



(12) 发明专利申请

(10) 申请公布号 CN 105512320 A

(43) 申请公布日 2016. 04. 20

(21) 申请号 201510958540. 5

(22) 申请日 2015. 12. 18

(71) 申请人 北京金山安全软件有限公司  
地址 100085 北京市海淀区小营西路 33 号  
二层东区

(72) 发明人 莫广

(74) 专利代理机构 北京柏杉松知识产权代理事  
务所(普通合伙) 11413  
代理人 马敬 项京

(51) Int. Cl.  
G06F 17/30(2006. 01)

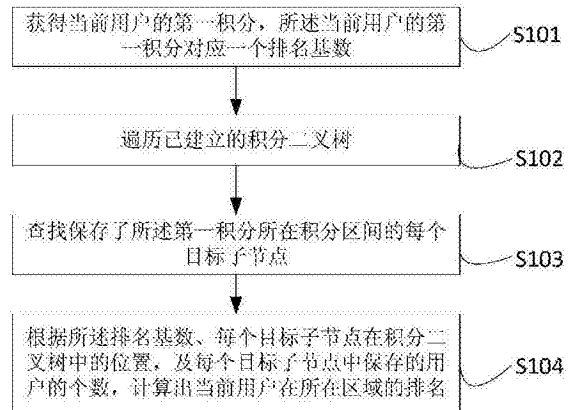
权利要求书3页 说明书11页 附图4页

(54) 发明名称

一种用户排名获得方法、装置及服务器

(57) 摘要

本发明实施例提供了一种用户排名获得方法、装置及服务器,获得当前用户的第一积分,所述当前用户的第一积分对应一个排名基数,遍历已建立的积分二叉树,所述积分二叉树的每个节点中保存了一个积分区间及积分处于该积分区间内的用户个数,查找保存了所述第一积分所在积分区间的每个目标子节点,根据所述排名基数、每个目标子节点在积分二叉树中的位置,及每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名。本发明通过遍历累加积分二叉树中每个目标子节点中保存的用户个数获得各个用户的排名,由于累加次数较少且累加次数与用户量无关,达到减少数据计算量,减少系统资源占用的目的。



1. 一种用户排名获得方法,其特征在于,应用于每个区域的服务器,所述方法包括:  
获得当前用户的第一积分,所述当前用户的第一积分对应一个排名基数;  
遍历已建立的积分二叉树,所述积分二叉树为当前用户所在区域的积分二叉树,所述积分二叉树的每个节点中保存了一个积分区间及积分处于该积分区间内的用户个数;  
查找保存了所述第一积分所在积分区间的每个目标子节点;  
根据所述排名基数、每个目标子节点在积分二叉树中的位置,及每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名。

2. 根据权利要求1所述的方法,其特征在于,所述获得当前用户的第一积分包括:  
获得当前用户的备份数据;  
根据所述获得的备份数据计算出当前用户的第一积分。

3. 根据权利要求1所述的方法,其特征在于,所述根据所述排名基数、每个目标子节点在积分二叉树中的位置,及每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名,包括:

在所述积分二叉树的根节点的子节点中找到目标子节点,将找到的目标子节点确定为当前节点,判断当前节点是否为左子节点,如果是,则将右子节点包含的用户个数累加到所述排名基数上,将累加后得到的排名作为当前用户的排名,如果否,则将所述排名基数作为当前用户的排名,其中,每个非叶子节点有两个子节点,分别为左子节点和右子节点,所述左子节点为低分子节点,所述右子节点为高分子节点;

在当前节点的子节点中查找下一个目标子节点,将该找到的目标子节点确定为当前节点,判断当前节点是否为左子节点,如果是,则将右子节点包含的用户个数累加到所述当前用户的排名上,将累加后得到的排名作为当前用户的排名,如果否,则当前用户的排名不变,在当前节点的子节点中查找再下一个目标子节点,将该找到的再下一个目标子节点确定为当前节点,返回执行所述判断当前节点是否为左子节点的步骤;

在当前节点为叶子节点时,将当前用户的排名确定为用户在所在区域的排名。

4. 根据权利要求1所述的方法,其特征在于,所述积分二叉树的根节点保存的积分区间为:当前用户所在区域中的积分中的最小值与最大值组成的积分区间;所述积分二叉树的根节点保存的积分处于该积分区间内的用户个数为当前用户所在区域中的用户的总个数;每个节点包含的积分区间的长度为积分区间中的最大值与最小值的差,且叶子节点所包含的积分区间的长度为1。

5. 根据权利要求1所述的方法,其特征在于,所述方法还包括:

定时统计服务器所在区域内积分处于各个积分区间内的用户个数,并根据统计结果更新服务器所在区域的积分二叉树。

6. 根据权利要求1所述的方法,其特征在于,所述方法还包括:

将当前用户及第一积分发送给所有区域的各个服务器;  
接收各个服务器根据所述第一积分计算出的所述当前用户在各个区域的排名;  
将接收到的各个区域的排名累加;  
将累加后的排名减一作为当前用户在所有区域中的总排名。

7. 一种用户排名获得装置,其特征在于,应用于每个区域的服务器,所述装置包括:获得模块、遍历模块、查找模块和计算模块,

所述获得模块,用于获得当前用户的第一积分,所述当前用户的第一积分对应一个排名基数;

所述遍历模块,用于遍历已建立的积分二叉树,所述积分二叉树为当前用户所在区域的积分二叉树,所述积分二叉树的每个节点中保存了一个积分区间及积分处于该积分区间内的用户个数;

所述查找模块,用于查找保存了所述第一积分所在积分区间的每个目标子节点;

所述计算模块,用于根据所述排名基数、每个目标子节点在积分二叉树中的位置,及每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名。

8. 根据权利要求7所述的装置,其特征在于,所述获得模块,包括:第一获得子模块和第一计算子模,

所述第一获得子模块,用于获得当前用户的备份数据;

所述第一计算子模块,用于根据所述获得的备份数据计算出当前用户的第一积分,所述当前用户的第一积分对应有一个排名基数。

9. 根据权利要求7所述的装置,其特征在于,所述计算模块,包括:第一确定子模块、第一判断子模块、第二确定子模块、第二判断子模块、第三确定子模块和确定子模块,

所述第一确定子模块,用于在所述积分二叉树的根节点的子节点中找到目标子节点,将找到的目标子节点确定为当前节点;

所述第一判断子模块,用于判断当前节点是否为左子节点,如果是,则触发第一排名子模块,如果不是,则触发第二排名子模块,其中,每个非叶子节点有两个子节点,分别为左子节点和右子节点,所述左子节点为低分子节点,所述右子节点为高分子节点;

所述第二确定子模块,用于在当前节点的子节点中查找下一个目标子节点,将该找到的目标子节点确定为当前节点;

所述第二判断子模块,用于判断当前节点是否为左子节点,如果是,则触发第三排名子模块,如果不是,则触发第四排名子模块;

所述第三确定子模块,用于在当前节点的子节点中查找再下一个目标子节点,将该找到的再下一个目标子节点确定为当前节点,触发第二判断子模块;

所述确定子模块,用于在当前节点为叶子节点时,将当前用户的排名确定为用户在所在区域的排名;

所述第一排名子模块,用于将右子节点包含的用户个数累加到所述排名基数上,将累加后得到的排名作为当前用户的排名;

所述第二排名子模块,用于将所述排名基数作为当前用户的排名;

所述第三排名子模块,用于将右子节点包含的用户个数累加到所述当前用户的排名上,将累加后得到的排名作为当前用户的排名;

所述第四排名子模块,用于当前用户的排名不变。

10. 一种服务器,其特征在于,所述服务器包括:

壳体、处理器、存储器、电路板和电源电路,其中,电路板安置在壳体围成的空间内部,处理器和存储器设置在电路板上;电源电路,用于为电子设备的各个电路或器件供电;存储器用于存储可执行程序代码;处理器通过读取存储器中存储的可执行程序代码来运行与可执行程序代码对应的程序,以用于执行以下步骤:

获得当前用户的第一积分,所述当前用户的第一积分对应一个排名基数;

遍历已建立的积分二叉树,所述积分二叉树为当前用户所在区域的积分二叉树,所述积分二叉树的每个节点中保存了一个积分区间及积分处于该积分区间内的用户个数;

查找保存了所述第一积分所在积分区间的每个目标子节点;

根据所述排名基数、每个目标子节点在积分二叉树中的位置,及每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名。

## 一种用户排名获得方法、装置及服务器

### 技术领域

[0001] 本发明涉及计算机应用技术领域,特别是涉及一种用户排名获得方法、装置及服务器。

### 背景技术

[0002] 目前,很多应用系统,例如:视频应用系统或购物应用系统等,需要对用户进行排名,使用户可以根据获得的排名兑换相应的礼品。

[0003] 现有技术中,通过以下方式来获得用户排名:

[0004] 首先,每个区域的服务器获得区域中每个用户的备份数据,也就是用户备份数据的备份/还原操作次数等信息。然后,基于每个用户的备份数据,确定每个用户的积分,保存每个用户及对应的积分。再根据保存的各个用户的积分,将所有用户的积分按多少顺序进行排序,将排序的结果作为各个用户在其所在区域的排名结果,并保存。当需要获得某用户的排名时,在排名结果中查找该用户,如果能够查找到该用户的排名,则将查找到的排名作为该用户在所在区域的排名。如果在排名结果中没有查找到该用户的排名,则重新将包括该用户的所有用户的积分按多少顺序进行排序,以获得该用户在所在区域的排名。

[0005] 上述过程中,由于是通过对所有用户的积分进行排序而获得各个用户的排名,因此,对每个区域的服务器来说计算数据量大,系统资源占用较多。

### 发明内容

[0006] 本发明实施例的目的在于提供一种用户排名获得方法、装置及服务器,以解决计算数据量大和系统资源占用较多的问题。具体技术方案如下:

[0007] 本发明提供了一种用户排名获得方法,应用于每个区域的服务器,所述方法包括:

[0008] 获得当前用户的第一积分,所述当前用户的第一积分对应一个排名基数;

[0009] 遍历已建立的积分二叉树,所述积分二叉树为当前用户所在区域的积分二叉树,所述积分二叉树的每个节点中保存了一个积分区间及积分处于该积分区间内的用户个数;

[0010] 查找保存了所述第一积分所在积分区间的每个目标子节点;

[0011] 根据所述排名基数、每个目标子节点在积分二叉树中的位置,及每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名。

[0012] 可选的,所述获得当前用户的第一积分包括:

[0013] 获得当前用户的备份数据;

[0014] 根据所述获得的备份数据计算出当前用户的第一积分。

[0015] 可选的,所述根据所述排名基数、每个目标子节点在积分二叉树中的位置,及每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名,包括:

[0016] 在所述积分二叉树的根节点的子节点中找到目标子节点,将找到的目标子节点确定为当前节点,判断当前节点是否为左子节点,如果是,则将右子节点包含的用户个数累加到所述排名基数上,将累加后得到的排名作为当前用户的排名,如果否,则将所述排名基数

作为当前用户的排名,其中,每个非叶子节点有两个子节点,分别为左子节点和右子节点,所述左子节点为低分子节点,所述右子节点为高分子节点;

[0017] 在当前节点的子节点中查找下一个目标子节点,将该找到的目标子节点确定为当前节点,判断当前节点是否为左子节点,如果是,则将右子节点包含的用户个数累加到所述当前用户的排名上,将累加后得到的排名作为当前用户的排名,如果否,则当前用户的排名不变,在当前节点的子节点中查找再下一个目标子节点,将该找到的再下一个目标子节点确定为当前节点,返回执行所述判断当前节点是否为左子节点的步骤;

[0018] 在当前节点为叶子节点时,将当前用户的排名确定为用户在所在区域的排名。

[0019] 可选的,所述积分二叉树的根节点保存的积分区间为:当前用户所在区域中的积分中的最小值与最大值组成的积分区间;所述积分二叉树的根节点保存的积分处于该积分区间内的用户个数为当前用户所在区域中的用户的总个数;每个节点包含的积分区间的长度为积分区间中的最大值与最小值的差,且叶子节点所包含的积分区间的长度为1。

[0020] 可选的,所述方法还包括:

[0021] 定时统计服务器所在区域内积分处于各个积分区间内的用户个数,并根据统计结果更新服务器所在区域的积分二叉树。

[0022] 可选的,所述方法还包括:

[0023] 将当前用户及第一积分发送给所有区域的各个服务器;

[0024] 接收各个服务器根据所述第一积分计算出的所述当前用户在各个区域的排名;

[0025] 将接收到的各个区域的排名累加;

[0026] 将累加后的排名减一作为当前用户在所有区域中的总排名。

[0027] 本发明提供了一种用户排名获得装置,应用于每个区域的服务器,所述装置包括:获得模块、遍历模块、查找模块和计算模块,

[0028] 所述获得模块,用于获得当前用户的第一积分,所述当前用户的第一积分对应一个排名基数;

[0029] 所述遍历模块,用于遍历已建立的积分二叉树,所述积分二叉树为当前用户所在区域的积分二叉树,所述积分二叉树的每个节点中保存了一个积分区间及积分处于该积分区间内的用户个数;

[0030] 所述查找模块,用于查找保存了所述第一积分所在积分区间的每个目标子节点;

[0031] 所述计算模块,用于根据所述排名基数、每个目标子节点在积分二叉树中的位置,及每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名。

[0032] 可选的,所述获得模块,包括:第一获得子模块和第一计算子模,

[0033] 所述第一获得子模块,用于获得当前用户的备份数据;

[0034] 所述第一计算子模块,用于根据所述获得的备份数据计算出当前用户的第一积分,所述当前用户的第一积分对应有一个排名基数。

[0035] 可选的,所述计算模块,包括:第一确定子模块、第一判断子模块、第二确定子模块、第二判断子模块、第三确定子模块和确定子模块,

[0036] 所述第一确定子模块,用于在所述积分二叉树的根节点的子节点中找到目标子节点,将找到的目标子节点确定为当前节点;

[0037] 所述第一判断子模块,用于判断当前节点是否为左子节点,如果是,则触发第一排

名子模块, 如果否, 则触发第二排名子模块, 其中, 每个非叶子节点有两个子节点, 分别为左子节点和右子节点, 所述左子节点为低分子节点, 所述右子节点为高分子节点;

[0038] 所述第二确定子模块, 用于在当前节点的子节点中查找下一个目标子节点, 将该找到的目标子节点确定为当前节点;

[0039] 所述第二判断子模块, 用于判断当前节点是否为左子节点, 如果是, 则触发第三排名子模块, 如果否, 则触发第四排名子模块;

[0040] 所述第三确定子模块, 用于在当前节点的子节点中查找再下一个目标子节点, 将该找到的再下一个目标子节点确定为当前节点, 触发第二判断子模块;

[0041] 所述确定子模块, 用于在当前节点为叶子节点时, 将当前用户的排名确定为用户在所在区域的排名;

[0042] 所述第一排名子模块, 用于将右子节点包含的用户个数累加到所述排名基数上, 将累加后得到的排名作为当前用户的排名;

[0043] 所述第二排名子模块, 用于将所述排名基数作为当前用户的排名;

[0044] 所述第三排名子模块, 用于将右子节点包含的用户个数累加到所述当前用户的排名上, 将累加后得到的排名作为当前用户的排名;

[0045] 所述第四排名子模块, 用于当前用户的排名不变。

[0046] 可选的, 所述积分二叉树的根节点保存的积分区间为: 当前用户所在区域中的积分中的最小值与最大值组成的积分区间; 所述积分二叉树的根节点保存的积分处于该积分区间内的用户个数为当前用户所在区域中的用户的总个数; 每个节点包含的积分区间的长度为积分区间中的最大值与最小值的差, 且叶子节点所包含的积分区间的长度为1。

[0047] 可选的, 所述装置还包括: 更新模块,

[0048] 所述更新模块, 用于定时统计服务器所在区域内积分处于各个积分区间内的用户个数, 并根据统计结果更新服务器所在区域的积分二叉树。

[0049] 可选的, 所述装置还包括: 发送模块、接收模块、累加模块和确定模块,

[0050] 所述发送模块, 用于将当前用户及第一积分发送给所有区域的各个服务器;

[0051] 所述接收模块, 用于接收各个服务器根据所述第一积分计算出的所述当前用户在各个区域的排名;

[0052] 所述累加模块, 用于将接收到的各个区域的排名累加;

[0053] 所述确定模块, 用于将累加后的排名减一作为当前用户在所有区域中的总排名。

[0054] 本发明提供了一种服务器, 所述服务器包括:

[0055] 壳体、处理器、存储器、电路板和电源电路, 其中, 电路板安置在壳体围成的空间内部, 处理器和存储器设置在电路板上; 电源电路, 用于为电子设备的各个电路或器件供电; 存储器用于存储可执行程序代码; 处理器通过读取存储器中存储的可执行程序代码来运行与可执行程序代码对应的程序, 以用于执行以下步骤:

[0056] 获得当前用户的第一积分, 所述当前用户的第一积分对应一个排名基数;

[0057] 遍历已建立的积分二叉树, 所述积分二叉树为当前用户所在区域的积分二叉树, 所述积分二叉树的每个节点中保存了一个积分区间及积分处于该积分区间内的用户个数;

[0058] 查找保存了所述第一积分所在积分区间的每个目标子节点;

[0059] 根据所述排名基数、每个目标子节点在积分二叉树中的位置, 及每个目标子节点

中保存的用户的个数,计算出当前用户在所在区域的排名。

[0060] 为达到上述目的,本发明还提供了一种存储介质,其中,该存储介质用于存储应用程序,所述应用程序用于在运行时执行本申请所述的一种用户排名获得方法。其中,本申请所述的一种用户排名获得方法,应用于每个区域的服务器,所述方法包括:

[0061] 获得当前用户的第一积分,所述当前用户的第一积分对应一个排名基数;

[0062] 遍历已建立的积分二叉树,所述积分二叉树为当前用户所在区域的积分二叉树,所述积分二叉树的每个节点中保存了一个积分区间及积分处于该积分区间内的用户个数;

[0063] 查找保存了所述第一积分所在积分区间的每个目标子节点;

[0064] 根据所述排名基数、每个目标子节点在积分二叉树中的位置,及每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名。

[0065] 为达到上述目的,本发明还提供了一种应用程序,其中,该应用程序用于在运行时执行本申请所述的一种用户排名获得方法。其中,本申请所述的一种用户排名获得方法,应用于每个区域的服务器,所述方法包括:

[0066] 获得当前用户的第一积分,所述当前用户的第一积分对应一个排名基数;

[0067] 遍历已建立的积分二叉树,所述积分二叉树为当前用户所在区域的积分二叉树,所述积分二叉树的每个节点中保存了一个积分区间及积分处于该积分区间内的用户个数;

[0068] 查找保存了所述第一积分所在积分区间的每个目标子节点;

[0069] 根据所述排名基数、每个目标子节点在积分二叉树中的位置,及每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名。

[0070] 本发明实施例提供的一种用户排名获得方法、装置及服务器,获得当前用户的第一积分,所述当前用户的第一积分对应一个排名基数,遍历已建立的积分二叉树,所述积分二叉树为当前用户所在区域的积分二叉树,所述积分二叉树的每个节点中保存了一个积分区间及积分处于该积分区间内的用户个数,查找保存了所述第一积分所在积分区间的每个目标子节点,根据所述排名基数、每个目标子节点在积分二叉树中的位置,及每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名。本发明通过遍历累加积分二叉树中每个目标子节点中保存的用户个数获得各个用户的排名,由于累加次数较少且累加次数与用户量无关,达到减少数据计算量,减少系统资源占用的目的。

## 附图说明

[0071] 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。

[0072] 图1为本发明实施例提供的一种用户排名获得方法的流程示意图;

[0073] 图2为本发明实施例提供的当前用户A所在区域的积分二叉树的示意图;

[0074] 图3为本发明实施例提供的另一种用户排名获得方法的流程示意图;

[0075] 图4为本发明实施例提供的另一种用户排名获得方法的流程示意图;

[0076] 图5为本发明实施例提供的计算当前用户A在所有区域中的总排名的示意图;

[0077] 图6为本发明实施例提供的一种用户排名获得装置的结构示意图;



[0078] 图7为本发明实施例提供的一种服务器的结构示意图。

### 具体实施方式

[0079] 下面将结合本发明实施例中的附图,对本发明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发明一部分实施例,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前提下所获得的所有其他实施例,都属于本发明保护的范围。

[0080] 如图1所示,本发明实施例提供的一种用户排名获得方法,应用于每个区域的服务器,所述方法可以包括:

[0081] S101:获得当前用户的第一积分,所述当前用户的第一积分对应一个排名基数。

[0082] 服务器根据用户就近的原则存储数据,所述数据可以包括图片、短信或视频等。

[0083] 例如:新注册的用户来自印度,离印度最近的服务器部署在新加坡,那么用户所在区域的服务器为新加坡部署的服务器,新加坡部署的服务器中将存储该用户存储的图片、短信或视频等。

[0084] 在当前用户对存储数据进行备份/还原操作时,当前用户所在区域的服务器将备份/还原操作对应的精简数据写入缓冲区。

[0085] 例如:用户A第一次备份了10张图片,则在缓冲区写入{A:[bak,pic,10]},1分钟后,用户A第二次原了20条sms短信,则在缓冲区继续写入,此时缓冲区数据更新为{A:[[bak,pic,10],[res,sms,20]]}。

[0086] 当在缓冲区预设超时时间内,当前用户对存储数据没有进行备份/还原操作时,则将以上缓冲区的精简数据写入消息队列作为备份数据,等待服务器读取并进行下一步处理。

[0087] 例如:缓冲区预设超时时间为5分钟,5分钟后用户A没有操作,此时将{A:[[bak,pic,10],[res,sms,20]]}写入消息队列。

[0088] 其中,步骤S101可以包括:

[0089] 获得当前用户的备份数据;

[0090] 根据所述获得的备份数据计算出当前用户的第一积分。

[0091] 读取消息队列中的当前用户的备份数据,根据所述备份数据中备份/还原操作次数等信息计算出当前用户的第一积分。实际应用中,按照预设的积分获得原则,备份/还原操作针对的类型不同,则获得积分也可以不相同。

[0092] 例如:假设备份1张图片获得2分,还原1条sms短信记录获得1分,则对于以下的数据:{A:[[bak,pic,50],[res,sms,100]]},获得的积分为: $50*2+100*1=200$ ,因此用户A的第一积分为200分。

[0093] S102:遍历已建立的积分二叉树。

[0094] 其中,所述积分二叉树为当前用户所在区域的积分二叉树,所述积分二叉树的每个节点中保存了一个积分区间及积分处于该积分区间内的用户个数。

[0095] 具体的,所述积分二叉树的根节点保存的积分区间为:当前用户所在区域中的积分中的最小值与最大值组成的积分区间;所述积分二叉树的根节点保存的积分处于该积分区间内的用户个数为当前用户所在区域中的用户的总个数;每个节点包含的积分区间的长

度为积分区间中的最大值与最小值的差,且叶子节点所包含的积分区间的长度为1。

[0096] 由于用户的积分数据量庞大,本发明实施例对每个区域的用户的积分数据进行积分分区统计,建立积分二叉树,所述积分二叉树中的每个节点的形式为 $[a, b, c]$ , $a$ 为当前用户所在区域中的积分中的最小值, $b$ 为当前用户所在区域中的积分中的最大值, $c$ 表示积分处于该积分区间内的用户个数。

[0097] 例如:当前用户A所在区域中的最高积分不超过1000000分,用户A所在区域中用户的总个数为200000人,其中,所在区域中用户积分为0分的有2人,用户积分为1分的有4人,用户积分为499000分的有30人,用户积分为999998分的有1人,用户积分为999999分的有1人等,则将 $[0, 1000000, 200000)$ 作为积分二叉树的根节点,根节点的积分区间的长度为1000000,然后把根节点保存的积分区间平均分为两个积分区间 $[0, 500000, 150000)$ 和 $[500000, 1000000, 50000)$ 保存在根节点的子节点中,依次类推,最终得到1000000个叶子节点保存的区间 $[0, 1, 2)$ 、 $[1, 2, 4)$ 、……、 $[499000, 499001, 30)$ 、……、 $[999998, 999999, 1)$ 、 $[999999, 1000000, 1)$ ,由此建立用户A所在区域的积分二叉树如图2所示。

[0098] S103:查找保存了所述第一积分所在积分区间的每个目标子节点。

[0099] 由于保存所述第一积分所在积分区间的目标子节点有多个,因此,在已建立的积分二叉树中,查找保存了所述第一积分所在积分区间的每个目标子节点。

[0100] S104:根据所述排名基数、每个目标子节点在积分二叉树中的位置,及每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名。

[0101] 具体的,本步骤中找到保存所述第一积分所在积分区间的每个目标子节点,判断找到的每个目标子节点在积分二叉树中的位置,根据判断结果及当前用户的第一积分对应的排名基数累加找到的每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名。

[0102] 其中,步骤S104可以包括:

[0103] 在所述积分二叉树的根节点的子节点中找到目标子节点,将找到的目标子节点确定为当前节点,判断当前节点是否为左子节点,如果是,则将右子节点包含的用户个数累加到所述排名基数上,将累加后得到的排名作为当前用户的排名,如果否,则将所述排名基数作为当前用户的排名,其中,每个非叶子节点有两个子节点,分别为左子节点和右子节点,所述左子节点为低分子节点,所述右子节点为高分子节点;

[0104] 在当前节点的子节点中查找下一个目标子节点,将该找到的目标子节点确定为当前节点,判断当前节点是否为左子节点,如果是,则将右子节点包含的用户个数累加到所述当前用户的排名上,将累加后得到的排名作为当前用户的排名,如果否,则当前用户的排名不变,在当前节点的子节点中查找再下一个目标子节点,将该找到的再下一个目标子节点确定为当前节点,返回执行所述判断当前节点是否为左子节点的步骤;

[0105] 在当前节点为叶子节点时,将当前用户的排名确定为当前用户在所在区域的排名。

[0106] 具体的,由于左子节点为低分子节点,右子节点为高分子节点,如果在根节点的子节点中找到的目标子节点为左子节点,那么当前用户的排名在右子节点包含的用户个数之后,则将右子节点包含的用户个数累加到所述排名基数上,如果在除根节点的子节点以外的子节点中找到的目标子节点为左子节点,则将右子节点包含的用户个数累加到所述当前用户的排名上。

[0107] 例如:当前用户A的积分为499000分,当前用户A对应的排名基数为0,首先,在图2所示的已建立的积分二叉树中,找到保存了积分区间为 $[0, 500000, 150000)$ 的节点,该节点为左子节点,所以将右子节点包含的用户个数累加到0上,也就是当前用户A的排名为50000名,依次类推,继续遍历累加直到找到保存了499000分的叶子节点 $[499000, 499001, 30)$ ,找到的节点约有20多个,即累加20多次,整个累加过程完成,得到当前用户A的排名。

[0108] 由于积分二叉树结构的体积实现起来相对较小,因此它存储的积分数据可以保存在内存中,以实现快速查询和修改。

[0109] 在图1所示方法实施例基础上,如图3所示,本发明提供的另一种用户排名获得方法,应用于每个区域的服务器,该方法在图1所示的步骤S101-S104后还可以包括:

[0110] S301:定时统计服务器所在区域内积分处于各个积分区间内的用户个数,并根据统计结果更新服务器所在区域的积分二叉树。

[0111] 由于每个区域内的用户个数是动态变化的,所以本实施例中定时统计服务器所在区域内积分处于各个积分区间内的用户个数,将新统计的用户个数及其积分存入其所在区域已建立的积分二叉树匹配的节点中,以保证排名的实时性。具体定时的时间,可以依据需要来设置,例如可以每天更新一次,或者每小时更新一次等等。

[0112] 在当前用户的积分发生改变时,在已建立的积分二叉树中涉及修改的节点为所述当前用户的积分发生改变的积分区间所在的节点。

[0113] 例如:当前用户A的积分由3分变到5分,当前用户A所在区域的积分二叉树涉及保存积分区间为 $[3, 5]$ 的节点。

[0114] 当用户的积分发生变化时,每次所需要更新的节点数量是用户积分变化的 $\log(n)$ 级别,每次查询或更新的复杂度为积分最大值的 $O(\log(n))$ ,也就是说如果当前用户积分一次变化在百万级,那么更新的节点的数量在二十级别,与用户数量无关,因此,在海量用户下也可以快速完成查询并计算出当前用户的排名。

[0115] 在图1所示方法实施例基础上,如图4所示,本发明提供的另一种用户排名获得方法,应用于每个区域的服务器,该方法在图1所示的步骤S101-S104后还可以包括:

[0116] S401:将当前用户及第一积分发送给所有区域的各个服务器。

[0117] 为了获得当前用户在所有区域中的总排名,需要将当前用户及第一积分发送给除当前用户所在区域之外的所有区域的各个服务器。各个服务器根据接收到的当前用户及第一积分,遍历本区域所建立的积分二叉树,计算出接收到的当前用户在本区域的排名,并将计算出的排名发送给当前用户所在区域的服务器。当然,各个服务器也可以按照现有技术的方法根据接收到的当前用户及第一积分,获得当前用户在该区域的排名。

[0118] S402:接收各个服务器根据所述第一积分计算出的所述当前用户在各个区域的排名。

[0119] S403:将接收到的各个区域的排名累加。

[0120] S404:将累加后的排名减一作为当前用户在所有区域中的总排名。

[0121] 例如:当前用户A在区域1,积分为10分,在区域1的排名为C1,将当前用户及积分10发送给区域1以外的其它区域,接收各个区域发送的排名,累加排名并减1即为当前用户A在所有区域中的总排名,如图5所示。

[0122] 在接收其它区域发送的排名时,可以接受一定的区域间数据延迟,为了提高响应

速度,可以在每个区域存储一份缓存数据,异步定期更新,这样在获取其他区域排名时,只要读取缓存而不是遍历各个区域的积分二叉树,可以提高一定的计算速度。

[0123] 本发明实施例提供一种用户排名获得方法,获得当前用户的第一积分,所述当前用户的第一积分对应一个排名基数,遍历已建立的积分二叉树,所述积分二叉树为当前用户所在区域的积分二叉树,所述积分二叉树的每个节点中保存了一个积分区间及积分处于该积分区间内的用户个数,查找保存了所述第一积分所在积分区间的每个目标子节点,根据所述排名基数、每个目标子节点在积分二叉树中的位置,及每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名。本发明通过遍历累加积分二叉树中每个目标子节点中保存的用户个数获得各个用户的排名,由于累加次数较少且累加次数与用户量无关,达到减少数据计算量,减少系统资源占用的目的。

[0124] 图6为本发明实施例提供一种用户排名获得装置的结构示意图,与图1所示的方法相对应,图6所示装置可以包括:获得模块601、遍历模块602、查找模块603和计算模块604,

[0125] 所述获得模块601,用于获得当前用户的第一积分,所述当前用户的第一积分对应一个排名基数;

[0126] 服务器根据用户就近的原则存储数据,所述数据可以包括图片、短信或视频等。

[0127] 在当用户对存储数据进行备份/还原操作时,当前用户所在区域的服务器将备份/还原操作对应的精简数据写入缓冲区。

[0128] 当在缓冲区预设超时时间内,当前用户对存储数据没有进行备份/还原操作时,则将以上缓冲区的精简数据写入消息队列作为备份数据,等待服务器读取并进行下一步处理。

[0129] 具体的,所述获得模块601,可以包括:第一获得子模块和第一计算子模,

[0130] 所述第一获得子模块,用于获得当前用户的备份数据;

[0131] 所述第一计算子模块,用于根据所述获得的备份数据计算出当前用户的第一积分,所述当前用户的第一积分对应有一个排名基数。

[0132] 读取消息队列中的当前用户的备份数据,根据所述备份数据中备份/还原操作次数等信息计算出当前用户的第一积分。实际应用中,按照预设的积分获得原则,备份/还原操作针对的类型不同,则获得积分也不相同。

[0133] 所述遍历模块602,用于遍历已建立的积分二叉树,所述积分二叉树为当前用户所在区域的积分二叉树,所述积分二叉树的每个节点中保存了一个积分区间及积分处于该积分区间内的用户个数;

[0134] 其中,所述积分二叉树的根节点保存的积分区间为:当前用户所在区域中的积分中的最小值与最大值组成的积分区间;所述积分二叉树的根节点保存的积分处于该积分区间内的用户个数为当前用户所在区域中的用户的总个数;每个节点包含的积分区间的长度为积分区间中的最大值与最小值的差,且叶子节点所包含的积分区间的长度为1。

[0135] 所述查找模块603,用于查找保存了所述第一积分所在积分区间的每个目标子节点;

[0136] 所述计算模块604,用于根据所述排名基数、每个目标子节点在积分二叉树中的位置,及每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名。

[0137] 具体的,所述计算模块604中找到保存所述第一积分所在积分区间的每个目标子节点,判断找到的每个目标子节点在积分二叉树中的位置,根据判断结果及当前用户的第一积分对应的排名基数累加找到的每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名。

[0138] 所述计算模块604,可以包括:第一确定子模块、第一判断子模块、第二确定子模块、第二判断子模块、第三确定子模块和确定子模块,

[0139] 所述第一确定子模块,用于在所述积分二叉树的根节点的子节点中找到目标子节点,将找到的目标子节点确定为当前节点;

[0140] 所述第一判断子模块,用于判断当前节点是否为左子节点,如果是,则触发第一排名子模块,如果否,则触发第二排名子模块,其中,每个非叶子节点有两个子节点,分别为左子节点和右子节点,所述左子节点为低分子节点,所述右子节点为高分子节点;

[0141] 所述第二确定子模块,用于在当前节点的子节点中查找下一个目标子节点,将该找到的目标子节点确定为当前节点;

[0142] 所述第二判断子模块,用于判断当前节点是否为左子节点,如果是,则触发第三排名子模块,如果否,则触发第四排名子模块;

[0143] 所述第三确定子模块,用于在当前节点的子节点中查找再下一个目标子节点,将该找到的再下一个目标子节点确定为当前节点,触发第二判断子模块;

[0144] 所述确定子模块,用于在当前节点为叶子节点时,将当前用户的排名确定为用户在所在区域的排名;

[0145] 所述第一排名子模块,用于将右子节点包含的用户个数累加到所述排名基数上,将累加后得到的排名作为当前用户的排名;

[0146] 所述第二排名子模块,用于将所述排名基数作为当前用户的排名;

[0147] 所述第三排名子模块,用于将右子节点包含的用户个数累加到所述当前用户的排名上,将累加后得到的排名作为当前用户的排名;

[0148] 所述第四排名子模块,用于当前用户的排名不变。

[0149] 在本发明其他实施例中,图6所示装置还可以包括:更新模块,

[0150] 所述更新模块,用于定时统计服务器所在区域内积分处于各个积分区间内的用户个数,并根据统计结果更新服务器所在区域的积分二叉树。

[0151] 当用户的积分发生变化时,每次所需要更新的节点数量是用户积分变化的 $\log(n)$ 级别,每次查询或更新的复杂度为积分最大值的 $O(\log(n))$ ,也就是说如果当前用户积分一次变化在百万级,那么更新的节点的数量在二十级别,与用户数量无关,因此,在海量用户下也可以快速完成查询并计算出当前用户的排名。

[0152] 在本发明其他实施例中,图6所示装置还可以包括:发送模块、接收模块、累加模块和确定模块,

[0153] 所述发送模块,用于将当前用户及第一积分发送给所有区域的各个服务器;

[0154] 为了获得当前用户在所有区域中的总排名,需要将当前用户及第一积分发送给除当前用户所在区域之外的所有区域的各个服务器。各个服务器可以按照图6所示的装置,根据接收到的当前用户及第一积分,遍历本区域所建立的积分二叉树,计算出接收到的当前用户在本区域的排名,并将计算出的排名发送给当前用户所在区域的服务器。当然,各个服

务器也可以按照现有技术的方法根据接收到的当前用户及第一积分,获得当前用户在该区域的排名。

[0155] 所述接收模块,用于接收各个服务器根据所述第一积分计算出的所述当前用户在各个区域的排名;

[0156] 所述累加模块,用于将接收到的各个区域的排名累加;

[0157] 所述确定模块,用于将累加后的排名减一作为当前用户在所有区域中的总排名。

[0158] 本发明实施例提供的一种用户排名获得装置,获得当前用户的第一积分,所述当前用户的第一积分对应一个排名基数,遍历已建立的积分二叉树,所述积分二叉树为当前用户所在区域的积分二叉树,所述积分二叉树的每个节点中保存了一个积分区间及积分处于该积分区间内的用户个数,查找保存了所述第一积分所在积分区间的每个目标子节点,根据所述排名基数、每个目标子节点在积分二叉树中的位置,及每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名。本发明通过遍历累加积分二叉树中每个目标子节点中保存的用户个数获得各个用户的排名,由于累加次数较少且累加次数与用户量无关,达到减少数据计算量,减少系统资源占用的目的。

[0159] 相应地,如图7所述,本发明实施例还提供了一种服务器,所述服务器包括:

[0160] 壳体(图中未示出)、处理器701、存储器702、电路板703和电源电路704,其中,电路板703安置在壳体围成的空间内部,处理器701和存储器702设置在电路板703上;电源电路704,用于为电子设备的各个电路或器件供电;存储器702用于存储可执行程序代码;处理器701通过读取存储器702中存储的可执行程序代码来运行与可执行程序代码对应的程序,以用于执行以下步骤:

[0161] 获得当前用户的第一积分,所述当前用户的第一积分对应一个排名基数;

[0162] 遍历已建立的积分二叉树,所述积分二叉树为当前用户所在区域的积分二叉树,所述积分二叉树的每个节点中保存了一个积分区间及积分处于该积分区间内的用户个数;

[0163] 查找保存了所述第一积分所在积分区间的每个目标子节点;

[0164] 根据所述排名基数、每个目标子节点在积分二叉树中的位置,及每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名。

[0165] 相应地,本申请实施例还提供了一种存储介质,其中,该存储介质用于存储应用程序,所述应用程序用于在运行时执行本申请所述的一种用户排名获得方法。其中,本申请所述的一种用户排名获得方法,应用于每个区域的服务器,该方法可以包括:

[0166] 获得当前用户的第一积分,所述当前用户的第一积分对应一个排名基数;

[0167] 遍历已建立的积分二叉树,所述积分二叉树为当前用户所在区域的积分二叉树,所述积分二叉树的每个节点中保存了一个积分区间及积分处于该积分区间内的用户个数;

[0168] 查找保存了所述第一积分所在积分区间的每个目标子节点;

[0169] 根据所述排名基数、每个目标子节点在积分二叉树中的位置,及每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名。

[0170] 相应地,本申请实施例还提供了一种应用程序,其中,该应用程序用于在运行时执行本申请所述的一种用户排名获得方法。其中,本申请所述的一种用户排名获得方法,应用于每个区域的服务器,该方法可以包括:

[0171] 获得当前用户的第一积分,所述当前用户的第一积分对应一个排名基数;

[0172] 遍历已建立的积分二叉树,所述积分二叉树为当前用户所在区域的积分二叉树,所述积分二叉树的每个节点中保存了一个积分区间及积分处于该积分区间内的用户个数;

[0173] 查找保存了所述第一积分所在积分区间的每个目标子节点;

[0174] 根据所述排名基数、每个目标子节点在积分二叉树中的位置,及每个目标子节点中保存的用户的个数,计算出当前用户在所在区域的排名。

[0175] 需要说明的是,在本文中,诸如第一和第二等之类的关系术语仅仅用来将一个实体或者操作与另一个实体或操作区分开来,而不一定要求或者暗示这些实体或操作之间存在任何这种实际的关系或者顺序。而且,术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含,从而使得包括一系列要素的过程、方法、物品或者设备不仅包括那些要素,而且还包括没有明确列出的其他要素,或者是还包括为这种过程、方法、物品或者设备所固有的要素。在没有更多限制的情况下,由语句“包括一个……”限定的要素,并不排除在包括所述要素的过程、方法、物品或者设备中还存在另外的相同要素。

[0176] 本说明书中的各个实施例均采用相关的方式描述,各个实施例之间相同相似的部分互相参见即可,每个实施例重点说明的都是与其他实施例的不同之处。尤其,对于系统实施例而言,由于其基本相似于方法实施例,所以描述的比较简单,相关之处参见方法实施例的部分说明即可。

[0177] 以上所述仅为本发明的较佳实施例而已,并非用于限定本发明的保护范围。凡在本发明的精神和原则之内所作的任何修改、等同替换、改进等,均包含在本发明的保护范围内。

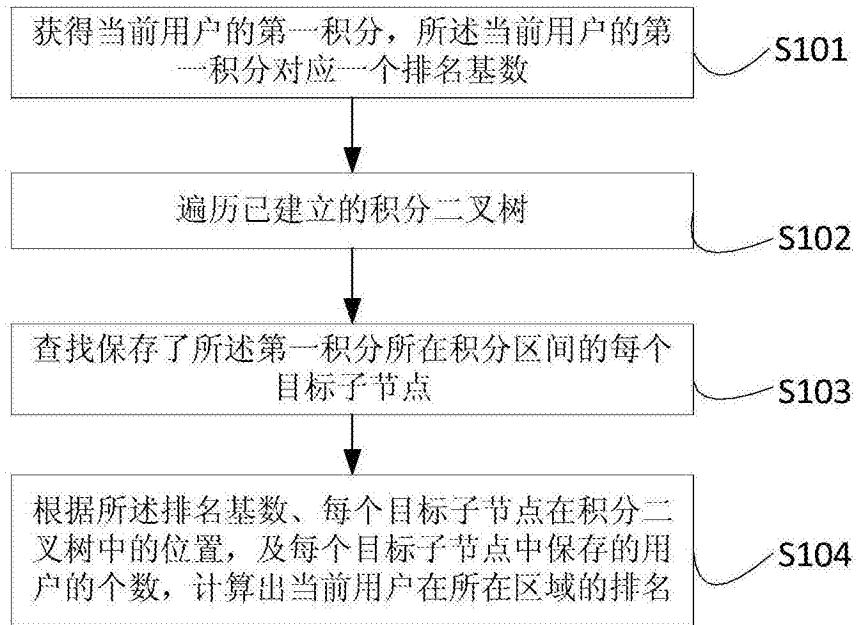


图1

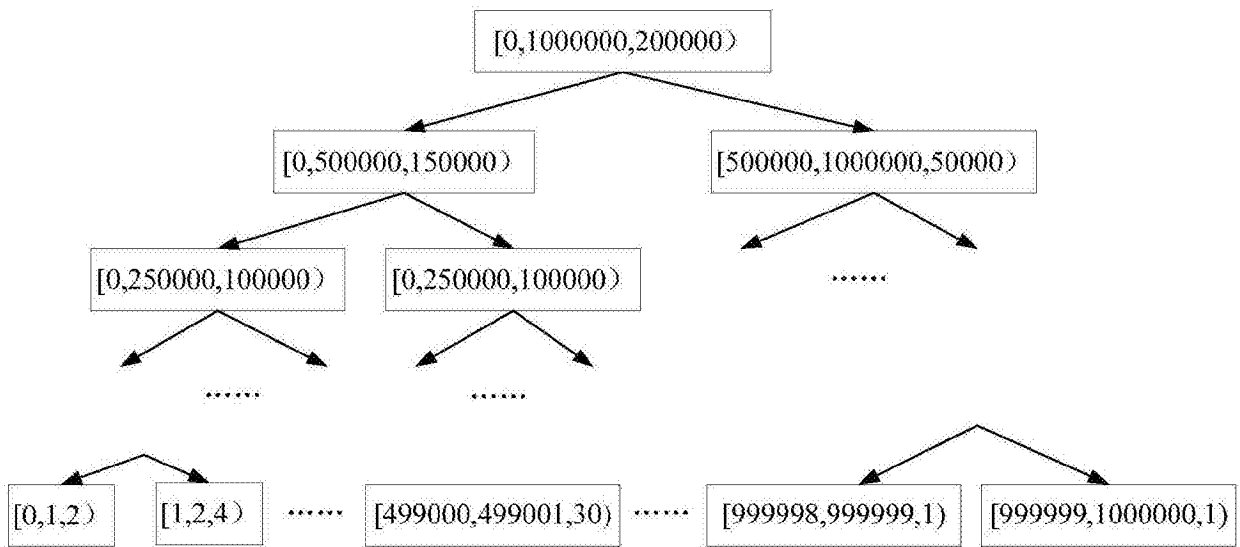


图2



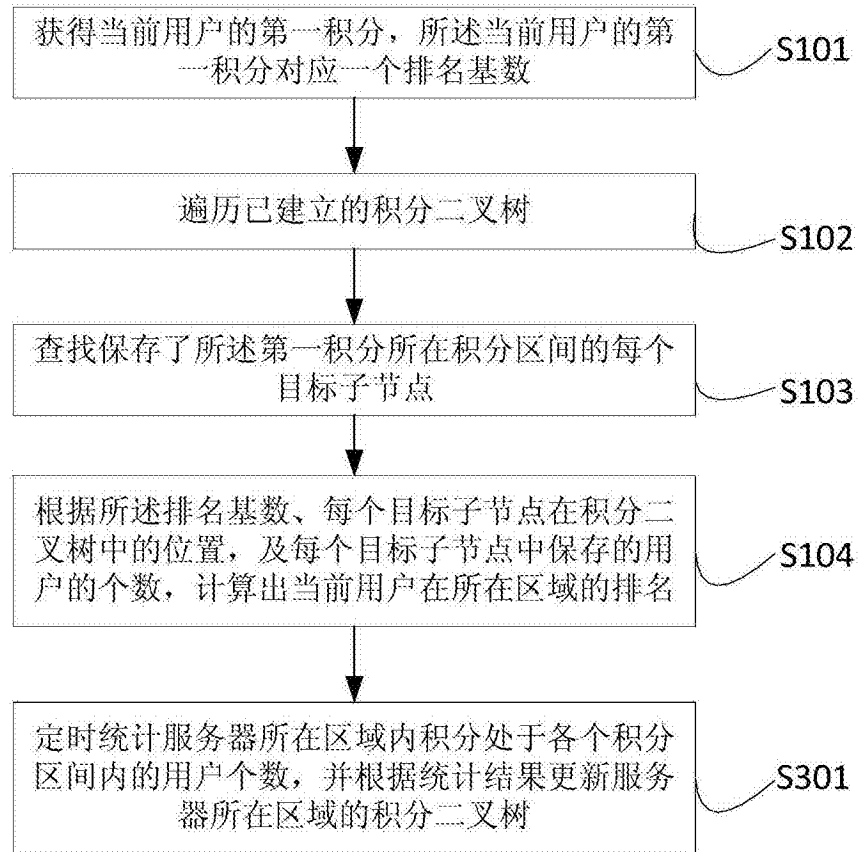


图3

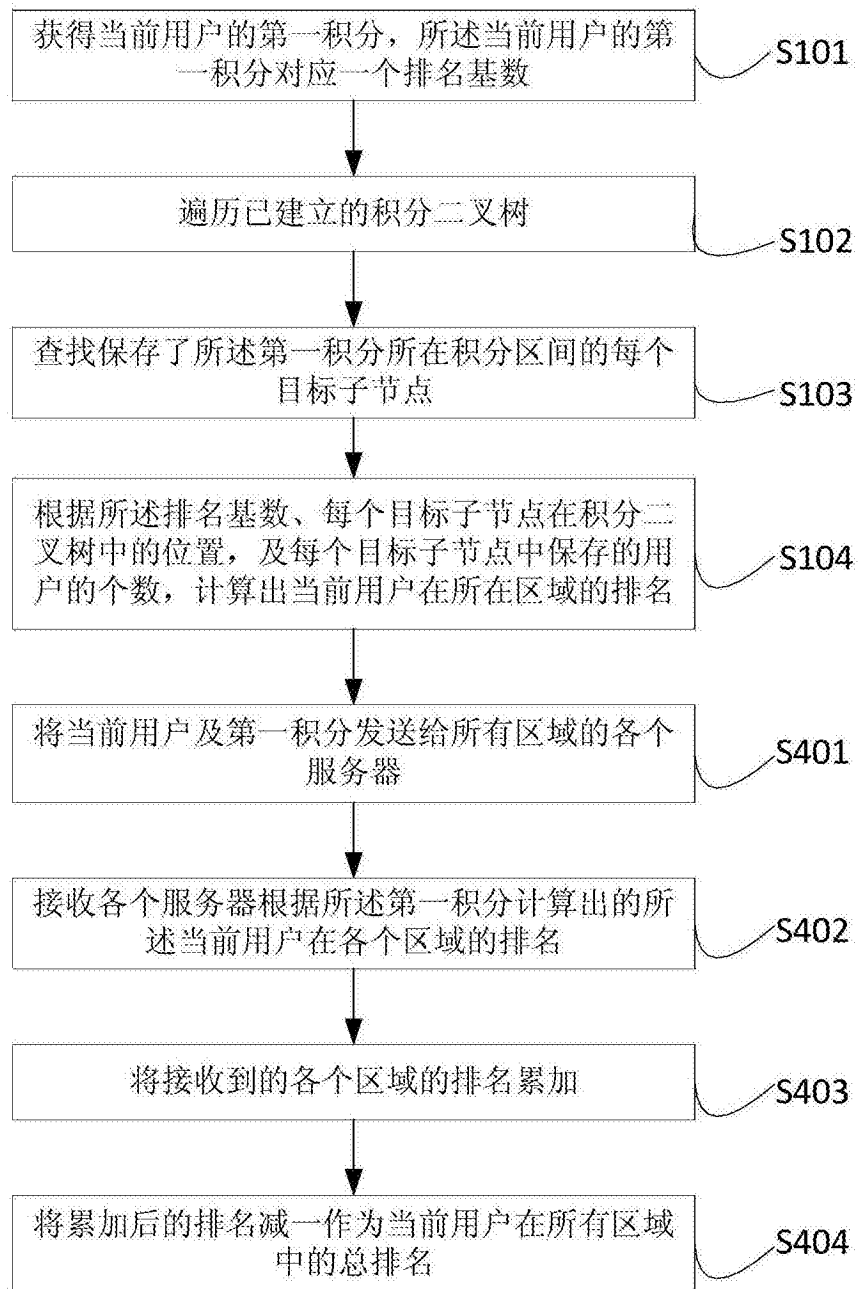


图4

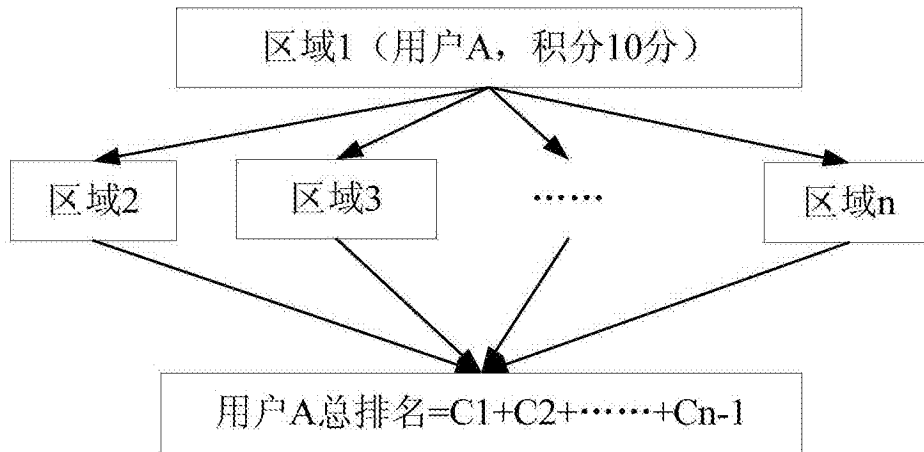


图5

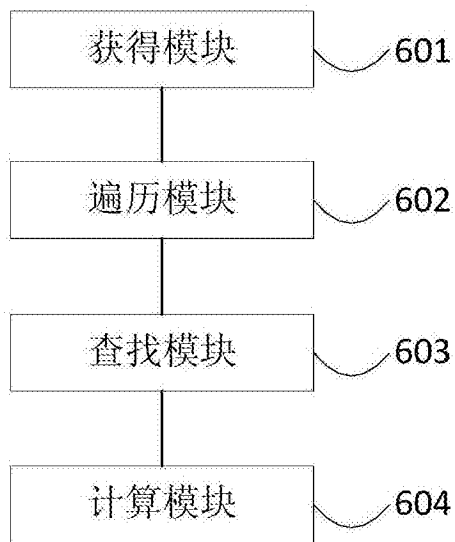


图6

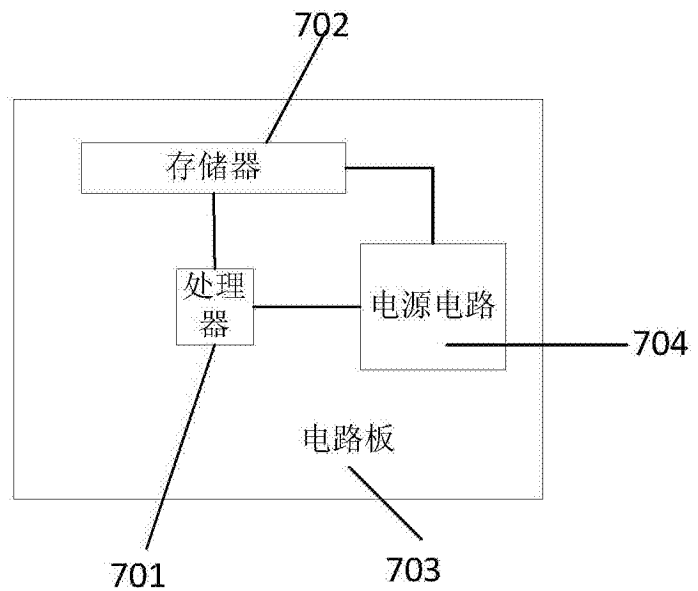


图7