



(12) 发明专利

(10) 授权公告号 CN 113836366 B

(45) 授权公告日 2024. 10. 25

(21) 申请号 202110949930.1

(22) 申请日 2021.08.18

(65) 同一申请的已公布的文献号

申请公布号 CN 113836366 A

(43) 申请公布日 2021.12.24

(73) 专利权人 广州致远电子股份有限公司

地址 510000 广东省广州市天河区软件园

思成路43号第3层306房

(72) 发明人 周立功 翁叶冬 魏科

(74) 专利代理机构 北京泽方誉航专利代理事务

所(普通合伙) 11884

专利代理师 白婉露

(51) Int. Cl.

G06F 16/903 (2019.01)

G06F 16/901 (2019.01)

(56) 对比文件

马申斌等. 基于多核框架实时信号处理的内存管理. 工业控制计算机. 2017, 第30卷(第3期), 50-51, 54页.

审查员 孙博

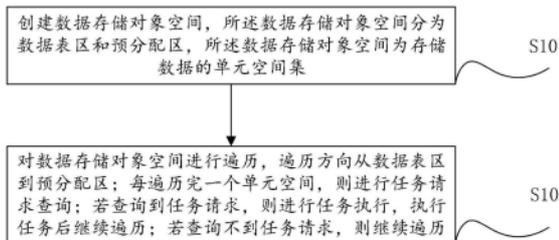
权利要求书2页 说明书8页 附图3页

(54) 发明名称

一种基于嵌入式系统的数据遍历方法及装置

(57) 摘要

本申请实施例公开了一种基于嵌入式系统的数据遍历方法及装置。本申请实施例提供的技术方案,通过在创建数据存储对象空间,所述数据存储对象空间分为数据表区和预分配区,所述数据存储对象空间为存储数据的单元空间集;对数据存储对象空间进行遍历,遍历方向从数据表区到预分配区;每遍历完一个单元空间,则进行任务请求查询;若查询到任务请求,则进行任务执行,执行任务后继续遍历;若查询不到任务请求,则继续遍历。本申请实施例能够在遍历过程中进行其他任务的查询与执行,避免了其他任务的长时间等待,同时避免了繁琐的加解锁操作,从而提高了数据遍历的效率和有效性。



1. 一种基于嵌入式系统的数据遍历方法,其特征在于,包括:
 - 创建数据存储对象空间,所述数据存储对象空间分为数据表区和预分配区,所述数据存储对象空间为存储数据的单元空间集;
 - 对数据存储对象空间进行遍历,遍历方向从数据表区到预分配区;
 - 每遍历完一个单元空间,则进行任务请求查询,所述任务请求包括增加数据任务和删除数据任务;
 - 若查询到任务请求,则进行任务执行,以对所述数据表区或所述预分配区中的单元空间进行数据增加或数据删除,执行任务后继续遍历;
 - 若查询不到任务请求,则继续遍历。
2. 根据权利要求1所述的基于嵌入式系统的数据遍历方法,其特征在于,所述数据为哈希结构数据,所述基于嵌入式系统的数据遍历方法,具体为:
 - 创建哈希存储对象空间,所述哈希存储对象空间分为哈希表区和预分配区,所述哈希存储对象空间为存储数据的单元空间集;
 - 对哈希存储对象空间进行遍历,遍历顺序方向从哈希表区到预分配区;
 - 每遍历完一个单元空间,则进行任务请求查询;
 - 若查询到任务请求,则进行任务执行,执行任务后继续遍历;
 - 若查询不到任务请求,则继续遍历。
3. 根据权利要求2所述的基于嵌入式系统的数据遍历方法,其特征在于,所述任务请求包括增加数据任务和删除数据任务;
 - 所述若查询到任务请求,则进行任务执行,执行任务后继续遍历,具体为:
 - 若查询到增加数据任务,则判断增加数据至哈希表区是否存在哈希冲突,所述哈希冲突为待增加数据的单元空间中已经存在数据,将待增加数据增加至所述单元空间时发生的冲突;
 - 若无哈希冲突,则将待增加数据增加至哈希表区;
 - 若存在哈希冲突,则将待增加数据增加至预分配区,并将待分配区链入哈希表区;
 - 若查询到删除数据任务,则直接删除对应单元空间中的数据。
4. 根据权利要求2所述的基于嵌入式系统的数据遍历方法,其特征在于,创建迭代器对象,所述迭代器对象中存在缓存空间;
 - 将遍历的单元空间中的键值对存储到所述迭代器对象中的缓存空间中,以供用户端获取遍历数据;
 - 若迭代器对象缓存空间中的键值对数据全部被用户端获取完毕,则继续从上一次遍历的最后一个单元空间之后继续遍历,并将遍历得到的单元空间中的键值对数据存储到迭代器对象的缓存空间中。
5. 根据权利要求1所述的基于嵌入式系统的数据遍历方法,其特征在于,所述数据为哈希结构数据、列表结构数据和数组结构数据任一种。
6. 根据权利要求5所述的基于嵌入式系统的数据遍历方法,其特征在于,创建迭代器对象,所述迭代器对象中存在缓存空间;
 - 将遍历的单元空间中的键值对存储到所述迭代器对象中的缓存空间中,以供用户端获取遍历数据;

若迭代器对象缓存空间中的键值对数据全部被用户端获取完毕,则继续从上一次遍历的最后一个单元空间之后继续遍历,并将遍历得到的单元空间中的键值对数据存储到迭代器对象的缓存空间中。

7. 根据权利要求1所述的基于嵌入式系统的数据遍历方法,其特征在于,所述数据存储对象空间分为数据表区和预分配区,具体为:

将所述数据存储对象空间分为数据表区和预分配区,所述预分配区空间为所述数据表区空间的2倍。

8. 一种基于嵌入式系统的数据遍历装置,其特征在于,包括:

存储对象空间创建模块,用于创建数据存储对象空间,所述数据存储对象空间分为数据表区和预分配区,所述数据存储对象空间为存储数据的单元空间集;

遍历模块,用于对数据存储对象空间进行遍历,遍历方向从数据表区到预分配区;每遍历完一个单元空间,则进行任务请求查询,所述任务请求包括增加数据任务和删除数据任务,所述单元空间存储对应数据信息;若查询到任务请求,则进行任务执行,以对所述数据表区或所述预分配区中的单元空间进行数据增加或数据删除,执行任务后继续遍历;若查询不到任务请求,则继续遍历。

9. 一种电子设备,其特征在于,包括:

存储器以及一个或多个处理器;

所述存储器,用于存储一个或多个程序;

当所述一个或多个程序被所述一个或多个处理器执行,使得所述一个或多个处理器实现如权利要求1-7任一所述的方法。

10. 一种包含计算机可执行指令的存储介质,其特征在于,所述计算机可执行指令在由计算机处理器执行时用于执行如权利要求1-7任一所述的方法。

一种基于嵌入式系统的数据遍历方法及装置

技术领域

[0001] 本申请实施例涉及数据遍历技术领域,尤其涉及一种基于嵌入式系统的数据遍历方法及装置。

背景技术

[0002] 对于建立一个嵌入式实时系统的数据库而言,哈希结构有着高效的查找性能,能够极大提高数据的存储效率。

[0003] 在传统的哈希结构中,一旦发生数据冲突,较为普遍的方式是在冲突的单元中制作一个链表,使得一个哈希单元可以容纳多个数据。当执行数据遍历的时候,在具有多个数据的单元中就要进行链表的遍历。此时如果对链表上正在遍历的数据执行插入或者删除等操作,就有可能对遍历操作产生破坏。

[0004] 在嵌入式实时系统中,通常的解决方式在哈希遍历的时候加上互斥锁,使得用户在执行遍历的时候无法进行增加和删除等操作,如果哈希表中的数据太多,则会导致遍历的时间太长,使得插入、删除等操作长时间无法得到执行。

[0005] 在哈希存储结构中,有许多操作需要依赖遍历,比如清空哈希表,重哈希等,而嵌入式实时系统的数据库在遍历过程中,无法进行其他操作会导致数据存储以及任务调度效率的下降。

发明内容

[0006] 本申请实施例提供一种基于嵌入式系统的数据遍历方法及装置,能够解决数据遍历过程增加互斥锁导致其他操作任务等待时间长的问题,提高了数据遍历的效率和有效性。

[0007] 在第一方面,本申请实施例提供了一种基于嵌入式系统的数据遍历方法,包括:

[0008] 创建数据存储对象空间,所述数据存储对象空间分为数据表区和预分配区,所述数据存储对象空间为存储数据的单元空间集;

[0009] 对数据存储对象空间进行遍历,遍历方向从数据表区到预分配区;

[0010] 每遍历完一个单元空间,则进行任务请求查询;

[0011] 若查询到任务请求,则进行任务执行,执行任务后继续遍历;

[0012] 若查询不到任务请求,则继续遍历。

[0013] 进一步的,所述数据为哈希结构数据,所述基于嵌入式系统的数据遍历方法,具体为:

[0014] 创建哈希存储对象空间,所述哈希存储对象空间分为哈希表区和预分配区,所述哈希存储对象空间为存储数据的单元空间集;

[0015] 对哈希存储对象空间进行遍历,遍历顺序方向从哈希表区到预分配区;

[0016] 每遍历完一个单元空间,则进行任务请求查询;

[0017] 若查询到任务请求,则进行任务执行,执行任务后继续遍历;

- [0018] 若查询不到任务请求,则继续遍历。
- [0019] 进一步的,所述任务请求包括增加数据任务和删除数据任务;
- [0020] 所述若查询到任务请求,则进行任务执行,执行任务后继续遍历,具体为:
- [0021] 若查询到增加数据任务,则判断增加数据至哈希表区是否存在哈希冲突,所述哈希冲突为待增加数据的单元空间中已经存在数据,将待增加数据增加至所述单元空间时发生的冲突;
- [0022] 若无哈希冲突,则将待增加数据增加至哈希表区;
- [0023] 若存在哈希冲突,则将待增加数据增加至预分配区,并将待分配区链入哈希表区;
- [0024] 若查询到删除数据任务,则直接删除对应单元空间中的数据。
- [0025] 进一步的,创建迭代器对象,所述迭代器对象中存在缓存空间;
- [0026] 将遍历的单元空间中的键值对存储到所述迭代器对象中的缓存空间中,以供用户端获取遍历数据;
- [0027] 若迭代器对象缓存空间中的键值对数据全部被用户端获取完毕,则继续从上一次遍历的最后一个单元空间之后继续遍历,并将遍历得到的单元空间中的键值对数据存储到迭代器对象的缓存空间中。
- [0028] 进一步的,所述数据为哈希结构数据、列表结构数据和数组结构数据任一种。
- [0029] 进一步的,创建迭代器对象,所述迭代器对象中存在缓存空间;
- [0030] 将遍历的单元空间中的键值对存储到所述迭代器对象中的缓存空间中,以供用户端获取遍历数据;
- [0031] 若迭代器对象缓存空间中的键值对数据全部被用户端获取完毕,则继续从上一次遍历的最后一个单元空间之后继续遍历,并将遍历得到的单元空间中的键值对数据存储到迭代器对象的缓存空间中。
- [0032] 进一步的,所述数据存储对象空间分为数据表区和预分配区,具体为:
- [0033] 将所述数据存储对象空间分为数据表区和预分配区,所述预分配区空间为所述数据表区空间的2倍。
- [0034] 在第二方面,本申请实施例提供了一种基于嵌入式系统的数据遍历装置,包括:存储对象空间创建模块,用于创建数据存储对象空间,所述数据存储对象空间分为数据表区和预分配区,所述数据存储对象空间为存储数据的单元空间集;
- [0035] 遍历模块,用于对数据存储对象空间进行遍历,遍历方向从数据表区到预分配区;每遍历完一个单元空间,则进行任务请求查询,所述单元空间存储对应数据信息;若查询到任务请求,则进行任务执行,执行任务后继续遍历;若查询不到任务请求,则继续遍历。
- [0036] 进一步的,所述存储对象空间创建模块,还用于创建哈希存储对象空间,所述哈希存储对象空间分为哈希表区和预分配区,所述哈希存储对象空间为存储数据的单元空间集;
- [0037] 进一步的,所述遍历模块,还用于对哈希存储对象空间进行遍历,遍历顺序方向从哈希表区到预分配区;每遍历完一个单元空间,则进行任务请求查询;若查询到任务请求,则进行任务执行,执行任务后继续遍历;若查询不到任务请求,则继续遍历。
- [0038] 进一步的,所述基于嵌入式系统的数据遍历装置还包括任务请求查询模块,所述任务请求查询模块,用于所述任务请求包括增加数据任务和删除数据任务;所述若查询到

任务请求,则进行任务执行,执行任务后继续遍历,具体为:若查询到增加数据任务,则判断增加数据至哈希表区是否存在哈希冲突,所述哈希冲突为待增加数据的单元空间中已经存在数据,将待增加数据增加至所述单元空间时发生的冲突;若无哈希冲突,则将待增加数据增加至哈希表区;若存在哈希冲突,则将待增加数据增加至预分配区,并将待分配区链入哈希表区;若查询到删除数据任务,则直接删除对应单元空间中的数据。

[0039] 进一步的,所述基于嵌入式系统的数据遍历装置还包括迭代器模块,所述迭代器模块,用于创建迭代器对象,所述迭代器对象中存在缓存空间;将遍历的单元空间中的键值对存储到所述迭代器对象中的缓存空间中,以供用户端获取遍历数据;若迭代器对象缓存空间中的键值对数据全部被用户端获取完毕,则继续从上一次遍历的最后一个单元空间之后继续遍历,并将遍历得到的单元空间中的键值对数据存储到迭代器对象的缓存空间中。

[0040] 进一步的,所述数据为哈希结构数据、列表结构数据和数组结构数据任一种。

[0041] 进一步的,所述迭代器模块,用于创建迭代器对象,所述迭代器对象中存在缓存空间;将遍历的单元空间中的键值对存储到所述迭代器对象中的缓存空间中,以供用户端获取遍历数据;若迭代器对象缓存空间中的键值对数据全部被用户端获取完毕,则继续从上一次遍历的最后一个单元空间之后继续遍历,并将遍历得到的单元空间中的键值对数据存储到迭代器对象的缓存空间中。

[0042] 进一步的,所述存储对象空间创建模块,还用于将所述数据存储对象空间分为数据表区和预分配区,所述预分配区空间为所述数据表区空间的2倍。

[0043] 在第三方面,本申请实施例提供了一种电子设备,包括:

[0044] 存储器以及一个或多个处理器;

[0045] 所述存储器,用于存储一个或多个程序;

[0046] 当所述一个或多个程序被所述一个或多个处理器执行,使得所述一个或多个处理器实现如第一方面所述的基于嵌入式系统的数据遍历方法。

[0047] 在第四方面,本申请实施例提供了一种包含计算机可执行指令的存储介质,所述计算机可执行指令在由计算机处理器执行时用于执行如第一方面所述的基于嵌入式系统的数据遍历方法。

[0048] 本申请实施例通过在遍历的过程中,每遍历完一个单元空间则进行任务请求查询,若查询到任务请求则进行任务执行后再继续遍历,若查询不到任务请求则继续遍历。采用上述技术手段,能够在遍历过程中进行其他任务的查询与执行,避免了其他任务的长时间等待,同时避免了繁琐的加解锁操作,从而提高了数据遍历的效率和有效性。

附图说明

[0049] 图1是本申请实施例一提供的一种基于嵌入式系统的数据遍历方法的流程图;

[0050] 图2是本申请实施例一提供的存储空间创建示意图;

[0051] 图3是本申请实施例一提供的无哈希冲突的示意图;

[0052] 图4是本申请实施例一提供的有哈希冲突的示意图;

[0053] 图5是本申请实施例一提供的遍历方向示意图;

[0054] 图6是本申请实施例二提供的一种基于嵌入式系统的数据遍历装置的结构示意图;

[0055] 图7是本申请实施例三提供的一种电子设备的结构示意图。

具体实施方式

[0056] 为了使本申请的目的、技术方案和优点更加清楚,下面结合附图对本申请具体实施例作进一步的详细描述。可以理解的是,此处所描述的具体实施例仅仅用于解释本申请,而非对本申请的限定。另外还需要说明的是,为了便于描述,附图中仅示出了与本申请相关的部分而非全部内容。在更加详细地讨论示例性实施例之前应当提到的是,一些示例性实施例被描述成作为流程图描绘的处理或方法。虽然流程图将各项操作(或步骤)描述成顺序的处理,但是其中的许多操作可以被并行地、并发地或者同时实施。此外,各项操作的顺序可以被重新安排。当其操作完成时所述处理可以被终止,但是还可以具有未包括在附图中的附加步骤。所述处理可以对应于方法、函数、规程、子例程、子程序等等。

[0057] 本申请提供的基于嵌入式系统的数据遍历方法及装置,旨遍历的过程中,每遍历完一个单元空间则进行任务请求查询,若查询到任务请求则进行任务执行后再继续遍历,若查询不到任务请求则继续遍历。以此实现在一个遍历过程中进行任务的查询与执行,避免了繁琐的加解锁操作,从而提高了数据遍历的效率和有效性。相对于传统的数据遍历方式,其通常是在遍历的过程中添加互斥锁使得用户在执行遍历的时候无法进行增加和删除等操作。若遍历的数据较多时则会导致遍历时间较长,使得增加和删除等操作长时间无法得到执行,使得任务的性能下降,而如果对遍历的每个单元进行单独的加解锁操作则会导致大量额外的开销。基于此,提供本申请实施例的基于嵌入式系统的数据遍历方法,以解决现有数据遍历过程中的增加互斥所导致其他操作任务等待时间长的问题。

[0058] 实施例一:

[0059] 图1给出了本申请实施例一提供的一种基于嵌入式系统的数据遍历方法的流程图,本实施例中提供的基于嵌入式系统的数据遍历方法可以由基于嵌入式系统的数据遍历设备执行,该基于嵌入式系统的数据遍历设备可以通过软件和/或硬件的方式实现,该基于嵌入式系统的数据遍历设备可以是两个或多个物理实体构成,也可以是一个物理实体构成。一般而言,该基于嵌入式系统的数据遍历设备可以是智能终端设备等。

[0060] 参照图1,该基于嵌入式系统的数据遍历方法具体包括:

[0061] S101、创建数据存储对象空间,所述数据存储对象空间分为数据表区和预分配区,所述数据存储对象空间为存储数据的单元空间集。

[0062] 具体的,所述数据为哈希结构数据、列表结构数据和数组结构数据任一种。

[0063] 具体的,将所述数据存储对象空间分为数据表区和预分配区,所述预分配区空间为所述数据表区空间的2倍。

[0064] 进一步的,参照图2,所述数据为哈希结构数据,创建哈希存储对象空间,所述哈希存储对象空间分为哈希表区1和预分配区2,所述哈希存储对象空间为存储数据的单元空间集。创建哈希数据存储对象时,先预设3倍哈希数据表容量空间,其中1倍哈希数据表容量空间是哈希表区1空间,2倍哈希数据表容量空间是预分配区2空间。

[0065] S102、对数据存储对象空间进行遍历,遍历方向从数据表区到预分配区;每遍历完一个单元空间,则进行任务请求查询;若查询到任务请求,则进行任务执行,执行任务后继续遍历;若查询不到任务请求,则继续遍历。

[0066] 具体的,所述数据为哈希结构数据,对哈希存储对象空间进行遍历,遍历顺序方向从哈希表区1到预分配区2;每遍历完一个单元空间,则进行任务请求查询;若查询到任务请求,则进行任务执行,执行任务后继续遍历;若查询不到任务请求,则继续遍历。

[0067] 具体的,所述任务请求包括增加数据任务和删除数据任务;若查询到增加数据任务,则判断增加数据至哈希表区是否存在哈希冲突,所述哈希冲突为待增加数据的单元空间中已经存在数据,将待增加数据增加至所述单元空间时发生的冲突;若无哈希冲突,则将待增加数据增加至哈希表区;若存在哈希冲突,则将待增加数据增加至预分配区,并将待分配区链入哈希表区;若查询到删除数据任务,则直接删除对应单元空间中的数据。

[0068] 具体的,参照图3,当查询到任务请求为增加数据任务请求时,所述增加任务为增加数据A和数据B到哈希结构中;当待增加的数据A和B对应的哈希表区的单位空间中不存在数据时,则为无哈希冲突,直接将待增加数据A和B存入哈希表区1中。当待增加数据C对应的哈希表区的单位空间中已经存在数据A,则属于有哈希冲突的情况,无法将待增加数据C存入哈希表区对应的A数据所处的单位空间的位置中,则将待增加数据C增加到预分配区2中的单位空间中,并将待分配区2链入哈希表区1。

[0069] 示例性的,嵌入式实时系统是多任务系统,允许在使用后台任务方式实现键值数据库,能够将遍历、增加等等操作转化为任务请求的方式进行。在数据库启动的时候创建一个相应的任务,等待执行用户端的任务请求指令,若查询到任务请求指令则执行对应的任务请求,若查询不到任务请求则阻塞等待或者执行数据的刷新等内部操作。数据库后台任务执行遍历操作时,为了防止遍历过程过长,每遍历完一个单元就查询是否有其他的任务请求,若查询到有其他任务请求则插入执行任务,进行任务的执行,执行任务完成后,继续遍历,若查询到没有其他任务请求,则继续遍历。通过上述方法可以在一个遍历过程中插入执行其他的任务请求,避免了繁琐的加解锁操作,提高的遍历的工作效率。在哈希遍历的时候允许执行数据的其他的增加、删除数据等短时间的任务的执行,避免了其他任务的长时间等待,优化实时系统中数据的存取性能;避免遍历期间为了让出CPU的执行而对每个遍历单元单独执行加解锁操作所导致的性能消耗。

[0070] 进一步,参照图4,遍历方向从数据表区1到预分配区2;每遍历完一个单元空间,则进行任务请求查询。执行遍历的时候,则按照数组结构的方式遍历,则每个数据单元都可以被遍历到,哈希结构本身就是无序的,因此从用户的角度这哈希结构数据遍历和数组结构数据遍历的方式并没有区别,而采取数组结构方式遍历在遍历期间不需要维护链表的完整性,执行插入删除数据都不会导致不可预测的后果。由于预先分配了预分配区空间,使得在发生哈希冲突的时候不需要再从哈希表区内存堆中分配空间,由此避免了频繁申请释放内存导致的外部碎片。

[0071] 进一步的,一个遍历任务开始之前新建一个记录任务,所述记录任务是对发生了执行其他增加或者删除的任务请求的相对变化位置段的记录,对发生了执行其他增加或者删除的任务请求的相对变化位置段的数据进行重新查询,以核对遍历数据的准确性。

[0072] 示例性的,参照图4,在遍历任务开始之前新建一个记录任务以记录发生了执行其他增加或者删除的任务请求的相对变化位置段。例如,遍历完空间数据C后执行了增加任务,增加的数据是在数据A和数据C所处的单位空间之间,则通过记录任务记录数据A和数据C所处的单位空间之间的位置段,在全部单位空间遍历完成后,再所记录的数据A和数据C所

处的单元空间之间的位置段内的数据进行查询,以核对遍历数据的准确性。

[0073] 进一步的,所述数据为哈希结构数据、列表结构数据和数组结构数据任一种。创建迭代器对象,所述迭代器对象中存在缓存空间;将遍历的单元空间中的键值对存储到所述迭代器对象中的缓存空间中,以供用户端获取遍历数据;若迭代器对象缓存空间中的键值对数据全部被用户端获取完毕,则继续从上一次遍历的最后一个单元空间之后继续遍历,并将遍历得到的单元空间中的键值对数据存储到迭代器对象的缓存空间中。

[0074] 具体的,数据遍历通常是提供一个回调函数,在遍历每个数据单元时执行用户回调,但是由于数据库是单后台任务运行的,如果用户对机制不熟悉,在回调函数内执行过多耗时的操作,则有导致整个数据效率降低。以此,可通过迭代器遍历方式实现用户获取数据。在用户使用迭代器进行数据获取时,创建迭代器对象,所述迭代器对象中存在缓存空间;调用迭代器获取数据的接口,将遍历的单元空间中的键值对存储到所述迭代器对象中的缓存空间中,然后返回,等待用户下一次使用迭代器获取数据。用户在迭代器中获取数据时,优先从迭代器对象中的缓存空间中进行查找获取如果缓存空间的键值对都已经被用户获取完成,则再继续从上一次遍历的最后一个单元空间之后继续遍历,并将遍历得到的单元空间中的键值对数据存储到迭代器对象的缓存空间中。使用迭代器遍历的方式,一方面优化了加锁操作带来的性能消耗,由于它预分配了一个缓冲区,使得一次加锁访问数据库可以获得多个数据,降低了加锁访问的频率,因为缓冲区是迭代器独有而非数据库管理的,所以在缓冲区内的键值对访问都无需加锁。此外,嵌入式系统的数据除哈希结构数据外,常常还需要提供列表结构数据,数组结构数据等,迭代器可以提供一个统一的遍历方式,降低接口的复杂度。

[0075] 上述通过在遍历的过程中,每遍历完一个单元空间则进行任务请求查询,若查询到任务请求则进行任务执行后再继续遍历,若查询不到任务请求则继续遍历。采用上述技术手段,能够在一个遍历过程中进行其他任务的查询与执行,避免了其他任务的长时间等待,同时避免了繁琐的加解锁操作,从而提高了数据遍历的效率和有效性。

[0076] 实施例二:

[0077] 在上述实施例的基础上,图5为本申请实施例二提供的一种基于嵌入式系统的数据遍历装置的结构示意图。参考图6,本实施例提供的基于嵌入式系统的数据遍历装置具体包括:存储对象空间创建模块21和遍历模块22。

[0078] 其中,存储对象空间创建模块22,用于创建数据存储对象空间,所述数据存储对象空间分为数据表区和预分配区,所述数据存储对象空间为存储数据的单元空间集;

[0079] 遍历模块23,用于对数据存储对象空间进行遍历,遍历方向从数据表区到预分配区;每遍历完一个单元空间,则进行任务请求查询,所述单元空间存储对应数据信息;若查询到任务请求,则进行任务执行,执行任务后继续遍历;若查询不到任务请求,则继续遍历。

[0080] 进一步的,所述存储对象空间创建模块22,还用于创建哈希存储对象空间,所述哈希存储对象空间分为哈希表区和预分配区,所述哈希存储对象空间为存储数据的单元空间集;

[0081] 进一步的,所述遍历模块23,还用于对哈希存储对象空间进行遍历,遍历顺序方向从哈希表区到预分配区;每遍历完一个单元空间,则进行任务请求查询;若查询到任务请求,则进行任务执行,执行任务后继续遍历;若查询不到任务请求,则继续遍历。

[0082] 进一步的,所述基于嵌入式系统的数据遍历装置还包括任务请求查询模块,所述任务请求查询模块,用于所述任务请求包括增加数据任务和删除数据任务;所述若查询到任务请求,则进行任务执行,执行任务后继续遍历,具体为:若查询到增加数据任务,则判断增加数据至哈希表区是否存在哈希冲突,所述哈希冲突为待增加数据的单元空间中已经存在数据,将待增加数据增加至所述单元空间时发生的冲突;若无哈希冲突,则将待增加数据增加至哈希表区;若存在哈希冲突,则将待增加数据增加至预分配区,并将待分配区链入哈希表区;若查询到删除数据任务,则直接删除对应单元空间中的数据。

[0083] 进一步的,所述基于嵌入式系统的数据遍历装置还包括迭代器模块,所述迭代器模块,用于创建迭代器对象,所述迭代器对象中存在缓存空间;将遍历的单元空间中的键值对存储到所述迭代器对象中的缓存空间中,以供用户端获取遍历数据;若迭代器对象缓存空间中的键值对数据全部被用户端获取完毕,则继续从上一次遍历的最后一个单元空间之后继续遍历,并将遍历得到的单元空间中的键值对数据存储到迭代器对象的缓存空间中。

[0084] 进一步的,所述数据为哈希结构数据、列表结构数据和数组结构数据任一种。

[0085] 进一步的,所述迭代器模块,用于创建迭代器对象,所述迭代器对象中存在缓存空间;将遍历的单元空间中的键值对存储到所述迭代器对象中的缓存空间中,以供用户端获取遍历数据;若迭代器对象缓存空间中的键值对数据全部被用户端获取完毕,则继续从上一次遍历的最后一个单元空间之后继续遍历,并将遍历得到的单元空间中的键值对数据存储到迭代器对象的缓存空间中。

[0086] 进一步的,所述存储对象空间创建模块22,还用于将所述数据存储对象空间分为数据表区和预分配区,所述预分配区空间为所述数据表区空间的2倍。

[0087] 上述通过在遍历的过程中,每遍历完一个单元空间则进行任务请求查询,若查询到任务请求则进行任务执行后再继续遍历,若查询不到任务请求则继续遍历。采用上述技术手段,能够在遍历过程中进行其他任务的查询与执行,避免了其他任务的长时间等待,同时避免了繁琐的加解锁操作,从而提高了数据遍历的效率和有效性。

[0088] 本申请实施例二提供的基于嵌入式系统的数据遍历装置可以用于执行上述实施例一提供的基于嵌入式系统的数据遍历方法,具备相应的功能和有益效果。

[0089] 实施例三:

[0090] 本申请实施例三提供了一种电子设备,参照图6,该电子设备包括:处理器31、存储器32、通信模块33、输入装置34及输出装置35。该电子设备中处理器的数量可以是一个或者多个,该电子设备中的存储器的数量可以是一个或者多个。该电子设备的处理器、存储器、通信模块、输入装置及输出装置可以通过总线或者其他方式连接。

[0091] 存储器32作为一种计算机可读存储介质,可用于存储软件程序、计算机可执行程序以及模块,如本申请任意实施例所述的基于嵌入式系统的数据遍历方法对应的程序指令/模块(例如,基于嵌入式系统的数据遍历装置中的存储对象空间创建模块和遍历模块)。存储器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;存储数据区可存储根据设备的使用所创建的数据等。此外,存储器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、或其他非易失性固态存储器件。在一些实例中,存储器可进一步包括相对于处理器远程设置的存储器,这些远程存储器可以通过网络连接至设备。上述网络的实例包括但不

限于互联网、企业内部网、局域网、移动通信网及其组合。

[0092] 通信模块33用于进行数据传输。

[0093] 处理器31通过运行存储在存储器中的软件程序、指令以及模块,从而执行设备的各种功能应用以及数据处理,即实现上述的基于嵌入式系统的数据遍历方法。

[0094] 输入装置34可用于接收输入的数字或字符信息,以及产生与设备的用户设置以及功能控制有关的键信号输入。输出装置35可包括显示屏等显示设备。

[0095] 上述提供的电子设备可用于执行上述实施例一提供的基于嵌入式系统的数据遍历方法,具备相应的功能和有益效果。

[0096] 实施例四:

[0097] 本申请实施例还提供一种包含计算机可执行指令的存储介质,所述计算机可执行指令在由计算机处理器执行时用于执行一种基于嵌入式系统的数据遍历方法,该基于嵌入式系统的数据遍历方法包括:创建数据存储对象空间,所述数据存储对象空间分为数据表区和预分配区,所述数据存储对象空间为存储数据的单元空间集;对数据存储对象空间进行遍历,遍历方向从数据表区到预分配区;每遍历完一个单元空间,则进行任务请求查询;若查询到任务请求,则进行任务执行,执行任务后继续遍历;若查询不到任务请求,则继续遍历。

[0098] 存储介质——任何的各种类型的存储器设备或存储设备。术语“存储介质”旨在包括:安装介质,例如CD-ROM、软盘或磁带装置;计算机系统存储器或随机存取存储器,诸如DRAM、DDR RAM、SRAM、EDO RAM,兰巴斯(Rambus)RAM等;非易失性存储器,诸如闪存、磁介质(例如硬盘或光存储);寄存器或其它相似类型的存储器元件等。存储介质可以还包括其它类型的存储器或其组合。另外,存储介质可以位于程序在其中被执行的第一计算机系统中,或者可以位于不同的第二计算机系统中,第二计算机系统通过网络(诸如因特网)连接到第一计算机系统。第二计算机系统可以提供程序指令给第一计算机用于执行。术语“存储介质”可以包括驻留在不同位置中(例如在通过网络连接的不同计算机系统中)的两个或更多存储介质。存储介质可以存储可由一个或多个处理器执行的程序指令(例如具体实现为计算机程序)。

[0099] 当然,本申请实施例所提供的一种包含计算机可执行指令的存储介质,其计算机可执行指令不限于如上所述的基于嵌入式系统的数据遍历方法,还可以执行本申请任意实施例所提供的基于嵌入式系统的数据遍历方法中的相关操作。

[0100] 上述实施例中提供的基于嵌入式系统的数据遍历装置、存储介质及电子设备可执行本申请任意实施例所提供的基于嵌入式系统的数据遍历方法,未在上述实施例中详尽描述的技术细节,可参见本申请任意实施例所提供的基于嵌入式系统的数据遍历方法。

[0101] 上述仅为本申请的较佳实施例及所运用的技术原理。本申请不限于这里所述的特定实施例,对本领域技术人员来说能够进行的各种明显变化、重新调整及替代均不会脱离本申请的保护范围。因此,虽然通过以上实施例对本申请进行了较为详细的说明,但是本申请不仅仅限于以上实施例,在不脱离本申请构思的情况下,还可以包括更多其他等效实施例,而本申请的范围由权利要求的范围决定。

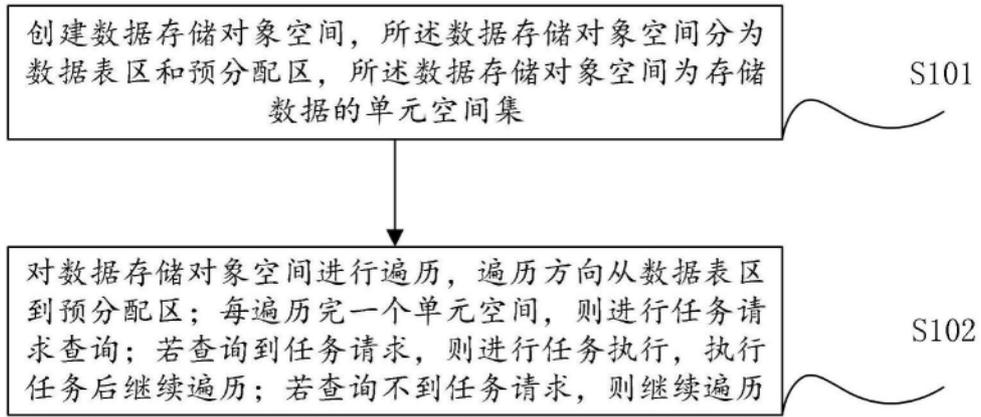


图1

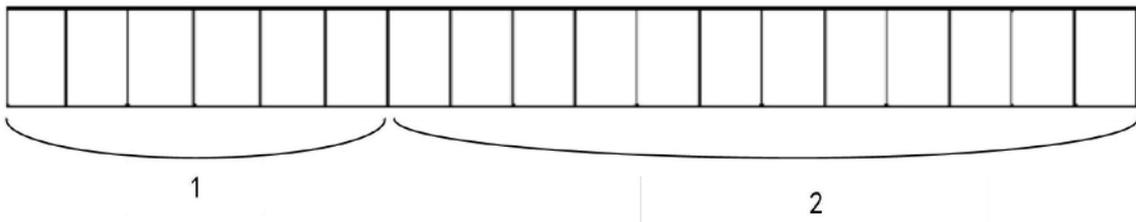


图2

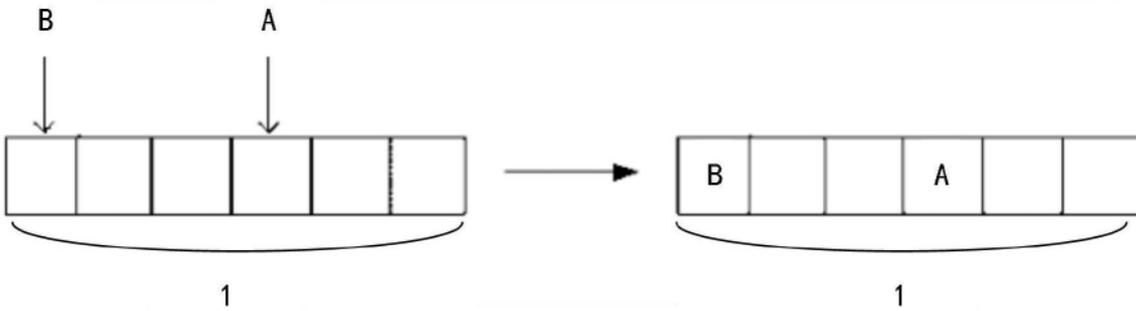


图3

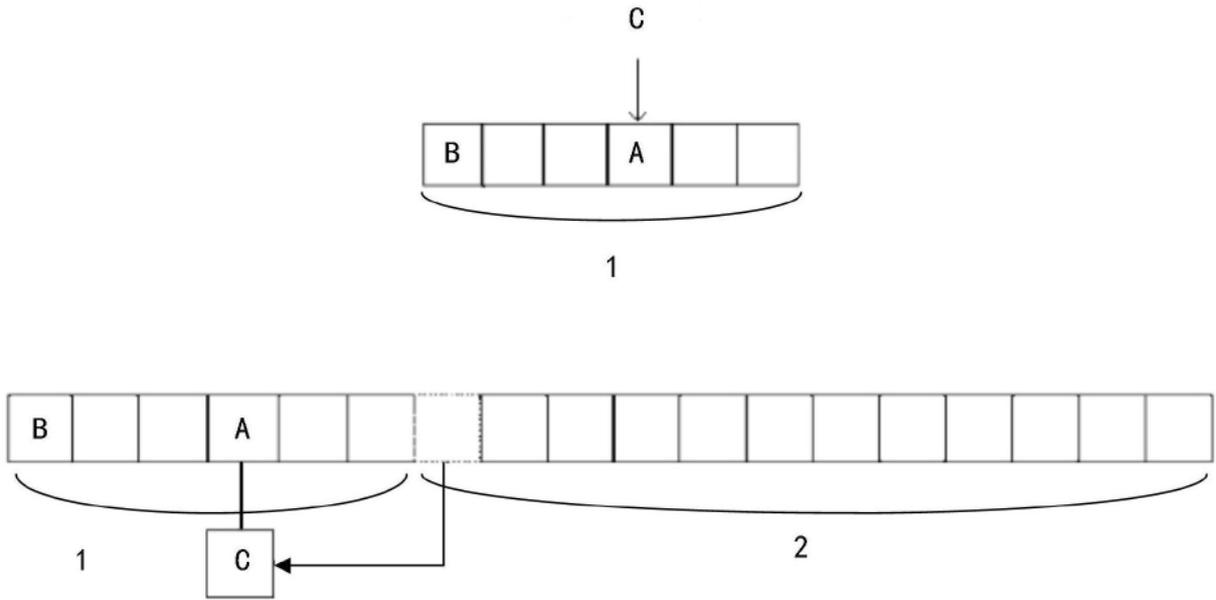


图4



图5

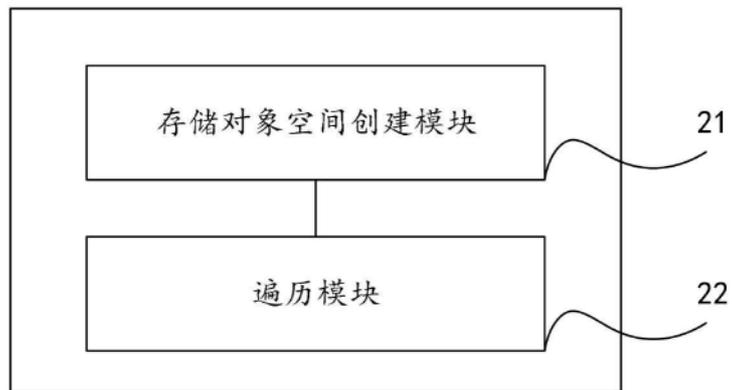


图6

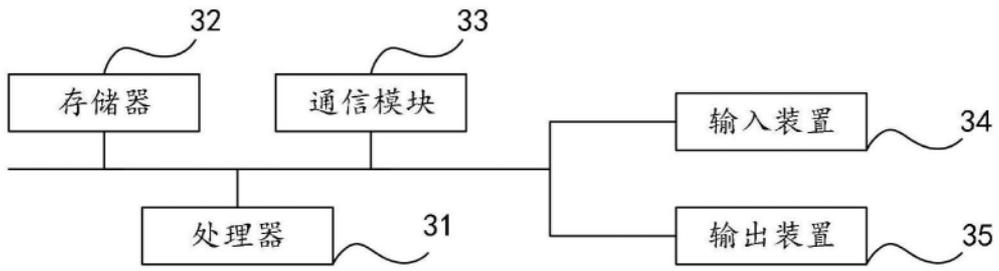


图7