



[12] 发明专利申请公开说明书

[21] 申请号 200410061911.1

[43] 公开日 2005年2月2日

[11] 公开号 CN 1573701A

[22] 申请日 2004.6.24

[21] 申请号 200410061911.1

[30] 优先权

[32] 2003.6.24 [33] US [31] 10/603, 294

[71] 申请人 微软公司

地址 美国华盛顿州

[72] 发明人 S·洛德韦克 J·科恩

R·布克哈特

[74] 专利代理机构 上海专利商标事务所

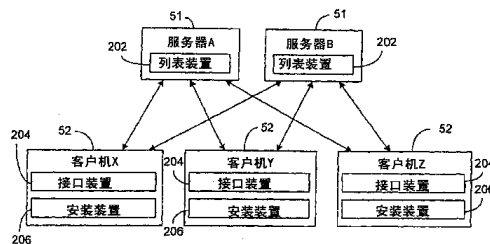
代理人 谢喜堂

权利要求书 3 页 说明书 15 页 附图 5 页

[54] 发明名称 分布式构建环境中的软件映像创建

[57] 摘要

在多个客户计算机之中分布软件映像创建和配置。服务器计算机定义与软件映像的创建相关的作业。每一客户计算机与服务器进行通信来标识、接受并完成作业。服务器和客户计算机维护指示作业可用性和状态的数据结构。在该分布式构建环境中，原始设备制造商(OEM)和系统构建者能够在工厂中简单地修改、创建软件并对其成像来显著地减少资源消耗和时间。



1. 一种分布式构建的系统，其特征在于，它包括：
维护一作业列表的第一计算机，每一所述作业有一与已安装的软件映像的创建相关联的操作；
5 第二计算机，从所述第一计算机接受所述作业之一并通过执行与其相关联的操作来执行所接受的作业。
2. 如权利要求 1 所述的系统，其特征在于，所述操作包括：
安装、配置软件映像以及对其进行成像。
- 10 3. 如权利要求 1 所述的系统，其特征在于，它还包括主计算机，所述第二计算机可通过它来从所述第一计算机接受所述作业之一。
4. 一种方法，用在一个或多个第一计算机向第二计算机授权软件映像的创建的分布式构建环境中，由所述第二计算机执行的所述方法包括：
从所述第一计算机之一接受作业，所述作业有与已安装的软件映像的创建相
15 关联的操作；以及
通过执行所述操作来完成所接受的作业，其中，所完成的作业代表已安装的软件映像。
5. 如权利要求 4 所述的方法，其特征在于，它还包括，从所述第一计算机之一请求所述作业。
- 20 6. 如权利要求 4 所述的方法，其特征在于，它还包括配置所述已安装的软件映像。
7. 如权利要求 4 所述的方法，其特征在于，所述第二计算机执行最小操作系统。
8. 如权利要求 4 所述的方法，其特征在于，所述接受包括：
25 接连地与所述第一计算机的每一个进行通信来标识与所述第一计算机相关联的多项作业；以及
边通信边选择所标识的多项作业中的第一项。
9. 如权利要求 8 所述的方法，其特征在于，每一所述第一计算机在与所述第一计算机相关联的特定文件中储存所述作业列表，并且其中，所述通信包括访问
30 每一所述第一计算机上的特定文件来标识所述作业。

10. 如权利要求 8 所述的方法, 其特征在于, 它还包括通过搜索与每一所述作业相关联并储存在所述第一计算机上的作业状态文件来确定每一所述第一计算机上的每一所述作业的状态。
11. 如权利要求 8 所述的方法, 其特征在于, 所述多项作业的每一个具有与其相应的优先级。
12. 如权利要求 11 所述的方法, 其特征在于, 所述多项作业的每一个根据其相应的优先级在列表中进行组织。
13. 如权利要求 4 所述的方法, 其特征在于, 所述完成包括执行与所述作业相关联的脚本。
- 10 14. 如权利要求 4 所述的方法, 其特征在于, 它还包括将所述已安装的软件映像发送到从中接受所述作业的第一计算机之一。
15. 如权利要求 4 所述的方法, 其特征在于, 所述已安装的软件映像代表一个或多个操作系统和应用程序。
16. 一种分布式构建环境中的数据结构, 在分布式构建环境中第一计算机向第二计算机授权已安装软件映像的创建, 所述数据结构储存在所述第一计算机上, 所述数据结构代表一项或多项作业, 所述数据结构包括:
- 与每一所述作业相关联的作业标识符; 以及
- 定义与所述作业标识符相关联、要由所述第二计算机执行来创建已安装软件映像的多个操作的脚本。
17. 如权利要求 16 所述的数据结构, 其特征在于, 所述每一作业具有与其相应的优先级, 并且其中, 所述每一作业根据其相应的优先级在所述数据结构中进行组织。
18. 如权利要求 16 所述的数据结构, 其特征在于, 所述脚本还定义了用于配置所述已安装软件映像的操作。
19. 一种用于分布式构建的系统, 在分布式构建中第一计算机向第二计算机授权已安装软件映像的创建, 所述系统包括:
- 用于第一计算机维护作业列表的列表装置, 每一所述作业与已安装的软件映像的创建相关;
- 用于第二计算机通过所述列表装置接受由所述第一计算机维护的所述作业之一的接口装置, 所述作业具有与已安装的软件映像的创建相关联的操作; 以及

用于第二计算机通过执行所述操作完成经所述接口装置接受的所述作业的安装装置，其中，所述完成的作业代表所述已安装的软件映像。

20. 如权利要求 19 所述的系统，其特征在于，它还包括用于第二计算机配置所述已安装的软件映像的装置。

分布式构建环境中的软件映像创建

5 技术领域

本发明的实施例涉及软件映像创建的领域，尤其涉及第一计算机向第二计算机授权软件映像创建的分布式构建环境。

背景技术

10 在典型的操作系统构建环境中，代码被计算机编译成二进制形式。这类环境无法使用外部资源来生成操作系统运行时刻映像。构建环境能够支持计算机内部的映像创建，但仅能使用有限的宝贵资源供应来满足最终操作系统映像的创建的需要。例如，一些现有环境专用并配置一个计算机来构建特定映像（即，操作系统或应用程序）。然而，这类系统效率较低，因为专用计算机仅用于构建特定映像（如
15 可能一天一次或一天两次）。由此，每一专用计算机在大部分时间都处于空闲状态。而且，当要创建的不同映像数增加时，所需要的处理硬件量也随之增加。例如，如果需要对7个不同产品的6个版本或配置构建映像，则需要42个专用计算机来构建所有的映像。初始硬件、维护以及对专用机器的支持的成本是基本的。需要一种能够使用外部资源来创建软件映像的系统。此外，还需要一种使用少量构建计算机
20 来有效地构建映像的系统。另外，由于操作系统的安装和配置通常在终端用户的计算机上执行，用户会遇到延迟和不便。需要一种安装和配置在映像创建过程中出现的系统。

一些现有系统向客户计算机分配任务来完成。例如，一些现有系统实现分布式编译，其中，客户计算机编译应用程序的一部分，中央服务器将编译的部分连接
25 为应用程序。在另一示例中，SETI@home项目将任务分布到客户计算机以分析对外星生物搜索中的数据。然而，在这类系统中，客户计算机并不创建最终产品。而且，这类系统与已安装的软件映像的创建无关。

因此，需要一种用于软件映像的分布式构建环境的系统来着眼于这些和其它
30 缺点中的一个或多个。

发明内容

本发明的实施例涉及分布式构建环境。在一个实施例中，本发明包括从第一计算机向第二计算机授权软件映像的安装和配置。在一个形式中，多个服务器计算机定义与软件映像安装和配置相关的作业。多个客户与服务器进行通信来标识、接受并完成作业。例如，本发明的分布式构建环境中的每一客户计算机可以连续地用来构建不同软件映像。在这一方式下，减少了创建映像所需的客户计算机的数量。

本发明提供了一种通过成像来安装和配置的公用方法。而且，由于安装和成像可由执行构建的客户定制，从而减少了操作系统安装次数。原始设备制造商(OEM)和系统构建者可以使用本发明的分布式成像进程在工厂中简便地修改、创建客户并对其成像，从而显著地减少资源消耗和时间。

依照本发明的一个方面，用于分布式构建的系统包括第一计算机和第二计算机。第一计算机维护作业的列表。每一作业具有与已安装的软件映像的创建相关联的操作。第二计算机从第一计算机接受一个作业并通过执行与其关联的操作来执行所接受的作业。

依照本发明的另一方面，一种方法在分布式构建环境中操作，在分布式构建环境中一个或多个第一计算机向第二计算机授权已安装的软件映像的创建。由第二计算机执行的该方法包括从第一计算机之一接受作业。该作业具有与已安装的软件映像的创建相关联的操作。该方法还包括通过执行该操作来完成所接受的作业。完成的作业代表已安装的软件映像。

依照本发明的再一方面，一种方法在分布式构建环境中操作，在分布式构建环境中第一计算机向一个或多个第二计算机授权已安装的软件映像的创建。由第一计算机执行的方法包括维护作业列表。每一作业与已安装的软件映像的创建相关。每一第二计算机选择一项或多项作业并执行所选择的作业。

依照本发明的再一方面，在第一计算机向第二计算机授权已安装的软件映像的创建的分布式构建环境中存在一种数据结构。该数据结构代表与软件映像的安装相关联的状态。该数据结构包括与第二计算机相关联的标识符。该数据结构也包括指示由标识符所标识的第二计算机所执行的安装状态的描述符。

依照本发明的另一方面，在第一计算机向第二计算机授权已安装的软件映像的创建的分布式构建环境中存在一种数据结构。该数据结构储存在第一计算机上。该数据结构代表一项或多项作业。该数据结构包括与每一作业相关联的作业标识

符。该数据结构还包括定义与作业标识符相关联的由第二计算机执行来安装软件映像的多个操作的脚本。

依照本发明的再一方面，一种系统提供了一种第一计算机向第二计算机授权已安装的软件映像的创建的分布式构建。该系统包括用于第一计算机的列表装置、
5 用于第二计算机的接口装置以及用于第二计算机的安装装置。列表装置维护作业列表。每一作业与所安装的软件映像的创建相关。接口装置通过列表装置接受一项由第一计算机维护的作业。该作业具有与软件映像的安装相关联的操作。安装装置通过执行该操作来完成通过接口装置接受的作业。完成的作业代表所安装的软件映像。

10 作为选择，本发明可以包括各种其它方法和装置。
其它特性将在后文中部分明确并部分指出。

附图说明

图 1 是客户/服务器网络系统的示例性实施例。

15 图 2 是说明依照本发明的客户和服务器的通信的示例性结构图。

图 3 是说明使用主服务器来对所有客户维护来自所有服务器的作业的示例性结构图。

图 4 是说明依照本发明的客户软件操作的示例性流程图。

图 5 是说明依照本发明的服务器软件操作的示例性流程图。

20 图 6 是说明客户计算机可访问的示例性计算机可读媒质的结构图。

图 7 是说明服务器计算机可访问的示例性计算机可读媒质的结构图。

图 8 是说明可实现本发明的合适计算系统环境的一个示例的结构图。

所有附图中，相应标号标识相应部件。

25 具体实施方式

在一个实施例中，本发明包括分布式构建环境，其中一个或多个第一计算机向一个或多个第二计算机授权软件映像的创建（如安装和配置）。具体地，至少一个服务器生成要由至少一个分布式客户完成的成映像工作项目列表（如作业）。该列表在服务器配置和生成构建的环境的基础上生成。在创建工作项目列表之后，任
30 一可用的成像客户机能够从任一服务器选择并完成工作项目。每一工作项目仅由一

个客户机选择并完成。客户机安装、配置操作系统并对其成像，然后将最终映像上传回服务器。服务器向客户机提供指令（如，通过脚本机制）来完成作业。当客户机完成作业之后，客户机用完成状态来更新服务器。然后客户机搜索另一作业来接受并完成。

- 5 本发明的分布式构建环境包括客户机上的最小配置，简化了客户机的重新分配和重新分布。该分布式环境向任一服务器分配额外的客户机资源。另外，该进程是可伸缩的（scalable）。因为一个成像客户机可以创建多个不同的产品映像，因此可以通过添加额外的成像客户机来减少构建次数。安装多个成像客户机并配置多个软件映像显著地提高了客户机资源使用率和效率。例如，现有系统中，如果需要
- 10 为 7 个不同的产品的 6 个版本或配置构建映像，则需要 42 个专用计算机来构建所有的映像。然而，采用本发明，单个计算机可以接连地构建每一映像，由此减少了处理硬件的数量和成本。

首先参考图 1，结构图说明了在本发明的分布式构建环境中使用的客户机/服务器网络系统的示例性实施例。图 1 示出了网络系统 50 包括多个服务器 51 和客户

15 机 52。这些计算机 51、52 通过网络 53 采用著名的网络化技术连接，用于高速数据通信。因特网是网络 53 的一个示例。服务器 51 从大量远程网络客户机 53 接受请求。服务器 51 通过网络 53 向客户机 52 提供包括数据在内的响应，然而也可以使用其它的通信方式。本发明参考服务器 51 和客户机 52 来描述，然而发明人认为本发明在其它网络系统中也可操作。也即，本发明并非局限于图 1 所示的客户机/

20 服务器网络系统 50。例如，本发明可以在对等网络系统中应用。

下面参考图 2，示例结构图说明了依照本发明的客户机 52 和服务器 51 之间的通信。在图 2 中，服务器计算机 51，如服务器 A 和服务器 B，各自与多个客户计算机 52，如客户机 X、客户机 Y 和客户机 Z 进行通信，以在分布式构建环境中授权所

25 安装的软件映像的创建。每一服务器计算机 51 担当第一计算机，操作以维护作业列表。每一作业与至少一个与已安装的软件映像的创建相关联的操作有关。例如，每一软件产品可能有一项作业，每一服务器 51 可能有多项作业。每一客户计算机 52 担当第二计算机，操作以从第一计算机接受一项作业并通过执行与其关联的操作来完成所接受的作业。具体地，该操作包括安装、配置软件并对其成像。第二计算机（如客户计算机 52）将已安装并已配置的软件映像传送到第一计算机（如服

30 务器计算机）。

在一个形式中，服务器计算机 51 和服务器软件组成用于维护作业列表的列表装置 202。客户计算机 52 和客户机软件组成用于从服务器计算机 51 接受一项作业的接口装置 204。客户计算机 52 和客户机软件也组成用于通过执行一个或多个与该作业相关联的操作来完成所接受的作业的安装装置 206。完成的作业代表已安装 5 的软件映像。与列表装置、接口装置和安装装置相应的结构还包括图示并在这里描述的元件。此外，与配置已安装的软件映像的装置（未示出）相应的结构包括客户机计算机 52、客户机软件以及图示并在这里描述的元件。

下面参考图 3，示例性结构图说明了使用主服务器 302 来维护来自所有服务器 51 的对所有客户机 52 的作业。在图 3 的示例实施例中，每一客户计算机 52 轮询 10 或定位主服务器 302。主服务器 302 维护可用服务器计算机 51 的列表，并有规律地（如基于请求或广播）向每一客户计算机 52 传递该列表。客户计算机 52 访问该列表来找出具有将要授权并完成的作业的服务器 51（如服务器 A）。该作业而后可以通过主服务器 302 或者通过服务器 51 和客户机 52 之间的对等连接授权给客户机 52。在另一实施例中，客户计算机 52 与多个主服务器 302 的至少一个进行通信。

下面参考图 4，示例性流程图说明了依照本发明的客户机软件的操作。客户机 15 软件包括计算机可执行指令，用于在 402 从服务器 51 请求作业、在 404 接受该作业并在 406 通过执行该操作来完成所接受的作业。接受作业包括接连地与每一服务器 51 进行通信来标识与服务器 51 相关联的一项或多项作业。每一服务器 51 在与服务器 51 相关联的特定文件（如图 7 中的 `relbuild.xml`）中储存可用作业的列表。 20 客户计算机 52 访问服务器 51 上的特定文件来标识可用作业。在一个形式中，每一服务器 51 也储存与每一作业相关联并储存在服务器 51 可访问的媒质上的作业状态文件（如图 7 中的 `job_id.xml`）。客户计算机 52 通过与作业相关联的作业状态文件对每一作业确定其可用性状态。例如，缺少作业状态文件指示该作业的可用性。

每一作业具有其相应的优先级。作业列表依照每一作业的优先级来组织。由 25 此，客户计算机 52 选择最高优先级的作业（如通过选择标识的作业的第一项）。另外或作为选择，客户计算机 52 可能接受由服务器 51 之一分配的作业。接受作业之后，客户计算机 52 创建一个与该作业相关联的文件（如图 7 中的 `job_id.xml`）来指示客户计算机 52 当前在执行与该作业相关联的操作。该文件储存在与客户计算机 52 和/或服务器计算机相关联的计算机可读媒质上。为如图 7 所示的在与服务 30 器计算机相关联的媒质上储存该文件，客户计算机 52 向服务器 51 提供证书来鉴别。

本领域的技术人员可以注意到，也存在其它方法用于指示一项作业被客户计算机 52 之一接受。例如，每一服务器 51 上的作业列表可能包括指示客户 52 已接受该作业的字段。发明人认为，所有这类方法都包含在本发明的范围之内。

客户计算机 52 也可以下载指令，如任务列表或脚本来完成作业。在另一实施
5 例中，指令包括软件组件，如完成作业中使用的库或其它对象文件。客户机软件包括指令，用于在 408 配置已安装的软件映像，并在 410 向其接受作业的服务器 51 发送回已安装的软件映像。已安装的软件映像代表准备好展开的操作系统和/或应用程序。

计算机 52 可访问的一个或多个计算机可读媒质具有用于执行图 4 所示的方法
10 的计算机可执行指令。在一个实施例中，客户计算机 52 在运行、最小操作系统环境的语境下执行指令。

在特定的示例中，客户机软件执行以连续地向（多个）服务器轮询已标帖的
作业。对特定服务器 51 标帖的每一作业，客户机软件确定该作业的当前状态。在一个实施例中，客户机软件搜索与作业相关联的作业状态文件（见图 7）的缺乏或
15 存在。如果每一作业的作业状态文件指示没有来自特定服务器 51 的可用作业，则客户机软件接连地向另外的服务器 51 轮询已标帖的作业。

如果来自特定服务器 51 有一个或多个可用的作业，则客户机软件执行以接受
其中一项作业。例如，客户机软件可以接受第一可用作业（如果按优先级排列）或接受由标帖作业的服务器 51 分配的作业。接受与服务器计算机 51 之一相关联的作
20 业之一之后，客户机软件使用证明在服务器计算机上创建文件（如图 7 中的 job_id.xml）来指示该作业已接受。客户机软件还可以在本地创建相应的状态文件。客户机软件执行与所接受的作业相关联的操作来完成该作业。例如，操作可以包含
在脚本中，来创建、安装、定制、配置和/或修改软件映像（如操作系统或应用程序的映像）。完成作业之后，客户机 52 储存以后能够发送到其它地方（如，到标
25 帖该作业的服务器计算机）的已安装并已配置的软件映像。

客户机 52 的一个示例性脚本包括操作，如以下的一个或多个：配置或格式化
硬盘驱动器、下载特定的文件用于设立、开始或装入设立、完成设立、激活客户机
52、启动进入最小操作系统环境、创建操作系统的映像以及将完成的映像上传到服
务器 51。

30 下面参考图 5，示例性流程图说明了依照本发明的服务器软件的操作。服务器

软件包括计算机可执行指令，用于在 502 动态地创建并维护作业列表。每一作业与已安装的软件映像的创建（如安装并配置软件映像）相关。服务器计算机将已维护的列表标帖为客户机计算机 52 可访问的网络资源（如在存储媒质上）。每一客户机计算机 52 选择一项或多项作业，并执行所选择的作业。该列表包括可用于客户机计算机 52 选择的作业、已选择的作业和已执行（如已完成）的作业。

维护包括根据用户输入将作业添加到列表以及在列表中组织作业。作业列表储存在服务器 51 和客户机计算机 52 可访问的文件（如图 7 中的 relbuild.xml）中。维护包括在 504 从客户机计算机 52 之一接收与客户机计算机 52 选择并执行的作业之一相应的已创建并已配置的软件安装。作为对接收已完成作业的响应，在 10 506，服务器计算机将该作业从列表中移除。软件安装可以包括操作系统和/或应用程序。一个或多个计算机可读媒质具有用于执行图 5 所示的方法的计算机可读指令。

如果在某一时间段内作业未能被客户机计算机 52 接受，服务器 51 通知作业的创作者。例如，该时间段可预设或可由创作者配置。创作者可以根据特定的作业选择时间段。在一个实施例中，服务器 51 在时间段过去之后从列表中移除未接受的作业。创作者或服务器 51 也可以指定一个时间段来完成所接受的作业。在一个实施例中，如果客户机 52 未能在该时间段内完成所接受的作业（如，客户机 52 遇到故障和中止），服务器 51 收回所接受的作业使其能够被其它客户机 52 所用。在从服务器 51 的故障中恢复之后，服务器 51 向客户机 52 重新标帖未完成的作业列表，客户机 52 回收其相应的作业，并且成像工作如服务器 51 故障之前继续。

在一个具体示例中，服务器软件执行以对每一作业区分优先级并储存在一个文件中，如 relbuild.xml。优先级可以基于最后期限或其它任一因素，并通过用户输入接收或储存在配置文件中。服务器 51 还执行服务器软件来引发安装共享并使 relbuil.xml 对用户计算机 52 可用。服务器软件等待作业完成。在一个形式中，25 从客户机 52 接收已完成的映像之后，服务器软件使用完成的软件映像来创建网络共享。

下面参考图 6，结构图说明了与客户机计算机 52 相关联的计算机可读媒质 602。媒质 602 储存分布式构建环境中的数据结构 604。数据结构 604 包括服务器路径 606 和证书 608。服务器路径 606 定义了到服务器 51 的路径（例如，网络地址如//server/install）。证书 608 与客户机 52 相关的标识符相关联，以在服务30

器 51 和客户机 52 之间的通信过程中进行鉴别。客户机 52 通过标识符、证书 608 和服务器路径 606 与服务器 51 进行通信，来向服务器 51 指示客户机 52 正在执行与软件映像的安装相关联的操作。

下面参考图 7，结构图说明了与服务器计算机 51 相关联的示例计算机可读媒质 702。媒质 702 储存分布式构建环境中的数据结构 704。数据结构 704 代表一项或多项作业。数据结构 704 包括作业标识符 706 以及脚本 708。作业标识符 706 与特定的作业相关联。脚本 708 定义了与作业标识符 706 相关联、将要由客户机 52 执行来安装软件映像的多个操作。如上所述，每一作业具有相应的优先级，并且在数据结构 705 中按照相应的优先级组织每一作业。脚本 708 还定义了配置已安装的软件映像的操作。

图 7 中的计算机可读媒质 702 储存另一数据结构 710。数据结构 710 代表与软件映像的安装相关联的状态。客户机 52 和服务器计算机 51 访问数据结构 710 来确定安装的状态。数据结构 710 包括客户机标识符 712 和安装状态 714 或其它描述符。客户机标识符 712 标识客户计算机 52。安装状态 714 指示由客户标识符 712 标识的客户计算机 52 执行的安装的状态。例如，安装状态 714 可能指示安装已完成或正在进行。如果安装仍在进行，则安装状态 714 还可指示完成的百分比和/或估计完成剩余时间。客户计算机 52 使用客户计算机 52 完成作业的状态来更新数据结构 710。尽管在图 7 中示出作业状态数据结构 710 储存在服务器计算机可读媒质 702 中，发明者认为它也可以替换或另外储存在图 6 的客户计算机可读媒质 602 中。

图 8 以计算机 130 的形式示出了一个通用计算装置的示例。在本发明的一个实施例中，计算机，如计算机 130，适合于在这里说明并描述的其它示图中使用。计算机 130 具有一个或多个处理器或处理单元 132 以及系统存储器 134。在说明的实施例中，系统总线 136 将包括系统存储器 134 的各类系统组件耦合至处理器 132。总线 135 代表一个或多个任一各类总线结构，包括存储器总线或存储器控制器、外围总线、加速图形端口以及使用任一各类总线结构的处理器或本地总线。作为示例而非限制，这类结构包括工业标准体系结构 (ISA) 总线、微通道结构 (MCA) 总线、增强 ISA (EISA) 总线、视频电子标准协会 (VESA) 本地总线以及外设部件互连 (PCI) 总线，也称为 Mezzanine 总线。

计算机 130 通常至少具有某些形式的计算机可读媒质。计算机可读媒质包括易失和非易失媒质、可移动和不可移动媒质，可以是任一可由计算机 130 访问的可

用媒质。作为示例而非局限，计算机可读媒质包括计算机存储媒质和通信媒质。计算机存储媒质包括易失和非易失、可移动和不可移动媒质，以任一用于储存如计算机可读指令、数据结构、程序模块或其它数据的信息的方法或技术实现。例如，计算机存储媒质包括 RAM、ROM、EEPROM、闪存或其它存储器技术、CD-ROM、数字多功能盘（DVD）或其它光盘存储、磁盒、磁带、磁盘存储或其它磁存储设备、或任一可用来储存期望的信息并可由计算机 130 访问的媒质。通信媒质通常配备计算机可读指令、数据结构、程序模块或在如载波或其它传输机制的已调制数据信号中的其它数据，并包括任意信息传送媒质。本领域的技术人员熟悉已调制数据信号，该信号以某一方式设定或改变其一个或多个特征来对信号中的信息进行编码。有线媒质，如有线网络或直接连线连接，是通信媒质的示例。上述任一组合也包含在计算机可读媒质的范围内。

系统存储器 134 包括以可移动和/或不可移动、易失和/或非易失存储器形式的计算机存储媒质。在说明的实施例中，系统存储器 134 包括只读存储器（ROM）138 和随机存取存储器（RAM）140。基本输入/输出系统 142（BIOS），包含如在启动时协助在计算机 130 内的元件之间传输信息的基本例程，通常储存在 ROM 138 中。RAM 140 通常包含处理单元 132 立即可访问和/或当前正在执行的数据和/程序模块。作为示例而非局限，图 8 说明了操作系统 144、应用程序 146、其它程序模块 148 和程序数据 150。

计算机 130 也可包括其它可移动/不可移动、易失/非易失计算机存储媒质。例如，图 8 说明了对可移动、非易失磁媒质进行读写的硬盘驱动器 154。图 8 还示出了对可移动、非易失磁盘 158 进行读写的磁盘驱动器 156 以及对可移动、非易失光盘 162，如 CD-ROM 或其它光媒质进行读写的光盘驱动器 160。其它可用在本示例操作环境中的可移动/不可移动、易失/非易失计算机存储媒质包括但不限于，磁带盒、闪存卡、数字多功能盘、数字视频带、固态 RAM、固态 ROM 等等。硬盘驱动器 154、磁盘驱动器 156 和光盘驱动器 160 通常通过非易失存储器接口，如接口 166 连接至系统总线 146。

以上描述并在图 8 中说明的驱动器或其它大容量存储设备及其关联的计算机存储媒质为计算机 130 提供了计算机可读指令、数据结构、程序模块和其它数据的存储。在图 8 中，例如，说明硬盘驱动器 154 储存操作系统 170、应用程序 172、其它程序模块 174 以及程序数据 176。注意，这些组件可以与操作系统 144、应用

程序 146、其它程序模块 148 和程序数据 150 相同，也可以与它们不同。这里对操作系统 170、应用程序 172、其它程序模块 174 和程序数据 176 给予不同的标号说明至少它们是不同的副本。

5 用户可以通过输入设备或用户接口选择设备，如键盘 180 和指向设备 182（例如，鼠标、轨迹球、笔或触摸板）来向计算机 130 输入命令和信息。其它输入设备（未示出）可包括麦克风、操纵杆、游戏板、圆盘式卫星天线、扫描仪等等。这些和其它输入设备通过耦合到系统总线 136 的用户输入接口 184 连接到处理单元 132，也可以通过其它接口和总线结构，如并行端口、游戏端口或通用串行总线(USB)连接。监视器 188 或其它类型的显示设备也通过接口，如视频接口 190 连接到系统
10 总线 136。除监视器 188 之外，计算机经常包括其它外围输出设备（未示出），如打印机和扬声器，可通过输出外围接口（未示出）连接。

计算机 130 可以在使用到一个或多个远程计算机，如计算机 194 的逻辑连接的网络化环境中操作。远程计算机 194 可以是个人计算机、服务器、路由器、网络 PC、对等设备或其它公用网络节点，并通常包括许多或所有上述与计算机 130 相关的
15 的元件。图 8 描述的逻辑连接包括局域网（LAN）196 和广域网（WAN）198，也可以包括其它网络。LAN 136 和/或 WAN 138 可以是有线网络、无线网络、其组合等等。这类网络化环境常见于办公室、企业范围计算机网络、企业内部互联网以及全球计算机网络（如因特网）。

当在局域网环境中使用时，计算机 130 通过网络接口或适配器 186 连接到 LAN
20 196。当在广域网环境中使用时，计算机 130 通常包括调制解调器 178 或其它装置用于通过 WAN 198，如因特网建立通信。调制解调器 178 可以是内置的也可以是外置的，通过用户输入接口 184 或其它合适的机制连接到系统总线 136。在网络化环境中，所描述的与计算机 130 相关的程序程序模块或其部分可以储存在远程存储器存储设备（未示出）中。作为示例而非局限，图 8 说明了远程应用程序 192 驻留在
25 存储器设备中。可以理解，这里示出的网络连接是示例性的，并且可以使用其它在计算机之间建立通信链路的装置。

通常，计算机 130 的数据处理器用不同时间储存在计算机的各类计算机可读存储媒质上的指令来编程。程序和操作系统通常分布在，例如，软盘或 CD-ROM 中。由此，它们被安装或装载到计算机的二级存储器。当执行时，它们至少部分地被装
30 载到计算机的初级电子存储器中。这里描述的本发明包括这些和其它各类计算机可

读存储媒质,这类媒质包含用于结合微处理器或其它数据处理器实现下文描述的步骤的指令或程序。本发明也包括计算机其本身,它被依照这里所述的方法和技术来编程。

为说明目的,程序和其它可执行程序组件,如操作系统,在这里示为不连续的块。然而,可以认可,这类程序和组件在各种时间驻留在计算机的不同存储组件中,并由计算机的(多个)数据处理器执行。

尽管结合示例性计算系统环境,包括计算机 130 来描述,本发明可与其它通用或专用计算系统环境或结构来操作。计算系统环境不意味着任何对本发明的使用或功能范围的局限。而且,不应将计算系统环境解释为具有关于示例性操作环境中示出的任一组件或其组合的依赖或需求。著名的适合与本发明使用的计算系统、环境和/或结构的示例包括但不限于,个人计算机、服务器计算机、手持式或膝上设备、多处理器系统、基于微处理器的系统、机顶盒、可编程消费电子产品、网络 PC、小型机、大型机、包含上述系统或设备的分布式计算环境等等。

本发明可以在由一个或多个计算机或其它设备执行的计算机可执行指令,如程序模块的一般语境下描述。通常,程序模块包括但不限于,例程、程序、对象、组件以及数据结构,执行特定的任务或实现特定的抽象数据类型。本发明也可以在由通过通信网络连接的远程处理设备执行任务的分布式计算环境中实践。在分布式计算环境中,程序模块可以驻留在本地和远程包括存储器存储设备的计算机存储媒质上。

在操作中,客户计算机 52,如计算机 130 执行如图 4 所示的计算机可执行指令,来安装并配置为软件映像。另外,服务器 51,如计算机 130 执行如图 5 所示的计算机可执行指令来创建并维护软件成像作业列表。

例如,服务器计算机 51,如计算机 130 可以在构建后(postbuild)进程中执行两个脚本来准备、例示并完成成像进程。构建后进程在编译和连接之后,但在释放构建之前出现。一个脚本(例如, stagerel.cmd)引发网络上的成像共享并监控作业完成的阶段目录。在成像开始之后,服务器计算机 51 无限地等待客户机 52 上传完成的映像。阶段目录中的作业状态文件储存成像进程的当前进度。作业状态文件中的字段 MAJORSTATE(见下文)储存至少两个状态中的其中一个: INPROGRESS 和 FINISHED。所有作业完成之后,脚本通过降低共享并继续移动到下一构建后命令来继续。如果共享无法引发或者如果检测到之前的构建后错误,则脚本不会等待

映像。

另一脚本（例如，img_createskus.cmd）在构建后进程中执行两次。当第一次装入脚本时（即，在其它脚本 stagerel.cmd 之前），它创建名为 relbuild.xml 的文件，包含成像机器（例如，客户计算机 52）所需要的服务器特定的信息和客户机作业。这一脚本第二次在 stagerel.cmd 之后运行，并将 SKU/产品拼装在一起，将所得映像放置在特定目录中。stagerel.cmd 脚本等待客户计算机 52 完成映像，并监控储存作业状态文件的成像目录。在所有作业都被取走并完成之后，stagerel.cmd 脚本继续。映像创建可通过观察记录文件来检验。

当客户机 52 执行作业时，它可以通过删除本地客户机作业状态文件并重新启动客户机 52 来被回收。客户机 52 接受的成像作业也可以通过回收客户机 52 并删除服务器作业状态文件来重新开始。

以下储存在与服务器之一关联的计算机可读媒质上的数据结构示例进一步说明了本发明。以下示出了与 relbuild.xml（见图 6）相应的示例性服务器配置模式。尽管为方便，仅在模式中示出了一项作业，可以认为该模式可以定义多项作业。

```

15  <SERVER>
        <BUILDTIME></BUILDTIME>
        <BUILDNUMBER></BUILDNUMBER>
        <LOGFILE></LOGFILE>
        <TITLE></TITLE>
20  <JOBS>
        <JOB ID="">
                <TITLE></TITLE>
                <CONFIGTYPE></CONFIGTYPE>
                <INSTALLSCRIPT></INSTALLSCRIPT>
25  <IMAGESCRIPT></IMAGESCRIPT>
                <TIMEOUT></TIMEOUT>
        </JOB>
        </JOBS>
    </SERVER>

```

30 下列填充的数据结构代表服务器配置模式的一个具体示例。


```

<SERVER>
  <BUILDTIME>20021016:18:00:00</BUILDTIME>
  <BUILDNUMBER>3700</BUILDNUMBER>
  <LOGFILE>relbuild.log</LOGFILE>
5  <TITLE>Main Build 3700</TITLE>
  <JOBS>
    <JOB ID="1">
      <TITLE>Professional Installation</TITLE>
      <CONFIGTYPE>PRO</CONFIGTYPE>
10  <INSTALLSCRIPT>pro/install.cmd</INSTALLSCRIPT>
    <IMAGESCRIPT>pro/image.cmd</IMAGESCRIPT>
    <TIMEOUT>3600</TIMEOUT>
    </JOB>
  </JOBS>
15  </SERVER>

```

以下示出了与 job_id.xml（见图 7）相应的示例作业状态模式。

```

<JOBSTATUS>
  <CLIENTID></CLIENTID>
  <TIMESTART></TIMESTART>
20  <TIMECOMPLETE></TIMECOMPLETE>
  <CLIENTSTATUS>
    <MAJORSTATE></MAJORSTATE>
    <MINORSTATE></MINORSTATE>
  </CLINTSTATUS>
25  </JOBSATUS>

```

以下填充的数据结构代表作业状态模式的一个具体示例。

```

<JOBSTATUS>
  <CLIENTID>RACK05_MACHINE10</CLIENTID>
  <TIMESTART>20021016:18:01:00</TIMESTART>
30  <TIMECOMPLETE></TIMECOMPLETE>

```

```

    <CLIENTSTATUS>
        <MAJORSTATE>install</MAJORSTATE>
        <MINORSTATE>format</MINORSTATE>
    </CLIENTSTATUS>

```

5 </JOBSATUS

以下储存在与客户计算机 52 之一关联的计算机可读媒质上的数据结构示例说明了本发明。以下示出与 buildclient.xml (见图 6) 相应的示例性客户机配置模式。

```

    <BUILDCLIENT ID="">
10     <WAITTIME></WAITTIME>
        <CONFIGINCLUDE></CONFIGINCLUDE>
        <REDIRECTS>
            <REDIRECT>
                <USERNAME></USERNAME>
15                <PASSWORD></PASSWORD>
                <CONFIGLOC></CONFIGLOC>
            </REDIRECT>
        </REDIRECTS>
    </BUILDCLIENT>

```

20 以下填充的数据结构代表客户机配置模式的一个具体示例。

```

    <BUILDCLIENT ID="RACK05_MACHINE10">
        <WAITTIME>60</WAITTIME>
        <CONFIGINCLUDE>PRO</CONFIGINCLUDE>
        <REDIRECTS>
25         <REDIRECT>
            <USERNAME>redmound\bvt</USERNAME>
            <PASSWORD>bvtpassword</PASSWORD>
            <CONFIGLOC>\\ntre101\bvt$</CONFIGLOC>
        </REDIRECT>
30     </REDIRECTS>
    </BUILDCLIENT>

```

</BUILDCLIENT>

当介绍本发明或其（多个）实施例的元件时，冠词“一个”、“该”以及“所述”意指有一个或多个元件。术语“包括”、“包含”以及“具有”为包含性的，并意指除列出的元件外还有其它另外的元件。

5 综上所述，可以看到，达到了本发明的若干个目的，并获得了其它优点。

在不脱离本发明的范围的情况下可以在上述构造、产品和方法中作出各种改变，所有包含在以上描述以及在附图中示出的事物都应该解释为说明性的，而非局限。

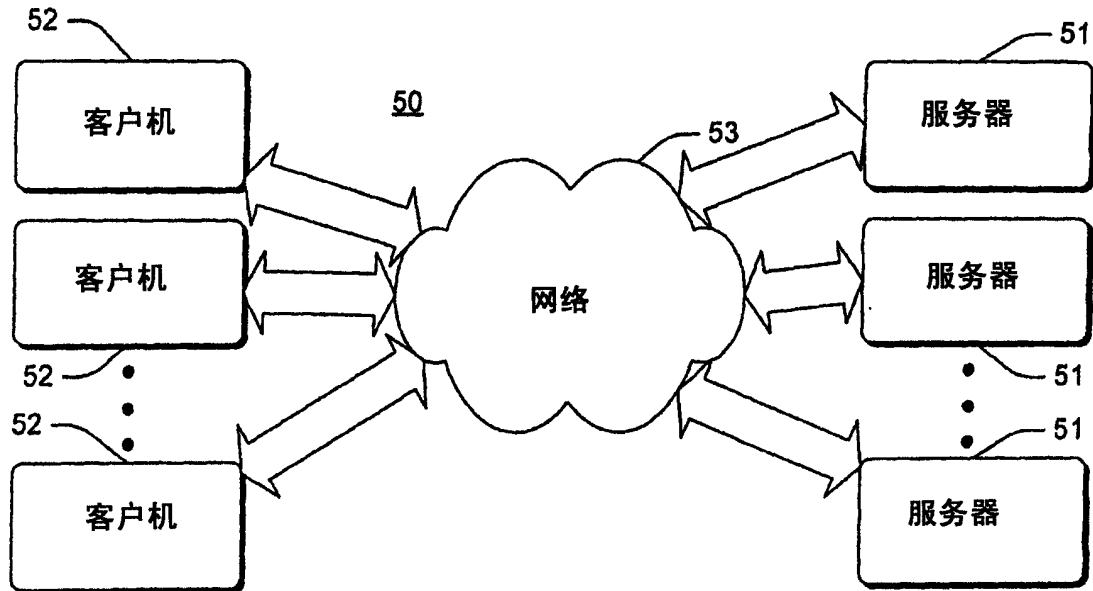


图 1

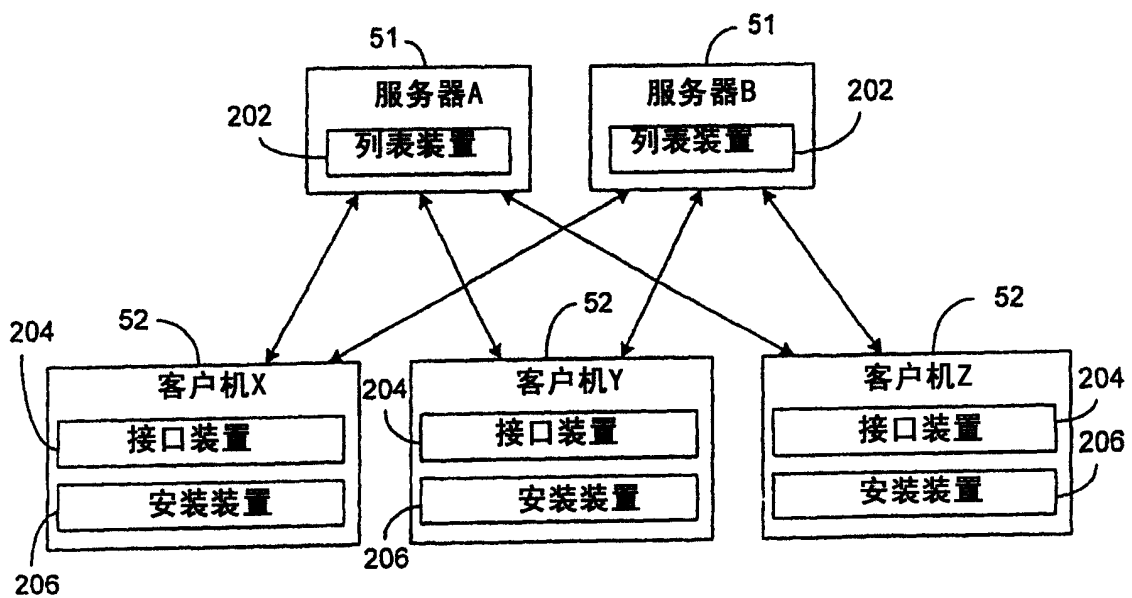


图 2

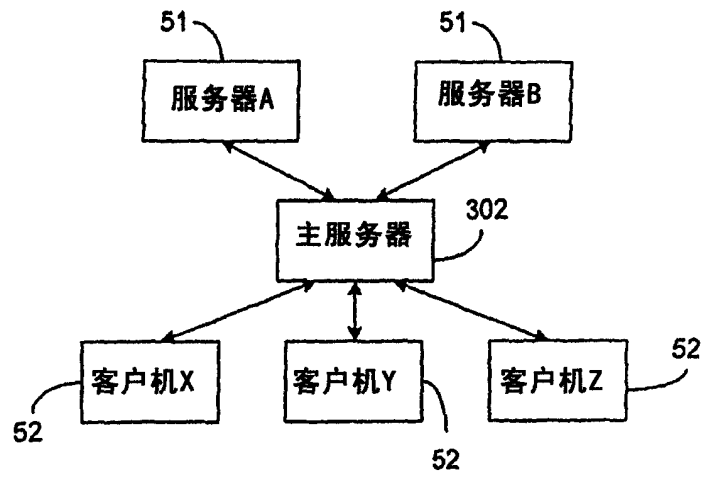


图 3

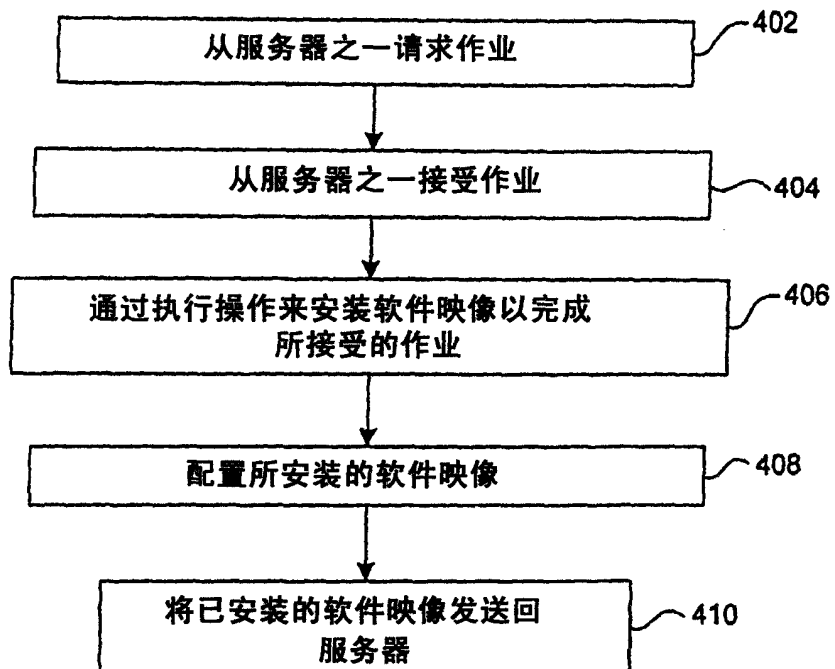


图 4

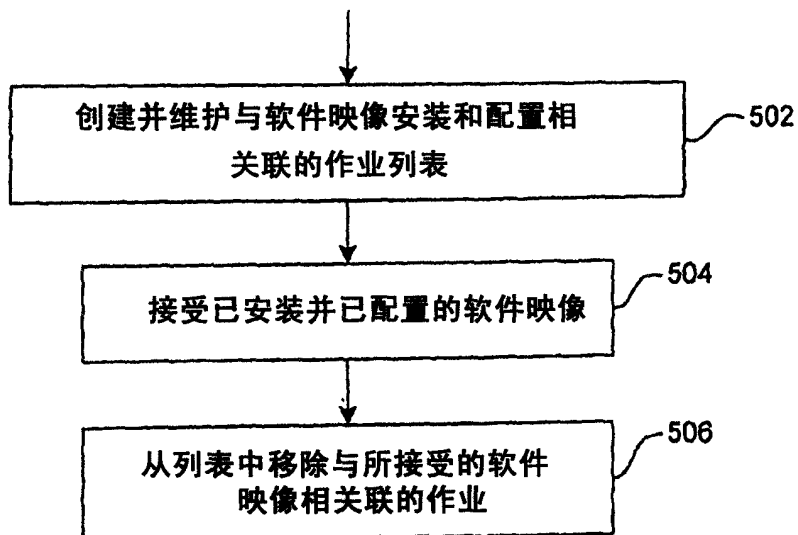


图 5

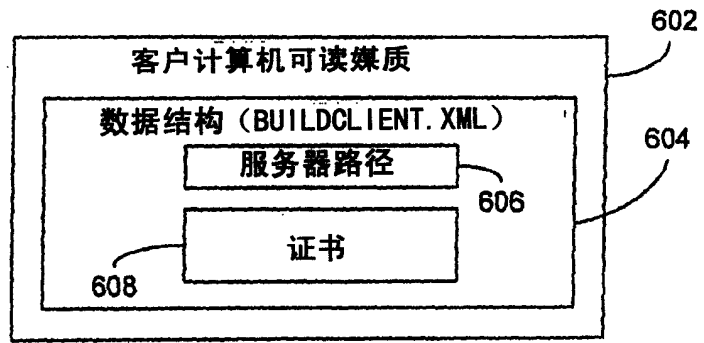


图 6

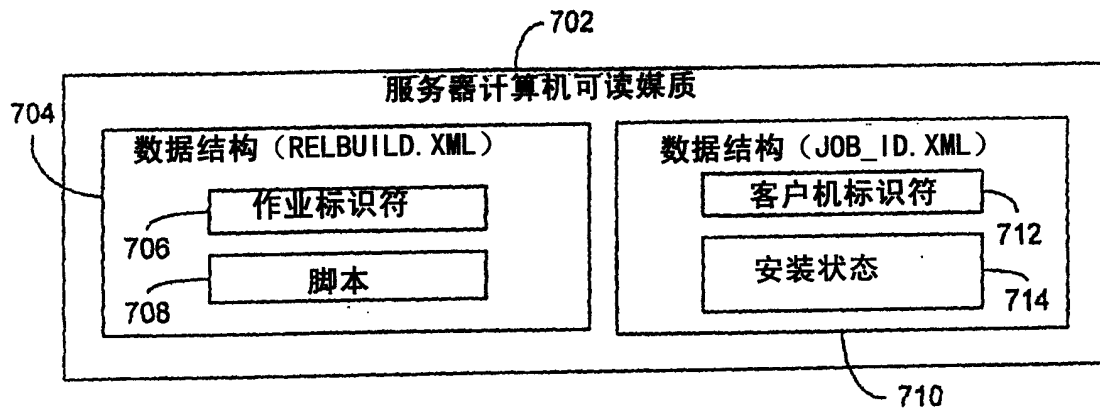


图 7

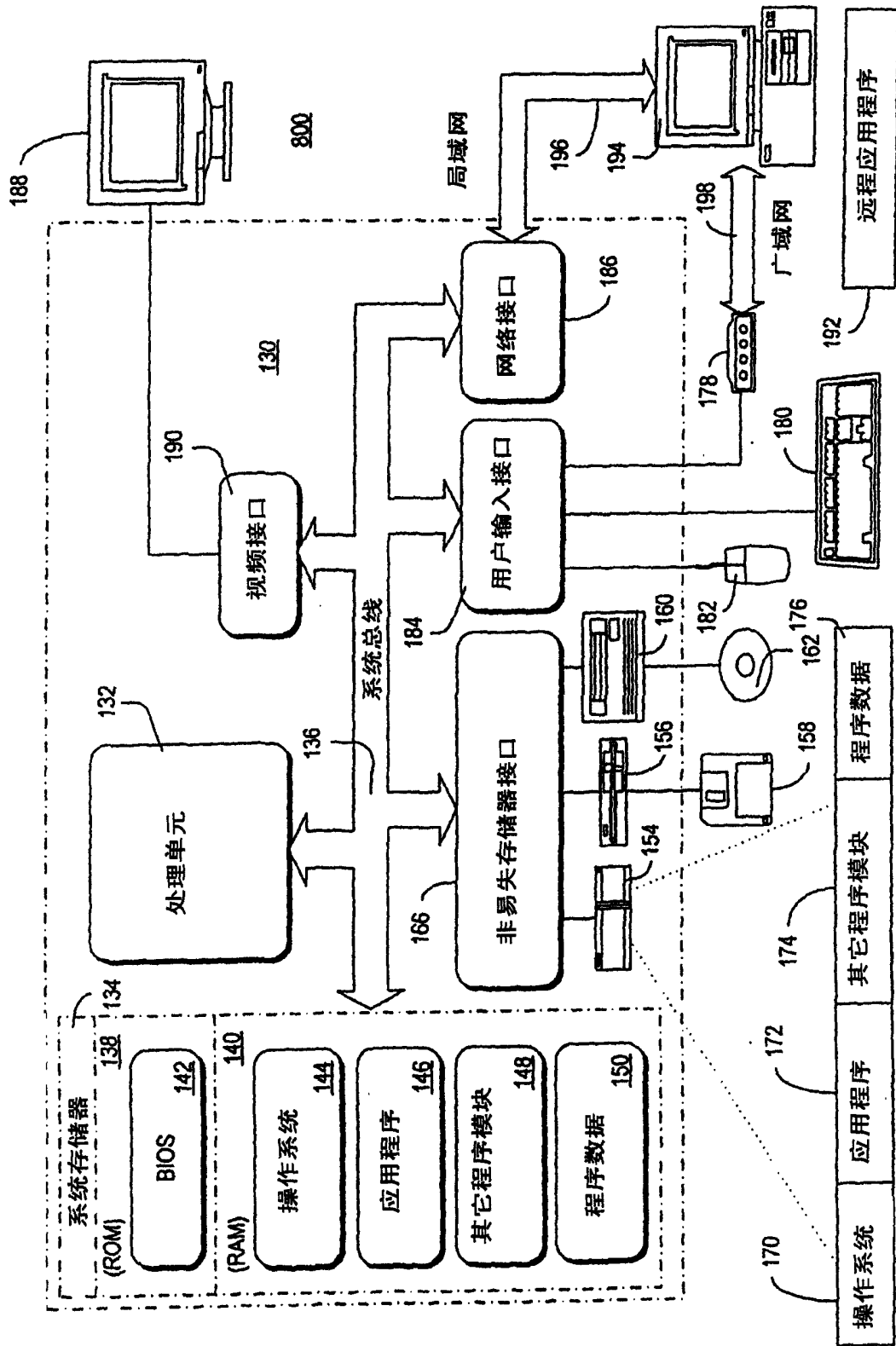


图 8