

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.
G06F 9/44 (2006.01)



[12] 发明专利说明书

专利号 ZL 200410032245.9

[45] 授权公告日 2009 年 6 月 3 日

[11] 授权公告号 CN 100495327C

[22] 申请日 2004.3.26

[21] 申请号 200410032245.9

[30] 优先权

[32] 2003.3.26 [33] US [31] 10/402,179

[73] 专利权人 微软公司

地址 美国华盛顿州

[72] 发明人 T·P·麦克基 F·A·迪伯里

[56] 参考文献

KR20010036672 A 2001.5.7

US6453311 B1 2002.9.17

KR20010037797 A 2001.5.15

审查员 周述虹

[74] 专利代理机构 上海专利商标事务所有限公司
代理人 陈 斌

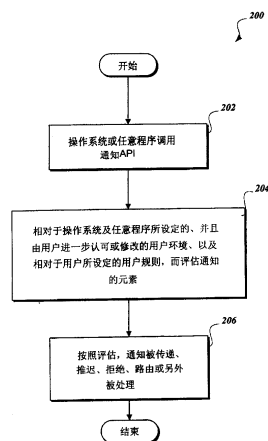
权利要求书 5 页 说明书 18 页 附图 14 页

[54] 发明名称

使用测试通知的系统和方法

[57] 摘要

一种使用测试通知的系统和方法。按照用户的当前环境，应用程序能够构造一测试通知，该测试通知被发送至用户环境系统，所述程序接收回一指示，指明通知在当前时间在屏幕上被绘制或不被绘制。在另一实施例中，调用应用程序接收回较丰富的指令，比如有关是否要推迟或路由通知的细节、它会被播放的声级等等。在轮询实现中，应用程序可以周期性地重新发送测试通知来反复轮询系统以确定是否发生变化。在预订实现中，应用程序可以向系统预订以接收在发生变化时提供的更新。



1. 一种在包括处理器、存储器和显示器的计算设备中使用测试通知的方法，包括：

响应于从所述计算设备上运行的应用程序调用的测试通知接口接收测试通知，评估所述测试通知但是不把所述测试通知传递给所述应用程序的用户；以及向所述应用程序返回有关对测试通知的评估结果的指示。

2. 如权利要求 1 所述的方法，其特征在于，所述测试通知模拟了实际通知，且模拟结果指明是否已应把实际通知传递给所述应用程序的用户。

3. 如权利要求 1 所述的方法，其特征在于，对所述测试通知的评估是按照用户环境完成的。

4. 如权利要求 3 所述的方法，其特征在于，所述用户环境包括或为真或为假的条件以及如果条件为真就要执行的指令。

5. 如权利要求 4 所述的方法，其特征在于，所述用户环境的条件是用于接收通知的所述应用程序的所述用户的可用性，而所述指令是是否应该向所述应用程序的所述用户传递实际通知。

6. 如权利要求 1 所述的方法，其特征在于，对所述测试通知的评估是按照用户规则完成的，所述用户规则规定了包含特定指定内容的测试通知要怎样被处理。

7. 如权利要求 1 所述的方法，其特征在于，还包括轮询附加的测试通知，以便使用附加的测试通知来反复轮询所述计算设备以确定对用户环境指定的变化何时已发生。

8. 如权利要求 1 所述的方法，其特征在于，还包括预订对用户环境的变化，使得在对用户环境的变化已发生时，所述计算设备向所述应用程序自动提供指示。

9. 如权利要求 1 所述的方法, 其特征在于, 向应用程序返回的有关对测试通知的评估结果的指示包括传递指示。

10. 如权利要求 9 所述的方法, 其特征在于, 所述传递指示或为真或为假。

11. 如权利要求 9 所述的方法, 其特征在于, 向应用程序返回的有关对测试通知的评估结果的指示还包括路由或推迟指示之一。

12. 如权利要求 9 所述的方法, 其特征在于, 向应用程序返回的有关对测试通知的评估结果的指示还包括侵入性指示。

13. 如权利要求 9 所述的方法, 其特征在于, 向应用程序返回的有关对测试通知的评估结果的指示还包括音量指示。

14. 一种在包括处理器、存储器和显示器的计算设备中使用测试通知的方法, 包括:

响应于从运行于所述计算设备上的应用程序调用的测试通知接口接收模拟实际通知的测试通知, 评估所述测试通知; 以及

向所述应用程序提供有关是否已应向所述应用程序的用户传递所述实际通知的指示。

15. 如权利要求 14 所述的方法, 其特征在于, 所述测试通知是从调用应用程序接收到的, 把所述有关是否会传递了所述实际通知的指示被返回给所述调用应用程序。

16. 如权利要求 14 所述的方法, 其特征在于, 对所述测试通知的评估是按照用户环境完成的。

17. 如权利要求 16 所述的方法, 其特征在于, 所述用户环境包括或为真或为假的条件以及如果条件为真则要被执行的指令。

18. 如权利要求 17 所述的方法，其特征在于，所述用户环境的条件是用于接收通知的所述应用程序的用户的可用性，而所述指令是是否应该向所述应用程序的所述用户传递实际通知。

19. 如权利要求 14 所述的方法，其特征在于，对所述测试通知的评估是按照用户规则完成的，所述用户规则规定了包含特定指定内容的实际通知要被怎样处理。

20. 如权利要求 14 所述的方法，其特征在于，还包括轮询附加的测试通知，以便使用附加的测试通知来反复轮询所述计算设备以确定对用户环境的指定的变化何时已发生。

21. 如权利要求 14 所述的方法，其特征在于，还包括预订对用户环境的改变，以使得在对用户环境的变化已发生时，所述计算设备向所述应用程序自动自动提供指示。

22. 如权利要求 14 所述的方法，其特征在于，向所述应用程序提供的有关是否已应向所述应用程序的用户传递所述实际通知的指示还包括与所述实际通知已应怎样被处理有关的附加指示。

23. 一种系统，包括：

一个或多个处理器；

由所述处理器可访问的存储器；

由所述处理器可访问的一个或多个易失性和非易失性存储设备；

用于提供通知的通知工具，所述通知工具包括：

用于从应用程序调用的测试通知接口接收测试通知的装置；

用于评估所述测试通知但不把所述测试通知传递给所述应用程序的用户的装置；以及

用于向所述应用程序返回与对所述测试通知的评估结果有关的指示的装置。

24. 如权利要求 23 所述的系统，其特征在于，所述通知工具还包括按照用户环境评估测试通知的装置。

25. 如权利要求 23 所述的系统，其特征在于，所述通知工具还包括按照用户规则评估测试通知的装置。

26. 如权利要求 23 所述的系统，其特征在于，所述通知工具还包括用于轮询附加的测试通知的装置，其中附加的测试通知用于反复轮询所述系统以确定指定的变化何时已发生。

27. 如权利要求 23 所述的系统，其特征在于，所述通知工具还包括预订对用户环境的变化装置的装置，其中当对用户环境的变化已发生时，所述系统向所述应用程序自动提供指示。

28. 一种系统，包括：

一个或多个处理器；

由所述处理器可访问的存储器；

由所述处理器可访问的一个或多个易失性和非易失性存储设备；

用于提供通知的通知工具，所述通知工具包括：

用于从应用程序调用的测试通知接口接收模拟实际通知的测试通知的装置；以及

用于评估测试通知并且向所述应用程序提供与是否已应向所述应用程序的用户传递所述实际通知有关的指示的装置。

29. 如权利要求 28 所述的系统，其特征在于，所述通知工具还包括按照用户环境评估测试通知的装置。

30. 如权利要求 28 所述的系统，其特征在于，所述通知工具还包括按照用户规则评估测试通知的装置。

31. 如权利要求 28 所述的系统，其特征在于，所述通知工具还包括用于轮询

附加的测试通知的装置,其中附加的测试通知用来反复轮询系统以确定指定的变化何时已发生。

32. 如权利要求 28 所述的系统,其特征在于,所述通知工具还包括预订对用户环境的变化变化的装置,其中当对用户环境的变化已发生时,所述系统向所述应用程序自动提供指示。

使用测试通知的系统和方法

技术领域

本发明涉及计算系统中的通知，尤其涉及使用测试通知的系统和方法。

背景技术

在计算机系统中，通知的形式可以是来自程序的信号，它向用户指明已经发生了指定事件。这种通知可以包含文本、声音和图形的各个元素。通知内也可以包括其它属性，比如优先级、发送该通知的人(对于电子邮件或即时消息传递这样的信道)以及通知到期时间。通知还可以包括编码的某些元素，使得用户可以与通知交互以及启动任意代码(例如点击通知内的按钮或文本，这会造成启动新程序或者对目前正在运行的程序采取行动)。

操作系统可以创建通知使用户获悉有关网络连接和更新。使用“联系人列表”的即时消息传递程序可以在屏幕上绘制通知来让用户得知联系人列表上正在发生的事项或者联系人何时开始即时消息对话。其它程序可以提供在显示器的类似区域内绘制的类似通知。有关这些通知类型的一个问题是：它们一般不知道其它通知，因此有时会导致在其它通知上绘制通知。

现有系统的另一个问题是：它们会造成不适当地传递通知，或者在不适当的时间传递通知。例如，对于提供全屏演示的用户而言，不宜在演示期间使其它程序在屏幕上绘制通知。可能绘制这种不适当通知的程序例子是一即时消息传递程序，它在操作系统的后台运行并且当联系人列表内的联系人指令或开始即时消息时绘制这种通知。演示期间的这类“中断”是用户不期望的。

本发明提供了克服上述及其他缺点的系统和方法。更具体地说，本发明针对使用测试通知的系统和方法。

发明内容

提供了一种使用测试通知的系统和方法。按照本发明一方面，调用程序构成了与实际通知类似的测试通知。与实际通知的关键差异是测试通知实际上不会被传递给用户。

按照本发明另一方面，在一个实施例中，测试通知返回一个真或假的指示。指示为真意指会在当前时间绘制实际通知，而指示为假意指不能在当前时间绘制实际通知。在另一实施例中，测试通知也可以返回较丰富的返回值。较丰富的返回值的例子包括通知是否会被推迟或路由、通知会在哪个声音级被播放等等。

按照本发明另一方面，在一个实施例中使用了轮询方法。在该实施例中，调用程序周期性地反复轮询(发送测试通知)以确定用户的当前环境以及调用程序发送的任何广播数据应该怎样变化。广播数据的例子可以是向联系人广播繁忙或空闲状态的即时消息传递程序。在另一实施例中使用了预订回叫方法。在该实施例中，调用程序预订以接收环境变化。换言之，随着用户的环境关于广播的信息类型而变化，环境引擎根据这些更新回叫程序，而不是周期性地反复轮询。后一种实施例的优点是：在环境变化和广播内容之间没有时滞。在其它情况下，轮询方法也许更为适当，比如对于一次性事件而言。

可以理解，本发明的测试系统和方法允许操作系统和任意程序来确定何时适于向用户发送通知。使用测试通知的一个优点在于：通过允许程序更多地得知用户环境，可以防止产生不希望的实际通知，直到用户处在能接收的状态为止。此外，为不使用相同通知传递的用户界面的程序提供了较大的灵活性，因为通过使用测试通知，它们仍能获得有关用户环境的信息。例如，将来的程序可以研发具有特殊图形的通知，也许不被当前的用户环境系统所支持，然而将来的程序仍能使用从测试通知获得的信息来确定是否适于发送或修改其当前的通知。

按照本发明另一方面，在用户环境系统中使用了测试通知。用户环境系统代理并连续排列来自多个源的通知传递。此外，提供了用户环境的共享概念，用于为每个通知确定适当的处理。按照这些方面，可以认为由用户环境系统传递的通知更有价值，因为它们是在用户更能接收它们时被传递的。这些方面还提供了公共规则，这些规则帮助用户删除不期望的通知。在一实施例中，用户环境系统中测试通知的使用基本上提供了一种系统工具或系统提供的机制，该工具或机制能够封装用户的总体环境并使其经受任意的处理。

按照本发明另一方面，在用户环境系统中，环境是由操作系统和任意程序宣布的。在一实施例中，用户环境包括可以为真或为假的条件以及如果条件为真则服从的指令。例如，条件可以是“当用户正在听音乐”，对此的指令会是“在屏幕上递送通知但没有声音”。通常，用户环境的条件可以被认为是系统可采取的一种状态，该状态使用户在某些方面不可用于通知传递，或者造成应该传递通知的方式与

它被开始的程序所发送的方式不同。用户可以处在称为“不可用”的状态，其中通知或者不被传递，或者被保留直到用户变得“可用”。例如，如果用户正在运行全屏应用程序，其中应用程序正在使用显示屏的全部区域或者在其上被显示，该用户就被视为不可用。或者，用户可以“可供使用”，但在这种状态下，需要修改通知以适用于该用户。

按照本发明另一方面，在用户环境系统中，除了宣布环境的操作系统以外，程序向系统注册，并且宣布它们提供的环境以及它对通知的影响(根据在屏幕上绘制是否适当以及适用于在屏幕上绘制的侵入性级别，以及声音是否适当或者声音应该以什么相对音量被播放)，然后告知系统该环境为真还是为假。在一实施例中，在要传递通知时可以评估环境为真或是为假。在一实施例中，系统还可以跟踪调用程序的过程，如果过程不再存在，就把环境重置为假。通过跟踪该过程，可以避免特定的不期望的情况，比如一应用程序，该程序宣布用户处于繁忙、然后崩溃、然后使用户陷于一状态，该状态中他们已被宣布为不可用于接收通知。

按照本发明另一方面，在用户环境系统中，存在为绘制通知指定的不同的扩散性级别。换言之，根据用户环境，绘制消息会有梯度，使得会有以所绘制通知为形式的不同侵入性级别。例如，正常的通知能够被自由地绘制在客户机区域中，并且简要地遮掩一个窗口。如果用户处于略微受限制的环境中，通知可以自由地示出，但是仅以较不侵入的方式示出，比如也许不允许在另一个窗口上面进行绘制。举另一个例子，如果用户正在运行最大化的应用程序，设置可能是用户环境略微受限制，因为用户已经明确地作出声明：他们希望他们当前的应用程序取得整个客户机区域。在这种环境下，仍旧允许绘制通知，但是仅让它们出现在侧边条(sidebar)内。这类以通知绘制为形式的降低了的侵入性减少了通知的影响，并且减少了认知负载。

按照本发明另一方面，在用户环境系统中，已经提供的环境暴露于用户，并且可以或者被关闭(例如用户不同意程序对环境的评价)，或者根据对传递的影响而变化。

按照本发明另一实施例，在用户环境系统中，用户可以定义规则，这些规则规定了包含指定元素的通知应该怎样被传递。例如，用户规则可以规定应该立即传递从“John Doe”接收到并且主题栏为“紧急”的任何通知。在一实施例中，这些用户规则比其它用户环境优先。用户规则可用于用户按照用户偏好进行修改。

附图说明

通过下面提出的结合附图的详细描述,本发明的特征、性质和优点将变得更加明显,附图中相同的元件具有相同的标识,其中:

图 1 是适用于实现本发明的通用计算机系统的框图;

图 2 是说明按照本发明用于处理通知的一般例程的流程图;

图 3 是说明宣布用户环境的操作系统或任意程序的例程的流程图;

图 4 是说明在调用通知 API 时评估用户环境为真或为假的例程的流程图;

图 5 是说明用于调整用户环境并创建新的用户规则的例程的流程图;

图 6 是说明用于按照用户环境和用户规则处理通知的例程的流程图;

图 7 是说明用于根据通知的内容以及用户环境来实现用户规则的例程的流程图;

图 8 是说明用于推迟通知传递的例程的流程图;

图 9 是说明用于按照各种限制性设置确定会怎样绘制通知的例程的流程图;

图 10 是说明用于按照各种限制性设置确定通知的音量级别的例程的流程图;

图 11 是说明按照本发明用于处理测试通知的总例程的流程图;

图 12 是说明用于处理测试通知并且返回真或假的指示的例程的流程图;

图 13 是说明用于处理测试通知并且返回具有全部细节的指示的例程的流程图;以及

图 14 是说明根据测试通知的内容和当前用户环境使用用户规则来处理测试通知的例程的流程图。

具体实施方式

本发明针对一种使用测试通知的系统和方法。在一实施例中,测试通知可以由用户环境系统来处理。用户环境系统控制通知的传递。

在现有系统中,一般有许多希望向用户发送通知的竞争元件,各个元件都设计了发送这种通知的自己的方式。一般没有一个竞争元件已经得知其它通知并因此意图在彼此的以及彼此应用程序的顶上绘制,当每个元件都选择同时提供它们相应通知的指示时,这会导致冲突。此外,还没有用户环境的共享概念,导致某些通知被不适当地传递,或者在不适当的时间被传递。用户环境系统通过把通知构建为操作系统的一大部分而解决了这些问题,这使得用户环境系统所提供的通知的用户界面变得相似并因此不再彼此冲突,因为系统适当地代理并连续排列了它们在屏幕上

的显示。此外，由用户环境系统所提供的通知可以被认为较有价值，因此它们是在用户更能接收它们的时候被传递的，此外，公共规则的使用也帮助用户删除了不期望的通知。

如上所述，可以使用用户环境系统来处理本发明的测试通知。本发明的测试通知用于确定在当前条件下是否会传递实际通知。换言之，测试通知意图提供一指示，指明用户目前是否可用于接收通知。例如，可以使用该指示来防止程序试图在当前时间发送实际通知，或者使程序修改该通知。

图 1 及下列讨论提供了对其中可实现本发明的适当计算环境的简要、一般的描述。尽管不需要，然而仍然会以计算机可读指令的一般上下文来描述本发明，比如个人计算机所执行的程序模块。一般而言，程序模块包括例程、程序、字符、组件、数据结构等等，它们执行特定的任务或者实现特定的抽象数据类型。本领域的技术人员会理解，本发明可以用其它计算机系统配置来实现，包括手提设备、微处理器系统、基于微处理器的或可编程的用户电子设备、网络 PC、小型计算机、大型计算机等等。本发明还可以在分布式计算环境中实现，其中由通过通信网络相连的远程处理设备来执行任务。在分布式计算环境中，程序模块既可位于本地存储设备中、又可位于远程存储设备中。

参照图 1，一种用于实现本发明的示例性系统包括形式为常规个人计算机 20 的通用计算设备，它包括处理单元 21、系统存储器 22 以及把包括系统存储器 22 在内的各种系统组件耦合到处理单元 21 的系统总线 23。系统总线 23 可能是若干总线结构类型中的任一种，包括存储器总线或存储器控制器、外围总线以及使用各种总线结构中任何一个的本地总线。系统存储器包括只读存储器(ROM)24 以及随机存取存储器(RAM)25。基本输入/输出系统(BIOS)26 存储在 ROM 24 内，它包括如启动时帮助在计算机 20 内的元件间传输信息的基本例程。计算机 20 还包括用于向硬盘 39 读写的硬盘驱动器 27、用于向可移动磁盘 29 读写的磁盘驱动器 28 以及用于向诸如 CD ROM 或其它光学媒质这样的可移动光盘 31 读写的光盘驱动器 30。硬盘驱动器 27、磁盘驱动器 28 和光盘驱动器 29 分别通过硬盘驱动接口 32、磁盘驱动接口 33 以及光盘驱动接口 34 连到系统总线 23。驱动器及其相关的计算机可读媒质为个人计算机 20 的计算机可读指令、数据结构、程序模块及其它数据提供了非易失性存储。尽管这里所述的示例性环境采用了硬盘 39、可移动磁盘 29 和可移动光盘 31，然而本领域内的技术人员可以理解，在示例性操作环境中也可以使用其它类型的计算机可读媒介，它们能存储可由计算机存取的数据，这些媒介有：磁带、

闪存卡、数字视频盘、贝努利(Bernouilli)盒带、随机存取存储器(RAM)、只读存储器(ROM)等等。

许多程序模块可以存储在硬盘 39、磁盘 29、光盘 31、ROM 24 或 RAM 25 上,包括操作系统 35、一个或多个应用程序 36、其它程序模块 37 以及程序数据 38。用户可以通过诸如键盘 40 和指示设备 42 这样的输入设备将命令和信息输入计算机 20。其它输入设备(未示出)可能包括麦克风、操纵杆、游戏垫、卫星式转盘、扫描仪等。这些和其它输入设备一般通过与系统总线 23 耦合的串行端口接口 46 连到处理单元 21,但也可能由其它接口连接,诸如并行端口、游戏端口或通用串行总线(USB)。形式为监视器 47 的显示器也通过诸如视频卡或适配器 48 这样的接口连到系统总线 23。一个或多个扬声器 57 也可以通过诸如音频适配器 56 这样的接口连到系统总线 23。除了显示器和扬声器以外,个人计算机一般包括其它外围输出设备(未示出),譬如打印机。

个人计算机 20 可以工作在使用到一个或多个诸如远程计算机 49 这样的远程计算机的逻辑连接的联网环境内。远程计算机 49 可以是另一计算机服务器、路由器、网络 PC、对等设备或其它公共网络节点,并且一般包括上述相对于计算机 20 所述的许多或所有元件。图 1 所述的逻辑连接包括局域网(LAN)51 和广域网(WAN)52。这种联网环境常见于办公室、企业范围的计算机网络、内联网以及因特网。

当用于 LAN 联网环境时,个人计算机 20 通过网络接口或适配器 53 连到局域网 51。当用于 WAN 联网环境时,个人计算机 20 一般包括用于在诸如因特网的广域网 52 上建立通信的调制解调器 54 或其它装置。调制解调器 54 可以是内置或外置的,它通过串行端口接口 46 连到系统总线 23。在联网环境内,相对应个人计算机 20 所述的程序模块或其中的一部分可以被存储在远程存储设备内。可以理解,所示的网络连接是示例性的,并且可以使用其它用于在计算机间建立通信连接的装置。

本发明可以在图 1 所述类型的系统上实现。如上所述,本发明的测试通知可由用户环境系统来评估。下面将参照图 2-10 主要描述用于评估测试通知的用户环境系统。在下面的段落将参照图 11-14 更详细地描述测试通知的评估。

关于图 2-10,提供了用于控制来自多个源的通知传递的用户环境系统。在一实施例中,用户环境系统可以包括三个要素,比较它们以决定怎样处理通知。第一个要素是用户的环境(可以由已经扩展它的操作系统和任意程序所提供)。第二个要

素是用户的规则和偏好。第三个要素是通知自身(它包含诸如可能与用户规则相匹配的数据和属性这样的要素)。

下面将详细描述, 用户环境系统通过操作系统序宣布用户环境的及其他程而进行操作, 其后系统代理用户的环境和规则。通知由其它调用到系统内的程序提出。用户环境、规则和通知的元素相互比较, 然后作出有关应该怎样处理通知的判决。应该怎样处理通知的各种选项的例子包括: 拒绝(如果不允许通知绘制或发出噪声, 且用户将永远看不到通知)、推迟(保持通知直到用户环境变化或者用户规则规定随后适于传递为止)、传递(允许按照用户的环境和规则来传递通知)以及路由(用户规则指示通知应被切换到另一系统, 无论在当前系统中是否也允许传递通知)。

下面将更详细地讨论各种用于传递通知的例程。通常, 用户可以处在称为“不可用”的状态, 其中通知或者不被传递、或者被保留直到用户变得“可用”为止。例如, 如果用户正在运行全屏应用程序, 该用户会被视为不可用。或者, 用户可以是“可用的”, 但这种情况下需要修改通知以适用于用户。例如, 如果用户正在听音乐或开会, 用户也许以指示应该把通知传递到用户的屏幕, 但它们发出的声音应该较轻或是静音。

如上所述, 用户环境部分地确定了通知是否在用户屏幕上示出。当示出通知时, 它可以根据用户环境内的特定梯度被示出。换言之, 可以为所绘制通知的形式指定不同级别的侵入性。例如, 正常的通知可以自由地弹出到客户机区域中并且简要地遮蔽一个窗口。如果用户处在略微限制性的环境中, 则通知可以自由显示, 但仅以较不侵入的方式显示, 比如可能不允许通知在另一窗口上方绘制。举另一个例子, 在用户正在运行最大化应用程序的一个实施例中, 缺省设置会是这意味着该环境是略微受限的, 且用户已清楚作出了他们希望该应用程序取得整个客户机区域的声明。在这个设置中, 仍可以允许绘制通知, 但仅使其出现在侧边条内。换言之, 通知绘制形式中的这类降低了的侵入性减少了通知的影响, 并且总的减少了认知负载。

图 2 是说明用于处理通知的例程 200 的流程图。在方框 202 处, 操作系统或任意程序调用通知应用程序编程接口(API)。在方框 204 处, 相对于操作系统及任意程序所设定的、并且由用户进一步认可或修改的用户环境、以及相对于用户所设定的用户规则, 而评估通知的元素。在方框 206 处, 按照用户环境和用户规则, 通知被传递、推迟、拒绝、路由或被另外处理。

下面将更详细地描述用户环境和用户规则。在一实施例中, 用户环境包括或

为真或为假的条件以及当条件为真时指示应该怎样处理的通知的指示。通常，用户环境的条件可以被认为系统采用的一状态，该状态下使用户在某些方面不可用于传递或者使传递通知的方式与通知被开始它的程序所发送的方式不同。换言之，在一个实施例中用户环境可以被认为是一声明：“当条件 X 为真时，这就是应该对到来的通知作出的处理”。一个例子是“当我的音乐播放器正在为我播放音乐时，到来的通知应该在屏幕上显示但是不放出声音”。另一个例子是“当任何应用程序正以全屏模式运行时，到来的通知应该被推迟到稍后”。

关于这种用户环境，在一个实施例中用户还可以定义用于处理到来通知的特殊规则，因此为用户环境的指令提供了特殊例外。例如，用户规则可以声明“当我从“John Doe”接收到新的电子邮件、文本中有“紧急”并且标记为“高优先级”时，无论其它用户环境是什么都传递这个电子邮件”。换言之，在该例中这个用户规则为用户环境提供了一个例外，该用户环境指出此时不适于为到来的电子邮件传递通知。关于相对用户规则评估的通知元素，这些可以包括类似文本、声音、图形及其他像优先级这样的属性、发送通知的时间、通知何时到期以及某些编码元素，这些元素使用户能与通知交互并且启动任意编码（例如点击通知内的按钮或文本会造成运行新程序或者对当前运行的程序采取新行动[比如删除电子邮件]）。

图 3 是说明宣布用户环境的操作系统或任意编码的例程 220 的流程图。在方框 222 处，操作系统或程序宣布缺省环境以及它们对用户繁忙状态的影响。换言之，程序向系统注册并且提供包括它们对通知的影响的用户环境（例如在屏幕上绘制是否适当，以及声音是否适当或者应该以什么相对音量进行播放）。例如，音乐播放器程序可以宣布一个缺省环境，该环境声明“当音乐播放器为用户播放音乐时，到来的通知应该显示在屏幕上但不播放声音”。举另一个例子，操作系统可以宣布一环境，该环境声明“当任何应用程序以全屏模式运行时，到来的通知应被推迟到稍后的时间”。

返回图 3，在方框 224 处，操作系统或程序把所宣布的环境设为真或为假。例如，关于宣告“当音乐播放器正在播放音乐时，到来的通知应该显示在屏幕上但不播放声音”的环境的音乐播放器，音乐播放器程序还可以把这个所宣布的环境设为当前为真或为假。换言之，音乐播放器程序指示音乐播放器当前正在播放音乐是其或假。下面将更详细描述，在一个实施例中，在调用通知 API 时，或者在重新评估用户规则和例外时，也可以评估环境是为真还是为假的判决。作为附加的特征，系统还可以跟踪调用程序的过程处理，使得如果过程终止而无须首先把环境值设为它

的缺省“假”值，那么只要系统检测到初始过程不再存在(在一个实施例中，当过程终止时把过程处理状态设为信号，该状态变化由观察过程处理的系统所提取)，系统就重置环境值。这确保如果过程意外地终止或者忘记重置环境，那么不会过度影响进一步通知的传递。例如，如果在上例中音乐播放器程序已被关闭并且过程不再存在，则环境被自动地重置为假。举另一个例子，如果程序最初宣布用户为繁忙，但接着程序失效，使得过程不再存在，则把环境自动地设为假，而不是使用户陷于不会接收到通知的状态中。在任何情况下，无论环境是否被主动地设置或者用函数来评估，在一个实施例中，环境一般可以被分析为或为真或为假。

返回图 3，在方框 226 处，向系统中存储的用户环境添加环境信息。这个过程通过附加的程序宣布环境来重复。此外，如上所述，当用户打开或关闭不同的程序并且从事不同的任务时，已经宣布的环境为真或为假的状态会随时间而变化。

如上所述，在一个实施例中环境的注册是一个宣布的过程。下面将更详细地描述，在用户环境系统中，通过注册用户环境，可以向用户给出一环境表列，使用户可以选择不接受特定的环境或者如果用户不同意环境参数则改变他们所指的内容。如上所述，在一个实施例中，环境可以包括可以为真或为假的条件、以及如果条件为真则对通知做什么处理的指令。在这一点上，用户环境可以包括特定的程序设计元素，比如：人类可读字符串(用于终端用户得知所指的意思)；唯一标识符(比如全局唯一标识符、aka GUID)，使得程序可以告知操作系统该环境为真还是不为真；以及可以包括一声明的指令，该声明通过屏幕上绘制的通知来指出该环境的意思(可以包括侵入性级别、声音和音量)。如下面更详细描述，环境也可以是动态的。

图 4 是说明在调用通知 API 时要把环境评估为真或为假的例程 230 的流程图。在判决框 232 处，确定在调用通知 API 时是否要评估用户环境。如果要评估用户环境，则例程进行到方框 234。如果在调用通知 API 时不要评估用户环境，则例程终止。在方框 234 中，用户环境被评估为真或为假。

如图 3 和 4 且如上所述，环境可以被前摄地设置，或者它可以是在有关时刻被评估的函数。例如，程序可以主动地注意到用户正在听音乐。举另一个例子，当评估通知时，程序也许已经注册了它的回叫，使得在评估通知时，系统询问该程序环境是否为真。其中这第二过程特别重要的情况的例子是当用户环境与用户规则结合以形成动态环境时。(下面会详细描述用户规则)。与用户规则结合的用户环境的特定例子会是用户已经设定了一个规则，该规则声明“我正在会见的人总是可以向

我发送通知无论我是否处在繁忙状态”。在这种情况下，“当用户在开会”的用户环境可以根据用户正在与谁会见而进一步被评估。在该例中，处理会议的程序可以将其注册为动态的环境，而且当评估通知时，相对于该环境而评估发送该通知的人（由于参见会议的人会随时间而变化，因此这不会被前摄地宣布为真或为假）。换言之，这个特定例子要求按照取决于其它用户环境的用户规则来评估用户的环境。

图 5 是说明用户可以调节环境并创建新规则的例程 240 的流程图。在方框 242 中，确定用户是否希望调节环境。如果用户不希望调节环境，则例程进行到判决框 246，下面将详细描述。如果用户希望调节环境，则例程进行到方框 244，其中用户对环境作出修改。

在一实施例中，可以向用户呈现已经被提供的环境，呈现方式是允许用户或者关闭环境（例如用户不同意程序对环境的评估）、或者根据对通知传递的影响而改变环境。更具体的例子是，用户环境可以包括类似“当任何应用程序正以全屏模式运行时”；“当我在播放音乐或影碟时”；“当我的会议管理器显示我正在开会时”；或者“当我的不在办公室助手被开启时”等等。对于这些中的每一个而言，会允许用户选择指定一个指令，当给定条件为真时，到来的通知应该服从所选的步骤。这些指令会指明例如是否或者怎样在屏幕上绘制通知、以及通知会作出的声音或音量。对于音量而言，用户可以在给定的调节选项指定期望音量的百分比。对于在屏幕上绘制通知的选项而言，可以向用户提供选项：比如根本不绘制通知、或者仅在指定的外部显示器上绘制通知、或者在当前屏幕上绘制通知。为了通知的绘制可以指定不同级别的侵入性。例如，如果用户正在运行最大化的应用程序，使得环境略微受限制，则侵入性设置可以是使通知仍能绘制，但可能仅出现在侧边条内。

返回图 5，在判决框 246 处，确定用户是否希望创建新的用户规则。如果用户不希望创建新的用户规则，则例程继续到判决框 250，下面将详述。如果用户希望创建新的用户规则，则例程继续到方框 248，其中创建新的规则。通常，用户规则规定了应该怎样处理包含指定元素的通知。例如，规则可以规定来自指定人的通知总是应该被立即传递，该规则可应用于全部通知，无论哪个程序始发该通知，只要它来自该指定人。更具体的例子是有，其它用户规则可以指向类似“Bremeton，华盛顿的 MSN 自动话务警示”以及“来自 John Doe 的重要电子邮件”。作为来自 John Doe 的重要电子邮件的用户规则的例子，规则可以规定来自 John Doe、文本中有“紧急”、且标记为“高优先级”的任何电子邮件都应该服从指定的处理条件。处理条件可指定通知应被立即传递，且应该要求用户确认这一点。通常，要求用户确认通

知意味着在通知侵入性的形式中有略微提升的高度,因为通知会留在屏幕上直到用户特别消除它为止。在一实施例中,要求用户确认仅仅可以通过用户规则来设置。举另一个例子,规则也可以指定以指定的音量为通知播放常规声音,以便向用户提供一报警,指示特殊的通知已经到达。如可由用户环境所确定的,可以为在用户的“正常”或“繁忙”条件期间应该怎样处理通知而选择不同的设置。处理指令也包括类似通知的路由选项,比如“把通知从 John Doe 传递到我的寻呼机”。在一实施例中,当评估环境时,最限制性的当前为真的环境是所应用的环境。当评估用户规则时,这意味着特定的通知已经与用户创建的规则相匹配,这种情况下从已经与通知匹配的用户规则应用最侵入性的设置。换言之,在用户规则中,用户指定了某些事物是重要的,该步骤会服从用户的偏好。如果在两个规则间存在冲突,则应用最侵入性的。

在一实施例中,用户规则也可以控制通知从特定通知服务而来的传递。例如,按照通知服务提供通知的操作系统可以为用户提供修改怎样传递通知的方式。例如,指定的通知服务可以提供“西雅图的话务警报”,用户可以编辑该传递,使得当接收到这种通知时,系统应该“显示该通知并且播放声音”。

参照图 5,在判决框 250 处,确定用户是否希望调节已经存在的用户规则中的任何规则。如果用户不希望调节任何规则,则例程终止。如果用户希望调节用户规则,则例程进行到方框 252,其中用户对规则作出修改。

如上参照图 3—5 所设,用户环境和用户规则是由操作系统、程序和用户所设定的。系统按照用户偏好适当地代理并连续排列通知的传递。可以向用户呈现用户环境和用户规则,使得用户可以修改或调节各种环境和规则,或者创建新的规则。这向用户提供了管理怎样处理通知的优先权的集中方式。可以理解,这允许用户有效地管理计算系统中希望向用户发送通知的许多竞争元件。

图 6 是说明用于按照用户环境和用户规则处理通知的例程 300 的流程图。在方框 302 处,操作系统或任意程序调用通知 API。在判决框 304 处,确定是否应该把通知记录到通知历史。如果要记录通知,则例程继续到方框 306,其中把通知记录历史。如果不要记录通知,则例程继续到判决框 310。

在判决框 310 处,确定通知是否与任何用户规则相匹配。如果通知与任何用户规则匹配,则例程继续到方框 312,其中服从用户规则(根据通知内容加上用户环境),例程继续到图 7 中继续到点 A。如果在判决框 310 中通知不与任何用户规则匹配,则例程继续到判决框 320。

在一实施例中，用户规则总是比当前的用户环境重要。如上所述，用户规则可以基于通知的任何元素。例如，可以向所有通知应用基于对开始该通知的人的评估的规则，无论哪个程序开始该通知，只要它是来自该规则所基于的人（例如“John Doe”总是可以联系到我）。此外，即使在会使通知不绘制的环境期间，通知也可以绘制在屏幕上（例如“与我会见的人总是可以向我发送通知”，即使用户环境一般声明该用户不会在会议期间接收通知）。

参照图 6，在判决框 320 处，确定当前时间是否绘制通知（仅根据用户环境）。如果可以在当前时间绘制通知，则例程继续到方框 322，其中绘制通知，并且提供适当的声音和音量。如果不适于在当前时间绘制通知，则例程继续到判决框 330。

在判决框 330 处，确定通知是否到期。如果通知已到期，则例程继续到方框 332，其中通知被破坏。如果通知未到期，则例程继续到方框 334，其中通知被推迟，且例程继续到图 7 中继续到点 B。

图 7 是说明用按照指定的用户规则处理通知的例程 350 的流程图。该例程从来自图 6 的点 A 继续，如上所述。如图 7 所述，在判决框 352 处，确定通知是否应被路由。如果通知不要被路由，则例程继续到判决框 362，下面会详述。如果通知要被路由，则例程继续到方框 354，其中通知根据指定被路由。当通知被路由时，这指示通知包含与用户规则相匹配的元素，这要求通知被切换到另一系统。这可以在用户繁忙时发生，或者在与用户规则内指定的标准相匹配的任何通知时发生，无论用户是否可用。例如，其中有单词“紧急”的通知总是会被转发到用户的寻呼机，而其它通知可能仅根据用户规则和环境组合而被路由。

某些路由指令的例子包括：“把该通知转发到邮件地址”；“把该通知转发到另一台 PC”；“把该通知转发到寻呼机”；“把该通知转发到蜂窝电话”；或者“把该通知转发到电子邮件服务器”。下面将详述，如果通知被路由，它也可以被传递并且在屏幕上绘制。此外，通知所转发到的设备可能实现这个相同的环境系统，在该设备上会有用户环境的其它或不同的知识，该设备上的环境系统会选择对通知作出不同的行动。

参照图 7，在判决框 362 处，确定是否要拒绝该通知。如果通知不被拒绝，则例程继续到判决框 366，下面将详述。如果通知要被拒绝，则例程继续到方框 364，其中通知被破坏并且不被用户所见。换言之，不允许被拒绝的通知绘制或发出噪声。例如，这会根据一用户规则而发生，该用户规则声明当通知到期时应该拒绝特定的通知，或如上面参照图 6 的方框 332 所述。

参照图 7, 在判决框 366 处, 确定通知是否应被推迟。如果通知不会被推迟, 则例程继续到判决框 370, 如下详述。如果通知要被推迟, 则例程继续到方框 368, 其中保留通知直到用户环境变化为止, 例程继续到在图 8 中继续的点 B。通常, 推迟通知指明会允许该通知传递, 但用户当前的环境或规则看上去不适于此时传递该通知。下面将参照图 8 详述, 一旦用户环境变化或者当用户规则指示随后适于传递时, 通知会被传递到用户的屏幕并且允许它绘制并/或发出声音, 这由用户规则或用户环境所规定。

参照图 7, 在判决框 370 处, 确定通知是否应被传递。如果通知不被传递, 则例程结束。如果通知要被传递, 则例程继续到方框 372, 其中按照适当的侵入性级别绘制通知, 并且提供适当的声音和音量。换言之, 允许传递通知, 尽管它是按照用户的环境和规则来传递的(例如可以允许绘制通知但要求静音)。

图 8 是说明用于推迟通知的传递的例程 380 的流程图。该例程从图 6 或 7 的点 B 继续, 如上所述。如图 8 所述, 在方框 382 处, 保留通知。在方框 384 处, 系统监视对为真或为假的所宣布环境的变化, 或者监视规定现在适于传递通知的用户规则。在判决框 386 处, 确定用户环境是否已变化, 或者用户规则规定现在适于传递通知。如果用户环境未变化或者如果没有用户规则另外规定, 则例程返回方框 382, 其中继续保留通知。如果用户环境已变化或者如果用户规则规定现在可能适于传递通知, 则例程继续到在图 6 中继续的点 C。图 6 中的点 C 返回到判决框 304, 其中用于评估通知的过程重新开始。

图 9 是说明用于按照各种限制确定通知的绘制的例程 400 的流程图。可以理解, 该例程可以作为通知处理的一部分来实现, 比如在图 6 的方框 322 处或者在图 7 的方框 372 处。通常, 当通知进入系统时, 评估当前为真的所有环境, 按照用户的当前状态应用通知最具限制性的设置。如图 9 所述, 在判决框 402 处, 确定是否根本不应绘制通知。如果根本不应绘制通知, 则例程继续到方框 404, 其中设置通知不绘制在任何显示器上。如果要绘制通知, 则例程继续到判决框 406。

在判决框 406 处, 确定通知是否应被绘制但是仅在外部绘制。如果通知仅在外部绘制, 则例程继续到方框 408, 其中绘制通知但是仅在外部硬件显示器上绘制。如果不在外部硬件显示器上绘制通知, 则例程继续到判决框 410。

在判决框 410 处, 确定是否应在当前显示器上绘制通知。如果要在当前显示器上绘制通知, 则例程继续到方框 412, 其中在当前显示器上按照适当的侵入性级别绘制通知。如果通知不被绘制在当前显示器上, 则例程结束。

图 10 是说明按照各种限制, 确定要为通知的声音播放的音量的例程 420 的流程图。如上面参照图 9 所述, 可以理解, 该例程可以作为通知处理的一部分来实现, 比如在图 6 的方框 322 处或者在图 7 的方框 372 处。当通知进入系统时, 评估当前为真的所有环境, 并且按照用户的当前状态对通知应用最具限制性的设置。如图 10 所述, 在判决框 422 处, 确定通知是否应被静音。如果通知要被静音, 则例程继续到方框 424, 其中不为通知提供任何音量。如果通知不要被静音, 则例程继续到判决框 426。

在判决框 426 处, 确定是否应为通知提供某些百分比的音量但低于满音量。如果要提供某些百分比的音量, 则例程继续到方框 428, 其中以指定的百分比音量播放通知。如果不要提供指定的百分比音量, 则例程继续到判决框 430。

在判决框 430 处, 确定是否应为通知提供满音量。如果要提供满音量, 则例程继续到方框 432, 其中以满音量级播放通知。如果不提供满音量, 则例程结束。在一实施例中, 除了为通知提供不同音量级别以外, 也可以按照用户环境和规则为通知选择不同的声音。

可以理解, 参照图 1-10 所述的用户环境系统控制来自各个源的通知传递, 使得通知彼此间不再冲突, 因为系统适当地代理并且连续排列它们在屏幕上的呈现。此外, 由用户环境系统所处理的通知可以被认为更有价值, 因为它们是在用户更能接收他们的时候被传递的, 此外, 公共规则的使用帮助用户删除了不期望的通知。

图 11-14 是指测试通知的评估。下面将详述, 按照本发明的一方面, 可以由任何程序使用测试通知来获得有关用户环境的当前状态的信息。这方面的一个优点是: 通过亲自解释什么通知应该看似如何或者通知应该被传递的方式, 用户环境信息可由任何程序获得, 无论程序是否要使用系统内已经内建的服务, 或者无论程序是否要扩展它。换言之, 将来的程序具有更高级的通知, 它们不被设计成被系统向它们提供的呈现所限制, 这些程序仍将能使用测试通知来获得有关用户当前环境的信息。这种更高级的通知可能随着通知内容继续增长和变化而发生, 通知的新用户界面继续发展。

例如, 将来的用户界面可以提供仅在用户不“繁忙”时绘制的丰富的全屏动画。例如, 当 CD-ROM 加速旋转时(由于技术约束条件, 从首次插入 CD 到可以读取 CD 之间会有一段时间, 即使知道 CD 在驱动器内——在这段时间内会使用动画向用户显示系统知道该 CD, 但只是还不能从中读取), 把 CD 放入 CD-ROM 驱动器会在屏

幕上给出 CD 的动画。通过使用本发明的测试通知，动画程序将能够得知用户的当前环境，并且如果用户现在未准备好接收通知，则选择不显示在屏幕上。

举另一个例子，将来的即时消息传递程序可以为通知研发新的用户界面，这不能用当前的通知引擎来实现。需要这种新用户界面的发展作为在当前竞争性市场中所需灵活性的一部分。按照本发明，即时消息传递程序会继续使用测试通知来确定它是否应该按照用户的当前环境显示或不显示其更高级的通知。

按照本发明另一方面，可以使用测试通知来防止产生不希望的通知。这方面可应用于试图向系统发送通知的任何程序。换言之，通过使系统能更多地知道用户的环境，可以防止由程序产生不希望的通知，因此前摄地终止这类通知的生成，知道用户处在能接收的状态为止。下面的例子帮助说明了本发明的这方面。

举一个例子，即时消息传递程序可以提供一个人列表。测试通知能够按照每个联系人接入环境系统(例如“如果 Tom 现在正向你发送即时消息，会显示么？”以及“如果 Chris 现在正向你发送即时消息，会显示么？”)。根据该信息，即时消息传递程序可以开始向单个联系人广播明确的繁忙或空闲状态。可以使用该技术来优先停止产生不希望的通知，而不是仅在接收到它们时才抑制它们。

举另一个例子，如果用户繁忙，邮件程序会利用这点向发送者提供自动回复(或者根据用户已提供的规则向所有发送者提供，比如“我的直接报告”或“我的管理器”)。自动回复可以指示“我现在很忙，但有机会会回复”。通常，总体的系统通信可以通过向任意程序呈现用户环境来改进。

如上所述，按照本发明，应用程序能够构造一测试通知并且特别接收回是否会在当前时间在屏幕上绘制实际通知。如上所述，这允许程序即使在发展了通知的新用户界面以后仍继续使用用户环境系统。此外，通过为其它程序启用这些新的较丰富的场景，根据能增加对有关用户行为和偏好的信息的访问，使用该系统的所有程序可以被认为较丰富且较智能。

图 11 是说明按照本发明用于处理测试通知的一般例程 500 的流程图。该例程类似于图 2 用于处理实际通知的例程。如图 11 所述，在方框 502 处，调用测试通知 API。在方框 504 处，相对于操作系统及任意程序所设定的、并且由用户进一步认可或修改的用户环境、以及相对于用户所设定的用户规则，而评估测试通知的元素。在方框 506 处，按照测试通知的评估结果提供有关应该怎样处理测试通知的指示。然后把该指示返回到调用应用程序。

在一实施例中，当操作系统或任意程序决定它需要理解用户目前有多繁忙时，

调用测试通知 API。这会在何时发生的一个例子是存在一个是否要在屏幕上绘制通知的判决点。另一个例子是使用该数据来通知程序希望代表用户采取的行为。

当调用测试通知 API 时，如果调用程序正在使用用户环境系统的通知方法来发送实际通知，调用程序就构造一接近于它会发送的通知，然后使用任选方法来进行测试（这返回结果并且还确保这个特定的通知不会被显示在屏幕上）。这个过程的一个例子是希望根据当前用户的环境向各个联系人广播适当的空闲或繁忙状态的即时消息传递程序。即时消息传递程序会为每个联系人创建一个测试通知，并且根据返回值按每个联系人逐一广播不同的空闲或繁忙状态。另一个例子将是程序希望根据用户环境显示动画（例如上述的 CD-ROM 动画示例）。希望显示动画的编码会构造一通知（在该情况下，内容只是一个具有图像或动画序列的简单通知，因为这只是有关给定的通知是否会绘制的测试），然后使用测试方法，接着会使用返回结果作为当前是否应播放动画的向导。在一实施例中，呼叫代码一般至少会把可能的最基本通知提升为测试通知。如果存在可用的较丰富数据（比如来自联系人列表的联系人），则包括该信息会使测试通知更为准确，因为用户可能会根据用户的不同而有定制的用户规则，这会影响所返回的结果。

可以为通知测试 API 使用的一种实现是轮询实现。在上述即时消息传递程序示例中，对于轮询实现而言，即时消息传递程序会周期性地反复轮询通知测试 API 以确定怎样改变广播数据。可以为通知测试 API 使用的另一种实现是预订回叫实现。在这种实现中，即时消息传递程序会“预订”环境变化。于是，当用户环境以即时消息传递程序能够广播的方式变化时，环境引擎可以根据适当的更新回叫即时消息传递程序，而不是周期性地反复轮询。这在某些情况下是有利的，因为在环境变化和广播状态之间没有时滞（而根据轮询实现，会有广播状态与当前用户环境不匹配的时刻）。对于其它情况而言，轮询实现可能更为适当（因为它们是对一次性事件的响应，例如当把 CD 插入 CD-ROM 时）。

图 12 是说明例程 520 的流程图，该例程 520 用于处理一测试通知，并且返回对于是否会在当前事件绘制测试通知的为真或为假的指示。在方框 522 处，调用通知测试 API。在判决框 530 处，确定测试通知是否与任何用户规则相匹配。如果测试通知不与任何用户规则相匹配，则例程继续到判决框 550，下面将详述。如果测试通知与任何用户规则匹配，则例程继续到判决框 540。

在判决框 540 处，确定用户规则是否指示会在当前时间绘制测试通知。如果会在当前时间绘制测试通知，则例程继续到方框 542，其中提供的指示为真。如果

不会在当前时间绘制测试通知，则例程继续到方框 544，其中提供的指示为假。

在判决框 550 处，确定测试通知是否能在当前时间绘制(仅关于用户环境)。如果测试通知能够在当前时间绘制，则例程继续到方框 552，其中提供的指示为真。如果通知不能在当前时间绘制，则例程继续到方框 554，其中提供的指示为假。例程根据指定的指示从方框 542、544、552 和 554 返回调用应用程序。

图 13 是说明用于处理通知并返回详细指示的例程 600 的流程图。如上所述，图 12 的例程 520 仅提供了指示为真或为假的返回值，关于是否在当前时间绘制通知。如下详述，图 13 的例程 600 返回了较丰富的返回值(例如通知不会在当前绘制，但只要用户环境变化它就会绘制，或者它会路由到另一设备，等等)。这提供了调用编码中较丰富的逻辑。这允许程序中的高级功能，它们能使用这种较丰富的返回值。

还应该注意，虽然返回值被描述为功能调用的一部分，在另一实施例中该数据可以作为回叫的一部分被传递。换言之，调用应用程序可以建立对通知的“预订”，使得当用户的环境随后变化时(这会影响通知从调用应用程序而来的传递)会通知调用应用程序。这不要轮询，因此在某些情况下对于系统的稳健性和性能较为有利。

如图 13 所述，在方框 602 处，调用通知测试 API 或者注册预订(关于轮询相对于上述的实施例)。在判决框 610 处，确定测试通知是否与任何用户规则相匹配。如果测试通知不与任何用户规则相匹配，则例程继续到判决框 620，下面将详述。如果测试通知与任何用户规则相匹配，则例程继续到方框 612。在方框 612 处，按照用户规则评估测试通知(根据测试通知内容加上用户环境)，且例程继续到在图 14 中继续的点 D，下面将详述。

在判决框 620 处，确定测试通知是否能在当前时间绘制(仅根据用户环境)。如果测试通知不能在当前时间绘制，则例程继续到判决框 630，下面将详述。如果测试通知能够在当前时间绘制，则例程继续到方框 622。在方框 622 处，例程按照用户环境确定什么声级会是适当的。在方框 624 处，提供一指示，指明通知会绘制，并且还包括会适用于该通知的百分比声级。

在判决框 630 处，确定是否会保留测试通知以留待稍后的传递(根据测试通知内容加上用户规则)。如果会保留测试通知留待以后传递，则例程继续到方框 632，其中提供了推迟的指示。如果不会保留测试通知留待稍后的传递，则例程继续到方框 634，其中提供了拒绝的指示。例程根据指定的指示从方框 624、632 和 634 返

回调应用程序。

图 14 是用于按照用户规则评估测试通知的例程 650 的流程图。例程从图 13 的点 D 继续。如图 14 所述，在判决框 652 处，确定测试通知是否应被路由。如果测试通知要被路由，则例程继续到方框 654，其中提供路由的指示，且例程继续到判决框 662。如果测试通知不要被路由，例程也继续到判决框 662。

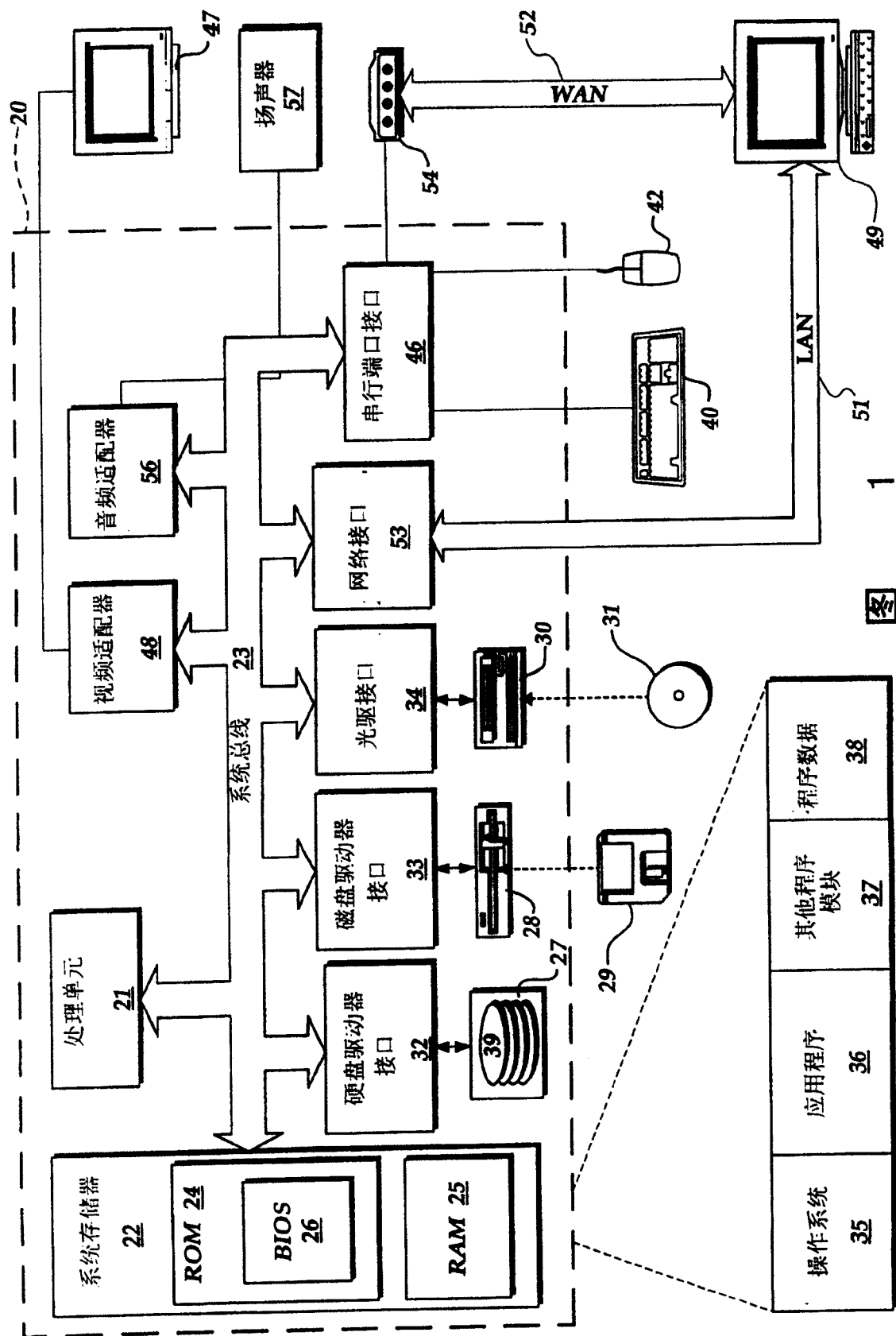
在判决框 662 处，确定测试通知是否会被拒绝。如果测试通知会被拒绝，则例程继续到方框 664，其中提供拒绝的指示。如果测试通知不会被拒绝，则例程继续到判决框 666。

在判决框 666 处，确定测试通知是否会被推迟。如果测试通知会被推迟，则例程继续到方框 668，其中提供推迟的指示。如果测试通知不会被推迟，则例程继续到判决框 670。

在判决框 670 处，确定测试通知是否会被传递。如果测试通知会被传递，则例程继续到方框 672，其中提供传递的指示。在一实施例中，传递指示还可以包括指定的侵入性指示以及声音和音量指示。如果测试通知不会被传递，则例程返回调用应用程序。例程根据指定的指示从方框 664、668 和 672 返回调用应用程序。

可以理解，上述关于图 1—14 描述的本发明提供了一种使用测试通知的系统和方法，它使程序能获得与用户的可用性有关的指示。通过使程序能更多地知道用户的环境，可以防止在源头处产生不希望的通知，因此允许仅当用户处在可接受的状态下时产生通知。此外，程序能够使用测试通知来确定用户的环境，即使程序一般为其自身的通知使用不同的用户界面。这些方面允许在用户环境系统的潜在使用中更大的灵活性。这些方面还允许其它程序更丰富的场景，使得系统总体根据用户行为和偏好变得较丰富和较智能。

虽然已经说明并描述了本发明的优选实施例，然而可以理解，这里可以作出各种变化而不背离本发明的精神和范围。



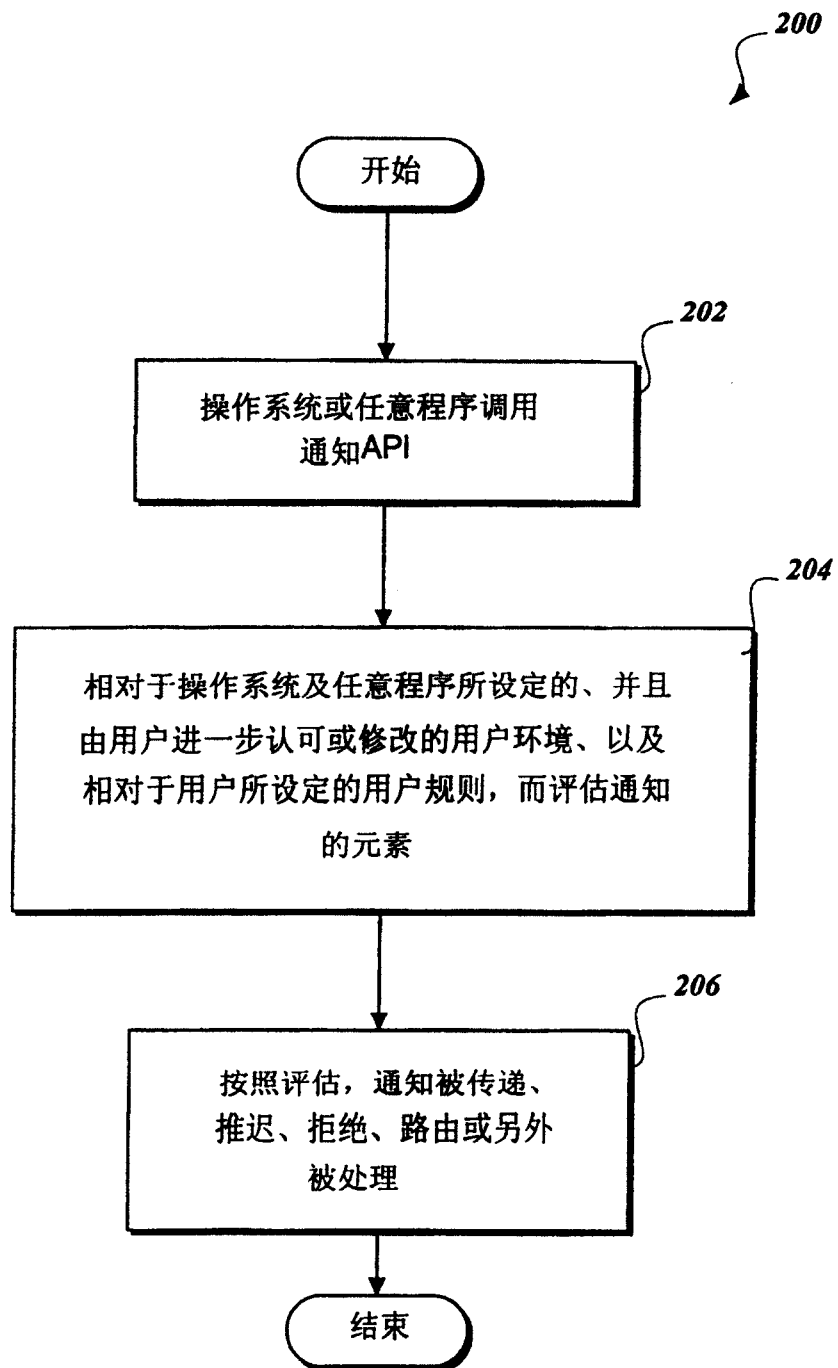


图 2

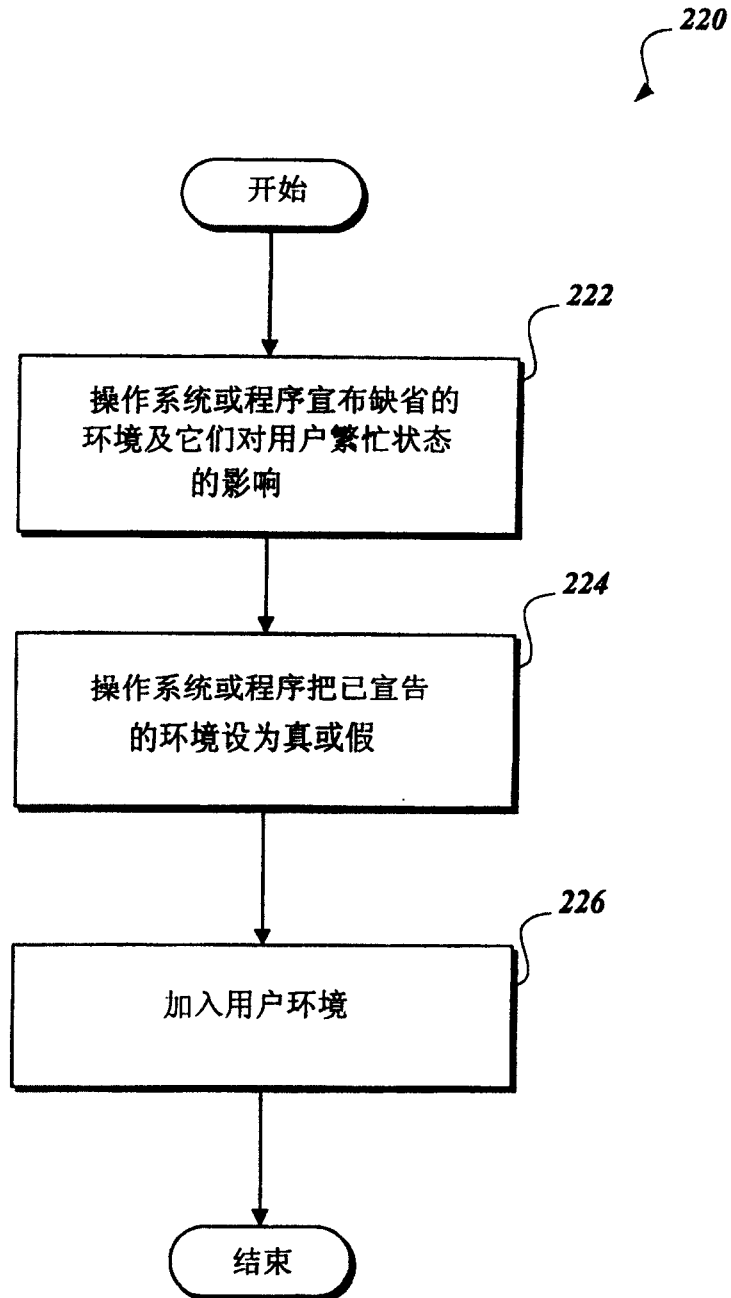


图 3

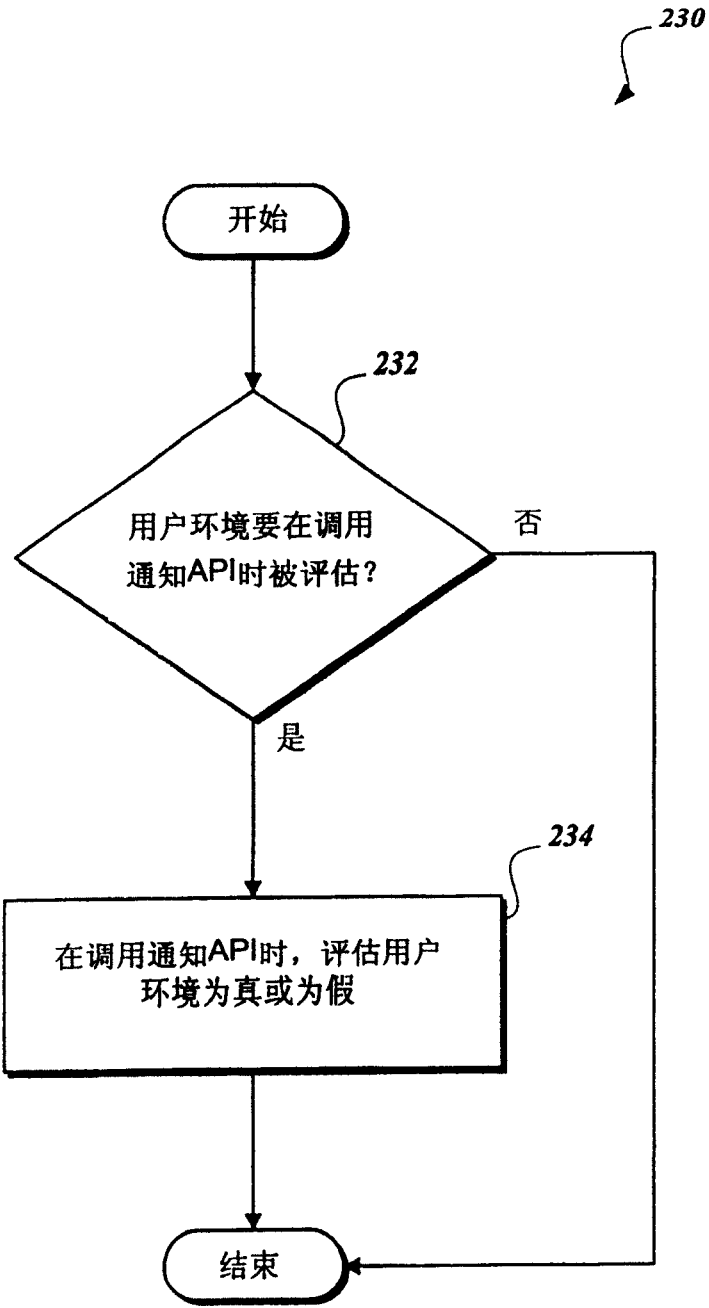


图 4

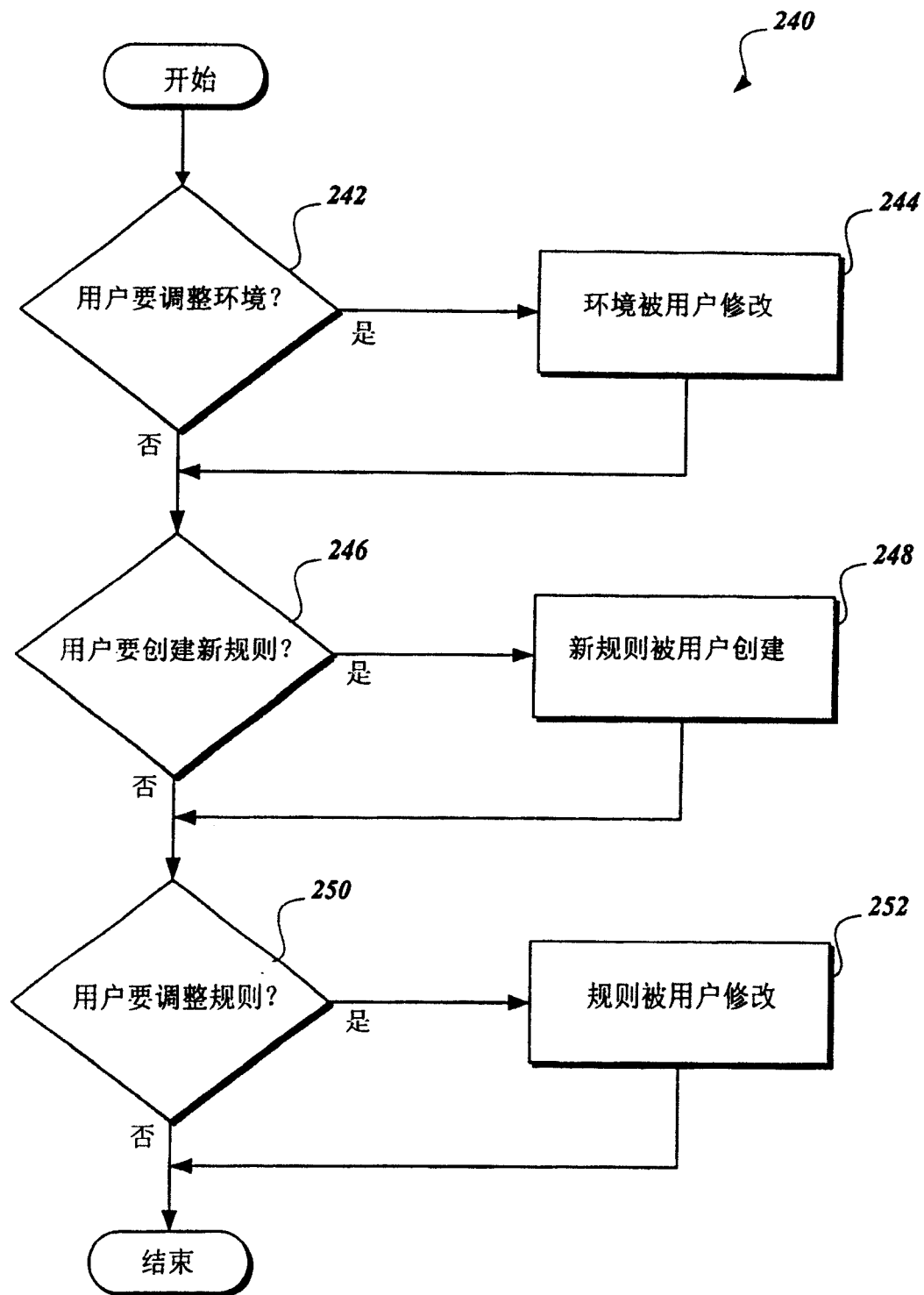


图 5

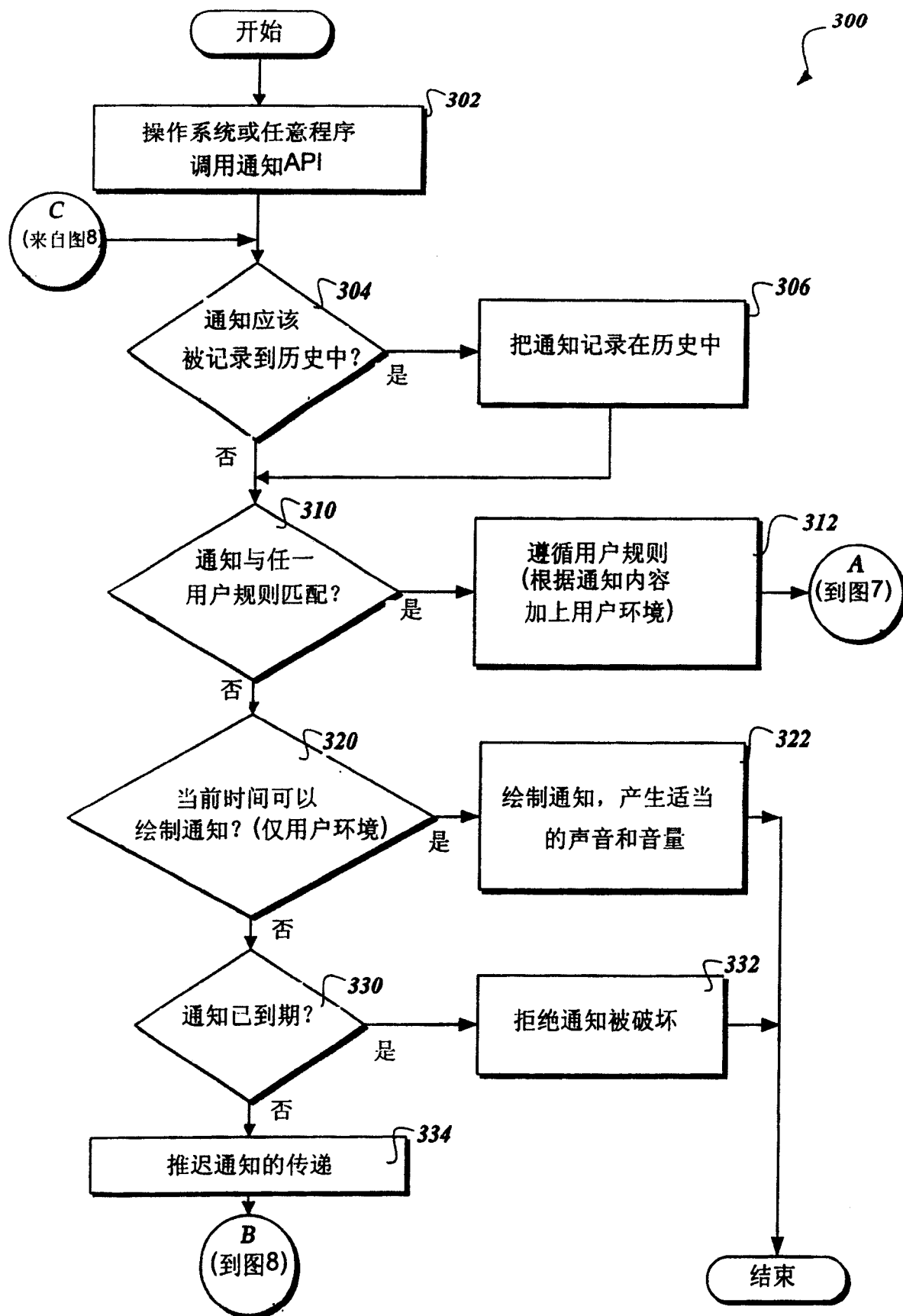


图 6

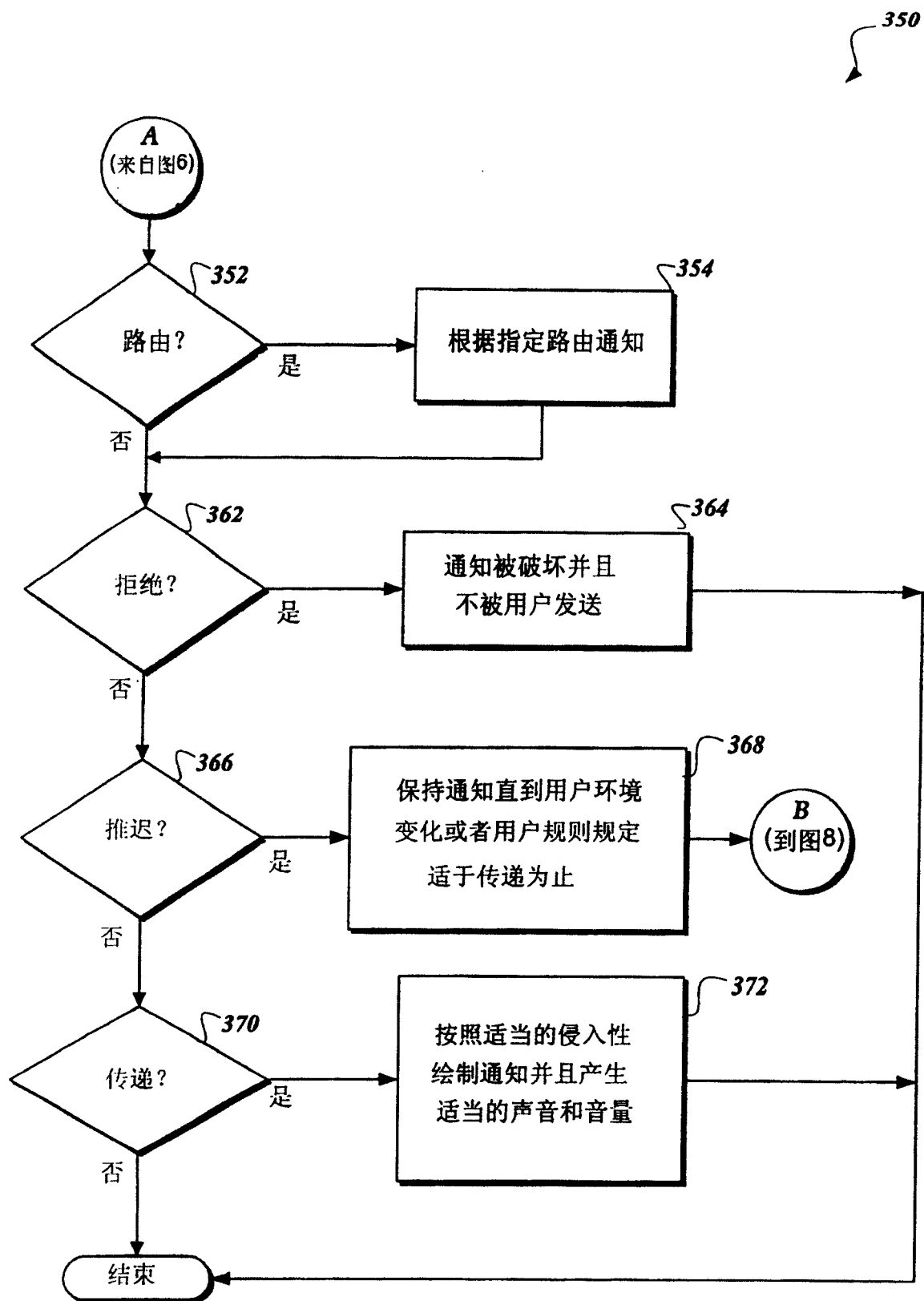


图 7

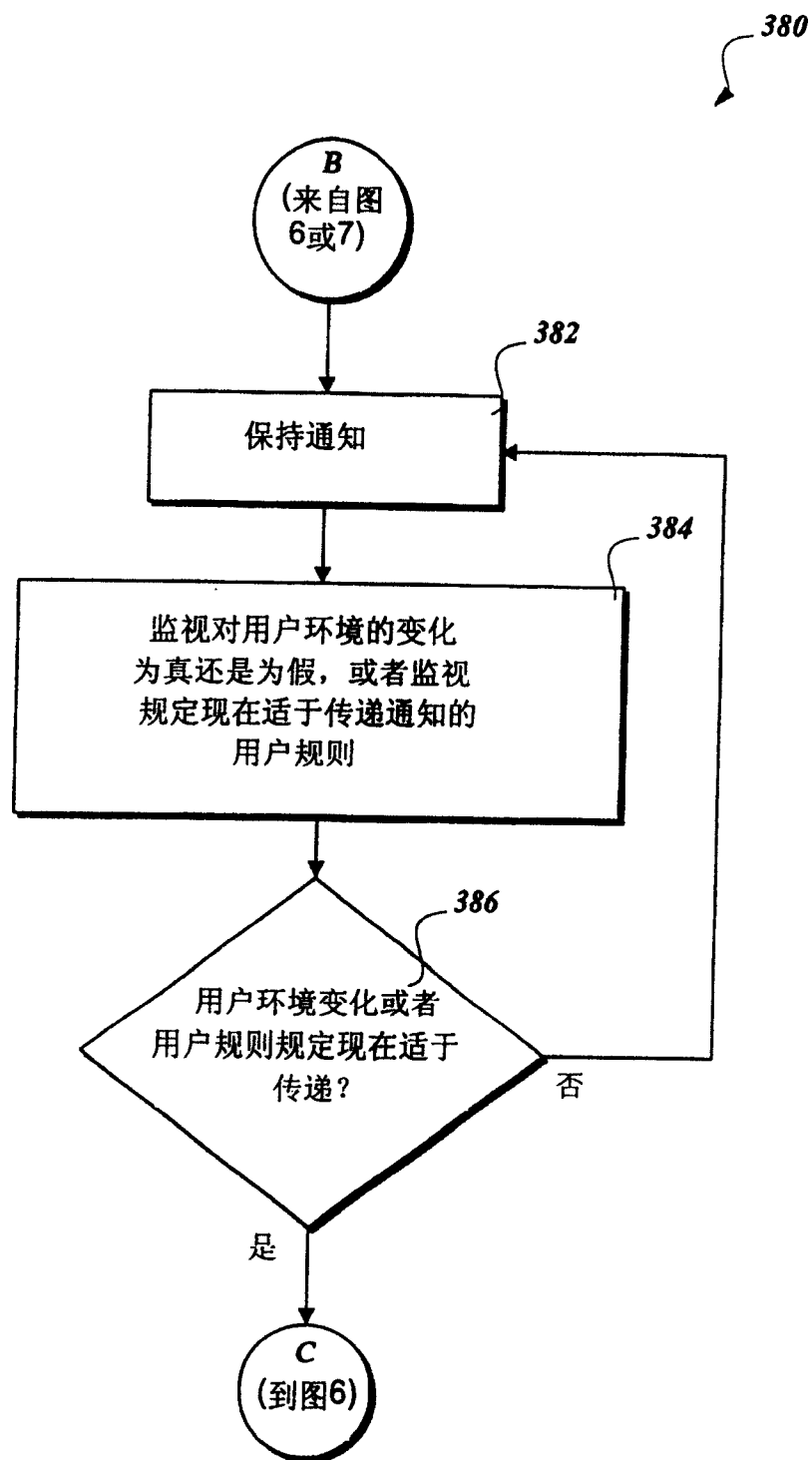


图 8

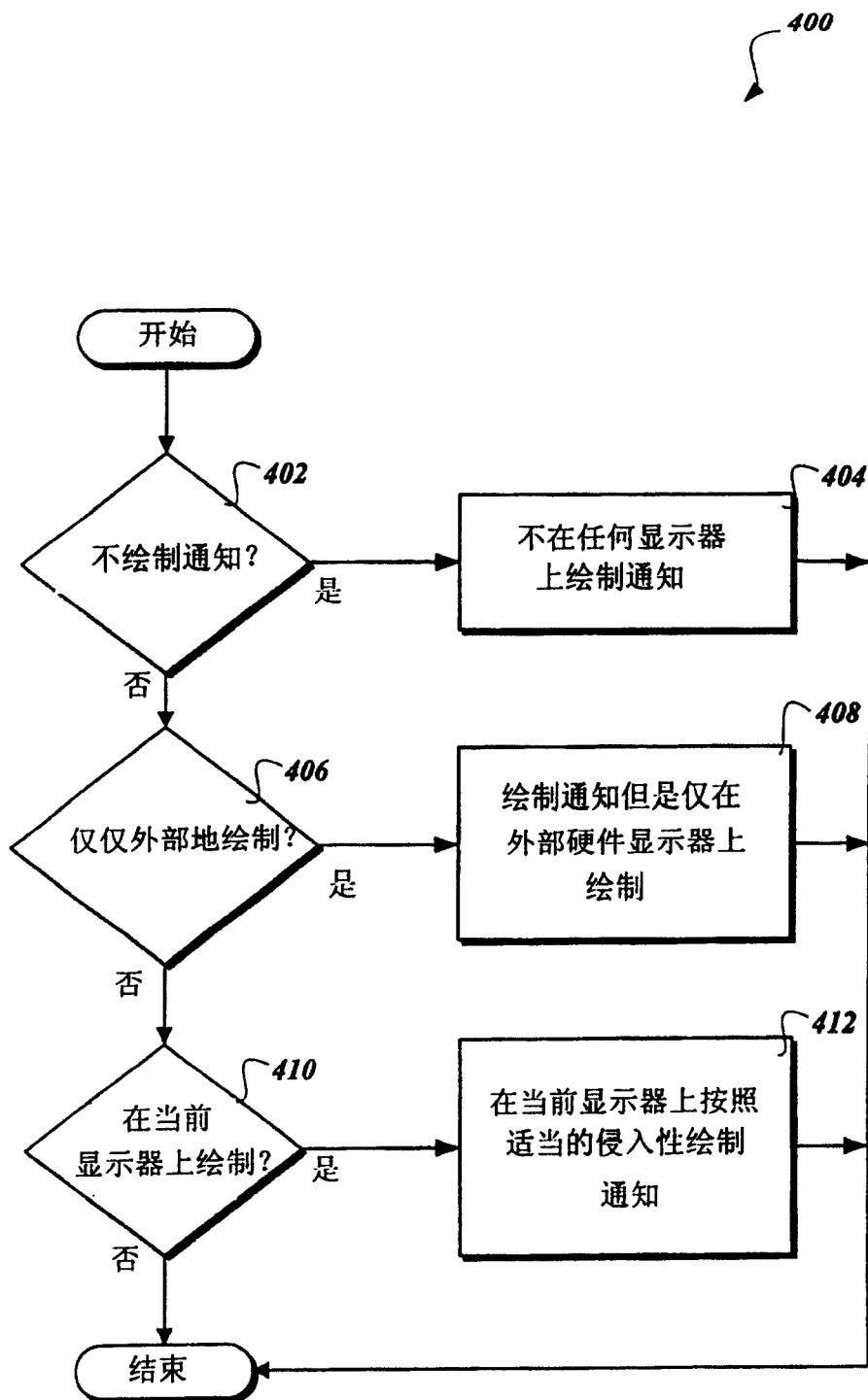


图 9

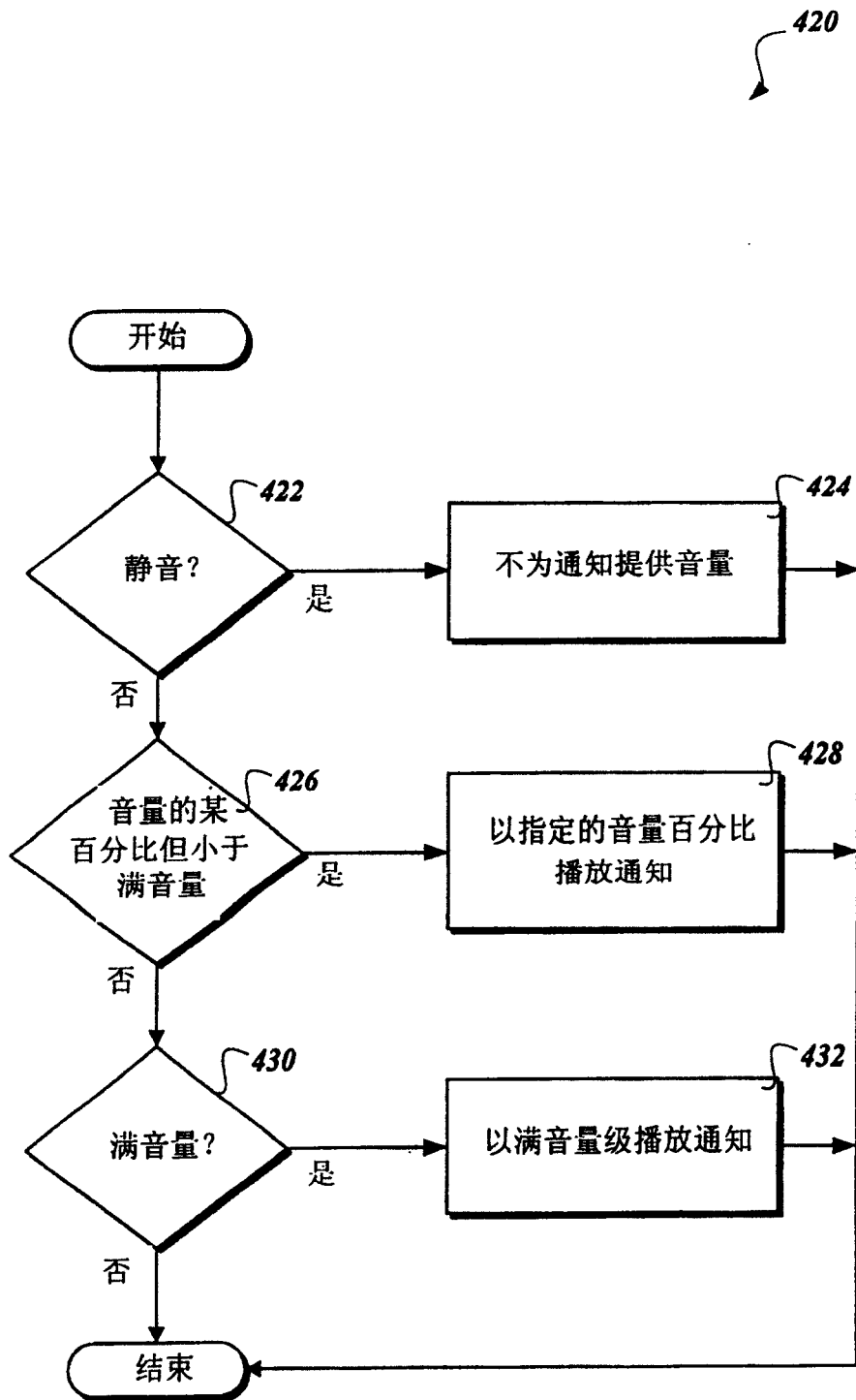


图 10

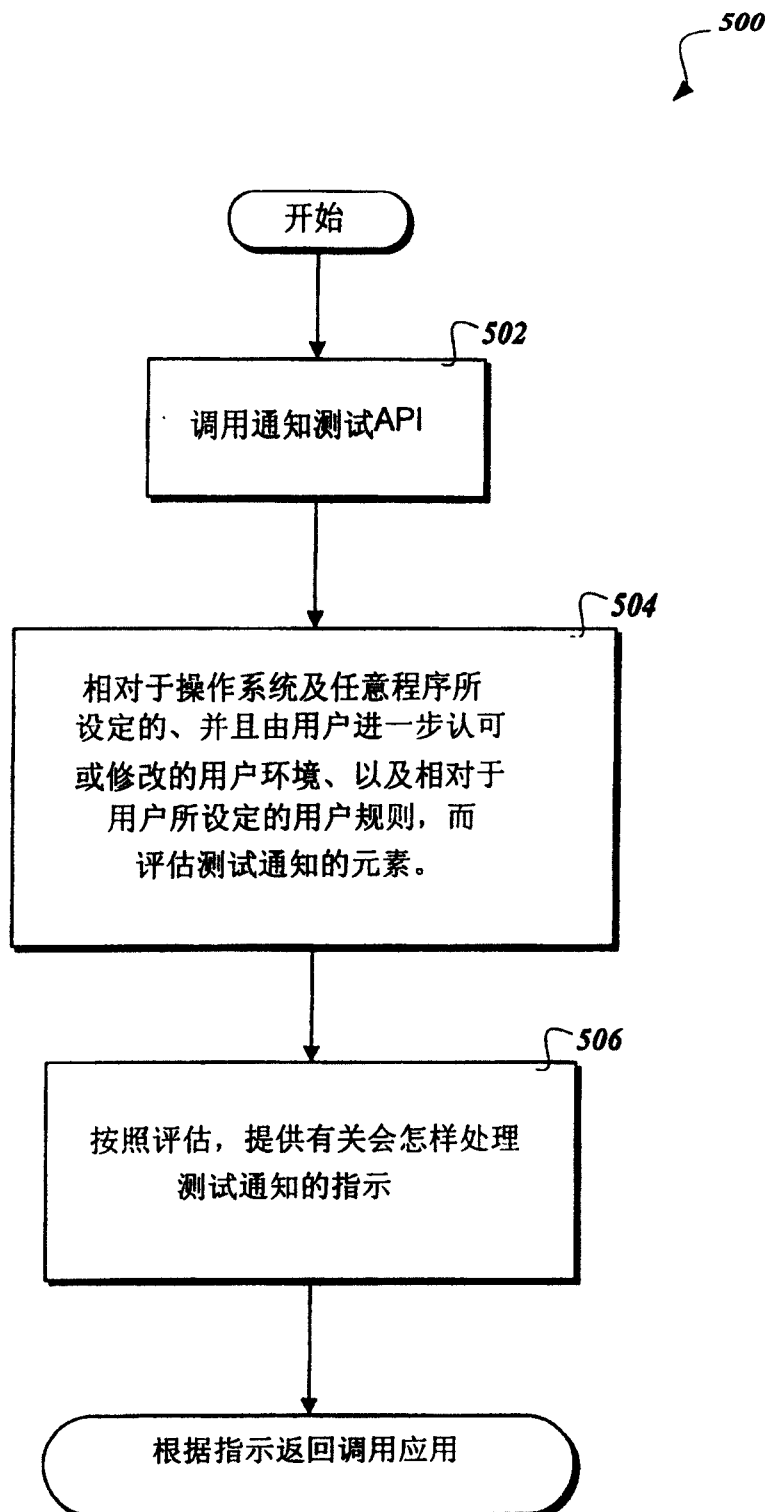


图 11

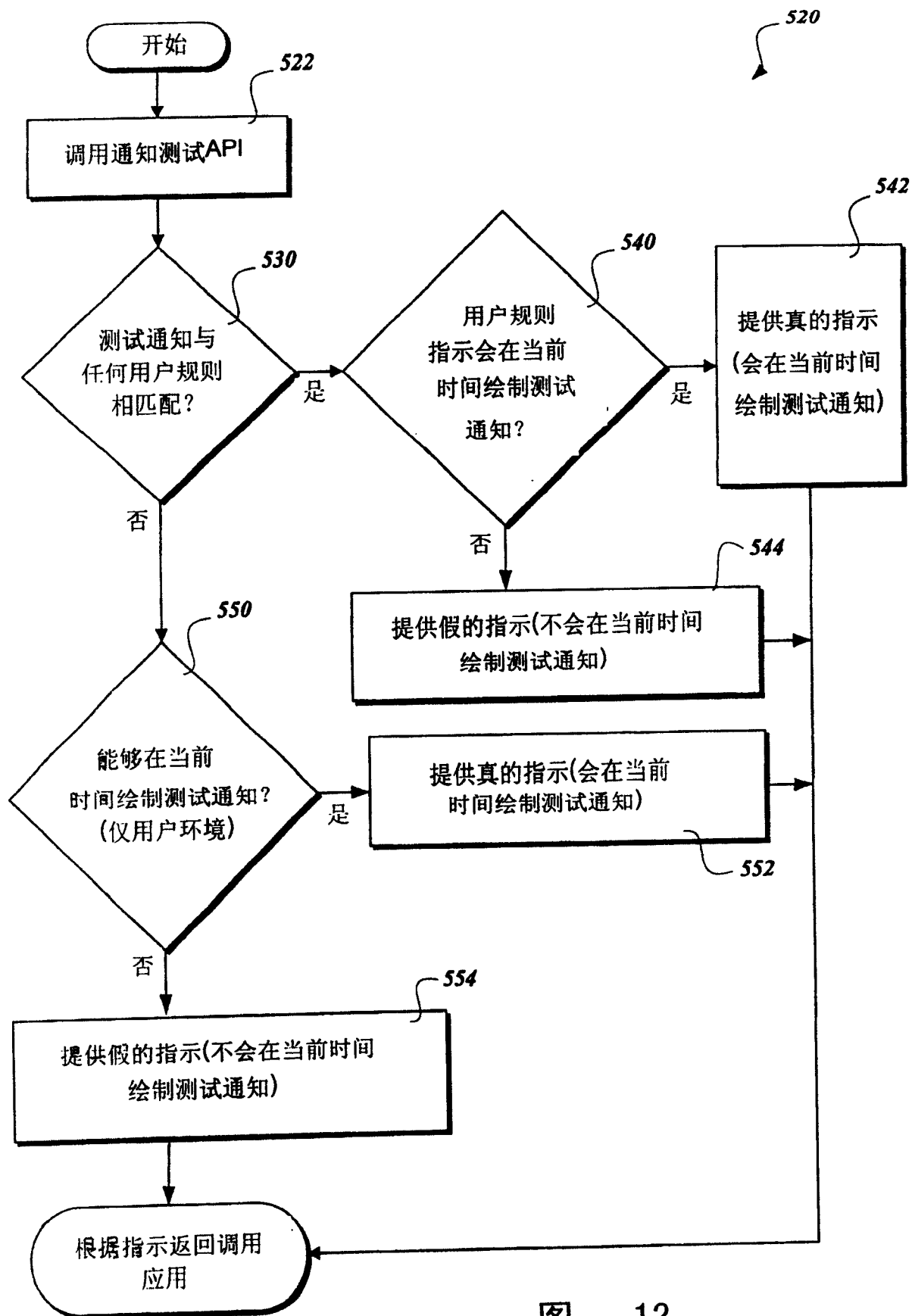


图 12

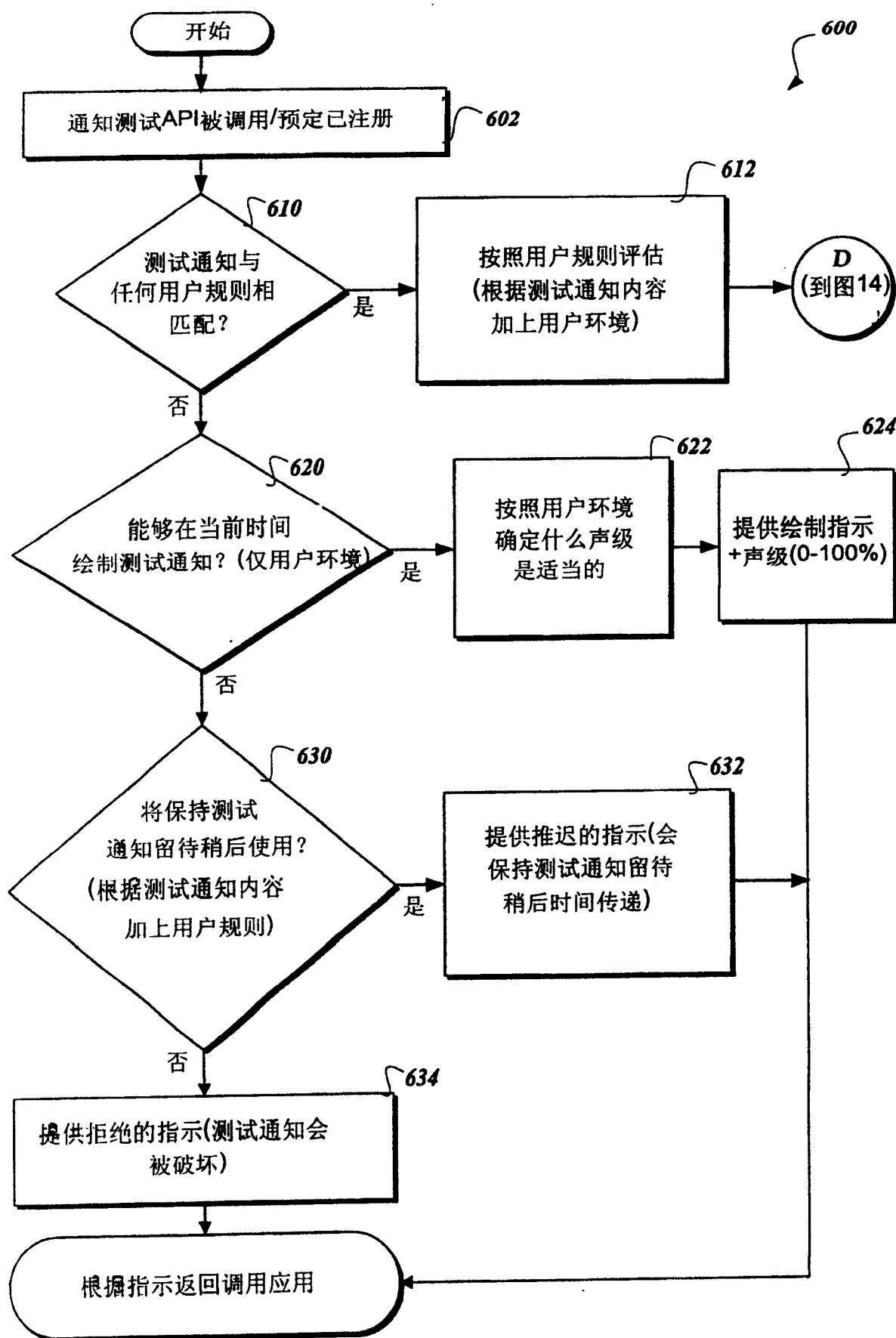


图 13

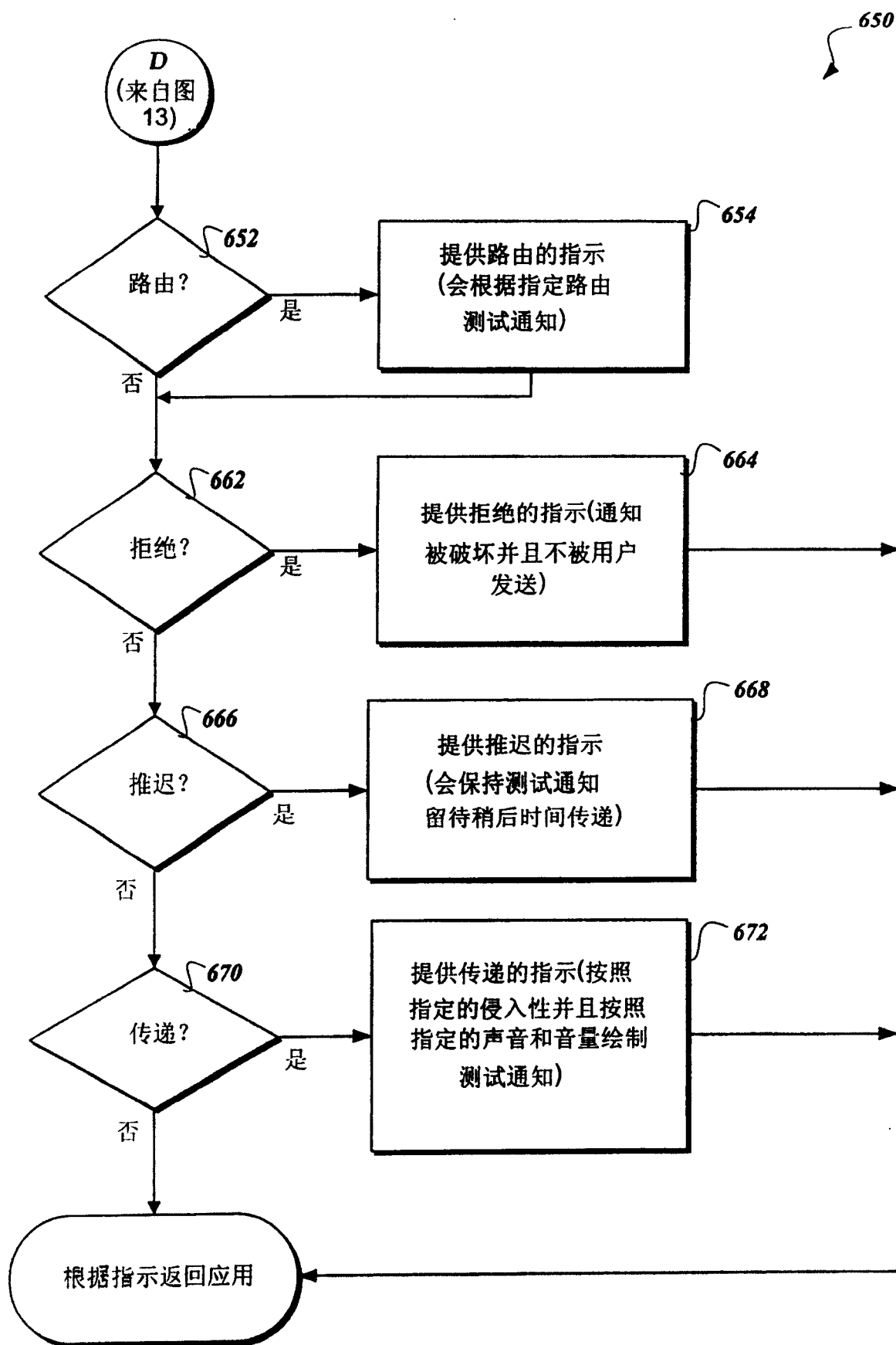


图 14