



(12) 发明专利申请

(10) 申请公布号 CN 105740264 A

(43) 申请公布日 2016. 07. 06

(21) 申请号 201410754323. X

(22) 申请日 2014. 12. 10

(71) 申请人 北大方正集团有限公司
地址 100871 北京市海淀区成府路 298 号中
关村方正大厦 808 室
申请人 北大方正信息产业集团有限公司

(72) 发明人 刘雨洋 郑程光 孙伟丰 罗正海
李浩 刘慧娟

(74) 专利代理机构 北京银龙知识产权代理有限
公司 11243
代理人 许静 黄灿

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

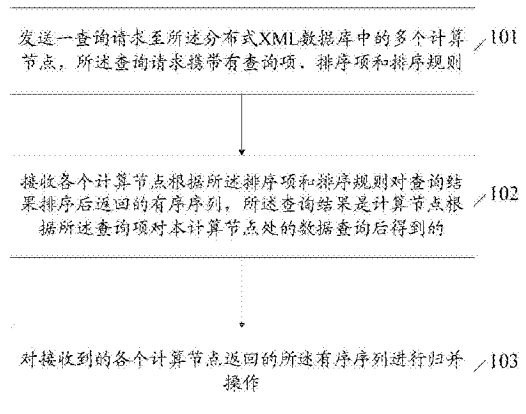
权利要求书2页 说明书5页 附图2页

(54) 发明名称

一种分布式 XML 数据库的排序方法及装置

(57) 摘要

本发明提供了一种分布式 XML 数据库的排序方法及装置,该方法包括:发送一查询请求至所述分布式 XML 数据库中的多个计算节点,该查询请求携带有查询项、排序项和排序规则;接收各个计算节点根据所述排序项和排序规则对查询结果排序后返回的有序序列,所述查询结果是计算节点根据所述查询项对本计算节点处的数据查询后得到的;对接收到的各个计算节点返回的有序序列进行归并操作。在本发明的方法中,每个计算节点根据查询项对本节点处的数据进行查询,并对查询结果根据排序项和排序规则进行排序,控制节点只需要归并各个计算节点已经排序好的有序序列,即可完成大量数据的排序,从而大大减少了 IO 时间,提高了排序效率,增强了用户体验。



1. 一种分布式 XML 数据库的排序方法,应用于 XML 数据库中的控制节点,其特征在于,所述方法包括:

发送一查询请求至所述分布式 XML 数据库中的多个计算节点,所述查询请求携带有查询项、排序项和排序规则;

接收各个计算节点根据所述排序项和排序规则对查询结果排序后返回的有序序列,所述查询结果是计算节点根据所述查询项对本计算节点处的数据查询后得到的;

对接收到的各个计算节点返回的所述有序序列进行归并操作。

2. 根据权利要求 1 所述的方法,其特征在于,所述对所述有序序列进行归并操作包括:若进行归并操作的有序序列的数据量未超过内存上限,则将有序序列直接保存在内存中。

3. 根据权利要求 2 所述的方法,其特征在于,所述对所述有序序列进行归并操作还包括:

若进行归并操作的有序序列的数据量超过内存上限,则将超过内存上限的有序序列保存在磁盘中,等待内存读取。

4. 一种分布式 XML 数据库的排序方法,应用于 XML 数据库中的计算节点,其特征在于,所述方法包括:

接收 XML 数据库中的控制节点发送的一查询请求,所述查询请求携带有查询项、排序项和排序规则;

根据所述查询项,对本计算节点处的数据进行查询,得到一查询结果,并根据所述排序项和排序规则,对查询结果进行排序,以得到一有序序列;

发送所述有序序列至所述控制节点,以使控制节点对所述有序序列进行归并操作。

5. 根据权利要求 4 所述的方法,其特征在于,在所述得到一有序序列的步骤之后,所述方法还包括:

将所述有序序列划分为多个子序列;

所述发送所述有序序列至所述控制节点包括:

分别将所述多个子序列依次发送至所述控制节点。

6. 一种分布式 XML 数据库的排序装置,应用于 XML 数据库中的控制节点,其特征在于,所述装置包括:

发送模块,用于发送一查询请求至所述分布式 XML 数据库中的多个计算节点,所述查询请求携带有查询项、排序项和排序规则;

接收模块,用于接收各个计算节点根据所述排序项和排序规则对查询结果排序后返回的有序序列,所述查询结果是计算节点根据所述查询项对本计算节点处的数据查询后得到的;

归并模块,用于对接收到的各个计算节点返回的所述有序序列进行归并操作。

7. 根据权利要求 6 所述的装置,其特征在于,所述归并模块包括:

第一归并单元,用于若进行归并操作的有序序列的数据量未超过内存上限,则将有序序列直接保存在内存中。

8. 根据权利要求 7 所述的装置,其特征在于,所述归并模块还包括:

第二归并单元,用于若进行归并操作的有序序列的数据量超过内存上限,则将超过内

存上限的有序序列保存在磁盘中,等待内存读取。

9. 一种分布式 XML 数据库的排序装置,应用于 XML 数据库中的计算节点,其特征在于,所述装置包括:

接收模块,用于接收 XML 数据库中的控制节点发送的一查询请求,所述查询请求携带有查询项、排序项和排序规则;

查询模块,用于根据所述查询项,对本计算节点处的数据进行查询,得到一查询结果,并根据所述排序项和排序规则,对查询结果进行排序,以得到一有序序列;

发送模块,用于发送所述有序序列至所述控制节点,以使控制节点对所述有序序列进行归并操作。

10. 根据权利要求 9 所述的装置,其特征在于,所述装置还包括:

划分模块,用于将所述有序序列划分为多个子序列;

所述发送模块包括:

发送单元,用于分别将所述多个子序列依次发送至所述控制节点。

一种分布式 XML 数据库的排序方法及装置

技术领域

[0001] 本发明涉及一种排序方法,尤其涉及一种分布式 XML 数据库的排序方法及装置。

背景技术

[0002] 在分布式 XML 数据库中包含两类节点:控制节点和计算节点,计算节点用于与控制节点交互,保存分片数据,完成局部查询和局部查询优化;控制节点用于对外提供查询接口,全局优化,结果合并。对于客户端来说,客户端不关心分布式 XML 数据库的数据分布情况和数据合并过程,只要拿到正确的查询结果即可,但是由于查询是在多个计算节点上同时展开的,控制节点需要合并所有的查询结果,对于非排序的用于 XML 数据查询的语言(XQuery)来说,控制节点只要把各个计算节点得到的数据简单合并起来即可;对于需要排序的 XQuery 来说,由于数据分布在不同的计算节点上,简单合并各个计算节点并不能得到全局的排序结果,需要在控制节点对数据进行排序操作。想要得到排序后的结果,控制节点需要对多个计算节点合并得到的结果进行排序。

[0003] 因为结果集可能会超过内容的容量,所以控制节点需要采用外部排序。外部排序指的是大文件的排序,即待排序的记录存储在外存储器上,待排序的文件无法一次装入内存,需要在内存和外部存储器之间进行多次数据交换,以达到排序整个文件的目的。

[0004] 外部排序最常用的算法是多路归并排序,即将原文件分解成多个能够一次性装入内存的部分,分别把每一部分调入内存完成排序。然后,对已经排序的子文件进行归并排序。根据有限的内存资源将大文件分为 L 个段,然后依次将这 L 个段读入内存并利用高效的内部排序算法对每个段进行排序,排序后的结果即为初始有序归并段直接写入外存文件。内部排序时要选择合适的排序算法,并且要考虑到内部排序需要的辅助空间以及有限的内存空间来决定究竟要把大文件分为几个段。接下来选择合适的路数 k 对这 L 个归并段进行多路归并排序,每一趟归并使 k 个归并段变为 1 个较大归并段写入文件,反复几趟归并后得到整个有序的文件。上述算法中,根据内存限制,L 和 K 的不同选择可能会涉及到多次 IO 操作。

发明内容

[0005] 为了克服现有技术中存在的上述问题,本发明实施例提供了一种分布式 XML 数据库的排序方法及装置,减少了排序过程中的 IO 时间,增强了用户体验。

[0006] 为了解决上述技术问题,本发明实施例采用如下技术方案:

[0007] 依据本发明实施例的一个方面,提供了一种分布式 XML 数据库的排序方法,应用于 XML 数据库中的控制节点,所述方法包括:发送一查询请求至所述分布式 XML 数据库中的多个计算节点,所述查询请求携带有查询项、排序项和排序规则;接收各个计算节点根据所述排序项和排序规则对查询结果排序后返回的有序序列,所述查询结果是计算节点根据所述查询项对本计算节点处的数据查询后得到的;对接收到的各个计算节点返回的所述有序序列进行归并操作。

- [0008] 其中,所述对所述有序序列进行归并操作包括:
- [0009] 若进行归并操作的有序序列的数据量未超过内存上限,则将有序序列直接保存在内存中。
- [0010] 其中,所述对所述有序序列进行归并操作还包括:
- [0011] 若进行归并操作的有序序列的数据量超过内存上限,则将超过内存上限的有序序列保存在磁盘中,等待内存读取。
- [0012] 依据本发明实施例的又一个方面,还提供了一种分布式 XML 数据库的排序方法,应用于 XML 数据库中的计算节点,所述方法包括:接收 XML 数据库中的控制节点发送的一查询请求,所述查询请求携带有查询项、排序项和排序规则;根据所述查询项,对本计算节点处的数据进行查询,得到一查询结果,并根据所述排序项和排序规则,对查询结果进行排序,以得到一有序序列;发送所述有序序列至所述控制节点,以使控制节点对所述有序序列进行归并操作。
- [0013] 其中,在所述得到一有序序列的步骤之后,所述方法还包括:
- [0014] 将所述有序序列划分为多个子序列;
- [0015] 所述发送所述有序序列至所述控制节点包括:
- [0016] 分别将所述多个子序列依次发送至所述控制节点。
- [0017] 依据本发明实施例的另一个方面,还提供了一种分布式 XML 数据库的排序装置,应用于 XML 数据库中的控制节点,所述装置包括:
- [0018] 发送模块,用于发送一查询请求至所述分布式 XML 数据库中的多个计算节点,所述查询请求携带有查询项、排序项和排序规则;
- [0019] 接收模块,用于接收各个计算节点根据所述排序项和排序规则对查询结果排序后返回的有序序列,所述查询结果是计算节点根据所述查询项对本计算节点处的数据查询后得到的;
- [0020] 归并模块,用于对接收到的各个计算节点返回的所述有序序列进行归并操作。
- [0021] 其中,所述归并模块包括:
- [0022] 第一归并单元,用于若进行归并操作的有序序列的数据量未超过内存上限,则将有序序列直接保存在内存中。
- [0023] 其中,所述归并模块还包括:
- [0024] 第二归并单元,用于若进行归并操作的有序序列的数据量超过内存上限,则将超过内存上限的有序序列保存在磁盘中,等待内存读取。
- [0025] 依据本发明实施例的另一个方面,还提供了一种分布式 XML 数据库的排序装置,应用于 XML 数据库中的计算节点,所述装置包括:
- [0026] 接收模块,用于接收 XML 数据库中的控制节点发送的一查询请求,所述查询请求携带有查询项、排序项和排序规则;
- [0027] 查询模块,用于根据所述查询项,对本计算节点处的数据进行查询,得到一查询结果,并根据所述排序项和排序规则,对查询结果进行排序,以得到一有序序列;
- [0028] 发送模块,用于发送所述有序序列至所述控制节点,以使控制节点对所述有序序列进行归并操作。
- [0029] 其中,所述装置还包括:

[0030] 划分模块,用于将所述有序序列划分为多个子序列;

[0031] 所述发送模块包括:

[0032] 发送单元,用于分别将所述多个子序列依次发送至所述控制节点。

[0033] 本发明实施例的有益效果:在本发明实施例的方法中,每个计算节点根据查询项对本节点处的数据进行查询,并对查询结果根据排序项和排序规则进行排序,控制节点只需要归并各个计算节点已经排序好的有序序列,即可完成大量数据的排序,从而大大减少了 IO 时间,提高了排序效率,增强了用户体验。

附图说明

[0034] 图 1 为本发明实施例一的分布式 XML 数据库的排序方法的流程图;

[0035] 图 2 为本发明实施例二的分布式 XML 数据库的排序方法的流程图;

[0036] 图 3 为本发明实施例一的分布式 XML 数据库的排序方法对应的装置的结构示意图;

[0037] 图 4 为本发明实施例二的分布式 XML 数据库的排序方法对应的装置的结构示意图。

具体实施方式

[0038] 为使本发明要解决的技术问题、技术方案和优点更加清楚,下面将结合附图及具体实施例进行详细描述。

[0039] 实施例一

[0040] 如图 1 所示,本发明实施例提供了一种分布式 XML 数据库的排序方法,该方法应用于 XML 数据库中的控制节点,该方法包括:

[0041] 步骤 101,发送一查询请求至所述分布式 XML 数据库中的多个计算节点,所述查询请求携带有查询项、排序项和排序规则。

[0042] 具体地,所述排序项为所述查询项的属性或类别。

[0043] 步骤 102,接收各个计算节点根据所述排序项和排序规则对查询结果排序后返回的有序序列,所述查询结果是计算节点根据所述查询项对本计算节点处的数据查询后得到的。

[0044] 步骤 103,对接收到的各个计算节点返回的所述有序序列进行归并操作。

[0045] 在本发明实施例的方法中,每个计算节点根据查询项对本节点处的数据进行查询,并对查询结果根据排序项和排序规则进行排序,控制节点只需要归并各个计算节点已经排序好的有序序列,即可完成大量数据的排序,从而大大减少了 IO 时间,提高了排序效率,增强了用户体验。

[0046] 进一步地,在本发明上述实施例的步骤 101 之前,本实施例的方法还包括:接收用户发送的查询请求,其中所述查询请求携带有查询项、排序项和排序规则。

[0047] 其中,本发明上述实施例的步骤 103 中的对有序序列进行归并操作具体包括两种情况:

[0048] 情况 1:若进行归并操作的有序序列的数据量未超过内存上限,则将有序序列直接保存在内存中。

[0049] 情况 2:若进行归并操作的有序序列的数据量超过内存上限,则将超过内存上限的有序序列保存在磁盘中,等待内存读取。

[0050] 在大部分情况下,即使是对海量 XML 数据库做出的查询,其查询结果也不会很大,是符合情况 1 的,也就是说在大多数的归并操作的过程中,不需要 IO 操作;即使是需要用磁盘缓存部分数据的情况下,也只需要一次写 IO(接收到的数据超过内存限制),一次读 IO(将超过内存上限的数据进行归并操作的时候),而不会像传统算法一样有多次的磁盘 IO 操作。另外,由于数据不需要持久化存储,即不需要调用 fsync 函数进行同步,还可以进一步节省 IO 开销。同时,与传统的归并排序算法需要完成所有的局部排序才能开始返回结果不同,本实施例的方法在即使只有部分排序数据的情况下,就开始向客户端输出结果了。

[0051] 实施例二

[0052] 如图 2 所示,本发明实施例提供了一种分布式 XML 数据库的排序方法,该方法应用于 XML 数据库中的计算节点,该方法包括:

[0053] 步骤 201,接收 XML 数据库中的控制节点发送的一查询请求,所述查询请求携带有查询项、排序项和排序规则。

[0054] 具体地,所述排序项为所述查询项的属性或类别。

[0055] 步骤 202,根据所述查询项,对本计算节点处的数据进行查询,得到一查询结果,并根据所述排序项和排序规则,对查询结果进行排序,以得到一有序序列。

[0056] 步骤 203,发送所述有序序列至所述控制节点,以使控制节点对所述有序序列进行归并操作。

[0057] 在本发明实施例的方法中,每个计算节点根据查询项对本节点处的数据进行查询,并对查询结果根据排序项和排序规则进行排序,控制节点只需要归并各个计算节点已经排序好的有序序列,即可完成大量数据的排序,从而大大减少了 IO 时间,提高了排序效率,增强了用户体验。

[0058] 在本发明上述实施例的步骤 202 之后,所述方法还包括:

[0059] 将所述有序序列划分为多个子序列;

[0060] 所述发送所述有序序列至所述控制节点包括:

[0061] 分别将所述多个子序列依次发送至所述控制节点。

[0062] 下面进一步解释该过程,在得到一有序序列之后,需要将该有序序列分为多个子序列,其中将该多个子序列按照一定的先后顺序依次发送至控制节点,控制节点对一定时间内接收的各个计算节点发送的第一批多个子序列进行归并操作,与此同时输出局部归并操作的排序结果,同时下一批子序列发送至控制节点,控制节点依照此时内存的情况,对接收到的所有子序列进行归并操作。

[0063] 实施例三

[0064] 对应于上述实施例一中的排序方法,如图 3 所示,本发明实施例还提供了一种排序装置,该排序装置应用于 XML 数据库中的控制节点,所述装置 30 包括:

[0065] 发送模块 31,用于发送一查询请求至所述分布式 XML 数据库中的多个计算节点,所述查询请求携带有查询项、排序项和排序规则。

[0066] 接收模块 32,用于接收各个计算节点根据所述排序项和排序规则对查询结果排序后返回的有序序列,所述查询结果是计算节点根据所述查询项对本计算节点处的数据查询

后得到的。

[0067] 归并模块 33,用于对接收到的各个计算节点返回的所述有序序列进行归并操作。

[0068] 在本发明实施例中,每个计算节点根据查询项对本节点处的数据进行查询,并对查询结果根据排序项和排序规则进行排序,控制节点只需要归并各个计算节点已经排序好的有序序列,即可完成大量数据的排序,从而大大减少了 IO 时间,提高了排序效率,增强了用户体验。

[0069] 其中,所述归并模块 33 包括:第一归并单元和第二归并单元,其中

[0070] 所述第一归并单元,用于若进行归并操作的有序序列的数据量未超过内存上限,则将有序序列直接保存在内存中。

[0071] 所述第二归并单元,用于若进行归并操作的有序序列的数据量超过内存上限,则将超过内存上限的有序序列保存在磁盘中,等待内存读取。

[0072] 实施例四

[0073] 对应于上述实施例二中的排序方法,如图 4 所示,本发明实施例还提供了一种排序装置,所述装置应用于 XML 数据库中的计算节点,所述装置 40 包括:

[0074] 接收模块 41,用于接收 XML 数据库中的控制节点发送的一查询请求,所述查询请求携带有查询项、排序项和排序规则;

[0075] 查询模块 42,用于根据所述查询项,对本计算节点处的数据进行查询,得到一查询结果,并根据所述排序项和排序规则,对查询结果进行排序,以得到一有序序列;

[0076] 发送模块 43,用于发送所述有序序列至所述控制节点,以使控制节点对所述有序序列进行归并操作。

[0077] 在本发明实施例中,每个计算节点根据查询项对本节点处的数据进行查询,并对查询结果根据排序项和排序规则进行排序,控制节点只需要归并各个计算节点已经排序好的有序序列,即可完成大量数据的排序,从而大大减少了 IO 时间,提高了排序效率,增强了用户体验。

[0078] 进一步地,所述装置 40 还包括:

[0079] 划分模块,用于将所述有序序列划分为多个子序列;

[0080] 所述发送模块 43 包括:

[0081] 发送单元,用于分别将所述多个子序列依次发送至所述控制节点。

[0082] 以上所述是本发明的优选实施方式,应当指出,对于本技术领域的普通技术人员来说,在不脱离本发明所述原理的前提下,还可以做出若干改进和润饰,这些改进和润饰也应视为本发明的保护范围。

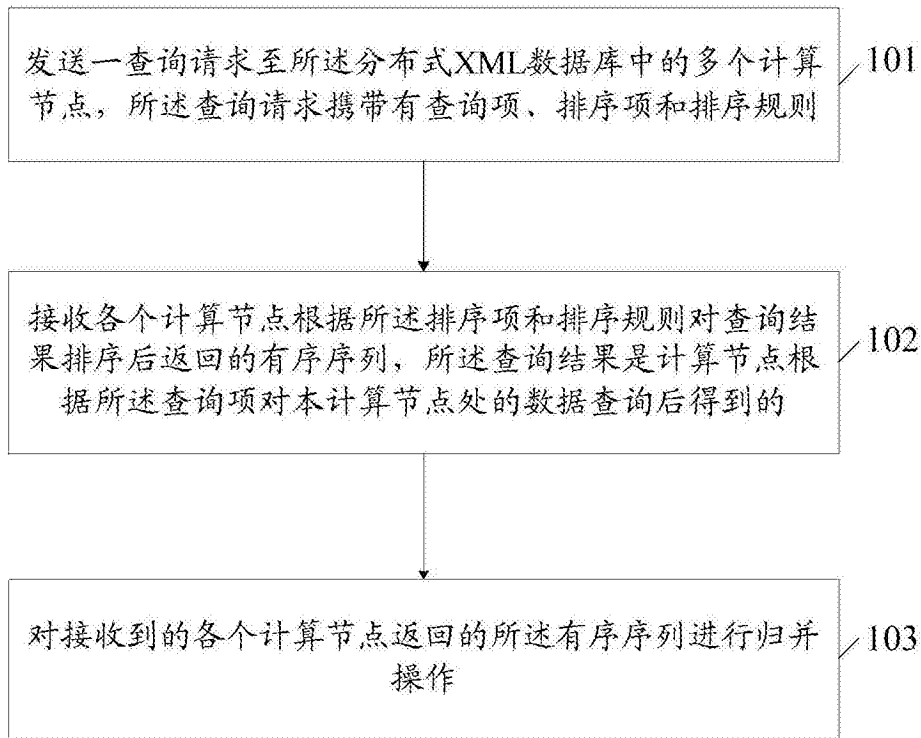


图 1

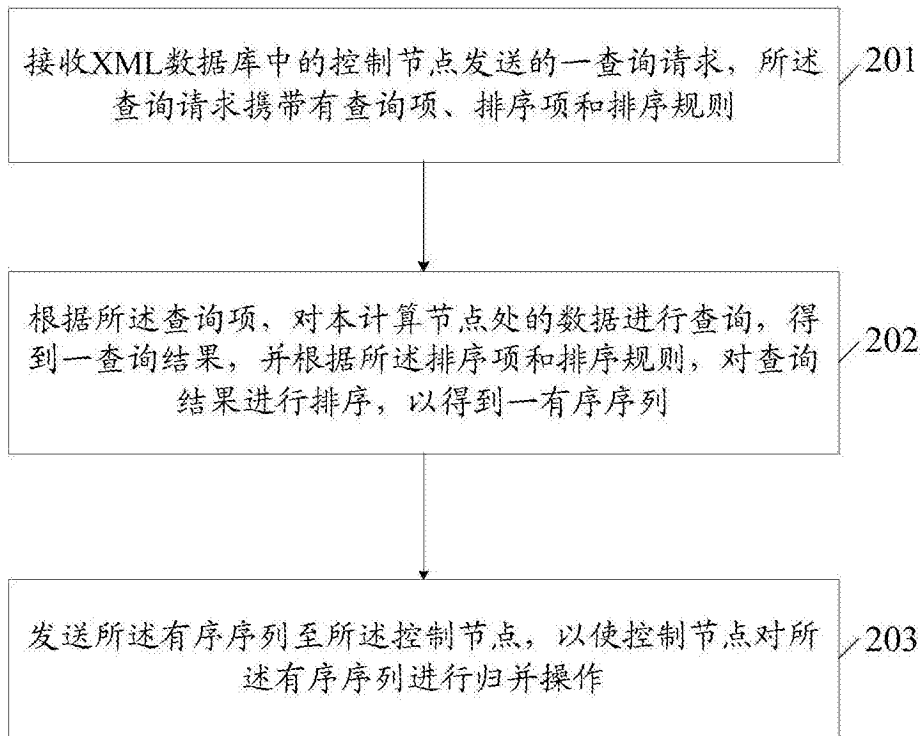


图 2

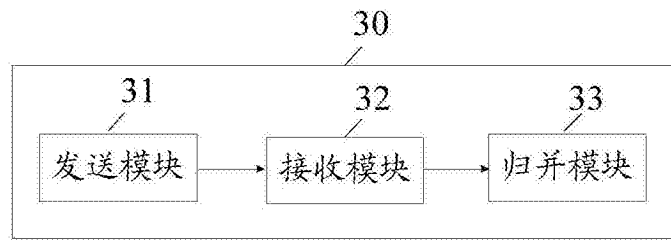


图 3

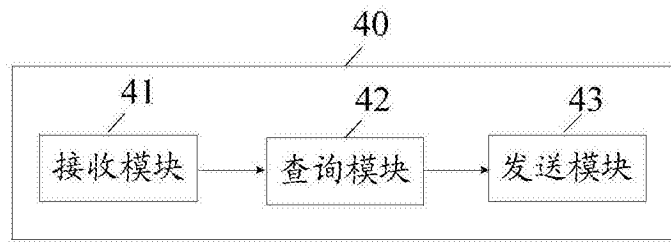


图 4