



(12) 发明专利

(10) 授权公告号 CN 102541619 B

(45) 授权公告日 2015. 09. 16

(21) 申请号 201010612360. 9

审查员 王晓飞

(22) 申请日 2010. 12. 23

(73) 专利权人 国际商业机器公司
地址 美国纽约

(72) 发明人 甘志 戴晓君 韩瑞波 刘显

(74) 专利代理机构 北京市中咨律师事务所
11247

代理人 于静 周良玉

(51) Int. Cl.

G06F 9/455(2006. 01)

G06F 9/50(2006. 01)

(56) 对比文件

CN 101847105 A, 2010. 09. 29,

CN 101158924 A, 2008. 04. 09,

US 2010/0217916 A1, 2010. 08. 26,

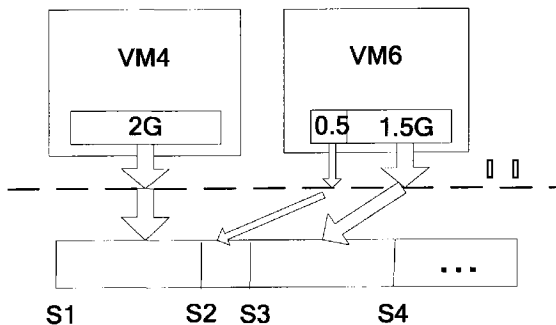
权利要求书2页 说明书8页 附图4页

(54) 发明名称

虚拟机管理装置和方法

(57) 摘要

提供了虚拟机管理装置和方法,所述装置包括:虚拟机信息获取单元,配置为获取当前虚拟机的软件层级信息;内存分析单元,配置为参照已安装的至少一个虚拟机的现有软件层级信息和所述当前虚拟机的软件层级信息,将分配给当前虚拟机的逻辑内存划分为专用部分和共享部分;以及内存映射单元,配置为将上述共享部分映射到物理内存的共享区段,所述共享区段由至少一个已经安装的虚拟机所使用。还提供了与上述装置对应的方法。通过以上装置和方法,可以使得同一物理平台上的多个虚拟机实现内存的部分共享,从而在有效利用内存资源的同时,保持虚拟机的相对独立和隔离。



1. 一种虚拟机管理装置,包括:

虚拟机信息获取单元,配置为获取当前虚拟机的软件层级信息;

内存分析单元,配置为参照已安装的至少一个虚拟机的现有软件层级信息和所述当前虚拟机的软件层级信息,将分配给当前虚拟机的逻辑内存划分为专用部分和共享部分,其中所述共享部分与所述至少一个虚拟机的现有软件层级信息和所述当前虚拟机的软件层级信息的共有的软件层级信息相关联;以及

内存映射单元,配置为将上述共享部分映射到物理内存的共享区段,所述共享区段由所述已安装的至少一个虚拟机中的至少一个所使用。

2. 如权利要求 1 所述的虚拟机管理装置,其中,所述虚拟机信息获取单元配置为,在所述当前虚拟机的安装过程中读取其安装文件的头部信息,从而获取当前虚拟机的软件层级信息。

3. 如权利要求 1 所述的虚拟机管理装置,其中,所述虚拟机信息获取单元配置为,以树形图形式和柱形图形式中的至少一个维持已安装的至少一个虚拟机的现有软件层级信息。

4. 如权利要求 1 所述的虚拟机管理装置,其中,所述内存分析单元配置为,将当前虚拟机的软件层级信息与所述现有软件层级信息进行比对,确定当前虚拟机与最接近的已安装虚拟机所共有的软件层级信息,并使得所述共享部分与上述共有的软件层级信息相关联。

5. 如权利要求 1 所述的虚拟机管理装置,其中所述内存映射单元配置为,维持映射表,通过所述映射表记录逻辑内存的共享部分与物理内存的共享区段的映射关系。

6. 如权利要求 5 所述的虚拟机管理装置,其中所述映射表包含状态位,用以标记各个内存区段的共享状态。

7. 如权利要求 1 所述的虚拟机管理装置,其中所述内存映射单元配置为,释放卸载的虚拟机的专用部分所对应的物理内存。

8. 如权利要求 1 所述的虚拟机管理装置,其中所述内存映射单元配置为,响应于对所述共享部分的修改请求,从空闲的物理内存中开辟出新的内存区段,将共享部分所对应的共享区段中的数据复制到该新的内存区段,并使得所述共享部分映射到该新的内存区段。

9. 一种虚拟机管理方法,包括:

获取当前虚拟机的软件层级信息;

参照已安装的至少一个虚拟机的现有软件层级信息和所述当前虚拟机的软件层级信息,将分配给当前虚拟机的逻辑内存划分为专用部分和共享部分,其中所述共享部分与所述至少一个虚拟机的现有软件层级信息和所述当前虚拟机的软件层级信息的共有的软件层级信息相关联,其中,划分逻辑内存的步骤包括,将当前虚拟机的软件层级信息与所述现有软件层级信息进行比对,确定当前虚拟机与最接近的已安装虚拟机所共有的软件层级信息,并使得所述共享部分与上述共有的软件层级信息相关联;以及

将上述共享部分映射到物理内存的共享区段,所述共享区段由所述已安装的至少一个虚拟机中的至少一个所使用。

10. 如权利要求 9 所述的方法,其中,获取当前虚拟机的软件层级信息的步骤包括,在所述当前虚拟机的安装过程中读取其安装文件的头部信息,从而获取当前虚拟机的软件层级信息。

11. 如权利要求 9 所述的方法,其中,所述已安装的至少一个虚拟机的现有软件层级信

息被维持为树形图形式和柱形图形式中的至少一个。

12. 如权利要求 9 所述的方法, 其中将共享部分映射到共享区段的步骤包括, 维持映射表, 通过所述映射表记录逻辑内存的共享部分与物理内存的共享区段的映射关系。

13. 如权利要求 12 所述的方法, 其中所述映射表包含状态位, 用以标记各个内存区段的共享状态。

14. 如权利要求 9 所述的方法, 还包括, 释放卸载的虚拟机的专用部分所对应的物理内存。

15. 如权利要求 9 所述的方法, 还包括, 响应于对所述共享部分的修改请求, 从空闲的物理内存中开辟出新的内存区段, 将共享部分所对应的共享区段中的数据复制到该新的内存区段, 并使得所述共享部分映射到该新的内存区段。

虚拟机管理装置和方法

技术领域

[0001] 本发明涉及虚拟机,更具体而言,涉及管理虚拟机的装置和方法。

背景技术

[0002] 近来,虚拟化技术已经广泛应用在各种应用环境之中,用以提高 IT 资源和应用程序的效率。虚拟化技术可以随时根据需要,将资源、应用程序甚至服务器分配到相应的位置,从而整合资源,提高计算机的可用性,由此节省 50%到 70%的 IT 总成本。

[0003] 虚拟化的典型实现是在一台物理计算机上模拟出一台或多台虚拟的计算机,即虚拟机 (Virtual Machine),也就是通过软件模拟的具有完整硬件系统功能的、运行在一个完全隔离环境中的完整计算机系统。这样的虚拟机与物理计算机一样,承载着自身的客户操作系统和应用程序,并具有物理计算机上的所有组件,包括主板、硬盘、网卡控制器等。虽然多个虚拟机可以共享一台计算机的物理资源,但它们彼此之间保持完全隔离状态,如同它们是不同的物理计算机一样。例如,如果在一台物理服务器上有四个虚拟机,而其中一个虚拟机崩溃了,其他三个虚拟机仍将可用。在可用性和安全性方面,虚拟环境中运行的应用程序之所以远优于在传统的非虚拟化系统中运行的应用程序,隔离就是一个重要的原因。

[0004] 为了保持虚拟机之间的隔离,虚拟机之间通常会保持硬件资源分配的独立性,尽量少地共享同一部分物理资源。具体地,在内存管理方面,处于物理平台和虚拟机之间的管理程序会为每个虚拟机分配一段独立的内存,每个虚拟机只对自己“专属”的内存进行存取。图 1 示出现有技术中多个虚拟机的内存管理。如图 1 所示,在同一物理平台上安装了 10 个虚拟机,分别标注为虚拟机 VM1-10。虚拟机的管理程序根据物理平台的硬件容量和每个虚拟机的运行需要,为每个虚拟机分配一段内存,并维持虚拟机内存和物理内存之间的映射关系。例如在一个例子中,物理平台的总内存容量为 15G,管理程序为每个虚拟机分配 1G 的内存,并且使得每个虚拟机所获得的内存之间相互独立,例如将物理内存的 0-1G 分配给虚拟机 VM1,将 1-2G 分配给虚拟机 VM2。由此,每个虚拟机只能看到分配给自己的 1G 内存,并对这部分内存进行读写。具体地,在 VM2 进行内存访问时,VM2 的内存管理单元首先将所请求的逻辑地址转化为自己能看到的 0-1G 物理地址中的相应地址,称为客户物理地址 (guest physical address)。然后,管理程序获得该客户物理地址,通过查阅虚拟机内存和物理内存之间的映射关系,将获得的客户物理地址映射到物理内存的真实物理地址,并返回给虚拟机。例如,如果 VM2 所请求的客户物理地址是 0.5G,那么管理程序返回的真实物理地址就是 1.5G。如此,每个虚拟机独立地享有专用的内存区段,彼此隔离地进行内存访问。

[0005] 然而,上述的隔离方法在资源利用上还存在一定不足。假定一个虚拟机流畅地运行一个操作系统和基本的上层应用所需的内存为 2G,那么在同一物理平台上安装 100 个虚拟机就需要至少 200G 物理内存,这样巨大的内存需要对物理平台提出了很高的要求。然而,实际上,在这 100 个虚拟机中,很可能大部分虚拟机是基于相同的操作系统,因此物理平台上存在对同一操作系统的多重备份,这就存在着一定的资源浪费。然而,对于整个系统来说,保持虚拟机之间的独立和隔离是虚拟化的基本要求。因此,如何更加有效地利用物理

平台的资源,同时保持虚拟机之间的隔离成为一项很有意义的挑战。

发明内容

[0006] 考虑到以上提出的问题,提出本发明,以提高承载多个虚拟机的物理平台的资源利用率。

[0007] 根据本发明第一方面,提供一种虚拟机管理装置,包括:虚拟机信息获取单元,配置为获取当前虚拟机的软件层级信息;内存分析单元,配置为参照已安装的至少一个虚拟机的现有软件层级信息和所述当前虚拟机的软件层级信息,将分配给当前虚拟机的逻辑内存划分为专用部分和共享部分;以及内存映射单元,配置为将上述共享部分映射到物理内存的共享区段,所述共享区段由所述已安装的至少一个虚拟机中的至少一个所使用。

[0008] 根据本发明第二方面,提供一种虚拟机管理方法,包括:获取当前虚拟机的软件层级信息;参照已安装的至少一个虚拟机的现有软件层级信息和所述当前虚拟机的软件层级信息,将分配给当前虚拟机的逻辑内存划分为专用部分和共享部分;以及将上述共享部分映射到物理内存的共享区段,所述共享区段由所述已安装的至少一个虚拟机中的至少一个所使用。

[0009] 通过本发明实施例的方法和装置,可以使得同一物理平台上的多个虚拟机在各个层级上进行内存的部分共享,从而在有效利用内存资源的同时,保持虚拟机的相对独立和隔离。

附图说明

[0010] 图 1 示出现有技术中多个虚拟机的内存管理;

[0011] 图 2 示出根据本发明一个实施例的虚拟机管理装置的框图;

[0012] 图 3 示意性示出根据本发明一个实施例的多个虚拟机的软件层级信息;

[0013] 图 4 示意性示出根据本发明另一实施例的多个虚拟机的软件层级信息;

[0014] 图 5 示出根据本发明一个实施例的内存映射方式的示意图;

[0015] 图 6 示出根据本发明一个实施例的内存映射改变的示意图;以及

[0016] 图 7 示出根据本发明一个实施例的管理虚拟机的方法的流程图。

具体实施方式

[0017] 以下结合附图描述本发明的具体实施例。但是应该理解,以下对具体实施例的描述仅仅是为了解释本发明的执行示例,而不对本发明的范围进行任何限定。

[0018] 在本发明的多个实施例中,考虑同一物理平台上安装的虚拟机所具有的操作系统和应用软件,也就是虚拟机在各个层级上安装的软件信息,对于在同一层级具有相似软件的虚拟机实施内存的部分共享,从而在有效利用内存资源的同时,保持虚拟机的相对独立和隔离。

[0019] 图 2 示出根据本发明一个实施例的虚拟机管理装置的框图,该虚拟机管理装置位于物理平台与虚拟机之间,用以管理在该物理平台上安装的多个虚拟机。如图 2 所示,虚拟机管理装置 20 包括虚拟机信息获取单元 202,配置为获取当前虚拟机的软件层级信息;内存分析单元 204,配置为参照已安装的至少一个虚拟机的现有软件层级信息和所述当前虚

拟机的软件层级信息,将分配给当前虚拟机的逻辑内存划分为专用部分和共享部分;以及内存映射单元 206,配置为将上述共享部分映射到物理内存的共享区段,所述共享区段由所述已安装的至少一个虚拟机中的至少一个所使用。

[0020] 具体地,虚拟机信息获取单元 202 可以在各个虚拟机的安装过程中捕获每个虚拟机的软件信息。典型地,虚拟机的安装过程包括首先安装最底层的软件,即虚拟机所运行的操作系统,然后在操作系统之上安装各种软件应用。软件应用的安装也是分层级进行的,例如可以先在特定操作系统上安装 Java 虚拟机 (JVM),然后在 Java 虚拟机环境中安装 WAS(WebSphere Application Server),接着在 WAS 基础上可以安装更为上层的功能,例如 WPS(WebSphere Process Server)。一般来说,各个软件应用的安装文件会在文件头部中指明该软件应用的信息,例如名称、版本、发行公司、发行时间等等。因此,在各个层级的软件应用的安装过程中,虚拟机信息获取单元 202 可以通过读取安装文件的头部信息,获取所安装的软件应用的信息。

[0021] 对于当前安装的虚拟机,虚拟机信息获取单元 202 如上所述例如通过读取安装文件中的头部信息获取其软件信息,并结合软件所在的层级,得到当前虚拟机的软件层级信息。进一步地,虚拟机信息获取单元 202 将获取的当前虚拟机的软件层级信息传送到内存分析单元 204。

[0022] 对于已经安装的虚拟机,虚拟机信息获取单元 202 已经在其安装过程中获取其软件信息,进而得到每个已安装的虚拟机的软件层级信息。进一步地,虚拟机信息获取单元 202 可以对多个已安装的虚拟机的软件层级信息进行综合和比较,以各种形式维持已安装的虚拟机的软件层级信息。

[0023] 在一个实施例中,已安装的虚拟机的软件层级信息被组织为树形图的形式,如图 3 所示。具体地,图 3 示意性示出根据本发明一个实施例的多个虚拟机的软件层级信息。在图 3 的示意图中,示出了已安装的 4 个虚拟机的软件层级信息,其中,通过树形图中节点之间的继承关系示出各个虚拟机的软件层级信息之间的关系。根据树形图的特点,继承自某一父节点的子节点具有父节点的全部软件层级信息,相应地,继承自同一父节点的两个子节点共享父节点处的软件层级信息。更具体而言,虚拟机 1 只在操作系统上安装了 Java 开发工具包 (JDK),虚拟机 2 基于虚拟机 1 在 JDK 中安装了 WAS,而虚拟机 3 在虚拟机 1 的基础上安装了 WXS。进一步地,相比于虚拟机 2,虚拟机 4 在虚拟机 2 的基础上又安装了 WPS。虚拟机 2 和虚拟机 3 共同具有父节点处的虚拟机 1 的软件层级信息。通过维持这样的树形图,虚拟机信息获取单元 202 可以清楚地表示出各个虚拟机的软件层级信息之间的关系。

[0024] 在另一个实施例中,已安装的虚拟机的软件层级信息被组织为柱形图的形式,如图 4 所示。具体地,图 4 示意性示出根据本发明另一实施例的多个虚拟机的软件层级信息。在图 4 的示意图中,对于每个虚拟机,以柱形图的形式自下而上地示出在各个层级安装的软件的信息。例如,虚拟机 1 在 AIX 操作系统上安装了 IBM Java 虚拟机,在此之上又相继安装了 WAS 和 WPS;虚拟机 2 的软件层级信息与虚拟机 1 完全相同;虚拟机 3 在 Java 虚拟机上安装的是 WXS;而虚拟机 4 和 5 则是基于 Linux 操作系统,其上安装的 Java 虚拟机版本也不相同,一个来自 IBM,一个来自 Sun 公司。通过维持类似图 4 的柱形图,虚拟机信息获取单元 202 可以清楚地表示出每个虚拟机的具体软件层级信息。

[0025] 可以理解,可以以其他形式维持已安装虚拟机的软件层级信息,例如将其组织为

列表、文本形式,或以其他图表形式示出。本领域技术人员能够需要对软件层级信息的维持形式进行选择或修改,这样的选择或修改都在本发明的构思范围之内。

[0026] 接下来,内存分析单元 204 通过参照虚拟机信息获取单元 202 所传送的当前虚拟机的软件层级信息和所维持的已安装虚拟机的软件层级信息,来确定当前虚拟机的内存共享方式。具体地,在虚拟机安装过程中,会通过用户指定或系统默认,为每个虚拟机分配一段内存。这段内存存在虚拟机看来是专属于自己使用的内存,但是在虚拟机管理装置看来,只是虚拟机在逻辑上可以使用的内存大小,因此可以将其称为是逻辑内存。为了确定各个虚拟机分配得到的逻辑内存存在物理意义上的共享方式,内存分析单元 204 将当前虚拟机的软件信息与已安装虚拟机的软件层级信息进行比对,从而确定哪个已安装的虚拟机与当前虚拟机的软件层级结构最为接近。通过确定当前虚拟机与最接近的已安装虚拟机所共有的软件层级信息,内存分析单元 204 将分配给当前虚拟机的逻辑内存划分为专用部分和共享部分,其中共享部分与上述共有的软件层级信息相关联。

[0027] 具体地,在一个例子中,已安装虚拟机的软件层级信息如图 4 所示,其中示出了已经安装的 5 个虚拟机的软件层级信息。在安装新的虚拟机,即第 6 个虚拟机时,虚拟机信息获取单元 202 获取该第 6 虚拟机的安装信息,例如,所安装的操作系统为 Linux,操作系统上安装的是 IBM Java 虚拟机,其上安装的是 WXS。内存分析单元 204 通过将当前的第 6 虚拟机的软件信息与图 4 所示的已安装虚拟机的软件层级信息相比较,可以判断得出,已安装的虚拟机 4 所具有的软件层级结构与当前虚拟机最为相似,两者具有共同的操作系统和 Java 虚拟机。

[0028] 实际上,在操作系统维持基本运行环境所需要的内存中,有一部分内存用于维持操作系统所需的各种库函数等固定条目。操作系统通常只对这部分内存进行读取,而不对其进行修改或写入。因此,这部分内存可以在多个虚拟机之间进行共享,而不会妨碍虚拟机之间的隔离。对于软件应用来说,也存在类似的固定内存区段,可在多个虚拟机之间进行共享。因此,在内存分析单元 204 确定当前虚拟机与虚拟机 4 具有相同的操作系统和 Java 虚拟机的情况下,就可以基于操作系统 Linux 和 Java 虚拟机所具有的固定内存区段的大小,使得当前虚拟机与虚拟机 4 共享一部分内存。

[0029] 具体地,内存分析单元 204 可以将分配给当前虚拟机的逻辑内存划分为专用部分和共享部分,专用部分是当前虚拟机私有的内存,共享部分是可能与其他虚拟机共同使用的内存部分。一般来说,共享部分的大小关联于当前虚拟机与已安装虚拟机的共同软件部分的大小,以及该共同软件部分所占用的内存中可共享的固定内存区段的大小。在以上例子中,假定 Linux 操作系统所占用的内存中有 0.3G 内存是固定内存区段,Java 虚拟机所占用的内存中有 0.2G 内存是固定内存区段,那么当前虚拟机与虚拟机 4 可以共享 0.5G 的内存。相应地,如果分配给当前虚拟机的内存为 2G,那么当前虚拟机的专用内存部分为 1.5G。

[0030] 如上所述,内存分析单元 204 在逻辑层面确定了内存的共享方案。进而,通过内存映射单元 206 在物理层面实现上述内存共享。

[0031] 具体地,在每个虚拟机中都存在内存管理单元,用以维持虚拟机运行过程中所请求的内存的逻辑地址与物理地址之间的映射表。这个物理地址是虚拟机所能看到的、分配给自己的内存区段中的地址,也就是客户物理地址。然而,客户物理地址并不是硬件平台中的物理内存的真实物理地址,需要通过内存映射单元 206 进行进一步转换。具体而言,内存

映射单元 206 维持虚拟机的客户物理地址与物理内存的真实物理地址之间的映射关系,该映射关系可以以页表的形式存储。在获得虚拟机所请求的客户物理地址之后,内存映射单元 206 通过查阅所维持的页表确定对应的真实物理地址,从而使得虚拟机能够对该真实物理地址处的内存进行访问。为了实现内存的共享,内存映射单元 206 需要维持特殊的映射表,从而将多个虚拟机的客户物理地址映射到同一真实物理地址,由此实现物理内存的共享。

[0032] 图 5 示出根据本发明一个实施例的内存映射方式的示意图。在图 5 中,虚拟机 4 和当前虚拟机 6 都被分配了 2G 内存,各自都可以通过客户物理地址对分配给自己的内存进行访问。并且,虚拟机 4 和当前虚拟机 6 具有如上所述的软件层级信息,因此可以基于共同的操作系统 Linux 和同样版本的 Java 虚拟机而共享 0.5G 逻辑内存。对于虚拟机 4,内存映射单元 206 如通常方式将 0-2G 的逻辑内存的客户物理地址映射到真实的物理内存区段 S1-S3,其中 S1-S3 总大小为 2G,并且在 S1-S3 中,操作系统和 Java 虚拟机所占用的固定内存区段为 S2-S3,大小为 0.5G。对于当前虚拟机 6,已经在逻辑上将分配给它的 2G 内存划分为专用部分和共享部分,其中共享部分为 0.5G,对应于操作系统和 Java 虚拟机所占用的固定内存区段。因此,内存映射单元 206 将虚拟机 6 的 0.5G 共享部分的客户物理地址映射到物理内存的共享内存区段 S2-S3,该内存区段是虚拟机 4 已经使用的内存区段,将剩余的 1.5G 逻辑内存的客户物理地址映射到专用于虚拟机 6 的内存区段 S3-S4,其大小为 1.5G。这样,经过内存映射单元的映射,虚拟机 4 和虚拟机 6 共享了 0.5G 的内存。在这种情况下,两个虚拟机各自能使用的内存存在逻辑上依然是 2G,但是所占用的真实物理内存的总大小仅为 3.5G。

[0033] 为了记录内存区段的使用状态,在一个实施例中,内存映射单元 206 可以在现有的映射表形式的基础上添加一个状态位,用以标记相应内存区段的共享状态。例如,在一个例子中,以 S 标记共享的内存区段,以 P 标记专用的内存区段。在上述例子中,在内存映射单元 206 将虚拟机 6 的 0.5G 共享部分映射到内存区段 S3-S4 时,就可以将该段内存 S3-S4 的状态位由 P 改变为 S。为了保证共享的安全,在虚拟机访问物理内存时,仅允许对状态位为 S 的内存区段进行读取,而不能进行修改或写入。由此,即使虚拟机 4 崩溃,也不会影响内存区段 S3-S4 的数据,因此虚拟机 6 仍然可以照常运行而不会受到影响,从而保持了所需的独立性和隔离性。

[0034] 在以上实施例中,通过内存分析单元 204 和内存映射单元 206,虚拟机之间可以在多个层级进行内存的部分共享。例如,当前虚拟机与虚拟机 4 在操作系统层级上共享 0.3G 内存,在 Java 虚拟机层级上共享 0.2G 内存,这进一步提高了内存的使用效率,并使得共享方案更加灵活地适应于各种应用场景。

[0035] 不过,虚拟机之间的共享关系并不是恒定不变的,而是会随着虚拟机软件层级信息的改变而变化。仍然参照以上描述的例子,在物理平台上已安装图 4 所示的虚拟机 1-5 的情况下,对于当前安装的虚拟机 6,可以采用如图 5 所示的共享方式。接着,在虚拟机 6 安装完成之后,有可能用户卸载了虚拟机 4,释放虚拟机 4 所占用的内存。在此情况下,虚拟机信息获取单元 202 会更新其中维持的软件层级信息,将其转变为已安装的虚拟机 1-3,5-6 的软件层级信息。在一个实施例中,为了使得虚拟机 4 的内存释放不影响其他虚拟机的运行,内存映射单元 206 完全释放虚拟机 4 所占用的专用内存部分 S1-S2,而不释放虚拟机 4

与虚拟机 6 共享的内存区段 S2-S3,也不改变内存区段 S2-S3 的状态位 S。接着,如果在后续进一步安装新的虚拟机 7,则内存分析单元 204 类似地参照已安装的虚拟机 1-3,4-6 的软件层级信息确定新的内存共享方式。

[0036] 此外,在虚拟机的配置状态发生变化时,虚拟机的共享关系和共享方式也会发生改变。图 6 示出根据本发明一个实施例的内存映射改变的示意图,其中图 6A 示出初始映射关系,图 6B 示出改变后的映射关系。如图 6A 所示,与上述例子相一致,虚拟机 4 和虚拟机 6 各自有 0.5G 的逻辑内存被映射到物理内存的同一区段 S2-S3。这一共享内存区段 S2-S3 可能维持有与操作系统配置方式相关的数据。在某种情况下,如图 6B 所示,虚拟机 6 需要对自己的系统配置方式进行改变,因此需要对共享的逻辑内存中的子区段 m 进行修改。在接收到虚拟机 6 对内存区段 m 进行修改的请求后,内存映射单元 206 通过原有的映射表可以查阅到,该逻辑内存区段 m 对应于物理内存的共享区段 S2-S3 中的子区段 L,该子区段的状态位为 S,不允许修改。这时,内存映射单元 206 可以从空闲的物理内存中开辟出一段内存 L',将区段 L 中的数据复制到区段 L',将内存区段 L 和 L' 的状态位都设定为 P。进一步地,内存映射单元 206 修改映射表,使得虚拟机 6 的逻辑内存 m 对应到物理内存区段 L'。由此,虚拟机 6 通过对状态位为 P 的内存区段 L' 进行写入而实现了对逻辑内存区段 m 的修改。此外,内存区段 S2-S3 中除子区段 L 之外的剩余部分仍然由虚拟机 4 和虚拟机 6 所共享,而子区段 L 的状态变为虚拟机 4 所专有。通过上述过程,可以实现虚拟机对共享内存部分的修改而不影响其他虚拟机。

[0037] 如上所述,本发明实施例的虚拟机管理装置根据虚拟机的软件层级信息确定虚拟机之间的共享方式,使得不同虚拟机之间在多个层级都可以实现内存的部分共享。可以看到,虚拟机之间共享的内存部分越多,在物理平台上节省的内存就越多。同时,由于共享的内存部分是经过特殊选择的,并且不能被单个虚拟机修改,因此虚拟机之间仍然可以保持相对的独立和隔离,安全性不会受到影响。

[0038] 基于同一发明构思,本发明还提供了管理虚拟机的方法。图 7 示出根据本发明一个实施例的管理虚拟机的方法的流程图,该方法用以管理在同一物理平台上安装的多个虚拟机。如图 7 所示,根据一个实施例的虚拟机管理方法包括:步骤 701,在其中获取当前虚拟机的软件层级信息;步骤 702,在其中参照已安装的至少一个虚拟机的软件层级信息和所述当前虚拟机的软件层级信息,将分配给当前虚拟机的逻辑内存划分为专用部分和共享部分;以及步骤 703,在其中将上述共享部分映射到物理内存的共享区段,所述共享区段由至少一个已经安装的虚拟机所使用。

[0039] 在一个实施例中,步骤 701 可以包括,在当前虚拟机的安装过程中,通过读取安装文件的头部信息,捕获当前虚拟机的软件信息。在此基础上,结合软件所处的层级,可以获得当前虚拟机的软件层级信息。

[0040] 类似地,对于各个已安装的虚拟机,在其安装过程中都可以获得其软件层级信息。通过对多个已安装的虚拟机的软件层级信息进行综合和比较,可以将其维持为各种形式。在一个实施例中,已安装的虚拟机的软件层级信息被组织为树形图的形式。在另一个实施例中,已安装的虚拟机的软件层级信息被组织为柱形图的形式。可以理解,虚拟机的软件层级信息也可以以其他形式示出,具体例子如参照虚拟机管理装置的描述,在此不再赘述。

[0041] 接着,在步骤 702 中,通过参照已安装虚拟机的软件层级信息和步骤 701 中获取的

当前虚拟机的软件层级信息,来确定当前虚拟机的内存划分。具体地,在步骤 702 中,将当前虚拟机的软件信息与已安装虚拟机的软件层级信息进行比对,从而确定哪个已安装的虚拟机与当前虚拟机的软件层级结构最为接近。基于当前虚拟机与最接近的已安装虚拟机共同的软件层级信息,将分配给当前虚拟机的逻辑内存划分为专用部分和共享部分,其中专用部分是当前虚拟机私有的内存,共享部分是可能与其他虚拟机共同使用的内存部分。具体地,共享部分的大小关联于当前虚拟机与已安装虚拟机的共同软件部分的大小,以及该共同软件部分所占用的内存中可共享的固定内存区段的大小。

[0042] 为了在物理层面实现内存共享,在步骤 703 中,通过维持特殊的映射表,可以将共享部分的地址映射到物理内存的共享区段,该共享区段是其他虚拟机已经在使用的内存区段。也就是说,通过将多个虚拟机的逻辑内存映射到同一物理内存地址,来实现物理内存的共享。

[0043] 为了记录内存区段的使用状态,在一个实施例中,步骤 703 还可以包括,在映射表中包含状态位,用以标记相应内存区段的共享状态。对于共享状态的内存区段,虚拟机仅可以对其进行读取,而不能进行修改或写入。

[0044] 此外,虚拟机之间的共享关系并不是恒定不变的。在卸载虚拟机的情况下,虚拟机管理方法还包括释放内存的步骤(未示出)。在该步骤中,完全释放所卸载的虚拟机占用的专用物理内存部分。对于该虚拟机所占用的共享内存区段,仅在使用该内存区段的最后一个虚拟机卸载的同时释放。在虚拟机的配置状态发生变化时,虚拟机的共享关系和共享方式也会发生改变。具体地,在虚拟机想要修改逻辑内存中的共享部分的情况下,虚拟机管理方法还包括共享修改步骤(未示出),在其中,从空闲的物理内存中开辟出新的内存区段,将共享部分原始映射到的物理内存的共享区段中的数据复制到该新的内存区段,修改映射表,使得共享部分映射到该新的内存区段。由此,就可以对共享部分进行安全地修改而不影响其他虚拟机。具体修改过程的例子如参照虚拟机管理装置所描述,不再赘述。

[0045] 通过以上结合具体例子的描述可以获知,利用本发明实施例的虚拟机管理装置和方法,可以使得同一物理平台上安装的虚拟机实现内存的部分共享,同时不影响虚拟机的隔离和安全性。

[0046] 本领域技术人员可以理解,上述管理虚拟机的装置和方法可以使用计算机可执行指令和/或包含在处理器控制代码中来实现,例如在诸如磁盘、CD 或 DVD-ROM 的载体介质、诸如只读存储器(固件)的可编程的存储器或者诸如光学或电子信号载体的数据载体上提供了这样的代码。各实施例的装置及其单元可以由诸如超大规模集成电路或门阵列、诸如逻辑芯片、晶体管等的半导体、或者诸如现场可编程门阵列、可编程逻辑设备等的可编程硬件设备的硬件电路实现,也可以用由各种类型的处理器执行的软件实现,也可以由上述硬件电路和软件的结合实现。用于执行本发明的操作的软件和程序代码,可以用一种或多种程序设计语言的组合来编写,包括但不限于,面向对象的程序设计语言,诸如 Java, Smalltalk, C++ 之类,以及常规的过程式程序设计语言,诸如 C 程序设计语言或类似的程序设计语言。程序代码可以本地地或远程地在计算机上执行,以完成设定的操作。

[0047] 虽然以上结合具体实施例,对本发明的管理虚拟机的装置和方法进行了详细描述,但本发明并不限于此。本领域普通技术人员能够在说明书教导之下对本发明进行多种变换、替换和修改而不偏离本发明的精神和范围。应该理解,所有这样的变化、替换、修改仍

然落入本发明的保护范围之内。本发明的保护范围由所附权利要求来限定。

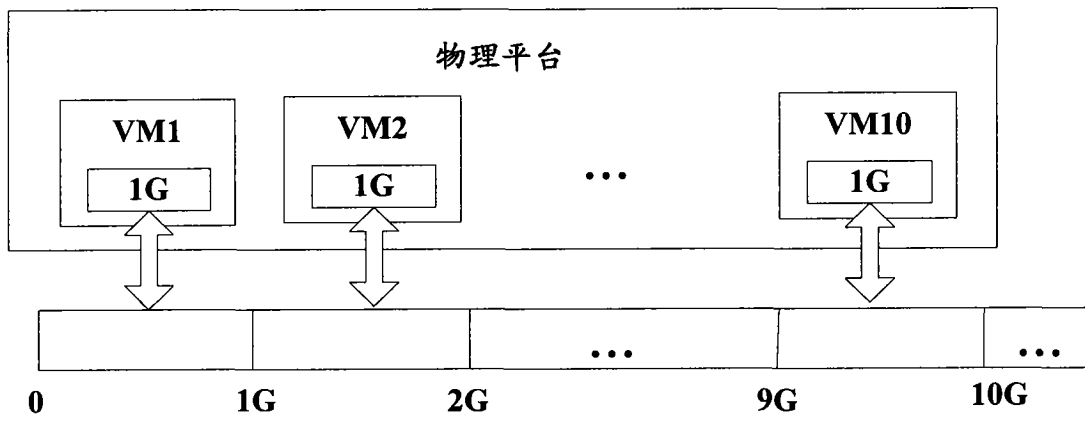


图 1

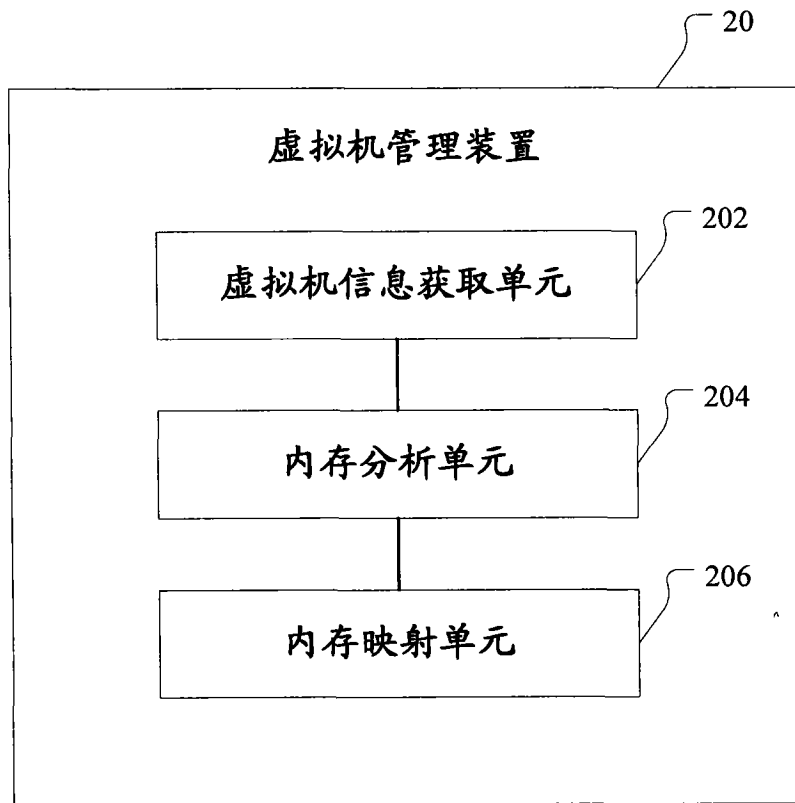


图 2

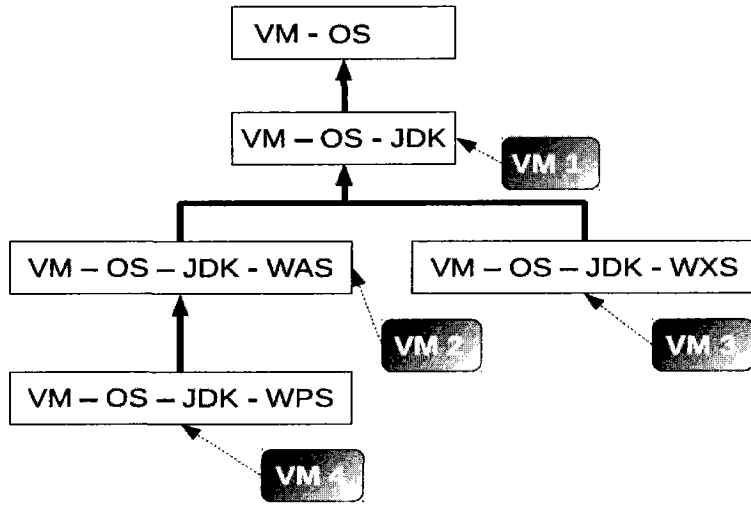


图 3

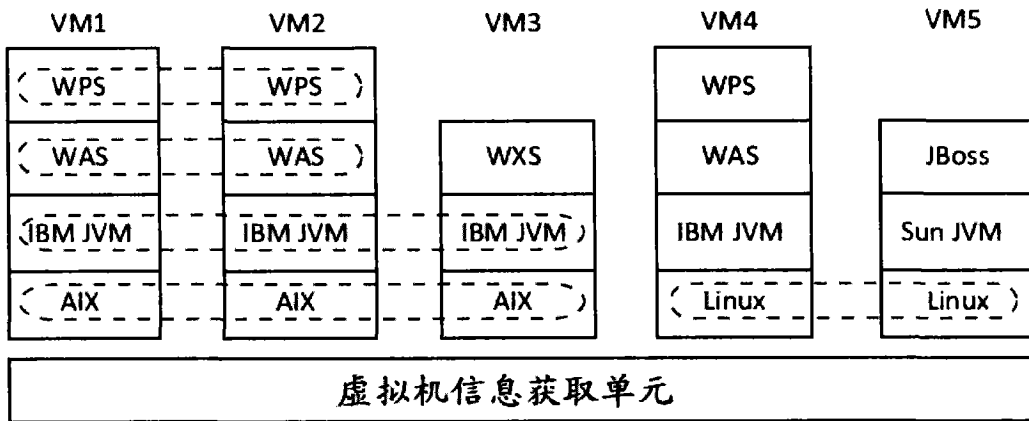


图 4

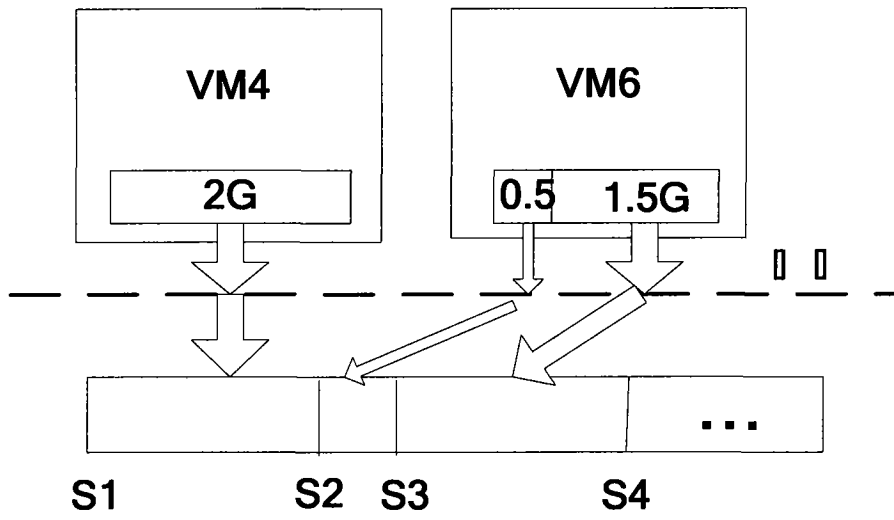


图 5

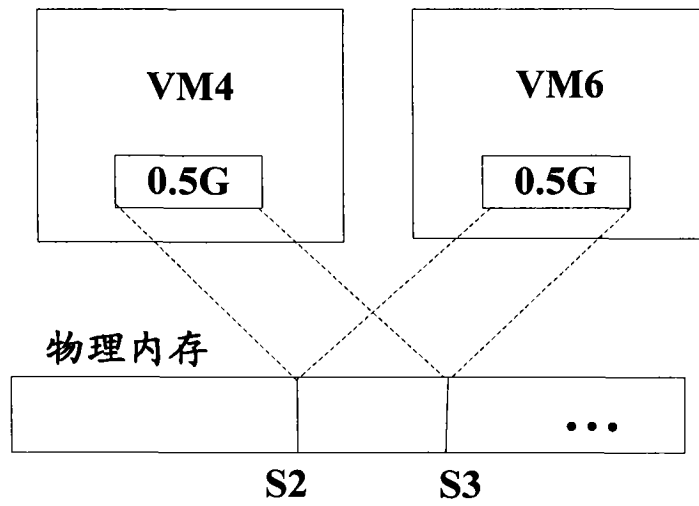


图 6A

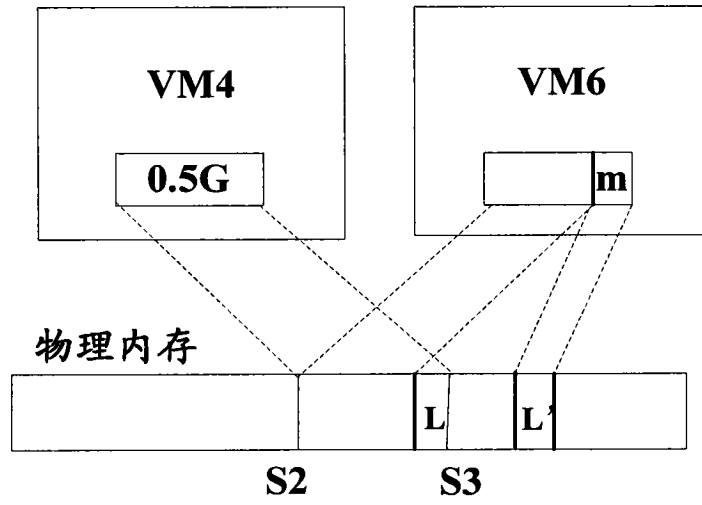


图 6B

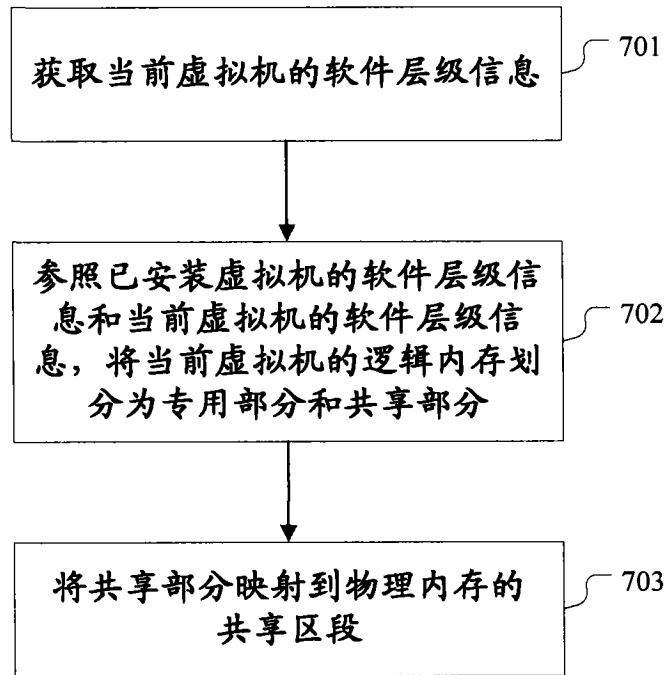


图 7