

(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 1 月 10 日 (10.01.2019)



(10) 国际公布号
WO 2019/007216 A1

(51) 国际专利分类号:
G06F 9/44 (2018.01)

(21) 国际申请号: PCT/CN2018/092625

(22) 国际申请日: 2018 年 6 月 25 日 (25.06.2018)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
201710551288.5 2017 年 7 月 7 日 (07.07.2017) CN

(71) 申请人: 阿里巴巴集团控股有限公司 (ALIBABA GROUP HOLDING LIMITED) [—/CN]; 开曼群岛大开曼资本大厦一座四层 847 号邮箱, Grand Cayman (KY)。

(72) 发明人: 金德才 (JIN, Decai); 中国浙江省杭州市余杭区文一西路 969 号 3 号楼 5 楼阿里巴巴集团法务部, Zhejiang 311121 (CN)。金卓军 (JIN, Zhuojun); 中国浙江省杭州市余杭区文一西路 969 号 3 号楼 5 楼阿里巴巴集团法务部, Zhejiang 311121 (CN)。

(74) 代理人: 北京三友知识产权代理有限公司 (BEIJING SANYOU INTELLECTUAL PROPERTY AGENCY LTD.); 中国北京市金融街 35 号国际企业大厦 A 座 16 层, Beijing 100033 (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,

(54) Title: GRAPHICS LAYER PROCESSING METHOD AND DEVICE

(54) 发明名称: 一种图层处理方法及装置

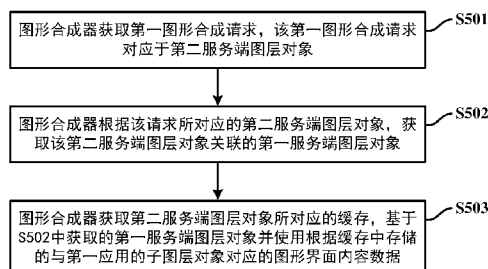


图 5

- S501 A GRAPHICS SYNTHESIZER OBTAINS A FIRST GRAPHICS SYNTHESIS REQUEST THAT CORRESPONDS TO A SECOND SERVER GRAPHICS LAYER OBJECT
- S502 THE GRAPHICS SYNTHESIZER OBTAINS A FIRST SERVER GRAPHICS LAYER OBJECT ASSOCIATED WITH THE SECOND SERVER GRAPHIC LAYER OBJECT ACCORDING TO THE SECOND SERVER GRAPHICS LAYER OBJECT CORRESPONDING TO THE REQUEST
- S503 THE GRAPHICS SYNTHESIZER OBTAINS A CACHE CORRESPONDING TO THE SECOND SERVER GRAPHICS LAYER OBJECT, AND RESPONDS TO THE FIRST GRAPHICS SYNTHESIS REQUEST ON THE BASIS OF THE FIRST SERVER GRAPHICS LAYER OBJECT OBTAINED IN S502 AND BY USING GRAPHICS INTERFACE CONTENT DATA CORRESPONDING TO THE SUB-GRAPHICS LAYER OBJECTS OF A FIRST APPLICATION STORED IN THE CACHE

(57) Abstract: Disclosed are a graphics layer processing method and device. In the present application, a graphics synthesizer obtains a first graphics synthesis request that corresponds to a second server graphics layer object in a second subsystem, and obtains a first server graphics layer object in a first subsystem associated with the second server graphics layer object according to the second server graphics layer object corresponding to the first graphics synthesis request, wherein the first server graphics layer object and the second server graphics layer object are created for sub-graphics layer objects of a first application; the graphics synthesizer obtains a cache

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, 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))。

corresponding to the second server graphics layer object, and responds to the first graphics synthesis request according to graphics interface content data corresponding to the sub-graphics layer objects stored in the cache. The present application can be used for integrating applications or services based on different operating systems in graphics synthesis level within an operating system framework.

(57) 摘要: 本申请公开了一种图层处理方法及装置。本申请中, 图形合成器获取第一图形合成请求, 所述第一图形合成请求对应于第二子系统内的第二服务端图层对象, 根据所述第一图形合成请求所对应的第二服务端图层对象, 获取所述第二服务端图层对象所关联的第一子系统内的第一服务端图层对象, 其中, 所述第一服务端图层对象和所述第二服务端图层对象是针对第一应用的子图层对象创建的; 图形合成器获取所述第二服务端图层对象所对应的缓存, 并根据所述缓存中存储的与所述子图层对象对应的图形界面内容数据, 响应第一图形合成请求。采用本申请可针对在一种操作系统框架内, 基于不同操作系统的应用或服务如何在图形合成层面融合。

一种图层处理方法及装置

本申请要求 2017 年 07 月 07 日递交的申请号为 201710551288.5、发明名称为“一种图层处理方法及装置”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。

5 技术领域

本申请涉及计算机技术领域，尤其涉及一种图层处理方法及装置。

背景技术

云操作系统又称云 OS，是以云计算、云存储技术作为支撑的操作系统。云操作系统
10 因其能够整合海量的计算能力并能够提供强大的资源调配能力，未来将得到更广泛的应用。

云操作系统既可以用于传统的桌面系统设备（如 PC）又适用于移动设备。基于云操作系统的应用或服务，以及基于传统操作系统的应用或服务，将会在云操作系统框架内运行，这就要求云操作系统框架能够将基于不同操作系统的应用或服务进行融合。

15 目前，针对在一种操作系统框架内，基于不同操作系统的应用或服务如何在图形合成层面融合，目前尚未有相应的解决方案。

发明内容

本申请实施例公开了一种图层处理方法及装置。

20 第一方面，提供一种图层处理方法，包括：

图形合成器获取第一图形合成请求，所述第一图形合成请求对应于第二子系统
中的第二服务端图层对象；

所述图形合成器根据所述第一图形合成请求所对应的第二服务端图层对象，获取所述
第二服务端图层对象所关联的第一子系统的第一服务端图层对象；其中，所述第一
25 服务端图层对象和所述第二服务端图层对象是针对第一应用的子图层对象创建的；

所述图形合成器获取所述第二服务端图层对象所对应的缓存，基于所述第一服务端
图层对象并使用所述缓存中存储的与所述第一应用的子图层对象对应的图形界面内容数
据，响应所述第一图形合成请求。

可选地，所述获取所述第二服务端图层对象所关联的第一服务端图层对象，包括：

30 所述图形合成器根据所述第二服务端图层对象对应的子图层对象的名称，获取对应于相

同子图层对象名称的第一服务端图层对象。

可选地，所述第一图形合成请求由第二客户端基于第二客户端图层对象对所述第一应用的图形界面渲染完成后生成，所述第二客户端图层对象与所述第二服务端图层对象对应；其中，所述第二客户端为所述第一应用在第二子系统中的客户端。

5 可选地，还包括：所述图形合成器获取第二图形合成请求，所述第二图形合成请求对应于所述第一服务端图层对象；所述图形合成器根据所述第一服务端图层对象所对应的子图层对象，获取该子图层对象对应的第一应用图形界面窗口状态数据；所述图形合成器基于所述第一服务端图层对象，并根据所述第一应用图形界面窗口状态数据，响应所述第二图形合成请求。

10 可选地，所述第二图形合成请求由第一客户端基于第一客户端图层对象对所述第一应用的图形界面的窗口状态进行更新的操作生成，所述第一客户端图层对象与所述第一服务端图层对象对应；其中，所述第一客户端为所述第一应用在第一子系统中的客户端。

可选地，所述第一子系统为 Host 子系统，所述第二子系统为 Guest 子系统。

第二方面，提供一种图层处理方法，包括：

15 第一客户端指示图形合成器针对第一应用的子图层对象创建第一服务端图层对象，并向第二客户端发送通知；其中，所述第一客户端为所述第一应用在第一子系统中的客户端，所述第二客户端为所述第一应用在第二子系统中的客户端；

所述第二客户端根据所述通知，指示所述图形合成器针对所述第一应用的子图层对象创建第二服务端图层对象，并建立所述第二服务端图层对象与所述第一服务端图层对象间的关联关系。
20

可选地，建立所述第二服务端图层对象与所述第一服务端图层对象间的关联关系，包括：将所述第二服务端图层对象对应的子图层对象的名称，设置为与所述第一服务端图层对象对应的子图层对象的名称相同。

可选地，还包括：所述第一客户端获得所述第一服务端图层对象对应的子图层对象的名称，并将所述名称携带于所述通知中。
25

可选地，所述第一客户端获得所述第一服务端图层对象对应的子图层对象的名称，包括：所述第一客户端接收所述图形合成器在创建完成所述第一服务端图层对象后返回的所述子图层对象的名称；或者，所述第一客户端通过注册的回调函数获得所述子图层对象的名称，所述回调函数用于在所述图形合成器创建完成所述第一服务端图层对象后将所述第一服务端图层对象对应的子图层对象的名称通知给所述第一客户端。
30

可选地，第一客户端指示图形合成器创建第一服务端图层对象，包括：所述第一客户端通过调用用于创建图层对象的第一库函数模块，为第一应用的图形界面创建第一客户端图层对象，所述第一库函数模块被调用后通知所述图形合成器创建所述第一服务端图层对象。

- 5 可选地，所述第二客户端根据所述通知，指示所述图形合成器创建第二服务端图层对象，包括：所述第二客户端根据所述通知，通过调用用于创建图层对象的第二库函数模块，为所述图形界面创建第二客户端图层对象，所述第二库函数模块被调用后通知所述图形合成器创建所述第二服务端图层对象。

可选地，所述第一子系统为 Host 子系统，所述第二子系统为 Guest 子系统。

- 10 第三方面，提供一种图层处理方法，包括：

图形合成器根据第一客户端的指示，针对第一应用的子图层对象创建第一服务端图层对象；其中，所述第一客户端为所述第一应用在第一子系统客户端；

- 所述图形合成器根据第二客户端的指示，针对所述第一应用的子图层对象创建第二服务端图层对象；其中，所述指示是所述第二客户端根据所述第一客户端发送的通知向
15 所述图形合成器发送的；其中，所述第二客户端为所述第一应用在第二子系统客户端；

所述图形合成器建立所述第二服务端图层对象与所述第一服务端图层对象间的关联关系。

- 可选地，所述图形合成器建立所述第二服务端图层对象与所述第一服务端图层对象
20 间的关联关系，包括：所述图形合成器将所述第二服务端图层对象对应的子图层对象的名称，设置为与所述第一服务端图层对象对应的子图层对象的名称相同。

可选地，还包括：所述图形合成器从所述第二客户端获得所述第一服务端图层对象对应的子图层对象的名称；其中，所述子图层对象的名称是所述第一客户端携带于通知中发送给所述第二客户端的。

- 25 可选地，所述第一子系统为 Host 子系统，所述第二子系统为 Guest 子系统。

第四方面，提供一种图形合成器，所述图形合成器为操作系统中的第一子系统和第二子系统共享使用的图形合成器，所述图形合成器包括：

请求获取模块，用于获取第一图形合成请求，所述第一图形合成请求对应于第二系统中的第二服务端图层对象；

- 30 关联对象获取模块，用于根据所述第一图形合成请求所对应的第二服务端图层对象，

获取所述第二服务端图层对象所关联的第一子系统中的第一服务端图层对象；其中，所述第一服务端图层对象和所述第二服务端图层对象是针对第一应用的子图层对象创建的；

合成模块，用于获取所述第二服务端图层对象所对应的缓存，基于所述第一服务端
5 图层对象并使用所述缓存中存储的与所述第一应用的子图层对象对应的图形界面内容数据，响应所述第一图形合成请求。

可选地，所述关联对象获取模块具体用于：根据所述第二服务端图层对象对应的子图层对象的名称，获取对应于相同子图层对象名称的第一服务端图层对象。

可选地，所述第一图形合成请求由第二客户端基于第二客户端图层对象对所述第一
10 应用的图形界面渲染完成后生成，所述第二客户端图层对象与所述第二服务端图层对象对应；其中，所述第二客户端为所述第一应用在第二子系统客户端。

可选地，所述请求获取模块还用于：获取第二图形合成请求，所述第二图形合成请求对应于所述第一服务端图层对象；所述合成模块还用于：根据所述第一服务端图层对象所对应的子图层对象，获取该子图层对象对应的第一应用图形界面窗口状态数据；基
15 于所述第一服务端图层对象，并根据所述第一应用图形界面窗口状态数据，响应所述第二图形合成请求。

可选地，所述第二图形合成请求由第一客户端基于第一客户端图层对象对所述第一应用的图形界面的窗口状态进行更新的操作生成，所述第一客户端图层对象与所述第一服务端图层对象对应；其中，所述第一客户端为所述第一应用在第一子系统客户端。

20 第五方面，提供一种装置，包括：

第一客户端，用于指示图形合成器针对第一应用的子图层对象创建第一服务端图层对象，并向第二客户端发送通知；其中，所述第一客户端为所述第一应用在第一子系统客户端，所述第二客户端为所述第一应用在第二子系统客户端；

第二客户端，用于根据所述通知，指示所述图形合成器针对所述第一应用的子图层
25 对象创建第二服务端图层对象，并建立所述第二服务端图层对象与所述第一服务端图层对象间的关联关系。

可选地，所述第二客户端具体用于：将所述第二服务端图层对象对应的子图层对象的名称，设置为与所述第一服务端图层对象对应的子图层对象的名称相同。

可选地，所述第一客户端还用于：获得所述第一服务端图层对象对应的子图层对象
30 的名称，并将所述名称携带于所述通知中。

可选地，所述第一客户端具体用于获得所述第一服务端图层对象对应的子图层对象的名称，包括：接收所述图形合成器在创建完成所述第一服务端图层对象后返回的所述子图层对象的名称；或者，通过注册的回调函数获得所述子图层对象的名称，所述回调函数用于在所述图形合成器创建完成所述第一服务端图层对象后将所述第一服务端图层对象对应的子图层对象的名称通知给所述第一客户端。

可选地，所述第一客户端具体用于：通过调用用于创建图层对象的第一库函数模块，为第一应用的图形界面创建第一客户端图层对象，所述第一库函数模块被调用后通知所述图形合成器创建所述第一服务端图层对象。

可选地，所述第二客户端具体用于：根据所述通知，通过调用用于创建图层对象的第二库函数模块，为所述图形界面创建第二客户端图层对象，所述第二库函数模块被调用后通知所述图形合成器创建所述第二服务端图层对象。

第六方面，提供一种图形合成器，所述图形合成器为操作系统中的第一子系统和第二子系统共享使用的图形合成器，所述图形合成器包括：

第一图层对象创建模块，用于根据第一客户端的指示，针对第一应用的子图层对象创建第一服务端图层对象；以及，根据第二客户端的指示，针对所述第一应用的子图层对象创建第二服务端图层对象，所述指示是所述第二客户端根据所述第一客户端发送的通知向所述图形合成器发送的；其中，所述第一客户端为所述第一应用在第一子系统内的客户端，所述第二客户端为所述第一应用在第二子系统内的客户端；

关联模块，用于建立所述第二服务端图层对象与所述第一服务端图层对象间的关联关系。

可选地，所述关联模块具体用于：将所述第二服务端图层对象对应的子图层对象的名称，设置为与所述第一服务端图层对象对应的子图层对象的名称相同。

可选地，所述关联模块还用于：从所述第二客户端获得所述第一服务端图层对象对应的子图层对象的名称；其中，所述子图层对象的名称是所述第一客户端携带于通知中发送给所述第二客户端的。

第七方面，提供一个或多个计算机可读介质，所述可读介质上存储有指令，所述指令被一个或多个处理器执行时，使得通信设备执行上述第一方面中任一项所述的方法。

第八方面，提供一个或多个计算机可读介质，所述可读介质上存储有指令，所述指令被一个或多个处理器执行时，使得通信设备执行上述第二方面中任一项所述的方法。

第九方面，提供一个或多个计算机可读介质，所述可读介质上存储有指令，所述指

令被一个或多个处理器执行时，使得通信设备执行上述第三方面中任一项所述的方法。

第十方面，提供一种装置，包括：一个或多个处理器；以及，一个或多个计算机可读介质，所述可读介质上存储有指令，所述指令被所述一个或多个处理器执行时，使得所述装置执行上述第一方面中任一项所述的方法。

5 第十一方面，提供一种装置，包括：一个或多个处理器；以及，一个或多个计算机可读介质，所述可读介质上存储有指令，所述指令被所述一个或多个处理器执行时，使得所述装置执行上述第二方面中任一项所述的方法。

第十二方面，提供一种装置，包括：一个或多个处理器；以及，一个或多个计算机可读介质，所述可读介质上存储有指令，所述指令被所述一个或多个处理器执行时，使
10 得所述装置执行上述第三方面中任一项所述的方法。

本申请的上述实施例中，图形合成器获取到第一图形合成请求，由于所述第一图形合成请求对应于第二子系统第二服务端图层对象，因此，图形合成器可根据所述第一图形合成请求所对应的第二服务端图层对象，获取所述第二服务端图层对象所关联的第一子系统的第一服务端图层对象；所述图形合成器获取所述第二服务端图层对象所
15 对应的缓存，并根据所述缓存中存储的与所述子图层对象对应的图形界面内容数据，响应所述第一图形合成请求。其中，由于所述第一服务端图层对象和所述第二服务端图层对象是针对第一应用的子图层对象创建的，因此图形合成器通过上述方式响应第一图形合成请求，从而使用第一应用的子图层对象对应的图形界面内容数据进行图形合成，实现了在一种操作系统框架内，基于不同操作系统的应用或服务在图形合成层面的融合。

20

附图说明

本申请的实施例通过示例而非限制的方式示出在所附附图中，类似的附图标记表示类似的元素。

图1示例性地示出了本申请实施例所使用的系统架构示意图；

25 图2示例性地示出了本申请实施例中图层对象的Z-order顺序示意图；

图3示例性地示出了本申请实施例提供的图层对象的创建流程示意图；

图4示例性地示出了本申请实施例创建得到的图层对象的示意图；

图5示例性地示出了本申请实施例提供的图形合成的流程示意图；

图6示例性地示出了本申请另外的实施例提供的图形合成的流程示意图；

30 图7示例性地示出了本申请实施例中Host_client_appA和Guest_client_appA创建与

图形合成相关的对象的过程；

图 8 示例性地示出了本申请实施例提供的图形合成器的结构示意图；

图 9 示例性地示出了本申请另外的实施例提供的图形合成器结构示意图；

图 10 示例性地示出了本申请实施例提供的通信装置的结构示意图。

5

具体实施方式

虽然本申请的概念易于进行各种修改和替代形式，但是其具体实施例已经通过附图中的示例示出并且将在本文中详细描述。然而，应当理解，没有意图将本申请的概念限制为所公开的特定形式，而是相反，意图是覆盖与本申请以及所附权利要求一致的所有修改、等同物和替代物。

10

说明书中对“一个实施例”、“实施例”、“说明性实施例”等的引用，指示所描述的实施例可包括特定特征、结构或特性，但是每个实施例可以或可以不必包括特定特征、结构或特性。此外，这样的短语不一定指的是相同的实施例。进一步地，认为在本领域技术人员的知识范围内，当结合实施例描述特定特征、结构或特性时，结合无论是否明确描述的其它实施例影响这样的特征、结构或特性。另外，应当理解，以“A，B 和 C 中的至少一个”的形式包括在列表中的项目可以表示 (A)；(B)；(C)；(A 和 B)；(A 和 C)；(B 和 C)；或 (A，B 和 C)。类似地，以“A，B 或 C 中的至少一个”的形式列出的项目可以表示 (A)；(B)；(C)；(A 和 B)；(A 和 C)；(B 和 C) 或 (A，B 和 C)。

15

20

在一些情况下，所公开的实施例可以在硬件、固件、软件或其任何组合中实现。所公开的实施例还可以被实现为由一个或多个暂时性或非暂时性机器可读（例如，计算机可读）存储介质携带或存储的指令，其可以由一个或多个处理器读取和执行。机器可读存储介质可以体现为用于以机器可读形式（例如，易失性或非易失性存储器、介质盘或其他介质）存储或传输信息的任何存储设备，机制或其他物理结构的设备）。

25

在附图中，一些结构或方法特征可以以特定布置和/或顺序示出。然而，应当理解，可能不需要这样的具体布置和/或排序。相反，在一些实施例中，这些特征可以以与说明性附图中所示不同的方式和/或顺序来布置。另外，在特定图中包括结构或方法特征并不意味着暗示这种特征在所有实施例中都是需要的，并且在一些实施例中可以不包括或可以与其他特征组合。

30

为了在一种操作系统框架内更好地兼容另一种操作系统，为用户提供更好的用户体

验，本申请实施例提供了在一种操作系统框架内针对不同操作系统在图形合成层面上的融合和统一处理操作，为用户提供无缝的双系统用户体验。

本申请实施例可适用于操作系统框架内包含第一子系统和第二子系统，且第一子系统和第二子系统共享第一子系统的图形合成器的场景。其中，第一子系统可以是主系统或称 Host 子系统，第二子系统可以是辅系统或称 Guest 子系统。具体地，第一子系统可以是云操作系统，第二子系统可以是传统的操作系统。更具体地，第一子系统可以是 YunOS，第二子系统可以是 Android。

下面结合附图对本申请实施例进行详细描述。

图 1 示例性地示出了本申请实施例使用的一种包括多个子系统的操作系统框架示意图。以 YunOS 作为第一子系统，Android 为第二子系统为例，图 1 示出了在操作系统框架内的双系统兼容架构。

如图 1 所示，操作系统框架中包括 Host 子系统与 Guest 子系统，其中，YunOS 即为 Host 子系统，Android 即为 Guest 子系统，两个子系统通过容器（container）技术隔离。该框架内的硬件抽象层（Hardware Abstraction Layer，简称 HAL）提供了图形显示等方面的接口和驱动服务。

Host 子系统中的 Glibc，即 C 运行库，是 linux 中最底层的应用程序编程接口（Application Programming Interface，简称 API）。Libhybris 为兼容层，能够使基于 Glibc 的操作系统重用现有的 Android 的驱动（driver）服务。Gralloc 模块位于硬件抽象层中，封装了对帧缓冲区的所有访问操作。Compositor 是 Host 子系统的图形合成器，该图形合成器为 Host 子系统和 Guest 子系统共享使用。

需要说明的是，上述系统架构仅以 YunOS 和 Android 为例进行说明，对于其他类型的操作系统，也可基于上述原理实现双系统在图形合成层面的兼容以及有效融合。

为了理解本申请实施例，下面首先描述与图形合成相关的技术术语。

一个应用程序可包含一个或多个图形界面（或称窗口），一个图形界面可至少对应一个图层对象，称为 surface。每个图形界面在屏幕上有它的位置、大小，以及需要显示的内容。一个图形界面对应的图层对象（surface）可被理解为该图形界面对应的一个容器，用来存储该图形界面的控制信息（比如该图形界面在屏幕上的大小、位置等信息）和/或图形界面的数据（即图形界面需要显示的内容）。图层对象（surface）可由应用程序在图形合成器（compositor）中的客户端（client）创建。

多个图层对象（surface）之间可能有重叠，图层对象（surface）之间的层叠关系可

用 Z-order 描述。在屏幕平面的垂直方向存在一个 Z 轴，所有的图层对象（surface）根据在 Z 轴上的坐标来确定前后顺序，即描述各个图层对象（surface）之间的上下覆盖关系。这个在 Z 轴上的顺序，可称为 Z-order。

图 2 示例性地示出了多个图层对象（surface）在 Z 轴上的顺序。其中，按照在屏幕上显示时从顶至底的顺序，屏幕上包括 surface0、surface1 和 surface2。surface0 部分覆盖 surface1，surface1 部分覆盖 surface2。surface0、surface1 和 surface2 的 Z-order 分别表示为：1,2,3。

一个图层对象（surface）对应 Z 轴上的一个层对象，称为 layer。一个层对象（layer）提供了对该 layer 所对应的图层对象（surface）的控制信息的操作，以及对该 surface 的内容的处理。

图形合成器（compositor）主要用于对图形合成过程进行管理，即对不同图层对象（surface）的融合进行控制，比如，计算两个图层对象（surface）重叠的区域，而图层对象（surface）需要显示的内容则可在 layer 中进行。

在包含第一子系统和第二子系统的操作系统框架内，针对一些应用程序的图形界面创建有主图层对象（即主 surface）和子图层对象（即子 surface）。对于子 surface，其所对应的图形界面窗口状态数据保存在第一子系统中，其所对应的图形界面内容数据保存在第二子系统中。对于上述应用，当其图形界面的内容发生改变时，需要在图形合成层面由第一子系统和第二子系统共用的图形合成器基于子 surface 对应的窗口状态数据和图形界面内容数据进行图形合成。

其中，窗口状态数据可在窗口状态更新事件发生时生成或更新。所述窗口状态更新事件可包括但不限于：基于窗口移动生成的事件、基于窗口缩放生成的事件、基于窗口旋转生成的事件。所述窗口状态数据可包括但不限于：用于描述窗口位置和/或方向的数据。

为了实现此目的，本申请实施例中，针对应用程序的图形界面（更具体地，针对该图形界面的子 surface）在第一子系统中创建第一服务端图层对象，针对该图形界面（更具体地，针对该图形界面的子 surface）在第二子系统中创建第二服务端图层对象以及对应的缓存（该缓存用于存储该图形界面的内容数据），并建立第一服务端图层对象和第二服务端图层对象间的关联。这样，第一服务端图层对象和第二服务端图层对象均关联于该子 surface。当该应用程序的图形界面的内容发生变化时，该应用程序在第二子系统

中的客户端基于上述关联关系确定出该图形界面在第一子系统中对应的第一服务端图层

对象,并基于第一服务端图层对象使用该子 Surface 对应的缓存中的图形界面内容数据进行图形合成。进一步地,当该应用程序的图形界面的窗口状态发生变化时,该应用程序在第一子系统客户端基于第一服务端图层对象使用子 surface 对应的图形界面窗口状态数据进行图形合成。

- 5 由此可见,本申请的上述实施例中,通过不同子系统服务端图层对象之间的关联关系,使得第二子系统相关的图形合成事件发生时,第一子系统的第一服务端图层对象可基于第二子系统第二服务端图层对象对应的缓存进行图形合成,实现了不同子系统在图形合成层面的融合。

下面结合图 3 和图 4 对本申请实施例提供的图层对象(surface)创建过程进行描述,
10 结合图 5 和图 6 对本申请实施例提供的图形合成过程进行描述。

本申请实施例中,对于包含有两个子系统(第一子系统和第二子系统)的操作系统架构来说,针对基于第二子系统的应用,可创建两个客户端:一个是该应用在第一系统中的客户端,本申请实施例中称为第一客户端,另一个是该应用在第二系统中的客户端,本申请实施例中称为第二客户端。其中,该应用的两个客户端可对应于该应用在
15 操作系统中的两个不同的进程,每个进程拥有各自的内存区域。具体实施时,可在应用程序启动时创建上述应用客户端。

基于第二子系统的应用,其在第一系统中的客户端和在第二系统中的客户端,可分别创建服务端图层对象(surface),并建立相互的关联关系。

图 3 示例性地示出了本申请实施例提供的图层对象(surface)的创建流程。该流程
20 可在应用程序被启动时执行,也可在打开新的图形界面(窗口)时执行,或者在其他情况下执行。以下流程以第一应用为例进行描述,其中,该应用在第一子系统(如主系统)中的客户端称为第一客户端,该应用在第二子系统(如辅系统)中的客户端称为第二客户端。第一客户端可创建子图层对象(以下简称子 surface)。

如图所示,该流程可包括:

- 25 S301:第一客户端指示图形合成器为第一应用的图形界面创建第一服务端图层对象。

可选地,第一客户端可通过调用用于创建图层对象的第一库函数模块,为第一应用的图形界面创建第一客户端图层对象,该第一库函数模块被调用后通知图形合成器创建与第一客户端图层对象对应的第一服务端图层对象。其中,第一库函数模块可以是第一
30 子系统函数库,其中可包含多种函数,这些函数中包括用于创建图层对象的函数。该函数库可将函数接口(比如应用程序编程接口(Application Programming Interface,简

称 API)) 提供给应用层使用。第一客户端可基于用于创建图层对象的函数的接口, 调用相应函数创建图层对象。

可选地, 第一客户端可基于第一应用程序的子 surface 创建第一服务端图层对象, 使得第一服务端图层对象与该子 surface 对应或与该子 surface 关联。比如, 第一服务端图层对象中可包括第一应用程序的子 surface 的名称, 从而使第一服务端图层对象与该子 surface 关联。

S302: 第一客户端向第二客户端发送通知。该通知用于指示为该第一应用的图形界面创建第二服务端图层对象。

可选地, S302 中, 第一客户端可采用进程间通信 (Inter-Process Communication, 简称 IPC) 协议, 向第二客户端发送通知消息。进一步地, 该通知消息中可包括该子 surface 的名称, 该子 surface 为第一服务端图层对象所对应的子 surface。

可选地, 第一客户端可采用以下方式获得第一服务端图层对象所对应的子 surface 的名称:

方式 1: 第一客户端接收图形合成器在创建完成该第一服务端图层对象后返回的该第一服务端图层对象所对应的子 surface 的名称;

方式 2: 第一客户端可预先注册回调函数, 当图形合成器创建完成第一服务端图层对象后, 该回调函数可将该第一服务端图层对象所对应的子 surface 的名称通知给第一客户端。

S303: 第二客户端根据该通知, 指示图形合成器为该第一应用的图形界面创建第二服务端图层对象, 并建立所述第二服务端图层对象与所述第一服务端图层对象间的关联关系。

可选地, 第二客户端可基于第一客户端所通知的子 surface 的名称, 基于该子 surface 指示图形合成器为第一应用的图形界面创建第二服务端图层对象, 使得创建得到的第二服务端图层对象与该子 surface 对应或关联。

可选地, 第二客户端可通过调用用于创建图层对象的第二库函数模块, 为第一应用的图形界面创建第二客户端图层对象, 该第二库函数模块被调用后通知图形合成器创建与第二客户端图层对象对应的第二服务端图层对象。其中, 第二库函数模块可以是第二子系统函数库, 其中可包含多种函数, 这些函数中包括用于创建图层对象的函数。该函数库可将函数接口 (比如应用程序编程接口 (API) 提供给应用层使用。第二客户端可基于用于创建图层对象的函数的接口, 调用相应函数创建图层对象。

在 S303 中,图形合成器可采用多种方式建立第一服务端图层对象和第二服务端图层对象之间的关联关系。比如,可将第二服务端图层对象所对应的图层对象的名称(即子 surface 的名称),设置为与第一服务端图层对象所对应的图层对象的名称(即子 surface 的名称)相同,这样,第一服务端图形对象和第二服务端图形对象对应于相同名称的子 surface,即关联与同一子 surface。再比如,将第一服务端图层对象的标识和第二服务端图层对象的标识以标识对的形式存储。

进一步地,第二客户端还可为第一应用的图形界面创建缓存,该缓存与第一应用的图形界面的子 surface 对应。由于该子 surface 与第二服务端图层对象和/或第二客户端图层对象相对应,因此该缓存也与第二服务端图层对象和/或第二客户端图层对象相对应。通过该缓存可获取到该图形界面的内容数据,所述内容数据可以是指图形界面渲染处理后的数据。该缓存用于存储该图形界面的内容数据更新事件对应的内容数据。所述事件可包括图形界面渲染事件,比如在对图形界面渲染完成后可生成该事件。

上述路程的步骤顺序没有严格要求,比如,S301 和 S302 的先后顺序可以调换。

基于上述流程,图 4 示例性地示出了针对第一应用的图形界面创建得到的图层对象。如图所示,第一客户端图层对象和第一服务端图层对象是第一应用在第一子系统的第一客户端创建的图层对象,第二客户端图层对象和第二服务端图层对象是第一应用在第二子系统的第一客户端创建的图层对象。其中,在第一子系统中,首先创建第一客户端图层对象,并以第一客户端图层对象作为父对象创建子对象——第一服务端图层对象;在第二子系统中,首先创建第二客户端图层对象,并以第二客户端图层对象作为父对象创建子对象——第二服务端图层对象。具体创建过程可参见前述实施例。第一服务端图层对象和第二服务端图层对象之间关联,并均关联于第一应用程序的图形界面的子 surface。第二客户端图层对象与用于存储第一应用的图形界面内容数据的缓存相对应。

上述基于第二子系统的第一应用程序,其在运行过程中可响应用户操作,根据用户的操作更新该应用程序的图形界面,也可以根据设定周期刷新该应用程序的图形界面。

图形界面更新可包括两个方面:

(1) 图形界面的内容更新,该过程即为图形渲染过程。当图形渲染过程完成后(即得到图形界面内容数据后),图形合成器进行图形合成。

(2) 图形界面的窗口状态更新。当图形界面的窗口状态更新后,图形合成器进行图形合成。

上述两个方面可能同时发生,也可能仅发生其中一个方面描述的更新过程。

图 5 示例性地示出了本申请实施例提供的图形合成的流程示意图。该流程描述了第一应用的图形界面渲染完成后，图形合成器进行图形合成。其中，第一应用在第二子系统中的第二客户端基于第二客户端图层对象，对相应图形界面进行渲染，渲染后的图形界面内容数据存储于第二子系统中该第一应用的子 surface 所对应的缓存中。渲染完成后，

5 可触发以下流程：

S501：图形合成器获取第一图形合成请求，该第一图形合成请求对应于第二服务端图层对象。

该步骤可基于异步事件机制实现，也可基于同步事件机制实现。在采用异步事件机制的情况下，第二客户端基于第二服务端图层对象完成图形界面渲染后生成第一图形合成请求（或称事件），并将该请求存储于事件队列，图形合成器基于异步事件机制从事

10 件队列中获取第一图形合成请求进行响应。在采用同步事件机制的情况下，第二客户端基于第二服务端图层对象完成图形界面渲染后，向图形合成器发送请求，以请求图形合成器立即响应该请求进行图形合成。

其中，在具体实施时，第二客户端可调用第二客户端图层对象，再由第二客户端图

15 层对象调用第二服务端图层对象进行图形渲染。

S502：图形合成器根据该请求所对应的第二服务端图层对象，获取该第二服务端图层对象关联的第一服务端图层对象。

在具体实施时，第一服务端图层对象所对应的子 surface 的名称，与第二服务端图层对象所对应的子 surface 的名称相同，相应地，图形合成器可根据第二服务端图层对象对

20 应的子 surface 的名称，获取对应于相同子 surface 名称的第一服务端图层对象。

S503：图形合成器获取第二服务端图层对象所对应的缓存，基于 S502 中获取的第一服务端图层对象并使用缓存中存储的与第一应用的子图层对象对应的图形界面内容数据，响应该第一图形合成请求。

该步骤中，图形合成器可根据第一服务端图层对象，获取第一服务端图层对象关联

25 的第二服务端图层对象所对应的缓存，该缓存中存储有第二服务端图层对象所对应的子 surface 的图形界面内容数据。图形合成器基于获取到的第一服务端图层对象，并根据该缓存中存储的图形界面内容数据，响应第一图形合成请求。在图形合成器响应该图形合成请求时，进一步地可根据相应子 surface（即该第一服务端图层对象所对应的子 surface）的窗口状态数据以及各图层对象的 Z-order 等控制信息进行图形合成。

30 图 6 示例性地示出了本申请另外的实施例提供的图形合成的流程示意图。该流程描

述了第一应用的图形界面的窗口状态发生变化后，图形合成器进行图形合成。其中，第一应用在第一子系统的第一客户端更新图形界面的窗口状态。窗口状态更新完成后，可触发以下流程：

S601：图形合成器获取第二图形合成请求，该第二图形合成请求根据第一应用程序的图形界面窗口状态更新事件生成。

该步骤可基于异步事件机制实现，也可基于同步事件机制实现。在采用异步事件机制的情况下，第一客户端基于第一服务端图层对象完成图形界面的窗口状态更新后生成第二图形合成请求（或称事件），并将该请求存储于事件队列，图形合成器基于异步事件机制从事件队列中获取第二图形合成请求进行响应。在采用同步事件机制的情况下，第一客户端基于第一服务端图层对象完成图形界面的窗口状态更新后，向图形合成器发送请求，以请求图形合成器立即响应该请求进行图形合成。

其中，在具体实施时，第一客户端可调用第一客户端图层对象，再由第一客户端图层对象调用第一服务端图层对象进行窗口状态更新。

S602：图形合成器根据该第一服务端图层对象响应该第二图形合成请求。

该步骤中，图形合成器可根据第一服务端图层对象，获取第一服务端图层对象对应的子 surface，基于该子 surface 对应的图形界面的窗口状态数据，响应第二图形合成请求。

为了更清楚地理解本申请实施例，下面结合具体场景对本申请实施例进行描述。

智能手机使用的操作系统的框架中包括 Host 子系统（对应于云操作系统）和 Guest 子系统（对应于 Android 系统），这两个子系统共享使用一个图形合成器（Compositor server）。该智能手机上安装了应用程序 A，应用程序 A 是基于 Andorid 系统的应用程序窗口小部件（Widget）应用。

当用户点击操作系统的主界面中应用程序 A 的图标，该操作被操作系统捕获，生成应用程序 A 的启动事件，应用程序 A 的启动事件被发送给 Host 子系统和 Guest 子系统。在 Host 子系统中，Host 子系统根据应用程序 A 的启动事件为应用程序 A 在 Host 子系统中创建一个客户端（client），为描述方便，此处将该客户端称为 Host_client_appA，Host_client_appA 创建应用程序 A 的子 surface，该子 surface 的名称为“N”。在 Guest 子系统中，Guest 子系统根据应用程序 A 的启动事件为应用程序 A 在 Guest 子系统中创建一个客户端（client），为描述方便，此处将该客户端称为 Guest_client_appA。Host_client_appA 与 Guest_client_appA 之间可基于 IPC 机制进行通信。

Host_client_appA 和 Guest_client_appA 创建完成后，可创建与图形合成相关的对象。

图 7 示例性地示出了 Host_client_appA 和 Guest_client_appA 创建与图形合成相关的对象的过程。

在图 7 的 S701 中, Host_client_appA 调用 Host 子系统中的图形化用户界面库模块 (libgui 模块) 为应用程序 A 创建 Host 子系统中客户端的 surface 对象, 这里将该 surface 对象称为 Host Client Surface, 简称为 HC_surface 对象, HC_surface 对象与应用程序 A 的子 surface 对应, 是针对应用程序 A 的子 surface 创建的。其中, libgui 模块中封装有 native 层针对 surface 对象的相关函数, 可用于创建 surface 对象。libgui 模块被调用时通知图形合成器 (Compositor server) 为应用程序 A 创建 Host 子系统中服务端的 surface 对象, 这里将该 surface 对象称为 Host Server Surface, 简称为 HS_surface 对象, HS_surface 对象与应用程序 A 的子 surface 对应, 是针对应用程序 A 的子 surface 创建的。HS_surface 对象创建完成后, Host_client_appA 注册的回调函数将 HS_surface 所对应的子 surface 名字 “N” 通知给 Host_client_appA。

在图 7 的 S702 中, Host_client_appA 向 Guest_client_appA 发送请求, 其中携带 HS_surface 对象所对应的子 surface 名字 “N”, 以请求为应用程序 A 创建 Guest 子系统 15 中的相应 surface 对象。

在图 7 的 S703 中, Guest_client_appA 接收到 Host_client_appA 发送的请求后, 调用 Guest 子系统中的函数库模块为应用程序 A 创建 layer 对象, 根据该请求消息中的 HS_surface 对象对应的子 surface 名字 “N” 为应用程序 A 创建 Guest 子系统中客户端的 surface 对象, 这里将该 surface 对象称为 Guest Client Surface, 简称为 GC_surface 对象, GC_surface 对象与应用程序 A 的子 surface 对应, 是针对应用程序 A 的子 surface 创建的。该函数库模块被调用时通知图形合成器 (Compositor server) 根据 GC_surface 对象所对应的子 surface 名字为应用程序 A 创建 Guest 子系统中的服务端 surface 对象, 这里将该 surface 对象称为 Guest Client Surface, 简称为 GS_surface 对象, 该 GS_surface 对象所对应的子 surface 名称为 “N”。

25 Guest_client_appA 还创建该子 surface 对应的 buffer, 该 buffer 用于存储应用程序 A 的图形界面内容数据。由于 GC_surface 对象和 GS_surface 对象是基于该子 surface 创建的, 因此, 该 buffer 也与 GC_surface 对象和 GS_surface 对象相对应。

应用程序 A 运行过程中, Guest_client_appA 进行图形界面渲染, 渲染后的图形界面内容数据存储于应用程序 A 在 Guest 子系统内的 GC_surface 对象所对应的 buffer 中。

30 当该 buffer 中的图形界面内容数据已渲染完成或者是达到其他进行图形界面合成的

条件时, Guest_client_appA 通过 GC_surface 对象获取 GS_surface 对象, 基于 GS_surface 对象并采用 IPC 通信机制向 compositor 发送第一图形合成请求; compositor 根据 GS_surface 对象获取关联的 HS_surface 对象, 并基于该 HS_surface 对象, 利用 GS_surface 对象所对应的 buffer 中的图形界面内容数据以及 Host 子系统中保存的相应子 surface 的图形界面窗口状态数据, 并结合 Z-order 等信息进行图形合成。

应用程序 A 运行过程中, 图形界面的位置发生变化, Host_client_appA 通过 IPC 通信机制通知 compositor 针对位置更新后的图形界面进行图形合成, compositor 基于 HS_surface 对象获取相应子 surface 所对应的图形界面窗口状态数据, 并结合 Z-order 等信息进行图形合成。

本申请的上述实施例中, 图形合成器获取到第一图形合成请求, 由于所述第一图形合成请求对应于第二子系统内的第二服务端图层对象, 因此, 图形合成器可根据所述第一图形合成请求所对应的第二服务端图层对象, 获取所述第二服务端图层对象所关联的第一子系统内的第一服务端图层对象; 所述图形合成器获取所述第二服务端图层对象所对应的缓存, 并根据所述缓存中存储的与所述子图层对象对应的图形界面内容数据, 响应所述第一图形合成请求。其中, 由于所述第一服务端图层对象和所述第二服务端图层对象是针对第一应用的子图层对象创建的, 因此图形合成器通过上述方式响应第一图形合成请求, 从而使用第一应用的子图层对象对应的图形界面内容数据进行图形合成, 实现了在一种操作系统框架内, 基于不同操作系统的应用或服务在图形合成层面的融合。

基于相同的技术构思, 本申请实施例还提供了一种能够实现上述实施例描述的方法的装置。

参见图 8, 为本申请实施例提供的图形合成器的结构示意图, 如图所示, 所述图形合成器为操作系统中的第一子系统和第二子系统共享使用的图形合成器, 所述图形合成器包括: 请求获取模块 801、关联对象获取模块 802、合成模块 803, 其中:

请求获取模块 801 用于获取第一图形合成请求, 所述第一图形合成请求对应于第二子系统内的第二服务端图层对象;

关联对象获取模块 502 用于根据所述第一图形合成请求所对应的第二服务端图层对象, 获取所述第二服务端图层对象所关联的第一子系统内的第一服务端图层对象; 其中, 所述第一服务端图层对象和所述第二服务端图层对象是针对第一应用的子图层对象创建的;

合成模块 803 用于获取所述第二服务端图层对象所对应的缓存, 基于所述第一服务

端图层对象并使用所述缓存中存储的与所述第一应用的子图层对象对应的图形界面内容数据，响应所述第一图形合成请求。

可选地，关联对象获取模块 802 具体用于：根据所述第二服务端图层对象对应的子图层对象的名称，获取对应于相同子图层对象名称的第一服务端图层对象。

5 可选地，所述第一图形合成请求由第二客户端基于第二客户端图层对象对所述第一应用的图形界面渲染完成后生成，所述第二客户端图层对象与所述第二服务端图层对象对应；其中，所述第二客户端为所述第一应用在所述第二系统中的客户端。

可选地，请求获取模块 801 还用于：获取第二图形合成请求，所述第二图形合成请求对应于所述第一服务端图层对象；所述合成模块还用于：根据所述第一服务端图层对象所对应的子图层对象，获取该子图层对象对应的第一应用图形界面窗口状态数据；基
10 于所述第一服务端图层对象，并根据所述第一应用图形界面窗口状态数据，响应所述第二图形合成请求。

可选地，所述第二图形合成请求由第一客户端基于第一客户端图层对象对所述第一应用的图形界面的窗口状态进行更新的操作生成，所述第一客户端图层对象与所述第一
15 服务端图层对象对应；其中，所述第一客户端为所述第一应用在第一系统中的客户端。

基于相同的技术构思，本申请实施例还提供了一个或多个计算机可读介质，所述可读介质上存储有指令，所述指令被一个或多个处理器执行时，使得通信设备执行前述实施例描述的图形界面处理方法。

基于相同的技术构思，本申请实施例还提供了一种能够实现上述实施例描述的方法
20 的装置。

基于相同的技术构思，本申请实施例还提供了一种能够实现上述实施例描述的方法的装置。

参见图 9，为本申请实施例提供的图形合成器的结构示意图，如图所示，所述图形合成器为操作系统中的第一子系统和第二子系统共享使用的图形合成器，所述图形合成
25 器包括：第一图层对象创建模块 901、关联模块 902，其中：

第一图层对象创建模块 901 用于根据第一客户端的指示，针对第一应用的子图层对象创建第一服务端图层对象；以及，根据第二客户端的指示，针对所述第一应用的子图层对象创建第二服务端图层对象，所述指示是所述第二客户端根据所述第一客户端发送的通知向所述图形合成器发送的；其中，所述第一客户端为所述第一应用在第一子系统
30 中的客户端，所述第二客户端为所述第一应用在所述第二系统中的客户端；

关联模块 902 用于建立所述第二服务端图层对象与所述第一服务端图层对象间的关联关系。

可选地，关联模块 902 具体用于：将第二服务端图层对象对应的子图层对象的名称，设置为与所述第一服务端图层对象对应的子图层对象的名称相同。

5 可选地，关联模块 902 还用于：从所述第二客户端获得所述第一服务端图层对象对应的子图层对象的名称；其中，所述子图层对象的名称是所述第一客户端携带于通知中发送给所述第二客户端的。

基于相同的技术构思，本申请实施例还提供了一个或多个计算机可读介质，所述可读介质上存储有指令，所述指令被一个或多个处理器执行时，使得通信设备执行前述实
10 施例描述的图形界面处理方法。

基于相同的技术构思，本申请实施例还提供了一种装置，该装置中包括第一客户端和第二客户端。所述第一客户端和所述第二客户端的功能可如前述实施例所述，在此不再重复。

基于相同的技术构思，本申请实施例还提供了一个或多个计算机可读介质，所述可读介质上存储有指令，所述指令被一个或多个处理器执行时，使得通信设备执行前述实
15 施例描述的图形界面处理方法。

基于相同的技术构思，本申请实施例还提供了一种装置 1000，该装置 1000 可实现前述实施例中服务器侧的流程。

图 10 示例性地示出了根据各种实施例的示例装置 1000，装置 1000 可包括一个或多个
20 处理器 1002，系统控制逻辑 1001 耦合于至少一个处理器 1002，非易失性存储器（non-volatile memory, NMV）/存储器 1004 耦合于系统控制逻辑 1001，网络接口 1006 耦合于系统控制逻辑 1001。

处理器 1002 可包括一个或多个单核处理器或多核处理器。处理器 1002 可包括任何一般用途处理器或专用处理器（如图像处理器、应用处理器基带处理器等）的组合。

25 一个实施例中的系统控制逻辑 1001，可包括任何适当的接口控制器，以提供到处理器 1002 中的至少一个的任何合适的接口，和/或提供到与系统控制逻辑 1001 通信的任何合适的设备或组件的任何合适的接口。

一个实施例中的系统控制逻辑 1001，可包括一个或多个内存控制器，以提供到系统内存 1003 的接口。系统内存 1003 用来加载以及存储数据和/或指令。例如，对应装置 1000，
30 在一个实施例中，系统内存 1003 可包括任何合适的易失性存储器。

NVM/存储器 1004 可包括一个或多个有形的非暂时的计算机可读介质，用于存储数据和/或指令。例如，NVM/存储器 1004 可包括任何合适的非易失性存储装置，如一个或多个硬盘（hard disk device, HDD），一个或多个光盘（compact disk, CD），和/或一个或多个数字通用盘（digital versatile disk, DVD）。

- 5 NVM/存储器 1004 可包括存储资源，该存储资源物理上是该系统所安装的或者可以被访问的设备的一部分，但不一定是设备的一部分。例如，NVM/存储器 1004 可经由网络接口 1006 被网络访问。

- 系统内存 1003 以及 NVM/存储器 1004 可分别包括临时的或持久的指令 1010 的副本。指令 1010 可包括当由处理器 1002 中的至少一个执行时导致装置 1000 实现图 4 至图 6 描述的方法之一或组合的指令。各实施例中，指令 1010 或硬件、固件，和/或软件组件可
10 另外地/可替换地被置于系统控制逻辑 1001，网络接口 1006 和/或处理器 1002。

- 网络接口 1006 可包括一个接收器来为装置 1000 提供无线接口来与一个或多个网络和/或任何合适的设备进行通信。网络接口 1006 可包括任何合适的硬件和/或固件。网络接口 1006 可包括多个天线来提供多输入多输出无线接口。在一个实施例中，网络接口
15 1006 可包括一个网络适配器、一个无线网络适配器、一个电话调制解调器，和/或无线调制解调器。

- 在一个实施例中，处理器 1002 中的至少一个可以与用于系统控制逻辑的一个或多个控制器的逻辑一起封装。在一个实施例中，处理器中的至少一个可以与用于系统控制逻辑的一个或多个控制器的逻辑一起封装以形成系统级封装。在一个实施例中，处理器中
20 的至少一个可以与用于系统控制逻辑的一个或多个控制器的逻辑集成在相同的管芯上。在一个实施例中，处理器中的至少一个可以与用于系统控制逻辑的一个或多个控制器的逻辑集成在相同的管芯上以形成系统芯片。

- 装置 1000 可进一步包括输入/输出装置 1005。输入/输出装置 1005 可包括用户接口旨在使用户与装置 1000 进行交互，可包括外围组件接口，其被设计为使得外围组件能够
25 与系统交互，和/或，可包括传感器，旨在确定环境条件和/或有关装置 1000 的位置信息。

权利要求书

1.一种图层处理方法，其特征在于，包括：

图形合成器获取第一图形合成请求，所述第一图形合成请求对应于第二子系统中的第二服务端图层对象；

5 所述图形合成器根据所述第一图形合成请求所对应的第二服务端图层对象，获取所述第二服务端图层对象所关联的第一子系统的第一服务端图层对象；其中，所述第一服务端图层对象和所述第二服务端图层对象是针对第一应用的子图层对象创建的；

 所述图形合成器获取所述第二服务端图层对象所对应的缓存，基于所述第一服务端图层对象并使用所述缓存中存储的与所述第一应用的子图层对象对应的图形界面内容
10 数据，响应所述第一图形合成请求。

2.如权利要求1所述的方法，其特征在于，所述获取所述第二服务端图层对象所关联的第一服务端图层对象，包括：

 所述图形合成器根据所述第二服务端图层对象对应的子图层对象的名称，获取对应于相同子图层对象名称的第一服务端图层对象。

15 3.如权利要求1或2所述的方法，其特征在于，所述第一图形合成请求由第二客户端基于第二客户端图层对象对所述第一应用的图形界面渲染完成后生成，所述第二客户端图层对象与所述第二服务端图层对象对应；其中，所述第二客户端为所述第一应用在第二子系统中的客户端。

4.如权利要求1或2所述的方法，其特征在于，还包括：

20 所述图形合成器获取第二图形合成请求，所述第二图形合成请求对应于所述第一服务端图层对象；

 所述图形合成器根据所述第一服务端图层对象所对应的子图层对象，获取该子图层对象对应的第一应用图形界面窗口状态数据；

 所述图形合成器基于所述第一服务端图层对象，并根据所述第一应用图形界面窗口
25 状态数据，响应所述第二图形合成请求。

 5.如权利要求4所述的方法，其特征在于，所述第二图形合成请求由第一客户端基于第一客户端图层对象对所述第一应用的图形界面的窗口状态进行更新的操作生成，所述第一客户端图层对象与所述第一服务端图层对象对应；其中，所述第一客户端为所述第一应用在第一子系统中的客户端。

30 6.如权利要求1中任一项所述的方法，其特征在于，所述第一子系统为Host子系统，

所述第二子系统为 Guest 子系统。

7.一种图层处理方法，其特征在于，包括：

第一客户端指示图形合成器针对第一应用的子图层对象创建第一服务端图层对象，并向第二客户端发送通知；其中，所述第一客户端为所述第一应用在第一子系统
5 中的客户端，所述第二客户端为所述第一应用在第二子系统
中的客户端；

所述第二客户端根据所述通知，指示所述图形合成器针对所述第一应用的子图层对象创建第二服务端图层对象，并建立所述第二服务端图层对象与所述第一服务端图层对象间的关联关系。

8.如权利要求 7 所述的方法，其特征在于，建立所述第二服务端图层对象与所述第一服务端图层对象间的关联关系，包括：
10

将所述第二服务端图层对象对应的子图层对象的名称，设置为与所述第一服务端图层对象对应的子图层对象的名称相同。

9.如权利要求 7 所述的方法，其特征在于，还包括：

所述第一客户端获得所述第一服务端图层对象对应的子图层对象的名称，并将所述名称携带于所述通知中。
15

10.如权利要求 9 所述的方法，其特征在于，所述第一客户端获得所述第一服务端图层对象对应的子图层对象的名称，包括：

所述第一客户端接收所述图形合成器在创建完成所述第一服务端图层对象后返回的所述子图层对象的名称；或者，

20 所述第一客户端通过注册的回调函数获得所述子图层对象的名称，所述回调函数用于在所述图形合成器创建完成所述第一服务端图层对象后将所述第一服务端图层对象对应的子图层对象的名称通知给所述第一客户端。

11.如权利要求 7 至 10 中任一项所述的方法，其特征在于，第一客户端指示图形合成器创建第一服务端图层对象，包括：

25 所述第一客户端通过调用用于创建图层对象的第一库函数模块，为第一应用的图形界面创建第一客户端图层对象，所述第一库函数模块被调用后通知所述图形合成器创建所述第一服务端图层对象。

12.如权利要求 7 至 10 中任一项所述的方法，其特征在于，所述第二客户端根据所述通知，指示所述图形合成器创建第二服务端图层对象，包括：

30 所述第二客户端根据所述通知，通过调用用于创建图层对象的第二库函数模块，为

所述图形界面创建第二客户端图层对象，所述第二库函数模块被调用后通知所述图形合成器创建所述第二服务端图层对象。

13.如权利要求 7 所述的方法，其特征在于，所述第一子系统为 Host 子系统，所述第二子系统为 Guest 子系统。

5 14.一种图层处理方法，其特征在于，包括：

图形合成器根据第一客户端的指示，针对第一应用的子图层对象创建第一服务端图层对象；其中，所述第一客户端为所述第一应用在第一子系统客户端；

所述图形合成器根据第二客户端的指示，针对所述第一应用的子图层对象创建第二服务端图层对象；其中，所述指示是所述第二客户端根据所述第一客户端发送的通知向
10 所述图形合成器发送的；其中，所述第二客户端为所述第一应用在第二子系统客户端；

所述图形合成器建立所述第二服务端图层对象与所述第一服务端图层对象间的关联关系。

15.如权利要求 14 所述的方法，其特征在于，所述图形合成器建立所述第二服务端
15 图层对象与所述第一服务端图层对象间的关联关系，包括：

所述图形合成器将所述第二服务端图层对象对应的子图层对象的名称，设置为与所述第一服务端图层对象对应的子图层对象的名称相同。

16.如权利要求 14 或 15 所述的方法，其特征在于，还包括：

所述图形合成器从所述第二客户端获得所述第一服务端图层对象对应的子图层对象的名称；其中，所述子图层对象的名称是所述第一客户端携带于通知中发送给所述第二客户端的。
20

17.如权利要求 14 或 15 所述的方法，其特征在于，所述第一子系统为 Host 子系统，所述第二子系统为 Guest 子系统。

18.一种图形合成器，其特征在于，所述图形合成器为操作系统中的第一子系统和第二子系统共享使用的图形合成器，所述图形合成器包括：
25

请求获取模块，用于获取第一图形合成请求，所述第一图形合成请求对应于第二系统中的第二服务端图层对象；

关联对象获取模块，用于根据所述第一图形合成请求所对应的第二服务端图层对象，获取所述第二服务端图层对象所关联的第一子系统的第一服务端图层对象；其中，
30 所述第一服务端图层对象和所述第二服务端图层对象是针对第一应用的子图层对象创

建的；

合成模块，用于获取所述第二服务端图层对象所对应的缓存，基于所述第一服务端图层对象并使用所述缓存中存储的与所述第一应用的子图层对象对应的图形界面内容数据，响应所述第一图形合成请求。

- 5 19.如权利要求 18 所述的图形合成器，其特征在于，所述关联对象获取模块具体用于：

根据所述第二服务端图层对象对应的子图层对象的名称，获取对应于相同子图层对象名称的第一服务端图层对象。

- 20.如权利要求 18 或 19 所述的图形合成器，其特征在于，所述第一图形合成请求由
10 第二客户端基于第二客户端图层对象对所述第一应用的图形界面渲染完成后生成，所述第二客户端图层对象与所述第二服务端图层对象对应；其中，所述第二客户端为所述第一应用在第二子系统客户端。

- 21.如权利要求 18 或 19 所述的图形合成器，其特征在于，

- 所述请求获取模块还用于：获取第二图形合成请求，所述第二图形合成请求对应于
15 所述第一服务端图层对象；

所述合成模块还用于：根据所述第一服务端图层对象所对应的子图层对象，获取该子图层对象对应的第一应用图形界面窗口状态数据；基于所述第一服务端图层对象，并根据所述第一应用图形界面窗口状态数据，响应所述第二图形合成请求。

- 22.如权利要求 21 所述的图形合成器，其特征在于，所述第二图形合成请求由第一
20 客户端基于第一客户端图层对象对所述第一应用的图形界面的窗口状态进行更新的操作生成，所述第一客户端图层对象与所述第一服务端图层对象对应；其中，所述第一客户端为所述第一应用在第一子系统客户端。

- 23.一种图层处理装置，其特征在于，包括：

- 第一客户端，用于指示图形合成器针对第一应用的子图层对象创建第一服务端图层
25 对象，并向第二客户端发送通知；其中，所述第一客户端为所述第一应用在第一子系统客户端，所述第二客户端为所述第一应用在所述第二子系统客户端；

第二客户端，用于根据所述通知，指示所述图形合成器针对所述第一应用的子图层对象创建第二服务端图层对象，并建立所述第二服务端图层对象与所述第一服务端图层对象间的关联关系。

- 30 24.如权利要求 23 所述的装置，其特征在于，所述第二客户端具体用于：

将所述第二服务端图层对象对应的子图层对象的名称, 设置为与所述第一服务端图层对象对应的子图层对象的名称相同。

25. 如权利要求 23 所述的装置, 其特征在于, 所述第一客户端还用于:

5 获得所述第一服务端图层对象对应的子图层对象的名称, 并将所述名称携带于所述通知中。

26. 如权利要求 25 所述的装置, 其特征在于, 所述第一客户端具体用于获得所述第一服务端图层对象对应的子图层对象的名称, 包括:

接收所述图形合成器在创建完成所述第一服务端图层对象后返回的所述子图层对象的名称; 或者,

10 通过注册的回调函数获得所述子图层对象的名称, 所述回调函数用于在所述图形合成器创建完成所述第一服务端图层对象后将所述第一服务端图层对象对应的子图层对象的名称通知给所述第一客户端。

27. 如权利要求 23 至 26 中任一项所述的装置, 其特征在于, 第一客户端具体用于: 通过调用用于创建图层对象的第一库函数模块, 为第一应用的图形界面创建第一客户端
15 图层对象, 所述第一库函数模块被调用后通知所述图形合成器创建所述第一服务端图层对象。

28. 如权利要求 23 至 26 中任一项所述的装置, 其特征在于, 所述第二客户端具体用于:

根据所述通知, 通过调用用于创建图层对象的第二库函数模块, 为所述图形界面创
20 建第二客户端图层对象, 所述第二库函数模块被调用后通知所述图形合成器创建所述第二服务端图层对象。

29. 一种图形合成器, 其特征在于, 所述图形合成器为操作系统中的第一子系统和第二子系统共享使用的图形合成器, 所述图形合成器包括:

第一图层对象创建模块, 用于根据第一客户端的指示, 针对第一应用的子图层对象
25 创建第一服务端图层对象; 以及, 根据第二客户端的指示, 针对所述第一应用的子图层对象创建第二服务端图层对象, 所述指示是所述第二客户端根据所述第一客户端发送的通知向所述图形合成器发送的; 其中, 所述第一客户端为所述第一应用在第一子系统
中的客户端, 所述第二客户端为所述第一应用在第二子系统
中的客户端;

30 关联模块, 用于建立所述第二服务端图层对象与所述第一服务端图层对象间的关联关系。

30.如权利要求 29 所述的图形合成器，其特征在于，所述关联模块具体用于：

将所述第二服务端图层对象对应的子图层对象的名称，设置为与所述第一服务端图层对象对应的子图层对象的名称相同。

31.如权利要求 29 或 30 所述的图形合成器，其特征在于，所述关联模块还用于：从
5 所述第二客户端获得所述第一服务端图层对象对应的子图层对象的名称；其中，所述子图层对象的名称是所述第一客户端携带于通知中发送给所述第二客户端的。

32.一个或多个计算机可读介质，其特征在于，所述可读介质上存储有指令，所述指令被一个或多个处理器执行时，使得通信设备执行如权利要求 1-6 中任一项所述的方法。

33.一个或多个计算机可读介质，其特征在于，所述可读介质上存储有指令，所述指令
10 被一个或多个处理器执行时，使得通信设备执行如权利要求 7-13 中任一项所述的方法。

34.一个或多个计算机可读介质，其特征在于，所述可读介质上存储有指令，所述指令被一个或多个处理器执行时，使得通信设备执行如权利要求 14-17 中任一项所述的方法。

15 35.一种图层处理装置，其特征在于，包括：

一个或多个处理器；以及

一个或多个计算机可读介质，所述可读介质上存储有指令，所述指令被所述一个或多个处理器执行时，使得所述装置执行如权利要求 1-6 中任一项所述的方法。

36.一种图层处理装置，其特征在于，包括：

20 一个或多个处理器；以及

一个或多个计算机可读介质，所述可读介质上存储有指令，所述指令被所述一个或多个处理器执行时，使得所述装置执行如权利要求 7-13 中任一项所述的方法。

37.一种图层处理装置，其特征在于，包括：

一个或多个处理器；以及

25 一个或多个计算机可读介质，所述可读介质上存储有指令，所述指令被所述一个或多个处理器执行时，使得所述装置执行如权利要求 14-17 中任一项所述的方法。

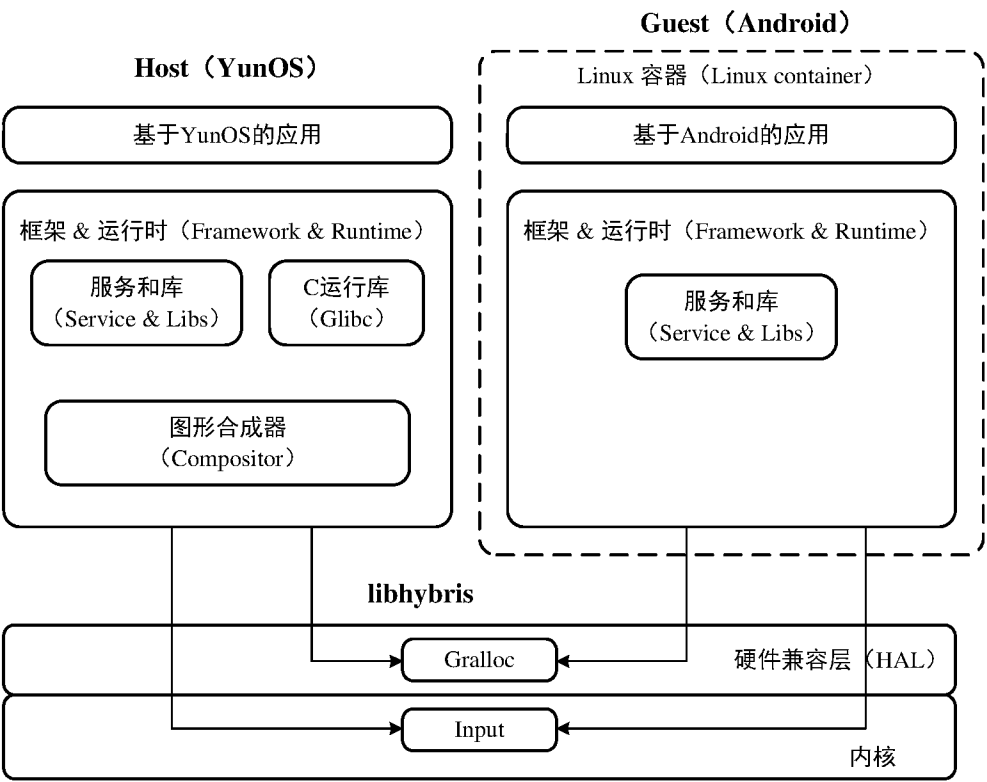


图 1

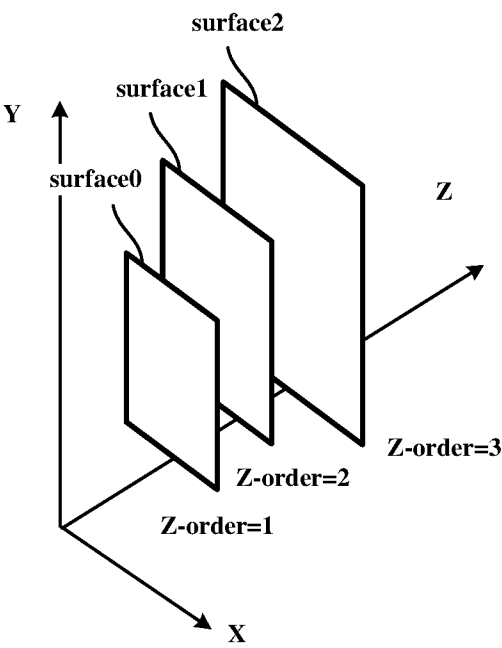


图 2

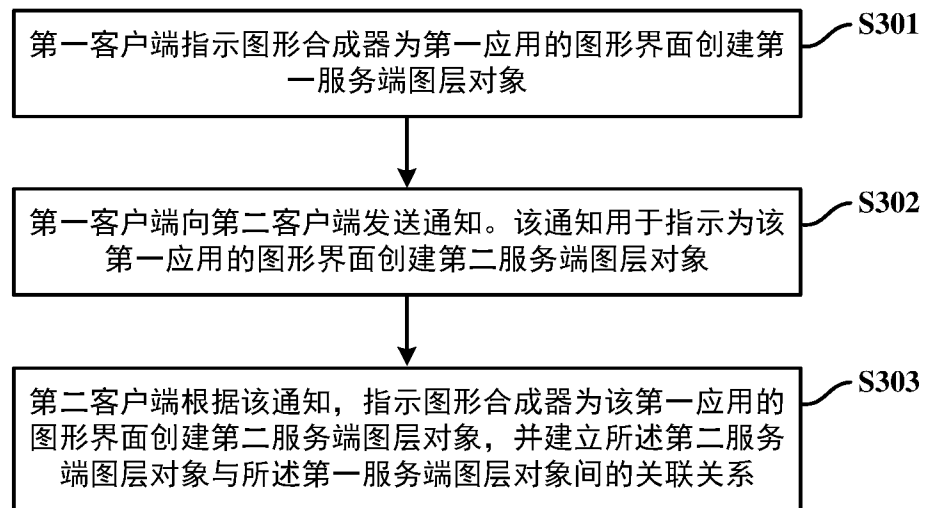


图 3

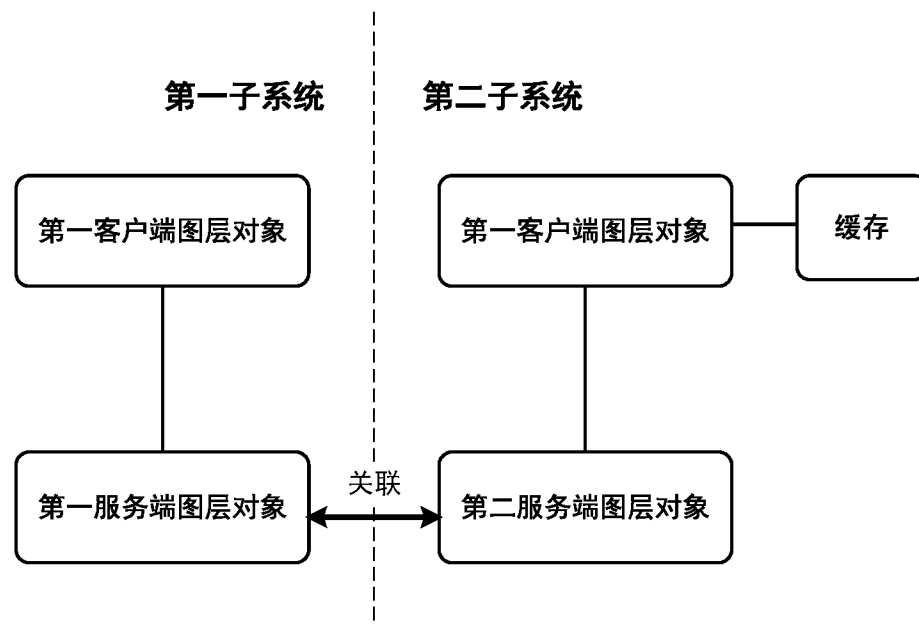


图 4

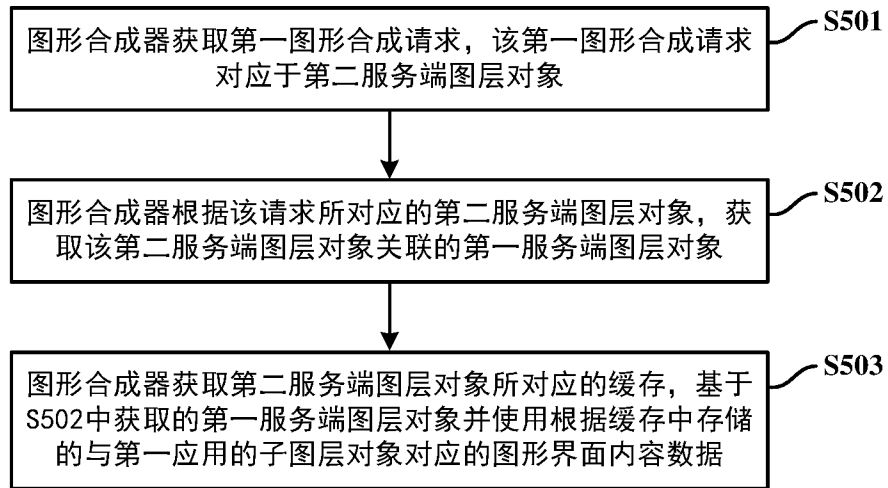


图 5

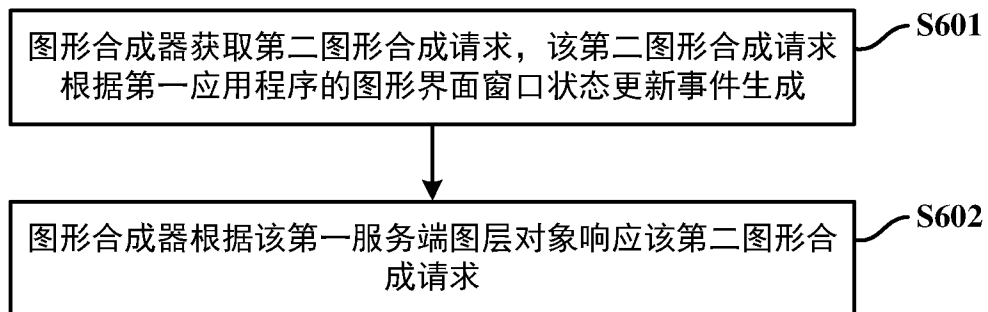


图 6

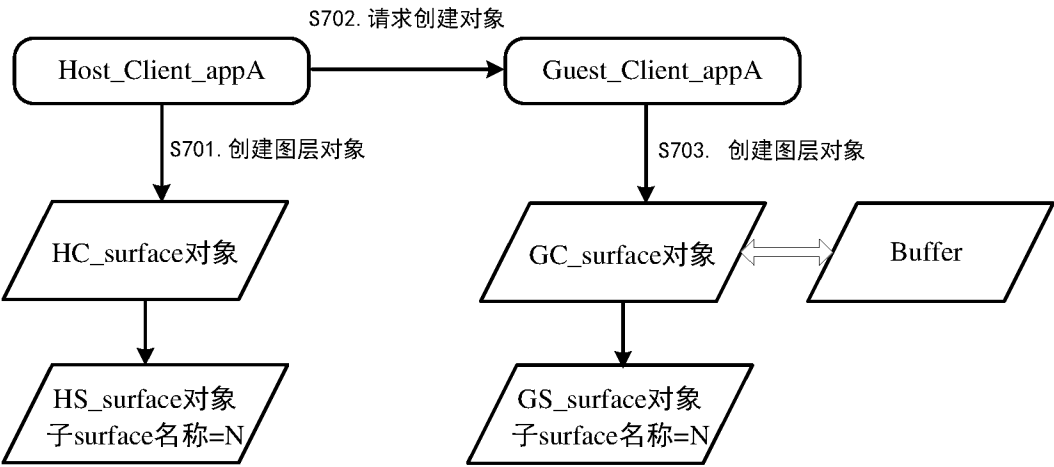


图 7

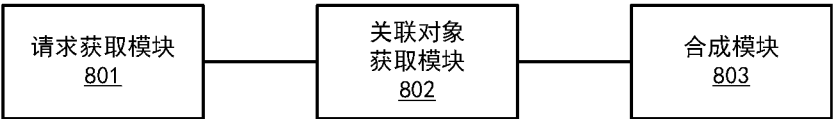


图 8

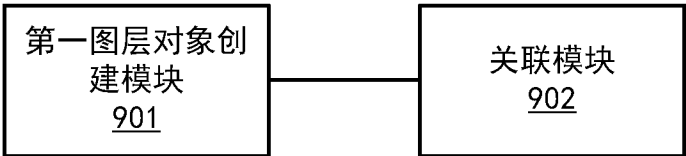


图 9

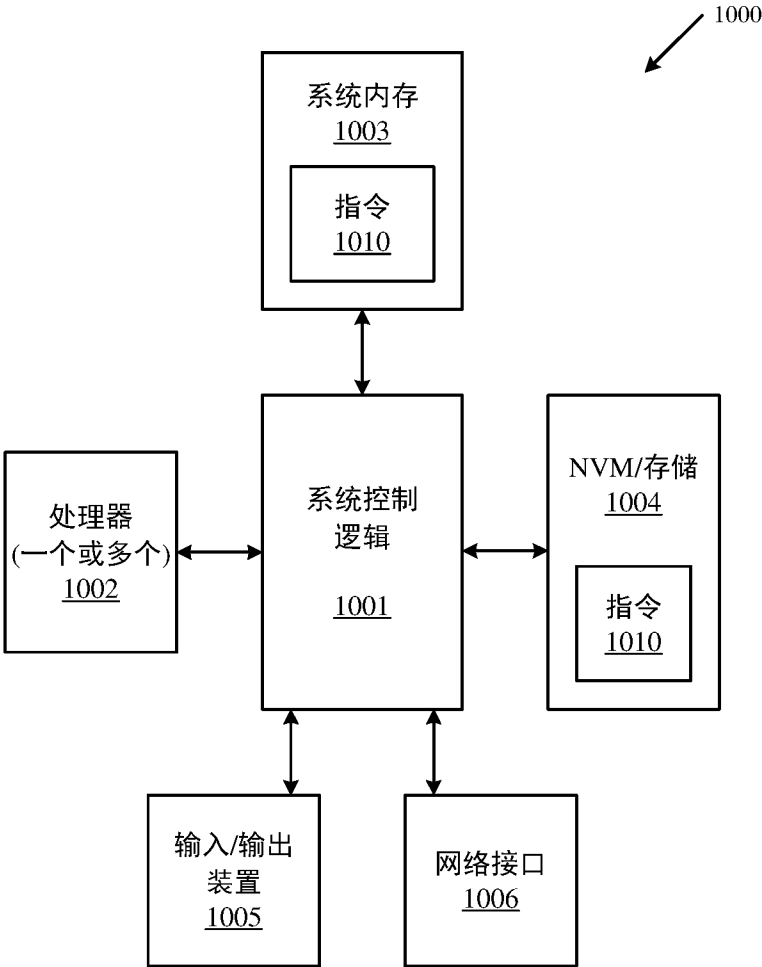


图 10

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2018/092625

A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/44(2018.01)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)

CNABS; CNTXT; CNKI; DWPI; SIPOABS: 阿里巴巴, 金德才, 金卓军, 图层, 图, 层, 界面, 合成, 生成, 关联, 联系, 相关, 第一, 第二, 系统, 主, 辅, 云, surface, layer, composit+, compos+, host, guest, image, picture, yunos, yun, OS, first, android, second, system, conjunction, associat+, correlation

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	CN 101820524 A (ZTE CORPORATION) 01 September 2010 (2010-09-01) entire document	1-37
A	CN 106201398 A (LE HOLDINGS (BEIJING) CO., LTD. ET AL.) 07 December 2016 (2016-12-07) entire document	1-37
A	US 2009225093 A1 (JOHN, H. ET AL.) 10 September 2009 (2009-09-10) entire document	1-37

☐ 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

13 September 2018

Date of mailing of the international search report

26 September 2018

Name and mailing address of the ISA/CN

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

Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2018/092625

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
CN	101820524	A	01 September 2010	WO	2011116611	A1	29 September 2011
CN	106201398	A	07 December 2016	None			
US	2009225093	A1	10 September 2009	WO	2009111045	A1	11 September 2009
				EP	2250558	A1	17 November 2010
				US	8477143	B2	02 July 2013

国际检索报告

国际申请号

PCT/CN2018/092625

A. 主题的分类

G06F 9/44(2018.01) 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))

CNABS;CNTXT;CNKI;DWPI;SIPOABS: 阿里巴巴, 金德才, 金卓军, 图层, 图, 层, 界面, 合成, 生成, 关联, 联系, 相关, 第一, 第二, 系统, 主, 辅, 云, surface, layer, composit+, compos+, host, guest, image, picture, yunos, yun, OS, first, android, second, system, conjunction, associat+, correlation

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 101820524 A (中兴通讯股份有限公司) 2010年 9月 1日 (2010 - 09 - 01) 全文	1-37
A	CN 106201398 A (乐视控股北京有限公司等) 2016年 12月 7日 (2016 - 12 - 07) 全文	1-37
A	US 2009225093 A1 (HARPER JOHN等) 2009年 9月 10日 (2009 - 09 - 10) 全文	1-37

☐ 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 9月 13日

国际检索报告邮寄日期

2018年 9月 26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

传真号 (86-10)62019451

受权官员

孙蕾

电话号码 86-(010)-62412294

国际检索报告
关于同族专利的信息

国际申请号

PCT/CN2018/092625

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	101820524	A	2010年 9月 1日	WO	2011116611	A1	2011年 9月 29日
CN	106201398	A	2016年 12月 7日	无			
US	2009225093	A1	2009年 9月 10日	WO	2009111045	A1	2009年 9月 11日
				EP	2250558	A1	2010年 11月 17日
				US	8477143	B2	2013年 7月 2日

表 PCT/ISA/210 (同族专利附件) (2015年1月)