(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 118819871 A
(43)申请公布日 2024.10.22

(21)申请号 202411305193.1

(22)申请日 2024.09.19

(71)申请人 阿里云计算有限公司
地址 310030 浙江省杭州市西湖区三墩镇灯彩街1008号云谷园区1-2-A06室

(72)发明人 钟江 庞训磊 马涛 杨勇

(74)专利代理机构 北京辰权知识产权代理有限公司 11619
专利代理师 包莉莉

(51)Int.Cl.
G06F 9/50(2006.01)
G06F 9/455(2018.01)

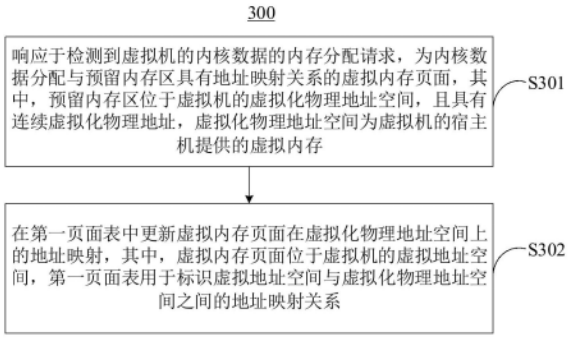
权利要求书2页 说明书13页 附图4页

(54)发明名称

内存管理方法、宿主机、电子设备、存储介质和程序产品

(57)摘要

本申请实施例提供一种内存管理方法、宿主机、电子设备、存储介质和程序产品，涉及计算机技术领域，方法包括：响应于检测到虚拟机的内核数据的内存分配请求，为内核数据分配与预留内存区具有地址映射关系的虚拟内存页面，其中，预留内存区位于虚拟机的虚拟化物理地址空间，且具有连续的虚拟化物理地址，虚拟化物理地址空间为虚拟机的宿主机提供的虚拟内存；在第一页面表中更新虚拟内存页面在虚拟化物理地址空间上的地址映射，其中，虚拟内存页面位于虚拟机的虚拟地址空间，第一页面表用于表示虚拟地址空间与虚拟化物理地址空间之间的地址映射关系。本申请实施例的技术方案可以避免内核数据内存分配的随机性，减少内存碎片的产生。



CN 118819871 A

1.一种内存管理方法,应用于虚拟机,所述方法包括:

响应于检测到所述虚拟机的内核数据的内存分配请求,为所述内核数据分配与预留内存区具有地址映射关系的虚拟内存页面,其中,所述预留内存区位于所述虚拟机的虚拟化物理地址空间,且具有连续的虚拟化物理地址,所述虚拟化物理地址空间为所述虚拟机的宿主机提供的虚拟内存;

在第一页面表中更新所述虚拟内存页面在所述虚拟化物理地址空间上的地址映射,其中,所述虚拟内存页面位于所述虚拟机的虚拟地址空间,所述第一页面表用于表示所述虚拟地址空间与所述虚拟化物理地址空间之间的地址映射关系。

2.根据权利要求1所述的方法,还包括:

响应于在至少一个所述虚拟内存页面中识别到冷页面,向所述宿主机发送针对所述冷页面的内存释放请求,所述内存释放请求用于请求所述宿主机从所述宿主机的物理地址空间中释放所述冷页面对应的物理内存页面,所述冷页面为在预设时长内被访问的频率小于第一预设频率阈值的页面;

从所述预留内存区中释放所述冷页面对应的虚拟化物理内存页面,并在所述第一页面表中解除所述冷页面在所述虚拟化物理地址空间上的地址映射。

3.根据权利要求1所述的方法,还包括:

响应于在至少一个所述虚拟内存页面中识别到热页面,向所述宿主机发送页面整合请求,所述页面整合请求用于请求所述宿主机将所述热页面对应的物理内存页面整合为目标页面,所述热页面为在预设时长内被访问的频率大于第二预设频率阈值的页面,所述目标页面的尺寸大于尺寸阈值。

4.根据权利要求1至3中任一项所述的方法,其中,所述预留内存区包括多个具有连续虚拟化物理地址的子区域,所述子区域的大小与目标页面的大小相同,所述目标页面的尺寸大于尺寸阈值,从预留内存区中为所述内核数据分配对应的虚拟内存页面,包括:

从多个所述子区域中选择至少一个所述子区域分配给所述虚拟内存页面。

5.根据权利要求1至3中任一项所述的方法,其中,所述虚拟内存页面对应连续的虚拟化物理地址。

6.一种内存管理方法,应用于宿主机,所述方法包括:

从所述宿主机的物理地址空间中为虚拟机分配虚拟化物理地址空间,其中,所述虚拟化物理地址空间中包括具有连续虚拟化物理地址的预留内存区,所述虚拟机用于为所述虚拟机的内核数据分配与所述预留内存区具有地址映射关系的虚拟内存页面;

利用第二页面表表示所述物理地址空间与所述虚拟化物理地址空间之间的地址映射关系。

7.根据权利要求6所述的方法,还包括:

接收所述虚拟机针对冷页面发送的内存释放请求,其中,所述冷页面为所述虚拟机从至少一个所述虚拟内存页面中识别,且在预设时长内被访问的频率小于第一预设频率阈值的页面;

从所述物理地址空间中释放所述冷页面对应的物理内存页面,并在所述第二页面表中解除所述冷页面对应的虚拟化物理内存页面在所述物理地址空间上的地址映射。

8.根据权利要求6所述的方法,还包括:

接收所述虚拟机针对热页面发送的页面整合请求,其中,所述热页面为所述虚拟机从至少一个所述虚拟内存页面中识别,且在预设时长内被访问的频率大于第二预设频率阈值的页面;

将所述热页面对应的物理内存页面整合为目标页面,并在所述第二页面表中更新整合后的目标页面在所述虚拟化物理地址空间上的地址映射,所述目标页面的尺寸大于尺寸阈值。

9.一种宿主机,包括:

非易失性存储器,存储有内存管理程序;

物理内存,提供物理地址空间;

处理器,承载一个或多个虚拟机,所述处理器在执行所述内存管理程序时实现权利要求1至8中任一项所述的方法。

10.一种电子设备,包括存储器、处理器及存储在存储器上的计算机程序,所述处理器在执行所述计算机程序时实现权利要求1至8中任一项所述的方法。

11.一种计算机可读存储介质,所述计算机可读存储介质内存储有计算机程序,所述计算机程序被处理器执行时实现权利要求1至8中任一项所述的方法。

12.一种计算机程序产品,包括计算机程序,所述计算机程序在被处理器执行时实现根据权利要求1至8中任一项所述的方法。

内存管理方法、宿主机、电子设备、存储介质和程序产品

技术领域

[0001] 本申请涉及计算机技术领域,尤其涉及一种内存管理方法、宿主机、电子设备、计算机可读存储介质和计算机程序产品,可应用于虚拟化内存管理技术领域。

背景技术

[0002] 内存碎片化是指内存中可用的空闲块被分成了许多不连续的小块。操作系统内核数据分布在系统的不可移动内存中,由于这些内核数据在系统运行时被频繁分配和释放,它们可能会在内存中形成许多小而分散的块。相关技术中,支持以大尺寸页面(Huge Page,HP)的粒度(如2MB)进行内存热插拔操作。但是,内核数据可能分散在这些2MB的内存块中,导致这些内存块不能被释放,尽管总的空闲内存量可能足够,但由于这些内存块是分散的,无法进行大块连续内存的分配。因此,如何降低内存碎片化是亟待解决的技术问题。

发明内容

[0003] 本申请实施例提供一种内存管理方法、宿主机、电子设备、计算机可读存储介质和计算机程序产品,以缓解或解决现有技术中存在的一项或多项技术问题。

[0004] 第一方面,本申请实施例提供了一种内存管理方法,应用于虚拟机,所述方法包括:

响应于检测到所述虚拟机的内核数据的内存分配请求,为所述内核数据分配与所述预留内存区具有地址映射关系的虚拟内存页面,其中,所述预留内存区位于所述虚拟机的虚拟化物理地址空间,且具有连续的虚拟化物理地址,所述虚拟化物理地址空间为所述虚拟机的宿主机提供的虚拟内存;在第一页面表中更新所述虚拟内存页面在所述虚拟化物理地址空间上的地址映射,其中,所述虚拟内存页面位于所述虚拟机的虚拟地址空间,所述第一页面表用于表示所述虚拟地址空间与所述虚拟化物理地址空间之间的地址映射关系。

[0005] 第二方面,本申请实施例提供了一种内存管理方法,应用于宿主机,所述方法包括:从所述宿主机的物理地址空间中为虚拟机分配虚拟化物理地址空间,其中,所述虚拟化物理地址空间中包括具有连续虚拟化物理地址的预留内存区,所述虚拟机用于为所述虚拟机的内核数据分配与所述预留内存区具有地址映射关系的虚拟内存页面;利用第二页面表表示所述物理地址空间与所述虚拟化物理地址空间之间的地址映射关系。

[0006] 第三方面,本申请实施例提供了一种宿主机,包括:非易失性存储器,存储有内存管理程序;物理内存,提供物理地址空间;处理器,承载一个或多个虚拟机,所述处理器在执行所述内存管理程序时实现本申请实施例任一项的方法。

[0007] 第四方面,本申请实施例提供了一种电子设备,包括存储器、处理器及存储在存储器上的计算机程序,处理器在执行计算机程序时实现本申请实施例任一项的方法。

[0008] 第五方面,本申请实施例提供了一种计算机可读存储介质,计算机可读存储介质内存储有计算机程序,计算机程序被处理器执行时实现本申请实施例任一项的方法。

[0009] 第六方面,本申请实施例提供了一种计算机程序产品,包括计算机程序,计算机程

序在被处理器执行时实现本申请实施例任一项的方法。

[0010] 根据本申请实施例的技术方案,在Guest侧的虚拟化物理地址空间中固定设置一个预留内存区,并在后续检测到针对内核数据的内存分配请求时,从该预留内存区中为内核数据分配对应的虚拟内存页面,如此便可以避免内核数据内存分配的随机性,将内核数据所占用的内存集中在固定区域,从而减少内存碎片的产生。

[0011] 可选地,通过识别虚拟内存页面中的冷页面,从预留内存区中释放冷页面对应的虚拟化物理内存页面,下线冷页面对应的虚拟化物理内存页面,避免它们被重新分配,一方面可以实现不可移动内存区中的内核内存页面迁移,另一方面,由于释放的内存资源可以被其他内核数据使用,从而可以优化内存资源的使用。

[0012] 可选地,通过识别虚拟内存页面中的热页面,将热页面对应的物理内存页面合并为尺寸大于尺寸阈值的目标页面,如大尺寸页面HP,基于大尺寸页面HP的使用,可以进一步减少内存碎片,降低页表项过多带来的内存管理开销。

[0013] 上述说明仅是本申请技术方案的概述,为了能够更清楚了解本申请的技术手段,可依照说明书的内容予以实施,并且为了让本申请的上述和其他目的、特征和优点能够更明显易懂,以下特举本申请的具体实施方式。

附图说明

[0014] 在附图中,除非另外规定,否则贯穿多个附图相同的附图标记表示相同或相似的部件或元素。这些附图不一定是按照比例绘制的。应该理解,这些附图仅描绘了根据本申请的一些实施方式,而不应将其视为是对本申请范围的限制。

[0015] 图1示出相关技术中虚拟机的内存管理方法;

图2A示出本申请实施例提供的一种宿主机200的结构示意图;

图2B示出本申请实施例提供的一种宿主机200的系统架构示意图;

图2C示出本申请实施例提供的内存管理系统210的架构图;

图2D示出本申请实施例提供的内存管理系统210的内存地址空间的示意图;

图3示出了本申请实施例的内存管理方法300的流程图;

图4示出了本申请实施例的内存管理方法400的流程图;

图5示出了本申请实施例提供的电子设备的框图。

具体实施方式

[0016] 在下文中,仅简单地描述了某些示例性实施例。正如本领域技术人员可认识到的那样,在不脱离本申请的构思或范围的情况下,可通过各种不同方式修改所描述的实施例。因此,附图和描述被认为本质上是示例性的,而非限制性的。

[0017] 为便于理解本申请实施例的技术方案,以下对本申请实施例的相关技术进行说明。以下相关技术作为可选方案与本申请实施例的技术方案可以进行任意结合,其均属于本申请实施例的保护范围。

[0018] 为了便于理解,首先介绍下文中将使用的术语:

内核态:是操作系统内核(Kernel)运行的状态或空间域,内核态下运行的是操作系统内核本身,包括内存管理、进程调度、文件系统管理、驱动程序等关键功能。

[0019] 用户态:是指应用程序在较低权限级别下运行的状态或空间域,用户态运行的是所有普通的应用程序,如浏览器、文本编辑器、游戏等。

[0020] 32位直接内存访问(Direct Memory Access 32,DMA32):是一个特殊的内存区,用于支持需要直接内存访问(Direct Memory Access,DMA)的设备,这些设备只能访问低于4GB 的物理内存(当其位于虚拟机侧时为虚拟化物理内存),用于解决一些 32 位硬件设备在 64 位系统上进行 DMA 时的兼容性问题。

[0021] 常规(Normal)内存:是操作系统中常用的内存区,主要用于内核数据的内存分配和操作系统的常规操作。

[0022] 可移动(Movable)内存:是可移动的内存区,通常用于用户态下的内存分配(如匿名内存映射和页面缓存)。

[0023] malloc函数:用户空间中的标准内存分配接口,用于动态分配指定大小的内存页面,分配的内存页面的大小和数量可以在程序运行时动态调整。

[0024] Slab 分配器:是Linux操作系统的一种内存分配机制,它通过将内存分成固定大小的缓存(称为Slab)来减少内存碎片并提高分配效率,Slab 分配器通常用于分配和管理固定大小的内存块。

[0025] vmalloc函数:是一个内核接口,用于分配虚拟地址连续的虚拟内存页面,而不要求物理内存是连续的,通过映射非连续的物理页面来实现虚拟内存页面的连续性。

[0026] 页面表(Page Table,pgtable)接口:是页面表的管理和分配接口。页面表是操作系统用来表示页面的虚拟地址到物理地址的映射关系。

[0027] 伙伴系统(Buddy System):是一种用于内存管理的算法,通过将内存块分割成大小为2的幂次方的内存块,从而有效地管理和分配内存,减少内存碎片。当系统需要分配一块内存时,首先找到满足需求的最小块。如果没有合适大小的块,系统会将更大的块分成两个相等的小块,直到找到合适大小的块为止。当内存不再使用时,系统会释放这块内存,并尝试将它与相邻的“伙伴”块合并。如果两个“伙伴”块都空闲且大小相同,它们就可以合并为一个更大的块。这个过程可以递归进行,直到无法再合并。

[0028] 内存碎片化:是指内存中可用的空闲块被分成了许多不连续的小块。

[0029] 图1示出相关技术中虚拟机的内存管理方法,如图1所示,虚拟机(Guest)的虚拟化物理地址空间包括直接内存、不可移动内存和可移动内存。其中,虚拟化物理地址空间为虚拟机的宿主机分配的虚拟内存。直接内存可以实现某些硬件设备进行直接内存访问,例如为DMA32;不可移动内存主要用于内核数据的内存分配,例如为Normal内存;可移动内存主要用于用户数据的内存分配。

[0030] 在虚拟机(Guest)上进行内存分配包括内核态下的内存分配和用户态下的内存分配。其中,在用户态下的内存分配过程包括:应用程序进程调用用户数据内存分配接口发送内存分配请求,系统将从可移动内存中为其分配虚拟内存页面;在内核态下的内存分配过程包括内核进程调用内核数据内存分配接口发送内存分配请求,系统将从不可移动内存中为其分配虚拟内存页面。其中,虚拟内存页面可以理解为具有虚拟地址的内存页面。用户数据内存分配接口例如为malloc函数,用于用户态空间的标准内存分配接口。内核数据内存分配接口例如为Slab 分配器或vmalloc函数或pgtable接口。

[0031] 由于操作系统内核数据分布在系统的不可移动内存中,而这些内核数据在系统运

行时被频繁分配和释放,它们可能会在内存中形成许多小而分散的块。即使采用以大尺寸页面HP的粒度(如2MB粒度)进行内存热插拔操作。但是,内核数据可能分散在这些2MB的内存块中,导致这些内存块不能被释放,尽管总的空闲内存量可能足够,但由于这些内存块是分散的,无法进行大块连续内存的分配,因此内存碎片化的问题仍然存在,这将影响虚拟化场景下的容器缩容。

[0032] 本申请实施例旨在提供一种内存管理方法、宿主机、电子设备、计算机可读存储介质和计算机程序产品,以降低内存碎片化,提高内存资源的利用率。为了便于理解,首先结合图2A和图2B介绍本申请实施例的硬件架构和软件架构。

[0033] 图2A示出本申请实施例提供的一种宿主机(Host)200的结构示意图,图2B示出本申请实施例提供的一种宿主机的系统架构示意图。具体地,如图2A和图2B所示,该宿主机200的硬件层结构可以包括处理器201、非易失性存储器202和物理内存203。

[0034] 处理器201例如为中央处理器(Central Processing Unit,CPU);物理内存203例如为随机访问存储器(Random Access Memory,RAM),提供宿主机(Host)200的物理地址空间。处理器201、非易失性存储器202和物理内存203的数量可以是一个或多个,它们通过总线204相互连接以完成相互间的通信。

[0035] 当宿主机(Host)200启动后,处理器201首先从非易失性存储器202中读取启动程序从而加载Host操作系统206。Host操作系统206运行在Host内核空间,控制和管理实际的硬件资源(包括处理器201、非易失性存储器202和物理内存203等),以及运行一个或多个Host应用程序207。Host用户空间提供了在Host操作系统206上运行Host应用程序207的环境。进一步地,在虚拟化场景中,Host操作系统206会加载基于内核的虚拟机(Kernel-based Virtual Machine,KVM)模块208,当用户在HOST操作系统206上启动一个或多个虚拟机(Guest)205时,KVM模块208负责创建虚拟机205的虚拟硬件资源。Guest操作系统运行在Guest内核空间,负责管理虚拟机205中的虚拟硬件资源,并提供在Guest用户空间运行Guest应用程序的环境。

[0036] 本申请实施例中,非易失性存储器202上存储有内存管理程序,当处理器201执行该内存管理程序时,可实现本申请实施例的内存管理方法。下面结合图2C和图2D进行具体介绍。

[0037] 图2C示出本申请实施例提供的内存管理系统210的架构图,该内存管理系统210可运行在宿主机(Host)200上。非易失性存储器202上存储的内存管理程序包括Host内存管理子程序和Guest内存管理子程序。如图2C所示,Host操作系统启动后,从非易失性存储器202中读取Host内存管理子程序,从而加载内存管理单元211,内存管理单元211实现Host侧的内存管理方法。Guest操作系统启动后,通过与KVM模块交互,从非易失性存储器202中读取Guest内存管理子程序,从而加载内存分配器212,内存分配器212实现Guest侧的内存管理方法。

[0038] 图2D示出了内存管理系统210的内存地址空间的示意图。内存管理系统210会按照设定的内存管理粒度进行内存管理,每一内存管理粒度可称为页面(page),或者也可称为内存块。每一个页面均对应有其内存地址。内存地址空间可以划分为Host侧的物理地址空间、Guest侧的虚拟化物理地址空间和Guest侧的虚拟地址空间。

[0039] 如图2C和图2D所示,物理内存202提供Host侧的物理地址空间,物理地址空间包括

多个物理地址页面,即物理地址页面位于物理地址空间。

[0040] Host侧的内存管理单元211可以从物理地址空间中选择物理地址连续或物理地址不连续的物理内存页面,并将与该物理内存页面具有地址映射关系的虚拟内存页面分配给Guest上的每个进程,由此构建了Guest侧的虚拟化物理地址空间。也就是说,Guest侧的虚拟化物理地址空间包含了多个由Host侧分配的虚拟内存页面,但在Guest看来,这些虚拟内存页面是其物理内存页面。因此,本申请实施例中,将Host为Guest分配的虚拟内存页面叫做虚拟化物理内存页面。虚拟化物理内存页面对应有虚拟化物理地址,且位于Guest侧的虚拟化物理地址空间。

[0041] 其中,物理地址空间与虚拟化物理地址空间之间的地址映射关系由第二页面表表示,即第二页面表中包括虚拟化物理内存页面与物理内存页面之间的地址映射。示例性地,第二页面表由Host侧的内存管理单元211管理。

[0042] 虚拟化物理地址空间包括不可移动内存区和可移动(Movable)内存区,不可移动内存区例如为Normal内存。其中,Host会为Guest上的每个进程分配虚拟化物理内存页面。这些进程可以来自于Guest内核空间,例如为Guest操作系统进程,分配的虚拟化物理内存页面可进一步用于Guest操作系统内核数据的内存分配,相应地,这部分虚拟化物理内存页面位于不可移动内存区。这些进程也可以来自于Guest用户空间,例如为Guest应用程序进程,分配的虚拟化物理内存页面可进一步用于Guest用户数据的内存分配,相应地,这部分虚拟化物理内存页面位于可移动内存区。

[0043] 如图2D所示,Guest侧的内存管理模块可以从虚拟化物理地址空间中选择虚拟化物理地址连续或不连续的虚拟化物理内存页面,并将与该虚拟化物理内存页面具有地址映射关系的虚拟内存页面分配给Guest上的每个进程,由此构建了Guest侧的虚拟地址空间。也就是说,Guest侧的虚拟地址空间包含了多个虚拟内存页面。

[0044] 其中,虚拟化物理地址空间与虚拟地址空间之间的地址映射关系由第一页面表表示,即第一页面表中包括虚拟化物理内存页面与虚拟内存页面之间的地址映射。示例性地,第一页面表由Guest操作系统内核管理。

[0045] 如图2C所示,内存分配器212可以向Host申请一个预留内存区213,该预留内存区213位于虚拟化物理地址空间,且预留内存区213具有连续的虚拟化物理地址。具体地,如图2C和图2D所示,预留内存区213位于不可移动内存区,用于为Guest操作系统的内核数据分配虚拟内存页面。

[0046] 示例性地,可以在KVM模块208的配置文件中设置Guest的初始内存,例如配置文件中的init_mem_size参数决定了Guest在创建或启动时会获得多大的虚拟化物理内存。在Guest启动后,Guest操作系统内核会根据init_mem_size参数预留一块虚拟化物理地址连续的区域作为预留内存区。

[0047] 也就是说,本申请实施例中,在Guest侧的虚拟化物理地址空间中固定设置一个预留内存区,后续内核数据的虚拟内存页面均从该预留内存区中分配,如此便可以避免内核数据内存分配的随机性,将内核数据所占用的内存集中在固定区域,从而减少内存碎片的产生。

[0048] 下面以具体的实施例对本申请的Host侧的内存管理方法和Guest侧的内存管理方法进行详细说明。所列举的若干具体的实施例可以相互结合,对于相同或相似的概念或过

程可能在某些实施例中不再赘述。

[0049] 图3示出了本申请实施例的内存管理方法300的流程图,该内存管理方法300可应用于虚拟机(Guest),例如由虚拟机的内存分配器实现。需要说明的是,内存分配器运行在Guest内核空间,即内存管理方法300用于实现Guest内核数据的内存管理。如图3所示,内存管理方法300可以包括步骤S301和步骤S302。

[0050] 步骤S301:响应于检测到虚拟机的内核数据的内存分配请求,为内核数据分配与预留内存区具有地址映射关系的虚拟内存页面,其中,预留内存区位于虚拟机的虚拟化物理地址空间,且具有连续虚拟化物理地址,虚拟化物理地址空间为虚拟机的宿主机提供的虚拟内存。

[0051] 其中,内核数据即Guest操作系统在运行过程中使用和管理的各种数据结构和信息,包括但不限于进程信息、内存管理数据、文件系统数据、驱动程序数据等。内存分配请求通常由不同模块或子系统发出,如进程、内存管理、设备驱动、文件系统或应用程序等。如图2C所示,Guest侧的内存分配器212将检测内存分配请求是否是针对内核数据的内存分配请求。

[0052] 在一个示例中,如果内存分配请求来自Guest操作系统进程,则可确定该内存分配请求是针对内核数据的内存分配请求。

[0053] 在另一个示例中,Guest进程可以调用内存分配接口来实现内存分配请求。当Guest进程需要分配内存时,将带有内存分配标识(如“GFP_KERNEL”或“GFP_Movable”等)的信息传递给内存分配接口(如kmalloc函数、vmalloc函数或alloc_pages 函数或malloc函数等),来发送内存分配请求。这些信息可以指示内存分配的来源(如可移动内存区或不可移动内存区或DMA内存区)。其中,内核数据进程对应其专属的内核内存分配标识,如“GFP_KERNEL”,内存分配器212通过对各进程的内存分配请求进行检测,包括筛选和过滤,以识别内核内存分配标识,一旦在内存分配请求中识别到内核内存分配标识,则判定该内存分配请求是针对内核数据的内存分配请求。

[0054] 如果确定内存分配请求是针对内核数据的内存分配请求,则交由内存分配器212进行管理,内存分配器212会从预留内存区213中选择虚拟化物理内存页面分配给该内核数据,与该虚拟化物理内存页面具有地址映射关系的虚拟内存页面即为该内核数据的虚拟内存页面。如果确定内存分配请求并非针对内核数据的内存分配请求,则交由Guest上的伙伴系统进行内存管理。本申请实施例中对内存的“管理”包括分配内存、释放(回收)内存、内存初始化、内存扩容、内存缩容、内存对齐等中的任意一种或多种操作,具体可基于实际 情况而定。

[0055] 示例性地,内存分配器212通过调用内核数据内存分配接口如vmalloc函数或alloc_contig_pages函数来为内核数据分配虚拟内存页面。

[0056] 其中,vmalloc函数可以分配虚拟地址连续但其对应的虚拟化物理地址可能不能连续的虚拟内存页面。vmalloc函数适用需要大量连续虚拟地址空间的情况,从而可以更灵活地管理和映射内存。当需要将 vmalloc 分配的虚拟内存页面从当前位置迁移到其他位置时,可以解除(unmap)之前分配的虚拟内存页面在虚拟化物理地址空间上的地址映射,便于重新分配这块区域,或将这块区域上的内核数据迁移到其他位置,从而可以在虚拟化环境中实现内核页面的迁移目标,从而更灵活地管理和优化内存资源。

[0057] alloc_contig_pages函数可以分配对应虚拟化物理地址连续的虚拟内存页面。在使用 alloc_contig_pages 函数分配内存时,如果目标虚拟化物理内存区域内存在可迁移页面,系统会先将这些页面迁移到其他位置,确保分配的虚拟内存页面所对应的虚拟化物理地址是连续的。

[0058] 步骤S302:在第一页面表中更新虚拟内存页面在虚拟化物理地址空间上的地址映射,其中,虚拟内存页面位于虚拟机的虚拟地址空间,第一页面表用于表示虚拟地址空间与虚拟化物理地址空间之间的地址映射关系。

[0059] 内存分配器212在从预留内存区213中选择虚拟化物理内存页面分配给该内核数据,用作该内核数据的虚拟内存页面后,会在第一页面表中更新所选择的虚拟化物理内存页面与该内核数据的虚拟内存页面之间的地址映射。

[0060] 根据本申请实施例的内存管理方法300,在Guest侧的虚拟化物理地址空间中固定设置一个预留内存区,并在后续检测到针对内核数据的内存分配请求时,从该预留内存区中为内核数据分配对应的虚拟内存页面,如此便可以避免内核数据内存分配的随机性,将内核数据所占用的内存集中在固定区域, 从而减少内存碎片的产生。

[0061] 在一种实施方式中,预留内存区包括多个具有连续虚拟化物理地址的子区域,子区域的大小与目标页面的大小相同,目标页面的尺寸大于尺寸阈值,在步骤S301中,从预留内存区中为内核数据分配对应的虚拟内存页面,可以包括:从多个子区域中选择至少一个子区域分配给虚拟内存页面。

[0062] 其中,标准页面的大小通常为4KB,而大尺寸页面(Huge Pages,HP)的大小可以是2MB、1GB或更大。由于在内存管理系统中,需要页面表(如第一页面表和第二页面表)来表示地址映射关系,每组地址映射关系对应一个页表项(Page Table Entry,PTE),如果采用标准页面为内存管理粒度,将会产生大量的页表项,因此会消耗大量内存来管理页面表。其中,尺寸阈值例如为2M或4M,即目标页面为大尺寸页面HP。

[0063] 本申请实施例中,将预留内存区划分为多个HP大小的子区域,从而实现以HP大小为内存管理粒度,例如,以2MB为粒度对预留内存区进行管理,包括内存分配时以HP大小为粒度以及释放内存以HP大小为粒度。基于此,可以减少页表项过多带来的内存管理开销,并可以加速转换后备缓冲区(Translation Lookaside Buffer,TLB)的地址转换,提高TLB的命中率。其中,可以根据内存需求和负载从多个子区域中选择至少一个子区域。

[0064] 在一种实施方式中,在步骤S301中分配的虚拟内存页面对应连续的虚拟化物理地址。例如:内存分配器212通过调用alloc_contig_pages函数,从预留内存区中选择虚拟化物理地址连续的虚拟内存页面分配给内核数据。在虚拟化环境中,内存的分配和释放是频繁的,通过分配具有连续虚拟化物理地址的虚拟内存页面,可以更好地提高内存回收和重用的效率,并进一步减少碎片化的影响。

[0065] 在一种实施方式中,本申请实施例的内存管理方法300还可以包括:响应于在至少一个虚拟内存页面中识别到冷页面,向宿主机发送针对冷页面的内存释放请求,内存释放请求用于请求宿主机从宿主机的物理地址空间中释放冷页面对应的物理内存页面,其中,冷页面为在预设时长内被访问的频率小于第一预设频率阈值的页面;从预留内存区中释放冷页面对应的虚拟化物理内存页面,并在第一页面表中解除冷页面在虚拟化物理地址空间上的地址映射。

[0066] 示例性地,Guest操作系统内核会定期检查和统计各个虚拟内存页面的访问情况。例如可以通过软中断来记录虚拟内存页面的访问时间戳;又如可以通过第一页面表中页表项中记录的访问位(Accessed bit)来判断该虚拟内存页面是否被访问过;再如,Guest操作系统内核可以周期性地扫描虚拟内存页面的访问情况。内存分配器212可以获取各个虚拟内存页面的访问情况,进而统计虚拟内存页面在预设时长内被访问的频率是否小于第一预设频率阈值,如果小于第一预设频率阈值,则将该虚拟内存页面标记为冷页面。

[0067] 内存分配器212在识别到冷页面后,可以向Host发送针对冷页面的内存释放请求。例如,内存分配器212可以使用CPU指令Hypercall,向Host的内存管理单元211发送内存释放请求,并通知内存管理单元211该冷页面对应的虚拟化物理地址。其中,内存分配器212可以通过查找第一页面表确定该冷页面对应的虚拟化物理地址。

[0068] 内存管理单元211在收到内存释放请求后,将查找第二页面表,根据该冷页面对应的虚拟化物理地址确定该冷页面对应的物理地址,并在物理地址空间中释放该物理地址对应的物理内存页面。内存管理单元211会将该物理内存页面上的内核数据写入磁盘或其他存储设备,实现页面换出。如果这些页面再次被访问,内存管理单元211可以将其重新加载到物理地址空间中。

[0069] 进一步地,内存分配器212从预留内存区中释放冷页面对应的虚拟化物理内存页面,并在第一页面表中解除冷页面在虚拟化物理地址空间上的地址映射,从而下线冷页面对应的虚拟化物理内存页面,避免它们被重新分配。

[0070] 基于此,一方面可以实现不可移动内存区中的内核内存页面迁移,另一方面,由于释放的内存资源可以被其他内核数据使用,从而可以优化内存资源的使用。

[0071] 在一种实施方式中,本申请实施例的内存管理方法300还可以包括:响应于在至少一个虚拟内存页面中识别到热页面,向宿主机发送页面整合请求,页面整合请求用于请求宿主机将热页面对应的物理内存页面整合为目标页面,其中,热页面为在预设时长内被访问的频率大于第二预设频率阈值的页面,目标页面的尺寸大于尺寸阈值。

[0072] 示例性地,内存分配器212可以获取各个虚拟内存页面的访问情况,进而统计虚拟内存页面在预设时长内被访问的频率是否大于第二预设频率阈值,如果大于第二预设频率阈值,则将该虚拟内存页面标记为热页面。其中,第一预设频率阈值小于或等于第二预设频率阈值。

[0073] 内存分配器212在识别到热页面后,可以向Host发送针对热页面的页面整合请求。例如,内存分配器212可以使用CPU指令Hypercall,向Host的内存管理单元211发送页面整合请求,并通知内存管理单元211该热页面对应的虚拟化物理地址。其中,内存分配器212可以通过查找第一页面表确定该热页面对应的虚拟化物理地址。

[0074] 其中,尺寸阈值例如为2M或4M,即目标页面为大尺寸页面HP。内存管理单元211在收到页面整合请求后,将查找第二页面表,根据该热页面对应的虚拟化物理地址确定该热页面对应的物理地址,并在物理地址空间中释放该物理地址对应的物理内存页面合并为大尺寸页面HP。

[0075] 基于HP的使用,可以进一步减少内存碎片,降低页表项过多带来的内存管理开销,并可以加速TLB的地址转换,提高TLB的命中率。

[0076] 图4示出了本申请实施例的内存管理方法400的流程图,该内存管理方法400可应

用于宿主机(Host),例如由宿主机的内存管理单元实现。如图4所示,内存管理方法400可以包括:

步骤S401:从宿主机的物理地址空间中为虚拟机分配虚拟化物理地址空间,其中,虚拟化物理地址空间中包括具有连续虚拟化物理地址的预留内存区,虚拟机用于为虚拟机的内核数据分配与预留内存区具有地址映射关系的虚拟内存页面;

步骤S402:利用第二页面表表示物理地址空间与虚拟化物理地址空间之间的地址映射关系。

[0077] 在一种实施方式中,内存管理方法400还可以包括:接收虚拟机针对冷页面发送的内存释放请求,其中,冷页面为虚拟机从至少一个虚拟内存页面中识别,且在预设时长内被访问的频率小于第一预设频率阈值的页面;从物理地址空间中释放冷页面对应的物理内存页面,并在第二页面表中解除冷页面对应的虚拟化物理内存页面在物理地址空间上的地址映射。

[0078] 在一种实施方式中,内存管理方法400还可以包括:接收虚拟机针对热页面发送的页面整合请求,其中,热页面为虚拟机从至少一个虚拟内存页面中识别,且在预设时长内被访问的频率大于第二预设频率阈值的页面;将热页面对应的物理内存页面整合为目标页面,并在第二页面表中更新整合后的目标页面在虚拟化物理地址空间上的地址映射,目标页面的尺寸大于尺寸阈值。

[0079] 其中,内存管理方法400中的各步骤的实施方式以及相对应的技术效果可以参见上述宿主机200和内存管理方法300中的对应描述,在此不再赘述。

[0080] 下面介绍本申请实施例的内存管理方法的示例性应用场景。

[0081] 数据库系统包括Runc模式和Rund模式。其中,Runc模式是在Host侧创建和运行轻量级容器,Rund模式是在Host侧创建和运行彼此强隔离的安全容器。Rund模式适用于多租户云场景下为租户提供与裸机几乎相同的安全隔离。

[0082] 在Rund模式下,为了更好实现弹性扩缩容,需要针对容器进行扩缩容。扩容时,需要为容器分配更多的内存和资源。如果内存碎片化严重,即使系统有足够的总内存量,扩容操作也可能会因无法分配到所需的连续内存块而失败。缩容时,容器试图释放部分内存,但如果这些内存碎片被系统的其他部分占用或锁定,内存可能无法有效释放,导致缩容失败或效率低下。这会浪费内存资源,使得其他容器或应用程序无法利用这些内存。

[0083] 而根据本申请实施例提供的技术方案,在Guest侧的虚拟化物理地址空间中固定设置一个预留内存区,后续内核数据的虚拟内存页面均从该预留内存区中分配,如此便可以避免内核数据内存分配的随机性,将内核数据所占用的内存集中在固定区域,从而减少内存碎片的产生,使内存资源可以更容易地被回收和重新分配。对于扩容操作,系统能够更有效地分配连续内存;对于缩容操作,系统能够更容易地释放内存。通过减少内核内存碎片化,容器在需要扩容时能够迅速获取所需的资源,在缩容时能够快速释放资源,从而提升整体弹性。这使得容器能够更灵活地响应负载变化,可以在更大的负载范围内正常运行,减少了扩缩容操作失败的风险。

[0084] 需要说明的是,本申请实施例中提供的上述应用场景或应用示例是为了便于理解,本申请实施例对技术方案的应用不作具体限定。

[0085] 此外,本申请所涉及的用户信息(包括但不限于用户设备信息、用户个人信息等)

和数据(包括但不限于用于分析的数据、存储的数据、展示的数据等),均为经用户授权或者经过各方充分授权的信息和数据,并且相关数据的收集、使用和处理需要遵守相关国家和地区的相关法律法规和标准,并提供有相应的操作入口,供用户选择授权或者拒绝。

[0086] 与本申请实施例提供的内存管理方法300相对应地,本申请实施例还提供一种内存管理装置,应用于虚拟机,该装置包括:虚拟内存页面分配模块,用于响应于检测到所述虚拟机的内核数据的内存分配请求,为所述内核数据分配与预留内存区具有地址映射关系的虚拟内存页面,其中,所述预留内存区位于所述虚拟机的虚拟化物理地址空间,且具有连续的虚拟化物理地址,所述虚拟化物理地址空间为所述虚拟机的宿主机提供的虚拟内存;第一页面表更新模块,用于在第一页面表中更新所述虚拟内存页面在所述虚拟化物理地址空间上的地址映射,其中,所述虚拟内存页面位于所述虚拟机的虚拟地址空间,所述第一页面表用于表示所述虚拟地址空间与所述虚拟化物理地址空间之间的地址映射关系。

[0087] 在一种实施方式中,该装置还可以包括:内存释放请求发送模块,用于响应于在至少一个所述虚拟内存页面中识别到冷页面,向所述宿主机发送针对所述冷页面的内存释放请求,所述内存释放请求用于请求所述宿主机从所述宿主机的物理地址空间中释放所述冷页面对应的物理内存页面,其中,所述冷页面为在预设时长内被访问的频率小于第一预设频率阈值的页面;虚拟化物理内存页面释放模块,用于从所述预留内存区中释放所述冷页面对应的虚拟化物理内存页面,并在所述第一页面表中解除所述冷页面在所述虚拟化物理地址空间上的地址映射。

[0088] 在一种实施方式中,该装置还可以包括:页面整合请求发送模块,用于响应于在至少一个所述虚拟内存页面中识别到热页面,向所述宿主机发送页面整合请求,所述页面整合请求用于请求所述宿主机将所述热页面对应的物理内存页面整合为目标页面,其中,所述热页面为在预设时长内被访问的频率大于第二预设频率阈值的页面,所述目标页面的尺寸大于尺寸阈值。

[0089] 在一种实施方式中,所述预留内存区包括多个具有连续虚拟化物理地址的子区域,所述子区域的大小与目标页面的大小相同,所述目标页面的尺寸大于尺寸阈值,虚拟内存页面分配模块还用于从多个所述子区域中选择至少一个所述子区域分配给所述虚拟内存页面。

[0090] 在一种实施方式中,所述虚拟内存页面对应连续的虚拟化物理地址。

[0091] 与本申请实施例提供的内存管理方法400相对应地,本申请实施例还提供一种内存管理装置,应用于宿主机,该装置包括:虚拟化物理地址空间分配模块,用于从所述宿主机的物理地址空间中为虚拟机分配虚拟化物理地址空间,其中,所述虚拟化物理地址空间中包括具有连续虚拟化物理地址的预留内存区,所述虚拟机用于为所述虚拟机的内核数据分配与预留内存区具有地址映射关系的虚拟内存页面;第二页面表表示模块,用于利用第二页面表表示所述物理地址空间与所述虚拟化物理地址空间之间的地址映射关系。

[0092] 在一种实施方式中,该装置还可以包括:内存释放请求接收模块,用于接收所述虚拟机针对冷页面发送的内存释放请求,其中,所述冷页面为所述虚拟机从至少一个所述虚拟内存页面中识别,且在预设时长内被访问的频率小于第一预设频率阈值的页面;物理内存页面释放模块,用于从所述物理地址空间中释放所述冷页面对应的物理内存页面,并在所述第二页面表中解除所述冷页面对应的虚拟化物理内存页面在所述物理地址空间上的

地址映射。

[0093] 在一种实施方式中,该装置还可以包括:页面整合请求接收模块,用于接收所述虚拟机针对热页面发送的页面整合请求,其中,所述热页面为所述虚拟机从至少一个所述虚拟内存页面中识别,且在预设时长内被访问的频率大于第二预设频率阈值的页面;页面整合模块,用于将所述热页面对应的物理内存页面整合为目标页面,并在所述第二页面表中更新整合后的目标页面在所述虚拟化物理地址空间上的地址映射,所述目标页面的尺寸大于尺寸阈值。

本申请实施例各装置中的各模块的功能可以参见上述方法中的对应描述,并具备相应的有益效果,在此不再赘述。

[0094] 图5为用来实现本申请实施例的电子设备的框图。如图5所示,该电子设备包括:存储器501和处理器502,存储器501内存储有可在处理器502上运行的计算机程序。处理器502执行该计算机程序时实现上述实施例中的方法。存储器501和处理器502的数量可以为一个或多个。在具体实现上,该电子设备还可以包括通信接口503,用于与外界设备进行通信,进行数据交互传输。

[0095] 在具体实现上,如果存储器501、处理器502和通信接口503独立实现,则存储器501、处理器502和通信接口503可以通过总线相互连接并完成相互间的通信。该总线可以是工业标准体系结构(Industry Standard Architecture,ISA)总线、外部设备互连(Peripheral Component Interconnect,PCI)总线或扩展工业标准体系结构(Extended Industry Standard Architecture,EISA)总线等。该总线可以分为地址总线、数据总线、控制总线等。为便于表示,图5中仅用一条粗线表示,但并不表示仅有一根总线或一种类型的总线。

[0096] 可选地,在具体实现上,如果存储器501、处理器502及通信接口503集成在一块芯片上,则存储器501、处理器502及通信接口503可以通过内部接口完成相互间的通信。

[0097] 本申请实施例提供了一种计算机可读存储介质,其存储有计算机程序,该程序被处理器执行时实现本申请实施例中提供的方法。

[0098] 本申请实施例提供了一种计算机程序产品,包括计算机程序,该程序被处理器执行时实现本申请实施例中提供的方法。

[0099] 本申请实施例还提供了一种芯片,该芯片包括处理器,用于从存储器中调用并运行存储器中存储的指令,使得安装有芯片的通信设备执行本申请实施例提供的方法。

[0100] 本申请实施例还提供了一种芯片,包括:输入接口、输出接口、处理器和存储器,输入接口、输出接口、处理器以及存储器之间通过内部连接通路相连,处理器用于执行存储器中的代码,当代码被执行时,处理器用于执行申请实施例提供的方法。

[0101] 应理解的是,上述处理器可以是中央处理器(Central Processing Unit,CPU),还可以是其他通用处理器、数字信号处理器(Digital Signal Processor,DSP)、专用集成电路(Application Specific Integrated Circuit,ASIC)、现场可编程门阵列(Field Programmable Gate Array,FPGA)或者其他可编程逻辑器件、分立门或者晶体管逻辑器件、分立硬件组件等。通用处理器可以是微处理器或者是任何常规的处理器等。值得说明的是,处理器可以是支持进阶精简指令集机器(Advanced RISC Machines,ARM)架构的处理器。

[0102] 进一步地,可选地,上述存储器可以包括只读存储器和随机访问存储器。该存储器

可以是易失性存储器或非易失性存储器,或可包括易失性和非易失性存储器两者。其中,非易失性存储器可以包括只读存储器(Read-Only Memory,ROM)、可编程只读存储器(Programmable ROM,PROM)、可擦除可编程只读存储器(Erasable PROM,EPROM)、电可擦除可编程只读存储器(Electrically EPROM,EEPROM)或闪存。易失性存储器可以包括随机访问存储器(Random Access Memory,RAM),其用作外部高速缓存。通过示例性但不是限制性说明,许多形式的RAM均可用。例如,静态随机访问存储器(Static RAM,SRAM)、动态随机访问存储器(Dynamic Random Access Memory,DRAM)、同步动态随机访问存储器(Synchronous DRAM,SDRAM)、双倍数据速率同步动态随机访问存储器(Double Data Rate SDRAM,DDR SDRAM)、增强型同步动态随机访问存储器(Enhanced SDRAM,ESDRAM)、同步链接动态随机访问存储器(Sync link DRAM,SLDRAM)和直接内存总线随机访问存储器(Direct Rambus RAM,DR RAM)。

[0103] 在上述实施例中,可以全部或部分地通过软件、硬件、固件或者其任意组合来实现。当使用软件实现时,可以全部或部分地以计算机程序产品的形式实现。计算机程序产品包括一个或多个计算机指令。在计算机上加载和执行计算机程序指令时,全部或部分地产生依照本申请的流程或功能。计算机可以是通用计算机、专用计算机、计算机网络、或者其他可编程装置。计算机指令可以存储在计算机可读存储介质中,或者从一个计算机可读存储介质向另一个计算机可读存储介质传输。

[0104] 在本说明书的描述中,参考术语“一个实施例”、“一些实施例”、“示例”、“具体示例”、或“一些示例”等的描述意指结合该实施例或示例描述的具体特征、结构、材料或者特点包括于本申请的至少一个实施例或示例中。而且,描述的具体特征、结构、材料或者特点可以在任一个或多个实施例或示例中以合适的方式结合。此外,在不相互矛盾的情况下,本领域的技术人员可以将本说明书中描述的不同实施例或示例以及不同实施例或示例的特征进行结合和组合。

[0105] 此外,术语“第一”、“第二”仅用于描述目的,而不能理解为指示或暗示相对重要性或者隐含指明所指示的技术特征的数量。由此,限定有“第一”、“第二”的特征可以明示或隐含地包括至少一个该特征。在本申请的描述中,“多个”的含义是两个或两个以上,除非另有明确具体的限定。

[0106] 流程图中描述的或在此以其他方式描述的任何过程或方法可以被理解为,表示包括一个或更多个用于实现特定逻辑功能或过程的步骤的可执行指令的代码的模块、片段或部分。并且本申请的优选实施方式的范围包括另外的实现,其中可以不按所示出或讨论的顺序,包括根据所涉及的功能按基本同时的方式或按相反的顺序,来执行功能。

[0107] 在流程图中描述的或在此以其他方式描述的逻辑和/或步骤,例如,可以被认为是用于实现逻辑功能的可执行指令的定序列表,可以具体实现在任何计算机可读介质中,以供指令执行系统、装置或设备(如基于计算机的系统、包括处理器的系统或其他可以从指令执行系统、装置或设备取指令并执行指令的系统)使用,或结合这些指令执行系统、装置或设备而使用。

[0108] 应理解的是,本申请的各部分可以用硬件、软件、固件或它们的组合来实现。在上述实施方式中,多个步骤或方法可以用存储在存储器中且由合适的指令执行系统执行的软件或固件来实现。上述实施例方法的全部或部分步骤是可以通过程序来指令相关的硬件完

成,该程序可以存储于一种计算机可读存储介质中,该程序在执行时,包括方法实施例的步骤之一或其组合。

[0109] 此外,在本申请各个实施例中的各功能单元可以集成在一个处理模块中,也可以是各个单元单独物理存在,也可以两个或两个以上单元集成在一个模块中。上述集成的模块既可以采用硬件的形式实现,也可以采用软件功能模块的形式实现。上述集成的模块如果以软件功能模块的形式实现并作为独立的产品销售或使用时,也可以存储在一个计算机可读存储介质中。该存储介质可以是只读存储器,磁盘或光盘等。

[0110] 以上所述,仅为本申请的示例性实施方式,但本申请的保护范围并不局限于此,任何熟悉本技术领域的技术人员在本申请记载的技术范围内,可轻易想到其各种变化或替换,这些都应涵盖在本申请的保护范围之内。因此,本申请的保护范围应以权利要求的保护范围为准。

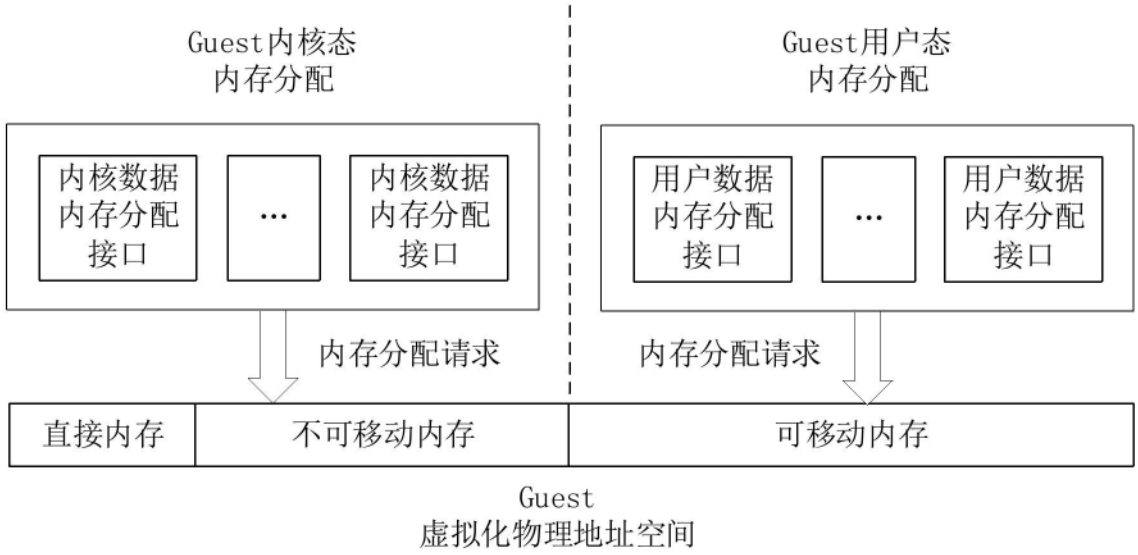

Guest内核态
内存分配
Guest用户态
内存分配
内核数据
内存分配
接口
...
内核数据
内存分配
接口
用户数据
内存分配
接口
...
用户数据
内存分配
接口
内存分配请求
内存分配请求
直接内存
不可移动内存
可移动内存
Guest
虚拟化物理地址空间

图1

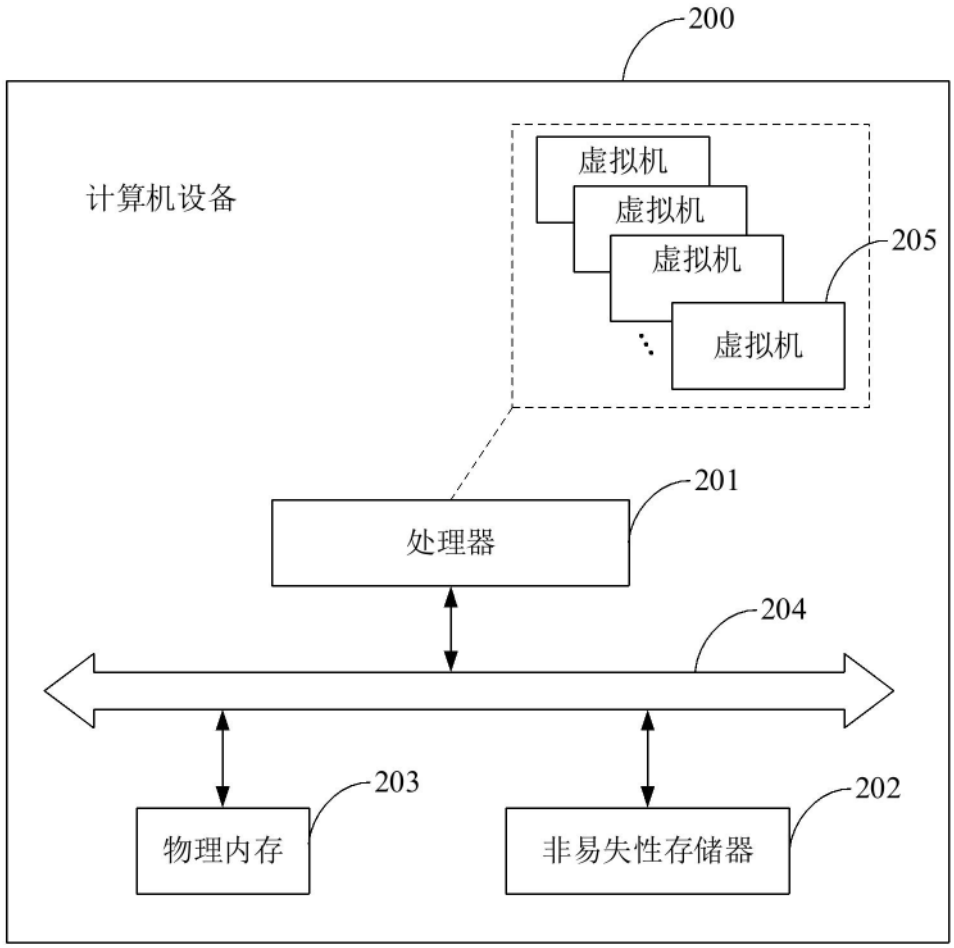

200
计算机设备
虚拟机
虚拟机
虚拟机
虚拟机
205
201
处理器
204
203
物理内存
202
非易失性存储器

图2A

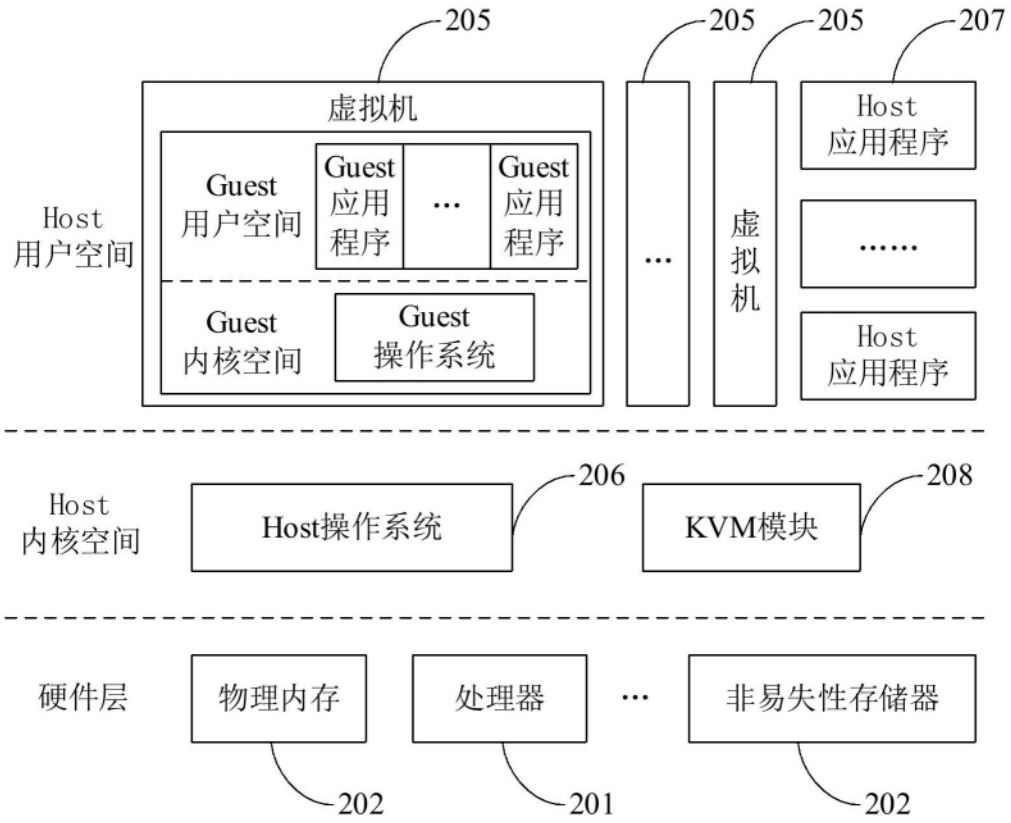

205
205
205
207
虚拟机
Host
应用程序
Guest
用户空间
Guest
应用
程序
...
Guest
应用
程序
Host
用户空间
...
虚拟机
.....
Guest
内核空间
Guest
操作系统
Host
应用程序
Host
内核空间
206
Host操作系统
208
KVM模块
硬件层
物理内存
处理器
...
非易失性存储器
202
201
202

图2B

210

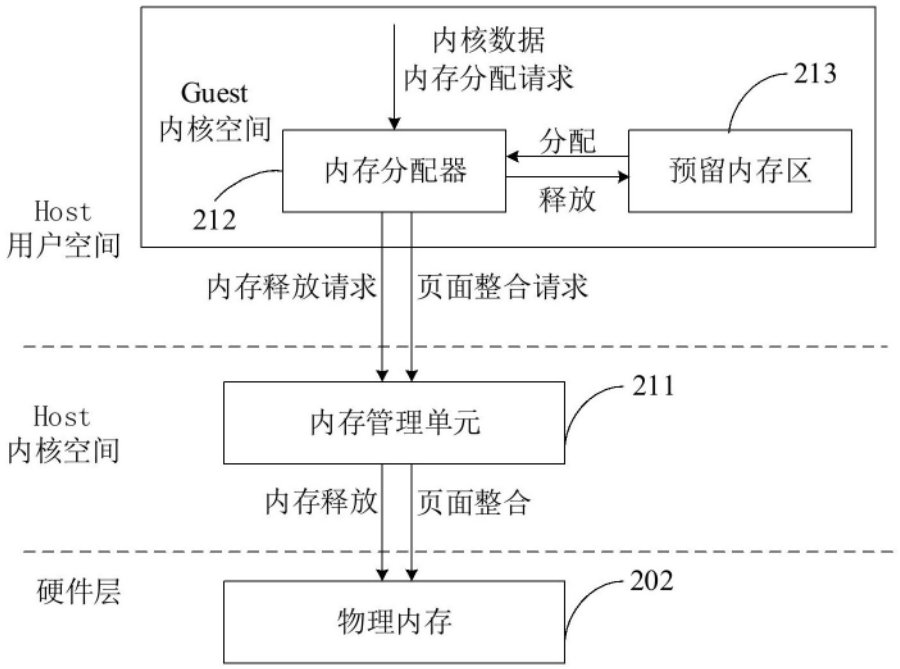

内核数据
内存分配请求
213
Guest
内核空间
分配
内存分配器
预留内存区
释放
212
Host
用户空间
内存释放请求
页面整合请求
Host
内核空间
211
内存管理单元
内存释放
页面整合
硬件层
202
物理内存

图2C

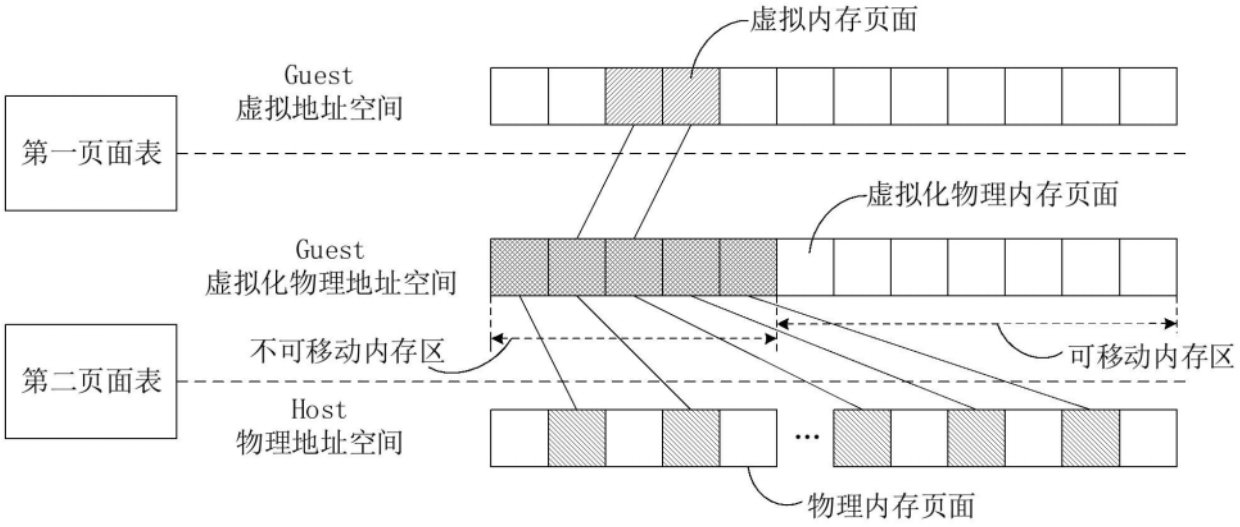

虚拟内存页面
Guest
虚拟地址空间
第一页面表
虚拟化物理内存页面
Guest
虚拟化物理地址空间
不可移动内存区
可移动内存区
第二页面表
Host
物理地址空间
物理内存页面

图2D

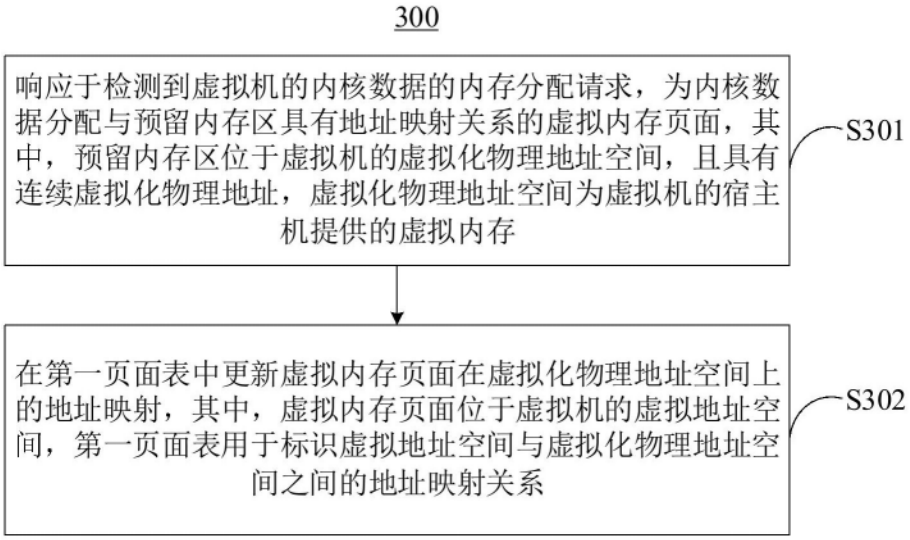

300
响应于检测到虚拟机的内核数据的内存分配请求，为内核数据分配与预留内存区具有地址映射关系的虚拟内存页面，其中，预留内存区位于虚拟机的虚拟化物理地址空间，且具有连续虚拟化物理地址，虚拟化物理地址空间为虚拟机的宿主机提供的虚拟内存
S301
在第一页面表中更新虚拟内存页面在虚拟化物理地址空间上的地址映射，其中，虚拟内存页面位于虚拟机的虚拟地址空间，第一页面表用于标识虚拟地址空间与虚拟化物理地址空间之间的地址映射关系
S302

图3

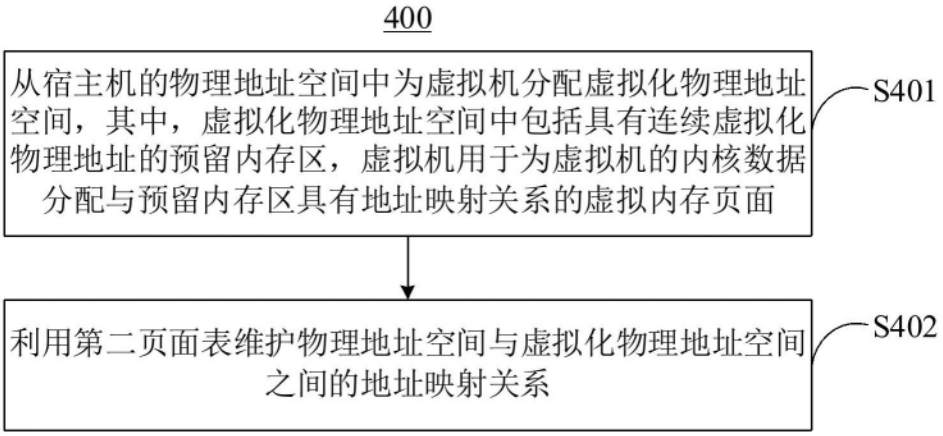

400
从宿主机的物理地址空间中为虚拟机分配虚拟化物理地址空间，其中，虚拟化物理地址空间中包括具有连续虚拟化物理地址的预留内存区，虚拟机用于为虚拟机的内核数据分配与预留内存区具有地址映射关系的虚拟内存页面
S401
利用第二页面表维护物理地址空间与虚拟化物理地址空间之间的地址映射关系
S402

图4

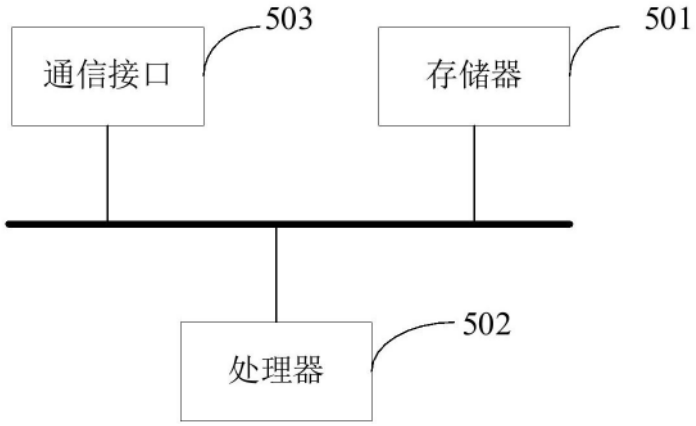

503
通信接口
501
存储器
502
处理器

图5