

(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
2024 年 3 月 28 日 (28.03.2024)



(10) 国际公布号
WO 2024/061036 A1

(51) 国际专利分类号:
G06F 9/54 (2006.01) **G06F 15/163** (2006.01)

(21) 国际申请号: PCT/CN2023/117917

(22) 国际申请日: 2023 年 9 月 11 日 (11.09.2023)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
202211137787.7 2022年9月19日 (19.09.2022) CN

(71) 申请人: 华为技术有限公司 (HUAWEI TECHNOLOGIES CO., LTD.) [CN/CN]; 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。

(72) 发明人: 陈本智 (CHEN, Benzhi); 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。 兰守忍 (LAN, Shouren); 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。 余枝强 (YU, Zhiqiang); 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。

(74) 代理人: 广州三环专利商标代理有限公司 (SCIHEAD IP LAW FIRM); 中国广东省广州市越秀区先烈中路 80 号汇华商贸大厦 1508 室, Guangdong 510070 (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, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI,

(54) Title: INTER-ENGINE COMMUNICATION METHOD AND RELATED DEVICE

(54) 发明名称: 一种引擎间的通信方法及相关设备

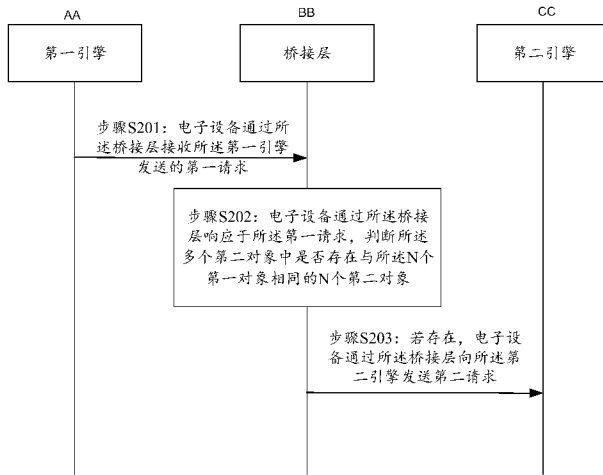


图 3

Step S201 An electronic device receives a first request sent by the first engine by means of the bridging layer

Step S202 In response to the first request, and by means of the bridging layer, the electronic device determines whether there are a number N of second objects which are the same as N first objects among the multiple second objects

Step S203 If so, the electronic device sends a second request to the second engine by means of the bridging layer

AA First engine

BB Bridging layer

CC Second engine

(57) Abstract: Embodiments of the present application disclose an inter-engine communication method and a related device, which are characterized by being applied to an operating system. The operating system comprises a first engine, a second engine and a bridging layer, the first engine comprising multiple first objects, and the second engine comprising multiple second objects. The method comprises: receiving a first request sent by the first engine by means of the bridging layer, the first request being used to request execution of a number N of first objects among the multiple first objects, N being an integer greater than 0; by means of the bridging

GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, 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, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, 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, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, 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, ME, MK, MT, NL, NO, PL, PT, 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)。

根据细则4.17的声明:

- 关于申请人有权要求在先申请的优先权(细则4.17(iii))

本国际公布:

- 包括国际检索报告(条约第21条(3))。

layer, and in response to the first request, determining whether there are N second objects which are the same as the N first objects among the multiple second objects; if so, sending a second request to the second engine by means of the bridging layer, the second request being used to request that the second engine execute the N second objects according to a corresponding function parameter. Using an embodiment of the present application, communication between different engines may be implemented.

(57) 摘要: 本申请实施例公开了一种引擎间的通信方法及相关设备, 其特征在于, 应用于操作系统, 所述操作系统包括第一引擎、第二引擎和桥接层, 所述第一引擎包括多个第一对象, 所述第二引擎包括多个第二对象, 所述方法包括: 通过所述桥接层接收所述第一引擎发送的第一请求, 所述第一请求用于请求执行所述多个第一对象中的N个第一对象, N为大于0的整数; 通过所述桥接层响应于所述第一请求, 判断所述多个第二对象中是否存在与所述N个第一对象相同的N个第二对象; 若存在, 通过所述桥接层向所述第二引擎发送第二请求, 所述第二请求用于请求所述第二引擎按照对应的函数参数执行所述N个第二对象。采用本申请实施例能够实现不同引擎之间的通信。

一种引擎间的通信方法及相关设备

本申请要求于 2022 年 09 月 19 日提交中国专利局、申请号为 202211137787.7、申请名称为“一种引擎间的通信方法及相关设备”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

技术领域

本申请涉及计算机技术领域，尤其涉及一种引擎间的通信方法及相关设备。

背景技术

Ets 是一种可扩展的 TS (TypeScript) 语言，是鸿蒙应用主要的开发语言，具有与 JS (JavaScript) 生态融合，高效的类型系统、简洁高效的声明式开发范式的优点。为了进一步扩展鸿蒙生态，需要引入 Web 生态，Web 生态主要是 HTML5 (H5) 语言开发的，通过鸿蒙的方舟开发框架 (ArkUI) 中的 web 组件进行加载并显示 H5 页面。

鸿蒙 Ets 代码运行在 Ark 引擎的 JS 运行环境 (JS runtime) 中，H5 的 JS 代码运行在网络视图 (WebView) 的 V8 引擎的 JS 运行环境 (JS runtime) 中。由于这两种语言运行在不同的 JS 引擎中，也可以理解为这两种语言运行在两个彼此独立的沙箱隔离的上下文 (context) 中，导致 H5 和 Ets 之间无法直接进行通信，同时在 H5 中无法复用 Ets 侧开发者自定义或鸿蒙系统的应用程序编程接口 (Application Programming Interface, API)，严重影响了鸿蒙应用生态的推广和扩展。

因此，如何提供一种引擎间的通信方法及相关设备，以实现不同引擎之间的通信，如实现 H5 侧和 Ets 侧的信息互通，是亟待解决的问题。

发明内容

本申请实施例所要解决的技术问题在于，提供一种引擎间的通信方法及相关设备，以实现不同 JS 引擎之间的通信。

第一方面，本申请实施例提供一种引擎间的通信方法，其特征在于，应用于操作系统，所述操作系统包括第一引擎、第二引擎和桥接层，所述第一引擎包括多个第一对象，所述第二引擎包括多个第二对象，所述方法包括：通过所述桥接层接收所述第一引擎发送的第一请求，所述第一请求用于请求执行所述多个第一对象中的 N 个第一对象，N 为大于 0 的整数；通过所述桥接层响应于所述第一请求，判断所述多个第二对象中是否存在与所述 N 个第一对象相同的 N 个第二对象；若存在，通过所述桥接层向所述第二引擎发送第二请求，所述第二请求用于请求所述第二引擎按照对应的函数参数执行所述 N 个第二对象。

本发明实施例中，可以在第一引擎和第二引擎之间增加一个桥接层，以通过桥接层实现不同引擎之间的通信。具体的，可以在第一引擎上运行第一应用，第一应用中可以包括多个第一对象；可以在第二引擎上运行第二应用，第二应用中可以包括多个第二对象，且第一应用和第二应用运行在两个不同的上下文中。进一步地，在两个引擎运行过程中，第一引擎可以向桥接层发送执行一个或多个第一对象的请求，然后桥接层判断多个第二对象中是否存在与上述一个或多个第一对象相同的第二对象，若存在，桥接层再向第二引擎发送执行对应的一个或多个第二对象的请求，且在该请求中可以包括执行过程中需要使用的函数参数，从而第二引擎可以基于函数参数执行对应的第二对象，实现了跨引擎通信。由于在本申请中，避免将第二引擎的对象直接在第一引擎中注册，且通过桥接层高效实现两个不同引擎之间的信息交互，因此在保证安全的前提下，解决了两个运行环境相互隔离无法调用彼此对象（或函数）的问题。

在一种可能的实现方式中，所述判断所述多个第二对象中是否存在与所述 N 个第一对象相同的 N 个第二对象，包括：判断所述多个第二对象中是否存在与所述 N 个第一对象的对象标识相同的 N 个第二对象；其中，若存在则所述 N 个第二对象与所述 N 个第一对象相同。

在本发明实施例中，桥接层可以从第一引擎获取到多个第一对象的对象名（也可以称为对象标识），桥接层也可以从第二引擎获取到多个第二对象的对象名。进一步地，当桥接层接收第一引擎发送的请求后，可以判断多个第二对象中是否存在与请求中的 N 个第一对象的对象名相同的 N 个第二对象，若存在，则桥接层可以向第二引擎发送执行请求，从而实现在运行过程中第一引擎可以通过桥接层调用第二引擎中的

对象。由于在本申请中，避免将第二引擎的对象直接在第一引擎中注册，且通过桥接层高效实现两个不同引擎之间的信息交互，因此在保证安全的前提下，解决了两个运行环境相互隔离无法调用彼此对象（或函数）的问题。

在一种可能的实现方式中，所述判断所述多个第二对象中是否存在与所述N个第一对象相同的N个第二对象，包括：判断所述多个第二对象中是否存在与所述N个第一对象的对象标识相同的N个第二对象；若存在，则通过第一记录表确定所述N个第一对象中每个第一对象包括的第一函数；所述第一记录表包括所述多个第一对象中每个第一对象包括的所述第一函数；基于第二记录表，判断所述N个第一对象中每个第一对象包括的所述第一函数与所述N个第二对象中每个第二对象包括的第二函数是否一致；其中，若一致则所述多个第二对象中存在与所述N个第一对象相同的所述N个第二对象；所述第二记录表包括所述第二对象中每个所述第二对象所包括的第二函数。

在本发明实施例中，桥接层不仅需要判断对象名是否相同，还需要通过查表确定对象名下的函数是否相同，若相同，则桥接层可以向第二引擎发送执行请求，从而实现在运行过程中第一引擎可以通过桥接层调用第二引擎中的对象。由于在本申请中，避免将第二引擎的对象直接在第一引擎中注册，且通过桥接层高效实现两个不同引擎之间的信息交互，因此在保证安全的前提下，解决了两个运行环境相互隔离无法调用彼此对象（或函数）的问题。

在一种可能的实现方式中，所述方法还包括：通过所述桥接层获取所述多个第一对象中每个所述第一对象对应的对象标识，以及每个所述第一对象包括的所述第一函数，生成所述第一记录表；通过所述桥接层获取所述多个第二对象中每个所述第二对象对应的对象标识，以及每个所述第二对象包括的所述第二函数，生成所述第二记录表。

在本发明实施例中，桥接层可以从第一引擎获取到多个第一对象的对象名（也可以称为对象标识）以及对象名下的函数名并生成第一记录表，桥接层也可以从第二引擎获取到多个第二对象的对象名以及对象名下的函数名并生成第二记录表，进而桥接层可以基于第一记录表和第二记录表判断请求中的对象名下的函数是否相同。由于在本申请中，避免将第二引擎的对象直接在第一引擎中注册，且通过桥接层高效实现两个不同引擎之间的信息交互，因此在保证安全的前提下，解决了两个运行环境相互隔离无法调用彼此对象（或函数）的问题。

在一种可能的实现方式中，所述方法还包括：响应于所述第二请求，通过所述第二引擎按照对应的函数参数执行所述N个第二对象，并生成所述N个第二对象对应的执行结果；通过所述桥接层向所述第一引擎转发所述第二引擎发送的所述N个第二对象对应的所述执行结果。

在本发明实施例中，第二引擎可以响应于桥接层的请求，可以基于函数参数执行对应的第二对象，并生成相应的执行结果，然后通过桥接层向第一引擎转发执行结果，从而实现跨引擎的数据交互。由于在本申请中，避免将第二引擎的对象直接在第一引擎中注册，且通过桥接层高效实现两个不同引擎之间的信息交互，因此在保证安全的前提下，解决了两个运行环境相互隔离无法调用彼此对象（或函数），以及无法进行数据传递的问题。

在一种可能的实现方式中，所述方法还包括：若所述多个第二对象中不存在与所述N个第一对象相同的所述N个第二对象，则通过所述桥接层向所述第一引擎发送第一通知，所述第一通知包括所述N个第一对象执行错误的消息。

在本发明实施例中，若不存在相同的对象，则表示第二引擎无法执行请求中的对象，进而可以通过桥接层通知第一引擎请求中的对象不会被执行。由于在本申请中，避免将第二引擎的对象直接在第一引擎中注册，且通过桥接层高效实现两个不同引擎之间的信息交互，因此在保证安全的前提下，解决了两个运行环境相互隔离无法调用彼此对象（或函数），以及无法进行数据传递的问题。

第二方面，本申请提供了一种电子设备，其特征在于，所述电子设备运行操作系统，所述操作系统包括第一引擎、第二引擎和桥接层，所述第一引擎包括多个第一对象，所述第二引擎包括多个第二对象，所述装置包括：第一接收单元，用于通过所述桥接层接收所述第一引擎发送的第一请求，所述第一请求用于请求执行所述多个第一对象中的N个第一对象，N为大于0的整数；第一处理单元，用于通过所述桥接层响应于所述第一请求，判断所述多个第二对象中是否存在与所述N个第一对象相同的N个第二对象；第一发送单元，用于，若存在，通过所述桥接层向所述第二引擎发送第二请求，所述第二请求用于请求所述第二引擎按照对应的函数参数执行所述N个第二对象。

在一种可能的实现方式中，所述第一处理单元，具体用于：判断所述多个第二对象中是否存在与所述

N 个第一对象的对象标识相同的 N 个第二对象；其中，若存在则所述 N 个第二对象与所述 N 个第一对象相同。

在一种可能的实现方式中，所述第一处理单元，具体用于：判断所述多个第二对象中是否存在与所述 N 个第一对象的对象标识相同的 N 个第二对象；若存在，则通过第一记录表确定所述 N 个第一对象中每个第一对象包括的第一函数；所述第一记录表包括所述多个第一对象中每个第一对象包括的所述第一函数；基于第二记录表，判断所述 N 个第一对象中每个第一对象包括的所述第一函数与所述 N 个第二对象中每个第二对象包括的第二函数是否一致；其中，若一致则所述多个第二对象中存在与所述 N 个第一对象相同的所述 N 个第二对象；所述第二记录表包括所述第二对象中每个所述第二对象所包括的第二函数。

在一种可能的实现方式中，所述设备还包括：第二接收单元，用于通过所述桥接层获取所述多个第一对象中每个所述第一对象对应的对象标识，以及每个所述第一对象包括的所述第一函数，生成所述第一记录表；第三接收单元，用于通过所述桥接层获取所述多个第二对象中每个所述第二对象对应的对象标识，以及每个所述第二对象包括的所述第二函数，生成所述第二记录表。

在一种可能的实现方式中，所述设备还包括：第二处理单元，用于响应于所述第二请求，通过所述第二引擎按照对应的函数参数执行所述 N 个第二对象，并生成所述 N 个第二对象对应的执行结果；第二发送单元，用于通过所述桥接层向所述第一引擎转发所述第二引擎发送的所述 N 个第二对象对应的所述执行结果。

在一种可能的实现方式中，所述设备还包括：第三发送单元，用于若所述多个第二对象中不存在与所述 N 个第一对象相同的所述 N 个第二对象，则通过所述桥接层向所述第一引擎发送第一通知，所述第一通知包括所述 N 个第一对象执行错误的消息。

第三方面，本申请提供了一种计算机存储介质，其特征在于，所述计算机存储介质存储有计算机程序，该计算机程序被处理器执行时实现上述第一方面任意一项所述的方法。

第四方面，本申请实施例提供一种电子设备，该电子设备中包括处理器，处理器被配置为支持该电子设备实现第一方面提供的一种引擎间的通信方法中相应的功能。该电子设备还可以包括存储器，存储器用于与处理器耦合，其保存该电子设备必要的程序指令和数据。该电子设备还可以包括通信接口，用于该电子设备与其他设备或通信网络通信。

第五方面，本申请提供了一种芯片系统，该芯片系统包括处理器，用于支持电子设备实现上述第一方面中所涉及的功能，例如，生成或处理上述引擎间的通信方法中所涉及的信息。在一种可能的设计中，所述芯片系统还包括存储器，所述存储器，用于保存电子设备必要的程序指令和数据。该芯片系统，可以由芯片构成，也可以包含芯片和其他分离器件。

第六方面，本申请提供一种计算机程序，其特征在于，所述计算机程序包括指令，当所述计算机程序被计算机执行时，使得所述计算机执行上述第一方面中任意一项所述的方法。

附图说明

图 1 为本发明实施例提供的一种电子设备 100 的结构示意图。

图 2 为本发明实施例提供的一种跨引擎通信的系统架构示意图。

图 3 是本申请实施例中的一种引擎间的通信方法的流程示意图。

图 4 为本发明实施例提供的一种 H5 侧与 Ets 侧交互的示意图。

图 5 为本发明实施例提供的一种电子设备的界面示意图。

图 6 为本发明实施例提供的一种 H5 侧与 Ets 侧交互的流程示意图。

图 7 为本发明实施例提供的本申请提供的一种电子设备的示意图。

具体实施方式

下面将结合本申请实施例中的附图，对本申请实施例进行描述。

本申请的说明书和权利要求书及所述附图中的术语“第一”、“第二”、“第三”和“第四”等是用于区别不同对象，而不是用于描述特定顺序。此外，术语“包括”和“具有”以及它们任何变形，意图在于覆盖不排他的包含。例如包含了一系列步骤或单元的过程、方法、系统、产品或设备没有局限于已列出的步骤或单元，而是可选地还包括没有列出的步骤或单元，或可选地还包括对于这些过程、方法、产品或设备固有的其它步骤或单元。

在本文中提及“实施例”意味着，结合实施例描述的特定特征、结构或特性可以包含在本申请的至少一个实施例中。在说明书中的各个位置出现该短语并不一定均是指相同的实施例，也不是与其它实施例互斥的独立的或备选的实施例。本领域技术人员显式地和隐式地理解的是，本文所描述的实施例可以与其它实施例相结合。

图1示出了电子设备100的结构示意图。

下面以电子设备100为例对实施例进行具体说明。应该理解的是，电子设备100可以具有比图中所示的更多的或者更少的部件，可以组合两个或多个的部件，或者可以具有不同的部件配置。图中所示出的各种部件可以在包括一个或多个信号处理和/或专用集成电路在内的硬件、软件、或硬件和软件的组合中实现。

电子设备100可以包括：处理器110，外部存储器接口120，内部存储器121，通用串行总线（universal serial bus, USB）接口130，充电管理模块140，电源管理模块141，电池142，天线1，天线2，移动通信模块150，无线通信模块160，音频模块170，扬声器170A，受话器170B，麦克风170C，耳机接口170D，传感器模块180，按键190，马达191，指示器192，摄像头193，显示屏194，以及用户标识模块（subscriber identification module, SIM）卡接口195等。其中，传感器模块180可以包括压力传感器180A，陀螺仪传感器180B，气压传感器180C，磁传感器180D，加速度传感器180E，距离传感器180F，接近光传感器180G，指纹传感器180H，温度传感器180J，触摸传感器180K，环境光传感器180L，骨传导传感器180M等。

可以理解的是，本发明实施例示意的结构并不构成对电子设备100的具体限定。在本申请另一些实施例中，电子设备100可以包括比图示更多或更少的部件，或者组合某些部件，或者拆分某些部件，或者不同的部件布置。图示的部件可以以硬件，软件或软件和硬件的组合实现。

处理器110可以包括一个或多个处理单元，例如：处理器110可以包括应用处理器（application processor, AP），调制解调处理器，图形处理器（graphics processing unit, GPU），图像信号处理器（image signal processor, ISP），控制器，存储器，视频编解码器，数字信号处理器（digital signal processor, DSP），基带处理器，和/或神经网络处理器（neural-network processing unit, NPU）等。其中，不同的处理单元可以是独立的器件，也可以集成在一个或多个处理器中。

其中，控制器可以是电子设备100的神经中枢和指挥中心。控制器可以根据指令操作码和时序信号，产生操作控制信号，完成取指令和执行指令的控制。

处理器110中还可以设置存储器，用于存储指令和数据。在一些实施例中，处理器110中的存储器为高速缓冲存储器。该存储器可以保存处理器110刚用过或循环使用的指令或数据。如果处理器110需要再次使用该指令或数据，可从所述存储器中直接调用。避免了重复存取，减少了处理器110的等待时间，因而提高了系统的效率。

在一些实施例中，处理器110可以包括一个或多个接口。接口可以包括集成电路（inter-integrated circuit, I2C）接口，集成电路内置音频（inter-integrated circuit sound, I2S）接口，脉冲编码调制（pulse code modulation, PCM）接口，通用异步收发传输器（universal asynchronous receiver/transmitter, UART）接口，移动产业处理器接口（mobile industry processor interface, MIPI），通用输入输出（general-purpose input/output, GPIO）接口，用户标识模块（subscriber identity module, SIM）接口，和/或通用串行总线（universal serial bus, USB）接口等。

可以理解的是，本发明实施例示意的各模块间的接口连接关系，只是示意性说明，并不构成对电子设备100的结构限定。在本申请另一些实施例中，电子设备100也可以采用上述实施例中不同的接口连接方式，或多种接口连接方式的组合。

充电管理模块140用于从充电器接收充电输入。其中，充电器可以是无线充电器，也可以是有线充电器。

电源管理模块141用于连接电池142，充电管理模块140与处理器110。电源管理模块141接收电池142和/或充电管理模块140的输入，为处理器110，内部存储器121，外部存储器，显示屏194，摄像头193，和无线通信模块160等供电。

电子设备100的无线通信功能可以通过天线1，天线2，移动通信模块150，无线通信模块160，调制解调处理器以及基带处理器等实现。

电子设备100通过GPU，显示屏194，以及应用处理器等实现显示功能。GPU为图像处理的微处理器，连接显示屏194和应用处理器。GPU用于执行数学和几何计算，用于图形渲染。处理器110可包括一个或多个GPU，其执行程序指令以生成或改变显示信息。

显示屏 194 用于显示图像, 视频等。显示屏 194 包括显示面板。显示面板可以采用液晶显示屏 (liquid crystal display, LCD), 有机发光二极管 (organic light-emitting diode, OLED), 有源矩阵有机发光二极管或主动矩阵有机发光二极管 (active-matrix organic light emitting diode 的, AMOLED), 柔性发光二极管 (flex light-emitting diode, FLED), Miniled, MicroLed, Micro-oLed, 量子点发光二极管 (quantum dot light emitting diodes, QLED) 等。在一些实施例中, 电子设备 100 可以包括 1 个或 N 个显示屏 194, N 为大于 1 的正整数。

电子设备 100 可以通过 ISP, 摄像头 193, 视频编解码器, GPU, 显示屏 194 以及应用处理器等实现拍摄功能。

ISP 用于处理摄像头 193 反馈的数据。例如, 拍照时, 打开快门, 光线通过镜头被传递到摄像头感光元件上, 光信号转换为电信号, 摄像头感光元件将所述电信号传递给 ISP 处理, 转化为肉眼可见的图像。ISP 还可以对图像的噪点, 亮度, 肤色进行算法优化。ISP 还可以对拍摄场景的曝光, 色温等参数优化。在一些实施例中, ISP 可以设置在摄像头 193 中。

摄像头 193 用于捕获静态图像或视频。物体通过镜头生成光学图像投射到感光元件。感光元件可以是电荷耦合器件 (charge coupled device, CCD) 或互补金属氧化物半导体 (complementary metal-oxide-semiconductor, CMOS) 光电晶体管。感光元件把光信号转换成电信号, 之后将电信号传递给 ISP 转换成数字图像信号。ISP 将数字图像信号输出到 DSP 加工处理。DSP 将数字图像信号转换成标准的 RGB, YUV 等格式的图像信号。在一些实施例中, 电子设备 100 可以包括其他摄像头。电子设备还可以包括点阵发射器 (图中未示出), 用于发射光线。摄像头采集人脸反射的光线, 得到人脸图像, 处理器对人脸图像进行处理和分析, 通过与存储的人脸图像的信息进行比较以进行验证。

数字信号处理器用于处理数字信号, 除了可以处理数字图像信号, 还可以处理其他数字信号。例如, 当电子设备 100 在频点选择时, 数字信号处理器用于对频点能量进行傅里叶变换等。

视频编解码器用于对数字视频压缩或解压缩。电子设备 100 可以支持一种或多种视频编解码器。这样, 电子设备 100 可以播放或录制多种编码格式的视频, 例如: 动态图像专家组 (moving picture experts group, MPEG) 1, MPEG2, MPEG3, MPEG4 等。

NPU 为神经网络 (neural-network, NN) 计算处理器, 通过借鉴生物神经网络结构, 例如, 借鉴人脑神经元之间传递模式, 对输入信息快速处理, 还可以不断的自学习。通过 NPU 可以实现电子设备 100 的智能认知等应用, 例如: 图像识别, 人脸识别, 语音识别, 文本理解等。

外部存储器接口 120 可以用于连接外部存储卡, 例如 Micro SD 卡, 实现扩展电子设备 100 的存储能力。外部存储卡通过外部存储器接口 120 与处理器 110 通信, 实现数据存储功能。例如将音乐, 视频等文件保存在外部存储卡中。

内部存储器 121 可以用于存储计算机可执行程序代码, 所述可执行程序代码包括指令。处理器 110 通过运行存储在内部存储器 121 的指令, 从而执行电子设备 100 的各种功能应用以及数据处理。内部存储器 121 可以包括存储程序区和存储数据区。其中, 存储程序区可存储操作系统, 至少一个功能所需的应用 (比如人脸识别功能, 指纹识别功能、移动支付功能等)。存储数据区可存储电子设备 100 使用过程中所创建的数据 (比如人脸信息模板数据, 指纹信息模板等)。此外, 内部存储器 121 可以包括高速随机存取存储器, 还可以包括非易失性存储器, 例如至少一个磁盘存储器件, 闪存器件, 通用闪存存储器 (universal flash storage, UFS) 等。

电子设备 100 可以通过音频模块 170, 扬声器 170A, 受话器 170B, 麦克风 170C, 耳机接口 170D, 以及应用处理器等实现音频功能。例如音乐播放, 录音等。

音频模块 170 用于将数字音频信息转换成模拟音频信号输出, 也用于将模拟音频输入转换为数字音频信号。

扬声器 170A, 也称“喇叭”, 用于将音频电信号转换为声音信号。

受话器 170B, 也称“听筒”, 用于将音频电信号转换为声音信号。

麦克风 170C, 也称“话筒”, “传声器”, 用于将声音信号转换为电信号。

耳机接口 170D 用于连接有线耳机。耳机接口 170D 可以是 USB 接口 130, 也可以是 3.5mm 的开放移动电子设备平台 (open mobile terminal platform, OMTp) 标准接口, 美国蜂窝电信工业协会 (cellular telecommunications industry association of the USA, CTIA) 标准接口。

压力传感器 180A 用于感受压力信号, 可以将压力信号转换成电信号。在一些实施例中, 压力传感器 180A 可以设置于显示屏 194。压力传感器 180A 的种类很多, 如电阻式压力传感器, 电感式压力传感器,

电容式压力传感器等。

陀螺仪传感器 180B 可以用于确定电子设备 100 的运动姿态。在一些实施例中，可以通过陀螺仪传感器 180B 确定电子设备 100 围绕三个轴（即，x，y 和 z 轴）的角速度。

接近光传感器 180G 可以包括例如发光二极管（LED）和光检测器，例如光电二极管。发光二极管可以是红外发光二极管。

环境光传感器 180L 用于感知环境光亮度。电子设备 100 可以根据感知的环境光亮度自适应调节显示屏 194 亮度。环境光传感器 180L 也可用于拍照时自动调节白平衡。

指纹传感器 180H 用于采集指纹。电子设备 100 可以利用采集的指纹特性实现指纹解锁，访问应用锁，指纹拍照，指纹接听来电等。其中，该指纹传感器 180H 可以设置在触控屏下方，电子设备 100 可以接收用户在触控屏上该指纹传感器对应的区域的触摸操作，电子设备 100 可以响应于该触摸操作，采集用户手指的指纹信息。

温度传感器 180J 用于检测温度。在一些实施例中，电子设备 100 利用温度传感器 180J 检测的温度，执行温度处理策略。

触摸传感器 180K，也称“触控面板”。触摸传感器 180K 可以设置于显示屏 194，由触摸传感器 180K 与显示屏 194 组成触摸屏，也称“触控屏”。触摸传感器 180K 用于检测作用于其上或附近的触摸操作。触摸传感器可以将检测到的触摸操作传递给应用处理器，以确定触摸事件类型。可以通过显示屏 194 提供与触摸操作相关的视觉输出。在另一些实施例中，触摸传感器 180K 也可以设置于电子设备 100 的表面，与显示屏 194 所处的位置不同。

按键 190 包括开机键，音量键等。按键 190 可以是机械按键。也可以是触摸式按键。电子设备 100 可以接收按键输入，产生与电子设备 100 的用户设置以及功能控制有关的键信号输入。

指示器 192 可以是指示灯，可以用于指示充电状态，电量变化，也可以用于指示消息，未接来电，通知等。

SIM 卡接口 195 用于连接 SIM 卡。SIM 卡可以通过插入 SIM 卡接口 195，或从 SIM 卡接口 195 拔出，实现和电子设备 100 的接触和分离。在一些实施例中，电子设备 100 采用 eSIM，即：嵌入式 SIM 卡。eSIM 卡可以嵌在电子设备 100 中，不能和电子设备 100 分离。

下面结合附图对本申请的实施例进行描述。

基于上述提出的技术问题，也为了便于理解本发明实施例，下面先对本发明实施例所基于的系统架构进行描述。请参见图 2，图 2 是本发明实施例提供的一种跨引擎通信的系统架构示意图，该系统可以运行在上述电子设备 100 的软件系统中，可以解决不同 JS 引擎在运行时不能相互通讯的问题。该系统架构中可以包括方舟 JS 引擎模块（Ark JavaScript Engine）、方舟开发架构模块（ArkUI Framework）、H5 JS 引擎模块、JS 引擎桥接模块、JS 方法识别模块、容错模块。其中，Ark JS 引擎模块可以负责执行 JS 代码，解析生成抽象语法树（Abstract Syntax Tree，AST），转成字节码，并生成机器码等，并执行机器码且将执行结果传给 ArkUI 的 Framework 模块；ArkUI Framework 模块可以负责与 JS 引擎进行交互，页面生命周期管理，事件分发，加载 UI 组件，路由跳转、数据绑定等；H5 JS 引擎模块可以负责解析执行 H5 的 JS 代码，将执行的 JS 对象名和函数名、函数参数传给 JS 引擎桥接模块；JS 引擎桥接模块可以负责监测 Ark 引擎和 H5 的 JS 引擎的传递过来的对象名，方法名（也可以称为函数名），和函数参数，函数执行结果等消息，后续将详细描述如何基于 JS 引擎桥接模块实现不同引擎之间的通信，在此先不赘述；JS 方法识别模块可以负责从 JS 引擎桥接模块获取 H5 侧和 Ets 侧的方法名，判断 H5 侧的 JS 方法是否与 Ets 侧的方法相同；容错模块可以负责处理 H5 侧非法的输入，并在 Ets 实时监测方法是否存在，完成应用侧的容错保护，避免出现异常情况。

下面对本发明实施例所基于的具体方法架构进行描述。

请参见图 3，图 3 是本申请实施例中的一种引擎间的通信方法的流程示意图，下面将结合附图 3 并基于上述图 2 中的跨引擎通信的系统架构从交互侧对本申请实施例中的引擎间的通信方法进行描述。该方法应用于操作系统，所述操作系统包括第一引擎、第二引擎和桥接层，所述第一引擎包括多个第一对象，所述第二引擎包括多个第二对象，多个第一对象为第一应用中运行的对象，多个第二对象为第二应用中运行的对象，第一应用与第二应用为两个不同的应用，即第一应用和第二应用运行在两个不同的上下文中。该方法可以用于解决不同引擎在运行时不能相互通讯的问题。需要说明的是，为了更详细的描述本申请实施例中的引擎间的通信方法，本申请在各个流程步骤中描述了相应的执行主体分别电子设备（可以为上述图 1 中的电子设备 100），但不代表本申请实施例只能通过所描述的执行主体进行对应的方法流程。

步骤 S201: 电子设备通过所述桥接层接收所述第一引擎发送的第一请求。

具体的, 所述第一请求用于请求执行所述多个第一对象中的 N 个第一对象, N 为大于 0 的整数。第一引擎可以为上述提及的 H5 侧 H5 JS 引擎, H5 JS 引擎可以用于解析执行 H5 的 JS 代码。

步骤 S202: 电子设备通过所述桥接层响应于所述第一请求, 判断所述多个第二对象中是否存在与所述 N 个第一对象相同的 N 个第二对象。

具体的, 当桥接层接收第一引擎发送的第一请求后, 确定第一请求中请求执行的目标对象, 即 N 个第一对象, 然后桥接层判断第二引擎的多个第二对象中是否存在与目标对象相同的第二对象。第二引擎可以为上述提及的 Ets 侧的 Ark JS 引擎, Ark JS 引擎可以用于执行 JS 代码, 解析生成抽象语法树, 转成字节码, 并生成机器码等。

在一种可能的实现方式中, 所述判断所述多个第二对象中是否存在与所述 N 个第一对象相同的 N 个第二对象, 包括: 判断所述多个第二对象中是否存在与所述 N 个第一对象的对象标识相同的 N 个第二对象; 其中, 若存在则所述 N 个第二对象与所述 N 个第一对象相同。

具体的, 桥接层可以从第一引擎获取到多个第一对象的对象名 (也可以称为对象标识), 桥接层也可以从第二引擎获取到多个第二对象的对象名。进一步地, 当桥接层接收第一引擎发送的请求后, 可以判断多个第二对象中是否存在与请求中的 N 个第一对象的对象名相同的 N 个第二对象, 若存在, 则桥接层可以向第二引擎发送执行请求, 从而实现在运行过程中第一引擎可以通过桥接层调用第二引擎中的对象。由于在本申请中, 避免将第二引擎的对象直接在第一引擎中注册, 且通过桥接层高效实现两个不同引擎之间的信息交互, 因此在保证安全的前提下, 解决了两个运行环境相互隔离无法调用彼此对象 (或函数) 的问题。

在一种可能的实现方式中, 所述判断所述多个第二对象中是否存在与所述 N 个第一对象相同的 N 个第二对象, 包括: 判断所述多个第二对象中是否存在与所述 N 个第一对象的对象标识相同的 N 个第二对象; 若存在, 则通过第一记录表确定所述 N 个第一对象中每个第一对象包括的第一函数; 所述第一记录表包括所述多个第一对象中每个第一对象包括的所述第一函数; 基于第二记录表, 判断所述 N 个第一对象中每个第一对象包括的所述第一函数与所述 N 个第二对象中每个第二对象包括的第二函数是否一致; 其中, 若一致则所述多个第二对象中存在与所述 N 个第一对象相同的所述 N 个第二对象; 所述第二记录表包括所述第二对象中每个所述第二对象所包括的第二函数。

具体的, 桥接层不仅需要判断对象名是否相同, 还需要通过查表确定对象名下的函数是否相同, 若相同, 则桥接层可以向第二引擎发送执行请求, 从而实现在运行过程中第一引擎可以通过桥接层调用第二引擎中的对象。由于在本申请中, 避免将第二引擎的对象直接在第一引擎中注册, 且通过桥接层高效实现两个不同引擎之间的信息交互, 因此在保证安全的前提下, 解决了两个运行环境相互隔离无法调用彼此对象 (或函数) 的问题。

在一种可能实现方式中, 所述方法还包括: 通过所述桥接层获取所述多个第一对象中每个所述第一对象对应的对象标识, 以及每个所述第一对象包括的所述第一函数, 生成所述第一记录表; 通过所述桥接层获取所述多个第二对象中每个所述第二对象对应的对象标识, 以及每个所述第二对象包括的所述第二函数, 生成所述第二记录表。

具体的, 桥接层可以从第一引擎获取到多个第一对象的对象名 (也可以称为对象标识) 以及对象名下的函数名并生成第一记录表, 桥接层也可以从第二引擎获取到多个第二对象的对象名以及对象名下的函数名并生成第二记录表, 进而桥接层可以基于第一记录表和第二记录表判断请求中的对象名下的函数是否相同。由于在本申请中, 避免将第二引擎的对象直接在第一引擎中注册, 且通过桥接层高效实现两个不同引擎之间的信息交互, 因此在保证安全的前提下, 解决了两个运行环境相互隔离无法调用彼此对象 (或函数) 的问题。

例如, 如图 4 所示, 图 4 为本发明实施例提供的一种 H5 侧与 Ets 侧交互的示意图, 可以在 ArkUI (鸿蒙的应用开发框架) 的 JS 引擎侧 (即第二引擎, 可用于运行 Ets 侧的代码) 和 H5 的 JS 引擎 (即第一引擎, 可用于运行 H5 侧的代码) 侧构建两个映射表, 且可以在两个 JS 运行环境间构建一个桥接层用于传递消息, 将 H5 函数中的参数通过回调传给 Ark 引擎侧, 在 Ark 引擎侧执行 Ets 对象的 JS 方法 (也可以称为函数), 并通过桥接层将 Ark 引擎的执行结果回传给 H5 中的 JS 函数, 实现 H5 与 Ets 侧的消息的传递。

具体的, 在 JS 引擎桥接模块 (可以理解为上述桥接层) 解析 Ark 引擎 (即第二引擎) 上开发者自定义的对象名和方法名 (也可以称为函数名) 并注册, 以获得 Ark 引擎所包括的对象名以及每个对象名下的方法列表 (也可以理解为函数列表) 并记录在第二记录表中; 通过 JS 引擎桥接模块在 WebView 的 JS 引擎

(即第一引擎)注册上述方法(即函数);H5 页面通过 WebView 的 JS 引擎执行 H5 的方法;WebView 的 JS 引擎向 JS 引擎桥接模块发送请求判断方法是否存在;若 JS 引擎桥接模块基于上述第二记录表判断方法存在,则可以通知 WebView 的 JS 引擎方法存在;WebView 的 JS 引擎向 JS 引擎桥接模块发送执行请求并且可以携带上执行过程中所需的函数参数;JS 引擎桥接模块向 Ark 引擎发送执行方法的请求;Ark 引擎执行该方法并将结果返回给 WebView 内核,并通过 JS 引擎桥接模块转发给 WebView 的 JS 引擎;WebView 的 JS 引擎再将执行结果回传给 H5 页面。

步骤 S203: 若存在,电子设备通过所述桥接层向所述第二引擎发送第二请求。

具体的,所述第二请求用于请求所述第二引擎按照对应的函数参数执行所述 N 个第二对象。第二请求中可以包括执行 N 个第二请求过程中需要使用到的函数参数。

在一种可能的实现方式中,所述方法还包括:响应于所述第二请求,通过所述第二引擎按照对应的函数参数执行所述 N 个第二对象,并生成所述 N 个第二对象对应的执行结果;通过所述桥接层向所述第一引擎转发所述第二引擎发送的所述 N 个第二对象对应的所述执行结果。

具体的,第二引擎可以响应于桥接层的请求,可以基于函数参数执行对应的第二对象,并生成相应的执行结果,然后通过桥接层向第一引擎转发执行结果,从而实现跨引擎的数据交互。由于在本申请中,避免将第二引擎的对象直接在第一引擎中注册,且通过桥接层高效实现两个不同引擎之间的信息交互,因此在保证安全的前提下,解决了两个运行环境相互隔离无法调用彼此对象(或函数),以及无法进行数据传递的问题。

可选的,第二引擎响应于桥接层的第二请求,第二引擎可以再次判断是否存在对应的对象(或函数),若存在,则基于函数参数执行对应的对象(或函数);若不存在,第二引擎不会执行对应的对象(或函数),并通知第一引擎执行失败,返回异常日志信息,从而完成应用侧的容错保护。

例如,如图 5 所示,图 5 为本发明实施例提供的一种电子设备的界面示意图,图 5 的(a)中电子设备上显示第一界面,第一界面可以为浏览器当前打开的页面,用户可以在此界面上实现实时计算,如用户输入数据 1 和 2,此时第一引擎会向桥接层发送一个调用加法运算(可以理解为一个函数)的请求。进一步地,桥接层会判断第二引擎中是否存在该函数,若存在,则桥接层将函数参数(如 1 和 2)发送给第二引擎,第二引擎基于函数参数执行对应的函数,并将执行结果通过桥接层发送给第一引擎。最终,如图 5 的(b)所示可以在电子设备上直接显示执行结果。本申请提供的引擎间的通信方法,不仅能解决两个运行环境相互隔离无法调用彼此对象(或函数)的问题,在 H5 与 Ets 频繁交互的场景下,能够降低响应时延,即点即响应。

在一种可能的实现方式中,所述方法还包括:若所述多个第二对象中不存在与所述 N 个第一对象相同的所述 N 个第二对象,则通过所述桥接层向所述第一引擎发送第一通知,所述第一通知包括所述 N 个第一对象执行错误的消息。

具体的,若不存在相同的对象,则表示第二引擎无法执行请求中的对象,进而可以通过桥接层通知第一引擎请求中的对象不会被执行。由于在本申请中,避免将第二引擎的对象直接在第一引擎中注册,且通过桥接层高效实现两个不同引擎之间的信息交互,因此在保证安全的前提下,解决了两个运行环境相互隔离无法调用彼此对象(或函数),以及无法进行数据传递的问题。

接下来,以第一引擎为 H5 的 JS 引擎,第二引擎为 Ets 侧的 Ark 引擎为例进行说明,如图 6 所示,图 6 为本发明实施例提供的一种 H5 侧与 Ets 侧交互的流程示意图,图中结合上述图 2 中的跨引擎通信的系统架构从交互侧对本申请实施例中的引擎间的通信方法进行描述,需要说明的是,在本申请中可以实现一种映射表机制,完成 ArkUI 侧只执行 Ets 侧的方法,H5 侧执行 H5 中的 JS 方法,通过桥接层,将 H5 的对象传给 ArkUI 侧,ArkUI 框架将该对象转成 Ark 引擎的对象(JsValue 对象),并执行开发者在 Ets 侧定义的对象下的 JS 方法;在本申请中也设计了一种 H5 JS 引擎与 ArkUI JS 引擎交互的桥接机制,负责将 Ark 引擎的对象和方法名传给 H5 的 JS 引擎,识别到该方法存在后,JS 引擎桥接模块将函数参数传给 Ark 引擎,Ark 引擎根据函数参数执行 JS 方法后,再将执行结果返回给 H5 的 JS 引擎,完成两个不同 JS 引擎在相互隔离的 JS 运行环境下的通讯;本申请还设计了一种 H5 与 Ark 引擎通讯的容错保护机制,通过在两个不同 JS 引擎侧分别判断执行的 JS 方法是否存在,若不存在,Ark 的 JS 引擎不会执行该方法,并通知应用侧该方法不存在,返回异常日志信息,完成应用侧的容错保护。详细步骤如下:

步骤 1: 开启应用,Ark JS 引擎模块执行应用开发者的 JS 代码,解析开发者自定义的对象名和方法名,同时 JS 对象和方法名等通过 JS 到 C++ 的跨语言接口调用传给 ArkUI Framework 模块。

步骤 2: ArkUI Framework 模块根据对象名和方法名等信息创建 C++ 对象,并在对象上绑定对应的 C++

的方法，并向 JS 引擎桥接模块注册 C++ 函数，并将对象名和对象名下的方法列表保存到一个哈希表中，发送给 JS 方法识别模块。

步骤 3：JS 方法识别模块智能识别 ArkUI 中 JS 引擎中的 JS 方法是否存在，并通知容错模块。

步骤 4：在 ArkUI JS 引擎模块解析 Ets 侧的对象和方法时，H5 的 JS 模块在应用启动并执行 H5 的 JS 代码时，H5 的 JS 引擎模块也会解析 H5 侧使用的 JS 对象和方法名，通过 JS 引擎桥接模块来判断对象下的方法是否存在来通知容错模块。

步骤 5：容错模块发送消息给 Ark JS 引擎模块，如果识别到对应的方法存在，执行对应的 JS 方法，如果方法不存在，通过 JS 回调到应用侧，可以通过弹框通知用户执行的方法无效。

步骤 6：ArkUI JS 引擎模块执行 JS 方法，并将 JS 执行结果返回给 JS 引擎桥接模块，然后 JS 引擎桥接模块将通过注册的回调将 JS 函数执行结果传给 H5 的 JS 引擎；

步骤 7：H5 的 JS 引擎模块将执行结果回传给 H5 页面，通过数据绑定触发页面更新，将执行结果更新的 H5 界面上，并最终通知到用户。

综上，本发明实施例中，可以在第一引擎和第二引擎之间增加一个桥接层，以通过桥接层实现不同引擎之间的通信。具体的，可以在第一引擎上运行第一应用，第一应用中可以包括多个第一对象；可以在第二引擎上运行第二应用，第二应用中可以包括多个第二对象，且第一应用和第二应用运行在两个不同的上下文中。进一步地，在两个引擎运行过程中，第一引擎可以向桥接层发送执行一个或多个第一对象的请求，然后桥接层判断多个第二对象中是否存在与上述一个或多个第一对象相同的第二对象，若存在，桥接层再向第二引擎发送执行对应的一个或多个第二对象的请求，且在该请求中可以包括执行过程中需要使用的函数参数，从而第二引擎可以基于函数参数执行对应的第二对象，实现了跨引擎通信。由于在本申请中，避免将第二引擎的对象直接在第一引擎中注册，且通过桥接层高效实现两个不同引擎之间的信息交互，因此在保证安全的前提下，解决了两个运行环境相互隔离无法调用彼此对象（或函数）的问题。

上述详细阐述了本发明实施例的方法，下面提供了本发明实施例的相关设备。

请参见图 7，图 7 是本发明实施例提供的本申请提供了一种电子设备的示意图，该电子设备 30 运行操作系统，所述操作系统包括第一引擎、第二引擎和桥接层，所述第一引擎包括多个第一对象，所述第二引擎包括多个第二对象。该电子设备 30 可以包括第一接收单元 301、第一处理单元 302、第一发送单元 303、第二接收单元 304、第三接收单元 305、第二处理单元 306、第二发送单元 307、第三发送单元 308，其中，各个模块的详细描述如下。

第一接收单元 301，用于通过所述桥接层接收所述第一引擎发送的第一请求，所述第一请求用于请求执行所述多个第一对象中的 N 个第一对象，N 为大于 0 的整数；

第一处理单元 302，用于通过所述桥接层响应于所述第一请求，判断所述多个第二对象中是否存在与所述 N 个第一对象相同的 N 个第二对象；

第一发送单元 303，用于，若存在，通过所述桥接层向所述第二引擎发送第二请求，所述第二请求用于请求所述第二引擎按照对应的函数参数执行所述 N 个第二对象。

在一种可能的实现方式中，所述第一处理单元 302，具体用于：判断所述多个第二对象中是否存在与所述 N 个第一对象的对象标识相同的 N 个第二对象；其中，若存在则所述 N 个第二对象与所述 N 个第一对象相同。

在一种可能的实现方式中，所述第一处理单元 302，具体用于：判断所述多个第二对象中是否存在与所述 N 个第一对象的对象标识相同的 N 个第二对象；若存在，则通过第一记录表确定所述 N 个第一对象中每个第一对象包括的第一函数；所述第一记录表包括所述多个第一对象中每个第一对象包括的所述第一函数；基于第二记录表，判断所述 N 个第一对象中每个第一对象包括的所述第一函数与所述 N 个第二对象中每个第二对象包括的第二函数是否一致；其中，若一致则所述多个第二对象中存在与所述 N 个第一对象相同的所述 N 个第二对象；所述第二记录表包括所述第二对象中每个所述第二对象所包括的第二函数。

在一种可能的实现方式中，所述设备还包括：第二接收单元 304，用于通过所述桥接层获取所述多个第一对象中每个所述第一对象对应的对象标识，以及每个所述第一对象包括的所述第一函数，生成所述第一记录表；第三接收单元 305，用于通过所述桥接层获取所述多个第二对象中每个所述第二对象对应的对象标识，以及每个所述第二对象包括的所述第二函数，生成所述第二记录表。

在一种可能的实现方式中，所述设备还包括：第二处理单元 306，用于响应于所述第二请求，通过所述第二引擎按照对应的函数参数执行所述 N 个第二对象，并生成所述 N 个第二对象对应的执行结果；第二发送单元 307，用于通过所述桥接层向所述第一引擎转发所述第二引擎发送的所述 N 个第二对象对应的

所述执行结果。

在一种可能的实现方式中，所述设备还包括：第三发送单元 308，用于若所述多个第二对象中不存在与所述 N 个第一对象相同的所述 N 个第二对象，则通过所述桥接层向所述第一引擎发送第一通知，所述第一通知包括所述 N 个第一对象执行错误的消息。

需要说明的是，本发明实施例中所描述的电子设备 30 中各功能单元的功能可参见上述图 3 中所述的方法实施例中的步骤 S201-步骤 S203 的相关描述，此处不再赘述。

本申请提供了一种计算机存储介质，其特征在于，所述计算机存储介质存储有计算机程序，该计算机程序被处理器执行时实现上述任意一种引擎间的通信方法。

本申请实施例提供一种电子设备，该电子设备中包括处理器，处理器被配置为支持该电子设备实现上述任意一种引擎间的通信方法中相应的功能。该电子设备还可以包括存储器，存储器用于与处理器耦合，其保存该电子设备必要的程序指令和数据。该电子设备还可以包括通信接口，用于该电子设备与其他设备或通信网络通信。

本申请提供了一种芯片系统，该芯片系统包括处理器，用于支持电子设备实现上述所涉及的功能，例如，生成或处理上述一种引擎间的通信方法中所涉及的信息。在一种可能的设计中，所述芯片系统还包括存储器，所述存储器，用于保存电子设备必要的程序指令和数据。该芯片系统，可以由芯片构成，也可以包含芯片和其他分离器件。

本申请提供一种计算机程序，其特征在于，所述计算机程序包括指令，当所述计算机程序被计算机执行时，使得所述计算机执行上述一种引擎间的通信方法。

在上述实施例中，对各个实施例的描述都各有侧重，某个实施例中未详述的部分，可以参见其他实施例的相关描述。

需要说明的是，对于前述的各方法实施例，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，本申请并不受所描述的动作顺序的限制，因为依据本申请，某些步骤可能可以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，所涉及的动作和模块并不一定是本申请所必须的。

在本申请所提供的几个实施例中，应该理解到，所揭露的装置，可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，例如上述单元的划分，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，可以是电性或其它的形式。

上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。

另外，在本申请各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。上述集成的单元既可以采用硬件的形式实现，也可以采用软件功能单元的形式实现。

上述集成的单元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存储介质中。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或者该技术方案的全部或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（可以为个人计算机、服务器或者网络设备等，具体可以是计算机设备中的处理器）执行本申请各个实施例上述方法的全部或部分步骤。其中，而前述的存储介质可包括：U 盘、移动硬盘、磁碟、光盘、只读存储器（Read-Only Memory，缩写：ROM）或者随机存取存储器（Random Access Memory，缩写：RAM）等各种可以存储程序代码的介质。

以上所述，以上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。

权 利 要 求 书

1.一种引擎间的通信方法，其特征在于，应用于操作系统，所述操作系统包括第一引擎、第二引擎和桥接层，所述第一引擎包括多个第一对象，所述第二引擎包括多个第二对象，所述方法包括：

通过所述桥接层接收所述第一引擎发送的第一请求，所述第一请求用于请求执行所述多个第一对象中的N个第一对象，N为大于0的整数；

通过所述桥接层响应于所述第一请求，判断所述多个第二对象中是否存在与所述N个第一对象相同的N个第二对象；

若存在，通过所述桥接层向所述第二引擎发送第二请求，所述第二请求用于请求所述第二引擎按照对应的函数参数执行所述N个第二对象。

2.如权利要求1所述的方法，其特征在于，所述判断所述多个第二对象中是否存在与所述N个第一对象相同的N个第二对象，包括：

判断所述多个第二对象中是否存在与所述N个第一对象的对象标识相同的N个第二对象；其中，若存在则所述N个第二对象与所述N个第一对象相同。

3.如权利要求1所述的方法，其特征在于，所述判断所述多个第二对象中是否存在与所述N个第一对象相同的N个第二对象，包括：

判断所述多个第二对象中是否存在与所述N个第一对象的对象标识相同的N个第二对象；

若存在，则通过第一记录表确定所述N个第一对象中每个第一对象包括的第一函数；所述第一记录表包括所述多个第一对象中每个第一对象包括的所述第一函数；

基于第二记录表，判断所述N个第一对象中每个第一对象包括的所述第一函数与所述N个第二对象中每个第二对象包括的第二函数是否一致；其中，若一致则所述多个第二对象中存在与所述N个第一对象相同的所述N个第二对象；所述第二记录表包括所述第二对象中每个所述第二对象所包括的第二函数。

4.如权利要求3所述的方法，其特征在于，所述方法还包括：

通过所述桥接层获取所述多个第一对象中每个所述第一对象对应的对象标识，以及每个所述第一对象包括的所述第一函数，生成所述第一记录表；

通过所述桥接层获取所述多个第二对象中每个所述第二对象对应的对象标识，以及每个所述第二对象包括的所述第二函数，生成所述第二记录表。

5.如权利要求1-4任意一项所述的方法，其特征在于，所述方法还包括：

响应于所述第二请求，通过所述第二引擎按照对应的函数参数执行所述N个第二对象，并生成所述N个第二对象对应的执行结果；

通过所述桥接层向所述第一引擎转发所述第二引擎发送的所述N个第二对象对应的所述执行结果。

6.如权利要求1-5任意一项所述的方法，其特征在于，所述方法还包括：

若所述多个第二对象中不存在与所述N个第一对象相同的所述N个第二对象，则通过所述桥接层向所述第一引擎发送第一通知，所述第一通知包括所述N个第一对象执行错误的消息。

7.一种电子设备，其特征在于，所述电子设备运行操作系统，所述操作系统包括第一引擎、第二引擎和桥接层，所述第一引擎包括多个第一对象，所述第二引擎包括多个第二对象，所述装置包括：

第一接收单元，用于通过所述桥接层接收所述第一引擎发送的第一请求，所述第一请求用于请求执行所述多个第一对象中的N个第一对象，N为大于0的整数；

第一处理单元，用于通过所述桥接层响应于所述第一请求，判断所述多个第二对象中是否存在与所述N个第一对象相同的N个第二对象；

第一发送单元，用于，若存在，通过所述桥接层向所述第二引擎发送第二请求，所述第二请求用于请求所述第二引擎按照对应的函数参数执行所述N个第二对象。

8.如权利要求7所述的设备，其特征在于，所述第一处理单元，具体用于：

判断所述多个第二对象中是否存在与所述N个第一对象的对象标识相同的N个第二对象；其中，若存在则所述N个第二对象与所述N个第一对象相同。

9.如权利要求7所述的设备，其特征在于，所述第一处理单元，具体用于：

判断所述多个第二对象中是否存在与所述N个第一对象的对象标识相同的N个第二对象；

若存在，则通过第一记录表确定所述N个第一对象中每个第一对象包括的第一函数；所述第一记录表包括所述多个第一对象中每个第一对象包括的所述第一函数；

基于第二记录表，判断所述N个第一对象中每个第一对象包括的所述第一函数与所述N个第二对象中每个第二对象包括的第二函数是否一致；其中，若一致则所述多个第二对象中存在与所述N个第一对象相同的所述N个第二对象；所述第二记录表包括所述第二对象中每个所述第二对象所包括的第二函数。

10.如权利要求9所述的设备，其特征在于，所述设备还包括：

第二接收单元，用于通过所述桥接层获取所述多个第一对象中每个所述第一对象对应的对象标识，以及每个所述第一对象包括的所述第一函数，生成所述第一记录表；

第三接收单元，用于通过所述桥接层获取所述多个第二对象中每个所述第二对象对应的对象标识，以及每个所述第二对象包括的所述第二函数，生成所述第二记录表。

11.如权利要求7-10任意一项所述的设备，其特征在于，所述设备还包括：

第二处理单元，用于响应于所述第二请求，通过所述第二引擎按照对应的函数参数执行所述N个第二对象，并生成所述N个第二对象对应的执行结果；

第二发送单元，用于通过所述桥接层向所述第一引擎转发所述第二引擎发送的所述N个第二对象对应的所述执行结果。

12.如权利要求7-11任意一项所述的设备，其特征在于，所述设备还包括：

第三发送单元，用于若所述多个第二对象中不存在与所述N个第一对象相同的所述N个第二对象，则通过所述桥接层向所述第一引擎发送第一通知，所述第一通知包括所述N个第一对象执行错误的消息。

13.一种计算机存储介质，其特征在于，所述计算机存储介质存储有计算机程序，该计算机程序被处理器执行时实现上述权利要求1-6中任意一项所述的方法。

14.一种计算机程序，其特征在于，所述计算机程序包括指令，当所述计算机程序被计算机执行时，使得所述计算机执行如权利要求1-6中任意一项所述的方法。



图 1

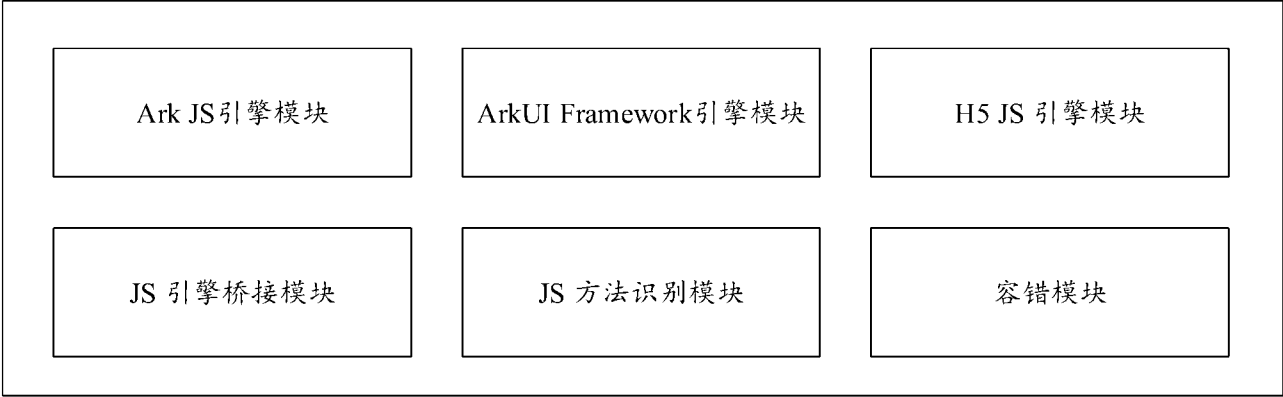


图 2

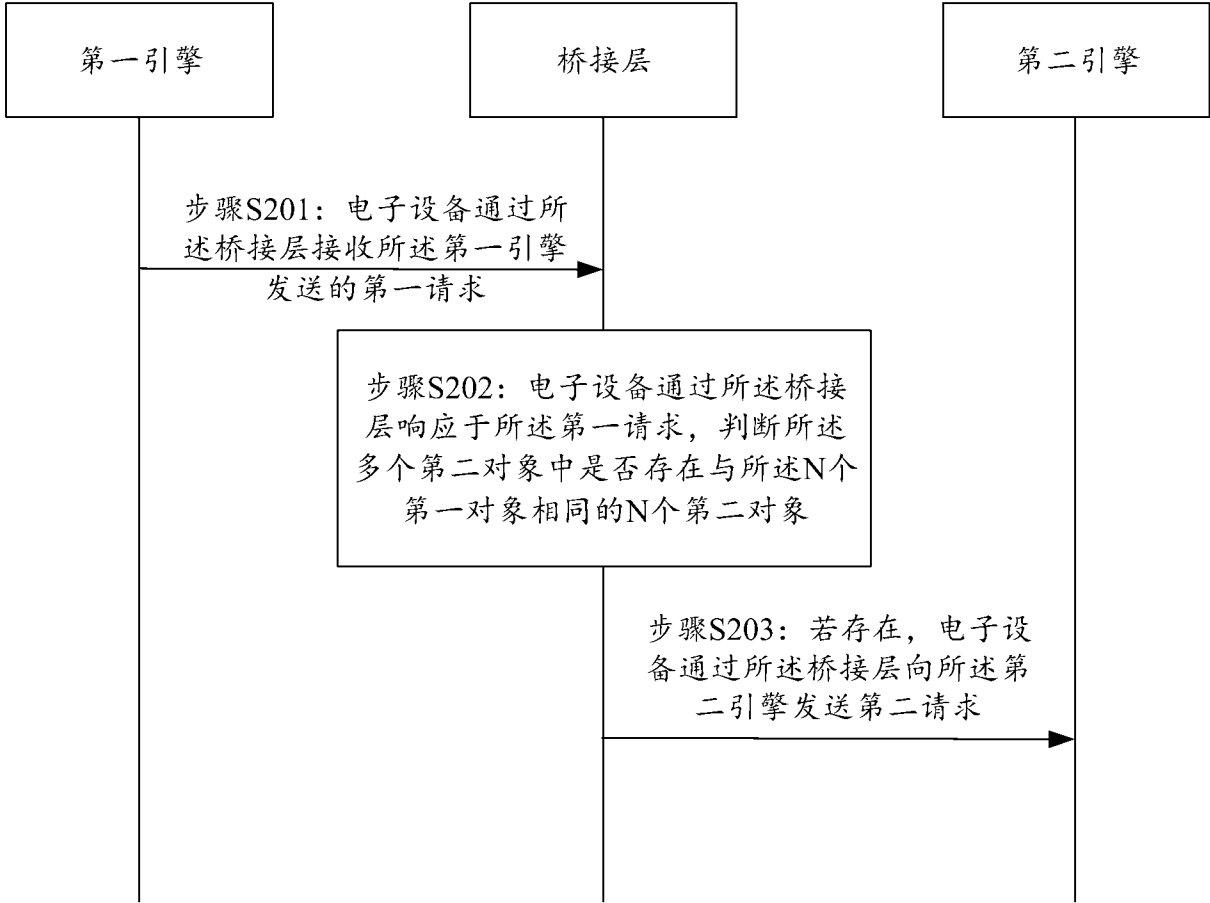


图 3

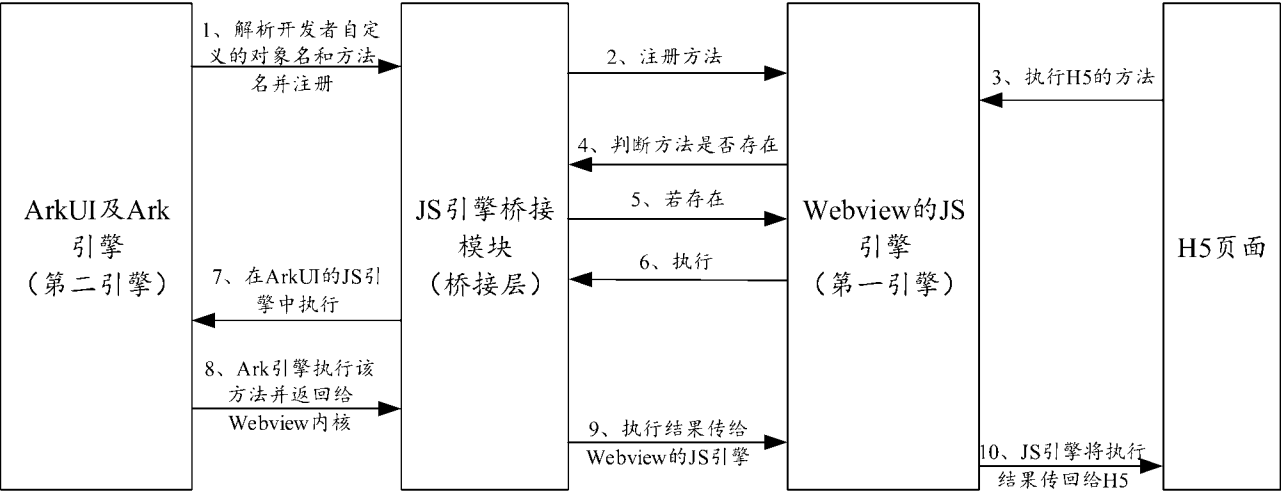


图 4

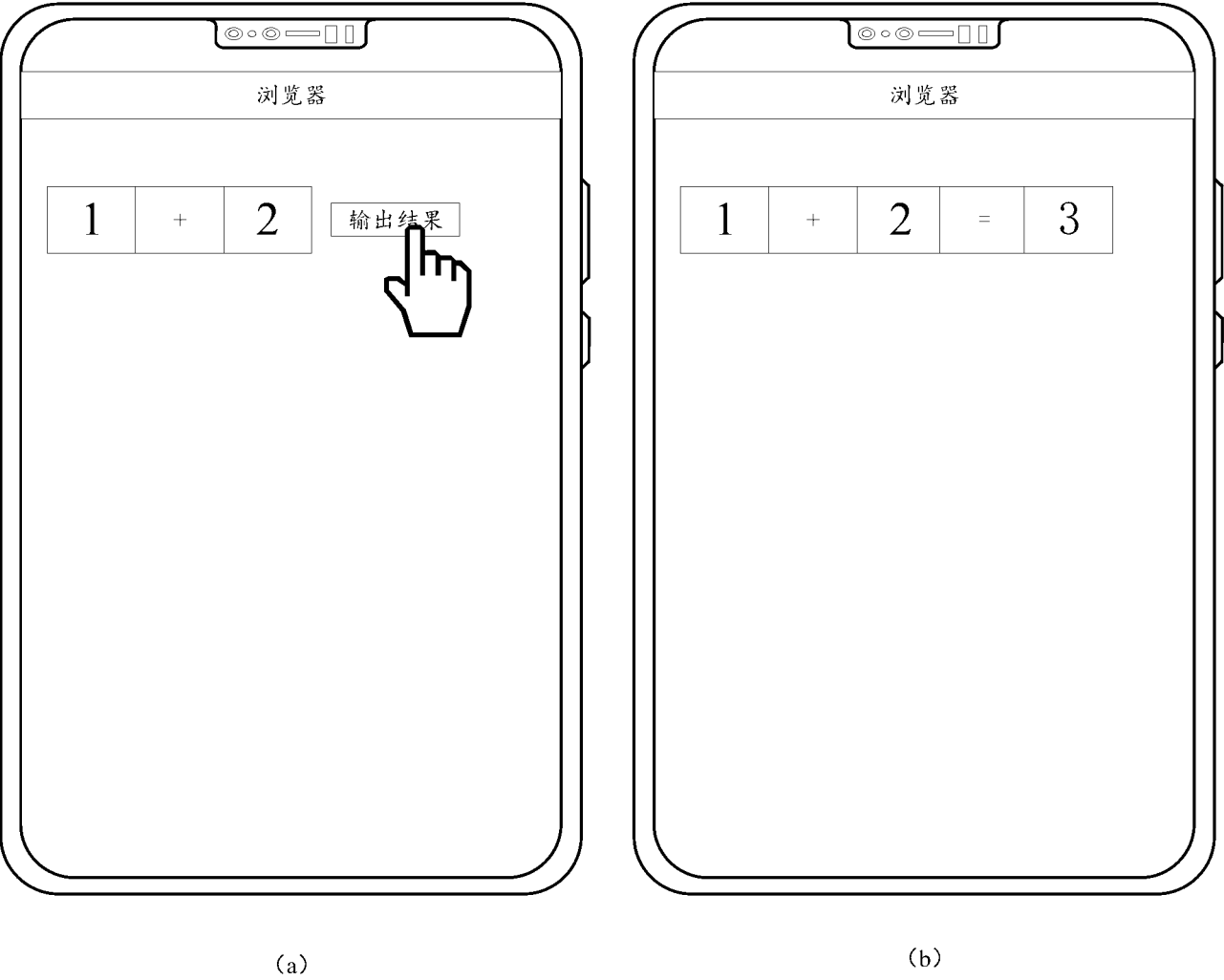


图 5

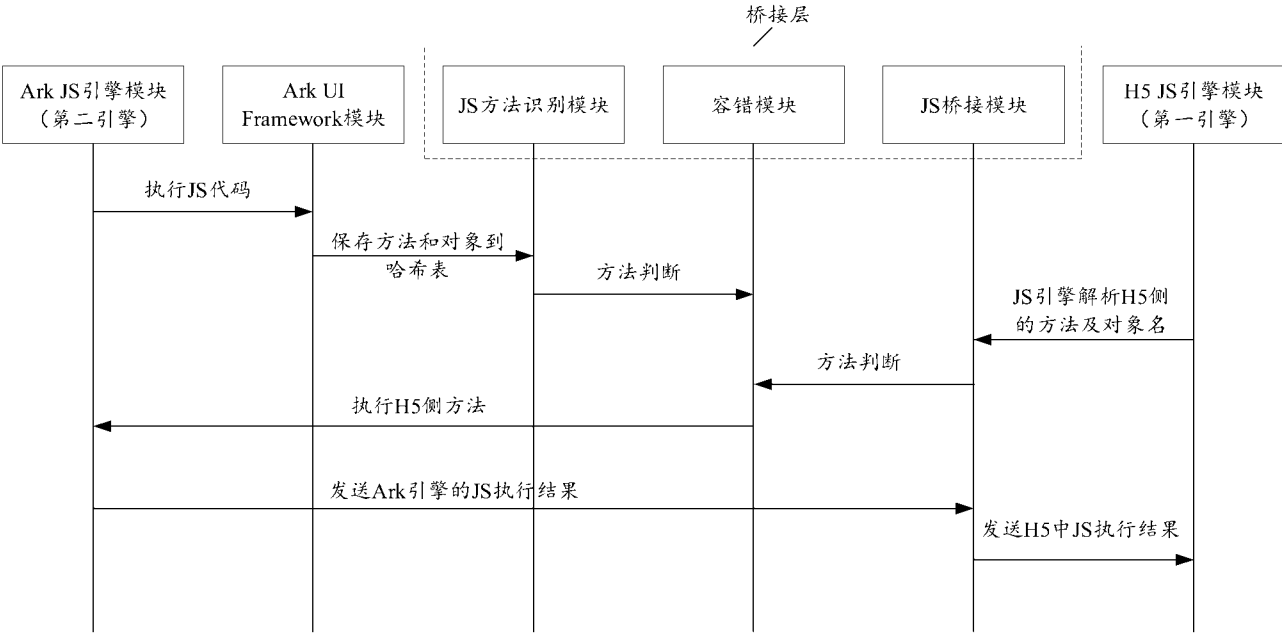


图 6

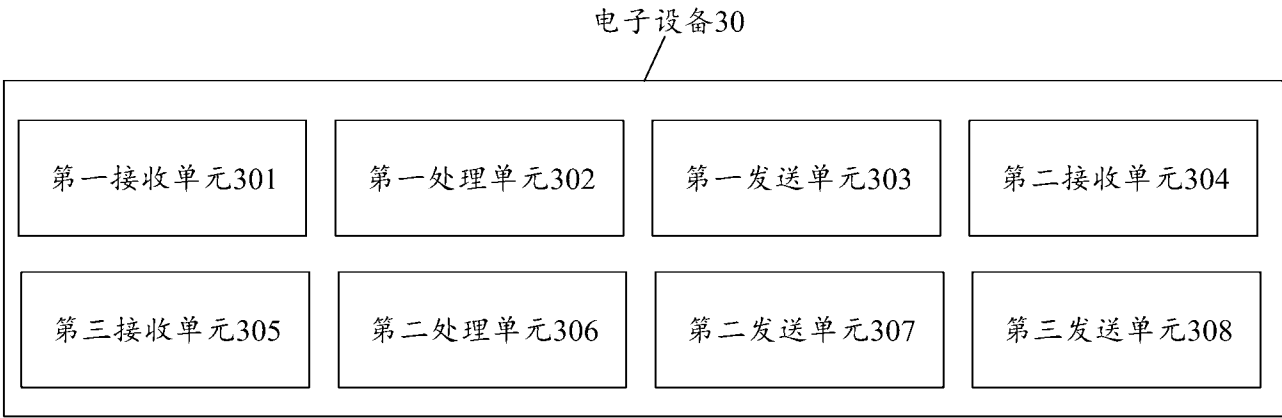


图 7

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2023/117917

A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/54(2006.01)i; G06F15/163(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)

IPC: 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)

VEN, CNABS, CNTXT, ENTXTC, CNKI, IEEE: 引擎, 桥接, 对象, 函数, 标识, 相同, Bridge, Engine, Identification, Object, Function, Same

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	CN 110688601 A (ZHENGZHOU APUS TECHNOLOGY CO., LTD.) 14 January 2020 (2020-01-14) description, paragraphs [30]-[122]	1-4
Y	US 8490117 B1 (ADOBE SYSTEMS INC.) 16 July 2013 (2013-07-16) description, columns 4-22	1-14
Y	CN 113791790 A (58 CO., LTD.) 14 December 2021 (2021-12-14) description, paragraphs [45]-[114]	1-14
A	CN 110727504 A (BAIDU ONLINE NETWORK TECHNOLOGY (BEIJING) CO., LTD.) 24 January 2020 (2020-01-24) entire document	1-14
A	US 2022035799 A1 (SAP SE) 03 February 2022 (2022-02-03) entire document	1-14



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

“D” document cited by the applicant in the international application

“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

09 November 2023

Date of mailing of the international search report

20 November 2023

Name and mailing address of the ISA/CN

China National Intellectual Property Administration (ISA/
CN)
China No. 6, Xitucheng Road, Jimenqiao, Haidian District,
Beijing 100088

Authorized officer

Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2023/117917

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
CN	110688601	A	14 January 2020	None			
US	8490117	B1	16 July 2013	US	2008098296	A1	24 April 2008
				US	7614003	B2	03 November 2009
CN	113791790	A	14 December 2021	None			
CN	110727504	A	24 January 2020	JP	2021068415	A	30 April 2021
				JP	7048663	B2	05 April 2022
				US	2021117169	A1	22 April 2021
				US	11294651	B2	05 April 2022
US	2022035799	A1	03 February 2022	US	11403286	B2	02 August 2022
				EP	3945431	A1	02 February 2022

A. 主题的分类 G06F 9/54(2006.01)i; G06F15/163(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) IPC: 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)) VEN, CNABS, CNTXT, ENTXTC, CNKI, IEEE: 引擎, 桥接, 对象, 函数, 标识, 相同, Bridge, Engine, Identification, Object, Function, Same		
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
Y	CN 110688601 A (郑州阿帕斯科技有限公司) 2020年1月14日 (2020 - 01 - 14) 说明书第[30]-[122]段	1-4
Y	US 8490117 B1 (ADOBE SYSTEMS INCORPORATED) 2013年7月16日 (2013 - 07 - 16) 说明书第4-22栏	1-14
Y	CN 113791790 A (五八有限公司) 2021年12月14日 (2021 - 12 - 14) 说明书第[45]-[114]段	1-14
A	CN 110727504 A (百度在线网络技术(北京)有限公司) 2020年1月24日 (2020 - 01 - 24) 全文	1-14
A	US 2022035799 A1 (SAP SE) 2022年2月3日 (2022 - 02 - 03) 全文	1-14
☐ 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 “D” 申请人在国际申请中引证的文件 “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 2023年11月9日		国际检索报告邮寄日期 2023年11月20日
ISA/CN的名称和邮寄地址 中国国家知识产权局 中国北京市海淀区蓟门桥西土城路6号 100088		授权官员 苏玉兰 电话号码 (+86) 010-53961547

国际检索报告
关于同族专利的信息

国际申请号
PCT/CN2023/117917

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	110688601	A	2020年1月14日	无			
US	8490117	B1	2013年7月16日	US	2008098296	A1	2008年4月24日
				US	7614003	B2	2009年11月3日
CN	113791790	A	2021年12月14日	无			
CN	110727504	A	2020年1月24日	JP	2021068415	A	2021年4月30日
				JP	7048663	B2	2022年4月5日
				US	2021117169	A1	2021年4月22日
				US	11294651	B2	2022年4月5日
US	2022035799	A1	2022年2月3日	US	11403286	B2	2022年8月2日
				EP	3945431	A1	2022年2月2日