



(12) 发明专利申请

(10) 申请公布号 CN 104166624 A

(43) 申请公布日 2014. 11. 26

(21) 申请号 201310180657. 6

(51) Int. Cl.

(22) 申请日 2013. 05. 15

G06F 12/02 (2006. 01)

(71) 申请人 上海贝尔股份有限公司

G06F 9/455 (2006. 01)

地址 201206 上海市浦东新区浦东金桥宁桥路 388 号

(72) 发明人 沈志宏 叶磊 龚永杰

(74) 专利代理机构 北京市金杜律师事务所

11256

代理人 郑立柱

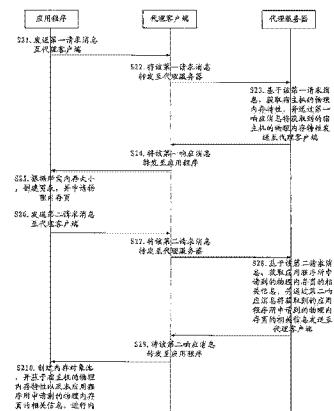
权利要求书3页 说明书5页 附图4页

(54) 发明名称

在虚拟环境下基于物理内存的内存优化方法和装置

(57) 摘要

本发明提供了在虚拟环境下基于物理内存的内存优化方案。Application 发送第一请求消息至 MM_Agent_Client，用于查询 Host 的物理内存特性。MM_Agent_Client 将该第一请求消息转发至 MM_Agent_Server。MM_Agent_Server 基于该第一请求消息，获取 Host 的物理内存特性，并通过第一响应消息将其发送至 MM_Agent_Client。MM_Agent_Client 将该第一响应消息转发至 Application。Application 根据所需内存大小，创建页表，并申请物理内存页。Application 发送第二请求消息至 MM_Agent_Client，用于查询所申请到的物理内存页的相关信息。MM_Agent_Client 将该第二请求消息转发至 MM_Agent_Server。MM_Agent_Server 基于该第二请求消息，获取 Application 所申请到的物理内存页的相关信息，并通过第二响应消息将其发送至 MM_Agent_Client。MM_Agent_Client 将该第二响应消息转发至 Application。Application 创建内存对象池，并基于物理内存特性以及物理内存页的相关信息，进行内存优化。



1. 一种在虚拟机的应用程序中基于物理内存的内存优化方法,其中,所述方法包括以下步骤:

- 发送第一请求消息至所述虚拟机的客户操作系统中的代理客户端,所述第一请求消息用于查询宿主机的物理内存特性;

- 接收由所述代理客户端转发的来自虚拟机监视器中的代理服务器的第一响应消息,所述第一响应消息中包括所述宿主机的所述物理内存特性;

- 根据所需内存大小,创建页表,并申请物理内存页;

- 发送第二请求消息至所述代理客户端,所述第二请求消息用于查询本应用程序所申请到的所述物理内存页的相关信息;

- 接收由所述代理客户端转发的来自所述代理服务器的第二响应消息,所述第二响应消息中包括本应用程序所申请到的所述物理内存页的相关信息。

2. 根据权利要求 1 所述的方法,其特征在于,所述方法还包括以下步骤:

- 创建内存对象池,并基于所述宿主机的所述物理内存特性以及本应用程序所申请到的所述物理内存页的相关信息,进行内存优化。

3. 根据权利要求 1 所述的方法,其特征在于,所述宿主机的所述物理内存特性包括内存页大小,内存 channel 数以及内存 rank 数。

4. 根据权利要求 1 所述的方法,其特征在于,本应用程序所申请到的所述物理内存页的相关信息包括所述物理内存页的物理内存地址以及 socket 数。

5. 一种在虚拟机的客户操作系统的代理客户端中基于物理内存的内存优化的方法,其中,所述方法包括以下步骤:

- 接收来自所述虚拟机的应用程序的第一请求消息,所述第一请求消息用于查询宿主机的物理内存特性;

- 将所述第一请求消息转发至虚拟机监视器中的代理服务器;

- 接收来自所述代理服务器的第一响应消息,所述第一响应消息中包括所述宿主机的所述物理内存特性;

- 将所述第一响应消息转发至所述应用程序;

- 接收来自所述应用程序的第二请求消息,所述第二请求消息用于查询所述应用程序所申请到的物理内存页的相关信息;

- 将所述第二请求消息转发至所述代理服务器;

- 接收来自所述代理服务器的第二响应消息,所述第二响应消息中包括所述应用程序所申请到的所述物理内存页的相关信息;

- 将所述第二响应消息转发至所述应用程序。

6. 一种在虚拟机监视器的代理服务器中基于物理内存的内存优化的方法,其中,所述方法包括以下步骤:

- 接收由虚拟机的客户操作系统中的代理客户端转发的来自所述虚拟机中的应用程序的第一请求消息,所述第一请求消息用于查询宿主机的物理内存特性;

- 基于所述第一请求消息,获取所述宿主机的物理内存特性,并通过第一响应消息将获取到的所述宿主机的所述物理内存特性发送至所述代理客户端;

- 接收由所述代理客户端转发的来自所述应用程序的第二请求消息,所述第二请求消

息用于查询所述应用程序所申请到的物理内存页的相关信息；

- 基于所述第二请求消息，获取所述应用程序所申请到的所述物理内存页的相关信息，并通过第二响应消息将获取到的所述应用程序所申请到的所述物理内存页的相关信息发送至所述代理客户端。

7. 一种在虚拟机的应用程序中基于物理内存进行内存优化的装置，其中，所述装置包括：

第一发送单元，用于发送第一请求消息至所述虚拟机的客户操作系统中的代理客户端，所述第一请求消息用于查询宿主机的物理内存特性；

第一接收单元，用于接收由所述代理客户端转发的来自虚拟机监视器中的代理服务器的第一响应消息，所述第一响应消息中包括所述宿主机的所述物理内存特性；

分配单元，用于根据所需内存大小，创建页表，并申请物理内存页；

第二发送单元，用于发送第二请求消息至所述代理客户端，所述第二请求消息用于查询本应用程序所申请到的所述物理内存页的相关信息；

第二接收单元，用于接收由所述代理客户端转发的来自所述代理服务器的第二响应消息，所述第二响应消息中包括本应用程序所申请到的所述物理内存页的相关信息。

8. 根据权利要求 7 所述的装置，其特征在于，所述装置还包括：

内存优化单元，用于创建内存对象池，并基于所述宿主机的所述物理内存特性以及本应用程序所申请到的所述物理内存页的相关信息，进行内存优化。

9. 根据权利要求 7 所述的装置，其特征在于，所述宿主机的所述物理内存特性包括内存页大小，内存 channel 数以及内存 rank 数。

10. 根据权利要求 7 所述的装置，其特征在于，本应用程序所申请到的所述物理内存页的相关信息包括所述物理内存页的物理内存地址以及 socket 数。

11. 一种在虚拟机的客户操作系统的代理客户端中基于物理内存进行内存优化的装置，其中，所述装置包括：

第三接收单元，用于接收来自所述虚拟机的应用程序的第一请求消息，所述第一请求消息用于查询宿主机的物理内存特性；

第三发送单元，用于将所述第一请求消息转发至虚拟机监视器中的代理服务器；

第四接收单元，用于接收来自所述代理服务器的第一响应消息，所述第一响应消息中包括所述宿主机的所述物理内存特性；

第四发送单元，用于将所述第一响应消息转发至所述应用程序；

第五接收单元，用于接收来自所述应用程序的第二请求消息，所述第二请求消息用于查询所述应用程序所申请到的物理内存页的相关信息；

第五发送单元，用于将所述第二请求消息转发至所述代理服务器；

第六接收单元，用于接收来自所述代理服务器的第二响应消息，所述第二响应消息中包括所述应用程序所申请到的所述物理内存页的相关信息；

第六发送单元，用于将所述第二响应消息转发至所述应用程序。

12. 一种在虚拟机监视器的代理服务器中基于物理内存进行内存优化的装置，其中，所述装置包括：

第七接收单元，用于接收由虚拟机的客户操作系统中的代理客户端转发的来自所述虚

拟机中的应用程序的第一请求消息，所述第一请求消息用于查询宿主机的物理内存特性；

第一获取单元，用于基于所述第一请求消息，获取所述宿主机的物理内存特性，并通过第一响应消息将获取到的所述宿主机的所述物理内存特性发送至所述代理客户端；

第八接收单元，用于接收由所述代理客户端转发的来自所述应用程序的第二请求消息，所述第二请求消息用于查询所述应用程序所申请到的物理内存页的相关信息；

第二获取单元，用于基于所述第二请求消息，获取所述应用程序所申请到的所述物理内存页的相关信息，并通过第二响应消息将获取到的所述应用程序所申请到的所述物理内存页的相关信息发送至所述代理客户端。

在虚拟环境下基于物理内存的内存优化方法和装置

技术领域

[0001] 本申请涉及通信系统,尤其涉及在虚拟环境下基于物理内存的内存优化方法和装置。

背景技术

[0002] 内存优化对于传统的“IP 网络转发”是非常重要的,其与特定的硬件关系密切。应用程序的开发者需要感知物理内存的硬件配置、分布及特性,从而基于例如物理内存的 channel 数目, rank 数目等参数优化应用程序。内存优化能够极大地提高应用的性能。

[0003] 在虚拟化的云环境下,应用程序运行于虚拟机 (virtual machine, VM) 中的客户操作系统 (Guest OS) 之上。真正的物理内存对于 Guest OS 来说是不可见的。应用程序仅能够访问由虚拟机监视器 (virtual machine monitor, VMM) 虚拟出来的虚拟物理内存。

[0004] 当虚拟机中的应用程序访问内存时,首先由 Guest OS 将虚拟内存映射成虚拟物理内存,然后由 VMM 将虚拟物理内存再次映射到真正的物理内存。由于物理内存对于 Guest OS 是不可见的,因此在 Guest OS 中的内存优化是针对虚拟物理内存的,而经由 VMM 再次映射后,所有的内存优化 (如连续内存分配,内存 channel, rank 优化, NUMA 管理) 将可能失效。例如,虚拟机中连续的内存分配,在映射后可能变成不连续的,使得程序运行效率低下。

[0005] 此外,在虚拟化的云环境里,当虚拟机进行离线迁移时,宿主机 (host machine) 的物理内存的格局有可能发生变化,比如由 NUMA 转到非 NUMA 的架构,在此情形下,可能会导致原先虚拟机中应用的内存优化失效。

发明内容

[0006] 基于上述考量,本发明提出了一种在虚拟环境下基于物理内存的内存优化方案。

[0007] 本发明的主要构思在于:在传统的 VMM 的内存映射之外,提供一种机制,使得在虚拟机中看到的是宿主机的真实物理内存架构及特性,从而使得在虚拟机中所做的内存优化是直接针对真实物理内存而非虚拟物理内存的。

[0008] 在实现上,在虚拟机启动的时候,由 VMM 从宿主机的物理内存中为该虚拟机分配连续的宏大页面物理内存,并且这些内存是不被交换到硬盘上的 (这在目前的虚拟机产品中已经实现)。同时,在 VMM 中实现一个代理服务器 (MM_Agent_Server),负责向 Guest OS 中的代理客户端 (MM_Agent_Client) 提供 API,以便 Guest OS 中的应用程序在运行优化过程中,查询底层物理内存的特性 (包括这些内存的物理连接,配置特性等)。

[0009] 对于虚拟机的离线迁移,在离线迁移的过程中,由 MM_Agent_Server 动态地检测当前宿主机的物理内存,并在虚拟机中的应用程序执行内存分配和优化时,将此物理内存信息传递给应用程序。

[0010] 根据本发明的一个方面,提出了一种在虚拟机的应用程序中基于物理内存的内存优化方法,其中,所述方法包括以下步骤:发送第一请求消息至所述虚拟机的客户操作系统中的代理客户端,所述第一请求消息用于查询宿主机的物理内存特性;接收由所述代理客

户端转发的来自虚拟机监视器中的代理服务器的第一响应消息，所述第一响应消息中包括所述宿主机的所述物理内存特性；根据所需内存大小，创建页表，并申请物理内存页；发送第二请求消息至所述代理客户端，所述第二请求消息用于查询本应用程序所申请到的所述物理内存页的相关信息；以及接收由所述代理客户端转发的来自所述代理服务器的第二响应消息，所述第二响应消息中包括本应用程序所申请到的所述物理内存页的相关信息。

[0011] 有利的，所述方法还包括以下步骤：创建内存对象池，并基于所述宿主机的所述物理内存特性以及本应用程序所申请到的所述物理内存页的相关信息，进行内存优化。

[0012] 有利的，所述宿主机的所述物理内存特性包括内存页大小，内存 channel 数以及内存 rank 数。

[0013] 有利的，本应用程序所申请到的所述物理内存页的相关信息包括所述物理内存页的物理内存地址以及 socket 数。

[0014] 根据本发明的另一个方面，提出了一种在虚拟机的客户操作系统的代理客户端中基于物理内存的内存优化的方法，其中，所述方法包括以下步骤：接收来自所述虚拟机的应用程序的第一请求消息，所述第一请求消息用于查询宿主机的物理内存特性；将所述第一请求消息转发至虚拟机监视器中的代理服务器；接收来自所述代理服务器的第一响应消息，所述第一响应消息中包括所述宿主机的所述物理内存特性；将所述第一响应消息转发至所述应用程序；接收来自所述应用程序的第二请求消息，所述第二请求消息用于查询所述应用程序所申请到的物理内存页的相关信息；将所述第二请求消息转发至所述代理服务器；接收来自所述代理服务器的第二响应消息，所述第二响应消息中包括所述应用程序所申请到的所述物理内存页的相关信息；以及将所述第二响应消息转发至所述应用程序。

[0015] 根据本发明的又一个方面，提出了一种在虚拟机监视器的代理服务器中基于物理内存的内存优化的方法，其中，所述方法包括以下步骤：接收由虚拟机的客户操作系统中的代理客户端转发的来自所述虚拟机中的应用程序的第一请求消息，所述第一请求消息用于查询宿主机的物理内存特性；基于所述第一请求消息，获取所述宿主机的物理内存特性，并通过第一响应消息将获取到的所述宿主机的所述物理内存特性发送至所述代理客户端；接收由所述代理客户端转发的来自所述应用程序的第二请求消息，所述第二请求消息用于查询所述应用程序所申请到的物理内存页的相关信息；以及基于所述第二请求消息，获取所述应用程序所申请到的所述物理内存页的相关信息，并通过第二响应消息将获取到的所述应用程序所申请到的所述物理内存页的相关信息发送至所述代理客户端。

[0016] 本发明的各个方面将通过下文中的具体实施例的说明而更加清晰。

附图说明

[0017] 通过阅读参照以下附图所作的对非限制性实施例所作的详细描述，本发明的上述及其他特征将会更加清晰：

[0018] 图 1 示出了根据本发明的一个实施例的内存直接路径架构图；

[0019] 图 2 示出了根据本发明的一个实施例的基于物理内存进行内存优化的方法流程图；

[0020] 图 3 示出了根据本发明的一个实施例的 userplane 应用架构图；

[0021] 图 4 示出了本发明的基于物理内存的内存优化的一个例子的流程图。

[0022] 附图中相同或者相似的附图标示表示相同或者相似的部件。

具体实施方式

[0023] 以下参考附图对本发明的实施例进行描述。

[0024] 参照图 1 和图 2, 在创建虚拟机时, 虚拟机监视器 (VMM) 为该虚拟机配置固定大小的连续的物理内存。

[0025] 当客户操作系统 (Guest OS) 中的应用程序启动时, 首先, 在步骤 S21 中, 应用程序发送第一请求消息至虚拟机的客户操作系统中的代理客户端 (MM_Agent_Client)。该第一请求消息用于查询宿主机 (Host) 的物理内存特性。有利的, 该物理内存特性包括内存页大小, 内存 channel 数以及内存 rank 数等。

[0026] 代理客户端接收到来自应用程序的第一请求消息后, 在步骤 S22 中, 将该第一请求消息转发至虚拟机监视器中的代理服务器 (MM_Agent_Server)。

[0027] 代理服务器接收到由代理客户端转发的来自应用程序的第一请求消息后, 在步骤 S23 中, 基于该第一请求消息, 获取宿主机的物理内存特性, 并通过第一响应消息将获取到的宿主机的物理内存特性发送至代理客户端。

[0028] 代理客户端接收到来自代理服务器的第一响应消息后, 在步骤 S24 中, 将该第一响应消息转发至应用程序。

[0029] 应用程序接收到由代理客户端转发的来自代理服务器的第一响应消息后, 在步骤 S25 中, 根据所需内存大小, 创建页表, 并申请物理内存页。

[0030] 然后, 在步骤 S26 中, 应用程序发送第二请求消息至代理客户端。该第二请求消息用于查询本应用程序所申请到的物理内存页的相关信息。有利的, 本应用程序所申请到的物理内存页的相关信息包括物理内存页的物理内存地址以及 socket 数。

[0031] 代理客户端接收到来自应用程序的第二请求消息后, 在步骤 S27 中, 将该第二请求消息转发至代理服务器。

[0032] 代理服务器接收到由代理客户端转发的来自应用程序的第二请求消息后, 在步骤 S28 中, 基于该第二请求消息, 获取应用程序所申请到的物理内存页的相关信息, 并通过第二响应消息将获取到的应用程序所申请到的物理内存页的相关信息发送至代理客户端。

[0033] 代理客户端接收到来自代理服务器的第二响应消息后, 在步骤 S29 中, 将该第二响应消息转发至应用程序。

[0034] 应用程序接收到由代理客户端转发的来自代理服务器的第二响应消息后, 有利的, 在步骤 S210 中, 该应用程序创建内存对象池, 并基于宿主机的物理内存特性以及本应用程序所申请到的物理内存页的相关信息, 进行内存优化。例如, 可以创建一个通用的对象池, 每个对象的长度是 channel 和 rank 乘积的整数倍, 使得对象能够散布在不同的内存通道上, 以便能够并行访问, 提高内存访问效率。

[0035] 由于在虚拟机中看到的是物理内存的分布, 其应用程序所做的优化将直接应用于真实的物理内存上。

[0036] 在云环境下, 由于是动态检测物理内存, 所以即使做了虚拟机离线迁移, 虚拟机中应用程序的内存优化也能够保持有效。

[0037] 以下将针对一个具体应用场景, 对本发明的技术方案进行进一步描述。

[0038] 电信网络从 2G/3G 到 4G,逐渐向全 IP 网络演进。在用户面的实现过程中,由于硬件性能的提升,许多用户面也采用软件方式在云环境下来实现。如图 3 所示,以 KVM 为例描述 userplane 应用的实现。但是本领域技术人员可以理解,本发明的技术方案不仅可以应用于 KVM,其也可以应用于其它的虚拟化平台,如 XEN, lguest 等。在虚拟机与 KVM 的通信中,采用了 KVM 的半虚拟化技术 virtio。

[0039] 以下参照图 4 对具体流程进行描述。

[0040] 1) 用户创建虚拟机,设置虚拟机的内存容量,NUMA 等信息。

[0041] 2) 当虚拟机启动时,VMM 为 VM 保留物理内存,其大小为创建虚拟机中设置的内存大小。

[0042] 3) MM_Agent_Server 收集相应的物理内存信息,如 socket 信息,物理内存地址,channel 数,rank 数等。

[0043] 以上步骤和常规流程相同。

[0044] 4) 当 userplane 应用启动时,需要建立一个基于物理地址连续的对象池。

[0045] 5) 应用程序发送第一请求消息 sys_memory_info_req 给 MM_Agent_Client,用于查询宿主机的物理内存特性。

[0046] 6) MM_Agent_Client 转发第一请求消息 sys_memory_info_req 给 MM_Agent_Server。

[0047] 7) MM_Agent_Server 接收到第一请求消息 sys_memory_info_req,查询宿主机的物理内存特性,例如页面大小,channel 数,rank 数等;然后发送第一响应消息 sys_memory_info_ack 给 MM_Agent_Client。

[0048] 8) MM_Agent_Client 转发第一响应消息 sys_memory_info_ack 给 userplane application。

[0049] 9) 用户创建页表数组,数组大小为 REQUEST_SIZE/PAGE_SIZE。

[0050] 10) userplane 应用程序使用传统方式申请内存页,并且在上一步创建的页表数组中保存相应的页信息。

[0051] 11) userplane 应用程序向 MM_Agent_Client 发送第二请求消息 host_memory_info_req,用于查询相应的物理地址信息。

[0052] 12) MM_Agent_Client 转发第二请求消息 host_memory_info_req 给 MM_Agent_Server。

[0053] 13) MM_Agent_Server 填写页表项的各种信息,如真实的物理地址,socket 数等;然后通过第二响应消息 host_memory_info_ack 返回给 MM_Agent_Client。

[0054] 14) MM_Agent_Client 转发第二响应消息 host_memory_info_ack 给 userplane 应用程序。

[0055] 15) userplane 应用程序根据收到的信息创建对象池。

[0056] * 在对页表数组中,根据内存页的物理地址进行排序。

[0057] * 在应用程序中,创建连续虚拟地址到相应连续物理地址的映射。

[0058] * 在创建出来的连续的内存空间上,创建对象池。

[0059] * 在创建对象池的过程中,根据之前取得的宿主机的 channel 数和 rank 数,对于对象的长度做优化,使得对象长度是 channel 数和 rank 数乘积的整数倍,从而使得各个

channel1 能够同时均衡的加载数据。

[0060] 16) 至此,应用程序创建的对象池,其对应的真实的地物理内存也是连续的。同时在应用层上,每个对象的访问都是优化的。

[0061] 对于本领域技术人员而言,显然本发明不限于上述示范性实施例的细节,而且在不背离本发明的精神或基本特征的情况下,能够以其他的具体形式实现本发明。因此,无论从哪一点来看,均应将实施例看作是示范性的,而且是非限制性的,不应将权利要求中的任何附图标视为限制所涉及的权利要求。此外,明显的,“包括”一词不排除其他元件或步骤,在元件前的“一个”一词不排除包括“多个”该元件。产品权利要求中陈述的多个元件也可以由一个元件通过软件或者硬件来实现。第一,第二等词语用来表示名称,而并不表示任何特定的顺序。

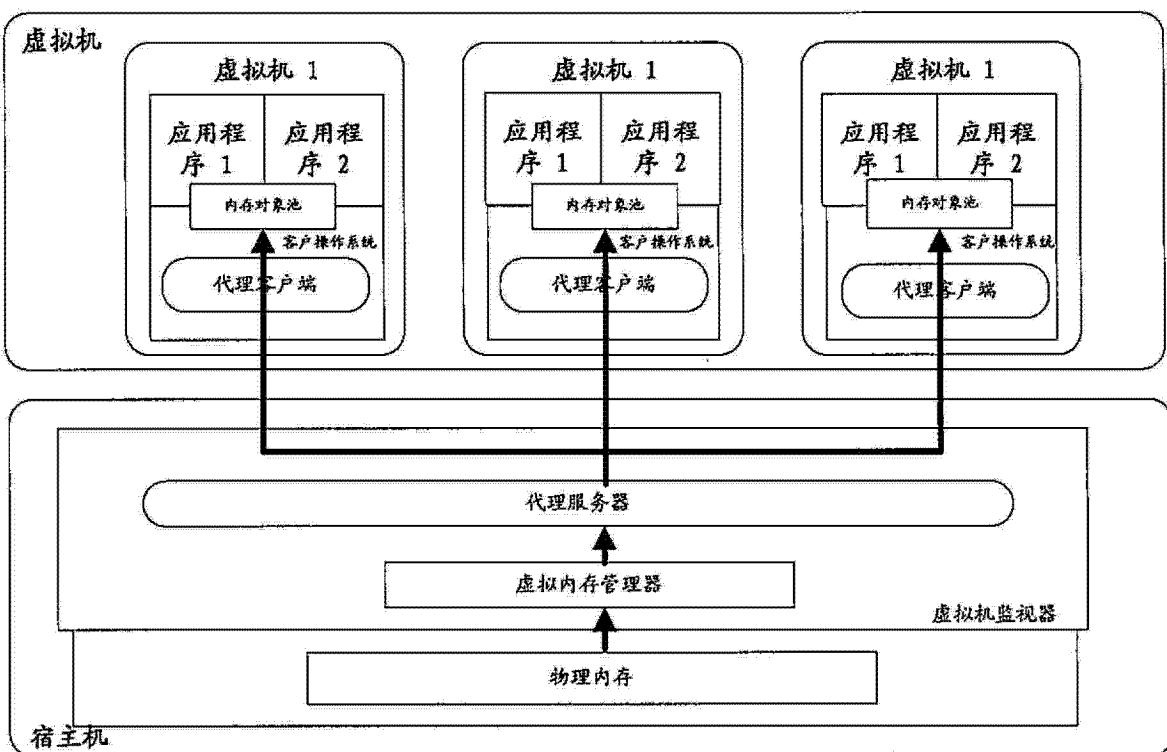


图 1

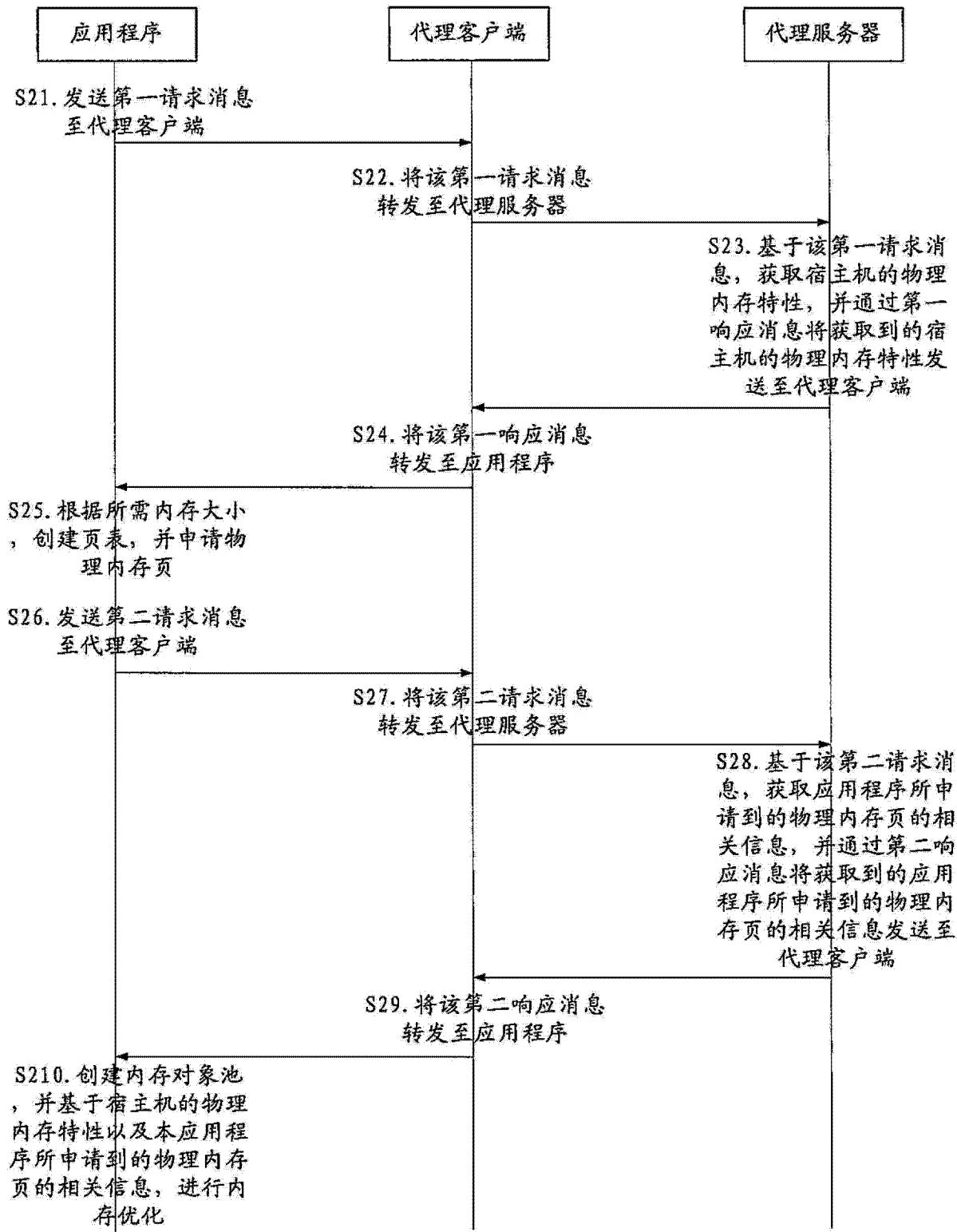


图 2

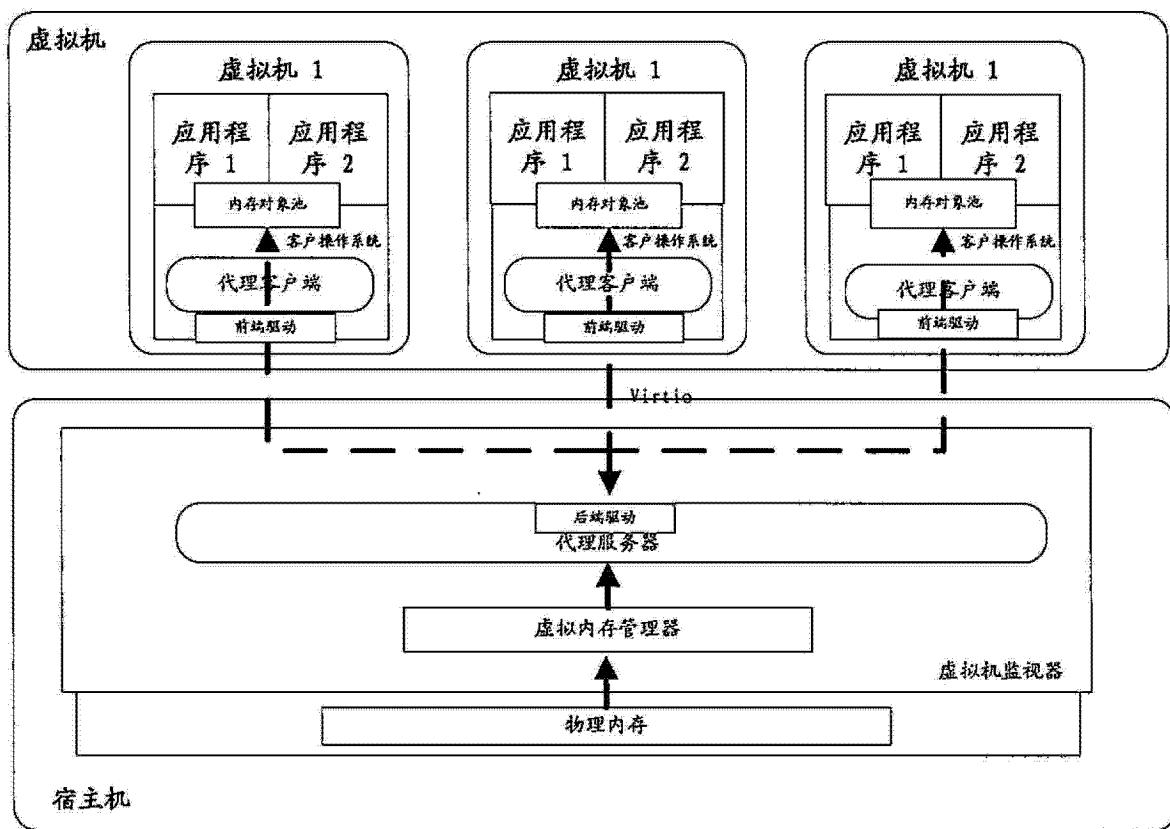


图 3

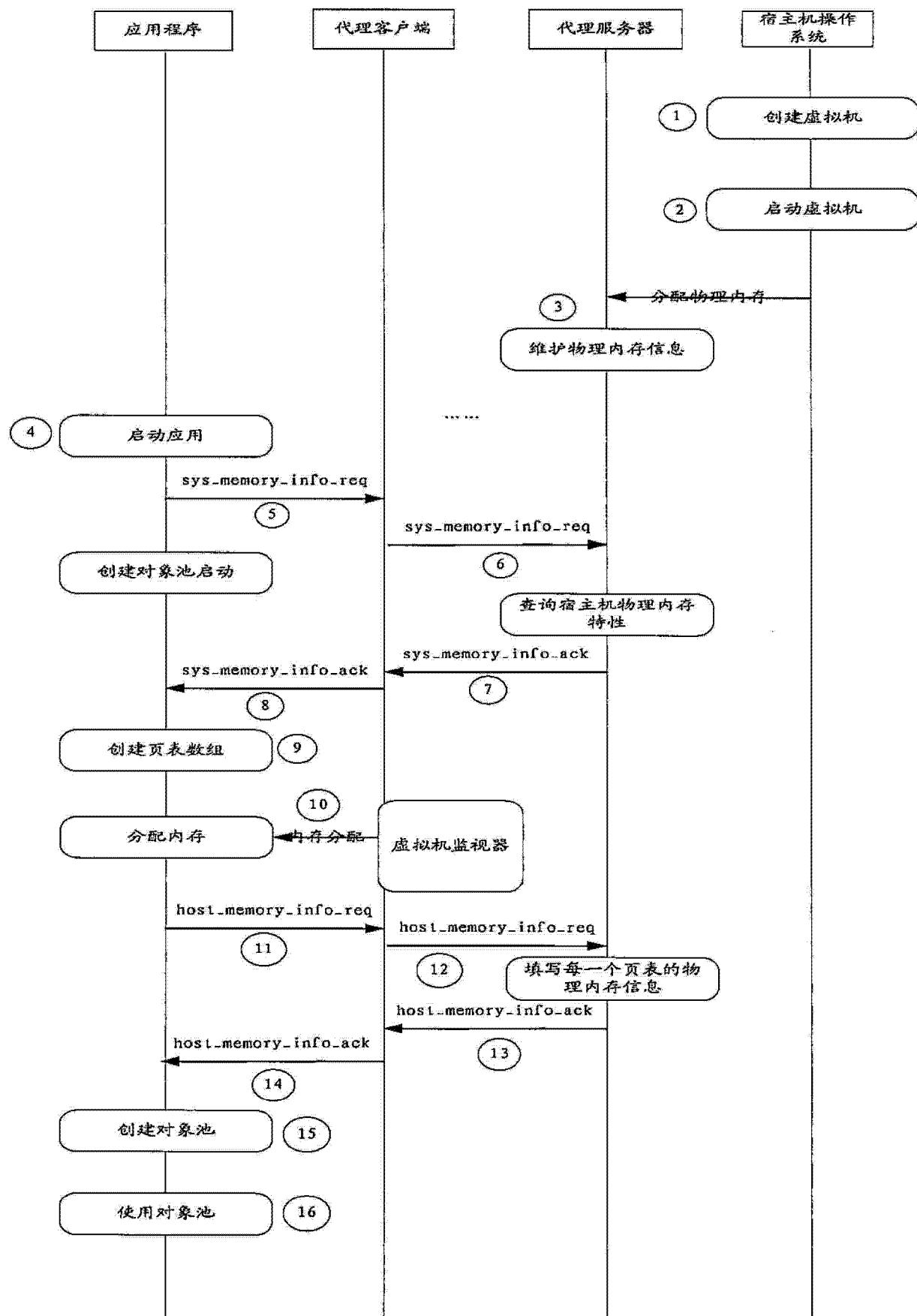


图 4