

(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国际局

(43) 国际公布日
2019 年 4 月 18 日 (18.04.2019)



(10) 国际公布号
WO 2019/071891 A1

- (51) 国际专利分类号:
G06F 11/36 (2006.01)
- (21) 国际申请号: PCT/CN2018/076141
- (22) 国际申请日: 2018 年 2 月 10 日 (10.02.2018)
- (25) 申请语言: 中文
- (26) 公布语言: 中文
- (30) 优先权:
201710933579.0 2017 年 10 月 10 日 (10.10.2017) CN
- (71) 申请人: 平安科技 (深圳) 有限公司 (PING AN TECHNOLOGY (SHENZHEN) CO., LTD.) [CN/CN];
中国广东省深圳市福田区八卦岭工业区平安大厦六楼, Guangdong 518000 (CN)。
- (72) 发明人: 熊星 (XIONG, Xing); 中国广东省深圳市福田区八卦岭工业区平安大厦六楼, Guangdong 518000 (CN)。
- (74) 代理人: 深圳市沃德知识产权代理事务所 (普通合伙) (SHENZHEN WORLD INTELLECTUAL PROPERTY AGENCY (GENERAL

PARTNERSHIP)); 中国广东省深圳市福田区园岭街道八卦四路 10 号中浩大厦 1528-1530 室于志光, Guangdong 518000 (CN)。

(81) 指定国 (除另有指明, 要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW。

(84) 指定国 (除另有指明, 要求每一种可提供的地区保护): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT,

(54) Title: CODE COVERAGE ANALYSIS METHOD AND APPLICATION SERVER

(54) 发明名称: 代码覆盖率分析方法及应用服务器

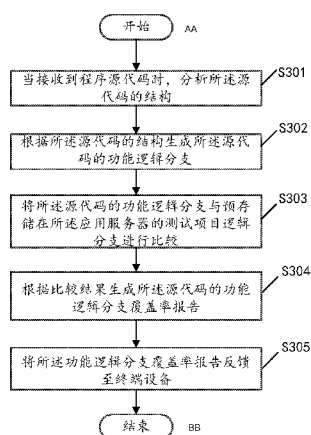


图 6

S301 ANALYZE A STRUCTURE OF A PROGRAM SOURCE CODE WHEN RECEIVING SAID SOURCE CODE
S302 GENERATE FUNCTION LOGIC BRANCHES OF THE SOURCE CODE ACCORDING TO THE STRUCTURE OF THE SOURCE CODE
S303 COMPARE THE FUNCTION LOGIC BRANCHES OF THE SOURCE CODE TO TEST PROJECT LOGIC BRANCHES THAT ARE PRE-STORED IN THE APPLICATION SERVER
S304 GENERATE A FUNCTION LOGIC BRANCH COVERAGE REPORT OF THE SOURCE CODE ACCORDING TO A COMPARISON RESULT
S305 FEED THE FUNCTION LOGIC BRANCH COVERAGE REPORT BACK TO THE TERMINAL DEVICE
AA START
BB END

(57) Abstract: A code coverage analysis method, an application server and a computer-readable storage medium, the method comprising: analyzing the structure of a program source code when receiving said source code which is sent by a terminal device (S301); generating function logic branches of the source code according to the structure of the source code (S302); comparing the function logic branches of the source code to test project logic branches that are pre-stored in the application server (S303); generating a function logic branch coverage report of the source code according to a comparison result (S304); and feeding the function logic branch coverage report back to the terminal device (S305). With the described code coverage analysis method, whether a source code already covers all function logic branches may be determined, so that the function integrity of a test project is guaranteed.

(57) 摘要: 一种代码覆盖率分析方法、应用服务器以及计算机可读存储介质, 所述方法包括: 当接收到所述终端设备发送的程序源代码时, 分析所述源代码的结构 (S301); 根据所述源代码的结构生成所述源代码的功能逻辑分支 (S302); 将所述源代码的功能逻辑分支与预存储在所述应用服务器的测试项目逻辑分支进行比较 (S303); 根据比较结果生成所述源代码的功能逻辑分支覆盖率报告 (S304); 将所述功能逻辑分支覆盖率报告反馈至终端设备 (S305)。所述代码覆盖率分析方法可以确认源代码是否已经覆盖了所有的功能逻辑分支, 保证测试项目的功能完整性。

RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI,
CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布:

- 包括国际检索报告(条约第21条(3))。

代码覆盖率分析方法及应用服务器

本申请要求于2017年10月10日提交中国专利局、申请号为201710933579.0、发明名称为“代码覆盖率分析方法及应用服务器”的中国专利申请的优先权，其全部内容通过引用结合在申请中。

技术领域

本申请涉及代码测试技术领域，尤其涉及一种代码覆盖率分析方法及应用服务器。

背景技术

代码覆盖率(codecoverage)是反映测试用例对被测程序覆盖程度的重要指标，也是衡量测试完整性的一个参考值。通过代码覆盖率数据可以评估测试是否充分。在软件开发的过程中，通过客户端获取代码覆盖率数据是基于被监控项目的全部代码生成的。传统的测试覆盖率分析都是针对代码行的覆盖情况，通过代码覆盖率统计，我们能了解到程序中有多少的代码被执行了，但是并不能真正了解一个程序模块的完整覆盖情况。

发明内容

有鉴于此，本申请提出一种代码覆盖率分析方法及应用服务器，能够确认源代码是否已经覆盖了所有的功能逻辑分支，保证测试项目的功能完整性。

首先，为实现上述目的，本申请提出一种代码覆盖率分析方法，该方法应用于应用服务器，所述方法包括：

当接收到终端设备发送的程序源代码时，分析所述源代码的结构；

根据所述源代码的结构生成所述源代码的功能逻辑分支；

将所述源代码的功能逻辑分支与预存储在所述应用服务器的测试项目逻

辑分支进行比较；

根据比较结果生成所述源代码的功能逻辑分支覆盖率报告；

将所述功能逻辑分支覆盖率报告反馈至所述终端设备。

此外，为实现上述目的，本申请还提供一种应用服务器，所述应用服务
5 器包括存储器、处理器，所述存储器上存储有可在所述处理器上运行的代码
覆盖率分析系统，所述代码覆盖率分析系统被所述处理器执行时实现如下步
骤：

当接收到终端设备发送的程序源代码时，分析所述源代码的结构；

根据所述源代码的结构生成所述源代码的功能逻辑分支；

10 将所述源代码的功能逻辑分支与预存储在所述应用服务器的测试项目逻辑
分支进行比较；

根据比较结果生成所述源代码的功能逻辑分支覆盖率报告；

将所述功能逻辑分支覆盖率报告反馈至所述终端设备。

进一步地，为实现上述目的，本申请还提供一种计算机可读存储介质，
15 所述计算机可读存储介质存储有代码覆盖率分析系统，所述代码覆盖率分析
系统可被至少一个处理器执行，以使所述至少一个处理器执行如上述的代码
覆盖率分析方法的步骤。

相较于现有技术，本申请所提出的应用服务器、代码覆盖率分析方法及
计算机可读存储介质，首先当接收到所述终端设备发送的程序源代码时，分
20 析所述源代码的结构；然后，根据所述源代码的结构生成所述源代码的功能
逻辑分支；进一步地，将所述源代码的功能逻辑分支与预存储在所述应用服
务器的测试项目逻辑分支进行比较；接着，根据比较结果生成所述源代码的
功能逻辑分支覆盖率报告；最后将所述功能逻辑分支覆盖率报告反馈至终端
设备。这样，既可以避免现有技术中通过代码覆盖率统计，仅能了解到程序
25 中有多少的代码被执行了，但是并不能真正了解一个程序模块的完整覆盖情
况的缺陷，也可以使得通过逻辑分支的比较，确认源代码是否已经覆盖了所

有的功能逻辑分支，保证测试项目的功能完整性。

附图说明

图1是本申请各个实施例一可选的应用环境示意图；

5 图2是图1中应用服务器一可选的硬件架构的示意图；

图3是本申请代码覆盖率分析系统第一实施例的程序模块示意图；

图4是本申请代码覆盖率分析系统第二实施例的程序模块示意图；

图5是本申请代码覆盖率分析系统第三实施例的程序模块示意图；

图6为本申请代码覆盖率分析方法第一实施例的流程示意图；

10 图7为本申请代码覆盖率分析方法第二实施例的流程示意图；

图8为本申请代码覆盖率分析方法第三实施例的流程示意图。

本申请目的的实现、功能特点及优点将结合实施例，参照附图做进一步说明。

15 具体实施方式

为了使本申请的目的、技术方案及优点更加清楚明白，以下结合附图及实施例，对本申请进行进一步详细说明。应当理解，此处所描述的具体实施例仅用以解释本申请，并不用于限定本申请。基于本申请中的实施例，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
20 属于本申请保护的范围。

需要说明的是，在本申请中涉及“第一”、“第二”等的描述仅用于描述目的，而不能理解为指示或暗示其相对重要性或者隐含指明所指示的技术特征的数量。由此，限定有“第一”、“第二”的特征可以明示或者隐含地包括至少一个该特征。另外，各个实施例之间的技术方案可以相互结合，但是必须是以本领域普通技术人员能够实现为基础，当技术方案的结合出现相互矛盾或无法实现
25 现时应当认为这种技术方案的结合不存在，也不在本申请要求的保护范围之内。

内。

参阅图 1 所示，是本申请各个实施例一可选的应用环境示意图。

在本实施例中，本申请可应用于包括，但不限于，终端设备 1、应用服务器 2、网络 3 的应用环境中。其中，所述终端设备 1 可以是移动电话、智能
5 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)、导航装置等等的可移动设备，以及诸如数字 TV、台式计算机、笔记本、服务器等等的固定终端。所述应用服务器 2 可以是机架式服务器、刀片式服务器、塔式服务器或机柜式服务器等计算设备，该应用服务器 2 可以是独立的服务器，也可以是多个服务器所组成的服务器集群。
10 所述网络 3 可以是企业内部网 (Intranet)、互联网 (Internet)、全球移动通讯系统(Global System of Mobile communication, GSM)、宽带码分多址(Wideband Code Division Multiple Access, WCDMA)、4G 网络、5G 网络、蓝牙(Bluetooth)、Wi-Fi 等无线或有线网络。

其中，所述应用服务器 2 中通过所述网络 3 分别与一个或多个所述终端设备 1 通信连接，以进行数据传输和交互。
15

本实施例中，所述应用服务器 2 内安装并运行有代码覆盖率分析系统 200，当所述代码覆盖率分析系统 200 运行时，所述应用服务器 2 接收到所述终端设备 1 发送的程序源代码时，分析所述源代码的结构；然后，根据所述源代码的结构生成所述源代码的功能逻辑分支；进一步地，将所述源代码的功能
20 逻辑分支与预存储在所述应用服务器的测试项目逻辑分支进行比较；接着，根据比较结果生成所述源代码的功能逻辑分支覆盖率报告；最后将所述功能逻辑分支覆盖率报告反馈至终端设备 1。这样，既可以避免现有技术中通过代码覆盖率统计，仅能了解到程序中有多少的代码被执行了，但是并不能真正了解一个程序模块的完整覆盖情况的缺陷，也可以使得通过逻辑分支的比较，
25 确认源代码是否已经覆盖了所有的功能逻辑分支，保证测试项目的功能完整性。

参阅图 2 所示，是图 1 中应用服务器 2 一可选的硬件架构的示意图。本实施例中，所述应用服务器 2 可包括，但不仅限于，可通过系统总线相互通信连接存储器 11、处理器 12、网络接口 13。需要指出的是，图 2 仅示出了具有组件 11-13 的终端设备 1，但是应理解的是，并不要求实施所有示出的组件，
5 可以替代的实施更多或者更少的组件。

其中，所述存储器 11 至少包括一种类型的可读存储介质，所述可读存储介质包括闪存、硬盘、多媒体卡、卡型存储器（例如，SD 或 DX 存储器等）、随机访问存储器（RAM）、静态随机访问存储器（SRAM）、只读存储器（ROM）、电可擦除可编程只读存储器（EEPROM）、可编程只读存储器（PROM）、磁性
10 存储器、磁盘、光盘等。在一些实施例中，所述存储器 11 可以是所述应用服务器 2 的内部存储单元，例如该应用服务器 2 的硬盘或内存。在另一些实施例中，所述存储器 11 也可以是所述应用服务器 2 的外部存储设备，例如该应用服务器 2 上配备的插接式硬盘，智能存储卡（Smart Media Card, SMC），安全数字（Secure Digital, SD）卡，闪存卡（Flash Card）等。当然，所述存储器
15 11 还可以既包括所述应用服务器 2 的内部存储单元也包括其外部存储设备。本实施例中，所述存储器 11 通常用于存储安装于所述应用服务器 2 的操作系统和各类应用软件，例如所述代码覆盖率分析系统 200 的程序代码等。此外，所述存储器 11 还可以用于暂时地存储已经输出或者将要输出的各类数据。

所述处理器 12 在一些实施例中可以是中央处理器（Central Processing
20 Unit, CPU）、控制器、微控制器、微处理器、或其他数据处理芯片。该处理器 12 通常用于控制所述应用服务器 2 的总体操作，例如执行与所述终端设备 1 进行数据交互或者通信相关的控制和处理等。本实施例中，所述处理器 12 用于运行所述存储器 11 中存储的程序代码或者处理数据，例如运行所述的代码覆盖率分析系统 200 等。

所述网络接口 13 可包括无线网络接口或有线网络接口，该网络接口 13
25 通常用于在所述应用服务器 2 与其他电子设备之间建立通信连接。本实施例

中，所述网络接口 13 主要用于通过所述网络 3 将所述应用服务器 2 与一个或多个所述终端设备 1 相连，在所述应用服务器 2 与所述一个或多个终端设备 1 之间的建立数据传输通道和通信连接。

至此，已经详细介绍了本申请各个实施例的应用环境和相关设备的硬件
5 结构和功能。下面，将基于上述应用环境和相关设备，提出本申请的各个实施例。

首先，本申请提出一种代码覆盖率分析系统 200。

参阅图 3 所示，是本申请代码覆盖率分析系统 200 第一实施例的程序模块图。

10 本实施例中，所述代码覆盖率分析系统 200 包括一系列的存储于存储器 11 上的计算机程序指令，当该计算机程序指令被处理器 12 执行时，可以实现本申请各实施例的代码覆盖率分析操作。在一些实施例中，基于该计算机程序指令各部分所实现的特定的操作，代码覆盖率分析系统 200 可以被划分为一个或多个模块。例如，在图 3 中，所述代码覆盖率分析系统 200 可以被分
15 割成分析模块 201、生成模块 202、比较模块 203、以及反馈模块 204。其中：

所述分析模块 201，用于当接收到所述终端设备 1 发送的程序源代码时，分析所述源代码的结构。

通常地，传统的测试程序代码的覆盖率分析都是针对代码行的覆盖情况，但是这种通过分析代码行的覆盖情况，能够了解程序中有多少代码被执行了，
20 并不能正代表一个功能单元的完整覆盖情况。假设一个功能单元根据不同条件需要调用 4 个函数组合来完成这个功能。若该功能会形成 16 个逻辑分支，而按代码行覆盖率统计，源代码只需要写 4 个逻辑分支就能把 4 个函数都覆盖掉，但从功能完整性上看，所述功能单元是不完整的。

举例而言，所述源代码的功能是完成用户名注册，完成注册的用户可以
25 有不同类型的登陆方式，每一种登陆方式为一个功能逻辑分支。例如，通过调用函数 A，实现用户可以通过手机号码发送认证码登陆，调用函数 B，实现

用户可以通过邮箱激活链接登陆，还可以同时调用函数 A 及函数 B，实现通过邮箱激活链接的同时通过手机号码发送认证码登陆，这种情况下，如果按照传统的测试程序代码的覆盖分析，则只能分析出程序中的函数 A 和函数 B 被调用，而不能分析出同时调用函数 A 和函数 B 这一逻辑分支，从而导致测试案例出现功能覆盖不完整的情况。

因此，在本实施例中，所述应用服务器 2 与所述终端设备 1 建立长连接，当所述应用服务器 2 在接收到终端设备 1 发送的程序源代码时，通过分析模块 201 分析所述源代码的结构，具体地，分析函数及方法的调用关系，结构的包含关系，类的继承关系等等。

所述生成模块 202，用于根据所述源代码的结构生成所述源代码的功能逻辑分支。

具体地，根据所述源代码的结构，分析每一个功能单元调用到的功能函数，以及每一个功能函数的调用关系，进而生成源代码关于实现每一个功能的功能逻辑分支。

在本实施例中，所述应用服务器 2 分析所述源代码的结构，形成所述源代码的语法树。根据所述语法树查找功能函数及与功能函数的调用关系，同时去除注释以及不需要统计覆盖率信息的公共函数等，查找功能函数，分析功能函数的调用关系。根据所述功能函数的调用关系，生成所述源代码的逻辑分支。

所述比较模块 203，用于将所述源代码的功能逻辑分支与预存储在所述应用服务器的测试项目逻辑分支进行比较。

通常地，开发人员进行源代码设计时，必然了解待开发软件的功能。在本实施例中，为了验证所述源代码完整性，开发人员预先把源代码需要的逻辑分支存在所述应用服务器的存储空间。在本实施例中，当生成了所述源代码的逻辑分支时，所述应用服务器 2 调预先存储有的待开发软件的功能逻辑分支，并将源代码的逻辑分支与待开发软件的功能逻辑分支进行分析比较，

通过比较结果分析所述功能逻辑分支覆盖率。

所述生成模块 202, 还用于根据比较结果生成所述源代码的功能逻辑分支覆盖率报告。

具体地, 所述应用服务器 2 通过所述生成模块 202 根据比较结果生成所述源代码的功能逻辑分支覆盖率报告。在本实施例中, 所述逻辑分支覆盖率报告包括标识出已经覆盖的函数及功能逻辑分支和没有覆盖的函数及功能逻辑分支。

所述反馈模块 204, 用于将所述功能逻辑分支覆盖率报告反馈至终端设备 1。

具体地, 所述应用服务器 2 将所述功能逻辑分支覆盖率报告发送至终端设备 1 的 web 页面显示, 开发人员可以根据所述功能逻辑分支覆盖率报告检查所述源代码的完整性, 即所述源代码是否已经覆盖了所有的功能逻辑分支, 如有遗漏, 即可进行针对性修改。

在本实施例中, 所述应用服务器 2 还设定一存储空间, 并存储所述功能逻辑分支覆盖率报告以及所述源代码至所述存储空间, 便于开发人员从存储空间访问所述功能逻辑分支覆盖率报告以及所述源代码, 跟进源代码的设计进展。

进一步地, 基于本申请代码覆盖率分析系统 200 的上述第一实施例, 提出本申请的第二实施例 (如图 4 所示)。本实施例中, 所述代码覆盖率分析系统 200 还包括获取模块 205, 匹配模块 206 以及调取模块 207, 其中,

所述获取模块 205, 用于当接收到所述终端设备 1 发送的程序源代码时, 获取所述源代码的标识。

所述匹配模块 206, 用于根据所述标识匹配预存储在所述应用服务器的测试项目逻辑分支。

具体地, 所述应用服务器 2 会接收多个不同的源代码, 每一个源代码都会有对应的预存储在所述应用服务器 2 内存中的测试项目逻辑分支。为了区

别每个源代码，每个源代码都有特定的标识，所述测试项目逻辑分支设置有与源代码对应的标识。当所述应用服务器 2 在接收到从终端设备 1 发送的源代码时，通过获取模块 205 获取所述源代码的标识，进而匹配模块 206 通过标识匹配预存储在所述应用服务器 2 的测试项目逻辑分支，从而确定对应的测试项目逻辑分支。

所述调取模块 207，用于调取所述测试项目逻辑分支。

具体地，所述应用服务器 2 通过所述调取模块 207 调取所述测试项目逻辑分支，进而与源代码的功能逻辑分支进行比较。

进一步地，基于本申请代码覆盖率分析系统 200 的上述第一实施例，提出本申请的第三实施例（如图 5 所示）。本实施例中，所述的代码覆盖率分析系统 200 还包括判断模块 208 及标注模块 209，其中：

所述判断模块 208，用于判断所述源代码的功能逻辑分支是否已经全部覆盖所述测试项目逻辑分支。

所述标注模块 209，用于当所述源代码的功能逻辑分支没有全部覆盖所述测试项目逻辑分支时，标注未覆盖的测试项目逻辑分支。

所述生成模块 202，还用于生成包含有标注有未覆盖的测试项目逻辑分支的功能逻辑分支覆盖率报告。

在本实施例中，当所述源代码的功能逻辑分支已经全部覆盖了所述测试项目逻辑分支时，所述应用服务器 2 不生成功能逻辑分支覆盖率报告。在其他实施例中，当所述源代码的功能逻辑分支已经全部覆盖了所述测试项目逻辑分支时，所述应用服务器 2 反馈已经全部覆盖的信息至所述终端设备 1 的 web 界面，提醒开发人员所述源代码的功能性已经完整，可以进行下一步的测试。

此外，本申请还提出一种代码覆盖率分析方法。

参阅图 6 所示，是本申请代码覆盖率分析方法第一实施例的流程示意图。在本实施例中，根据不同的需求，图 6 所示的流程图中的步骤的执行顺序可

以改变，某些步骤可以省略。

步骤 S301，当接收到所述终端设备 1 发送的程序源代码时，分析所述源代码的结构。

通常地，传统的测试程序代码的覆盖率分析都是针对代码行的覆盖情况，
5 但是这种通过分析代码行的覆盖情况，能够了解程序中有多少代码被执行了，并不能正代表一个功能单元的完整覆盖情况。假设一个功能单元根据不同条件需要调用 4 个函数组合来完成这个功能。若该功能会形成 16 个逻辑分支，而按代码行覆盖率统计，源代码只需要写 4 个逻辑分支就能把 4 个函数都覆盖掉，但从功能完整性上看，所述功能单元是不完整的。

10 举例而言，所述源代码的功能是完成用户名注册，完成注册的用户可以有不同类型的登陆方式，每一种登陆方式为一个功能逻辑分支。例如，通过调用函数 A，实现用户可以通过手机号码发送认证码登陆，调用函数 B，实现用户可以通过邮箱激活链接登陆，还可以同时调用函数 A 及函数 B，实现通过邮箱激活链接的同时通过手机号码发送认证码登陆，这种情况下，如果按
15 照传统的测试程序代码的覆盖分析，则只能分析出程序中的函数 A 和函数 B 被调用，而不能分析出同时调用函数 A 和函数 B 这一逻辑分支，从而导致测试案例出现功能覆盖不完整的情况。

因此，在本实施例中，所述应用服务器 2 与所述终端设备 1 建立长连接，当所述应用服务器 2 在接收到终端设备 1 发送的程序源代码时，所述应用服
20 务器 2 分析所述源代码的结构，具体地，分析函数及方法的调用关系，结构的包含关系，类的继承关系等等。

步骤 S302，根据所述源代码的结构生成所述源代码的功能逻辑分支。

具体地，根据所述源代码的结构，分析每一个功能单元调用到的功能函数，以及每一个功能函数的调用关系，进而生成源代码关于实现每一个功能
25 的功能逻辑分支。

在本实施例中，所述应用服务器 2 分析所述源代码的结构，形成所述源

代码的语法树。根据所述语法树查找功能函数及与功能函数的调用关系，同时去除注释以及不需要统计覆盖率信息的公共函数等，查找功能函数，分析功能函数的调用关系。根据所述功能函数的调用关系，生成所述源代码的逻辑分支。

- 5 步骤 S303，将所述源代码的功能逻辑分支与预存储在所述应用服务器的测试项目逻辑分支进行比较。

通常地，开发人员进行源代码设计时，必然了解待开发软件的功能。在本实施例中，为了验证所述源代码完整性，开发人员预先把源代码需要的逻辑分支存在所述应用服务器 2 的存储空间。在本实施例中，当生成了所述
10 源代码的逻辑分支时，所述应用服务器 2 调预先存储有的待开发软件的功能逻辑分支，并将源代码的逻辑分支与待开发软件的功能逻辑分支进行分析比较，通过比较结果分析所述功能逻辑分支覆盖率。

步骤 S304，根据比较结果生成所述源代码的功能逻辑分支覆盖率报告。

具体地，所述应用服务器 2 通过所述生成模块 202 根据比较结果生成所述
15 源代码的功能逻辑分支覆盖率报告。在本实施例中，所述逻辑分支覆盖率报告包括标识出已经覆盖的函数及功能逻辑分支和没有覆盖的函数及功能逻辑分支。

步骤 S305，将所述功能逻辑分支覆盖率报告反馈至终端设备 1。

具体地，所述应用服务器 2 将所述功能逻辑分支覆盖报告发送至终端设
20 备 1 的 web 页面显示，开发人员可以根据所述功能逻辑分支覆盖率报告检查所述源代码的完整性，即所述源代码是否已经覆盖了所有的功能逻辑分支，如有遗漏，即可进行针对性修改。

在本实施例中，所述应用服务器 2 还设定一存储空间，并存储所述功能逻辑分支覆盖率报告以及所述源代码至所述存储空间，便于开发人员从存储空间访问所述功能逻辑分支覆盖率报告以及所述源代码，跟进源代码的设计
25 进展。

通过上述步骤 S301-305，本申请所提出的代码覆盖率分析方法，首先当接收到所述终端设备 1 发送的程序源代码时，分析所述源代码的结构；然后，根据所述源代码的结构生成所述源代码的功能逻辑分支；进一步地，将所述源代码的功能逻辑分支与预存储在所述应用服务器 2 的测试项目逻辑分支进行比较；接着，根据比较结果生成所述源代码的功能逻辑分支覆盖率报告；最后将所述功能逻辑分支覆盖率报告反馈至终端设备 1。这样，既可以避免现有技术中通过代码覆盖率统计，仅能了解到程序中有多少的代码被执行了，但是并不能真正了解一个程序模块的完整覆盖情况的缺陷，也可以使得通过逻辑分支的比较，确认源代码是否已经覆盖了所有的功能逻辑分支，保证测试项目的功能完整性。

进一步地，基于本申请代码覆盖率分析方法的上述第一实施例，提出本申请代码覆盖率分析方法的第二实施例。

如图 7 所示，是本申请代码覆盖率分析方法第二实施例的流程示意图。本实施例中，所述将所述源代码的功能逻辑分支与预存储在应用服务器 2 的测试案例逻辑分支进行比较，生成所述源代码的功能逻辑分支覆盖率报告的步骤之前，还包括如下步骤：

步骤 S401，当接收到所述终端设备 1 发送的程序源代码时，获取所述源代码的标识。

步骤 S402，根据所述标识匹配预存储在所述应用服务器的测试项目逻辑分支。

具体地，所述应用服务器 2 会接收多个不同的源代码，每一个源代码都会有对应的预存储在所述应用服务器 2 内存中的测试项目逻辑分支。为了区别每个源代码，每个源代码都有特定的标识，所述测试项目逻辑分支设置有与源代码对应的标识。当所述应用服务器 2 在接收到从终端设备 1 发送的源代码时，通过获取模块 205 获取所述源代码的标识，进而匹配模块 206 通过标识匹配预存储在所述应用服务器 2 的测试项目逻辑分支，从而确定对应的

测试项目逻辑分支。

步骤 S403，调取所述测试项目逻辑分支。

具体地，所述应用服务器 2 通过所述调取模块 207 调取所述测试项目逻辑分支，进而与源代码的功能逻辑分支进行比较。

- 5 通过上述步骤 S401-S402，本申请所提出的代码覆盖率分析方法，可以通过获取所述源代码的标识，进而匹配预存储在所述应用服务器 2 中的测试项目逻辑分支，以从不同的测试项目逻辑分支中确定与所述源代码对应的测试项目逻辑分支，提高测试效率。

- 进一步地，基于本申请代码覆盖率分析方法的上述第一和第二实施例，
10 提出本申请代码覆盖率分析方法的第三实施例。

如图 8 所示，是本申请代码覆盖率分析方法第三实施例的流程示意图。本实施例中，所述生成所述源代码的功能逻辑分支覆盖率报告的步骤，具体包括：

- 步骤 S501，判断所述源代码的功能逻辑分支是否已经全部覆盖所述测试
15 项目逻辑分支。

步骤 S502，当所述源代码的功能逻辑分支没有全部覆盖所述测试项目逻辑分支时，标注未覆盖的测试项目逻辑分支。

步骤 S503，生成包含有标注有未覆盖的测试项目逻辑分支的功能逻辑分支覆盖率报告。

- 20 在本实施例中，当所述源代码的功能逻辑分支已经全部覆盖了所述测试项目逻辑分支时，所述应用服务器 2 不生成功能逻辑分支覆盖率报告。在其他实施例中，当所述源代码的功能逻辑分支已经全部覆盖了所述测试项目逻辑分支时，所述应用服务器 2 反馈已经全部覆盖的信息至所述终端设备 1 的 web 界面，提醒开发人员所述源代码的功能性已经完整，可以进行下一步的
25 测试。

通过上述步骤 S501-S503，本申请所提出的代码覆盖率分析方法，还能够

在判断出所述源代码的功能逻辑分支没有全部覆盖所述测试项目逻辑分支时，标注未覆盖的测试项目逻辑分支，并生成包含有标注有未覆盖的测试项目逻辑分支的功能逻辑分支覆盖率报告，方便开发人员直观地了解到没有被覆盖的功能逻辑分支，便于开发人员针对性修改。

5 上述本申请实施例序号仅仅为了描述，不代表实施例的优劣。

通过以上的实施方式的描述，本领域的技术人员可以清楚地了解到上述实施例方法可借助软件加必需的通用硬件平台的方式来实现，当然也可以通过硬件，但很多情况下前者是更佳的实施方式。基于这样的理解，本申请的技术方案本质上或者说对现有技术做出贡献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（如ROM/RAM、磁碟、光盘）
10 中，包括若干指令用以使得一台终端设备（可以是手机，计算机，服务器，空调器，或者网络设备等等）执行本申请各个实施例所述的方法。

以上仅为本申请的优选实施例，并非因此限制本申请的专利范围，凡是利用本申请说明书及附图内容所作的等效结构或等效流程变换，或直接或间接运用在其他相关的技术领域，均同理包括在本申请的专利保护范围内。
15

权利要求书

1. 一种代码覆盖率分析方法，应用于应用服务器，其特征在于，所述方法包括：

当接收到终端设备发送的程序源代码时，分析所述源代码的结构；

5 根据所述源代码的结构生成所述源代码的功能逻辑分支；

将所述源代码的功能逻辑分支与预存储在所述应用服务器的测试项目逻辑分支进行比较；

根据比较结果生成所述源代码的功能逻辑分支覆盖率报告；

将所述功能逻辑分支覆盖率报告反馈至所述终端设备。

10 2. 如权利要求 1 所述的代码覆盖率分析方法，其特征在于，所述根据所述源代码的结构生成所述源代码的逻辑分支的步骤，具体包括如下步骤：

分析所述源代码的结构，形成所述源代码的语法树；

根据所述语法树查找功能函数及与功能函数的调用关系；

根据所述功能函数的调用关系，生成所述源代码的逻辑分支。

15 3. 如权利要求 1 所述的代码覆盖率分析方法，其特征在于，所述将所述源代码的功能逻辑分支与预存储在应用服务器的测试案例逻辑分支进行比较，生成所述源代码的功能逻辑分支覆盖率报告的步骤之前，还包括如下步骤：

当接收到所述终端设备发送的程序源代码时，获取所述源代码的标识；

20 根据所述标识匹配预存储在所述应用服务器的测试项目逻辑分支；

调取所述测试项目逻辑分支。

4. 如权利要求 2 所述的代码覆盖率分析方法，其特征在于，所述将所述源代码的功能逻辑分支与预存储在应用服务器的测试案例逻辑分支进行比较，生成所述源代码的功能逻辑分支覆盖率报告的步骤之前，还包括如下步
25 骤：

当接收到所述终端设备发送的程序源代码时，获取所述源代码的标识；

根据所述标识匹配预存储在所述应用服务器的测试项目逻辑分支；
调取所述测试项目逻辑分支。

5. 如权利要求 1 所述的代码覆盖率分析方法，其特征在于，所述生成所述源代码的功能逻辑分支覆盖率报告的步骤，具体包括：

5 判断所述源代码的功能逻辑分支是否已经全部覆盖所述测试项目逻辑分支；

当所述源代码的功能逻辑分支没有全部覆盖所述测试项目逻辑分支时，
标注未覆盖的测试项目逻辑分支；

生成包含有标注有未覆盖的测试项目逻辑分支的功能逻辑分支覆盖率报
10 告。

6. 如权利要求 1 所述的代码覆盖率分析方法，其特征在于，所述方法还包括：

设定一个存储空间；

存储所述功能逻辑分支覆盖率报告以及所述源代码至所述存储空间。

15 7. 如权利要求 2 所述的代码覆盖率分析方法，其特征在于，所述方法还包括：

设定一个存储空间；

存储所述功能逻辑分支覆盖率报告以及所述源代码至所述存储空间。

8. 如权利要求 3 所述的代码覆盖率分析方法，其特征在于，所述方法还
20 包括：

设定一个存储空间；

存储所述功能逻辑分支覆盖率报告以及所述源代码至所述存储空间。

9. 如权利要求 5 所述的代码覆盖率分析方法，其特征在于，所述方法还包括：

25 设定一个存储空间；

存储所述功能逻辑分支覆盖率报告以及所述源代码至所述存储空间。

10. 一种应用服务器，其特征在于，所述应用服务器包括存储器、处理器，所述存储器上存储有可在所述处理器上运行的代码覆盖率分析系统，所述代码覆盖率分析系统被所述处理器执行时实现如下步骤：

当接收到终端设备发送的程序源代码时，分析所述源代码的结构；

5 根据所述源代码的结构生成所述源代码的功能逻辑分支；

将所述源代码的功能逻辑分支与预存储在所述应用服务器的测试项目逻辑分支进行比较；

根据比较结果生成所述源代码的功能逻辑分支覆盖率报告；

将所述功能逻辑分支覆盖率报告反馈至所述终端设备。

10 11. 如权利要求 10 所述的应用服务器，其特征在于，所述根据所述源代码的结构生成所述源代码的逻辑分支的步骤，具体包括如下步骤：

分析所述源代码的结构，形成所述源代码的语法树；

根据所述语法树查找功能函数及与功能函数的调用关系；

根据所述功能函数的调用关系，生成所述源代码的逻辑分支。

15 12. 如权利要求 10 所述的应用服务器，其特征在于，所述将所述源代码的功能逻辑分支与预存储在应用服务器的测试案例逻辑分支进行比较，生成所述源代码的功能逻辑分支覆盖率报告的步骤之前，还包括如下步骤：

当接收到所述终端设备发送的程序源代码时，获取所述源代码的标识；

根据所述标识匹配预存储在所述应用服务器的测试项目逻辑分支；

20 调取所述测试项目逻辑分支。

13. 如权利要求 11 所述的应用服务器，其特征在于，所述将所述源代码的功能逻辑分支与预存储在应用服务器的测试案例逻辑分支进行比较，生成所述源代码的功能逻辑分支覆盖率报告的步骤之前，还包括如下步骤：

当接收到所述终端设备发送的程序源代码时，获取所述源代码的标识；

25 根据所述标识匹配预存储在所述应用服务器的测试项目逻辑分支；

调取所述测试项目逻辑分支。

14. 如权利要求 10 所述的应用服务器，其特征在于，所述生成所述源代码的功能逻辑分支覆盖率报告的步骤，具体包括：

判断所述源代码的功能逻辑分支是否已经全部覆盖所述测试项目逻辑分支；

5 当所述源代码的功能逻辑分支没有全部覆盖所述测试项目逻辑分支时，标注未覆盖的测试项目逻辑分支；

生成包含有标注有未覆盖的测试项目逻辑分支的功能逻辑分支覆盖率报告。

15. 如权利要求 10 所述的应用服务器，其特征在于，所述代码覆盖率分析系统被所述处理器执行时还实现如下步骤：

设定一个存储空间；

存储所述功能逻辑分支覆盖率报告以及所述源代码至所述存储空间。

16. 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代码覆盖率分析系统，所述代码覆盖率分析系统可被至少一个处理器执行，以使所述至少一个处理器执行时实现如下步骤：

当接收到终端设备发送的程序源代码时，分析所述源代码的结构；

根据所述源代码的结构生成所述源代码的功能逻辑分支；

将所述源代码的功能逻辑分支与预存储在所述应用服务器的测试项目逻辑分支进行比较；

20 根据比较结果生成所述源代码的功能逻辑分支覆盖率报告；

将所述功能逻辑分支覆盖率报告反馈至所述终端设备。

17. 如权利要求 16 所述的计算机可读存储介质，其特征在于，所述根据所述源代码的结构生成所述源代码的逻辑分支的步骤，具体包括如下步骤：

分析所述源代码的结构，形成所述源代码的语法树；

25 根据所述语法树查找功能函数及与功能函数的调用关系；

根据所述功能函数的调用关系，生成所述源代码的逻辑分支。

18. 如权利要求 16 所述的计算机可读存储介质，其特征在于，所述将所述源代码的功能逻辑分支与预存储在应用服务器的测试案例逻辑分支进行比较，生成所述源代码的功能逻辑分支覆盖率报告的步骤之前，还包括如下步骤：

- 5 当接收到所述终端设备发送的程序源代码时，获取所述源代码的标识；
根据所述标识匹配预存储在所述应用服务器的测试项目逻辑分支；
调取所述测试项目逻辑分支。

19. 如权利要求 16 所述的计算机可读存储介质，其特征在于，所述生成所述源代码的功能逻辑分支覆盖率报告的步骤，具体包括：

- 10 判断所述源代码的功能逻辑分支是否已经全部覆盖所述测试项目逻辑分支；

当所述源代码的功能逻辑分支没有全部覆盖所述测试项目逻辑分支时，
标注未覆盖的测试项目逻辑分支；

- 15 生成包含有标注有未覆盖的测试项目逻辑分支的功能逻辑分支覆盖率报告。

20. 如权利要求 16 所述的计算机可读存储介质，其特征在于，所述代码覆盖率分析系统被所述处理器执行时还实现如下步骤：

设定一个存储空间；

存储所述功能逻辑分支覆盖率报告以及所述源代码至所述存储空间。

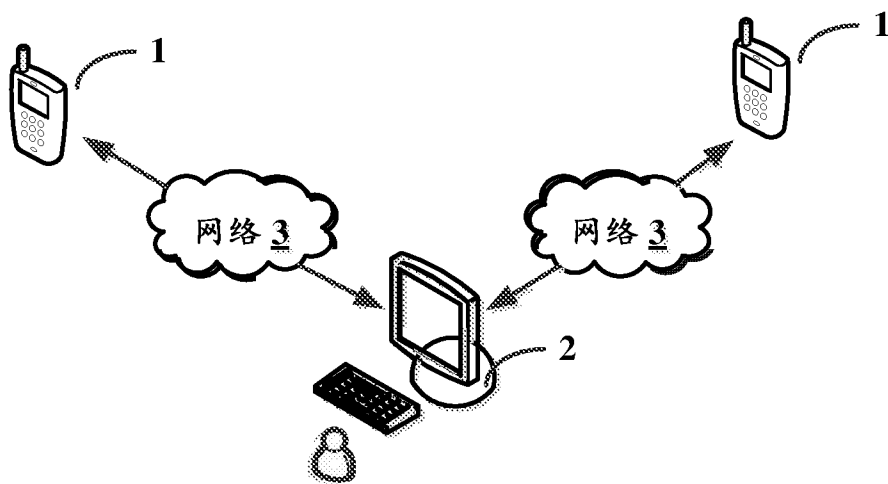


图 1

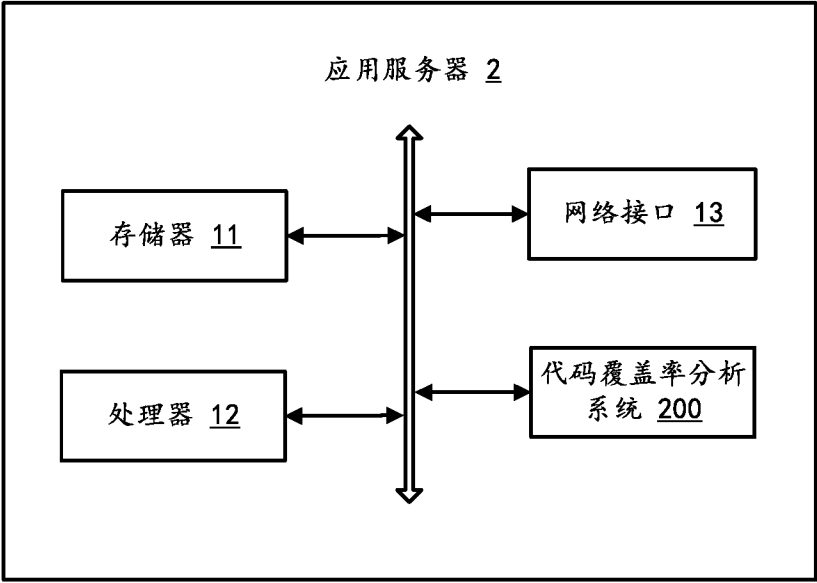


图 2



图 3



图 4



图 5

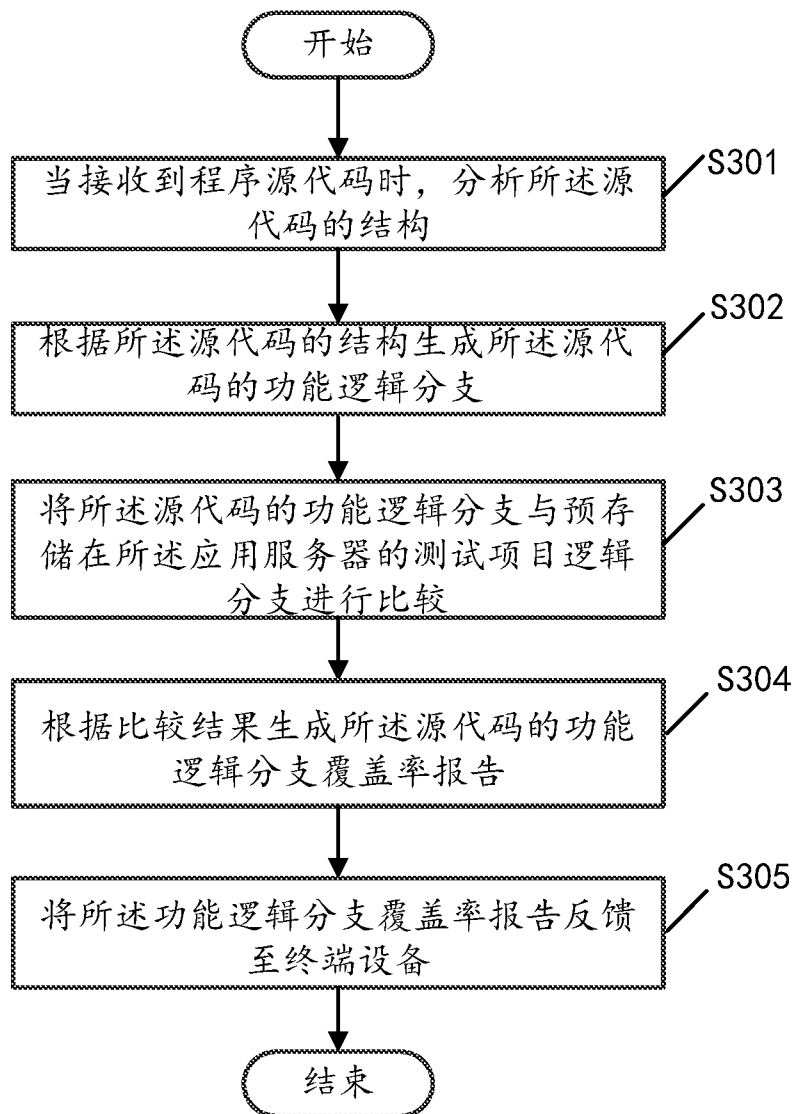


图 6

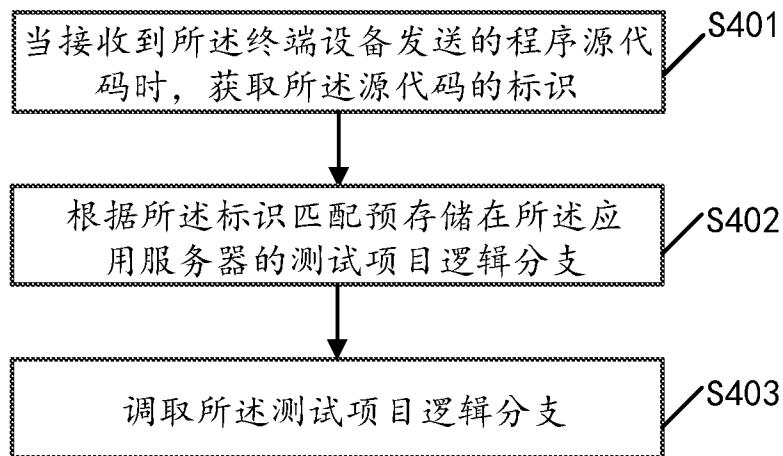


图 7

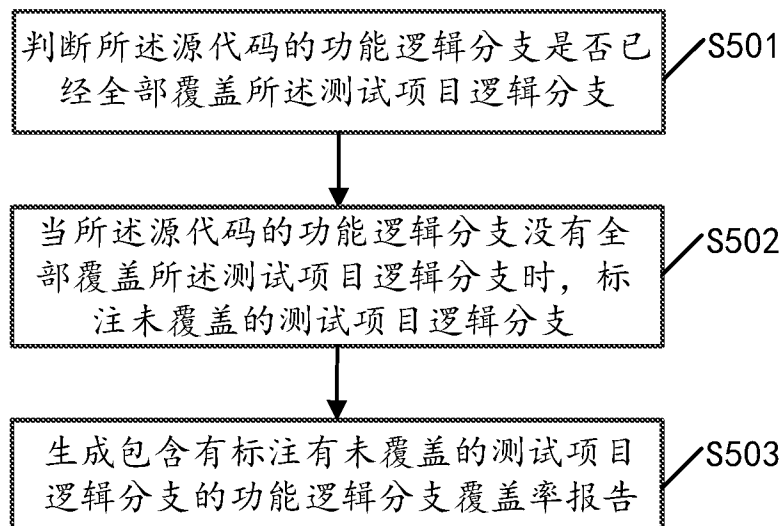


图 8

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2018/076141

A. CLASSIFICATION OF SUBJECT MATTER

G06F 11/36 (2006.01) i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

WPI, EPODOC, CNPAT, CNKI: 源, 程序, 代码, 软件, 覆盖率, 功能, 逻辑分支, 模块, 单元, 测试, source, program, code, software, coverage, rate, function, logic, branch, module, unit, test

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	CN 106649110 A (CHINA STANDARD SOFTWARE CO., LTD.) 10 May 2017 (10.05.2017), description, paragraphs [0070]-[0080]	1-20
A	CN 101706751 A (ZTE CORPORATION) 12 May 2010 (12.05.2010), entire document	1-20
A	CN 105630670 A (BEIJING QIHOO TECHNOLOGY CO., LTD. et al.) 01 June 2016 (01.06.2016), entire document	1-20
A	US 2016328314 A1 (HCL TECHNOLOGIES LIMITED) 10 November 2016 (10.11.2016), entire document	1-20
A	US 2009287729 A1 (MICROSOFT CORPORATION) 19 November 2009 (19.11.2009), entire document	1-20

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents:

“A” document defining the general state of the art which is not considered to be of particular relevance

“E” earlier application or patent but published on or after the international filing date

“L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

“O” document referring to an oral disclosure, use, exhibition or other means

“P” document published prior to the international filing date but later than the priority date claimed

“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

“X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

“Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

“&” document member of the same patent family

Date of the actual completion of the international search
27 June 2018

Date of mailing of the international search report
12 July 2018

Name and mailing address of the ISA
State Intellectual Property Office of the P. R. China
No. 6, Xitucheng Road, Jimenqiao
Haidian District, Beijing 100088, China
Facsimile No. (86-10) 62019451

Authorized officer
WU, Qiong
Telephone No. (86-10) 53961338

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.
PCT/CN2018/076141

Patent Documents referred in the Report	Publication Date	Patent Family	Publication Date
CN 106649110 A	10 May 2017	None	
CN 101706751 A	12 May 2010	WO 2011060655 A1	26 May 2011
CN 105630670 A	01 June 2016	None	
US 2016328314 A1	10 November 2016	IN 201501249 I1	05 June 2015
US 2009287729 A1	19 November 2009	None	

国际检索报告

国际申请号

PCT/CN2018/076141

A. 主题的分类

G06F 11/36(2006.01)i

按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类

B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

G06F

包含在检索领域中的除最低限度文献以外的检索文献

在国际检索时查阅的电子数据库(数据库的名称, 和使用的检索词(如使用))

WPI, EPODOC, CNPAT, CNKI: 源, 程序, 代码, 软件, 覆盖率, 功能, 逻辑分支, 模块, 单元, 测试, source, program, code, software, coverage, rate, function, logic, branch, module, unit, test

C. 相关文件

类 型*	引用文件, 必要时, 指明相关段落	相关的权利要求
A	CN 106649110 A (中标软件有限公司) 2017年 5月 10日 (2017 - 05 - 10) 说明书第[0070]-[0080]段	1-20
A	CN 101706751 A (中兴通讯股份有限公司) 2010年 5月 12日 (2010 - 05 - 12) 全文	1-20
A	CN 105630670 A (北京奇虎科技有限公司 等) 2016年 6月 1日 (2016 - 06 - 01) 全文	1-20
A	US 2016328314 A1 (HCL TECHNOLOGIES LIMITED) 2016年 11月 10日 (2016 - 11 - 10) 全文	1-20
A	US 2009287729 A1 (MICROSOFT CORPORATION) 2009年 11月 19日 (2009 - 11 - 19) 全文	1-20

☐ 其余文件在C栏的续页中列出。☒ 见同族专利附件。

* 引用文件的具体类型:

“A” 认为不特别相关的表示了现有技术一般状态的文件

“E” 在国际申请日的当天或之后公布的在先申请或专利

“L” 可能对优先权要求构成怀疑的文件, 或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件 (如具体说明的)

“O” 涉及口头公开、使用、展览或其他方式公开的文件

“P” 公布日先于国际申请日但迟于所要求的优先权日的文件

“T” 在申请日或优先权日之后公布, 与申请不相抵触, 但为了理解发明之理论或原理的在后文件

“X” 特别相关的文件, 单独考虑该文件, 认定要求保护的发明不是新颖的或不具有创造性

“Y” 特别相关的文件, 当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时, 要求保护的发明不具有创造性

“&” 同族专利的文件

国际检索实际完成的日期

2018年 6月 27日

国际检索报告邮寄日期

2018年 7月 12日

ISA/CN的名称和邮寄地址

中华人民共和国国家知识产权局(ISA/CN)
中国北京市海淀区蓟门桥西土城路6号 100088

受权官员

吴琼

传真号 (86-10)62019451

电话号码 86-(10)-53961338

国际检索报告
关于同族专利的信息

国际申请号

PCT/CN2018/076141

检索报告引用的专利文件			公布日 (年/月/日)	同族专利			公布日 (年/月/日)
CN	106649110	A	2017年 5月 10日	无			
CN	101706751	A	2010年 5月 12日	WO	2011060655	A1	2011年 5月 26日
CN	105630670	A	2016年 6月 1日	无			
US	2016328314	A1	2016年 11月 10日	IN	201501249	I1	2015年 6月 5日
US	2009287729	A1	2009年 11月 19日	无			

表 PCT/ISA/210 (同族专利附件) (2015年1月)