



(12) 发明专利申请

(10) 申请公布号 CN 112965719 A

(43) 申请公布日 2021.06.15

(21) 申请号 202110310967.X

(22) 申请日 2020.03.06

(62) 分案原申请数据
202010153657.7 2020.03.06

(71) 申请人 北京五八信息技术有限公司
地址 100083 北京市海淀区学清路甲18号
中关村东升科技园学院园三层301室

(72) 发明人 李中秋

(74) 专利代理机构 北京润泽恒知识产权代理有
限公司 11319
代理人 莎日娜

(51) Int.Cl.
G06F 8/40 (2018.01)

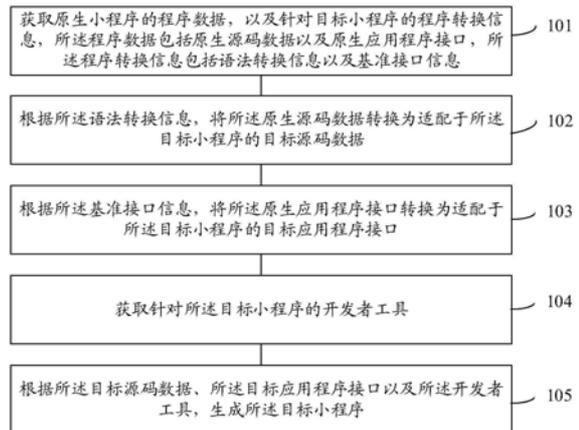
权利要求书2页 说明书11页 附图5页

(54) 发明名称

一种小程序的生成方法和装置

(57) 摘要

本发明实施例提供了一种小程序的生成方法及装置,通过获取原生小程序的程序数据,以及针对目标小程序的程序转换信息,然后根据程序转换信息对原生小程序的原生源码数据以及原生应用程序接口进行转换,获得适配于目标小程序的目标源码数据以及目标应用程序接口,接着集成开发者工具,并输出目标小程序的实例,从而通过原生小程序进行源码数据的语法结构转换,对源码数据无侵入,且不需要额外引入框架与引入框架带来的维护风险,并且对应用程序接口进行适配,抹平了小程序之间的差异,保证了小程序转换之后的稳定性。



1. 一种小程序的生成方法,其特征在于,包括:

获取原生小程序的程序数据,以及针对目标小程序的程序转换信息,所述程序数据包括原生成源数据以及原生应用程序接口,所述程序转换信息包括语法转换信息以及基准接口信息;

根据所述语法转换信息,将所述原生成源数据转换为适配于所述目标小程序的目标源数据;

获取所述原生小程序的运行框架,所述运行框架包括所述原生应用程序接口的原生参数信息;

根据所述基准接口信息,将所述原生参数信息转换为适配于所述目标小程序的目标参数信息;

采用所述目标参数信息更新所述运行框架中的原生参数信息,获得所述目标小程序的目标应用程序接口;

根据所述目标源数据与所述目标应用程序接口,生成所述目标小程序。

2. 根据权利要求1所述的方法,其特征在于,所述根据所述语法转换信息,将所述原生成源数据转换为适配于所述目标小程序的目标源数据,包括:

遍历所述原生小程序的源码目录,获取所述原生成源数据的原生程序文件;

根据所述语法转换信息,对所述原生程序文件进行语法结构转换,生成适配于所述目标小程序的目标程序文件,所述目标程序文件包括目标配置文件、目标结构文件、目标样式表文件以及目标逻辑文件;

采用所述目标配置文件、目标结构文件、目标样式表文件以及目标逻辑文件,生成适配于所述目标小程序的目标源数据。

3. 根据权利要求1或2所述的方法,其特征在于,所述根据所述目标源数据与所述目标应用程序接口,生成所述目标小程序,包括:

获取针对所述目标小程序的开发者工具;

采用所述目标源数据、所述目标应用程序接口以及所述开发者工具,生成所述目标小程序。

4. 根据权利要求1所述的方法,其特征在于,所述基准接口信息包括参数映射关系与结果映射关系,所述原生参数信息包括原生参数名以及原生返回值,所述根据所述基准接口信息,将所述原生参数信息转换为适配于所述目标小程序的目标参数信息,包括:

采用所述参数映射关系,将所述原生参数名转换为目标参数名;

采用所述结果映射关系,将所述原生返回值转换为目标返回值;

所述采用所述目标参数信息更新所述运行框架中的原生参数信息,获得所述目标小程序的目标应用程序接口,包括:

采用所述目标参数名更新所述运行框架中的原生参数名,以及采用所述目标返回值更新所述运行框架中的原生返回值;

确定一针对所述目标参数名与所述目标返回值的平台标识,生成与所述平台标识匹配的目标应用程序接口。

5. 根据权利要求4所述的方法,其特征在于,所述基准接口信息通过如下方式生成:

获取基准小程序的基准应用程序接口的基准参数信息,所述基准参数信息包括基准参

数名以及基准返回值；

获取所述目标小程序的目标应用程序接口的目标参数信息，所述目标参数信息包括所述目标参数名以及所述目标返回值；

当所述目标参数名与所述基准参数名不同时，则建立所述目标参数名与所述基准参数名之间的参数映射关系；

当所述目标返回值与所述基准返回值不同时，则建立所述目标返回值与所述基准返回值之间的结果映射关系。

6. 根据权利要求5所述的方法，其特征在于，还包括：

当所述目标返回值与所述基准返回值不同时，则将所述原生返回值封装为与所述基准返回值对应的返回值。

7. 根据权利要求3所述的方法，其特征在于，所述开发者工具包括命令行展示工具、日志工具、语法检测工具中的至少一种，所述方法还包括：

通过所述命令行展示工具展示所述目标小程序的目标源码数据；

通过所述语法检测工具对所述目标源码数据进行语法结构检测，生成检测结果；

通过所述日志工具，输出与所述检测结果对应的日志。

8. 一种小程序的生成装置，其特征在于，包括：

数据获取模块，用于获取原生小程序的程序数据，以及针对目标小程序的程序转换信息，所述程序数据包括原生源码数据以及原生应用程序接口，所述程序转换信息包括语法转换信息以及基准接口信息；

语法转换模块，用于根据所述语法转换信息，将所述原生源码数据转换为适配于所述目标小程序的目标源码数据；

运行框架获取子模块，用于获取所述原生小程序的运行框架，所述运行框架包括所述原生应用程序接口的原生参数信息；

参数信息转换子模块，用于根据所述基准接口信息，将所述原生参数信息转换为适配于所述目标小程序的目标参数信息；

应用程序接口获得子模块，用于采用所述目标参数信息更新所述运行框架中的原生参数信息，获得所述目标小程序的目标应用程序接口；

小程序生成模块，用于根据所述目标源码数据与所述目标应用程序接口，生成所述目标小程序。

9. 一种电子设备，其特征在于，包括：

一个或多个处理器；和

其上存储有指令的一个或多个机器可读介质，当由所述一个或多个处理器执行时，使得所述电子设备执行如权利要求1-7所述的一个或多个的方法。

10. 一个或多个机器可读介质，其上存储有指令，当由一个或多个处理器执行时，使得所述处理器执行如权利要求1-7所述的一个或多个的方法。

一种小程序的生成方法和装置

[0001] 本发明专利申请是申请日为2020年3月6日、申请号为202010153657.7、名称为“一种小程序的生成方法和装置”的中国发明专利申请的分案申请。

技术领域

[0002] 本发明涉及数据处理技术领域,特别是涉及一种小程序的生成方法和一种小程序的生成装置。

背景技术

[0003] 随着信息技术的发展,越来越多的企业相应推出小程序,小程序是一种无需下载安装,即可使用的手机应用。只需要扫描二维码,或搜一搜,就能立即使用。其具有无需安装、无需卸载、用完即走,不占手机内存的特点。

[0004] 不同的企业遵循相应的开发规范,使用相应的语言进行小程序的开发,则容易开发出多种不同平台的小程序。为了实现小程序之间的相互转换,开发人员可以通过antmove(蚂蚁搬家)或多端同一开发框架taro进行小程序的生成。然而,通过antmove进行小程序转换仅能够支持小程序转换,而无法支持小程序中插件的迁移;通过taro进行小程序转换则需要小程序中引入框架,容易导致更新维护比原生小程序更加滞后,并且框架无法支持原生小程序能够支持的功能,影响了小程序的使用。

发明内容

[0005] 鉴于上述问题,提出了本发明实施例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一种小程序的生成方法和相应的一种小程序的生成装置。

[0006] 为了解决上述问题,本发明实施例公开了一种小程序的生成方法,包括:

[0007] 获取原生小程序的程序数据,以及针对目标小程序的程序转换信息,所述程序数据包括原生源码数据以及原生应用程序接口,所述程序转换信息包括语法转换信息以及基准接口信息;

[0008] 根据所述语法转换信息,将所述原生源码数据转换为适配于所述目标小程序的目标源码数据;

[0009] 根据所述基准接口信息,将所述原生应用程序接口转换为适配于所述目标小程序的目标应用程序接口;

[0010] 获取针对所述目标小程序的开发者工具;

[0011] 根据所述目标源码数据、所述目标应用程序接口以及所述开发者工具,生成所述目标小程序。

[0012] 可选地,所述根据所述语法转换信息,将所述原生源码数据转换为适配于所述目标小程序的目标源码数据,包括:

[0013] 遍历所述原生小程序的源码目录,获取所述原生源码数据的原生程序文件;

[0014] 根据所述语法转换信息,对所述原生程序文件进行语法结构转换,生成适配于所

述目标小程序的目标程序文件,所述目标程序文件包括目标配置文件、目标结构文件、目标样式表文件以及目标逻辑文件;

[0015] 采用所述目标配置文件、目标结构文件、目标样式表文件以及目标逻辑文件,生成适配于所述目标小程序的目标源码数据。

[0016] 可选地,所述根据所述基准接口信息,将所述原生应用程序接口转换为适配于所述目标小程序的目标应用程序接口,包括:

[0017] 获取所述原生小程序的运行框架,所述运行框架包括所述原生应用程序接口的原生参数信息;

[0018] 根据所述基准接口信息,将所述原生参数信息转换为适配于所述目标小程序的目标参数信息;

[0019] 采用所述目标参数信息更新所述运行框架中的原生参数信息,获得所述目标小程序的目标应用程序接口。

[0020] 可选地,所述基准接口信息包括参数映射关系与结果映射关系,所述原生参数信息包括原生参数名以及原生返回值,所述根据所述基准接口信息,将所述原生参数信息转换为适配于所述目标小程序的目标参数信息,包括:

[0021] 采用所述参数映射关系,将所述原生参数名转换为目标参数名;

[0022] 采用所述结果映射关系,将所述原生返回值转换为目标返回值;

[0023] 所述采用所述目标参数信息更新所述运行框架中的原生参数信息,获得所述目标小程序的目标应用程序接口,包括:

[0024] 采用所述目标参数名更新所述运行框架中的原生参数名,以及采用所述目标返回值更新所述运行框架中的原生返回值;

[0025] 确定一针对所述目标参数名与所述目标返回值的平台标识,生成与所述平台标识匹配的目标应用程序接口。

[0026] 可选地,所述基准接口信息通过如下方式生成:

[0027] 获取基准小程序的基准应用程序接口的基准参数信息,所述基准参数信息包括基准参数名以及基准返回值;

[0028] 获取所述目标小程序的目标应用程序接口的目标参数信息,所述目标参数信息包括所述目标参数名以及所述目标返回值;

[0029] 当所述目标参数名与所述基准参数名不同时,则建立所述目标参数名与所述基准参数名之间的参数映射关系;

[0030] 当所述目标返回值与所述基准返回值不同时,则建立所述目标返回值与所述基准返回值之间的结果映射关系。

[0031] 可选地,还包括:

[0032] 当所述目标返回值与所述基准返回值不同时,则将所述原生返回值封装为与所述基准返回值对应的返回值。

[0033] 可选地,所述开发者工具包括命令行展示工具、日志工具、语法检测工具中的至少一种,所述方法还包括:

[0034] 通过所述命令行展示工具展示所述目标小程序的目标源码数据;

[0035] 通过所述语法检测工具对所述目标源码数据进行语法结构检测,生成检测结果;

- [0036] 通过所述日志工具,输出与所述检测结果对应的日志。
- [0037] 本发明实施例还公开了一种小程序的生成装置,包括:
- [0038] 数据获取模块,用于获取原生小程序的程序数据,以及针对目标小程序的程序转换信息,所述程序数据包括原生源码数据以及原生应用程序接口,所述程序转换信息包括语法转换信息以及基准接口信息;
- [0039] 语法转换模块,用于根据所述语法转换信息,将所述原生源码数据转换为适配于所述目标小程序的目标源码数据;
- [0040] 接口转换模块,用于根据所述基准接口信息,将所述原生应用程序接口转换为适配于所述目标小程序的目标应用程序接口;
- [0041] 开发者工具获取模块,用于获取针对所述目标小程序的开发者工具;
- [0042] 小程序生成模块,用于根据所述目标源码数据、所述目标应用程序接口以及所述开发者工具,生成所述目标小程序。
- [0043] 可选地,所述语法转换模块包括:
- [0044] 原生程序文件获取子模块,用于遍历所述原生小程序的源码目录,获取所述原生源码数据的原生程序文件;
- [0045] 程序文件转换子模块,用于根据所述语法转换信息,对所述原生程序文件进行语法结构转换,生成适配于所述目标小程序的目标程序文件,所述目标程序文件包括目标配置文件、目标结构文件、目标样式表文件以及目标逻辑文件;
- [0046] 源码数据生成子模块,用于采用所述目标配置文件、目标结构文件、目标样式表文件以及目标逻辑文件,生成适配于所述目标小程序的目标源码数据。
- [0047] 可选地,所述接口转换模块包括:
- [0048] 运行框架获取子模块,用于获取所述原生小程序的运行框架,所述运行框架包括所述原生应用程序接口的原生参数信息;
- [0049] 参数信息转换子模块,用于根据所述基准接口信息,将所述原生参数信息转换为适配于所述目标小程序的目标参数信息;
- [0050] 应用程序接口获得子模块,用于采用所述目标参数信息更新所述运行框架中的原生参数信息,获得所述目标小程序的目标应用程序接口。
- [0051] 可选地,所述基准接口信息包括参数映射关系与结果映射关系,所述原生参数信息包括原生参数名以及原生返回值,所述参数信息转换子模块具体用于:
- [0052] 采用所述参数映射关系,将所述原生参数名转换为目标参数名;
- [0053] 采用所述结果映射关系,将所述原生返回值转换为目标返回值;
- [0054] 所述应用程序接口获得子模块具体用于:
- [0055] 采用所述目标参数名更新所述运行框架中的原生参数名,以及采用所述目标返回值更新所述运行框架中的原生返回值;
- [0056] 确定一针对所述目标参数名与所述目标返回值的平台标识,生成与所述平台标识匹配的目标应用程序接口。
- [0057] 可选地,所述基准接口信息通过如下模块生成:
- [0058] 基准参数信息获取模块,用于获取基准小程序的基准应用程序接口的基准参数信息,所述基准参数信息包括基准参数名以及基准返回值;

- [0059] 目标参数信息获取模块,用于获取所述目标小程序的目标应用程序接口的目标参数信息,所述目标参数信息包括所述目标参数名以及所述目标返回值;
- [0060] 参数映射关系建立模块,用于当所述目标参数名与所述基准参数名不同时,则建立所述目标参数名与所述基准参数名之间的参数映射关系;
- [0061] 结果映射关系建立模块,用于当所述目标返回值与所述基准返回值不同时,则建立所述目标返回值与所述基准返回值之间的结果映射关系。
- [0062] 可选地,还包括:
- [0063] 返回值封装模块,用于当所述目标返回值与所述基准返回值不同时,则将所述原生返回值封装为与所述基准返回值对应的返回值。
- [0064] 可选地,所述开发者工具包括命令行展示工具、日志工具、语法检测工具中的至少一种,所述装置还包括:
- [0065] 源码数据展示模块,用于通过所述命令行展示工具展示所述目标小程序的目标源码数据;
- [0066] 源码数据检测模块,用于通过所述语法检测工具对所述目标源码数据进行语法结构检测,生成检测结果;
- [0067] 日志输出模块,用于通过所述日志工具,输出与所述检测结果对应的日志。
- [0068] 本发明实施例还公开了一种电子设备,包括:
- [0069] 一个或多个处理器;和
- [0070] 其上存储有指令的一个或多个机器可读介质,当由所述一个或多个处理器执行时,使得所述电子设备执行如上所述的一个或多个的方法。
- [0071] 本发明实施例还公开了一个或多个机器可读介质,其上存储有指令,当由一个或多个处理器执行时,使得所述处理器执行如上所述的一个或多个的方法。
- [0072] 本发明实施例包括以下优点:
- [0073] 在本发明实施例中,通过获取原生小程序的程序数据,以及针对目标小程序的程序转换信息,然后根据程序转换信息对原生小程序的原生源码数据以及原生应用程序接口进行转换,获得适配于目标小程序的目标源码数据以及目标应用程序接口,接着集成开发者工具,并输出目标小程序的实例,从而通过原生小程序进行源码数据的语法结构转换,对源码数据无侵入,且不需要额外引入框架与引入框架带来的维护风险,并且对应用程序接口进行适配,抹平了小程序之间的差异,保证了小程序转换之后的稳定性。

附图说明

- [0074] 图1是本发明的一种小程序的生成方法实施例的步骤流程图;
- [0075] 图2是本发明实施例中语法结构转换的示意图;
- [0076] 图3是本发明实施例中小程序转换架构的示意图;
- [0077] 图4是本发明实施例中运行检测的示意图;
- [0078] 图5是本发明的一种小程序的生成装置实施例的结构框图。

具体实施方式

- [0079] 为使本发明的上述目的、特征和优点能够更加明显易懂,下面结合附图和具体实

施方式对本发明作进一步详细的说明。

[0080] 参照图1,示出了本发明的一种小程序的生成方法实施例的步骤流程图,具体可以包括如下步骤:

[0081] 步骤101,获取原生小程序的程序数据,以及针对目标小程序的程序转换信息,所述程序数据包括原生源码数据以及原生应用程序接口,所述程序转换信息包括语法转换信息以及基准接口信息;

[0082] 作为一种示例,开发人员可以遵循相应的开发规范,使用相应的语言进行小程序的开发,从而可以开发出适配不同平台的小程序。对于不同平台之间的小程序,小程序的程序数据之间存在一定的差异,例如:小程序的页面显示数据不同、程序数据中的参数名不同、API(Application Programming Interface,应用程序接口)参数不同以及返回值不同等等。

[0083] 在本发明实施例中,原生小程序的程序数据可以包括原生源码数据以及原生应用程序接口,其中,原生源码数据可以为描述原生小程序页面的页面数据,小程序可以通过应用程序接口获取相应的返回结果,不同的应用程序接口可以对应不同的返回结果。

[0084] 需要说明的是,不同平台之间描述整体程序的程序主体部分,可以包括逻辑文件、公共配置文件以及公共样式表文件等等,程序主体部分为小程序运行的基本数据,此外,一个小程序的页面可以由原生源码数据组成,可以包括页面配置文件、页面结构文件、页面样式表文件以及页面逻辑文件等等。在小程序的运行框架中,提供了丰富的原生API,可以方便地调起应用程序提供的能力,如通过API获取用户信息、读取本地存储以及支付功能等等。

[0085] 在具体实现中,可以先获取目标平台的目标小程序的程序转换信息,以便根据该程序转换信息,将原生小程序的原生源码数据以及原生应用程序接口,转换为适配目标小程序的目标源码数据以及目标应用程序接口。其中,程序转换信息可以包括语法转换信息以及基准接口信息,语法转换信息可以对原生小程序的原生源码数据进行语法结构的转换,基准接口信息可以对应用程序接口进行适配,抹平小程序之间的差异性。

[0086] 在一种示例中,假设目标小程序为某度的小程序,则可以获取针对目标小程序的程序转换信息,以及原生小程序的原生源码数据以及原生API,然后根据程序转换信息,对原生源码数据与原生API进行转换,以适配目标小程序。

[0087] 步骤102,根据所述语法转换信息,将所述原生源码数据转换为适配于所述目标小程序的目标源码数据;

[0088] 在一种示例中,原生源码数据可以为描述小程序的页面数据,可以包括原生配置文件、原生结构文件、原生样式表文件以及原生逻辑文件等程序文件,则可以根据语法转换信息将原生源码数据中各个文件转换为适配于目标小程序的文件,实现源码数据的转换。

[0089] 在本发明的一种可选实施例中,可以通过遍历原生小程序的源码目录,获取原生源码数据的原生程序文件,接着根据语法转换信息,对原生程序文件进行语法结构转换,生成适配于目标小程序的目标程序文件,然后采用目标程序文件,生成适配于目标小程序的目标源码数据。其中,目标程序文件可以包括目标配置文件、目标结构文件、目标样式表文件以及目标逻辑文件。

[0090] 在具体实现中,配置文件可以包括小程序页面的配置信息,例如窗口表现、网络超

时时间等等;结构文件可以包括运行框架的标签语言,通过结合小程序的基础组件、时间系统等,可以构建出页面的结构,结构文件可以提供数据绑定、列表渲染、条件渲染、模板以及事件等功能;样式表文件可以为样式语言,用于描述结构文件的组件样式,例如可以用于决定结构文件的组件应该如何进行显示,可以包括尺寸单位、样式导入、内联样式以及选择器等信息;逻辑文件可以为小程序页面运行的相关逻辑。

[0091] 在一种示例中,如图2所示,示出了本发明实施例中语法结构转换的示意图,首先获取原生小程序的原生源码数据,原生源码数据可以包括json文件(即配置文件)、wxss(WeiXin Style Sheets)文件(即样式表文件)、wxml(WeiXin Markup Language)文件(即结构文件)以及js文件(即逻辑文件)等,接着可以根据语法转换信息,对程序文件进行语法结构转换。具体的,当程序文件为json文件时,可以进行格式转换、命名规则转换等实现语法结构转换。例如将原生小程序的全局对象,按照目标小程序的命名规则进行命名;当程序文件为wxss文件时,可以进行文件后缀处理、URL(Uniform Resource Locator,统一资源定位符)转换、选择器修复以及重命名等处理,例如可以把源码数据中“.wxss”文件,修改为后缀名为“.css”的文件,也可以对选择器中的组件进行修复,以保证小程序转换后能够稳定运行;当程序文件为wxml文件时,可以进行属性名替换、约定规则转换、后缀转换以及组件命名转换等等,例如可以把源码数据中“.wxml”文件,修改为后缀名为“.swan”的文件,也可以按照目标小程序中的命名规则对组件进行重命名等;当程序文件为js文件时,可以进行bug修复,例如可以对运行逻辑出现错误的地方进行修复等,从而可以将原生小程序的程序文件转换为适配于目标小程序的程序文件。

[0092] 需要说明的是,对于不同平台的小程序,并不需要对所有的程序文件进行转换,可以根据不同平台的特性,进行选择性地转换,每个平台可以对应一个处理子类,例如,对于平台①,其只需要转换json文件;对于平台②,其只需要转换wxss文件;对于平台③,其只需要转换wxml文件;对于平台④,其只需要转换js文件等等,从而可以根据针对目标小程序的语法转换信息,进行语法结构的转换,对源码数据无侵入,且不需要额外引入框架与引入框架带来的维护风险。

[0093] 步骤103,根据所述基准接口信息,将所述原生应用程序接口转换为适配于所述目标小程序的目标应用程序接口;

[0094] 在本发明实施例中,为了抹平不同平台小程序之间的差异,保证API的稳定性,可以将某一小程序的API作为基准,获取基准API的基准接口信息,然后将原生小程序的原生API转换为适配于目标小程序的目标API。

[0095] 在具体实现中,基准接口信息可以包括参数映射关系以及结果映射关系,可以通过获取基准小程序的基准应用程序接口的基准参数信息,基准参数信息包括基准参数名以及基准返回值,以及获取目标小程序的目标应用程序接口的目标参数信息,目标参数信息包括目标参数名以及目标返回值,接着将基准参数信息与目标参数信息进行匹配,当目标参数名与基准参数名不同时,则建立目标参数名与基准参数名之间的参数映射关系;当目标返回值与基准返回值不同时,则建立目标返回值与基准返回值之间的结果映射关系,或,将原生返回值封装为与基准返回值对应的返回值,从而建立目标小程序与基准小程序之间API的映射关系。可选地,对于其他平台的小程序也可以建立与基准小程序之间API的映射关系,从而将基准小程序作为其他平台之间的桥梁,进行API的参数转换。

[0096] 在具体实现中,基准小程序与其他平台小程序进行参数映射的过程中,若基准小程序拥有的API,其他平台没有对应的API,由于平台之间的不兼容,则降级为空方法;若基准小程序和其他平台小程序之间API的参数名不同,则建立参数名的参数映射关系,如表1所示:

基准小程序	平台①	平台②	平台③
API-1	API-①	API-I	API-一
API-2	API-②	API-II	API-二
API-3	API-③	API-III	API-三
API-4	API-④	API-IV	API-四
API-5	API-⑤	API-V	API-五

[0098] 表1

[0099] 若基准小程序和其他平台小程序之间API返回值不同时,则建立返回值之间的结果映射关系,或对返回结果进行封装,从而可以根据映射关系或封装结果,获取对应的返回值,进而将基准小程序作为其他平台之间的桥梁,进行API的参数转换,抹平了不同小程序之间的差异性。其中,返回值的结果映射关系参考参数名的参数映射关系,以基准小程序作为桥梁,建立对应关系,在此不再赘述。

[0100] 在本发明的一种可选实施例中,小程序的运行框架提供了丰富的API,不同平台的小程序之间,API的差异可以包括参数名不同、返回值不同等情况,则可以先获取原生小程序的运行框架,并从中获取原生API的原生参数信息,然后根据基准小程序的基准接口信息,将原生参数信息转换为适配于目标小程序的目标参数信息,以便采用目标参数信息更新运行框架中的原生参数信息,获取目标小程序的目标应用程序接口。

[0101] 需要说明的是,本发明实施例以原生小程序与目标小程序采用相同的框架为例进行说明,可以理解的是,当采用不同的运行框架时,可以先将原生参数信息转换为目标参数信息,然后将目标参数信息输入目标小程序的运行框架,构建目标小程序的API。

[0102] 在具体实现中,采用参数映射关系,将原生参数名转换为目标参数名,以及采用结果映射关系,将原生返回值转换为目标返回值,接着采用目标参数名更新运行框架中的原生参数名,以及采用目标返回值更新运行框架中的原生返回值,确定一针对目标参数名与目标返回值的平台标识,生成与平台标识匹配的目标应用程序接口。具体的,可以将根据参数映射关系以及结果映射关系,对API的参数进行转换,然后生成一个与目标小程序对应的平台标识,以便标识目标小程序属于哪个平台,接着进行编写兼容,从而实现小程序之间API的适配转换。

[0103] 步骤104,获取针对所述目标小程序的开发者工具;

[0104] 在一种示例中,开发者工具可以包括CLI(Command-Line Interface,命令行界面)命令行展示工具、日志工具、语法检测工具等等。其中,命令行展示工具可以用于展示目标小程序的目标源码数据,语法检测工具可以用于目标源码数据的语法结构检测,日志工具可以根据检测结果输出对应的日志。

[0105] 步骤105,根据所述目标源码数据、所述目标应用程序接口以及所述开发者工具,生成所述目标小程序。

[0106] 在一种示例中,如图3所示,示出了本发明实施例中小程序转换架构的示意图,架

构可以包括编译层、运行成以及集成层。当获取了原生小程序的程序数据后,可以将程序数据输入编译层,进行语法结构的集中处理,得到适配于目标小程序的目标源码数据,接着通过运行层进行API的适配转换,然后通过集成层中集成的工具进行运行调试以及语法检测。如图4所示,示出了本发明实施例中运行检测的示意图,可以通过集成层中的linter检测器(即语法检测工具)进行规则总结、规则包装,以及按照规则扫描两遍(命令行检测、文件整体检测),并通过日志logger工具输出总结日志。具体的,若检测到不支持attr、events、fns以及tags等时,可以转换为对应的检测规则,如wxmlFileRules、wxmlLineRules、wxssFileRules、wxssLineRules、scriptFileRules以及scriptLineRules等检测规则,然后扫描检测工具,包括LintLine以及LintFlie,接着接收检测规则,进行检测并总结检测结果,输入日志logger工具,以不同效果打印不同检测结果。当检测完毕后,则可以以目标源码数据作为目标小程序运行的代码,在运行框架中构建目标API,并集成各开发工具,生成目标小程序的实例,从而通过原生小程序进行源码数据的语法结构转换,对源码数据无侵入,且不需要额外引入框架与引入框架带来的维护风险,并且对应用程序接口进行适配,抹平了小程序之间的差异,保证了小程序转换之后的稳定性。

[0107] 在本发明实施例中,通过获取原生小程序的程序数据,以及针对目标小程序的程序转换信息,然后根据程序转换信息对原生小程序的原生源码数据以及原生应用程序接口进行转换,获得适配于目标小程序的目标源码数据以及目标应用程序接口,接着集成开发者工具,并输出目标小程序的实例,从而通过原生小程序进行源码数据的语法结构转换,对源码数据无侵入,且不需要额外引入框架与引入框架带来的维护风险,并且对应用程序接口进行适配,抹平了小程序之间的差异,保证了小程序转换之后的稳定性。

[0108] 需要说明的是,对于方法实施例,为了简单描述,故将其都表述为一系列的动作组合,但是本领域技术人员应该知悉,本发明实施例并不受所描述的动作顺序的限制,因为依据本发明实施例,某些步骤可以采用其他顺序或者同时进行。其次,本领域技术人员也应该知悉,说明书中所描述的实施例均属于优选实施例,所涉及的动作并不一定是本发明实施例所必须的。

[0109] 参照图5,示出了本发明的一种小程序的生成装置实施例的结构框图,具体可以包括如下模块:

[0110] 数据获取模块501,用于获取原生小程序的程序数据,以及针对目标小程序的程序转换信息,所述程序数据包括原生物码数据以及原生应用程序接口,所述程序转换信息包括语法转换信息以及基准接口信息;

[0111] 语法转换模块502,用于根据所述语法转换信息,将所述原生物码数据转换为适配于所述目标小程序的目标源码数据;

[0112] 接口转换模块503,用于根据所述基准接口信息,将所述原生应用程序接口转换为适配于所述目标小程序的目标应用程序接口;

[0113] 开发者工具获取模块504,用于获取针对所述目标小程序的开发者工具;

[0114] 小程序生成模块505,用于根据所述目标源码数据、所述目标应用程序接口以及所述开发者工具,生成所述目标小程序。

[0115] 在本发明的一种可选实施例中,所述语法转换模块502包括:

[0116] 原生程序文件获取子模块,用于遍历所述原生小程序的源码目录,获取所述原生

源码数据的原生程序文件；

[0117] 程序文件转换子模块,用于根据所述语法转换信息,对所述原生程序文件进行语法结构转换,生成适配于所述目标小程序的目标程序文件,所述目标程序文件包括目标配置文件、目标结构文件、目标样式表文件以及目标逻辑文件；

[0118] 源码数据生成子模块,用于采用所述目标配置文件、目标结构文件、目标样式表文件以及目标逻辑文件,生成适配于所述目标小程序的目标源码数据。

[0119] 在本发明的一种可选实施例中,所述接口转换模块503包括：

[0120] 运行框架获取子模块,用于获取所述原生小程序的运行框架,所述运行框架包括所述原生应用程序接口的原生参数信息；

[0121] 参数信息转换子模块,用于根据所述基准接口信息,将所述原生参数信息转换为适配于所述目标小程序的目标参数信息；

[0122] 应用程序接口获得子模块,用于采用所述目标参数信息更新所述运行框架中的原生参数信息,获得所述目标小程序的目标应用程序接口。

[0123] 在本发明的一种可选实施例中,所述基准接口信息包括参数映射关系与结果映射关系,所述原生参数信息包括原生参数名以及原生返回值,所述参数信息转换子模块具体用于：

[0124] 采用所述参数映射关系,将所述原生参数名转换为目标参数名；

[0125] 采用所述结果映射关系,将所述原生返回值转换为目标返回值；

[0126] 所述应用程序接口获得子模块具体用于：

[0127] 采用所述目标参数名更新所述运行框架中的原生参数名,以及采用所述目标返回值更新所述运行框架中的原生返回值；

[0128] 确定一针对所述目标参数名与所述目标返回值的平台标识,生成与所述平台标识匹配的目标应用程序接口。

[0129] 在本发明的一种可选实施例中,所述基准接口信息通过如下模块生成：

[0130] 基准参数信息获取模块,用于获取基准小程序的基准应用程序接口的基准参数信息,所述基准参数信息包括基准参数名以及基准返回值；

[0131] 目标参数信息获取模块,用于获取所述目标小程序的目标应用程序接口的目标参数信息,所述目标参数信息包括所述目标参数名以及所述目标返回值；

[0132] 参数映射关系建立模块,用于当所述目标参数名与所述基准参数名不同时,则建立所述目标参数名与所述基准参数名之间的参数映射关系；

[0133] 结果映射关系建立模块,用于当所述目标返回值与所述基准返回值不同时,则建立所述目标返回值与所述基准返回值之间的结果映射关系。

[0134] 在本发明的一种可选实施例中,还包括：

[0135] 返回值封装模块,用于当所述目标返回值与所述基准返回值不同时,则将所述原生返回值封装为与所述基准返回值对应的返回值。

[0136] 在本发明的一种可选实施例中,所述开发者工具包括命令行展示工具、日志工具、语法检测工具中的至少一种,所述装置还包括：

[0137] 源码数据展示模块,用于通过所述命令行展示工具展示所述目标小程序的目标源码数据；

[0138] 源码数据检测模块,用于通过所述语法检测工具对所述目标源码数据进行语法结构检测,生成检测结果;

[0139] 日志输出模块,用于通过所述日志工具,输出与所述检测结果对应的日志。

[0140] 对于装置实施例而言,由于其与方法实施例基本相似,所以描述的比较简单,相关之处参见方法实施例的部分说明即可。

[0141] 本发明实施例还提供了一种电子设备,包括:

[0142] 一个或多个处理器;和

[0143] 其上存储有指令的一个或多个机器可读介质,当由所述一个或多个处理器执行时,使得所述电子设备执行本发明实施例所述的方法。

[0144] 本发明实施例还提供了一个或多个机器可读介质,其上存储有指令,当由一个或多个处理器执行时,使得所述处理器执行本发明实施例所述的方法。

[0145] 本说明书中的各个实施例均采用递进的方式描述,每个实施例重点说明的都是与其他实施例的不同之处,各个实施例之间相同相似的部分互相参见即可。

[0146] 本领域内的技术人员应明白,本发明实施例的实施例可提供为方法、装置、或计算机程序产品。因此,本发明实施例可采用完全硬件实施例、完全软件实施例、或结合软件和硬件方面的实施例的形式。而且,本发明实施例可采用在一个或多个其中包含有计算机可用程序代码的计算机可用存储介质(包括但不限于磁盘存储器、CD-ROM、光学存储器、EEPROM、Flash以及eMMC等)上实施的计算机程序产品的形式。

[0147] 本发明实施例是参照根据本发明实施例的方法、终端设备(系统)、和计算机程序产品的流程图和/或方框图来描述的。应理解可由计算机程序指令实现流程图和/或方框图中的每一流程和/或方框、以及流程图和/或方框图中的流程和/或方框的结合。可提供这些计算机程序指令到通用计算机、专用计算机、嵌入式处理机或其他可编程数据处理终端设备的处理器以产生一个机器,使得通过计算机或其他可编程数据处理终端设备的处理器执行的指令产生用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的装置。

[0148] 这些计算机程序指令也可存储在能引导计算机或其他可编程数据处理终端设备以特定方式工作的计算机可读存储器中,使得存储在该计算机可读存储器中的指令产生包括指令装置的制造品,该指令装置实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能。

[0149] 这些计算机程序指令也可装载到计算机或其他可编程数据处理终端设备上,使得在计算机或其他可编程终端设备上执行一系列操作步骤以产生计算机实现的处理,从而在计算机或其他可编程终端设备上执行的指令提供用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的步骤。

[0150] 尽管已描述了本发明实施例的优选实施例,但本领域内的技术人员一旦得知了基本创造性概念,则可对这些实施例做出另外的变更和修改。所以,所附权利要求意欲解释为包括优选实施例以及落入本发明实施例范围的所有变更和修改。

[0151] 最后,还需要说明的是,在本文中,诸如第一和第二等之类的关系术语仅仅用来将一个实体或者操作与另一个实体或操作区分开来,而不一定要求或者暗示这些实体或操作之间存在任何这种实际的关系或者顺序。而且,术语“包括”、“包含”或者其任何其他变体意

在涵盖非排他性的包含,从而使得包括一系列要素的过程、方法、物品或者终端设备不仅包括那些要素,而且还包括没有明确列出的其他要素,或者是还包括为这种过程、方法、物品或者终端设备所固有的要素。在没有更多限制的情况下,由语句“包括一个……”限定的要素,并不排除在包括所述要素的过程、方法、物品或者终端设备中还存在另外的相同要素。

[0152] 以上对本发明所提供的一种小程序的生成方法和一种小程序的生成装置,进行了详细介绍,本文中应用了具体个例对本发明的原理及实施方式进行了阐述,以上实施例的说明只是用于帮助理解本发明的方法及其核心思想;同时,对于本领域的一般技术人员,依据本发明的思想,在具体实施方式及应用范围上均会有改变之处,综上所述,本说明书内容不应理解为对本发明的限制。

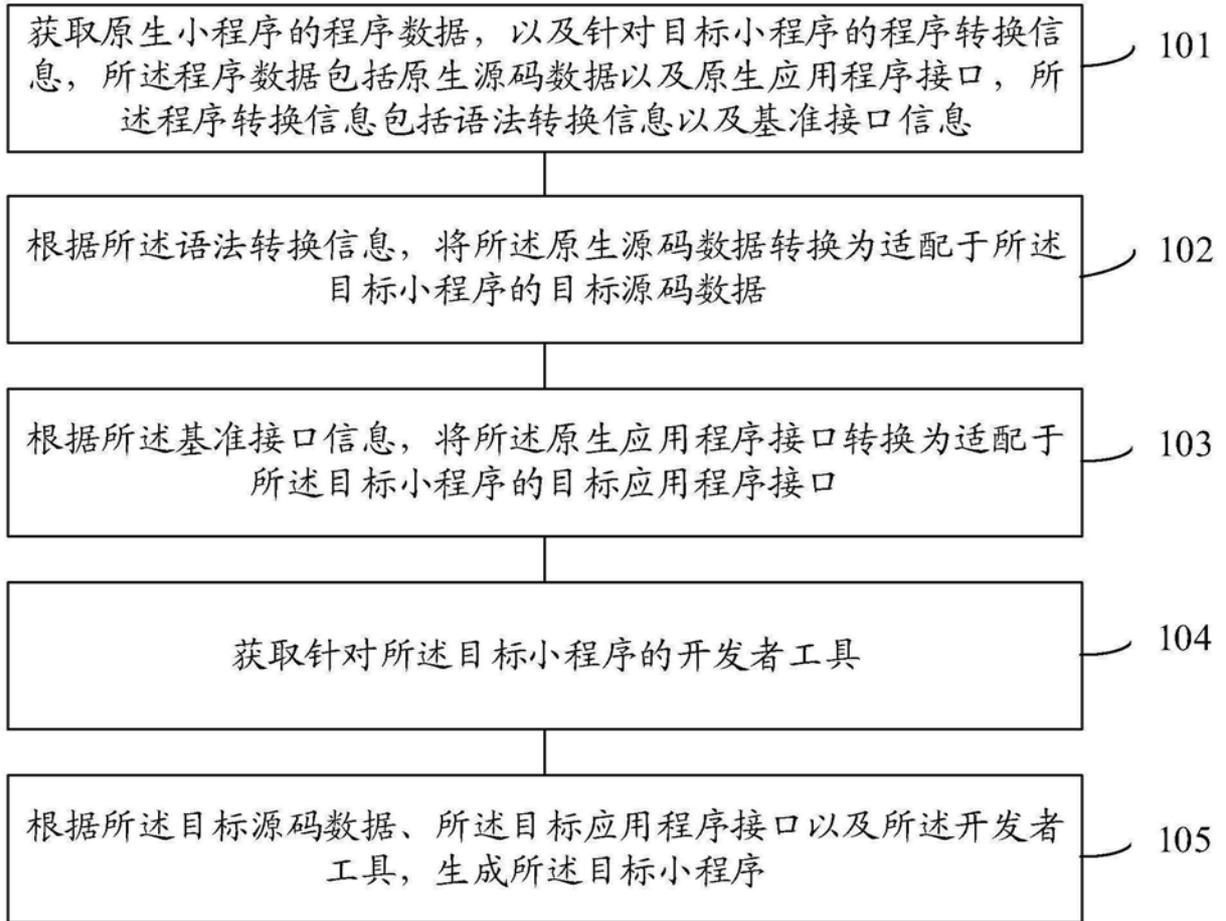


图1

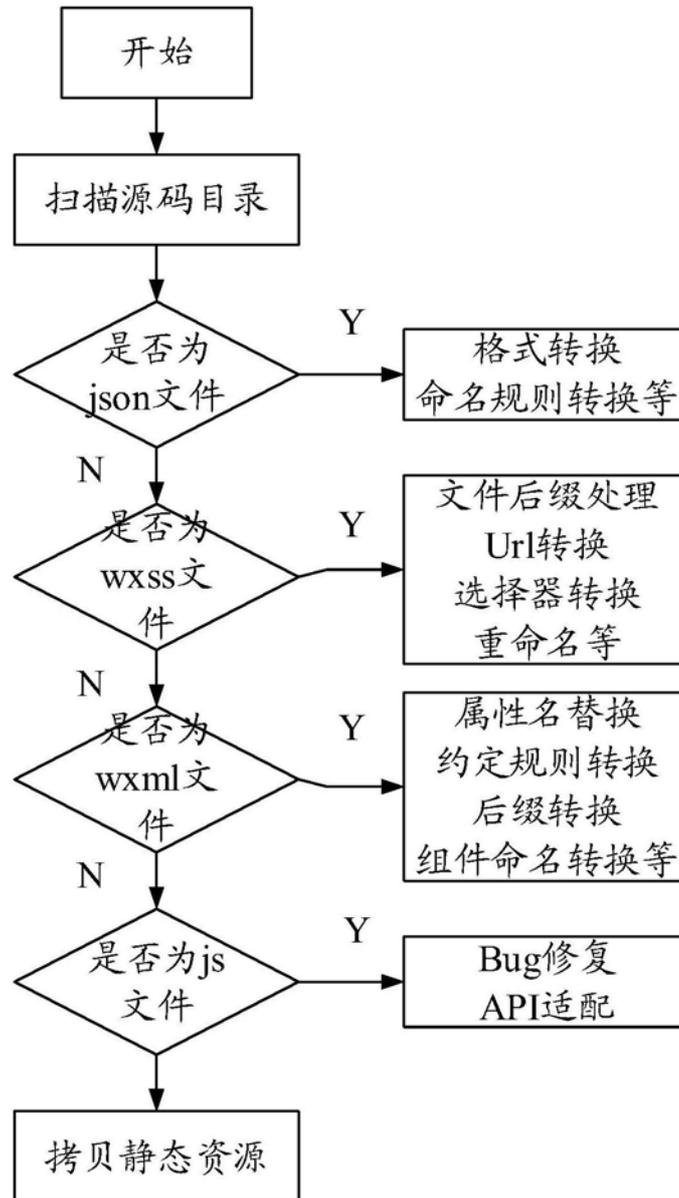


图2

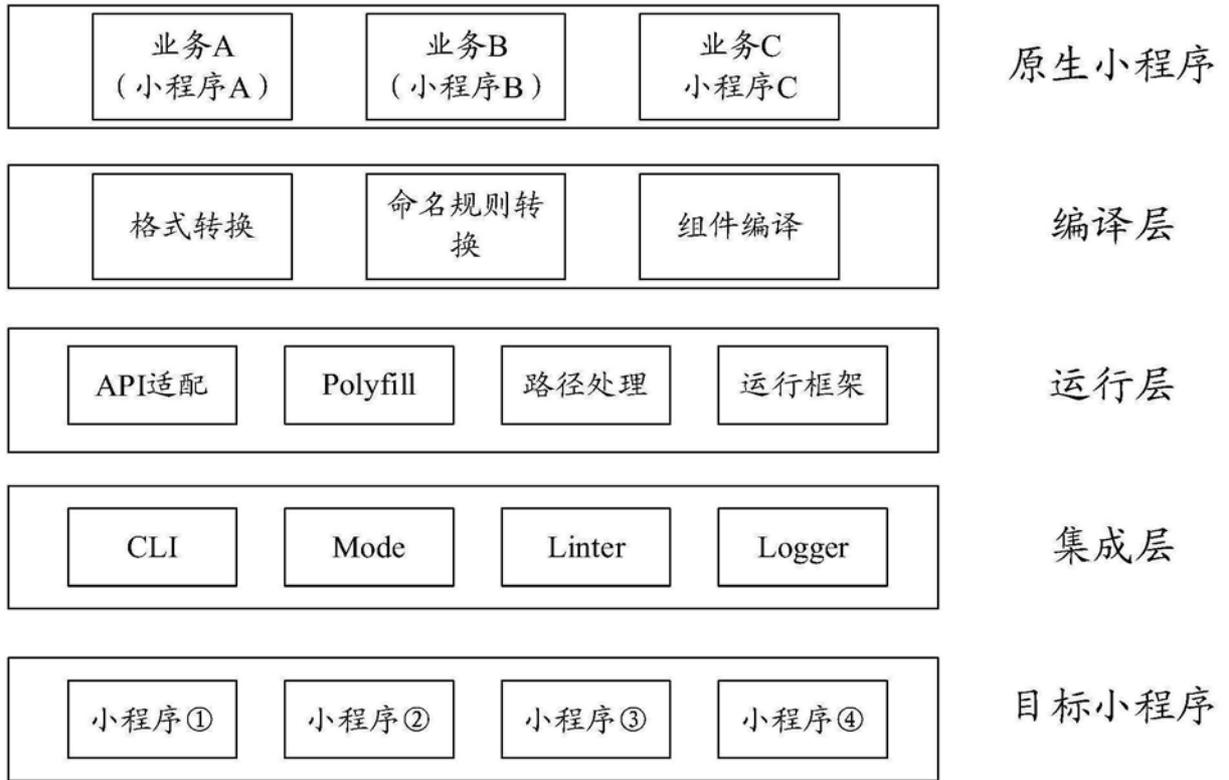


图3

Linter检测器

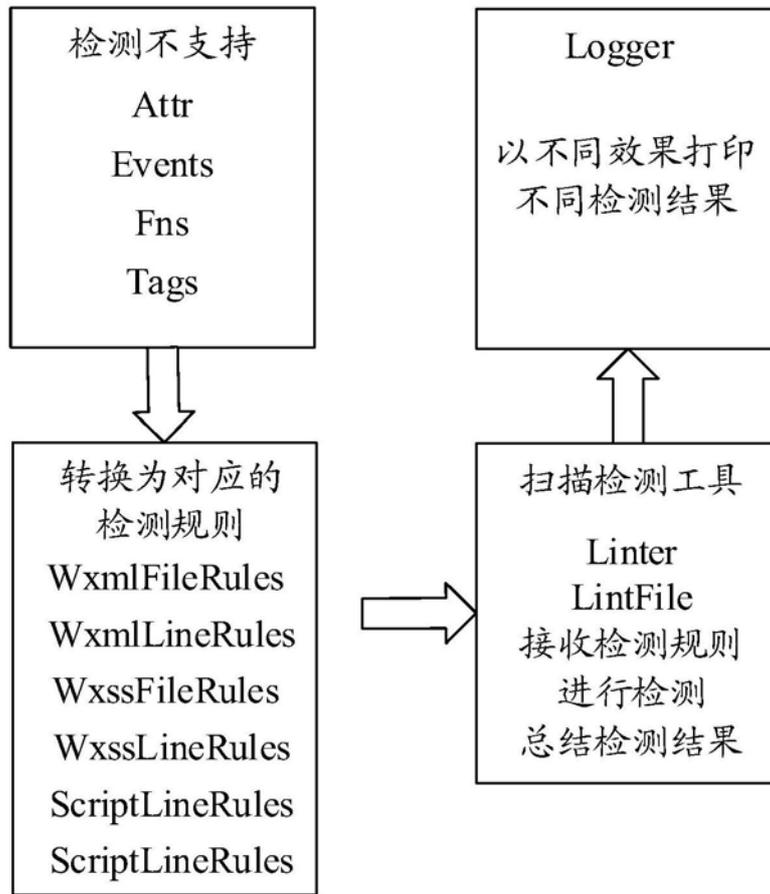


图4



图5