(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 109871319 A
(43)申请公布日 2019.06.11

(21)申请号 201910048267.0

(22)申请日 2019.01.18

(71)申请人 深圳壹账通智能科技有限公司
地址 518000 广东省深圳市前海深港合作区前湾一路1号A栋201室(入驻深圳市前海商务秘书有限公司)

(72)发明人 刘慧众

(74)专利代理机构 深圳众鼎专利商标代理事务所(普通合伙) 44325
代理人 黄章辉

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

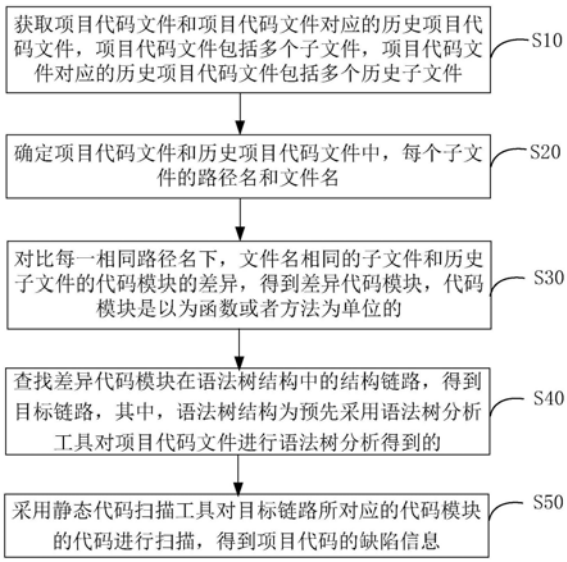
权利要求书2页 说明书9页 附图5页

(54)发明名称

项目代码扫描方法、装置、计算机设备及存储介质

(57)摘要

本申请公开了一种项目代码扫描方法、装置、计算机设备及存储介质，该项目代码扫描方法首先通过获取项目代码文件和所述项目代码文件对应的历史项目代码文件，所述项目代码文件包括多个子文件，所述项目代码文件对应的历史项目代码文件包括多个历史子文件；再确定所述项目代码文件和所述历史项目代码文件中，每个子文件的路径名和文件名；然后对比每一相同路径名下，文件名相同的所述子文件和所述历史子文件的代码模块的差异，得到差异代码模块；之后查找所述差异代码模块在语法树结构中的结构链路，得到目标链路；最后采用静态代码扫描工具对所述目标链路所对应的项目代码模块进行扫描，得到项目代码的扫描结果，采用静态代码扫描工具对项目代码进行针对性扫描，能够提高项目代码扫描的效率和准确率。



CN 109871319 A

1.一种项目代码扫描方法,其特征在于,包括:

获取项目代码文件和所述项目代码文件对应的历史项目代码文件,所述项目代码文件包括多个子文件,所述项目代码文件对应的历史项目代码文件包括多个历史子文件;

确定所述项目代码文件和所述历史项目代码文件中,每个子文件的路径名和文件名;

对比每一相同路径名下,文件名相同的所述子文件和所述历史子文件的代码模块的差异,得到差异代码模块,所述代码模块是以函数或方法为单位的;

查找所述差异代码模块在语法树结构中的结构链路,得到目标链路,其中,所述语法树结构为预先采用语法树分析工具对项目代码文件进行语法树分析得到的;

采用静态代码扫描工具对所述目标链路所对应的代码模块进行扫描,得到项目代码的扫描结果。

2.如权利要求1所述的项目代码扫描方法,其特征在于,所述对比每一相同路径名下,文件名相同的所述子文件和所述历史子文件的代码模块的差异,得到差异代码模块,包括:

判断所述每一个相同文件路径名下,是否存在文件名相同的所述子文件和所述历史子文件;

若相同文件路径名下,存在文件名相同的所述子文件和所述历史子文件,则对比所述子文件与所述历史子文件的代码模块之间的差异,将每一存在差异的代码模块作为所述差异代码模块。

3.如权利要求2所述的项目代码扫描方法,其特征在于,

若相同文件路径名下只存在所述子文件,则将所述子文件中的每一代码模块作为所述差异代码模块;

若相同文件路径名下只存在所述历史子文件,则将所述历史子文件中的每一代码模块作为所述差异代码模块。

4.如权利要求1所述的项目代码扫描方法,其特征在于,所述查找所述差异代码模块在语法树结构中的结构链路,得到目标链路,包括:

以所述差异代码模块的名字为关键字,查找所述差异代码模块在所述语法树结构的结构链路中所处的节点,其中所述差异代码模块的名字为所述差异代码模块所对应的函数名或者方法名;

将查找出来的所述节点所在的每一条结构链路作为所述目标链路。

5.如权利要求1所述的项目代码扫描方法,其特征在于,所述采用静态代码扫描工具对所述目标链路所对应的代码模块进行扫描,得到项目代码的缺陷信息,包括:

提取所述目标链路上的所有节点对应的代码模块;

采用静态代码扫描工具对提取出来的所述代码模块进行静态代码扫描,得到所述扫描结果。

6.一种项目代码扫描装置,其特征在于,包括:

获取模块,用于获取项目代码文件和所述项目代码文件对应的历史项目代码文件,所述项目代码文件包括多个子文件,所述项目代码文件对应的历史项目代码文件包括多个历史子文件;

确定模块,用于确定所述项目代码文件和所述历史项目代码文件中,每个子文件的路径名和文件名;

对比模块，用于对比每一相同路径名下，文件名相同的所述子文件和所述历史子文件的代码模块的差异，得到差异代码模块，所述代码模块是以为函数或者方法为单位的；

查找模块，用于查找所述差异代码模块在语法树结构中的结构链路，得到目标链路，其中，所述语法树结构为预先采用语法树分析工具对项目代码文件进行语法树分析得到的；

扫描模块，用于采用静态代码扫描工具对所述目标链路所对应的代码模块进行扫描，得到项目代码的缺陷信息。

7.如权利要求6所述的项目代码扫描装置，其特征在于，所述对比模块包括：

判断单元，用于判断所述每一个相同文件路径名下，是否存在文件名相同的所述子文件和所述历史子文件；

对比单元，用于若相同文件路径名下，存在文件名相同的所述子文件和所述历史子文件，则对比所述子文件与所述历史子文件的代码模块之间的差异，将每一存在差异的所述代码模块作为所述差异代码模块。

8.如权利要求6所述的项目代码扫描装置，其特征在于，所述查找模块包括：

查找单元，用于以所述差异代码模块的名字为关键字，查找所述差异代码模块在所述语法树结构的结构链路中所处的节点，其中差异代码模块的名字为所述差异代码模块所对应的函数名或者方法名；

确定单元，用于将查找出来的所述节点所在的每一条结构链路作为所述目标链路。

9.一种计算机设备，包括存储器、处理器以及存储在所述存储器中并可在所述处理器上运行的计算机程序，其特征在于，所述处理器执行所述计算机程序时实现如权利要求1至5任一项所述项目代码扫描方法。

10.一种计算机可读存储介质，所述计算机可读存储介质存储有计算机程序，其特征在于，所述计算机程序被处理器执行时实现如权利要求1至5任一项所述项目代码扫描方法。

项目代码扫描方法、装置、计算机设备及存储介质

技术领域

[0001] 本申请涉及代码质量检测领域，尤其涉及一种项目代码扫描方法、装置、计算机设备及存储介质

背景技术

[0002] 现如今，随着计算机科学技术的高速发展，应用软件开发的分工不断细化，在一个项目开发过程中，工程师们需要负责开发的模块各不相同。但是这样的分工往往容易出现源代码混乱，从而导致程序出错。为了避免这些错误，需要对应用软件开发项目的源代码进行管理。

[0003] 传统方案通常的做法是对项目代码做全量的代码扫描。在一些代码量大的项目中，采用全量代码扫描的方法目的性不强，效率低，还十分容易出错。

发明内容

[0004] 本申请提供一种项目代码扫描方法、装置、计算机设备及存储介质，采用静态代码扫描工具对修改过的项目代码进行针对性地扫描，能够提高项目代码扫描的效率和准确率。

[0005] 一种项目代码扫描方法，包括：

[0006] 获取项目代码文件和所述项目代码文件对应的历史项目代码文件，所述项目代码文件包括多个子文件，所述项目代码文件对应的历史项目代码文件包括多个历史子文件；

[0007] 确定所述项目代码文件和所述历史项目代码文件中，每个子文件的路径名和文件名；

[0008] 对比每一相同路径名下，文件名相同的所述子文件和所述历史子文件的代码模块的差异，得到差异代码模块，所述代码模块是以为函数或者方法为单位的；

[0009] 查找所述差异代码模块在语法树结构中的结构链路，得到目标链路，其中，所述语法树结构为预先采用语法树分析工具对项目代码文件进行语法树分析得到的；

[0010] 采用静态代码扫描工具对所述目标链路所对应的代码模块进行扫描，得到项目代码的扫描结果。

[0011] 一种项目代码扫描装置，包括：

[0012] 获取模块，用于获取项目代码文件和所述项目代码文件对应的历史项目代码文件，所述项目代码文件包括多个子文件，所述项目代码文件对应的历史项目代码文件包括多个历史子文件；

[0013] 确定模块，用于确定所述项目代码文件和所述历史项目代码文件中，每个子文件的路径名和文件名；

[0014] 对比模块，用于对比每一相同路径名下，文件名相同的所述子文件和所述历史子文件的代码模块的差异，得到差异代码模块，所述代码模块是以为函数或者方法为单位的；

[0015] 查找模块，用于查找所述差异代码模块在语法树结构中的结构链路，得到目标链

路，其中，所述语法树结构为预先采用语法树分析工具对项目代码文件进行语法树分析得到的；

[0016] 扫描模块，用于采用静态代码扫描工具对所述目标链路所对应的代码模块进行扫描，得到项目代码的扫描结果。

[0017] 一种计算机设备，包括存储器、处理器以及存储在所述存储器中并可在所述处理器上运行的计算机程序，所述处理器执行所述计算机程序时实现上述项目代码扫描方法。

[0018] 一种计算机可读存储介质，所述计算机可读存储介质存储有计算机程序，所述计算机程序被处理器执行时实现上述项目代码扫描方法。

[0019] 上述项目代码扫描方法、装置、计算机设备及存储介质，通过对比来得出项目代码文件的自文件中，与历史项目代码文件的历史子文件存在差异的差异代码模块，再采用静态代码扫描工具对差异代码模块和与差异代码模块存在调用关系的代码模块进行针对性扫描，能够提高项目代码扫描的效率和准确率。

附图说明

[0020] 为了更清楚地说明本申请实施例的技术方案，下面将对本申请实施例的描述中所需要使用的附图作简单地介绍，显而易见地，下面描述中的附图仅仅是本申请的一些实施例，对于本领域普通技术人员来讲，在不付出创造性劳动性的前提下，还可以根据这些附图获得其他的附图。

[0021] 图1是本申请一实施例中项目代码扫描方法的一应用环境示意图；

[0022] 图2是本申请一实施例中项目代码扫描方法的一示例图；

[0023] 图3是本申请一实施例中项目代码扫描方法的语法树结构示例图；

[0024] 图4是本申请一实施例中项目代码扫描方法的一示例图；

[0025] 图5是本申请一实施例中项目代码扫描方法的一示例图；

[0026] 图6是本申请一实施例中项目代码扫描方法的一示例图；

[0027] 图7是本申请一实施例中项目代码扫描方法的一示例图；

[0028] 图8是本申请一实施例中项目代码扫描装置的一原理框图；

[0029] 图9是本申请一实施例中项目代码扫描装置的一示意图；

[0030] 图10是本申请一实施例中项目代码扫描装置的一示意图；

[0031] 图11是本申请一实施例中计算机设备的一示意图。

具体实施方式

[0032] 下面将结合本申请实施例中的附图，对本申请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申请一部分实施例，而不是全部的实施例。基于本申请中的实施例，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，都属于本申请保护的范围。

[0033] 本申请实施例提供的项目代码扫描方法，可应用在如图1的应用环境中，其中，计算机设备通过网络与服务器进行通信。计算机设备在接收到项目代码扫描指令之后，服务器开始获取项目代码文件和项目代码文件对应的历史项目代码文件，并对项目代码文件和项目代码文件对应的历史项目代码文件进行对比操作，再经过一系列操作之后对项目代码

进行扫描，得到项目代码的扫描结果。其中，计算机设备可以但不限于各种个人计算机、笔记本电脑、智能手机、平板电脑和便携式可穿戴设备。服务器可以用独立的服务器或者是多个服务器组成的服务器集群来实现。

[0034] 在一实施例中，如图2所示，提供一种项目代码扫描方法，以该方法应用在图1中的服务器为例进行说明，包括如下步骤：

[0035] S10：获取项目代码文件和项目代码文件对应的历史项目代码文件，项目代码文件包括多个子文件，项目代码文件对应的历史项目代码文件包括多个历史子文件。

[0036] 其中，项目代码文件是指程序开发人员对项目代码进行增加、删除或修改之后产生的项目文件。项目代码文件对应的历史项目代码文件指程序开发人员对项目代码进行修改前的项目文件，该历史项目代码文件与项目代码文件的关系为：项目代码文件是在历史项目代码文件的基础之上进行增加、删除或修改。

[0037] 其中，项目代码文件包括多个子文件。例如，项目代码文件中包含kol_online、sub_online等子文件，而kol_online子文件中又包含index.html、main.js等子文件。

[0038] 项目代码文件对应的历史项目代码文件包括多个历史子文件。例如，项目代码文件对应的历史项目代码文件中包含kol_online、sub_online等多个历史子文件，而kol_online历史子文件中又包含index.html、main.js等多个历史子文件。

[0039] 具体地，可以从版本管理工具中获取项目代码文件和历史项目代码文件。其中，版本管理工具是指对不同版本项目代码进行保存和管理的工具，例如，常见的版本管理工具一般有Subversion和Git等。

[0040] S20：确定项目代码文件和历史项目代码文件中，每个子文件的路径名和文件名。

[0041] 具体地，通过项目代码文件的子文件之间的包含关系确定项目代码文件中的每个子文件的文件路径名以及文件名；通过历史项目代码文件的历史子文件之间的包含关系确定历史项目代码文件中的每个历史子文件的文件路径名以及文件名。例如，项目代码文件中pack文件夹下包含下kol_onlin文件夹，而kol_onlin文件夹下又包含index.html、main.js子文件，则根据该包含关系得出项目代码文件中，子文件的文件路径名为：/pack/kol_online/，文件名为index.html、main.js。

[0042] S30：对比每一相同路径名下，文件名相同的子文件和历史子文件的代码模块的差异，得到差异代码模块，代码模块是以为函数或者方法为单位的。

[0043] 其中，代码模块是以函数或者方法进行划分的，每一个函数或者方法作为一个代码模块。

[0044] 例如，在函数hello中，

[0045] function hello(){

[0046] var string＝“hello world!”；

[0047] alert string；}为一个代码模块。

[0048] 可以理解地，若子文件或历史子文件中全为函数，则代码模块是以函数来划分的；若子文件或历史子文件中全为方法，则代码模块是以函数来划分的；若子文件或历史子文件中由方法和函数构成，则每一函数为一个代码模块，每一方法为一个代码模块。

[0049] 具体地，先判断每一相同路径名下，是否存在文件名相同的子文件和历史子文件，再采用对比工具来对比子文件和历史子文件的代码模块的差异，将存在差异的代码模块作

为差异代码模块。示例性地，对比工具可以是Linux环境下的diff指令，该指令是采用字符串匹配的方式来逐行对比指定路径下的两个文件之间的差异，并输出差异的地方；还可以采用beyond compare软件来对比指定文件之间的差异。对比工具包括但不限于上述两种。

[0050] 例如，路径名为/pack/kol_online/中，文件名为index.html子文件中，名为OSInit的函数的第3-7行与历史子文件中，路径名为/pack/kol_online/中，文件名为index.html子文件中，名为OSInit的函数中的第3-7行有差异，其余地方的内容都相同，则将OSInit函数的代码模块作为差异代码模块。

[0051] S40：查找差异代码模块在语法树结构中的结构链路，得到目标链路，其中，语法树结构为预先采用语法树分析工具对项目代码文件进行语法树分析得到的。

[0052] 其中，语法树结构为根据项目代码文件所采用的编程语言来选择语法树分析工具所生成的。具体做法可以是将项目代码文件导入到语法是分析工具中，再采用语法树分析工具对项目代码文件进行语法树分析。如图3所示，在语法树结构中，每一个项目代表一颗树，将项目中的方法或者函数抽象成一个个节点，多个具有相互引用关系的节点组成了一条结构链路，结构链路反映了各个节点的代码之间的调用关系。

[0053] 具体地，以差异代码模块的名字为关键字来查找差异代码模块在语法树结构中的节点位置，并获取该节点所在的结构链路作为目标链路。

[0054] 可以理解地，目标链路的数量至少为1条，具体目标链路的数量还需要根据差异代码模块在项目代码文件之间的调用次数而定，调用次数越多，目标链路就越多。

[0055] S50：采用静态代码扫描工具对目标链路所对应的代码模块进行扫描，得到项目代码的扫描结果。

[0056] 其中，静态代码扫描工具是一种对代码进行静态代码检查的辅助工具，该工具能够检查出代码可能存在的缺陷。静态代码扫描工具种类很多，示例性地，主流的Java静态代码扫描工具有Checkstyle、FindBugs和Sonar等；主流的C/C++静态代码扫描工具有Cppcheck、FxCop Integrato和Microsoft Code Analysis等这里仅列举部分静态代码扫描工具，不同工具有不同的特点，具体选择哪种工具需要根据程序开发语言来选择和代码检查需求来定。

[0057] 可以理解地，扫描结果可能包含有项目代码的缺陷信息。若被扫描的代码模块存在漏洞、质量问题和/或语法规范问题，则扫描结果就会包含有项目代码的缺陷信息。

[0058] 其中，项目代码的缺陷信息是指项目代码中可能存在漏洞、项目代码存在的质量问题和/或项目代码的语法规范问题。

[0059] 项目代码中可能存在漏洞是指项目代码中可能存在的bug，例如因为一些空的try/catch/finally/switch语句可能导致的异常情况、因为定义的变量是数据类型太短而导致的数据的精确度失准或者数据溢出、参数不匹配、有歧义的嵌套语句、错误的递归、非法计算或者可能出现的空指针引用。

[0060] 项目代码存在的质量问题是指项目代码的重复率、复杂度或者覆盖率等。代码质量的高低能够影响代码的可读性和运行速率，重复率或者复杂度过高的代码会降低代码的可读性，严重情况下还会降低代码的运行速率。例如，代码模块中存在多句相同代码，并且没有将该多句相同的代码进行封装，则会降低代码的可读性。

[0061] 项目代码的语法规范问题是指项目代码是否符合语法规范，例如，在网页开发中，

常常有一些旧标签或者旧语句被废弃以及一些新标签或者新语句产生。虽然这些标签或者语句的废弃或产生不会影响程序的执行，但是通常会存在一些兼容性问题。

[0062] 具体地，需要先提取目标链路对应的代码模块，再采用静态代码扫描工具来对这些代码模块进行静态代码扫描，得出扫描结果。

[0063] 在该实施例中，先获取项目代码文件和项目代码文件对应的历史项目代码文件，再确定项目代码文件和历史项目代码文件中，每个子文件的路径名和文件名，然后对比项目代码文件的子文件和历史项目代码文件的历史子文件之间的代码模块的差异，能够增加项目代码扫描的针对性和目的性；通过先查找差异代码模块在语法树结构中的结构链路，得到目标链路，再采用静态代码扫描工具对目标链路所对应的代码模块进行扫描，有针对性的对部分代码进行扫描减少了代码扫描花费的时间，从而提高了项目代码检查的项目代码检查效率；采用静态代码扫描工具对目标链路对应的代码模块进行扫描，能够提高项目代码检查的准确性。

[0064] 在一实施例中，如图4所示，步骤S30中，即对比每一相同路径名下，文件名相同的子文件和历史子文件的代码模块的差异，得到差异代码模块，具体包括如下步骤：

[0065] S31：判断每一个相同文件路径名下，是否存在文件名相同的子文件和历史子文件。

[0066] 具体地，判断每一个相同文件路径名下，是否存在文件名相同的子文件和历史子文件，若只存在子文件，则证明该子文件相对于历史子文件来说是新增的；若只存在历史子文件，则证明该历史子文件对应的子文件被删除了。

[0067] S32：若相同文件路径名下，存在文件名相同的子文件和历史子文件，则对比子文件与历史子文件的代码模块之间的差异，将每一存在差异的代码模块作为差异代码模块。

[0068] 具体地，若相同文件路径名下存在文件名相同的子文件和历史子文件，则采用对比工具来对比相同文件路径名下存在文件名相同的子文件和历史子文件的代码模块之间的差异，将每一存在差异的代码模块作为差异代码模块。

[0069] 在本实施例中，通过判断每一个相同文件路径名下，是否存在文件名相同的子文件和历史子文件，若相同文件路径名下，存在文件名相同的子文件和历史子文件，则对比子文件与历史子文件的代码模块之间的差异，将每一存在差异的代码模块作为差异代码模块，能够根据判断结果生成差异代码模块，并且根据不同的情况具体分析，提高项目代码扫描的准确性。

[0070] 在一实施例中，如图5所示，步骤S31中，即判断每一个相同文件路径名下，是否存在文件名相同的子文件和历史子文件之后，还包括如下步骤：

[0071] S33：若相同文件路径名下只存在子文件，则将子文件中的每一代码模块作为差异代码模块。

[0072] 具体地，若相同文件路径名下只存在项目代码文件的子文件，则证明该子文件相对与历史子文件来说是新增的，因此，将该子文件中的每一代码模块作为一个差异代码模块。

[0073] S34：若相同文件路径名下只存在历史子文件，则将历史子文件中的每一代码模块作为差异代码模块。

[0074] 具体地，若相同文件路径名下只存在历史项目代码文件的历史子文件，则证明该

历史子文件对应的项目代码文件的子文件被删除了，因此，将该历史子文件中的每一代码模块作为一个差异代码模块。

[0075] 在该实施例中，若相同文件路径名下只存在子文件，则将子文件中的每一代码模块作为差异代码模块；若相同文件路径名下只存在历史子文件，则将历史子文件中的每一代码模块作为差异代码模块；根据判断结果对不同情况进行处理，能够提高后续项目代码扫描的准确性。

[0076] 在一实施例中，如图6所示，步骤S40中，即查找差异代码模块在语法树结构中的结构链路，得到目标链路，包括以下步骤：

[0077] S41：以差异代码模块的名字为关键字，查找差异代码模块在语法树结构的结构链路中所处的节点，其中差异代码模块的名字为差异代码模块所对应的函数名或者方法名；

[0078] 其中，差异代码模块的名字是指差异代码模块所对应的函数名或者方法名。例如，差异代码模块所对应的函数名为OSIint，则差异代码模块的名字也为OSIint。

[0079] 具体地，以差异代码模块的名字为关键字，查找差异代码模块在语法树结构的结构链路中所处的节点。

[0080] S42：将查找出来的节点所在的每一条结构链路作为目标链路。

[0081] 具体地，将查找出来的节点所在的每一条结构链路作为目标链路。例如，查找出来的节点为OSTaskCreate，该节点所在的结构链路为：main—OSTaskCreate—OS_Sched—OS_TASK_SW和main—OSTaskCreate—OSTaskStkInit，则将这两个结构链路作为目标链路。

[0082] 在该实施例中，通过以差异代码模块的名字为关键字，查找差异代码模块在语法树结构的结构链路中所处的节点，再将查找出来的节点所在的每一条结构链路作为目标链路，能够提高代码扫描的目的性，进一步地，能够快速地获取到待扫描的项目代码，提高后续的代码扫描的速率。

[0083] 在一实施例中，如图7所示，步骤S50中，即采用静态代码扫描工具对目标链路所对应的项目代码进行扫描，得到项目代码的扫描结果，包括如下步骤：

[0084] S51：提取目标链路上的所有节点对应的项目代码模块。

[0085] 具体地，从项目代码文件中将目标链路上的所有节点对应的项目代码模块提取出来。例如，目标链路为main—OSTaskCreate—OSTaskStkInit，则提取main、OSTaskCreate和OSTaskStkInit这三个节点所对应的代码模块。

[0086] S52：采用静态代码扫描工具对提取出来的代码模块进行静态代码扫描，得到扫描结果。

[0087] 其中，静态代码扫描是指在不运行代码的方式下，采用特定的静态代码扫描工具进行词法分析、语法分析、控制流分析等技术对程序代码进行扫描，验证代码是否满足规范性、安全性、可靠性、可维护性等指标的一种代码分析技术。静态代码检查能够发现编译器不能发现的问题，也可以自定义一些检查规范，来检查代码编写中不规范、存在安全隐患的地方。

[0088] 具体地，先将提取出来的代码模块(即目标链路上所有节点的代码模块)导入入进静态代码扫描工具中，再进行扫描。

[0089] 可选地，在进行静态代码扫描前，可以自定义一些检查规则来约束扫描条件和输出规则，再将该规则和提取出来的代码模块输入进静态代码扫描工具中，再开启静态代码

扫描工具，对代码模块进行扫描。经过一段时间的扫描后，得出一个项目代码的扫描结果。其中，只要提取出来的代码模块存在与自定义的检查规则相符的地方，则根据检查规则中的输出规则来输出该部分的信息。例如，自定义的检查规则为：函数命名不能出现全部大写字母的情况，若检查出目标链路上的节点的名字符合该检查规则的情况，则输出规则为：函数(全大写字母的函数名)命名为全大写字母。

[0090] 在该实施例中，先提取目标链路上的所有节点对应的项目代码模块，然后采用静态代码扫描工具对提取出来的代码模块进行静态代码扫描，得到项目代码的扫描结果，对项目代码进行针对性扫描，能够提高项目代码扫描的目的性。

[0091] 应理解，上述实施例中各步骤的序号的大小并不意味着执行顺序的先后，各过程的执行顺序应以其功能和内在逻辑确定，而不应对本申请实施例的实施过程构成任何限定。

[0092] 在一实施例中，提供一种项目代码扫描装置，该项目代码扫描装置与上述实施例中项目代码扫描方法一一对应。如图8所示，该项目代码扫描装置包括获取模块10、确定模块20、对比模块30、查找模块40和扫描模块50。

[0093] 各功能模块详细说明如下：

[0094] 获取模块10，用于获取项目代码文件和项目代码文件对应的历史项目代码文件，项目代码文件包括多个子文件，项目代码文件对应的历史项目代码文件包括多个历史子文件。

[0095] 确定模块20，用于确定项目代码文件和历史项目代码文件中，每个子文件的路径名和文件名。

[0096] 对比模块30，用于对比每一相同路径名下，文件名相同的子文件和历史子文件的代码模块的差异，得到差异代码模块，代码模块是以为函数或者方法为单位的。

[0097] 查找模块40，用于查找差异代码模块在语法树结构中的结构链路，得到目标链路，其中，语法树结构为预先采用语法树分析工具对项目代码文件进行语法树分析得到的。

[0098] 扫描模块50，用于采用静态代码扫描工具对目标链路所对应的代码模块进行扫描，得到项目代码的扫描结果。

[0099] 优选地，在一实施例中，如图9所示，对比模块30包括判断单元31和对比单元32，各单元详细说明如下：

[0100] 判断单元31，用于判断每一个相同文件路径名下，是否存在文件名相同的子文件和历史子文件。

[0101] 对比单元32，用于若相同文件路径名下，存在文件名相同的子文件和历史子文件，则对比子文件与历史子文件的代码模块之间的差异，将每一存在差异的代码模块作为差异代码模块。

[0102] 优选地，在一实施例中，如图10所示，查找模块40包括查找单元41和确定单元42，各单元详细说明如下：

[0103] 查找单元41，用于以差异代码模块的名字为关键字，查找差异代码模块在语法树结构的结构链路中所处的节点，其中差异代码模块的名字为差异代码模块所对应的函数名或者方法名。

[0104] 确定单元42，用于将查找出来的节点所在的每一条结构链路作为目标链路。

[0105] 可选地，在一实施例中，在判断单元31之后还包括第一差异子单元和第二差异子单元，各单元详细说明如下：

[0106] 第一差异子单元，用于若相同文件路径名下只存在子文件，则将子文件中的每一代码模块作为差异代码模块。

[0107] 第二差异子单元，用于若相同文件路径名下只存在历史子文件，则将历史子文件中的每一代码模块作为差异代码模块。

[0108] 可选地，在一实施例中，扫描模块50包括提取单元和扫描单元，各单元详细说明如下：

[0109] 提取单元，用于提取目标链路上的所有节点对应的项目代码模块。

[0110] 扫描单元，用于采用静态代码扫描工具对提取出来的代码模块进行静态代码扫描，得到项目代码的扫描结果。

[0111] 关于项目代码扫描装置的具体限定可以参见上文中对于项目代码扫描方法的限定，在此不再赘述。上述项目代码扫描装置中的各个模块可全部或部分通过软件、硬件及其组合来实现。上述各模块可以硬件形式内嵌于或独立于计算机设备中的处理器中，也可以以软件形式存储于计算机设备中的存储器中，以便于处理器调用执行以上各个模块对应的操作。

[0112] 在一个实施例中，提供了一种计算机设备，该计算机设备可以是服务器，其内部结构图可以如图11所示。该计算机设备包括通过系统总线连接的处理器、存储器、网络接口和数据库。其中，该计算机设备的处理器用于提供计算和控制能力。该计算机设备的存储器包括非易失性存储介质、内存储器。该非易失性存储介质存储有操作系统、计算机程序和数据库。该内存储器为非易失性存储介质中的操作系统和计算机程序的运行提供环境。该计算机设备的数据库用于存储项目代码扫描方法所需要的数据。该计算机设备的网络接口用于与外部的终端通过网络连接通信。该计算机程序被处理器执行时以实现一种项目代码扫描方法。

[0113] 在一个实施例中，提供了一种计算机设备，包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序，处理器执行计算机程序时实现以下步骤：

[0114] 获取项目代码文件和项目代码文件对应的历史项目代码文件，项目代码文件包括多个子文件，项目代码文件对应的历史项目代码文件包括多个历史子文件；

[0115] 确定项目代码文件和历史项目代码文件中，每个子文件的路径名和文件名；

[0116] 对比每一相同路径名下，文件名相同的子文件和历史子文件的代码模块的差异，得到差异代码模块，代码模块是以为函数或者方法为单位的；

[0117] 查找差异代码模块在语法树结构中的结构链路，得到目标链路，其中，语法树结构为预先采用语法树分析工具对项目代码文件进行语法树分析得到的；

[0118] 采用静态代码扫描工具对目标链路所对应的代码模块进行扫描，得到项目代码的扫描结果。

[0119] 在一个实施例中，提供了一种计算机可读存储介质，其上存储有计算机程序，计算机程序被处理器执行时实现以下步骤：

[0120] 获取项目代码文件和项目代码文件对应的历史项目代码文件，项目代码文件包括多个子文件，项目代码文件对应的历史项目代码文件包括多个历史子文件；

[0121] 确定项目代码文件和历史项目代码文件中，每个子文件的路径名和文件名；

[0122] 对比每一相同路径名下，文件名相同的子文件和历史子文件的代码模块的差异，得到差异代码模块，代码模块是以为函数或者方法为单位的；

[0123] 查找差异代码模块在语法树结构中的结构链路，得到目标链路，其中，语法树结构为预先采用语法树分析工具对项目代码文件进行语法树分析得到的；

[0124] 采用静态代码扫描工具对目标链路所对应的代码模块进行扫描，得到项目代码的扫描结果。

[0125] 本领域普通技术人员可以理解实现上述实施例方法中的全部或部分流程，是可以通过计算机程序来指令相关的硬件来完成，所述的计算机程序可存储于一非易失性计算机可读取存储介质中，该计算机程序在执行时，可包括如上述各方法的实施例的流程。其中，本申请所提供的各实施例中所使用的对存储器、存储、数据库或其它介质的任何引用，均可包括非易失性和/或易失性存储器。非易失性存储器可包括只读存储器(ROM)、可编程ROM(PROM)、电可编程ROM(EPROM)、电可擦除可编程ROM(EEPROM)或闪存。易失性存储器可包括随机存取存储器(RAM)或者外部高速缓冲存储器。作为说明而非局限，RAM以多种形式可得，诸如静态RAM(SRAM)、动态RAM(DRAM)、同步DRAM(SDRAM)、双数据率SDRAM(DDRSDRAM)、增强型SDRAM(ESDRAM)、同步链路(Synchlink)DRAM(SLDRAM)、存储器总线(Rambus)直接RAM(RDRAM)、直接存储器总线动态RAM(DRDRAM)、以及存储器总线动态RAM(RDRAM)等。

[0126] 所属领域的技术人员可以清楚地了解到，为了描述的方便和简洁，仅以上述各功能单元、模块的划分进行举例说明，实际应用中，可以根据需要而将上述功能分配由不同的功能单元、模块完成，即将所述装置的内部结构划分成不同的功能单元或模块，以完成以上描述的全部或者部分功能。

[0127] 以上所述实施例仅用以说明本申请的技术方案，而非对其限制；尽管参照前述实施例对本申请进行了详细的说明，本领域的普通技术人员应当理解：其依然可以对前述各实施例所记载的技术方案进行修改，或者对其中部分技术特征进行等同替换；而这些修改或者替换，并不使相应技术方案的本质脱离本申请各实施例技术方案的精神和范围，均应包含在本申请的保护范围之内。

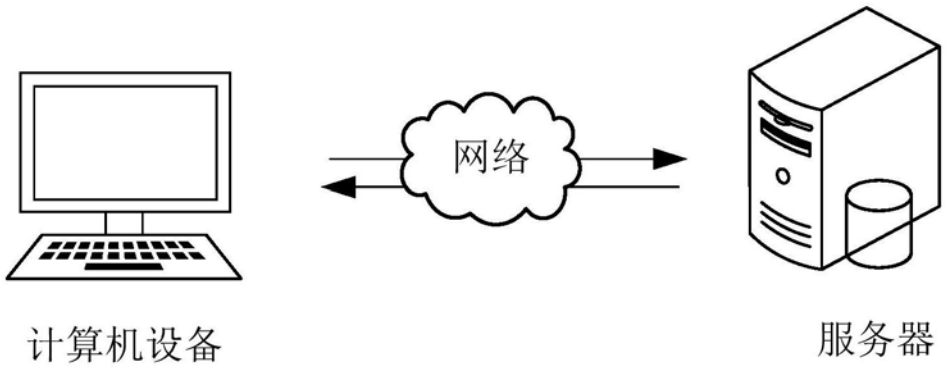

网络
计算机设备
服务器

图1

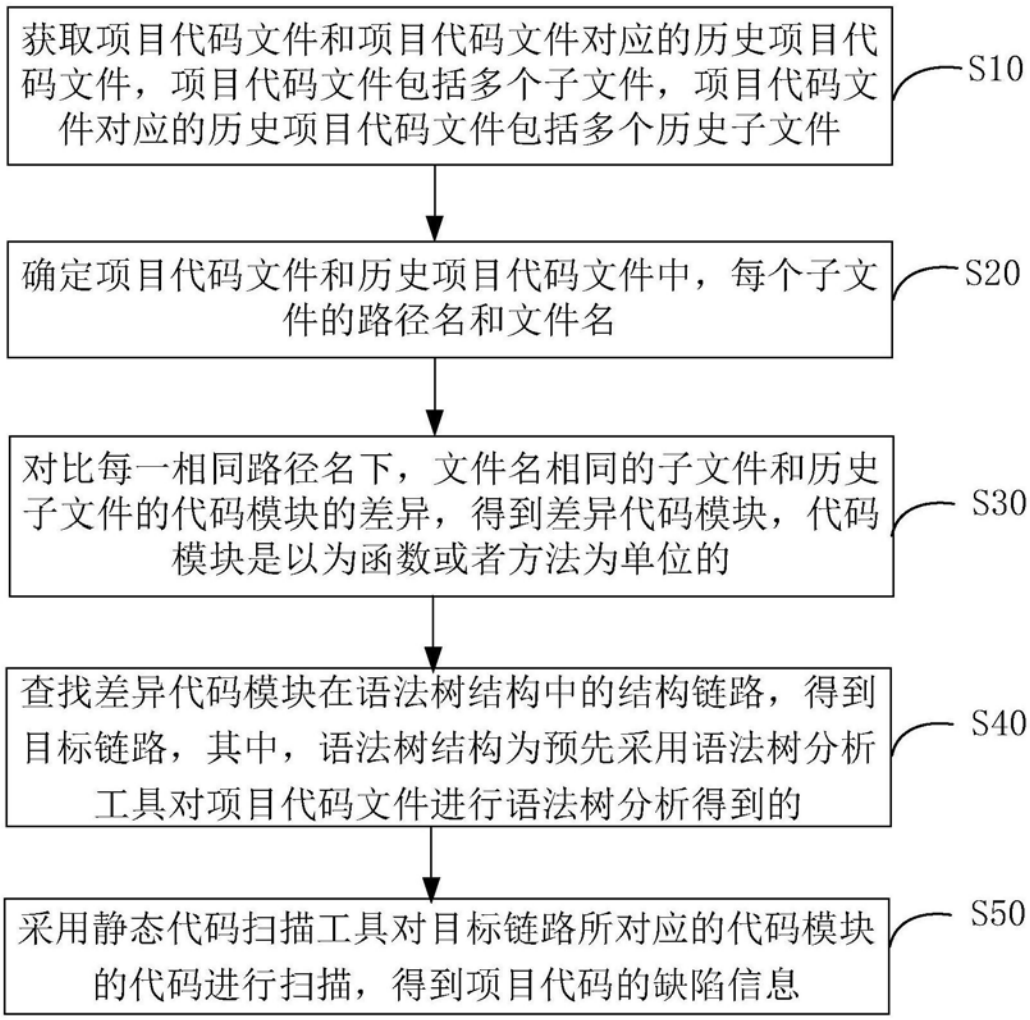

获取项目代码文件和项目代码文件对应的历史项目代码文件，项目代码文件包括多个子文件，项目代码文件对应的历史项目代码文件包括多个历史子文件
S10
确定项目代码文件和历史项目代码文件中，每个子文件的路径名和文件名
S20
对比每一相同路径名下，文件名相同的子文件和历史子文件的代码模块的差异，得到差异代码模块，代码模块是以为函数或者方法为单位的
S30
查找差异代码模块在语法树结构中的结构链路，得到目标链路，其中，语法树结构为预先采用语法树分析工具对项目代码文件进行语法树分析得到的
S40
采用静态代码扫描工具对目标链路所对应的代码模块的代码进行扫描，得到项目代码的缺陷信息
S50

图2

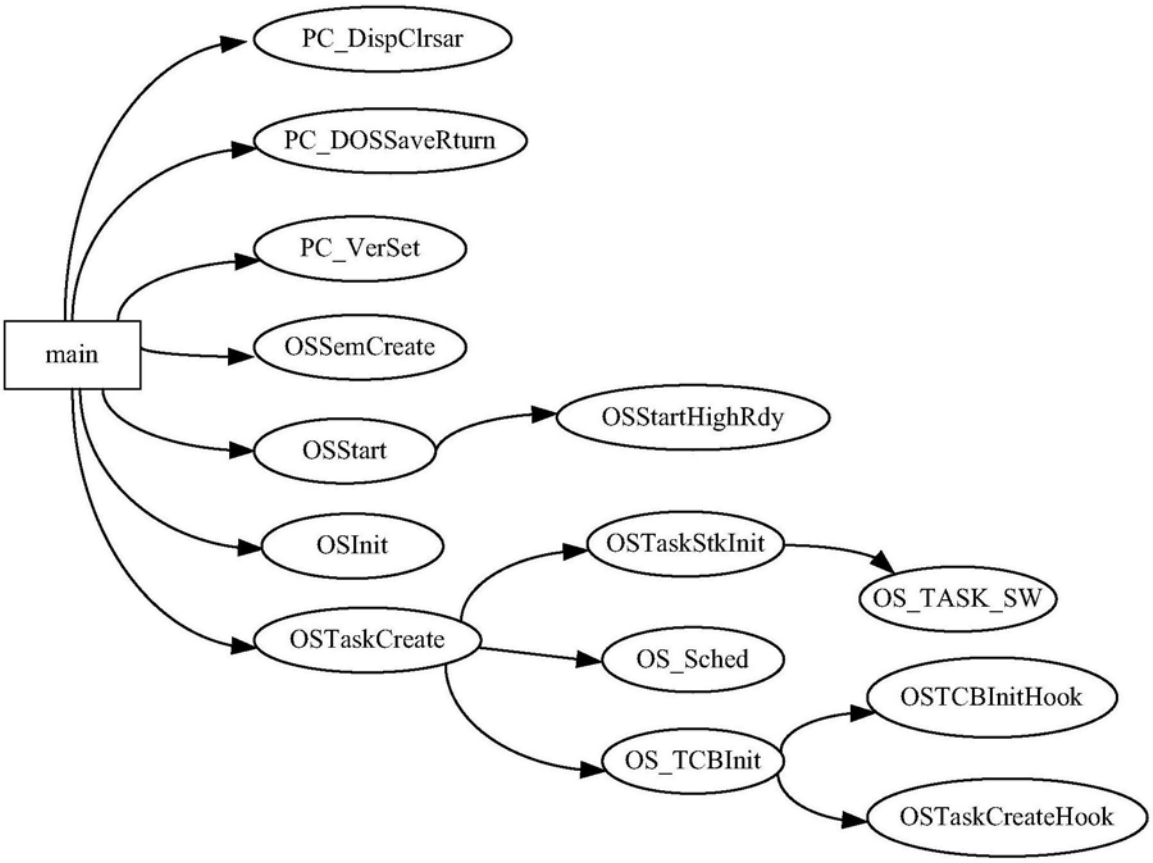

PC_DispClrsar
PC_DOSSaveRturn
PC_VerSet
main
OSSemCreate
OSStart
OSStartHighRdy
OSInit
OSTaskStkInit
OS_TASK_SW
OSTaskCreate
OS_Sched
OSTCBInitHook
OS_TCBInit
OSTaskCreateHook

图3

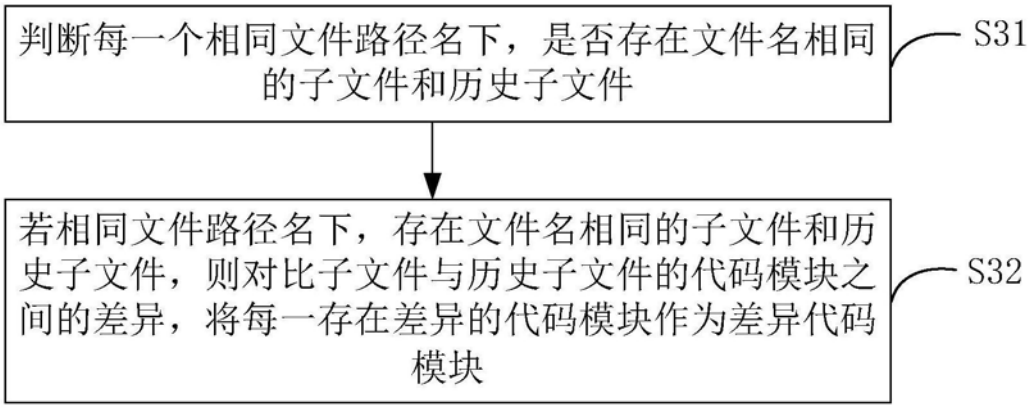

判断每一个相同文件路径名下，是否存在文件名相同的子文件和历史子文件
S31
若相同文件路径名下，存在文件名相同的子文件和历史子文件，则对比子文件与历史子文件的代码模块之间的差异，将每一存在差异的代码模块作为差异代码模块
S32

图4

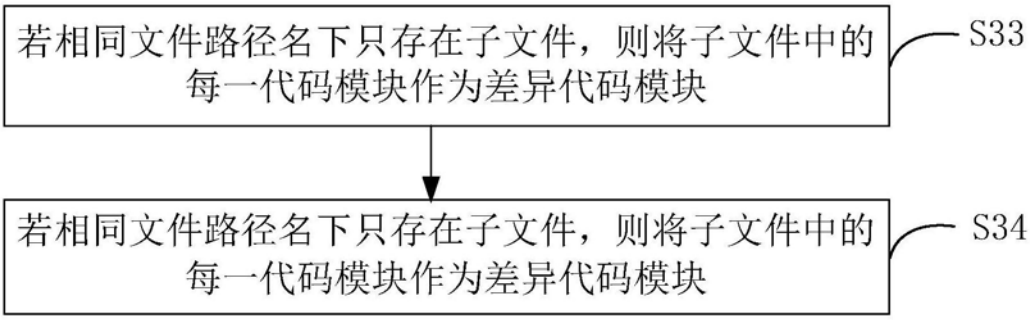

若相同文件路径名下只存在子文件，则将子文件中的每一代码模块作为差异代码模块
S33
若相同文件路径名下只存在子文件，则将子文件中的每一代码模块作为差异代码模块
S34

图5

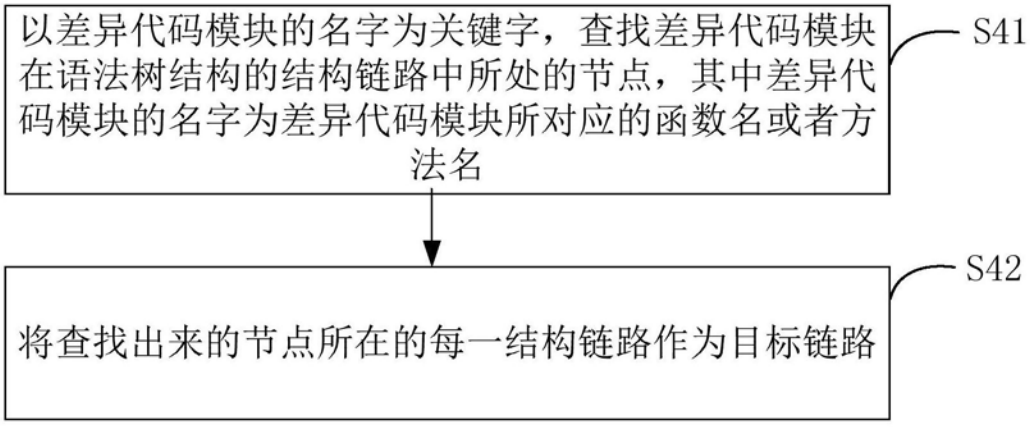

以差异代码模块的名字为关键字，查找差异代码模块在语法树结构的结构链路中所处的节点，其中差异代码模块的名字为差异代码模块所对应的函数名或者方法名
S41
将查找出来的节点所在的每一结构链路作为目标链路
S42

图6

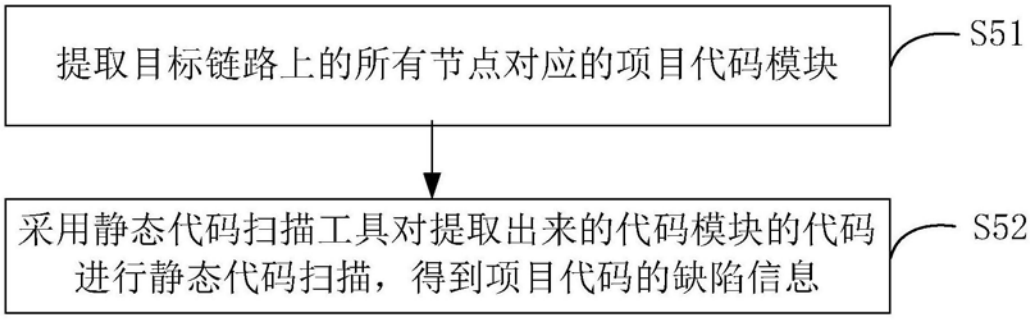

提取目标链路上的所有节点对应的项目代码模块
S51
采用静态代码扫描工具对提取出来的代码模块的代码进行静态代码扫描，得到项目代码的缺陷信息
S52

图7

图8

图9


查找单元
41
确定单元
42

图10

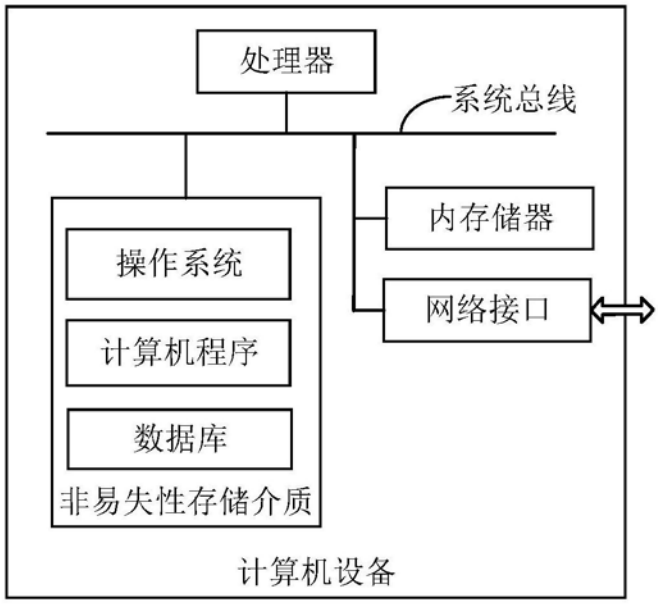

处理器
系统总线
内存储器
操作系统
网络接口
计算机程序
数据库
非易失性存储介质
计算机设备

图11