



(12)发明专利申请

(10)申请公布号 CN 110062926 A

(43)申请公布日 2019.07.26

(21)申请号 201780075697.0

V·帕布雷加 李俊操

(22)申请日 2017.12.04

(74)专利代理机构 北京市金杜律师事务所

11256

(30)优先权数据

代理人 王茂华

15/374,780 2016.12.09 US

(85)PCT国际申请进入国家阶段日

(51)Int.Cl.

2019.06.06

G06F 11/34(2006.01)

(86)PCT国际申请的申请数据

G06F 11/36(2006.01)

PCT/US2017/064393 2017.12.04

G06F 11/30(2006.01)

(87)PCT国际申请的公布数据

W02018/106555 EN 2018.06.14

(71)申请人 微软技术许可有限责任公司

地址 美国华盛顿州

(72)发明人 M·W·米尔克 J·F·利克滕贝格

V·A·勒文 R·L·德维杜维

权赫俊 N·L·戴欣格

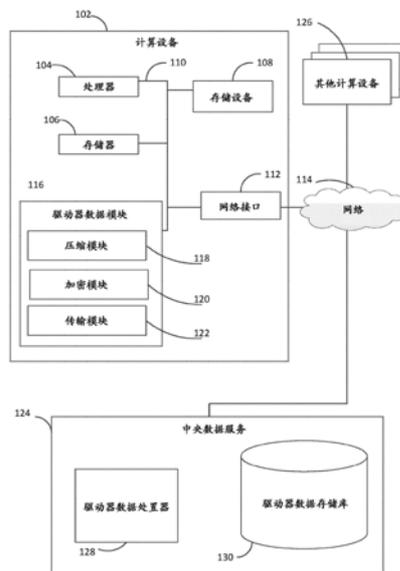
权利要求书2页 说明书16页 附图8页

(54)发明名称

设备驱动器遥测

(57)摘要

描述了用于验证计算设备的功能性的系统和方法。接收规则,该规则可用于配置驱动器验证器功能以捕获与由该规则标识的设备驱动器相关联的信息。经配置的驱动器验证器功能在计算设备上运行。响应于由规则标识的驱动器状况而捕获信息。当驱动器状况包括经标识的设备驱动器的错误状况时,允许计算设备继续操作。发起通信以将所捕获的信息传送到驱动器验证分析服务。



1. 一种用于验证计算设备的功能性的计算机实施的方法,所述方法包括:  
接收一个或多个规则,所述一个或多个规则可用于配置驱动器验证器功能,以捕获与由所述一个或多个规则标识的设备驱动器相关联的信息;  
在计算设备上配置所述驱动器验证器功能,以根据所述一个或多个规则捕获与所标识的所述设备驱动器相关联的所述信息;  
在所述计算设备上执行经配置的所述驱动器验证器功能;  
响应于由所述一个或多个规则标识的一个或多个驱动器状况,使用经配置的所述驱动器验证器功能来捕获所述信息;  
在所述一个或多个驱动器状况包括如由所述一个或多个规则限定的经标识的所述设备驱动器的错误状况的情况下,允许所述计算设备继续操作,并且否则对经标识的所述设备驱动器执行错误恢复响应;以及  
发起通信以将所捕获的所述信息传送到驱动器验证分析服务。
2. 根据权利要求1所述的方法,其中从驱动器验证遥测管理器接收所述一个或多个规则。
3. 根据权利要求1所述的方法,其中所述驱动器验证器功能可操作以在数据捕获模式或记录模式下运行。
4. 根据权利要求1所述的方法,其中所述驱动器验证器功能在预定时间之后被自动禁用。
5. 根据权利要求1所述的方法,其中所述一个或多个规则将所述驱动器验证器功能配置成以指定类别的驱动器为目标。
6. 根据权利要求5所述的方法,其中所述指定类别的驱动器包括音频、NDIS和相机驱动器中的一种或多种。
7. 根据权利要求1所述的方法,其中所述捕获包括检测对所述规则中的一个规则的违反,并且作为响应,记录事件并存储相关联的驱动器数据。
8. 根据权利要求1所述的方法,其中所述一个或多个规则实施一般驱动器检查和基于规则的检查。
9. 根据权利要求8所述的方法,其中一般检查包括I/O验证和存储器管理中的一个或多个。
10. 根据权利要求1所述的方法,其中一次仅启用所述规则中的一个规则。
11. 根据权利要求1所述的方法,其中所述驱动器验证器功能可操作以在学习模式下运行,其中驱动器问题被收集并且基线简档被创建而不是生成系统错误。
12. 一种被配置成验证计算设备的功能性的系统,所述系统包括:  
至少一个存储器,其中存储有计算机指令,所述计算机指令在由所述系统的一个或多个处理器执行时至少使所述系统:  
执行驱动器验证器功能,所述驱动器验证器功能被配置成捕获与由一个或多个规则标识的设备驱动器相关联的信息;  
响应于由所述一个或多个规则标识的一个或多个驱动器状况,使用经配置的所述驱动器验证器功能来捕获所述信息;  
在所述一个或多个驱动器状况包括如由所述一个或多个规则限定的经标识的所述设

备驱动器的错误状况的情况下,继续操作而不允许程式错误检查,并且否则对经标识的所述设备驱动器执行错误恢复响应;以及

发起通信以将所捕获的所述信息传送到驱动器验证分析服务。

13. 根据权利要求12所述的系统,其中所述驱动器验证器功能在预定时间之后被自动禁用。

14. 根据权利要求12所述的系统,其中所述一个或多个规则将所述驱动器验证器功能配置成以指定类别的驱动器为目标。

15. 一种被配置成验证计算设备的功能性的系统,所述系统包括:

用于执行驱动器验证器功能的装置,所述驱动器验证器功能被配置成捕获与由一个或多个规则标识的设备驱动器相关联的信息;

用于响应于由所述一个或多个规则标识的一个或多个驱动器状况而使用经配置的所述驱动器验证器功能来捕获所述信息的装置;

用于在所述一个或多个驱动器状况包括如由所述一个或多个规则限定的经标识的所述设备驱动器的错误状况的情况下维持所述系统的操作而不允许程式错误检查,并且否则对经标识的所述设备驱动器执行错误恢复响应的装置;以及

用于发起通信以将所捕获的所述信息传送到驱动器验证分析服务的装置。

## 设备驱动器遥测

### 背景技术

[0001] 如今的计算设备具有越来越多的附接的和安装的外围设备。为了与这些外围设备接口,计算设备通常运行设备驱动器以操作或控制每种类型的设备。随着外围设备的种类和数目增加以及计算设备的数目增长,在生产和发布之前跟踪和验证设备驱动器的正确操作变得越来越困难。

### 发明内容

[0002] 本文公开了用于验证计算设备的功能性的方法和系统。执行驱动器验证器功能,该驱动器验证器功能被配置成捕获与由一个或多个规则标识的设备驱动器相关联的信息。响应于由规则标识的一个或多个驱动器状况,使用经配置的驱动器验证器功能来捕获信息。在驱动器状况包含由规则限定的错误状况的情况下,允许系统的操作而不进行程序错误(bug)检查。否则,针对所标识的设备驱动器执行错误恢复响应。发起将所捕获的信息传送到驱动器验证分析服务的通信。

[0003] 应当理解,上述主题可以被实施为计算机控制的装置、计算机进程、计算系统,或者诸如一个或多个计算机可读存储介质的制品。通过阅读以下具体实施方式和评阅相关联的附图,这些和各种其他特征将是显而易见的。

[0004] 提供本发明内容是为了以简化的形式介绍一些概念,这些概念将在下面的具体实施方式中进一步描述。本发明内容不旨在标识所要求保护的主题的关键特征或必要特征,也不旨在用于帮助确定所要求保护的主题的范围。另外,所要求保护的主题不限于解决在本公开的任何部分中提到的任何或所有缺点的实施方式。

### 附图说明

[0005] 贯穿附图,可以重复使用附图标记来指示所引用的元件之间的对应关系。提供附图是为了说明本文描述的示例实施例,并且不旨在限制本公开的范围。

[0006] 图1是图示了根据本公开的用于驱动器的验证的机制的图;

[0007] 图2图示了用于收集驱动器数据的示例过程;

[0008] 图3是图示与中央数据服务协调数据的过程的框图;

[0009] 图4是图示生成与驱动器数据相关联的数据对象的方法的框图;

[0010] 图5是图示根据本公开的用于驱动器的验证的机制的图;

[0011] 图6a和图6b是描绘根据本公开的用于驱动器的验证的示例流程的流程图;

[0012] 图7是可以在一些实施例中使用的示例计算机系统。

### 具体实施方式

[0013] 平台或软件提供商可以具有数千个计算设备的安装基础,计算设备包括移动设备和诸如服务器的其他设备。可以在这些设备上调整许多计算设置。例如,可以跨用户基础不同地配置硬件设置和网络设置。由于设备配置的数目和种类繁多,可能难以测试和验证所

有组件和软件,更不用说表征计算设备相对于特定度量的性能。当设备驱动器遇到服务问题时,由于必须评估的数据的量,对问题进行故障排除可能是一个复杂的过程。即使在发现难题的原因时,计算设备的使用的损失也可能导致数据丢失、停机时间和客户不满意。

[0014] 设备驱动器的开发和验证本身就是一项复杂的任务。正确测试设备驱动器对于确保特定驱动器的高质量安装和用户体验非常重要。设计涵盖驱动器可能经历的所有情况的测试是困难的,并且在产品发布之前可能无法实施。例如,在驱动器仍处于生产中时进行设备测试可能会导致测试环境未覆盖的错误。确认驱动器在具有与其他驱动器的所有交互的实时系统中工作使难题更加复杂。当驱动器被释放并被允许在生产环境中执行时,驱动器可能处置关键任务并从其他组件接收连续请求。因此,在生产操作期间发生的任何故障都可能具有对设备功能性和终端用户对系统质量的感知的直接影响。

[0015] 以下详细描述涉及用于评估设备驱动器的技术。在本文描述的示例中,提供驱动器评估和支持服务是可以实施所描述的实施例的一个示例场景。然而,所描述的概念通常可以应用于其他计算组件和服务。

[0016] 描述了用于经由驱动器数据收集来收集和分析设备驱动器的信息的各种实施例。所描述的方法可以包括经由驱动器数据模块枚举与设备驱动器有关的数据。所描述的方法还可以包括在执行驱动器的计算设备中生成数据对象。

[0017] 驱动器数据收集方法可以用于捕获设备驱动器的健康和状态的详细快照并将其发送到中央存储库。当该功能与诸如机器学习的分析方法结合时,可以利用各种方式使服务提供商能够标识和编译驱动器问题的根本原因。例如,可以向支持人员呈现针对驱动器问题的汇总数据。汇总数据还可以对类似设备被交叉参考。这种中央存储库在本文也可以被称为驱动器验证分析服务或集中式设备驱动器数据收集服务,其中任一个都可以包括如本文进一步描述的遥测管理器。通过提供具有详细的和服务中的数据的数据的这种驱动器验证分析服务,可以为任何类型的设备驱动器和设备驱动器的任何组合提供服务。

[0018] 另外,通过提供具有设备驱动器数据的这种驱动器验证分析服务,可以增强可以由服务提供商提供的分析和服务的级别。例如,可以使对基于大量信息的信息分析和评估自动化,从而允许针对设备驱动器的更高级别的基于证据的分析和服务。

[0019] 通过使用所描述的技术,在一些情况下,可以在错误可以显著影响终端用户体验之前标识和纠正驱动器错误。可以在服务中操作期间收集本地设备驱动器信息,以诊断在预发布测试环境中未标识的驱动器错误。在一些实施例中,驱动器验证器功能可以被配置成在执行驱动器的本地计算设备上以数据捕获或记录模式执行。本地计算设备可以与遥测管理器通信并且接收场景以配置设备验证器功能。设备验证器功能可以被配置成利用特定检查和数据捕获指令来以特定类别的驱动器(例如,音频、NDIS、相机)为目标。当满足指定条件时,诸如检测到违反从遥测管理器接收的规则,可以记录和收集事件。不允许本地设备崩溃(例如,蓝屏),这是响应于驱动器故障的通常情况。在指定时间后,可以自动禁用或删除设备验证器,以避免对设备的性能影响。通过在生产和服务中期间允许这种数据捕获,可以在开发期间早期检测到一些设备驱动器故障,并且可以在驱动器在生产中被更广泛地发布之前并入修复。

[0020] 为了允许分析和评估在运行时期间捕获的错误,可以提供用于察看最重要的错误的机制。为了便于标识在生产期间发生的驱动器错误,本文描述了用于优先化和分析驱动

器错误的机制。在一个实施例中，提供遥测管理器以允许构建场景以配置设备验证器。可以构建用于验证特定驱动器的场景并将其发送到各种用户设备。场景可以包括以所标识的设备驱动器为目标并且在目标用户设备上的规则集合。在一些实施例中，可以创建规则的层级结构以建立针对相同设备类别中的驱动器应当如何表现的基线。可以将场景部署到生产用户设备。可以使用指定的格式或声明性语言将规则实施为任何声明性语句。该规则声明可以用于配置驱动器验证器功能的语句。

[0021] 在实施例中，可以从用户设备收集驱动器数据并将其组织成可以被查询和分析的数据桶。遥测管理器可以被配置成定期地且针对每个驱动器/驱动器版本对收获所收集的数据。遥测管理器可以被配置成保持对驱动器性能和规则违反的跟踪。在一些实施例中，可以将设备驱动器的性能与相同设备类别中的其他驱动器以及先前版本进行比较。遥测管理器还可以被配置成将分析转换为分数以提供规则排名。可以提供驱动器健康仪表盘以组织来自用户设备的数据，以允许深入了解指定驱动器的质量。仪表盘可以允许分析度量以确定是否将驱动器提升到更高的验证级别。

[0022] 例如，在一个示例场景中，可以观察到驱动器在25%的时间记录规则违反，而观察到针对相同类别中的驱动器的所建立的基线仅在10%的时间违反。可能发现规则违反中的一个规则违反发生在30%的计算设备中，其中所建立的基线仅在5%的时间经历此相同的规则违反。与针对驱动器类型的所建立的基线相比，可能发现其他规则违反在更少的情况中发生。因此，可以确定应当评估第一违反以尝试提高性能以便与其同类别中的其他驱动器一致。

[0023] 在一个实施例中，集中式设备驱动器数据收集服务可以收集独特的驱动器违反的列表，该独特的驱动器违反具有关于相关联的驱动器的信息，诸如针对其发生违反的驱动器版本和设备标识符。独特的驱动器违例的列表可以用于确定针对给定驱动器和驱动器版本的独特的违反的数目。在一个实施例中，可以生成所有驱动器版本和驱动器名称对以及每个的违反频率的列表。另外，针对给定的驱动器和驱动器版本，可以确定已经下载了驱动器的计算设备的数目。

[0024] 可以收集与计算设备上的驱动器有关的信息，并且在没有计算设备的明显中断或锁定/关机的情况下按照该信息进行动作。在一个实施例中，驱动器数据收集和验证可以在计算设备上以记录模式进行，以扩展验证覆盖范围。例如，可以使用遥测管理器开发场景和规则并将其发送到计算设备。计算设备可以执行本地设备驱动器验证器功能，其可以被实施为在计算设备上执行的本地代理。

[0025] 设备驱动器验证器功能可以被配置成基于所接收的规则利用指定的检查来以特定类别的驱动器（例如，音频、NDIS、相机）为目标。在检测到规则违反时，违反事件可以由设备驱动器验证器功能记录并被提供给遥测组件以用于发送到集中式设备驱动器数据收集服务。在预定时间或指定时间之后，或者根据命令或其他方式，可以自动禁用设备驱动器验证器功能以避免任何未来的性能影响。由集中式设备驱动器数据收集服务收集的遥测数据可以用于检测违反设备验证规则的使用中的驱动器并根据需要采取响应动作。

[0026] 本文描述的技术可以在执行驱动器的任何类型的用户设备（诸如膝上型计算机和移动设备）上实施。为了最小化性能和可恢复性影响，在一些实施例中，某些类别的驱动器可以被选择并被作为针对特定于该平台的验证检查的目标。在一个实施例中，一次仅可以

选择一种类别的驱动器。

[0027] 为了进一步避免用户中断,设备驱动器验证器功能可以被配置成:如果违反了规则,则避免程式错误检查(致命系统错误)。相反,计算设备可以继续操作,并且设备驱动器验证器功能可以继续记录规则违反而无需程式错误检查。另外,可以针对未被任何验证规则覆盖的故障禁用程式错误检查。在一些实施例中,可以在故障的启动时重置设备驱动器验证器功能,并且还可以在预定的时间段或预定数目的启动之后自动禁用设备驱动器验证器功能。

[0028] 在示例实施例中,可以生成覆盖在目标用户设备上运行的各种所选择的驱动器检查的场景集合。可以将场景部署到由遥测管理器限定的用户设备池的集合。已经接收到场景的用户设备可以使用设备驱动器验证器功能中的场景并开始收集针对指定驱动器的数据。用户设备可以在遇到任何驱动器错误时开始发送回遥测。当由集中式设备驱动器数据收集服务接收到数据时,该数据可以被分析。集中式设备驱动器数据收集服务还可以根据需要调整场景以特定驱动器或用户设备为目标来收集附加数据。集中式设备驱动器数据收集服务还可以将信息发送给其他服务以供进一步的分析和调试。

[0029] 在实施例中,所描述的验证检查可以包括一般检查和基于规则的检查。一般检查可以包括例如I/O验证、存储器管理检查以及未在特定状态机方面限定的其他验证行为。可以使用规则限定语言提前限定基于规则的检查。规则可以被构造成检测当驱动器进入非法状态或其他状况时出现的问题。例如,规则可以应用于对诸如设备驱动器接口(DDI)的驱动器应用程序编程接口的调用。通常,可以通过在操作系统注册表中设置标志来激活或停用检查。

[0030] 在通常的驱动器配置下,违反验证检查可能导致程式错误检查。对于包含本文描述的设备驱动器验证器功能的实施方式,设备驱动器验证器功能可以在违反规则时对驱动器行为进行控制并且防止程式错误检查。

[0031] 在一些实施例中,计算设备可以在正常模式和驱动器测试模式下操作,其中设备驱动器验证管理器仅在记录状态下操作。在此状态下,将记录任何驱动器违反,而不是导致程式错误检查。违反可以被记录到事件日志,该事件日志可以由遥测代理访问以发送到集中式设备驱动器数据收集服务。

[0032] 在一些实施例中,自动检查可以被配置成仅在设置与程式错误检查分离的特定标志时激活。另外,设备驱动器验证器功能可以被配置成仅启用对应于活动标志的检查。

[0033] 在一些实施例中,被记录的信息可以包括被违反的规则和从其记录事件的模块。可以记录附加信息以实现进一步分析并允许诊断错误。例如,可以记录与在故障点处违反的规则的状态有关的信息。另外,可以收集实时转储。可以在配置时控制附加数据的收集。

[0034] 为了最小化对计算设备的性能影响,可以将集中式设备驱动器数据收集服务配置成使得在驱动器上一次仅启用一个规则集合,规则可以应用到该驱动器。例如,设备驱动器验证管理器可以被配置成在NDIS驱动器上运行NDIS规则,但不同时在音频驱动器上运行音频规则。

[0035] 为了减少在发送遥测时的网络拥塞,设备驱动器验证器功能可以发送关于违反的基本信息。另外,如果重复发生相同的问题,则设备驱动器验证器功能可以最终停止生成事件。这可能在主动运行故障驱动器时发生。设备驱动器验证器功能可以被配置成仅在设置

特定标志时发送详细信息。

[0036] 对于功率敏感设备,诸如以电池功率操作的那些设备,设备驱动器验证器功能可以被配置成在功率节省模式下操作,以操作使得在连接不良期间最小化错误数据的传输或者在电池电量低时存储数据以供后续传输。

[0037] 在一些实施例中,可以指定计算设备的自定义组。例如,可以通过收集用户设备的列表来指定组,可以选择该用户设备以关联仅包含特定设备和驱动器的系统的数据。以这种方式,可以以特定类型或制造商或指定的驱动器版本的设备为目标进行聚焦分析。在其他示例中,可以以指定OS版本、区域或其他因素的设备为目标。

[0038] 集中式设备驱动器数据收集服务可以根据规则从所有启用遥测的终端用户设备收集遥测数据,这些终端用户设备被连接并且能够传送信息。集中式设备驱动器数据收集服务可以将遥测数据组织成流或桶,该流或桶可以被查询、分析和进一步分类。为了分析数据,可以使用查询来搜索独特的事件标识符。例如,事件标识符可以包括多种类别,诸如遇到过指定驱动器错误的用户设备、错误的特征、违反检查的驱动器、所违反的检查、违反的频率等。

[0039] 数据可以被调度成以定期或根据其他调度而被收集。可以分析所收集的数据以标识趋势和阈值。例如,如果特定驱动器具有与其相关联的异常大量的事件,则可以确定需要对该驱动器执行进一步的分析。作为响应,可以构建场景以讨论中的驱动器为目标并启用附加的记录选项。

[0040] 可以提供用户界面以呈现汇总所收集的驱动器数据的图形仪表板。在一些实施例中,仪表板可以包括用于选择日期范围、按平台过滤、按驱动器类别过滤、按驱动器过滤、按规则过滤等的选项。

[0041] 在一些实施例中,可以如下跟踪特定驱动器的信息:

[0042]  $(\# \text{已发生的机器特定的违反} / \# \text{实验中的机器}) * 100 = \text{驱动器违反频率};$

[0043]  $(\# \text{总的具有违反的机器} / \# \text{实验中的机器}) * 100 = \text{整体违反频率}。$

[0044] 在一些实施例中,可以如下跟踪相同类别中的驱动器的信息:

[0045]  $(\# \text{已发生的机器特定的违反} / \# \text{实验中的机器}) * 100 = \text{基线频率};$

[0046]  $(\# \text{总的具有违反的机器} / \# \text{实验中的机器}) * 100 = \text{整体基线频率}。$

[0047] 在一些实施例中,可以如下跟踪特定违反的信息:

[0048] 对于每次违反:

[0049]  $\text{违反频率} * (\text{违反权重}) - \text{基线频率} = \text{违反分数};$

[0050] 如果:  $\Sigma \text{违反分数} > 15 \rightarrow \text{指示 (例如,红色)};$

[0051] 否则如果:  $\Sigma \text{违反分数} < 5 \rightarrow \text{继续}。$

[0052] 在一些实施例中,可以如下跟踪用于回归的信息(将新版本与先前版本进行比较):

[0053] 如果:  $\Sigma \text{旧违反分数} - \Sigma \text{当前违反分数} < -5 \rightarrow \text{指示 (例如,红色)};$

[0054] 否则如果:  $\Sigma \text{旧违反分数} - \Sigma \text{当前违反分数} > 0 \rightarrow \text{指示 (例如,绿色)}。$

[0055] 在一些实施例中,计算设备可以在学习模式下操作,其中收集驱动器问题并且创建基线简档,而不是生成系统错误。

[0056] 在一些实施例中,详细的设备驱动器信息可以补充有来自服务提供商的信息、来

自行业源的信息和众包信息。

[0057] 在一个示例中,在检测到驱动器难题或问题之后,集中式设备驱动器数据收集服务可以被配置成从所部署的用户设备收集数据、分析数据并呈现经分析的数据。在标识了对难题或问题的修复之后,集中式设备驱动器数据收集服务可以自动推送修复或自动回滚(roll back)驱动器更新。

[0058] 在一些实施例中,集中式设备驱动器数据收集服务可以被实施成基于人的分析的备选或增强,其可以允许管理许多变量和设备状态,这在没有这种系统的情况下可能是困难或不可能的。设备和系统提供商可以通过集中式设备驱动器数据收集服务发布新的驱动器代码。因此,可以利用诸如因特网的网络以连续的方式执行测试和验证,并且在自动化的基础上提供该测试和验证同时减少设备驱动器问题和易受攻击性。

[0059] 本文描述的技术中的一些包括本地于计算设备的可执行模块。可执行模块在本文可以被称为驱动器数据模块,其可以是设备驱动器验证器功能的一部分。驱动器数据模块可以在存储器中生成数据对象。在一些实施例中,可以压缩和加密数据对象以供跨包括非安全网络路径的网络进行传输。例如,经压缩和经加密的数据对象可以是持久的,使得可以在其他路径不可用时使用跨非安全网络路径的传输,同时降低暴露私密或机密数据的风险。

[0060] 图1是被配置成提供计算系统和环境的驱动器数据的计算系统的示例的框图。系统可以包括计算设备102。计算设备102可以是例如移动电话、膝上型计算机、台式计算机或平板电脑等。计算设备102可以包括适于执行存储的指令的处理器104,以及存储由处理器102可执行的指令的存储器单元106。存储器单元106可以是与存储设备108相关的非持久存储器,存储设备108可以例如包括硬盘驱动。处理器104可以是单核处理器、多核处理器、计算集群或任何数目的其他配置。存储器单元106可以包括随机存取存储器(例如,静态随机存取存储器(SRAM)、动态随机存取存储器(DRAM)、零电容器RAM、硅氧化物-氮化物-氧化物-硅SONOS、嵌入式DRAM、扩展数据输出RAM、双倍数据速率(DDR)RAM、电阻随机存取存储器(RRAM)、参数随机存取存储器(PRAM)等)、只读存储器(ROM)(例如,掩模ROM、可编程只读存储器(PROM)、可擦除可编程只读存储器(EPROM)、电可擦除可编程只读存储器(EEPROM)等)、闪速存储器或任何其他合适的存储器系统。由处理器104执行的指令可以用于收集设备驱动器数据。

[0061] 处理器104可以通过系统总线110(例如,PCI、ISA、PCI-Express、NuBus等)连接到网络接口116,网络接口116也可以适于通过系统总线106将计算系统100连接到网络114。

[0062] 存储设备108可以包括硬盘驱动、光盘驱动、USB闪速驱动、驱动阵列或其任何组合。在一些情况下,驱动器数据模块116可以是要被存储在存储设备108中的可执行指令。在一些情况下,驱动器数据模块116可以被实施成逻辑,其至少部分地包括硬件逻辑。驱动器数据模块116可以包括子模块,子模块包括压缩模块118、加密模块120和传输模块122。尽管图1将模块118、120和122中的每个图示为驱动器数据模块116的子模块,但是模块116、118、120和122中的每个可以被实施成分立组件,或者被实施成更广泛的过程的组件或逻辑硬件构造。

[0063] 驱动器数据模块116可以被配置成从计算设备102的驱动器收集数据并生成被存储在计算设备102的存储器单元中的数据对象。在一些情况下,驱动器数据模块116可以被

配置成通过网络114提供到中央数据中心116的通信,中央数据中心116可以对应于所描述的集中式设备驱动器数据收集服务,其中可以解密、解压缩和分析经压缩和经加密的数据对象。在一些实施例中,如果驱动器数据模块116被配置成仅具有对中央数据中心124的单向访问,则这可以减少任何后门安全威胁潜在性,否则该后门安全威胁潜在性可以实现对驱动器数据模块116的恶意或甚至是非有意访问。另外,在一些实施例中,可以压制由驱动器数据模块116提供的数据。例如,取决于在计算设备102上运行的给定应用,驱动器数据模块116可以延迟数据报告一段时间,使得可以不影响任何其他操作或应用的性能。换言之,驱动器数据模块116可以被配置成在可能由与驱动器数据收集操作相关的增加的负载引起的性能降级方面不可检测。

[0064] 驱动器数据模块116可以被配置成基于所收集的数据生成被存储在计算设备102的存储器单元(诸如存储器单元106)中的数据对象。

[0065] 被存储在存储器单元106中的数据对象可以包括定界列表。数据对象可以包括嵌入到单个文件中的多个定界部分。可以将驱动器数据的多个部分提取出,作为单独的定界文件。多个定界列表可以各自与不同类别的驱动器信息相关。

[0066] 可以在对经由中央数据中心124提供的一个或多个外部操作进行操作时部署驱动器数据模块116。例如,中央数据中心124可以运行结构化语言查询(SQL)以询问计算设备102。在部署SQL时,可以启动驱动器数据模块116。在一些情况下,驱动器数据模块116可以嵌入到计算设备102的操作系统(OS)中。在该场景中,OS可以启动驱动器数据模块116的操作。在任何情况下,驱动器数据模块116都可以被配置成在预定的可配置时间段之后部署。例如,在一些情况下,驱动器数据模块116可以被配置成每小时部署一次。

[0067] 在驱动器数据模块116生成了与驱动器数据有关的数据对象之后,压缩模块118可以被配置成压缩数据对象。然后可以经由加密模块120对经压缩的数据对象进行加密。加密模块120可以包括同步加密的两部分过程,诸如高级加密标准(AES),以及诸如RSA的异步加密。预期同步和异步加密的其他加密方法。

[0068] 计算系统100可以包括其他联网设备,诸如一个或多个联网计算设备126。联网计算设备126可以类似于计算设备102被实施,并且可以包括驱动器数据模块,诸如本地于每个设备的驱动器数据模块116。可以将经压缩和经加密的格式的驱动器数据提供给中央数据中心124,以供驱动器数据处置器128分析。驱动器数据处置器128可以被实施成逻辑,其至少部分地包括硬件逻辑、软件、固件或其任何组合。驱动器数据处置器128可以被配置成对驱动器数据进行解密和解压缩,以及将组合的定界文件分解成与组合的定界文件中的多个定界文件中的每个相关联的数据的维度。数据可以被存储在中央数据存储库中,诸如中央数据中心124的驱动器数据存储库120。如下面更详细地讨论的,被提供给中央数据中心124的驱动器数据可以用于:基于所接收的驱动器数据对象的变化,更新当前或先前已知的驱动器数据。

[0069] 应当理解,图1的框图并不旨在指示计算系统将包括图1中所示的所有组件。相反,计算系统可以包括更少的组件或图1中未图示的更多的组件(例如,附加应用、附加模块、附加存储器设备、附加网络接口等)。另外,压缩模块118、加密模块120和传输模块122的任何功能可以部分地或全部地以硬件实施和/或在处理器104中实施。例如,可以利用专用集成电路实施功能性、可以在处理器104中实施的逻辑中实施功能性,或在任何其他设备中实施

功能性。

[0070] 图2是用于收集用户设备的驱动器数据的示例过程的过程流程图。在框202处,可以在联网计算设备(诸如图1的计算设备102)处本地查询驱动器数据。

[0071] 在框204处,可以组装定界字符串。对定界字符串的组装可以包括基于在框202处查询的所收集的驱动器数据来生成被存储在存储器单元(诸如计算设备102的存储器单元106)中的数据对象。在框206处,可以压缩数据对象,并且在框208处,可以加密经压缩的数据对象。

[0072] 在框208处的加密可以是同步加密,诸如上面关于图1讨论的AES加密。同步加密208可以使用动态生成的密钥和初始化向量以及经加密字节的所计算的散列值。在框210处,可以对所计算的散列值、动态生成的密钥和初始化向量执行异步加密。可以使用目的地设备的嵌入式公共密钥将加密210封装为预封装,目的地设备诸如与图1的中央数据中心124相关联的设备。在一些情况下,如果存在临时目录,则可以在计算设备102处创建具有驱动器数据的纯文本、人类可读的文件。

[0073] 在框212处,可以将预封装和包括来自同步加密208的经加密字节的所计算的散列值的有效负载组合成输出文件。可以将输出文件提供给网络,诸如网络114。在一些情况下,网络114可以包括一个或多个最接近的联网设备,诸如联网计算设备126。图1的驱动器数据模块114可以包括用于收件箱的地址的列表以潜在地用作目的地收件箱。

[0074] 在框214处,可以对封装以及预封装进行排队。在214处的队列可以被配置成从收件箱中拉取驱动器数据对象并对它们进行排队以用于分解。预先地,可以在216处执行解密,解密包括对预封装的解密和对有效负载封装的解密。另外,可以在218处对驱动器数据对象进行解压缩。在220处,可以分解来自框204的经组装的定界字符串。换言之,在222处发生的合并期间,可以将定界文件的组合分离并分解为序列化文本以被存储在关系数据库结构中。合并可以包括经由SQL将驱动器数据加入到合适的数据库。

[0075] 在一些情况下,在220处的分解可以包括批量分解成存储器中的数据表以增加吞吐量。在给定数目的封装被分解成数据表的集之后,它们就可以在合并222期间被传递给SQL。每个存储器中数据表可以被直接传递给SQL,而没有任何输入/输出开销。例如,可以组装数据表字典。可以从队列中取回最旧的封装。可以独占地锁定该封装,并且可以从与同步加密相关联的所计算的散列值以及密钥和向量中分离出有效负载的前128个字节、RSA解密。可以计算第二同步散列(SHA)以与经解密的RSA散列进行比较。如果散列不匹配,则可以将该封装标记为可疑。

[0076] 可以使用初始化向量和密钥来解密有效负载。在数据被解压缩之后,就可以解析定界的和经去标记的枚举对象数据。例如,可以以数据表为目标,其中经去标记的值涉及字典中的密钥。可以基于批量大小分阶段地进行解压缩。具有多于0行的字典的每个数据表可以被发送到SQL,并且在被提交到SQL之后,就可以删除驱动器数据对象。另外,合并222可以包括被配置成跟踪性能、可缩放性、变化率的运行时度量。

[0077] 图3是图示使用中央资产服务(诸如所描述的集中式设备驱动器数据收集服务)来协调驱动器数据的过程的框图。在一些情况下,可以将所接收的驱动器数据与驱动器数据、先前存储的驱动器数据进行比较,并且可以基于驱动器数据的任何变化来提供更新。在图3中,可以在302处生成数据对象。

[0078] 数据对象302可以是如上面关于图1和图2所讨论的从一个或多个计算设备(诸如计算设备102、联网计算设备126等)生成的驱动器数据对象。在302处的驱动器数据对象可以被传递到驱动器数据聚合器304。驱动器数据聚合器304可以是数据存储库,诸如图1的驱动器数据存储库130,或者可以是被配置成临时存储驱动器数据的单独的数据存储库。如在306处所指示的,中央资产服务308可以指向驱动器数据聚合器304。中央资产服务308可以是中央库存系统,其被配置成接收驱动器数据并且在驱动器数据变化时经由协调模块310协调驱动器数据。可以向中央驱动器数据存储库312提供驱动器数据的差异或变化。诸如源聚合器314的附加工具可以用于从一个或多个部署模块316接收更新。源聚合器314的操作可以向各种一个或多个部署模块316提供更新,该更新基于在协调模块310处执行的协调并且被存储在中央驱动器数据存储库312中。

[0079] 图4是用于收集驱动器数据的示例计算机可读存储介质的框图。有形的和非暂态的计算机可读存储介质400可以由处理器402通过计算机总线404访问。另外,有形的和非暂态的计算机可读存储介质400可以包括指示处理器402执行所描述的方法的步骤的代码。

[0080] 如图4中所指示的,本文讨论的各种软件组件可以被存储在有形的和非暂态的计算机可读存储介质400上。例如,有形的和非暂态的计算机可读存储介质400可以包括驱动器数据收集模块406,以用于收集用户设备的驱动器数据。驱动器数据收集模块406还可以基于所收集的驱动器数据生成被存储在联网设备的存储器单元中的数据对象,其中所生成的对象包括在单个文件中表示的多个定界部分。驱动器数据收集模块406还可以基于所收集的驱动器数据生成被存储在联网设备的存储器单元中的数据对象。可以压缩和加密数据。可以将经加密和经压缩的数据对象传送到目的地设备。

[0081] 本文描述的数据包括可以由设备驱动器评估系统从设备驱动器收集的信息的示例。可以收集其他设备健康和组件级别数据。可以跨各种环境将数据的收集扩展到无限数目的设备。

[0082] 图5是图示根据本公开的用于标识驱动器问题的设备驱动器评估系统500的图。在图5中,设备外服务可以包括资源和应用,其可以例如在一个或多个服务器530上执行。应当理解,一些实施例可以涉及可以被提供的各种类型的附加计算资源540。

[0083] 图5还图示了可以由用户510可访问的一个或多个设备提供通信的网络520。根据一个实施例,在服务器530上执行的资源可以被配置成经由网络520向用户510提供驱动器支持服务。图5还图示了数据存储库550,其可以存储与用户510的设备的各种驱动器数据有关的数据。

[0084] 可以将请求发送到设备驱动器评估系统500,以用于监控和分析一个或多个设备驱动器。设备驱动器评估系统500可以与服务器530和数据存储库550共同主存,或者可以驻留在用户510的设备上,或者其组合。在一些实施例中,可以从用户510中的一个用户的设备接收请求以供分析服务。在其他实施例中,可以从设备制造商处的一个或多个服务接收请求。响应于请求的接收,设备驱动器评估系统500可以记录该请求并提供关于请求的状态的更新。设备驱动器评估系统500可以与其他服务通信以有助于:(1)处理请求,(2)收集与请求有关的数据,以及(3)生成界面以提供请求的结果。例如,设备驱动器评估系统500可以提供用于有助于提交请求的界面。设备驱动器评估系统500还可以提供用于察看请求的结果以及修改或取消请求的界面。

[0085] 基于配置的评估系统500可以被配置成提供分析和诊断,以用于基于对各种设备驱动器的实时的或累积的和/或存档的监控来提供分析。设备驱动器评估系统500可以访问诸如设备类型和设备使用活动的度量。可以经由应用编程接口(API)或可以经由Web浏览器或其他输入机制访问的用户界面使设备驱动器评估系统500可访问。

[0086] 在一些实施例中,设备驱动器评估系统500可以确定故障或其他事件的预测的概率。基于配置的评估系统500可以从操作环境的其他组件,诸如数据存储库550,收集驱动器数据。数据存储库550可以从用户510的设备等收集信息。设备驱动器评估系统500还可以收集被存储在日志文件中和从计算资源540可用的其他位置中的信息。还可以通过向设备查询当前未被存储在日志文件中的数据来获得该信息。

[0087] 应当理解,图5中所图示的网络拓扑结构已经被极大地简化,并且可以利用更多的网络和联网设备来互连本文公开的各种计算系统。这些网络拓扑结构和设备对于本领域技术人员来说应当是显而易见的。

[0088] 另外,应当理解,本文公开的功能性可以用软件、硬件或软件和硬件的组合来实施。其他实施方式对于本领域技术人员来说应当是显而易见的。还应当理解,服务器、网关或其他计算设备可以包括可以交互并执行所描述的功能性的硬件或软件的任何组合,包括但不限于台式或其他计算机、数据库服务器、网络存储设备和其他网络设备、PDA、平板电脑、手机、无线电话、寻呼机、电子整理器、互联网电器、基于电视的系统(例如,使用机顶盒和/或个人/数字视频记录器)和包括合适通信能力的各种其他消费产品。另外,在一些实施例中,由所图示的模块提供的功能性可以被组合在更少的模块中或分布在附加的模块中。类似地,在一些实施例中,可以不提供所图示的模块中的一些模块的功能性和/或其他附加功能性可以是可用的。

[0089] 在一些实施例中,设备驱动器评估系统500可以自动监控驱动器数据。例如,服务提供商可以以预定频率监控多个预先选择的设备驱动器。可以与服务提供商指定的相同或不同的频率收集附加的驱动器数据。

[0090] 另外,设备驱动器评估系统500可以根据设备驱动器事件向用户提供通知或采取指定的动作。例如,服务提供商可以使经更新的驱动器下载到设备,该设备被预测为故障或以其他方式导致设备中的问题。

[0091] 在一些实施例中,可以提供API以有助于对设备驱动器信息的请求。例如,可以利用诸如设备标识符的信息来调用API。在调用API之后,在一个实施例中,设备驱动器评估系统500可以采取诸如以下的动作:

[0092] • 访问设备的活动记录。

[0093] • 取回相关设备的驱动器数据。

[0094] 可以聚合和分析所存储的数据以标识简档、模式和各种统计。例如,通过分析错误率,可以确定设备驱动器的健康和性能。所收集的驱动器数据的数据挖掘可以有用于标识用于推断设备驱动器或设备驱动器组的当前状态的各种特征。

[0095] 在一些实施例中,可以分析所收集的数据以确定统计信息。另外,可以执行数据挖掘以标识与设备功能、健康和性能相关的趋势和模式。在一个实施例中,设备驱动器评估系统可以被配置成分析数据。例如,设备驱动器评估系统可以被实施成速率检测向量引擎。检测向量引擎可以包含朴素贝叶斯分类器、模糊逻辑和其他方法。检测向量引擎可以被配置

成确定哪些数据模式与错误分析相关。

[0096] 在一些实施例中,设备驱动器评估系统可以实施模式分类系统以标识相关数据。例如,检测向量引擎可以包括学习系统,该学习系统可以包括学习功能以持续地学习哪些数据模式与特定潜在难题相关。

[0097] 通常,设备驱动器评估系统可以向各种用户提供信息。例如,设备驱动器评估系统可以允许数据经由网站可访问,或者可以允许数据经由基于云的系统对用户可访问。设备驱动器评估系统还可以允许设备支持和操作服务经由网站或通过边信道访问数据。因此,设备支持和操作服务可以访问错误级别、警告和信息状态,以向用户提供合适和及时的服务。例如,设计和质量保证人员可以访问数据以做出对设计和生产过程的改进。

[0098] 设备驱动器评估系统可以经由设备的用户界面向设备的用户提供有用的警报和信息。作为示例,在一个场景中,可以向用户提供健康设备指示。

[0099] 在另一客户警报场景中,可以向用户提供设备健康警报。警报可以向用户通知检测到的可能影响用户体验的异常行为。

[0100] 设备驱动器评估系统可以通过提供对可能不在单个设备上执行的设备驱动器的更准确的诊断结果和可跟踪的趋势的计算来改善用户体验。

[0101] 图6a图示了用于验证计算设备的功能性的示例操作流程。在实施例中,该流程可以至少部分地在对应于图1的计算设备上的系统或功能中实施。

[0102] 参考图6a,操作600开始操作流程。操作602可以跟随操作600。操作602图示了接收一个或多个规则,一个或多个规则可用于配置驱动器验证器功能以捕获与由一个或多个规则标识的设备驱动器相关联的信息。操作604可以跟随操作602。操作604图示了在计算设备上配置驱动器验证器功能以根据一个或多个规则捕获与所标识的设备驱动器相关联的信息。操作606可以跟随操作604。操作606图示了在计算设备上执行经配置的驱动器验证器功能。

[0103] 操作608可以跟随操作606。操作606图示了:响应于由一个或多个规则标识的一个或多个驱动器状况,使用经配置的驱动器验证器功能来捕获信息。操作610可以跟随操作608。操作610图示了:在一个或多个驱动器状况包括由一个或多个规则限定的所标识的设备驱动器的错误状况的情况下,允许计算设备继续操作,否则对所标识的设备驱动器执行错误恢复响应。操作612可以跟随操作610。操作612图示了发起通信以将所捕获的信息传送到驱动器验证分析服务。

[0104] 图6b图示了用于验证多个计算设备的功能性的示例操作流程。在实施例中,该流程可以至少部分地在对应于图3和图5的系统上的系统或功能中实施。

[0105] 参考图6b,操作650开始操作流程。操作652可以跟随操作650。操作652图示了向计算设备发送与被安装在计算设备上的设备驱动器相关联的规则集合。操作654可以跟随操作652。操作654图示了从计算设备接收指示计算设备的驱动器数据的数据。在实施例中,可以通过在计算设备上执行的驱动器验证器功能来收集驱动器数据。驱动器验证器功能可以被配置成根据规则集合来捕获与设备驱动器相关联的信息。

[0106] 操作656可以跟随操作654。操作656图示了解析驱动器数据以利用先前收集的其他计算设备的驱动器数据对驱动器数据进行分类。可以至少部分地基于驱动器数据的属性来对驱动器数据进行分类。操作658可以跟随操作656。操作658图示了提供经分类的驱动器

数据以用于分析计算设备上的驱动器健康。

[0107] 本公开的实施例可以如以下条款中所描述的那样实施：

[0108] 一种用于验证计算设备的功能性的计算机实施的方法，该方法包括：

[0109] 接收一个或多个规则，一个或多个规则可用于配置驱动器验证器功能以捕获与由一个或多个规则标识的设备驱动器相关联的信息；

[0110] 在计算设备上配置驱动器验证器功能，以根据一个或多个规则捕获与所标识的设备驱动器相关联的信息；

[0111] 在计算设备上执行经配置的驱动器验证器功能；

[0112] 响应于由一个或多个规则标识的一个或多个驱动器状况，使用经配置的驱动器验证器功能来捕获该信息；

[0113] 在一个或多个驱动器状况包括如由一个或多个规则限定的经标识的设备驱动器的错误状况的情况下，允许计算设备继续操作，并且否则对经标识的设备驱动器执行错误恢复响应；以及

[0114] 发起通信以将所捕获的信息传送到驱动器验证分析服务。

[0115] 在该方法中，从驱动器验证遥测管理器接收一个或多个规则。

[0116] 在该方法中，驱动器验证器功能可操作以在数据捕获模式或记录模式下运行。

[0117] 在该方法中，驱动器验证器功能在预定时间之后被自动禁用。

[0118] 在该方法中，一个或多个规则将驱动器验证器功能配置成以指定类别的驱动器为目标。

[0119] 在该方法中，指定类别的驱动器包括音频、NDIS和相机驱动器中的一种或多种。

[0120] 在该方法中，捕获包括检测对规则中的一个规则的违反，并且作为响应，记录事件并存储相关联的驱动器数据。

[0121] 在该方法中，一个或多个规则实施一般驱动器检查和基于规则的检查。

[0122] 在该方法中，一般检查包括I/O验证和存储器管理中的一个或多个。

[0123] 在该方法中，一次仅启用规则中的一个规则。

[0124] 在该方法中，驱动器验证器功能可操作以在学习模式下运行，其中驱动器问题被收集并且基线简档被创建而不是生成系统错误。

[0125] 一种被配置成验证计算设备的功能性的系统，该系统包括：

[0126] 至少一个存储器，其中存储有计算机指令，该计算机指令在由系统的一个或多个处理器执行时至少使系统：

[0127] 执行驱动器验证器功能，该驱动器验证器功能被配置成捕获与由一个或多个规则标识的设备驱动器相关联的信息；

[0128] 响应于由一个或多个规则标识的一个或多个驱动器状况，使用经配置的驱动器验证器功能来捕获该信息；

[0129] 在一个或多个驱动器状况包括如由一个或多个规则限定的经标识的设备驱动器的错误状况的情况下，继续操作而不允许程式错误检查，并且否则对经标识的设备驱动器执行错误恢复响应；以及

[0130] 发起通信以将所捕获的信息传送到驱动器验证分析服务。

[0131] 在该系统中，驱动器验证器功能在预定时间之后被自动禁用。

- [0132] 在该系统中,一个或多个规则将驱动器验证器功能配置成以指定类别的驱动器为目标。
- [0133] 在该系统中,一个或多个规则实施一般驱动器检查和基于规则的检查。
- [0134] 一种被配置成验证计算设备的功能性的系统,该系统包括:
- [0135] 用于执行驱动器验证器功能的装置,驱动器验证器功能被配置成捕获与由一个或多个规则标识的设备驱动器相关联的信息;
- [0136] 用于响应于由一个或多个规则标识的一个或多个驱动器状况,而使用经配置的驱动器验证器功能来捕获该信息的装置;
- [0137] 用于在一个或多个驱动器状况包括如由一个或多个规则限定的经标识的设备驱动器的错误状况的情况下维持系统的操作而不允许程式错误检查,并且否则对经标识的设备驱动器执行错误恢复响应的装置;以及
- [0138] 用于发起通信以将所捕获的信息传送到驱动器验证分析服务的装置。
- [0139] 在该系统中,驱动器验证器功能可操作以在数据捕获模式或记录模式下运行。
- [0140] 在该系统中,一个或多个规则将驱动器验证器功能配置成以指定类别的驱动器为目标。
- [0141] 在该系统中,用于捕获的装置包括用于检测对规则中的一个规则的违反,并且作为响应记录事件并存储相关联的驱动器数据的装置。
- [0142] 在该系统中,一个或多个规则实施一般驱动器检查和基于规则的检查。
- [0143] 在至少一些实施例中,实施本文描述的技术,包括实施设备驱动器评估系统700的功能性的技术中的一种或多种中的一部分或全部的计算设备可以包括通用计算机系统,该通用计算机系统包括或被配置成访问一个或多个计算机可访问介质。图7图示了这种通用计算设备700。在所图示实施例中,计算设备700包括经由输入/输出(I/O)接口730耦合到系统存储器720的一个或多个处理器710a、710b和/或710n(在本文可以以单数称为“处理器710”或以复数称为“多个处理器710”)。计算设备700还包括耦合到I/O接口730的网络接口740。
- [0144] 在各种实施例中,计算设备700可以是包括一个处理器710的单处理器系统,或是包括若干处理器710(例如,两个、四个、八个或另一合适数目)的多处理器系统。处理器710可以是能够执行指令的任何合适的处理器。例如,在各种实施例中,处理器710可以是实施各种指令集架构(ISA)中的任何一种的通用或嵌入式处理器,诸如x86、PowerPC、SPARC或MIPS ISA,或任何其他合适的ISA。在多处理器系统中,处理器710中的每个可以共同地但不是必须地实施相同的ISA。
- [0145] 系统存储器720可以被配置成存储由(多个)处理器710可访问的指令和数据。在各种实施例中,系统存储器720可以使用任何合适的存储器技术来实施,诸如静态随机存取存储器(SRAM)、同步动态RAM(SDRAM)、非易失性/闪速型存储器或任何其他类型的存储器。在所图示的实施例中,实施一个或多个所需功能,诸如上述那些方法、技术和数据的程序指令和数据被示为被存储在系统存储器720内作为代码725和数据726。
- [0146] 在一个实施例中,I/O接口730可以被配置成调节处理器710、系统存储器720和设备中的任何外围设备包括网络接口740或其他外围接口之间的I/O流量。在一些实施例中,I/O接口730可以执行任何必要的协议、时序或其他数据变换,以将来自一个组件(例如,系

统存储器720)的数据信号转换成适于由另一组件(例如,处理器710)使用的格式。在一些实施例中,I/O接口730可以包括对通过各种类型的外围总线(例如,诸如外围组件互连(PCI)总线标准的变型或通用串行总线(USB)标准的变型)附接的设备的支持。在一些实施例中,I/O接口730的功能可以被分成两个或更多个单独的组件,例如,诸如北桥接器和南桥接器。而且,在一些实施例中,I/O接口730的功能性中的一些或全部,诸如到系统存储器720的接口,可以被直接合并到处理器710中。

[0147] 网络接口740可以被配置成允许数据在计算设备700和其他设备或附接到网络或(多个)网络750的多个设备760(例如,诸如图1到图11中所图示的其他计算机系统或设备)之间交换。在各种实施例中,网络接口740可以支持经由任何合适的有线或无线通用数据网络的通信,例如,诸如以太网网络的类型。另外,网络接口740可以支持经由诸如模拟语音网络或数字光纤通信网络的电信/电话网络、经由诸如光纤通道SAN的存储区域网络或经由任何其他合适类型的网络和/或协议的通信。

[0148] 在一些实施例中,系统存储器720可以是计算机可访问介质的一个实施例,该计算机可访问介质被配置成存储如上面针对图1-图6所描述的用于实施对应的方法和装置的实施例的程序指令和数据。然而,在其他实施例中,可以在不同类型的计算机可访问介质上接收、发送或存储程序指令和/或数据。一般而言,计算机可访问介质可以包括非暂态存储介质或存储器介质,诸如磁介质或光学介质,例如经由I/O接口730耦合到计算设备700的磁盘或DVD/CD。非暂态计算机可访问存储介质还可以包括任何易失性或非易失性介质,诸如RAM(例如SDRAM、DDR SDRAM、RDRAM、SRAM等)、ROM等,其可以被包括在计算设备700的一些实施例中作为系统存储器720或其他类型的存储器。另外,计算机可访问介质可以包括传输介质或诸如电信号、电磁信号或数字信号的信号,该信号经由诸如可以经由网络接口740实施的诸如网络和/或无线链路的通信介质传送。多个计算设备(诸如图7中所示的那些)中的部分或全部可以用于实施在各种实施例中的所描述的功能性;例如,在各种不同设备和服务器上运行的软件组件可以协作以提供功能性。在一些实施例中,除了使用通用计算机系统实施之外或代替使用通用计算机系统实施,可以使用存储设备、网络设备或专用计算机系统来实施所描述的功能性的部分。本文使用的术语“计算设备”是指至少所有这些类型的设备,并且不限于这些类型的设备。出于本说明书和权利要求的目的,短语“计算机可读存储介质”及其变型不包括波、信号和/或其他暂态和/或无形的通信介质。

[0149] 由诸如公司或制造商的实体建立的、用于向分布式客户端集合提供经由因特网和/或其他网络可访问的一个或多个服务(诸如各种类型的基于云的分析服务)的网络可以被称为服务提供商。这种提供商网络可以包括主存各种资源池的许多数据中心,诸如实施和分布由服务提供商提供的基础设施和服务所需的物理的和/或虚拟化的计算机服务器、存储设备、联网装备等的集。

[0150] 可以单独或组合使用多种不同类型的计算设备以在不同实施例中实施服务提供商的资源,资源包括通用或专用计算机服务器、存储设备、网络设备等。在一些实施例中,例如,通过给用户管理员登录名和密码,可以向客户端或用户提供对资源实例的直接访问。在其他实施例中,提供商网络运营商可以允许客户端指定针对指定的客户端应用的执行要求,并且代表客户端在适合于应用的执行平台(诸如应用服务器实例、Java™虚拟机(JVM)、通用或专用操作系统、支持各种解释的或编译的编程语言(诸如Ruby、Perl、Python、C、C++

等)的平台或高性能计算平台)上调度应用的执行,而无需例如客户端直接访问实例或执行平台。在一些实施方式中,给定的执行平台可以利用一个或多个资源实例;在其他实施方式中,可以将多个执行平台映射到单个资源实例。

[0151] 如本文所使用的,短语“被配置成”可以涵盖任何类型的结构组件可以被构造成执行所标识的操作的任何方式。结构组件可以被配置成使用软件、硬件、固件等或其任何组合来执行操作。术语“逻辑”可以涵盖用于执行任务的任何功能性。例如,流程图中图示的每个操作对应于用于执行该操作的逻辑。可以使用软件、硬件、固件、物理电子电路等或其任何组合来执行操作。术语“组件”、“系统”、“客户端”等旨在指与计算机相关的实体,无论是(例如,执行中的)硬件、软件和/或固件还是其组合。例如,组件可以是在处理器上运行的进程、对象、可执行文件、程序、函数、库、子例程和/或计算机或软件和硬件的组合。作为说明,在服务器上运行的应用和服务器都可以是组件。一个或多个组件可以驻留在进程内,并且组件可以被本地化于一台计算机上和/或分布在两台或更多台计算机之间。

[0152] 另外,所要求保护的主题可以使用标准编程和/或工程技术被实施成方法、装置或制造的物品,以产生软件、固件、硬件或其任何组合以控制计算机来实施所公开的主题。本文使用的术语“制造的物品”旨在涵盖从任何有形的计算机可读设备或介质可访问的计算机程序。

[0153] 计算机可读存储介质可以包括但不限于磁存储设备(例如,硬盘、软盘和磁条等)、光学盘(例如,光盘(CD)和数字通用盘(DVD)等)、智能卡和闪速存储器设备(例如,卡、棒和密钥驱动等)。计算机可读存储介质通常不包括计算机可读介质的所有实施方式,诸如信号本身。因此,相反地,通常计算机可读介质(即,不是计算机可读存储介质)可以附加地包括通信介质,诸如用于无线信号的传输介质等。

[0154] 前面部分中描述的过程、方法和算法中的每个可以被体现在由一个或多个计算机或计算机处理器执行的代码模块中,并且完全地或部分地由该代码模块自动化。代码模块可以被存储在任何类型的非暂态计算机可读介质或计算机存储设备上,诸如硬盘驱动、固态存储器、光学盘等。过程和算法可以部分地或全部地在专用电路装置中实施。所公开的过程和过程步骤的结果可以持久地或以其他方式被存储在任何类型的非暂态计算机存储设备(诸如易失性或非易失性存储设备)中。

[0155] 上述各种特征和过程可以彼此独立地使用,或者可以以各种方式组合。所有可能的组合和子组合都旨在落入本公开的范围。另外,在一些实施方式中可以省略某些方法或过程框。本文描述的方法和过程也不限于任何特定序列,并且与其相关的框或状态可以以合适的其他序列执行。例如,所描述的框或状态可以以不同于具体公开的顺序执行,或者多个框或状态可以被组合在单个框或状态中。可以串行、并行或以一些其他方式执行示例框或状态。可以向所公开的示例实施例添加框或状态,或从其中移除框或状态。本文描述的示例系统和组件可以与所描述的不同地被配置。例如,与所公开的示例实施例相比,可以添加、移除或重新布置元件。

[0156] 还应当理解,各种项目被图示为在使用时被存储在存储器中或存储设备上,并且出于存储器管理和数据完整性的目的,这些项目或其部分可以在存储器与其他存储设备之间传送。备选地,在其他实施例中,软件模块和/或系统中的一些或全部可以在另一设备上的存储器中执行并且经由计算机间通信与所图示的计算系统进行通信。另外,在一些实施

例中,可以以其他方式来实施或提供系统和/或模块中的一些或全部,诸如至少部分地以固件和/或硬件实施或提供,包括但不限于一个或多个专用集成电路(ASIC)、标准集成电路、控制器(例如,通过执行适当的指令并且包括微控制器和/或嵌入式控制器)、现场可编程门阵列(FPGA)、复杂可编程逻辑器件(CPLD)等。模块、系统和数据结构中的一些或全部也可以被存储在计算机可读介质(例如硬盘、存储器、网络或便携式介质物品)上(例如,作为软件指令或结构化数据),以由适当的驱动或经由适当的连接进行读取。系统、模块和数据结构还可以作为生成的数据信号(例如,作为载波或者其他模拟或数字传播信号的一部分)在各种计算机可读传输介质(包括基于无线的和基于有线/电缆的介质)上被传输,并且可以采用各种形式(例如,作为单个或多路复用模拟信号的一部分,或者作为多个离散数字分组或帧)。在其他实施例中,这些计算机程序产品还可以采用其他形式。因此,本发明可以利用其他计算机系统配置来实施。

[0157] 除非另外明确指出,或在如所使用的上下文中有其他理解,否则本文使用的条件性语言,诸如“能够”、“可以”、“可能”、“可”、“例如”等通常旨在传达:某些实施例包括而其他实施例不包括某些特征、元件和/或步骤。因此,这种条件性语言通常并不旨在暗示:特征、元件和/或步骤在任何情况下都是一个或多个实施例所必需的,或者一个或多个实施例在具有或没有用户输入或提示的情况下必须包括用于决定这些特征、元件和/或步骤是否被包括在任何特定实施例中或将在其中执行的逻辑。术语“包括”、“包含”、“具有”等是同义的并且以开放式方式被包括性地使用,并且不排除附加的元件、特征、动作、操作等。另外,术语“或”以其包括性意义(并且不以其排除性意义)被使用,使得在用于例如连接元件的列表时,术语“或”意指列表中的一个、一些或全部元件。

[0158] 虽然已经描述了某些示例实施例,但是这些实施例仅以示例的方式呈现,并且不旨在限制本文公开的发明的范围。因此,前面的描述中的任何内容都不旨在暗示任何特定的特征、特性、步骤、模块或框是必需的或必不可少的。实际上,本文描述的新方法和系统可以以各种其他形式体现;另外,在不脱离本文公开的本发明的精神的情况下,可以对本文描述的方法和系统的形式进行各种省略、替换和变化。所附权利要求及其等同物旨在覆盖落入本文公开的某些发明的范围和精神内的这样的形式或修改。

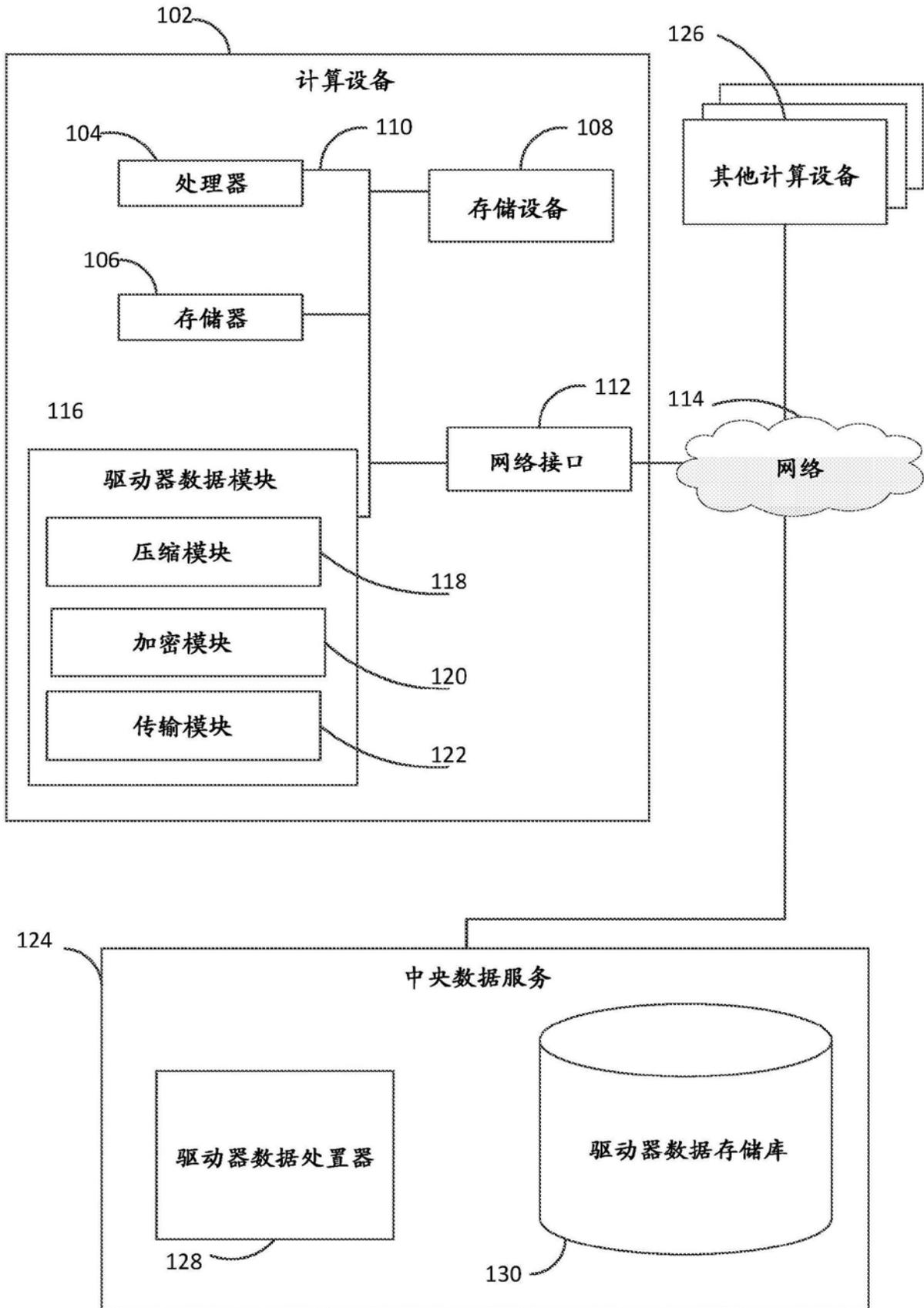


图1

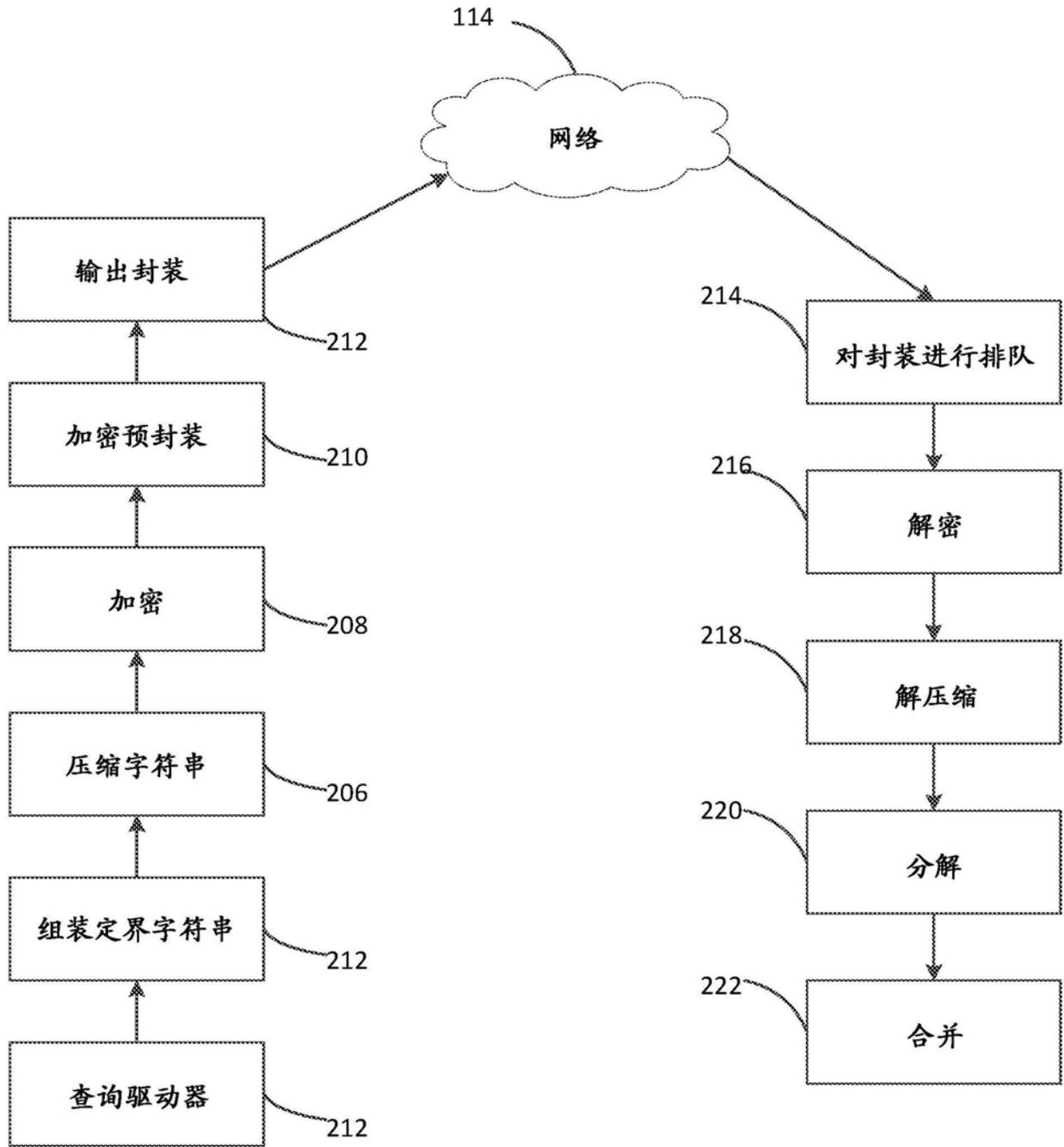


图2

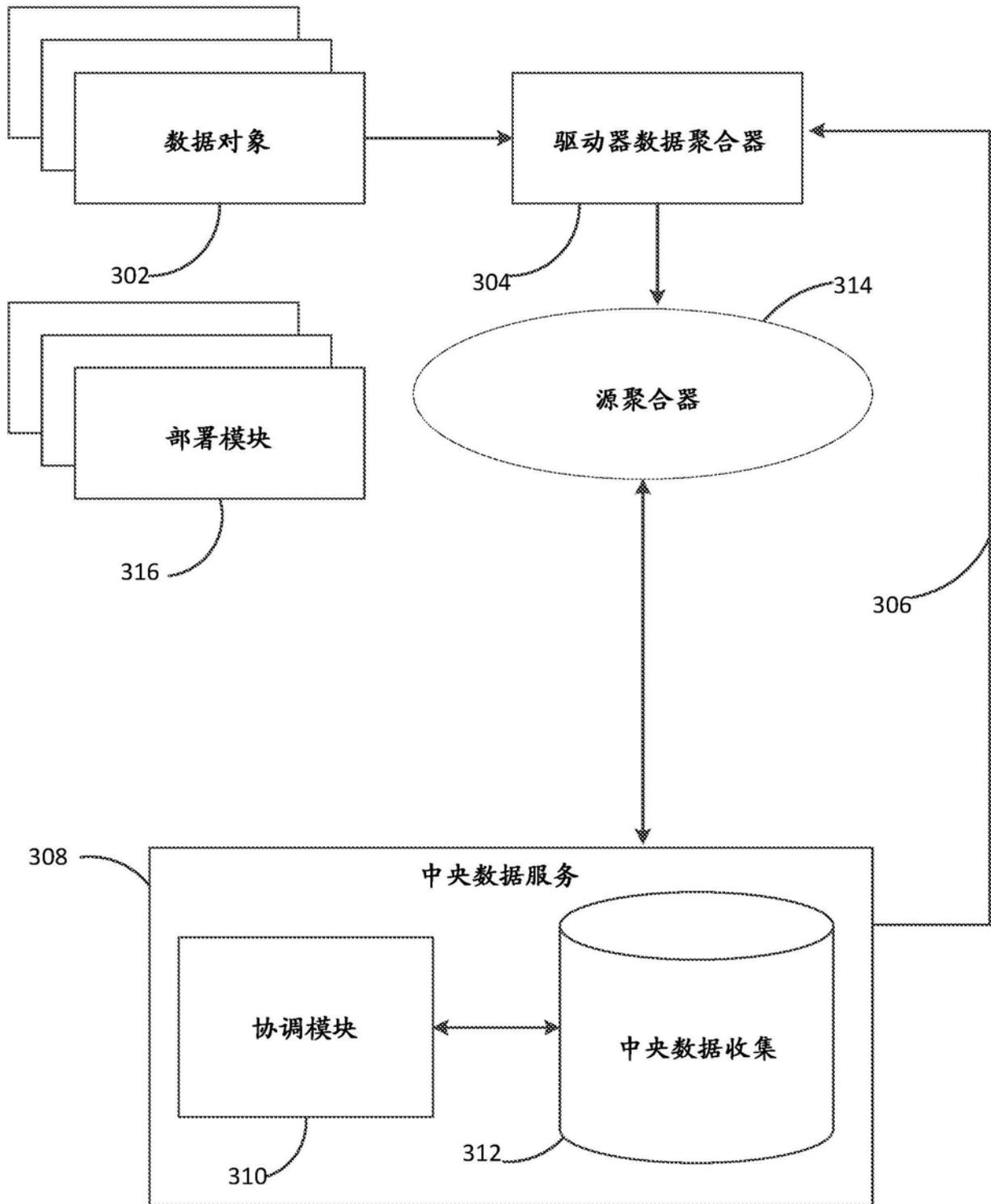


图3

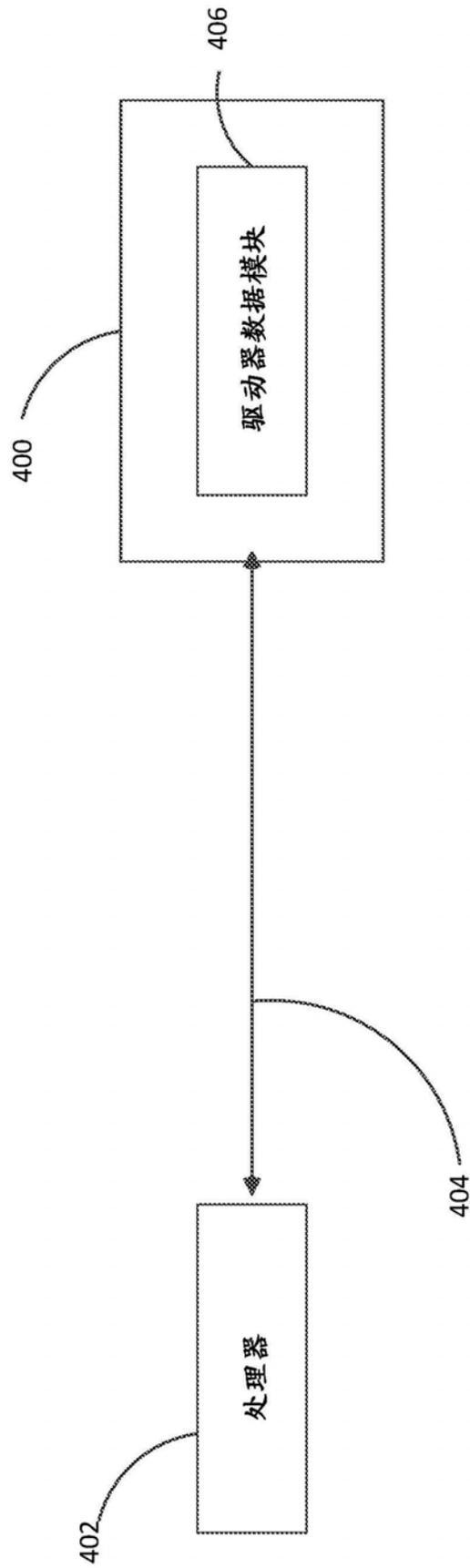


图4

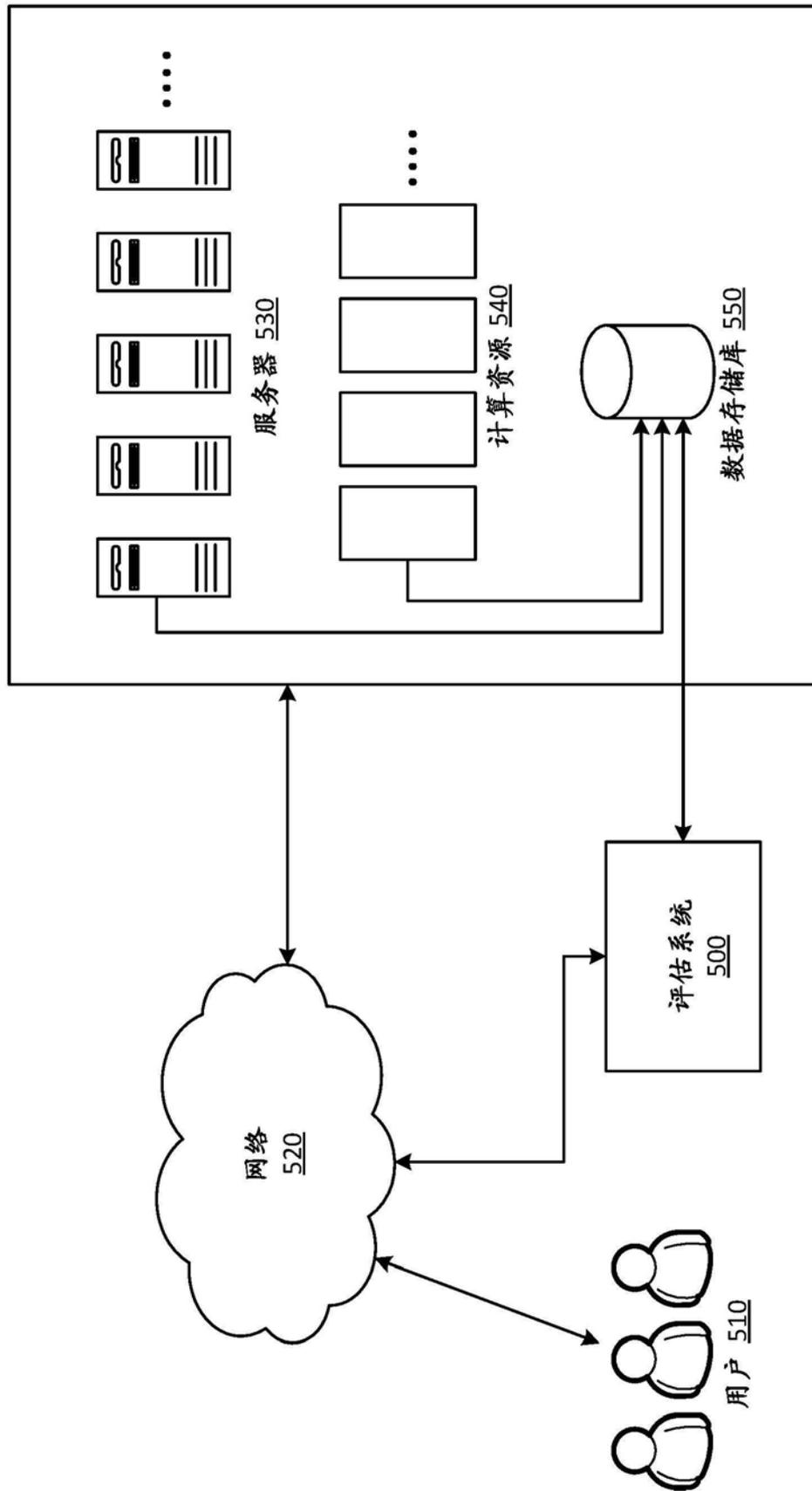


图5

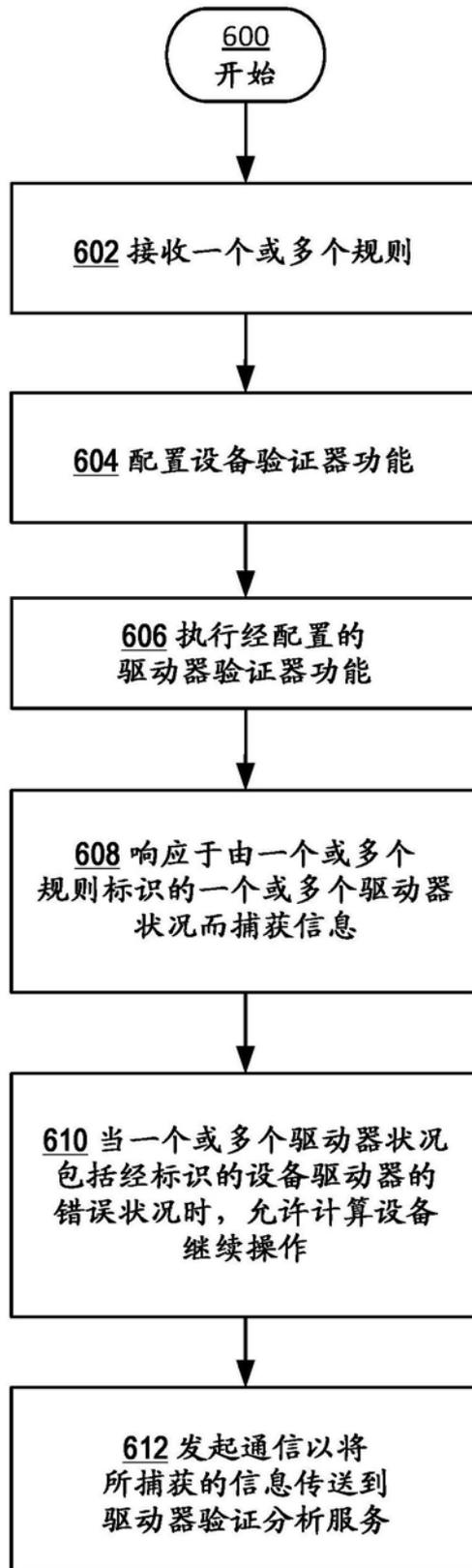


图6a

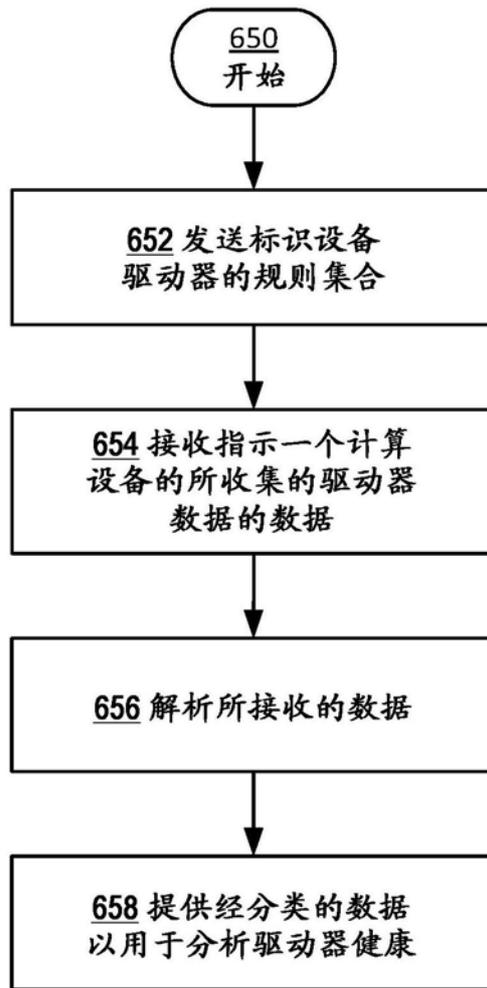


图6b

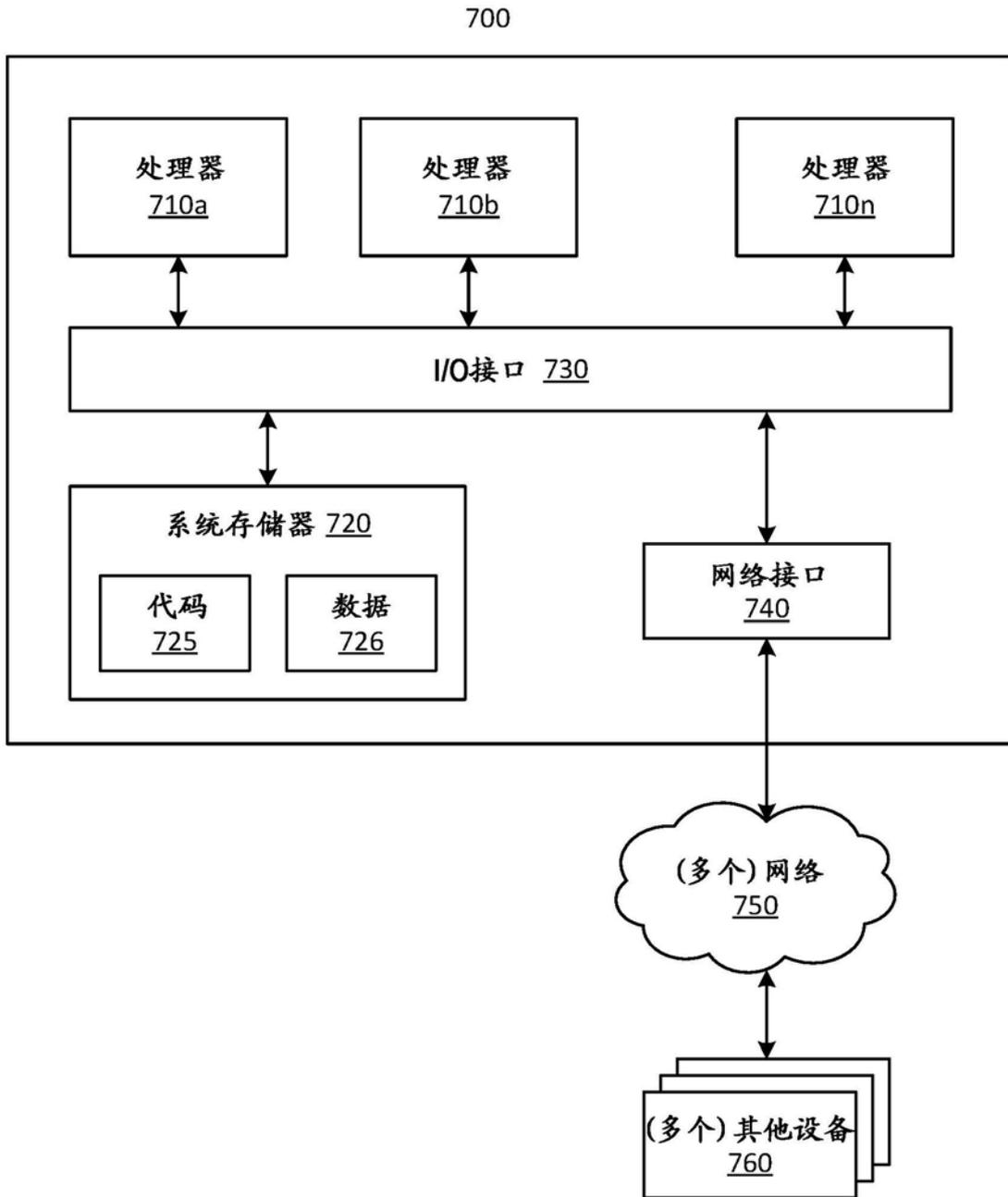


图7