



(12) 发明专利

(10) 授权公告号 CN 109800020 B

(45) 授权公告日 2023. 02. 10

(21) 申请号 201910107623.1

(22) 申请日 2019.02.02

(65) 同一申请的已公布的文献号
申请公布号 CN 109800020 A

(43) 申请公布日 2019.05.24

(73) 专利权人 北京字节跳动网络技术有限公司
地址 100080 北京市石景山区实兴大街30
号院3号楼2层B-0035房间

(72) 发明人 李续铖

(74) 专利代理机构 北京品源专利代理有限公司
11332
专利代理师 孟金喆

(51) Int. Cl.
G06F 8/76 (2018.01)

(56) 对比文件

CN 109191605 A, 2019.01.11

CN 107590385 A, 2018.01.16

审查员 刘艳萍

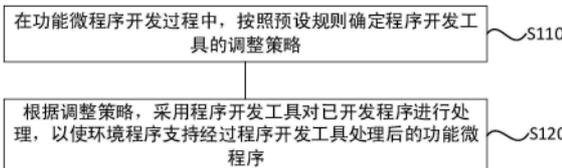
权利要求书2页 说明书8页 附图3页

(54) 发明名称

一种程序开发的代码自动调整方法、装置、设备和介质

(57) 摘要

本公开实施例公开了一种程序开发的代码自动调整方法、装置、终端设备及存储介质。该方法包括在小程序开发过程中,按照预设规则确定程序开发工具的调整策略,根据所述调整策略,采用程序开发工具对已开发程序进行处理,以使环境程序支持经过程序开发工具处理后的小程序。在小程序开发过程中,通过程序开发工具根据调整策略对小程序已开发的程序进行处理,使得环境程序支持经过程序开关工具处理后的小程序的同时,小程序能够使用程序开发工具,降低了小程序开发过程中的复杂度,同时降低了程序开发者的操作难度,提高了小程序的开发效率。



1. 一种程序开发的代码自动调整方法,其特征在于,所述方法包括:

在小程序开发过程中,按照预设规则确定程序开发工具的调整策略,其中,所述预设规则是程序开发工具处理小程序的程序语句时,不同状况对应不同的调整策略;

根据所述调整策略,采用程序开发工具对已开发程序进行处理,以使环境程序支持经过程序开发工具处理后的小程序;

所述根据所述调整策略,采用程序开发工具对已开发程序进行处理包括:

根据所述调整策略,预先采用程序开发工具对已开发程序进行处理;

监听获取程序开发工具处理过程中的变更数据;

按照所述调整策略,调整所述变更数据;

所述根据所述调整策略,采用程序开发工具对已开发程序进行处理还包括:

根据所述调整策略,在已开发程序中确定目标处理数据,并采用所述程序开发工具对所述目标处理数据进行处理;

所述程序开发工具可在APP开发和网页程序开发中使用;

所述根据所述调整策略,在已开发程序中确定目标处理数据,并采用所述程序开发工具对所述目标处理数据进行处理包括:

从小程序的页面文件夹中获取至少一个页面的文件存储路径;

根据所述页面的文件存储路径获取页面对应的至少一个文件,作为目标处理数据;

采用webpack对每个页面对应的各目标处理数据进行打包处理,形成一个打包文件。

2. 根据权利要求1所述的方法,其特征在于,所述程序开发工具为webpack。

3. 根据权利要求2所述的方法,其特征在于,按照所述调整策略,调整所述变更数据包括:

识别所述变更数据中的运行变量,其中,所述运行变量为环境程序运行所述小程序的页面时所需的变量;

将环境程序不支持的运行变量替换为环境程序支持的运行变量。

4. 根据权利要求2所述的方法,其特征在于,监听获取程序开发工具处理过程中的变更数据包括:

接收webpack处理过程中上报的变更数据信息,并根据所述变更数据信息确定变更数据。

5. 根据权利要求3所述的方法,其特征在于,所述不支持的运行变量为网页对象变量,所述支持的运行变量为应用软件作为环境程序时支持的对象变量。

6. 一种程序开发的代码自动调整装置,其特征在于,包括:

调整策略确定模块,用于在小程序开发过程中,按照预设规则确定程序开发工具的调整策略,其中,所述预设规则是程序开发工具处理小程序的程序语句时,不同状况对应不同的调整策略;

处理模块,用于根据所述调整策略,采用程序开发工具对已开发程序进行处理,以使环境程序支持经过程序开发工具处理后的小程序;

所述处理模块用于根据调整策略,预先采用程序开发工具对已开发程序进行处理,在处理过程中,监听获取程序开发工具处理过程中的变更数据,并按照预设规则,调整变更数据;

所述处理模块还用于根据调整策略,在已开发程序中确定目标处理数据,并采用程序开发工具对目标处理数据进行处理;

所述程序开发工具可在APP开发和网页程序开发中使用;

所述处理模块还用于:从小程序的页面文件夹中获取至少一个页面的文件存储路径;

根据所述页面的文件存储路径获取页面对应的至少一个文件,作为目标处理数据;

采用webpack对每个页面对应的各目标处理数据进行打包处理,形成一个打包文件。

7.一种终端设备,其特征在于,包括:

一个或多个处理器;

存储器,用于存储一个或多个程序;

当所述一个或多个程序被所述一个或多个处理器执行,使得所述一个或多个处理器实现如权利要求1-5任一所述的程序开发的代码自动调整方法。

8.一种计算机可读存储介质,其上存储有计算机程序,其特征在于,该程序被处理器执行时实现如权利要求1-5任一所述的程序开发的代码自动调整方法。

一种程序开发的代码自动调整方法、装置、设备和介质

技术领域

[0001] 本公开实施例涉及计算机软件技术,尤其涉及一种程序开发的代码自动调整方法、装置、设备和介质。

背景技术

[0002] 随着智能终端的普及,各种功能的应用软件(APP)逐渐增多,并且应用软件的功能也在持续升级更新中。APP客户端通常需要用户将软件下载到终端,并安装,而后运行。为了适应不同的功能需求,目前还出现了小程序。小程序的特点是,能够实现某种业务功能,但小程序并不需要用户下载独立的软件到本地、安装再使用。小程序通常是以某个APP客户端或终端内的某个应用软件,作为运行平台来实现的,支撑小程序运行的应用软件记为环境程序。例如微信的小程序是以微信客户端作为运行平台来实现的,微信客户端为微信的小程序的环境程序。小程序的开发者,通过开发小程序所需展示的各个页面来完成开发。

[0003] 由于小程序是由环境程序来支撑运行的,所以环境程序也就约束了小程序的开发,特别是一些软件工具无法使用,没法使用成熟开发工具,限制了程序开发者的操作,也降低了程序开发效率。

发明内容

[0004] 本公开实施例提供了一种程序开发的代码自动调整方法、装置、设备和介质,以使小程序能够使用一些程序开发工具,并仍然兼容于环境程序,能够正常运行。

[0005] 第一方面,本公开实施例提供了一种程序开发的代码自动调整方法,所述方法包括:

[0006] 在小程序开发过程中,按照预设规则确定程序开发工具的调整策略;

[0007] 根据所述调整策略,采用程序开发工具对已开发程序进行处理,以使环境程序支持经过程序开发工具处理后的小程序。

[0008] 第二方面,本公开实施例还提供了一种程序开发的代码自动调整装置,包括:

[0009] 调整策略确定模块,用于在小程序开发过程中,按照预设规则确定程序开发工具的调整策略;

[0010] 处理模块,用于根据所述调整策略,采用程序开发工具对已开发程序进行处理,以使环境程序支持经过程序开发工具处理后的小程序。

[0011] 第三方面,本公开实施例还提供了一种终端设备,包括:

[0012] 一个或多个处理器;

[0013] 存储器,用于存储一个或多个程序;

[0014] 当所述一个或多个程序被所述一个或多个处理器执行,使得所述一个或多个处理器实现如本公开实施例第一方面任意的程序开发的代码自动调整方法。

[0015] 第四方面,本公开实施例还提供了一种计算机可读存储介质,其上存储有计算机程序,该程序被处理器执行时实现如本公开实施例第一方面任意的程序开发的代码自动调

整方法。

[0016] 本公开实施例所提供的技术方案,在小程序开发过程中,通过程序开发工具根据调整策略对小程序已开发的程序进行处理,使得环境程序支持经过程序开关工具处理后的小程序的同时,小程序能够使用程序开发工具,降低了小程序开发过程中的复杂度,同时降低了程序开发者的操作难度,提高了小程序的开发效率。

附图说明

[0017] 为了更清楚地说明本公开实施例或现有技术中的技术方案,下面将对实施例或现有技术描述中所需要使用的附图做一简单地介绍,显而易见地,下面描述中的附图是本公开的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其他的附图。

[0018] 图1为本公开实施例一提供的一种程序开发的代码自动调整方法的流程示意图;

[0019] 图2为本公开实施例二提供的一种程序开发的代码自动调整方法的流程示意图;

[0020] 图3为本公开实施例二提供的另一种程序开发的代码自动调整方法的流程示意图;

[0021] 图4为本公开实施例四提供的一种程序开发的代码自动调整装置的结构示意图;

[0022] 图5为本公开实施例五提供的一种终端设备的结构示意图。

具体实施方式

[0023] 为使本公开的目的、技术方案和优点更加清楚,以下将参照本公开实施例中的附图,通过实施方式清楚、完整地描述本公开的技术方案,显然,所描述的实施例是本公开一部分实施例,而不是全部的实施例。基于本公开中的实施例,本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例,都属于本公开保护的范围。

[0024] 下述各实施例中,每个实施例中同时提供了可选特征和示例,实施例中记载的各个特征可进行组合,形成多个可选方案,不应将每个编号的实施例仅视为一个技术方案。

[0025] 本公开实施例提供一种程序开发的代码自动调整方法、装置、终端设备及存储介质,可以实现在小程序开发过程中,通过程序开发工具根据调整策略对小程序进行调整,使得环境程序支持经过程序开关工具处理后的小程序的同时,小程序能够使用程序开发工具,降低了小程序开发过程中的复杂度,同时降低了程序开发者的操作难度,提高了小程序的开发效率。

[0026] 实施例一

[0027] 图1为本公开实施例一提供的一种程序开发的代码自动调整方法的流程示意图。本公开实施例公开的方法主要适用于小程序开发过程中程序开发工具不适用的情况。具体的,可以通过程序开发的代码自动调整装置实现。如图1所示,该方法具体包括如下步骤:

[0028] S110、在小程序开发过程中,按照预设规则确定程序开发工具的调整策略。

[0029] 具体地,小程序是一种能够实现某种业务功能,而且不需要用户下载独立的软件到本地、安装再使用的一种程序。一般情况下,小程序可以通过某个APP作为运行平台,例如小程序可以是一个小程序,运行平台可以是微信APP,小程序通过以微信客户端作为运行平台实现小程序的运行。因此,小程序的运行平台与APP和网页的运行平台不同。APP的运行平

台可以为终端设备的操作系统,网页的运行平台可以为浏览器,小程序的运行平台可以为某个APP。因运行平台不同,运行程序与其对应的运行平台之间的交互也不同。例如,APP与操作系统通过预先配置的一种协议实现交互,当APP运行时,操作系统根据协议解析APP的请求,为APP的运行配置资源,从而实现APP的运行。而小程序与其运行平台APP通过另一种预先配置的协议实现交互,在小程序运行过程中,APP根据协议解析小程序的请求,为小程序配置自己能够支配的资源,从而实现小程序的运行。因此,APP与操作系统之间的交互和APP与小程序之间的交互不同,对应程序语句等也不同。

[0030] 在开发APP和网页程序的过程中,可以使用一些成熟的程序开发工具,以降低开发过程中开发人员的难度,提高了开发效率。小程序一般通过开发小程序所需展示的各个页面来完成开发。因APP的程序语句、网页的程序语句与小程序的程序语句各不相同,一些在开发APP和网页程序过程中使用的程序开发工具无法兼容小程序的程序语句,因此可以按照一定的预设规则确定对程序开发工具的调整策略,使其在小程序开发过程中调整小程序的程序语句,从而程序开发工具能够兼容小程序。即功能开发工具可以应用到小程序的开发中,通过程序开发工具处理的小程序的程序语句可以在环境程序中运行,环境程序可以支持采用了程序开发工具的小程序。

[0031] 预设规则是一种预先设定的规则,用于设定小程序开发的程序语句在自动调整时的情况与调整策略的对应关系。示例性地,预设规则可以是程序开发工具处理小程序的程序语句时不同状况对应不同的调整策略。调整策略可以为程序开发工具调整小程序的程序语句的具体调整规则,以使环境程序支持经过程序开发工具处理后的小程序。例如,当小程序的程序语句中包括不能被环境程序兼容的代码时,对应的调整策略可以为替换策略,使小程序的程序语句中不能被环境程序兼容的代码在不影响程序语句的功能的前提下,被替换为环境程序能够兼容的代码。

[0032] 一般情况下,一个功能可以包括网页版、APP版和小程序版。一个功能的不同版本在实现的功能相同时,对应的程序语句不同。示例性地,程序语句中的语法和定时规则可能不同。例如根据语法的不同,不同版本的同一程序的程序语句中代表同一变量的后缀名和变量名的代码可能不同。调整策略可以为将程序语句中一种协议识别的字符填充为另一种协议识别的字符。例如可以是将程序语句的后缀名和变量名等的代码替换为环境程序可以兼容的后缀名和变量名的代码。程序开发工具通过调整策略对小程序的程序语句的调整,使得程序开发工具处理过的已开发程序可以得到环境程序的支持。

[0033] 需要说明的是,程序开发工具的调整策略由预设规则确定。当预设规则发生变化时,调整策略也相应的变化。

[0034] S120、根据调整策略,采用程序开发工具对已开发程序进行处理,以使环境程序支持经过程序开发工具处理后的小程序。

[0035] 具体地,环境程序可以是一个APP。在小程序运行过程中,环境程序根据协议解析小程序的请求,为小程序配置自己能够支配的资源,从而实现小程序的运行。程序开发工具是一种用于程序开发过程中的工具,可以对开发的程序进行处理,提高程序开发的效率。例如,webpack是一种现代JavaScript应用程序静态模块打包器的程序开发工具,可以用于对现代JavaScript应用程序进行打包压缩,提高程序加载速度,节省流量。当webpack直接应用在程序开发中,可以减少程序开发时webpack功能的开发,进而提高了程序开发的效率。

[0036] 当根据预设规则确定调整策略后,程序开发工具根据调整策略对小程序已开发程序的程序语句进行处理,使得环境程序支持经过程序开发工具处理后的小程序,因此程序开发工具可以应用到小程序中,以降低开发过程中开发人员的难度,提高小程序的开发效率。

[0037] 本公开实施例的技术方案,在小程序开发过程中,通过程序开发工具根据调整策略对小程序已开发的程序进行处理,使得环境程序支持经过程序开发工具处理后的小程序的同时,小程序能够使用程序开发工具,降低了小程序开发过程中的复杂度,同时降低了程序开发者的操作难度,提高了小程序的开发效率。

[0038] 实施例二

[0039] 在实施例一的基础上,实施例二示例性地示出了根据预设规则确定一种调整策略的情况。调整策略为程序开发工具对已开发程序进行预处理后再调整。图2为本公开实施例二提供的一种程序开发的代码自动调整方法的流程示意图。如图2所示,具体地,图1中的步骤S120可以包括:

[0040] S121、根据调整策略,预先采用程序开发工具对已开发程序进行处理。

[0041] 具体地,程序开发工具对小程序中已开发的程序进行预处理,在预处理过程中,程序开发工具对小程序已开发的程序进行处理,提高程序开发的效率。例如,程序开发工具为webpack,webpack对小程序已开发的程序进行打包压缩,提高小程序的加载速度,同时节省流量。

[0042] S122、监听获取程序开发工具处理过程中的变更数据。

[0043] 具体地,在确定调整策略后,程序开发工具可以根据调整策略监听小程序已开发的程序。即程序开发工具根据调整策略对小程序已开发的程序进行响应,以确定程序开发工具需要变更的数据。例如,调整策略为替换策略,在程序开发工具对小程序已开发的程序具体进行处理的过程中,需要根据替换策略对小程序的已开发的程序进行监听,以获得需要变更的数据。

[0044] 一般情况下,需要变更的数据可以包括多种。示例性地,变更数据可以为运行变量。例如,代表同一语义的运行变量height,在网页版的程序中即为Windows height,在小程序中即为global height。两者的运行变量的语义相同,在不同版本的程序语句中运行变量不同。为了将网页版的运行变量替换为小程序能够识别兼容的运行变量,需要将Windows替换为global,因此运行变量height即为变更数据。

[0045] 示例性地,当程序开发工具为webpack时,监听获取程序开发工具处理过程中的变更数据可以包括:

[0046] 接收webpack处理过程中上报的变更数据信息,并根据变更数据信息确定变更数据。

[0047] 具体地,在webpack处理过程中,webpack对程序进行监听,并对变更数据信息进行上报。webpack根据webpack处理过程中上报的变更数据信息可以确定变更数据。例如,webpack处理过程中上报的变更数据信息可以为小程序已开发程序中无法被环境程序支持的数据信息,变更数据信息所在的数据即为变更数据。或者,webpack处理过程中上报的变更数据信息也可以为调整策略中的变更数据的信息,根据调整策略中变更数据的信息确定变更数据。

[0048] S123、按照调整策略,调整变更数据。

[0049] 当确定变更数据后,按照调整策略调整变更数据。例如,调整策略为替换策略,当确定需要替换的变更数据后,对变更数据进行替换,使得替换后的变更数据使得程序开发工具处理过的已开发程序可以得到环境程序的支持。

[0050] 示例性地,图3为本公开实施例二提供的另一种程序开发的代码自动调整方法的流程示意图。如图3所示,当程序开发工具为webpack时,图2中的S123步骤可以包括以下步骤:

[0051] S1231、识别变更数据中的运行变量,其中,运行变量为环境程序运行小程序的页面时所需的变量。

[0052] 具体地,环境程序在运行小程序的页面时,需要对小程序进行解析识别,使得小程序能够正常运行。在此过程中,环境程序需要识别的变量即为运行变量。

[0053] S1232、将环境程序不支持的运行变量替换为环境程序支持的运行变量。

[0054] 具体地,不支持的运行变量为网页对象变量,支持的运行变量为应用软件作为环境程序时支持的对象变量。所谓对象,是计算设备运行时分配资源的控制句柄,也可以是指针。该对象名称,用于统一管理计算设备分配给该小程序的资源,例如缓存空间、进程等。

[0055] 本公开实施例的技术方案,在小程序开发过程中,通过程序开发工具对已开发程序进行预处理,并监听获取预处理过程中的变更数据,根据调整策略调整变更数据,使得环境程序支持根据调整策略调整后的小程序的同时,小程序能够使用程序开发工具,降低了小程序开发过程中的复杂度,同时降低了程序开发者的操作难度,提高了小程序的开发效率。

[0056] 实施例三

[0057] 在实施例一的基础上,实施例三示例性地示出了根据预设规则确定另一种调整策略的情况。调整策略为程序开发工具先确定目标处理数据再进行后处理。具体地,图1中的步骤S120可以包括:

[0058] 根据调整策略,在已开发程序中确定目标处理数据,并采用程序开发工具对目标处理数据进行处理。

[0059] 具体地,小程序一般包括至少一个页面,每个页面可以设置为一个文件夹,每个页面文件夹对应的页面文件的存储路径不同。在小程序运行的过程中,需要对每个页面文件夹中的页面文件进行处理。一个页面文件夹中的至少一个文件的存储路径相同,因此一个目标处理数据可以为一个页面对应的至少一个文件。在已开发程序中确定目标处理数据后,程序开发工具对目标处理数据根据调整策略进行处理。

[0060] 示例性地,当程序开发工具为webpack时,webpack从小程序的页面文件夹中获取至少一个页面的文件存储路径,根据页面的文件存储路径获取页面对应的至少一个文件,作为目标处理数据。其中,一个页面文件夹中对应的文件作为一个目标处理数据。采用webpack对每个页面对应的各目标处理数据进行打包处理,形成一个打包文件。在确定打包文件后,webpack根据调整策略对每个打包文件中的数据进行后处理,以使环境程序支持经过程序开发工具处理后的小程序。当webpack对目标处理数据处理后,webpack还需对小程序进行打包压缩等过程,提高程序加载速度,节省流量。

[0061] 需要说明的是,一个页面文件夹可以对应多个文件,此处不做限定。

[0062] 实施例四

[0063] 本公开实施例还提供一种程序开发的代码自动调整装置。图4为本公开实施例四提供的一种程序开发的代码自动调整装置的结构示意图。如图4所示,该装置包括:

[0064] 调整策略确定模块11,用于在小程序开发过程中,按照预设规则确定程序开发工具的调整策略。

[0065] 处理模块12,用于根据所述调整策略,采用程序开发工具对已开发程序进行处理,以使环境程序支持经过程序开发工具处理后的小程序。

[0066] 可选地,处理模块12可以根据调整策略,预先采用程序开发工具对已开发程序进行处理,在处理过程中,监听获取程序开发工具处理过程中的变更数据,并按照预设规则,调整变更数据。当程序开发工具为webpack时,在按照预设规则,调整变更数据的过程中,需要识别变更数据中的运行变量,其中,运行变量为环境程序运行小程序的页面时所需的变量,将环境程序不支持的运行变量替换为环境程序支持的运行变量。其中,不支持的运行变量为网页对象变量,支持的运行变量为应用软件作为环境程序时支持的对象变量。

[0067] 可选地,处理模块12还可以根据调整策略,在已开发程序中确定目标处理数据,并采用程序开发工具对目标处理数据进行处理。在确定目标处理数据的过程中,从小程序的页面文件夹中获取至少一个页面的文件存储路径,根据页面的文件存储路径获取页面对应的至少一个文件,作为目标处理数据。程序开发工具对目标处理数据进行打包处理,形成打包文件。在确定打包文件后,程序开发工具根据调整策略对每个打包文件中的数据进行处理,以使环境程序支持经过程序开发工具处理后的小程序。

[0068] 本公开实施例提供的程序开发的代码自动调整装置,可执行本公开方法实施例所提供的程序开发的代码自动调整方法中程序开发工具所执行的步骤,具备执行方法相应的功能模块和有益效果。

[0069] 实施例五

[0070] 图5为本公开实施例五提供的一种终端设备的结构示意图。其示出了适于用来实现本公开实施例的终端设备500的结构示意图。本公开实施例中的终端设备可以包括但不限于诸如移动电话、智能电话、笔记本电脑、数字广播接收器、PDA(个人数字助理)、PAD(平板电脑)、PMP(便携式多媒体播放器)、车载终端(例如车载导航终端)等等的移动终端以及诸如数字TV、台式计算机等等的固定终端。图5示出的终端设备仅仅是一个示例,不应对本公开实施例的功能和使用范围带来任何限制。

[0071] 如图5所示,终端设备500可以包括处理装置(例如中央处理器、图形处理器等)501,其可以根据存储在只读存储器(ROM)502中的程序或者从存储装置508加载到随机访问存储器(RAM)503中的程序而执行各种适当的动作和处理。在RAM 503中,还存储有电子设备500操作所需的各种程序和数据。处理装置501、ROM 502以及RAM 503通过总线504彼此相连。输入/输出(I/O)接口505也连接至总线504。

[0072] 通常,以下装置可以连接至I/O接口505:包括例如触摸屏、触摸板、键盘、鼠标、摄像头、麦克风、加速度计、陀螺仪等的输入装置506;包括例如液晶显示器(LCD)、扬声器、振动器等的输出装置507;包括例如磁带、硬盘等的存储装置508;以及通信装置509。通信装置509可以允许终端设备500与其他设备进行无线或有线通信以交换数据。虽然图5示出了具有各种装置的终端设备500,但是应理解的是,并不要求实施或具备所有示出的装置。可以

替代地实施或具备更多或更少的装置。

[0073] 特别地,根据本公开的实施例,上文参考流程图描述的过程可以被实现为计算机软件程序。例如,本公开的实施例包括一种计算机程序产品,其包括承载在计算机可读介质上的计算机程序,该计算机程序包含用于执行流程图所示的方法的程序代码。在这样的实施例中,该计算机程序可以通过通信装置509从网络上被下载和安装,或者从存储装置508被安装,或者从ROM 502被安装。在该计算机程序被处理装置501执行时,执行本公开实施例的方法中限定的上述功能。

[0074] 需要说明的是,本公开上述的计算机可读介质可以是计算机可读信号介质或者计算机可读存储介质或者是上述两者的任意组合。计算机可读存储介质例如可以是——但不限于——电、磁、光、电磁、红外线、或半导体的系统、装置或器件,或者任意以上的组合。计算机可读存储介质的更具体的例子可以包括但不限于:具有一个或多个导线的电连接、便携式计算机磁盘、硬盘、随机访问存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPROM或闪存)、光纤、便携式紧凑磁盘只读存储器(CD-ROM)、光存储器件、磁存储器件、或者上述的任意合适的组合。在本公开中,计算机可读存储介质可以是任何包含或存储程序的有形介质,该程序可以被指令执行系统、装置或者器件使用或者与其结合使用。而在本公开中,计算机可读信号介质可以包括在基带中或者作为载波一部分传播的数据信号,其中承载了计算机可读的程序代码。这种传播的数据信号可以采用多种形式,包括但不限于电磁信号、光信号或上述的任意合适的组合。计算机可读信号介质还可以是计算机可读存储介质以外的任何计算机可读介质,该计算机可读信号介质可以发送、传播或者传输用于由指令执行系统、装置或者器件使用或者与其结合使用的程序。计算机可读介质上包含的程序代码可以用任何适当的介质传输,包括但不限于:电线、光缆、RF(射频)等等,或者上述的任意合适的组合。

[0075] 上述计算机可读介质可以是上述电子设备中所包含的;也可以是单独存在,而未装配入该电子设备中。

[0076] 上述计算机可读介质承载有一个或者多个程序,当上述一个或者多个程序被该电子设备执行时,使得该电子设备:在小程序开发过程中,按照预设规则确定程序开发工具的调整策略;根据调整策略,采用程序开发工具对已开发程序进行处理,以使环境程序支持经过程序开发工具处理后的小程序。

[0077] 可以以一种或多种程序设计语言或其组合来编写用于执行本公开的操作的计算机程序代码,上述程序设计语言包括面向对象的程序设计语言——诸如Java、Smalltalk、C++,还包括常规的过程式程序设计语言——诸如“C”语言或类似的程序设计语言。程序代码可以完全地在用户计算机上执行、部分地在用户计算机上执行、作为一个独立的软件包执行、部分在用户计算机上部分在远程计算机上执行、或者完全在远程计算机或服务器上执行。在涉及远程计算机的情形中,远程计算机可以通过任意种类的网络——包括局域网(LAN)或广域网(WAN)——连接到用户计算机,或者,可以连接到外部计算机(例如利用因特网服务提供商来通过因特网连接)。

[0078] 附图中的流程图和框图,图示了按照本公开各种实施例的系统、方法和计算机程序产品的可能实现的体系架构、功能和操作。在这点上,流程图或框图中的每个方框可以代表一个模块、程序段、或代码的一部分,该模块、程序段、或代码的一部分包含一个或多个用

于实现规定的逻辑功能的可执行指令。也应当注意,在有些作为替换的实现中,方框中所标注的功能也可以以不同于附图中所标注的顺序发生。例如,两个接连地表示的方框实际上可以基本并行地执行,它们有时也可以按相反的顺序执行,这依所涉及的功能而定。也要注意,框图和/或流程图中的每个方框、以及框图和/或流程图中的方框的组合,可以用执行规定的功能或操作的专用的基于硬件的系统来实现,或者可以用专用硬件与计算机指令的组合来实现。

[0079] 描述于本公开实施例中涉及到的模块可以通过软件的方式实现,也可以通过硬件的方式来实现。其中,模块的名称在某种情况下并不构成对该模块本身的限定,例如,调整策略确定模块还可以被描述为“在小程序开发过程中,按照预设规则确定程序开发工具的调整策略”。

[0080] 以上描述仅为本公开的较佳实施例以及对所运用技术原理的说明。本领域技术人员应当理解,本公开中所涉及的公开范围,并不限于上述技术特征的特定组合而成的技术方案,同时也应涵盖在不脱离上述公开构思的情况下,由上述技术特征或其等同特征进行任意组合而形成的其它技术方案。例如上述特征与本公开中公开的(但不限于)具有类似功能的技术特征进行互相替换而形成的技术方案。

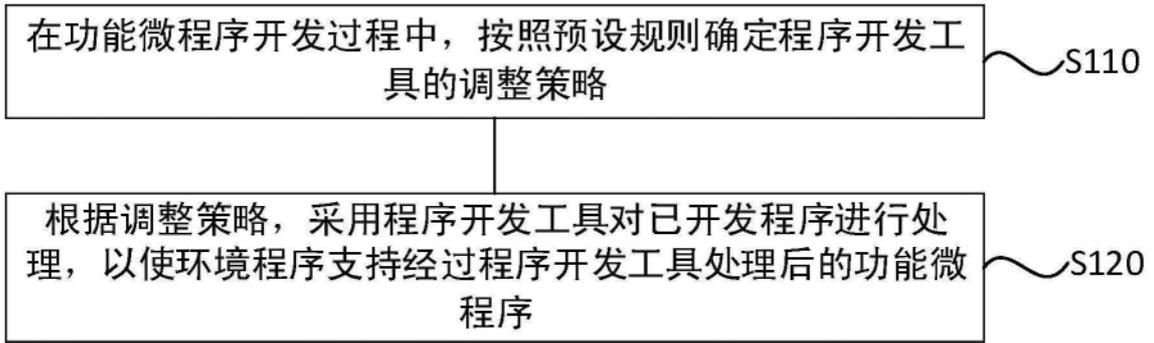


图1

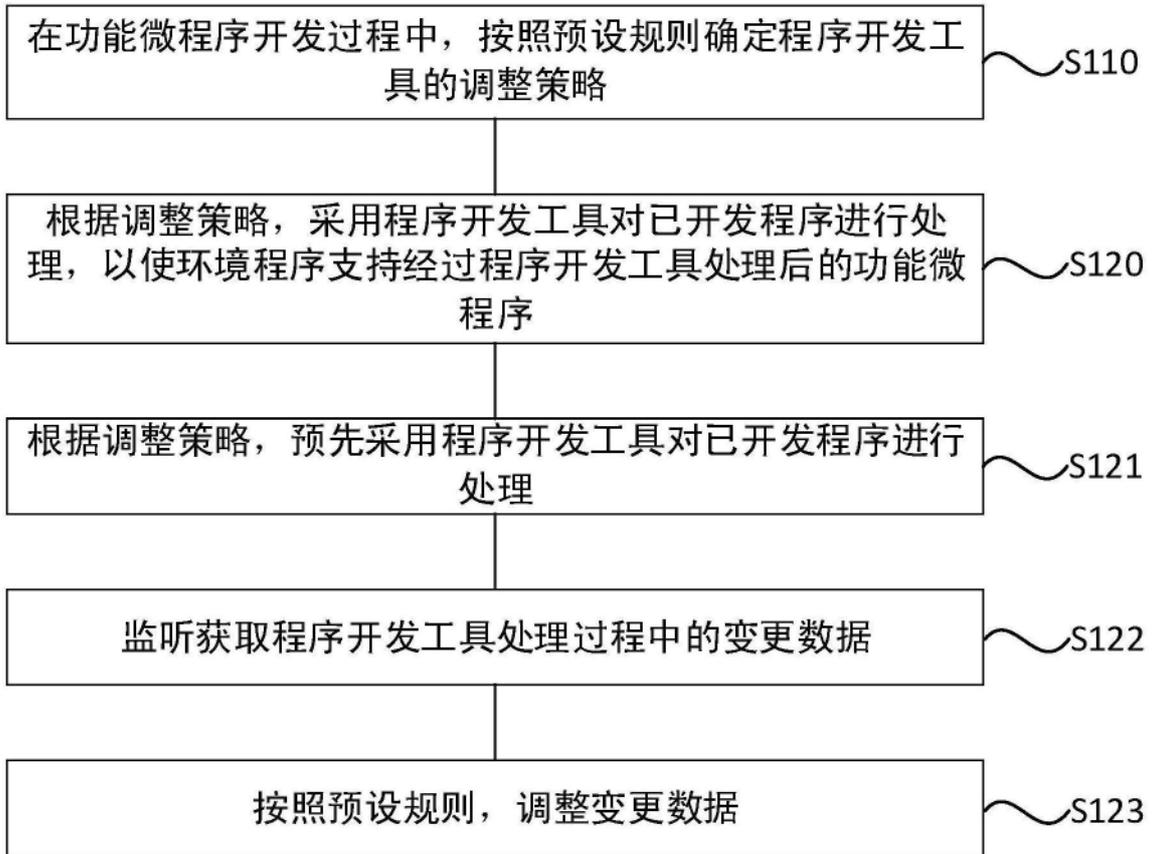


图2

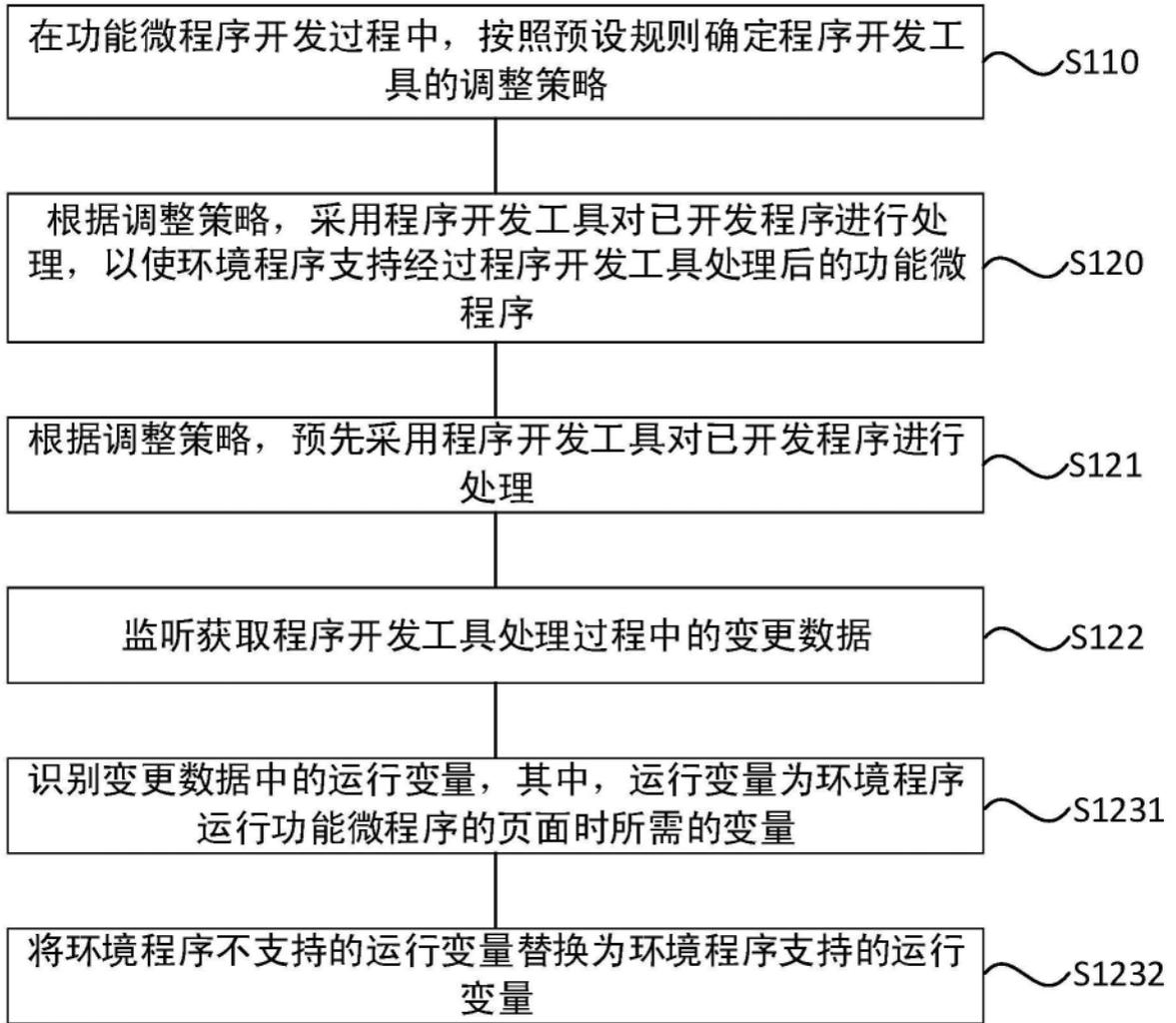


图3

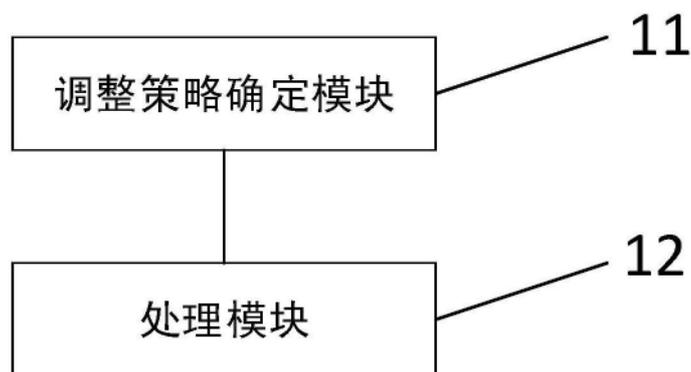


图4

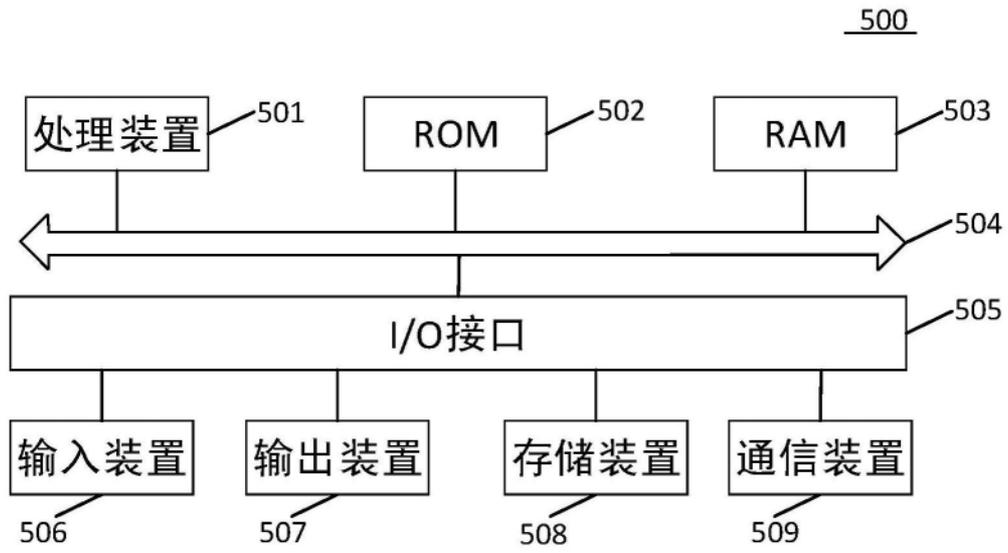


图5