

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.
G06F 9/455 (2006.01)
G06F 17/30 (2006.01)



[12] 发明专利申请公布说明书

[21] 申请号 200810184903.4

[43] 公开日 2009年7月1日

[11] 公开号 CN 101470621A

[22] 申请日 2008.12.19
[21] 申请号 200810184903.4
[30] 优先权
[32] 2007.12.28 [33] US [31] 11/966,719
[71] 申请人 埃森哲环球服务有限公司
地址 瑞士沙夫豪森
[72] 发明人 刘 桓

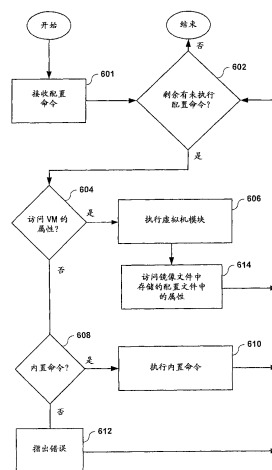
[74] 专利代理机构 北京市金杜律师事务所
代理人 朱海波 姜 彦

权利要求书 5 页 说明书 23 页 附图 6 页

[54] 发明名称
虚拟机配置系统

[57] 摘要

一种计算设备，使用配置命令来配置虚拟机。每个虚拟机可以从相应的虚拟机镜像文件来引导。在虚拟机引导或者执行时，虚拟机可以从相应的虚拟机镜像文件中所包括的应用文件和操作系统文件中进行读取或者向其进行写入。在虚拟机引导之前，通过从相应的虚拟机镜像文件中包括的应用文件和操作系统文件进行读取或者向其进行写入，计算设备可以得到或者设置将要在相应的虚拟机中引导的操作系统和应用的配置属性。计算设备可以执行配置命令来得到或者设置操作系统以及应用的配置属性。



1. 一种用于配置虚拟机的计算设备,所述计算设备包括计算机可读存储器和处理器,其中所述计算机可读存储器包括计算机代码和多个镜像文件,所述计算机代码包括:

多个虚拟机模块,至少一个所述虚拟机模块可以通过所述处理器来执行,以修改与镜像文件相关联的至少一个软件程序的配置属性,其中:

每个所述镜像文件包括多个文件集,每个文件集与计算机系统的软件程序唯一地关联;

所述镜像文件中包括的文件集与所述至少一个软件程序唯一地关联,并且包括所述配置属性;

每个所述镜像文件可引导为虚拟机,其被配置用于仿真所述计算机系统的硬件; 以及

所述至少一个所述虚拟机模块可执行以修改所述镜像文件,以修改包括在所述文件集中的所述配置属性。

2. 根据权利要求1所述的计算设备,其中,所述至少一个软件程序是应用,并且所述镜像文件中包括的另一文件集与另一软件程序唯一地关联,所述另一软件程序包括操作系统。

3. 根据权利要求1所述的计算设备,其中,所述计算机代码进一步包括:

容器模块,其可以通过所述处理器来执行,用以指引存储在所述计算机可读存储器中的多个配置命令的执行,其中,至少一个所述配置命令可以通过所述至少一个所述虚拟机模块来执行,以访问所述镜像文件,以访问所述至少一个软件程序的所述配置属性。

4. 根据权利要求3所述的计算设备,其中,所述至少一个所述配置命令所访问的所述配置属性可由随后执行的配置命令访问。

5. 根据权利要求1所述的计算设备,其中,每个所述虚拟机模块包括在所述镜像文件中一个不同的镜像文件中。

6. 根据权利要求5所述的计算设备,其中,每个所述镜像文件包括声明文件,所述声明文件被配置用于指定每个所述镜像文件中所包括的所述虚拟机模块可访问的所述配置属性的集合。

7. 根据权利要求3所述的计算设备,其中,所述配置命令包括移除命令,并且所述至少一个所述虚拟机模块可以执行以:响应于所述至少一个所述虚拟机模块从所述移除命令接收到预定值,而从所述镜像文件删除所述文件集。

8. 根据权利要求3所述的计算设备,其中,所述至少一个所述虚拟机模块包括获得器子模块,并且所述至少一个所述虚拟机模块可执行以利用所述获得器子模块来从所述镜像文件读取所述配置属性。

9. 根据权利要求3所述的计算设备,其中,所述至少一个所述虚拟机模块包括设置器子模块,并且所述至少一个所述虚拟机模块可执行以利用所述设置器子模块来设置所述镜像文件中的所述配置属性。

10. 根据权利要求9所述的计算设备,其中,所述至少一个所述虚拟机模块包括通知子模块,并且所述至少一个所述虚拟机模块可执行以:利用所述通知子模块来设置在已引导虚拟机中执行的所述软件程序的所述配置属性。

11. 根据权利要求3所述的计算设备,其中,所述容器模块可通过处理器来执行以:指引所述配置命令的第一子集与所述配置命令的第二子集并行执行。

12. 一种用于配置虚拟机的计算设备,所述计算设备包括:

存储器;

与所述存储器通信的处理器;

存储在所述存储器中的多个镜像文件,其包括第一镜像文件和第二镜像文件,其中:

所述第一镜像文件和所述第二镜像文件的每一个都包括多个软件程序,所述软件程序包括操作系统和至少一个应用;

所述处理器可执行以从至少一个所述镜像文件引导虚拟机;
以及

所述虚拟机被配置用于仿真计算机系统的硬件，以便在所述虚拟机中执行所述至少一个所述镜像文件的所述软件程序；以及多个配置命令，其包括第一配置命令和第二配置命令，每个所述配置命令存储在所述存储器中，并且可以通过所述处理器来执行，其中，所述第一配置命令可执行以从所述第一镜像文件读取第一软件程序的第一属性，并且所述第二配置命令可执行以向所述第二镜像文件写入以便将第二软件程序的第二属性设置为利用所述第一配置命令获取的所述第一属性。

13. 根据权利要求 12 所述的计算设备，其中，所述第一软件程序是所述第一镜像文件的所述操作系统，所述第二软件程序是所述第二镜像文件中包括的所述至少一个应用，并且所述第二配置命令可执行以将所述第二属性写入到所述第二镜像文件中包括的所述至少一个应用的文件。

14. 根据权利要求 12 所述的计算设备，进一步包括：

多个命令模块，其包括应用命令模块，每个所述命令模块存储在所述存储器中，并且可通过所述处理器来执行以实现至少一个所述配置命令，其中，所述应用命令模块被配置用于实现一个或多个所述配置命令，以访问所述第二软件程序的属性。

15. 根据权利要求 14 所述的计算设备，进一步包括：

容器模块，其存储在存储器中，并且可通过所述处理器来执行以：获取所述配置命令、将所述应用命令模块装载到所述存储器中，以及指引所述应用命令模块来执行所述一个或多个所述配置命令。

16. 根据权利要求 12 所述的计算设备，进一步包括解析器模块，其存储在所述存储器中，并且可通过所述处理器来执行以：解析所述配置命令的表示以识别每个所述配置命令、将所述配置命令编组为所确定的顺序，以及识别所述表示中的语法错误和语义错误。

17. 根据权利要求 12 所述的计算设备，其中，所述配置命令进一步包括实例化命令，其可执行以将标识符映射至所述第一镜像文件的副本；以及被配置用于参考所述标识符的任意一个或多个所述配置命

令可执行以访问所述副本中包括的、而不是所述第一镜像文件中包括的、所述软件程序的文件。

18. 根据权利要求 12 所述的计算设备, 其中, 所述配置命令进一步包括引导虚拟机命令, 其可执行以从所述至少一个所述镜像文件来引导所述虚拟机。

19. 根据权利要求 15 所述的计算设备, 其中, 所述配置命令进一步包括设置变量命令, 其可执行以将所述存储器中的变量设置为预定的值; 以及所述容器模块可执行以使得所述预定的值对于后续执行的配置命令可用。

20. 根据权利要求 12 所述的计算设备, 进一步包括已引导虚拟机, 其包含在所述存储器中并可由所述处理器来执行, 其中, 所述已引导虚拟机是从所述第二镜像文件被引导的; 在所述已引导虚拟机中, 代理模块存储在所述存储器中, 并且可通过所述处理器执行, 所述代理模块被配置用于设置在所述已引导虚拟机中执行的所述第二软件程序的第三属性; 以及所述配置命令进一步包括动态配置命令, 所述动态配置命令可执行以与所述代理模块通信, 并且进一步可执行以通过与所述代理模块通信来设置所述第三属性。

21. 一种用于配置多个虚拟机的方法, 虚拟机包括: 硬件仿真软件; 操作系统, 其被配置用于在所述硬件仿真软件上执行; 以及至少一个应用, 其被配置用于在所述操作系统内执行, 所述方法包括:

接收多个配置命令, 所述配置命令包括: 第一命令, 其可执行以读取第一虚拟机的第一属性; 以及第二命令, 其可执行以写入第二虚拟机的第二属性, 其中, 每个所述虚拟机包括与每个所述虚拟机的所述操作系统的以及所述至少一个应用的配置设置相对应的属性;

执行所述第一命令, 以读取所述第一属性;

使得由所述第一命令的执行而读取的所述第一属性对于随后执行的命令可用; 以及

执行所述第二命令, 其中, 所述第二命令的执行包括: 将所述第一属性写入到所述第二虚拟机的镜像文件中存储的配置文件中; 以及其

中，所述第二虚拟机的应用从所述配置文件读取相应配置设置。

22. 根据权利要求 21 所述的方法，其中，所述镜像文件是第一镜像文件，所述相应配置设置是第一相应配置设置，并且执行所述第一命令包括：从所述第一虚拟机的第二镜像文件中存储的操作系统配置文件读取所述第一属性，并且所述第一虚拟机的已配置操作系统从所述操作系统配置文件读取第二相应配置设置。

23. 根据权利要求 21 所述的方法，其中，所述镜像文件是第一镜像文件，所述相应配置设置是第一相应配置设置，并且执行所述第一命令包括：使用虚拟机管理系统来读取所述第一属性，并且所述虚拟机管理系统从所述第一虚拟机的第二镜像文件读取所述第一虚拟机的已配置操作系统的第二相应配置设置。

24. 根据权利要求 21 所述的方法，其中，所述镜像文件是第一镜像文件，所述第一属性是所述第一虚拟机的网络地址，并且执行所述第一命令包括：

从所述第一虚拟机的第二镜像文件获取所述第一虚拟机的唯一标识符；以及

与服务器通信，以根据所述唯一标识符来获取所述第一虚拟机的所述网络地址。

25. 根据权利要求 22 所述的方法，所述方法进一步包括：

将在所述第一虚拟机上操作的全部所述配置命令存储到包括在所述第一虚拟机的所述第二镜像文件中的文件中。

虚拟机配置系统

技术领域

本发明的领域总体上涉及软件配置。具体地，本发明涉及虚拟机的配置。

背景技术

正确地配置应用可能是费时费力的人工过程。正确地配置应用可能需要人们使用图形用户接口（GUI）来进行配置设置，其中轻微的错误便可能导致应用故障。一些应用的配置是非常复杂的，因为应用可能包括多个层级（tier）。层级可以定义为执行类似功能的一个或多个服务器的群组。例如，两层级应用可以包括配置 web 服务器和数据库服务器。三层级应用可以包括配置 web 服务器、应用服务器以及数据库服务器。每个层级可以包括多个机器，以处理负载平衡和机器故障转移。

每个服务器可能需要至少一些人工配置，并且动态地增大了复杂性和引入配置错误的可能性。典型的企业级应用，诸如客户关系管理（CRM）应用或者企业资源规划（ERP）应用，可能需要几天甚至几周的时间来安装和配置。

配置的难点之一涉及从层级到机器的映射。在开发期间，为了节省硬件成本，可能将所有层级映射至同一物理机器。在此典型配置中，每个层级中使用的所有应用都可以安装在同一物理机器上。相反，在产品配置中，每个层级可以映射至一个或者多个物理机器，以使可扩展性和可靠性最大化。在此典型的产品配置中，每个层级中的应用可以仅安装在属于该层级的每个机器上。由此，每个机器具有不同的配置。

虚拟化技术已经缓解了与一些类型的这些复杂应用相关联的一

些配置问题。虚拟机使用硬件仿真层，从而使多个操作系统可以在同一物理硬件上同时执行，因为硬件仿真层使得物理硬件看起来好像是每个操作系统的专用硬件。

虚拟机可以从镜像文件来引导。镜像文件可以是安装在计算机系统中软件（可以包括操作系统和各种应用）的镜像。有时，为了特定目的而安装在计算机系统中的软件的镜像文件公知为虚拟工具。例如，给定应用的特定层级中的软件可以安装在计算机系统中。从该计算机系统创建的镜像文件可以称为虚拟工具。随后可以使用镜像文件来引导属于该层级的虚拟机。

在典型的开发者配置中，每个层级的虚拟机可以在同一物理机器上引导，每个虚拟机从不同的镜像文件引导。备选地，在产品环境中，可以从同一镜像文件引导用于给定层级的多个虚拟机。每个虚拟机可以在独立的物理机器上运行。

尽管镜像文件可以包括预先安装的软件的集合，该软件的其他配置或者附加组件的安装常常还是留给用户来人工完成。其一个原因是：在创建镜像文件时，用以配置机器的某些信息是未知的。例如，网络设置可能仅当在从目标配置中的镜像文件引导了虚拟机之后才是已知的。一般地，如果应用需要使用多个虚拟机，所引导的虚拟机至少在一定程度上必须人工配置。

发明内容

一种计算设备，其使用配置命令来配置虚拟机。每个虚拟机可以从相应的虚拟机镜像文件来引导。在虚拟机引导或者执行时，虚拟机从相应的虚拟机镜像文件中所包括的应用文件以及操作系统文件进行读取或向其进行写入。在虚拟机引导之前，计算设备可以从相应的虚拟机镜像文件中所包括的应用和操作系统中读取或者向其写入，以此来得到或者设置将要在相应的虚拟机中引导的操作系统和应用的配置属性。该计算设备可以执行配置命令，以得到或者设置操作系统和应用的配置属性。

与从应用文件和操作系统文件进行读取或者向其进行写入相关联的逻辑可以封装在虚拟机模块中。虚拟机模块可以具有应用编程接口（API），其包括用于得到或者设置虚拟机的已命名属性的一个或多个方法。已命名属性可以称为虚拟机的配置属性。虚拟机的配置属性可以对应于该虚拟机的操作系统和应用的配置属性。

该计算设备可以执行配置命令，所述配置命令可执行以访问虚拟机的配置属性。该计算设备可以执行虚拟机模块的 API 以执行配置命令。可以使得配置命令所获得的虚拟机的配置属性对于一个或多个随后执行的配置命令可用。

配置命令可以对虚拟机镜像文件的副本、而不是虚拟机镜像文件本身进行参考。以此方式，可以使用相同的虚拟机镜像文件来引导不同的虚拟机，其中每个不同的虚拟机具有不同的配置。

可以通过 XML 来表示配置命令。可以将用以得到或者设置虚拟机属性的配置命令表示为单个 XML 元素。还可以将用以将虚拟机镜像文件映射为虚拟机标识符的配置命令表示为单个 XML 元素。还可以将用以引导给定虚拟机标识符的虚拟机的配置命令表示为单个 XML 元素。一个配置命令可以通过被表示为父 XML 元素的子 XML 元素，来将值传递给另一配置命令，其中该另一配置命令被表示为所述父 XML 元素。

该计算设备的令人感兴趣的方面在于：配置命令可以使用从一个虚拟机获得的配置属性来配置多个虚拟机，以设置另一虚拟机的配置属性。例如，一个配置命令可以得到一个虚拟机的操作系统的配置属性，并且使用该配置属性来设置不同虚拟机中的应用的配置属性。

该计算设备的另一令人感兴趣的方面在于：配置命令还可以配置已引导的虚拟机。例如，配置命令可以通过与已引导虚拟机内执行的代理模块进行通信来设置该已引导虚拟机的属性。

通过下文参考附图的描述，本发明的其他目的和优点将变得易见，其中附图清楚地示出了本发明的优选实施方式。

附图说明

参考下文的附图和描述将更好地理解本系统。图中的组件不一定是按比例绘制的，其重点强调的是示出本发明的原理。

图 1 是具有预先安装软件的示例计算机系统的图示；

图 2 是用于配置虚拟机的第一示例计算设备的图示；

图 3 是用于配置虚拟机的第二示例计算设备的图示；

图 4 是用于配置虚拟机的第三示例计算设备的图示；

图 5 是用于配置虚拟机的第四示例计算设备的图示；以及

图 6 是示出了使用配置命令来配置虚拟机的示例操作的框图。

具体实施方式

图 1 是具有预先安装软件的示例计算机系统的图示。示例计算机系统 100 具有已安装的操作系统 102 以及两个应用 104 和 106。操作系统 102 可以具有硬件驱动器 108，操作系统 102 使用该硬件驱动器 108 来与硬件设备通信。一个这样的硬件设备可以是存储设备 110。存储设备 110 可以是任何类型的存储器、存储器存储设备或者不同类型存储器或者存储器存储设备的组合，包括但不限于：随机访问存储器 (RAM)、只读存储器 (ROM)、硬盘、光盘、数字视频盘 (DVD)、压缩盘只读存储器 (CD-ROM) 等。

存储设备 110 可以包含操作系统文件 112 以及一个或多个应用文件 114 和 116。操作系统文件 112 中的一些可以是可执行的二进制码。当计算机系统 100 引导或者开始执行指令时，可以将可执行二进制码从存储在存储设备 110 上的操作系统文件 112 装载至存储器并执行，由此开始计算机系统 100 的操作。在向硬件加电之后，当计算机系统执行将该计算机系统置于常规操作配置所需的操作时，可以说该计算机系统进行引导（或者处于引导过程中）。当操作系统已经启动时，可以认为计算机系统处于常规操作配置中，此时应用可以在操作系统内执行。一旦应用可以在操作系统内执行，就认为操作系统和计算机系统已经经过了引导。在一些示例中，当应用可以在操作系统内执行

时，操作系统可能仍然处于启动过程中，在这样的例子中，认为操作系统和计算机系统已经经过了引导，即使操作系统仍然正在启动。应用是需要操作系统以便执行的任何软件，诸如 web 服务器、数据库服务器、文字处理器、网络驱动器、网络服务、浏览器插件等。软件程序可以是操作系统或者应用。如果可以从包含操作系统文件 112 的存储设备 110 来引导计算机系统，则认为该存储设备 110 是可引导的。

操作系统 102 在其引导时可以读取一个或多个配置文件，以读取存储在配置文件中的配置设置。例如，图 1 中示出了操作系统 (OS) 配置文件 118。OS 配置文件 118 可以是用于存储配置设置的任意格式，诸如：二进制、可扩展标记语言 (XML)、INI 格式 (也即，由符号 “=” 隔开的名称值对) 等。在所示的例子中，OS 配置文件 118 包含至少三个属性 120，其名称分别为 Property1、Property2 和 Property3，值分别是 A、B 和 C。在其他示例中，OS 配置文件 188 可以包括更多或者更少的属性，具有不同名称或者值并且不同格式的属属性。

在操作系统 102 引导时，操作系统 102 可能需要确定特定属性的值，诸如属性 “Property1”。操作系统 102 还可能需要在操作系统 102 引导之后确定特定属性的值。为了确定特定属性的值，操作系统 102 可以读取 OS 配置文件 118，并继而解析 OS 配置文件 118 的内容，以确定特定属性的值。取决于操作系统 102 和将要读取的特定属性，可以在操作系统 102 引导之前修改 OS 配置文件 118 中的属性值，以配置该特定属性。在一些情况下，为了改变操作系统 102 的一个属性，可能需要改变多个 OS 配置文件。

类似地，当应用 104 或者 106 在操作系统 102 内加载时，可以将应用文件 114 或者 116 中包含的该应用的可执行二进制码从存储设备 110 装载到存储器中，并继而执行。而且，当应用 104 或者 106 执行时，应用可以从应用配置文件 122 读取配置设置。

取决于应用 104 和 106 以及将要读取的特定属性 124，可以在应用 104 加载之前修改应用配置文件 122 中的属性值，以便配置应用的该特定属性 124。在一些情况下，为了改变应用 104 的一个属性，可

能需要改变多个应用配置文件 122。

如上所述，计算机系统 100 可以包括 (1) 操作系统 102 在其上执行的硬件，(2) 操作系统 102，以及 (3) 在操作系统 102 内执行的应用 104 和 106。相反，在此使用的术语“虚拟机”定义为软件，其包括 (1) 仿真软件，其对计算机系统 100 的硬件进行仿真，以及 (2) 操作系统 102，其被配置为在仿真软件上执行。短语“被配置为在仿真软件上执行”包括：操作系统 102 被配置为在计算机系统 100 的硬件上执行，但是其实际上是在仿真软件上执行。仿真软件可以包括硬件仿真驱动器，其对操作系统 102 用来与硬件设备通信的硬件驱动器 108 进行仿真。在此使用的术语“虚拟机”可以表示包括在操作系统 102 内执行的一个或多个应用 104 和 106 的软件。在此使用的术语“虚拟机”可以表示被配置为在仿真软件上执行的操作系统 102 以及在操作系统 102 内执行的一个或多个应用 104 和 106，而不是表示仿真软件本身。

计算机系统 100 的虚拟镜像文件可以包含操作系统文件 112 以及应用文件 114 和 116 的全部。虚拟镜像文件的格式可以是将要使用的虚拟化软件所特定的。可用的供应商虚拟化软件包括但不限于：Xen、VMware 以及微软的“Viridian”项目。虚拟镜像文件还可以包括除了操作系统文件 112 以及应用文件 114 和 116 之外的信息，诸如仅由供应商虚拟化软件所使用的信息。通过至少包括操作系统文件 112 以及应用文件 114 和 116，供应商虚拟化软件能够作为虚拟机来引导计算机系统 100 中最初所安装和配置的系统。当安装在虚拟机中的操作系统启动（响应于硬件仿真软件仿真对硬件加电）并且操作系统内的一个或多个应用可以执行的时候，认为该虚拟机进行了引导。能够从虚拟镜像文件来引导虚拟机的供应商虚拟化软件的组件将被称为虚拟机管理系统（VMMS）。虚拟机管理系统可以具有附加的功能，诸如查询功能。

在逻辑上，可以将每个虚拟机视为具有可访问（也即，读取或者修改）的属性集合。该虚拟机属性集合是操作系统的属性以及存储在

应用配置文件 118 和 122 中的应用的属性的集合或者子集。该虚拟机属性集合也可以包括逻辑属性。逻辑属性是表示虚拟机状态的属性，其中状态未必存储在应用配置文件中。例如，逻辑属性可以向供应商指明安装在虚拟机中的应用的类型。有两类虚拟机属性：静态的或者动态的。静态属性可以在引导虚拟机之前改变。动态属性可以在引导虚拟机之前和之后改变。虚拟机属性可以具有不同的数据类型。

图 2 是用于配置虚拟机的第一示例计算设备的图示。计算设备 200 可以包括操作系统 201。操作系统 201 可以在包括硬件仿真器 203 的虚拟机中运行，或者可以在不包括硬件仿真器 203 的非虚拟化计算机系统中运行。硬件仿真器 203 是上面讨论的仿真软件。容器模块 230 可以在操作系统 201 内执行。容器模块 230 可以获取配置命令 235、装载虚拟机模块 236 和 238，以及指引虚拟机模块 236 和 238 执行：以便根据一个或多个配置命令 235 来配置镜像文件 232 和 234。一个或多个配置命令 235 例如可以通过 XML 来表示。配置命令 235 可以包括用于得到和设置虚拟机操作系统 202 和/或虚拟机应用 204 的配置属性的命令。配置命令 235 可以指明配置命令是针对哪个虚拟机的，是 260 还是 262。

每个虚拟机 260 或者 262 可以对应于镜像文件 232 和 234。备选地，每个虚拟机 260 或者 262 可以对应于指定的镜像文件 232 或者 234 的副本。通过对镜像文件的副本进行操作，可以不同地配置基于相同镜像文件的两个或者更多虚拟机。

为了执行用于得到或者设置虚拟机 260 或者 262 的配置属性的配置命令，容器模块 230 可以执行与镜像文件 232 或者 234 相对应的虚拟机模块 236 或者 238。为了将配置属性设置为配置命令中所指定的值，虚拟机模块 236 或者 238 中的代码可以修改镜像文件 232 或者 234 中的适当配置文件。为了得到配置命令中所指定的配置属性，虚拟机模块 236 或者 238 中的代码可以解析镜像文件 232 或者 234 中的适当配置文件。

配置命令还可以包括用于引导虚拟机 260 或者 262 的一个或多个

配置命令。为了执行这样的引导配置命令，容器模块 230 可以调用虚拟机管理系统 240 (VMMS) 中的代码，并且容器模块 230 可以将对与虚拟机 260 或者 262 相对应的镜像文件 232 或者 234 的参考传递给 VMMS 240。对镜像文件 232 和 234 的参考可以是镜像文件 232 或者 234 的路径名称、网络地址、快捷方式、预定义别名或者任何其他标识符。可以执行 VMMS 240，以在操作系统当前执行的同一计算设备 200 上或者在计算设备 200 外部的机器上引导虚拟机 260 或者 262。当虚拟机 260 或者 262 引导时，将根据对镜像文件 232 或者 234 所进行的修改来配置虚拟机操作系统 202 和/或虚拟机应用 204。

图 3 是用于配置虚拟机的计算设备的第二示例的图示。在此示例中，计算设备 300 包括处理器 301 和计算机可读存储器 302。处理器 301 可以是能够执行模拟逻辑操作或数字逻辑操作的任何设备或者系统。存储器 302 可以是任何已知的或者稍后开发的、用于存储数据或代码的计算机可读介质，诸如随机访问存储器 (RAM)、只读存储器 (ROM)、硬盘、光盘、数字视频盘 (DVD)、压缩盘只读存储器 (CD-ROM) 等。计算机可读存储器 302 可以包括计算机代码和虚拟机镜像文件 332 和 334。计算机代码可以是以任意计算机语言编写的，诸如 C、C++、Java、C#、汇编语言、Perl 等。计算机代码可以（或者可以不）在操作系统内运行。计算设备 300 可以包括一个、两个或者多于两个虚拟机镜像文件 332 和 334。

计算机代码可以包括访问模块 336，其可以由处理器 301 来执行，用以访问每个虚拟机镜像文件 332 和 334 中所包括的一个或多个应用文件 340 以及操作系统文件 338。访问模块 336 可以是 VMMS 的部分。例如，访问模块 336 可以包括 VMMS 所提供的操作系统扩展，其将虚拟机镜像文件 332 或者 334 作为操作系统内执行的代码可访问的驱动或者目录而安装在操作系统内。在另一示例中，如果虚拟机镜像文件 332 和 334 是标准文件压缩格式的，诸如 ZIP、TAR、GZ 等，则访问模块 336 可以是能够处理这些标准文件压缩格式的代码。每个虚拟机镜像文件 332 和 334 可以是虚拟机中可引导的。供应商虚拟化软件

的具体内容可以确定是否使用 VMMS 来从虚拟机镜像文件引导虚拟机。

在图 3 示出的示例计算设备 300 中，计算机代码可以包括一个或多个虚拟机模块 341。每个虚拟机模块 342 可以由处理器 301 来执行，以访问至少一个虚拟机镜像文件 332 中所配置的应用和/或操作系统的配置属性。下文将给出虚拟机模块 341 如何操作的细节。

在图 3 所示的示例计算设备 300 中，计算机代码可以包括容器模块 330。容器模块 330 可以从存储在计算机可读存储器 201 中的文件处获取配置命令 344。该文件的格式可以是 XML 文本文件。备选地，该文件可以是目前已知的或者以后开发的、用于表示命令的任何格式，诸如 JavaScript、VBScript、Perl、各种构建文件语法等。配置命令 344 例如可被配置用于访问虚拟机镜像文件 332 或者 334 中所配置的应用和操作系统的配置属性。

容器模块 330 可以执行配置命令 344。容器模块 330 可以调用至少一个虚拟机模块 341 来访问虚拟机镜像文件 332 或者 334 中所包括的文件 338 和 340 中所存储的配置属性。

图 4 是用于配置虚拟机的计算设备 400 的第三示例的图示。图 4 中的计算设备 400 示出了虚拟机模块 442 是如何操作的。计算设备 400 包括处理器 401 以及处理器 401 可执行的容器模块 430。计算设备 400 还可以包括虚拟机模块 442 以及访问模块 436，二者都可以由处理器 401 来执行。计算设备 400 还可以包括虚拟机镜像文件 432。虚拟机镜像文件 432 可以包括虚拟机 (VM) 操作系统文件 438 以及一个或多个 VM 应用文件 440。配置属性 433 可以存储在应用文件 440 之一中。在其他示例中，计算设备 400 可以包括不止一个虚拟机镜像文件 432。

容器模块 430 可以获取配置命令 444，其包括用以访问存储在应用文件 440 之一中的配置属性 433 的一个或多个配置命令 444。容器模块 430 还可以装载虚拟机模块 442，并且指引虚拟机模块 442 执行以访问配置属性 433，而不是直接调用访问模块 436 或者直接访问应

用文件 440。容器模块 430 可被配置用于执行以预定编程语言集合之一编写的虚拟机模块 442。该预定集合可以是编程语言的任意集合，诸如 Java 和 C++。备选地，容器模块 430 可被配置用于以一个预定编程语言编写的虚拟机模块 442。编程语言可以是解释语言，诸如 Java、BASIC、C#等。当软件代码执行时，以解释语言编写的软件代码可以由语言特定的代码模块来解释。编程语言还可以是编译语言，诸如 C、C++、汇编语言等。可以首先利用称为编译器的代码模块对以编译语言编写的软件代码进行编译，以生成二进制可执行文件。该二进制可执行文件可以包括处理器 401 所理解的代码指令。在此使用的短语“通过处理器可执行”一般地适用于以解释语言或者编译语言编写的软件代码。

在此使用的术语“子模块”被定义为与方法、功能或者子例程等同。子模块可以（或者可以不）具有任何参数，并且可以（或者可以不）具有返回值。

虚拟机模块 442 可以外露应用程序接口（API）446，其包含用于通过名称来得到和/或设置属性的方法。继而可以将对存储配置属性 433 的具体应用或者操作系统文件进行定位的逻辑封装到该 API 的实现代码中。还可以将用于访问和解析特定应用或者操作系统文件的代码封装到虚拟机模块 442 的实现中。备选地，在不同的示例中，可以将用于访问和解析特定应用或者操作系统文件的代码封装到容器模块 430 中。在后面的例子中，虚拟机模块 442 中的代码对于所使用的供应商虚拟化软件而言可以是中立（neutral）的。

在一个示例中，API 446 可以包括具有值参（例如，“firstValue”）的设置器方法（例如“setFirstConfigurationProperty”）。该设置器方法可以具有前缀（例如，“set”），其连接有配置属性 433 之一的名称（例如，“FirstConfigurationProperty”）。可以执行设置器方法来将配置属性 433 设置为值参。在其他示例中，可以使用不同的命名惯例。在其他示例中，可以不遵循命名惯例。

API 446 可以包括具有返回值的获得器方法（例如，

“getFirstConfigurationProperty”)。该获得器方法可以具有前缀(例如,“get”),其连接有配置属性 433 之一的名称(例如,“FirstConfigurationProperty”)。可以执行获得器方法来返回配置属性 433 的值。在其他示例中,可以使用不同的命名惯例。在其他示例中,可以不遵循命名惯例。

API 446 中访问配置属性 433 的方法的实现可以执行访问模块 436 来访问存储在应用文件 440 之一中的配置属性 433。在一些示例中,API 446 中方法的实现可以仅仅是确定配置属性 433 存储在什么文件中、以及该文件的格式的代码。

具有获得器方法和/或设置器方法的 API 446 使得容器模块 430 能够将配置命令 444 转换为对虚拟机模块 442 的 API 446 中所包括的方法的调用。虚拟机模块 442 可以编写一次,并被随后编写配置命令 444 的多个不同的脚本开发者复用。

由于特定虚拟机镜像文件 432 上的应用文件 440 与被编写以访问这些文件的虚拟机 442 之间的紧密关系,可以将虚拟机模块 442 作为文件包括在该特定虚拟机镜像文件 432 中。容器模块 430 可以直接从虚拟机镜像文件 432 读取和装载虚拟机模块 442。

在一些示例中,容器模块 430 可以可编程地检查给定虚拟机模块 442 的 API 446,以确定给定的虚拟机模块 442 支持什么属性。例如,如果容器模块 430 和虚拟机模块 442 是通过 Java 编写的,则容器模块 430 可以使用 Java Reflection 来找到 API 446 中方法的名称,以确定可以得到或者设置什么属性。

备选地,每个虚拟机镜像文件 432 可以包括声明文件。声明文件可被配置用于指定该虚拟机镜像文件 432 中所包括的虚拟机模块 442 可以访问的配置属性。声明文件还可被配置用于指定 API 446 的方法。在另一示例中,API 446 的属性声明和/或方法声明可以存储在虚拟机镜像文件 432 中,但是当从虚拟机镜像文件来引导虚拟机 452 时,其存储在对于操作系统 456 或者应用 450 而言不可见的虚拟机镜像文件 432 的部分中。

在一个示例中，属性声明可以遵循编程语言（例如，C 编程语言）的语法。例如，可以使用 C 语言中用于声明变量的语法来声明配置属性；并且可以使用 C 语言中用于声明方法的语法来声明 API 446 的方法。每个方法声明还可以包括从虚拟机模块 442 的开始（其指明了 API 446 的相应方法的二进制实现的开始）的偏移。容器模块 430 可以使用该偏移来计算存储器中包含 API 446 的预定方法的二进制实现的地址。

通过抽象对访问模块 436 的访问逻辑，虚拟机模块 442 可以对所使用的供应商虚拟化软件中立，并且可以在不同的供应商虚拟化软件之间移动。此外，配置命令 444 对于所使用的供应商虚拟化软件而言也可以是中立的，因此可以在不同的供应商虚拟化软件之间移动。

还可以执行容器模块 430 来使得由配置命令 444 之一获取的配置属性对后续执行的配置命令可用。这例如允许将存储在一个虚拟机镜像文件 432 中的配置属性 433 设置为从在不同的虚拟机镜像文件中获取的另一配置属性获取的值。

在另一示例中，虚拟机模块 442 的 API 446 可以包括可执行以删除至少一个虚拟机镜像文件 432 中所包括的应用文件 440 的子集的方法。通过删除应用文件的子集，该方法可执行以从虚拟机镜像文件 432 处卸载一个或多个应用。

在另一示例中，虚拟机模块 442 的 API 446 可以包括通知方法。通知方法可以具有方法名称（例如，“notifySecondConfigurationProperty”），其形式是前缀（例如，“notify”）连接有配置属性 448 的名称（例如，“SecondConfigurationProperty”）。可以从调用通知方法的模块向该通知方法传递值参。可以执行通知方法，以便在从虚拟机镜像文件 432 引导虚拟机 452 之后将虚拟机镜像文件 432 中所配置的应用 450 的配置属性 448 设置为该值参。通知方法可能是有用的，因为虚拟机镜像文件 432 中的配置属性 433 在引导虚拟机 452 之后的改变可能并未被已引导虚拟机 452 内执行的应用 450 读取到。

虚拟机模块 442 的通知方法可以与在已引导虚拟机 452 中执行的代理 454 通信。在一个示例中，代理 454 可以存储在虚拟机镜像文件 432 中。在一些示例中，通知方法可以使用诸如 TCP/IP、HTTP、SOAP、SMP 等的标准协议来与代理 454 通信。在其他示例中，通知方法可以使用 VMMS 来与代理 454 通信。VMMS 可以提供以下能力：使用 VMMS 可编程地加载已引导虚拟机 452 中的进程。例如，如果已引导的虚拟机 452 使用寄宿（hosted）硬件仿真器在宿主操作系统中执行并且 VMMS 在该宿主操作系统中执行，则 VMMS 可以使用该功能。寄宿硬件仿真器是在宿主操作系统内运行、并且对硬件进行仿真的软件。“客户”操作系统 456 可以在寄宿硬件仿真器上运行，并由此在宿主操作系统内执行第二操作系统。

在一个示例中，容器模块 430 可以确定是调用 API 446 中包括的通知方法还是设置器方法。如果容器模块 430 执行设置配置属性 433 或者 448 的配置命令 444，则容器模块 430 可以在虚拟机 452 被引导的情况下调用通知方法，而在虚拟机 452 未被引导的情况下调用设置器方法。在另一示例中，容器模块 430 总是调用设置器方法，但是仅在虚拟机 452 被引导的情况下调用通知方法。在又一示例中，配置命令 444 本身指明是调用通知方法还是设置器方法。

在一个示例中，代理 454 可以调用虚拟机模块 442 中的方法来设置应用 450 的配置属性 448。在该示例中，虚拟机模块 442 可以附加地存储在应用文件 458 中。换言之，虚拟机模块 442 可以在客户操作系统 456 中运行，并由此被认为是应用——因此，虚拟机模块 442 可以恰当地包括在应用文件中。代理 454 可以是多个虚拟机模块 442 使用的通用代码。

可以执行容器模块 430 来将作为虚拟机模块 442 的 API 中所包括方法的值参而从配置命令 444 之一传递的文本值从文本数据类型转换为该值参的预定数据类型。该功能可以使得开发者更为容易地创建虚拟机模块 442，因为开发者可以使用 API 中所包括的方法中的适当数据类型，而无需编写代码来转换从配置命令 444 之一传递的文本值。

容器模块 430 也可以将使用预定数据类型的变量中的值保留为虚拟机模块 442 中的方法所返回的样子。容器模块 430 继而可以将该变量中保留的未转换值传递给虚拟机 442 中的方法（如果随后执行的配置命令 444 之一提出了这样的请求）。

图 5 是用于配置虚拟机的计算设备 500 的第四示例的图示。计算设备 500 可以包括处理器 501 和存储器 502。处理器 501 与存储器 502 通信。镜像文件 532 和 533 可以存储在存储器中。每个镜像文件（包括第一镜像文件 532 和第二镜像文件 533）可以是虚拟机镜像格式的，并且包括操作系统 538 的文件以及一个或多个应用 540 的文件。计算设备 500 还可以包括包含在存储器 502 中的虚拟机管理系统 504（VMMS）。

可以由处理器 501 来执行 VMMS 504，以便从虚拟机镜像格式的镜像文件 532 或者 533 来引导虚拟机 552。虚拟机 552 可被配置用于对计算机系统的硬件驱动器 558 进行仿真，以便使一个或多个应用 550 以及镜像文件 532 或者 533 的 VM 操作系统 556 能够在虚拟机 552 中执行。

计算设备 500 还可以包括包含在存储器 502 中的配置命令 544。配置命令 544 可以由处理器 501 来执行。可以从一个或多个文件中读取配置命令 544。在另一示例中，可以使用任何已知的或者以后发明的协议（诸如 TCP/IP、HTTP、SOAP 等）来通过网络或者计算设备 500 外部的一些其他位置来接收配置命令 544。

配置命令 544 可以包括第一配置命令。可以执行第一配置命令来得到第一镜像文件 532 的操作系统 538 的第一属性。第一配置命令可以使用操作系统命令模块 506 来获取第一属性。操作系统命令模块 506 可以是特定于特定类型的操作系统的，诸如 Windows、Linux、Macintosh 等。可以执行操作系统命令模块 506，以便例如从操作系统 538 的一个或多个文件定位给定的属性。

配置命令 544 可以包括第二配置命令，可执行该第二配置命令以设置第二镜像文件 533 的应用 535 的一个或多个文件中所包括的第二

属性 534。第二配置命令可以使用应用命令模块 508 来设置该属性。

计算设备 500 可以包括任意数目的命令模块，诸如操作系统命令模块 506 和应用命令模块 508。可以执行每个命令模块来实现至少一个配置命令。应用命令模块 508 可被配置用于实现被配置为访问应用 550 之一的属性的所有配置命令，其中，属性存储在第二镜像文件 533 中所包括的一个或多个应用 535 的文件中。实际上，如果同一应用包括在多个镜像文件 532 和 533 中，则可以使用相同的应用命令模块来访问该多个镜像文件 532 或者 533 的任意镜像文件中的该应用的属性。

计算设备 500 可以包括容器模块 530，其包含在存储器 502 中，并且可以由处理器 501 来执行。容器模块 530 可被配置用于执行与每个配置命令 544 相对应的命令模块 506 或者 508，其中命令模块实际上实现了配置命令。在一个示例中，每个镜像文件 532 和 533 可以包含任意操作系统命令模块 506 或者应用命令模块 508 的副本，所述模块可执行以访问存储在操作系统 536 或者 538 的文件中、或者存储在镜像文件 532 或者 533 中所包括的应用 535 或者 540 的文件中的属性。容器模块 530 可被配置用于从每个镜像文件 532 和 533 来读取每个命令模块 506 或者 508。

在另一示例中，计算设备 500 可以包括解析器模块，其存储在存储器 502 中并且可以由处理器 501 来执行，以从配置命令的表示（例如，XML）获得配置命令 544。解析器可进一步执行以识别每个配置命令 544 以便将配置命令编组为所确定的顺序，以及识别该表示中的语法错误和语义错误。示例语义错误是访问不存在的虚拟机属性。在错误的情况下，解析器模块可以指明该错误和/或停止配置命令的执行。解析器模块可以包括在容器模块 530 中。

可以采用任何形式的 XML 表示。下面给出了具体配置命令的一些示例形式，不应将其理解为限制可能使用的形式。

配置命令 544 可以包括实例化命令，其可执行以将标识符映射到指定镜像文件 532 或者 533 的副本。标识符可以是唯一标识虚拟机（其

可被引导或者不可引导)的任何文本(包括数字或者符号)。在一个示例中,实例化命令可以表示为如下 XML 命令:

```
<VM id="dbserver" image="c:\ubuntu.vmx"/>
```

该 XML 元素的“image”(镜像)属性指明了将要使用的基本虚拟机镜像的位置。“id”属性指明了虚拟机标识符的名称,随后执行的配置命令可以参考该名称来访问该虚拟机的配置属性。在一个示例中,实例化命令可以制作镜像属性中指定的基本虚拟机镜像文件的副本,并且参考该虚拟机的所有后续配置命令可以访问该副本。为了效率原因,可能只需要复制镜像文件的一部分。备选地,参考该虚拟机标识符的后续配置命令可以访问镜像属性中所指定的镜像文件而不是该镜像文件的副本。

可执行以得到第一镜像文件 532 或者 533 的操作系统的第二属性的第二配置命令可以表示为如下 XML 命令:

```
<property-path id="dbserverip" path="dbserver.ipaddr"/>
```

<property-path/>XML 元素的“path”(路径)属性的值可以指明第一属性,并且可以遵循以下形式:“虚拟机标识符.属性名称”。“id”属性可以指明可被随后执行的配置命令参考的变量名称,其中随后执行的配置命令可以获取第一属性的值。“id”属性可以是可选的。在一个示例中,在上述 XML 命令的执行期间,容器模块 430 可以利用第一属性的值来替换<property-path/>XML 元素。

一个配置命令可以通过表示为子 XML 元素来将值传递给另一配置命令,其中该另一配置命令表示为父 XML 元素。例如,可执行以将第二镜像文件 533 的应用 535 的文件之一的文件中的第二属性 534 设置为第一配置命令获得的第一属性的第二配置命令可以表示为以下 XML 命令:

```
<VM id="webserver">
```

```
<property name="databaseIpAddress">
```

```
<variable id="dbserverip"/>
```

```
</property>
```

```
</VM>
```

具有“id”属性的<VM>XML 元素可以指定将要配置的虚拟机的标识符。在此示例中，一些之前的实例化命令可能已经将“webserver”标识符映射到了特定的镜像文件。<property>XML 元素指明：其后跟随的命令对“name”（名称）属性所标识的属性进行操作。在此示例中，属性名称是“databaseIpAddress”。因为<property>XML 元素是<VM>元素的子元素，因此所标识的属性是<VM>元素中标识的虚拟机的属性。因此，上面的示例 XML 表示利用标识符“webserver”来设置虚拟机的“databaseIpAddress”属性的命令。<variable>元素赋值为“id”属性所指明的变量的值。因为<variable>XML 元素是<property>XML 元素的子元素，因此该 XML 表示将该属性设置为变量值的命令。在此示例中，XML 表示利用“webserver”标识符将虚拟机的“databaseIpAddress”属性设置为“dbserverip”变量。

取代使用变量，上面讨论的第一配置命令和第二配置命令可以表示为如下 XML 命令：

```
<VM id= “webserver”>
```

```
  <property name= “databaseIpAddress”>
```

```
    <property-path path= “dbserver.ipaddr”/>
```

```
  </property>
```

```
</VM>
```

<property-path> XML 元素得到具有“dbserver”标识符的虚拟机的“ipaddr”属性的值。<property> XML 元素将具有“webserver”标识符的虚拟机的属性“databaseIpAddress”设置为从得到“ipaddr”属性获得的值。取代使用<property-path> XML 元素，可以将 property-path 属性用于相同的效果，但是具有更简练的表示：

```
<VM id= “webserver”>
```

```
<property name= "databaseIpAddress" property-path="dbserver.ipaddr"/>
</VM>
```

配置命令还可以指定常量值。例如，下面的命令将“webserver”所标识的虚拟机的“databaseIpAdderss”属性设置为值“192.168.33.190”。

```
<VM id= "webserver">
  <property name="databaseIpAddress">
    <value>192.168.33.190<value/>
  </property>
</VM>
```

备选地，可以使用<property> XML 元素上的“value”属性来取代子<“value”> XML 元素。在另一示例中，下面的配置命令是设置变量命令，其可以将存储器 201 中的变量设置为指定值，以便随后执行的配置命令通过参考该变量来使用：

```
<variable id="dbserverip" value="192.168.33.190"/>
```

属性“value”包含特定的值。可以对“id”属性的值进行参考。例如，随后执行的配置命令可以利用上面示例之一中指出的 XML 元素<variable id="dbserverip"/>来对示例变量进行参考。

配置命令还可以包括连接命令，其可执行以连接两个或者更多值。在其他示例中，可以支持更为精细的字符串操作。例如，下面以 XML 表示的连接命令将两个变量的值“firstValue”和“secondValue”连接为存储在称为“largerstring”中的字符串。

```
<concatenate id= "largerstring">
  <variable id="firstValue"/>
  <variable id="secondValue"/>
</concatenate>
```

配置命令还可以包括引导虚拟机命令，其可执行以从镜像文件

532 或者 533 引导指定的虚拟机 552。例如，引导虚拟机命令可以表示为：

```
<VM id="webserver" action=poweron/>
```

“id”属性可以指明实例化命令中先前指定的虚拟机的标识符。“action”属性可以指明对指定虚拟机的执行的动作。在此示例中，“poweron”指明，应当从实例化命令中先前指定的镜像文件 532 或者 533 引导虚拟机 552。备选地，可以在<VM> XML 元素上的“image”属性中指定镜像文件，以用于实例化和引导虚拟机 552。“action”属性可以指定其他动作，诸如虚拟机 552 关闭。

容器模块 430 可以按照容器模块 430 获取配置命令 444 的顺序所指明的顺序来指引配置命令 444 的执行。备选地，配置命令 444 可以包括改变执行顺序的逻辑命令。

容器模块 430 可以串行地执行配置命令 444。在不同示例中，容器模块 430 可以并行执行配置命令 444。如果在配置命令 444 中没有指明两个虚拟机之间存在依赖关系，则在两个虚拟机之一上进行操作的配置命令 444 可以与在两个虚拟机中另一个上执行的配置命令 444 并行执行。指明两个虚拟机之间的依赖关系的一个示例可以是显式地这样做，配置命令 444：

```
<VM id="webserver" image="c:\webserver.vmx" depends-on="dbserver" />
```

在一些示例中，容器模块 501 可以处理<include> XML 元素或者 XML 处理器指令，以包括 XML 文件。在这些示例中，仅对某些应用或者虚拟机进行操作的配置命令可以包含在专用文件中。单个 XML 文件继而可以包括多个专用文件。

图 5 所示的示例计算设备 500 包括已引导的虚拟机 552，其包含在存储器 502 中，并且由处理器 501 来执行。可以从镜像文件 532 或者 533 之一来引导已引导虚拟机 552。已引导虚拟机 552 可以包括已引导虚拟机（BVM）操作系统 556 以及一个或多个 BVM 应用 550。已引导虚拟机 552 还可以包括代理模块 554。

已引导虚拟机 552 可以访问 BVM 操作系统 560 的文件以及一个或多个 BVM 应用 562 的文件。BVM 操作系统 560 的文件可以不是操作系统 536 或者 538 的文件的相同副本，并且一个或多个 BVM 应用 562 的文件可以不是镜像文件 532 或者 533 中存储的一个或多个应用 535 或者 540 的文件的相同副本。例如，如果 BVM 操作系统 560 的文件与一个或多个 BVM 应用 562 的文件是不同的，则访问第二属性 534 的任何配置命令 544 将不需要访问一个或多个 BVM 应用 562 的文件中所包括的第三属性 564 的值，即使第二属性和第三属性的名称是相同的，并且虚拟机 552 是从第二镜像文件 533 引导的。

为了防止该问题，应用命令模块 508 可以与虚拟机 552 中执行的代理模块 554 通信。代理模块 554 可被配置用于访问在已引导虚拟机 552 中执行的应用 550 之一的第三属性 564。代理模块 554 可被配置用于得到或者设置第三属性 564。

如果配置命令 544 包括动态配置命令，该命令可以执行应用命令模块 508，其中应用命令模块 508 可以与代理模块 554 通信以访问第三属性 564。在一些示例中，动态配置命令还可以修改第二镜像文件 533 中的第二属性 534。

图 6 是示出了使用配置命令来配置虚拟机的示例操作的流程图。如前所述，虚拟机可以包括：硬件仿真软件；操作系统，其被配置用于执行硬件仿真软件；和至少一个应用，其被配置用于在操作系统内执行。此外，虚拟机具有与操作系统以及至少一个应用的配置设置相对应的属性。

例如，操作系统的配置属性可以是网络地址配置设置。在该示例中，具有该操作系统的虚拟机的属性是网络地址，其对应于网络地址配置设置。类似于操作系统，诸如 web 服务器的应用可以具有配置设置，诸如监听地址。在该示例中，具有该 web 服务器应用的虚拟机的属性是 web 服务器监听地址，其对应于 web 服务器应用的监听地址配置设置。

在块 601 处，操作开始，接收诸如 XML 配置命令的配置命令。

可以从网络接收配置命令。可以使用诸如 TCP/IP、HTTP、SOAP、专利技术等协议来接收命令。备选地，接收配置命令可以包括从文件读取配置命令。配置命令可被配置用于访问虚拟机的静态属性。在一些示例中，配置命令还可被配置用于访问虚拟机的动态属性。

一些示例中的操作可以包括：对在特定虚拟机上进行操作的所有配置命令存储到该虚拟机的镜像文件中所包括的文件中。继而，当在块 601 接收到配置命令时，从镜像文件中读取在特定虚拟机上操作的配置命令。

在此示例中，执行配置命令至少可以发生在块 602、604、606 和 608。操作可以包括在块 602 检查任何未执行的剩余配置命令。如果没有未执行的剩余配置命令，则操作完成。

然而，如果在块 602 检测到了剩余的未执行配置命令，则操作可以包括在块 604 处检查将要执行的下一命令是否访问了虚拟机 (VM) 的属性。如果下一命令访问了虚拟机的属性，操作可以进行到在块 606 处执行虚拟机模块。

然而，如果下一命令没有访问虚拟机的属性，操作可以进行到在块 608 处检查内置命令模块以执行下一命令。内置命令模块可以独立于虚拟机的内容而实现。例如，可以将引导虚拟机实现为内置命令模块。

如果针对下一命令的内置命令模块存在，操作可以进行到在块 610 处执行该内置命令模块，并且返回块 602。如果针对下一命令不存在内置命令，可以在块 612 指出错误，并且操作返回块 602 以检查任何剩余的未执行命令。在块 604 处的检查和在块 612 处的检查可以反序执行。

如上所述，操作可以进行到在块 606 处执行虚拟机模块。在块 606 处，操作可以包括确认虚拟机模块支持访问下一命令中指定的属性。操作可以包括执行支持访问下一命令中指定的属性的虚拟机模块上的方法。操作还可以包括使得该方法返回的属性值对后续执行的命令可用。

执行虚拟机模块上的方法可以包括在块 614 访问存储在镜像文件中的配置文件中的属性。例如，当下一命令是读取第一虚拟机的第一属性、并且第一属性是静态属性的情况下，操作可以包括从存储在第一虚拟机的第一镜像文件中的操作系统配置文件读取第一属性。第一属性可以是第一虚拟机的属性，因为操作系统具有存储在操作系统配置文件中的第一对应配置设置。

在另一示例中，当下一命令是可执行以写入第二虚拟机的第二属性的命令、并且第二属性是静态的情况下，操作可以包括将第二属性写入到存储在第二虚拟机的第二镜像文件中的应用配置文件。第二属性可以是第二虚拟机的属性，因为第二虚拟机的应用具有存储在应用配置文件中的对应配置设置。

在其他示例中，在块 606 处在虚拟机模块上执行方法可以不包括直接访问在块 614 中指明的配置文件中的虚拟机属性。例如，访问与操作系统设置相对应的虚拟机的属性可以包括：使用虚拟机管理系统（VMMS）来获取该属性。VMMS 可以能够从镜像文件读取特定的操作系统属性，诸如媒体访问控制地址（MAC 地址）或者以太网硬件地址（EHA）或者硬件地址或者适配器地址。

在又一示例中，在块 606 处在虚拟机模块上执行方法可以包括访问网络服务。例如，如果上面提到的第一属性是第一虚拟机的网络地址，可以在第一虚拟机引导时动态地获取该网络地址。然而，可能期望的是：在引导第一虚拟机之前访问网络地址属性。例如，如果在引导第一虚拟机之前要使用网络地址来设置第一虚拟机的其他属性，则可能期望的是：在引导之前访问网络地址。

在该示例中，访问网络地址属性可以包括访问网络服务。在块 606 处在虚拟机上执行方法可以包括：首先从第一虚拟机的第一镜像文件获取第一虚拟机的唯一标识符。该唯一标识符可以是 MAC 地址、主机名、或者能够从第一镜像文件获取的任何其他唯一标识符。操作继而可以包括与诸如动态主机配置协议（DHCP）的网络服务进行通信，以根据该唯一标识符来动态获取网络地址。

通过使用虚拟机配置系统，可以在很少或者没有人工配置的情况下快速配置虚拟机。只要编写了配置命令，便可以按照多种不同方式来配置虚拟机，而只需对配置命令进行很小的修改或者无需修改，并且需要很少的或者不需要其他人工配置。

尽管已经参考具体的示例性实施方式描述了本发明，但是显而易见，可以对这些实施方式进行各种修改和改变，而不脱离权利要求所记载的本发明的较宽泛的精神和范围。因此，应当将说明书和附图认为是示例性而非限制性意义的。

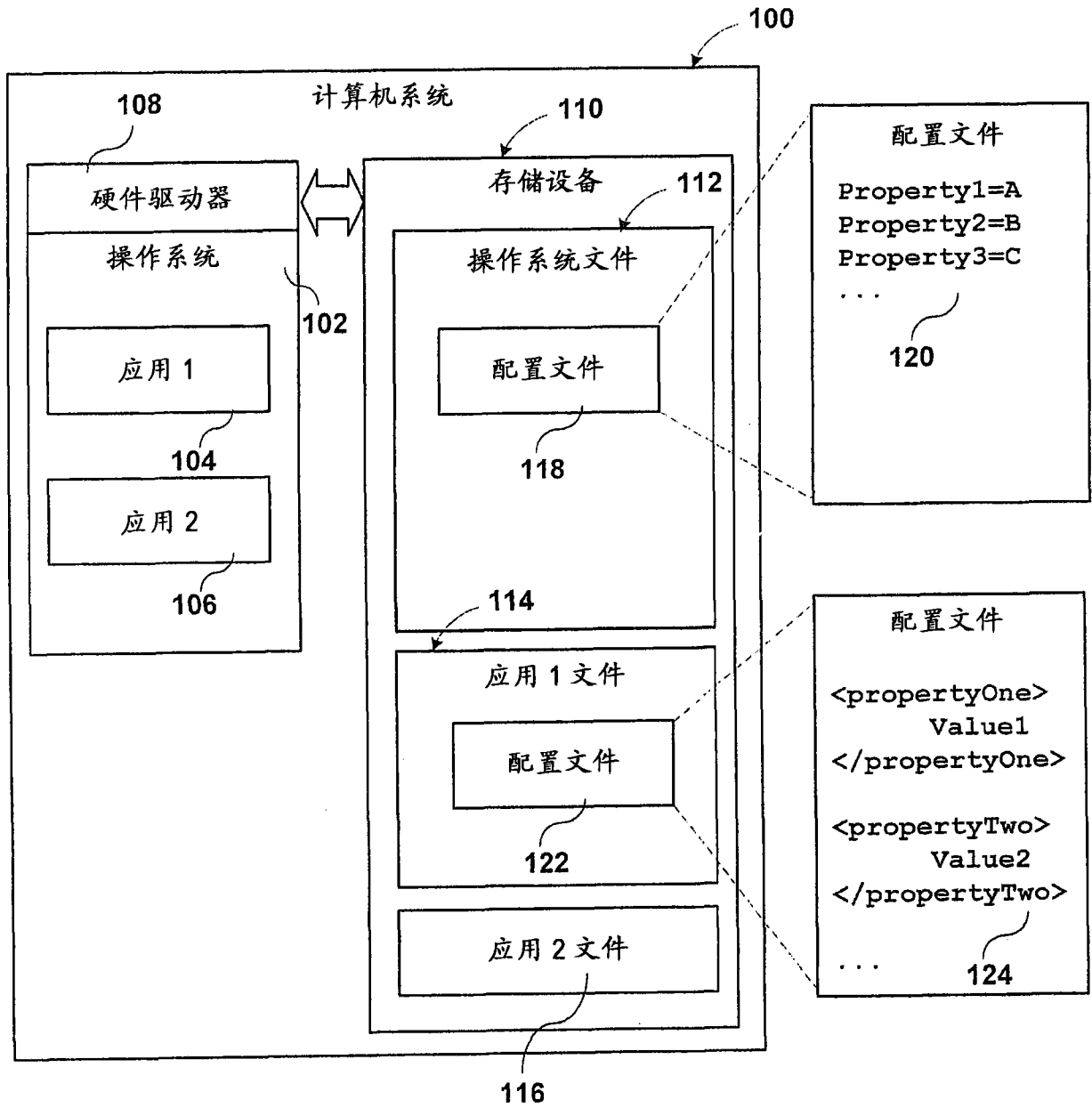


图 1

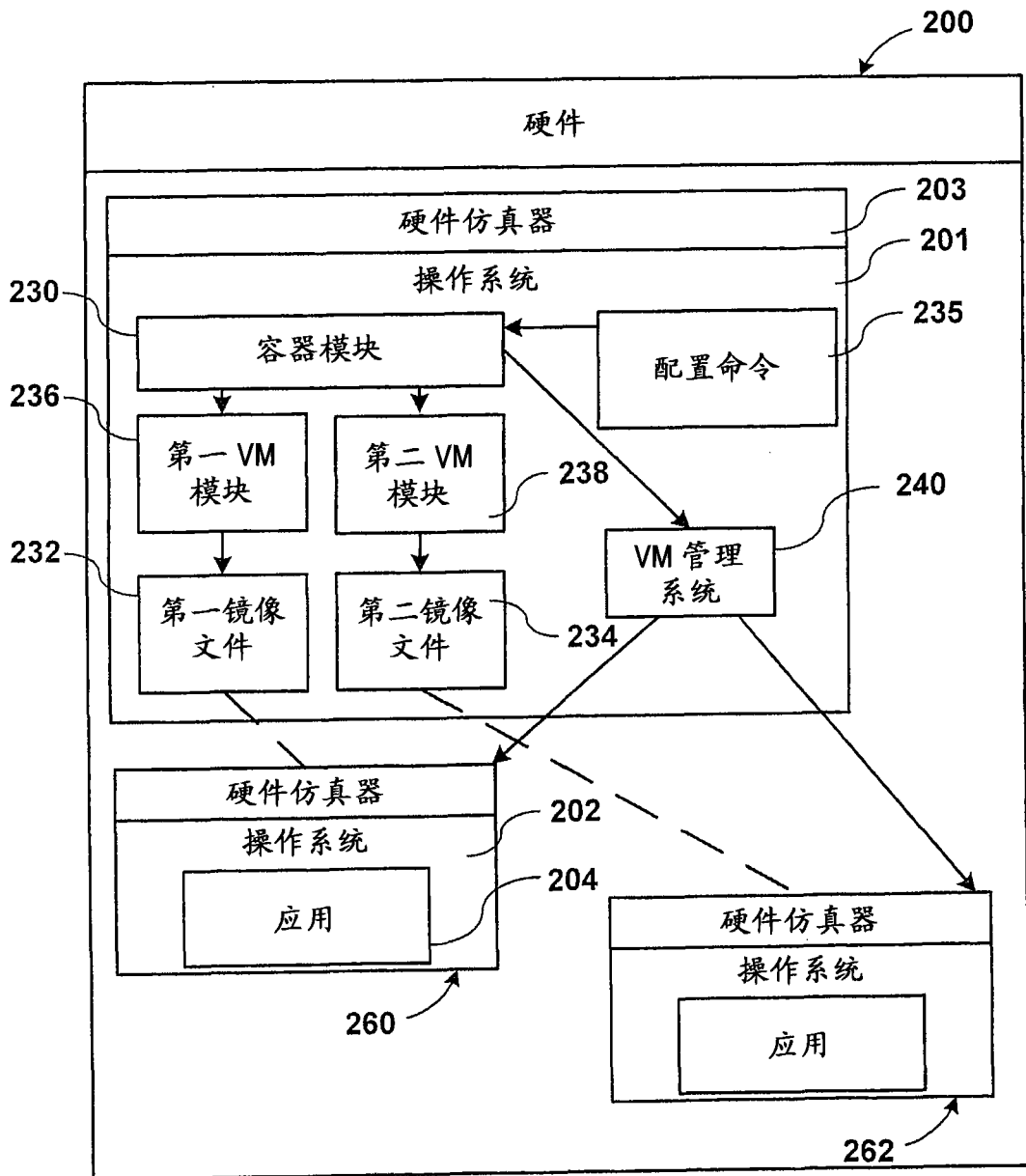


图 2

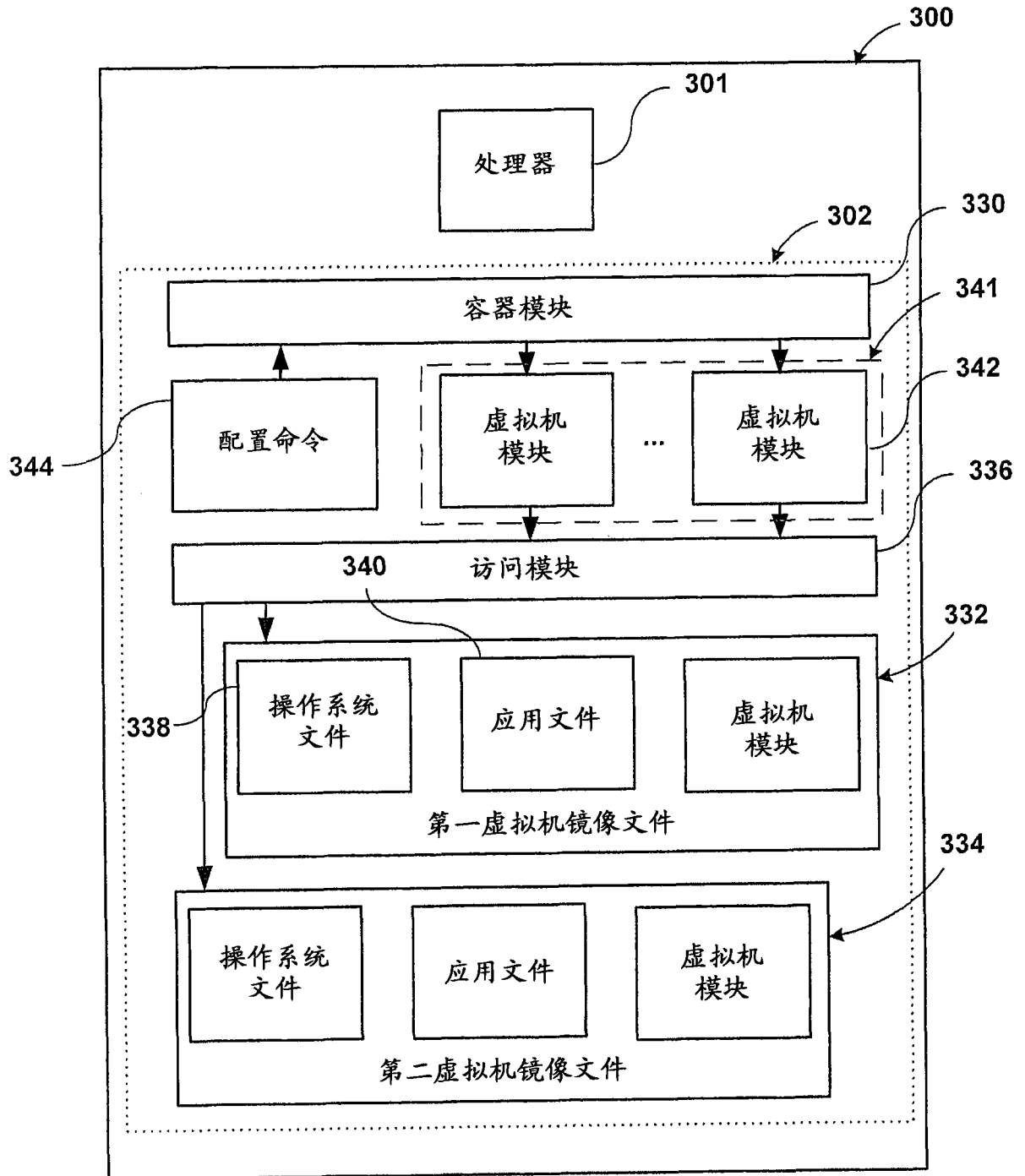


图 3

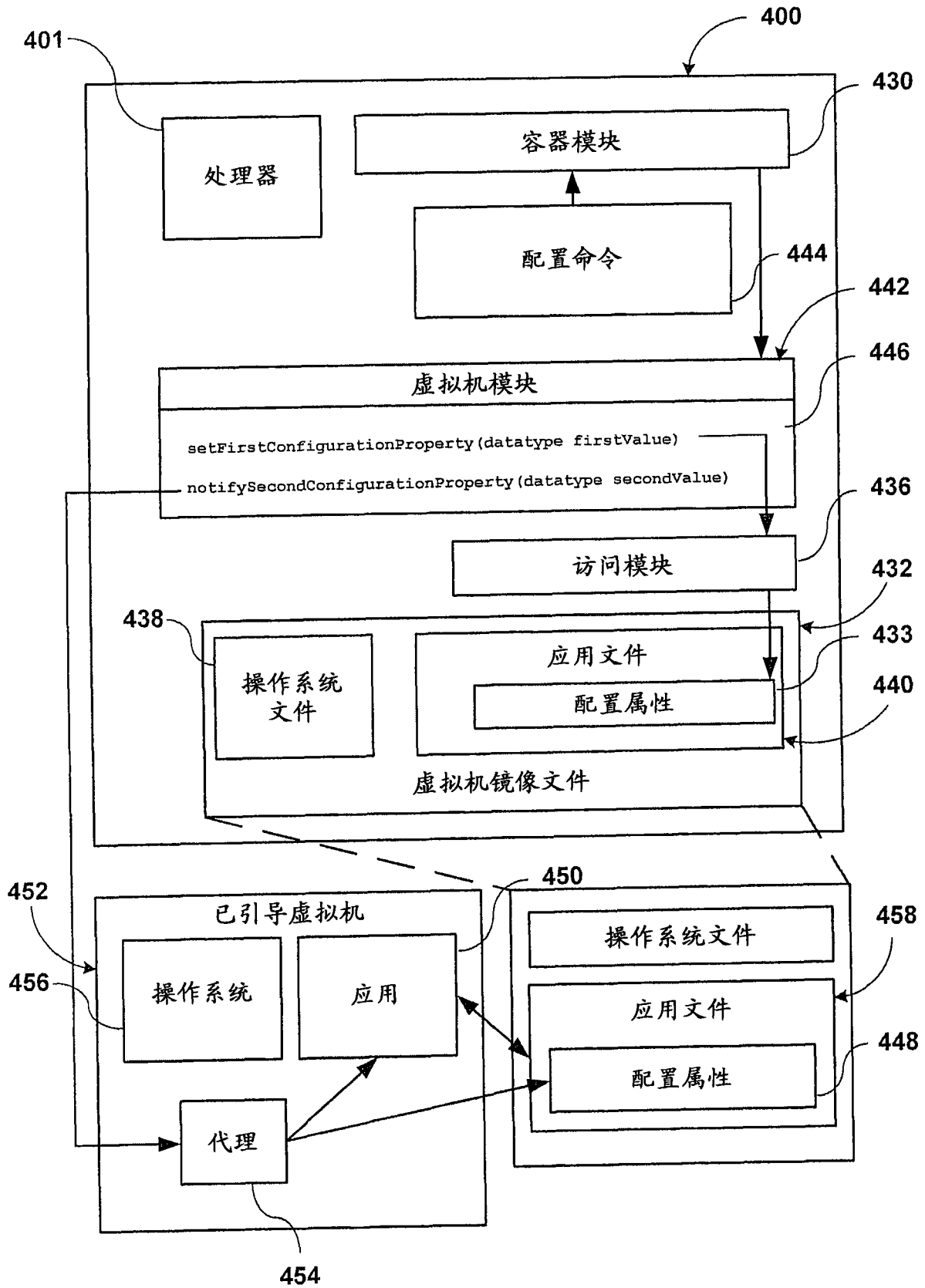


图 4

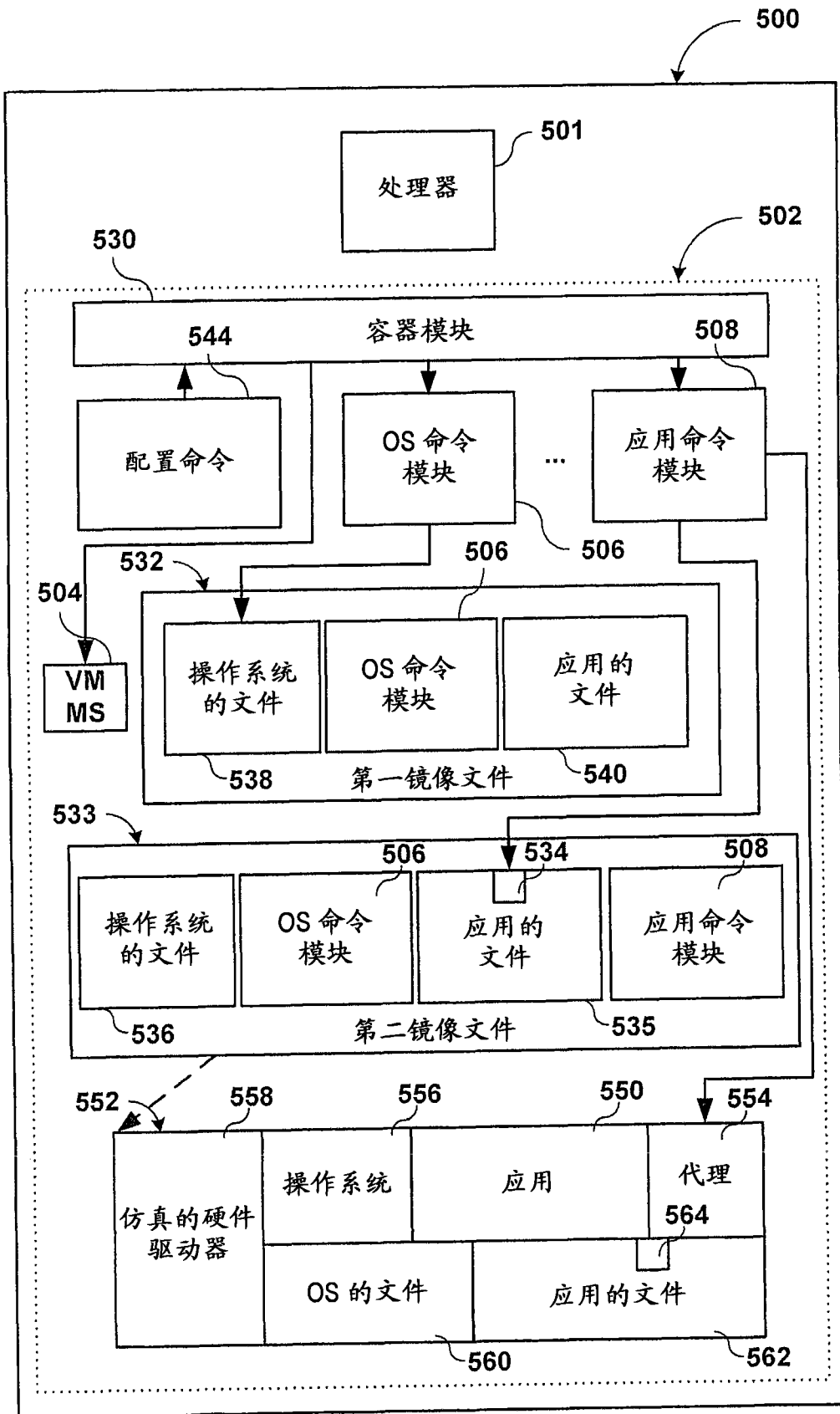


图 5

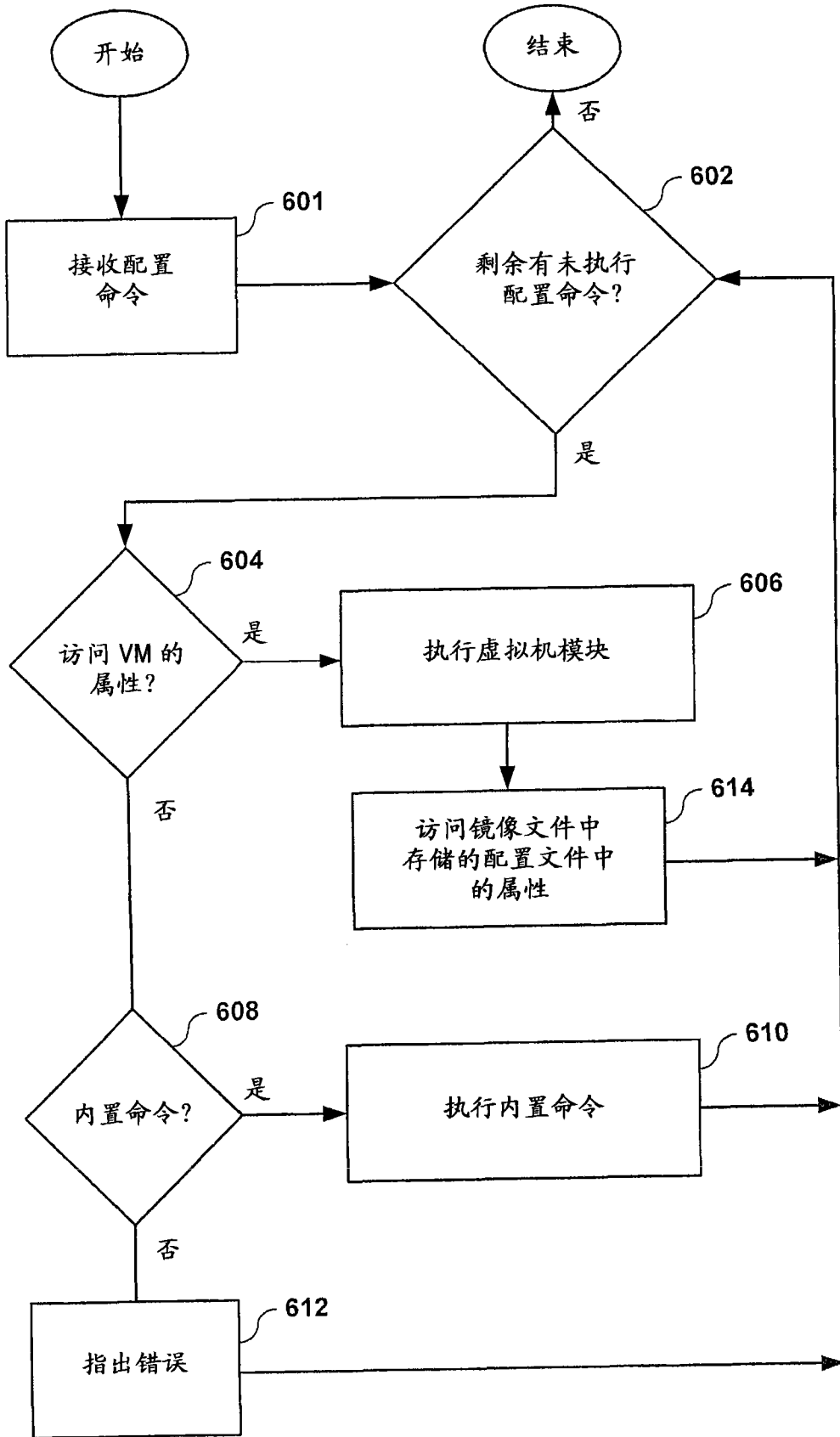


图 6