

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

G06F 9/445 (2006.01)

G06F 11/22 (2006.01)

G06F 12/08 (2006.01)



[12] 发明专利说明书

专利号 ZL 02140671.5

[45] 授权公告日 2007 年 5 月 9 日

[11] 授权公告号 CN 1315042C

[22] 申请日 2002.7.12 [21] 申请号 02140671.5

[30] 优先权

[32] 2001. 7.24 [33] US [31] 09/912,864

[73] 专利权人 微软公司

地址 美国华盛顿

[72] 发明人 瑞安·伯卡特

西萨雷曼·哈里卡里沙南

汤姆·亚彦 理查德·邦德

维杰什·谢蒂

维加亚谢德兰·杰亚西兰

[56] 参考文献

US6098097A 2000. 8. 1

WO99/49394A1 1999. 9. 30

US555416A 1996. 9. 10

WO99/59059A1 1999. 11. 18

US6066182A 2000. 5. 23

JP7 - 219747A 1995. 8. 18

审查员 解 欣

[74] 专利代理机构 中国国际贸易促进委员会专利
商标事务所

代理人 付建军

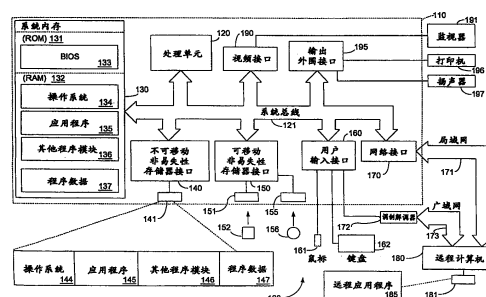
权利要求书 6 页 说明书 27 页 附图 5 页

[54] 发明名称

创建和使用具有选定功能的操作系统的方法和系统

[57] 摘要

一个创建和使用具有选定功能的操作系统 (OS) 映像的系统和方法。诸如个人计算机 (PC) 的原始设备制造商 (OEM) 或信息技术专业人士之类的用户可以从安装型 OS 映像中提供的许多 OS 组件中选择一组 OS 组件。选定的该组 OS 组件作为 OS 映像可以安装在计算机可读的介质, 如 CD-ROM 上。该 OS 映像包括一个脚本, 用于与 OS 映像进行交互以执行用户需要的功能, 比如启动一台目标计算机、安装一个参考 OS 映像, 或者从参考 OS 映像的失败安装中恢复目标计算机。



1.一种计算机化的方法，包括：

提供用户对基于文本的脚本的访问；

由用户编辑基于文本的脚本，以提供用户自定义的基于文本的脚本；

从计算机可读的介质上的一个简化的操作系统映像启动第一计算机，其中所述简化的操作系统映像是第二计算机的完整操作系统映像，所述第二计算机具有独立于第一计算机的硬件；以及

根据用户自定义的基于文本的脚本在第一计算机上执行一个或多个功能，其中所述计算机化的方法独立于体系结构。

2.根据权利要求 1 所述的计算机化的方法，其特征在于执行过程包括在第一计算机上安装一个完整的操作系统映像。

3.根据权利要求 2 所述的计算机化的方法，其特征在于所述第一计算机具有一个海量存储设备，并且其特征在于执行过程包括对海量存储设备进行分区。

4.根据权利要求 2 所述的计算机化的方法，其特征在于所述第一计算机具有一个海量存储设备，并且其特征在于执行过程包括对海量存储设备进行格式化。

5.根据权利要求 4 所述的计算机化的方法，其特征在于格式化包括用简化的操作系统映像支持的格式对带有文件系统的海量存储设备进行格式化。

6.根据权利要求 2 所述的计算机化的方法，其特征在于执行过程还包括对所述第一计算机的硬件进行验证。

7.根据权利要求 6 所述的计算机化的方法，其特征在于所述简化的操作系统映像包括可在简化的操作系统映像的保护模式下运行的保护模式硬件驱动程序，其特征在于验证过程包括：

在安装之前，使用保护模式硬件驱动程序对硬件进行验证。

8.根据权利要求 2 所述的计算机化的方法，其特征在于安装过程

包括将与完整的操作系统映像关联的一个或多个文件复制到计算机上并集成这些文件。

9.根据权利要求 8 所述的计算机化的方法,其特征在於脚本确定复制和集成文件的许多步骤,并且其特征在於脚本确定执行许多步骤的顺序。

10.根据权利要求 1 所述的计算机化的方法,其特征在於第一计算机的体系结构是 64 位。

11.根据权利要求 1 所述的计算机化的方法,其特征在於第一计算机的体系结构大于 32 位。

12.根据权利要求 1 所述的计算机化的方法,其特征在於启动包括:设置一个标记;以及

作为对设置的标记的响应,启用简化操作系统映像内的功能。

13.根据权利要求 12 所述的计算机化的方法,进一步包括允许用户模式软件响应设置的标记进行配置。

14.根据权利要求 12 所述的计算机化的方法,其特征在於功能包括将配置信息加载到易失性存储器中。

15.根据权利要求 1 所述的计算机化的方法,其特征在於计算机可读的介质上包含脚本。

16.根据权利要求 1 所述的计算机化的方法,其特征在於脚本远离计算机可读的介质。

17.根据权利要求 1 所述的计算机化的方法,其特征在於执行包括从软件的故障中恢复计算机。

18.根据权利要求 17 所述的计算机化的方法,其特征在於软件包括操作系统或应用程序。

19.根据权利要求 1 所述的计算机化的方法,其特征在於计算机可读的介质可由计算机通过网络进行访问。

20.根据权利要求 1 所述的计算机化的方法,进一步包括对用户自定义的基于文本的脚本进行编辑以使执行过程包括安装另一个操作系统。

21.一个系统，包括：

由用户编辑基于文本的脚本，以提供用户自定义的基于文本的脚本的装置；

从计算机可读的介质上的一个简化的操作系统映像启动第一计算机的装置，其中所述简化的操作系统映像是具有独立于第一计算机的硬件的第二计算机的完整操作系统映像，并且所述简化的操作系统映像是硬件独立的；以及

根据用户自定义的基于文本的脚本在第一计算机上执行一个或多个功能以在第一计算机上安装简化的操作系统映像的装置。

22.一种计算机化的方法，包括：

提供用户对基于脚本的文本的访问；

由用户编辑基于文本的脚本，以提供用户自定义的基于文本的脚本；

从计算机可读的介质上的简化的第一操作系统映像启动第一计算机，其中所述简化的操作系统映像是具有独立于第一计算机的硬件的第二计算机的完整操作系统；以及

根据用户自定义的基于文本的脚本在第一计算机上安装完整操作系统映像，其中启动和安装过程只需要重新启动计算机一次，并且所述计算机方法是体系结构独立的。

23.根据权利要求 22 所述的计算机化的方法，其特征在于脚本位于计算机可读的介质上。

24.根据权利要求 22 所述的计算机化的方法，其特征在于还包括对第一计算机的硬件进行验证。

25.根据权利要求 22 所述的计算机化的方法，其特征在于第一计算机具有一个海量存储设备，并且其特征在于安装过程包括对海量存储设备进行格式化。

26.根据权利要求 22 的计算机化的方法，其特征在于安装过程包括将与完整操作系统映像关联的一个或多个文件复制到第一计算机上并集成这些文件。

27.根据权利要求 26 的计算机化的方法，其特征在于脚本确定复制和集成文件的许多步骤，并且其特征在于脚本确定执行许多步骤的顺序。

28.根据权利要求 22 所述的计算机化的方法，其特征在于所述第一计算机有计算机可读的介质。

29.一个系统，包括：

具有用户选定的操作系统组件的第一操作系统映像，所述第一操作系统映像是硬件独立的；

具有第二操作系统映像的第一计算机，所述第一操作系统映像相对所述第二操作系统是简化的；以及

可由用户自定义的基于文本的脚本文件，该脚本文件与第一操作系统映像进行交互，以在具有独立于第一计算机的硬件的第二计算机上安装第二操作系统映像，其中所述基于文本的脚本文件包括用户可编辑的文本。

30.根据权利要求 29 所述的系统，其特征在于操作系统的每一组件都至少包括一个文件。

31.根据权利要求 29 所述的系统，其特征在于操作系统组件包括硬件驱动程序。

32.根据权利要求 29 所述的系统，其特征在于第二操作系统映像有许多操作系统组件，并且其特征在于第一操作系统映像具有多个操作系统组件的子集。

33.根据权利要求 29 所述的系统，其特征在于第二计算机的体系结构是 64 位。

34.根据权利要求 29 所述的系统，其特征在于脚本文件可以在大于 32 位体系结构的计算机运行。

35.一种计算机化的方法，所述方法包括：

基于要被启用的需要的功能，从第一计算机的多个操作系统组件中选择操作系统组件的一个子集；

生成与选定的操作系统组件的子集关联的文件列表，所述生成的文件列表是基于文本的，并可由用户编辑；以及

在目标介质上安装选定的操作系统组件的子集作为简化的操作系统映像，其中安装过程包括从安装介质中将文件复制到目标介质中；以及

使用目标介质引导具有独立于第一计算机的硬件的第二计算机，其中所述计算机化的方法是体系结构独立的。

36.根据权利要求 35 所述的计算机化的方法，其特征在于操作系统组件包括硬件驱动程序。

37.根据权利要求 35 所述的计算机化的方法，进一步包括至少创建一个配置单元，其中安装过程包括将创建的配置单元复制到目标介质中。

38.根据权利要求 35 所述的计算机化的方法，其中安装过程进一步包括在目标介质上集成这些文件。

39.根据权利要求 35 所述的计算机化的方法，进一步包括选择更多的操作系统组件，并且其中安装过程包括在目标介质上安装更多的操作系统组件。

40.一个系统，所述系统包括：

基于要被启用的需要的功能，从第一计算机的多个操作系统组件中选择操作系统组件的一个子集的装置；

生成与选定的操作系统组件的子集关联的文件列表的装置，所述生成的文件列表是基于文本的，并可由用户编辑；

在目标介质上安装选定的操作系统组件的子集作为独立于硬件的简化的操作系统映像的装置，其中安装过程包括从安装介质中将文件复制到目标介质中。

41.用来用许多操作系统组件在目标介质上创建简化的操作系统映像的系统，所述系统包括：

用户界面，允许用户从第一计算机的多个操作系统组件中选择操作系统组件的子集，生成与选定的操作系统组件的子集关联的文件列表，所述选择的操作系统组件被选择以启用需要的功能，所述生成的文件列表是基于文本的，并且可由用户编辑；以及

操作系统组件修改，它们允许选定的操作系统组件集作为第二计算机上简化的操作系统映像执行，其中简化的操作系统映像是硬件独

立的，并且第二计算机具有与第一计算机独立的硬件。

42.根据权利要求 41 所述的系统，进一步包括一个基于文本的脚本，用于指导简化的操作系统映像执行一个或多个功能。

43.根据权利要求 42 所述的系统，其特征在于这些功能包括在另一台计算机上安装一个操作系统。

44.根据权利要求 42 所述的系统，其特征在于这些功能包括从下列一组任务中选定的一个或多个任务：从软件故障中恢复第二计算机，在第二计算机上备份和/或恢复数据，远程排除另一计算机上的故障，以及使用计算机执行应用程序作为第二计算机的远程瘦客户端。

45.根据权利要求 41 所述的系统，其特征在于操作系统组件集是许多操作系统组件的一个子集。

46.根据权利要求 41 所述的系统，进一步包括许多操作系统组件中没有的更多操作系统组件，并且其中在执行应用程序时，进一步使一个或多个处理器执行如下操作：包括允许用户在选定的操作系统组件集中至少添加一个其他操作系统组件。

47.根据权利要求 41 所述的系统，其特征在于操作系统组件修改包括将状态信息写入易失性存储器。

48.根据权利要求 41 所述的系统，其特征在于在执行应用程序时，进一步使一个或多个处理器执行各种操作，包括生成简化的操作系统映像。

49.根据权利要求 41 所述的系统，其特征在于在执行应用程序时，进一步使一个或多个处理器执行各种操作，包括至少创建一个配置单元。

50. 根据权利要求 21 的系统，其中用于编辑的装置包括一个文本界面。

创建和使用具有选定功能的操作系统的方法和系统

技术领域

本发明涉及计算机操作系统领域。具体来说，本发明涉及一种具有脚本支持的操作系统，该操作系统具有用户选定的组件用以执行用户所需要的功能。

发明背景

操作系统 (OS) 映像与计算机中的硬件进行交互以执行各种各样的功能。典型的 OS 映像要求在硬盘驱动器之类的海量存储设备上有大量的空间。由于要求大量的磁盘空间，使得典型的 OS 映像不适用于那些只需要 OS 映像中某些功能的情况。此外，由于要求大量磁盘空间以及需要对海量存储设备进行读写访问，因而使典型的 OS 映像无法驻留在一个单独的只读的计算机可读的介质中。

为了在计算机上安装一个 OS 映像，历史上一直在使用另一个具有最小功能的并能从只读介质中启动计算机的 OS 映像（如 MS-DOS）。然而，MS-DOS 却有一些局限性，包括最高只能使用 640 千字节的易失性存储器的限制以及与 64 位计算机不兼容。此外，在使用 MS-DOS 启动和安装 OS 映像期间，往往还需要多次重新启动计算机。此外，用于使硬件运转的软件例程（即通常所说的由硬件制造商提供的硬件驱动程序）必须经过修改才能在 MS-DOS 的约束内进行工作以使用 MS-DOS 对硬件进行验证。

同样，某些版本的 Windows NT 操作系统 (NT) 要通过文本模式进行安装，以使计算机开始进入到 NT 的初始启动状态。图形用户界面模式安装则要根据用户的输入来对安装进行配置。还有一种叫做网上安装的方法，这种安装方法包括一个在计算机上自举 NT 本身的过程。然而，网上安装也需要重新启动计算机若干次。

大多数 OS 映像都包括一个内核，里面包含用于执行 OS 的必需的基本功能的软件例程。其他的功能则由内核外部的软件来实现。某些 OS 映像还包括一个只有最少功能和大小的微型内核。在这样的微型内核 OS 映像中，内核外部的软件执行 OS 映像的大量的必需功能。然而，这些微型内核 OS 映像一般来讲要依赖于硬件，并且不包括可由用户自定义的以执行用户所需要的特定功能的基于文本脚本。

下面描述的本发明解决了这些缺点以及其他一些缺点。

发明内容

本发明能从一个安装型 OS 映像创建简化的操作系统 (OS) 映像。本发明还包括产生简化的 OS 映像的方法。诸如个人计算机 (PC) 的原始设备制造商 (OEM) 之类的用户可以从一个安装型 OS 映像中提供的许多 OS 组件中选择一个 OS 组件集。选定的该 OS 组件集作为简化的 OS 映像可以安装在计算机可读的介质 (CRM) (如 CD-ROM) 上。简化的 OS 映像包括一个脚本，用于与简化的 OS 映像进行交互以执行用户需要的功能，比如显示命令提示符、安装一个参考 OS 映像，或者从参考 OS 映像的故障中恢复。简化的 OS 映像对易失性和非易失性存储器存储空间要求较小。本发明可以用于在体系结构大于 32 位的计算机上安装参考 OS 映像。本发明还可以在体系结构为 32 位或小于 32 的计算机上运行。简化的 OS 映像使用硬件驱动程序 (用于控制硬件，一般来讲为保护模式编写) 来验证硬件的操作是否正确。本发明的简化的 OS 映像是独立于硬件的，因为简化的 OS 映像包括许多用户指定的硬件驱动程序。本发明提供了一个框架，可以让用户只需要重新启动计算机一次即可在计算机上安装参考 OS 映像。脚本可以执行许多功能，包括但不限于启用网络连接、对海量存储设备进行分区，以及用一种文件系统将海量存储设备格式化。例如，脚本可以根据 WINDOWS 操作系统中提供的一种文件系统格式将海量存储设备格式化。

根据本发明的一个方面，一种计算机化的方法，包括：提供用户对基于文本的脚本的访问；由用户编辑基于文本的脚本，以提供用户自定义的基于文本的脚本；从计算机可读的介质上的一个简化的操作系统映像启动第一计算机，其中所述简化的操作系统映像是第二计算机的完整操作系统映像，所述第二计算机具有独立于第一计算机的硬件；以及根据用户自定义的基于文本的脚本在第一计算机上执行一个或多个功能，其中所述计算机化的方法独立于体系结构。计算机可以是 32 位或 64 位体系结构。

根据本发明的另一个方面，一种系统包括：由用户编辑基于文本的脚本，以提供用户自定义的基于文本的脚本的装置；从计算机可读的介质上的一个简化的操作系统映像启动第一计算机的装置，其中所述简化的操作系统映像是具有独立于第一计算机的硬件的第二计算机的完整操作系统映像，并且所述简化的操作系统映像是硬件独立的；以及根据用户自定义的基于文本的脚本在第一计算机上执行一个或多个功能以在第一计算机上安装简化的操作系统映像的装置。

根据本发明的另一个方面，一种计算机化的方法，包括提供用户对基于脚本的文本的访问；由用户编辑基于文本的脚本，以提供用户自定义的基于文本的脚本；从计算机可读的介质上的简化的第一操作系统映像启动第一计算机，其中所述简化的操作系统映像是具有独立于第一计算机的硬件的第二计算机的完整操作系统；以及根据用户自定义的基于文本的脚本在第一计算机上安装完整操作系统映像，其中启动和安装过程只需要重新启动计算机一次，并且所述计算机方法是体系结构独立的。

根据本发明的另一个方面，一种系统包括：具有用户选定的操作系统组件的第一操作系统映像，所述第一操作系统映像是硬件独立的；具有第二操作系统映像的第一计算机，所述第一操作系统映像相对所述第二操作系统是简化的；以及可由用户自定义的基于文本的脚本文件，该脚本文件与第一操作系统映像进行交互，以在具有独立于第一计算机的硬件的第二计算机上安装第二操作系统映像，其中所述基于

文本的脚本文件包括用户可编辑的文本。计算机可以是 32 位或 64 位体系结构。

根据本发明的另一个方面，与计算机一起使用的计算机可读的恢复介质上有一个操作系统映像和基于文本的脚本文件。脚本文件与该操作系统映像进行交互以从计算机上的软件故障中恢复。

根据本发明的另一个方面，一种计算机化的方法，所述方法包括：基于要被启用的需要的功能，从第一计算机的多个操作系统组件中选择操作系统组件的一个子集；生成与选定的操作系统组件的子集关联的文件列表，所述生成的文件列表是基于文本的，并可由用户编辑；以及在目标介质上安装选定的操作系统组件的子集作为简化的操作系统映像，其中安装过程包括从安装介质中将文件复制到目标介质中；以及使用目标介质引导具有独立于第一计算机的硬件的第二计算机，其中所述计算机化的方法是体系结构独立的。

根据本发明的另一个方面，一个系统，所述系统包括：基于要被启用的需要的功能，从第一计算机的多个操作系统组件中选择操作系统组件的一个子集的装置；生成与选定的操作系统组件的子集关联的文件列表的装置，所述生成的文件列表是基于文本的，并可由用户编辑；在目标介质上安装选定的操作系统组件的子集作为独立于硬件的简化的操作系统映像的装置，其中安装过程包括从安装介质中将文件复制到目标介质中。

根据本发明的另一个方面，一种系统能在目标介质上从操作系统的许多组件中创建一个操作系统映像。该系统包括至少一个应用程序，当由计算机上的一个或多个处理器执行该应用程序时，使一个或多个处理器执行如下操作，包括允许用户从第一计算机的多个操作系统组件中选择操作系统组件的子集，生成与选定的操作系统组件的子集关联的文件列表，所述选择的操作系统组件被选择以启用需要的功能，所述生成的文件列表是基于文本的，并且可由用户编辑；以及操作系统组件修改，它们允许选定的操作系统组件集作为第二计算机上简化的操作系统映像执行，其中简化的操作系统映像是硬件独立的，并且

第二计算机具有与第一计算机独立的硬件。

附图说明

图 1 是本发明的方法和系统的一个实施例的方框图，说明了应用了本发明的计算系统环境的一个示例。

图 2 是一个发明的方法和系统的一个实施例的方框图，说明了一个操作系统组件。

图 3 是本发明的方法和系统的一个实施例的流程图，说明了在目标介质上安装简化的操作系统映像的过程

图 4 是本发明的方法和系统的一个实施例的方框图，说明了在目标介质上安装简化的操作系统映像的过程

图 5 是本发明的方法和系统的一个实施例的流程图，说明了使用简化的操作系统映像安装另一个操作系统映像的过程

图 6 是本发明的方法和系统的一个实施例的方框图，说明了从一台远程启动计算机启动目标计算机以在目标计算机上安装引用操作系统映像的过程。

图 7 是本发明的方法和系统的一个实施例的方框图，说明了从一台本地计算机可读的介质启动目标计算机以在目标计算机上安装引用操作系统映像的过程。

图 8 是本发明的方法和系统的一个实施例的方框图，说明了其软件发生故障的计算机和一个恢复介质之间的交互。

对应的参考字符表示所有附图的对应部件。

具体实施方式

一般来说，本发明涉及一种方法和系统，至少提供一个应用程序和对操作系统 (OS) 组件的必需的修改，以从一个安装型 OS 映像创建一个可启动的简化的 OS 映像，该简化的 OS 映像有脚本支持，以执行诸如原始设备制造商 (OEM) 之类的用户指定的一个或多个功能。简化的 OS 映像相对于完全安装的安装型 OS 映像来说功能和/或大小缩小了。简化的 OS 映像具有用户选定的功能。

用户选择 OS 组件以包含在简化的 OS 映像中，还可以添加其他 OS 组件以实现需要的功能。应用程序针对安装型 OS 映像应用选择和补充以创建可启动的简化的 OS 映像。简化的 OS 映像将写入到计算机可读的介质 (CRM) (如 CD-ROM)，存储在远程服务器上，或安装到本地海量存储设备上。简化的 OS 映像包括一个用于执行用户指定功能的脚本。例如，用户可以对脚本进行编辑以与简化的 OS 映像交互，从而安装软件、从故障 OS 安装中恢复、更新硬件驱动程序、重新密封和审核计算机、执行应用程序 (包括其他脚本)，或执行任何其他任务。

尽管本发明的示范 OS 是 WINDOWS 操作系统，但是本发明也可适用于其他操作系统以创建和使用简化的 OS 映像。在示范 WINDOWS 操作系统实施例中，简化的 OS 映像包括与 WINDOWS NT 操作系统关联的基码，并带有最低限度的 WIN32 应用程序编程接口 (API) 子系统。简化的 OS 映像有启动到 Win32 子系统中必需的最小数量的 OS 组件。最小数量的 OS 组件包括内核、硬件驱动程序，以及系统文件。简化的 OS 映像在一个数据库中存储了关于其本身的最低限度的状态信息，如与 WINDOWS 操作系统中的注册表关联的配置单元。

如需了解有关 WINDOWS 操作系统中提供的启动过程和 OS 组件的一般信息，请参阅 Solomon 和 Russinovich 所著的 “Inside

Microsoft Windows 2000”第三版，2000，第 46-87、177-206、以及 215-236 页，在此处作为参考。

在一个实施例中，简化的 OS 映像包括对微处理器的受保护操作方式的支持。微处理器的保护模式是相对于微处理器操作的实模式而言的。用户可以对脚本进行编辑，以使用硬件制造商提供的并用于控制硬件的驱动程序对硬件进行验证。对于所提供的为保护模式而编写的硬件驱动程序，用户没有必要编写单独的硬件驱动程序或以别的方式修改保护模式硬件驱动程序即可在实模式下运行。在简化的 OS 映像中，用户使用未修改的硬件驱动程序以在简化的 OS 映像的保护模式下对硬件进行验证。当简化的 OS 映像用于安装另一个 OS 时，硬件验证的结果也适用于已安装的 OS。

通过对脚本进行编辑，用户配置参考 OS 映像的自动安装。简化的 OS 映像可用于安装任何 OS 映像。此外，简化的 OS 映像还可用于修复 OS 安装或执行端对端的图形用户界面 (GUI) 应用程序。

在一个实施例中，OEM 使用简化的 OS 映像和个人计算机 (PC) 上安装一个参考 OS 映像，该个人计算机的体系结构可以是 64 位或 32 位。OEM 接收一个软件开发工具包 (SDK)，该包中包含用于创建简化的 OS 映像的应用程序和必需的 OS 组件修改。OEM 在 OEM 工厂根据特定的 PC 硬件和该 OEM 必须现场考虑的其他事项选择将参考 OS 映像安装到目标计算机上所必需的 OS 组件。具体来说，OEM 通过一个文本文件添加或删除 OS 组件（如硬件驱动程序）并指定海量存储设备配置。应用程序将用选定的 OS 组件集创建简化的 OS 映像。然后 OEM 使用简化的 OS 映像以启动一个硬盘驱动器尚未格式化的目标计算机（如 PC）、对硬件进行验证、将目标计算机的硬盘格式化，并在目标计算机上安装参考 OS 映像。

目标计算机是根据各种体系结构设计的，包括但不限于 32 位和 64 位。每一台计算机一般来讲都是按其体系结构进行分类的。64 位体系结构一般是指计算机在内部以 64 位分组对数据进行操作。64 位计算机体系结构基于 64 位的元素，包括处理单元（参见图 1，引

用字符 120)，处理单元内部或外部至少有一个存储器寄存器，以及一个数据总线。同样，当一般谈及一类体系结构大于 32 位的计算机时，该计算机可能是 64 位体系结构、128 位体系结构，或者在内部至少以 32 位对数据进行操作的任何体系结构。那些精通本技术的人可能会注意到，本发明的系统和方法不仅限于当前的计算机体系结构。本发明还适用于未来的计算机体系结构，包括但不限于 128 位和 256 位体系结构。如果计算机有基于软件的控制计算机及其外围设备的 OS 概念的话，本发明还适用于完全不同类型的计算机，甚至还可适用于非数字计算机。此外，本发明还适用于 32 位或较小位数的计算机体系结构，包括但不限于 32 位体系结构、16 位体系结构、8 位体系结构或者单个位数体系结构。一般来讲，本发明的系统和方法是独立于体系结构的，因为本发明可以适用于受简化的 OS 映像支持的任何计算机体系结构。例如，如果简化的 OS 映像支持 32 位体系结构计算机和 64 位体系结构计算机，那么本发明就适用于 32 位或 64 位体系结构的计算机。在一个实施例中，启动和执行的步骤可以在 32 位或较小位数的体系结构计算机上进行。此外，脚本文件可以在 32 位或较小位数的体系结构的计算机上运行。

诸如 MICROSOFT 磁盘操作系统 (MS-DOS) 之类的操作系统历史上一直支持在 16 位体系结构的计算机使用脚本功能。MS-DOS 利用中断与基本输入输出系统 (BIOS) 通信以实现计算机上的功能。BIOS 在运行了 MS-DOS 和 BIOS 的特定计算机的上下文中执行 MS-DOS 命令。可以对 BIOS 进行修改以在 32 位计算机上执行 16 位 MS-DOS 命令。然而，却无法通过修改 BIOS 来在体系结构大于 32 位的计算机上执行 16 位 MS-DOS 命令，如果不完全重新编写 BIOS 就无法对这样的体系结构提供支持。此外，也无法通过修改 BIOS 来生成可在体系结构大于 32 位的计算机上操作的命令，如果不完全重新编写 MS-DOS 也无法对这样的体系结构提供支持。由于 MS-DOS 和 BIOS 的固有的设计，MS-DOS 从根本上只限于 32 位或较小的体系结构。完全重写 MS-DOS 和/或 BIOS 就等于编写了一

个新的 OS，如此处描述的 OS。

例如，在其他局限性中，MS-DOS 一般来讲也无法访问大量的内存，因为 MS-DOS 主要是为 8 位和早期 16 位体系结构处理器设计的，这些处理器一般来讲都有地址局限性，只能访问 1 兆字节的内存，因为地址总线只有二十位宽。如果不完全重写 MS-DOS（将包括特殊的 MS-DOS 扩展名），在 MS-DOS 下运行的应用程序将不能够访问 1 兆字节以上的内存，因而限制了地址总线比 20 位宽的处理器。如果不完全重新编写，将无法通过修改 MS-DOS 来支持这样的内存访问。一般来讲，大多数在 MS-DOS 下运行的应用程序最大只有 640 千字节的内存。

首先请参看图 1，一个方框图说明了可以应用本发明的计算系统环境 100 的示例。该计算系统环境 100 只是一个合适的计算或操作环境的示例，不对本发明的应用范围或功能作任何限制。计算系统环境 100 也不应被解释为与示范计算系统环境 100 中所示的任何一个组件或组件组合有任何依赖关系或要求。

本发明可以应用于很多其他一般用途或特殊用途的计算系统环境或配置。可以应用本发明的已知的计算系统、环境、和/或配置的示例包括但不限于个人计算机、服务器、手提或膝上型设备、多处理器系统、基于微处理器的系统、机顶盒、可进行设定的消费类电子产品、网络 PC、小型机、大型机、包括上述任何系统或设备的分布式计算环境。

本发明可以在由一台或多台计算机或其他设备执行的计算机可执行的指令的一般上下文中进行描述，如程序模块。一般来讲，程序模块包括但不限于例程、程序、对象、组件和执行特定任务或实现特定抽象数据类型的数据结构。本发明还可以在分布式计算环境中应用，在这种环境中，各种任务可以由通过通信网络链接在一起的远程处理设备来执行。在分布式计算环境中，程序模块可以位于本地和远程计算机存储介质（包括记忆存储设备）。

请看图 1，一个用于实现本发明的示范系统包括一个以计算机 110

为形式的一般用途计算设备。计算机 110 的组件可以包括但不限于处理单元 120、系统内存 130，以及系统总线 121，它把各种系统组件（包括系统内存）连接到处理单元 120。在本发明的一个实施例中，处理单元 120 是 32 位或 64 位体系结构。系统总线 121 可以是任何类型的总线结构，包括内存总线或内存控制器、外围总线，以及使用任何种类的总线体系结构的本地总线。作为示例，而不作为限制，这样的总线体系结构包括工业标准体系结构 (ISA) 总线、微通道体系结构 (MCA) 总线、增强的 ISA (EISA) 总线、视频电子产品标准协会 (VESA) 本地总线，以及外围组件互连 (PCI) 总线（也就是通常所说的附加板总线）。

计算机 110 通常至少包括某些形式的 CRM。CRM 可以是计算机 110 可以访问的任何介质，并包括易失性和非易失性介质、可移动的和不可移动的介质。作为示例，而不作为限制，CRM 可以包括计算机存储介质和通信介质。计算机存储介质包括以任何方法或技术实现的易失性和非易失性、可移动的和不可移动的介质，以用于存储信息，如计算机可读的指令、数据结构、程序模块或其他数据。计算机存储介质包括，但不限于，RAM、ROM、EEPROM、闪存或其他内存技术，CD-ROM、数字多功能磁盘 (DVD) 或其他光盘存储器、盒式磁带、磁带、磁盘存储器或其他磁存储设备，或者可用于存储所需要的信息并且计算机 110 可以访问的任何其他介质。通信介质通常在一个调制数据信号（如载波）或其他传输机制中包含计算机可读的指令、数据结构、程序模块或者其他数据，并且包括任何信息提供介质。那些精通本技术的人会熟悉调制数据信号，该信号设置或更改了一个或多个其特性，以便对该信号中的信息进行编码。作为示例，而不作为限制，通信介质包括有线介质，如有线网络或者直接有线连接，以及无线介质，如声控、RF、红外和其他无线介质。上述任何几项的组合也应包括在 CRM 的范围内。

系统内存 130 包括易失性和/或非易失性存储器形式的计算机存储介质，如只读存储器 (ROM) 131 和随机访问存储器 (RAM) 132。

ROM 131 中通常存储了一个基本输入/输出系统 133 (BIOS), 里面包含帮助在计算机 110 内的各个元素之间 (如在启动过程中) 传输信息的基本例程。RAM 132 通常包含处理单元 120 立即可访问和/或目前正在对其进行操作的数据和/或程序模块。作为示例, 而不作为限制, 图 1 说明了操作系统 134、应用程序 135、其他程序模块 136, 以及程序数据 137。

计算机 110 还可以包括其他可移动的/不可移动的、易失性/非易失性计算机存储介质。仅作为示例, 图 1 说明了一个可以从不可移动的非易失性磁介质中读取或写入的硬盘驱动器, 一个从可移动的非易失性磁盘 152 读取或写入的硬磁驱动器, 以及可以从可移动的非易失性光盘 156 (如 CD ROM 或其他光学介质) 读取或写入的光盘驱动器 155。其他可用于示范操作环境的可移动/不可移动的易失性/非易失性计算机存储介质包括但不限于磁带、快速内存卡、数字多功能磁盘、数字视频磁带、固态 RAM、固态 ROM 等等。硬盘驱动器 141 通常通过一个不可移动的内存接口 (如接口 140) 连接到系统总线 121, 硬磁盘驱动器 151 和光盘驱动器 155 通常由可移动的内存接口 (如接口 150) 连接到系统总线 121。

驱动器或其他海量存储设备以及上面讨论并在图 1 中显示的与它们关联的计算机存储介质, 将为计算机 110 存储计算机可读的指令、数据结构、程序模块以及其他数据。例如, 在图 1 中, 硬盘驱动器 141 存储了操作系统 144、应用程序 145、其他程序模块 146, 以及程序数据 147。请注意, 这些组件可能与操作系统 134、应用程序 135、其他程序模块 136, 以及程序数据 137 相同, 也可能不同。操作系统 144、应用程序 145、其他程序模块 146, 以及程序数据 147 在这里使用了不同的编号, 以至少说明它们是不同的副本。根据本发明的 OS, 如操作系统 144, 是计算机 110 的基本软件控制程序, 可执行各种功能, 包括提供用户界面, 管理应用程序 145 的执行, 以及控制各种输入/输出 (I/O) 设备的数据的输入和输出。应用程序 145 代表一个或多个可以在计算机 110 上执行的各种各样的软件例程。应用

程序的示例包括,但不仅限于,教育程序、引用程序、办公程序(例如,文字处理软件、电子表格、数据库)、娱乐程序,以及实用程序(例如,通信程序)。应用程序 145 可以由用户安装在计算机 110 上,或者由计算机 110 的 OEM 和/或经销商预装。

用户可以通过诸如键盘 162 和指示设备 161(通常是指鼠标、轨迹球或触模板)之类的输入设备向计算机 20 中输入命令和信息。其他输入设备(未显示)可以包括麦克风、游戏杆、游戏板、卫星电视天线、扫描仪或类似的装置。这些和其他输入设备通常通过一个用户输入接口 160(它连接到系统总线)连接到处理单元 120,但也可以通过其他接口和总线结构进行连接,如并行端口、游戏端口或通用串行总线(USB)。监视器 191 或其他类型的显示设备也通过一个接口(如视频接口 190)连接到系统总线 121。除了监视器外,计算机还可包括其他外围输出设备,如扬声器 197 和打印机 196,它们可以通过一个输出外围接口 195 进行连接。

计算机 110 可以在一个网络环境中运行,使用逻辑连接与一台或多台远程计算机(如远程计算机 180)相连接。远程计算机 180 可以是一台个人计算机、服务器、路由器、网络 PC、一台对等设备或其他通用网络节点,通常包括上文关于计算机 110 的描述中的许多或者全部元素,虽然在图 1 中只显示了记忆存储设备 181。图 1 中描述的逻辑连接包括局域网(LAN) 171 和广域网(WAN) 173,但也可以包括其他网络。这样的网络环境在办公室、企业计算机网络、Intranet 以及全球计算机网络(例如,Internet)中是常见的。

当计算机 110 在一个 LAN 网络环境中使用时,计算机 110 通过网络接口或适配器 170 连接到 LAN 171。当计算机 110 在一个 WAN 网络环境中使用时,计算机 110 通常包括一个调制解调器 172 或其他用于在 WAN 173(如 Internet)建立通信的装置。调制解调器 172 可以是内置的,也可以是外置的,可以通过用户输入接口 160 或其他相应的机制连接到系统总线 121。在网络环境中,涉及计算机 110 的程序模块,或者它的一部分,可以存储在远程记忆存储设备中。

作为示例，而不作为限制，图 1 说明了驻留在记忆设备 181 上的远程应用程序 185。显然，这里显示的网络连接是示范性的，也可以使用其他装置在计算机之间建立通信链接。

下面请参看图 2，一个方框图说明了 OS 组件 202。在某些计算机系统中，OS 映像将被组织进 OS 组件 202，其中每一个 OS 组件 202 都执行一个功能或功能组合。每一个 OS 组件 202 都最多可包括计算机可访问的 N 个文件 204。

下面请看图 3，一个流程图说明了在目标介质上安装简化的 OS 映像的过程。目标介质是任何 CRM，可由计算机在本地或远程访问。用户在 302 至少从许多 OS 组件中选择一个 OS 组件（参见图 2，引用字符 202）。在一个实施例中，用户选择了一组 OS 组件。许多 OS 组件包括用于启用计算机中的硬件的驱动程序，这些硬件包括海量存储设备、网卡，和/或图 1 中所显示的或未显示的任何设备。如上文所描述，许多 OS 组件中的每一个组件都至少包括一个文件（参见图 2，引用字符 204）。安装简化的 OS 映像的过程包括将与选定的 OS 组件集关联的文件和相关的 OS 组件修改复制到目标介质上并进行集成。与选定的 OS 组件关联的文件在 304 复制到目标介质上，并在 306 集成为简化的 OS 映像。

在一个实施例中，简化的 OS 映像中的选定的 OS 组件集包括安装型 OS 映像中的操作系统许多组件中的一部分。用户还可以将其其他的 OS 组件添加到选定的 OS 组件集中。其他 OS 组件可包括用户提供的 OS 许多组件中所没有的任何组件。其他 OS 组件包括但不限于特定的硬件驱动程序、脚本文件，以及应用程序。例如，特定的硬件驱动程序包括海量存储设备驱动程序、视频设备驱动程序、输入设备驱动程序，以及网络设备驱动程序。要包括在简化的 OS 映像中的硬件驱动程序可由用户进行自定义，以包括安装型 OS 映像中提供的或者第三方提供的任何和全部硬件驱动程序。同样，用户还可以添加工具或实用工具，如网络客户端、测试工具以及报告实用工具。与其他 OS 组件关联的文件由用户提供并复制目标介质上，用于集成

到简化的 OS 映像中。例如，要添加对特定的海量存储设备的支持，用户可以在硬件驱动程序列表中添加特定的海量存储设备的条目，并在一个包含所有驱动程序的文件目录中添加与特定的海量存储设备关联的硬件驱动程序。用户可以在将简化的 OS 映像安装在目标介质上之前或之后添加或删除对特定硬件设备的支持。

在一个实施例中，简化的 OS 映像大约需要 100 兆字节的非易失性记忆存储空间，而完全安装的 OS 映像典型情况下却需要 950 兆字节到 1.2 千兆字节。那些精通本技术的会注意到，随着操作系统的发展，在不影响本发明的功能的情况下，简化的 OS 映像的非易失性记忆存储空间要求可能会大大地少于 100 兆字节。

简化的 OS 映像还包括一个最小的文件列表和最低限度的状态信息。例如，在 WINDOWS 操作系统中，状态信息存储在一个被称为注册表的数据库中，该数据库包括一系列叫做配置单元的不连续的文件。有一个主文件布局列表，该列表包含与安装型 OS 映像所在的介质中所包含的 OS 组件关联的每个文件名。主文件布局列表中的条目表示它们对于简化的 OS 映像是否为必需的。例如，在 WINDOWS 操作系统中，主文件布局列表是 layout.inf。本发明包括映像生成软件，用来分析主文件布局列表和识别简化的 OS 映像所必需的文件。该映像生成软件将把简化的 OS 映像安装在目标介质上，方法是将识别的文件复制到目标介质中的一个文件目录中。然后该文件目录可以被镜像到一个非易失性介质中。此外，映像生成软件还在必要时将识别的文件解压缩到目标介质上。映像生成软件还可让用户在创建目录结构时在简化的 OS 映像中添加其他必需的文件。用户也可以采用这种方式在安装型 OS 映像中添加里面所没有的较新的驱动程序文件和其他实用程序文件。

一个 GUI 和一个文本界面可用来从 OS 的许多组件中选择一部分 OS 组件。用户可以通过一个应用程序（如 SDK）与任一界面进行交互，以选择放在简化的 OS 映像中的一部分 OS 组件。上文图 3 中描述的 SDK 包括生成与选定的 OS 组件子集关联的文件的

列表的装置和用于把选定的 OS 组件子集在目标介质上安装为简化的 OS 映像的装置。安装的方法是将文件从安装介质复制到目标介质。此外，上文描述的示例和其它地方描述的示例构成了选择 OS 组件子集的装置、生成文件列表的装置，以及安装选定的 OS 组件子集的装置。

下面请看图 4，一个方框图说明了在目标介质 410 上安装简化的 OS 映像 412 的过程。软件开发工具包 (SDK) 405 或至少一个其他应用程序向用户显示一个列出了 OS 组件 402 的列表，如一个包含硬件驱动程序的列表。用户从列表 402 中选择一组 OS 组件。计算机上的一个或多个处理器执行 SDK 405 以使用户从 OS 组件列表 402 中选择 OS 组件子集 404。可被 SDK 405 访问的安装型 OS 映像 406 提供对应于列表 402 的许多 OS 组件 407。例如，安装型 OS 映像 406 位于包括 WINDOWS 操作系统产品的一个介质上。SDK 405 针对安装型 OS 映像 406 应用选定的 OS 组件集以识别与选定的 OS 组件集 404 关联的文件。SDK 405 也访问和识别许多 OS 组件修改 408 中的任何一个修改以及与选定的 OS 组件集 404 关联的对应文件。被识别的 OS 组件修改 416 允许选定的 OS 组件 414 在目标介质 410 上作为简化的 OS 映像 412 执行。OS 组件修改 408 将替换、补充或以别的方式修改许多 OS 组件 407。

具体来说，对许多 OS 组件 407 的修改 408 可以包括修改 OS 组件 407 以不向非易失性存储器写入数据。例如，在 WINDOWS 操作系统环境中，本发明的 OS 组件将把注册表信息写到易失性存储器而不是写到非易失性存储器中。注册表信息包括存储在一个数据库中的被称为“配置单元”的不连续的文件中的 OS 配置数据。在启动期间，一个安装型加载程序将向与简化的 OS 映像关联的内核传递一个标记，以向内核表明正在启动的是简化的 OS 映像 412。在内核初始化过程中，注册表管理器将看到该标记并将注册表写入到易失性存储器，即使在从一个只读 CRM 启动也是如此。对注册表中的条目进行的更改是在易失性存储器中进行的。由于易失性存储器的性质，

如果简化的 OS 映像 412 重新启动，所有的更改都将被扔掉。将注册表信息存储在易失性存储器中的好处是，每一次启动时简化的 OS 映像 412 都有一个清洁的注册表。在另一个实施例中，如果目标计算机能够对非易失性存储器进行写入访问，那么就会为用户提供一个选项，是否将注册表信息或其他配置信息保存到非易失性存储器中以使用户能够在不同的重新启动之间对该信息进行维护。

也可以对配置单元中的条目进行修改。配置单元条目存储在一个文本文件中。配置单元创建软件在创建简化的 OS 映像 412 时还将创建该文本文件的二进制版本。配置单元创建软件将分析该文本文件中的配置单元条目，并为目标介质创建一个二进制配置单元。配置单元创建软件还对现有的二进制配置单元进行修改，其做法是，将配置单元加载到存储器中，应用所请求的更改，然后以二进制格式保存经修改的配置单元。

例如，WINDOWS 操作系统启动时需要四个配置单元。必需的配置单元包括 SYSTEM、SOFTWARE、SECURITY，以及安全帐户管理器 (SAM) 配置单元。默认的 SYSTEM 配置单元包括有关在自举 WINDOWS 操作系统时要运行哪些子系统、硬件驱动程序和服务的信息。本发明的 SYSTEM 是当前 WINDOWS 操作系统安装程序的 SYSTEM 配置单元的修改版本。配置单元创建软件将把一个文本文件（如 minint.inf）中的修改应用到默认 SYSTEM 配置单元。SOFTWARE 配置单元是由配置单元创建软件从一个文本文件（如 hivesft.inf）创建的。SECURITY 和 SAM 配置单元是长度为零的文件，表示在简化的 OS 映像 412 上没有相应的安全信息。

根据传递到内核的加载程序标记（该标记表示正在启动简化的 OS 映像 412），注册表管理器将在一个特定位置下面创建一个注册表项。例如，注册表管理器将在创建易失项 HKLM\System\CurrentControlSet\Control\MiniNT 之后创建一个叫做“Option”的 DWORD 值。“Option”将被设置为 1 以表示用户是在简化的 OS 映像 412 环境中。在用户模式下执行的应用程

序将按需要查询注册表项以根据该值进行配置。

在本发明的示范 WINDOWS 操作系统实施例中，简化的 OS 映像拥有 “system” 帐户的安全特权。简化的 OS 映像可使用的内存仅限于可用的易失性存储器的量，因为没有分页文件提供对其他存储器的访问。此外，默认情况下自动启动的一些其他不需要的服务（例如，系统文件保护）在本发明的示范 WINDOWS 操作系统实施例中对简化的 OS 映像是禁用的。

SDK 405 将把选定的 OS 组件 414 和识别的 OS 组件修改 416 作为简化的 OS 映像 412 安装到目标介质 410 中，做法是复制与选定的 OS 组件集 404 关联的文件和识别的 OS 组件修改 416，然后进行集成。用户还可以添加 OS 组件集 407 中不存在的其他 OS 组件。其他 OS 组件将通过 SDK 405 来添加到选定的 OS 组件 414 中。SDK 405 将在目标介质 410 中添加一个脚本文件以执行用户指定的功能。如果目标介质 410 是一个只读 CRM，那么用户应在简化的 OS 映像 412 安装在 CRM 上之前对该脚本文件进行自定义。如果目标介质 410 是一个可读写 CRM，那么用户可以在简化的 OS 映像 412 安装在 CRM 上之前或之后对该脚本进行自定义。

那些精通本技术的人将会注意到，图 4 中的安装型 OS 映像 406、OS 组件修改集 408，以及目标介质 410 可以由 SDK 405 本地或远程进行访问，或者可以包括在 SDK 405 中。远程访问的示例如图 1 的网络环境所示。

下面请看图 5，一个流程图说明了使用简化的 OS 映像安装另一个 OS 映像的过程。一台计算机在 502 从 CRM 上的简化的 OS 映像启动。在一个实施例中，简化的 OS 映像支持文本用户界面和图形用户界面 (GUI)。具体来说，GUI 支持至少有 256 色和分辨率至少为 640 x 480 像素的视频图形阵列 (VGA) 视频模式。用户自定义的、基于文本的脚本与简化的 OS 映像进行交互，以指导在计算机上执行一个或多个功能。脚本将确定执行功能的步骤。脚本也将确定执行步骤的顺序。脚本还将支持多任务处理以及多个步骤的连续执行。即，

用户一步一步地指定脚本中的每一个步骤是否只能在前面的步骤完成之后才能执行。脚本还可采用这种方式允许一次执行一个步骤以及同时执行多个步骤。在一个实施例中，脚本是一个计算机可读的文件并位于 CRM 中。计算机可执行的指令驻留在一个 CRM 中以根据该脚本协调功能的启动和执行。在其他实施例中，脚本位于计算机本地或远离计算机。CRM 可以位于计算机本地或者远离计算机（请分别参见图 6 和 7）。例如，计算机以远程方式访问 CRM 或脚本，如图 1 的网络环境所示。

在一个实施例中，诸如 OEM 之类的用户对脚本进行编辑，以使脚本执行必需的功能以在计算机上安装参考 OS 映像。在一个实施例中，计算机是一台空白 PC，上面的海量存储设备尚未格式化，也没有分区。在另一个实施例中，计算机是一台具有 CRM 的 PC，并且有一个需要更新的 OS。用户还对脚本进行编辑以在 504 验证计算机上的硬件。计算机上的硬件包括网卡、海量存储设备、视频卡，和/或图 1 中所显示的或未显示的任何其他设备。脚本通过启用指定的硬件设备来验证硬件，还可以进行测试以验证适当的操作并发现有缺点的硬件。对每一个硬件设备进行验证的过程一般来讲需要硬件驱动程序，以使简化的 OS 映像与硬件设备进行通信。硬件驱动程序是一个硬件设备的制造商提供的专门软件，它可在计算机上安装的 OS 中运行，与硬件设备进行通信。在一个实施例中，简化的 OS 映像支持即插即用 (PNP) 协议以检测计算机中的硬件，并能够识别相应的硬件驱动程序。对于简化的 OS 映像，PNP 管理器服务也在运行。在内核初始化期间，PNP 管理器的内核模式部分将检测计算机上的各种硬件设备，并将检测到的硬件设备上的信息写入到注册表中的特定位置。在本发明的简化的 OS 映像中，用户模式 PNP 管理器被禁用，以便不自动安装检测到的硬件设备的驱动程序。这一点与其他 OS 映像 PNP 管理器的用户模式部分确定是否要安装检测到的硬件设备的驱动程序不同。如果硬件驱动程序没有安装，那么 PNP 管理器的用户模式部分将为这些硬件设备安装硬件驱动程序。

在本发明中，如果相应的硬件驱动程序正确地加载和初始化，那么简化的 OS 映像就会假设特定的硬件会正确地运行。还可以通过硬件制造商提供的测试套件对硬件进行其他测试。例如，用户可以向硬件驱动程序发送一个输入/输出控制 (IOCTL) 命令以与硬件进行通信。通过分析 IOCTL 命令返回的值，用户可以验证该硬件是否正常运行。作为另一个示例，要验证 ACME 网络接口卡 (NIC) 是否正常地运转，用户可以加载硬件驱动程序 `ACMENIC.SYS`。一旦 `ACMENIC.SYS` 成功地加载，用户还可以使用 `ACMENIC.SYS` 执行回送测试，以验证 NIC 是否正确地配置并正常运转。

OEM 提供的硬件驱动程序或者简化的 OS 映像内的硬件驱动程序用于控制硬件，并且它们是为供支持保护模式的 OS 使用而编写的。在一个实施例中，本发明的简化的 OS 映像是一个支持保护模式的 OS，如 WINDOWS 操作系统环境。简化的 OS 映像包括可用于计算机上出现的一些或所有可能的硬件的一些或所有硬件驱动程序。如此，简化的 OS 映像是独立于硬件的。在一个实施例中，至少有一个文本文件（如 `txtsetup.sif` 或 `txtsetup.oem`）列出了可能会出现在计算机中的所有可能的硬件可用的所有硬件驱动程序。用户通过向列表中添加或从中删除硬件设备对该文本文件进行编辑，以使该文本文件代表可能的硬件范围。通过最大限度地缩小硬件驱动程序列表，进行硬件检测和验证所需要的时间也会最大限度地缩短。

在示范 WINDOWS 操作系统实施例中，硬件驱动程序可以是启动型硬件驱动程序，也可以是非启动型硬件驱动程序。简化的 OS 映像自动加载启动型硬件驱动程序。启动型硬件驱动程序包括但不限于输入、存储（硬盘驱动器和文件系统），以及显示器驱动程序。由于简化的 OS 映像可以在各种计算机上启动，不同的计算机可能有不同的显示器、输入和存储设备。简化的 OS 映像包括可以在大多数计算机上运行的驱动程序的动态超集列表。安装型加载程序（在常规安装期间使用的加载程序）将加载驱动程序，这些驱动程序包括但不限于默认 VGA 驱动程序、用于各种经常使用的硬盘驱动器的通用海

量存储驱动程序、常规文件系统驱动程序以及用于键盘和鼠标的输入驱动程序。由于与 VGA 兼容的显示设备是大多数 WINDOWS 操作系统所必需的，因此默认的 VGA 驱动程序几乎可以在 WINDOWS 操作系统环境中的所有计算机上运行。安装型加载程序将作为启动型驱动程序加载这些驱动程序，创建一个列表，并将该列表传递到内核以将这些驱动程序初始化。在驱动程序初始化之前，将会在 SYSTEM 配置单元中的服务键值下面创建一个正确的条目。由于这些驱动程序是动态地加载的，因此在生成简化的 OS 映像期间无法创建必需的注册表条目。本发明的另一个驱动程序（如 setupdd.sys）将在启动型驱动程序初始化过程中执行，以在其他驱动程序初始化之前在注册表中创建必需的条目。这将有助于驱动程序的正确初始化。用户将按脚本指定的方式加载非启动型硬件驱动程序。

脚本将指导在 506 对海量存储设备进行分区。对海量存储设备的分区过程包括在 508 在海量存储设备上创建、删除特定区域或将它们格式化。在一个实施例中，简化的 OS 映像将引用一个文本文件（如 partinfo.txt）以获得特定的分区信息。用户对 partinfo.txt 进行编辑以添加用户所需要的特定的分区信息。脚本将用一个简化的 OS 映像支持的文件系统格式对海量存储设备进行格式化。例如，在 WINDOWS 操作系统中，受支持的文件系统格式包括 CDFS、UDF、FAT12、FAT16、FAT32 和 NTFS。

如果带有参考 OS 映像的 CRM 远离计算机，那么脚本将启用网络连接。脚本将执行一个实用程序（如 factory.exe）以检测计算机中的特定网络接口，并为该特定的网络接口安装相应的硬件驱动程序。factory.exe 实用程序将使用 PNP 管理器 API。网络驱动程序安装之后，脚本将通过一个实用程序（如 netcfg.exe）初始化传输控制协议/因特网协议 (TCP/IP) 堆栈以及相关的服务。TCP/IP 堆栈以及相关的服务将允许计算机通过一个 “net use”（例如）命令访问远程计算机。启用网络连接之后，脚本在 510 从远程 CRM 获取一个参考 OS 映像并在 512 将该参考 OS 映像安装在计算机上。在一个实施例

中，简化的 OS 映像支持网络连接协议（如服务器消息块，SMB）以便可以在各个计算机之间进行文件访问。

在计算机上安装参考 OS 映像的过程包括将与参考 OS 映像关联的一个或多个文件复制到计算机上，并将这些文件集成。在示范 WINDOWS 操作系统环境中，将文件集成的过程包括但不限于创建或更新注册表以及创建或更新桌面图标。脚本确定复制文件并进行集成的步骤。脚本也将确定执行步骤的顺序。在计算机上安装参考 OS 映像之后，可以将计算机关闭以便提供给客户或者在 514 重新启动。重新启动的过程包括关闭计算机，然后加载和初始化安装的参考 OS 映像。在一个实施例中，本发明提供了一个框架，根据该框架，在计算机上启动和安装参考 OS 映像只需要重新启动一次即可完成。即，脚本允许用户只需要一次重新启动即可将参考 OS 映像镜像到计算机上。本发明还可使用户将参考 OS 映像镜像到与计算机关联的一个 CRM 中。用户可通过脚本实现所需要的镜像方法。镜像方法可以是任何一种传送文件的方法，包括但不限于文件复制和完整的 OS 安装。重新启动之后，安装的引用 OS 镜像将为最终用户协调 OS 映像安装后的任何个性化工作。

可本地或远程访问的易失性和非易失性 CRM 构成了用于提供用户自定义的基于文本的脚本的装置。从本地 CRM 启动，或者通过 PXE 远程启动，或者任何其他远程启动协议构成了从 CRM 启动具有 32 位或 64 位体系结构的计算机的装置。示范 WINDOWS 操作系统环境中的应用程序（如 cmd.exe 和 csh.exe）构成了根据用户自定义的基于文本的脚本在计算机上执行一个或多个功能的装置。此外，上文描述的示例和其它地方描述的示例构成了提供脚本的装置和启动计算机的装置。

下面请看图 6，一个方框图说明了从一台远程启动计算机 602 启动目标计算机以在目标计算机 616 上安装参考 OS 映像 612 的过程。例如，在 WINDOWS 操作系统实施例中，远程启动计算机 602 是一台远程安装服务器。远程启动计算机 602 包括一个远程启动 CRM

604, 该 CRM 中包括一个简化的 OS 映像 606 和一个远程启动 CRM 脚本 608。在一个实施例中, 目标计算机 616 连接到远程启动计算机 602, 如图 1 中的示范网络环境所示。目标计算机 616 通过许多方法和协议 (包括但不限于启动之前执行环境, PXE) 从远程启动 CRM 604 进行启动。PXE 启动遵循动态主机配置协议 (DHCP)。目标计算机 616 将在网络上广播或以别的方式传达 DHCP 启动请求。远程启动计算机 602 接收到 DHCP 启动请求并从远程启动 CRM 604 启动目标计算机 616。启动目标计算机 616 的过程包括从远程启动计算机 602 将简化的 OS 映像 606 加载到目标计算机 616 上的易失性存储器, 并在目标计算机 616 上初始化简化的 OS 映像 606。

在 WINDOWS 操作系统中有一个适用于本地和远程启动的启动过程的示例。在根据本发明的 WINDOWS 操作系统实施例中, 简化的 OS 映像 606 包括与 WINDOWS NT 操作系统关联的基码, 并带有最低限度的 WIN32 应用程序编程接口 (API) 子系统。最低限度的 WIN32 API 子系统包括但不限于输入/输出 API 和核心 Win32 API。简化的 OS 映像 606 使用安装型加载程序而不使用启动型加载程序来加载简化的 OS 映像。用户在一个配置文件中指定一个加载程序标记, 以便让安装型加载程序用来识别远程启动 CRM 604 上的 OS 映像是简化的 OS 映像 606。在启动期间, 安装型加载程序将设置该标记。作为对设置的标记的响应, 简化的 OS 映像内的功能被启用。例如, 在 boot.ini 中指定了标记 “/minint”。“/minint” 标记类似于诸如 “/debug” 和 “/vga” 之类的其他加载程序。如此, 安装型加载程序将简化的 OS 映像 606 存在这一信息传达到简化的 OS 映像中的 OS 组件。OS 组件将按此处描述改变它们的行为以作为简化的 OS 映像 606 来运行。例如, 如上文所描述, 功能包括将配置信息加载到易失性存储器中。此外, 用户模式应用程序还根据设置的标记进行配置。在此实施例中, 对安装型 OS 映像中的 OS 组件的修改 (参见图 4, 引用字符 406) 在 OS 组件内存在, 但在被特

定的加载程序标记激活之前是被动的。

安装型加载程序将按照一个文本文件（如 `txtsetup.sif`）中指定的方式初始化各种硬件驱动程序。安装型加载程序还允许用户加载其他启动型硬件驱动程序和/或更新一个硬件抽象层，以促进简化的 OS 映像 606 和目标计算机 616 中的硬件之间的通信。

安装型加载程序将使用常规会话管理器而不使用与安装型加载程序关联的会话管理器。在启动期间，与简化的 OS 映像 606 关联的一个内核将执行一个实用程序（如 `smss.exe`）。当 `smss.exe` 执行时，它将查找在内核初始化期间创建的表示该 OS 是简化的 OS 映像 606 的注册表项。如果该注册表项存在，那么 `smss.exe` 将忽略通常在启动到 WINDOWS 操作系统环境中时执行的其他操作，这些操作包括但不限于创建一个分页文件。实用程序 `smss.exe` 还执行另一个实用程序（如 `winlogon.exe`）以查找该注册表项。如果该注册表项存在，那么 `winlogon.exe` 将注册表中指定的另一个进程（如 `cmd.exe`）而不是执行身份验证对话框和 `explorer.exe`。诸如 `cmd.exe` 或 `csh.exe` 之类的外壳支持 WINDOWS 操作系统内的脚本功能。是选择 `cmd.exe` 还是选择另一个外壳可以由用户在生成简化的 OS 映像 606 之前或之后进行配置。用户还可以指定一个自定义外壳环境作为简化的 OS 映像 606 的默认外壳。该外壳支持具有一个或多个脚本文件（如 `winBOM.ini` 文件或 `floppy.cmd` 文件）的脚本，以执行用户指定的功能。在此实施例中，简化的 OS 映像 606 将在简化的 OS 映像 606 中或其他位置（如用户指定的可移动的非易失性存储器）搜索该脚本文件。如果能找到该脚本，那么简化的 OS 映像 606 将在该脚本内执行命令。如果找不到该脚本，那么简化的 OS 映像 606 将显示一个命令提示符以接受用户输入的命令。例如，请看图 6，用户将指定是执行远程启动 CRM 脚本 608 还是执行引用 CRM 脚本 614，还是两者都执行，还是两者都不执行。

然后实用程序 `winlogon.exe` 等待 `cmd.exe` 或其他外壳进程完成。例如，`winlogon.exe` 可能等待二十四小时。如果外壳进程在二十

四小时之后还没有完成,那么 winlogon.exe 就会强制目标计算机 616 重新启动。实施时间限制是为了让用户不会将简化的 OS 映像 606 当作完全安装的 OS 映像来使用。

简化的 OS 映像 606 使用用户指定的脚本来指导参考 OS 映像 612 在目标计算机 616 上的安装。本发明允许用户配置安装过程,以便通过提供一个多级别的脚本环境实现自动安装。简化的 OS 映像 606 将使用远程启动 CRM 604、目标计算机 616 的本地 CRM、或目标计算机 616 可通过网络访问的 CRM 中的任何一个或任何组合中的一个代表脚本的文件,如 winBOM.ini。脚本环境允许用户协调许多计算机上任何一个脚本或许多脚本的执行。用户还可以在一个特定的脚本内指定另一个脚本的执行。例如,远程启动 CRM 脚本 608 的最后一个步骤就可以执行引用 CRM 脚本 614。

在目标计算机 616 启动之后,用户指定的脚本将查找包含参考 OS 映像 612 和引用 CRM 脚本 614 的引用 CRM 610。引用 CRM 610 位于目标计算机 616 本地或以别的方式连接到目标计算机 616,例如,如图 1 中的网络环境。脚本将通过把与参考 OS 映像 612 关联的文件复制到目标计算机 616 的本地海量存储设备并进行集成来将参考 OS 映像 612 安装在目标计算机 616 上。在其他实施例中,脚本首先用一个文件系统对海量存储设备进行分区和格式化。在目标计算机 616 上安装参考 OS 映像 612 之后,脚本将关闭或重新启动目标计算机 616。

下面是一个 winBOM.ini 脚本文件的示例,它与简化的 OS 映像 612 一起使用以将海量存储设备格式化,并将参考 OS 映像 612 安装在目标计算机 616 上。

[Version]

signature=\$version\$

[Factory]

[NetCards]

[WinPE]

Restart=Reboot

Lang=ENG

SKU=pro

Arch--x86

ConfigSet = Awesome 2001

SourceRoot=\\machinename\WhistlerOPK

Username=

Password=

[DiskConfig]

Disk1=Disk1 .config

[Disk1 .config]

Size1=*

PartitionType 1 =primary

FileSystem 1 =ntfs

QuickFormat 1 =yes

[OemRunOnce]

下面请看图 7, 一个方框图说明了从本地 CRM 704 启动目标计算机 702 以在目标计算机 702 上安装参考 OS 映像 712 的过程。本地 CRM 704 包括一个简化的 OS 映像 706 和一个本地 CRM 脚本 708。目标计算机 702 从本地 CRM 704 上的简化的 OS 映像 706 启动。

在启动之后，简化的 OS 映像 706 使用用户指定的脚本来指导在目标计算机 702 上安装参考 OS 映像 712 的过程。简化的 OS 映像 706 将查找用户指定的脚本来识别包含一个参考 OS 映像 712 和一个引用 CRM 脚本 714 的引用 CRM 710。引用 CRM 710 位于目标计算机 702 本地或者远离目标计算机 702，类似于图 6 中的示例。例如，请看图 7，用户将指定是执行本地 CRM 脚本 708 还是执行引用 CRM 脚本 714，还是两者都执行，还是两者都不执行。脚本将通过把与参考 OS 映像 712 关联的文件复制到与目标计算机 702 关联的海量存储设备并进行集成来将参考 OS 映像 712 安装在目标计算机 702 上。在其他实施例中，脚本首先用一个文件系统对海量存储设备进行分区和格式化。在目标计算机 702 上安装参考 OS 映像 712 之后，脚本将关闭或重新启动目标计算机 702。

下面请看图 8，一个方框图说明了带有故障软件 810 的计算机 808 与带有简化的 OS 映像 804(含有脚本 806)的恢复介质 802 之间的交互。用户将对脚本 806 进行自定义以恢复计算机 808 中故障软件 810。例如，脚本 806 可以是图 5、6、或 7 中的脚本。恢复介质 802 包含简化的 OS 映像 804 和脚本 806。计算机 808 上有一个软件（如 OS 或应用程序）发生了故障。故障软件 810 包括但不限于被损坏、有运行故障或有别的故障的软件。脚本 806 将根据用户指定的指令指导故障软件 810 的恢复，可以通过启动计算机 808，重新安装软件，向软件中添加纠正代码，或者以别的方式消除故障。在其他实施例中，恢复介质 802 可由计算机 808 通过网络进行访问。

在一个实施例中，用户使用一台引用计算机上的恢复介质 802 从引用计算机和其他计算机上的软件故障中恢复。其他计算机可以由引用计算机在本地或远程访问。如此，恢复介质 802 可用于自动地纠正许多计算机上的特定的类似的问题。例如，从故障软件中恢复一台计算机之后，脚本 806 可以将一个补丁程序应用到脚本 806 中标识的所有其他计算机。

在其他实施例中，带有脚本 806 的简化的 OS 映像 804 将与一

台或多台计算机进行交互以执行各种任务。例如，备份和/或恢复计算机上的数据，从一台执行简化的 OS 映像 804 的位于中心位置的服务器远程排除计算机的故障，以及使用简化的 OS 映像 804 作为远程服务器的瘦客户端以代替计算机使用。另一个示例是从一个只读介质执行反病毒软件以对另一个 OS 映像上的被病毒感染的文件以及用简化的 OS 映像中的或者只读介质上的安装型 OS 映像上的正确版本更换其他 OS 映像上的被病毒感染的文件。

综上所述，本发明的一些优势就已经体现出来了。由于在不偏离本发明的范围可以对产品和方法进行各种更改，因此，上述说明中包含的所有内容以及附图中所显示的内容都应该解释为说明性的，没有限制性的意义。

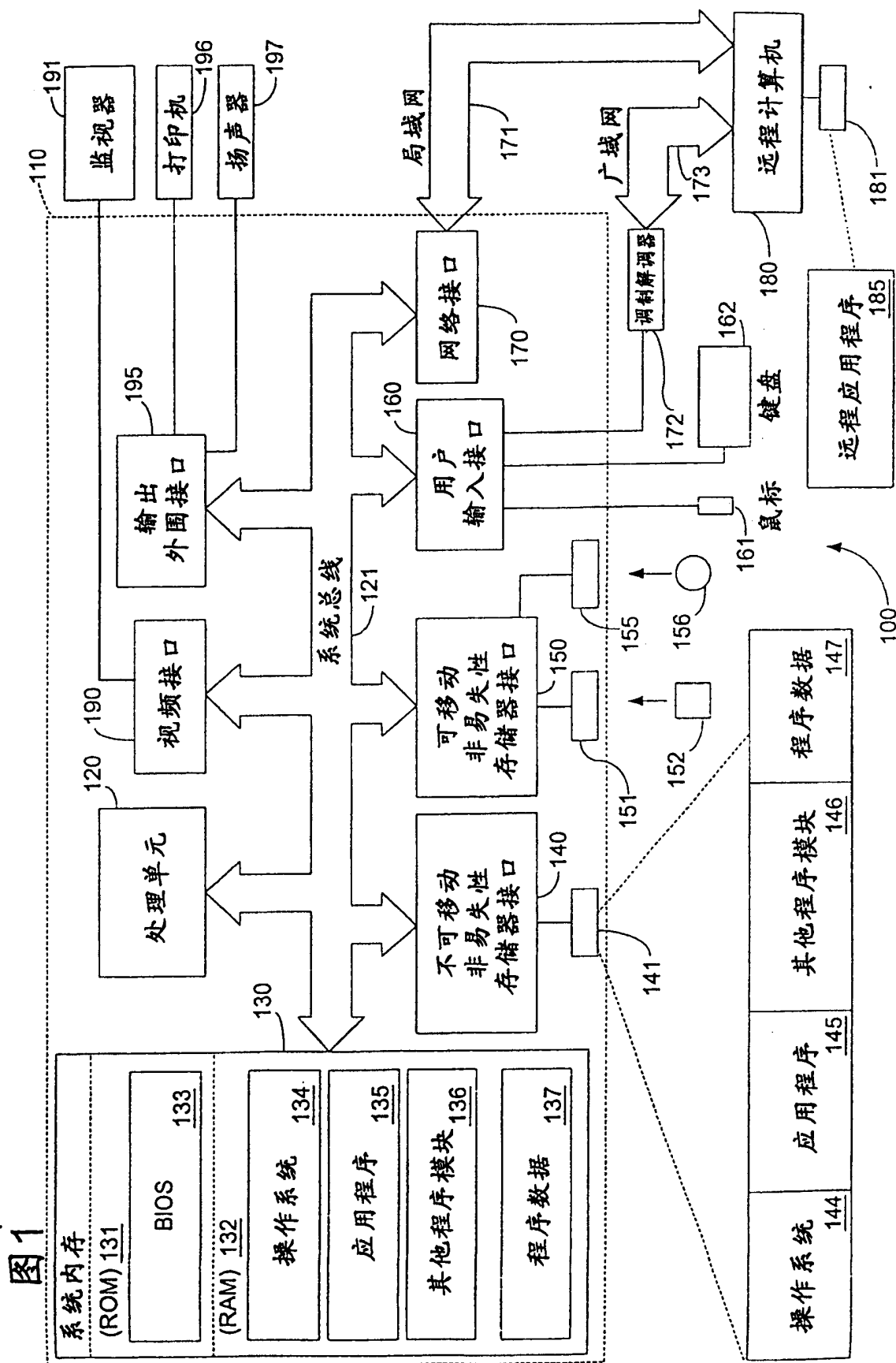


图3

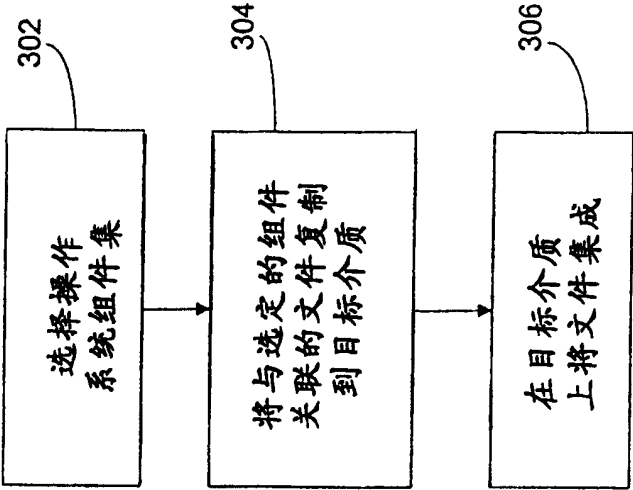
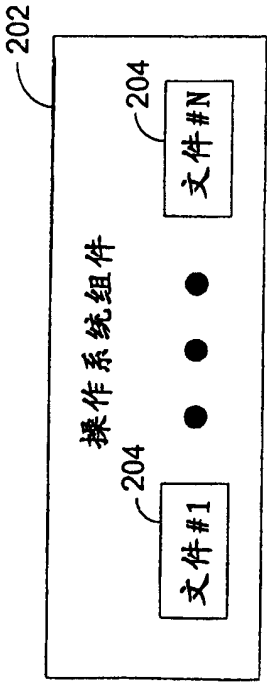


图2



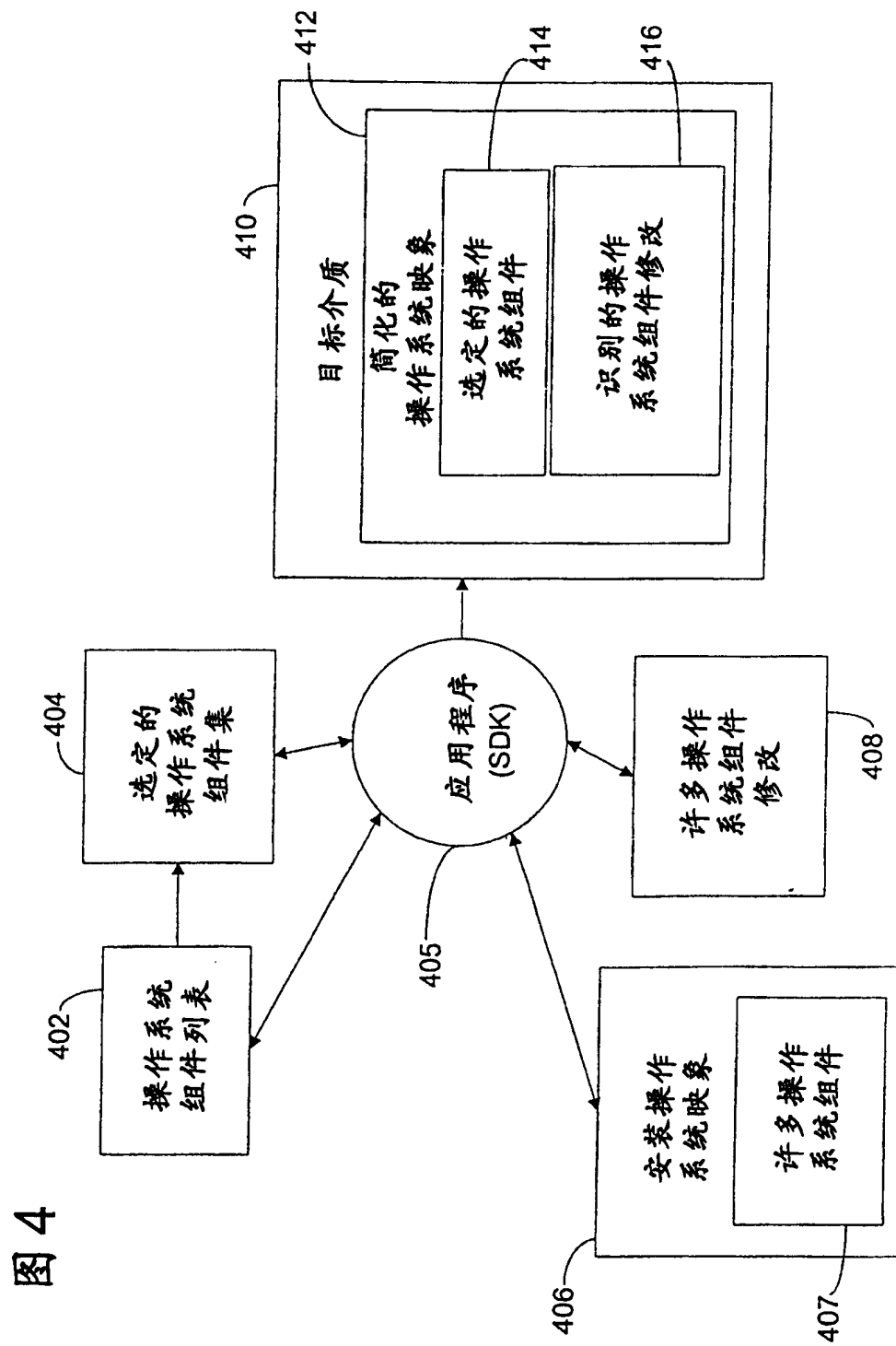


图5

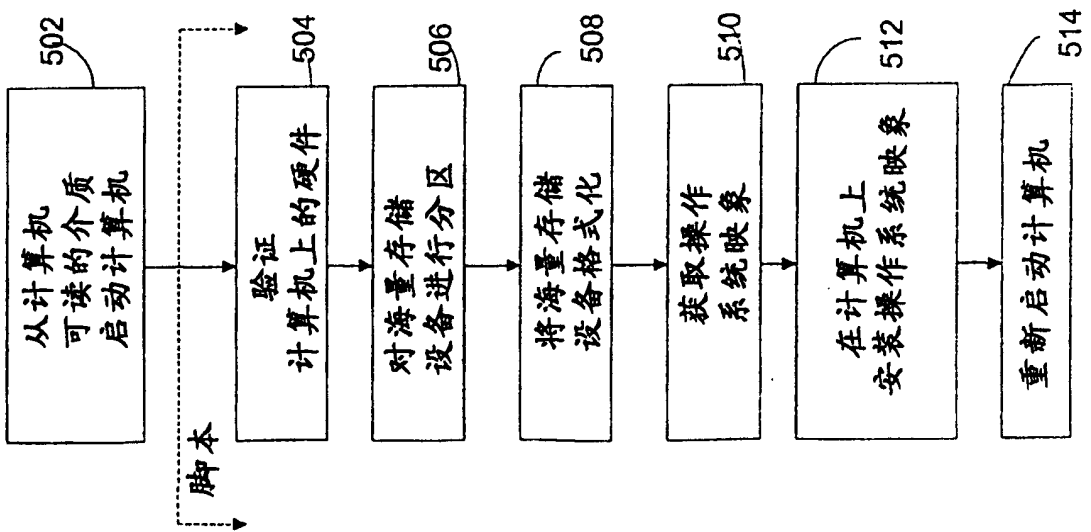


图6

