



(12) 发明专利申请

(10) 申请公布号 CN 105630566 A

(43) 申请公布日 2016. 06. 01

(21) 申请号 201610109091. 1

(22) 申请日 2016. 02. 26

(71) 申请人 北京奇虎科技有限公司
地址 100088 北京市西城区新街口外大街
28号D座112室(德胜园区)
申请人 奇智软件(北京)有限公司

(72) 发明人 刘刚 陈佩翰

(74) 专利代理机构 北京智汇东方知识产权代理
事务所(普通合伙) 11391
代理人 康正德 孙晓芳

(51) Int. Cl.
G06F 9/445(2006. 01)
G06F 11/34(2006. 01)

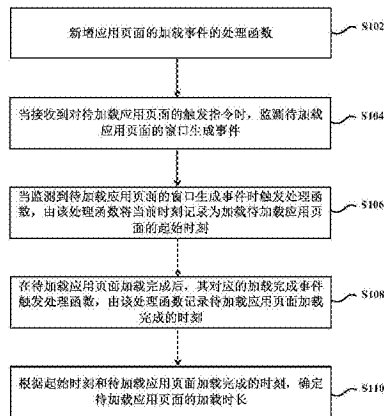
权利要求书2页 说明书11页 附图2页

(54) 发明名称

应用页面加载时长的确定方法及装置

(57) 摘要

本发明提供了一种应用页面加载时长的确定方法及装置。该方法包括:新增应用页面的加载事件的触发指令时,监测待加载应用页面的窗口生成事件;当监测到窗口生成事件时触发处理函数,由处理函数将当前时刻记录为加载待加载应用页面的起始时刻;在待加载应用页面加载完成后,其对应的加载完成事件触发处理函数,由处理函数记录待加载应用页面加载完成的时刻;根据起始时刻和待加载应用页面加载完成的时刻,确定待加载应用页面的加载时长。本发明实施例利用新增的处理函数来确定页面加载时长,解决了现有技术中通过人工手动计时确定页面加载时长效率低下的问题,提高测试效率和测试的准确性。



1. 一种应用页面加载时长的确定方法,包括:
 - 新增应用页面的加载事件的处理函数;
 - 当接收到对待加载应用页面的触发指令时,监测所述待加载应用页面的窗口生成事件;
 - 当监测到所述窗口生成事件时触发所述处理函数,由所述处理函数将当前时刻记录为加载所述待加载应用页面的起始时刻;
 - 在所述待加载应用页面加载完成后,其对应的加载完成事件触发所述处理函数,由所述处理函数记录所述待加载应用页面加载完成的时刻;
 - 根据所述起始时刻和所述待加载应用页面加载完成的时刻,确定所述待加载应用页面的加载时长。
2. 根据权利要求1所述的方法,其中,监测所述待加载应用页面的窗口生成事件,包括:
 - 监测生成所述待加载应用页面的窗口的onCreate方法;
 - 当监测到所述onCreate方法被回调时,确定监测到所述待加载应用页面的窗口生成事件。
3. 根据权利要求2所述的方法,其中,当监测到所述onCreate方法被回调时,确定监测到所述待加载应用页面的窗口生成事件,包括:
 - 当监测到所述onCreate方法被所述待加载应用页面的主线程中的mInstrumentation函数调用时,确定监测到所述待加载应用页面的窗口生成事件。
4. 根据权利要求1-3任一项所述的方法,其中,根据所述起始时刻和所述待加载应用页面加载完成的时刻,确定所述待加载应用页面的加载时长,包括:
 - 计算所述待加载应用页面加载完成的时刻与所述起始时刻的差值;
 - 将所述差值确定为所述待加载应用页面的加载时长。
5. 根据权利要求1-4任一项所述的方法,其中,确定所述待加载应用页面的加载时长之后,所述方法还包括:
 - 获取所述待加载应用页面的标识;
 - 在所述待加载应用页面的标识与加载时长之间建立关联关系,并将所述关联关系存储在指定数据库中。
6. 根据权利要求5所述的方法,其中,将所述关联关系存储在指定数据库中之后,所述方法还包括:
 - 在所述关联关系中查找指定应用页面对应的多个加载时长;
 - 根据查找到的所述多个加载时长,计算所述指定应用页面的加载性能。
7. 根据权利要求5所述的方法,其中,当所述待加载应用页面的标识中包含所述待加载应用页面对应应用的版本标识时,将所述关联关系存储在指定数据库中之后,所述方法还包括:
 - 在所述关联关系中查找指定应用页面对应的多个加载时长;
 - 在所述多个加载时长中分类出不同应用版本下的加载时长;
 - 根据分类出的所述不同应用版本下的加载时长,计算所述指定应用页面在不同应用版本下的加载性能。
8. 根据权利要求5所述的方法,其中,将所述关联关系存储在指定数据库中之后,所述

方法还包括：

确定指定应用中的多个应用页面；

在所述关联关系中查找所述多个应用页面各自对应的加载时长；

根据查找到的所述多个应用页面各自对应的加载时长，计算所述指定应用的页面加载性能。

9. 一种应用页面加载时长的确定装置，包括：

写入模块，适于新增应用页面的加载事件的处理函数；

监测模块，适于当接收到对待加载应用页面的触发指令时，监测所述待加载应用页面的窗口生成事件；

第一记录模块，适于当监测到所述窗口生成事件时触发所述处理函数，由所述处理函数将当前时刻记录为加载所述待加载应用页面的起始时刻；

第二记录模块，适于在所述待加载应用页面加载完成后，其对应的加载完成事件触发所述处理函数，由所述处理函数记录所述待加载应用页面加载完成的时刻；

确定模块，适于根据所述起始时刻和所述待加载应用页面加载完成的时刻，确定所述待加载应用页面的加载时长。

10. 根据权利要求9所述的装置，其中，所述监测模块还适于：

监测生成所述待加载应用页面的窗口的onCreate方法；

当监测到所述onCreate方法被回调时，确定监测到所述待加载应用页面的窗口生成事件。

应用页面加载时长的确定方法及装置

技术领域

[0001] 本发明涉及计算机技术领域,特别是一种应用页面加载时长的确定方法及装置。

背景技术

[0002] 随着计算机以及移动终端技术的不断发展,越来越多的基于电子终端(如手机、平板、智能手表、智能眼镜等)的应用正在极大地丰富着人们的生活。同时,用户对应用的体验要求也变得越来越,其中,页面加载时长是用户体验较为重要的一项指标。

[0003] 目前,开发者主要通过人工测试的方法确定应用页面加载时长,具体地,测试人员对应用进行手动操作,例如手动点击启动应用、点击应用中的某页面等。同时,测试人员利用计时器记录手动点击的时刻和应用的页面加载完成的时刻,计算手动点击的时刻和页面加载完成的时刻的差值,将差值作为应用的页面加载时长,并且需要人工记录数据。

[0004] 由此可见,人工测试的方法会消耗大量的人力资源,测试效率低且准确率不高,亟需解决这一问题。

发明内容

[0005] 鉴于上述问题,提出了本发明以便提供一种克服上述问题或者至少部分地解决上述问题的应用页面加载时长的确定方法及相应的装置。

[0006] 依据本发明的一方面,提供了一种应用页面加载时长的确定方法,包括:

[0007] 新增应用页面的加载事件的处理函数;

[0008] 当接收到对待加载应用页面的触发指令时,监测所述待加载应用页面的窗口生成事件;

[0009] 当监测到所述窗口生成事件时触发所述处理函数,由所述处理函数将当前时刻记录为加载所述待加载应用页面的起始时刻;

[0010] 在所述待加载应用页面加载完成后,其对应的加载完成事件触发所述处理函数,由所述处理函数记录所述待加载应用页面加载完成的时刻;

[0011] 根据所述起始时刻和所述待加载应用页面加载完成的时刻,确定所述待加载应用页面的加载时长。

[0012] 可选地,监测所述待加载应用页面的窗口生成事件,包括:

[0013] 监测生成所述待加载应用页面的窗口的onCreate方法;

[0014] 当监测到所述onCreate方法被回调时,确定监测到所述待加载应用页面的窗口生成事件。

[0015] 可选地,当监测到所述onCreate方法被回调时,确定监测到所述待加载应用页面的窗口生成事件,包括:

[0016] 当监测到所述onCreate方法被所述待加载应用页面的主线程中的mInstrumentation函数调用时,确定监测到所述待加载应用页面的窗口生成事件。

[0017] 可选地,根据所述起始时刻和所述待加载应用页面加载完成的时刻,确定所述待

加载应用页面的加载时长,包括:

[0018] 计算所述待加载应用页面加载完成的时刻与所述起始时刻的差值;

[0019] 将所述差值确定为所述待加载应用页面的加载时长。

[0020] 可选地,确定所述待加载应用页面的加载时长之后,所述方法还包括:

[0021] 获取所述待加载应用页面的标识;

[0022] 在所述待加载应用页面的标识与加载时长之间建立关联关系,并将所述关联关系存储在指定数据库中。

[0023] 可选地,将所述关联关系存储在指定数据库中之后,所述方法还包括:

[0024] 在所述关联关系中查找指定应用页面对应的多个加载时长;

[0025] 根据查找到的所述多个加载时长,计算所述指定应用页面的加载性能。

[0026] 可选地,当所述待加载应用页面的标识中包含所述待加载应用页面对应应用的版本标识时,将所述关联关系存储在指定数据库中之后,所述方法还包括:

[0027] 在所述关联关系中查找指定应用页面对应的多个加载时长;

[0028] 在所述多个加载时长中分类出不同应用版本下的加载时长;

[0029] 根据分类出的所述不同应用版本下的加载时长,计算所述指定应用页面在不同应用版本下的加载性能。

[0030] 可选地,将所述关联关系存储在指定数据库中之后,所述方法还包括:

[0031] 确定指定应用中的多个应用页面;

[0032] 在所述关联关系中查找所述多个应用页面各自对应的加载时长;

[0033] 根据查找到的所述多个应用页面各自对应的加载时长,计算所述指定应用的页面加载性能。

[0034] 依据本发明的另一方面,还提供了一种应用页面加载时长的确定装置,包括:

[0035] 写入模块,适于新增应用页面的加载事件的处理函数;

[0036] 监测模块,适于当接收到对待加载应用页面的触发指令时,监测所述待加载应用页面的窗口生成事件;

[0037] 第一记录模块,适于当监测到所述窗口生成事件时触发所述处理函数,由所述处理函数将当前时刻记录为加载所述待加载应用页面的起始时刻;

[0038] 第二记录模块,适于在所述待加载应用页面加载完成后,其对应的加载完成事件触发所述处理函数,由所述处理函数记录所述待加载应用页面加载完成的时刻;

[0039] 确定模块,适于根据所述起始时刻和所述待加载应用页面加载完成的时刻,确定所述待加载应用页面的加载时长。

[0040] 可选地,所述监测模块还适于:

[0041] 监测生成所述待加载应用页面的窗口的onCreate方法;

[0042] 当监测到所述onCreate方法被回调时,确定监测到所述待加载应用页面的窗口生成事件。

[0043] 可选地,所述监测模块还适于:

[0044] 当监测到所述onCreate方法被所述待加载应用页面的主线程中的mInstrumentation函数调用时,确定监测到所述待加载应用页面的窗口生成事件。

[0045] 可选地,所述确定模块还适于:

- [0046] 计算所述待加载应用页面加载完成的时刻与所述起始时刻的差值；
- [0047] 将所述差值确定为所述待加载应用页面的加载时长。
- [0048] 可选地,所述装置还包括:
- [0049] 建立模块,适于在所述确定模块确定所述待加载应用页面的加载时长之后,获取所述待加载应用页面的标识;在所述待加载应用页面的标识与加载时长之间建立关联关系,并将所述关联关系存储在指定数据库中。
- [0050] 可选地,所述装置还包括:
- [0051] 第一计算模块,适于在所述建立模块将所述关联关系存储在指定数据库中之后,在所述关联关系中查找指定应用页面对应的多个加载时长;根据查找到的所述多个加载时长,计算所述指定应用页面的加载性能。
- [0052] 可选地,所述装置还包括:
- [0053] 第二计算模块,适于当所述待加载应用页面的标识中包含所述待加载应用页面对应应用的版本标识时,在所述建立模块将所述关联关系存储在指定数据库中之后,在所述关联关系中查找指定应用页面对应的多个加载时长;在所述多个加载时长中分类出不同应用版本下的加载时长;根据分类出的所述不同应用版本下的加载时长,计算所述指定应用页面在不同应用版本下的加载性能。
- [0054] 可选地,所述装置还包括:
- [0055] 第三计算模块,适于在所述建立模块将所述关联关系存储在指定数据库中之后,确定指定应用中的多个应用页面;在所述关联关系中查找所述多个应用页面各自对应的加载时长;根据查找到的所述多个应用页面各自对应的加载时长,计算所述指定应用的页面加载性能。
- [0056] 本发明实施例新增应用页面的加载事件的处理函数,当监测到待加载应用页面的窗口生成事件时触发该处理函数,由该处理函数将当前时刻记录为加载待加载应用页面的起始时刻;以及在待加载应用页面加载完成后,其对应的加载完成事件触发该处理函数,由该处理函数记录待加载应用页面加载完成的时刻,由此根据起始时刻和待加载应用页面加载完成的时刻,确定待加载应用页面的加载时长。可见,本发明实施例利用新增的处理函数来确定页面加载时长,解决了现有技术中通过人工手动计时确定页面加载时长效率低下的问题,节省了人力资源,提高测试效率和测试的准确性。
- [0057] 上述说明仅是本发明技术方案的概述,为了能够更清楚了解本发明的技术手段,而可依照说明书的内容予以实施,并且为了让本发明的上述和其它目的、特征和优点能够更明显易懂,以下特举本发明的具体实施方式。
- [0058] 根据下文结合附图对本发明具体实施例的详细描述,本领域技术人员将会更加明了本发明的上述以及其他目的、优点和特征。

附图说明

- [0059] 通过阅读下文优选实施方式的详细描述,各种其他的优点和益处对于本领域普通技术人员将变得清楚明了。附图仅用于示出优选实施方式的目的,而并不认为是对本发明的限制。而且在整个附图中,用相同的参考符号表示相同的部件。在附图中:
- [0060] 图1示出了根据本发明一实施例的应用页面加载时长的确定方法的流程图;

[0061] 图2示出了根据本发明一个实施例的应用页面加载时长的确定装置的结构示意图;以及

[0062] 图3示出了根据本发明另一个实施例的应用页面加载时长的确定装置的结构示意图。

具体实施方式

[0063] 下面将参照附图更详细地描述本公开的示例性实施例。虽然附图中显示了本公开的示例性实施例,然而应当理解,可以以各种形式实现本公开而不应被这里阐述的实施例所限制。相反,提供这些实施例是为了能够更透彻地理解本公开,并且能够将本公开的范围完整的传达给本领域的技术人员。

[0064] 为解决上述技术问题,本发明实施例提供了一种应用页面加载时长的确定方法。图1示出了根据本发明一实施例的应用页面加载时长的确定方法的流程图。如图1所示,该方法至少包括以下步骤S102至步骤S110。

[0065] 步骤S102,新增应用页面的加载事件的处理函数。

[0066] 步骤S104,当接收到对待加载应用页面的触发指令时,监测待加载应用页面的窗口生成事件。

[0067] 步骤S106,当监测到待加载应用页面的窗口生成事件时触发处理函数,由该处理函数将当前时刻记录为加载待加载应用页面的起始时刻。

[0068] 步骤S108,在待加载应用页面加载完成后,其对应的加载完成事件触发处理函数,由该处理函数记录待加载应用页面加载完成的时刻。

[0069] 步骤S110,根据起始时刻和待加载应用页面加载完成的时刻,确定待加载应用页面的加载时长。

[0070] 本发明实施例新增应用页面的加载事件的处理函数,当监测到待加载应用页面的窗口生成事件时触发该处理函数,由该处理函数将当前时刻记录为加载待加载应用页面的起始时刻;以及在待加载应用页面加载完成后,其对应的加载完成事件触发该处理函数,由该处理函数记录待加载应用页面加载完成的时刻,由此根据起始时刻和待加载应用页面加载完成的时刻,确定待加载应用页面的加载时长。可见,本发明实施例利用新增的处理函数来确定页面加载时长,解决了现有技术中通过人工手动计时确定页面加载时长效率低下的问题,节省了人力资源,提高测试效率和测试的准确性。

[0071] 一般来说,应用程序的外表是通过Activity来展示的,Activity是一个应用程序组件,提供一个屏幕,用户可以用来交互为了完成某项任务。首先,需要创建Activity,在安卓系统中,一个应用的入口函数是ActivityThread类的main函数,ActivityThread类中有一个handleLaunchActivity函数,它就是创建Activity的地方。在创建Activity后,调用Activity的onCreate方法就开始了Activity的生命周期。基于此,在上文步骤S104中监测待加载应用页面的窗口生成事件,本发明实施例提供了一种可选的方案,在该方案中,可以监测生成待加载应用页面的窗口的onCreate方法,当监测到该onCreate方法被回调时,确定监测到待加载应用页面的窗口生成事件。

[0072] 在安卓系统中,通过主线程中的mInstrumentation.callActivityOnCreate函数来调用Activity的onCreate函数。因而,在本发明一实施例中,当监测到onCreate方法被回

调时,确定监测到待加载应用页面的窗口生成事件,本发明实施例提供了一种可选的方案,即,当监测到onCreate方法被待加载应用页面的主线程中的mInstrumentation函数调用时,确定监测到待加载应用页面的窗口生成事件。

[0073] 在安卓系统默认情况下,待加载应用页面的加载完成事件不会触发任何处理过程,在本发明实施例中,可以定义应用页面的加载事件的处理函数,由该处理函数记录待加载应用页面加载完成的时刻。在其他实施例中,新增的应用页面的加载事件的处理函数可以为第一处理函数和第二处理函数,在步骤S106中当监测到待加载应用页面的窗口生成事件时触发第一处理函数,由第一处理函数将当前时刻记录为加载待加载应用页面的起始时刻。在步骤S108在待加载应用页面加载完成后,其对应的加载完成事件触发第二处理函数,由第二处理函数记录待加载应用页面加载完成的时刻。

[0074] 在上文步骤S110根据起始时刻和待加载应用页面加载完成的时刻,确定待加载应用页面的加载时长,本发明实施例提供了一种可选的方案,在该方案中,计算待加载应用页面加载完成的时刻与起始时刻的差值,将该差值确定为待加载应用页面的加载时长。例如,加载待加载应用页面的起始时刻为 t_1 ,待加载应用页面加载完成的时刻为 t_2 ,则可以将 t_2-t_1 确定为待加载应用页面的加载时长。

[0075] 为了便于分析页面加载性能,以改善应用的页面加载时长,在确定待加载应用页面的加载时长之后,可以获取待加载应用页面的标识,随后在待加载应用页面的标识与加载时长之间建立关联关系,并将关联关系存储在指定数据库中,以便后续查找使用。进一步,在分析页面加载性能时,本发明实施例提供了多种分析页面加载性能的方法,下面将分别进行详细介绍。

[0076] 方法一,针对指定应用页面被多次加载的情形。可以在关联关系中查找指定应用页面对应的多个加载时长,随后根据查找到的多个加载时长,计算指定应用页面的加载性能。具体地,可以设定一指定阈值,若查找到的多个加载时长均大于该设定的指定阈值,则认为该指定应用页面加载时长较长,则需要提示开发人员改善该指定应用页面加载时长。若查找到的多个加载时长中的一个或几个大于该指定阈值,则认为可能是网络连接或其他问题导致的,可以暂且不用关注。

[0077] 方法二,针对不同应用版本下的指定应用页面。待加载应用页面的标识中包含待加载应用页面对应应用的版本标识,可以在关联关系中查找指定应用页面对应的多个加载时长,随后在多个加载时长中分类出不同应用版本下的加载时长,进而根据分类出的不同应用版本下的加载时长,计算指定应用页面在不同应用版本下的加载性能。具体地,可以比较指定应用页面在不同应用版本下的加载时长,例如,A应用版本下的指定应用页面加载时长为 t_{11} ,B应用版本下的指定应用页面加载时长为 t_{22} ,若 t_{22} 大于 t_{11} ,则认为B应用版本下的指定应用页面加载时长较长,则需要提示开发人员改善B应用版本下的该指定应用页面加载时长。又例如,设定一指定阈值,A应用版本下的指定应用页面加载时长为 t_{11} ,B应用版本下的指定应用页面加载时长为 t_{22} ,若 t_{22} 大于 t_{11} ,然而 t_{11} 和 t_{22} 均小于该指定阈值,则可以暂且不用关注。

[0078] 方法三,针对指定应用中的多个应用页面的情形。可以确定指定应用中的多个应用页面,随后在关联关系中查找多个应用页面各自对应的加载时长,进而根据查找到的多个应用页面各自对应的加载时长,计算指定应用的页面加载性能。具体地,可以设定一指定

阈值,将多个应用页面各自对应的加载时长与该指定阈值进行比较,若其中的多个均大于该指定阈值,则认为该指定应用的页面加载时长较长,则需要提示开发人员改善该指定应用的页面加载时长。

[0079] 需要说明的是,在实际应用中,上述所有可选实施方式可以采用结合的方式任意组合,形成本发明的可选实施例,在此不再一一赘述。

[0080] 基于同一发明构思,本发明实施例还提供一种应用页面加载时长的确定装置,图2示出了根据本发明一个实施例的应用页面加载时长的确定装置的结构示意图。如图2所示,该装置至少可以包括写入模块210、监测模块220、第一记录模块230、第二记录模块240以及确定模块250。

[0081] 现介绍本发明实施例的应用页面加载时长的确定装置的各组成或器件的功能以及各部分间的连接关系:

[0082] 写入模块210,适于新增应用页面的加载事件的处理函数;

[0083] 监测模块220,适于当接收到对待加载应用页面的触发指令时,监测待加载应用页面的窗口生成事件;

[0084] 第一记录模块230,与写入模块210和监测模块220相耦合,适于当监测到窗口生成事件时触发处理函数,由处理函数将当前时刻记录为加载待加载应用页面的起始时刻;

[0085] 第二记录模块240,与写入模块210和第一记录模块230相耦合,适于在待加载应用页面加载完成后,其对应的加载完成事件触发处理函数,由处理函数记录待加载应用页面加载完成的时刻;

[0086] 确定模块250,与第一记录模块230和第二记录模块240相耦合,适于根据起始时刻和待加载应用页面加载完成的时刻,确定待加载应用页面的加载时长。

[0087] 一般来说,应用程序的外表是通过Activity来展示的,Activity是一个应用程序组件,提供一个屏幕,用户可以用来交互为了完成某项任务。首先,需要创建Activity,在安卓系统中,一个应用的入口函数是ActivityThread类的主函数,ActivityThread类中有一个handleLaunchActivity函数,它就是创建Activity的地方。在创建Activity后,调用Activity的onCreate方法就开始了Activity的生命周期。基于此,监测模块220还适于:监测生成待加载应用页面的窗口的onCreate方法;当监测到onCreate方法被回调时,确定监测到待加载应用页面的窗口生成事件。

[0088] 在安卓系统中,通过主线程中的mInstrumentation.callActivityOnCreate函数来调用Activity的onCreate函数。因而,监测模块220还适于:当监测到onCreate方法被待加载应用页面的主线程中的mInstrumentation函数调用时,确定监测到待加载应用页面的窗口生成事件。

[0089] 在安卓系统默认情况下,待加载应用页面的加载完成事件不会触发任何处理过程,在本发明实施例中,可以定义应用页面的加载事件的处理函数,由该处理函数记录待加载应用页面加载完成的时刻。在其他实施例中,新增的应用页面的加载事件的处理函数可以为第一处理函数和第二处理函数,第一记录模块230还适于当监测到待加载应用页面的窗口生成事件时触发第一处理函数,由第一处理函数将当前时刻记录为加载待加载应用页面的起始时刻。第二记录模块240还适于在待加载应用页面加载完成后,其对应的加载完成事件触发第二处理函数,由第二处理函数记录待加载应用页面加载完成的时刻。

[0090] 在本发明一实施例中,确定模块250还适于:计算待加载应用页面加载完成的时刻与起始时刻的差值;将差值确定为待加载应用页面的加载时长。例如,加载待加载应用页面的起始时刻为 t_1 ,待加载应用页面加载完成的时刻为 t_2 ,则可以将 t_2-t_1 确定为待加载应用页面的加载时长。

[0091] 在本发明一实施例中,为了便于分析页面加载性能,以改善应用的页面加载时长,如图3所示,图2展示的装置还可以包括建立模块260,与确定模块250相耦合,适于在确定模块250确定待加载应用页面的加载时长之后,获取待加载应用页面的标识;在待加载应用页面的标识与加载时长之间建立关联关系,并将关联关系存储在指定数据库中。

[0092] 在本发明一实施例中,图2或图3展示的装置还可以包括:

[0093] 第一计算模块(附图中未示出),与建立模块260相耦合,适于在建立模块260将关联关系存储在指定数据库中之后,在关联关系中查找指定应用页面对应的多个加载时长;根据查找到的多个加载时长,计算指定应用页面的加载性能。

[0094] 在本发明一实施例中,图2或图3展示的装置还可以包括:

[0095] 第二计算模块(附图中未示出),与建立模块260相耦合,适于当待加载应用页面的标识中包含待加载应用页面对应应用的版本标识时,在建立模块260将关联关系存储在指定数据库中之后,在关联关系中查找指定应用页面对应的多个加载时长;在多个加载时长中分类出不同应用版本下的加载时长;根据分类出的不同应用版本下的加载时长,计算指定应用页面在不同应用版本下的加载性能。

[0096] 在本发明一实施例中,图2或图3展示的装置还可以包括:

[0097] 第三计算模块(附图中未示出),与建立模块260相耦合,适于在建立模块260将关联关系存储在指定数据库中之后,确定指定应用中的多个应用页面;在关联关系中查找多个应用页面各自对应的加载时长;根据查找到的多个应用页面各自对应的加载时长,计算指定应用的页面加载性能。

[0098] 根据上述任意一个优选实施例或多个优选实施例的组合,本发明实施例能够达到如下有益效果:

[0099] 本发明实施例新增应用页面的加载事件的处理函数,当监测到待加载应用页面的窗口生成事件时触发该处理函数,由该处理函数将当前时刻记录为加载待加载应用页面的起始时刻;以及在待加载应用页面加载完成后,其对应的加载完成事件触发该处理函数,由该处理函数记录待加载应用页面加载完成的时刻,由此根据起始时刻和待加载应用页面加载完成的时刻,确定待加载应用页面的加载时长。可见,本发明实施例利用新增的处理函数来确定页面加载时长,解决了现有技术中通过人工手动计时确定页面加载时长效率低下的问题,节省了人力资源,提高测试效率和测试的准确性。

[0100] 在此处所提供的说明书中,说明了大量具体细节。然而,能够理解,本发明的实施例可以在没有这些具体细节的情况下实践。在一些实例中,并未详细示出公知的方法、结构和技术,以便不模糊对本说明书的理解。

[0101] 类似地,应当理解,为了精简本公开并帮助理解各个发明方面中的一个或多个,在上面对本发明的示例性实施例的描述中,本发明的各个特征有时被一起分组到单个实施例、图、或者对其的描述中。然而,并不应将该公开的方法解释成反映如下意图:即所要求保护的本发明要求比在每个权利要求中所明确记载的特征更多的特征。更确切地说,如下面

的权利要求书所反映的那样,发明方面在于少于前面公开的单个实施例的所有特征。因此,遵循具体实施方式的权利要求书由此明确地并入该具体实施方式,其中每个权利要求本身都作为本发明的单独实施例。

[0102] 本领域那些技术人员可以理解,可以对实施例中的设备中的模块进行自适应性地改变并且把它们设置在与该实施例不同的一个或多个设备中。可以把实施例中的模块或单元或组件组合成一个模块或单元或组件,以及此外可以把它分成多个子模块或子单元或子组件。除了这样的特征和/或过程或者单元中的至少一些是相互排斥之外,可以采用任何组合对本说明书(包括伴随的权利要求、摘要和附图)中公开的所有特征以及如此公开的任何方法或者设备的所有过程或单元进行组合。除非另外明确陈述,本说明书(包括伴随的权利要求、摘要和附图)中公开的每个特征可以由提供相同、等同或相似目的的替代特征来代替。

[0103] 此外,本领域的技术人员能够理解,尽管在此所述的一些实施例包括其它实施例中包括的某些特征而不是其它特征,但是不同实施例的特征的组合意味着处于本发明的范围之内并且形成不同的实施例。例如,在权利要求书中,所要求保护的实施例的任意之一都可以以任意的组合方式来使用。

[0104] 本发明的各个部件实施例可以以硬件实现,或者以在一个或者多个处理器上运行的软件模块实现,或者以它们的组合实现。本领域的技术人员应当理解,可以在实践中使用微处理器或者数字信号处理器(DSP)来实现根据本发明实施例的应用页面加载时长的确定装置中的一些或者全部部件的一些或者全部功能。本发明还可以实现为用于执行这里所描述的方法的一部分或者全部的设备或者装置程序(例如,计算机程序和计算机程序产品)。这样的实现本发明的程序可以存储在计算机可读介质上,或者可以具有一个或者多个信号的形式。这样的信号可以从因特网网站上下载得到,或者在载体信号上提供,或者以任何其他形式提供。

[0105] 应该注意的是上述实施例对本发明进行说明而不是对本发明进行限制,并且本领域技术人员在不脱离所附权利要求的范围的情况下可设计出替换实施例。在权利要求中,不应将位于括号之间的任何参考符号构造成对权利要求的限制。单词“包含”不排除存在未列在权利要求中的元件或步骤。位于元件之前的单词“一”或“一个”不排除存在多个这样的元件。本发明可以借助于包括有若干不同元件的硬件以及借助于适当编程的计算机来实现。在列举了若干装置的单元权利要求中,这些装置中的若干个可以是通过同一个硬件项来具体体现。单词第一、第二、以及第三等的使用不表示任何顺序。可将这些单词解释为名称。

[0106] 至此,本领域技术人员应认识到,虽然本文已详尽示出和描述了本发明的多个示例性实施例,但是,在不脱离本发明精神和范围的情况下,仍可根据本发明公开的内容直接确定或推导出符合本发明原理的许多其他变型或修改。因此,本发明的范围应被理解和认定为覆盖了所有这些其他变型或修改。

[0107] 本发明实施例还公开了:A1、一种应用页面加载时长的确定方法,包括:

[0108] 新增应用页面的加载事件的处理函数;

[0109] 当接收到对待加载应用页面的触发指令时,监测所述待加载应用页面的窗口生成事件;

- [0110] 当监测到所述窗口生成事件时触发所述处理函数,由所述处理函数将当前时刻记录为加载所述待加载应用页面的起始时刻;
- [0111] 在所述待加载应用页面加载完成后,其对应的加载完成事件触发所述处理函数,由所述处理函数记录所述待加载应用页面加载完成的时刻;
- [0112] 根据所述起始时刻和所述待加载应用页面加载完成的时刻,确定所述待加载应用页面的加载时长。
- [0113] A2、根据A1所述的方法,其中,监测所述待加载应用页面的窗口生成事件,包括:
- [0114] 监测生成所述待加载应用页面的窗口的onCreate方法;
- [0115] 当监测到所述onCreate方法被回调时,确定监测到所述待加载应用页面的窗口生成事件。
- [0116] A3、根据A2所述的方法,其中,当监测到所述onCreate方法被回调时,确定监测到所述待加载应用页面的窗口生成事件,包括:
- [0117] 当监测到所述onCreate方法被所述待加载应用页面的主线程中的mInstrumentation函数调用时,确定监测到所述待加载应用页面的窗口生成事件。
- [0118] A4、根据A1-A3任一项所述的方法,其中,根据所述起始时刻和所述待加载应用页面加载完成的时刻,确定所述待加载应用页面的加载时长,包括:
- [0119] 计算所述待加载应用页面加载完成的时刻与所述起始时刻的差值;
- [0120] 将所述差值确定为所述待加载应用页面的加载时长。
- [0121] A5、根据A1-A4任一项所述的方法,其中,确定所述待加载应用页面的加载时长之后,所述方法还包括:
- [0122] 获取所述待加载应用页面的标识;
- [0123] 在所述待加载应用页面的标识与加载时长之间建立关联关系,并将所述关联关系存储在指定数据库中。
- [0124] A6、根据A5所述的方法,其中,将所述关联关系存储在指定数据库中之后,所述方法还包括:
- [0125] 在所述关联关系中查找指定应用页面对应的多个加载时长;
- [0126] 根据查找到的所述多个加载时长,计算所述指定应用页面的加载性能。
- [0127] A7、根据A5所述的方法,其中,当所述待加载应用页面的标识中包含所述待加载应用页面对应应用的版本标识时,将所述关联关系存储在指定数据库中之后,所述方法还包括:
- [0128] 在所述关联关系中查找指定应用页面对应的多个加载时长;
- [0129] 在所述多个加载时长中分类出不同应用版本下的加载时长;
- [0130] 根据分类出的所述不同应用版本下的加载时长,计算所述指定应用页面在不同应用版本下的加载性能。
- [0131] A8、根据A5所述的方法,其中,将所述关联关系存储在指定数据库中之后,所述方法还包括:
- [0132] 确定指定应用中的多个应用页面;
- [0133] 在所述关联关系中查找所述多个应用页面各自对应的加载时长;
- [0134] 根据查找到的所述多个应用页面各自对应的加载时长,计算所述指定应用的页面

加载性能。

[0135] B9、一种应用页面加载时长的确定装置,包括:

[0136] 写入模块,适于新增应用页面的加载事件的处理函数;

[0137] 监测模块,适于当接收到对待加载应用页面的触发指令时,监测所述待加载应用页面的窗口生成事件;

[0138] 第一记录模块,适于当监测到所述窗口生成事件时触发所述处理函数,由所述处理函数将当前时刻记录为加载所述待加载应用页面的起始时刻;

[0139] 第二记录模块,适于在所述待加载应用页面加载完成后,其对应的加载完成事件触发所述处理函数,由所述处理函数记录所述待加载应用页面加载完成的时刻;

[0140] 确定模块,适于根据所述起始时刻和所述待加载应用页面加载完成的时刻,确定所述待加载应用页面的加载时长。

[0141] B10、根据B9所述的装置,其中,所述监测模块还适于:

[0142] 监测生成所述待加载应用页面的窗口的onCreate方法;

[0143] 当监测到所述onCreate方法被回调时,确定监测到所述待加载应用页面的窗口生成事件。

[0144] B11、根据B10所述的装置,其中,所述监测模块还适于:

[0145] 当监测到所述onCreate方法被所述待加载应用页面的主线程中的mInstrumentation函数调用时,确定监测到所述待加载应用页面的窗口生成事件。

[0146] B12、根据B9-B11任一项所述的装置,其中,所述确定模块还适于:

[0147] 计算所述待加载应用页面加载完成的时刻与所述起始时刻的差值;

[0148] 将所述差值确定为所述待加载应用页面的加载时长。

[0149] B13、根据B9-B12任一项所述的装置,其中,还包括:

[0150] 建立模块,适于在所述确定模块确定所述待加载应用页面的加载时长之后,获取所述待加载应用页面的标识;在所述待加载应用页面的标识与加载时长之间建立关联关系,并将所述关联关系存储在指定数据库中。

[0151] B14、根据B13所述的装置,其中,还包括:

[0152] 第一计算模块,适于在所述建立模块将所述关联关系存储在指定数据库中之后,在所述关联关系中查找指定应用页面对应的多个加载时长;根据查找到的所述多个加载时长,计算所述指定应用页面的加载性能。

[0153] B15、根据B13所述的装置,其中,还包括:

[0154] 第二计算模块,适于当所述待加载应用页面的标识中包含所述待加载应用页面对应应用的版本标识时,在所述建立模块将所述关联关系存储在指定数据库中之后,在所述关联关系中查找指定应用页面对应的多个加载时长;在所述多个加载时长中分类出不同应用版本下的加载时长;根据分类出的所述不同应用版本下的加载时长,计算所述指定应用页面在不同应用版本下的加载性能。

[0155] B16、根据B13所述的装置,其中,还包括:

[0156] 第三计算模块,适于在所述建立模块将所述关联关系存储在指定数据库中之后,确定指定应用中的多个应用页面;在所述关联关系中查找所述多个应用页面各自对应的加载时长;根据查找到的所述多个应用页面各自对应的加载时长,计算所述指定应用的页面

加载性能。

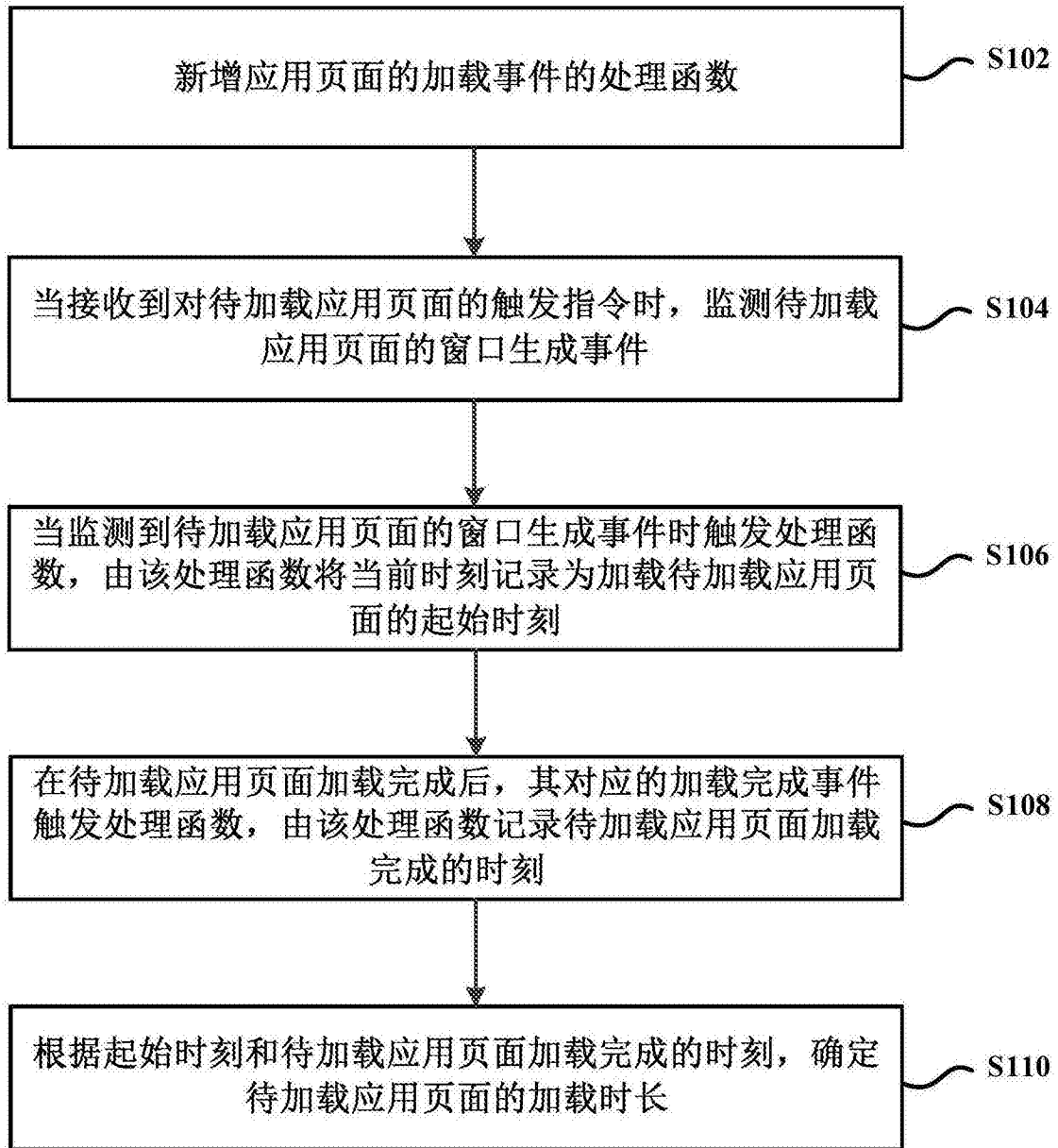


图1

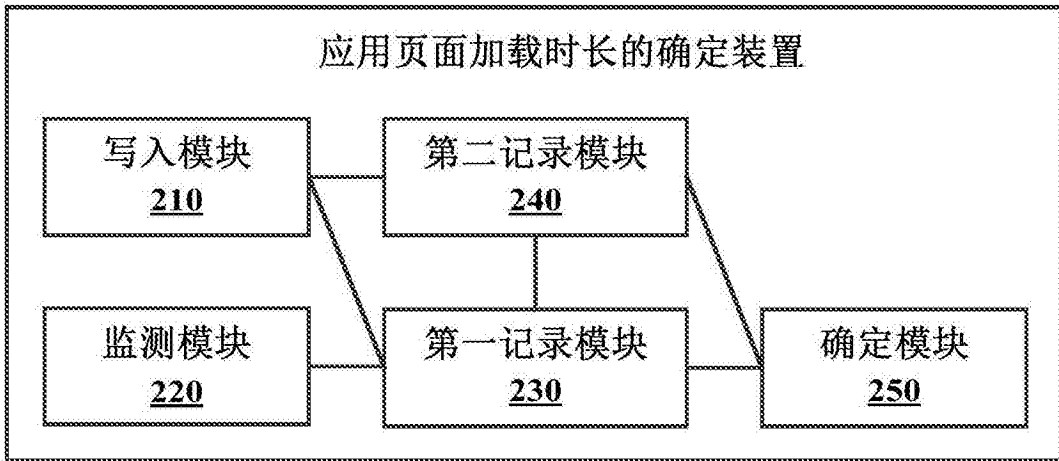


图2

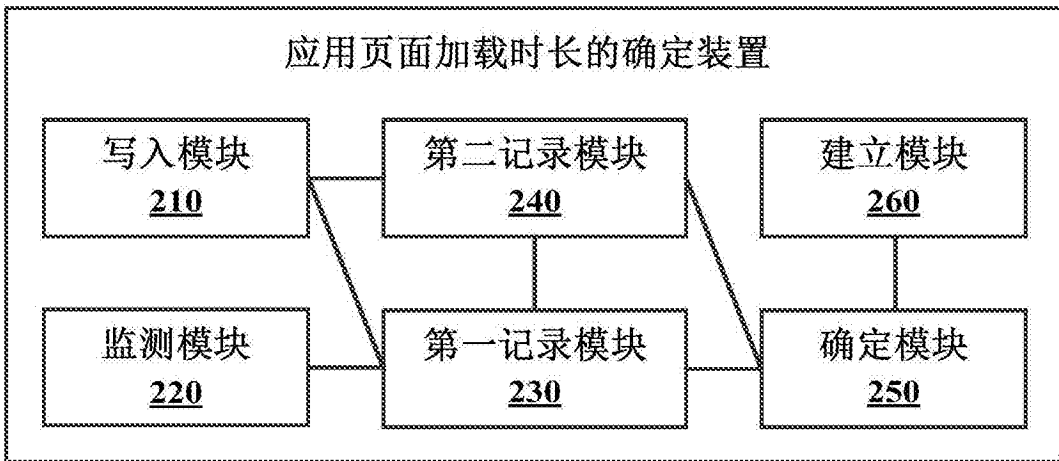


图3