



(12)发明专利

(10)授权公告号 CN 103765342 B

(45)授权公告日 2017.07.14

(21)申请号 201180072785.8

(22)申请日 2011.10.07

(65)同一申请的已公布的文献号

申请公布号 CN 103765342 A

(43)申请公布日 2014.04.30

(30)优先权数据

13/207138 2011.08.10 US

(85)PCT国际申请进入国家阶段日

2014.02.10

(86)PCT国际申请的申请数据

PCT/US2011/055464 2011.10.07

(87)PCT国际申请的公布数据

W02013/022463 EN 2013.02.14

(73)专利权人 微软技术许可有限责任公司

地址 美国华盛顿州

(72)发明人 J. 贝里

(74)专利代理机构 中国专利代理(香港)有限公司 72001

代理人 刘鹏 汪扬

(51)Int.Cl.

G06F 1/32(2006.01)

G06F 9/44(2006.01)

(56)对比文件

CN 101470509 A, 2009.07.01,

GB 2475902 A, 2011.06.08,

US 2005066006 A1, 2005.03.24,

审查员 周飞

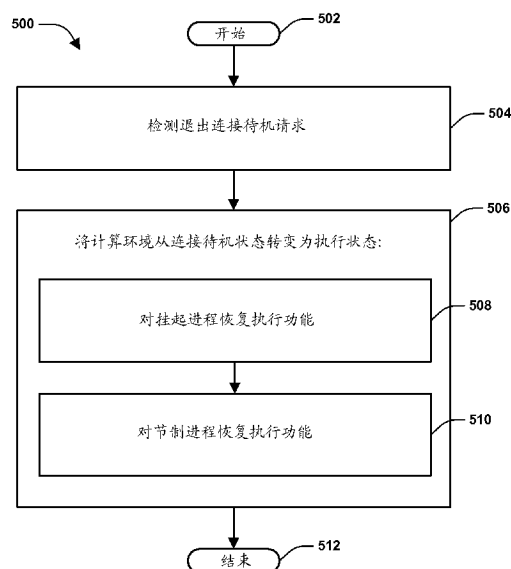
权利要求书2页 说明书13页 附图9页

(54)发明名称

用于连接待机进程的挂起和/或节制

(57)摘要

提供了用于将功率管理分类分配给进程、基于分配给进程的功率管理分类将计算环境转变为连接待机状态、以及将计算环境从连接待机状态转变为执行状态的一种或多种技术和/或系统。即,可以基于各种因素将诸如豁免、节制和/或挂起之类的功率管理分类分配给进程,比如,进程是否提供所希望的功能和/或进程是否提供计算环境的基本操作所依赖的功能。按照这种方式,可以将计算环境转变为低功率连接待机状态,该状态可以继续执行所希望的功能,同时通过挂起和/或节制其他功能而降低功率消耗。因为一些功能可以仍然执行,所以计算环境可以向用户快速提供最新信息的快速响应方式转变为执行状态。



1. 一种从连接待机状态转变为执行状态的方法,包括:
检测与计算环境相关联的退出连接待机请求;以及
将计算环境从连接待机状态转变为执行状态,包括:
对被分配了挂起分类的挂起进程恢复执行功能;
对被分配了节制分类的节制进程恢复执行功能,在对挂起进程恢复执行功能之前,实施对节制进程恢复执行功能,挂起进程被配置为使用来自节制进程的资源;以及
继续进行被分配了豁免分类的硬件功能进程的执行功能,所述硬件功能进程被允许在连接待机状态期间执行。
2. 如权利要求1所述的方法,计算环境包括操作系统。
3. 如权利要求2所述的方法,操作系统包括以下项中的至少一项:平板设备操作系统、桌面型设备操作系统或移动设备操作系统。
4. 如权利要求1所述的方法,包括:
在对子挂起进程恢复执行功能之前,对父挂起进程恢复执行功能。
5. 如权利要求4所述的方法,子挂起进程被配置为使用来自父挂起进程的资源。
6. 如权利要求1所述的方法,挂起分类对应于:
进程是否能够在没有计算系统故障的情况下被挂起并且与该进程相关联的有限运行时间功能是否不是所希望的。
7. 如权利要求1所述的方法,节制分类对应于:
进程是否能够在没有计算系统故障的情况下被节制并且与该进程相关联的有限运行时间功能是否是所希望的。
8. 如权利要求1所述的方法,节制进程包括在系统会话内创建的提供所希望状态功能的服务。
9. 如权利要求1所述的方法,挂起进程包括在用户会话内创建的用户应用。
10. 一种从连接待机状态转变为执行状态的系统,包括:
转变组件,其被配置为:
检测与计算环境相关联的退出连接待机请求;以及
将计算环境从连接待机状态转变为执行状态,包括:
对被分配了挂起分类的挂起进程恢复执行功能;以及
对被分配了节制分类的节制进程恢复执行功能,在对挂起进程恢复执行功能之前,实施对节制进程恢复执行功能,挂起进程被配置为使用来自节制进程的资源;以及
继续进行被分配了豁免分类的硬件功能进程的执行功能,所述硬件功能进程被允许在连接待机状态期间执行。
11. 如权利要求10所述的系统,计算环境包括操作系统。
12. 如权利要求11所述的系统,操作系统包括以下项中的至少一项:平板设备操作系统、桌面型设备操作系统或移动设备操作系统。
13. 如权利要求10所述的系统,转变组件被配置为:
在对子挂起进程恢复执行功能之前,对父挂起进程恢复执行功能,子挂起进程被配置为使用来自父挂起进程的资源。
14. 如权利要求10所述的系统,转变组件被配置为:

基于应用于节制进程的节制时间表,对节制进程恢复执行功能。

15. 如权利要求14所述的系统,转变组件被配置为:

不应用被应用于节制进程的节制时间表。

用于连接待机的进程的挂起和/或节制

背景技术

[0001] 围绕节省功率消耗和/或延长电池寿命而开发许多计算设备,比如桌面型设备、膝上型设备、智能电话、以及平板设备。例如,如果膝上型设备已经空闲了设定时间段,则可以调暗膝上型设备屏幕。节电可以在为计算设备(再)充电之间提供延长的操作和/或时间,这可能对于用户而言是所希望且方便的。不幸的是,许多应用可能不包括节电功能,并且可能在计算设备可以以另外的方式空闲(例如,空闲应用可以不必要地消耗处理器周期)时,通过不使硬件退出高功率状态和/或通过利用资源而显著地减弱节电和/或具有拙劣的功率“保健”。

[0002] 许多节电技术可能劣化与计算设备的用户体验,这是因为将计算设备从诸如休眠/睡眠状态之类的低功率状态转变为执行状态可能需要用户输入、花费相对长的时间量、和/或提供过时数据直到被刷新为止。例如,用户可能在家时周期性地与膝上型设备交互。为了节省膝上型设备的电池寿命,休眠/睡眠策略可能是适当的,其中,如果用户在设定时间段内,比如10分钟内不与膝上型设备交互,则该膝上型设备可以被置于挂起状态。不幸的是,常规休眠/睡眠策略可能花费相当数量的时间(例如,数分钟)来将计算设备再次唤醒到用户可使用的最新执行状态,这是因为计算系统的实质性部分可能已经被挂起(例如,处理器可能停止执行,操作系统内核可能被挂起,等等)。等待计算系统被完全再次唤醒可能提供不太满意的用户体验。

发明内容

[0003] 本发明内容被提供来以简要形式介绍概念的选择,以下将在具体实施方式中对这些概念进行进一步的描述。本发明内容既不是旨在标识所要求保护主题的关键因素或必要特征,也不是旨在被用来限制所要求保护主题的范围。

[0004] 除了别的以外,本文公开了用于将功率管理分类分配给与计算环境相关联的一个或多个进程、基于分配给进程的功率管理分类将计算环境转变为连接待机状态、以及将计算环境从连接待机状态转变为执行状态的一种或多种系统和/或技术。

[0005] 可以意识到,在一个示例中,计算环境的执行状态可以对应于计算环境的一般操作状态。在处于执行状态时,可以创建并执行进程,并且计算环境可以对用户输入作出响应(例如,在计算环境处于执行状态时,用户可以与字处理器和/或其他应用交互)。可以意识到,在一个示例中,连接待机状态可以对应于计算环境的低功率消耗状态,然而,其可以提供始终开着/始终连接的体验。在处于连接待机状态时,特定进程可以被允许执行,而其他进程可以被挂起而进入非执行挂起状态,和/或被节制在执行状态与非执行挂起状态之间,以降低功率消耗。例如,当处于连接待机状态时,计算环境可以允许执行和/或节制执行关键系统进程(例如,联网功能进程、特定核心内核进程、硬件功能进程、消息发送功能进程等)、机智功率应用(例如,具有被管理的生命周期的应用)、和/或可以提供所希望的连接的其他进程(例如,简单电子邮件功能进程),使得计算环境可以在从连接待机状态转变为执行状态时为用户提供快速响应且最新的体验(例如,计算环境可以被快速唤醒(例如,在数

秒内)并且可以向用户提供最新电子邮件、信息和/或连接)。然而,为了节电,可以节制和/或挂起非关键系统服务(例如,打印假脱机程序、系统更新检查器等)和/或用户应用(例如,字处理器)。

[0006] 在将功率管理分类分配给进程的一个示例中,可以识别进程(例如,新创建的进程)。可以基于各种因素(例如,该进程对于基本功能的重要性如何?该进程与系统服务相关联吗?该进程与用户应用相关联吗?该进程能够被节制而没有导致故障吗?该进程能够被挂起而没有失去所希望的功能和/或导致故障吗?等等)给该进程分配功率管理分类,比如豁免分类、挂起分类或节制分类。如果进程的生命周期由该进程和/或生命周期管理组件管理,则该进程可以被分配豁免分类,这是因为该进程可能已经与节电功能相关联(例如,分配功率管理分类给进程并实施可能是多余的和/或可能导致该进程的故障)。如果该进程可以被挂起而没有计算系统故障(例如,相片观看进程的挂起可以不导致计算环境的故障和/或相片观看应用的故障)并且与该进程相关联的有限运行时间功能不是所希望的(例如,在计算环境处于连接待机状态时,相片观看功能可以不是所希望的),则可以将挂起分类分配给该进程。

[0007] 如果该进程可以被节制而没有计算系统故障并且与该进程相关联的有限运行时间功能是所希望的(例如,在计算环境处于连接待机状态时,用户可能希望接收消息通知),则可以将节制分类分配给该进程。如果该进程不能在计算系统故障的情况下被节制并且与该进程相关联的有限运行时间功能是所希望的,则可以分配豁免分类给该进程。按照这种方式,可以将功率管理分类分配给该进程。在一个示例中,在将计算环境转变为连接待机状态时,可以利用功率管理分类。例如,豁免进程可以被允许执行,可挂起进程可以被挂起,并且可节制进程可以被节制。

[0008] 在基于分配给进程的功率管理分类将计算环境转变为连接待机状态的一个示例中,可以检测与计算环境相关联的进入连接待机请求(例如,用户可以调用平板设备的待机按钮,不活动的阈值时间段可能已经超过,等等)。例如,可以将计算环境从执行状态转变为连接待机状态。例如,对于相应的可挂起进程,可以将可挂起进程置于非执行挂起状态(例如,可挂起进程可能不适合于执行,因此可能不能调用硬件组件来实施功率消耗活动)。对于相应的可节制进程,可以将节制时间表分配给和/或应用于可节制进程(例如,电子邮件进程可以被允许每8分钟执行5秒以检查新消息,否则,该电子邮件进程可以处于非执行挂起状态,以降低功率消耗)。对于相应的豁免进程,豁免进程可以被允许在执行状态中执行(例如,网络进程可以被允许执行以提供所希望的功能,比如接收新电子邮件)。在一个示例中,可挂起进程可以在可节制进程之前被挂起,这是因为可挂起进程可能依赖于来自可节制进程的功能,因此如果可节制进程在可挂起进程被挂起之前进入非执行挂起状态,则该可挂起进程可能失败(例如,如果潜在的网络相片上载进程不可用,则可挂起在线相片应用可能失败)。按照这种方式,计算环境可以被转变为连接待机状态以便节电,同时仍然提供所希望的功能(例如,通知新消息)。

[0009] 在将计算环境从连接待机状态转变为执行状态的一个示例中,可以检测与计算环境相关联的退出连接待机请求(例如,用户可以尝试在平板屏幕上用手指触摸动作来与平板设备交互)。可以基于该退出连接待机请求将计算环境从连接待机状态转变为执行状态。例如,对于相应的挂起进程,可以对挂起进程恢复执行功能(例如,在线相片应用进程可以

以快速响应的方式可用,比如在数秒内)。对于相应的节制进程,可以对节制进程恢复执行功能(例如,可以以快速响应的方式将处于非执行挂起状态中的基于节制时间表的电子邮件进程转变为执行状态)。按照这种方式,可以以快速响应的方式将计算环境转变为最新执行状态,而不用用户为再用的功能和/或最新信息等待相当数量的时间。

[0010] 为了完成前述和相关方面,以下描述和附图阐述了某些说明性的方面和实现方式。这些说明了可以据以采用一个或多个方面的各种方式,但是是这些方式中的一些。当考虑结合附图时,从以下详细描述,本公开的其他方面、优点和新颖特征将变得显而易见。

附图说明

[0011] 图1是图示出将功率管理分类分配给进程的示例性方法的流程图。

[0012] 图2是图示出将功率管理分类分配给进程的示例性系统的组件框图。

[0013] 图3是图示出基于分配给进程的功率管理分类将计算环境转变为连接待机状态的示例性方法的流程图。

[0014] 图4是图示出基于分配给进程的功率管理分类将计算环境转变为连接待机状态的示例性系统的组件框图。

[0015] 图5是图示出将计算环境从连接待机状态转变为执行状态的示例性方法的流程图。

[0016] 图6是图示出将计算环境从连接待机状态转变为执行状态的示例性系统的组件框图。

[0017] 图7是从连接待机状态转变为执行状态的平板设备的示例的图示。

[0018] 图8是示例性计算机可读介质的图示,其中,可以包括被配置为具体实现本文所阐述的规定的一个或多个的处理器可执行指令。

[0019] 图9图示出示例性计算环境,其中,可以实施本文所阐述的规定的一个或多个。

具体实施方式

[0020] 现在参照附图描述所要求保护的主体,其中,贯穿全文,同样的附图标记一般用于指代同样的元素。在以下描述中,为了说明的目的,阐述了许多具体细节,以提供对所要求保护主题的透彻理解。然而,可能明显的是,可以在没有这些具体细节的情况下实践所要求保护的主体。在其他实例中,以框图形式图示了结构和设备,以便于描述所要求保护的主体。

[0021] 许多计算环境可以结合节电技术,以减轻功率消耗和/或节省电池寿命。例如,操作系统可以提供一种休眠/睡眠策略,该策略可以阻止否则可能调用计算资源来执行功率消耗任务的应用、系统进程和/或处理器功能的执行。不幸的是,许多节电技术可能挂起执行所希望功能的进程(例如,休眠策略可能不包括用于选择性地允许某些进程在休眠状态期间执行的功能),因此计算环境可能在唤醒到执行状态时提供过时信息(例如,从休眠状态唤醒的计算环境可能不提供最新消息信息,这是因为消息功能进程可能在休眠状态期间已经被挂起)。附加地,在计算环境完全响应和/或能够提供最新信息之前,从休眠/睡眠状态唤醒可能花费相当大量的时间。因此,在提供足够的节电与在向用户提供快速响应和连接体验之间进行平衡,是所希望的。

[0022] 相应地,本文提供了用于将功率管理分类分配给与计算环境相关联的进程、基于分配给进程的功率管理分类将计算环境转变为连接待机状态、以及将计算环境从连接待机状态转变为执行状态的一种或多种系统和/或技术。特别地,可以基于各种因素,比如进程是否提供所希望的功能和/或关键系统功能,用豁免分类、挂起分类和/或节制分类对进程进行分类。按照这种方式,可以将计算环境转变为提供始终开着/始终连接的体验的连接待机状态,比如低功率消耗状态。特别地,豁免进程可以被允许执行,可挂起进程可以被从执行挂起,以及可节制进程可以基于节制时间表而在执行状态和非执行挂起状态之间循环。对进程进行挂起和/或节制可以降低功率消耗,这是因为这样的进程可以被置于非执行挂起状态,该非执行挂起状态可以限制这样的进程调用资源来实施功率消耗任务。可以以快速响应的方式将计算环境从连接待机状态转变为执行状态,该快速响应方式用极少量的时间向用户提供了最新体验(例如,用户可能能够在数秒内与计算环境的最新状态进行交互)。特别地,挂起进程和/或节制进程可以被置于执行状态,使得计算环境被转变为执行状态。

[0023] 图1中的示例性方法100图示出了将功率管理分类分配给进程的一个实施例。在102处,所述方法开始。在104处,可以识别要被分配功率管理分类的进程(例如,诸如平板设备操作系统、桌面型设备操作系统、移动设备操作系统等计算环境内的进程)。例如,新创建的进程可以被识别为所述进程(例如,诸如等级工作对象之类的逻辑容器内的进程)。在106处,可以将功率管理分类分配给该进程。可以意识到,各种因素可以被评价,以分配功率管理分类(例如,进程相对系统操作的关键程度、由进程提供的所希望的功能、对进程进行节制的能力、等等)。

[0024] 在分配功率管理分类的一个示例中,进程的逻辑容器可以被确定为与父进程的父逻辑容器相关联(例如,拼写检查器进程的逻辑容器可以关联于(例如,嵌入到)父字处理进程的父逻辑容器)。分配给父进程的父功率管理分类可以分配给该进程,作为分配给该进程的功率管理分类(例如,分配给父字处理进程的挂起分类也可以分配给拼写检查器进程,使得在连接待机状态期间两个进程都可以一起被挂起,与此相对照的是,一个进程被挂起,而另一个进程(例如拼写检查器)继续不必要地执行)。

[0025] 在分配功率管理分类的另一示例中,在108处,如果进程的生命周期由该进程和/或生命周期管理组件管理,则豁免分类可以分配给该进程。例如,应用(例如,相对新地创建的应用)可以在提供广泛功率消耗管理的应用模型下开发。因此,可以分配豁免分类给该应用,这是因为附加的功率消耗管理可能是多余的和/或妨碍该应用的执行。

[0026] 在分配功率管理分类的另一示例中,在110处,如果该进程可以在没有计算系统故障(例如,计算环境和/或与该进程相关联的应用的故障)的情况下被挂起并且与该进程相关联的有限运行时间功能不是所希望的(例如,在计算环境处于低功率连接待机状态时,用户可能不想要打印功能),则可以分配挂起分类给该进程。例如,如果该进程与用户应用相关联,则该进程可以被确定为是可挂起的,并且可以分配挂起分类给该进程(例如,在用户会话内初始化的相片分享应用可能不提供所希望的功能用于在计算环境处于低功率连接待机状态时(例如,用户在他/她的平板设备处于连接待机状态时并不希望观看/分享相片)和/或可能不影响计算环境的核心系统功能)。

[0027] 在分配功率管理分类的另一示例中,在112处,如果该进程可以在没有计算系统故

障的情况下被节制并且与该进程相关联的有限运行时间功能是所希望的,则可以分配节制分类给该进程。例如,如果该进程包括提供所希望状态功能的服务(例如,在系统会话内初始化的消息发送系统服务)(例如,该消息系统服务可以监听新消息,使得可以向用户通知最新通信),则该进程可以被确定为是可节制的,并且可以分配节制分类给该进程(例如,可以分配节制分类给与防病毒功能、消息发送服务功能、网络服务功能、和/或多种其他所希望功能相关联的进程)。

[0028] 在分配功率管理分类的另一示例中,在114处,如果不能在没有计算系统故障的情况下挂起或节制该进程,则可以将豁免分类分配给该进程。可以意识到,计算环境可能依赖于用于基本操作功能的多种进程,因此这样的进程的挂起和/或节制可能导致计算系统故障。在一个示例中,可以将豁免分类分配给特定核心内核进程、设备驱动器和/或提供基本操作功能的其他进程。在另一示例中,可以查阅依赖性图。可能已经从核心内核进程到提供该核心内核进程所依赖的功能的一个或多个支持进程,构建了该依赖性图(例如,促进分布式软件组件之间的通信的软件组件网络通信模块可以是与这样的软件组件相关联的一个或多个核心内核进程所依赖的)。如果该进程包括在该依赖性图内指定的支持进程,则该进程可以被确定为不能够被挂起,因此可以将豁免分类分配给该进程。

[0029] 在分配功率管理分类的另一示例中,如果该进程提供与用户定义的策略相关联的功能,则可以将用户定义的功率管理分类分配给该进程。例如,用户可以指定,希望计算环境保持与到来消息同步(例如,不论计算环境是处于低功率连接待机状态还是执行状态)。相应地,可以分配用户定义的功率管理分类,比如豁免分类给提供消息功能的进程(例如,消息监听者进程、消息通知进程等)。按照这种方式,可以将可以用于将计算环境转变为连接待机状态的功率管理分类分配给各种进程。在116处,所述方法结束。

[0030] 图2图示出被配置为将功率管理分类分配给进程的系统200的示例。系统200可以包括分类组件204。分类组件204可以被配置为将功率管理分类分配给与计算环境相关联的进程。例如,计算环境可以与计算系统进程202相关联,计算系统进程202包括网络分组处理器208、硬件事件处理器210、打印假脱机程序212、防病毒214、字处理器216、拼写检查器218(例如,拼写检查器218可以是字处理器216的子进程)、即时消息器220和/或消息通知器222(例如,消息通知器222可以是即时消息器222的子进程)。

[0031] 在一个示例中,分类组件204可以被配置为确定进程是否与父进程相关联。如果该进程与父进程相关联,则分类组件204可以将分配给该父进程的父功率管理分类进行分配,作为分配给该进程的功率管理分类。例如,字处理器216可以是拼写检查器218的父进程。字处理器216可能已经被分配了挂起分类(例如,字处理器216可以与字处理器应用相关联,该字处理器应用在低功率连接待机状态期间不提供所希望的功能和/或不提供关键系统功能)。相应地,可以将分配给字处理器216的挂起分类分配给拼写检查器218。类似地,可以将分配给即时消息器220的用户功率管理豁免分类分配给消息通知器222,这是因为即时消息器220是消息通知器222的父进程。按照这种方式,可以类似地对关联进程进行分类,使得这样的功率管理分类可以被实现为效率组(例如,较少的分类操作需要被实施,进一步推动了节电)和/或实现成避免不一致的结果(例如,如果消息通知器222被分配了豁免分类,则在连接待机状态期间,消息通知器222可以不必要地执行,而即时消息器220被挂起)。

[0032] 在另一示例中,分类组件204可以基于确定该进程的生命周期由该进程和/或生命

周期管理组件来管理,而将豁免分类分配给该进程。即,节电可能已经由该进程或另一组件来管理。

[0033] 在另一示例中,分类组件204可以基于确定进程可以在没有计算系统故障的情况下被挂起和/或与该进程相关联的有限运行时间功能不是所希望的,来将挂起分类分配给该进程。例如,可以将挂起分类分配给字处理器216,这是因为字处理器216是一种用户应用,在计算环境处于连接待机状态时,该用户应用的挂起可以不创建计算系统故障和/或用户可能不想要有限运行时间字处理功能。附加地,可以将挂起分类分配给打印机假脱机程序212,这是因为在计算环境处于连接待机状态时,打印假脱机程序212的挂起可以不创建计算系统故障和/或用户可能不想要有限运行时间打印功能。

[0034] 在另一示例中,分类组件204可以基于确定进程可以被节制并且与该进程相关联的有限运行时间功能是所希望的,来将节制分类分配给该进程。例如,防病毒214(例如,在系统会话内初始化的防病毒系统服务)可以提供所希望状态的功能,这是因为例如对存储装置的写访问可以被阻止,直到防病毒214对要写入的数据实施检查为止(例如,用户可能希望在计算环境处于连接待机状态时将到来消息保存到存储装置)。

[0035] 在另一示例中,分类组件204可以基于确定该进程不能在计算系统故障的情况下被挂起或被节制,来将豁免分类分配给该进程。在一个示例中,分类组件204可以确定特定核心内核进程、设备驱动器、和/或其他进程不能在计算系统故障的情况下被挂起。例如,网络分组处理器208和/或硬件事件处理器210可以提供计算环境所依赖的核心功能,因此可以被分配豁免分类。在另一示例中,依赖性图可以被查阅,以确定该进程是否是核心内核进程所依赖的支持进程。

[0036] 在另一示例中,分类组件204可以基于确定该进程提供与用户定义的策略相关联的功能,来将用户定义的功率管理分类分配给该进程。例如,可以将用户功率管理豁免分类分配给即时消息器,这是因为即使计算环境处于连接待机状态,用户也可能希望接收即时到来消息。

[0037] 按照这种方式,可以在将功率管理分类分配给进程时创建和/或更新功率管理分类分配206。可以在将计算环境转变到连接待机状态以实现可以降低功率消耗和/或提高电池寿命的低功率状态中,利用这样的分配。可以意识到,这样的分配可以用于多种其他场景,比如在处于执行状态时前台/后台任务的挂起和/或进程的节制。

[0038] 图3中的示例性方法300图示出基于分配给进程的功率管理分类将计算环境转变为连接待机状态的一个实施例。在302处,所述方法开始。在304处,可以检测与计算环境相关联的进入连接待机请求。可以意识到,连接待机状态可以对应于低功率消耗状态,其中,各种进程可以被挂起、被节制、或被允许继续执行。按照这种方式,功率消耗可以被降低,并且计算环境可以继续执行所希望的功能,以在转变(回)到执行状态时提供快速响应且最新的体验。可以在诸如平板计算设备、桌面型计算设备、膝上型设备、移动电话等各种计算设备上以多种方式实施连接待机状态。

[0039] 在306处,可以将计算环境转变为连接待机状态。在一个示例中,在308处,对于被分配了挂起分类的相应可挂起进程,可挂起进程可以被置于非执行挂起状态。即,可以阻止可挂起进程调用资源来实施功率消耗任务(例如,字处理器可以被挂起,使得字处理器不访问硬件资源,该硬件资源否则可以处于低功率空闲状态)。挂起分类可以对应于进程是否能

够被挂起而没有计算系统故障和/或与该进程相关联的有限运行时间功能是否不是所希望的。按照这种方式,可挂起进程可以被置于非执行挂起状态,而没有创建计算机系统故障和/或失去所希望的功能。

[0040] 在另一个示例中,在310处,对于被分配了节制分类的相应可节制进程,可以分配节制时间表给可节制进程。节制分类可以对应于进程是否可以被节制而没有计算系统故障和/或与该进程相关联的有限运行时间功能是否是所希望的。节制时间表可以在处于连接待机状态时应用于可节制进程。例如,可以基于节制时间表而使可节制进程在执行状态和非执行挂起状态之间循环(例如,节制时间表可以指定用于执行状态的执行时间段以及用于非执行挂起状态的挂起时间段)。例如,可节制消息进程可以被允许每5分钟执行20秒来检查新消息。

[0041] 在另一示例中,在312处,对于被分配了豁免分类的相应豁免进程,豁免进程可以被允许继续在执行状态中执行。豁免分类可以对应于进程是否不能在计算系统故障的情况下被挂起或被节制(例如,核心内核进程、设备驱动器、和/或用于核心内核进程的支持进程、等等)。按照这种方式,提供用于计算环境的核心功能和/或提供所希望功能的豁免进程可以被允许在计算环境处于连接待机状态时在执行状态中执行。

[0042] 在将计算环境转变为连接待机状态的一个示例中,在可节制进程被节制之前,可挂起进程可以被置于非执行挂起状态。因为可挂起进程可能依赖于可节制进程(例如,可挂起字处理器应用可以依赖于用于访问存储的文本文档的可节制存储装置访问进程)。在节制之前的挂起可以减轻这样的实例,其中可挂起进程可能失败,因为可挂起进程所依赖的节制进程可能在可挂起进程挂起之前就已经被置于非执行挂起状态。

[0043] 在将计算环境转变为连接待机状态的一个示例中,彼此相关联的可挂起进程组可以被挂起,这例如可以减轻兼容性风险。在一个示例中,第一可挂起进程可以与第二可挂起进程聚集,这是因为第一可挂起进程可以使用与第二可挂起资源相关联的资源(例如,字处理器应用进程可以使用来自拼写检查器进程的拼写检查数据,因此这些进程可以被聚集在一起)。在另一个示例中,第一可挂起进程可以与第二可挂起进程聚集,这是因为第一可挂起进程可以是第二可挂起进程的子进程(例如,相片更新进程可以是在线相片分享进程的子进程,因此这些进程可以被聚集在一起)。在另一个示例中,第一可挂起进程可以与第二可挂起进程聚集,这是因为第一可挂起进程可以在与第二可挂起进程在其中被初始化的第二逻辑容器相关联的第一逻辑容器中被初始化(例如,在线相片分享进程可以在父工作组内初始化,而相片更新进程可以在嵌入到该父工作组中的子工作组内初始化,因此这些进程可以被聚集在一起)。按照这种方式,对于挂起而言,可以彼此相关联和/或彼此依赖的可挂起进程可以被聚集在一起成为组。

[0044] 在处于连接待机状态时,执行状态内的进程可以创建新进程。在创建时,可以用功率管理分类对该新进程分类。即,即使计算环境可能已经处于连接待机状态,也可以将功率管理分类应用于新进程。

[0045] 按照这种方式,计算环境可以被转变为连接待机状态,以降低功率使用。可以意识到,可以实施附加的省电技术,比如,关掉不必要的硬件和/或资源,比如将与计算环境相关联的计算设备的屏幕断电。在314处,所述方法结束。

[0046] 图4图示出系统400的示例,该系统400被配置为,基于分配给进程的功率管理分类

分配404而将计算环境转变为连接待机状态410。系统400可以包括转变组件406和/或节制组件408。转变组件406可以被配置为检测与计算环境相关联的进入连接待机请求402。转变组件406可以被配置为将计算环境转变为连接待机状态410,该连接待机状态410可以降低与托管计算环境的计算设备相关联的功率消耗(例如,一个或多个进程可以被挂起和/或节制,以降低与这样的进程相关联的功率消耗)。

[0047] 对于功率管理分类分配404内被分配了挂起分类的相应可挂起进程,转变组件402可以被配置为将可挂起进程置于非执行挂起状态。例如,在计算环境处于连接待机状态410时,打印假脱机程序进程、字处理器进程、拼写检查器进程和/或其他可挂起进程可以被置于非执行挂起状态。在一个示例中,彼此相关联的可挂起进程组可以被一起挂起(例如,第一可挂起进程可以基于以下项中的至少一项而关联于第二可挂起进程:第一可挂起进程是第二可挂起进程的子进程,第一可挂起进程使用来自第二可挂起进程的资源,和/或第一可挂起进程等待被第二可挂起进程锁定的资源)。例如,字处理器进程和拼写检查器进程可以被一起挂起,以减轻在一个进程被挂起而另一个进程留下来执行一段时间的情况下可能发生的兼容性风险(例如,字处理器进程可以使用来自拼写检查器进程的数据,并且在该拼写检查器进程在该字处理器进程之前被挂起的情况下,该字处理器进程可能失败)。

[0048] 对于功率管理分类分配404内被分配了节制分类的相应可节制进程,转变组件406可以将节制时间表分配给可节制进程。例如,防病毒进程可以被分配节制时间表,其中,防病毒进程可以每2分钟执行5秒。节制组件408可以被配置为通过基于节制时间表使可节制进程在执行状态和非执行挂起状态之间循环,将节制时间表应用于可节制进程。在一个示例中,转变组件406可以被配置为在将节制时间表分配给可节制进程之前和/或在节制组件408应用节制时间表之前,将可挂起进程置于非执行挂起状态。在对进程进行节制之前对进程进行挂起可以减轻这样的冲突,其中,可挂起进程依赖于来自可节制进程的数据,但是可节制进程已经在该可挂起进程被挂起之前处于非执行挂起状态。

[0049] 对于功率管理分类分配404内被分配了豁免分类的相应豁免进程,转变组件406可以允许豁免进程继续执行。例如,当计算环境处于连接待机状态410时,网络分组处理器进程、硬件事件处理器进程、即时消息器进程、消息通知器进程、和/或其他豁免进程可以被允许继续执行(例如,网络分组处理器进程和/或硬件事件处理器进程可以被豁免,这是因为这样的功能被计算环境依赖于基本职能,同时,即时消息器进程和消息通知器进程可以被豁免,这是因为这样的功能可能是用户所希望的)。按照这种方式,转变组件406可以将计算环境转变为连接待机状态410。

[0050] 图5中的示例性方法500图示出将计算环境从连接待机状态转变为执行状态的一个实施例。在502处,所述方法开始。在504处,可以接收与计算环境相关联的退出连接待机请求。即,计算环境可能已经被转变为连接待机状态,以降低功率消耗。例如,当处于连接待机状态时,一个或多个进程可能已经被挂起和/或被节制以降低功率消耗,同时,计算环境的基本操作所依赖的进程和/或提供所希望的功能的进程可能已经被允许执行,使得计算环境可以以快速响应的方式无缝地转变为最新执行状态(例如,处于连接待机数小时的平板设备可以在数秒内被转变为执行状态,在该执行状态中,电子邮件消息和其他信息可以是最新的和/或用户数秒内可访问的)。

[0051] 在506处,可以将计算环境从连接待机状态转变为执行状态。在一个示例中,在508

处,对于被分配了挂起分类的相应挂起进程,可以对挂起进程恢复执行功能。挂起分类可以对应于进程是否能够被挂起而没有计算系统故障(例如,计算环境和/或进程的故障)和/或与进程相关联的有限运行时间功能是否不是所希望的(例如,在用户会话内创建的字处理用户应用可以被挂起,这是因为字处理用户应用的功能在计算环境处于连接待机状态时可以不是所希望的,和/或这样的功能可以不被计算环境依赖于基本操作)。

[0052] 在另一个示例中,在510处,对于被分配了节制分类的相应节制进程,可以对节制进程恢复执行功能。例如,对节制进程的执行功能可以基于应用到节制进程上的节制时间表,并且节制时间表可以不作为对该进程恢复执行功能的一部分而应用到该节制进程(例如,如果节制进程处于非执行挂起状态,则节制进程可以被转变为执行状态)。节制分类可以对应于进程是否可以被节制而没有计算系统故障和/或与该进程相关联的有限运行时间功能是否是所希望的(例如,防病毒进程可以被分配了节制分类,这是因为该防病毒进程在数据被写到存储装置之前提供所希望状态的扫描数据功能,并且对该防病毒进程进行节制可以不创建防病毒进程和/或计算环境的故障)。

[0053] 在从连接待机状态转变到执行状态的一个示例中,可以在对挂起进程恢复执行功能之前对节制进程恢复执行功能(例如,510,然后508,即使图5在510之前图示出508),这是因为挂起进程可能依赖于由节制进程提供的数据和/或功能。例如,可以在挂起的相片编辑进程之前对节制的存储装置访问进程恢复执行功能,这是因为如果节制的存储装置访问进程不能够向挂起的相片编辑进程提供对存储装置的访问,则挂起的相片编辑进程可能崩溃。在另一示例中,可以在子挂起进程之前对父挂起进程恢复执行功能,这是因为子挂起进程可以被配置为使用来自父挂起进程的资源。按照这种方式,可以以快速响应的方式将计算环境从连接待机状态转变为执行状态,该快速响应的方式可以向用户提供最新信息而没有过度的延迟(例如,用户可能能够在提交退出连接待机请求的数秒内查看新消息)。在512处,所述方法结束。

[0054] 图6图示出系统600的示例,该系统600被配置为,将计算环境从连接待机状态604转变为执行状态608。系统600可以包括转变组件606。转变组件606可以被配置为检测与计算环境相关联的退出连接待机请求602(例如,用户横过平板设备屏幕猛击)。转变组件606可以被配置为将计算环境从连接待机状态604转变为执行状态608。

[0055] 对于被分配了挂起分类的相应挂起进程,转变组件606可以对挂起进程恢复执行功能。例如,在计算环境处于连接待机状态604时,打印假脱机程序进程、字处理器进程、拼写检查器进程和/或其他进程可以被挂起,这可以降低功率消耗。在恢复了执行功能时,这样的进程可以像在计算环境处于执行状态608时一样正常地执行。

[0056] 对于被分配了节制分类的相应节制进程,转变组件606可以对节制进程恢复执行功能。例如,在计算环境处于连接待机状态604时,防病毒进程可以基于节制时间表而被节制,这是因为防病毒进程可以提供所希望的有限运行时间功能(例如,防病毒进程可以以各种间隔检查与对存储装置的写命令相关联的数据,使得在计算环境转变到执行状态时,到来消息可以被写入存储装置和/或是用户可用的)。在一个示例中,可以通过不应用被应用到节制进程的节制时间表来恢复执行功能(例如,如果节制进程处于非执行挂起状态,则节制进程就可以被转变为执行状态)。

[0057] 在一个示例中,转变组件606可以被配置为在对挂起进程恢复执行功能之前对节

制进程恢复执行功能,这是因为挂起进程可以使用来自节制进程的资源。例如,可以在挂起的相片编辑进程之前对节制的存储装置访问进程恢复执行功能,这是因为如果节制的存储装置访问进程不能够向挂起的相片编辑进程提供对存储装置的访问,则挂起的相片编辑进程可能崩溃。在另一示例中,可以在对子挂起进程恢复执行功能之前对父挂起进程恢复执行功能,这是因为子挂起进程可以使用来自父挂起进程的资源。

[0058] 可以意识到,在计算环境处于连接待机状态604时,一个或多个豁免进程可以被允许在执行状态内执行(例如,网络分组处理器进程、硬件事件处理器进程、即时消息器进程、消息通知器进程、和/或其他进程可以被允许执行)。因此,这样的进程可以被允许在计算环境转变为执行状态608时继续处理。按照这种方式,转变组件606可以以快速响应的方式将计算环境从连接待机状态604无缝地转变为执行状态608,该快速响应的方式可以向用户提供最新信息而没有过度的延迟(例如,用户可能能够在提交退出连接待机请求602的数秒内查看新消息)。

[0059] 图7图示出从连接待机状态704转变为执行状态708的平板设备的示例700。该平板设备可以进入连接待机状态704,以通过允许执行一些进程而节制和/或挂起其他进程,来降低功率消耗。例如,被平板设备依赖于基本操作的核心系统进程和/或不能在不导致计算系统故障的情况下被节制或挂起的其他进程可以被允许以正常的方式执行。可以在不创建计算系统故障的情况下被节制和/或提供所希望的有限运行时间功能的系统服务进程和/或其他进程可以被节制。对进程进行节制可以节电,这是因为节制进程可以在相当数量的时间段内被置于非执行挂起状态。因此,处于非执行挂起状态的节制进程可以不调用资源(例如,处理器、硬件等)来实施功率消耗功能。附加地,可以在不创建计算系统故障的情况下被挂起和/或不提供所希望的有限运行时间功能的进程可以被挂起。对进程进行挂起可以节电,这是因为挂起进程可以被置于非执行挂起状态,使得挂起进程可以不调用资源来实施功率消耗功能。在连接待机状态704的一个示例中,网络功能进程、消息发送服务进程、和/或其他进程可以被节制和/或被允许完全执行。相应地,处于连接待机状态704的平板设备可以继续接收和/或处理到来消息702。按照这种方式,平板设备可以在低功率消耗状态下操作,同时继续提供所希望的功能。

[0060] 平板设备可以被转变为执行状态708。例如,可以接收与退出连接待机请求706相对应的用户输入(例如,可以检测在平板设备上的手指猛击)。可以通过将节制进程和/或挂起进程置于执行状态而将平板设备无缝地转变为执行状态708。因为在平板设备处于连接待机状态704时,至少某功能被允许执行,所以可以以快速响应的方式向用户提供最新信息和/或使最新信息可被用户使用。例如,在输入退出连接待机请求706的短时间跨度(例如,几秒)内,可以向用户呈现与在平板设备处于连接待机状态704时处理的到来消息702相关联的最新信息。例如,在平板设备转变为执行状态时,新消息、RSS提要更新、事件邀请和/或大量的其他信息可以被用户使用。因此,平板设备可以由于连接待机状态而提供显著提高的电池寿命,同时在处于执行状态时仍然向用户提供快速响应且最新的体验。

[0061] 再一实施例涉及包括处理器可执行指令的计算机可读介质,所述处理器可执行指令被配置为实施本文所介绍的技术中的一个或多个。图8中图示出可以按照这些方式进行设计的示例性计算机可读介质,其中,实现方式800包括其上编码了计算机可读数据814的计算机可读介质816(例如,CD-R、DVD-R、或硬盘驱动器盘)。该计算机可读数据814转而包括

计算机指令集812,该指令集被配置为根据本文所阐述的原理中的一个或多个进行操作。在一个这样的实施例800中,处理器可执行计算机指令812可以被配置为实施方法810,比如例如图1的示例性方法100、图3的示例性方法300、和/或图5的示例性方法500中的至少一些。在另一个这样的实施例中,处理器可执行指令812可以被配置为实施系统,比如例如图2的示例性系统200、图4的示例性系统400、和/或图6的示例性系统600中的至少一些。本领域普通技术人员可以设计许多这样的计算机可读介质,它们被配置为根据本文所介绍的技术进行操作。

[0062] 尽管用特定于结构特征和/或方法动作的语言对本主题进行了描述,但应理解,定义在所附权利要求中的主题未必局限于以上描述的特定特征或动作。相反,上述特定特征和动作是作为实施权利要求的示例形式来公开的。

[0063] 如本申请中所使用的,术语“组件”、“模块”、“系统”、“接口”等一般旨在指代计算机相关的实体,或者硬件、硬件和软件的组合、软件、或者执行中的软件。例如,组件可以是,但不限于,在处理器上运行的进程、处理器、对象、可执行文件、执行的线程、程序、和/或计算机。作为说明,在控制器上运行的应用以及该控制器都可以是组件。一个或多个组件可以驻留在进程和/或执行的线程内,并且组件可以位于一个计算机上和/或分布在两个或更多个计算机之间。

[0064] 此外,所要求保护的主体可以被实现为方法、装置或制品,以便使用标准编程和/或工程技术生成软件、固件、硬件、或其任意组合,来控制计算机实施所公开的主题。本文所使用的术语“制品”旨在涵盖可从任何计算机可读设备、载体或介质得到的计算机程序。当然,本领域技术人员应认识到,可以在不背离所要求保护的主体范围或精神的情况下对该配置进行许多修改。

[0065] 图8和以下讨论提供了实施本文所阐述的规定中的一个或多个的实施例的适当计算环境的简要而概括的描述。图8的操作环境仅是适当操作环境的一个示例,并非旨在就该操作环境的用途或功能的范围建议任何限制。示例计算设备包括但不限于,个人计算机、服务器计算机、手持式或膝上型设备、移动设备(比如,移动电话、个人数字助理(PDA)、媒体播放器等)、多处理器系统、消费电子设备、小型计算机、大型计算机、包括任何以上系统或设备的分布式计算环境、等。

[0066] 尽管没要求,但是在一个或多个计算设备执行计算机可读指令的一般情景下描述实施例。可以经由(以下讨论的)计算机可读介质分发计算机可读指令。计算机可读指令可以被实现为实施特定任务或实施特定抽象数据类型的程序模块,比如函数、对象、应用编程接口(API)、数据结构等。典型地,可以在各种环境中按照希望合并或分布计算机可读指令的功能。

[0067] 图9图示出包括计算设备912的系统910的示例,其被配置为实施本文所提供的一个或多个实施例。在一个配置中,计算设备912包括至少一个处理单元916和存储器919。根据计算设备的确切配置和类型,存储器919可以是易失性的(比如例如RAM)、非易失性的(比如例如,ROM、闪存等)或者二者的某种组合。在图9中用虚线914图示了该配置。

[0068] 在其他实施例中,设备912可以包括附加的特征和/或功能。例如,设备912还可以包括附加存储装置(例如,可移除的和/或不可移除的),附加存储装置包括但不限于磁性存储装置、光学存储装置等。图9中用存储装置920图示出这样的附加存储装置。在一个实施例

中,用于实施本文所提供的一个或多个实施例的计算机可读指令可以位于存储装置920中。存储装置920还可以存储用于实施操作系统、应用程序等的其他计算机可读指令。例如,计算机可读指令可以被加载到存储器919中,以便由处理单元916执行。

[0069] 本文所使用的术语“计算机可读介质”包括计算机存储介质。计算机存储介质包括,以用于存储信息(比如,计算机可读指令或其他数据)的任何方法或技术实施的易失性和非易失性的、可移除和不可移除的介质。存储器919和存储装置920是计算机存储介质的示例。计算机存储介质包括但不限于RAM、ROM、EEPROM、闪速存储器或其他存储器技术,CD-ROM、数字多功能盘(DVD)或其他光学存储装置,磁盒、磁带、磁盘存储装置或其他磁存储设备,或能够用于存储所希望的信息并且能够由设备912访问的任何其他介质。任何这样的计算机存储介质可以是设备912的一部分。

[0070] 设备912还可以包括允许设备912与其他设备进行通信的(多个)通信连接926。(多个)通信连接926可以包括但不限于,调制解调器、网络接口卡(NIC)、综合网络接口、无线电频率发射机/接收机、红外端口、USB连接、或用于将计算设备912与其他计算设备相连的其他接口。(多个)通信连接926可以包括有线连接或无线连接。(多个)通信连接926可以发送和/或接收通信介质。

[0071] 术语“计算机可读介质”可以包括通信介质。典型地,通信介质在诸如载波或其他传输机制之类的“调制数据信号”中具体实现计算机可读指令或其他数据,并且通信介质包括任何信息传递介质。术语“经调制的数据信号”可以包括,以在信号中编码信息这样的方式设置或改变其特性中的一个或多个的信号。

[0072] 设备912可以包括(多个)输入设备924,比如键盘、鼠标、笔、语音输入设备、触摸输入设备、红外线照相机、视频输入设备和/或任何其他输入设备。也可以在设备912中包括(多个)输出设备922,比如一个或多个显示器、扬声器、打印机、和/或任何其他输出设备。(多个)输入设备924和(多个)输出设备922可以经由有线连接、无线连接或其任何组合而连接至设备912。在一个实施例中,来自另一计算设备的输入设备或输出设备可以用作计算设备912的(多个)输入设备924或(多个)输出设备922。

[0073] 计算设备912的组件可以通过诸如总线之类的各种互连相连接。这样的互连可以包括诸如PCI Express之类的外围组件互连(PCI)、通用串行总线(USB)、火线(IEEE 1394)、光学总线结构等。在另一实施例中,计算设备912的组件可以通过网络互连。例如,存储器919可以包括位于不同物理位置的由网络互连的多个物理存储器单元。

[0074] 本领域技术人员应认识到,用于存储计算机可读指令的存储器设备可以横跨网络分布。例如,可经由网络929访问的计算设备930可以存储用于实施本文所提供的一个或多个实施例的计算机可读指令。计算设备912可以访问计算设备930,并且下载计算机可读指令的部分或全部来执行。可替换地,计算设备912可以根据需要下载计算机可读指令段,或者一些指令可以在计算设备912处执行,而一些指令在计算设备930处执行。

[0075] 本文提供了实施例的各种操作。在一个实施例中,所描述的操作中的一个或多个可以组成在一个或多个计算机可读介质上存储的计算机可读指令,如果这些计算机可读指令被计算设备执行,则其将使得该计算设备实施所描述的操作。一些或所有操作被描述的顺序应该不被解释为暗示这些操作必然是依赖于顺序的。得益于本描述的本领域技术人员将意识到可替换的排序。此外,应该理解,并非所有操作都必然存在于本文所提供的每个实

施例中。

[0076] 而且,词语“示例性”在本文中用来表示用作示例、实例或说明。本文描述为“示例性”的任何方面或设计并非必然解释为优于其他方面或设计。相反,词语示例性的使用旨在以具体方式表达概念。如本申请中所使用的,术语“或”旨在表示包括性的“或者”而不是排他性的“或者”。即,除非另外指定,或从上下文清楚可见,否则“X采用A或B”旨在表示自然包括性排列中的任何一个。即,如果X采用A;X采用B;或者X采用A和B二者,则在前述实例中的任何一个的情况下,都满足“X采用A或B”。附加地,本申请和所附权利要求中使用的冠词“一”和“一个”一般可以被解释成表示“一个或多个”,除非另外指定或从上下文清楚可见针对单数形式。此外,A和B中的至少一个和/或类似物一般表示A或B或A和B二者。

[0077] 此外,尽管已经相对于一个或多个实现方式示出并描述了本公开,但是本领域技术人员基于对本说明书和附图的阅读和理解将得到等同改变和修改。本公开包括所有这样的修改和改变,并且仅仅受以下权利要求的范围的限制。特别地,对于由以上描述的组件(例如,元件、资源等)实施的各种功能,除非另外指示,否则用于描述这样的组件的术语旨在对应于实施所描述组件的所指定功能的任何组件(例如,其在功能上等同),即使在结构上并不等同于实施在本文所图示的本公开的示例性实现方式中的功能的所公开的结构。此外,尽管可能相对于若干实现方式中的唯一一个公开了本公开的特定特征,但是这样的特征可以与其他实现方式的一个或多个其他特征结合,这对于任何给定或特定应用而言,可能都是所希望和有利的。此外,到了在详细描述或权利要求中使用术语“包括”、“具有”、“含有”、“带有”或者它们的变型的程度,这样的术语旨在以类似于术语“包含”的方式进行包括。

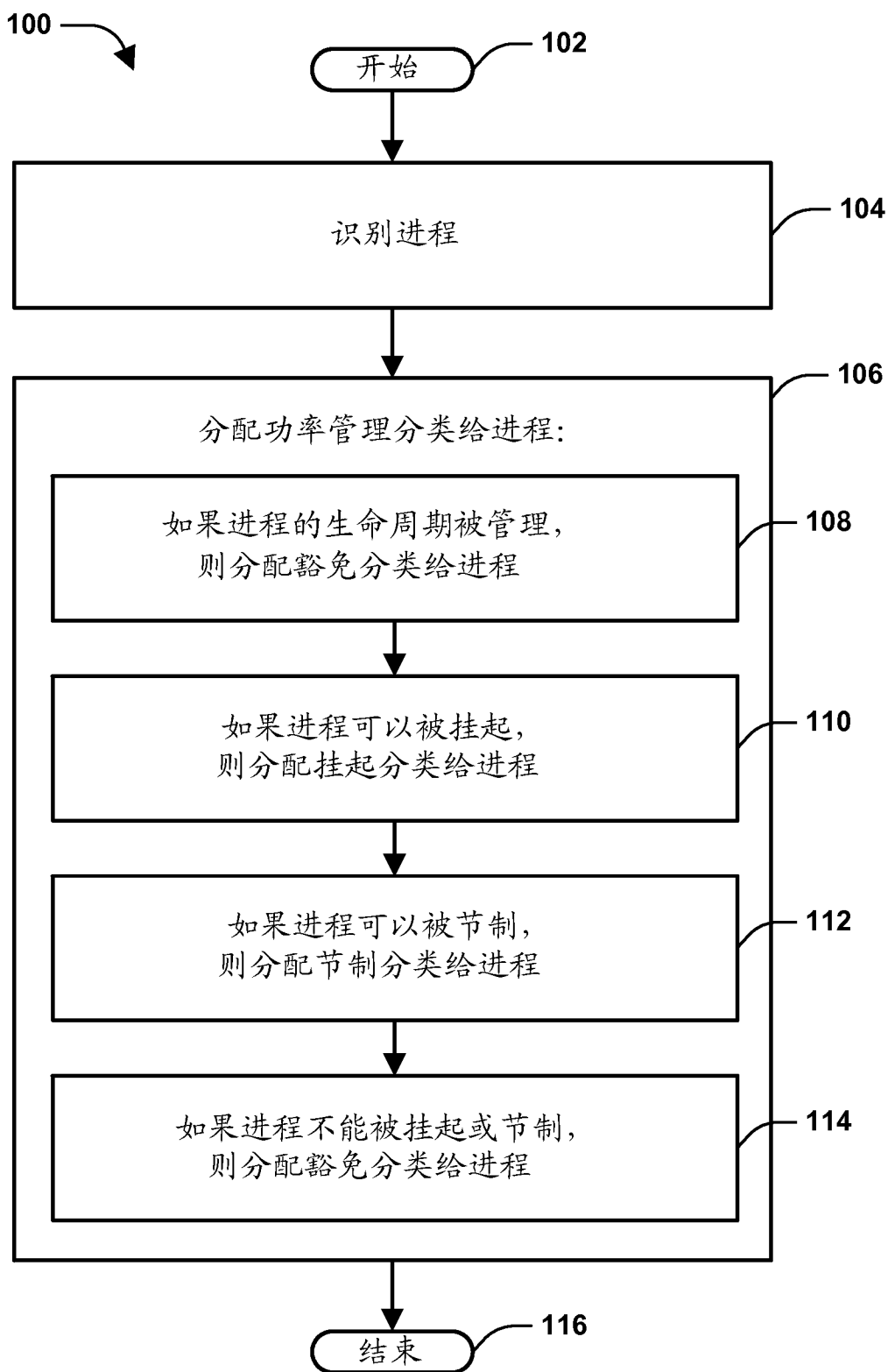


图 1

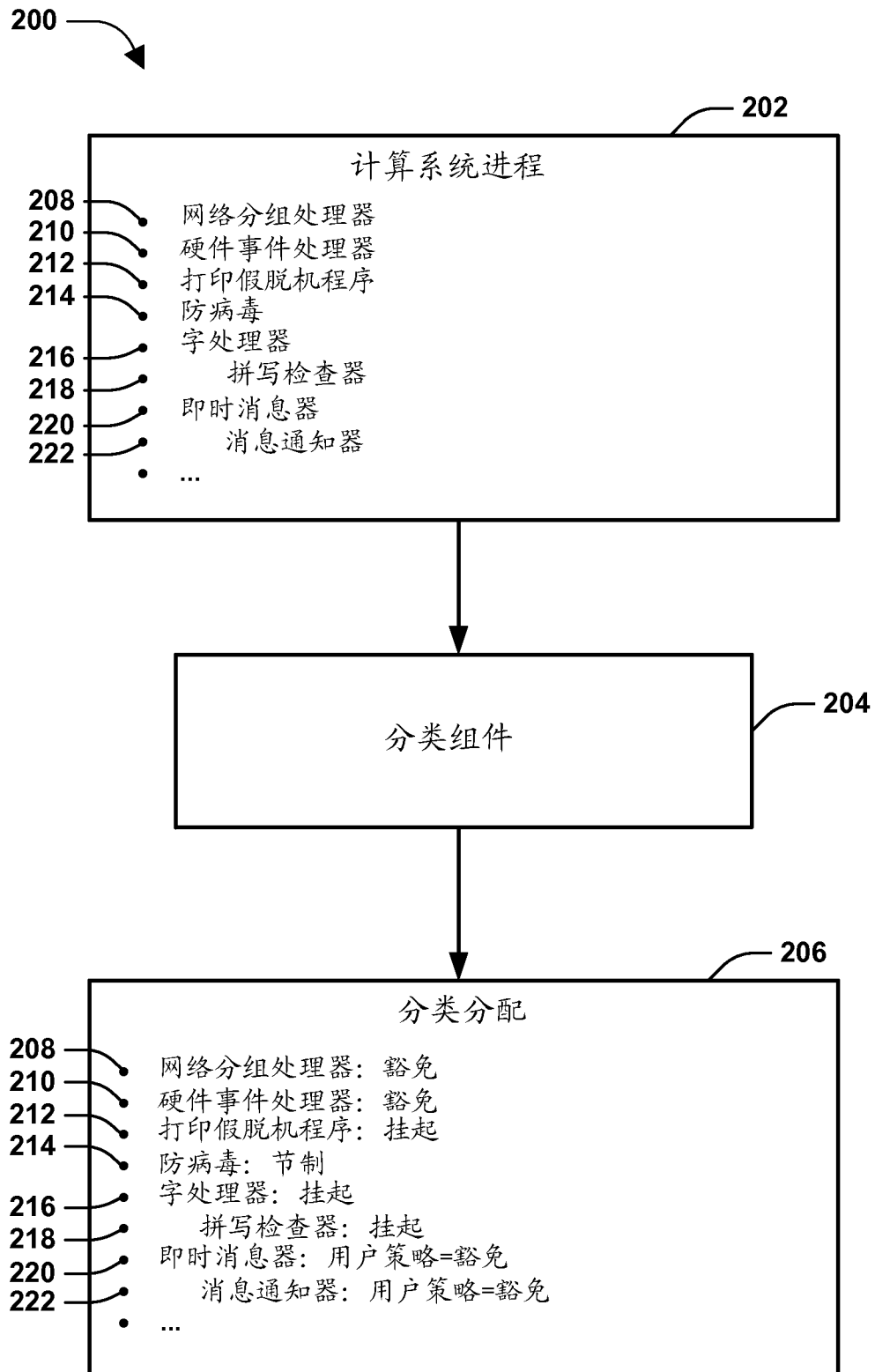


图 2

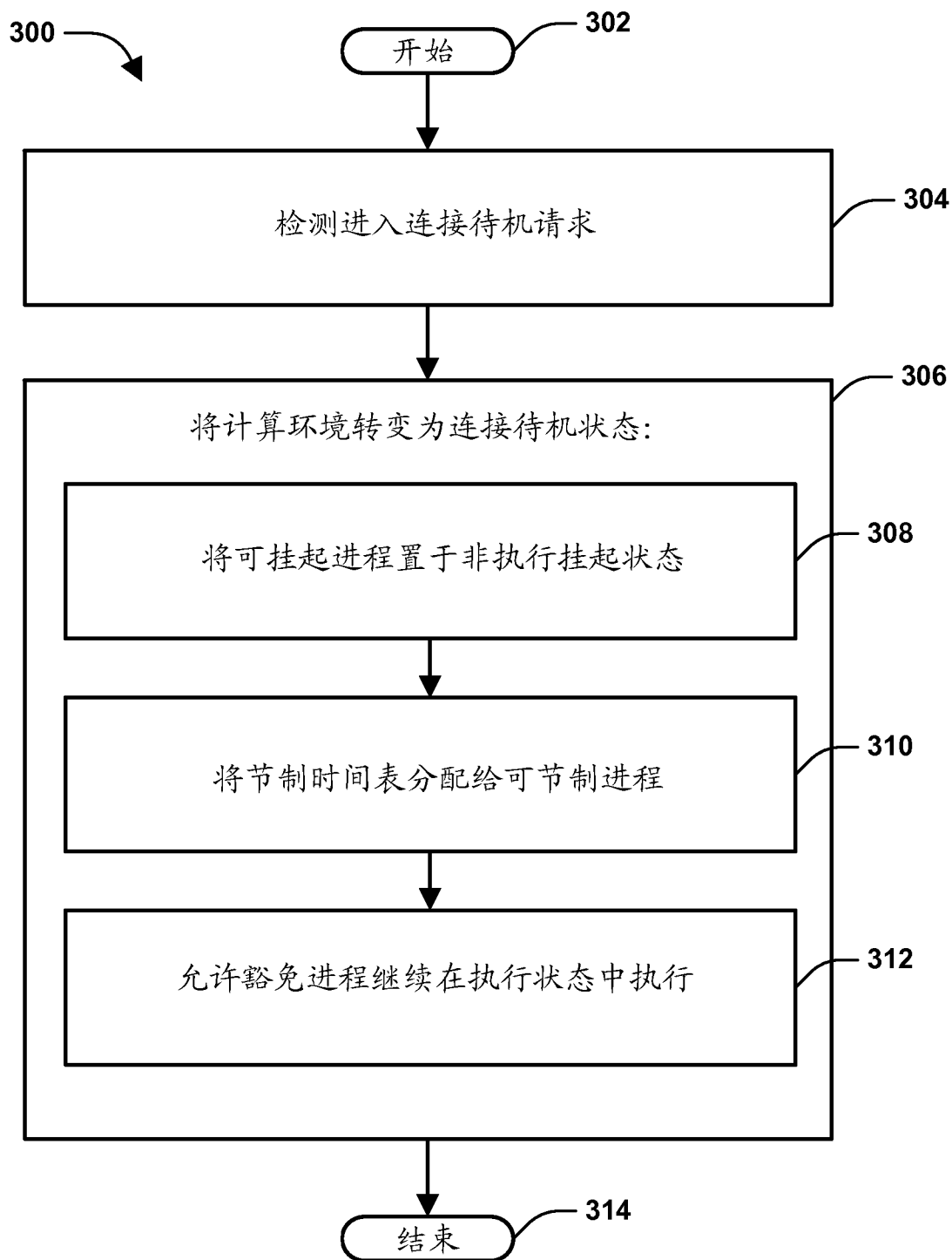


图 3

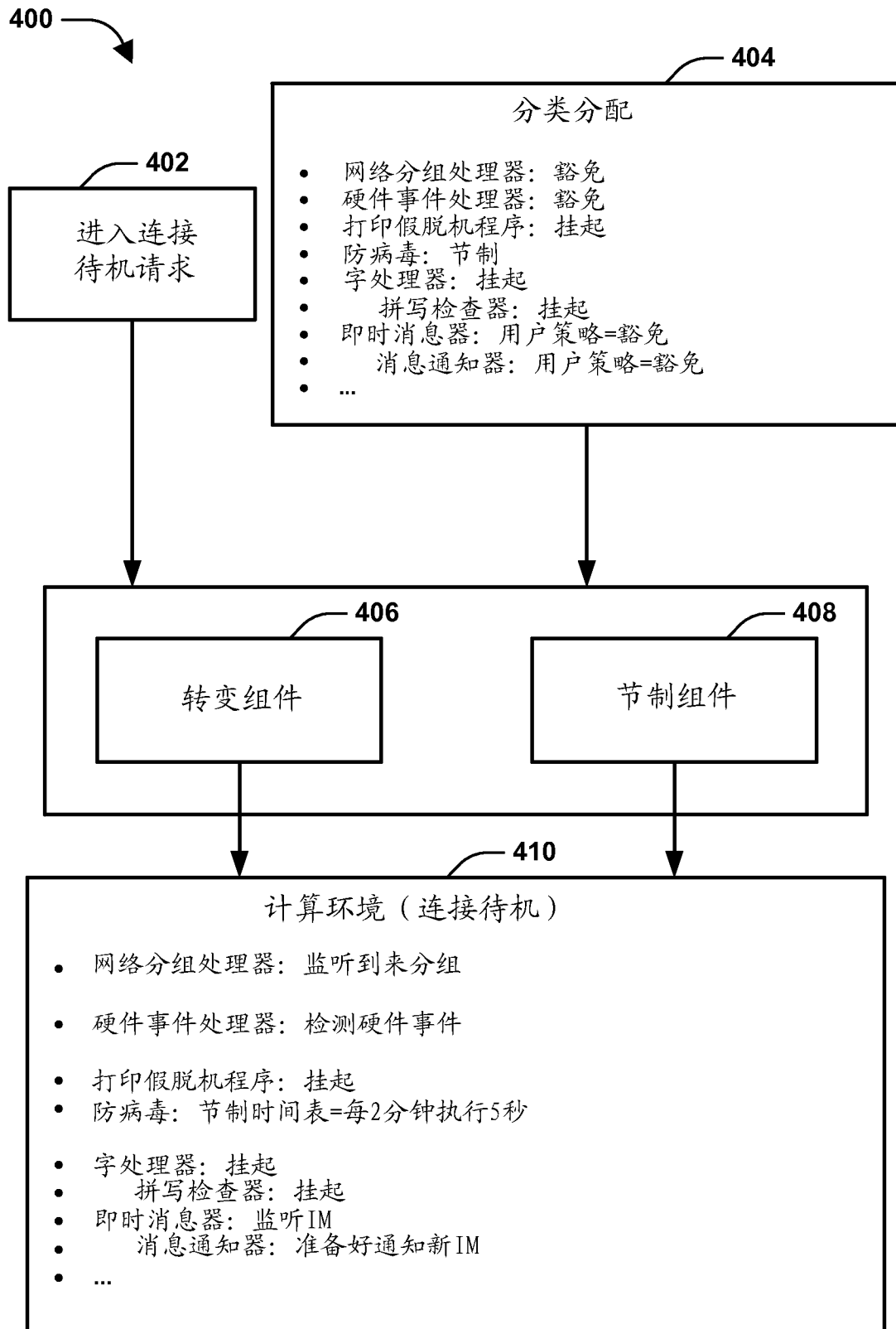


图 4

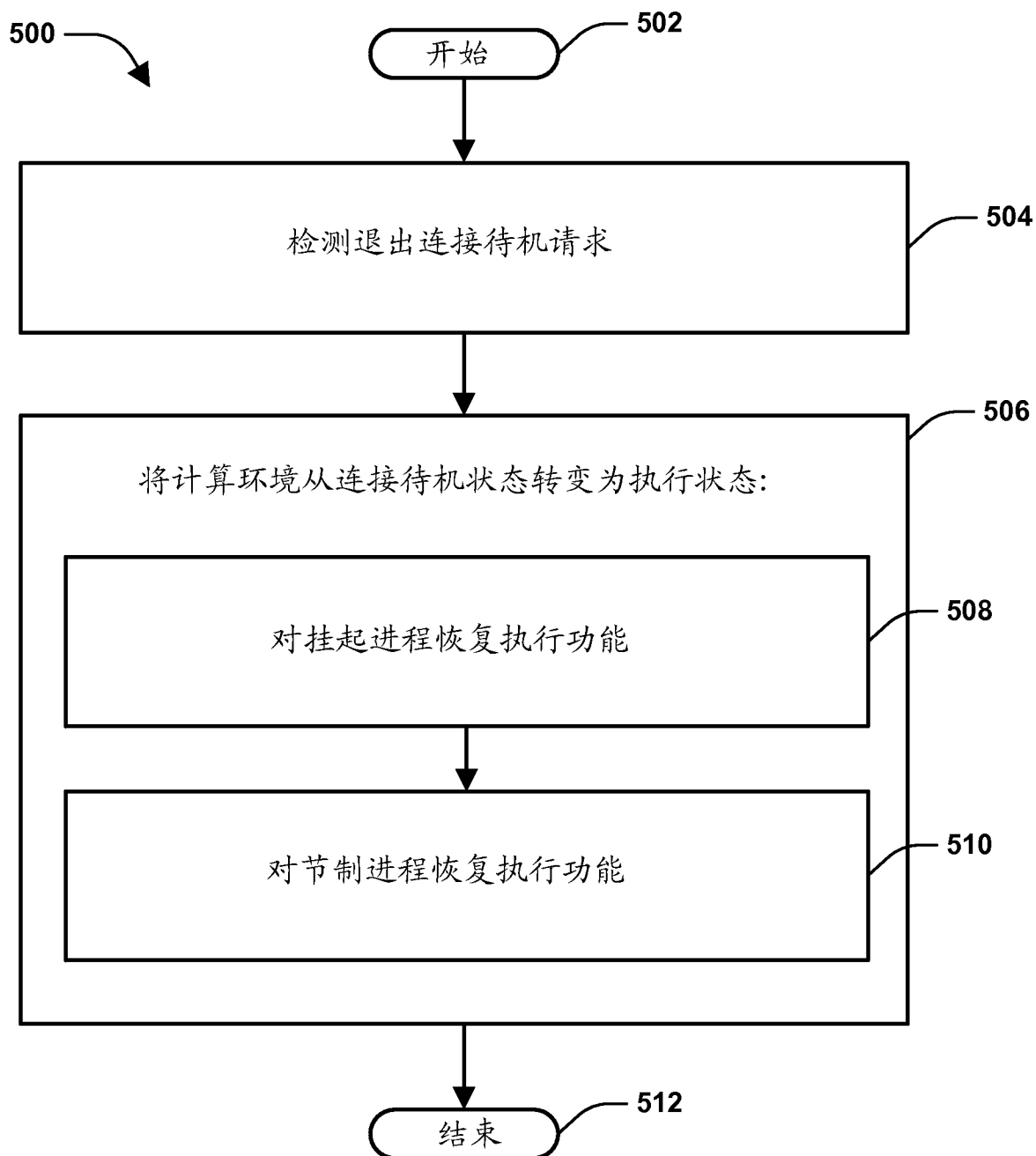


图 5

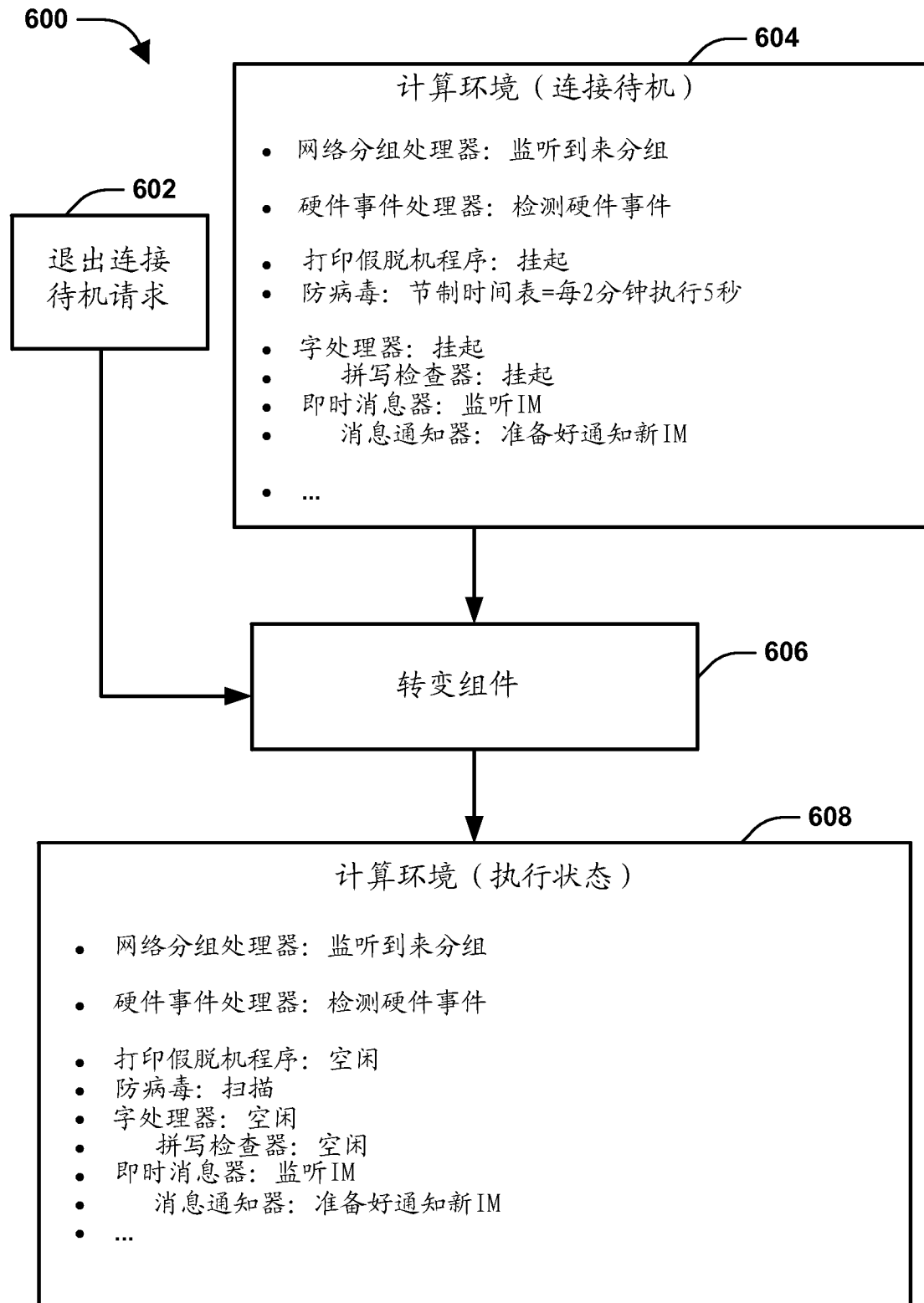


图 6

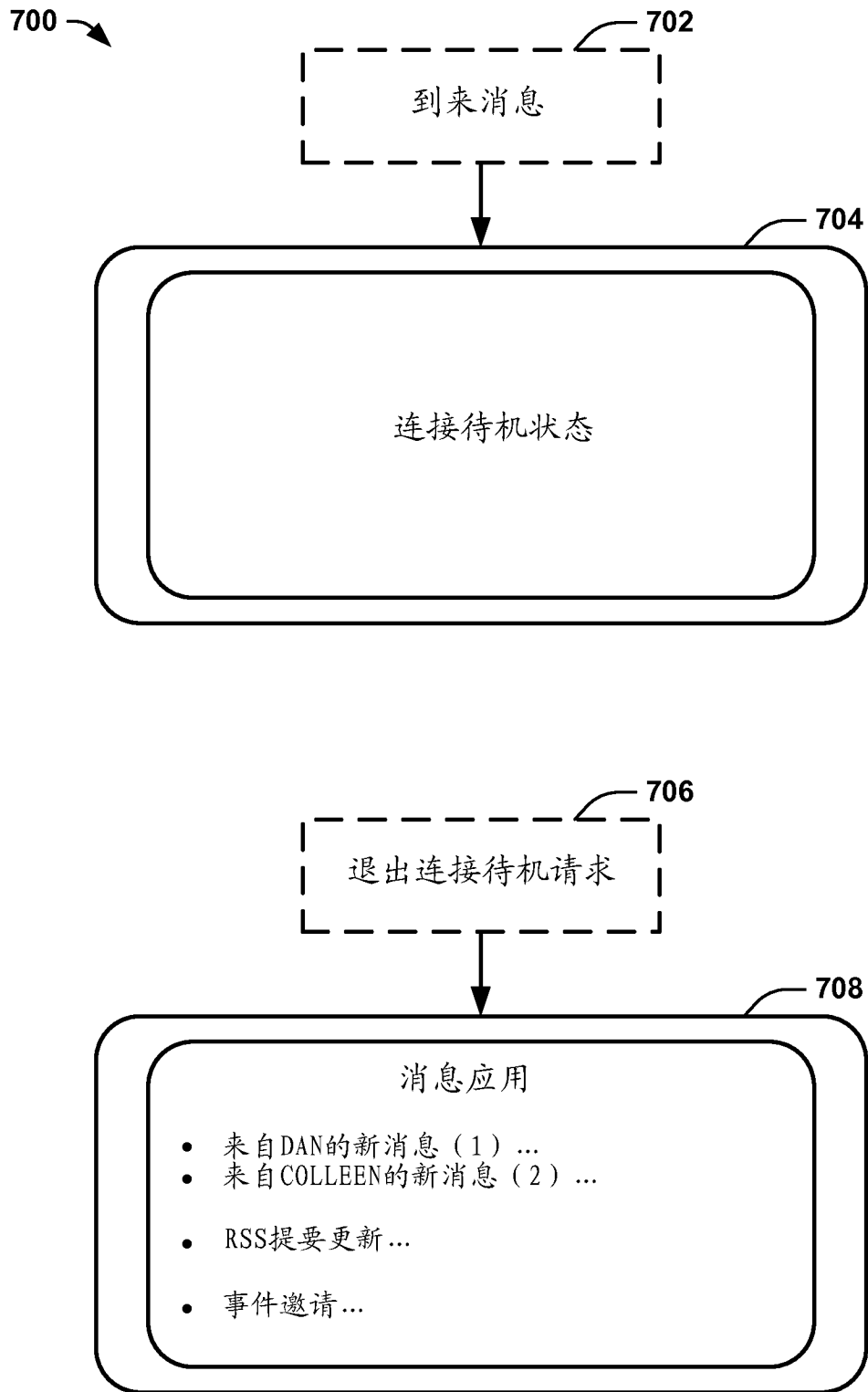


图 7

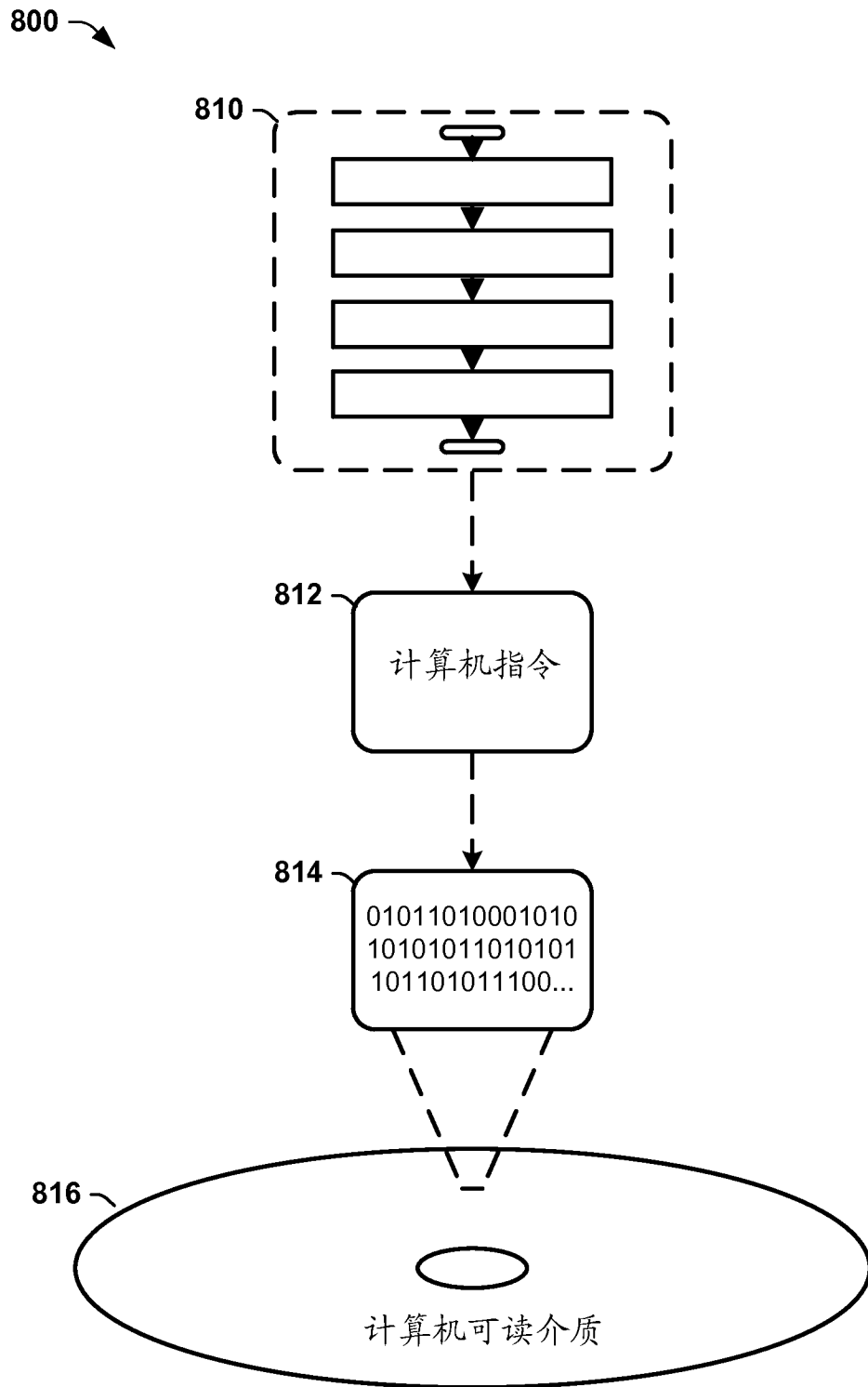


图 8

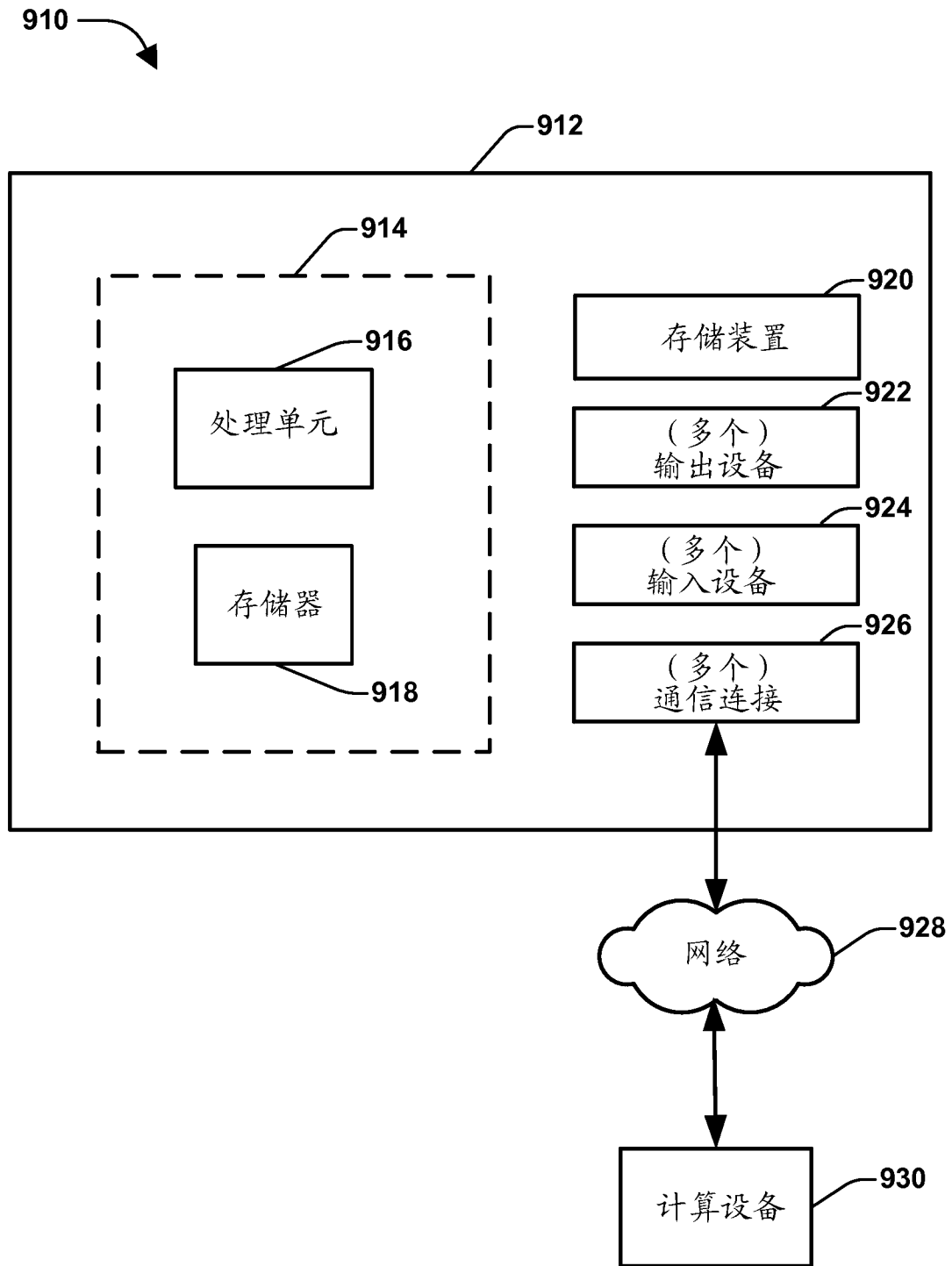


图 9