

1. 一种计算机实现的方法,包括:

自动发现软件应用的一个或多个配置项;

打包所述应用以形成虚拟应用包;

部署虚拟应用包以实例化所述应用;以及

在经实例化的所述应用中应用部署配置,所述部署配置包括用于所述一个或多个配置项的一个或多个设置,其中部署所述虚拟应用包以及应用所述部署配置交错进行。

2. 如权利要求 1 所述的方法,其特征在于,所述一个或多个配置项是一个或多个自动发现的配置项,并且其中所述方法还包括响应于标识一个或多个手动发现的配置项的用户输入发现所述软件应用的所述一个或多个手动发现的配置项,所述部署配置还包括用于所述一个或多个手动发现的配置项的一个或多个设置。

3. 如权利要求 1 所述的方法,其特征在于,自动发现包括将所述发现的至少一部分托付给与所述一个或多个配置项中的至少一个有关的子系统。

4. 如权利要求 1 所述的方法,其特征在于,所述部署配置是第一部署配置,并且其中所述方法还包括在所述经实例化的应用中应用第二部署配置,以改变用于一个或多个配置项的一个或多个设置,所述第二部署配置与所述第一部署配置不同。

5. 如权利要求 1 所述的方法,其特征在于,还包括通过用户接口设备来呈现所述一个或多个配置项。

6. 如权利要求 5 所述的方法,其特征在于,还包括接收用户输入,以及响应于所述用户输入编辑所述一个或多个配置项中的一个。

7. 如权利要求 1 所述的方法,其特征在于:

部署所述虚拟应用包以实例化所述应用包括:部署所述虚拟应用包以将所述应用实例化为所述应用的第一实例;

应用所述部署配置包括:在所述应用的所述第一实例中应用所述部署配置;

所述方法还包括部署所述虚拟应用包以将所述应用实例化为所述应用的第二实例;以及

所述方法还包括在所述应用的第二实例中应用所述部署配置。

8. 如权利要求 7 所述的方法,其特征在于,在所述应用的所述第一实例中应用所述部署配置以及在所述应用的所述第二实例中应用所述部署配置包括在所述应用的所述第一实例和所述第二实例中应用所述一个或多个配置项中的至少一个的不同设置。

9. 如权利要求 1 所述的方法,其特征在于:

所述方法还包括在打包所述应用之后执行以下操作:

通过用户接口设备呈现所述配置项;

接收用户输入;

响应于所述用户输入编辑所述一个或多个配置项中的至少一个的设置;

部署所述虚拟应用包以实例化所述应用包括:部署所述虚拟应用包以将所述应用实例化为所述应用的第一实例;

应用所述部署配置包括在所述应用的所述第一实例中应用所述部署配置;

所述方法还包括部署所述虚拟应用包以将所述应用实例化为所述应用的第二实例;以及

所述方法还包括在所述应用的所述第二实例中应用所述部署配置,其中在所述应用的所述第一实例和所述第二实例中对所述一个或多个配置项中的至少一个应用不同的设置。

10. 一种计算机实现的方法,包括:

打包软件应用以形成虚拟应用包;

在打包所述软件应用之后设置所述虚拟应用包的部署配置;

部署虚拟应用包以实例化所述应用;以及

在经实例化的所述应用中应用所述部署配置,其中部署所述虚拟应用包以及应用所述部署配置交错进行。

11. 如权利要求 10 所述的方法,其特征在于,设置所述部署配置包括编辑所述部署配置中的一个或多个配置项。

12. 如权利要求 10 所述的方法,其特征在于,设置所述部署配置是响应于接收到用户输入来进行的。

13. 如权利要求 10 所述的方法,其特征在于,所述方法还包括在完成所述打包之前发现所述部署配置的一个或多个配置项。

14. 如权利要求 13 所述的方法,其特征在于,所述方法还包括:将所述一个或多个配置项中的至少一个标记为任选或强制的,并且其中设置所述部署配置包括对所述一个或多个配置项中的每一个判断所述配置项是否被标记为强制的,并且如果所述配置项被标记为强制的,则在部署所述虚拟应用包之前要求针对所述配置项的用户输入。

15. 如权利要求 10 所述的方法,其特征在于,

所述部署配置是第一部署配置;

部署所述虚拟应用包以实例化所述应用包括:部署所述虚拟应用包以在所述应用的第一实例中实例化所述应用;

在所述经实例化的应用中应用所述部署配置包括:在所述应用的所述第一实例中应用所述第一部署配置;

所述方法还包括:

在打包所述软件应用之后设置所述虚拟应用包的第二部署配置;

部署所述虚拟应用包以在所述应用的第二实例中实例化所述应用;以及

在所述应用的所述第二实例中应用所述第二部署配置。

用于部署虚拟软件应用部署配置的方法

[0001] 背景

[0002] 虚拟应用可以一系列操作来实现。最初,应用可在打包环境中被打包以生成虚拟应用包。打包是取得应用并定义其数据(诸如文件和注册表条目)和元数据(配置项,诸如 *.ini 文件中的项目和注册表设置)的过程,以使得包可包括用以在没有作为本机应用被实际安装的情况下在目标环境中运行的应用数据和元数据。因此,该包可被部署到目标环境以实例化虚拟应用实例。虚拟应用实例可在胶囊(capsule)中运行,该胶囊包括应用以及用于运行该应用的虚拟资源。这样的胶囊在本文中称为“沙箱”。如本文中所使用的,包包括从对应用进行打包获得的信息(文件、元数据等),并且该包可被部署以实例化应用。包可被拆分成多个部分,并且一些部分可以不同的方式和/或在不同的时间被传送。

[0003] 概述

[0004] 本文中描述并要求保护的工具和技术可允许部署虚拟应用包和部署配置以实例化应用的特定实例。部署配置可包括一个或多个配置项,诸如在打包应用时发现的项目等。部署配置可与虚拟应用包一起被包括或与之分开,而部署配置可在应用被打包之后被编辑。部署配置可被应用于经实例化的虚拟应用中。例如,可在部署应用的同时和/或在应用已被部署之后应用部署配置来配置应用。

[0005] 在一个实施例中,工具和技术可包括自动发现软件应用的配置项,以及打包应用以形成虚拟应用包。可部署虚拟应用包以实例化应用。可在经实例化的应用中应用部署配置,其中部署配置包括配置项的设置。

[0006] 在工具和技术另一实施例中,可打包软件应用以形成虚拟应用包。可在打包应用之后设置用于应用的部署配置。设置部署配置包括确认和/或改变配置。例如,用于部署配置中的配置项的一个或多个默认配置可被自动确认或响应于特定用户输入来确认。作为另一示例,用于配置项的一个或多个设置可被添加或修改,诸如自动地或响应于特定用户输入来添加或修改。可部署虚拟应用包以实例化应用,并且可在经实例化的应用中应用部署配置。例如,可在部署应用包的同时(例如,通过将部署和配置操作交错进行)和/或在已部署应用包之后,在应用中应用部署配置中的配置项的设置。

[0007] 在工具和技术的又一实施例中,可发现软件应用的配置项。可打包应用以形成虚拟应用包。在打包应用之后,可通过用户接口设备来呈现配置项。可响应于指示改变的用户输入来改变所选配置项。另外,可部署虚拟应用包以实例化应用,并且可将配置项应用于经实例化的应用。

[0008] 提供本概述是为了以简化的形式介绍一些概念。这些概念将在以下详细描述中进一步描述。本概述并不旨在标识所要求保护主题的关键特征或必要特征,也不旨在用于限制所要求保护主题的范围。类似地,本发明不限于解决在背景、详细描述、或附图中讨论的特定技术、工具、环境、缺点、或优点的实现。

[0009] 附图简述

[0010] 图1是其中可实现所描述的各实施例中的一个或多个实施例的合适的计算环境的框图。

[0011] 图 2 是虚拟应用部署配置计算环境的框图。

[0012] 图 3 是虚拟应用部署配置技术的流程图。

[0013] 图 4 是另一虚拟应用部署配置技术的流程图。

[0014] 图 5 是又一虚拟应用部署配置技术的流程图。

[0015] 详细描述

[0016] 本文中描述的实施例涉及用于虚拟应用的改进的打包、部署、和 / 或实例化的技术和工具。这样的改善可源于分开或组合地使用各种技术和工具。

[0017] 此类技术和工具可包括允许部署虚拟应用包和部署配置,以在目标环境中实例化应用的特定实例。例如,可自动发现软件应用的一个或多个配置项。包括各项目的部署配置可在应用被打包为虚拟应用包之后被设置。例如,在打包应用之后,配置项可通过用户接口设备(例如,通过在计算机显示器上显示配置项的表示)来呈现给用户。在打包之后,还可响应于指示作出改变的用户输入改变配置项中的一个。可部署包以实例化应用的实例,并且可在应用实例中应用部署配置。

[0018] 在打包时间,打包工程师(进行打包的用户)和 / 或打包器(执行打包的计算组件)可知晓一些配置项是什么,但是可能不知晓那些配置项的设置特定应用实例中将会如何。例如,一些配置项可被绑定到特定机器、用户或应用实例。作为一个特定示例,用户接收度测试组织可将应用连接到名为“sql_qa”的数据库。然而,当应用在其预期生产目标环境中运行时,应用可连接到名为“sql_production(sql_生产)”的数据库。通过在打包时间标识配置项,但在部署包以实例化应用的特定实例时设置和应用那些配置项,可使进行部署的管理员(进行部署的用户)免于必须标识配置项。然而,管理员仍可输入配置项的恰当设置。而且,可部署单个包来就不同的机器和 / 或用户实例化不同的实例。这可在无需修改包自身(除了在配置项被包括在包中的情况下可能修改配置项之外)且无需重新打包应用的情况下进行。例如,这可能是有益的,因为可打包并测试一个通用包,但是此经测试的通用包可被部署到数个具有不同部署配置设置的不同实例。如果对于每个部署,包中除部署配置以外的项被修改,或者如果应用被整体重新打包,则原始测试可能不是有效的。

[0019] 由此,从此处描述的工具和技术中可以实现一个或多个实质的益处。所附权利要求中定义的主题不必限于本文描述的益处。本发明的特定实现可提供本文描述的益处的全部、一些、或未提供本文描述的益处。尽管本文出于呈现的目的以特定的顺序次序描述了用于各种技术的操作,但应理解除非要求特定的排序,否则这种描述方式涵盖了操作顺序上的重新安排。例如,在某些情况下,可以重新安排或并发执行顺序地描述的操作。本文参照流程图描述的技术可被用于本文描述的一个或多个系统和 / 或用于一个或多个其他系统。例如,本文描述的各种过程可用硬件或软件、或两者的组合来实现。此外,为了简单起见,流程图可能未示出可结合其他技术来使用特定技术的各种方式。

[0020] 示例性计算环境

[0021] 图 1 示出其中可实现所描述的各实施例中的一个或多个的合适的计算环境(100)的通用示例。例如,可将一个或多个这样的计算环境用作打包环境、部署环境、和 / 或目标环境。一般而言,可使用各种不同的通用或专用计算系统配置。适用于此处所描述的工具和技术的公知计算系统配置的示例包括,但不限于,服务器场和服务器群集、个人计算机、服务器计算机、手持式或膝上型设备、多处理器系统、基于微处理器的系统、可编程消费电

子产品、网络 PC、小型机、大型计算机、包括上述系统或设备中的任一个的分布式计算环境等。

[0022] 计算环境 (100) 不旨在对本发明的使用范围或功能提出任何限制,因为本发明可以在完全不同的通用或专用计算环境中实现。

[0023] 参考图 1, 计算环境 (100) 包括至少一个处理单元 (110) 和存储器 (120)。在图 1 中, 这一最基本的配置 (130) 被包括在虚线内。处理单元 (110) 执行计算机可执行指令, 并且可以是真实或虚拟处理器。在多处理系统中, 多个处理单元执行计算机可执行指令以提高处理能力。存储器 (120) 可以是易失性存储器 (例如, 寄存器、高速缓存、RAM)、非易失性存储器 (例如, ROM、EEPROM、闪存)、或两者的某种组合。存储器 (120) 存储实现虚拟软件应用部署配置的软件 (180)。

[0024] 尽管为了清楚起见用线条示出了图 1 的各框, 但是, 实际上, 描绘各组件并不是那样清楚, 并且用比喻方法, 图 1 以及下文讨论的其他附图的线条更精确地将是灰色的和模糊的。例如, 可以将诸如显示设备等呈现组件认为是 I/O 组件。而且, 处理器具有存储器。发明人关于此点认识到, 这是本领域的特性, 并且重申, 图 1 的图示只是例示可结合本发明的一个或多个实施例来使用的示例性计算设备。诸如“工作站”、“服务器”、“膝上型计算机”、“手持式设备”等分类之间没有区别, 它们全部都被认为是在图 1 的范围之内的并且被称为“计算机”、“计算环境”、或“计算设备”。

[0025] 计算环境 (100) 可具有附加特征。在图 1 中, 计算环境 (100) 包括存储 (140)、一个或多个输入设备 (150)、一个或多个输出设备 (160) 以及一个或多个通信连接 (170)。诸如总线、控制器或网络等互连机制 (未示出) 将计算环境 (100) 的各组件互连。通常, 操作系统软件 (未示出) 为在计算环境 (100) 中执行的其它软件提供操作环境, 并协调计算环境 (100) 的各组件的活动。

[0026] 存储 (140) 可以是可移动或不可移动的, 并可包括诸如磁盘、磁带或磁带盒、CD-ROM、CD-RW、DVD 之类的非瞬态计算机可读存储介质, 或者可用于储存信息并可在计算环境 (100) 内访问的任何其它介质。存储 (140) 存储用于软件 (180) 的指令。

[0027] 输入设备 (150) 可以是诸如键盘、鼠标、笔或跟踪球等触摸输入设备; 语音输入设备; 扫描设备; 网络适配器; CD/DVD 读取器; 或可向计算环境 (100) 提供输入的另一设备。输出设备 (160) 可以是显示器、打印机、扬声器、CD/DVD 刻录机、网络适配器、或从计算环境 (100) 提供输出的另一设备。

[0028] 通信连接 (170) 允许通过通信介质与另一计算实体进行通信。因此, 计算环境 (100) 可使用通往诸如个人计算机、服务器、路由器、网络 PC、对等设备或另一常见网络节点等一个或多个远程计算设备的逻辑连接而工作在联网环境中。通信介质以已调制数据信号的形式传达诸如数据或计算机可执行指令或请求等信息。已调制数据信号是其一个或多个特征以在信号中编码信息的方式设置或改变的信号。作为示例而非局限, 通信介质包括以电、光、RF、红外、声学或其他载波实现的有线或无线技术。

[0029] 各种工具和技术可以在计算机可读介质的一般上下文中描述。计算机可读介质可以是可在计算环境内访问的任何可用介质。作为示例而非局限, 对于计算环境 (100), 计算机可读介质包括存储器 (120)、存储 (140)、和以上的组合。

[0030] 这些工具和技术可在诸如程序模块中所包括的在目标真实或虚拟处理器上的计

算环境中执行的计算机可执行指令的一般上下文中描述。一般而言,程序模块包括执行特定任务或实现特定抽象数据类型的例程、程序、库、对象、类、组件、数据结构等。程序模块的功能可以如各实施例中所需的组合或在程序模块之间分离。用于程序模块的计算机可执行指令可以在本地或分布式计算环境中执行。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中。

[0031] 出于演示的目的,详细描述使用了如“确定”、“选择”、“调整”和“操作”等术语来描述计算环境中的计算机操作。这些以及其他类似术语是对计算机执行的操作的高层抽象,并且不应混淆于人类执行的动作,除非明确指出人类(诸如“用户”)的动作执行。对应于这些术语的实际的计算机操作取决于实现而不同。

[0032] 虚拟软件应用部署配置系统和环境

[0033] 图 2 是虚拟软件应用部署配置计算环境 (200) 的框图,可结合该计算环境实现所描述的各实施例中的一个或多个。计算环境 (200) 可包括打包环境 (210)。在打包环境 (210) 中,非虚拟应用 (212) 可被打包器 (214) 打包成虚拟应用包 (220)。例如,打包器 (214) 可以是来自诸如微软的应用虚拟化 (App-V) 软件之类的应用虚拟化软件的打包器。当打包应用 (212) 时,打包器 (214) 还可在非虚拟应用中标识要包括在部署配置 (230) 中作为配置项 (232) 的一个或多个配置。标识可以是自动进行的(即,搜索和标识可以是自动的,尽管此自动标识可由请求配置项被自动标识的用户输入请求来提示)。另外,打包器 (214) 可建议配置项 (232) 中的一个或多个是成为在部署时间必须响应于用户输入而被配置的强制配置项,还是在部署时间可响应于用户输入被设置或者可以不响应于用户输入被设置的任选配置项。对配置项 (232) 的标识以及对是否建议项目为任选还是强制的判断可被托付给打包器 (214) 的子系统。部署配置 (230) 可以是分开的文件的形式,诸如具有按位置(例如,文件或注册表位置)索引的配置项的 XML 文件。替换地,部署配置 (230) 可以是某一其他形式的,诸如多个文件和 / 或还包括其他应用数据和 / 或元数据的一个或多个文件中的属性。

[0034] 配置项 (232) 还可在用户接口设备(诸如以上参照图 1 讨论的输出设备 (160) 中的一个)上被呈现给用户。用户可提供编辑部署配置 (230) 中的配置项 (232) 中的一个或多个的用户输入,并且配置项 (232) 可响应于用户输入来编辑。例如,用户可提供请求进行以下各项的用户输入:请求从部署配置 (230) 删除项目 (232) 中的一个;请求改变项目 (232) 中的一个的名称或标签;请求将强制项 (232) 改变为任选项 (232);请求将任选项 (232) 改变为强制项 (232);请求添加、改变或删除项目 (232) 的默认配置;或者请求对项目 (232) 中的一个或多个作出某一其他改变。

[0035] 现在将讨论一些特定类型的配置项 (232)。然而,这些特定类型仅提供作为示例;作为这些类型的项目的补充或替代,其他类型的配置项 (232) 可被标识并包括在部署配置项 (230) 中。作为一个示例,配置项可以是数据库连接串。例如,打包器 (214) 可将数据库相关项的自动标识托付给打包器 (214) 的数据库子系统。数据库子系统可搜索数据库相关项。例如,数据库子系统可在非虚拟应用 (212) 被安装时监视 API(应用编程接口)调用,并且从这些调用提取特定串。数据库子系统可在随后产生配置项或搜索与处于应用状态中的那些所提取的串有关的项目,诸如在应用文件(例如,*.ini 和 xml 文件)和注册表条目中。当找到串时,数据库子系统可将这些串标识为配置项 (232)。子系统还可作出

与配置项 (232) 有关的其他判断, 诸如项目 (232) 是强制的还是任选的, 项目的名称是什么, 以及对于配置项 (232) 什么样的默认配置 (若有的话) 将被包括。打包器 (214) 可将配置项 (232) 包括在部署配置 (230) 中。此包括可由数据库子系统或者被配置成将配置项 (232) 包括在部署配置 (230) 中的某一其他更通用的组件来执行。例如, 如果在注册表条目 “HKLM\foo\app’odbcdatabase = mydb” 中找到数据库连接串 “mydb”, 则打包器 (214) 可将条目 “HKLM\foo\app, value odbcatabse = mydb” 存储在部署配置 (230) 中。类似条目可被用于诸如 xml 和 *.ini 元素之类的其他项目。例如, 对于是 xml 元素的配置项 (232), 打包器 (214) 可存储文件路径和 XPath 表达式来标识位置, 从而允许在具有相同名称的元素之间进行区分。作为另一示例, 打包器 (214) 可将诸如 “aproot\config\db.xml element ‘db’ = mydb” 之类的条目存储在部署配置 (230) 中。

[0036] 作为另一示例, 配置项 (232) 可以是 NT 服务的帐户名, 并且处理 NT 服务的打包器 214 的子系统可标识该帐户名。帐户名可按标识符列在部署配置 (230) 中, NT 服务组件在实例化应用时将识别该标识符。例如, 帐户名可通过 API 调用来设置。作为配置项的其他示例, 配置项 (232) 可包括由微软的因特网信息服务 (IIS) 管理的用户帐户和口令。打包器 (214) 的 IIS 子系统可识别由 IIS 管理的用户帐户和口令, 并且打包器 (214) 可将帐户和口令的配置项包括在部署配置 (230) 中, 按适当的标识符列出。

[0037] 可包括部署配置 (230) 的包 (220) 可被发送到部署环境 (240)。在部署环境 (240) 中, 部署配置 (230) 可被呈现给用户, 并且部署配置可被设置 (诸如通过确认默认设置、编辑设置等)。对于强制配置项 (232), 在允许包 (220) 的部署继续进行之前, 设置可能必须响应于用户输入而被确认和 / 或编辑。作为一个示例, 部署配置 (230) 可被包括在文件 (例如, xml 文件) 中, 并且该文件可被打开并作为要编辑的文档来显示。替换地, 可呈现部署配置的图形表示 (例如, 可在一个或多个对话框中呈现配置项目 (232))。可 (例如, 从进行包 (220) 的部署的管理员处) 接收用户输入, 并且响应于用户输入, 编辑组件 (242) 可如用户输入所指示地编辑部署配置 (230) 中的配置项 (232)。部署配置 (244) 可取得包并将其部署到目标环境 (260)。部署组件 (244) 和编辑组件 (242) 两者均可可是同一软件产品的部分或者是不同软件产品的部分。例如, 编辑组件 (242) 和用于在用户接口设备上呈现部署配置的组件可以是数据中心管理器软件产品的部分, 而部署组件 (244) 可以是应用虚拟化软件产品的部分。作为另一示例, 在使用基于 XML 的部署配置文件的情形中, 编辑组件 (242) 可以是诸如通用 XML 编辑器之类的简单编辑器, 其可被用来直接查看和编辑部署配置文件。

[0038] 如以上所述的, 包 (220) 可被部署到目标环境 (260) 以便被实例化。例如, 目标环境 (260) 中的实例化可由虚拟应用主机 (262) 来进行, 该虚拟应用主机可以是还包括编辑组件 (242) 和打包器 (214) 的应用虚拟软件的客户机组件。具体地, 应用可被实例化为目标环境 (260) 中的沙箱 (266) 里的虚拟应用实例 (264)。部署配置 (230) 中的配置项 (232) 可被应用于虚拟应用实例 (264) (即, 实例化的应用) 中。例如, 整个应用可被部署, 并在随后被配置为虚拟应用实例 (264)。作为替换, 包 (220) 的部署以及虚拟应用实例 (264) 的配置可交错进行, 其中包 (220) 的一部分被部署和配置, 接下来是包 (220) 的另一部分被部署和配置等。这种交错的部署和配置可继续进行, 直至整个应用被实例化为虚拟应用实例 (264)。

[0039] 应用部署配置 (230) 可以各种方式来进行, 诸如重写元数据和 / 或作出 API 调用

来配置应用实例 (264)。此配置可在部署环境 (240) 和 / 或目标环境 (260) 中进行。作为另一示例,虚拟应用可在存储中保留在未配置状态下。部署配置信息可在运行时与应用信息合并。例如,当应用打开 *.ini 文件时,可从存储读取 *.ini 文件并将其与部署配置项合并。合并的结果可被给予应用以供消费。作为另一示例,API 调用可被钩住,并且在运行时,API 调用的行为可被改变以返回来自部署配置 (230) 的值。例如,注册表 API 可被钩住以检测虚拟应用实例 (264) 的虚拟过程何时在读取已被部署配置项 (232) 覆盖的注册表值。当这种情况发生时,可返回来自部署配置项 (232) 的值,而非注册表中的值。

[0040] 对部署配置 (230) 的标识、编辑和 / 或应用可在与以上所讨论的计算环境不同的环境中进行。例如,可将虚拟应用包 (220) 从打包环境 (210) 直接部署到目标环境 (260)。实际上,对虚拟应用的打包、部署、实例化和执行可全都在同一环境中进行。

[0041] 虚拟软件应用部署配置技术

[0042] 现在将讨论若干虚拟软件应用部署配置技术。可以在计算环境中执行这些技术中的每一个。例如,每种技术可在包括至少一个处理器和存储器的计算机系统中执行,该存储器包括存储于其上的、在由该至少一个处理器执行时使该至少一个处理器执行该技术的指令 (存储器存储指令 (例如,对象代码),并且当处理器执行这些指令时,处理器执行该技术)。类似地,方法可具有收录于其上的计算机可执行指令,该些指令在由至少一个处理器执行时使该至少一个处理器执行该技术。

[0043] 参看图 3,将讨论虚拟软件应用部署配置技术。该技术可包括自动发现软件应用的一个或多个配置项目 (310)。可通过用户接口设备来呈现配置项 (312)。可接收用户输入 (314),并且响应于接收用户输入 (314),可编辑配置项中的一个 (316)。例如,此对配置的改变或编辑 (316) 可包括删除配置项、修改与配置项相关联的名称或标签、或者删除、添加、或修改配置项的设置。可打包应用以形成虚拟应用包 (320)。

[0044] 在打包应用 (320) 之后,可再次通过用户接口设备来呈现配置项 (322)。而且在打包 (320) 之后,可接收用户输入 (324),并且可编辑一个或多个配置项设置 (326)。还可在此时编辑配置项的其他特征。因此,对配置项的呈现、对用户输入的接收、以及响应性编辑可在打包 (320) 完成之前 (例如,在打包时间) 和 / 或在打包 (320) 完成之后 (例如,在部署时间) 进行。另外,可部署包以实例化应用 (330)。可在经实例化的应用中应用可包括配置项的一个或多个设置的部署配置 (340)。如果确定要再次部署包 (350),则可再次执行呈现配置项 (322)、接收用户输入 (324)、编辑配置项设置 (326)、部署包 (330)、以及应用部署配置 (340)。

[0045] 该技术也可包括在经实例化的应用中应用第二部署配置,以改变配置项的设置。可在已用原始部署配置配置了经实例化的应用之后应用第二部署配置。此另一部署配置可与第一部署配置不同,因此可通过应用该第二部署配置来重新配置经实例化的应用。此外,技术还可包括响应于标识配置项的用户输入发现软件应用的一个或多个配置项,并且部署配置还可包括这些手动发现的配置项的一个或多个设置。

[0046] 因此,同一个包可被部署来多次实例化应用,不同的部署配置设置可被应用到应用的不同实例中。例如,可部署应用以便在多个目标环境中实例化,这多个目标环境具有用于被不同地配置的目标环境的不同的部署配置设置。因此,部署虚拟应用包以实例化应用 (330) 可包括部署虚拟应用包以将应用实例化为应用的第一实例,而应用部署配置 (340)

可包括在应用的第一实例中应用部署配置。技术还可包括部署虚拟应用包以将应用实例化为应用的第二实例 (330), 以及在应用的第二实例中应用部署配置 (340)。而且, 在应用的第一实例中应用部署配置 (340) 以及在应用的第二实例中应用部署配置 (340) 可包括在应用的第一和第二实例中应用配置项中的至少一个的不同设置。

[0047] 现在参看图 4, 将讨论另一虚拟软件应用部署配置技术。该技术可包括发现部署配置的一个或多个配置项 (410), 以及打包软件应用 (420) 以形成虚拟应用包。发现 (410) 可在完成打包 (420) 之前执行。例如, 发现 (410) 可在打包 (420) 期间和 / 或在打包 (420) 之前执行。然而, 发现 (410) 可在打包 (420) 完成之后执行, 诸如通过使用扫描包来标识和 / 或产生部署配置文档的工具。技术还可包括在打包 (420) 之后设置虚拟应用包的部署配置 (430)。设置部署配置 (430) 可包括编辑部署配置中的一个或多个配置项 (例如, 通过编辑配置项的设置) 和 / 或确认部署配置中的现有配置项。例如, 设置部署配置 (430) 可以响应于接收到用户输入来进行。配置项中的至少一个可被标记为任选或强制的。设置部署配置 (430) 可包括对每个配置项判断该项目是否被标记为强制的, 并且如果是, 则在部署虚拟应用包之前要求针对此项目的用户输入。例如, 此类用户输入可确认现有默认设置, 修改现有默认设置, 或输入其中没有默认设置的设置。另外, 可部署虚拟应用包以实例化应用 (440), 并且可在经实例化的应用中应用部署配置 (450)。

[0048] 包可被部署多次, 有可能对不同的部署具有不同的部署配置。因此, 部署配置可以是第一部署配置。部署虚拟应用包以实例化应用 (440) 可包括部署虚拟应用包以在应用的第一实例中实例化应用。此外, 在经实例化的应用中应用部署配置 (450) 可包括在应用的第一实例中应用第一部署配置。此外, 该技术还可包括: 在打包软件应用之后设置虚拟应用包的第二部署配置; 部署虚拟应用包以在应用的第二实例中实例化应用; 以及在应用的第二实例中应用第二部署配置。应用的第一实例可以是在具有第一机器配置的第一机器上, 而应用的第二实例可以是在具有第二机器配置的第二机器上。替换地, 应用的第一和第二实例可以在同一机器上, 或者在都具有相同配置的不同的机器上。

[0049] 现在参看图 5, 将讨论又一虚拟软件应用部署配置技术。该技术可包括发现软件应用的一个或多个配置项 (510)。发现 (510) 中的一些或全部可自动进行。可在用户接口设备上呈现配置项 (520), 并且可改变配置项中的至少一个 (530)。可响应于接收指示改变的用户输入进行改变 (530)。

[0050] 可打包应用 (540) 以形成虚拟应用包, 但是呈现 (520) 和改变 (530) 可在打包 (540) 完成之前进行。另外, 在打包应用 (540) 之后, 可通过用户接口设备来呈现配置项 (550)。响应于指示对配置项中的所选配置项的改变的用户输入, 所选配置项可被改变 (560)。可部署虚拟应用包以实例化应用 (570), 并且可在经实例化的应用中应用配置项 (580)。

[0051] 尽管用结构特征和 / 或方法动作专用的语言描述了本主题, 但可以理解, 所附权利要求书中定义的主题不必限于上述具体特征或动作。更确切而言, 上述具体特征和动作是作为实现权利要求的示例形式公开的。

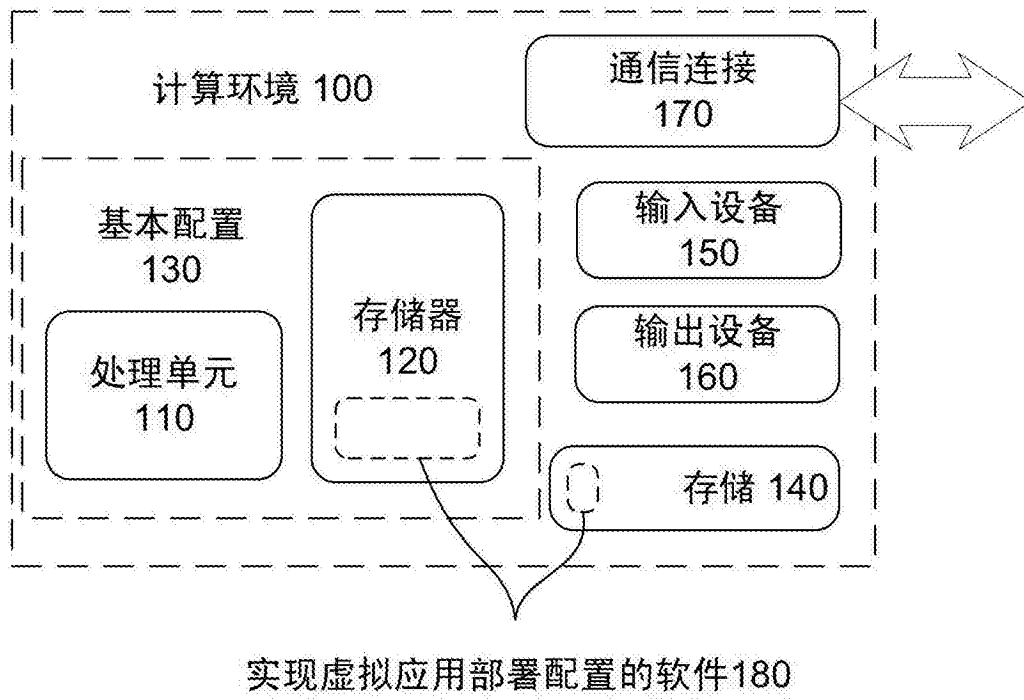


图 1

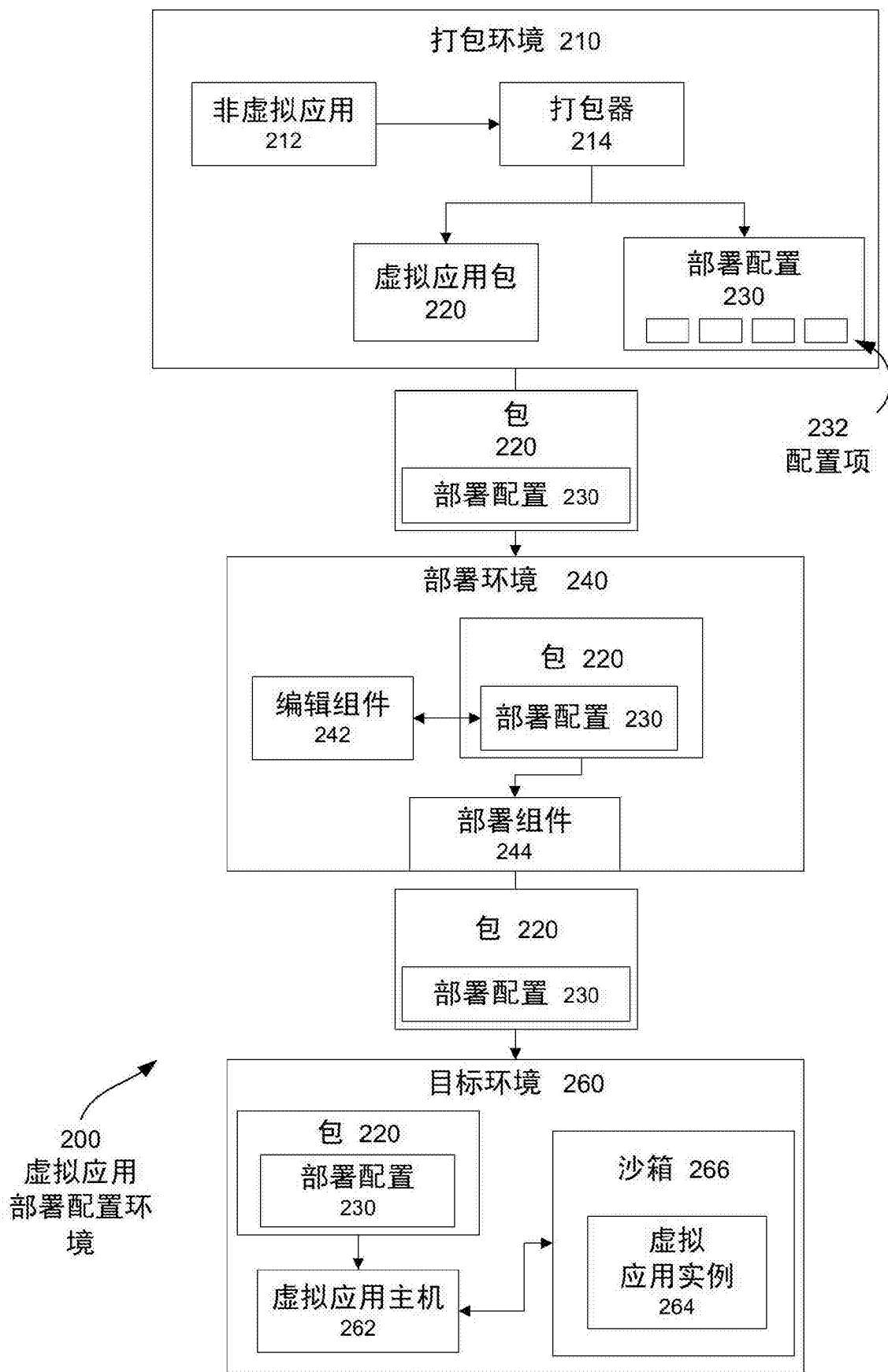


图 2

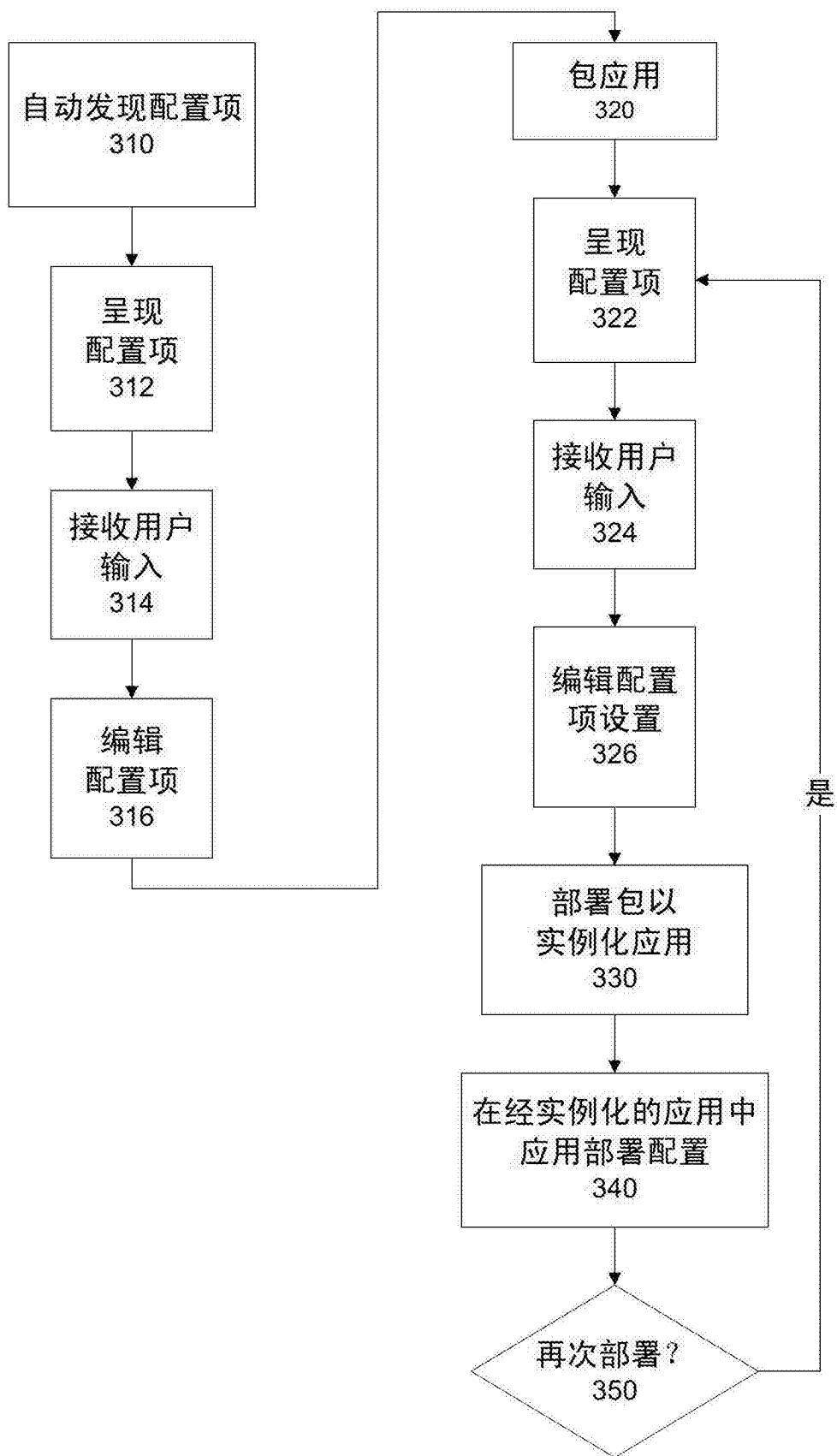


图 3

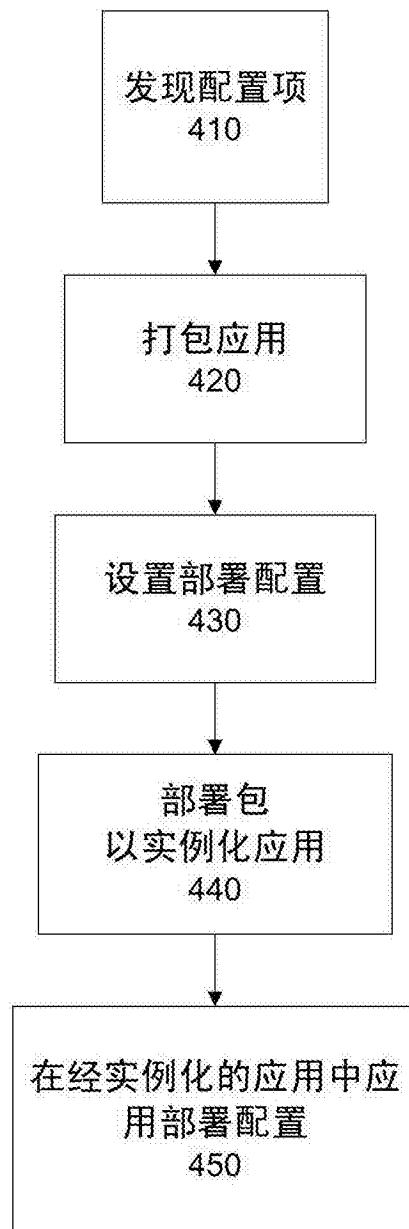


图 4

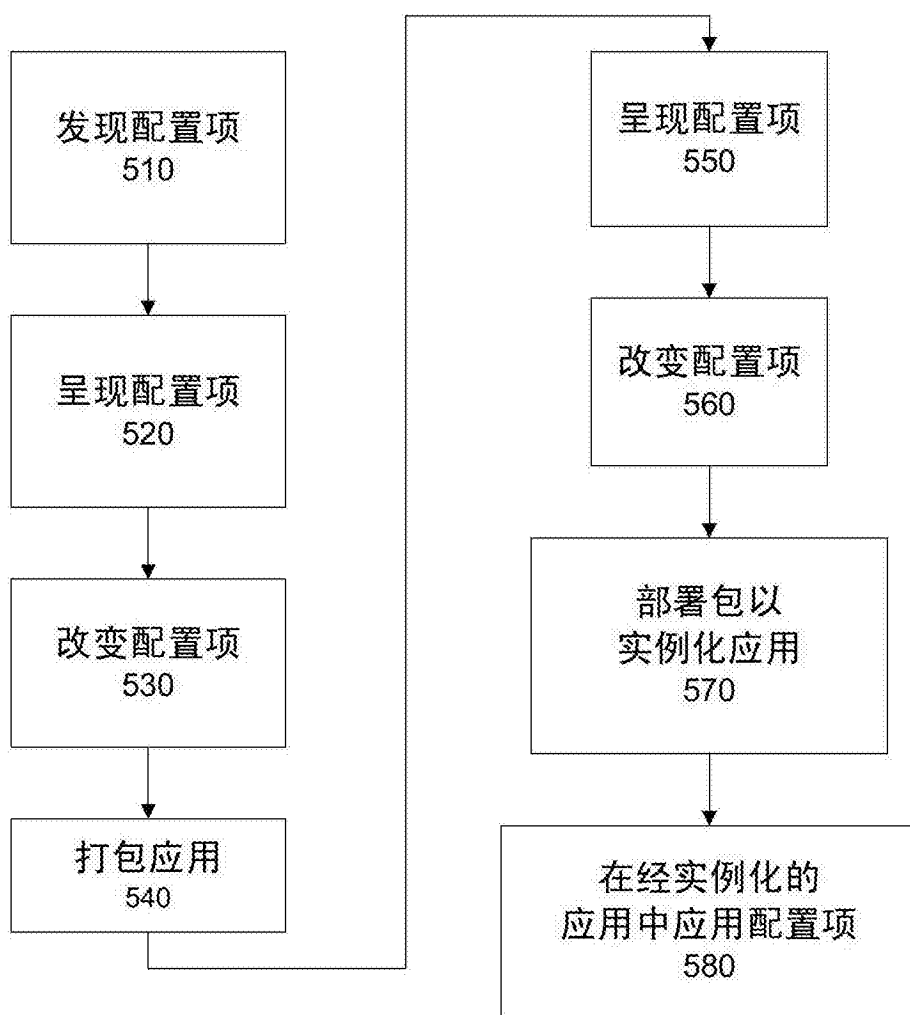


图 5