



(12) 发明专利

(10) 授权公告号 CN 110968460 B

(45) 授权公告日 2020.10.27

(21) 申请号 201911211736.2

(22) 申请日 2019.11.27

(65) 同一申请的已公布的文献号
申请公布号 CN 110968460 A

(43) 申请公布日 2020.04.07

(73) 专利权人 上海众言网络科技有限公司
地址 200030 上海市徐汇区宜山路700号B2
幢楼2003、2004、2005单元

(72) 发明人 居晓仁

(74) 专利代理机构 北京卓唐知识产权代理有限公司 11541

代理人 唐海力

(51) Int. Cl.
G06F 11/14 (2006.01)

(56) 对比文件

CN 109543123 A, 2019.03.29

CN 103428309 A, 2013.12.04

CN 103067389 A, 2013.04.24

AU 2011101328 A4, 2011.12.08

youxin.短网址生成思路算法.《https://
www.cnblogs.com/youxin/p/3533389.html》
.2014,第1-4页.

审查员 张也

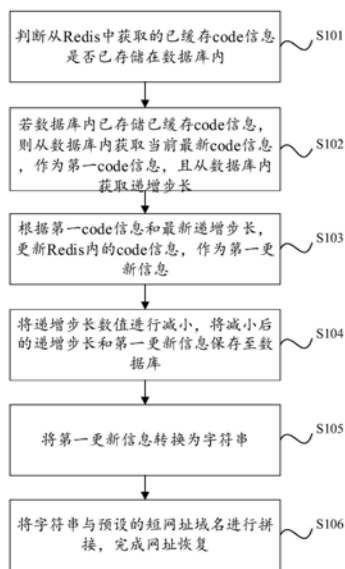
权利要求书2页 说明书7页 附图3页

(54) 发明名称

系统崩溃状态下网址恢复的方法和装置

(57) 摘要

本发明提供一种系统崩溃状态下网址恢复的方法和装置,其中方法包括:判断从Redis中获取的已缓存code信息是否已存储在数据库内;若数据库内已存储已缓存code信息,则从数据库内获取当前最新code信息,作为第一code信息,且从数据库内获取递增步长;根据第一code信息和最新递增步长,更新Redis内的code信息,作为第一更新信息;将最新递增步长数值进行减小,将减小后的递增步长和第一更新信息保存至数据库;将第一更新信息转换为字符串;将字符串与预设的短网址域名进行拼接,完成网址恢复。本发明通过Redis中已缓存code信息,数据库内的code信息,递增步长,来确定保存至数据库内的最新信息,将最新信息转换字符串与短网址固定域名进行拼接,增强网址恢复的稳定性和一致性。



CN 110968460 B

1. 一种系统崩溃状态下网址恢复的方法,其特征在于,所述方法包括:
 - 判断从Redis中获取的已缓存code信息是否已存储在数据库内;
 - 若所述数据库内已存储所述已缓存code信息,则从所述数据库内获取当前最新code信息,作为第一code信息,且从数据库内获取递增步长;
 - 判断所述Redis中已缓存code信息是否大于所述第一code信息;
 - 确定所述Redis中已缓存code信息大于所述第一code信息,则获取所述Redis中已缓存code信息与所述第一code信息的差值;
 - 根据所述差值和所述从数据库内获取递增步长,确定最新递增步长;
 - 将所述差值与所述从数据库内获取递增步长作比较;
 - 若所述差值大于所述从数据库内获取递增步长,则将所述差值作为最新递增步长;
 - 若所述差值小于所述从数据库内获取递增步长,则将所述从数据库内获取递增步长作为最新递增步长;
 - 根据所述第一code信息和最新递增步长,更新Redis内的code信息,作为第一更新信息;
 - 将所述最新递增步长数值进行减小,将减小后的递增步长和所述第一更新信息保存至数据库;
 - 将所述第一更新信息转换为字符串;
 - 将所述字符串与预设的短网址域名进行拼接,完成网址恢复。
2. 根据权利要求1所述的方法,其特征在于,所述方法还包括:
 - 若所述第一更新信息已存储在数据库内的次数超过预设阈值,则从数据库内获取所有code信息列表,确定数值最大的code信息,对其自增1;
 - 将自增1后的最大code信息缓存至Redis中;
 - 将自增1后的最大code信息,作为第二更新信息,与所述最新递增步长保存至数据库内;
 - 将所述第二更新信息转换为字符串;
 - 将所述字符串与预设的短网址域名进行拼接,完成网址恢复。
3. 根据权利要求1所述的方法,其特征在于,所述将所述第一更新信息转换为字符串包括:
 - 利用计算机进制算法将所述第一更新信息转换成对应的进制数值;
 - 根据预设的映射表将所述进制数值转换为字符串。
4. 根据权利要求1所述的方法,其特征在于,所述判断从Redis中获取的已缓存code信息是否已在数据库内之前还包括:
 - 判断是否能从Redis的缓存中获取当前长网址映射的code信息;
 - 若能获取,则直接与预设的短网址域名进行拼接;
 - 若不能获取,则进入判断从Redis中获取的已缓存code信息是否已存储在数据库内的步骤。
5. 根据权利要求1所述的方法,其特征在于,所述若所述数据库内已存储所述已缓存code信息,则从所述数据库内获取当前最新code信息,作为第一code信息,且从数据库内获取递增步长之后还包括:

判断所述Redis中已缓存code信息是否大于所述第一code信息；

确定所述Redis中已缓存code信息大于所述第一code信息，则获取所述Redis中已缓存code信息与所述第一code信息的差值；

根据所述差值和所述从数据库内获取递增步长，确定最新递增步长。

6. 根据权利要求5所述的方法，其特征在于，所述根据所述差值和所述从数据库内获取递增步长，确定最新递增步长包括：

将所述差值与所述从数据库内获取递增步长作比较；

若所述差值大于所述从数据库内获取递增步长，则将所述差值作为最新递增步长；

若所述差值小于所述从数据库内获取递增步长，则将所述从数据库内获取递增步长作为最新递增步长。

7. 根据权利要求4所述的方法，其特征在于，所述判断是否能从Redis的缓存中获取当前长网址映射的code信息之后还包括：

设置所述长网址和所述code信息的有效期。

8. 一种系统崩溃状态下网址恢复的装置，其特征在于，所述装置包括：

判断模块，用于判断从Redis中获取的已缓存code信息是否已存储在数据库内；

获取模块，用于若所述数据库内已存储所述已缓存code信息，则从所述数据库内获取当前最新code信息，作为第一code信息，且从数据库内获取递增步长；判断所述Redis中已缓存code信息是否大于所述第一code信息；确定所述Redis中已缓存code信息大于所述第一code信息，则获取所述Redis中已缓存code信息与所述第一code信息的差值；根据所述差值和所述从数据库内获取递增步长，确定最新递增步长；将所述差值与所述从数据库内获取递增步长作比较；若所述差值大于所述从数据库内获取递增步长，则将所述差值作为最新递增步长；若所述差值小于所述从数据库内获取递增步长，则将所述从数据库内获取递增步长作为最新递增步长；

第一更新信息确定模块，用于根据所述第一code信息和最新递增步长，更新Redis内的code信息，作为第一更新信息；

保存模块，用于将所述最新递增步长数值进行减小，将减小后的递增步长和所述第一更新信息保存至数据库；

转换模块，用于将所述第一更新信息转换为字符串；

网址恢复模块，用于将所述字符串与预设的短网址域名进行拼接，完成网址恢复。

9. 一种电子设备，包括存储器、处理器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，所述处理器执行所述程序时实现如权利要求1至7任一项所述系统崩溃状态下网址恢复的方法的步骤。

10. 一种非暂态计算机可读存储介质，其上存储有计算机程序，其特征在于，该计算机程序被处理器执行时实现如权利要求1至7任一项所述系统崩溃状态下网址恢复的方法的步骤。

系统崩溃状态下网址恢复的方法和装置

技术领域

[0001] 本发明涉及计算机技术领域,尤其涉及一种系统崩溃状态下网址恢复的方法和装置。

背景技术

[0002] 日常工作中正在网上浏览时,会出现计算机突然崩溃现象,正浏览的网页会消失,所以就面临着如何找回丢失的网页的问题。常用的网址恢复方法是基于短网址恢复机制。网址缩短的发展历史不长,它的出现主要是基于微博(如新浪微博,t.cn,url.cn)、短博客、移动营销的崛起而发展起来的。在这些环境中,过长的网址使得用户视觉体验度极差,所以有必要将很长的网址进行缩短。

[0003] 现有方案中主流长网址缩短有两种方案,方案1是将长网址通过某种算法映射为短网址(比如Hash算法处理),方案2是将长网址映射为一个Code。方案1存在短网址不可逆,有潜在冲突风险。方案2虽然能保证唯一性,但是操作比较复杂。且基于上述的现有方式会因为提供短网址服务的程序因为某种原因停止了服务,短网址服务重新提供服务时会出现和之前发放的短网址相冲突情况,基于此,现急需一种能够在系统崩溃状态下网址快速完成恢复的方法。

发明内容

[0004] 为了解决现有技术的上述问题,本发明提供一种系统崩溃状态下网址恢复的方法和装置。

[0005] 第一方面,本发明提供一种系统崩溃状态下网址恢复的方法,该方法包括:

[0006] 判断从Redis中获取的已缓存code信息是否已存储在数据库内;

[0007] 若数据库内已存储已缓存code信息,则从数据库内获取当前最新code信息,作为第一code信息,且从数据库内获取递增步长;

[0008] 根据第一code信息和最新递增步长,更新Redis内的code信息,作为第一更新信息;

[0009] 将递增步长数值进行减小,将减小后的递增步长和第一更新信息保存至数据库;

[0010] 将第一更新信息转换为字符串;

[0011] 将字符串与预设的短网址域名进行拼接,完成网址恢复。

[0012] 进一步地,该方法还包括:

[0013] 若第一更新信息已存储在数据库内的次数超过预设阈值,则从数据库内获取所有code信息列表,确定数值最大的code信息,对其自增1;

[0014] 将自增1后的最大code信息缓存至Redis中;

[0015] 将自增1后的最大code信息,作为第二更新信息,与所述最新递增步长保存至数据库内;

[0016] 将第二更新信息转换为字符串;

- [0017] 将字符串与预设的短网址域名进行拼接,完成网址恢复。
- [0018] 进一步地,将第一更新信息转换为字符串包括:
- [0019] 利用计算机进制算法将第一更新信息转换成对应的进制数值;
- [0020] 根据预设的映射表将进制数值转换为字符串。
- [0021] 进一步地,判断从Redis中获取的已缓存code信息是否已在数据库内之前还包括:
- [0022] 判断是否能从Redis的缓存中获取当前长网址映射的code信息;
- [0023] 若能获取,则直接与预设的短网址域名进行拼接;
- [0024] 若不能获取,则进入判断从Redis中获取的已缓存code信息是否已存储在数据库内的步骤。
- [0025] 进一步地,若数据库内已存储已缓存code信息,则从数据库内获取当前最新code信息,作为第一code信息,且从数据库内获取递增步长之后还包括:
- [0026] 判断Redis中已缓存code信息是否大于第一code信息;
- [0027] 确定Redis中已缓存code信息大于第一code信息,则获取Redis中已缓存code信息与第一code信息的差值;
- [0028] 根据差值和从数据库内获取递增步长,确定最新递增步长。
- [0029] 进一步地,根据差值和从数据库内获取递增步长,确定最新递增步长包括:
- [0030] 将差值与从数据库内获取递增步长作比较;
- [0031] 若差值大于从数据库内获取递增步长,则将差值作为最新递增步长;
- [0032] 若差值小于从数据库内获取递增步长,则将从数据库内获取递增步长作为最新递增步长。
- [0033] 进一步地,判断是否能从Redis的缓存中获取当前长网址映射的code信息之后还包括:
- [0034] 设置长网址和code信息的有效期。
- [0035] 第二方面,本发明提供一种系统崩溃状态下网址恢复的装置,该装置包括:
- [0036] 判断模块,用于判断从Redis中获取的已缓存code信息是否已存储在数据库内;
- [0037] 获取模块,用于若数据库内已存储已缓存code信息,则从数据库内获取当前最新code信息,作为第一code信息,且从数据库内获取递增步长;
- [0038] 第一更新信息确定模块,用于根据第一code信息和最新递增步长,更新Redis内的code信息,作为第一更新信息;
- [0039] 保存模块,用于将递增步长数值进行减小,将减小后的递增步长和第一更新信息保存至数据库;
- [0040] 转换模块,用于将第一更新信息转换为字符串;
- [0041] 网址恢复模块,用于将字符串与预设的短网址域名进行拼接,完成网址恢复。
- [0042] 第三方面,本发明提供一种电子设备,包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序,处理器执行所述程序时实现第一方面提供的系统崩溃状态下网址恢复的方法的步骤。
- [0043] 第四方面,本发明提供一种非暂态计算机可读存储介质,其上存储有计算机程序,该计算机程序被处理器执行时实现第一方面提供的系统崩溃状态下网址恢复的方法的步骤。

[0044] 本发明通过Redis中已缓存code信息,数据库内的code信息以及递增步长,来确定保存至数据库内的最新信息,再将最新信息转换字符串,与短网址固定域名进行拼接,增强了网址快速恢复的稳定性和一致性。

附图说明

[0045] 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,下面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,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的前提下,还可以根据这些附图获得其他的附图。

[0046] 图1为本发明实施例提供的系统崩溃状态下网址恢复的方法流程示意图;

[0047] 图2为本发明另一实施例提供的系统崩溃状态下网址恢复的方法流程示意图;

[0048] 图3为本发明实施例提供的系统崩溃状态下网址恢复的装置框图;

[0049] 图4为本发明实施例提供的电子设备框图。

具体实施方式

[0050] 为了使本技术领域的人员更好地理解本申请方案,下面将结合本申请实施例中的附图,对本申请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申请一部分的实施例,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前提下所获得的所有其他实施例,都应当属于本申请保护的范畴。

[0051] 需要说明的是,本申请的说明书和权利要求书及上述附图中的术语“第一”、“第二”等是用于区别类似的对象,而不必用于描述特定的顺序或先后次序。应该理解这样使用的数据在适当情况下可以互换,以便这里描述的本申请的实施例。此外,术语“包括”和“具有”以及他们的任何变形,意图在于覆盖不排他的包含,例如,包含了一系列步骤或单元的过程、方法、系统、产品或设备不必限于清楚地列出的那些步骤或单元,而是可包括没有清楚地列出的或对于这些过程、方法、产品或设备固有的其它步骤或单元。

[0052] 需要说明的是,在不冲突的情况下,本申请中的实施例及实施例中的特征可以相互组合。下面将参考附图并结合实施例来详细说明本申请。

[0053] 现有的主流长网址缩短成短网址的方法会出现短网址不可逆,有潜在冲突风险以及操作比较复杂等问题,为了解决上述问题,本发明实施例提供一种系统崩溃状态下网址恢复的方法,如图1所示,该方法包括:

[0054] 步骤S101,判断从Redis中获取的已缓存code信息是否已存储在数据库内;

[0055] 步骤S102,若数据库内已存储已缓存code信息,则从数据库内获取当前最新code信息,作为第一code信息,且从数据库内获取递增步长;

[0056] 步骤S103,根据第一code信息和最新递增步长,更新Redis内的code信息,作为第一更新信息;

[0057] 步骤S104,将递增步长数值进行减小,将减小后的递增步长和第一更新信息保存至数据库;

[0058] 步骤S105,将第一更新信息转换为字符串;

[0059] 步骤S106,将字符串与预设的短网址域名进行拼接,完成网址恢复。

[0060] 具体为,日常工作中常使用数据库来进行数据的存储,由于一般的系统任务中通常不会存在高并发的情况,所以这样看起来并没有什么问题,可是一旦涉及大数据量的需求,比如主页访问量瞬间较大的时候,这个时候往往不是数据库能够承受的,极其容易造成数据库系统瘫痪,最终导致服务宕机的严重问题。需要说明的是,长网址是指原始网址,比如https://survey.wjcem.com/openapi/get_survey/?source=1&survey_code=593&store_id=5be0141754bc141665768be9&signature=77a28590337f5cd732658c6093196187,短网址就是以固定域名加混淆后的Code来体现(<https://smgk.cn/5RRwzr>)。

[0061] 在本发明实施例中长网址与短网址的对应关系以key:value格式(以长网址为key,短网址的Code为value)保存在Redis中,也即是将短网址信息缓存在Redis内,根据长网址到Redis缓存中获取对应的短网址。需要补充的是,本发明实施例中的短网址恢复机制是指提供短网址服务的程序因为某种原因停止了服务,解决在该短网址服务重新提供服务时能够与之前发放的短网址不冲突的问题,在本发明实施例里是每隔30秒保存一次当前状态下发放的code,所以如果在两次保存中间发生断电时,此时恢复就依赖于上一次保存的code和断电之前已经发放过多少(即递增步长)。

[0062] 首先,计算机服务器可以判断是否能从Redis的缓存中获取当前长网址映射的code信息;若能获取,则直接与预设的短网址域名进行拼接;若不能获取,则判断从Redis中获取的已缓存code信息是否已存储在数据库内,若数据库内已存储已缓存code信息,则从数据库内获取当前最新code信息,作为第一code信息,且从数据库内获取递增步长,其中获取当前最新code信息是指数据库内存储的时间最新对应的存储的code信息,另外,递增步长是直接数据库中随着code信息一起取出来。

[0063] 将从数据库中取出的第一code信息和最新递增步长求和,得到的求和数值作为存储在Redis内的缓存信息,依此更新Redis存储数据,得到第一更新信息。

[0064] 由于网址在出现故障后进行恢复时,若递增步长设置过大,也即是跨度过大,会出现多个重复信息的情况,将过大数值的递增步长信息存在数据库会导致恢复机制不准确的情况,此时将最新递增步长数值进行减小,将减小后的递增步长和第一更新信息保存至数据库,在这里可以对最新递增步长数值进行减半的处理,综合举例:从数据库中取出来的信息是Code:100,递增步长:50,那么此时计算出来的最新Code是150(100+50),将数值150更新Redis缓存code信息,然后存储到数据库中是(150,25);其中25是50/2求得的。

[0065] 使用62进制算法将CacheCode转化成对应的62进制数值,根据预设的映射表,将62进制转化为字符串。

[0066] 在进行网址恢复时,以固定域名加混淆后的字符串完成拼接,达到网址恢复的目的。

[0067] 本发明通过Redis中已缓存code信息,数据库内的code信息以及递增步长,来确定保存至数据库内的最新信息,再将最新信息转换字符串,与短网址固定域名进行拼接,增强了网址快速恢复的稳定性和一致性。

[0068] 基于上述各实施例的内容,作为一种可选实施例:如图2,该方法还包括:

[0069] 步骤S201,若第一更新信息已存储在数据库内的次数超过预设阈值,则从数据库内获取所有code信息列表,确定数值最大的code信息,对其自增1;

- [0070] 步骤S202,将自增1后的最大code信息缓存至Redis中;
- [0071] 步骤S203,将自增1后的最大code信息,作为第二更新信息,与最新递增步长保存至数据库内;
- [0072] 步骤S204,将第二更新信息转换为字符串;
- [0073] 步骤S205,将字符串与预设的短网址域名进行拼接,完成网址恢复。
- [0074] 具体为,在本发明实施例中介绍另外一种情况的恢复机制:在查询数据库内的已存储的code信息时,若每次更新后的第一更新信息已出现在数据库内的次数超过预设阈值,在这里优选为3次,则从数据库内获取所有code信息列表,确定数值最大的code信息,对其自增1,在这里自增1是指每次获取的数值比上一次加1;将自增1后的code信息缓存至Redis中;将该自增1后的最大code信息,作为第二更新信息,与最新递增步长一同保存至数据库内。其中最新递增步长和前述的最新递增步长是同一个概念。将第二更新信息转换为字符串,将该字符串与预设的短网址域名进行拼接,完成网址恢复。
- [0075] 基于上述各实施例的内容,作为一种可选实施例:将第一更新信息转换为字符串包括:
- [0076] 利用计算机进制算法将第一更新信息转换成对应的进制数值;
- [0077] 根据预设的映射表将进制数值转换为字符串。
- [0078] 具体为,使用62进制算法将第一更新信息转化成对应的62进制数值,根据预设的映射表(包含26个大写字母,26个小写字母,0~9的阿拉伯数字,将这62个字符随机排列组合成长62的字符串)将62进制转化为字符串。
- [0079] 基于上述各实施例的内容,作为一种可选实施例:若数据库内已存储已缓存code信息,则从数据库内获取当前最新code信息,作为第一code信息,且从数据库内获取递增步长之后还包括:
- [0080] 判断Redis中已缓存code信息是否大于第一code信息;
- [0081] 确定Redis中已缓存code信息大于第一code信息,则获取Redis中已缓存code信息与第一code信息的差值;
- [0082] 根据差值和从数据库内获取递增步长,确定最新递增步长。
- [0083] 具体为,在本发明实施例中主要说明最新递增步长的确定方式:判断Redis中已缓存code信息是否大于第一code信息,若Redis中已缓存code信息大于第一code信息,则获取Redis中已缓存code信息与第一code信息的差值,比如恢复前Code (Redis中已缓存code信息)是3245,恢复时的Code (第一code信息)是3256,那么差值就是11(3256-3245)。之后再比较该差值和从数据库内获取递增步长(即数据库内已存储的递增步长)之间的大小,进而确定最新递增步长。
- [0084] 基于上述各实施例的内容,作为一种可选实施例:根据差值和从数据库内获取递增步长,确定最新递增步长包括:
- [0085] 将差值与从数据库内获取递增步长作比较;
- [0086] 若差值大于从数据库内获取递增步长,则将差值作为最新递增步长;
- [0087] 若差值小于从数据库内获取递增步长,则将从数据库内获取递增步长作为最新递增步长。
- [0088] 具体为,比较Redis中已缓存code信息与第一code信息间的差值,与从数据库内获

取递增步长之间的大小,若差值大于从数据库内获取递增步长,则将差值作为最新递增步长;若差值小于从数据库内获取递增步长,则将从数据库内获取递增步长作为最新递增步长。

[0089] 基于上述各实施例的内容,作为一种可选实施例:判断是否能从Redis的缓存中获取当前长网址映射的code信息之后还包括:

[0090] 设置长网址和code信息的有效期。

[0091] 具体为,在本发明实施例中设置长网址和code信息数据保存在Redis缓存中的时间,超过这个时间,就由Redis自动将这些存储的数据删除。

[0092] 本发明实施例设置有效期,起到减少Redis缓存空间占用的目的。

[0093] 根据本发明的再一个方面,本发明实施例提供系统崩溃状态下网址恢复的装置,参见图3,图3为本发明实施例提供的系统崩溃状态下网址恢复的装置框图。该装置用于在前述各实施例中完成本发明实施例提供的系统崩溃状态下网址恢复。因此,在前述各实施例中的本发明实施例提供的系统崩溃状态下网址恢复的方法中的描述和定义,可以用于本发明实施例中各执行模块的理解。

[0094] 该装置包括:

[0095] 判断模块301,用于判断从Redis中获取的已缓存code信息是否已存储在数据库内;

[0096] 获取模块302,用于若数据库内已存储已缓存code信息,则从数据库内获取当前最新code信息,作为第一code信息,且从数据库内获取递增步长;

[0097] 第一更新信息确定模块303,用于根据第一code信息和最新递增步长,更新Redis内的code信息,作为第一更新信息;

[0098] 保存模块304,用于将最新递增步长数值进行减小,将减小后的递增步长和第一更新信息保存至数据库;

[0099] 转换模块305,用于将第一更新信息转换为字符串;

[0100] 网址恢复模块306,用于将字符串与预设的短网址域名进行拼接,完成网址恢复。

[0101] 具体的,本实施例的装置中各模块实现其功能的具体过程可参见对应的方法实施例中的相关描述,此处不再赘述。

[0102] 本发明通过Redis中已缓存code信息,数据库内的code信息以及递增步长,来确定保存至数据库内的最新信息,再将最新信息转换字符串,与短网址固定域名进行拼接,增强了网址快速恢复的稳定性和一致性。

[0103] 图4为本发明实施例提供的电子设备框图,如图4所示,该设备包括:处理器401、存储器402和总线403;

[0104] 其中,处理器401及存储器402分别通过总线403完成相互间的通信;处理器401用于调用存储器402中的程序指令,以执行上述实施例所提供的系统崩溃状态下网址恢复的方法,例如包括:判断从Redis中获取的已缓存code信息是否已存储在数据库内;若数据库内已存储已缓存code信息,则从数据库内获取当前最新code信息,作为第一code信息,且从数据库内获取递增步长;根据第一code信息和最新递增步长,更新Redis内的code信息,作为第一更新信息;将最新递增步长数值进行减小,将减小后的递增步长和第一更新信息保存至数据库;将第一更新信息转换为字符串;将字符串与预设的短网址域名进行拼接,完成

网址恢复。

[0105] 本发明实施例提供一种非暂态计算机可读存储介质,其上存储有计算机程序,该计算机程序被处理器执行时实现系统崩溃状态下网址恢复的方法的步骤。例如包括:判断从Redis中获取的已缓存code信息是否已存储在数据库内;若数据库内已存储已缓存code信息,则从数据库内获取当前最新code信息,作为第一code信息,且从数据库内获取递增步长;根据第一code信息和最新递增步长,更新Redis内的code信息,作为第一更新信息;将最新递增步长数值进行减小,将减小后的递增步长和第一更新信息保存至数据库;将第一更新信息转换为字符串;将字符串与预设的短网址域名进行拼接,完成网址恢复。

[0106] 以上所描述的装置实施例仅仅是示意性的,其中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模块来实现本实施例方案的目的。本领域普通技术人员在不付出创造性的劳动的情况下,即可以理解并实施。

[0107] 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各实施方式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。基于这样的理解,上述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,该计算机软件产品可以存储在计算机可读存储介质中,如ROM/RAM、磁碟、光盘等,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)执行各个实施例或者实施例的某些部分的方法。

[0108] 最后,本发明中应用了具体实施例对本发明的原理及实施方式进行了阐述,以上实施例的说明只是用于帮助理解本发明的方法及其核心思想;同时,对于本领域的一般技术人员,依据本发明的思想,在具体实施方式及应用范围上均会有改变之处,综上所述,本说明书内容不应理解为对本发明的限制。

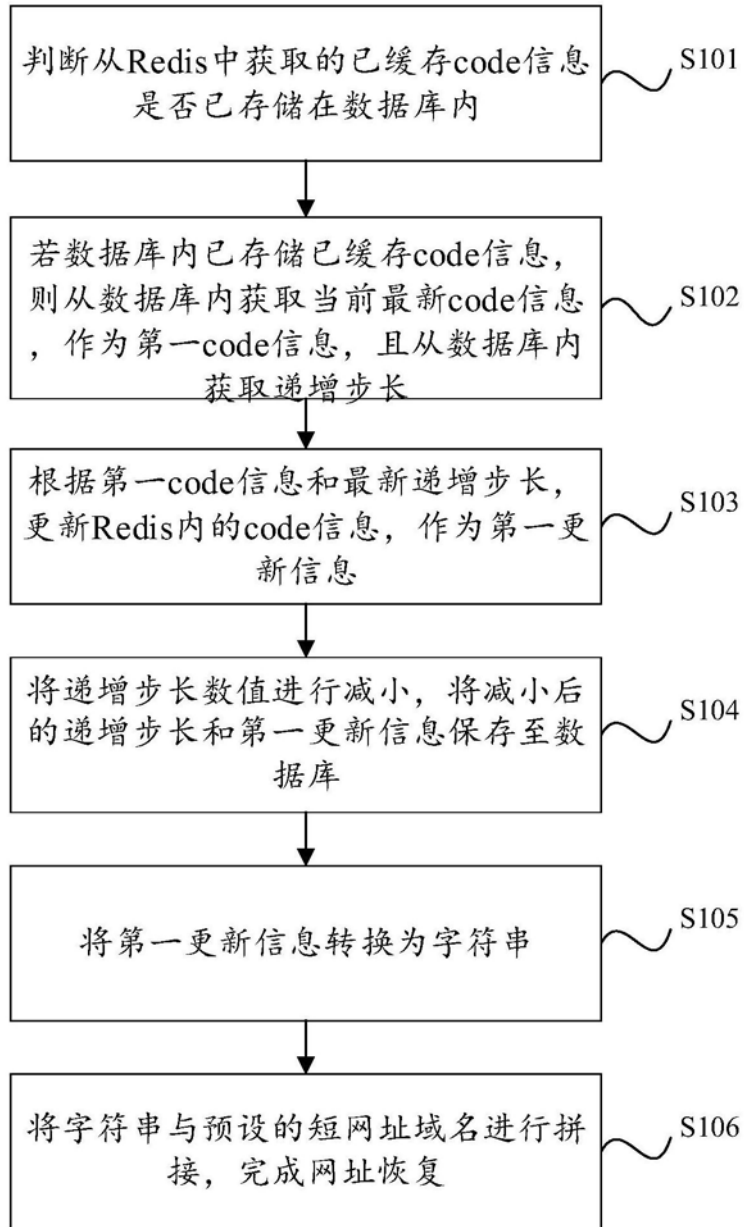


图1

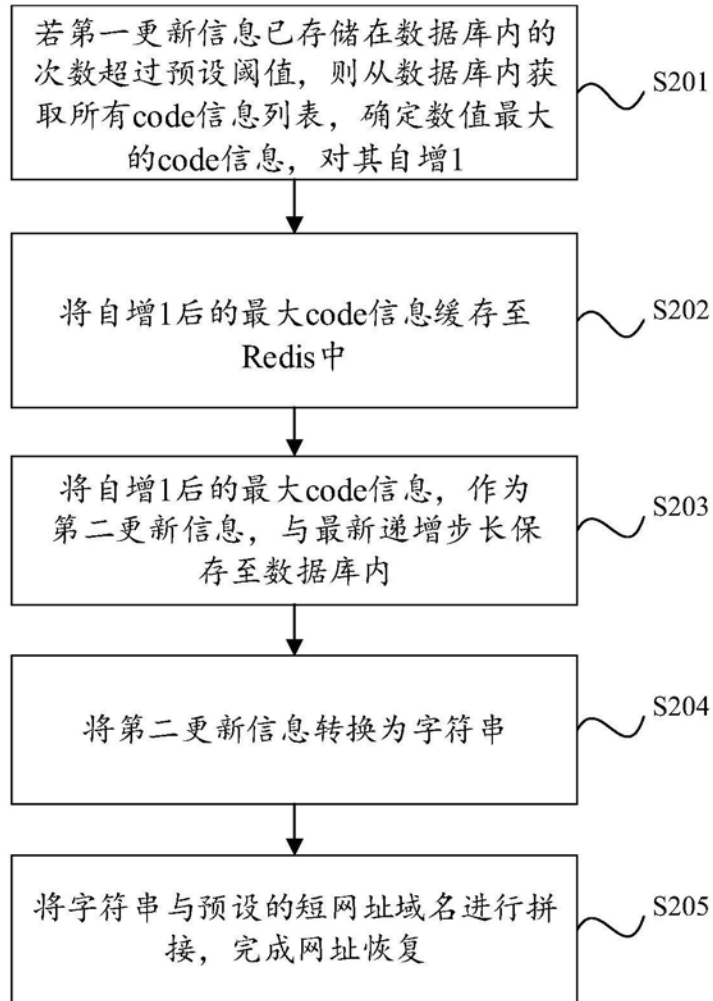


图2

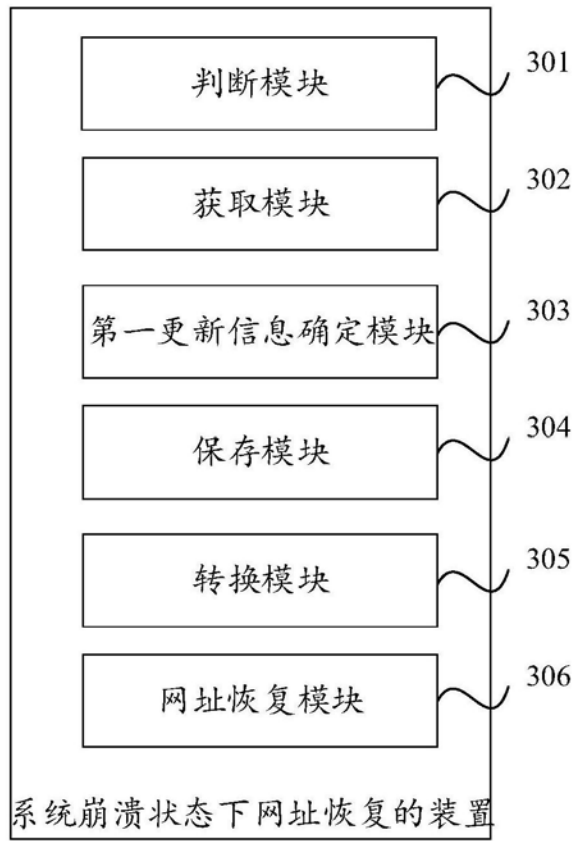


图3

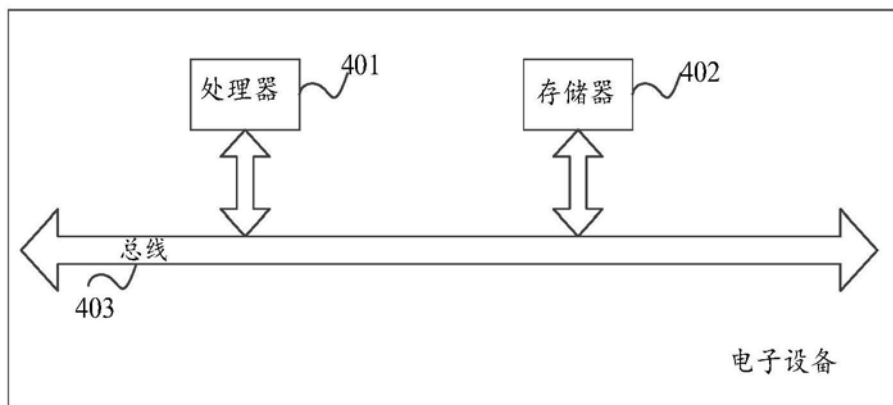


图4