



(12) 发明专利

(10) 授权公告号 CN 113612869 B

(45) 授权公告日 2023. 12. 19

(21) 申请号 202110982714.7

(22) 申请日 2021.08.25

(65) 同一申请的已公布的文献号
申请公布号 CN 113612869 A

(43) 申请公布日 2021.11.05

(73) 专利权人 福建凯米网络科技有限公司
地址 350018 福建省福州市仓山区城门镇
南江滨西大道198号福州海峡国际会
展中心地下一层东区办公中心A-029
号(自贸试验区内)

(72) 发明人 王明超 陈节省 刘旺

(74) 专利代理机构 福州市景弘专利代理事务所
(普通合伙) 35219
专利代理师 张忠波 徐宝珺

(51) Int.Cl.

H04L 61/5046 (2022.01)

H04L 67/02 (2022.01)

G06F 16/955 (2019.01)

H04L 101/672 (2022.01)

(56) 对比文件

CN 106412140 A, 2017.02.15

CN 109313660 A, 2019.02.05

CN 106933854 A, 2017.07.07

CN 110365806 A, 2019.10.22

CN 110213397 A, 2019.09.06

CN 102402558 A, 2012.04.04

US 2019339872 A1, 2019.11.07

审查员 高文娟

权利要求书1页 说明书5页 附图3页

(54) 发明名称

一种短网址生成方法、终端及存储介质

(57) 摘要

本发明公开了一种短网址生成方法、终端及存储介质,包括步骤:接收根据长网址生成对应的短网址的网址转换请求;对所述长网址进行哈希计算得到对应的计算值;判断所述计算值是否已存在于Redis中,所述Redis中存储有短网址和对应的计算值;若是,则返回所述短网址;若否,则生成所述长网址的唯一ID,将所述唯一ID转换为对应的短网址,以及将所述长网址的计算值和所述短网址存入所述Redis中。本发明可避免重复数据的产生,因此可无需进行冲突检测,同时可降低计算量以及存储压力。并且上述技术方案对同一长网址多次转换生成相同的短网址,可避免相同长网址多次转换生成不同短网址。



1. 一种短网址生成方法,其特征在于,包括步骤:
接收根据长网址生成对应的短网址的网址转换请求;
对所述长网址进行哈希计算得到对应的计算值;
判断所述计算值是否已存在于Redis中,所述Redis中存储有短网址和对应的计算值;
若是,则返回所述短网址;若否,则生成所述长网址的唯一ID,将所述唯一ID转换为对应的短网址,以及将所述长网址的计算值和所述短网址存入所述Redis中;
在所述接收根据长网址生成对应的短网址的网址转换请求之前,还包括步骤:
预设短网址的动态参数配置模式;所述动态参数配置模式用于在访问所述短网址时,根据所述动态参数配置模式调整所述短网址对应的长网址的参数,以实现动态参数透传,包括:
判断短网址预设的动态参数传递模式,并根据所述动态参数传递模式结合短网址中的动态参数合并,重新整理编排长网址的参数;以及短网址服务中心进行重定向跳转到长网址地址,并携带重新整理后的参数,其中,所述动态参数配置模式包括:不支持动态参数传递、支持动态参数传递或支持动态参数传递但优先使用原长网址中的参数值。
2. 根据权利要求1所述的短网址生成方法,其特征在于,所述对所述长网址进行哈希计算得到对应的计算值包括步骤:
通过预设的排序算法对所述长网址进行排序得到的字符串,对所述字符串进行消息-摘要计算得到摘要值,所述摘要值为所述计算值。
3. 根据权利要求2所述的短网址生成方法,其特征在于,所述消息-摘要计算为MD5计算,所述摘要值为所述长网址进行MD5计算得到MD5值。
4. 根据权利要求1所述的短网址生成方法,其特征在于,所述生成所述长网址的唯一ID,将所述唯一ID转换为对应的短网址,包括步骤:
将所述长网址插入至mysql数据库,得到所述长网址的自增主键ID;
将所述自增主键ID按照预设的进制转换算法转换成短网址。
5. 根据权利要求4所述的短网址生成方法,其特征在于,所述预设的进制转换算法包括62进制算法。
6. 一种生成短网址的终端,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计算机程序时实现权利要求1至5任一项所述的短网址生成方法。
7. 一种存储介质,其上存储有计算机程序,其特征在于,所述程序被处理器执行时实现如权利要求1至5任意一项所述的短网址生成方法。

一种短网址生成方法、终端及存储介质

技术领域

[0001] 本发明涉及互联网技术领域,特别是涉及一种短网址生成方法、终端及存储介质。

背景技术

[0002] 短网址(ShortURL)是指在形式上比较短的网址。短网址已经广泛被用于互联网,例如二维码生成、短信发送等,借助短网址可以用简短的网址替代原来冗长的网址,因此更便于分享链接。

[0003] 目前短网址主要有以下几种生成方案:

[0004] 1、将长网址利用hash散列算法,计算得到一个hash值作为短码,从而实现长网址和短码直接的映射关系。

[0005] 2、预先批量生成不重复的短码,当有长网址需要转换时,随机读取一个进行返回。

[0006] 3、通过Redis使用设置自增值,然后进行进制转换得到短网址。

[0007] 上述方案在计算得到短网址时都需要进行冲突检测,并且还存在着相同长网址多次转换生成不同短码的问题,以及短网址不支持携带参数,灵活性差的技术问题。

发明内容

[0008] 为此,需要提供一种短网址生成方法用于解决上述短网址需要冲突检测以及相同长网址多次转换生成不同短网址的技术问题。

[0009] 为实现上述目的,发明人提供了一种短网址生成方法,包括步骤:

[0010] 接收根据长网址生成对应的短网址的网址转换请求;

[0011] 对所述长网址进行哈希计算得到对应的计算值;

[0012] 判断所述计算值是否已存在于Redis中,所述Redis中存储有短网址和对应的计算值;

[0013] 若是,则返回所述短网址;若否,则生成所述长网址的唯一ID,将所述唯一ID转换为对应的短网址,以及将所述长网址的计算值和所述短网址存入所述Redis中。

[0014] 进一步的,所述对所述长网址进行哈希计算得到对应的计算值包括步骤:

[0015] 通过预设的排序算法对所述长网址进行排序得到的字符串,对所述字符串进行消息-摘要计算得到摘要值,所述摘要值为所述计算值。

[0016] 进一步的,所述消息-摘要计算为MD5计算,所述摘要值为所述长网址进行MD5计算得到MD5值。

[0017] 进一步的,所述步骤生成所述长网址的唯一ID,将所述唯一ID转换为对应的短网址,包括步骤:

[0018] 将所述长网址插入至mysql数据库,得到所述长网址的自增主键ID;

[0019] 将所述自增主键ID按照预设的进制转换算法转换成短网址。

[0020] 进一步的,在所述接收根据长网址生成对应的短网址的网址转换请求之前,还包括步骤:

[0021] 预设短网址的动态参数配置模式;所述动态参数配置模式用于在访问所述短网址时,根据所述动态参数配置模式调整所述短网址对应的长网址的参数。

[0022] 进一步的,所述动态参数配置模式包括:不支持动态参数传递、支持动态参数传递或支持动态参数传递但优先使用原长网址中的参数值。

[0023] 进一步的,所述预设的进制转换算法包括62进制算法。

[0024] 为解决上述技术问题,本发明还提供了另一技术方案:

[0025] 一种生成短网址的终端,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,所述处理器执行所述计算机程序时实现以上任一技术方案所述的短网址生成方法。

[0026] 为解决上述技术问题,本发明还提供了另一技术方案:

[0027] 一种存储介质,其上存储有计算机程序,所述程序被处理器执行时实现以上任一技术方案所述的短网址生成方法。

[0028] 区别于现有技术,上述技术方案通过长网址的计算值判断该长网址是否有对应的短网址存储于Redis,若有则直接返回该短网址,若不存在则对长网址转换生成短网址并存入Redis中,从而避免重复数据的产生,因此可无需进行冲突检测,同时可降低计算量以及存储压力。并且上述技术方案对同一长网址多次转换生成相同的短网址,可避免相同长网址多次转换生成不同短网址。

附图说明

[0029] 图1为具体实施方式所述短网址生成方法的流程图;

[0030] 图2为具体实施方式所述短网址生成方法的具体流程图;

[0031] 图3为具体实施方式所述生成短网址的终端的模块框图;

[0032] 图4为具体实施方式所述存储介质的示意图;

[0033] 附图标记说明:

[0034] 300、生成短网址的终端;

[0035] 310、处理器;

[0036] 320、存储器;

[0037] 400、存储介质;

具体实施方式

[0038] 为详细说明技术方案的技术内容、构造特征、所实现目的及效果,以下结合具体实施例并配合附图详予说明。

[0039] 请参阅图1,本实施例提供了一种短网址生成方法。通过该短网址生成方法可将长网址转换成对应的短网址,该短网址生成方法包括步骤:

[0040] S101、接收根据长网址生成对应的短网址的网址转换请求;其中,长网址和短网址通常采用不一样的快照和不一样的排名,在形式上长网址的长度要大于短网址,并且长网址中通常包括有“www”而短网址中通常不包括“www”。

[0041] S102、对所述长网址进行哈希计算得到对应的计算值;其中,对长网址进行哈希计算可以使不同长度的长网址得到相同长度的计算值,因此便于存储,可有效减少存储量。

[0042] S103、判断所述计算值是否已存在于Redis中,所述Redis中存储有短网址和对应的计算值;

[0043] 若是,则返回所述短网址;若否,则生成所述长网址的唯一ID,将所述唯一ID转换为对应的短网址,以及将所述长网址的计算值和所述短网址存入所述Redis中。

[0044] 其中,Redis是一个高性能的key-value数据库,Redis中的数据都是缓存在内存中,从而保证了读写的效率。并且Redis会周期性的把更新的数据写入磁盘或者把修改操作写入追加的记录文件,并且在此基础上实现了master-slave(主从)同步。

[0045] 在步骤S103中,若长网址的计算值存储于Redis,则直接从Redis中返回对应的短网址,因此无需进行重复生成短网址,减小系统计算量和存储压力。若长网址的计算值不存在于Redis中,则根据长网址的唯一ID转换为对应的短网址,并且存储于Redis中,由于每个长网址的唯一ID都是不同的,从而保证根据长网址的唯一ID得到的短网址也是不同,从而无需进行冲突检测。

[0046] 由上述描述可知,在该短网址生成方法可避免重复计算生成短网址,降低计算量以及存储压力,并且生成的每个短网址都是不同的,因此无需进行冲突检测。

[0047] 在一实施方式中,所述对所述长网址进行哈希计算得到对应的计算值包括步骤:

[0048] 通过预设的排序算法对所述长网址进行排序得到的字符串,对所述字符串进行消息-摘要计算得到摘要值,所述摘要值为所述计算值。其中,消息-摘要计算优选为MD5计算。由于排序完之后的网址还是很长,匹配性能较差,通过MD5计算则可以将网址长度缩短为32或16位,从而提高其性能。在其他实施方式中,也可以采用MD4、MD3、sha-1、RIPEMD以及Haval等其他的消息-摘要计算算法。MD5即信息-摘要算法5,通过MD5可以让大容量信息被“压缩”成一种保密的格式(就是把一个任意长度的字节串变换成一定长的十六进制数字串)。

[0049] 其中,对长网址进行排序可以得到唯一的字符串A,并将字符串A进行MD5计算可以得到MD5值B(即计算值)。其中,长网址进行排序是指对长网址中的参数进行排序,排序方式可以为长网址的参数名字典序升序或者降序排序,通过排序可以保证一批参数排序之后是得到的字符串A是唯一的。例如:

[0050] 长网址http://www.ktvme.com?ab=122&kk=32&bg=333,升序排序之后,固定为: http://www.ktvme.com?ab=122&bg=333&kk=32。

[0051] 在一实施方式中,所述步骤生成所述长网址的唯一ID,将所述唯一ID转换为对应的短网址,包括步骤:

[0052] 将所述长网址插入至mysql数据库,得到所述长网址的自增主键ID;

[0053] 将所述自增主键ID按照预设的进制转换算法转换成短网址。

[0054] 其中,所述预设的进制转换算法包括62进制算法、52进制算法、42进制算法等。

[0055] mysql数据库是一个关系型数据库,mysql数据库具有体积小、速度快、成本低、开放源代码等优点。

[0056] 在mysql数据库中,每新增一条数据,自增的主键ID会加1。每条数据有一个唯一的标识(即主键ID)。通过执行插入语句将长网址插入至mysql数据库时,可以得到该长网址的自增的主键ID,通过PHP的mysql_insert_ID函数可以获取该自增主键ID。

[0057] 在该实施方式中,通过将长网址插入至mysql数据库中,并通过长网址的自增主键

ID进制转换算法得到短网址,从而可保证短网址的唯一性,因此无需冲突检测。在其他实施方式中,也可以将所述长网址插入至sqlsvr数据库、sqlite数据库等其他数据库中得到对的自增主键ID。

[0058] 考虑到在现有技术中短网址不支持携带参数,短网址的使用灵活性差。在一实施方式中,短网址生成方法在所述接收根据长网址生成对应的短网址的网址转换请求之前,还包括步骤:预设短网址的动态参数配置模式,所述动态参数配置模式用于在访问所述短网址时,根据所述动态参数配置模式调整所述短网址对应的长网址的参数,从而实现动态参数透传的效果,即通过短网址访问长网址时将动态参数传递至对应的长网址。

[0059] 其中,所述动态参数配置模式包括:不支持动态参数传递、支持动态参数传递、支持部分参数传递(即支持动态参数传递但优先使用原长网址中的参数值)中的任意一种或两种上。

[0060] 在实际使用时,可根据短网址的使用场景选择相应的动态参数传递模式(例如支持动态参数传递)。在用户访问短网址行为时,将根据用户预设的参数传递模式判断,对于短网址的动态参数以及长网址的参数进行重新的合并、编排计算,从而达到一个动态参数透传的效果。

[0061] 以下以用户访问短网址为例进行说明。

[0062] 用户点击短网址进行页面访问的步骤包括:

[0063] 步骤一:用户点击短网址。

[0064] 步骤二:短网址服务中心校验短网址正确性。

[0065] 步骤三:判断短网址预设的动态参数传递模式,并根据动态参数传递模式并结合短网址中的动态参数合并、重新整理编排长网址的参数。

[0066] 步骤四:短网址服务中心进行重定向跳转到长网址地址,并携带重新整理后的参数。

[0067] 步骤五:用户看到期望网页。

[0068] 为了更清楚的理解短网址动态参数透传过程,以下以一具体实例进行说明。

[0069] 假设有一个长网址A为:`http://www.ktvme.com/a/b/c?id=1&type=4&abc=222`;该长网址在三种动态参数传递模式下都转换了一个短网址如下:

[0070] 1)、模式1:`https://s.ktvme.com/asbxjs`;模式1不支持动态参数传递;

[0071] 2)、模式2:`https://s.ktvme.com/sxsscs`;模式2支持动态参数传递,但是原长网址中的参数值优先使用;

[0072] 3)、模式3:`https://s.ktvme.com/xssacsa`;模式3支持动态参数,新参数值会覆盖原长网址的参数值;

[0073] 对于以上三种模式的短网址的生成差别仅在于动态参数模式的选项不一样,其他生成逻辑均一样。

[0074] 三种动态参数传递模式得到短网址(即短链接)假设以相同的方式投放,并携带相同的参数如下:

[0075] 4)、模式1:`https://s.ktvme.com/asbxjs?type=10&kk=2`;

[0076] 5)、模式2:`https://s.ktvme.com/sxsscs?type=10&kk=2`;

[0077] 6)、模式3:`https://s.ktvme.com/xssacsa?type=10&kk=2`;

[0078] 当用户访问4)时,由于选择的是模式1,模式1不传递动态参数,因此短网址(即短链接)上的参数 $type=10\&kk=2$ 将会被抛弃掉,用户直接访问到原长网址A。

[0079] 当用户访问5)时,选择的是模式2,该模式允许传递动态参数,但是原长网址中的参数值优先使用。因此 $type=10$ 被抛弃, $kk=2$ 被合并到参数,用户访问到 $http://www.ktvme.com/a/b/c?id=1\&type=4\&abc=222\&kk=2$ 。

[0080] 当用户访问6)时,选择模式3,该允许传递动态参数,新参数值会覆盖原长网址的参数值,因此 $type=10$ 会覆盖 $type=4$,用户访问到地址 $http://www.ktvme.com/a/b/c?id=1\&type=10\&abc=222\&kk=2$ 。

[0081] 通过以上的动态参数传递,既可以通过短网址服务获取到更短的短网址链接,而且可以传递参数,从而提高短网址应用的灵活性和扩展性。

[0082] 如图2所示,为短网址生成方法的一具体实施方式,该短网址生成该当包括步骤:

[0083] S201、接收长网址A转换请求,以及预设动态参数配置模式B;其中,所述态参数配置模式B即是否支持动态参数B,所述动态参数配置模型包括:不支持动态参数传递、支持动态参数传递、支持部分参数传递(即支持动态参数传递但优先使用原长网址中的参数值;

[0084] S202、使用排序算法将长网址转换成唯一字符串C;

[0085] S203、将字符串C进行MD5计算得到对应的MD5值D;

[0086] S204、判断MD5值D是否存在于Redis中;若是,则跳转至S208;若否,则执行S205;

[0087] S205、将长网址以及动态参数配置模式B插入至mysql数据库,得到自增主键ID;

[0088] S206、使用进制算法将自增主键ID转换成短码E;

[0089] S207、将MD5值D与短码E以Key-value存储在Redis中;

[0090] S208、返回短网址。

[0091] 通过该具体实施方式可知,该短网址生成方法使用mysql数据库的自增主键ID进行进制转换得到短码,因此转换快速且无需冲突检测。并且,在排序算法后采用MD5算法可以拦截重复请求,减少存储空间浪费。进一步的,该短网址生成方法通过预设动态参数配置模式还支持动态参数透传,动态参数的应用,使得大量近似的长网址(如仅有一个参数不一致的链接)无需重复转换,仅需要转换一次,其余链接只需要使用动态参数进行传递即可实现,大大提高灵活性,减少大量不必要的请求。

[0092] 如图3所示,在一实施方式中提供了一种生成短网址的终端300。生成短网址的终端300用于将长网址转换成对应的短网址。生成短网址的终端300包括存储器320,处理器310及存储在存储器320上并可在处理器上运行的计算机程序,所述处理器310执行所述计算机程序时实现任一上述实施方式中所述的短网址生成方法。

[0093] 如图4所示,在一实施方式中提供了一种存储介质400,存储介质400上存储有计算机程序,所述程序被处理器执行所述计算机程序时实现任一上述实施方式中所述的短网址生成方法。

[0094] 需要说明的是,尽管在本文中已经对上述各实施例进行了描述,但并非因此限制本发明的专利保护范围。因此,基于本发明的创新理念,对本文所述实施例进行的变更和修改,或利用本发明说明书及附图内容所作的等效结构或等效流程变换,直接或间接地将以上技术方案运用在其他相关的技术领域,均包括在本发明的专利保护范围之内。

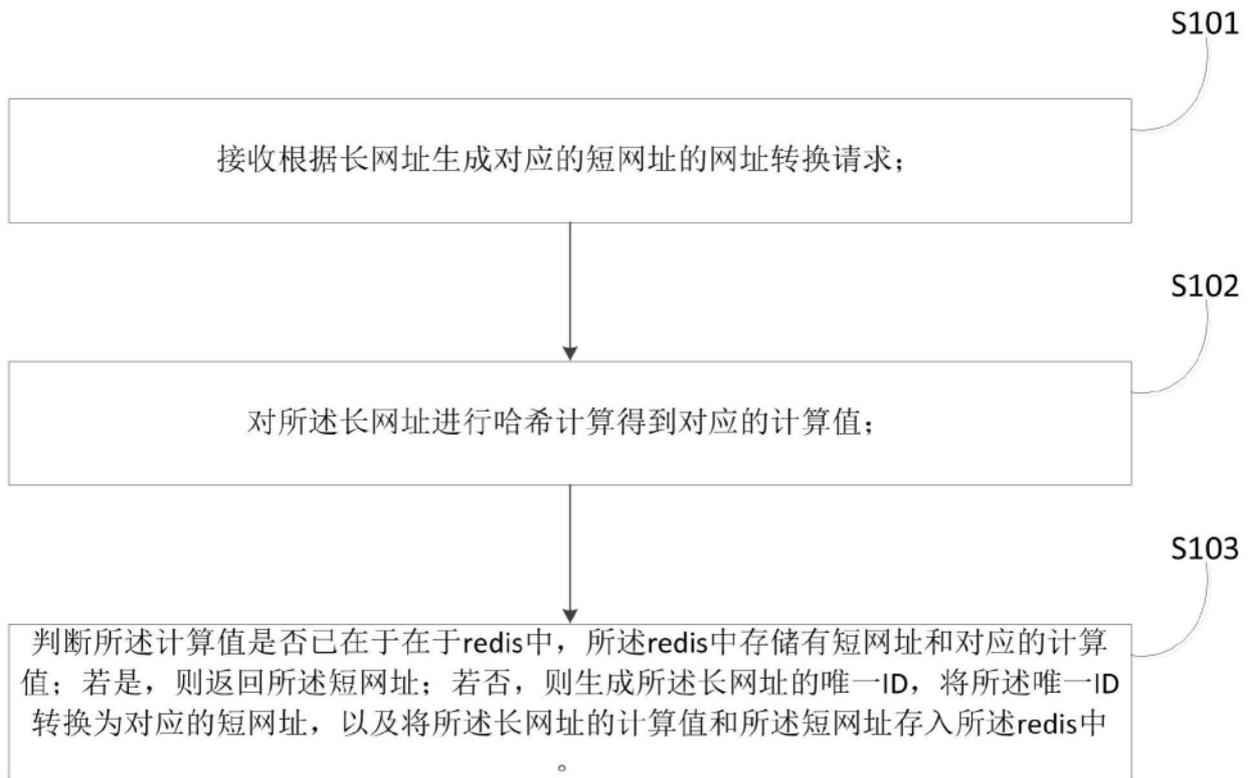


图1

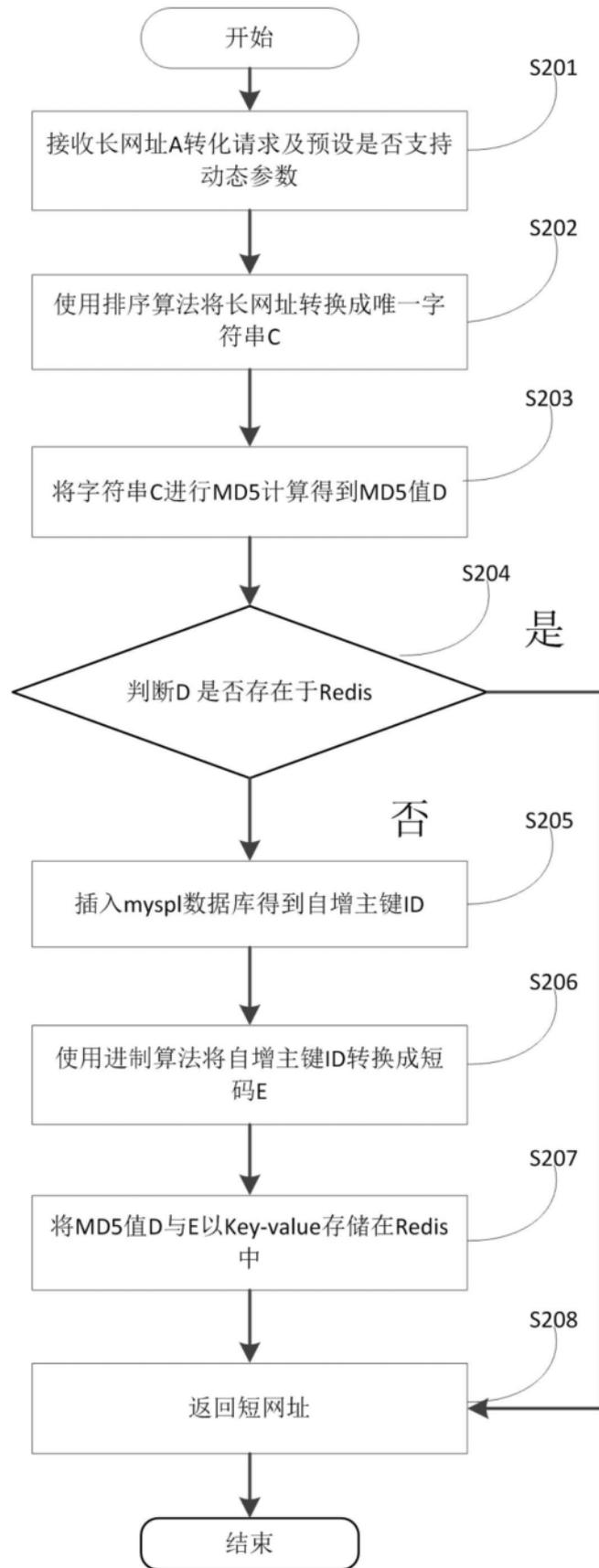


图2



图3



图4