



(12) 发明专利

(10) 授权公告号 CN 119201190 B

(45) 授权公告日 2025.06.13

(21) 申请号 202411376852.0

(22) 申请日 2024.09.29

(65) 同一申请的已公布的文献号
申请公布号 CN 119201190 A

(43) 申请公布日 2024.12.27

(73) 专利权人 北京火山引擎科技有限公司
地址 100190 北京市海淀区紫金数码园4号
楼13层1309

(72) 发明人 冯志敏

(74) 专利代理机构 上海光栅知识产权代理有限公司 31340
专利代理师 刘文娜 关浩

(51) Int. Cl.
G06F 8/656 (2018.01)
G06F 9/455 (2018.01)

(56) 对比文件

CN 103793288 B, 2017.07.18

CN 113296802 A, 2021.08.24

审查员 张帆

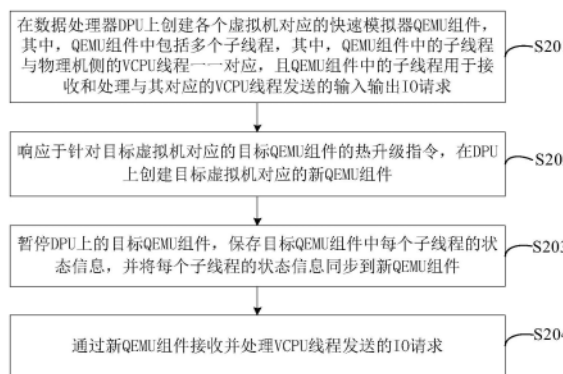
权利要求书2页 说明书11页 附图4页

(54) 发明名称

虚拟机的热升级方法及设备

(57) 摘要

本公开实施例提供一种虚拟机的热升级方法及设备,该方法包括:在数据处理器DPU上创建各个虚拟机对应的QEMU组件,其中,所述QEMU组件中的子线程与物理机侧的VCPU线程一一对应,且用于接收和处理VCPU线程发送的IO请求;响应于针对目标虚拟机对应的目标QEMU组件的热升级指令,在DPU上创建目标虚拟机对应的新的QEMU组件;暂停DPU上的目标QEMU组件,保存目标QEMU组件中每个子线程的状态信息,并将每个子线程的状态信息同步到新QEMU组件;通过新QEMU组件接收并处理VCPU线程发送的IO请求。该方法可以减少虚拟机热升级过程中的宕机时间。



1. 一种虚拟机的热升级方法,其特征在于,包括:

在数据处理器DPU上创建各个虚拟机对应的快速模拟器QEMU组件,其中,所述QEMU组件中包括多个子线程,其中,所述QEMU组件中的子线程与物理机侧的VCPU线程一一对应,且所述QEMU组件中的子线程用于接收和处理与其对应的VCPU线程发送的输入输出IO请求;

响应于针对目标虚拟机对应的目标QEMU组件的热升级指令,在DPU上创建所述目标虚拟机对应的新QEMU组件;

暂停所述DPU上的所述目标QEMU组件,保存所述目标QEMU组件中每个子线程的状态信息,并将每个子线程的状态信息同步到所述新QEMU组件,其中,在暂停所述DPU上的所述目标QEMU组件时,所述目标虚拟机的VCPU线程处于正常运行状态;

通过所述新QEMU组件接收并处理所述VCPU线程发送的IO请求。

2. 根据权利要求1所述的方法,其特征在于,所述目标QEMU组件包括多个目标子线程;所述目标子线程对应物理机上的一个目标VCPU线程;所述方法还包括:

响应于针对目标虚拟机对应的目标QEMU组件的热升级指令,继续运行所述物理机上的目标VCPU线程。

3. 根据权利要求1所述的方法,其特征在于,所述通过所述新QEMU组件接收并处理所述VCPU线程发送的IO请求之后,所述方法还包括:

关闭所述DPU上的所述目标QEMU组件。

4. 根据权利要求1所述的方法,其特征在于,所述物理机上部署有虚拟机对应内核虚拟机KVM组件,所述虚拟机在物理机上部署的多个VCPU线程通过所述KVM组件来处理业务请求;相应地,所述方法还包括:

响应于针对目标虚拟机对应的目标KVM组件的热升级指令,在DPU上创建热升级线程,其中所述目标KVM组件对应所述目标虚拟机在物理机上部署的多个目标VCPU线程;

通过调用所述热升级线程在所述物理机上创建每个目标VCPU线程对应的新VCPU线程;

针对每个目标VCPU线程,通过所述目标VCPU线程以及与其对应的新VCPU线程,保存和恢复所述目标VCPU线程的状态信息。

5. 根据权利要求4所述的方法,其特征在于,所述针对每个目标VCPU线程,通过所述目标VCPU线程以及与其对应的新VCPU线程,保存和恢复所述目标VCPU线程的状态信息,包括:

针对每个目标VCPU线程,通过所述目标VCPU线程保存所述目标VCPU线程的状态信息;

将所述目标VCPU线程的状态信息同步到与其对应的新VCPU线程。

6. 根据权利要求5所述的方法,其特征在于,所述将所述目标VCPU线程的状态信息同步到与其对应的新VCPU线程,包括:

若所述目标VCPU线程保存所述目标VCPU线程的状态信息完成,向所述热升级线程反馈保存完成信息;

通过所述热升级线程向所述目标VCPU线程对应的新VCPU线程发送信息加载指令,通过所述新VCPU线程加载所述目标VCPU线程的状态信息。

7. 根据权利要求4所述的方法,其特征在于,所述针对每个目标VCPU线程,通过所述目标VCPU线程以及与其对应的新VCPU线程,保存和恢复所述目标VCPU线程的状态信息之前,所述方法还包括:

通过调用所述热升级线程暂停所述目标虚拟机在物理机上部署的多个目标VCPU线程。

8. 根据权利要求4所述的方法,其特征在于,所述针对每个目标VCPU线程,通过所述目标VCPU线程以及与其对应的新VCPU线程,保存和恢复所述目标VCPU线程的状态信息之后,所述方法还包括:

通过所述每个目标VCPU线程对应的新VCPU线程处理所述目标虚拟机的业务请求;
关闭所述目标虚拟机在物理机上部署的多个目标VCPU线程。

9. 一种虚拟机的热升级设备,其特征在于,包括:

创建单元,用于在数据处理器DPU上创建各个虚拟机对应的快速模拟器QEMU组件,其中,所述QEMU组件中包括多个子线程,其中,所述QEMU组件中的子线程与物理机侧的VCPU线程一一对应,且所述QEMU组件中的子线程用于接收和处理与其对应的VCPU线程发送的输入输出IO请求;

热升级单元,用于响应于针对目标虚拟机对应的目标QEMU组件的热升级指令,在DPU上创建所述目标虚拟机对应的新QEMU组件;

同步单元,用于暂停所述DPU上的所述目标QEMU组件,保存所述目标QEMU组件中每个子线程的状态信息,并将每个子线程的状态信息同步到所述新QEMU组件,其中,在暂停所述DPU上的所述目标QEMU组件时,所述目标虚拟机的VCPU线程处于正常运行状态;

处理单元,用于通过所述新QEMU组件接收并处理所述VCPU线程发送的IO请求。

10. 一种电子设备,其特征在于,包括:处理器和存储器;

所述存储器存储计算机执行指令;

所述处理器执行所述存储器存储的计算机执行指令,使得所述处理器执行如权利要求1至8任一项所述的虚拟机的热升级方法。

11. 一种计算机可读存储介质,其特征在于,所述计算机可读存储介质中存储有计算机执行指令,当处理器执行所述计算机执行指令时,实现如权利要求1至8任一项所述的虚拟机的热升级方法。

12. 一种计算机程序产品,包括计算机程序,其特征在于,所述计算机程序被处理器执行时实现如权利要求1至8任一项所述的虚拟机的热升级方法。

虚拟机的热升级方法及设备

技术领域

[0001] 本公开实施例涉及虚拟机技术领域,尤其涉及一种虚拟机的热升级方法及设备。

背景技术

[0002] 在虚拟化场景下,虚拟机的功能迭代及缺陷修复需要发布新的组件版本,为了不影响运行中的虚拟机,通常会借助热升级能力将虚拟机的组件升级到最新版本。

[0003] 现有技术中,虚拟机的热升级过程类似虚拟机的热迁移,具体包括:使用新的KVM (Kernel-Based Virtual Machine的,内核虚拟机) 组件和QEMU (Quick Emulator,快速模拟器) 组件启动一个新虚拟机,然后暂停正在运行的虚拟机,将虚拟机所需要保证的状态同步到新的虚拟机,然后启动新的虚拟机运行,同时将老的虚拟机退出,这样无需重启虚拟机来实现QEMU组件、KVM组件的热升级。尽管热升级无需虚拟机停机和重启,但是也需要暂停虚拟机进行状态的保存和恢复操作,因此在热升级期间会存在短时间的虚拟机业务不可用状态,这段时间称为虚拟机热升级的宕机时间。

[0004] 发明人发现现有技术至少存在如下技术问题:通过现有技术对虚拟机进行热升级时,虚拟机的宕机时间较长。

发明内容

[0005] 本公开实施例提供一种虚拟机的热升级方法及设备,可以减少虚拟机热升级过程中的宕机时间。

[0006] 第一方面,本公开实施例提供一种虚拟机的热升级方法,包括:

[0007] 在数据处理器DPU上创建各个虚拟机对应的快速模拟器QEMU组件,其中,所述QEMU组件中包括多个子线程,其中,所述QEMU组件中的子线程与物理机侧的VCPU线程一一对应,且所述QEMU组件中的子线程用于接收和处理与其对应的VCPU线程发送的IO (Input/Output,输入输出) 请求;

[0008] 响应于针对目标虚拟机对应的目标QEMU组件的热升级指令,在DPU上创建所述目标虚拟机对应的新的QEMU组件;

[0009] 暂停所述DPU上的所述目标QEMU组件,保存所述目标QEMU组件中每个子线程的状态信息,并将每个子线程的状态信息同步到所述新的QEMU组件;

[0010] 通过所述新的QEMU组件接收并处理所述VCPU线程发送的IO请求。

[0011] 第二方面,本公开实施例提供一种虚拟机的热升级设备,包括:

[0012] 创建单元,用于在数据处理器DPU上创建各个虚拟机对应的快速模拟器QEMU组件,其中,所述QEMU组件中包括多个子线程,其中,所述QEMU组件中的子线程与物理机侧的VCPU线程一一对应,且所述QEMU组件中的子线程用于接收和处理与其对应的VCPU线程发送的输入输出IO请求;

[0013] 热升级单元,用于响应于针对目标虚拟机对应的目标QEMU组件的热升级指令,在DPU上创建所述目标虚拟机对应的新的QEMU组件;

[0014] 同步单元,用于暂停所述DPU上的所述目标QEMU组件,保存所述目标QEMU组件中每个子线程的状态信息,并将每个子线程的状态信息同步到所述新QEMU组件;

[0015] 处理单元,用于通过所述新QEMU组件接收并处理所述VCPU线程发送的IO请求。

[0016] 第三方面,本公开实施例提供一种电子设备,包括:处理器和存储器;

[0017] 所述存储器存储计算机执行指令;

[0018] 所述处理器执行所述存储器存储的计算机执行指令,使得所述至少一个处理器执行如上第一方面以及第一方面各种可能的设计所述的虚拟机的热升级方法。

[0019] 第四方面,本公开实施例提供一种计算机可读存储介质,所述计算机可读存储介质中存储有计算机执行指令,当处理器执行所述计算机执行指令时,实现如上第一方面以及第一方面各种可能的设计所述的虚拟机的热升级方法。

[0020] 第五方面,本公开实施例提供一种计算机程序产品,包括计算机程序,所述计算机程序被处理器执行时实现如上第一方面以及第一方面各种可能的设计所述的虚拟机的热升级方法。

[0021] 本实施例提供的虚拟机的热升级方法及设备,该方法包括:在数据处理器DPU上创建各个虚拟机对应的快速模拟器QEMU组件,其中,QEMU组件中包括多个子线程,其中,QEMU组件中的子线程与物理机侧的VCPU线程一一对应,且QEMU组件中的子线程用于接收和处理与其对应的VCPU线程发送的输入输出IO请求;响应于针对目标虚拟机对应的目标QEMU组件的热升级指令,在DPU上创建目标虚拟机对应的新的QEMU组件;暂停DPU上的目标QEMU组件,保存目标QEMU组件中每个子线程的状态信息,并将每个子线程的状态信息同步到新QEMU组件;通过新QEMU组件接收并处理VCPU线程发送的IO请求。该技术方案的,由于当升级QEMU组件时,只需要暂停运行在DPU卡上的QEMU组件,而无需暂停物理机上的VCPU线程,这样一来,升级QEMU组件时,虚拟机的VCPU线程是正常运行的,实现了QEMU组件热升级的零宕机时间,进而减少了虚拟机热升级过程中的宕机时间。

附图说明

[0022] 为了更清楚地说明本公开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一简单地介绍,显而易见地,下面描述中的附图是本公开的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动性的前提下,还可以根据这些附图获得其他的附图。

[0023] 图1为本公开实施例提供的虚拟机的热升级方法的应用场景示意图;

[0024] 图2为本公开实施例提供的虚拟机的热升级方法的流程图一;

[0025] 图3为本公开实施例提供的虚拟机的热升级方法的示意图一;

[0026] 图4为本公开实施例提供的虚拟机的热升级方法的流程图二;

[0027] 图5为本公开实施例提供的虚拟机的热升级方法的示意图二;

[0028] 图6为本公开实施例提供的虚拟机的热升级设备的结构示意图;

[0029] 图7为本公开实施例提供的电子设备的结构示意图。

具体实施方式

[0030] 为使本公开实施例的目的、技术方案和优点更加清楚,下面将结合本公开实施例

中的附图,对本公开实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例是本公开一部分实施例,而不是全部的实施例。基于本公开中的实施例,本领域普通技术人员在没有作出创造性劳动前提下所获得的所有其他实施例,都属于本公开保护的范围。

[0031] 为使本公开实施例的目的、技术方案和优点更加清楚,下面将结合本公开实施例中的附图,对本公开实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例是本公开一部分实施例,而不是全部的实施例。基于本公开中的实施例,本领域普通技术人员在没有作出创造性劳动前提下所获得的所有其他实施例,都属于本公开保护的范围。

[0032] 在虚拟化场景下,虚拟机的功能迭代及缺陷修复需要发布新的组件版本,为了不影响运行中的虚拟机,通常会借助热升级能力将虚拟机的组件升级到最新版本。在一些实施例中,虚拟化数据面组件包括QEMU组件和KVM组件。其中,QEMU组件是一个用户态程序,而KVM组件是内核模块。一个虚拟机对应一个QEMU组件,该QEMU组件包含了多个子线程,虚拟机中的每个虚拟处理器(VCPU)对应QEMU组件中的一个子线程。

[0033] 现有技术中,虚拟机的热升级过程类似虚拟机的热迁移,具体包括:使用新的KVM和QEMU启动一个新虚拟机,然后暂停正在运行的虚拟机,将虚拟机所需要保证的状态同步到新的虚拟机,然后启动新的虚拟机运行,同时将老的虚拟机退出,这样无需重启虚拟机来实现QEMU、KVM组件的热升级。尽管热升级无需虚拟机停机和重启,但是也需要暂停虚拟机进行状态的保存和恢复操作,因此在热升级期间会存在短时间的虚拟机业务不可用状态,这段时间称为虚拟机热升级的宕机时间。可以理解的是,虚拟机热升级的宕机时间越少,则对客户虚拟机的业务性能抖动影响越小。

[0034] 发明人发现现有技术至少存在如下技术问题:通过现有技术对虚拟机进行热升级时,虚拟机的宕机时间较长。

[0035] 由此可见,如何降低虚拟机热升级时虚拟机的宕机时间,以提高用户虚拟机的性能是目前亟待解决的一个技术问题。

[0036] 针对现有技术中的技术问题,发明人的技术构思如下:首先,在数据处理器DPU上创建各个虚拟机对应的快速模拟器QEMU组件,其中,QEMU组件中包括多个子线程,其中,QEMU组件中的子线程与物理机侧的VCPU线程一一对应,且QEMU组件中的子线程用于接收和处理与其对应的VCPU线程发送的输入输出IO请求。其次,响应于针对目标虚拟机对应的目标QEMU组件的热升级指令,在DPU上创建目标虚拟机对应的新的QEMU组件;最后,暂停DPU上的目标QEMU组件,保存目标QEMU组件中每个子线程的状态信息,并将每个子线程的状态信息同步到新QEMU组件;通过新QEMU组件接收并处理VCPU线程发送的IO请求。

[0037] 在该技术方案中,由于当升级QEMU组件时,只需要暂停运行在DPU卡上的QEMU组件,而无需暂停物理机上的VCPU线程,这样一来,升级QEMU组件时,虚拟机的VCPU线程是正常运行的,实现了QEMU组件热升级的零宕机时间,进而减少了虚拟机热升级过程中的宕机时间。

[0038] 下面对本公开实施例的应用场景进行解释:

[0039] 本公开实施例提供的虚拟机的热升级方法,可以应用于DPU场景中。图1为本公开实施例提供的一种虚拟机的热升级方法的应用场景示意图。如图1所示,利用DPU(Data Processing Unit,数据处理器)架构,将虚拟化组件中的QEMU组件卸载到DPU卡上运行,而KVM组件还是运行在物理机上。这样,基于DPU架构,可以实现QEMU组件、KVM组件独立热升级

能力。其中,QEMU组件包括子线程X和子线程Y。线程X对应该虚拟机在物理机上部署的VCPU线程X;线程Y对应该虚拟机在物理机上部署的VCPU线程Y。其中,QEMU组件中的子线程X用于接收和处理与其对应的VCPU线程X发送的IO请求。QEMU组件中的子线程Y用于接收和处理与其对应的VCPU线程Y发送的IO请求。

[0040] 当需要对虚拟机进行热升级时,可以通过本公开实施例提供的虚拟机的热升级方法,对QEMU组件、KVM组件进行热升级。

[0041] 下述为本公开实施例涉及的虚拟机的热升级方法及设备的具体实现过程,部分示例仅为举例,不作限定。本公开实施例涉及的虚拟机的热升级方法的执行主体为电子设备,可以是终端、服务器等。

[0042] 图2为本公开实施例提供的虚拟机的热升级方法流程图一,如图2所示,该虚拟机的热升级方法可以包括:

[0043] S201、在数据处理器DPU上创建各个虚拟机对应的快速模拟器QEMU组件,其中,QEMU组件中包括多个子线程,其中,QEMU组件中的子线程与物理机侧的VCPU线程一一对应,且QEMU组件中的子线程用于接收和处理与其对应的VCPU线程发送的输入输出IO请求。

[0044] 在本公开实施例中,由于QEMU组件运行在DPU卡上,而KVM组件部署在物理机上,因此在DPU架构下,物理机侧同样会存在一个VCPU线程来执行KVM代码,并且物理机侧的VCPU线程与QEMU组件中的子线程一一对应。

[0045] 需要说明的是,当虚拟机有需要QEMU组件介入处理的IO请求时,物理机上的VCPU线程才会将IO请求发送给DPU上对应的子线程来处理。在一些实施例中,在启动的时候会存在大量的IO请求需要QEMU组件处理;在虚拟机正常运行过程中,不存在退出到QEMU组件的IO请求。

[0046] 示例性的,虚拟机对应的QEMU组件包括子线程X和子线程Y。物理机侧的VCPU线程包括VCPU线程X和VCPU线程Y。其中,子线程X与VCPU线程X一一对应,子线程Y与VCPU线程Y一一对应。

[0047] S202、响应于针对目标虚拟机对应的目标QEMU组件的热升级指令,在DPU上创建目标虚拟机对应的新QEMU组件。

[0048] 在本公开实施例中,在DPU上创建的新QEMU组件中包括的多个子线程与目标QEMU组件中包括的多个子线程相同。

[0049] 示例性的,如图3所示,目标虚拟机对应的目标QEMU组件包括子线程X和子线程Y。新QEMU组件中包括子线程X和子线程Y。

[0050] S203、暂停DPU上的目标QEMU组件,保存目标QEMU组件中每个子线程的状态信息,并将每个子线程的状态信息同步到新QEMU组件。

[0051] 在本公开实施例中,在QEMU组件热升级过程中,首先需要暂停处理来自物理机侧的VCPU线程的IO请求。然后将老QEMU组件(也即目标QEMU组件)的设备状态进行保存并且发送给新QEMU组件。新QEMU组件将设备状态加载后开始接收物理机上的VCPU线程递交的IO请求。其中,老QEMU组件的设备状态包括各个子线程的状态信息。

[0052] 在一些实施例中,暂停DPU上的目标QEMU组件也即是停止处理IO请求。

[0053] 示例性的,如图3所示,目标虚拟机对应的目标QEMU组件包括子线程X和子线程Y。相应地,暂停DPU上的目标QEMU组件,包括:暂停子线程X处理IO请求,以及,停止子线程Y处

理IO请求。

[0054] 可选地,将每个子线程的状态信息同步到新QEMU组件的过程,也即是每个子线程的状态信息加载到新QEMU组件中的过程。

[0055] 需要说明的是,在暂停DPU上的目标QEMU组件时,无需暂停物理机上的VCPU线程。也即是,升级QEMU组件时,虚拟机的VCPU线程是正常运行的。

[0056] 在一些实施例中,目标QEMU组件包括多个目标子线程;目标子线程对应物理机上的一个目标VCPU线程;相应地,该方法还包括:响应于针对目标虚拟机对应的目标QEMU组件的热升级指令,继续运行物理机上的目标VCPU线程。

[0057] S204、通过新QEMU组件接收并处理VCPU线程发送的IO请求。

[0058] 在一些实施例中,如图3所示,本步骤包括:通过新QEMU组件中的子线程X接收并处理VCPU线程X发送的IO请求,以及,通过新QEMU组件中的子线程Y接收并处理VCPU线程Y发送的IO请求。

[0059] 在一些实施例中,如图3所示,通过新QEMU组件接收并处理VCPU线程发送的IO请求之后,该方法还包括:关闭DPU上的目标QEMU组件。

[0060] 本公开实施例提供了一种虚拟机的热升级方法:首先,在数据处理器DPU上创建各个虚拟机对应的快速模拟器QEMU组件,其中,所述QEMU组件中包括多个子线程,其中,所述QEMU组件中的子线程与物理机侧的VCPU线程一一对应,且所述QEMU组件中的子线程用于接收和处理与其对应的VCPU线程发送的输入输出IO请求。其次,响应于针对目标虚拟机对应的目标QEMU组件的热升级指令,在DPU上创建目标虚拟机对应的新的QEMU组件;最后,暂停DPU上的目标QEMU组件,保存目标QEMU组件中每个子线程的状态信息,并将每个子线程的状态信息同步到新QEMU组件;通过新QEMU组件接收并处理VCPU线程发送的IO请求。在该技术方案中,由于当升级QEMU组件时,只需要暂停运行在DPU卡上的QEMU组件,而无需暂停物理机上的VCPU线程,这样一来,升级QEMU组件时,虚拟机的VCPU线程是正常运行的,实现了QEMU组件热升级的零宕机时间,进而减少了虚拟机热升级过程中的宕机时间。

[0061] 在本公开实施例中,物理机上部署有虚拟机对应KVM组件,虚拟机在物理机上部署的多个VCPU线程通过KVM组件来处理业务请求。在此情况下,通过重构虚拟化VCPU架构,实现了虚拟化组件(QEMU组件、KVM组件)的独立热升级,在此基础上实现了消除热升级QEMU组件对虚拟机性能的影响。

[0062] 在一些实施例中,针对KVM组件热升级的宕机时间进行优化,可以通过并行方式的对VCPU状态进行保存和恢复,减少宕机时间。相应地,如图4所示,该方法还包括:

[0063] S401、响应于针对目标虚拟机对应的目标KVM组件的热升级指令,在DPU上创建热升级线程,其中目标KVM组件对应目标虚拟机在物理机上部署的多个目标VCPU线程。

[0064] 在本步骤中,如图5所示,创建的热升级线程只负责控制升级流程,虚拟机状态的保存和恢复都是由新旧VCPU线程各自完成,相对比传统的热升级方案(所有流程都是由升级线程串行执行完成),新的热升级方案对宕机时间的优化比较明显。其中,虚拟机状态包括各个VCPU线程的状态信息。

[0065] S402、通过调用热升级线程在物理机上创建每个目标VCPU线程对应的新的VCPU线程。

[0066] 在本步骤中,如图5所示,可以通过热升级线程初始化新的VCPU线程,创建每个目标

VCPU线程对应的新VCPU线程。其中,初始化新VCPU线程可以向物理机发送新VCPU线程的创建指令,物理机接收该创建指令,创建新VCPU线程。可以理解的是,创建的新VCPU线程与目标VCPU线程一一对应。

[0067] 可选地,如图5所示,针对每个目标VCPU线程,通过目标VCPU线程以及与其对应的新VCPU线程,保存和恢复目标VCPU线程的状态信息之前,该方法还包括:通过调用热升级线程暂停目标虚拟机在物理机上部署的多个目标VCPU线程。这样可以通过目标VCPU线程保存该目标VCPU线程的状态信息。

[0068] S403、针对每个目标VCPU线程,通过目标VCPU线程以及与其对应的新VCPU线程,保存和恢复目标VCPU线程的状态信息。

[0069] 在一些实施例中,可以通过目标VCPU线程(旧VCPU线程)保存VCPU线程的状态信息,通过创建的新VCPU线程加载VCPU线程的状态信息,通过新旧VCPU线程的升级,保存和恢复目标VCPU线程的状态信息。相应地,本步骤可以包括:针对每个目标VCPU线程,通过目标VCPU线程保存目标VCPU线程的状态信息;将目标VCPU线程的状态信息同步到与其对应的新VCPU线程。

[0070] 可选地,如图5所示,将目标VCPU线程的状态信息同步到与其对应的新VCPU线程的步骤可以包括:若目标VCPU线程保存目标VCPU线程的状态信息完成,向热升级线程反馈保存完成信息;通过热升级线程向目标VCPU线程对应的新VCPU线程发送信息加载指令,通过新VCPU线程加载目标VCPU线程的状态信息。

[0071] 需要说明的是,热升级线程可以向多个目标VCPU线程对应的新VCPU线程发送信息加载指令,此时,多个新VCPU线程加载目标VCPU线程的状态信息可以并行处理,也即是,通过并行方式的对VCPU状态进行保存和恢复,减少宕机时间。

[0072] 可选地,如图5所示,针对每个目标VCPU线程,通过目标VCPU线程以及与其对应的新VCPU线程,保存和恢复目标VCPU线程的状态信息之后,该方法还包括:通过每个目标VCPU线程对应的新VCPU线程处理目标虚拟机的业务请求;关闭目标虚拟机在物理机上部署的多个目标VCPU线程。

[0073] 在本公开实施例中,借助VCPU线程保存和恢复VCPU线程的热升级数据,提升数据的保存和恢复的并发度,降低大规格虚拟机热升级KVM组件时,对该虚拟机性能的影响,提高了用户体验。

[0074] 图6为本公开实施例提供的虚拟机的热升级设备的结构示意图,如图6所示,该虚拟机的热升级设备包括:

[0075] 创建单元601,用于在数据处理器DPU上创建各个虚拟机对应的快速模拟器QEMU组件,其中,所述QEMU组件中包括多个子线程,其中,所述QEMU组件中的子线程与物理机侧的VCPU线程一一对应,且所述QEMU组件中的子线程用于接收和处理与其对应的VCPU线程发送的输入输出IO请求;

[0076] 热升级单元602,用于响应于针对目标虚拟机对应的目标QEMU组件的热升级指令,在DPU上创建目标虚拟机对应的新QEMU组件;

[0077] 同步单元603,用于暂停DPU上的目标QEMU组件,保存目标QEMU组件中每个子线程的状态信息,并将每个子线程的状态信息同步到新QEMU组件;

[0078] 处理单元604,用于通过新QEMU组件接收并处理VCPU线程发送的IO请求。。

[0079] 根据本公开的一个或多个实施例,所述目标QEMU组件包括多个目标子线程;所述目标子线程对应物理机上的一个目标VCPU线程;所述热升级单元602,还用于响应于针对目标虚拟机对应的目标QEMU组件的热升级指令,继续运行所述物理机上的目标VCPU线程。

[0080] 根据本公开的一个或多个实施例,所述同步单元603,还用于关闭所述DPU上的所述目标QEMU组件。

[0081] 根据本公开的一个或多个实施例,所述物理机上部署有虚拟机对应KVM组件,所述虚拟机在物理机上部署的多个VCPU线程通过所述KVM组件来处理业务请求;相应地,所述设备还包括:保存单元,所述保存单元,用于响应于针对目标虚拟机对应的目标KVM组件的热升级指令,在DPU上创建热升级线程,其中所述目标KVM组件对应所述目标虚拟机在物理机上部署的多个目标VCPU线程;通过调用所述热升级线程在所述物理机上创建每个目标VCPU线程对应的新VCPU线程;针对每个目标VCPU线程,通过所述目标VCPU线程以及与其对应的新VCPU线程,保存和恢复所述目标VCPU线程的状态信息。

[0082] 根据本公开的一个或多个实施例,所述保存单元,针对每个目标VCPU线程,通过所述目标VCPU线程以及与其对应的新VCPU线程,保存和恢复所述目标VCPU线程的状态信息,包括:针对每个目标VCPU线程,通过所述目标VCPU线程保存所述目标VCPU线程的状态信息;将所述目标VCPU线程的状态信息同步到与其对应的新VCPU线程。

[0083] 根据本公开的一个或多个实施例,所述保存单元,将所述目标VCPU线程的状态信息同步到与其对应的新VCPU线程,包括:若所述目标VCPU线程保存所述目标VCPU线程的状态信息完成,向所述热升级线程反馈保存完成信息;通过所述热升级线程向所述目标VCPU线程对应的新VCPU线程发送信息加载指令,通过所述新VCPU线程加载所述目标VCPU线程的状态信息。

[0084] 根据本公开的一个或多个实施例,所述保存单元,还用于通过调用所述热升级线程暂停所述目标虚拟机在物理机上部署的多个目标VCPU线程。

[0085] 根据本公开的一个或多个实施例,所述处理单元604,还用于通过所述每个目标VCPU线程对应的新VCPU线程处理所述目标虚拟机的业务请求;关闭所述目标虚拟机在物理机上部署的多个目标VCPU线程。

[0086] 参考图7,其示出了适于用来实现本公开实施例的电子设备700的结构示意图,该电子设备700可以为终端设备或服务器。其中,终端设备可以包括但不限于诸如移动电话、笔记本电脑、数字广播接收器、个人数字助理(Personal Digital Assistant,简称PDA)、平板电脑(Tablet Computer)、便携式多媒体播放器(Portable Media Player,简称PMP)、车载终端(例如车载导航终端)等等的移动终端以及诸如数字TV、台式计算机等等的固定终端。图7示出的电子设备仅仅是一个示例,不应对本公开实施例的功能和使用范围带来任何限制。

[0087] 如图7所示,电子设备700可以包括处理装置(例如中央处理器、图形处理器等)701,其可以根据存储在只读存储器(Read Only Memory,简称ROM)702中的程序或者从存储装置708加载到随机访问存储器(Random Access Memory,简称RAM)703中的程序而执行各种适当的动作和处理。在RAM 703中,还存储有电子设备700操作所需的各种程序和数据。处理装置701、ROM 702以及RAM 703通过总线704彼此相连。输入/输出(I/O)接口705也连接至总线704。

[0088] 通常,以下装置可以连接至I/O接口705:包括例如触摸屏、触摸板、键盘、鼠标、摄像头、麦克风、加速度计、陀螺仪等的输入装置706;包括例如液晶显示器(Liquid Crystal Display,简称LCD)、扬声器、振动器等输出装置707;包括例如磁带、硬盘等的存储装置708;以及通信装置709。通信装置709可以允许电子设备700与其他设备进行无线或有线通信以交换数据。虽然图7示出了具有各种装置的电子设备700,但是应理解的是,并不要求实施或具备所有示出的装置。可以替代地实施或具备更多或更少的装置。

[0089] 特别地,根据本公开的实施例,上文参考流程图描述的过程可以被实现为计算机软件程序。例如,本公开的实施例包括一种计算机程序产品,其包括承载在计算机可读介质上的计算机程序,该计算机程序包含用于执行流程图所示的方法的程序代码。在这样的实施例中,该计算机程序可以通过通信装置709从网络上被下载和安装,或者从存储装置708被安装,或者从ROM 702被安装。在该计算机程序被处理装置701执行时,执行本公开实施例的方法中限定的上述功能。

[0090] 需要说明的是,本公开上述的计算机可读介质可以是计算机可读信号介质或者计算机可读存储介质或者是上述两者的任意组合。计算机可读存储介质例如可以是一——但不限于——电、磁、光、电磁、红外线、或半导体的系统、装置或器件,或者任意以上的组合。计算机可读存储介质的更具体的例子可以包括但不限于:具有一个或多个导线的电连接、便携式计算机磁盘、硬盘、随机访问存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPROM或闪存)、光纤、便携式紧凑磁盘只读存储器(CD-ROM)、光存储器件、磁存储器件、或者上述的任意合适的组合。在本公开中,计算机可读存储介质可以是任何包含或存储程序的有形介质,该程序可以被指令执行系统、装置或者器件使用或者与其结合使用。而在本公开中,计算机可读信号介质可以包括在基带中或者作为载波一部分传播的数据信号,其中承载了计算机可读的程序代码。这种传播的数据信号可以采用多种形式,包括但不限于电磁信号、光信号或上述的任意合适的组合。计算机可读信号介质还可以是计算机可读存储介质以外的任何计算机可读介质,该计算机可读信号介质可以发送、传播或者传输用于由指令执行系统、装置或者器件使用或者与其结合使用的程序。计算机可读介质上包含的程序代码可以用任何适当的介质传输,包括但不限于:电线、光缆、RF(射频)等等,或者上述的任意合适的组合。

[0091] 上述计算机可读介质可以是上述电子设备中所包含的;也可以是单独存在,而未装配入该电子设备中。

[0092] 上述计算机可读介质承载有一个或者多个程序,当上述一个或者多个程序被该电子设备执行时,使得该电子设备执行上述实施例所示的方法。

[0093] 可以以一种或多种程序设计语言或其组合来编写用于执行本公开的操作的计算机程序代码,上述程序设计语言包括面向对象的程序设计语言——诸如Java、Smalltalk、C++,还包括常规的过程式程序设计语言——诸如“C”语言或类似的设计语言。程序代码可以完全地在用户计算机上执行、部分地在用户计算机上执行、作为一个独立的软件包执行、部分在用户计算机上部分在远程计算机上执行、或者完全在远程计算机或服务器上执行。在涉及远程计算机的情形中,远程计算机可以通过任意种类的网络——包括局域网(Local Area Network,简称LAN)或广域网(Wide Area Network,简称WAN)——连接到用户计算机,或者,可以连接到外部计算机(例如利用因特网服务提供商来通过因特网连接)。

[0094] 附图中的流程图和框图,图示了按照本公开各种实施例的系统、方法和计算机程序产品的可能实现的体系架构、功能和操作。在这点上,流程图或框图中的每个方框可以代表一个模块、程序段、或代码的一部分,该模块、程序段、或代码的一部分包含一个或多个用于实现规定的逻辑功能的可执行指令。也应当注意,在有些作为替换的实现中,方框中所标注的功能也可以以不同于附图中所标注的顺序发生。例如,两个接连地表示的方框实际上可以基本并行地执行,它们有时也可以按相反的顺序执行,这依所涉及的功能而定。也要注意,框图和/或流程图中的每个方框、以及框图和/或流程图中的方框的组合,可以用执行规定的功能或操作的专用的基于硬件的系统来实现,或者可以用专用硬件与计算机指令的组合来实现。

[0095] 描述于本公开实施例中所涉及到的单元可以通过软件的方式实现,也可以通过硬件的方式来实现。其中,单元的名称在某种情况下并不构成对该单元本身的限定,例如,第一获取单元还可以被描述为“获取至少两个网际协议地址的单元”。

[0096] 本文中以上描述的功能可以至少部分地由一个或多个硬件逻辑部件来执行。例如,非限制性地,可以使用的示范类型的硬件逻辑部件包括:现场可编程门阵列(FPGA)、专用集成电路(ASIC)、专用标准产品(ASSP)、片上系统(SOC)、复杂可编程逻辑设备(CPLD)等等。

[0097] 第一方面,根据本公开的一个或多个实施例,提供了一种虚拟机的热升级方法,包括:

[0098] 在数据处理器DPU上创建各个虚拟机对应的快速模拟器QEMU组件,其中,所述QEMU组件中包括多个子线程,其中,所述QEMU组件中的子线程与物理机侧的VCPU线程一一对应,且所述QEMU组件中的子线程用于接收和处理与其对应的VCPU线程发送的输入输出IO请求;

[0099] 响应于针对目标虚拟机对应的目标QEMU组件的热升级指令,在DPU上创建所述目标虚拟机对应的新的QEMU组件;

[0100] 暂停所述DPU上的所述目标QEMU组件,保存所述目标QEMU组件中每个子线程的状态信息,并将每个子线程的状态信息同步到所述新的QEMU组件;

[0101] 通过所述新的QEMU组件接收并处理所述VCPU线程发送的IO请求。

[0102] 根据本公开的一个或多个实施例,所述目标QEMU组件包括多个目标子线程;所述目标子线程对应物理机上的一个目标VCPU线程;所述方法还包括:响应于针对目标虚拟机对应的目标QEMU组件的热升级指令,继续运行所述物理机上的目标VCPU线程。

[0103] 根据本公开的一个或多个实施例,所述通过所述新的QEMU组件接收并处理所述VCPU线程发送的IO请求之后,所述方法还包括:关闭所述DPU上的所述目标QEMU组件。

[0104] 根据本公开的一个或多个实施例,所述物理机上部署有虚拟机对应KVM组件,所述虚拟机在物理机上部署的多个VCPU线程通过所述KVM组件来处理业务请求;相应地,所述方法还包括:响应于针对目标虚拟机对应的目标KVM组件的热升级指令,在DPU上创建热升级线程,其中所述目标KVM组件对应所述目标虚拟机在物理机上部署的多个目标VCPU线程;通过调用所述热升级线程在所述物理机上创建每个目标VCPU线程对应的新的VCPU线程;针对每个目标VCPU线程,通过所述目标VCPU线程以及与其对应的新的VCPU线程,保存和恢复所述目标VCPU线程的状态信息。

[0105] 根据本公开的一个或多个实施例,所述针对每个目标VCPU线程,通过所述目标

VCPU线程以及与其对应的新VCPU线程,保存和恢复所述目标VCPU线程的状态信息,包括:针对每个目标VCPU线程,通过所述目标VCPU线程保存所述目标VCPU线程的状态信息;将所述目标VCPU线程的状态信息同步到与其对应的新VCPU线程。

[0106] 根据本公开的一个或多个实施例,所述将所述目标VCPU线程的状态信息同步到与其对应的新VCPU线程,包括:若所述目标VCPU线程保存所述目标VCPU线程的状态信息完成,向所述热升级线程反馈保存完成信息;通过所述热升级线程向所述目标VCPU线程对应的新VCPU线程发送信息加载指令,通过所述新VCPU线程加载所述目标VCPU线程的状态信息。

[0107] 根据本公开的一个或多个实施例,所述针对每个目标VCPU线程,通过所述目标VCPU线程以及与其对应的新VCPU线程,保存和恢复所述目标VCPU线程的状态信息之前,所述方法还包括:通过调用所述热升级线程暂停所述目标虚拟机在物理机上部署的多个目标VCPU线程。

[0108] 根据本公开的一个或多个实施例,所述针对每个目标VCPU线程,通过所述目标VCPU线程以及与其对应的新VCPU线程,保存和恢复所述目标VCPU线程的状态信息之后,所述方法还包括:通过所述每个目标VCPU线程对应的新VCPU线程处理所述目标虚拟机的业务请求;关闭所述目标虚拟机在物理机上部署的多个目标VCPU线程。

[0109] 第二方面,根据本公开的一个或多个实施例,提供了一种虚拟机的热升级设备,包括:

[0110] 创建单元,用于在数据处理器DPU上创建各个虚拟机对应的快速模拟器QEMU组件,其中,所述QEMU组件中包括多个子线程,其中,所述QEMU组件中的子线程与物理机侧的VCPU线程一一对应,且所述QEMU组件中的子线程用于接收和处理与其对应的VCPU线程发送的输入输出IO请求;

[0111] 热升级单元,用于响应于针对目标虚拟机对应的目标QEMU组件的热升级指令,在DPU上创建所述目标虚拟机对应的新QEMU组件;

[0112] 同步单元,用于暂停所述DPU上的所述目标QEMU组件,保存所述目标QEMU组件中每个子线程的状态信息,并将每个子线程的状态信息同步到所述新QEMU组件;

[0113] 处理单元,用于通过所述新QEMU组件接收并处理所述VCPU线程发送的IO请求。

[0114] 根据本公开的一个或多个实施例,所述目标QEMU组件包括多个目标子线程;所述目标子线程对应物理机上的一个目标VCPU线程;所述热升级单元,还用于响应于针对目标虚拟机对应的目标QEMU组件的热升级指令,继续运行所述物理机上的目标VCPU线程。

[0115] 根据本公开的一个或多个实施例,所述同步单元,还用于关闭所述DPU上的所述目标QEMU组件。

[0116] 根据本公开的一个或多个实施例,所述物理机上部署有虚拟机对应KVM组件,所述虚拟机在物理机上部署的多个VCPU线程通过所述KVM组件来处理业务请求;相应地,所述设备还包括:保存单元,所述保存单元,用于响应于针对目标虚拟机对应的目标KVM组件的热升级指令,在DPU上创建热升级线程,其中所述目标KVM组件对应所述目标虚拟机在物理机上部署的多个目标VCPU线程;通过调用所述热升级线程在所述物理机上创建每个目标VCPU线程对应的新VCPU线程;针对每个目标VCPU线程,通过所述目标VCPU线程以及与其对应的新VCPU线程,保存和恢复所述目标VCPU线程的状态信息。

[0117] 根据本公开的一个或多个实施例,所述保存单元,针对每个目标VCPU线程,通过所

述目标VCPU线程以及与其对应的新VCPU线程,保存和恢复所述目标VCPU线程的状态信息,包括:针对每个目标VCPU线程,通过所述目标VCPU线程保存所述目标VCPU线程的状态信息;将所述目标VCPU线程的状态信息同步到与其对应的新VCPU线程。

[0118] 根据本公开的一个或多个实施例,所述保存单元,将所述目标VCPU线程的状态信息同步到与其对应的新VCPU线程,包括:若所述目标VCPU线程保存所述目标VCPU线程的状态信息完成,向所述热升级线程反馈保存完成信息;通过所述热升级线程向所述目标VCPU线程对应的新VCPU线程发送信息加载指令,通过所述新VCPU线程加载所述目标VCPU线程的状态信息。

[0119] 根据本公开的一个或多个实施例,所述保存单元,还用于通过调用所述热升级线程暂停所述目标虚拟机在物理机上部署的多个目标VCPU线程。

[0120] 根据本公开的一个或多个实施例,所述处理单元,还用于通过所述每个目标VCPU线程对应的新VCPU线程处理所述目标虚拟机的业务请求;关闭所述目标虚拟机在物理机上部署的多个目标VCPU线程。

[0121] 第三方面,根据本公开的一个或多个实施例,提供了一种电子设备,包括:至少一个处理器和存储器;

[0122] 所述存储器存储计算机执行指令;

[0123] 所述至少一个处理器执行所述存储器存储的计算机执行指令,使得所述至少一个处理器执行如上第一方面以及第一方面各种可能的设计所述的虚拟机的热升级方法。

[0124] 第四方面,根据本公开的一个或多个实施例,提供了一种计算机可读存储介质,所述计算机可读存储介质中存储有计算机执行指令,当处理器执行所述计算机执行指令时,实现如上第一方面以及第一方面各种可能的设计所述的虚拟机的热升级方法。

[0125] 第五方面,根据本公开的一个或多个实施例,提供了一种计算机程序产品,包括计算机程序,所述计算机程序被处理器执行时实现如上第一方面以及第一方面各种可能的设计所述的虚拟机的热升级方法。

[0126] 以上描述仅为本公开的较佳实施例以及对所运用技术原理的说明。本领域技术人员应当理解,本公开中所涉及的公开范围,并不限于上述技术特征的特定组合而成的技术方案,同时也应涵盖在不脱离上述公开构思的情况下,由上述技术特征或其等同特征进行任意组合而形成的其它技术方案。例如上述特征与本公开中公开的(但不限于)具有类似功能的技术特征进行互相替换而形成的技术方案。

[0127] 此外,虽然采用特定次序描绘了各操作,但是这不应理解为要求这些操作以所示出的特定次序或以顺序次序执行来执行。在一定环境下,多任务和并行处理可能是有利的。同样地,虽然在上面论述中包含了若干具体实现细节,但是这些不应被解释为对本公开的范围的限制。在单独的实施例的上下文中描述的某些特征还可以组合地实现在单个实施例中。相反地,在单个实施例的上下文中描述的各种特征也可以单独地或以任何合适的子组合的方式实现在多个实施例中。

[0128] 尽管已经采用特定于结构特征和/或方法逻辑动作的语言描述了本主题,但是应当理解所附权利要求书中所限定的主题未必局限于上面描述的特定特征或动作。相反,上面所描述的特定特征和动作仅仅是实现权利要求书的示例形式。

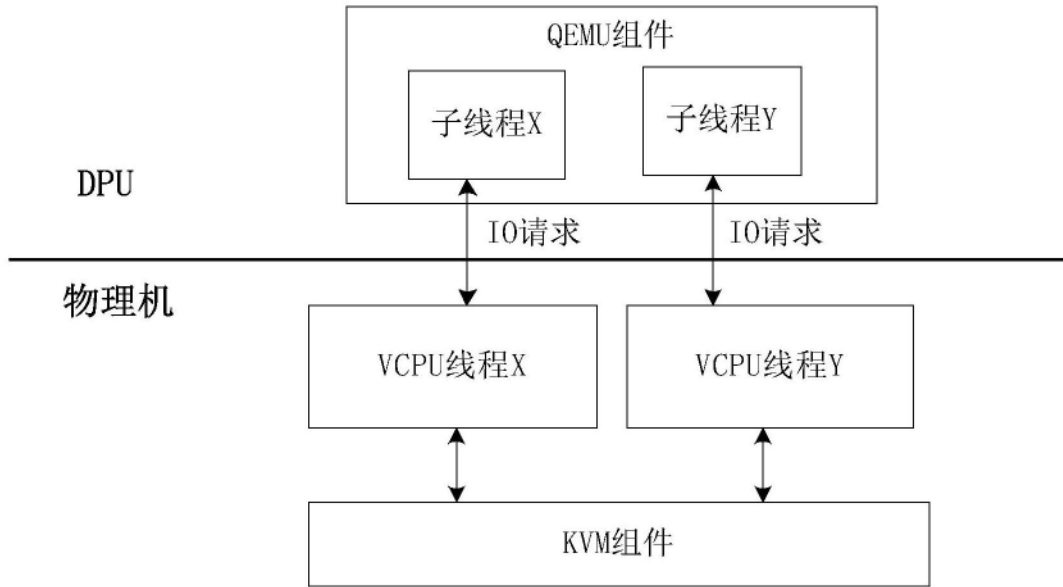


图1

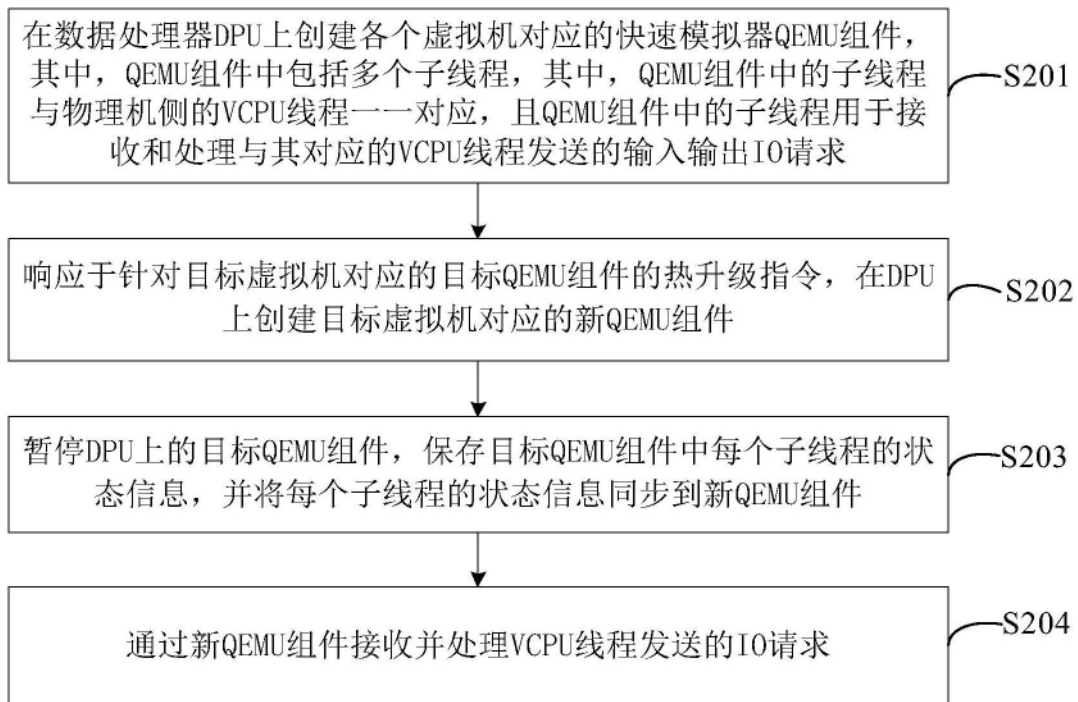


图2

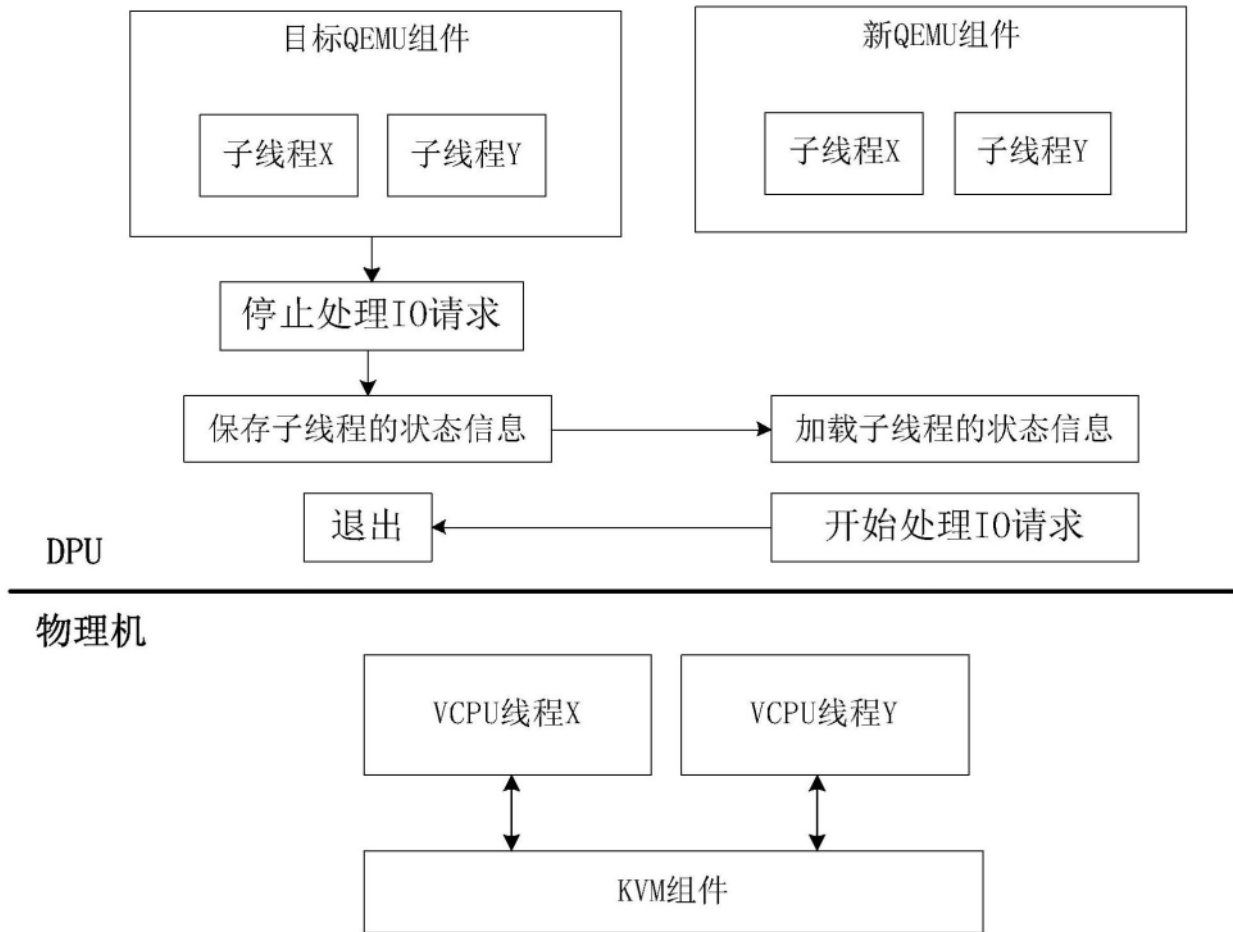


图3

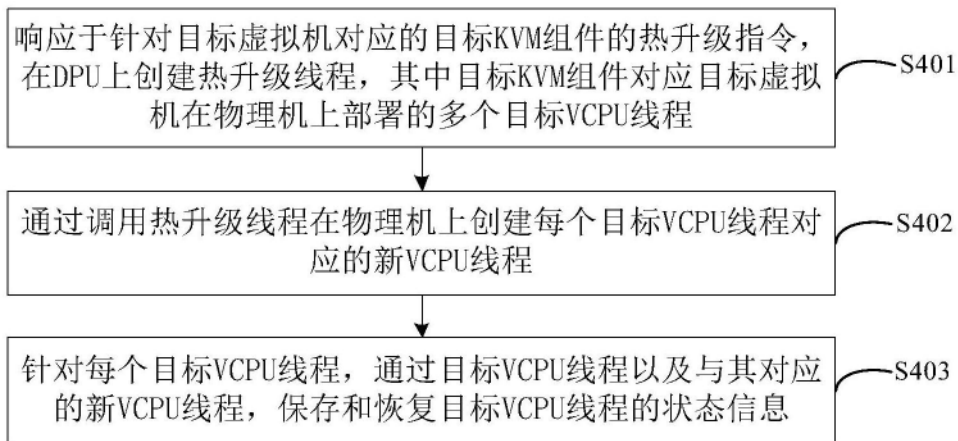


图4

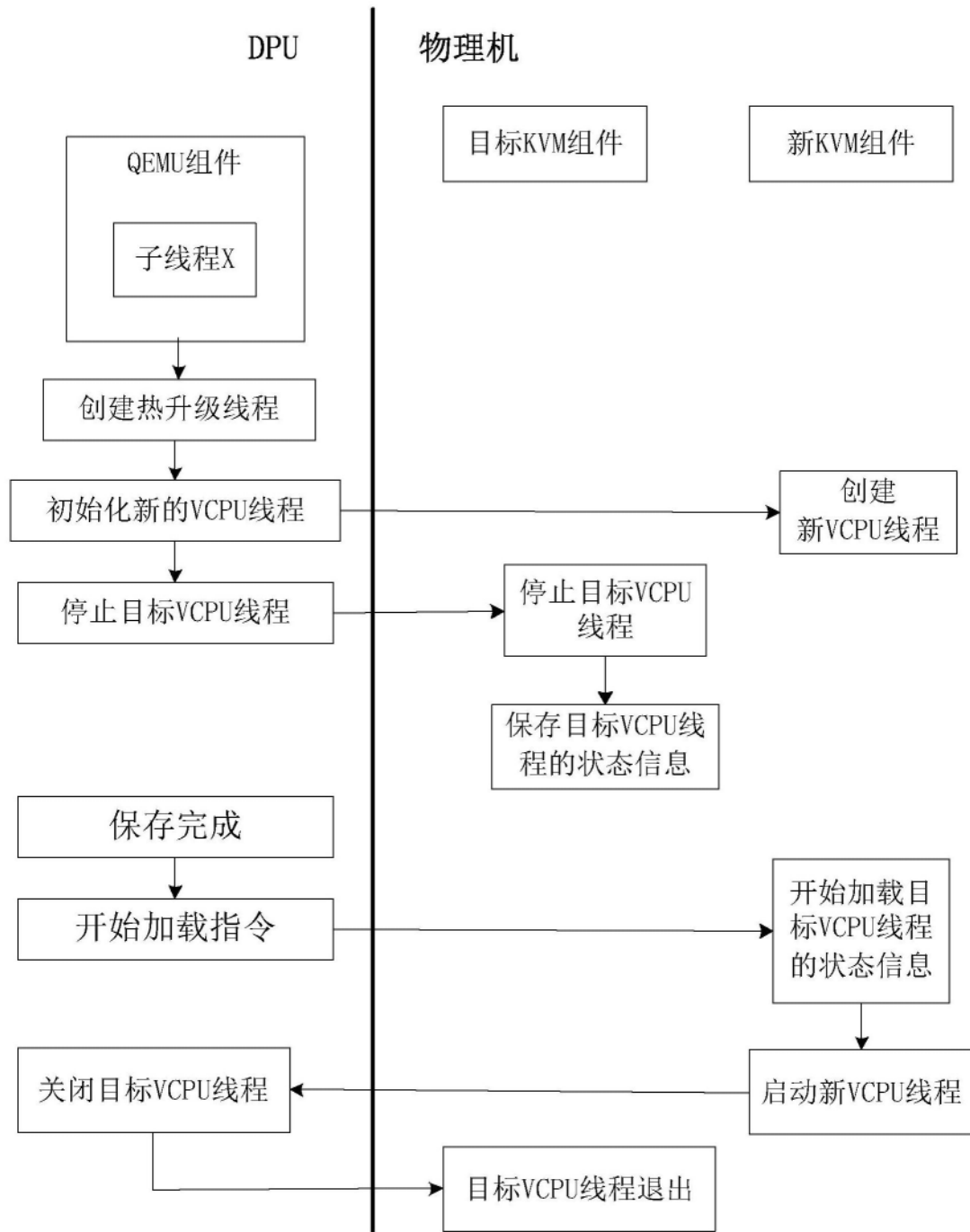


图5

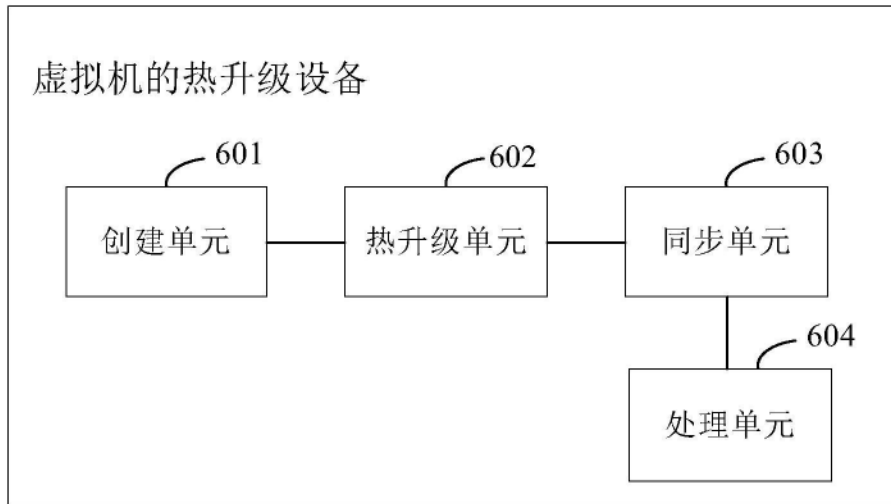


图6

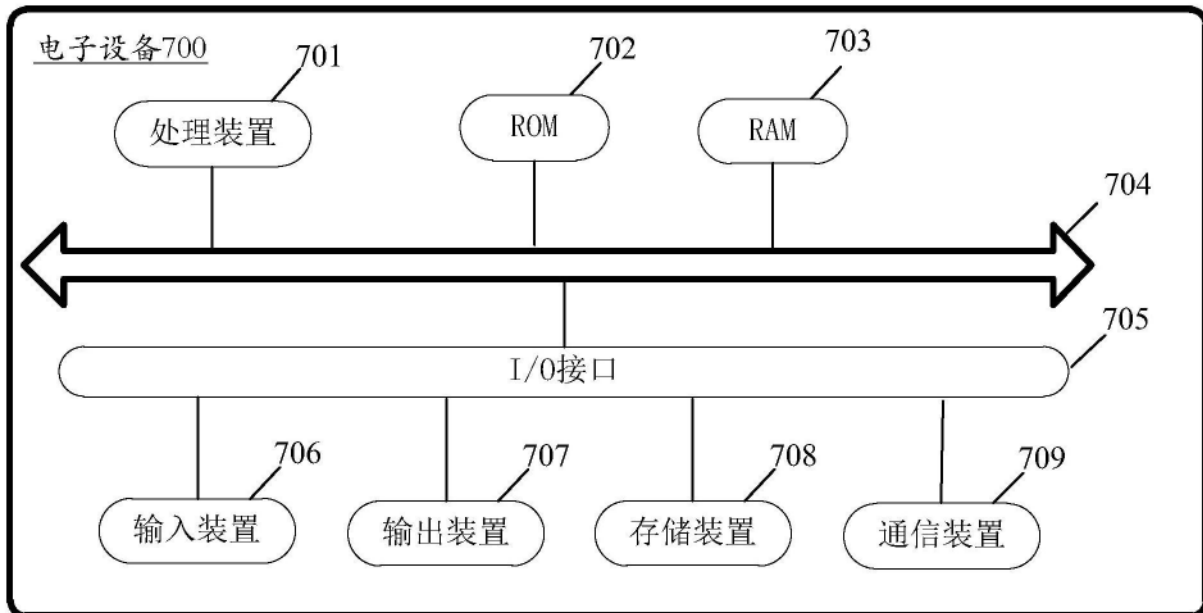


图7