



(12)发明专利

(10)授权公告号 CN 105912468 B

(45)授权公告日 2019.08.16

(21)申请号 201610220622.4

(22)申请日 2016.04.11

(65)同一申请的已公布的文献号  
申请公布号 CN 105912468 A

(43)申请公布日 2016.08.31

(73)专利权人 华为软件技术有限公司  
地址 210012 江苏省南京市雨花台区安德  
门94号

(72)发明人 朱继先

(74)专利代理机构 北京龙双利达知识产权代理  
有限公司 11329  
代理人 王君 肖鹂

(51)Int.Cl.  
G06F 11/36(2006.01)

(56)对比文件

CN 102289365 A,2011.12.21,说明书第  
[0040]-[0200]段及附图1-7.

US 2006/0195894 A1,2006.08.31,全文.  
CN 101241444 A,2008.08.13,全文.

审查员 林坚

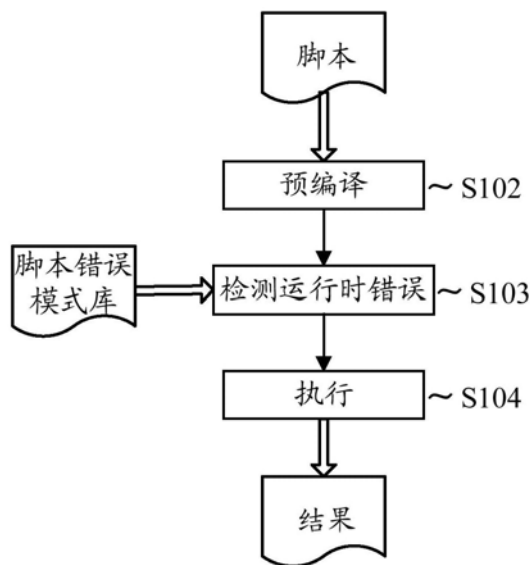
权利要求书2页 说明书8页 附图3页

(54)发明名称

开放平台运行脚本的方法和开放平台系统

(57)摘要

本发明提供一种开放平台运行脚本的方法和开放平台系统和开放平台系统。该方法包括:开放平台获取脚本;开放平台对脚本进行预编译,得到脚本的二进制文件;开放平台确定二进制文件中是否存在脚本错误模式库中的运行时错误,脚本错误模式库包括至少一个运行时错误;若开放平台确定二进制文件中存在脚本错误模式库中的运行时错误,则不执行所述二进制文件。本发明的技术方案,通过开放平台在执行脚本前对脚本中的运行时错误进行检查,不需要提供独立测试服务器就能实现脚本中的运行时错误的检测,以提高开放平台的系统稳定性,避免开放平台因为运行有运行时错误的脚本而造成的业务服务能力下降。



1. 一种开放平台运行脚本的方法,其特征在于,包括:
  - 所述开放平台获取脚本;
  - 所述开放平台对所述脚本进行预编译,得到所述脚本的二进制文件;
  - 所述开放平台确定所述二进制文件中是否存在脚本错误模式库中的运行时错误,所述脚本错误模式库包括至少一个运行时错误;
  - 若所述开放平台确定所述二进制文件中存在所述脚本错误模式库中的运行时错误,则不执行所述二进制文件;
  - 所述脚本错误模式库包括死循环错误的错误特征,所述死循环错误的错误特征包括循环标识、循环条件值为真和循环体中无退出语句;
  - 其中,所述开放平台确定所述二进制文件中是否存在脚本错误模式库中的运行时错误,包括:
    - 所述开放平台检查所述二进制文件中是否存在所述循环标识;
    - 所述开放平台检查所述二进制文件中存在所述循环标识时,检查所述二进制文件中所述循环标识对应的循环的循环条件值是否为真;
    - 所述开放平台检查所述二进制文件中所述循环的循环条件值为真时,检查所述循环的循环体中是否有退出循环的语句;
    - 所述开放平台检查所述循环的循环体中无退出循环的语句时,确定所述二进制文件中存在所述死循环错误。
2. 根据权利要求1所述的方法,其特征在于,所述方法还包括:
  - 所述开放平台确定所述二进制文件中存在所述脚本错误模式库中的运行时错误时,生成错误提示信息。
3. 根据权利要求1所述的方法,其特征在于,所述方法还包括:
  - 若所述开放平台确定所述二进制文件中不存在所述脚本错误模式库中的运行时错误,则执行所述二进制文件。
4. 根据权利要求1至3中任一项所述的方法,其特征在于,所述脚本为使用MVEL编写的脚本。
5. 一种开放平台系统,其特征在于,包括:
  - 获取模块,用于获取脚本;
  - 预编译模块,用于对所述脚本进行预编译,得到所述脚本的二进制文件;
  - 确定模块,用于确定所述二进制文件中是否存在脚本错误模式库中的运行时错误,所述脚本错误模式库包括至少一个运行时错误;
  - 执行模块,用于在所述确定模块确定所述二进制文件中存在所述脚本错误模式库中的运行时错误时,不执行所述二进制文件;
  - 所述脚本错误模式库包括死循环错误的错误特征,所述死循环错误的错误特征包括循环标识、循环条件值为真和循环体中无退出语句;
  - 其中,所述确定模块具体用于:
    - 检查所述二进制文件中是否存在所述循环标识;
    - 在所述二进制文件中存在所述循环标识时,检查所述二进制文件中所述循环标识对应的循环的循环条件值是否为真;

在所述二进制文件中所述循环的循环条件值为真时,检查所述循环的循环体中是否有退出循环的语句;

在所述循环的循环体中无退出循环的语句时,确定所述二进制文件中存在所述死循环错误。

6.根据权利要求5所述的开放平台系统,其特征在于,所述开放平台系统还包括错误提示模块,用于在所述确定模块确定所述二进制文件中存在所述脚本错误模式库中的运行时错误时,生成错误提示信息。

7.根据权利要求5所述的开放平台系统,其特征在于,所述执行模块还用于在所述确定模块确定所述二进制文件中不存在所述脚本错误模式库中的运行时错误时,执行所述二进制文件。

8.根据权利要求5至7中任一项所述的开放平台系统,其特征在于,所述脚本为使用MVCL编写的脚本。

## 开放平台运行脚本的方法和开放平台系统

### 技术领域

[0001] 本发明涉及信息技术领域,尤其涉及开放平台运行脚本的方法和开放平台系统。

### 背景技术

[0002] 开放平台可以通过公开软件系统的应用程序编程接口(Application Programming Interface,API)或函数(function)来使外部的程序可以增加该软件系统的功能或使用该软件系统的资源,而不需要更改该软件系统的源代码。开放平台也可以称为能力开放平台。

[0003] 开放平台在整合和利用现有软件系统资源的基础上,采用统一的多层级的开放接口来开放软件系统的能力,聚集互联网上有潜力、有创造力的开发者,让开发者能利用这些能力不断地创造出更好的商业应用和服务,实现开发者的无限的创造能力,形成一大批新的移动互联网应用及服务。

[0004] 使用开放平台的能力的过程为:开发者开发脚本,部署到开放平台;终端用户调用脚本的统一资源定位符(Uniform Resource Locator,URL);开放平台运行脚本。

[0005] 但是开发者可能会在开放平台上部署包括运行时错误的脚本,这会导致开放平台执行脚本时,影响开放平台的稳定性。

### 发明内容

[0006] 本发明提供一种开放平台运行脚本的方法和开放平台系统,能够提高开放平台的系统稳定性,避免开放平台因为运行有运行时错误的脚本而造成的业务服务能力下降。

[0007] 第一方面,提供一种开放平台运行脚本的方法,包括:所述开放平台获取脚本;所述开放平台对所述脚本进行预编译,得到所述脚本的二进制文件;所述开放平台确定所述二进制文件中是否存在脚本错误模式库中的运行时错误,所述脚本错误模式库包括至少一个运行时错误;若所述开放平台确定所述二进制文件中存在所述脚本错误模式库中的运行时错误,则不执行所述二进制文件。

[0008] 本发明实施例中,开放平台根据脚本错误模式库,在执行脚本前对脚本中的运行时错误进行检查,不需要提供独立测试服务器就能实现脚本中的运行时错误的检测,以提高开放平台的系统稳定性,避免开放平台因为运行有运行时错误的脚本而造成的业务服务能力下降。

[0009] 在一种可能的实现方式中,所述脚本错误模式库包括死循环错误的错误特征;其中,所述开放平台确定所述二进制文件中是否存在脚本错误模式库中的运行时错误,包括:所述开放平台根据所述二进制文件中存在所述错误特征,确定所述二进制文件中存在所述死循环错误。

[0010] 本发明实施例中,在脚本执行之前判断出脚本存在死循环,从而禁止死循环脚本运行,避免资源被完全占用,从而避免影响其他脚本程序运行。

[0011] 在一种可能的实现方式中,所述死循环错误的错误特征包括循环标识、循环条件

值为真和循环体中无退出语句;其中,所述开放平台根据所述二进制文件中存在所述错误特征,确定所述二进制文件中存在所述死循环错误,包括:所述开放平台检查所述二进制文件中是否存在所述循环标识;所述开放平台检查所述二进制文件中存在所述循环标识时,检查所述二进制文件中所述循环标识对应的循环的循环条件值是否为真;所述开放平台检查所述二进制文件中所述循环的循环条件值为真时,检查所述循环的循环体中是否有退出循环的语句;所述开放平台检查所述循环的循环体中无退出循环的语句时,确定所述二进制文件中存在所述死循环错误。

[0012] 在一种可能的实现方式中,所述方法还包括:所述开放平台确定所述二进制文件中存在所述脚本错误模式库中的运行时错误时,生成错误提示信息。

[0013] 本发明实施例中,开放平台检测出脚本中存在运行时错误时,提示该错误,从而可以使得开发者或终端用户了解不执行脚本的原因,提高开发者或终端用户的满意度。

[0014] 在一种可能的实现方式中,所述方法还包括:若所述开放平台确定所述二进制文件中不存在所述脚本错误模式库中的运行时错误,则执行所述二进制文件。

[0015] 本发明实施例中,执行不存在运行时错误的脚本,以保证正确脚本的功能能够实现,满足开发者或终端用户的需求。

[0016] 在一种可能的实现方式中,所述脚本为使用MVCL编写的脚本。

[0017] 第二方面,提供一种开放平台系统,包括:获取模块,用于获取脚本;预编译模块,用于对所述脚本进行预编译,得到所述脚本的二进制文件;确定模块,用于确定所述二进制文件中是否存在脚本错误模式库中的运行时错误,所述脚本错误模式库包括至少一个运行时错误;执行模块,用于在所述确定模块确定所述二进制文件中存在所述脚本错误模式库中的运行时错误时,不执行所述二进制文件。

[0018] 本发明实施例中,开放平台系统根据脚本错误模式库,在执行脚本前对脚本中的运行时错误进行检查,不需要提供独立测试服务器就能实现脚本中的运行时错误的检测,以提高开放平台系统的稳定性,避免开放平台系统因为运行有运行时错误的脚本而造成的业务服务能力下降。

[0019] 在一种可能的实现方式中,所述脚本错误模式库包括死循环错误的错误特征;其中,所述确定模块具体用于根据所述二进制文件中存在所述错误特征,确定所述二进制文件中存在所述死循环错误;所述执行模块具体用于在所述确定模块确定所述二进制文件中存在所述死循环错误时,不执行所述二进制文件。

[0020] 本发明实施例中,在脚本执行之前判断出脚本存在死循环,从而禁止死循环脚本运行,避免资源被完全占用,从而避免影响其他脚本程序运行。

[0021] 在一种可能的实现方式中,所述死循环错误的错误特征包括循环标识、循环条件值为真和循环体中无退出语句;其中,所述确定模块具体用于:检查所述二进制文件中是否存在所述循环标识;在所述二进制文件中存在所述循环标识时,检查所述二进制文件中所述循环标识对应的循环的循环条件值是否为真;在所述二进制文件中所述循环的循环条件值为真时,检查所述循环的循环体中是否有退出循环的语句;在所述循环的循环体中无退出循环的语句时,确定所述二进制文件中存在所述死循环错误。

[0022] 在一种可能的实现方式中,所述开放平台系统还包括错误提示模块,用于在所述确定模块确定所述二进制文件中存在所述脚本错误模式库中的运行时错误时,生成错误提

示信息。

[0023] 本发明实施例中,开放平台检测出脚本中存在运行时错误时,提示该错误,从而可以使得开发者或终端用户了解不执行脚本的原因,提高开发者或终端用户的满意度。

[0024] 在一种可能的实现方式中,所述执行模块还用于在所述确定模块确定所述二进制文件中不存在所述脚本错误模式库中的运行时错误时,执行所述二进制文件。

[0025] 本发明实施例中,执行不存在运行时错误的脚本,以保证正确脚本的功能能够实现,满足开发者或终端用户的需求。

[0026] 在一种可能的实现方式中,所述脚本为使用MVEL编写的脚本。

[0027] 第三方面,提供了一种开放平台系统,包括输入接口、输出接口、至少一个处理器、存储器,所述处理器用于执行所述存储器中的代码,当所述代码被执行时,所述处理器实现第一方面中的方法。

[0028] 第四方面,提供了一种计算机可读介质,所述计算机可读介质存储用于在开放平台中运行脚本的程序代码,所述程序代码包括用于执行第一方面中的方法的指令。

## 附图说明

[0029] 为了更清楚地说明本发明实施例的技术方案,下面将对本发明实施例中所需要使用的附图作简单地介绍,显而易见地,下面所描述的附图仅仅是本发明的一些实施例,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。

[0030] 图1是现有技术中开放平台运行脚本的方法的示意性流程图。

[0031] 图2是本发明实施例的开放平台运行脚本的方法的示意性流程图。

[0032] 图3是本发明实施例的开放平台运行脚本的方法的示意性流程图。

[0033] 图4是本发明实施例的开放平台系统的示意性结构图。

[0034] 图5是本发明实施例的开放平台系统的示意性结构图。

## 具体实施方式

[0035] 下面将结合本发明实施例中的附图,对本发明实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例是本发明一部分实施例,而不是全部的实施例。基于本发明中的实施例,本领域普通技术人员在没有作出创造性劳动前提下所获得的所有其他实施例,都属于本发明保护的范围。

[0036] 为了更好地理解本发明实施例,下面先介绍本发明实施例中的一些术语。

[0037] 脚本是批处理文件的延伸,是一种纯文本保存的程序,一般说的计算机脚本程序是确定的一系列控制计算机进行运算操作动作的组合,在其中可以实现一定的逻辑分支等。

[0038] 简单地说,脚本就是一条条的文字命令,这些文字命令是可以看到的(如可以用记事本打开查看、编辑),脚本程序在执行时,是由系统的一个解释器,将其一条条的翻译成机器可识别的指令,并按程序顺序执行。因为脚本在执行时多了一道翻译的过程,所以它比二进制程序执行效率要稍低一些。

[0039] 运行时错误与语法错误不同,其是指运行时才有可能产生的问题,并且这些问题

可以进行处理,从而避免中断程序,产生不好得用户体验。

[0040] 为了更好地理解本发明实施例于现有技术中的开放平台运行脚本的方法的区别,下面先结合图1介绍现有技术中开放平台运行脚本的方法。

[0041] 图1是现有技术中开放平台运行脚本的方法的示意性结构图。

[0042] S102,开放平台获取脚本。

[0043] S104,开放平台对脚本进行预编译,得到二进制文件。

[0044] S106,开放平台执行预编译得到的二进制文件,得到执行结果。

[0045] 有上述步骤可以看出,开放平台对脚本预编译成功,得到二进制文件后即执行该二进制文件,这样会导致脚本中存在如死循环之类的错误时,造成资源被完全占用,影响其他脚本程序运行,即导致开放平台的资源被不合理使用。

[0046] 现有技术里解决上述问题的方法是在独立的测试服务器上运行该脚本,确认该脚本没有如死循环的运行错误时,再将脚本部署到开放平台上,由开放平台运行。但这种解决方法需要提供与开放平台相同的运行环境等额外资源。

[0047] 针对上述问题,本发明实施例提出了一种开放平台运行脚本的方法,既不需要提供额外独立的资源,又能确保开放平台不会运行存在运行时错误的脚本。

[0048] 图2是本发明实施例的开放平台运行脚本的方法的示意性流程图。应理解,图2示出了开放平台运行脚本的方法的详细步骤或操作,但这些步骤或操作仅是示例,本发明实施例中还可以执行其他操作或者图2中的各个操作的变形。此外,图2中的部分步骤可以按照与图1呈现的不同的顺序来执行,并且有可能并非要执行图2中的全部操作。

[0049] S202,开放平台获取脚本。本发明实施例中的开放平台为开放平台系统中的组成部分,该开放平台系统还包括运行该开放平台的物理装置,本发明实施例中的开放平台均是在开放平台系统的物理装置上来完成该步骤及后续步骤的。开放平台获取的脚本可以是开发者在开放平台上使用开放平台支持的语言编写的脚本,也可以是开发者在其他平台上编写后,部署到开放平台上的脚本。

[0050] 本发明实施例中,可以是终端用户调用该脚本的逻辑接口,触发开放平台获取该脚本,以便于执行后续操作。

[0051] S204,开放平台对脚本进行预编译,得到脚本的二进制文件。开放平台对脚本进行预编译时,可以检查脚本中的语法错误等。

[0052] S206,开放平台确定二进制文件中是否存在脚本错误模式库中的运行时错误,所述脚本错误模式库包括至少一个运行时错误。

[0053] 开放平台对脚本进行预编译成功后(如没有检查出语法错误),可以根据脚本错误模式库对预编译所得的二进制文件进行检查,以确定该二进制文件中是否存在脚本错误模式库中包括的一个或多个运行时错误。

[0054] 开放平台对脚本进行预编译不成功,则不再继续执行步骤S206及后续的步骤。此时,开放平台可以退出脚本的运行过程。

[0055] S208,开放平台确定二进制文件中不存在脚本错误模式库中的运行时错误时,运行该二进制文件。此时,由于开放平台运行的二进制文件经过步骤S206的检查,确定无运行时错误,因此开放平台运行二进制文件的过程中不会出现诸如死循环之类的错误。

[0056] S210,开放平台确定二进制文件中存在脚本错误模式库中的运行时错误时,不运

行该二进制文件。

[0057] 开放平台确定二进制文件中存在脚本错误模式库中的运行时错误时,不再运行该二进制文件,以避免出现错误。此时,开放平台可以退出脚本的运行过程,也可以生成错误提示信息,向开发者或终端用户提示该错误,以便于开发者更改错误,或终端用户终止该脚本的运行。

[0058] 本发明实施例中,开放平台根据脚本错误模式库,在执行脚本前对脚本中的运行时错误进行检查,不需要提供独立测试服务器就能实现脚本中的运行时错误的检测,以提高开放平台的系统稳定性,避免开放平台因为运行有运行时错误的脚本而造成的业务服务能力下降。

[0059] 本发明实施例中,脚本错误模式库中具体可以包括运行时错误的错误特征。如脚本错误模式库中可以包括死循环错误的错误特征。此时,开放平台确定二进制文件中是否存在脚本错误模式库中的运行时错误的实施方式为:开放平台根据二进制文件中存在脚本错误模式库中的错误特征,确定二进制文件中存在死循环错误。

[0060] 本发明实施中,脚本错误模式库中用于表示死循环错误的错误特征可以包括循环标识(如“while”)、循环条件值为真(如“true”)和循环体中无退出语句,即包括循环标识的脚本中,若该循环标识对应的循环的条件值为真时,其循环体中没有催出语句,则说明该循环为死循环。

[0061] 此时,开放平台根据二进制文件中存在错误特征,确定二进制文件中存在死循环错误的实施方式为:开放平台检查二进制文件中是否存在循环标识;开放平台检查二进制文件中存在循环标识时,继续检查二进制文件中所述循环标识对应的循环的循环条件值是否为真;开放平台检查二进制文件中所述循环的循环条件值为真时,继续检查循环的循环体中是否有退出循环的语句;开放平台检查所述循环的循环体中无退出循环的语句时,确定二进制文件中存在死循环错误。

[0062] 下面根据一个具体的脚本例子来介绍本发明实施例中确定脚本中是否存在死循环错误的方法。

[0063] 如脚本错误模式库中包括的死循环错误的特征为:模式名(死循环检测),循环标识(while),条件为常量(true)和循环体无退出语句。

[0064] 开发者部署在开放平台上的脚本中包括以下语句:

```
while (true)
{
[0065]     i=i+1;
}
```

[0066] 则在开放平台对脚本预编译成功得到二进制文件后,取出脚本错误模式库中的死循环检测的“循环标识(while)”,并在二进制文件中进行匹配。在二进制文件中匹配到“while”后,继续取出“条件为常量(true)”在二进制文件中进行匹配,在二进制文件的上述“while”循环中匹配其条件是否为“true”;在二进制文件中的上述“while”循环中匹配到其条件为“true”后,在二进制文件中的上述“while”循环中查找有无退出语句;

[0067] 若上述“while”循环中无退出语句,则说明该循环为死循环,即该二进制文件中存



在死循环错误;若上述“while”循环中有退出语句,则说明该循环不是循环,即该二进制文件中不存在死循环错误。

[0068] 本发明实施例中,开放平台运行的脚本可以是MVFLEX表达式语言(MVFLEX Expression Language, MVEL)编写的。MVEL是一个功能强大的基于Java应用程序的表达式语言,是一种动态/静态的可嵌入的表达式语言和为Java平台提供运行时(Runtime)的语言。

[0069] 图3是本发明实施例的开放平台运行脚本的方法的示意性流程图。图3中与图1中相同的标记表示的含义相同,此处不再赘述。但是根据图1与图3中不同的标记S103可知,本发明实施例的开放平台运行脚本的方法与现有技术中开放平台运行脚本的方法的不同之处在于:开放平台对脚本进行预编译后,在执行脚本前,根据脚本错误模式库检查脚本中的运行时错误,且在确定脚本中不存在运行时错误时才执行脚本的二进制文件。

[0070] 因此,本发明实施例的开放平台运行脚本的方法不需要提供独立测试服务器就能实现脚本中的运行时错误的检测,以提高开放平台的系统稳定性,避免开放平台因为运行有运行时错误的脚本而造成的业务服务能力下降。

[0071] 图4是本发明实施例的开放平台系统的示意性结构图。应理解,图4中的开放平台系统400中的各个模块可以部署在同一个物理设备上,也可以部署在不同的物理设备上。图2中的方法中的开放平台为开放平台系统400的组成部分。

[0072] 获取模块410,用于获取脚本。

[0073] 预编译模块420,用于对所述脚本进行预编译,得到所述脚本的二进制文件。

[0074] 确定模块430,用于确定所述二进制文件中是否存在脚本错误模式库中的运行时错误,所述脚本错误模式库包括至少一个运行时错误。

[0075] 执行模块440,用于在所述确定模块430确定所述二进制文件中存在所述脚本错误模式库中的运行时错误时,不执行所述二进制文件。

[0076] 本发明实施例中,开放平台系统根据脚本错误模式库,在执行脚本前对脚本中的运行时错误进行检查,不需要提供独立测试服务器就能实现脚本中的运行时错误的检测,以提高开放平台系统的稳定性,避免开放平台因为运行有运行时错误的脚本而造成的业务服务能力下降。

[0077] 可选地,所述脚本错误模式库包括死循环错误的错误特征;其中,所述确定模块具体用于根据所述二进制文件中存在所述错误特征,确定所述二进制文件中存在所述死循环错误;所述执行模块具体用于在所述确定模块确定所述二进制文件中存在所述死循环错误时,不执行所述二进制文件。

[0078] 本发明实施例中,在脚本执行之前判断出脚本存在死循环,从而禁止死循环脚本运行,避免资源被完全占用,从而避免影响其他脚本程序运行。

[0079] 可选地,所述死循环错误的错误特征包括循环标识、循环条件值为真和循环体中无退出语句;其中,所述确定模块具体用于:检查所述二进制文件中是否存在所述循环标识;在所述二进制文件中存在所述循环标识时,检查所述二进制文件中所述循环标识对应的循环的循环条件值是否为真;在所述二进制文件中所述循环的循环条件值为真时,检查所述循环的循环体中是否有退出循环的语句;在所述循环的循环体中无退出循环的语句时,确定所述二进制文件中存在所述死循环错误。

[0080] 可选地,所述开放平台系统还包括错误提示模块,用于在所述确定模块430确定所述二进制文件中存在所述脚本错误模式库中的运行时错误时,生成错误提示信息。

[0081] 本发明实施例中,开放平台检测出脚本中存在运行时错误时,提示该错误,从而可以使得开发者或终端用户了解不执行脚本的原因,提高开发者或终端用户的满意度。

[0082] 可选地,所述执行模块还用于在所述开放平台确定所述二进制文件中不存在所述脚本错误模式库中的运行时错误时,执行所述二进制文件。

[0083] 本发明实施例中,执行不存在运行时错误的脚本,以保证正确脚本的功能能够实现,满足开发者或终端用户的需求。

[0084] 可选地,所述脚本为使用MVEL编写的脚本。

[0085] 图5是本发明实施例的开放平台系统的示意性结构图。图5的开放平台系统500包括输入接口510、至少一个处理器520、存储器530和输出接口540。

[0086] 输入接口510,用于接收终端用户发送的指示开放平台系统运行脚本的信息。

[0087] 至少一个处理器520,用于对所述脚本进行预编译,得到所述脚本的二进制文件。

[0088] 存储器530,用于存储脚本错误模式库,所述脚本错误模式库包括至少一个运行时错误。

[0089] 至少一个处理器520还用于确定所述二进制文件中是否存在脚本错误模式库中的运行时错误。

[0090] 至少一个处理器520还用于在确定所述二进制文件中存在所述脚本错误模式库中的运行时错误时,不执行所述二进制文件。

[0091] 本发明实施例中,开放平台系统根据脚本错误模式库,在执行脚本前对脚本中的运行时错误进行检查,不需要提供独立测试服务器就能实现脚本中的运行时错误的检测,以提高开放平台系统的稳定性,避免开放平台因为运行有运行时错误的脚本而造成的业务能力下降。

[0092] 可选地,所述脚本错误模式库包括死循环错误的错误特征;其中,至少一个处理器520具体用于根据所述二进制文件中存在所述错误特征,确定所述二进制文件中存在所述死循环错误;以及在所述确定模块确定所述二进制文件中存在所述死循环错误时,不执行所述二进制文件。

[0093] 本发明实施例中,在脚本执行之前判断出脚本存在死循环,从而禁止死循环脚本运行,避免资源被完全占用,从而避免影响其他脚本程序运行。

[0094] 可选地,所述死循环错误的错误特征包括循环标识、循环条件值为真和循环体中无退出语句;其中,至少一个处理器520具体用于:检查所述二进制文件中是否存在所述循环标识;在所述二进制文件中存在所述循环标识时,检查所述二进制文件中所述循环标识对应的循环的循环条件值是否为真;在所述二进制文件中所述循环的循环条件值为真时,检查所述循环的循环体中是否有退出循环的语句;在所述循环的循环体中无退出循环的语句时,确定所述二进制文件中存在所述死循环错误。

[0095] 可选地,至少一个处理器520还用于在所述确定模块确定所述二进制文件中存在所述脚本错误模式库中的运行时错误时,生成错误提示信息。此时,输出接口540用于向终端用户输出该提示信息。

[0096] 本发明实施例中,开放平台系统检测出脚本中存在运行时错误时,提示该错误,从

而可以使得开发者或终端用户了解不执行脚本的原因,提高开发者或终端用户的满意度。

[0097] 可选地,至少一个处理器520还用于在确定所述二进制文件中不存在所述脚本错误模式库中的运行时错误时,执行所述二进制文件。

[0098] 本发明实施例中,执行不存在运行时错误的脚本,以保证正确脚本的功能能够实现,满足开发者或终端用户的需求。

[0099] 可选地,所述脚本为使用MVEL编写的脚本。

[0100] 本发明实施例中,开放平台系统根据脚本错误模式库,在执行脚本前对脚本中的运行时错误进行检查,不需要提供独立测试服务器就能实现脚本中的运行时错误的检测,以提高开放平台的系统稳定性,避免开放平台因为运行有运行时错误的脚本而造成的业务服务能力下降。

[0101] 本领域普通技术人员可以意识到,结合本文中所公开的实施例描述的各示例的单元及算法步骤,能够以电子硬件、或者计算机软件和电子硬件的结合来实现。这些功能究竟以硬件还是软件方式来执行,取决于技术方案的特定应用和设计约束条件。专业技术人员可以对每个特定的应用来使用不同方法来实现所描述的功能,但是这种实现不应认为超出本发明的范围。

[0102] 所属领域的技术人员可以清楚地了解到,为描述的方便和简洁,上述描述的系统、装置和单元的具体工作过程,可以参考前述方法实施例中的对应过程,在此不再赘述。

[0103] 在本申请所提供的几个实施例中,应该理解到,所揭露的系统、装置和方法,可以通过其它的方式实现。例如,以上所描述的装置实施例仅仅是示意性的,例如,所述单元的划分,仅仅为一种逻辑功能划分,实际实现时可以有另外的划分方式,例如多个单元或组件可以结合或者可以集成到另一个系统,或一些特征可以忽略,或不执行。另一点,所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些接口,装置或单元的间接耦合或通信连接,可以是电性,机械或其它的形式。

[0104] 所述作为分离部件说明的单元可以是或者也可以不是物理上分开的,作为单元显示的部件可以是或者也可以不是物理单元,即可以位于一个地方,或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部单元来实现本实施例方案的目的。

[0105] 另外,在本发明各个实施例中的各功能单元可以集成在一个处理单元中,也可以是各个单元单独物理存在,也可以两个或两个以上单元集成在一个单元中。

[0106] 所述功能如果以软件功能单元的形式实现并作为独立的产品销售或使用,可以存储在一个计算机可读取存储介质中。基于这样的理解,本发明的技术方案本质上或者说对现有技术做出贡献的部分或者该技术方案的部分可以以软件产品的形式体现出来,该计算机软件产品存储在一个存储介质中,包括若干指令用以使得一台计算机设备(可以是个人计算机,服务器,或者网络设备)执行本发明各个实施例所述方法的全部或部分步骤。而前述的存储介质包括:U盘、移动硬盘、只读存储器(ROM,Read-Only Memory)、随机存取存储器(RAM,Random Access Memory)、磁碟或者光盘等各种可以存储程序代码的介质。

[0107] 以上所述,仅为本发明的具体实施方式,但本发明的保护范围并不局限于此,任何熟悉本技术领域的技术人员在本发明揭露的技术范围内,可轻易想到变化或替换,都应涵盖在本发明的保护范围之内。因此,本发明的保护范围应所述以权利要求的保护范围为准。

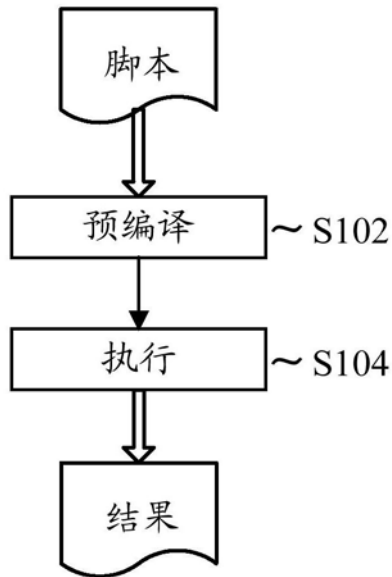


图1

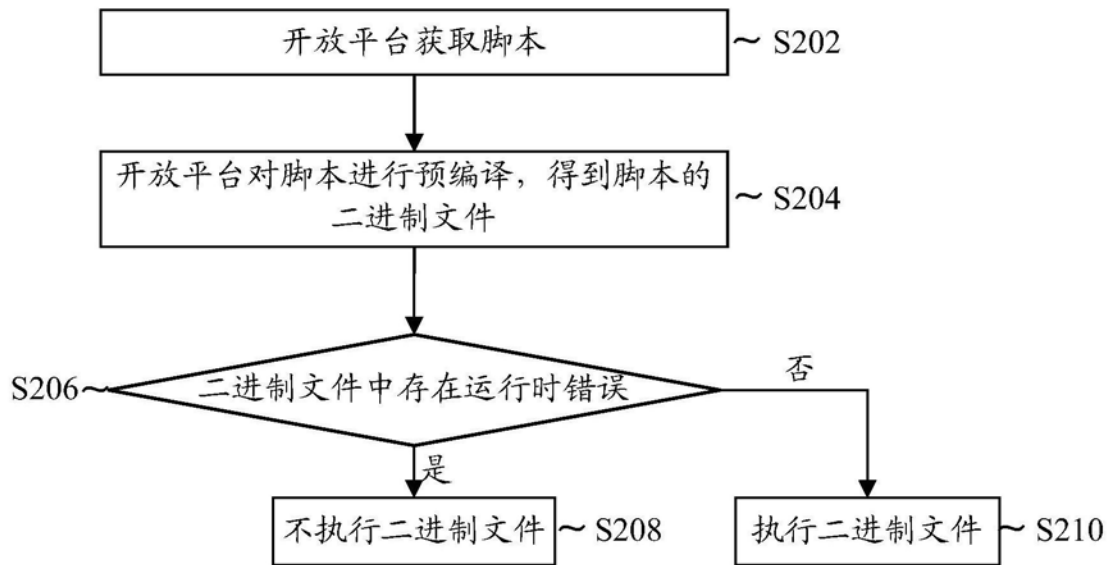


图2

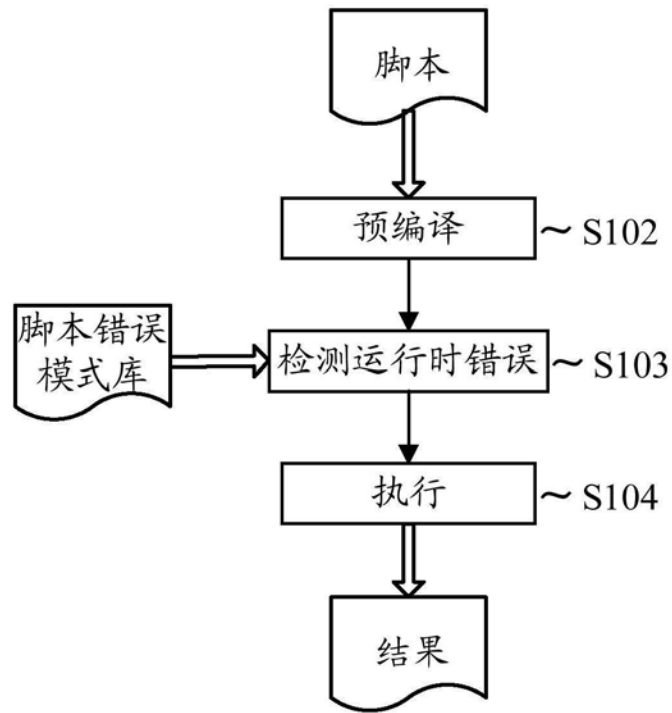


图3

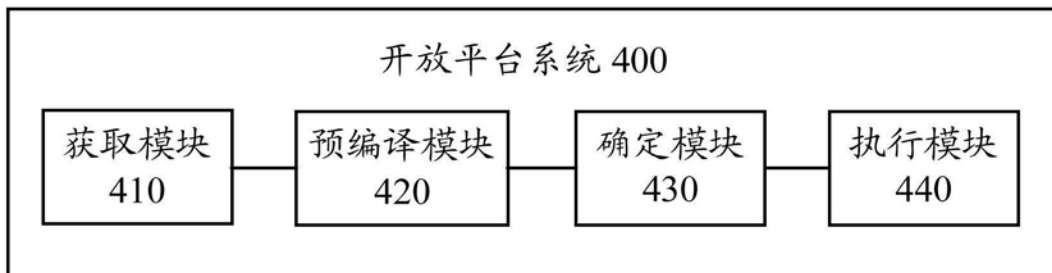


图4

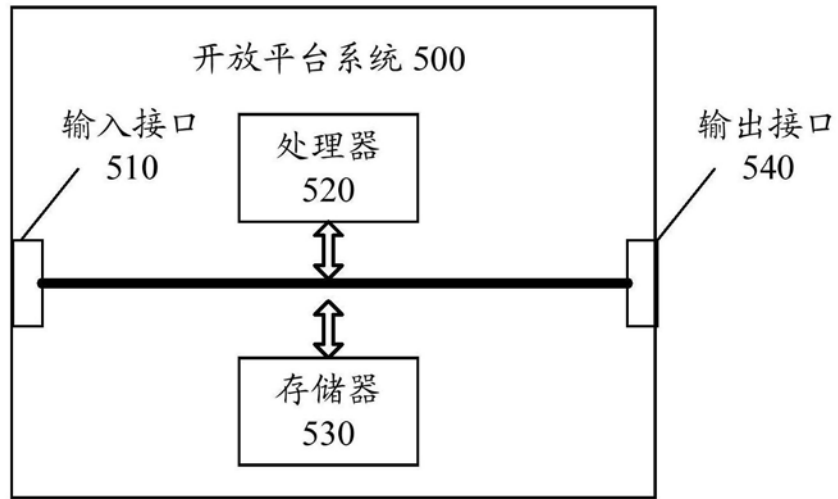


图5