总线330还可以接入其他设备,例如网络接口、显示设备等。

[0048] 所述服务器应至少包括CPU、芯片组、内存、磁盘系统等,其他构成部件在此不再赘述。

[0049] 在本发明实施例中,所述处理器320执行所述计算机程序3101时可以实现上述redis满载时键的复合过期方法。该终端所包括的处理器320执行的可运行程序具体为:一种redis满载时键的复合过期方法,其特征在于,包括如下步骤:

[0050] S101、按预设周期检查redis数据库的内存使用率;S102、当所述内存使用率大于预设阈值时,遍历redis数据库中所有具有生命周期的key,当存在占用内存大于设定的单key内存阈值的key,将所述占用内存大于设定的单key内存阈值的key加入第一删除子集,并删除所述第一删除子集中生命周期最短的key;S103、若redis数据库中所有具有生命周期的key删除完毕,当所述内存使用率大于所述预设阈值,遍历redis数据库中所有未设置生命周期的key,当存在占用内存大于设定的单key内存阈值的key,将所述占用内存大于设定的单key内存阈值的key加入第二删除子集,并删除所述第二删除子集中闲置时间最长的key。

[0051] 进一步的,所述步骤S102中当所述内存使用率大于所述预设阈值时,使用scan命令扫描redis中所有的key,并使用ttl命令判断各个key是否设置生命周期,之后遍历redis数据库中所有具有生命周期的key。

[0052] 进一步的,所述步骤S102还包括如下过程:每删除一个key,检查redis数据库的内存使用率,若所述内存使用率小于或等于所述预设阈值,结束删除,若所述内存使用率大于所述预设阈值,继续删除,直到所述第一删除子集中所有具有生命周期的key删除完毕。

[0053] 进一步的,所述步骤S102中,若所述第一删除子集中所有具有生命周期的key删除完毕,当所述内存占用率大于所述预设阈值时,若redis中存在具有生命周期的key,计算当前所有具有生命周期的各个key占用内存的平均值,设定单key内存阈值为小于所述平均值的任意值,并执行步骤S102。

[0054] 进一步的,所述单key内存阈值为单个key所占用内存的阈值,根据需求自主设置,若所述单key内存阈值设为动态阈值,每删除一个key,所述单key内存阈值相应减小。

[0055] 进一步的,步骤S103中将所述占用内存大于设定的单key内存阈值的key加入第二删除子集之后,并删除所述第二删除子集中闲置时间最长的key之前还包括:使用redis内存对象的idelttime命令获取所述第二删除子集中各个key的闲置时间。

[0056] 进一步的,所述步骤S103还包括如下内容:每删除一个key,检查redis的内存使用率,若所述内存使用率小于或等于所述预设阈值,结束删除,若所述内存使用率大于所述预

设阈值,继续删除,直到所述内存使用率小于或等于所述预设阈值。

[0057] 本发明实施例四:本发明还提供一种计算机可读存储介质,其中,该计算机可读存储介质上存储有计算机程序,所述计算机程序被处理器执行时实现实施例一中记载的redis满载时键的复合过期方法的部分或全部步骤。

[0058] 所属领域的技术人员可以清楚地了解到,为描述的方便和简洁,上述描述的系统,装置和单元的具体工作过程,可以参考前述方法实施例中的对应过程,在此不再赘述。

[0059] 上述实施例中,对各个实施例的描述都各有侧重,某个实施例没有详述或记载的部分,可以参见其他实施例的相关描述。

[0060] 本领域普通技术人员可以意识到,结合本文中所公开的实施例描述的各实施例的模块、单元和/或方法步骤,能够以电子硬件、或者计算机软件和电子硬件的结合来实现。这些功能究竟以硬件还是软件方式来执行,取决于技术方案的特定应用和设计约束条件。专业技术人员可以对每个特定的应用来使用不同方法来实现所描述的功能,但是这种实现不应认为超出本发明的范围。

[0061] 在本申请所提供的几个实施例中,应该理解到,所揭露的系统,装置和方法,可以通过其它的方式实现。例如,以上所描述的装置实施例仅仅是示意性的,例如,所述单元的划分,仅仅为一种逻辑功能划分,实际实现时可以有另外的划分方式,例如多个单元或组件可以结合或者可以集成到另一个系统,或一些特征可以忽略,或不执行。另一点,所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些接口,装置或单元的间接耦合或通信连接,可以是电性,机械或其它的形式。

[0062] 所述作为分离部件说明的单元可以是或者也可以不是物理上分开的,作为单元显示的部件可以是或者也可以不是物理单元,即可以位于一个地方,或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部单元来实现本实施例方案的目的。

[0063] 另外,在本发明各个实施例中的各功能单元可以集成在一个处理单元中,也可以是各个单元单独物理存在,也可以两个或两个以上单元集成在一个单元中。上述集成的单元既可以采用硬件的形式实现,也可以采用软件功能单元的形式实现。

[0064] 所述集成的单元如果以软件功能单元的形式实现并作为独立的产品销售或使用,可以存储在一个计算机可读存储介质中。基于这样的理解,本发明实现上述实施例方法中的全部或部分流程,也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个计算机可读存储介质中,该计算机程序在被处理器执行时,可实现上述各个方法实施例的步骤。其中,所述计算机程序包括计算机程序代码,所述计算机程序代码可以为源代码形式、对象代码形式、可执行文件或某些中间形式等。所述计算机可读介质可以包括:能够携带所述计算机程序代码的任何实体或装置、记录介质、U盘、移动硬盘、磁碟、光盘、计算机存储器、只读存储器(ROM, Read-Only Memory)、随机存取存储器(RAM, Random Access Memory)、电载波信号、电信信号以及软件分发介质等。需要说明的是,所述计算机可读介质包含的内容可以根据司法管辖区内立法和专利实践的要求进行适当的增减,例如在某些司法管辖区,根据立法和专利实践,计算机可读介质不包括电载波信号和电信信号。

[0065] 以上所述,以上实施例仅用以说明本发明的技术方案,而非对其限制;尽管参照前述实施例对本发明进行了详细的说明,本领域的普通技术人员应当理解:其依然可以对前

述各实施例所记载的技术方案进行修改,或者对其中部分技术特征进行等同替换;而这些修改或者替换,并不使相应技术方案的本质脱离本发明各实施例技术方案的精神和范围。

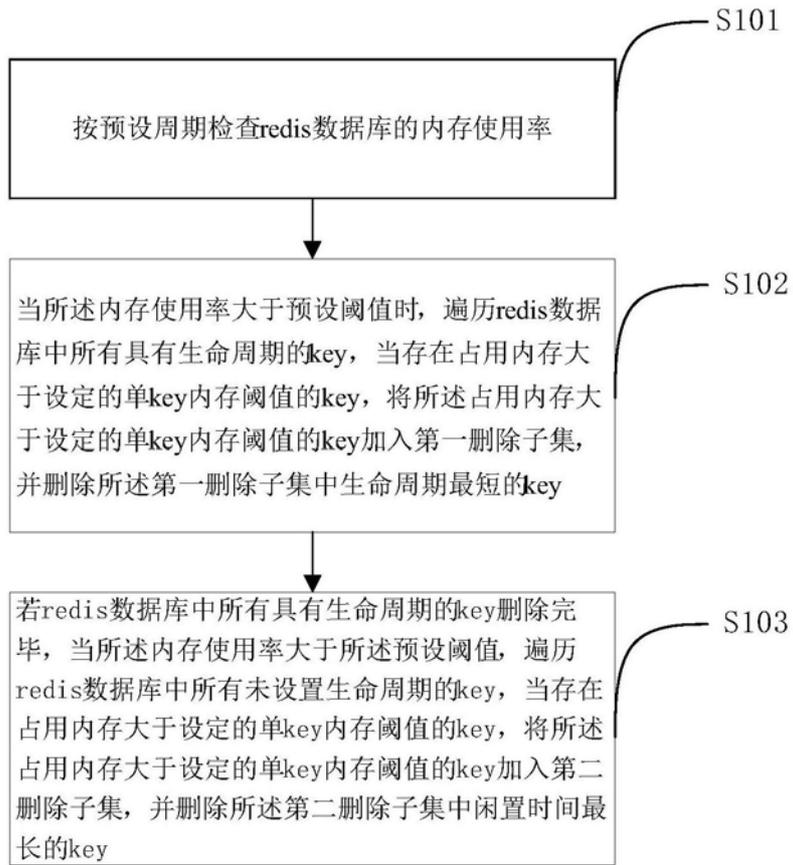


图1

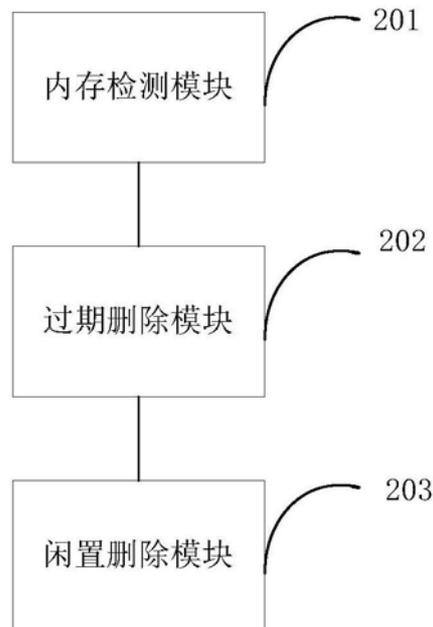


图2

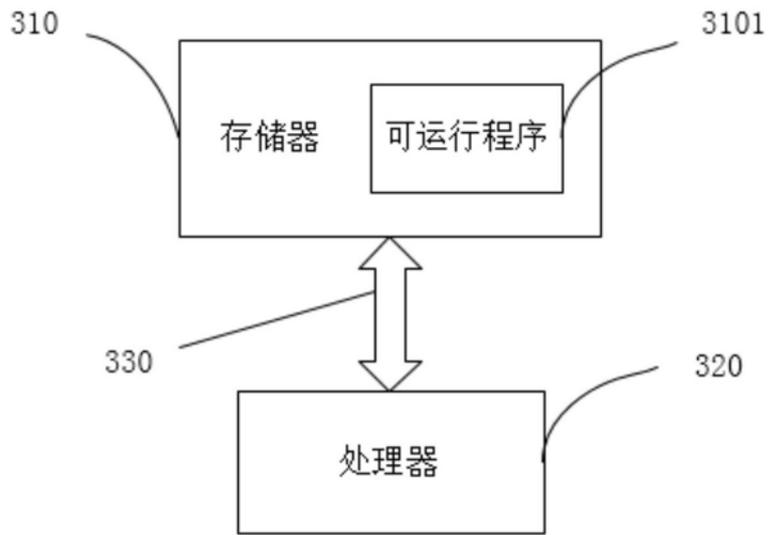


图3