



(12) 发明专利

(10) 授权公告号 CN 113168422 B

(45) 授权公告日 2025. 03. 14

(21) 申请号 201980077673.8

(22) 申请日 2019.10.15

(65) 同一申请的已公布的文献号
申请公布号 CN 113168422 A

(43) 申请公布日 2021.07.23

(30) 优先权数据
201841039490 2018.10.18 IN
16/566,490 2019.09.10 US
16/566,504 2019.09.10 US
16/566,511 2019.09.10 US

(85) PCT国际申请进入国家阶段日
2021.05.26

(86) PCT国际申请的申请数据
PCT/US2019/056364 2019.10.15

(87) PCT国际申请的公布数据
W02020/081587 EN 2020.04.23

(73) 专利权人 甲骨文国际公司

地址 美国加利福尼亚

(72) 发明人 S·N·安德拉比 R·莫迪
V·穆哈拉姆 M·帕拉尼萨米
M·黄

(74) 专利代理机构 中国贸促会专利商标事务所
有限公司 11038
专利代理师 冯薇

(51) Int.Cl.
G06F 16/908 (2006.01)
G06F 16/9035 (2006.01)

(56) 对比文件
US 2016358354 A1, 2016.12.08

审查员 王雨宁

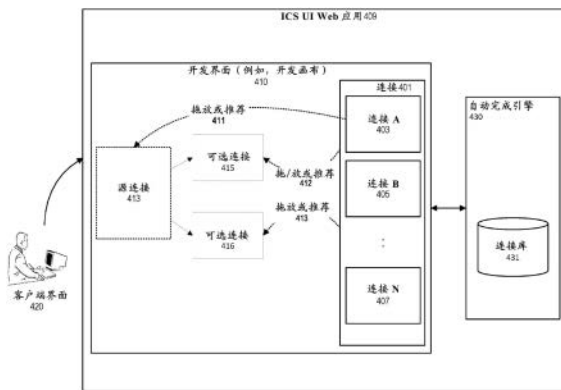
权利要求书2页 说明书30页 附图14页

(54) 发明名称

使用人工智能/机器学习自动完成ICS流程的系统和方法

(57) 摘要

根据实施例,本文描述了使用人工智能/机器学习自动完成ICS流程的系统和方法。下一动作预测是通过预测和建议用户可能正在考虑添加的下一组动作来帮助用户对流程进行快速建模的服务。该服务还帮助用户在创建集成流程时遵循一些最佳实践。



1. 一种用于使用人工智能/机器学习支持自动完成集成云服务ICS流程的系统,包括:
包括一个或多个微处理器的计算机;
在所述计算机上运行的集成平台,所述集成平台包括集成流程设计器,该集成流程设计器能够访问包括多个对象的库,每个对象能够在所述集成流程设计器内设计的集成流程内被利用;
其中所述计算机被配置为执行以下步骤:
启动所述集成流程设计器内的集成流程的设计模板;
收集用户上下文,所述用户上下文与所述集成流程设计器内的集成流程相关;以及
经由所述集成平台提供的用户界面,基于所收集的用户上下文来提供多个可选择流程预测,所述多个可选择流程预测中的每一个包括按顺序排列的连接器,每个连接器与所述集成流程设计器内的集成流程外部的应用的实例相关联。
2. 如权利要求1所述的系统,其中用户上下文考虑组织、子公司、部门、子部门和用户信息。
3. 如权利要求1或2所述的系统,其中在提供所述多个可选择流程预测时,利用处理模型的不变模式识别的分层聚类模型。
4. 如权利要求3所述的系统,其中所述分层聚类模型利用存储的处理设计的机器学习知识作为聚类的层次结构。
5. 如权利要求4所述的系统,其中所述分层聚类模型还基于所收集的用户上下文来利用机器学习模型。
6. 如权利要求1所述的系统,其中在提供所述多个可选择流程预测时使用排名生成器。
7. 如权利要求6所述的系统,其中所述排名生成器利用输入模式来对与所述输入模式匹配的多个输出模式进行排名,所述排名利用一个或多个输入;以及
其中所述输入模式是部分根据所收集的用户上下文创建的。
8. 一种用于使用人工智能/机器学习支持自动完成集成云服务ICS流程的方法,包括:
提供包括一个或多个微处理器的计算机;
提供在所述计算机上运行的集成平台,所述集成平台包括集成流程设计器,该集成流程设计器能够访问包括多个对象的库,每个对象能够在所述集成流程设计器内设计的集成流程内被利用;
启动所述集成流程设计器内的集成流程的设计模板;
收集用户上下文,所述用户上下文与所述集成流程设计器内的集成流程相关;以及
经由所述集成平台提供的用户界面,基于所收集的用户上下文来提供多个可选择流程预测,所述多个可选择流程预测中的每一个包括按顺序排列的连接器,每个连接器与所述集成流程设计器内的集成流程外部的应用的实例相关联。
9. 如权利要求8所述的方法,其中用户上下文考虑组织、子公司、部门、子部门和用户信息。
10. 如权利要求8或9所述的方法,其中在提供下一动作预测时,利用处理模型的不变模式识别的分层聚类模型。
11. 如权利要求10所述的方法,其中所述分层聚类模型利用存储的处理设计的机器学习知识作为聚类的层次结构。

12. 如权利要求11所述的方法,其中所述分层聚类模型还基于所收集的用户上下文来利用机器学习模型。

13. 如权利要求8所述的方法,其中在提供下一动作预测时使用排名生成器。

14. 如权利要求13所述的方法,其中所述排名生成器利用输入模式来对与所述输入模式匹配的多个输出模式进行排名,所述排名利用一个或多个输入;以及其中所述输入模式是部分根据所收集的用户上下文创建的。

15. 一种非暂态计算机可读存储介质,具有存储在其上的用于使用人工智能/机器学习支持自动完成集成云服务ICS流程的指令,所述指令在由一个或多个计算机读取和执行时,使所述一个或多个计算机执行包括以下的步骤:

提供包括一个或多个微处理器的计算机;

提供在所述计算机上运行的集成平台,所述集成平台包括集成流程设计器,该集成流程设计器能够访问包括多个对象的库,每个对象能够在所述集成流程设计器内设计的集成流程内被利用;

启动所述集成流程设计器内的集成流程的设计模板;

收集用户上下文,所述用户上下文与所述集成流程设计器内的集成流程相关;以及

经由所述集成平台提供的用户界面,基于所收集的用户上下文来提供多个可选择流程预测,所述多个可选择流程预测中的每一个包括按顺序排列的连接器,每个连接器与所述集成流程设计器内的集成流程外部的应用的实例相关联。

16. 如权利要求15所述的非暂态计算机可读存储介质,其中用户上下文考虑组织、子公司、部门、子部门和用户信息。

17. 如权利要求15或16所述的非暂态计算机可读存储介质,其中在提供下一动作预测时,利用处理模型的不变模式识别的分层聚类模型。

18. 如权利要求17所述的非暂态计算机可读存储介质,其中所述分层聚类模型利用存储的处理设计的机器学习知识作为聚类的层次结构。

19. 如权利要求18所述的非暂态计算机可读存储介质,其中所述分层聚类模型还基于所收集的用户上下文来利用机器学习模型。

20. 如权利要求15所述的非暂态计算机可读存储介质,

其中排名生成器用于提供下一动作预测;

其中所述排名生成器利用输入模式对与所述输入模式匹配的多个输出模式进行排名,所述排名利用一个或多个输入;以及

其中所述输入模式是部分根据所收集的用户上下文创建的。

21. 一种包括指令的计算机程序产品,所述指令在由计算机的一个或多个处理器执行时,使得所述计算机执行如权利要求8-14中任一项所述的方法。

使用人工智能/机器学习自动完成ICS流程的系统和方法

[0001] 版权声明

[0002] 本专利文档的公开内容的一部分包含受版权保护的素材。版权拥有者不反对任何人对专利文档或专利公开内容按照其在专利商标局的专利文件或记录中出现的那样进行传真复制,但是除此之外在任何情况下都保留所有版权。

[0003] 优先权要求:

[0004] 本申请要求以下专利申请的优先权:于2019年9月10日提交的标题为“SYSTEM AND METHOD FOR AUTO-COMPLETION OF ICS FLOW USING ARTIFICIAL INTELLIGENCE/MACHINE LEARNING”、申请号为16/566,490的美国专利申请;于2019年9月10日提交的标题为“SYSTEM AND METHOD FOR NEXT STEP PREDICTION FOR ICS FLOW USING ARTIFICIAL INTELLIGENCE/MACHINE LEARNING”、申请号为16/566,504的美国专利申请;于2019年9月10日提交的标题为“SYSTEM AND METHOD FOR NEXT OBJECT PREDICTION FOR ICS FLOW USING ARTIFICIAL INTELLIGENCE/MACHINE LEARNING”、申请号为16/566,511的美国专利申请;以及于2018年10月18日提交的标题为“SYSTEM AND METHOD FOR AUTO-COMPLETION OF ICS FLOW USING ARTIFICIAL INTELLIGENCE/MACHINE LEARNING”、申请号为201841039490的印度专利申请,这些申请通过引用并入本文。

技术领域

[0005] 本发明的实施例一般而言涉及集成云服务,并且特别地,涉及使用人工智能和/或机器学习来自动完成集成云服务中的流程。

背景技术

[0006] 集成云服务(ICS)(例如,Oracle集成云服务)是云中简单而强大的集成平台,该集成平台有助于对产品(诸如软件即服务(SaaS)和内部部署的应用)的利用。ICS可以作为集成平台即服务(iPaas)提供,并且可以包括用于在应用之间进行指向和点击集成的基于web的集成设计器、对交易提供实时洞察的丰富监控仪表盘。

发明内容

[0007] 根据实施例,本文描述了使用人工智能/机器学习自动完成ICS流程的系统和方法。下一(next)动作预测是通过预测和建议用户可能正在考虑添加的下一组动作来帮助用户对流程进行快速建模的服务。该服务还帮助用户在创建集成流程时遵循一些最佳实践。

附图说明

[0008] 图1图示了根据实施例的集成云服务。

[0009] 图2图示了根据实施例的集成云服务。

[0010] 图3图示了根据实施例的ICS设计时。

[0011] 图4图示了根据实施例的用于支持使用AI/ML自动完成ICS流程的系统。

- [0012] 图5图示了根据实施例的示例性流程图决策树。
- [0013] 图6示出了根据实施例的用于在支持系统使用AI/ML自动完成ICS流程中使用的分层(hierarchical)树结构。
- [0014] 图7示出了根据实施例的上下文到集成流程的映射。
- [0015] 图8示出了根据实施例的上下文到集成流程的映射。
- [0016] 图9示出了根据实施例的示例性排名模拟器。
- [0017] 图10示出了使用人工智能/机器学习来支持ICS(集成云服务)流程的自动完成的方法的流程图。
- [0018] 图11图示了根据实施例的示例性流程图决策树。
- [0019] 图12示出了使用人工智能/机器学习对ICS(集成云服务)流程进行下一步预测的示例性方法的流程图。
- [0020] 图13图示了根据实施例的示例性流程图决策树。
- [0021] 图14示出了使用人工智能/机器学习对ICS(集成云服务)流程进行下一对象预测的示例性方法的流程图。

具体实施方式

- [0022] 通过参考所附的说明书、权利要求书和附图,前述以及其它特征将变得显而易见。阐述了具体细节以便提供对各种实施例的理解。但是,将显而易见的是,可以在没有这些具体细节的情况下实践各种实施例。随附的说明书和附图并非旨在进行限制。
- [0023] 集成平台即服务(例如Oracle集成云访问(Oracle Integration Cloud Service)(ICS))可以提供基于云的平台,用于构建和部署连接驻留在云中或内部部署的应用的集成流程。
- [0024] 集成云服务
- [0025] 图1图示了根据实施例的用于设计和执行ICS集成流程的ICS平台。
- [0026] 如图1中所示,ICS平台可以包括设计时(design-time)环境120和运行时环境163。每种环境都可以在包括一个或多个处理器的计算机(例如计算机101或106)上执行。
- [0027] 根据实施例,设计时环境包括ICS web控制台122,该ICS web控制台122提供了基于浏览器的设计器,以允许集成流程开发人员使用客户端界面103来构建集成。
- [0028] 根据实施例,ICS设计时环境可以被预先加载到各种SaaS应用或其它应用的连接,并且可以包括源组件124和目标组件126。源组件可以为一个或多个源应用/对象提供定义和配置;并且目标组件可以为一个或多个目标应用/对象提供定义和配置。定义和配置可以用于识别应用类型、端点、集成对象和应用/对象的其它细节。
- [0029] 如图1中进一步所示,设计时环境可以包括用于将传入消息的内容映射到传出消息的映射/变换组件128,以及用于基于消息的内容或报头(header)信息来控制将哪些消息路由到哪些目标的消息路由组件130。此外,设计时环境可以包括用于基于消息内容或消息的报头信息来控制要路由哪些消息的消息过滤组件132,以及用于将相关但乱序(out-of-sequence)的消息的流重新布置回用户指定的顺序的消息排序组件134。
- [0030] 根据实施例,与源组件和目标组件一样,上述组件中的每个组件可以包括设计时设置,该设计时设置可以作为流程定义/配置的一部分被持久化。

[0031] 根据实施例,流程定义指定ICS集成流程的细节;并且涵盖集成流程的静态构造(例如,消息路由器)和可配置方面(例如,路由规则)两者。完全配置的流程定义和其它必需的工件(例如,jca和.wsdl文件)结合在一起可以被称为ICS项目。ICS项目可以完全定义集成流程,并且可以由底层实现层实现。

[0032] 根据实施例,策略组件136可以包括管理ICS环境的行为的多个策略。例如,可以为源应用的源-拉消息传递交互(即,查询样式集成)配置轮询策略,以经由基于时间的轮询来调用对源应用的出站调用。

[0033] 根据实施例,可以指定其它策略,用于在将消息路由到目标应用时的安全特权;用于在流程执行期间记录消息有效负载和报头字段,以经由监视控制台进行后续分析;以及用于消息节流,该消息节流用于定义企业服务总线(ESB)服务可以生成以容纳请求的多个实例。另外,可以指定策略,用于在流程级别监视/跟踪集成流程;以及用于根据已知方案验证由ICS平台正在处理的消息。

[0034] 根据实施例,集成开发人员可以将组件拖放到开发画布133上以进行编辑和配置,用于在设计集成流程时使用。

[0035] 如进一步所示,运行时环境可以在企业服务总线组件172的顶部包括应用服务器162、ICS运行时引擎166、存储服务168和消息传递服务170。用户界面控制台164可以用于监视和跟踪运行时环境的性能。

[0036] 图2图示了根据实施例的集成云服务。

[0037] 如图2中所示,ICS 207可以提供基于云的集成服务,用于设计、执行和管理ICS集成流程。ICS可以包括web应用209和在企业云环境(例如,Oracle公共云)201中的应用服务器217上执行的ICS运行时215。web应用可以提供设计时,该设计时暴露(expose)了多个用户界面供用户设计、激活、管理和监视ICS集成流程。可以在ICS运行时上部署和执行激活的ICS集成流程。

[0038] 根据实施例,可以通过处理连接到多个应用的底层复杂性来提供多个应用适配器213以简化配置到这些应用的连接的任务。该应用可以包括ICS供应商205的企业云应用、第三方云应用(例如,Salesforce)103和内部部署的应用219。ICS可以向这些应用公开简单对象访问协议(SOAP)和表示状态传输(REST)端点,以在与这些应用进行通信时使用。

[0039] 根据实施例,ICS集成流程(或ICS集成)可以包括源连接、目标连接以及两个连接之间的字段映射。每个连接可以基于应用适配器,并且可以包括应用适配器与应用的特定实例进行通信所需的附加信息。

[0040] 根据实施例,可以将ICS集成流程和多个其它所需的工件(例如,JCA和WSDL文件)编译为ICS项目,该ICS项目可以在ICS运行时中部署和执行。可以使用web UI应用创建多种不同类型的集成流程模式,包括数据映射集成流程、发布集成流程和订阅集成流程。要创建数据映射集成流程,ICS用户可以使用应用适配器或应用连接在开发界面中定义源应用和目标应用,并定义源应用和目标应用之间的路由路径和数据映射。在发布集成流程中,可以将源应用或服务配置为通过预定义的消息传递服务将消息发布到ICS。在订阅集成流程中,可以将目标应用或服务配置为通过消息传递服务订阅来自ICS的消息。

[0041] 图3图示了根据实施例的ICS设计时。

[0042] 根据实施例,用户320可以使用web UI应用中的开发界面(例如,开发画布)310使

用多个现有连接301(例如,连接A 303、连接B 305和连接N 307)来创建ICS集成流程。

[0043] 如图3中所示,可以将特定连接(例如,连接A)拖放311到开发界面作为源连接313,并且可以将连接N拖放309到开发界面作为目标连接315。源连接可以包括连接到源应用所需的信息,并且可以被ICS用来接收来自源应用的请求。目标连接可以包括连接到目标应用(例如,Salesforce云应用)所需的信息,并且可以被ICS用来将请求发送到目标应用。

[0044] 根据实施例,源连接和目标连接可以被进一步配置为包括附加信息。例如,附加信息可以包括要对与请求相关联的数据执行的操作的类型,以及针对这些操作的对象和字段。

[0045] 根据实施例,一旦配置了源连接和目标连接,就可以启用两个连接之间的映射器,并且可以显示映射器图标(例如,映射器图标A317和映射器图标B 318)以用于打开映射器,使得用户可以为请求消息和响应消息定义如何在源数据对象和目标数据对象之间传递信息。

[0046] 根据实施例,映射器可以为用户提供图形用户界面,以通过将源项目拖动到目标项目上来在源应用和目标应用之间映射项目(例如,字段、属性和元素)。当打开ICS集成流程中的用于请求或响应消息的映射器时,可以使用源连接和目标连接自动加载源数据对象和目标数据对象。

[0047] 根据实施例,可以提供查找以促进映射的创建。如本文中所使用的,查找是用于应用中用于描述相同项目的不同代码和术语的可重用映射。例如,一个应用使用一组特定的代码来描述国家,而另一个应用使用一组不同的代码来描述相同的国家。查找可以用于在不同的应用之间映射这些不同的代码。

[0048] 如上所述,集成流程的开发可能是一项复杂的工作,需要在可以成功部署和执行集成流程之前定义各种组件。集成流程中的一些组件需要被定义,而其它组件则是可选的。使开发过程更加复杂的是,定义可选组件可能导致附加的所需组件,并且取决于定义集成组件的顺序,在开发工作期间的任何时间点所需的组件可能不同。

[0049] 使用AI/ML自动完成ICS流程

[0050] 根据实施例,下一动作预测是通过预测和建议用户可能正在考虑添加的下一组动作来帮助客户端对流程进行快速建模的服务。该服务还帮助用户在创建集成流程时遵循一些最佳实践。

[0051] 根据实施例,本文提供的系统和方法包括对ICS系统用户所面临的问题的解决方案。例如,假设用户正在尝试创建“信用卡申请处理”流程。对于专家用户,任何应用处理流程都将具有基本的预处理步骤,例如地址和个人详细信息验证等。对于专家用户,这似乎是单调的工作,因为他们必须自己进行同一组流程。对于新手用户,创建这样的流程将给他们带来挑战,因为他们将很难从头开始创建该流程,并且他们在创建集成流程时可能会错过必要的步骤。然后,这可能会导致更耗时的问题,在这种情况下,他们必须咨询同行以了解创建流程的最佳实践、在互联网上阅读/了解业务流程等。

[0052] 根据实施例,本文描述的系统和方法可以通过允许客户以最小的工作量设计解决方案来利用AI/ML改善集成流程。这可以使得使用ICS流程设计器能够仅基于用户上下文;以及用户和处理的上下文来预测用户最有可能将构建的集成流程。例如,银行中的抵押贷款员登录以设计抵押贷款批准流程;可以向抵押贷款员呈现已实施抵押贷款批准流程并具

有生产价值的ICS流程列表。预测数据(包括工作流程和功能流程)可以被存储在可访问的位置。

[0053] 根据实施例,本文描述的系统和方法可以通过向当前正在对流程建模的用户提供实时提示来减少创建流程所需的时间量。另外,本文描述的系统和方法还可以允许降低关于流程的错误率、增加流程的性能,以及与现有机制相比在ICS内的流程的整体优化。

[0054] 图4图示了根据实施例的用于支持使用AI/ML自动完成ICS流程的系统。

[0055] 根据实施例,用户可以使用web UI应用中的开发界面(例如,开发画布)410使用多个连接401(例如,连接A 403、连接B 405和连接N 407)来创建ICS集成流程。

[0056] 如图4中所示,可以将特定连接(例如,连接A)拖放411到开发界面作为源连接413。替代地,取决于来自客户端接口420的上下文,可以推荐411源连接413并且基于与自动完成引擎430的交互来填充源连接413,该自动完成引擎430与连接库431相关联/通信。

[0057] 根据实施例,取决于从客户端接口接收到的上下文,开发界面可以允许用户从现有连接库中拖放(412、413)不同的连接,或者开发界面可以呈现填充的可以经由用户界面被选择的可选连接(415、416)的网格。根据实施例,上下文可以与源连接相关联,源连接可以包括连接到源应用所需的信息,并且可以被ICS用于接收来自源应用的请求。

[0058] 根据实施例,源连接和目标连接可以被进一步配置为包括附加信息。例如,附加信息可以包括要对与请求相关联的数据执行的操作的类型,以及针对这些操作的对象和字段。

[0059] 根据实施例,可以基于从客户端界面接收到的选择来实时更新经由客户端界面显示的推荐连接。另外,可以基于要建模的流程的上下文来提供推荐的连接。随着流程的增长,与流程相关联的上下文也随之增长。基于上下文,实时建议可以被更改为更细粒度和更准确。

[0060] 根据实施例,在选择实时推荐连接之一后,本文描述的系统和方法可以自动配置流程的选择部分。如果期望,可以覆盖这种自动配置。另外,基于随后对推荐连接的选择,ICS流程的其它推荐部分可以被更改、更新、重新配置、删除或添加...等等。

[0061] 根据实施例,可以简化创建集成流程的处理,因为所描述的自动完成可以将基本步骤聚集在一起,然后使用户自由地将精力集中在流程的更敏感的其它部分上。另外,随着上下文的生长,所描述的系统和方法可以基于流程的上下文自动添加到流程期望或必要的步骤。

[0062] 根据实施例,ICS设计器可以基于用户上下文来呈现推荐的ICS流程。用户上下文考虑组织、子公司(如果适用)、部门、子部门(如果适用)以及用于生成上下文的用户信息;上下文中最重要的是用户的工作描述。设计器可以将ICS流程(包括所有相关文件和连接器)呈现为可以被预览的单个虚拟项目。如果推荐的ICS流程满足用户要求;他们可以选择该ICS流程;将创建用户可以预览包括对其进行测试的虚拟项目。然后,如果项目符合要求(例如,如果流程经过认证且工作),那么该项目可以被保存到物理实体中。

[0063] 在本文描述的实施例中,流程文件可以被称为JSON或JSON文件。本领域技术人员将容易理解,本文描述的系统和方法可以利用不同或替代的文件格式并实现相同或相似的结果。

[0064] 模式/流程预测的相似性得分

[0065] 根据实施例,本文的系统和方法可以将模式识别模型机制用于JSON模型,以便预测应用流程,诸如ICS流程。在这个模型中,可以生成相似性得分以指示输入/源JSON与存储的JSON模式的匹配程度。

[0066] 根据实施例,方法/模型可以基于输入/源流程文件来识别结构上和语义上相似的流程文件模式。可以通过比较两个或更多个流程文件(例如,两个或更多个JSON)来确定结构和语义上的相似性,以确定文件的每个字段的相似程度。对于流程文件内的串字段,可以使用自然语言处理来确定单词或句子的相似程度。对于数值和日期字段,系统和方法可以确定它们彼此之间的偏离程度。可以为每个字段生成相似性得分。

[0067] 根据实施例,流程文件(诸如JSON文件)可以由对象、数组、以及串、数值和布尔类型的原始字段组成。JSON模式识别机制基于为每个字段生成相似性得分,进而将其整合为对象和数组的复合相似性得分,并且最终整合为根JSON对象或数组的相似性得分。

[0068] 根据实施例,使用键和值两者将每个源JSON原始字段与目标JSON字段进行比较。可以使用自然语言处理(NLP)来比较键,而值可以基于其类型来进行比较。即,例如,在计算数值字段距离的同时使用串字段NLP。字段比较可以导致生成相似性得分。例如,考虑采购订单的目标JSON,如下所示:

```
{
  "name"    : "John Smith",
  "sku"     : "20223",
  "price"   : 23.95,
  "shipTo" : {
    "name" : "Jane Smith",
    "address" : "123 Maple Street",
    "city" : "Pretendville",
    "state" : "NY",
    [0069] "zip" : "12345"
  },
  "billTo" : {
    "name" : "John Smith",
    "address" : "123 Maple Street",
    "city" : "Pretendville",
    "state" : "NY",
    "zip" : "12345"
  }
}
```

[0070] 根据实施例,本文描述的系统和方法可以从JSON购买订单中获取第一串,并针对输入JSON模式进行比较,以便确定相似性得分。例如,输入JSON模式可以包含片段(snippet):

```
[0071] {
[0072]   "name": "Jonn Smith"
[0073] }
```

[0074] 然后,根据实施例,该方法/模型可以基于两个JSON文件的比较来识别结构上和语义上相似的流程文件模式。在上述情况下,由于目标文件中的“name(名称)”字段与输入/源文件的“名称”字段完全匹配,因此该方法/模型可以产生完美的相似性得分(例如,“1.0”的相似性得分)。类似地,如果输入/源文件包含“title(称号)”字段而不是“name”字段,但值“John Smith”相同,那么该方法/模型可能导致高的相似性得分,但不是完美得分(例如,“0.9”的得分),因为字段“title”是“name”的同义词。

[0075] 根据实施例,可以使用以下公式生成字段(例如,JSON字段)的相似性得分:

$$[0076] \quad \text{字段相似性得分} = \frac{(\text{键相似性得分} + \text{值相似性得分})}{2}$$

[0077] 根据实施例,可以通过找到距目标值的百分比距离来找到数值字段的相似性得分。例如,在查看上述源JSON文件时,如果输入/源JSON片段包含“price (价格):23.95”,那么相似性得分将是完美的1.0,因为不仅自然语言是完美匹配,而且值也是完美匹配。但是,如果输入JSON文件的片段包含“price:22.95”,那么该方法/模型将产生“0.9791”的相似性得分。这可以使用以下公式找到:

[0078] $\text{keyFormula} = \text{NLP Match Similarity Score}$

$$[0079] \quad \text{valueFormula} = \frac{((\text{TargetValue} - \text{abs}(\text{TargetValue} - \text{SourceValue}))}{\text{TargetValue}}$$

$$[0080] \quad \text{fieldFormula} = \frac{(\text{keyFormula} + \text{valueFormula})}{2}$$

[0081] 根据实施例,如果输入/源代码片段包括“price:22.95”,那么上述“keyFormula”由于自然语言处理是完全匹配而导致1.0的得分。但是,值公式的得分为.9582,这意味着整个字段的相似性得分为(1+.9582)/2,从而得到.9791的相似性得分。

[0082] 根据实施例,本文描述的模型/方法可以支持字段对象类型模式识别(例如,JSON字段对象类型模式识别)。JSON对象类型模式识别可以基于从各个字段的相似性得分生成汇总平均相似性得分。例如,输入流程(例如,JSON流程)模式可以包含片段:

```
{
  "name"   : "Jonn Smith"
[0083]   "price" : 23.95
}
```

[0084] 根据实施例,由于“name”字段的字段相似性得分以及字段公式得分均是完美的1.0,因此上述示例所生成的相似性得分将是完美的1.0。

[0085] 根据实施例,例如,如果输入流程(例如,JSON流程)的片段不是自然语言处理的精确匹配,而是例如同义词,那么可以降低相似性得分。作为示例,假设输入流程中的片段包括:

```
{
  "title"  : "Jonn Smith"
[0086]   "price" : 23.95
}
```

[0087] 根据实施例,然后相似性得分将从1.0减少,因为根据自然语言处理(即,“key similarity score (键相似性得分)”字段)和值(即“price”字段和值)两者,输入流程的两个分段(segment)中的仅一个是精确匹配。另一个值包括“title”而不是“name”。但是,由于“title”是目标流程的同义词,因此相似性得分将不会显著降低。以上面的示例为例,此片段的总相似性得分可以为0.975。这可以从以下公式得出:

$$[0088] \quad \text{相似性得分} = \frac{(\text{键相似性得分} + \text{值相似性得分})}{2}$$

[0089] 根据实施例,如果“title”被给予0.9的键相似性得分,那么以上片段的总相似性得分可以被计算为:

$$[0090] \quad \text{相似性得分} = \frac{\left(\frac{(0.9+1)}{2} + \frac{(1+1)}{2}\right)}{2} = 0.975$$

[0091] 根据实施例,例如,如果输入流程(例如,JSON流程)的片段不是自然语言处理的精确匹配,而是例如同义词,那么相似性得分可能进一步降低。作为示例,假设输入流程中的片段包括:

```
{
  "title" : "Jonh Smith"
  "cost" : 23.95
}
```

[0093] 根据实施例,然后相似性得分将从1.0降低,因为输入流程的两个片段在自然语言处理方面都不是精确匹配(即,“key similarity score(键相似性得分)”字段)。这两个键字段都是目标流程的同义词。因为“title”是目标流程的同义词,并且“cost(成本)”是目标流程的同义词,因此相似性得分将不会显著降低。以上面的示例为例,此片段的总相似性得分可以为0.95。

[0094] 根据实施例,如果“title”和“cost”被赋予0.9的键相似性得分,那么上述片段的总相似性得分可以被计算为:

$$[0095] \quad \text{相似性得分} = \frac{\left(\frac{(0.9+0.9)}{2} + \frac{(1+1)}{2}\right)}{2} = 0.95$$

[0096] 相似性得分的字段数组类型模式识别

[0097] 根据实施例,本文描述的方法/模型可以利用字段数组类型模式识别。这样的数组类型模式识别(例如,JSON数组类型模式识别)可以基于考虑数组内的偏移量从各个字段的相似性得分生成聚合平均相似性得分。作为示例,假设以下内容包含输入源数组模式:

```
[
  "alpha",
  "bravo",
  "charlie"
]
```

[0099] 另外,假设目标数组模式包括精确匹配,即:

[
 “alpha”,
 [0100] ”bravo”,
 ”charlie”
]

[0101] 根据实施例,然后该数组的相似性得分将是完美的1.0。

[0102] 根据实施例,为了确定数组中的字段的相似性得分,可以使用以下公式:

$$[0103] \quad fieldSimilarityScore = \frac{\left(SimilarityScore + \left((ArrayLength - SourceArrayIndex) - \frac{TargetArrayIndex}{ArrayLength} \right) \right)}{2}$$

其中 ArrayLength =

if (SourceArrayLength > TargetArrayLength) {
 ArrayLength = SourceArrayLength
 [0104] }
 else {
 ArrayLength = TargetArrayLength
 }

[0105] 根据实施例,为了确定字段数组的相似性得分,可以使用以下公式:

$$[0106] \quad SimilarityScore_{Array} = \frac{(field1SimilarityScore + field2SimilarityScore + .. fieldNSimilarityScore)}{\text{字段的数量}}$$

[0107] 根据实施例,例如,如果与上述相同的输入源数组模式与紧接在下面显示的目标源数组模式匹配,那么数组相似性得分将为0.8:

[
 “x-ray”,
 “yankee”,
 [0108] “alpha”,
 ”bravo”,
 ”charlie”
]

[0109]]

[0110] 计算数组的相似性得分:

$$[0111] \quad SimilarityScore_{Array} = \frac{\frac{1 + \frac{(5-2)}{5}}{2} + \frac{1 + \frac{(5-2)}{5}}{2} + \frac{1 + \frac{(5-2)}{5}}{2}}{3} = 0.8$$

[0112] 图5图示了根据实施例的示例性流程图决策树。

[0113] 根据实施例,用户可以选择集成流程的第一部分,该第一部分显示在流程设计器500的非阴影部分501中。从该集成流程的开始,取决于用户的上下文,系统和方法可以向用户呈现多个选项503-507,以进行各种选择。如图所示,选项505当前正在窗口502(当前选择窗口)中显示。可以提供附加的流程推荐作为可以被选择的卡布局。在选择特定流程后,ICS设计器显示整个流程。还可以提供按钮导航。

[0114] 根据实施例,针对流程的推荐处理可能没有被选择(例如,在推荐不是正确推荐的情况下)。在这种情况下,其它处理推荐可以被一致地呈现为如图中阴影区域中的流程中所绘出的“预测平面”内的预测。然后,这提供了选择各个任务或整个流程的选项。该系统和方法还可以呈现卡布局,该卡布局可以针对所有可能的推荐进行轻拂(flick)以完成流程。这可以允许进行导航,以查看每个预测的ICS流程的外观以及其行为。在选择每个选项时,可以实时测试流程。测试可以在最终选择流程之前提供功能和性能简档。

[0115] 根据实施例,可以以多种不同的方式显示502中的流程。流程可以如图所示被显示为完整的流程(已经被测试和合格)。替代地,可以以覆盖的方式(例如,虚线轮廓)来呈现可选流程,其中仅下一组设定数量的流程步骤被显示给用户,直到接收到选择所呈现的选项之一的指令为止。替代地,可选流程可以被显示为已完成的流程,但以覆盖的方式显示,从而允许用户选择已完成的流程的不同部分,以查看可以被选择的不同分支选项。

[0116] 处理模型预测

[0117] 根据实施例,本文的系统和方法提供了处理模型的不变模式识别的分层聚类模型。在这个模型中,学习有关如何设计处理的知识,并将其封装为基于用户和处理上下文的处理模型的聚类的层次结构,其中最准确的聚类是聚类树的叶子。知识以树结构进行编码,该树结构可以基于用户和处理上下文进行查询。

[0118] 根据实施例,重组处理模型是复杂且耗时的工作。系统和方法可以通过匹配较大的现有模式的小片段来识别模式。该系统和方法可以通过基于用户和处理上下文通过处理模型的分层聚类来预测处理模型。在处理上下文上的聚类可以基于识别XML/JSON模式。

[0119] 图6示出了根据实施例的用于在支持系统使用AI/ML自动完成ICS流程中使用的分层树结构。更特别地,图6示出了根据实施例的自设计企业系统。

[0120] 根据实施例,学习和模式识别算法利用该分层结构。这些节点基于上下文对聚类区域进行建模。节点可以具有若干个子代(children)和一个父代(parent)。输入模型基于上下文进入各个级别的节点。在每个级别,节点具有子代,该子代是基于父上下文层次结构的相似模型的聚类。节点的层次结构是基于上下文的;顶层是根上下文,后面跟着可以具有子公司的公司类型上下文,后面跟着部门类型和子部门,后面跟着用户上下文;最后是工作类型上下文。这些树节点中的每一个都被索引。

[0121] 作为示例,结构可以包括可以与多个模型聚类601-603相关联的根上下文600。在根上下文之下可以是公司类型上下文610,该公司类型上下文610可以与多个模型聚类611-613相关联。公司类型上下文之下可以是部门类型620,部门类型620可以与多个模型聚类621-623相关联。在部门类型之下可以是工作类型630,工作类型630可以与多个模型聚类631-633相关联。

[0122] 根据实施例,可以用多个评估和认证的集成流程来预先填充每个模型聚类。这些集成流程中的每一个都可以被存储在例如与自动完成引擎相关联的存储装置中。例如,假

定模型聚类631处理销售点集成流程。然后,被分类为工作类型630的用户在登录并指示销售点流程时,可以被呈现有来自模型聚类631的多个集成流程。

[0123] 根据实施例,模型聚类621还可以与多个评估和认证的集成流程相关联。如果用户拒绝从模型聚类631中选择多个集成流程中的任何一个,那么树可以移回到部门类型620(也根据上下文与用户相关联),并向用户呈现来自模型聚类621的多个流程。只要用户继续拒绝选择任何呈现的流程,就可以继续这种递归情况。

[0124] 根据实施例,多个流程中的每个流程都可以作为完整流程被呈现给用户,或者可以以逐步的方式被呈现给用户,其中每个后续选择都会导致系统和方法呈现从正确的模型聚类631中选择的多个选项。

[0125] 根据实施例,自学习算法可以对图6中所示的分层数据结构进行操作。分层数据结构由已收集的数据初始化,该数据由已在生产中的模型组成,因此确保了数据的质量。在引入更改/添加后,该算法可以自动在层次结构中重建模型聚类。然后,针对特定上下文将新模型投入生产。自学习算法首先基于其句法和语义相似性将模型放入到根模型聚类之一。接下来,基于公司、部门、工作和用户上下文,它更新剩余的层次模型聚类。

[0126] 根据实施例,模式识别算法可以基于通过查看上下文信息并生成可以用于识别当前(发送的)上下文中的相似模型的键来识别发送的输入模式。基于公司、部门、工作和用户上下文与预先计算的模型聚类的匹配程度,识别从更准确到较不准确排名的模型模式;根据公司、部门、工作和用户上下文的组合,从最准确到最不准确对推荐进行进一步排名,即公司、部门、工作和用户上下文,后面跟着公司、部门和工作上下文,后面跟着公司和部门和公司上下文。

[0127] 根据实施例,系统和方法通常不会向用户呈现不太准确的推荐。但是,如果用户没有选择更准确的模型,那么算法可以确定期望的模型更通用。例如;对于只购买与量子力学相关的物理学书籍的物理学家;系统很可能会推荐有关量子力学的书籍。但是,假设物理学家有意阅读哲学,因此他/她拒绝有关量子力学的书籍的推荐。然后,推荐系统从用户上下文中得知该物理学家对哲学感兴趣,因此下一个推荐可以是基于用户对主题的兴趣的哲学书籍。因此,系统首先纯粹基于模式本身进行推荐;用户从未购买过与量子力学无关的书;因此,如果他/她拒绝这个推荐,那么我们呈现给他/她的下一个推荐是基于他/她喜欢的主题,在这种情况下是哲学。下一次同一用户登录时,我们将给出包括一些哲学书籍的推荐。

[0128] 根据实施例,该系统和方法可以基于用户上下文加上考虑模型的结构和语义的模型模式来提供推荐。考虑正在设计的如图7中所示的模型。用户可以将第一个活动创建为接收活动;所产生的推荐将基于用户上下文701加上所设计的处理的结构和语义。该系统和方法可以与具有相似上下文和以接收活动开始的模型加上与处理和接收活动的名称、文档等具有语义相似性的所有聚类705进行匹配。如图所示,仅示出了三个聚类702-704,但是更多的聚类可以被包含在系统内并且经由用户界面来呈现。

[0129] 根据实施例,可以扩展图7的模型,如图8中所示。通过扩展图7的模型;对具有接收和调用任务的模型的推荐将基于具有相似上下文801的聚类805加上具有接收和调用活动(结构)的处理加上具有其对于相似资源接收相似消息并用相似的输入、输出和故障模式调用相似资源方法的相似语义。如图所示,仅示出了三个聚类802-804,但是更多的聚类可以被包含在系统内并且经由用户界面来呈现。

[0130] 图9示出了根据实施例的示例性排名模拟器。

[0131] 根据实施例,可以接收JSON输入模式901。可以使用来自JSON输入模式的输入(例如,“swagger”、“info.version”和“info.title”)在JSON输入模式上运行基于相似性902的模式识别,以对已与JSON输入模式匹配的多个输出JSON模式903-905进行排名。

[0132] 图10示出了用于使用人工智能/机器学习来支持ICS(集成云服务)流程的自动完成的方法的流程图。

[0133] 在步骤1010处,该方法可以提供包括一个或多个微处理器的计算机。

[0134] 在步骤1020处,该方法可以启动(start)集成流程图(map)。

[0135] 在步骤1030处,该方法可以收集用户上下文。

[0136] 在步骤1040处,该方法可以基于所收集的用户上下文来提供多个流程预测。

[0137] 基于模式识别的下一活动预测

[0138] 根据实施例,本文描述的系统和方法可以基于文件格式(诸如JSON文件)中的流程(诸如ICS流程)内的模式识别来提供下一活动预测。这可以被描述为基于模式识别的下一活动预测(例如,在JSON模式内)。可以基于精确或部分上下文在多个上下文内进行下一活动预测。可以基于所有先前的任务来预测下一任务,或者可以基于考虑范围内可用的任务的频率的通用上下文来进行预测。

[0139] 根据实施例,在执行模式识别(例如,ICS JSON模式识别)并且识别出模式之后,下一步预测引擎可以基于精确或部分匹配(即,所有先前的任务完全或部分匹配)计算可能(例如,“最”可能)的下一步序列。该系统和方法还可以基于封闭范围来计算下一任务序列(即,在不依赖于先前任务的情况下,对于特定范围,最可能的下一任务是什么)。

[0140] 根据实施例,流程模式识别(例如,ICS流程JSON模式识别)可以导致与设计流程最紧密匹配的模型的子集。基于该子集,下一活动序列可以被预测为完美/部分匹配和范围预测。

[0141] 完美/部分匹配

[0142] 根据实施例,当本文描述的方法和系统执行完美或部分匹配以便预测下一活动时,该系统和方法可以使用现有流程的模式识别基于出现的频率来对潜在的下一任务序列执行排名。例如,在包含“销售账户创建”流程的以下流程中,假设用户正在设计该流程,并且已经到达将账户业务对象映射到销售云账户业务对象的分配步骤。基于该输入,模式识别系统最多识别与该输入模式紧密匹配的25个现有流程(即,正在被设计的流程)。

```

    {
      "name" : "Sales Account Creation",
      "sequence" : [
        {
          "name" : "receive",
          "type" : "receive",
          "application" : "sales",
          "operation" : "createAccount",
[0143]   "businessObject" : "Account"
        },
        {
          "name" : "assign",
          "type" : "assign",
          "mapping" : "Copy sales Account to cloud sales Account"
        }
      ]
    }

```

[0144] 根据实施例,本文的系统和方法可以包括或可以访问可以在下一活动预测中使用的多个目标模型。为了简单起见,假设在下一活动预测方法/模型中使用以下三个目标模型。

[0145] 模型1:

```

    {
      "name" : "Sales Account Creation – Company A, Department A",
      "sequence" : [
[0146]   {
          "name" : "receive",
          "type" : "receive",
          "application" : "sales",

```

```

        "operation" : "createAccount",
        "businessObject" : "Account"
    },
    {
        "name" : "assign",
        "type" : "assign",
        "mapping" : "Copy sales Account to cloud sales Account"
    },
[0147]    {
        "name" : "invoke",
        "type" : "invoke ",
        "application" : "cloud sales Cloud",
        "operation" : "createAccount",
        "businessObject" : "Account"
    }
]
}
[0148] 模型2:
{
    "name" : "Sales Account Creation – Company B, Department B",
    "sequence" : [
        {
[0149]     "name" : "receive",
        "type" : "receive",
        "application" : "sales",
        "operation" : "createAccount",
        "businessObject" : "Account"
    },
    {

```

```

        "name" : "assign",
        "type" : "assign",
        "mapping" : "Copy sales Account to cloud sales Account"
    },
    {
        "name" : "invoke",
[0150]     "type" : "invoke ",
        "application" : "cloud sales Cloud",
        "operation" : "createAccount",
        "businessObject" : "Account"
    }
]
}
[0151] 模型3:
{
    "name" : "Sales Account Creation – Company C, Department C",
    "sequence" : [
        {
            "name" : "receive",
            "type" : "receive",
            "application" : "sales",
[0152]     "operation" : "createAccount",
            "businessObject" : "Account"
        },
        {
            "name" : "assign",
            "type" : "assign",
            "mapping" : "Copy sales Account to cloud sales Account"
        },
    ],
}

```

```

        {
            "name" : "if",
            "type" : "if",
            "condition" : "$cloudsalesCloud",
            "activity" : {
                "name" : "invoke",
                "type" : "invoke",
                "application" : "cloud sales Cloud",
                "operation" : "createAccount",
                "businessObject" : "Account"
            }
        }
    [0153] else : {
            "activity" : {
                "name" : "invoke",
                "type" : "invoke",
                "application" : "S sales Cloud",
                "operation" : "createAccount",
                "businessObject" : "Account"
            }
        }
    ]
}

```

[0154] 根据实施例,模式识别可以匹配所有三个处理,并且基于匹配来生成下一任务最有可能是什么的排名。然后可以将其提供给用户/开发人员。这样的匹配的这种结果的示例如下所示:

```

    "name" : "Sales Account Creation",
    [0155] "sequence" : [
        {

```

```

    "name" : "receive",
    "type" : "receive",
    "application" : "sales",
    "operation" : "createAccount",
    "businessObject" : "Account"
  },
  {
    "name" : "assign",
    "type" : "assign",
    "mapping" : "Copy sales Account to cloud sales Account"
  },
  {
    "nextActivities" : {
      "3" : [{
[0156]   "name" : "invoke",
        "type" : "invoke ",
        "application" : "cloud sales Cloud",
        "operation" : "createAccount",
        "businessObject" : "Account"
      },
      {
        "name" : "if",
        "type" : "if",
        "condition" : "$cloudSalesCloud",
        "activity" : {
          "name" : "invoke",
          "type" : "invoke",
          "application" : "cloud Sales Cloud",
          "operation" : "createAccount",

```

```

        "businessObject" : "Account"
    }
    else : {
        "activity" : {
            "name" : "invoke",
            "type" : "invoke",
            "application" : "S sales Cloud",
            "operation" : "createAccount",
            "businessObject" : "Account"
        }
    }
}
]
"4" : {}
}
}
]
}

```

[0158] 根据实施例,该系统和方法以最可能到最不可能的顺序对潜在的下一任务进行排名:“invoke(调用)”任务后面跟着“if(如果)”,因为它在序列活动的索引3处具有较高的出现频率。

[0159] 根据实施例,基于模式识别,除了预测整个流程之外,该方法还可以生成其中可以预测“下一步”的机制。然后,这基于已经匹配的模型。使用统计方法来确定在来自第一预测的匹配流程中使用最多的“下一步”。例如,如果用户输入“Router(路由器)”任务,那么在最高索引(索引0)处,下一步的最高概率很可能是“Transformer(变换器)”,其目标是rest,然后下一个可能性是“rest”调用,然后下一个最有可能是transform rest。然后,系统可以转到索引1-这然后是下一步。在每个索引或序列号处,生成最可能进行哪个活动的概率。

[0160] 图11图示了根据实施例的示例性流程图决策树。

[0161] 根据实施例,用户可以选择集成流程的第一部分,该第一部分在流程设计器1100的非阴影部分1101中示出。从集成流程的开始,取决于用户的上下文,系统和方法可以向用户呈现多个下一任务选项1103-1107,以进行各种选择。如图所示,当前在窗口1102(当前选择窗口)中显示预测的下一任务选项1105。可以提供附加的下一任务推荐作为可以被选择的卡布局。在选择特定的推荐/预测的下一任务后,ICS设计器可以显示附加的预测的下一

任务(在后面的索引处)。

[0162] 根据实施例,虽然仅示出为流程的两个顺序索引,但是应当容易理解,本文描述的系统和方法可以显示预测任务的一个以上的下一索引级别,并且该下一任务预测可以基于不止一个任务索引。

[0163] 根据实施例,可能不为流程选择推荐的下一任务(例如,在推荐不是正确推荐的情况下)。在这种情况下,其它下一任务推荐可以一致地被呈现为如图中阴影区域中的流程所绘出的“预测平面”内的预测。然后,这提供了选择各个任务或整个流程的选项。该系统和方法还可以呈现卡布局,该卡布局可以针对所有可能的推荐进行轻拂以完成流程。这可以允许进行导航,以查看每个预测的ICS流程的外观以及其行为。在选择每个选项时,可以实时测试流程。测试可以在最终选择流程之前提供功能和性能简档。

[0164] 图12示出了使用人工智能/机器学习对ICS(集成云服务)流程进行下一步预测的示例性方法的流程图。

[0165] 在步骤1210处,该方法可以提供包括一个或多个微处理器的计算机。

[0166] 在步骤1220处,该方法可以启动集成流程图。

[0167] 在步骤1230处,该方法可以收集用户上下文。

[0168] 在步骤1240处,该方法可以基于所收集的用户上下文在集成流程图中提供多个下一步预测。

[0169] 在步骤1250处,该方法可以接收多个下一步预测中的所选择的下一步预测的选择。

[0170] 基于模式识别来预测活动的应用、操作和业务对象

[0171] 根据实施例,本文描述的系统和方法可以提供一种机制,用于基于模式识别(例如,JSON模式识别)对活动所引用的应用、操作和业务对象进行排名。活动引用的应用、操作和业务对象的排名可以在多个上下文中计算。例如,这些上下文可以基于精确/部分上下文(即,基于所有先前任务来计算排名)。根据实施例,预测可以附加地或替代地基于考虑到特定任务所引用的应用、操作和业务对象的频率的通用上下文。

[0172] 根据实施例,在基于输入模型(正在设计的处理的模型)识别模式(例如,ICS JSON模式)之后,排名引擎可以基于精确/部分匹配来计算模型排名。例如,这可以基于所有先前的任务匹配来完成。根据实施例,系统还可以纯粹基于封闭范围(即,对于特定任务,最可能的应用、操作和业务对象是什么)来计算模型排名。

[0173] 根据实施例,流程模式(例如,ICS JSON流程)识别导致与设计流程(输入)最紧密匹配的模型子集。基于匹配的ICS模型的这个子集,确定应用、操作和业务对象的排名。

[0174] 完美/部分匹配

[0175] 根据实施例,该系统和方法可以基于现有流程的模式识别(例如,ICS流程的JSON模式)基于特定任务所引用的应用、操作和业务对象的出现频率来对其进行排名。为了说明这一点,以下面的“销售账户创建”ICS流程为例。在这样的示例中,假设用户正在设计此ICS流程的过程中,并且已经到达将销售账户业务对象映射到销售云账户业务对象的分配步骤。基于此输入,模式识别系统识别与该输入模式紧密匹配的多个现有流程(正在设计的ICS流程)。下面提供了示例性输入流程:

[0176] {

```

"name" : "Sales Account Creation",
"sequence" : [
  {
    "name" : "receive",
    "type" : "receive",
    "application" : "sales",
    "operation" : "createAccount",
    "businessObject" : "Account"
[0177]   },
  {
    "name" : "assign",
    "type" : "assign",
    "mapping" : "Copy sales account to cloud sales account"
  }
]
}

```

[0178] 根据实施例,并且为了简化该示例,假设3个现有流程与来自现有数据库的该输入流程匹配。下面提供了这三个匹配的流程:

[0179] 模型1:

```

{
  "name" : "Sales Account Creation – Company A, Department A",
  "sequence" : [
    {
[0180]   "name" : "receive",
    "type" : "receive",
    "application" : "sales",
    "operation" : "createAccount",
    "businessObject" : "Account"

```

```

    },
    {
      "name" : "assign",
      "type" : "assign",
      "mapping" : "Copy sales account to cloud sales account"
    },
    {
[0181]     "name" : "invoke",
      "type" : "invoke ",
      "application" : "cloud sales cloud",
      "operation" : "createAccount",
      "businessObject" : "Account"
    }
  ]
}
[0182] 模型2:
{
  "name" : "Sales Account Creation – Company B, Department B",
  "sequence" : [
    {
[0183]     "name" : "receive",
      "type" : "receive",
      "application" : "sales",
      "operation" : "createAccount",
      "businessObject" : "Account"
    },
    {
      "name" : "assign",
      "type" : "assign",

```

```

        "mapping": "Copy sales account to cloud sales account"
    },
    {
        "name": "invoke",
        "type": "invoke ",
[0184]     "application": "cloud sales Cloud",
        "operation": "createAccount",
        "businessObject" : "Account"
    }
]
}
[0185] 模型3:
{
    "name" : "Sales Account Creation – Company C, Department C",
    "sequence": [
        {
            "name": "receive",
            "type": "receive",
            "application": "sales",
            "operation": "createAccount",
[0186]     "businessObject" : "Account"
        },
        {
            "name": "assign",
            "type": "assign",
            "mapping": "Copy sales account to cloud sales account"
        },
        {
            "name": "invoke",

```

```

        "type": "invoke",
        "application": "someother cloud sales Cloud",
        "operation": "createAccount",
[0187]     "businessObject" : "Account"
        }
    ]
}

```

[0188] 根据实施例,模式识别系统和方法(例如,JSON模式识别系统和方法)可以匹配合上述所有三个模型,并基于该匹配来计算应用、操作和业务对象的排名:

```

{
  "name" : "Sales Account Creation",
  "sequence": [
    {
      "name": "receive",
      "type": "receive",
      "application": "salesf",
      "operation": "createAccount",
      "businessObject" : "Account"
[0189]    },
    {
      "name": "assign",
      "type": "assign",
      "mapping": "Copy sales account to could sales account"
    },
    {
      "name": "invoke",
      "type": "invoke",
      "rankings": {

```

```
        "application": [  
            { //Higher Ranking => Higher Frequency  
                "name": "cloud sales cloud",  
                "operation": [  
                    {  
                        "name": "createAccount",  
                        "businessObject" : [  
                            {  
                                "name": "Account"  
                            }  
                        ]  
                    }  
                ]  
            },  
            [0190] {  
                "name": "someother cloud sales Cloud",  
                "operation": [  
                    {  
                        "name": "createAccount",  
                        "businessObject" : [  
                            {  
                                "name": "Account"  
                            }  
                        ]  
                    }  
                ]  
            }  
        ]  
    }  
}
```

```
    }
```

```
[0191] ]
```

```
    }
```

[0192] 根据实施例,对于调用任务,排名系统对“云销售云”应用的排名高于“其它云销售云”应用的排名。以这种方式,如果用户创建新的“调用”任务,那么系统可以向用户呈现选择“云销售云”作为主要/首选/第一选项的选项。这样做是因为系统更有可能是由销售中的创建账户事件触发的。也就是说,训练数据包含更多的流程,这些流程以“销售”接收开始,后面跟着分配,最后是对“云销售云”的调用。

[0193] 根据实施例,该系统和方法可以基于位置独立地对由特定任务引用的应用、操作和业务对象进行排名。例如以下面提供的“HR(人力资源)账户创建”ICS流程为例。这里,用户正在设计ICS流程的过程中,并且已经到达将“可行的HR账户”业务对象映射到云HR云账户业务对象的分配步骤。基于此输入,模式识别系统识别例如训练数据库中与该输入模式紧密匹配的多个现有流程(正在设计的ICS流程)。示例输入流程在此:

```
{
```

```
  "name" : "HR Account Creation",
```

```
  "sequence" : [
```

```
    {
```

```
      "name" : "receive",
```

```
      "type" : "receive",
```

```
[0194]      "application" : "workable",
```

```
      "operation" : "createAccount",
```

```
      "businessObject" : "Account"
```

```
    },
```

```
    {
```

```
      "name" : "assign",
```

```
      "type" : "assign",
```

```
      "mapping" : "Copy workable account to cloud HR account"
```

```
    }
```

```
[0195]
```

```
  ]
```

```
}
```

[0196] 根据实施例,虽然现有流程不具有“可行的HR账户”业务对象,但是该系统和方法

可以将输入流程与训练数据库中的多个相似的现有流程进行匹配。在这种情况下,对于以可行的创建账户开头的进程不具有精确匹配,因为系统在其进行通用预测之前尚未看到此信息,如果第三任务是调用,那么基于系统通用预测,用户可能希望创建销售账户;这是其它用户使用调用活动时通常会进行的操作。这可以从匹配的流程中理解,这里:

```
{
  "rankings": {
    "invoke": {
      "application": [
        { //Higher Ranking => Higher Frequency
          "name": "cloud Sales Cloud",
          "operation": [
            {
              "name": "createAccount",
              "businessObject" : [
                {
                  "name": "Account"
                }
              ]
            }
          ]
        }
      ]
    },
```

[0197]

```

        {
            "name" : "$ Sales Cloud",
            "operation" : [
                {
                    "name" : "createAccount",
                    "businessObject" : [
                        {
                            "name" : "Account"
                        }
                    ]
                }
            ]
        }
    ],
    [0198] },
    "name" : "HR Account Creation",
    "sequence" : [
        {
            "name" : "receive",
            "type" : "receive",
            "application" : "workable",
            "operation" : "createAccount",
            "businessObject" : "Account"
        },
        {
            "name" : "assign",
            "type" : "assign",
            "mapping" : "Copy Workday Account to Oracle HR Account"
        }
    ],

```

```
    {  
      "name": "invoke",  
      "type": "invoke"  
    }  
  ]  
}
```

[0200] 图13图示了根据实施例的示例性流程图决策树。

[0201] 根据实施例,用户可以选择集成流程的第一动作,其被显示在流程设计器1300的非阴影部分1301中。从集成流程的开始,系统和方法可以预测已经存储在训练数据库中的多个流程。例如,可以在各种窗口中呈现预测的下一动作,诸如1302-1307。但是,在用户选择或指示对下一个应用、操作或业务对象的请求后,系统可以在窗口1350中向用户呈现例如多个选项,诸如选项A-D。这些选项可以包括例如预测的下一个应用、操作或业务对象。然后,如果期望,用户可以选择所呈现的选项之一,其中可以基于输入流程和存储的已完成流程之间的匹配,以从最可能到不太可能的顺序显示所呈现的选项。

[0202] 图14示出了用于使用人工智能/机器学习对ICS(集成云服务)流程进行下一对象预测的示例性方法的流程图。

[0203] 在步骤1410处,该方法可以提供包括一个或多个微处理器的计算机。

[0204] 在步骤1420处,该方法可以启动集成流程图。

[0205] 在步骤1430处,该方法可以收集用户上下文。

[0206] 在步骤1440处,该方法可以基于所收集的用户上下文在集成流程图内提供多个下一对象预测,其中多个下一对象预测中的每一个包括应用、操作和业务对象中的一个。

[0207] 在步骤1450处,该方法可以接收多个下一步预测中的所选择的下一对象预测的选择。

[0208] 虽然上面已经描述了本发明的各种实施例,但是应当理解的是,它们是通过示例而非限制来呈现的。选择和描述实施例是为了解释本发明的原理及其实际应用。实施例说明了系统和方法,其中利用本发明通过提供新的和/或改进的特征和/或提供诸如降低的资源利用率、增加的容量、提高效率 and 减少的时延之类的益处来改进系统和方法的性能。

[0209] 在一些实施例中,本发明的特征全部或部分地在包括处理器、诸如存储器之类的存储介质和用于与其它计算机通信的网卡的计算机中实现。在一些实施例中,本发明的特征在分布式计算环境中实现,其中一个或多个计算机集群通过诸如局域网(LAN)、交换架构网络(例如,InfiniBand)或广域网(WAN)之类的网络连接。分布式计算环境可以使所有计算机处于单个地点,或者使处于不同远程地理位置的计算机集群通过WAN连接。

[0210] 在一些实施例中,本发明的特征全部或部分地在云中实现,作为基于使用Web技术以自助服务、计量方式向用户递送的共享的弹性资源的云计算系统的一部分或作为其服务。云有五个特性,如由国家标准与技术研究所定义:按需自助服务;广泛的网络接入;资源池化;快速弹性;以及测量的服务。云部署模型包括:公共、私有和混合。云服务模型包括软件即服务(SaaS)、平台即服务(PaaS)、数据库即服务(DBaaS)和基础架构即服务(IaaS)。如

本文所使用的,云是硬件、软件、网络和web技术的组合,其以自助服务、计量方式向用户递送共享的弹性资源。除非另有说明,否则如本文所使用的云包括公共云、私有云和混合云实施例,以及所有云部署模型,包括但不限于云SaaS、云DBaaS、云PaaS和云IaaS。

[0211] 在一些实施例中,使用硬件、软件、固件或其组合或在其帮助下实现本发明的特征。在一些实施例中,使用被配置或被编程为执行本发明的一个或多个功能的处理器来实现本发明的特征。在一些实施例中,处理器是单芯片或多芯片处理器、数字信号处理器(DSP)、片上系统(SOC)、专用集成电路(ASIC)、现场可编程门阵列(FPGA)或其它可编程逻辑器件、状态机、离散门或晶体管逻辑、分立硬件部件或被设计用于执行本文所述的功能的其任何组合。在一些实现中,本发明的特征可以由特定于给定功能的电路系统实现。在其它实现中,这些特征可以在处理器中实现,该处理器被配置为使用例如存储在计算机可读存储介质上的指令来执行特定功能。

[0212] 在一些实施例中,本发明的特征结合在软件和/或固件中,用于控制处理和/或网络系统的硬件,并用于使处理器和/或网络能够与利用本发明的特征的其它系统进行交互。这样的软件或固件可以包括但不限于应用代码、设备驱动程序、操作系统、虚拟机、管理程序、应用编程接口、编程语言和执行环境/容器。基于本公开的教导,熟练的程序员可以容易地准备适当的软件编码,这对于软件领域的技术人员来说是显而易见的。

[0213] 在一些实施例中,本发明包括计算机程序产品,其是具有存储在其上/内的指令的存储介质或计算机可读介质,指令可以用于编程或以其它方式配置诸如计算机之类的系统,以执行本发明的任何处理或功能。存储介质或计算机可读介质可以包括但不限于任何类型的盘,包括软盘、光盘、DVD、CD-ROM、微驱动器和磁光盘、ROM、RAM、EPROM、EEPROM、DRAM、VRAM、闪存设备、磁卡或光卡、纳米系统(包括分子存储器IC)、或适于存储指令和/或数据的任何类型的介质或设备。在特定实施例中,存储介质或计算机可读介质是非暂态存储介质或非暂态计算机可读介质。

[0214] 前面的描述并非旨在是详尽的或将本发明限制于所公开的精确形式。此外,在已经使用特定系列的事务和步骤描述了本发明的实施例的情况下,对于本领域技术人员来说显而易见的是,本发明的范围不限于所描述的一系列事务和步骤。另外,在已经使用硬件和软件的特定组合描述了本发明的实施例的情况下,应当认识到硬件和软件的其它组合也在本发明的范围内。另外,虽然各种实施例描述了本发明的特征的特定组合,但是应当理解的是,在本发明的范围内,特征的不同组合对于相关领域的技术人员来说是显而易见的,使得一个实施例的特征可以结合到另一个实施例中。而且,对于相关领域的技术人员来说显而易见的是,在不脱离本发明的精神和范围的情况下,可以在形式、细节、实现和应用中进行各种添加、减少、删除、变化以及其它修改和改变。旨在通过所附权利要求及其等同物来限定本发明的更广泛的精神和范围。

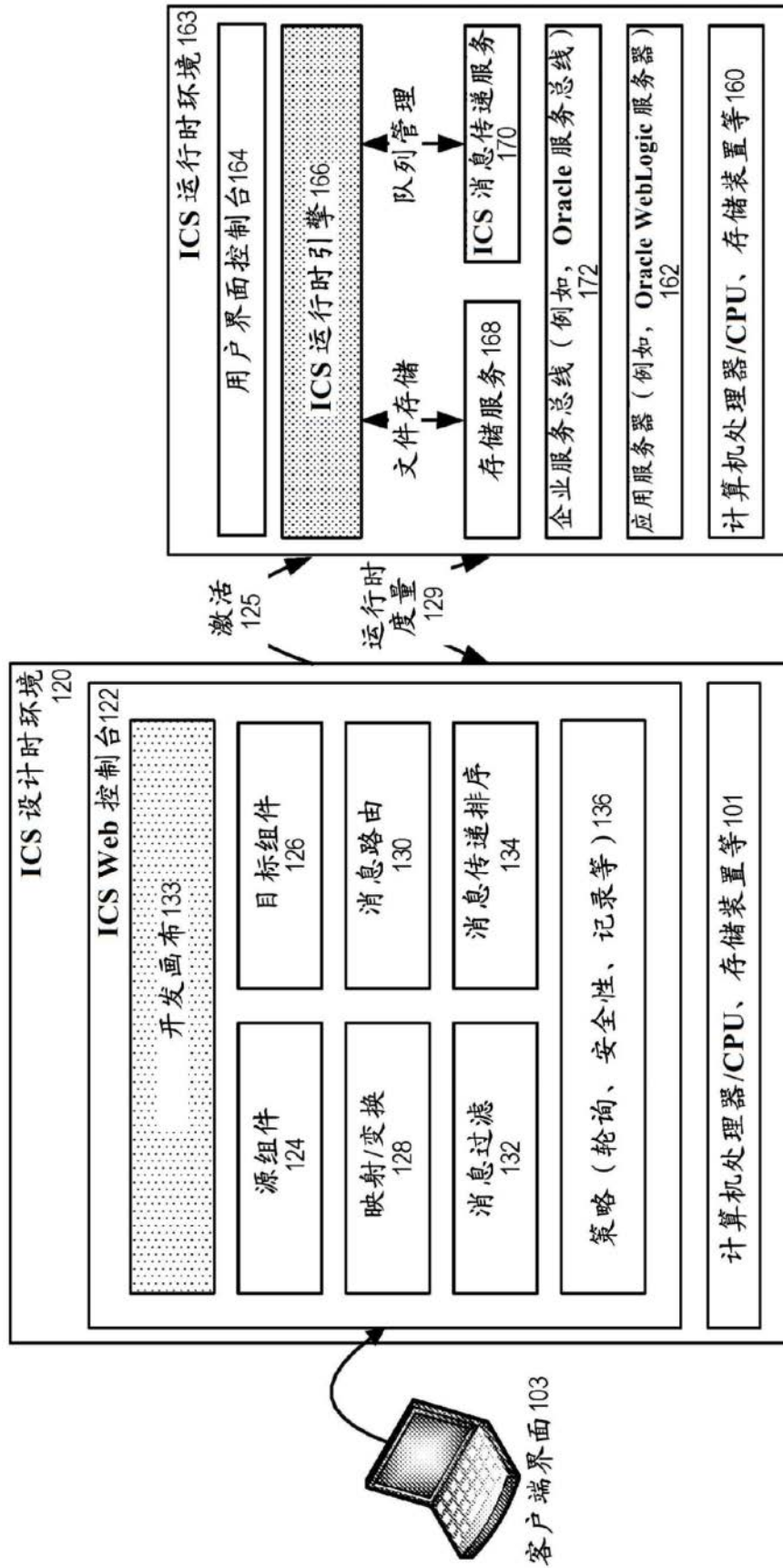


图1

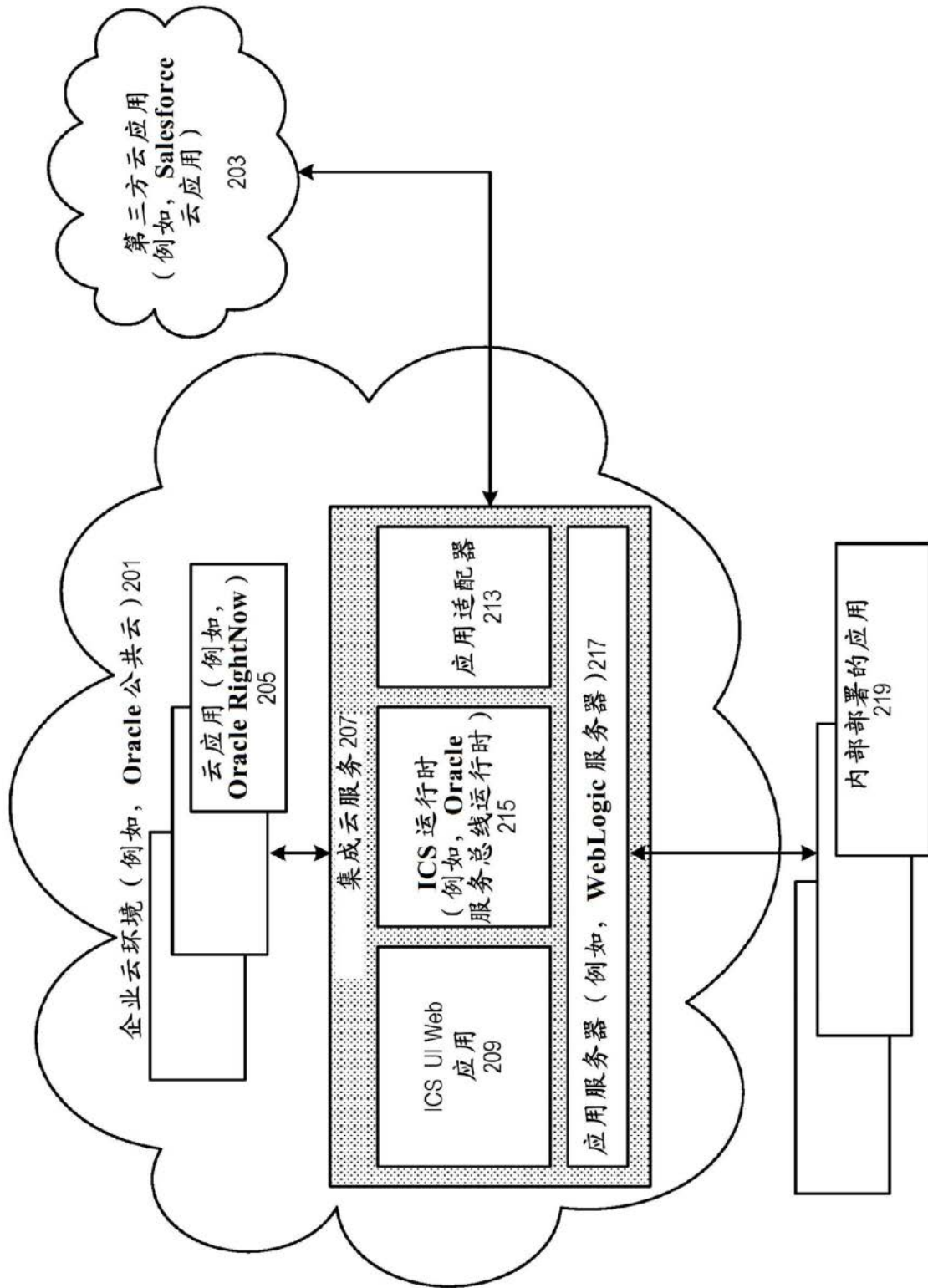


图2

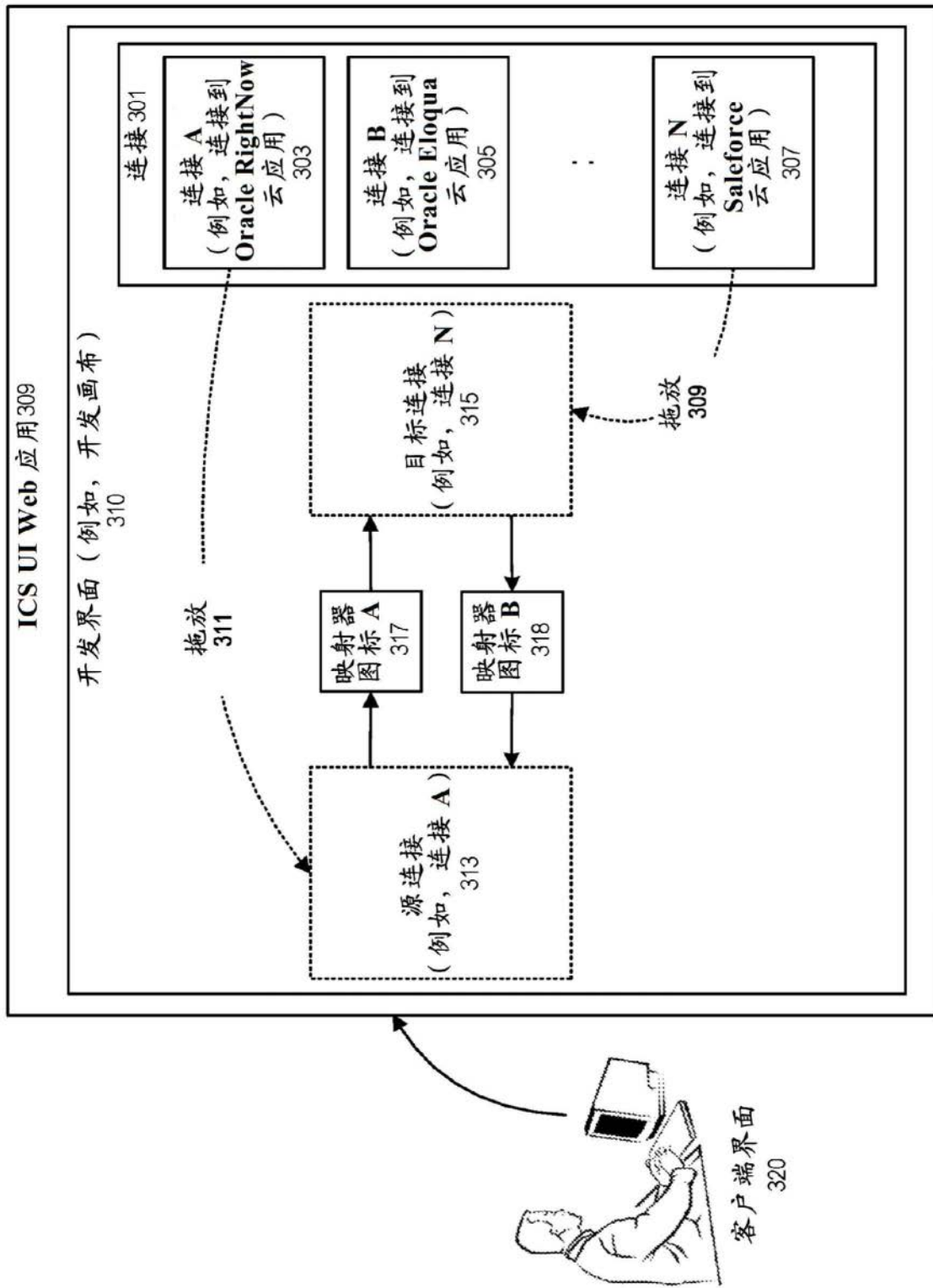


图3

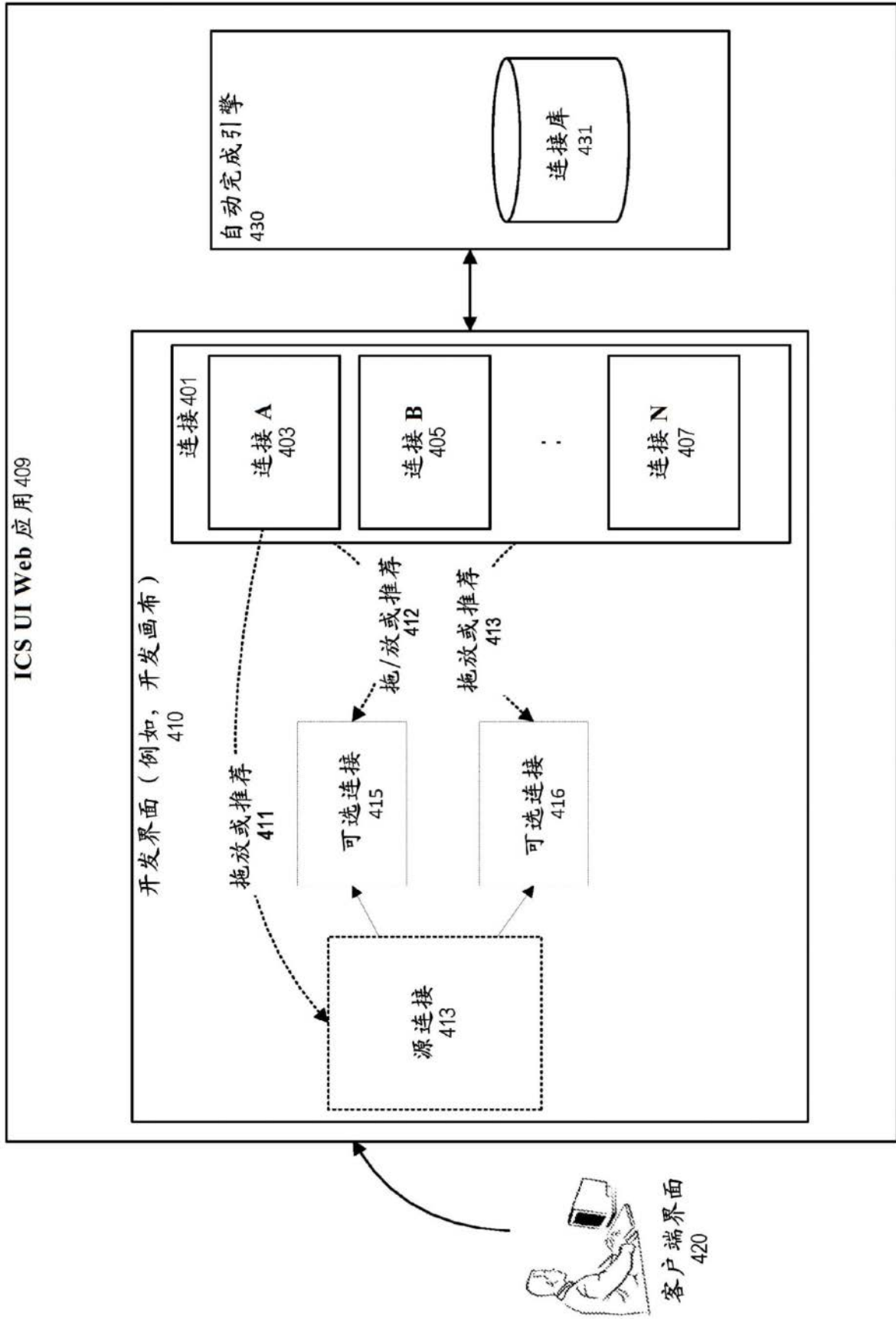


图4

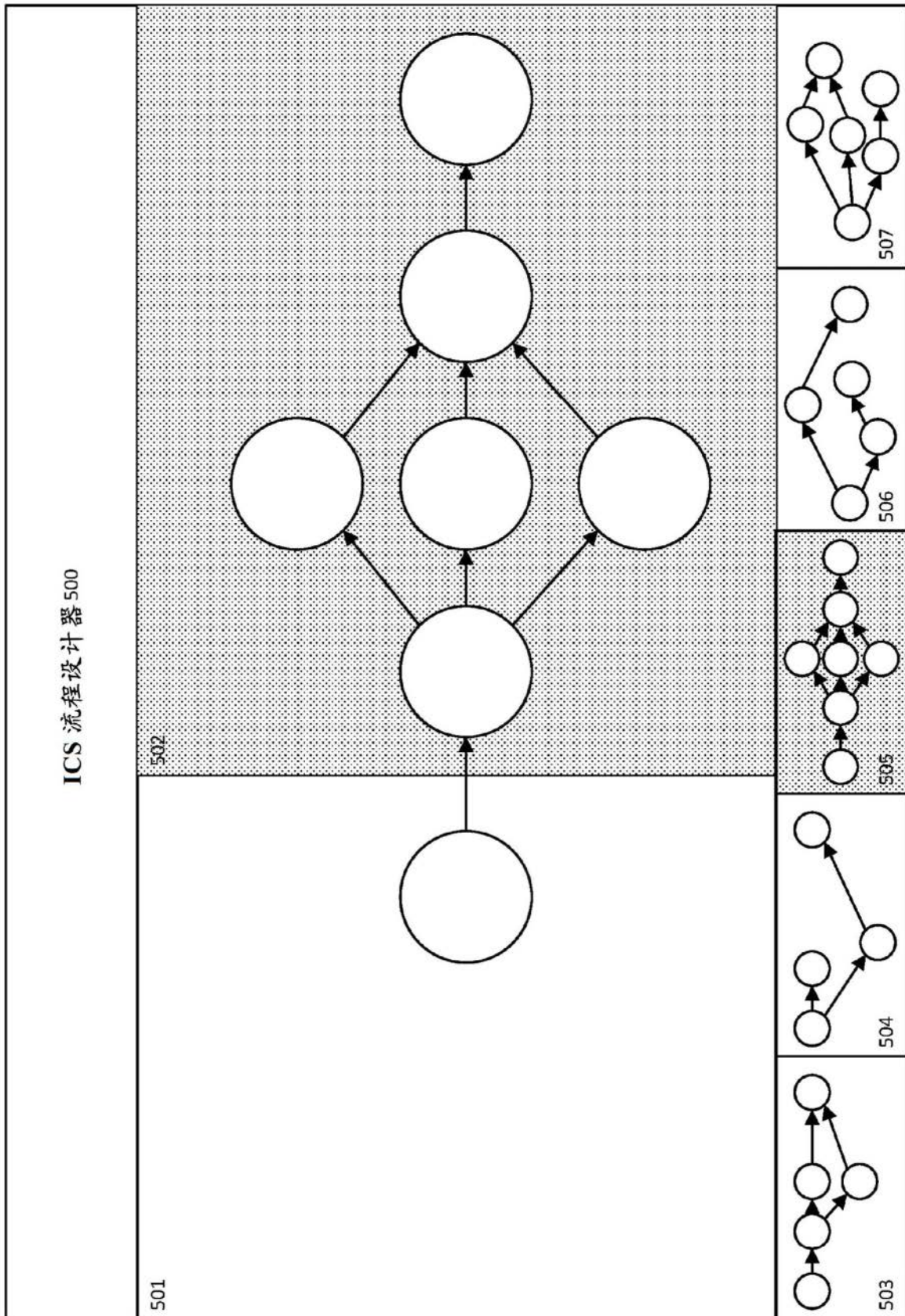


图5

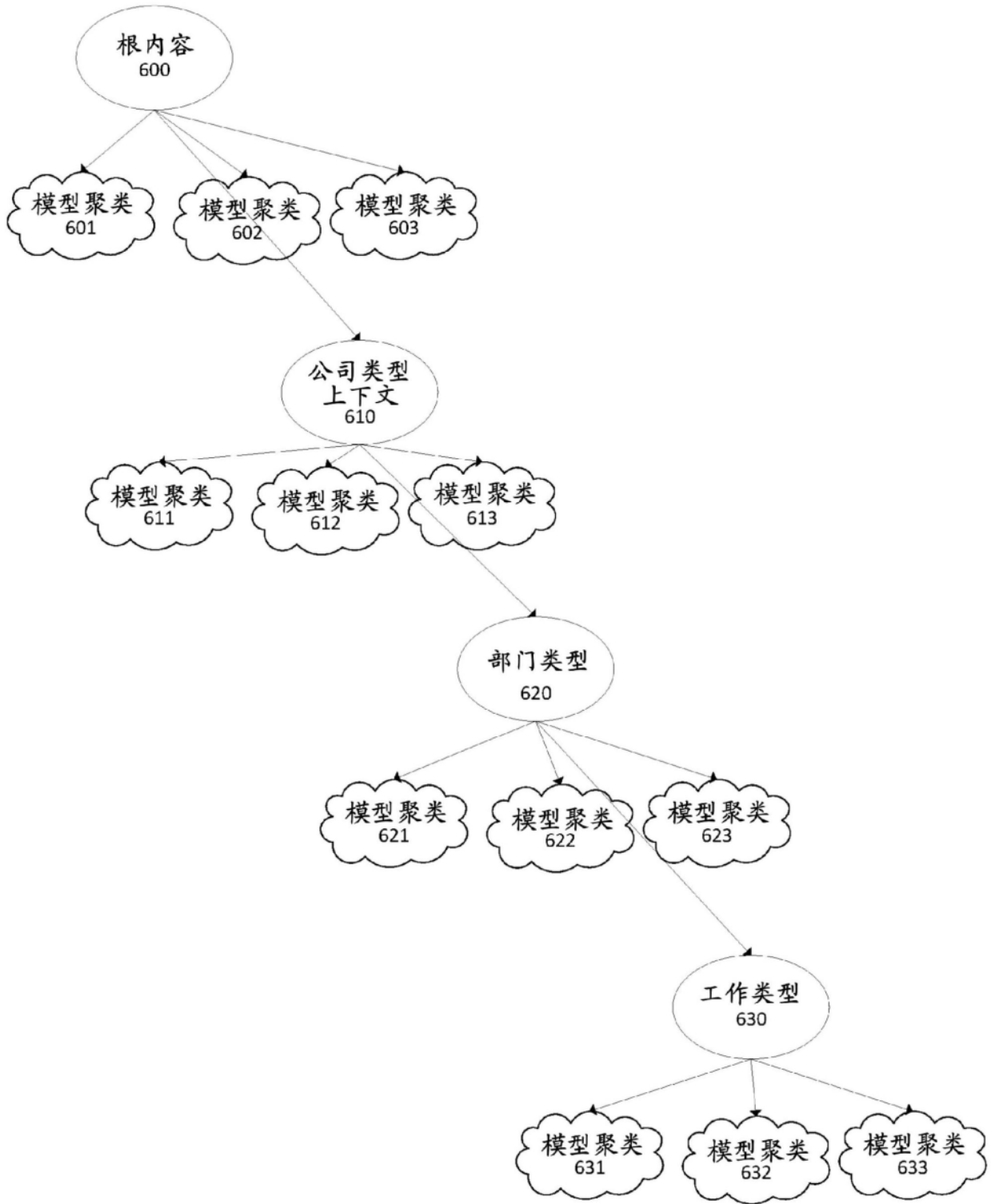


图6

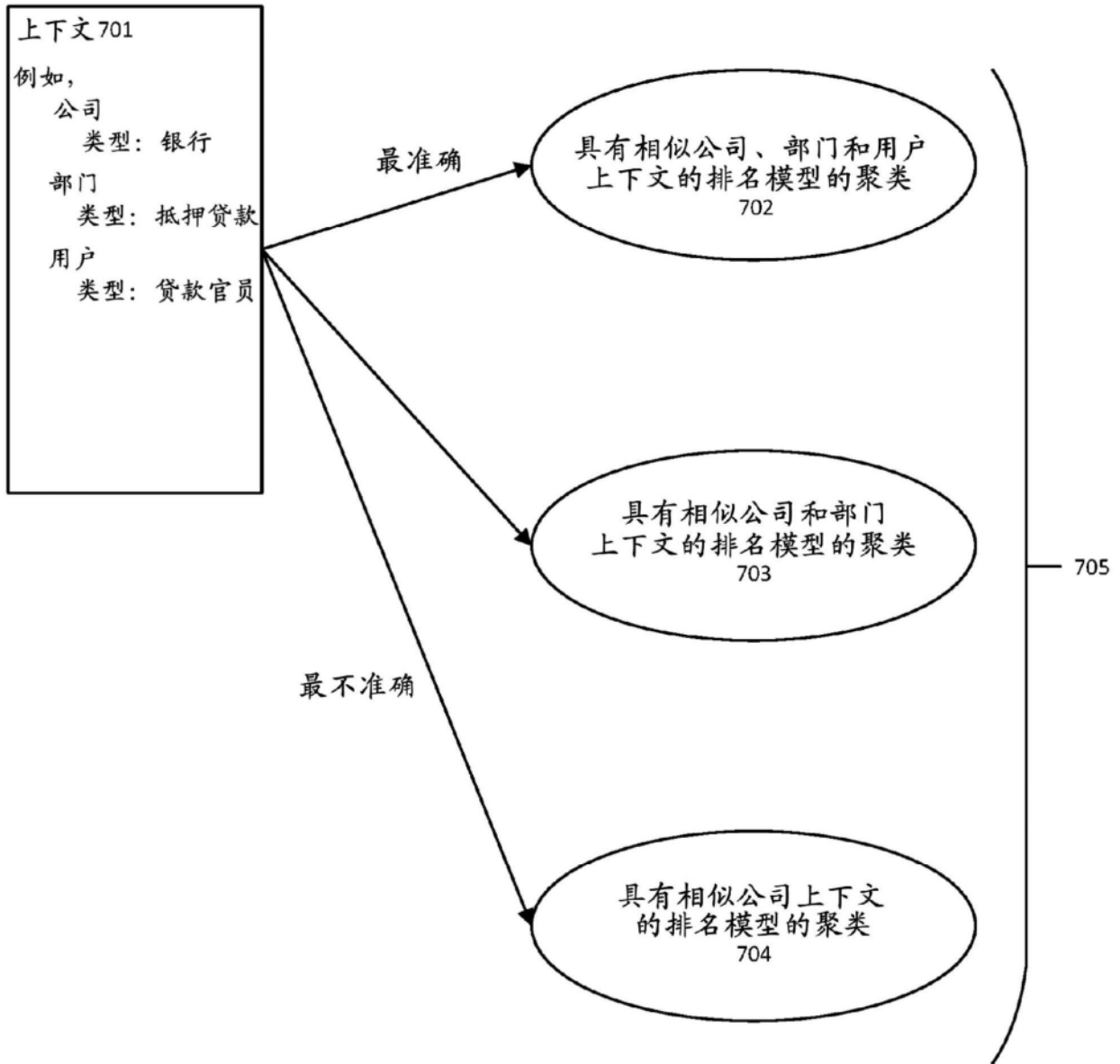


图7

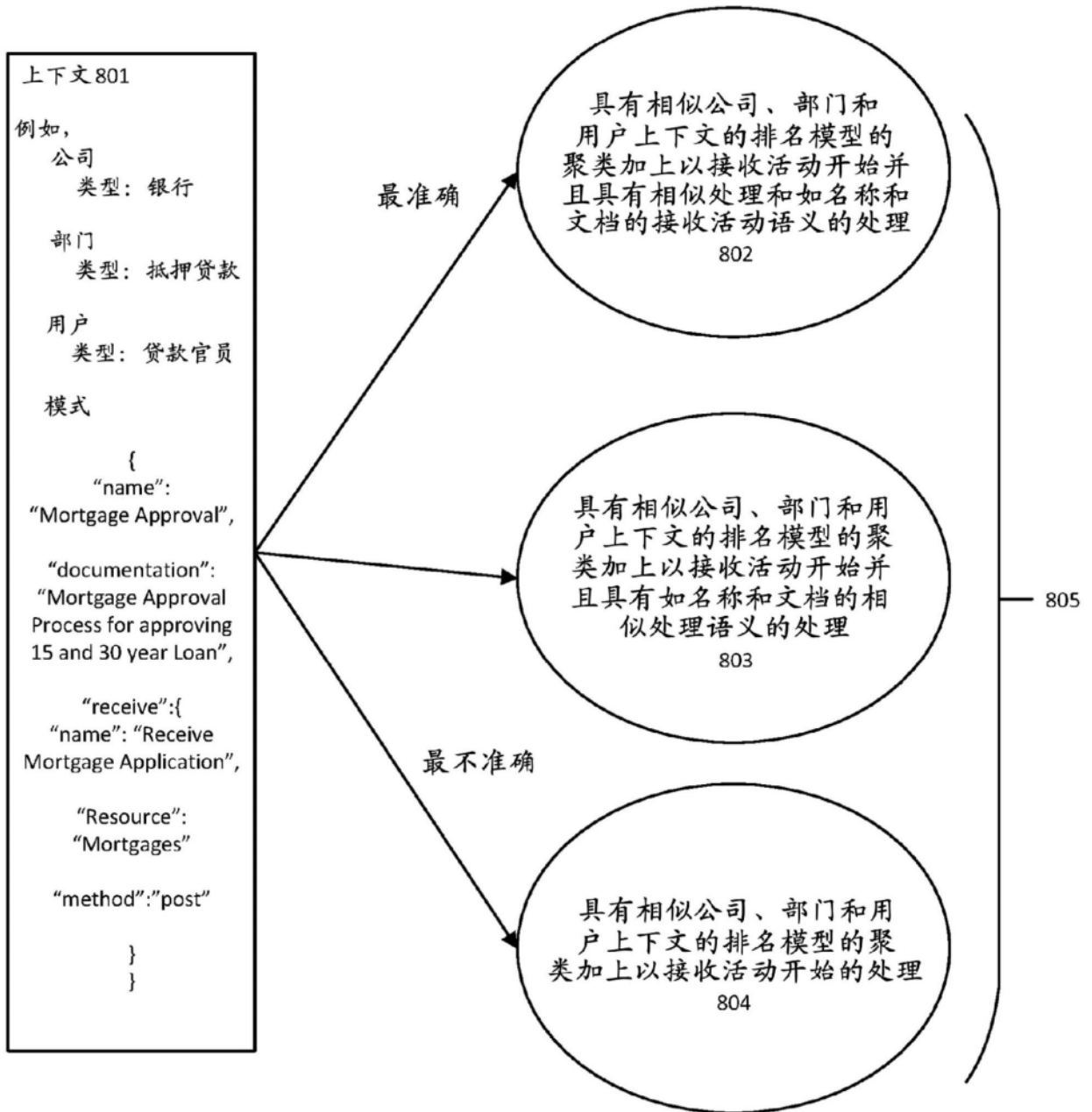


图8

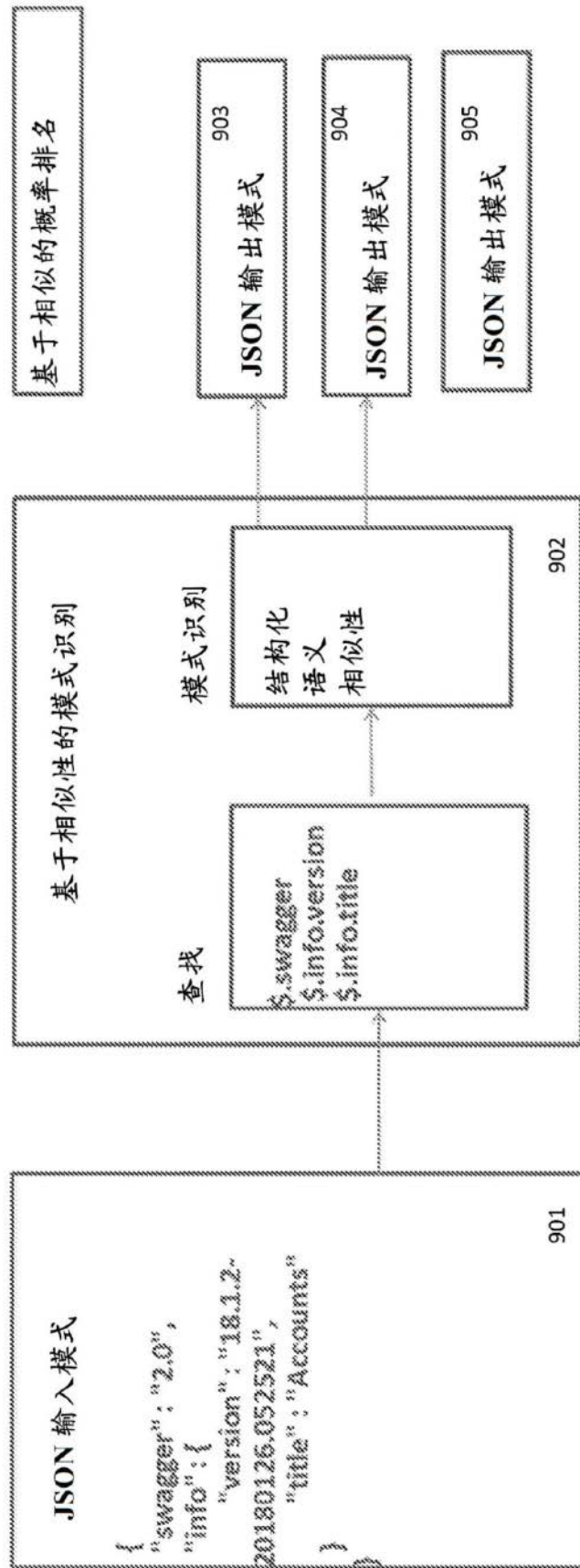


图9

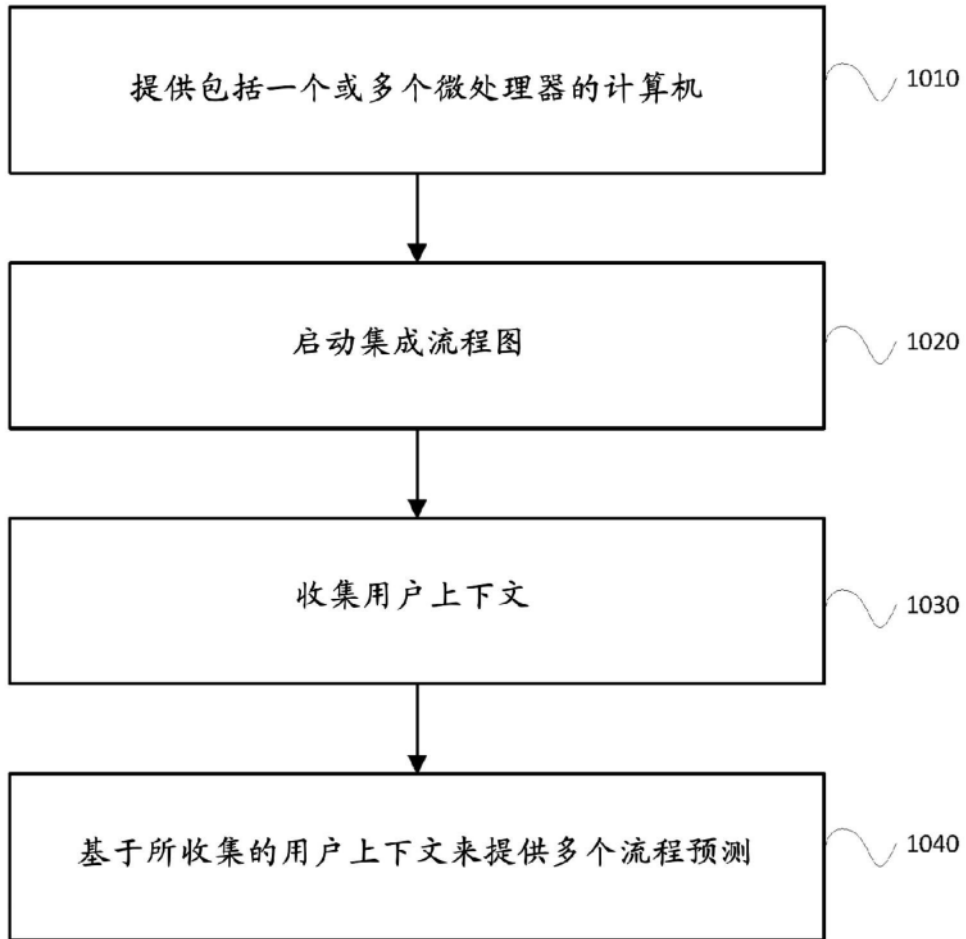


图10

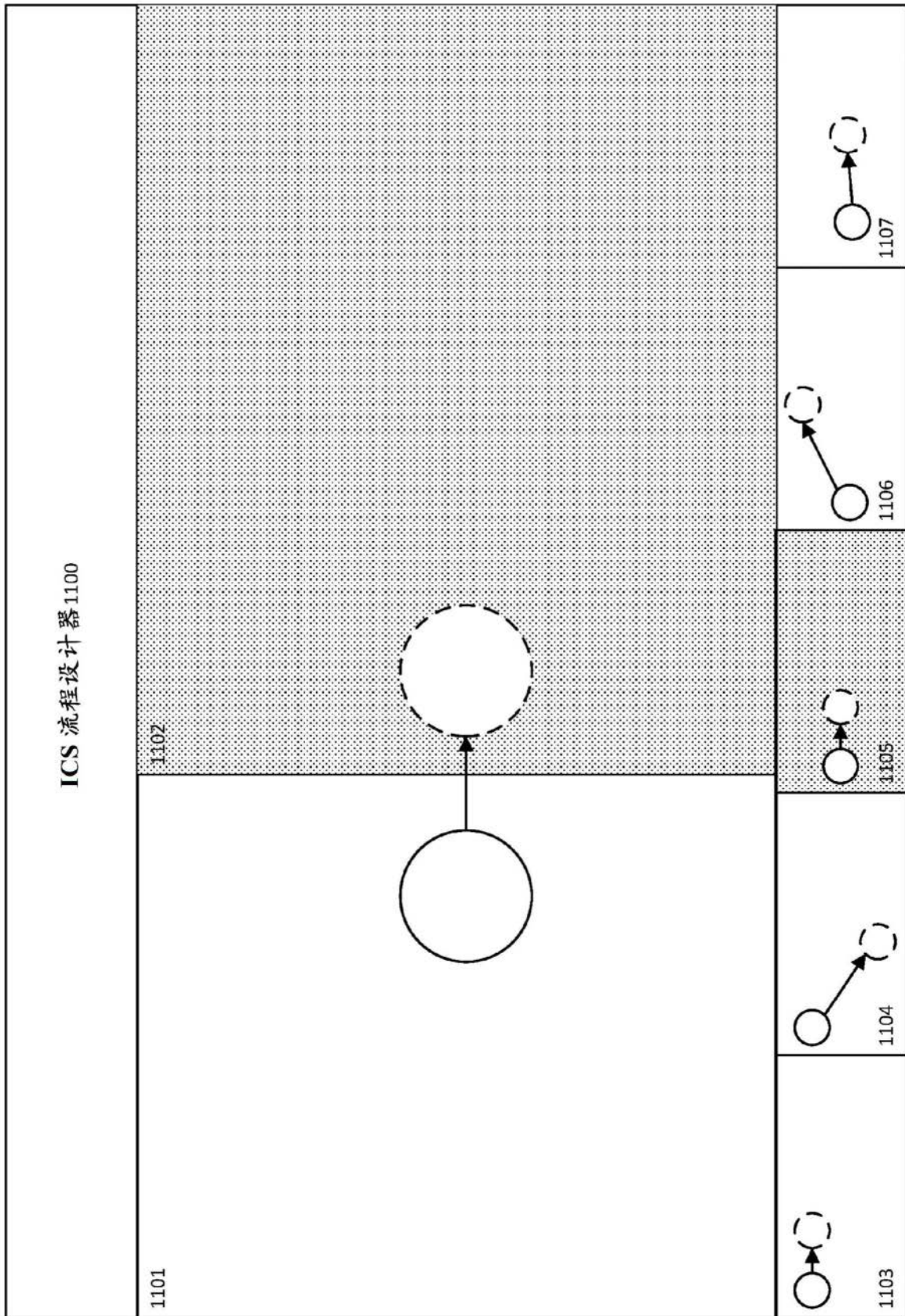


图11

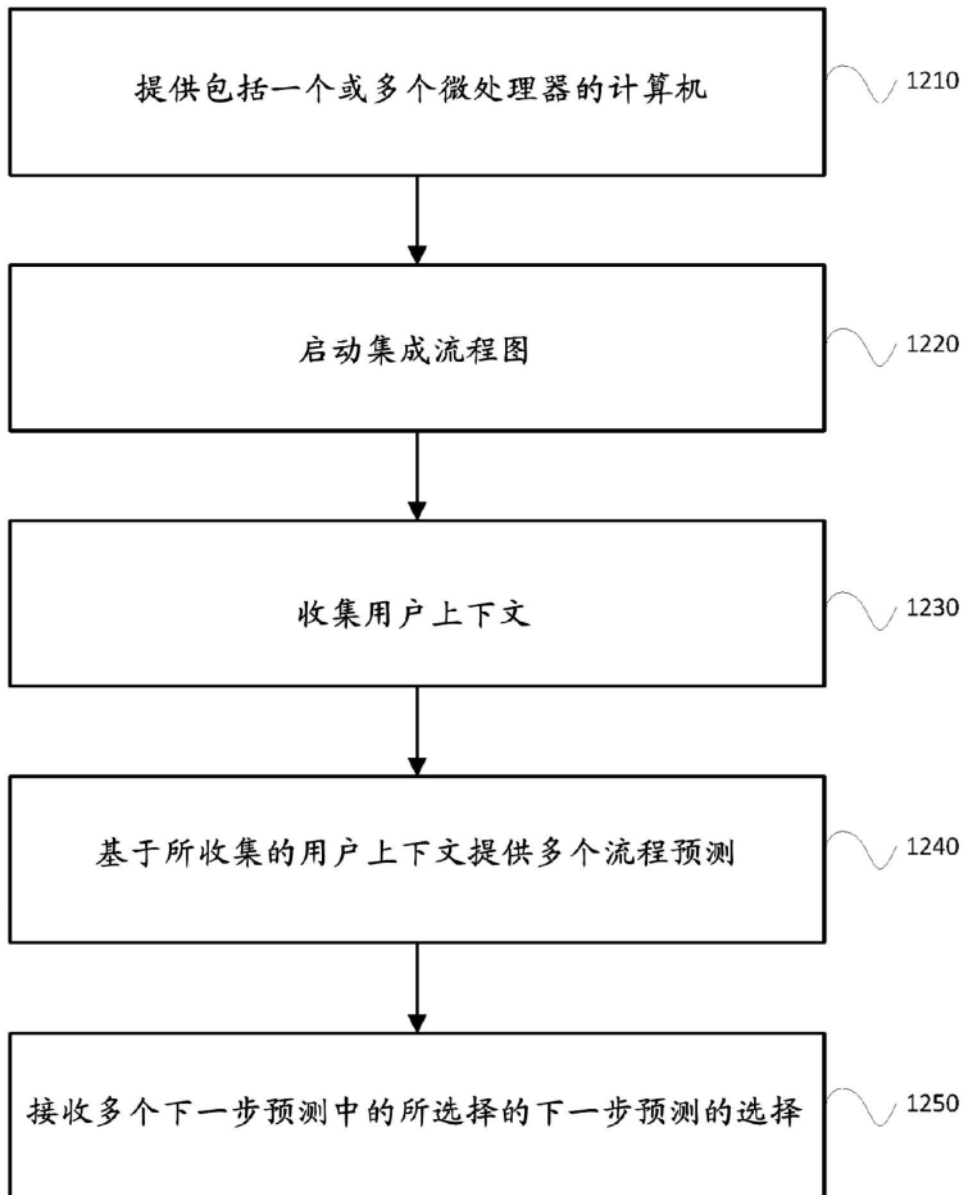


图12

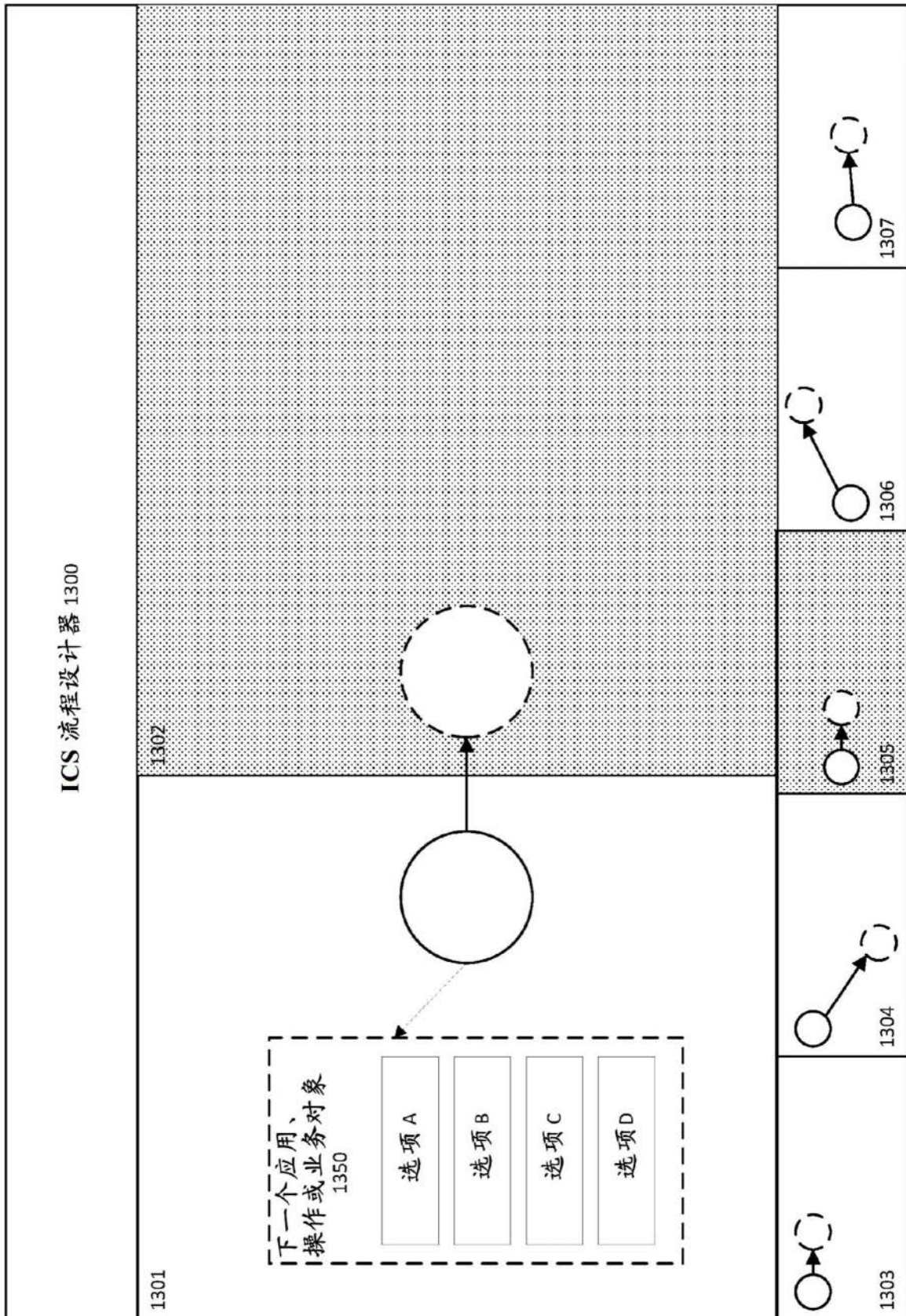


图13

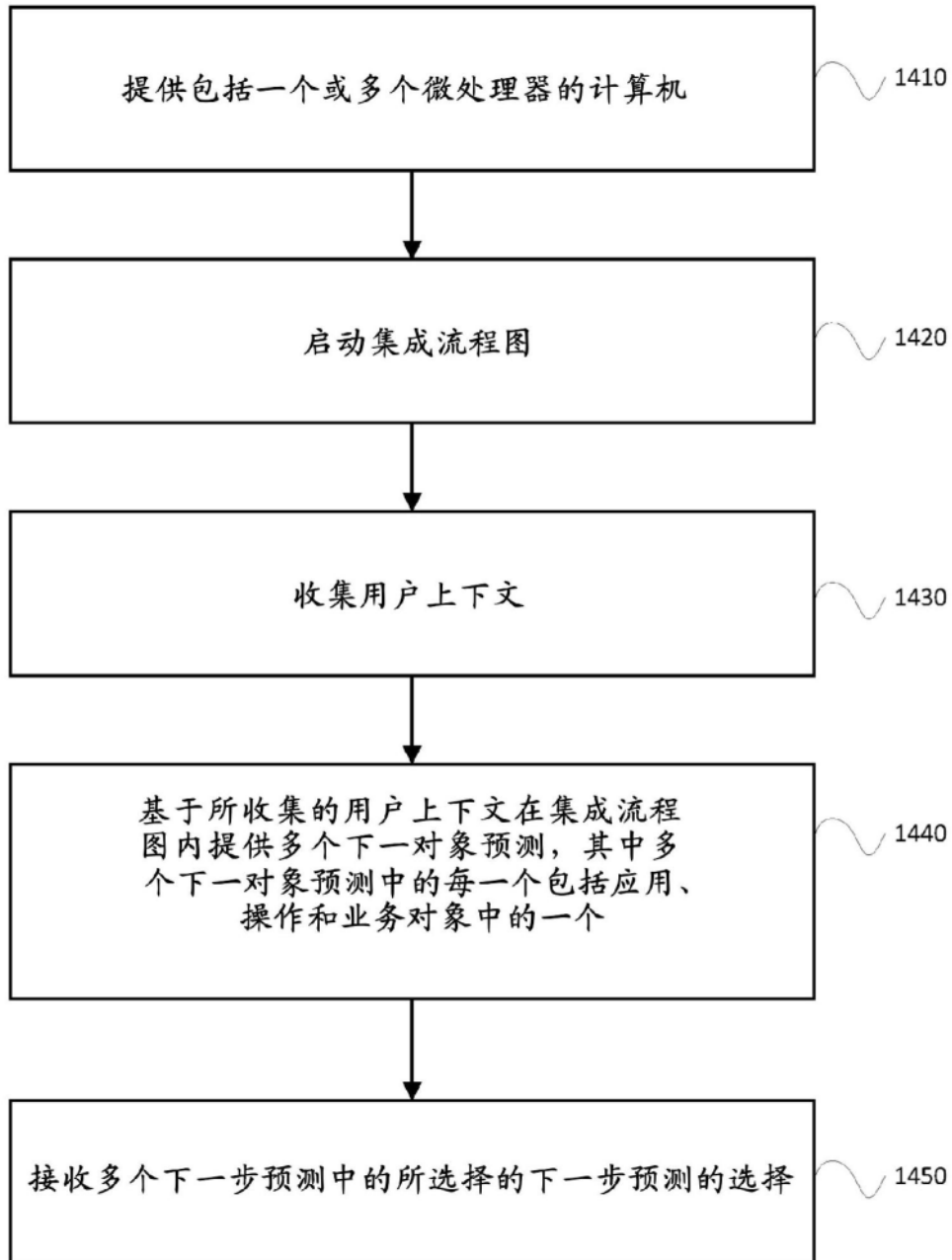


图14