



(12) 发明专利

(10) 授权公告号 CN 1527222 B

(45) 授权公告日 2012.03.21

(21) 申请号 200310123963.2

org/documents/swgc/swgc-final.

(22) 申请日 2003.12.19

pdf. 2002, 1, 16-17.

(30) 优先权数据

审查员 吴广平

60/434,432 2002.12.19 US

60/501,012 2003.09.09 US

60/511,741 2003.10.17 US

10/733,328 2003.12.12 US

(73) 专利权人 富士通株式会社

地址 日本神奈川县川崎市

(72) 发明人 益冈龙介 约翰·郎布罗 宋哲炫

(74) 专利代理机构 北京三友知识产权代理有限公司 11127

代理人 李辉

(51) Int. Cl.

G06F 15/76 (2006.01)

(56) 对比文件

GOBLE ET AL. Semantic Web and Grid

Computing. <http://www.semanticgrid>.

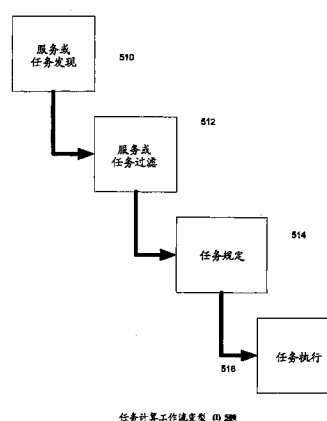
权利要求书 9 页 说明书 61 页 附图 62 页

(54) 发明名称

基于计算机的系统、方法和终端设备

(57) 摘要

基于计算机的系统包括任务计算,从而使用户能够通过将可用功能性组合在一起来定义任务并执行这种任务。基于计算机的系统包括可用功能性,可用功能性起源于设备、计算应用程序以及通过本地和远程过程调用可用的、包括 web 服务、UPnP、CORBA、RMI、RPC、DCE、DCOM 的电子服务,或者该可用功能性包括以前定义的任务。所有可用功能性对用户抽象为一服务,并以服务描述语言来表示每个服务,而且各服务具有与它们相关联的语义描述。



1. 一种基于计算机的系统,该基于计算机的系统包括:

计算机系统,其执行任务计算,从而使用户(110)能够通过将设备(304)或以计算机为介质的服务的功能性组合在一起定义任务并执行这种任务,所述功能性包括计算应用程序(306)以及通过远程过程调用可用的电子服务(312)或以前定义的任务;

其中,所有功能性对用户呈现为服务(1302),而且利用语义服务描述的服务描述语言来在语义上表示每个服务;

其中,每个服务具有至少一个与它相关联的语义服务描述,并且通过使其一个或者多个语义描述可用或不可用,能够使该服务可用或不可用;

该基于计算机的系统进一步包括:

服务发现装置,其用于基于与可用服务相关联的语义服务描述发现该可用服务;

服务过滤装置,其用于利用一个或多个标准来对服务进行过滤以从所发现的可用服务中选择服务子集;

任务规定装置,其用于通过将过滤的服务连接起来以规定与用户合作的可执行任务;以及

任务执行装置,其用于执行所规定的任务。

2. 根据权利要求1所述的基于计算机的系统,该基于计算机的系统进一步包括用于建立、移除以及管理服务(1302)的装置。

3. 根据权利要求1所述的基于计算机的系统,其中,所述远程过程调用包括Web服务、UPnP、CORBA、RMI、RPC、DCE、DCOM。

4. 根据权利要求1所述的基于计算机的系统,该基于计算机的系统被设置为自动地或者根据用户(110)的指示,把规定的任务保存为新的可用服务。

5. 根据权利要求1所述的基于计算机的系统,其中,服务发现装置被设置为利用一服务发现协议,该服务发现协议包括如下任何一种,或者它们的任意组合:

UPnP、UDDI、本地服务资料库、Jini、蓝牙SDP、会合以及红外线。

6. 根据权利要求5所述的基于计算机的系统,其中,服务过滤装置所用的标准包括如下任何一种,或者它们的任意组合:

用户概况、手头任务、用户设备特性、用户位置、用户运动状态、用户网络连通性、用户规定的关键字、在被总体看待时一组服务的特征、单独服务特征。

7. 根据权利要求6所述的基于计算机的系统,其中,任务规定装置包括如下任何一种,或者它们的任意组合:

基于计划的自动系统,或者

交互用户界面,其支持以下要素中的任何一种,或者它们的任意组合:可视、语音、文本、盲文、触觉。

8. 根据权利要求7所述的基于计算机的系统,其中,任务执行装置被设置为利用服务执行机制,该服务执行机制包括如下任何一种,或者它们的任意组合:web服务调用、UPnP动作、CORBA、RMI、RPC、DCE、DCOM、红外线、本地功能调用以及本地对象调用。

9. 根据权利要求2所述的基于计算机的系统,其中,用于建立、移除以及管理服务的装置包括如下任何一种,或者它们的任意组合:图形用户界面、以及被设置为在用户操作的设备上执行的应用程序。

10. 根据权利要求 9 所述的基于计算机的系统,其中,服务发现装置、服务过滤装置、任务规定装置以及任务执行装置包括:用户界面(1700-4100),该用户界面(1700-4100)帮助用户进行服务发现、服务过滤、任务规定、任务执行并保存规定的任务待未来之用,

其中,任务规定装置对用户关于可以如何组合服务(1204-1208)以构成可执行任务的信息,并且

其中,任务执行装置通过调用构成任务的服务并通过对用户提供在执行调用的服务时与调用的服务进行交互的装置,帮助用户执行任务。

11. 根据权利要求 10 所述的基于计算机的系统,其中,服务发现装置、服务过滤装置、任务规定装置以及任务执行装置分别是以下程序模块:发现模块、过滤模块、组成模块以及执行模块,并且

其中,这些模块被设置为在单个计算设备上执行,或者其中,这些模块或这些模块的子部件分布在多个计算设备上,而且可以被程序设计界面访问,所述程序设计界面可以访问被设置为运行在用户操作的设备上的用户界面(1700-4100)。

12. 根据权利要求 9 所述的基于计算机的系统,该基于计算机的系统被设置为允许用户(110)根据服务发现装置、服务过滤装置、任务规定装置以及任务执行装置,进行服务发现,接着进行服务过滤,接着进行任务规定,接着进行任务执行。

13. 根据权利要求 12 所述的基于计算机的系统,该基于计算机的系统被设置为允许正在进行任务规定的用户(110)把服务发现或服务过滤,或者它们二者重复进行一次或者多次。

14. 根据权利要求 13 所述的基于计算机的系统,该基于计算机的系统被设置为允许正在进行任务执行的用户(110)把服务发现或服务过滤或任务规定的任意组合重复进行一次或者多次。

15. 根据权利要求 12、13 和 14 中任何一个所述的基于计算机的系统,该基于计算机的系统进一步包括用户界面(1700-4100),该用户界面支持服务发现、服务过滤、任务规定以及任务执行的工作流。

16. 根据权利要求 15 所述的基于计算机的系统,其中,用户界面(1700)是图形用户界面,而且它包括下面的任意组合:

发现窗格,用于显示所发现的服务和过滤的服务;

详情窗格,用于显示从发现窗格上选择的服务的附加信息;

组成窗格,用于显示与将数据输出为制作者服务的服务相匹配的将数据输入接受为消费者服务的列表,其中,制作者服务的输出可以用作消费者服务的输入,从而将这些服务连接起来以组成任务;

构建窗格,用于通过显示在组成的任务中连接起来的服务的列表来构建任务;

信息窗格,用于显示任务计算操作消息和关于任务的执行进度的信息;

保存窗格,用于保存任务;

其中,所有窗格或窗格的子集可以同时对用户(110)显示,而且每个窗格上的动作导致自动更新显示在其他窗格上的有关信息。

17. 根据权利要求 16 所述的基于计算机的系统,该基于计算机的系统被设置为,允许用户或者通过从发现窗格上选择服务并选择构建功能,或者通过选择一对匹配服务并选择

构建功能,而移动到构建窗格。

18. 根据权利要求 16 所述的基于计算机的系统,该基于计算机的系统被设置为,允许用户 (110) 通过对任务规定装置自动确定的、在显示的服务的列表中可以进行服务的插入或删除的每个可能位置,选择显示为“+”号或“-”号的按钮,而对构建窗格上显示的组成的任务附加或者移除服务。

19. 根据权利要求 16 所述的基于计算机的系统,该基于计算机的系统被设置为,允许用户 (110) 在构建窗格上或者在组成窗格上,通过选择显示的执行按钮,而执行规定的任务。

20. 根据权利要求 15 所述的基于计算机的系统,其中,用户界面是图形用户界面,而且它包括包含下面的任意组合的选项卡:

发现选项卡 (1800),用于显示所发现的服务和过滤的服务;

详情选项卡,用于显示从发现选项卡上选择的服务的附加信息;

组成选项卡 (1900),用于显示与将数据输出为制作者服务的服务相匹配的将数据输入接受为消费者服务的服务的列表,其中,制作者服务的输出可以用作消费者服务的输入,从而将这些服务连接起来以组成任务;

构建选项卡 (2000,2100),用于通过显示在组成的任务中连接起来的服务的列表来构建任务;

信息选项卡 (2200),用于显示任务计算操作消息和关于执行进度的信息;

其中,在任何给定时间,只有一个选项卡对用户 (110) 可见,而且每个选项卡上的动作导致自动更新显示在其他选项卡上的有关信息。

21. 根据权利要求 20 所述的基于计算机的系统,该基于计算机的系统被设置为,允许用户 (110) 或者通过从发现选项卡 (1800) 上选择服务并选择构建功能,或者通过从组成选项卡上选择一对匹配服务并选择构建功能,而规定任务。

22. 根据权利要求 20 所述的基于计算机的系统,该基于计算机的系统被设置为,允许用户 (110) 通过对任务规定装置自动确定的、在所显示的服务的列表中可以进行插入或删除的每个可能位置,选择显示为“+”号或“-”号的按钮,而对构建选项卡上的显示的组成的任务附加或者移除服务。

23. 根据权利要求 20 所述的基于计算机的系统,该基于计算机的系统被设置为,允许用户 (110) 在构建选项卡上或者在组成选项卡上,通过选择显示的执行按钮,而执行规定的任务。

24. 根据权利要求 16 或 20 所述的基于计算机的系统,该基于计算机的系统被设置为,在用户 (110) 组成的任务中的服务之前或之后,自动附加任何、预定或用户定义的兼容可用服务组成,而且在所附加的任务可执行时,每当用户选择显示的执行按钮时,就将所附加的组成任务作为完整任务来执行。

25. 根据权利要求 16 或 20 所述的基于计算机的系统,该基于计算机的系统被设置为,允许用户 (110) 将任务保存为宏序列,该宏序列规定该宏序列的各个操作可以接受的服务条件。

26. 根据权利要求 15 所述的基于计算机的系统,其中,用户界面 (2400-2600) 是图形用户界面,而且它包括:第一窗格,用于显示所发现的服务和过滤的服务;以及第二窗格,用

于显示用户 (110) 规定的任务,并且该用户界面包括用于在任务的历史中来回移动的多个控件以及用于在每当可能时执行任务的控件。

27. 根据权利要求 26 所述的基于计算机的系统,其中,用户界面被设置为,当在任务的服务序列中针对选择的服务仅有一个正确位置时,自动将该选择的服务设置在该正确位置处。

28. 根据权利要求 27 所述的基于计算机的系统,其中,用户界面被设置为,每当可能时,包括在对于一个可以执行的任务、只能以单一方式组合可用的过滤服务时,自动使任务完整,其中,完整的任务是可执行的任务。

29. 根据权利要求 15 所述的基于计算机的系统,其中,用户界面 (3400-3800) 是 web 客户机。

30. 根据权利要求 29 所述的基于计算机的系统,该基于计算机的系统被设置为,在执行任务时,使浏览器弹出新窗口,或将该浏览器重定向到一个新链接,该新链接允许用户 (110) 输入与执行任务有关的信息或关于该执行的任务的反馈。

31. 根据权利要求 15 所述的基于计算机的系统,其中,用户界面是命令行界面 (4100),该命令行界面被设置为,每次在用户 (110) 输入可以与发现的和 / 或过滤的服务匹配的文本时,在用户按下预定键后,插入匹配的服务,或者,如果存在多个匹配,则提供它们的列表,以使用户通过进一步输入与多个服务中的一个服务匹配的文本,进行选择。

32. 根据权利要求 31 所述的基于计算机的系统,其中,用户通过命令行界面选择的服务序列包括所规定的可执行任务,该基于计算机的系统被设置为,通过用户 (110) 按下预定键来执行该任务。

33. 根据权利要求 11 所述的基于计算机的系统,其中,发现模块、过滤模块、组成模块以及执行模块和用户界面被设置为在同一个计算设备上执行。

34. 根据权利要求 33 所述的基于计算机的系统,其中,组成模块包括推理引擎,该推理引擎被设置为识别在给出服务的语义描述的情况下在任务中可以成功执行的各服务的所有可能的成对组合。

35. 根据权利要求 34 所述的基于计算机的系统,其中,推理引擎识别在给出服务的语义描述的情况下可以成功执行的各服务的所有可能组成。

36. 根据权利要求 35 所述的基于计算机的系统,该基于计算机的系统被设置为利用包括计算资源的上限、计算时间的上限、组成中的服务数量的标准,或者利用要计算的服务或服务组成的成对组合的预定总数,限制推理引擎返回的服务或服务组成的可能成对组合的数量。

37. 根据权利要求 36 所述的基于计算机的系统,其中,推理引擎包括组成逻辑,其中该组成逻辑是一个规则,或一组规则,或一组逻辑语句。

38. 根据权利要求 33 所述的基于计算机的系统,其中,发现模块被设置为使用本地语义服务描述数据库,或访问远程语义服务描述数据库,并利用通过其发现服务的发现机制的特定服务 ID,从这些数据库中检索所发现的服务的语义服务描述。

39. 根据权利要求 38 所述的基于计算机的系统,其中,发现机制是 UPnP,而 ID 是 UPnP 的唯一设备名。

40. 根据权利要求 33 所述的基于计算机的系统,该基于计算机的系统被设置为使用

UPnP 动作来检索指向服务的语义服务描述的一个或者多个指针。

41. 根据权利要求 11 所述的基于计算机的系统,其中,用户界面被设置为在与执行发现模块、过滤模块、组成模块以及执行模块的计算设备不同的计算设备上运行。

42. 根据权利要求 41 所述的基于计算机的系统,其中,用户界面是被设置为利用 http 与发现模块、过滤模块、组成模块以及执行模块通信的 web 客户机 (3400-3800),该 web 客户机是浏览器。

43. 根据权利要求 41 所述的基于计算机的系统,其中,用户界面是被设置为在用户的计算设备上执行的并通过发现模块、过滤模块、组成模块以及执行模块的程序设计界面分别与发现模块、过滤模块、组成模块以及执行模块通信的应用程序。

44. 根据权利要求 11 所述的基于计算机的系统,其中,用户的计算设备包括台式计算机、膝上型计算机、笔式计算机、PDA 或移动电话中的任何一种。

45. 根据权利要求 9 所述的基于计算机的系统,其中,服务包括一个或者多个界面,该一个或者多个界面被设置为使用户 (110) 在执行该服务之前或在执行该服务的过程中与该服务交互。

46. 根据权利要求 11 所述的基于计算机的系统,其中,服务 (1302) 包括如下任何一种:

“通过显示器观看”,允许用户 (110) 通过显示器观看用户规定的文档,并控制如何显示它;

“通过多个显示器观看”,允许用户通过一个以上的显示器观看用户规定的文档,并控制如何显示它;

“地图观看器”,允许用户观看用户规定位置的地图,并控制如何显示它;

“地图路线选择器”,允许用户观看从用户规定的位置到预定的或用户规定的位置 / 从预定或用户规定位置到用户规定的位置的路线的地图或文本描述,并控制如何显示它;

“实例提供服务”,在执行时,提供语义实例,其中语义实例是在本体中描述的类的实例;

“打印”,允许用户通过打印机以预定方式以特定格式打印用户规定的文档或语义实例,或者控制如何打印它;

“传真”,允许用户将用户规定文档的传真发送到用户规定的传真号码,并控制如何传真它;

“播放视频”,允许用户播放用户规定的视频内容,并控制如何播放它;

“播放音频”,允许用户播放用户规定的音频内容,并控制如何播放它;

“数码照片像框”,允许用户显示用户规定的图像内容,并控制如何显示它;

“位置确定”,自动确定用户操作的设备的位置;

“目录发布器”,允许用户从同一种类型的一组语义实例中选择一个语义实例;

“电话拨号器”,允许用户拨用户规定的电话号码;

“TC 存储器”,允许用户存储或检索文件;

“拷贝到可拆卸存储器”,允许用户将一个或一组文件存储到预定可拆卸存储介质,并控制如何保存它们;

“电子邮件处理器”,允许用户将具有附加的用户规定文档的电子邮件发送到用户规定

的电子邮件地址,并控制如何发送它;

“列表选择器”,允许用户从同一种类型的语义实例列表中选择语义实例;

“特性选择器”,允许用户从语义实例中选择一个特性;

“你的输入”,允许用户输入语义对象的属性值;

“实例观看器”,允许用户检查并修改语义实例;

“本地实例选择器”,允许用户选择计算设备本地的一个或者多个语义实例;

“语义实例拷贝器”,允许用户将语义实例拷贝到本地计算设备内;

“L 一便条”,允许用户阅读并留下关于位置的便条;

“数据库发布者”,从数据库发布全部或部分表格;

“地点信息”,提供关于地点的信息;以及

“宿主服务”,利用客户机端的浏览器来提供包括以下项目的服务:“上载文件”,允许用户从客户机设备上载文件;“到输入的 URL”,允许用户输入 URL;以及“通过浏览器观看”,允许用户通过客户机设备观看文件。

47. 根据权利要求 9 所述的基于计算机的系统,该基于计算机的系统还包括用于基于与服务相关联的语义服务描述发现服务的多个服务发现机制,

其中,用于建立、移除以及管理服务的装置包括如下任何一种或它们的组合:

用于通过发现服务的一个或者多个所述发现机制,使服务的语义服务描述可用的装置;

用于改变发现机制以便使用的装置;

用于保持语义服务描述的装置;

用于将语义服务描述恢复到发现机制之一的装置;

用于从发现机制移除语义服务描述的装置;以及

用于改变包括语义服务描述的有效期、调用、限制以及访问控制的发布参数的装置。

48. 根据权利要求 47 所述的基于计算机的系统,其中,用于建立、移除以及管理服务的装置包括如下任何一种或它们的组合:

“PIPE”,用于提供本地或远程 API,以发布和管理服务,并对用户(110)提供用户界面,以发布并管理服务;

“应用程序的语义实例剪贴程序”,用于在实例对应用程序可用时,发布语义实例,并允许用户管理服务提供过程;

“具有实例提供服务建立过程的实例选择器”,用于允许用户发布用户为实例选择器服务选择的语义实例,并允许用户管理服务提供过程;

“白洞”,用于允许用户通过用户界面发布来自本地计算设备的对象作为语义实例,并允许用户管理服务提供过程;

“公用目录”,用于允许用户通过设备发布语义实例,并允许用户管理服务提供过程;

“库”,允许用户通过服务调用过程发布语义实例,并允许用户管理服务提供过程;

“图像/音频/视频服务控制”,允许用户发布设备建立的语义实例,或来自设备的实例,并允许用户管理服务提供过程;

“可拆卸介质发布者”,用于允许用户通过插入计算设备中的可拆卸介质发布语义实例,并允许用户管理服务提供过程;以及

“传真”，用于发布收到的文档，并允许用户管理服务提供过程。

49. 一种基于计算机的方法，该基于计算机的方法包括：

进行任务计算，从而使用户能够通过将设备（304）或以计算机为介质的服务的功能性组合在一起来定义任务并执行这种任务，所述功能性包括计算应用程序（306）以及通过远程过程调用可用的电子服务（312）或以前定义的任务；

其中，所有功能性对用户（110）呈现为服务（1204-1208），而且以服务描述语言来表示每个服务；

其中，每个服务具有至少一个与它相关联的语义服务描述，以及

通过使其一个或者多个语义描述可用或不可用，可以使该服务可用或不可用；

该基于计算机的方法进一步包括：

作为服务发现，发现（510）可用服务；

作为服务过滤，利用一个或多个标准来过滤（512）服务，以从发现的可用服务中选择服务子集；

作为服务规定，通过将过滤的服务连接起来来规定（514）与用户合作（110）的可执行任务；以及

作为任务执行，执行（516）规定的任务。

50. 根据权利要求 49 所述的基于计算机的方法，该基于计算机的方法还包括创建、移除和管理服务。

51. 根据权利要求 49 所述的基于计算机的方法，其中，自动地，或者根据用户（110）的指示，把规定的任务保存为新可用服务。

52. 根据权利要求 49 所述的基于计算机的方法，其中，发现服务的过程利用服务发现协议，服务发现协议包括如下任何一种，或者它们的任意组合：

UPnP、UDDI、本地服务资料库、Jini、蓝牙 SDP、会合以及红外线。

53. 根据权利要求 52 所述的基于计算机的方法，其中，在过滤所发现的的服务的过程中使用的标准包括如下任何一种，或者它们的任意组合：

用户概况、手头任务、用户设备特性、用户位置、用户运动状态、用户网络连通性、用户规定的关键字、在被总体看待时一组服务的特征、单独服务特征。

54. 根据权利要求 53 所述的基于计算机的方法，其中，规定可执行任务的过程包括如下任何一种，或者它们的任意组合：

自动规定可执行的任务，或者

用户基于与计算机的可视、语音、文本、盲文、触觉的用户交互中的任一种或它们的任何组合，来交互地规定可执行的任务。

55. 根据权利要求 54 所述的基于计算机的方法，其中，执行规定的任务的过程利用服务执行机制，该服务执行机制包括如下任何一种，或者它们的任意组合：web 服务调用、UPnP 动作、CORBA、RMI、RPC、DCE、DCOM、红外线 IR、本地功能调用以及本地对象调用。

56. 根据权利要求 50 所述的基于计算机的方法，其中，建立、移除以及管理服务的过程基于图形用户界面以及被设置为在用户操作的设备上执行的应用程序的任意组合。

57. 一种基于终端设备的方法，该方法包括以下步骤：

提供用户界面，该用户界面使用户能够通过将设备（304）或以计算机为介质的服务的

功能性组合在一起来定义任务并执行这种任务,所述功能性包括计算应用程序(306)以及通过远程过程调用可用的电子服务(312)或以前定义的任务,

通过将各功能性关联有语义服务描述来将所有功能性对用户呈现为服务(1302),

作为服务发现功能,发现可用服务;

作为服务过滤功能,利用一个或多个标准来对服务进行过滤以从所发现的可用服务中选择服务子集;

作为组成功能,指示来自服务发现的匹配服务的列表并使用户能够将这些服务连接起来以组成可执行任务;

作为任务构建功能,显示在组成的任务中连接起来的服务的列表;以及

执行任务。

58. 根据权利要求 57 所述的基于终端设备的方法,其中,或者通过从服务发现功能选择服务并选择任务构建功能,或者通过在组成功能中选择一对匹配服务并选择任务构建功能,用户可以转到任务构建功能。

59. 根据权利要求 57 所述的基于终端设备的方法,其中,用户界面使用户能够通过对组成功能自动确定的、在显示的连接起来的服务中可以进行服务的插入或删除的每个可能位置,选择显示为“+”号或“-”号的按钮,来对显示的组成的任务添加或者去除服务。

60. 根据权利要求 57 所述的基于终端设备的方法,其中,正如自动确定的那样,在任务构建功能上或者在组成功能上,在任务变得可执行时,通过选择显示的执行按钮,用户可以执行组成的任务。

61. 根据权利要求 57 所述的基于终端设备的方法,其中,系统在用户组成的任务中的服务之前或之后,自动附加任何、预定或用户定义的兼容可用服务组成,而且在所附加的任务可执行时,每当用户选择显示的执行按钮时,就将所附加的任务作为完整组成任务来执行。

62. 根据权利要求 57 所述的基于终端设备的方法,其中,用户界面自动将选择的服务设置在组成任务中的服务序列中的恰当位置。

63. 根据权利要求 62 所述的基于终端设备的方法,其中,在对于组成的可执行任务只能以单一方式组合过滤服务时,用户界面将自动完成组成的任务。

64. 根据权利要求 57 所述的基于终端设备的方法,其中,用户界面是 web 客户端。

65. 根据权利要求 57 所述的基于终端设备的方法,其中,在执行任务时,用户的浏览器弹出新窗口,或将该浏览器重定向到一个新链接,该新链接允许用户输入与执行任务有关的信息或关于该执行的任务的反馈。

66. 根据权利要求 57 所述的基于终端设备的方法,其中,用户界面是命令行界面,其中,组成功能和任务构建功能包括:每当用户键入可以与发现的和/或过滤的服务匹配的文本时,在用户按下预定键后,用户界面插入匹配的服务,或者,如果存在多个匹配,则用户界面提供服务的列表,以便用户通过进一步键入与多个服务中的一个服务匹配的文本,进行选择。

67. 根据权利要求 66 所述的基于终端设备的方法,其中,用户选择的服务的序列包括可执行任务,而且可以随后通过按下预定键执行它。

68. 根据权利要求 57 所述的基于终端设备的方法,其中,服务包括如下任何一种:

“通过显示器观看”，允许用户通过显示器观看用户规定的文档，并控制如何显示它；

“通过多个显示器观看”，允许用户通过一个以上的显示器观看用户规定的文档，并控制如何显示它；

“地图观看器”，允许用户观看用户规定位置的地图，并控制如何显示它；

“地图路线选择器”，允许用户观看从用户规定的位置到预定的或用户规定的位置 / 从预定或用户规定位置到用户规定的位置的路线的地图或文本描述，并控制如何显示它；

“实例提供服务”，在执行时，提供语义实例，其中语义实例是在本体中描述的类的实例；

“打印”，允许用户通过打印机按预定方式以特定格式打印用户规定的文档或语义实例，或者控制如何打印它；

“传真”，允许用户将用户规定文档的传真发送到用户规定的传真号码，并控制如何传真它；

“播放视频”，允许用户播放用户规定的视频内容，并控制如何播放它；

“播放音频”，允许用户播放用户规定的音频内容，并控制如何播放它；

“数码照片像框”，允许用户显示用户规定的图像内容，并控制如何显示它；

“位置确定”，自动确定用户操作的设备的位置；

“目录发布器”，允许用户从同一种类型的一组语义实例中选择一个语义实例；

“电话拨号器”，允许用户拨用户规定的电话号码；

“TC 存储器”，允许用户存储或检索文件；

“拷贝到可拆卸存储器”，允许用户将一个或一组文件存储到预定可拆卸存储介质，并控制如何保存它们；

“电子邮件处理器”，允许用户将具有附加的用户规定文档的电子邮件发送到用户规定的电子邮件地址，并控制如何发送它；

“列表选择器”，允许用户从同一种类型的语义实例列表中选择语义实例；

“特性选择器”，允许用户从语义实例中选择一个特性；

“你的输入”，允许用户输入语义对象的属性值；

“实例观看器”，允许用户检查并修改语义实例；

“本地实例选择器”，允许用户选择计算设备本地的一个或者多个语义实例；

“语义实例拷贝器”，允许用户将语义实例拷贝到本地计算设备内；

“L- 便条”，允许用户阅读并留下关于位置的便条；

“数据库发布器”，从数据库发布全部或部分表格；

“地点信息”，提供关于地点的信息；以及

“宿主服务”，利用客户机端的浏览器功能性来提供服务，该服务包括：“上载文件”，允许用户从客户机设备上载文件；“到输入的 URL”，允许用户输入 URL；以及“通过浏览器观看”，允许用户通过客户机设备观看文件。

基于计算机的系统、方法和终端设备

技术领域

[0001] 本发明总体上涉及个人和遍布计算,更具体来说,涉及准备和执行计算任务。

背景技术

[0002] 计算的当前状态:个人计算

[0003] 个人计算用户操作单一设备和访问/使用驻留在该设备中的应用程序的典范。个人计算要求用户对其计算环境和其机器上可使用的应用程序有充分的理解,从而使得博学的用户可以充分地使用可利用的资源来执行复杂的任务。这就是我们大多数人日常体验的计算;学习如何完成复杂任务的负担仍落在用户身上,用户必须理解在其机器上运行的每个应用程序和它们所支持的各项功能,必须在应用程序之间手动传送数据(剪切和粘贴),必须手动地调用各个应用程序和与所述任务有关的特定功能,并且最终必须将其全部注意力(和时间)都投入到复杂任务的执行中。完成复杂任务一方面依赖于用户对任务的理解,另一方面依赖于用户对可利用资源(设备和应用程序)的理解,以使用户自己可以将它们结合到用户将要执行的工作流中,所述工作流的最终结果将是所完成的任务。

[0004] 早期的个人计算方法的示例

[0005] 我们简要讨论一些示例,对它们的技术方面进行概述并指出它们的局限性。这些示例显示了个人计算环境的更为面向任务的景象。

[0006] 在操作系统的一个特征中,当用户将音乐 CD 插入 CD 托盘中时,将弹出一个窗口,对用户提出从那个时刻起他/她可以执行的任务。这些选项的典型清单包括:

[0007] 播放音频 CD

[0008] 从 CD 复制音乐

[0009] 打开文件夹查看文件

[0010] 不采取动作

[0011] 这些选项中的每个还提及用以执行所述动作的应用程序。关注点放在将要执行的动作或任务而非用于执行任务的应用程序上。

[0012] 类似的,如果将一个数字摄像机连接到计算机上,那么就弹出带有下列用户可以执行的动作的弹出窗口:

[0013] 记录视频

[0014] 捕获视频

[0015] 不执行动作

[0016] 在任何一种情况下,操作系统都使用与特定事件(插入音乐 CD 或连接数字摄像机)的发生相关联的预定的动作或任务清单,以便在所述事件发生时,将相关的动作清单呈现给用户以遵照执行。在该情况下,系统的响应是硬连线的,并不包括超越它的灵活性,与作为触发事件的结果而被执行的可能动作有关的响应已经被编入系统中。换句话说,在数字摄像机连接到计算机上时,所述系统表现出可以发生的相同的动作集;操作系统的程序员已经为特定的事件准备了这个特定的动作清单。应用程序可以改变清单中的项目,但

对于最终用户就没有容易的办法来改变它。

[0017] 在操作系统的另一个特征中,用户有机会对动作进行选择。也就是说,对于每种下面的文件类型呈现给用户一个单独的任务列表:文档、图片、相册、音乐、音乐家、音乐图册(music album)、和视频。例如,如果文件类型为图片,那么就出现“图片任务”的列表:

[0018] 幻灯片放映查看(图片)

[0019] 联机顺序打印

[0020] 打印图片

[0021] 将图片设为背景

[0022] 将图片复制到 CD

[0023] 该任务列表也是预先编制的并且与特定的文件类型有关。对于最终用户没有容易的办法来改变该列表。如在另外的示例中那样,关注点集中在能被执行的任务上而不是集中在将要使用的用于执行所述任务的应用程序上。而且用户不需要知道哪个应用程序将用于所述任务。如果用户选择执行一个建议的任务,那么适当的应用程序将因此被初始化并被调用(启动)。

[0024] 在办公套装软件的示例中,智能标记特征是可利用的。智能标记特征使用编辑器高亮显示当前文档中的文本并给用户提供动作下拉菜单,所述动作可以利用文本表示的对象被执行。例如,如果所述文本表示一个名字,那么该特征可将与该名字关联的对象识别为人物,并且可提供下列可能的动作列表:

[0025] 发送邮件(给那个人)

[0026] 安排会议(与那个人)

[0027] 打开联系人(那个人的)

[0028] 创建联系人(用于那个人)

[0029] 通过识别所述文档中的字符串可能表示名字来启用所述选项。系统依靠文本的句法特征来识别该特殊的文本片断表示名字。注意,与典型的美国名字(例如, Lusheng Ji)不相似的字符串可能不会被识别为与人关联的名字。原因是用于将文本片断识别为名字的系统部分是非常简单的程序(脚本),它试图以文本的句法形式识别可容易地识别的模式。一旦文本的“性质”被(正确或不正确地)识别出来,例如,人,地址等,就将预先编制的可能动作的列表呈现给用户。对于应用程序编制者来说可以创建用于其它域和应用程序的智能标记,例如识别地址和调用地图应用程序等。

[0030] 现在讨论另一示例,其试图给用户提供更面向任务的景象。当用户在搜索引擎的搜索框内键入一个地址时,服务将返回(超过通常的搜索结果)到地图功能的链接,该链接如果继续将提供所述地址的地图。

[0031] 然而,并不见得用户可能正在搜索所键入的地址的地图。其它合理的可能性有:用户可能想要与该地址相关联的电话号码清单;或者如果该地址是企业,那么用户可能想要看到该企业的较好的商业局记录,或者检查在它附近的天气,等等。在搜索引擎当前的形式中,它将猜测键入的文本代表什么类型的“事物”(在该情况下为地址),并且它将返回与该类型的条目相关联的硬连线任务。

[0032] 总结

[0033] 上面提到的所有示例都表现出相似的特征:

[0034] 以某种方式来猜测用户的输入（文本或事件）的类型或性质；实际上系统依靠其句法特征试图推知串的意义（语义学）。

[0035] 系统对似乎可能的任务进行猜测，所述任务可能是用户希望针对所述输入来执行的；该猜测被硬连线到系统，所以系统不能实时有效地进行所述猜测，而是在对系统进行编程时，系统的程序设计人员远在用户与它交互之前就进行了猜测。

[0036] 根据用户利用合适的输入（无论系统在第一步骤猜测出什么）进行的选择（无论用户在第二步骤中选择什么），调用了适当的应用程序。

[0037] 现有技术的示例为用户增加了方便性，但也包括下列特征：

[0038] 功能已经被设计到应用程序中；应用程序的程序设计人员已经编制（硬件实现）了系统的响应。因此，这不是一个灵活的和可扩展的方法，因为在设计时间内就已确定了可能性的范围。

[0039] 系统已经限制了满足用户的动作和愿望的方法，它也不能精确地“感知”输入的性质（语义或意义）。虽然在每种示例中使用了不同的技术，但系统仍依赖于通过它的句法特征来正确地猜测所述输入的意义。

[0040] 在特定类型的输入导致单一动作（应用程序调用）的情况下，系统采用因-果（触发-响应）机制。复杂用户任务需要具有复杂的事件和动作序列的更加复杂的工作流，这对于在这些讨论的示例中使用的过分简单化的技术是不可能的（从技术上说）。

[0041] 总的来说，在个人计算的当前过渡状态中，我们注意到了这样的尝试：通过基于输入的句法特征猜测输入的意义，并且利用恰当的输入来调用正确的应用程序，来推测用户的任务；该方法是硬连线的并且是脆弱的。

[0042] 个人计算的局限性

[0043] 由于下面的原因，我们认为个人计算显示出本质的局限性：

[0044] 当个人计算达到海量时，期望普通的非专家用户拥有高级的计算经验和熟练程度是不切实际的。

[0045] 而且个人计算对于高级用户也是有问题的，因为它可能是耗时的。

[0046] 对于如下所述的遍布计算的新兴范例，个人计算是不适当的架构。

[0047] 对于普通用户，计算环境变得日益复杂。这是由于对用户可用的应用程序和设备的复杂度和激增引起的。虽然用户可以在计算环境中执行的任务已经快速增长了，但需要用户具有的熟练程度也提高了。计算机用户的用户基础已经快速扩展到具有非常少（或者根本不具有）的计算专门技术的用户，结果情况变糟了。计算业界的目标是使计算对非专家用户也是可用的。该目标与计算环境的日益增长的复杂性相冲突。

[0048] 即使最终用户掌握了丰富的专门知识，利用当今计算环境中可能的事物也需要花费相当可观的时间。这是因为高级用户已经变成其个人计算环境的“程序设计者”。考虑高级用户可能频繁执行的更加复杂的任务，例如决定参加她看到公布在网页上的谈话：用户必须找出谈话发生的地点的地址，检查与她的个人日程安排的冲突，找出到所述位置的方向，在日程安排应用程序中创建条目等。这样做要求在应用程序之间进行剪切和粘贴（假定用户足够熟练，知道使用哪个应用程序来完成所述任务）和执行使用可利用的资源的精心策划的工作流以完成期望的复杂任务。如果用户需要在将来执行相同或相似的任务，那么她必须重新执行所有这些动作，结果情况就变糟了。

[0049] 个人计算,也就是用户拥有和操作运行用户应用程序并“容纳”用户数据的思路正在让步于具有较少明确限定边界的计算环境。当计算机永久连接到计算机网络时,本地和远程应用程序与数据之间的区别变弱了,或者甚至更差,它们对于计算机用户而言是分不清的。而且,用户可以访问不是计算机的设备(照相机、打印机、智能设备等)并与其发生交互,这是针对个人计算机仍然具有非凡的计算能力并能给用户的目标提供服务和帮助他们完成各种任务而言的。这个通常称作遍布计算环境的新计算环境要求对于用于在其中完成事情的问题采用根本不同的方法。首先,普通用户可能不知道在这种环境中什么是可能的或切实可行的,因为可利用的资源(设备和应用程序)可能在不断地变化,而且此外用户不能依赖灵活且统一的用户界面来方便地管理与这种环境的交互。换言之,在充满对于用户来说不是已知的设备和应用程序的设置中,个人计算方法是不可行的。

发明内容

[0050] 任务计算是用户如何与设备和服务交互的新的范例,它在使用计算设备的同时强调用户想要完成的任务,而不是如何完成它们。任务计算填补了用户想要做的与在他们的环境中可利用的服务之间的空白。任务计算显示出超过传统方法(例如当前的个人计算范例)的本质优点,也就是说,它对于非专家计算机用户更为合适,它对于所有类型的用户都可节约时间,并且尤其适合于新兴的遍布计算类型的计算环境。

[0051] 我们通过提供对下述的支持来讨论对作为支持任务计算(TaskComputing)的基础的任务计算环境(TCE)的需求:

[0052] 任务计算的工作流

[0053] 任务和服务的说明

[0054] 由最终用户(他们依靠任务和服务的说明)进行的任务的指定、执行和再使用性

[0055] 包括由最终用户进行的创建和删除服务的操作

[0056] 我们详述任务计算环境(TCE)实施例;当看作具体实现的系统时,这样的实施例包括下述各项:

[0057] 一个或多个任务计算客户机(TCC)

[0058] 一个或多个语义描述服务(SDS)

[0059] 一个或多个语义服务发现机制(SSDM)

[0060] 以及,可选择地包括一个或多个语义描述服务控制机制(SDSCM)

[0061] 我们详细地给出两个称作TCC I和TCC II的任务计算客户机实施例、多个语义描述服务和语义描述服务控制机制的实施例。

[0062] 通过任务计算的详细预排我们推断使用STEER的应用情况。

[0063] 本发明的上述方面可通过包括执行任务计算的计算机系统的基于计算机的系统、方法和计算机可读介质来实现,从而使得用户通过结合可用的功能而可以定义任务并能执行这种任务。

[0064] 根据本发明的第一方面,提供了一种基于计算机的系统,该基于计算机的系统包括:

[0065] 计算机系统,其执行任务计算,从而使用户(110)能够通过将设备(304)或以计算机为介质的服务的功能性组合在一起来定义任务并执行这种任务,所述功能性包括计算应

用程序 (306) 以及通过远程过程调用可用的电子服务 (312) 或以前定义的任务 ;

[0066] 其中,所有功能性对用户呈现为服务 (1302),而且利用语义服务描述的服务描述语言来在语义上表示每个服务 ;

[0067] 其中,每个服务具有至少一个与它相关联的语义服务描述,并且通过使其一个或者多个语义描述可用或不可用,能够使该服务可用或不可用 ;

[0068] 该基于计算机的系统进一步包括 :

[0069] 服务发现装置,其用于基于与可用服务相关联的语义服务描述发现该可用服务 ;

[0070] 服务过滤装置,其用于利用一个或多个标准来对服务进行过滤以从所发现的可用服务中选择服务子集 ;

[0071] 任务规定装置,其用于通过将过滤的服务连接起来以规定与用户合作的可执行任务 ;以及

[0072] 任务执行装置其用于执行所规定的任务。

[0073] 根据本发明的第二方面,提供了一种基于计算机的方法,该基于计算机的方法包括 :

[0074] 进行任务计算,从而使用户能够通过将设备 (304) 或以计算机为介质的服务的功能性组合在一起来定义任务并执行这种任务,所述功能性包括计算应用程序 (306) 以及通过远程过程调用可用的电子服务 (312) 或以前定义的任务 ;

[0075] 其中,所有功能性对用户 (110) 呈现为服务 (1204-1208),而且以服务描述语言来表示每个服务 ;

[0076] 其中,每个服务具有至少一个与它相关联的语义服务描述,以及

[0077] 通过使其一个或者多个语义描述可用或不可用,可以使该服务可用或不可用 ;

[0078] 该基于计算机的方法进一步包括 :

[0079] 作为服务发现,发现 (510) 可用服务 ;

[0080] 作为服务过滤,利用一个或多个标准来过滤 (512) 服务,以从发现的可用服务中选择服务子集 ;

[0081] 作为服务规定,通过将过滤的服务连接起来来规定 (514) 与用户合作 (110) 的可执行任务 ;以及

[0082] 作为任务执行,执行 (516) 规定的任务。

[0083] 根据本发明的第三方面,提供了一种基于计算机的系统的终端设备,该终端设备包括控制器,该控制器被编程为使该终端设备执行包括以下在内的操作 :

[0084] 提供用户界面,该用户界面使用户能够通过将设备 (304) 或以计算机为介质的服务的功能性组合在一起来定义任务并执行这种任务,所述功能性包括计算应用程序 (306) 以及通过远程过程调用可用的电子服务 (312) 或以前定义的任务 ;

[0085] 通过将各功能性关联有语义服务描述来将所有功能性对用户呈现为服务 (1302),

[0086] 作为服务发现功能,发现可用服务 ;

[0087] 作为服务过滤功能,利用一个或多个标准来对服务进行过滤以从所发现的可用服务中选择服务子集 ;

[0088] 作为组成功能,指示来自服务发现的匹配服务的列表并使用户能够将这些服务连接起来以组成可执行任务 ;

- [0089] 作为任务构建功能,显示在组成的任务中连接起来的服务的列表 ;以及
- [0090] 执行任务。
- [0091] 通过参考作为其一部分的附图的说明,这些连同其它的方面和优点将是非常明显的,且它们存在于如之后更加充分描述的和所要求的结构和操作的细节中,其中始终用相同的标号表示相同的部分。

附图说明

- [0092] 图 1 示出由设备和服务包围的用户。
- [0093] 图 2 示出第一个以用户为中心的视图。
- [0094] 图 3 示出第二个以用户为中心的视图。
- [0095] 图 4 示出任务和服务。
- [0096] 图 5 示出第一计算机工作流变型。
- [0097] 图 6 示出第二计算机工作流变型。
- [0098] 图 7 示出第三计算机工作流变型。
- [0099] 图 8 示出第四计算机工作流变型。
- [0100] 图 9 示出第五计算机工作流变型。
- [0101] 图 10 示出第六计算机工作流变型。
- [0102] 图 11 示出复合任务计算工作流。
- [0103] 图 12 示出语义服务实例映射和基础连接 (GROUNDING)。
- [0104] 图 13 示出并入新服务的第一方法。
- [0105] 图 14 示出并入新服务的第二方法。
- [0106] 图 15 示出基于 S 备忘录的最终用户系统开发 / 部署。
- [0107] 图 16 示出 TCE 实施例的一般架构。
- [0108] 图 17 示出用户界面完全版本。
- [0109] 图 18 示出第一用户界面选项卡版本。
- [0110] 图 19 示出第二用户界面选项卡版本。
- [0111] 图 20 示出第三用户界面选项卡版本。
- [0112] 图 21 示出第四用户界面选项卡版本。
- [0113] 图 22 示出第五用户界面选项卡版本。
- [0114] 图 23 示出第一用户界面流版本。
- [0115] 图 24 示出第二用户界面流版本。
- [0116] 图 25 示出第三用户界面流版本。
- [0117] 图 26 示出第四用户界面流版本。
- [0118] 图 27 示出在开始处的任务计算 (Task Computing)。
- [0119] 图 28 示出用户选择之后的任务计算。
- [0120] 图 29 示出左面板刷新之后的任务计算。
- [0121] 图 30 示出用户选择之后的任务计算的另一视图。
- [0122] 图 31 示出在左面板刷新之后的任务计算的另一视图。
- [0123] 图 32 示出任务计算自动完成 (auto-complete) 信息。

- [0124] 图 33 示出用户在带有自动的自动完成选项的任务计算中选择自动完成。
- [0125] 图 34 示出第一用户界面 web 版本。
- [0126] 图 35 示出第二用户界面 web 版本。
- [0127] 图 36 示出第三用户界面 web 版本。
- [0128] 图 37 示出第四用户界面 web 版本。
- [0129] 图 38 示出第五用户界面 web 版本。
- [0130] 图 39 示出第一用户界面嵌套菜单版本。
- [0131] 图 40 示出第二用户界面嵌套菜单版本。
- [0132] 图 41 示出用户界面命令行版本。
- [0133] 图 42 示出语义实例类型之间的关系。
- [0134] 图 43 示出 TCE 实施例的架构。
- [0135] 图 44 示出 TCC I 内部结构。
- [0136] 图 45 示出 TCC I 主逻辑。
- [0137] 图 46 示出使用 IE 的发现和注册。
- [0138] 图 47 示出发现显示窗格。
- [0139] 图 48 示出带有服务图标的发现窗格。
- [0140] 图 49 示出细节窗格。
- [0141] 图 50 示出服务过滤。
- [0142] 图 51 示出带有“search web services”的发现窗格。
- [0143] 图 52 示出使用 IE 找出组成。
- [0144] 图 53 示出组成窗格。
- [0145] 图 54 示出构建窗格。
- [0146] 图 55 示出在语义层和句法层中的任务执行。
- [0147] 图 56 示出带有 web 服务支持的 web 服务的调用。
- [0148] 图 57 示出 TCC I 的服务调用流程。
- [0149] 图 58 示出带有服务控制 UI 的第一基于网络的 UI。
- [0150] 图 59 示出带有服务控制 UI 的第二基于网络的 UI。
- [0151] 图 60 示出带有服务控制 UI 的第三基于网络的 UI。
- [0152] 图 61 示出监视窗格。
- [0153] 图 62 示出保存窗格。
- [0154] 图 63 示出 TCC II。
- [0155] 图 64 示出在 SSD 中的多种自然语言支持。
- [0156] 图 65 示出银行服务。
- [0157] 图 66 示出 PIPE 管理器。
- [0158] 图 67 示出白洞推挽界面 (white hole drag and drop interface)。
- [0159] 图 68 示出在任务计算之前的能力金字塔。
- [0160] 图 69 示出在任务计算之后的倒转的能力金字塔。
- [0161] 图 70 示出应用电子家庭。
- [0162] 图 71 示出应用电子办公室。

[0163] 图 72 示出应用飞机场。

[0164] 图 73 示出应用咖啡馆。

[0165] 图 74 示出 :TCE 识别在网页上已经描述了什么,并且向用户提出一些给定当前内容和环境下的可能动作。

[0166] 图 75 示出 :在用户首先选择找出谈话位置的地址之后,TCE 找出该地址并向用户呈现新的可能动作的列表。

[0167] 图 76 示出 :在用户选择检查谈话位置在地图上的地址之后,TCE 呈现出修改后的动作列表,其带有选项以将到目前为止的用户动作保存为新任务,以在将来(再次)使用。

具体实施方式

[0168] 在本文中讨论任务计算和它的实施例。

[0169] 任务计算的概貌

[0170] 任务计算目的在于:帮助没有关于实现一定功能的环境的先验知识的用户使用由她支配的所述功能来实现复杂任务,并且不需要与该功能的资源进行交互或者理解该功能的资源。任务计算的基本要素是:用户知道和理解她的任务,并且任务计算环境(TCE)理解设备和服务以及知道如何与它们进行交互。用户和 TCE 两者进行结合,就能合作完成任意复杂的任务。

[0171] 在当今的计算环境中,用户由功能所包围,所述功能以设备或以计算机作为介质的服务的形式呈现给用户,以计算机为介质的服务例如有:在互联网上的电子服务(e-service),在用户操作的计算设备上运行的应用程序,或者支持特定功能的简单设备。这种设备和 e-服务的示例包括:电话、显示器、娱乐中心、电视、PDA、移动电话、音频播放器、传真机、打印机、天气服务、地图服务等。

[0172] 图 1 示出用户 110 由设备和服务包围的示例 100。图 1 所示的设备和服务包括:台式 PC、地图服务、PDA、传真机、MD 播放器、娱乐中心、天气服务、移动电话、联系人列表管理器、电视、便携式摄像机、显示器,以及电话。由这样的设备和 e-服务呈现的功能包括:听音乐、下载歌曲、观看流视频、听广播、提供联系信息、检查地图上的地址等。所有这些设备和服务被设计得通过用户直接与这些设备或服务进行交互(和/或操作这些设备或服务)使得它们的功能可为用户所用;例如,如果用户想通过为她正在参观的房间提供的电话给同事打电话并且同事的电话号码存储在用户的膝上型电脑的 MICROSOFT OUTLOOK 应用程序中,则用户必须启动 OUTLOOK 应用程序,查找想要的电话号码,然后在电话上手动拨号。即使当应用程序、e-服务和设备相互之间可以物理地通信,也就是在它们之间存在通信线路,但它们之间也不能以对用户任务是有意义的方式来交换数据,除非各个应用程序、服务、设备的设计者已经考虑到该特定的任务而对它们进行了设计。当面临着设备、e-服务和应用程序的剩余时,用户也不能执行任务利用来自所有这些资源的功能,除非所述设备、应用程序和 e-服务已经针对该任务进行了设计。而且,粗心的用户常常不知道哪种任务是可能的。

[0173] 图 2 是图 1 的“用户居中”的远景的面向历史技术的(technology-historical-oriented)视图 200。在技术层面,通过各种技术和协议进步,设

备、e- 服务和应用程序的功能已经变得可用,所述技术和协议进步包括,但不局限于:诸如 MICROSOFT. NET 的具有通过架构所暴露的功能的操作系统和应用程序,诸如 Jini 或 UpnP 的发现协议,web 服务、具有 WLAN(无线局域网)功能的设备等。

[0174] 尤其是,图 2 表明技术和协议对于用户 110 容易地访问设备和 e- 服务已经成熟。关于网络 202 方面,它们已经由独立网络发展成了因特网和无限网络。关于设备 204 方面,它们已经从独立设备发展成网络设备直至 UPnP 和 JINI。关于网络结构 206 方面,它们已经从服务器/客户机发展成 P2P/Ad-hoc。关于 web 技术 208 方面,它们已经从作为用于人们消费的“眼球式 web”发展成作为交易 web、语义 web(机器可操作、可读取的)、和可理解的 web 的 web 服务。关于 OS 和应用程序 210 方面,它们已经从面向对象发展成 .NET 和其它远程对象和远程过程调用技术。

[0175] 图 3 示出被由设备、OS 和应用程序、以及 e- 服务所提供的功能包围的用户,所述功能以服务的形式呈现给用户;服务是在任务计算范例中对于用户可见的抽象。也就是说,图 3 示出用户 110 被 web302、设备 304 和 OS/ 应用程序 306 包围的以用户为中心的视图 300。上面提到的设备 204 方面可以实现设备 304 的服务抽象,上面提到的网络 202、网络结构和 web 技术方面可以实现 web302 的服务抽象,而上面提到的 OS 应用程序 210 可以实现 OS/ 应用程序 306 的服务抽象。

[0176] 图 4 示出图 3 中所述思想的集中。基于设备的服务、基于操作系统和应用程序的服务、基于因特网的服务提供了功能,例如:播放(音频)、加入到 Outlook、天气信息等,这些都作为服务抽象并以将它们从提供它们的物理系统中分离出来的方式呈现给用户,使得用户可以“混合和匹配”它们,以满足任何她觉得合适的目的,例如在电视上观看来自她的 Outlook 的联系人地址的天气信息。

[0177] 更为具体地,图 4 示出任务和服务 400。如图 4 所示,来自 web302、设备 304 和 OS/ 应用程序 306 的输入分别导致 web 可访问服务 312、设备动作 308 和 OS 应用 310,所有这些都支持任务 314 的实现。

[0178] “任务计算”(Task Computing)被定义为填补任务(用户想要做的)与服务(可为用户所用的功能)之间的空隙的计算。

[0179] 任务计算环境(Task Computing Environment)(TCE)是支持任务计算的框架,其通过为下述内容提供支持来实现:

[0180] 任务计算的工作流

[0181] 任务和服务的描述

[0182] 由最终用户进行的任务的指定、执行和再利用性(依赖于任务和服务的描述)

[0183] 包括由最终用户进行的服务的创建和扇出的操作

[0184] 我们使用术语“任务”来指用户想要执行的动作(或一系列动作)。在此意义上,任务是用户设想的(潜在复杂的)动作,其在整体上可能是或可能不是适于计算的。

[0185] 正在由计算系统(或它的集合)执行的任务应该被认为是用户任务概念化的计算上可行的实现。在任务计算中,用户承担将任务概念化(按她的想法)“翻译”成任务的计算实现的部分(或全部)责任。就用户可能进一步定义其初始的任务概念化或将其初始的任务概念化细化为计算的任务实现而言,该处理可能是在用户与 TCE 之间交互和协作的。

[0186] 任务的计算实现是单一服务(原子服务)或在工作流中组织(构成)的一组服务

（复合服务）。这样，如果任务与服务精确地匹配（也就是，如果所述服务就是用户确实想要做的），则任务由单一服务构成；或者任务可以由多个服务的工作流构成。我们将原子服务和和复合服务都简单地称为服务。

[0187] 服务（原子或复合的）存在（在某处）的事实并不意味着它们可由特殊的用户使用来完成。换言之，服务本身的可用性没有启用由它自己进行的任务计算。这样的服务需要根据环境、特殊用户的需求和该服务的状态来进行发现和过滤，以用户利用它们可以创建任务的方式呈现给用户，最终为了用户并通过用户的参与执行所创建的任务。

[0188] 从用户角度看，任务计算涉及遍历任务计算工作流的这些步骤，在此意义上，服务的处理（发现、过滤、任务规定、任务执行）和可选操作是任务计算的最终用户视图。接着我们详细说明任务计算工作流。

[0189] 任务计算的工作流

[0190] 任务计算工作流描述了用户在执行任务计算时所经历的处理（步骤序列）。虽然工作流可以不必整体地显现在用户界面（User Interface）中（在该情形下，工作流的每个元素不必由用户界面中的元素明确地表示出来），但支持该工作流及其变型是任何 TCE 所必需的元素。

[0191] 任务计算工作流包括下列功能：

[0192] 由用户的 TCE 进行的设备和所提供服务的发现（由所述设备或宽带网络 / 因特网）的发现。发现既指找到存在的服务，又指如何访问它们。为了任务计算起见，发现包括与所述服务相关联的说明、它们的使用和状态的发现。除了设备和服务外，发现可包括诸如任务的项，该任务由当前用户或其它用户预先定义。各种发现机制在后面进行讨论，但应该注意，显式的用户驱动搜索也是发现的一种模式。

[0193] 基于用户环境的对所述已发现的设备和服务的过滤。过滤的目的是将可能难以控制数量的已发现服务降至可管理的子集，该子集与用户的环境和当前情形（使用各种潜在的用户可选择的准则）和 / 或用户试图完成的任务更为相关。各种过滤机制和 / 或准则将在后面讨论，但应该注意，显式的用户驱动过滤（其可包括对已发现服务的列表进行的搜索）是一种过滤模式。

[0194] 任务规定，该过程中用户知道和理解她想要完成的任务，该任务规定通过将可利用的（过滤过的）服务（和任务）连接起来组成所述任务。该任务规定可以用各种方式（优选地使用图形用户界面，GUI）来实现。对于实现任务计算工作流的所述步骤不存在唯一的 UI，尽管我们将在后面提出一些示例。

[0195] 用户也可保存特定的任务以备将来使用。

[0196] 特定任务的任务执行，在该过程中，用户仅与任务的执行进行交互，以便为所涉及的服务的子任务的正确执行提供必需的输入并监视任务的执行，但所有底层的调用和与受影响的设备和系统的交互都留给支持任务计算框架的计算系统。

[0197] 应该注意，发现 > 过滤 > 任务规定 > 任务执行规定了用于任务计算的总体工作流；它的元素用于描述用户使用任务计算环境时的任务计算经历。从 TCE 的角度看，将 TCE 看作“提供”任务计算的计算系统，任务计算工作流的四个要素构成 TCE 应该支持的基本功能和它们的内部关系。

[0198] 附加的元素可被引入到任务计算工作流中。另外，这些元素中的一些可从 TCE 中

的用户接口方面消除,尽管这种步骤仍然非常可能在背景中发生。例如,用户可以选择跳过任务规定阶段,因为过滤阶段的输出可以按与用户任务匹配的方式呈现给用户,而不需要用户作进一步的说明。或者,用户可以跳过滤,因为所发现的服务(和任务)非常少,或者使用了预先编制的任务和服务库,进一步的过滤没有必要。作为任务计算工作流的一个步骤,通常不能省略任务执行,因为它表示任务计算用户经历的最终结果。

[0199] 在图 5-11 中我们给出了基本的任务计算工作流及其变型的模式图。

[0200] 在图 5 所示的任务计算工作流变型 (I) 500 中,执行服务或任务发现 510,然后执行服务或任务过滤 512,接着执行任务规定 514,然后执行任务执行 516。

[0201] 在图 6 所示的任务计算工作流变型 (II) 600 中,执行服务或任务发现 510,然后执行服务或任务过滤 512,接着执行任务规定 514,然后执行任务执行 516。在任务规定 514 完成处,控制可返回至服务或任务发现 510。

[0202] 在图 7 所示的任务计算工作流变型 (III) 700 中,执行服务或任务发现 510,然后执行服务或任务过滤 512,接着执行任务规定 514,然后执行任务执行 516。在任务规定 514 完成处,控制可返回至任务过滤 512。

[0203] 在图 8 所示的任务计算工作流变型 (IV) 800 中,执行服务或任务发现 510,然后执行服务或任务过滤 512,接着执行任务规定 514,然后执行任务执行 516。在任务执行 516 完成处,控制可返回至服务或任务规定 510。

[0204] 在图 9 所示的任务计算工作流变型 (V) 900 中,执行服务或任务发现 510,然后执行服务或任务过滤 512,接着执行任务规定 514,然后执行任务执行 516。在任务执行 516 完成处,控制可返回至任务过滤 512。

[0205] 在图 10 所示的任务计算工作流变型 (VI) 1000 中,执行服务或任务发现 510,然后执行服务或任务过滤 512,接着执行任务规定 514,然后执行任务执行 516。在任务执行 516 完成处,控制可返回至任务规定 514。

[0206] 在图 11 所示的复合任务计算工作流 1100 中,执行服务或任务发现 510,然后执行服务或任务过滤 512,接着执行任务规定 514,然后执行任务执行 516。所述复合任务计算工作流表明:在服务或任务发现 510 之后的每个功能完成处(即,在服务或任务过滤 512、任务规定 514 或任务执行 516 的完成处),控制可返回至任何前面的功能。

[0207] 两种主要类别的变型在于包括从任务规定和任务执行阶段中的每一个到工作流前面步骤的迭代。这些变型的合理性在于任务计算的用户驱动性质。在任务规定阶段中,用户可能决定:她需要重新细化或修改她的任务,附加的发现→过滤或者只有过滤阶段对于用户发现其他的潜在有用的服务或进一步缩小过滤阶段的输出(通过修改过滤准则)是必需的。类似的,在执行过程中,由网络问题、服务失败或在执行特定任务的子任务中的某些其它困难所致的失败(或某些其它例外)可能要求用户寻找可选的/可替换的服务(通过进一步的发现和/或过滤)或者对任务本身进行修改。任务计算环境应该允许多重这样的迭代和/或这种工作流的“混合”;例如,如在一个工作流中所建议的,用户可以经历任务执行→过滤→任务指定,然后,如在另一个工作流中所建议的,可经历任务规定→发现→过滤→任务规定→任务执行。通过简单地允许从过滤、任务规定、和任务执行中的每一个到任何前面阶段的多重迭代,我们已经对上面进行了说明,但我们觉得我们所用的说明(和附加的评论)更能表示每个工作流类别的合理性。

[0208] 接着我们提供一些关于任务计算工作流的每个组成要素的附加评论。任务计算环境可以以采用带有不同窗格的完整 UI 的形式将其本身显现给用户,所述不同窗格表示任务计算工作流的每个(或多个)步骤。另外,根据命令可出现弹出窗口,以向最终用户呈现附加的信息或者为了辅助用户提供必要的输入。此外,UI 的形式可根据客户机设备的类型(例如,PDA 与桌上型或膝上型电脑)和/或用户环境的性质(例如,遍布计算环境与 web 浏览)而不同。可以有采用语音、盲人用点字法、触觉等的界面。语音界面对于移动电话和车内应用程序尤其有用。

[0209] 发现

[0210] 发现涉及找到能被 TCE 消费/使用的服务。发现包括识别用于与这些服务进行交互的程序界面和这种服务的说明,包括它们的输入和输出参数、关于它们执行的功能的细节、使用它们的条件、它们的效果等等。

[0211] 由多种服务发现机制。TCE 可以实现多重服务发现机制,并且任何单一服务都可通过多重服务发现机制被发现。总的来说,对于给定的服务不存在优选的发现机制,但是,根据服务的位置和/或其说明,存在更加有效或更容易实现的机制。

[0212] 发现可采用多种模式进行。我们将区分主动发现模式、被动发现模式和混合发现模式。

[0213] 发现近似地涉及找到计算设备及其所有者所处环境(或情形)的功能方面。发现找到环境所提供的功能集。在此意义上讲,每种发现机制都表示某一环境(或一组环境)。例如,局部服务发现机制表示计算设备本身的环境,给出了在其上可用的功能。UPnP 发现机制表示计算设备所在的子网络的环境。采用 web 的发现机制可以在因特网上为某一社区而被建立起来,通过该社区计算设备的所有者具有会员资格。蓝牙和红外(IR)通信与无线 LAN 技术相比具有受限的范围。但是伴随蓝牙和 IR 的发现机制不必次于那些伴随无线 LAN(例如,UPnP)的发现机制。由于它们受限的范围,它们给出对计算设备及其所有者的更好的定位,从而导致更好确定发现机制必须提供的功能。

[0214] 主动服务发现机制

[0215] TCE 查询某一设施(计算机系统或其网络),例如代理程序(broker)、服务商、目录(例如 UDDI 服务器)、电话黄页类服务、服务发现系统-(UPnP、蓝牙 SDP、红外(IR)、Jini、Rendezvous、招呼(Salutation))-实现网络节点,或者某种用于列出服务清单的服务的搜索引擎。该查询可能是对由该计算系统提供的可用(已知)服务的完整清单的请求,或者提交给该系统的某种更窄的搜索查询。这种请求可能提供给类似文件共享对等网络的对等网络(而不是单一的计算机系统)。对等网络可以是宽对等网络(例如前面提到的系统)或更局部化的对等网络,例如在相同的子网中作为当前用户的 TCE 对等端。这种查询可以由 TCE 本身(和/或它的一个组件)或由用户、或者通过用户与 TCE 之间的协作来预先确定或动态地创建。该搜索可以基于关键字(由用户提供),或者它可以由用户的环境和任务的机器级理解来驱动。任何以上类型的搜索还可被看作“交互式/用户辅助发现”。该查询还可以在当用户处于执行发现或任务计算工作流的某一其它步骤的处理中时来实时确定。例如,当用户试图指定任务时,TCE 可能推断用户可能对具有某种可用于她当前的任务的其它(尚未被发现的)服务或任务感兴趣,在用户没有明确地对此进行请求的条件下,TCE 可能试图发现这样的服务或任务(另选地,用户自己可以明确地执行这样的附加发现)。后者

是 TCE 的重要特征 ;给定用户动态地改变环境的情况下, TCE 可以前摄地执行发现。

[0216] 其他类型的主动发现可以包括 :依靠预先编制的可用设备、服务和任务的清单进行的查找或查询,所述清单存储在本地 (在用户的客户机设备上), 或者存储在可由用户的客户机设备访问的另一对等设备 (例如, 一个朋友的使用红外通信链路的 PDA) 上 ;无论用户在线 (与网络相连) 或离线, 或者甚至在与其它发现模式相结合的情况下, 都可使用本地存储。

[0217] 被动服务发现机制

[0218] 被动发现通常包括某种服务发现系统, 例如 UPnP、蓝牙 SDP、Jini、Rendezvous、Salutation 等。在这种情况下, TCE 具备了理解 and 处理发现系统的协议和机制的能力 (TCE 可以用于多个这种系统); 结果 TCE 依靠连接到 (或接入到) 网络可以发现服务, 在该网络上实现这些系统中的一个 (或多个) 的服务可能向相同网络或子网上的其它设备甚至跨网络广播 (或组播) 它们的存在和可用性。在该类型发现的变型中, 可以发现使用任何前述协议的设备、服务和任务的代理服务系统都可由 TCE 访问 (用于查询), 或者可以自动地就所述设备、任务和服务的可用性来更新 TCE。

[0219] 混合服务发现机制

[0220] 混合发现模式也是可能的 ;这种模式可以包括主动和被动方法。一个示例是将 web 服务用作发现机制。该发现过程包括两个步骤。在第一步骤中, TCC 向注册处注册一回传 (callback) 界面 (例如, WSDL URL)。无论什么时候发生了变化 (新的服务、删除的服务等), 注册处都调用给定的回叫界面来通知 TCC 该变化。第二步骤是可选择的。尽管回叫机制在使用网络带宽过程中非常有效, 但它具有一些缺陷。例如, 如果 TCC 在防火墙后并且回叫界面不能从注册处访问, 就不能使用回叫机制。为此设计了第二步骤。TCC 以给定的间隔或任何需要的时候轮询注册处。并且注册处返回最后的轮询之后的所有变化或所有可用服务。注意第二步骤可被看作是独立的发现机制, 并且在该情况下, 该机制应被分类为主动方法。

[0221] 过滤

[0222] 在任何给定的时间, 并非所有发现的服务都与针对任务计算的用户相关。任务计算 workflow 过滤阶段的目的是将已发现的服务缩减为适合用户的当前环境的那些服务。有时, 该过滤的输出不会进一步缩减发现的输出。在当用户的环境变化时过滤的输出应该动态地更新 (有时是根据用户的明确请求) 的情况下, 过滤应该是动态的。

[0223] 我们广泛地使用术语 “用户的环境”, 意指为从发现的输出中选择 (过滤) 一服务的子集所使用的各种准则。这种准则可以包括下面任何一个或其结合 :

[0224] 用户概貌

[0225] 手头的任务

[0226] 用户设备特性

[0227] 用户位置

[0228] 用户动作状态

[0229] 用户网络连通性

[0230] 用户指定的关键字 (与搜索查询类似)

[0231] 当看作整体时一组服务的特征, 例如

[0232] 服务数量

[0233] 服务之间的相似性

[0234] 关于服务相关值的数学或逻辑公式的应用输出,用于表达用户认为相关的任何约束或准则。该类的示例包括在该组中执行所有服务所需要的总资源,例如总费用或总时间。

[0235] 个体服务特征,例如

[0236] 使用 / 访问服务的成本,例如费用和时间

[0237] 服务提供商

[0238] 服务的声誉

[0239] 输入和输出参数以及它们的值,包括为了表达用户认为相关的任何约束或准则关于这种值的数学或逻辑公式的应用。

[0240] 等等

[0241] 用户可以保存环境(作为一个或多个过滤准则的应用的环境),从而她可以用名字来查阅它们并且用名字自动地调用它们。虽然可以由 TCE 自动地处理过滤,但用户可通过键入关键字或从各种准则中进行选择来明确地驱动过滤处理。也有可能的是, TCE 的前摄过滤与用户驱动过滤并行地和 / 或交替发生。

[0242] 任务规定

[0243] 任务规定是这样的处理:在该处理过程中,最终用户将任务“翻译”成服务(原子服务或复合服务)。为此,用户将利用作为过滤处理的输出的服务。该用户驱动处理可被看作是指定服务的工作流的处理,所述服务在执行时将完成用户的任务。服务组成是任务规定的输出;任务规定是这样的处理:在该处理过程中,用户(独立的或与 TCE 协作)浏览过滤后的服务,以试图创建与用户思想中的任务相匹配的服务组成。有时我们可能将任务规定称作任务组成。

[0244] 任务 [R. M. 3] 指定依赖于可用服务的输入、输出和其它要求,以便仅向用户呈现切实可行的可能组成。这样的要求包括:输入 / 输出参数的类型和 / 或值,使用、访问、调用服务的条件,如果被调用服务可能具有的效果,等。任务规定的结果是包括整体上可被执行的多个服务的有效序列(在通常的情况下是有向非循环图)。有时,在用户确定现有的服务或任务与用户考虑的任务相匹配的情况下,所述序列可能仅是一个服务长。

[0245] 在规定任务的过程中,用户得到用户界面的支持,用户界面试图以这样的方式给用户呈现可用的服务:(a) 帮助用户创建能被正确地执行的服务组成和 (b) 帮助用户识别与她的任务最相关的服务。当用户的环境或者甚至当用户选择了不同的服务(在任务规定中)时,可用的或相关的服务应该被更新,在这种情况下,该呈现应该是动态的。关于这些目标存在多种方式来使 TCE 可以帮助用户。例如,根据成对的服务之间的输出和输入的匹配或可能部分匹配,TCE 可以以成对的清单呈现服务;典型地(并非唯一地),这种匹配依赖于服务和它们的输入和输出参数的语义。代替使用清单,TCE 可以突出显示可能先于(或跟随)用户已突出显示(或选中)的服务的服务;而且,为了这样做,TCE 依赖于服务和它们的输入和输出的语义。另选地,以根据服务的类型或任何输入或输出参数的类型或值、或者某些其它的服务特征(位置、费用、新颖程度、优先化、先决条件、效果等)组织的层次,TCE 可呈现过滤后的服务。通常,在任务规定处理的过程中,可用于过滤服务的任何或所有准则或手段(按照过滤阶段的讨论)都可以用于引导服务的呈现;这种准则或手段可以用于以

这种方式来组织已过滤的服务：使用户为了完成任务而更加容易地选择和匹配服务。

[0246] 可以将服务构成在工作流中的方式通常将依赖于服务的属性，例如它们的语义。具体来说，为了决定如何构成服务而考虑的参数包括下述的任何一项或全部：它们的数据输入和输出的数据类型、它们的数据输入和输出的语义类型、在执行服务之前必须满足的条件、以及成功执行服务的效果。切实可行的和可能的任务规定可以通过域专用逻辑（例如，如果服务打印文档，则该服务的域包括打印机、打印和文档）、或应用程序逻辑、或商业逻辑、或任务逻辑（通常以某种逻辑的声明的形式来表达）来进一步限定。前述的逻辑可以说明在从服务集规定任务时要注意的附加限定。这种限定可以包括：在相同的任务规定中服务是否可以使用一次以上，或者是否可以以组成的形式一起使用两个服务。

[0247] 我们将为了决定构成服务的方式而考虑的参数称作组成逻辑。

[0248] 考虑服务的语义的组成逻辑的一个简单示例为：使服务的语义输出与另一服务的语义输入相匹配。如果第一服务的语义输出与第二服务的语义输入相匹配，那么就存在一个第二服务紧随第一服务的局部工作流。由于语义，这样一种匹配不需要是精确的，这意味着：输入和输出参数的语义类型之间的匹配可以是不精确的（例如，根据语义近似的匹配），或者匹配可能考虑参数在相关的实体层次中的相对位置。

[0249] 应着重注意：用户可以保存这种组成并且（为它们代表的任务）给以名字以便将来使用（类似应用程序中的宏）。这种预先编制的任务规定还可以被保存和用作用户的TCE客户机程序中的书签。用户可以选择共享这样的任务规定。如先前所述，为了任务规定的目的，这种先前定义的宏可如新服务一样使用。

[0250] 另外一种任务规定模式可以不是用户驱动的，而是TCE驱动的。在这种情况下，给定来自过滤阶段的输出的可用服务，TCE可以提出能呈现给用户以待认可的服务组成或局部服务组成。这种组合可能是或可以不是被最终执行的（参见下面的任务执行的讨论）。TCE可以使用用户可使用来创建服务组成的任何准则或手段，如先前在本部分中所讨论的。

[0251] 任务执行

[0252] 任务执行是执行特定任务的步骤。对任务执行的输入通常是任务规定阶段的输出，所述输入为服务。该服务是在任务规定过程中由用户指定的复合服务，或者是与用户的任务对应的原子服务。TCE对执行而非用户负责，尽管可以要求用户在服务组成的执行过程中提供追加的输入或者进行动作以便管理该执行期间的故障或例外。由于这个原因，任务执行也可以被看作是用户监视的任务执行，从这个意义上说，处于被动角色的用户不是必需出现的，但在等待被执行的组合的输出时，她可能必须提供输入，在失败的情况下选择另选的过程，取消执行并返回到发现、过滤或任务规定等。

[0253] 任务执行需要向服务传送正确的输入（当以一种对最终用户透明的方式处理系统与协议间的必要翻译的时候），需要在必要时请求用户输入，并且需要执行涉及组成任务的所有操作。由于一方面，使用所调用的服务的语义信息和该服务的输入和输出参数来指定特定的任务，另一方面，该语义信息却并不用于执行任务，所以，这些任务会出现挑战。换句话说，服务组成是一种抽象，其本身并不是一段可执行的代码。TCE负责将服务组成翻译成一系列可以直接用来执行与任务有关的服务的动作和调用。挑战的第二个原因在于，服务组成是可行的（在任务规定阶段TCE试图达到的目标之一），却并不意味着它就可以被正确地执行。潜在的问题包括：有关组成服务的错误的语义信息、这些服务实际运行在其上的

系统的故障、以及与网络相关的问题等。

[0254] 因此,在任务执行期间 TCE 必须处理基础连接,即将任务规定抽象中的服务绑定到实际的基础服务及其参数,并将(用于输入和输出的)语义术语正确地翻译成服务所用的实际语义术语,或者在引用不同本体的语义对象间进行翻译。

[0255] 任务计算的主要技术挑战之一在于必须处理下面的事实,即服务是在语义层定义的,服务的输入/输出参数也是如此。然而,当我们调用服务时,必须向服务提供句法参数,因为服务是在句法层中实现的。基础连接是将语义实例变换为句法实例的过程。

[0256] 分离语义层和句法层的好处如下。在某些情况下,虽然几种服务具有相同的语义输入,然而由于它们实现的详细情况,语义实例可以不同。我们能够在语义级组合这些服务,而不用担心那些差异。

[0257] 图 12 显示了语义服务实例映射和基础连接 1200。假设我们有“Semantic Contact Instance”1202 和三种服务。“Weather info of”是一种返回给定地址当前天气状况的服务 1204;“Aerial Photo of”是一种显示给定地址卫星图的服务 1206;“Route from FLACP”是显示从 Fujitsu Laboratories of America, College Park 到给定地址的逐级地址目录的服务 1208。按照语义,三种服务都将接受地址作为输入,但是在实际的实现中,“Weather info of”1204 将邮政编码作为一个串来接受;“Aerial Photo of”1206 将街道、城市、州和邮政分区当作四个串来接受;而“Route from FLACP”1208 将街道、城市、州和邮政分区当作一个串来接受。

[0258] 如果我们不分离语义层和句法层,那么三种服务就完全不同。如果我们有一个地址对象,它最多只能匹配一种服务。而现在我们有一种更灵活的构架。首先,我们定义“Semantic Address Instance”1210 作为所有三种服务的语义参数。接着创建语义层的实例映射。该映射从联系人项中提取商务地址信息。图中,该映射由虚箭头表示。从“SemanticAddress Instance”开始,我们定义三种基础连接,每种用于一种服务。基础连接描述了将地址实例变换成句法参数的详情并且由实箭头表示。

[0259] 在例如 OWL-S 的语义服务描述语言 1212 中描述了语义实例映射和基础连接。因为手工创建映射和基础连接很复杂,所以使用了名为 OntoLink 的工具。OntoLink 具有强大的图形用户界面,其有助于用户定义两个语义实例之间的关系以及接着创建映射和基础连接。

[0260] OntoLink 是一种图形工具,用于指定:(a) 服务的语义描述与服务的 WSDL 描述(这个过程也称作基础连接)之间的映射,(b) 本体之间或属于不同本体的实例之间的映射。该工具的输出是用称作 XSLT(业界公知)的语言表达的规约。

[0261] OWL-S 是用于描述本体知识的工具,并且在业界公知。

[0262] 另外,TEC 需要监视服务执行以及准备处理异常和故障。异常和故障在系统层(该层提供服务及其对于用户的连通性和/或可访问性)或者服务层中都可能会发生。当故障在服务层中发生时,这样的故障要么来自对服务的不正确调用或者绑定,要么就可能来自其他的、未预知的要求和/或用于存取和/或使用服务必需的凭据。在所有这些情况的任何一种中,TCE 需要尝试纠正这些问题,这可能包括需要用户键入(可能)附加的信息和/或采取一些动作,这些动作可能包括任务的重新指定。

[0263] 最后,任务规定和执行可能以一种交错的方式发生,即,执行部分任务规定,然后

用户回到任务规定来扩展服务组成并执行它等等。换句话说,任务规定和执行可以不一定是任务计算工作流的顺序连续阶段,相反二者可能以一种用户驱动、用户控制的方式,或以一种半自动地但用户监视的方式并行地发生。

[0264] 任务计算用户界面的元素

[0265] 任务计算工作流的每一步必须在用户界面中显示。但这并不意味着必须存在一个不同的画面(在图形用户界面或 GUI 这种情况中)或者必须存在专用于任务计算工作流的每一步的其它元素。实际上,在单个用户界面元素中也可显示(或者调用)多个工作流步骤。

[0266] 用于 TCE 的 GUI 可以使用多重用户界面特征,以在发现、过滤、指定和执行中的每一阶段中来辅助用户。这些特征包括下拉菜单、搜索框、校验表按钮、能弹出文本的高亮图标、和文本上光标等。例如,当用户在服务的图形表述上移动光标时(在使用 GUI 时),可以与光标指示的服务相匹配的其它服务的图标可以突出显示。或者,当发现或过滤的服务对于用户正在尝试的任务不够时,“Search for more Services(搜索更多服务)”选项就可变成可用的。

[0267] 启用用于任务计算的服务

[0268] 当讨论任务执行的挑战时,我们讨论了语义服务实例映射和基础连接的概念。所讨论的方法也有助于用户将新的服务合并到任务计算中,即向现有服务提供适当的语义描述以便该服务可以用于任务计算。图 13 和 14 显示了两种可能的方法。

[0269] 图 13 显示了合并一种新的服务 (I) 1300。如图 13 所示,现有服务 1302 具有将语义层 1304 映射至服务参数的 OWL-S 1212 基础连接。假设我们有一种新的服务 1306 与语义层 1304 中的现有服务 1302 相同,启用用于任务计算的新服务 1306 需要创建新的 OWL-S 1212 基础连接用于新的服务 1306。然后我们就可以使用用于新服务 1306 的新基础连接 1212 来调用新服务 1306;新的基础连接 1212 将会从语义实例中派生出服务参数。这个机制可以用于增加新的服务实现,该服务实现用于与现有的服务工具相同的目的,或者该机制用于下面的情况:即当我们希望通过向新的服务的基础连接有效地重新定向现存语义服务描述来升级服务实现时。

[0270] 图 14 显示了合并新的服务 (II) 1400。图 14 讨论了具有两种服务 1302、1306 的情况,这两种服务都具有语义服务描述但是各自都为自己的语义层使用不同的本体。为了合并第二种服务 1306,我们那可以使用语义实例映射 1308 将第一种服务 1302 的语义实例 1304 映射至第二种服务 1306 的语义实例 1304。接着,为了调用第二种服务 1306 我们首先使用语义实例映射 1308 将第一种服务 1302 的实例翻译成第二种服务 1306 的实例,然后再使用第二种服务 1306 的基础连接 1212 来准备服务调用参数。

[0271] 最终用户通常都不使用这些机制,这些机制主要由服务提供商或第三方来使用,为了让这些服务可用于任务计算,他们对这些服务感兴趣。

[0272] 服务描述(我们称之为 S 备注)本身是一个文件,或者是可以从服务所有者和管理者向最终用户分发的文件指针。如同所述,S 备注不仅包括服务的语义详情,它还包括服务调用方法。如果最终用户有权使用 S 备注,他们就可以使用用于任务计算的服务。即使当执行任务计算时,最终用户也将自动地发现服务,也可能的是,通过接受到 S 备注或者指向它的指针,用户被明确地告知有新的服务。这样的 S 备注可由电子邮件来分发,或者可以

在带有服务的相同的子网中通过服务本身自动向用户分发,或者从一个用户向另一个用户传送。图 15 描述了这些机制。

[0273] 更特别地,图 15 显示了基于 S 备注 1500 的最终用户系统的开发 / 部署。如图 15 所示,最终用户 110 可以通过 UPnP 从设备 204 接受 S 备注、通过电子邮件从开发者 /IT 部门 1502 接受 S 备注,和 / 或从她自己的计算设备的 OS 和应用程序接受 S 备注。然后最终用户 110 调用设备 204 的服务、服务 1302 和 / 或她自己的计算设备上的服务来完成她的任务。最终用户可以和其它的最终用户 110 共享 S 备注。

[0274] 关于任务计算环境 (TCE) 的实施例

[0275] 当看作具体实现的系统时,任务计算环境 (TCE) 包括以下:

[0276] 一个或多个任务计算客户机 (TCC)

[0277] 一个或多个语义描述服务 (SDS)

[0278] 一个或多个语义服务发现机制 (SSDM)

[0279] 可选的:

[0280] 一个或多个语义描述服务控制机制 (SDSCM)

[0281] “任务计算客户机 (TCC)”是支持与任务计算环境相关联的工作流的客户机。

[0282] 图 16 显示了示例 TCE 实施例 1600 的总体体系架构。当然,这只是许多可能的 TCE 实施例中的一种。TCC1602 是软件,它允许用户感受任务计算,而且当由计算机执行时,使计算机执行任务计算功能。TCC 发现服务 1302、过滤针对用户环境的服务、向用户提供用户界面用于规定任务或者帮助用户进行任务规定,而且在任务规定后,它通过调用服务来执行任务,有时候还请求用户通过用户界面输入,并且最后它向用户呈现任务执行的结果。

[0283] 如图 16 所示,用户 110 调用指向任务计算客户机 (TCC) 1602 的任务并从任务计算客户机 (TCC) 1602 接收表示。TCC 1602 执行任务并接收服务 1302 和语义服务发现 1604。

[0284] TCC 1602 的功能包括但不限于下列方面:

[0285] 用户界面 (UI):它允许用户与 TCE 支持的主要功能 (发现、过滤、组合、执行) 交互。这些功能被映射到用户通过用户界面可以访问的模块或引擎。

[0286] 发现功能 (DF):找出 (服务及其) 相关描述;即使 DF 在执行与接收自身通知相反的操作而仅仅查询第三方时,上面所述内容也是正确的。TCC 可以拥有对应于服务发现技术 (例如 UPnP、Jini、蓝牙 SDP,等) 的多个 DF。

[0287] 过滤功能 (FF):基于用户偏好和环境对所发现的服务进行过滤。

[0288] 指定功能 (CF 或任务规定功能):在给定服务语义的情况下对服务进行匹配,并且允许用户为后续指定 (认为) 指定服务的任意且复杂的序列 (不一定是线性序列)。

[0289] 执行 / 监视功能 (EF):执行服务调用界面的实际调用,负责执行期间所调用服务之间的协调,并且管理执行期间所调用服务与用户之间的交互。

[0290] TCC 的最小要求功能是发现功能 (DF)、任务规定功能 (TF) 和执行 / 监视功能 (EF)。

[0291] 实现 TCC 功能的组件 (模块) 在物理上可以位于除最终用户的设备之外的系统中。例如,在我们的 PDA 实现之一中,除用户界面 (在此情形中为 Web 浏览器) 之外的所有 TCC 组件都在一单独的计算设备中执行。许多 TCC 功能自身都可以作为服务提供。

[0292] 实现 TCC 功能的组件 (模块) 可以被不同的实体拥有和 / 或操作。这些组件自身

可以再次被作为服务发现。

[0293] “语义描述服务 (SDS)” 1604 包括一个或多个服务调用界面及其相伴的语义层描述。

[0294] 服务调用界面可以由任何方法 / 机制实现以调用它们 (并与它们进行交互), 这些方法 / 机制包括但不限于 SOAP&WSDL、UPuP、JINI、RPC、RMI、CORBA、COM、DCOM、KQML、FIPA-ACL, 等。一项服务可以拥有多重服务调用机制。

[0295] 服务调用界面不需要与其描述的服务位于相同机器上, 也不需要位于相同的管理权限下。

[0296] 服务的语义层描述或“语义服务描述 (SSD)” 1604 可能包括使用任何适当的语义语言的对其 I/O、进程模型等进行的语义描述。服务的 SSD 应该包括 (至少):

[0297] 服务的语义层描述

[0298] 如何执行服务

[0299] 语义服务描述在物理上可以位于任何位置。它不必在物理上和逻辑上与服务界面位于相同的实体中, 它也不必由管理和 / 或拥有所实现的服务和 / 或其界面的相同权限管理。例如:

[0300] 任何人都可以给出指向服务调用界面或一组服务调用界面 (界面可以跨多个实体分布) SSD (语义服务描述) 1604。

[0301] TCC 可以在内部保持 SSD。

[0302] 可以使用任意合适的语义语言编写语义服务描述。这样的合适语言的一个示例是 OWL-S。但是, 服务的语义描述也可能是一种仅仅显式地指定服务的 I/O 的简单的专用语言, 在此情形中, 描述 I/O 的任何语义语言都满足条件 (例如, RDF、DAML、DAML-S, 等)。

[0303] 语义服务发现机制 (SSDM) 是一种查找 SSD 的机制。可选地 SSDM 查找服务的状态 (活动, 不活动, 等)。

[0304] 有很多 SSDM。TCC 可能使用多个 SSDM, 并且一项服务可能被多个 SSDM 发现。

[0305] 在普遍环境中, UPnP、蓝牙 SDP、Salutation、Jini、Rendezvous 是非常动态的 SSDM, 并且积极地通知其它客户机设备服务的出现。一般而言, 在使用这些机制之一用于发现在与 TCC 执行的计算设备位于相同子网中的服务时, 使用一个 TCC。

[0306] 为在 Internet 上发现服务, 可以使用用于服务的 UDDI 或与 GOOGLE 等效物来进行服务发现; TCC 将在给定搜索准则下查询它们以获得可用服务。

[0307] 代理通信语言 (Agent communication language) (KQML、FIPAACL……) 为在网络上查询代理有关代理所知道的服务提供词典。

[0308] 对于在与 TCC 在相同的计算设备上运行的服务而言, 存在不同的机制, 例如使用 OS 的注册表机制, 或者下文将讨论的其它机制。

[0309] 此外, 用户可以键入 / 剪切并粘贴 URL 或用于服务的 SSD 的某种引用 (例如 S 备注)。

[0310] 用户扫描设备上与服务相关的条形码, 以获得 URL、某种引用或用于服务的 SSD 的描述自身。

[0311] Web 服务 (或任意其它远程过程调用机制) 可以用作一种 SSDM 服务。

[0312] 用于先前或经常使用的服务的本地高速缓存可被视为一种 SSDM 高速缓存。

[0313] 用于公知服务的 SSD 服务的本地或远程数据库可被视为 SSDM 的数据库。发现模块使用查找服务所专用的 ID 来查询用于 SSD 的数据库,以执行查找服务。例如,假定通过 UPnP 找到作为 UPnP 设备的服务。发现模块尝试使用获得其 SSD 的标准方法。当此种尝试失败时,发现模块使用其 UPnP 的唯一设备名称 (UDN) 向本地和 / 或远程数据库查询其 SSD。

[0314] 定义 (SDSCM)

[0315] 语义描述服务控制机制 (SDSCM) 是一种机制,它允许某种实体 (并不一定是 TCC 的用户,甚至不必是一个人) 管理并控制 SDS 的供应 (创建、提供、保存、删除,等)。

[0316] SDSCM 包括但不限于:

[0317] Semantic Instance Scraper for Applications (应用程序语义实例刮刀程序)

[0318] White Hole Service (白洞服务)

[0319] PIPE (遍布实例提供环境)

[0320] 通知服务。这些服务并不使用订阅模型,而是通过使信息提供服务需要时可用来通知用户。

[0321] 虽然 SDSCM 并非任务计算的绝对要求,但是它对于创建更加动态的任务计算环境至关重要 (下文将给出更多实施例)。

[0322] 用户界面

[0323] 作为 TCC 的重要部分,用户界面向用户呈现发现结果,帮助用户指定并提交用于调用的任务。本部分将给出六个版本的 TCC 用户界面。

[0324] 完整版本

[0325] 图 17 显示了用于任务计算的一种可能的用户界面 1700,它在一个窗口中显示了所有的信息。这种复杂的用户界面供高级用户使用。

[0326] 在此特殊实施例 1700 中的 UI 中有五个窗格。左上面的窗格是发现窗格。它包含一个显示出能够发现的所有服务 (功能) 的列表。在该列表上方有一个下拉列表,它允许用户基于诸如字母顺序、服务类型、发现时间等顺序以不同的方式对服务列表进行排序。若要查看详细信息,用户可以单击他们感兴趣的服务,与该服务相关的详细资料将显示在窗口左下方的详细窗格中。中上方的窗格是组合窗格。在组合窗格中有几个组,它们包括对象名、两个下拉列表和两个按钮。左侧下拉列表中的服务被称为消费者,右侧下列列表中的服务被称为生产者。在每个组中,任意消费者都可以与任意生产者结合,因为来自生产者的输出可以用作消费者的输入。这就允许该 UI 可以在紧凑的显示区域呈现更多组合。当由左向右看消费者 / 生产者对时,其次序与其实际执行次序相反。这是有意而为的,这样该消费者 / 生产者对就可以读做一个英语句子 (例如,“Open”“Local File”),因为可以很容易地将生产者视为名词短语而将消费者视为对生产者起作用的动词或形容词短语。这就允许用户更加迅速地理解可用的任务的种类。对于其它的动词及其宾语名词次序相反的语言 (例如日语),应该以其执行次序显示它们。在下拉菜单对上方显示的对象名 (例如“File (文件)”、“Schedule (计划)”、“Contact (联系人)”等) 是生产者的输出的对象名。该窗格中的按钮是“Execute (执行)”和“Construct (构建)”。当用户单击“Execute (执行)”时,当前选择的消费者和生产者将结组成一个任务并执行。当用户单击“Construct (构建)”时,当前的消费者和生产者将成为进一步构建的基础并且显示在构建窗格中。构建窗格是位于右上方的窗格,用户在其中创建自定义的任务。在构建窗格中显示当前正在创建的任

务。用户可以通过在工具栏中单击按钮或单击窗格中的“+”或“-”按钮向服务序列的顶端或底端添加新服务或删除序列中的第一个 / 最后一个服务。用户可以单击工具栏中的“Clear(清除)”按钮以删除当前序列并重新启动创建过程。当序列结束时,用户可以单击“Execute(执行)”按钮来执行它。右下方的窗格是信息窗格。在信息窗格中显示 TC 操作消息。并且对于每一步执行,都会向信息窗格新添一个显示当前执行监视信息的标签。用户还可以通过从发现窗格中选择一项服务并单击构建按钮来开始一个构建(任务规定)。

[0327] 如上所述,窗格彼此协同,这样用户即可迅速地完成任务。

[0328] 当用户使其光标悬停在组合窗格或构建窗格中的一个服务名称上方时,将在弹出框中显示由选自位于 OWL-S 中的 SSD 的服务的描述。

[0329] 除组合窗格之外,所有其它的四个窗格中的任何一个都可以通过选择对应的菜单项目而隐藏或显示。

[0330] 如果需要的话,也可以向该用户界面添加保存窗格以供用户保存其认为有价值的组合。

[0331] 选项卡版本

[0332] 图 18-22 使用选项卡隐喻显示了另一种可能的用户界面。这更适于具有较小显示器的设备,例如 PDA 和移动电话。

[0333] 类似于完整版本,在该用户界面中也有五个窗格。每一个窗格都是大型表格的选项卡。图 18 显示了发现窗格。发现窗格中的一个列表显示了所有的服务。用户可以通过选择工具栏中的下拉列表以六种不同的次序对该列表进行排序。若要在详细窗格中看到关于服务的更多内容,用户可以选择该服务和“Detail(详细信息)”按钮。用户可以由特定服务构建一个新任务。其方式是双击该服务,构建窗格将弹出,并且所选服务位于该构建窗格中。更特别地,图 18 显示了一个用户界面选项卡版本 (I) 1800。

[0334] 图 19 显示了一个用户界面选项卡版本 (II) 1900。图 19 显示了组合窗格。在组合窗格中有几个组,它们包括两个下拉列表和两个按钮。左侧下拉列表中的服务被称为消费者,右侧下列列表中的服务被称为生产者。在每个组中,任意消费者都可以与任意生产者结合。这就允许该 UI 可以在紧凑的显示区域呈现更多组合。当由左向右看消费者 / 生产者对时,其次序与其实际执行次序相反。这是有意而为的,这样该消费者 / 生产者对就可以读做一个英语句子(例如,“Open” “URL from Favorites”),因为可以很容易地将生产者视为名词短语而将消费者视为对生产者起作用的动词或形容词短语。这就允许用户更加迅速地理解可用的任务的种类。对于其它的动词及其宾语名词次序相反的语言(例如日语),应该以其执行次序显示它们。两个按钮是“Execute(执行)”和“Construct(构建)”。当用户单击“Execute(执行)”时,当前选择的消费者和生产者将结组成一个任务并执行。当用户单击“Construct(构建)”时,当前的消费者和生产者将成为进一步构建的基础并且显示在构建窗格中。

[0335] 图 20 显示了用户界面选项卡版本 (III) 2000。图 20 显示了构建窗格。在该窗格中有三个组合框,它表明当前任务包括三种服务。用户可以通过单击窗格中的“-”按钮来删除第一个和最后一个服务,或者可以通过单击第三个组合框下面的“+”按钮向按钮添加一项新服务。在该示例中,因为第一个服务并不从其它服务中接收任何参数,因此在第一项服务上方并没有“+”按钮。每个组合框还列出了与任务中当前服务类似的服务。用户可以

使用列表中任意一个服务替换当前任务。

[0336] 图 21 显示了用户界面选项卡版本 (IV) 2100。图 21 显示了用户单击“+”按钮之后显示的构建窗格。现在已经向构建窗格添加了一个新组合框。该组合框列出了可以与第三项服务（先前为最后一项服务）结合的所有服务。因为新的组合框中当前所选服务并不具有任何输出，所以并未显示“+”按钮。

[0337] 当用户对任务感到满意时，可以单击“Execute（执行）”按钮来运行该任务。

[0338] 图 22 显示了用户界面选项卡版本 (V) 2200。图 22 显示了信息窗格中的执行监视器。每个执行监视器都是信息窗格中的一选项卡。执行监视器具有两部分。左部是一树状结构，显示了任务中服务的组织。在每个服务名称的前面有一复选框。复选框最初是未选中的。在执行期间，当服务完成时，该复选框将被选中。右部是一文本框，它给出各个服务的调用状态。用户可以通过单击选项卡名称附近的“X”按钮来关闭监视器。

[0339] 在该特定的实施例中，每个选项卡仅仅包含一个窗格，并且图 18-22 按照上面描述的步骤执行。这样一个选项卡就可以包含一个以上紧密关联的窗格。

[0340] 流版本

[0341] 图 22-23 显示了另选的用于任务计算的用户界面，它提供了一种直觉的方法，可以容易地创建相对复杂的服务。此外，这一更加紧凑的用户界面更适于具有小型显示器的设备，例如 PDA 和移动电话（图 27-33 显示了用于移动电话的类似功能的用户界面）。

[0342] 接下来将描述对图 23-26 中的用户界面和图 27-33 中的用户界面的用户体验。

[0343] 在该 UI 中有两个窗格。左窗格包含一个列表，它列出了可以与出现在右窗格中的服务（功能）结合的所有服务（功能）。用户可以双击一项服务（或者在选中它之后再次单击）以将它由左窗格移至右窗格。若要从右窗格删除一项服务，用户可以双击该服务，或者可以选择它（通过在服务上单击），然后单击右上角的删除按钮。向右窗格添加服务或从右窗格删除服务会自动更新左窗格上的服务列表。当一项服务由左窗格移至右窗格时，TCC 会自动在左窗格的服务中的正确位置排列该服务。如果所选服务有多个正确位置，则会请用户选择一个正确的位置来插入该服务。

[0344] 右窗格中的服务序列应该从头到尾读做一个英语句子。当该序列“完成”时，意味着它表示了一个由 TCC 确定的可以执行的组合，右上角的执行按钮将处于活动状态。有时，当服务添加到右窗格上时，在服务之间会出现一个“+”；这意味着要完成组合，必须在“+”所在位置添加一项或多项服务。

[0345] 只要可能，即左窗格中的剩余服务或服务只能在与右窗格上的服务结合的一个组合中使用，系统将自动在右窗格上完成该组合。

[0346] 当用户创建一个组合时，她可能会使用左箭头和右箭头（位于顶部）来在组合创建过程中前后移动。用户可以通过按下左上角的重新启动按钮来重新开始。

[0347] UI 窗口的底部显示关于左窗格当前所选服务的信息，或者，如果未选择服务，则显示组合的当前状态信息，例如可能的组合的数目（“可能性”）。可以通过按帮助按钮的方式来禁用 UI 窗口的底线；底线将消失，然后用户即可通过在服务上单击（而不是双击）来添加或删除服务。

[0348] 接下来将描述使用图 23-26 所示用户界面创建一个组合的示范屏幕快照，其方式是详细说明各个图。

[0349] 图 23 显示了第一个用户界面流版本 2300。图 23 显示了在用户界面的开始阶段的所有可用服务。类似的显示出现在按下重新启动按钮时。

[0350] 图 24 显示了第二个用户界面流版本 2400。图 24 显示了用户连续选择两项服务之后的用户界面。

[0351] 图 25 显示了第三个用户界面流版本 2400。图 25 显示了用户选择了足够多的服务以达到一个可执行的服务组成后的用户界面。

[0352] 图 26 显示了第四个用户界面流版本 2500。图 26 显示了另一个可执行服务组成，但是禁用了底端信息线。

[0353] 图 27-33 所示的用户界面以与上面几乎相同的方式进行操作，不同之处在于图标被按钮替换；前向箭头（右箭头）（及其功能）和帮助功能（具有位于用户界面底部的相关解释）被省略。

[0354] 接下来将描述使用图 27-33 所示用户界面创建一个组合的示范屏幕快照，其方式是详细说明各个图。

[0355] 图 27 显示了开始处的任务计算 2700。图 27 显示了开始时的用户界面。

[0356] 图 28 显示了在用户选择后的任务计算 2800。图 28 显示了服务选择后的用户界面。

[0357] 图 29 显示了在左窗格刷新后的任务计算 2900。图 29 显示了在图 28 的用户选择之后左窗格刷新后的用户界面。

[0358] 图 30 显示了用户选择后任务计算的另一个视图 3000。图 30 显示了在第二个服务选择后的用户界面。

[0359] 图 31 显示了在左窗格刷新后任务计算的另一个视图 3100。图 31 显示了在图 30 的用户选择后左窗格刷新后的用户界面。

[0360] 图 32 显示了任务计算自动完成信息 3200。图 32 显示了在启用自动完成功能的图 31 的刷新后的用户界面。

[0361] 图 33 显示了在任务计算中用户使用自动完成选项选择自动结束 3300。图 33 显示了在用户调用自动完成功能之后的用户界面。右侧的服务是不可执行的。

[0362] 基于 Web 的版本

[0363] 图 34-38 显示了基于 Web 的任务计算用户界面。它适于与任务计算客户机 1602 一起使用，而不需要在用户的设备上安装特殊软件以供用户进行任务计算，不过用户界面运行在与用户用于任务计算的设备不同的设备上时除外。虽然所讨论的任何一种用户界面都可以在用户的设备上运行，而 TCC 的其余部分在一个或多个其他设备上执行，但是这一特殊用户界面并未阻止在用户的设备上安装软件。要注意的是，该用户界面没有排除在用户的设备上安装软件；用户可能会这样做，从而向其任务计算体验添加功能，即对运行在其设备上的其它应用程序的访问。

[0364] 执行（运行）任务计算客户机 1602 的设备（例如计算机）也承载 Web 服务器。用户可以使用 PDA 或移动电话来告诉 Web 服务器发现和过滤服务并构建和执行任务。用户界面是以 HTML 或者任何其它网页设计语言设计的，这样就可以使用任何网络客户机应用程序（例如 INTERNETEXPLORER、NETSCAPE NAVIGATOR、OPERA BROWSER、MOSAIC，等）进行浏览。

[0365] 图 34 显示了用户界面的 Web 版本 (I) 3400。图 34 显示了发现页面。该页面包括组合框、列表和几个按钮。该列表显示了所有的服务。用户可以通过从组合框中选择一种次序并单击“Sort(排序)”按钮来对列表进行排序。列表下方的按钮允许用户浏览其它页面。例如,若要浏览服务的详细信息,用户可以从列表中选择该服务并单击“Detail(详细信息)”按钮来访问详细资料页面。

[0366] 图 35 显示了用户界面的 Web 版本 (II) 3500。图 35 显示了组合页。在该页面的顶端给出了服务的数目。在此下方有两个按钮。用户可以单击“Update(更新)”按钮来下载组合页的最新版本或单击“Discovery(发现)”按钮来访问发现页。紧接着有几个由两个下拉列表和两个按钮组成的组。顶部下列列表中的服务被称为消费者,底部下列列表中的服务被称为生产者。在每个组中,任意消费者都可以与任意生产者结合。这就允许该 UI 可以在紧凑的显示区域呈现更多组合。当由上向下看消费者/生产者对时,其次序与其实际执行次序相反。这是有意而为的,这样该消费者/生产者对就可以读做一个英语句子(例如,“Weather info of” “Contact from Outlook”),因为可以很容易地将生产者视为名词短语而将消费者视为对生产者起作用的动词或形容词短语。这就允许用户更加迅速地理解可用的任务的种类。对于其它的动词及其宾语名词次序相反的语言(例如日语),应该以其执行次序显示它们。两个按钮是“Execute(执行)”和“Construct(构建)”。当用户单击“Execute(执行)”时,当前选择的消费者和生产者将结组成一个任务并执行。当用户单击“Construct(构建)”时,当前的消费者和生产者将成为进一步构建的基础并且显示在构建窗格中。

[0367] 图 36 显示了用户界面的 Web 版本 (III) 3600。图 36 显示了构建页面。在该页面中有两个组合框,这意味着当前任务包括两项服务。用户可以通过单击位于列表上方/下方的添加/删除按钮,来向列表的顶部/底部添加一项新服务或者删除列表中的第一/最后一项服务。

[0368] 图 37 显示了用户界面的 Web 版本 (III) 3700。图 37 显示了在用户单击了列表下方的“Add(添加)”按钮之后的构建页面。然后向该构建页面添加了一个新的组合框。用户可以通过单击“Discovery(发现)”和“Composition(组成)”按钮来停止构建并移至其它页面,或者通过单击“Execute(执行)”按钮来执行任务,或是通过单击“Save(保存)”按钮来将当前任务保存为一项新服务。

[0369] 图 38 显示了用户界面的 Web 版本 (V) 3800。图 38 显示了执行监视页面。在用户发布一条命令来执行任务之后将显示该页面。在该页面的中部列出任务中的服务。在每项服务之前,有一个箭头,用于指示该服务为当前正在的服务。用户可以通过单击“Composition(组成)”来停止监视,或通过单击“Discovery(发现)”按钮来移动至其它页面,或者通过单击链接“View Execution Detail(查看执行细节)”查看与执行相关的更多信息。在服务调用结束后,该箭头移动至下一项服务,并且在该服务之前出现一个红色的“Done(已完成)”的消息。在任务执行过程中,监视器页将以预定的频率刷新自身以反映在任务结束前或由异常中止之前的最新状态。

[0370] 图 33-38 按照上面所述结合执行。

[0371] 基于菜单的版本

[0372] 用户使用嵌套的菜单来组成/构建/执行任务。它可以是 TCC、OS 和应用程序的

一部分。嵌套的菜单是由当时对任务计算环境可用的服务动态创建的。这一嵌套的菜单尤其适用于 OS 和应用程序中对象的环境菜单。嵌套的菜单是为特定对象可以映射到其中的语义对象而构建的。

[0373] 图 39 显示了用户界面的嵌套菜单版本 (I) 3900。图 39 显示了不从对象构建组合的方法。菜单可以从 OS 中的某些图标开始,例如从 WindowsXP 中任务栏上的图标开始。菜单启动时没有输入服务 3910,并且它显示了服务的另一个菜单 3912,当用户的光标在服务上悬停特定时间时(例如),它可以与第一个服务 3910 组合。如果用户决定继续使用服务的当前组合,用户可以(例如)右键单击菜单项目(这是最后一个服务选择)来执行组合或构建进一步由当前选项启动的组合 3912 或显示所选服务的属性。图 39 显示了菜单 3910 中“Ryu’s Contact”、菜单 3912 中“BusinessAddress of”、菜单 3914 中“Aerial Photo of”和菜单 3916 中“Vies onprojector”的示例组合。

[0374] 如果组合以某些输出而结束,则系统可以设置为:(1) 系统不在右键单击菜单中显示“Execute(执行)”项目,或(2) 系统执行组合并省略输出,或(3) 系统在给定组合的结束时试着自动完成组成服务并执行它。

[0375] 系统也可以被设置成当(例如)用户的光标在服务上方悬停特定时间后执行/构建/属性菜单项目 3918 出现。

[0376] 在上面的描述中,我们讨论了从无输入服务开始扩展组合的方法。我们也可以由例如无输出服务开始并在其它方向对组合进行扩展。或者,我们可以启动任何服务(可能有或没有输入/输出),并让用户在两个方向扩展该组合。

[0377] 图 40 显示了用户界面的嵌套菜单版本 (II) 4000。图 40 显示了实质上从 OS 或应用程序中某些对象启动的同一个嵌套的菜单界面的示例。当用户右键单击对象 3908 时,将确定对象的相应的语义对象并显示使用该语义对象的服务。其它的与“嵌套菜单(从无对象)”中相同。当组合被调用时,会分析该对象以创建语义对象,(以某种内部一致性实现)创建提供服务并且执行服务的语义对象,并且将该语义对象传递给服务组成的其它部分。

[0378] 在 OS 广泛可用的情况下(Windows、Unix OS,等等),文件扩展名可用来判断对象所映射到的语义对象。例如,.jpg/“Image File”,.mpg/“Video File”,.vcf(vCard)/“Contact”,.ics(iCalendar)/“Schedule”,等。在文件自身包含语义对象时,会将文件中包含的一或多个语义对象用于服务的组成。例如,OWL(Web 本体语言)的.owl 文件可以包含语义对象。

[0379] 我们可以将该方法扩展至描述语义服务的 OWL-S 文件。当用户右键单击一个 OWL-S 文件时,就会显示可组成该 OWL-S 文件所描述服务的服务。

[0380] 在使用应用程序的情况下,可以从个人信息管理(PIM)软件中启动组合,例如,联系人或计划项目。或者可以由电子邮件软件中电子邮件中的附件启动组合。Web 浏览器软件可以动态地为其在页面或与页面链接的页面中找到的语义对象或非语义对象创建图标或菜单,并且允许用户由此开始组合。

[0381] 命令行版本

[0382] 图 41 显示了命令行版本的界面 4100。用户通过键盘或类似的输入设备与 TCC 进行交互。TCC 将按照命令(例如,通过按诸如 TAB 键之类的特殊键)使用已经键入的字符,显示对应于所发现服务名称的完成的短语。如果有一个以上用于完成的候选短语,界面将向

用户呈现这些候选项并等待用户的进一步输入。当用户已经输入了一个完整的服务名时，自动完成就基于服务的语义和 TCC 的组合逻辑。换句话说，在给定了输入的在先服务时，自动完成服务是一项可以很安全地组合的服务。例如，用户已经在“View on Projector”中键入并且单击 TAB 键来完成操作时的情形。随后，系统仅仅将与“View on Projector”匹配的服务（即生成“File（文件）”或其子类对象的服务）作为潜在的完成显示。当用户单击返回时，它就执行该组合（或者如果组合并未完成，它将显示潜在的完成）。这就模拟了用户通常比较熟悉的完成句子的过程，这样它就向用户提供了快捷并且简单的方法来组合并执行服务。

[0383] 并非总是要按照消费者到生产者的次序。用户可以将光标移动到一些服务前面并单击，例如，使用 SHIFT-TAB 来完成向左侧的移动。可以为实现效率而使光标仅仅跳到服务边界。

[0384] 任务计算客户机的详细描述 (TCC)

[0385] 接下来更加详细地描述两个任务计算客户机。我们将它们分别称为 TCC I 和 TCC II。

[0386] 这两个任务计算客户机对 TCE 做了关于语义描述服务 (SDS) 所使用技术的非必要的假设。其它技术可以可交换地使用。这些假设为：

[0387] 用于服务调用界面的 SOAP/WSDL 和 UPnP

[0388] 用于语义服务描述的 DAML-S 或 OWL-S

[0389] 用于与 TCC 在相同的子网中服务的发现的 UPnP。监听预先指定的套接字以发现与 TCC 在相同设备上运行的服务。

[0390] 该 TCE 中可用的服务使用并生成图 42 中所示本体中进行了语义描述的数据。

[0391] 图 42 显示了语义实例类型之间的关系 4200。一般而言，本体是某个领域内知识的机器可读表现形式；本体认为含义归于它所描述的领域内的实体，它位于其它东西附近或其中，描述实体的属性和该领域内实体之间的关系。类是具有相同属性和与其它实体的相同关系的实体的集合，实例是特定数目的语义类。此种本体的表示语言为 OWL-S。

[0392] 在图 42 中，“事物”是 RDF 名称空间中定义的类，其是所有语义实例类的根源。“联系人”、“时间表”、“地址”和“文件”是从“事物”直接继承而来的四个类。“FLA 位置”“地址”的子类，表明了 FLA 办公室中的地址。“媒体文件”有三个子类：“音频文件”、“视频文件”和“图像文件”，它是“文件”的子类。“网页”也是“文件”的另一个子类。最后，“地址”是“联系人”的属性。

[0393] 任务计算客户机 I

[0394] 图 43 显示了 TCE 实施例的体系架构 4300。即，图 43 显示了我们所描述的 TCE（在所述假设下）中的 TCC I 4400。在图 43 中，“用户”4302 是使用 TCE 的客户机；“本地服务”4304 是在与“TCC I”4400 相同的设备上运行的服务；“本地服务管理器”4306 控制本地服务 4304 的管理模块；“遍布服务”4308 是在与“TCC I”4400 相同的网络内运行的服务；“UPnP 设备 +Web 服务器”4310 是发布遍布服务的模块；“E-服务”4312 是另一种服务，并且也使用“UPnP 设备 +Web 服务器”来发布其自身。

[0395] 当在 TCE 框架中添加或删除本地服务 4304 时，向本地服务管理器 4306 发送一个登记或注销消息。本地服务管理器 4306 将其传递给“TCC I”4400。对于登记消息，本地服

务管理器 4306 也告诉“TCC I”4400 到哪里去寻找 SSD 1604。对于遍布服务 4308 和 E-服务 4312,UPnP 设备模块 4310 负责向网络环境发送添加或删除消息,该消息被“TCC I”4400 发现。对于添加消息,提供 UPnP 调用。当“TCC I”4400 发现该消息时,其将使用调用来寻找该服务的 SSD 1604。通常,SSD 1604 是 Web 服务器模块。下一个“TCCI”4400 连接到 Web 服务器并检索 SSD 1604。

[0396] “TCC I”4400 从 SSD 1604 中找出调用服务所用的必需信息。为调用它们,TCC I 1604 直接向本地服务提交内部 web 服务调用,向遍布服务提交 UPnP 或 WSDL/SOAP 服务调用,并向 e-服务提交 WSDL/SOAP 服务调用。

[0397] TCC I 4400 组件

[0398] TCC I 4400 主要包括五个部分:具有用户界面的主逻辑、发现和过滤引擎、组成引擎、执行监视器和执行引擎。

[0399] 图 44 显示了 TCC I 4400 的内部机制。下面将描述每个 TCC I 组件如何向用户提供任务计算环境而工作。

[0400] 具有用户界面的主逻辑 4402

[0401] “主逻辑”4402 是 TCC I 4400 的核心部分,原因有二。

[0402] 首先,其提供在任务计算环境中管理与用户的交互的(图形)用户界面((G)UI)模块,并支持任务计算的工作流。前述任何一个用户界面都可以用作 TCC 与用户交互的媒介。

[0403] 其次,主逻辑为其它模块传递消息,并将其整合到完整的系统中。其接收发现的结果并向登记和注销服务使用组成引擎的界面;其查询组成引擎,获取必要的服务相关信息并将其显示给用户;其帮助用户设计任务;其把任务传递给执行监视器以执行该任务;而且其显示监视信息并给用户一个清楚的执行观察。

[0404] 发现和过滤引擎 4404

[0405] “发现和过滤引擎(DE)”发现服务并检索服务的 SSD。该 DE 包含 UPnP 控制点和本地端口监听器,前者监视网络并接收遍布服务所发布的添加/删除消息,后者监视预定义的套接字并接收来自本地服务的添加/删除消息。当发现新的服务时,该 DE 也检索 SSD。对于遍布服务,该 DE 向 SSD 的 URI 发出“getDescriptionURL”调用;对于本地服务,SSD 的 URI 包含在消息中,因此不需要更多动作。因而若需要则该 DE 把变化和 SSD 一起告诉主逻辑。

[0406] 经常使用的远程本体描述文件在 TCC I 安装期间被安装在本体缓存中。在组成引擎中,当需要远程本体文件时,将首先尝试取回远程版本的连接。若不能成功建立连接,则改为使用已缓存的文件;若检索到远程文件,则也检查相应缓存版本,以确保其与最新版本一致。若否,则用最新版本替换缓存的版本。

[0407] 推论引擎(组成引擎)4406

[0408] “推论引擎(IE)”4406 解析服务的 SSD 并响应其它模块的服务相关查询。其为其它模块向登记和注销服务提供标准界面。当登记了新服务时,解析该服务的 SSD 并将服务信息添加到内部知识库中;当注销一个服务时,从知识库中删除与该服务相关的信息。该知识基础用于回答各种查询,诸如“列出所有可用服务”或“列出所有服务内容”或提供所有成对的匹配服务等。

[0409] 可用 Prolog 或某些其它语言来表现确定为了组成（任务说明）的目的而如何彼此匹配服务的组成逻辑。推论引擎也执行优先权的区分。根据诸如新颖、匹配精确性和用户偏好等准则而区分匹配的优先权。

[0410] 执行引擎 4408

[0411] “执行引擎 (EE)” 4408 解析执行计划并设计执行它们的策略。该策略包括服务调用顺序、从一个服务向另一个服务的参数传递、参数转换等。

[0412] 然后基于该策略，其执行适当的调用，以根据服务所使用的调用界面而调用诸如 UPnP、WSDL/SOAP 或其它远程过程调用界面等服务。这需要非常动态的调用途径。即，这需要动态调用函数的名称和参数列表，而不调用预编译的 API。

[0413] 这也需要支持异步调用。为支持这种调用，可包括用于异步服务的类似 JavaSpace 的组件，其能够处理 UPnP 事件回叫、普通轮询、SMTP SOAP 和其它机制。

[0414] 执行监视器 4410

[0415] “执行监视器” 4410 是处于 EE 4408 和主逻辑 4402 之间的模块。其为每个执行计划启动一个监视器。若没有监视器，用户可能只注意到是否已经成功地执行了任务，而不会有更多东西。若有监视器，用户能够检查任务的状态：哪个任务调用已经完成以及哪个还没有完成。若任务失败，则监视器告诉用户在哪个点上发生了错误。

[0416] TCC I 流

[0417] 主逻辑 4402

[0418] 图 45 给出了 TCC I 主逻辑 4402 的流程图。当主逻辑 4402 开始时，其首先通过解析在预先定义了的目录中的 OWL-S 文件而发现现存的本地服务 4402-1。然后其开动本地服务套接字监听器 4402-2。该监听器监视预定义的套接字并将被告知何时在系统中添加或删除本地服务。由于遍布服务使用 UPnP 来扩展其存在性，主逻辑然后开动 UPnP 控制点以发现遍布服务。当本地服务套接字监听器或 UPnP 控制点接收到添加 / 删除消息 4402-4 时，主逻辑向 IE 4402-5 登记 / 注销服务。接下来，主逻辑寻找所有服务部分 4402-6，并随后启动 TCC I GUI 并等待用户的请求 4402-7。

[0419] 每个用户的请求 4402-8 都在主逻辑 4402 中处理。若用户想观看服务 4402-10 的细节，则从组成引擎中检索该信息并显示在 GUI 4402-16 中。若用户想设计新的任务，则评估当前的结构状态 4402-11，从组成引擎 4402-12 中寻找所有可能的组件。该组件显示在 GUI 4402-16 中，其帮助用户做出决定。若用户想执行已创立的任务或给定的组件，则产生新的执行监视器 4402-3，并把该执行计划发送给监视器。执行监视器把计划提交给了 EE 4402-14。主逻辑的剩余工作是显示监视信息 4402-15 并等待新的用户请求。

[0420] 发现

[0421] 图 46 显示了 TCC I 4400 中的服务发现和登记过程的流程图，其由发现引擎 4404 和组成引擎 4406 执行。当听到添加的消息时，TCC I 4400 从消息 4600-1 中提取发现信息。若该消息与本地服务有关，则该消息包含 SSD 的 URI；若该消息关于遍布服务，则 TCC I 4400 调用 UPnPgetDescriptionURL 方法并检索 SSD 的 URI。接下来，在组成引擎 4406 中检索并解析该服务的 SSD。解析结果位于 RDF 模型 4600-2 中。此后，该 RDF 模型被转换为知识基础格式 4600-3。该知识基础断言被组成引擎 4406 插入到知识基础 4600-4 中。

[0422] TCC I 4400 把 UPnP 作为用于与 TCC 相同的子网络中的服务的服务发现的主要方

法。

[0423] 发现模块实现 UPnP 控制点,以向网络发送查询,并搜索遍布服务。有时候服务在报告其存在方面(若需要)速度慢,或者在某些情况下,由于各种原因而忽视或漏掉了查询。因此,在提示方式下不可能找到服务。我们在服务发现模块中引入多 UPnP(或其它服务发现)控制点的观念。我们不启动一个,而是多个(为 TCC I 启动通常 5 到 10 个)控制点。每个控制点单独发送查询。控制点的发现结果在发现模块中汇总并过滤重复的消息。试验表明,通过这种方式,以稍高的资源消耗就可以显著改善遍布服务发现的性能。

[0424] 当某些服务在提供正确的语义描述上有问题时,发现模块将忽略这些问题。然而,由于 UPnP 的属性,这些设备将重复宣告它们的存在,并且 TCC I 中每次宣告之后检查过程将耗费许多资源。为处理这个问题,TCCI 产生黑名单,以记住“坏”服务及其时间标记,该时间标记表明了发现该服务的最后时间。随后,若再次发现黑名单中的服务,则 TCC I 把所存储的时间标记与当前时间相比较。若时间跨度在预定范围内,则 TCC I 完全忽略该发现消息;否则,TCC I 将再次检查描述。若再次检查时描述变正确了,则 TCC I 将从黑名单中删除该服务;或者,TCC I 将为该服务把时间标记更新为当前时间标记并更长时间地阻挡该服务。

[0425] 本地服务的发现工作如下。

[0426] 为了这个目的而定义预定套接字。TCC I 总是监听套接字。当添加(删除)了服务时,把该服务的 SSD 拷贝到(从)预定义的目录(如 C:\ProgramFiles\tce\psmgr)(删除)。然后向套接字发送一个消息。消息的格式为“a<SSD filename>”(“r<SSD filename>”)。当 TCC I 找到该消息时,其首先知道 SSD 位于预定目录+<SSD filename>中。TCC I 为添加消息登记 SSD 并为删除消息登记 SSD。

[0427] 显示已发现的服务

[0428] 为了使用户更容易地理解和更好地决定,可以分类和/或解释已发现的服务。可以以多种准则进行分类,其包括但不限于:

[0429] 输入和/或输出信号

[0430] 服务种类

[0431] 内部、本地、遍布、远程

[0432] 离用户的距离

[0433] 收费或免费

[0434] 用户可以预先设置这些分类准则。可以通过下述方式但不限于该方式向用户呈现分类的结果:

[0435] 字体(字样、大小、颜色等)、背景(颜色、图案)、结构(笔画的力量等)、声音(男/女、声调、语速)、分类清单

[0436] 基于分类的分组可以映射到诸如下拉式菜单等(G)UI 元素中。

[0437] 可以利用许多方式解释服务,所述方式包括但并不局限于:

[0438] 在光标在其上时,或者在对该服务发出任何动作时,弹出菜单

[0439] 在诸如状态栏的特定固定区域内显示它

[0440] 分割页面以显示服务细节

[0441] 服务解释可以包括但并不局限于:

[0442] 服务的详细文本描述

[0443] 服务类别

[0444] 对服务的输入及其本体（到其的链接）

[0445] 对服务的输出及其本体（到其的链接）

[0446] 文本或图形表示的服务模型

[0447] 服务的其他属性 / 参数 / 度量

[0448] 多媒体（图标、语音、图像、视频等）描述

[0449] 其 OWL-S 文件（到其的链接）

[0450] 除了文本描述之外，我们还可以使服务在其 SSD 内具有服务的图标描述、图像描述、视频描述以及语音描述。在其 SSD 内，该服务可以具有到这些描述的链接或描述本身（可以以文本方式进行编码）。这些描述支持多峰接口 (multimodal interface)，而且可以更好地对用户进行导航。例如，对于盲人，或者在用户驾驶轿车时，可以建立语音接口。在其他情况下，我们可以将其视频描述用作关于如何使用该服务的教学式可视使用说明书。

[0451] 图 47 示出具有用于发现窗格的基于 web 的用户界面的 TCC I 的示例 4700。

[0452] 图 48 示出具有用于服务的图标的发现窗格 4800，包括利用 SDS 内 (OWL-S) 的图标用于发现的完全版本用户界面。

[0453] 图 49 示出详细窗格 4900，它示出服务的尽可能多的信息。

[0454] 过滤

[0455] 根据服务的参数来过滤服务。语义服务描述中的参数，例如 OWL-S 文件中的服务参数，用于过滤。示例过滤包括：

[0456] 位置，特别是相对于当前用户位置的位置，例如“在同一个房间内”、“在同一楼层”或“在同一个建筑内”。

[0457] 距离用户的距离。

[0458] 对于打印服务

[0459] 打印机的速度、黑白或彩色、免费或付费、估计的执行时间

[0460] 任务计算环境为用户提供根据语义服务描述过滤服务的动态方式和用户界面。用于服务参数的语义已知，所以通过根据本体约束用户界面，用户界面可以支持用户输入。

[0461] 图 50 示出服务过滤的示例 5000。

[0462] 搜索 E 服务

[0463] 对于 e 服务，搜索这些服务是通过其开始使用特定服务的主要管道 (conduit)。在其初始命中或使用后，可以将其语义服务描述 (SSD) 保持到 SSD 高速缓存内。对我们来说，我们假定在因特网上存在一些让人们查询服务的、公知的 SSD 库。但是可以将 UDDI, DISCO 以及其他服务搜索机制引入任务计算环境。这并不局限于 web 服务，而且还包括其他服务调用机制。

[0464] 主要从组成与构建窗格调用该“搜索”。该服务的下拉菜单中的一个项目是“搜索 web 服务”。在选择它时，它弹出一个页面，让用户搜索具有固定输入和输出的 e 服务。在该页面上，如果用户希望，则用户可以改变、添加、删除或修改搜索条件。

[0465] 图 51 示出具有“搜索 web 服务”的发现窗格的示例。

[0466] 组成

[0467] 组成引擎 (IE) 4406 支持任务规定过程。组成引擎 4406 采用被发现、被过滤服务的 SSD 及其组成逻辑,以确定如何组成服务。还根据诸如新奇、匹配的准确性以及用户偏好的标准,按优先序排列匹配。其他模块可以查询关于服务匹配的组成引擎。

[0468] 图 52 示出关于组成引擎 4406 如何应答各种查询 5200-1 的过程 5200。第一步是将查询变换为知识库查询 5200-2。IE 4406 利用本体推理器 5200-4 与服务 5200-3 匹配。查询的应答是匹配服务对表。接着,将列表发送到“基于用户环境的排序结果”5200-5。该过程检验“用户环境 / 偏好”5200-6,并选择排序顺序。在对该列表进行排序后,返回“排序组成结果”5200-7。

[0469] 根据以下内容但并不局限于以下内容,按优先序排列组成。

[0470] 发现服务的新奇性

[0471] 匹配的准确性

[0472] 本体翻译的质量

[0473] 使用频率

[0474] 用户偏好

[0475] 根据从主逻辑模块反馈的用户确定,可以引入关于组成中用户偏好的学习机制。可以采用日语 IME (输入方法编辑器) 中使用的类似学习方法。

[0476] 可以根据作为组成集自身的特征,再过滤组成,例如:

[0477] 总成本

[0478] 要执行的总时间

[0479] 组成集中所有服务的近似

[0480] 在光标悬停在“执行”按钮上时,例如,通过弹出窗口,可以为用户提供这些特征的信息。

[0481] 图 53 示出组成窗格的示例 5300。

[0482] 在 TCC I 内,根据输入 / 输出签名,可以将组成分组。我们还可以根据 SSD 内的服务类别,例如“浏览器”和“打印”服务类别 (它们基于同样的输入 / 输出类型),对这些组成进行分组。

[0483] 可以对组成结果进行分类,用于后面的表示。与对单个服务进行分类相同,可以利用几种标准进行分类,这些标准包括,但并不局限于:

[0484] 服务类别

[0485] 内部、本地、遍布、远程

[0486] 距离用户的距离

[0487] 免费或收费

[0488] 与对单个服务进行分类的情况类似,用户可以事先设置这些分类标准。

[0489] 为了对用户控制感觉,如何表示事务非常重要,我们非常关心表示和解释。

[0490] 为了更容易理解任务,我们将组成窗格上的组成服务的顺序颠倒。例如,执行顺序是首先“FLA, URL”,然后,“通过投影仪观看”。但是在组成窗格上,表示组成的顺序是首先“通过投影仪观看”,然后“FLA, URL”,以与英语语句的顺序匹配。

[0491] 对于组成与构建窗格,可以将基于分类的服务分组映射到诸如下拉式菜单的 (G) UI 单元。

[0492] 如何以交互方式向用户呈现、他 / 她如何能够与服务交互是任务计算的一个重要部分。其他示例包括：

[0493] 在 (G)UI 内反映根据用户动作改变优先次序。例如,如果用户在组成与构建窗格中的下拉式菜单中选择服务,则先前的或之后的下拉式菜单根据优先次序改变其菜单项目的排序。

[0494] 我们有用表示用户为了确定执行哪个服务或组成的所需信息的其他无障碍方式。在从组成窗格上的下拉式菜单中选择服务时,该信息框弹出显示服务描述。

[0495] 从其 SSD 中取出该服务的人类可读描述。

[0496] 我们还可以对用户关于服务组成的线索。在鼠标光标悬停在“执行”按钮上时,出现弹出框以示出总成本、估计时间以及用于执行特定组成的其他标准。

[0497] 本体翻译

[0498] 通过填充不同本体中各有关示例之间的间隙,该系统可以显著增强系统的利用率。不同本体中的不同概念化过程可以对表示同一个信息提供许多方式。例如,地址的不同概念化过程可以导致以下结果(但并不局限于此):

[0499] (地址)

[0500] 示例实例是(“8400Baltimore Avenue, College Park, Maryland, 20740-2396, USA”)

[0501] (街道地址、城市、州、邮政编码、国家),该概念化过程的示例实例是(“8400 Baltimore Avenue”, “College Park”, “Maryland”, “20740-2396”, “USA”)。

[0502] (街道地址、城市、州),该概念化过程的示例实例是(“8400Baltimore Avenue”, “College Park”, “Maryland”)。

[0503] (街道地址、城市),该概念化过程假定它所处理的每个地址均在 Maryland。示例实例是(“8400Baltimore Avenue”, “College Park”)。

[0504] (街道名称、街道编号、城市、州),该概念化过程的示例实例是(“Baltimore Avenue”, “8400”, “College Park”, “Maryland”)。

[0505] 为了填充该间隙,我们需要(但并不局限于此):

[0506] 省略某些信息部分

[0507] 增补附加信息

[0508] 解析文本部分

[0509] 重组

[0510] 可以利用 SSD 在内部,或者在外部(不包括搜索用于本体翻译的 web 服务)提供关于如何可以进行这些本体翻译的信息。可以利用以下内容(但并不局限于此)提供功能性。

[0511] XSLT

[0512] 脚本语言中的代码

[0513] 对本地可执行件的引用

[0514] 关于是否可以在两个对象之间进行本体翻译的知识也对服务组成产生影响。此外,关于本体翻译的某些其他标准也可能影响优先次序。例如,包括较准确本体翻译的组成可以比包括较不准确本体翻译的组成优先。

[0515] 构建

[0516] 如果用户选择构建,则通过附加、删除、改变服务等,用户可以修改服务组成。根据其 SSD,对可以附加的服务进行过滤和按优先序排列。如果执行“点击和选择”操作,则用户可以容易地修改该构建。

[0517] 服务不仅可以附加到组成的开头和尾部,而且可以附加到各服务之间。一个示例是,始终具有同样输入和输出的“实例观看器服务”(请参考“SDS 实施例”)。

[0518] 图 54 输出构建窗格的示例 5400。

[0519] 执行

[0520] 任务执行过程包括语义层和句法层。在语义层定义任务,而通过句法层执行任务。

[0521] 图 55 示出语义层和句法层中的任务执行过程 5500。图 55 利用示例示范如何执行任务。在该图中,在语义层定义任务,它是 3 个服务的序列。“Contact from Outlook(来自 Outlook 的联系人)”5502 是从用户的个人 outlook 文件夹中检索一个联系人项目的服务;“Businessaddress of(商务地址)”5504 是从联系人项目中提取商务地址的服务;以及“Route from FLACP(从 FLACP 开始的路线)”5506 是显示从 FujitsuLaboratories of America,College Park 到给定地址的路线的服务。采用 OWL-S 来编写 3 个服务的语义描述。每个语义描述分别包括被称为“基础连接”的部分,它描述关于如何将语义类型映射到句法类型/从句法类型映射到语义类型的细节。在该图中,“Contact(联系人)”和“Address(地址)”是两个语义类型,而“串”是句法类型。“Contact from Outlook”是实现“Contact from Outlook”的 web 服务,而“Route from FLACP”是实现“Route from FLACP”的 UPnP 服务。

[0522] 以下说明执行过程。首先,TCC I 4400 分析“Contact from Outlook”5502 的 SSD 1604,而且知道该服务未进行输入,并返回联系人项目作为输出。然后,TCC I 4400 将 web 服务调用发送到没有输入的“Contact from Outlook”5508。执行 web 服务后,返回结果。请注意,结果是句法字符串。然后,“Contact from Outlook”5502 的 SSD 的基础连接部分告知 TCC I 4400 如何将该字符串映射到语义“联系人”对象。TCC I 4400 根据该指令,产生“Contact”对象,并将它发送到第二服务,即“商务地址”5504。“商务地址”5504 服务指出如何从联系人项目中提取商务地址。TCC I 4400 完成操作并根据输入产生“地址”对象。然后,将该“地址”对象发送到“Route from FLACP”5506。同样,在建立基础连接之后,“地址”对象映射到句法字符串。然后,TCC I 调用 UPnP 调用,并将该字符串用作输入。最后,“从 FLACP UPnP 开始的路线”5510 完成该请求。

[0523] 服务调用

[0524] 为了执行任务中的服务,利用其 SSD 1604 内的基础连接信息,主逻辑模块 4402 确定调用哪个界面。(对于一个服务可以存在同样类型和不同类型的多个界面。服务可以具有例如一个 UPnP 界面和两个 WSDL/SOAP 界面)。调用哪个基础连接取决于以下内容,但是并不局限于此:

[0525] 系统能力。例如,PDA 可以仅具有 UPnP 服务调用模块并选择使用 UPnP 基础连接。

[0526] 系统或用户偏好。

[0527] 在执行序列组成服务期间,可能发生许多事件。事件包括:

[0528] 在执行之前,UPnP 服务不可用

[0529] 失败的服务调用,因为

[0530] 执行过程中位置发生变化

[0531] 丧失网络连通性

[0532] 用户可以执行服务组成而无需任何干预,或者如果用户希望,用户可以监视组成的执行过程。通过利用监视器窗格与用户充分交互,我们可以使这些事件不被用户讨厌。这种交互包括:

[0533] 如果任何一个要执行的服务变成不可用,则通知用户

[0534] 在需要时,对用户提供的重构该组成的简单方式

[0535] 在确定使用哪个基础连接后,系统需要不将主逻辑模块 4402 中使用的语义实例编组为用于服务调用的参数,而将调用结果编组为语义实例。

[0536] 这是编组 / 解组模块利用从要调用服务的 SSD 中提取的基础信息实现的。

[0537] 编组和解组

[0538] 基础连接信息可以位于(但并不局限于此):

[0539] XSLT

[0540] 用脚本描述语言编写的代码

[0541] 对用于编组 / 解组的 web 服务的引用

[0542] 对本地可执行件的引用

[0543] 例如,具有(“街道地址”,“8400Baltimore Avenue”)、(“城市”,“College Park”)、(“州”,“Maryland”)、(“邮政编码”,“20740-2396”)以及(“国家”,“USA”)属性 / 数值对的“地址”语义实例必须解组为用于将服务映射到美国地址的、诸如(“8400Baltimore Avenue”,“CollegePark”,“Maryland”)的参数集。或者,必须不将同一个“地址”语义实例编组为用于另一个映射服务的、诸如(“8400 Baltimore Avenue,College Park,Maryland, 20740-2396, USA”)的另一个单参数集。

[0544] 通常,困难是不将服务调用结果编组为语义对象。这些情况可以包括不将信息的文本表示编组为更结构化的语义实例。例如,地址提供服务可以提供“8400 Baltimore Avenue, College Park, Maryland, 20740-2396, USA”作为其调用结果,而且它需要不利用(“街道地址”,“8400 Baltimore Avenue”)、(“城市”,“College Park”)、(“州”,“Maryland”)、(“邮政编码”,“20740-2396”)以及(“国家”,“USA”)属性 / 数值对,编组回“地址”语义实例。在这种情况下,我们需要某种功能或服务解析该文本表示。可以利用 SSD 内的采用脚本语言编写的代码,利用 web 服务,或者利用本地可执行件,提供这种功能或服务。可以利用服务调用结果(不包括地址的文本表示)进行调用,以获得语义实例(不包括具有属性 / 数值对的“地址”语义实例)。

[0545] 利用在上述步骤获得的参数,执行模块调用服务。将该结果送到编组 / 解组模块以编组为要在主逻辑模块消费的语义实例。

[0546] 异步服务调用

[0547] 通常,WSDL+SOAP web 服务调用采用同步模型,即,在将调用请求发送到 web 服务时,作为响应消息发回服务结果。这种模型存在两个缺点。第一,如果执行该服务花费非常长的时间,则 TCC 可能遇到超时消息。否则,TCC 必须等待响应,而且不能做任何其他事情。第二,在服务处理周期内,如果用户需要更多的信息,则在当前的模型中,非常难以使 web

服务获得该信息。

[0548] 在 TCE 实施例中,通过引入异步 web 调用,解决第一个问题。以下说明异步 web 服务调用的工作过程。在 TCC 首先提交 web 服务请求时,web 服务产生唯一 id,并将它作为引用送回 TCC。此后,TCC 将新请求发送到具有 id 的 web 服务,并检验是否准备好对先前请求进行应答。存在 3 个可能响应。如果处理过程还未完成,则什么也不返回。否则,如果处理过程已经完成,则返回应答。或者,如果处理过程出现错误,将返回出错消息。TCC 可以以给定间隔轮询 web 服务,并对两个连续轮询操作期间的事务进行处理。

[0549] 以下说明第二个缺点的解决方案。在服务的 SSD 中,我们定义一个被称为服务控制 UI 的特殊字段。同时,除了 web 服务外,设计另一个 web 服务器,该 web 服务器允许用户输入特殊信息,然后,该 web 服务器将它发回 web 服务。

[0550] 图 56 示出关于它如何工作的示例。在该图中,“web 服务”5602 是提供服务 312 的模块。“服务控制 web 页面”5604 是帮助用户输入特殊信息的 web 服务器。该过程包括几个操作。首先,TCC 1602 将请求发送到“web 服务”5602。除了正常参数,TCC 1602 还产生特殊 TCEExecID,在该例中为 1234-1,并将它与请求附加在一起。与异步模型相同,“web 服务”5602 产生唯一 id,而且为了轮询引用,将该 id 返回 TCC 1602。在该例中,该 id 是 1。接着,“web 服务”5602 做一些特殊事情。在“web 服务”5602 内,存在用于存储 TCEExecID 与 id 之间的映射关系的“内部状态表”5606。该表 5606 有 3 列:TCEExecID、id 以及特殊信息。对于这种请求,在“内部状态表”5606 中插入新行。在上述示例中,这两个字段具有数值 1234-1 和 1。在 TCC 1602 端,在从“web 服务”5602 返回 id 时,它启动关于“服务控制 web 页面”5604、具有附加 TCEExecID 的 web 页面。利用“服务控制 web 页面”5604,用户成功提交特殊信息。接着,“服务控制 web 页面”5604 将该特殊信息发送到“web 服务”5602,并将 TCEExecID 用作引用。通过检验“内部状态表”5606,“web 服务”5602 容易发现特殊信息属于哪个请求,并完成该请求。对于 TCC 1602,在提交该信息后,TCC 1602 对“web 服务”5602 开始进行异步轮询过程并等待应答。

[0551] 我们具有同步和异步 web 服务调用,而且我们使任选的特殊 web 服务器支持某些 web 服务。TCC 1602 如何知道对于给定服务应该使用哪种调用方法。

[0552] 一种解决方案是,对该服务的 SSD 内的一切进行定义。在该实施例中,我们采用另一种方法。

[0553] 图 57 示出 TCC I 4400 的服务调用流程图 5700。图 57 示出 TCC 实施例的完整流程图。假定 SSD 指出利用 web 服务实现语义服务,并将该方法称为“GetURL”。首先,TCC 检验是否存在被称为“start_GetURL_with_ID”的方法。如果存在,则意味着,web 服务是具有 ServiceControlUI 的异步服务 5702。TCC 调用所述调用和唯一 id,在该例中为 1234-1,并进入轮询过程。如果不存在“start_GetURL_with_ID”,则 TCC 检验是否存在“start_GetURL”。如果存在,则 web 服务是异步服务。TCC 调用所述调用并进入轮询过程。如果“start_GetURL_with_ID”和“start_GetURL”均不存在,则 TCC 知道该 web 服务一定是同步服务 5704。接着,TCC 检验是否存在“GetURL_with_ID”。如果存在,则该方法是具有服务控制 UI 的同步服务。对于这种服务类型,TCC 利用唯一 id 调用所述调用,并利用该调用实现 TCC。请注意,对于同步调用,不要求进行轮询过程。最后,如果 3 个调用均不存在,则 TCC 认为这仅是一个正常同步 web 服务,并象往常一样进行处理。在上述示例中,可以利用

具有参数或不具有参数的任何函数代替“GetURL”。

[0554] 确定服务是异步 5702 还是同步 5704, 以及确定服务是具有用户界面还是不具有用户界面的方式有两种。以上描述了第一种情况, 在这种情况下, TCC 检查描述文件 (不必是语义文件。例如, 其 WSDL 文件), 而且, 不是利用将要调用的原始界面的名称, 而是利用可用界面的附加名称, 确定它。

[0555] 确定它的另一种方式是使用服务的 SSD。SSD 可以具有关于应该如何调用服务以及应该如何访问服务控制页面面的描述。

[0556] 在只有单页界面可用时, 执行监视器页面和服务控制 UI 之间的以下集成非常有用。例如, 在使用基于 web 的 UI 时, 客户机设备是容许非常有限的显示空间的、诸如 PDA 或移动电话的小型设备。如上所述, 为了调用具有服务控制 UI 的服务, 用户必须访问 web 服务器以输入特殊信息, 或者控制该服务。这意味着, 在基于 web 的 UI 中, 必须启动新浏览器。然而, 在 PDA 或移动电话中, 不支持多个浏览器。

[0557] 为了解决该问题, 我们修改基于 web 的 UI 内的执行监视器。图 58-60 示出一示例。此时, 我们执行具有两个服务“Jeff’s File (Jeff 的文件)”和“View on browser (通过浏览器观看)”的任务。“Jeff 的文件”是允许用户从表上选择文件的服务器, 并返回该文件。“通过浏览器观看”是在浏览器上显示文件的服务器。请注意用户的选择是第一服务的特殊输入。

[0558] 图 58 示出具有服务控制 UI (I) 的基于 web 的用户界面 (UI) 5800。图 58 显示初始执行页面。如上所述, 箭头指出当前正在运行第一服务。请注意, 链接在指向 web 服务器的服务名称之下。

[0559] 图 59 示出具有服务控制 (UI) (II) 的基于 web 的用户界面 (UI) 5900。图 59 显示用户跟踪该链接时的屏幕画面。然后, 要求她从表中选择文件。

[0560] 图 60 示出具有服务控制 (III) 的基于 web 的用户界面 (UI) 6000。图 60 显示在用户进行选择并回到执行页面时的屏幕画面。请注意, 两个服务现在均完成, 而且第二服务的服务控制 UI 的链接变得可用。用户可以跟踪第二页面的链接以观看浏览器上的文件。

[0561] 监视

[0562] 执行监视器 4410 是用于显示关于调用执行计划的信息的模块。在通过组成和构建确定执行计划后, 建立新监视器实例。在服务与监视器之间确定标准接口, 以使 TCC I 4400 不知道执行监视器的执行细节。服务执行线程将消息发送到每个监视器, 而每个监视器以用户可理解的方式显示它们。

[0563] 在执行组成服务序列期间可能发生许多事件。这些事件包括:

[0564] 在执行之前, UPnP 服务变得不可用

[0565] 失败的服务调用, 因为

[0566] 在执行期间改变位置

[0567] 丧失网络连通性

[0568] 用户可以执行服务组成, 而无需干预, 或者, 如果用户希望这样, 则用户可以监视组成的执行过程。尽管通过监视器窗格可以与用户充分交互, 但是我们可以使这些事件不被用户讨厌。这些交互包括:

[0569] 如果任何一个要执行的服务变成不可用, 则通知用户

[0570] 在需要时,对用户 提供重构该组成的简单方式

[0571] 图 61 示出监视器窗格 6100。

[0572] 保存

[0573] 如果用户希望这样做,可以将组成保存为宏。但是在遍布 (pervasive) 环境下,仅保存特定序列的服务并不始终有意义。与始终与服务连通的因特网环境不同,将在不同的地方发现不同组服务。例如,在一个房间内“通过投影仪观看”和在另一个房间内“通过投影仪观看”是独立服务。如果我们可以采用宏的方式在该序列中可互换地使用这些服务,这仍非常有用。

[0574] 因此,我们对宏序列的每个操作所做的指定并不始终完全是特定服务,而是该位置可以接受的服务条件。例如,不将如“在 123 房间通过投影仪观看”的服务指定为宏序列中的步骤,我们可以具有诸如对于该步骤“其分辨率高于 1024×768”的“通过投影仪观看”类别的服务”的条件。通过广义化,我们可以进入下一步。通过将该服务的特定服务表示为条件“服务与该特定服务 (或 SDS) 完全匹配”,我们可以说宏序列中的每个步骤均是该步骤的服务必须满足的条件。“服务与该特定服务 (或 SDS) 完全匹配”与“服务与该特定 SDS 完全匹配”之间的差别在于,前者在服务调用级 (例如,UPnP,WADL/SOAP) 解决该匹配问题,而后者解决语义级对应关系。在一种后者多种服务调用技术中,SDS 可以具有多个分组。

[0575] 这些宏还可以是任务计算工作流的发现过程中发现的服务。在特定环境下,检验服务的可用性,哪个满足宏中各步骤的条件。仅在启动宏的所有服务可用时,它表现得象被发现或者可用的。

[0576] 条件可以在从准确匹配 (即,指定特定服务) 变化至输入 / 输出签名到服务类别的范围内。可以利用该位置的服务的 SSD 动态产生用于指定该条件的 UI。

[0577] 图 62 示出保存窗格 6200。

[0578] TCC II,任务计算客户机 (TCC) II

[0579] 以上描述的 TCC I 是 TCC 的一个实施例。TCC II 是 TCC 的另一个实施例。在不同设备内运行用户界面和 TCC II 的其他 TCC 模块。这样就可以使诸如 PDA (个人数字助理)、移动电话或 web 浏览器启动的任何设备通过多种通信链接访问 TCE,而无需在该设备上安装 TCC。

[0580] TCC II 是对希望使用任务计算服务、但是又没有安装在其客户机设备上的 TCE 的用户设计的。TCC II 较低了使用 TCE 的要求,因为它仅需要 web 浏览器。

[0581] 作为 TCC 的实施例,TCC II 具有服务发现、服务组成、服务构建以及执行服务的全套模块。最重要的差别是,在用户设备上运行 TCC I,而在诸如 FNS 的单独设备上运行 TCC II。TCC II 具有内部 web 浏览器。用户连接到 web 浏览器以检索服务信息和消费服务。

[0582] 图 63 示出 TCC II 6300 的角色。一方面,就象 TCC I 4400 一样,“TCC II”6300 发现“任务计算服务”6304 并组织 SSD 1604,另一方面,“用户设备的 web 客户机”6302 与 TCC II 6300 对话,并共享任务计算环境 (TCE) 的所有好处。

[0583] SSD 实施例

[0584] 对于 SSD (语义服务描述),我们可以依赖类似于 DAML-S 和 OWL-S 的标准。对于 SSD 支持的多种自然语言,语义服务描述可以使用统一字符编码或特定自然语言编码以实现以一种语言进行描述。语义服务描述可以使用统一字符编码或可以处理多种自然语言编

码的某种编码,以在一个 SSD 内具有多种自然语言的描述。对于基于 XML 的语义服务描述,可以对希望多种自然语言中具有标记使用“编码”属性。利用相应自然语言,重复具有不同“编码”属性的标记。

[0585] 图 64 示出 SSD 支持的多种自然语言的示例 6400。

[0586] 对于 TCC, TCC 可以使用他或她自己的语言,通常是其设备的 OS 上的选择的自然语言。这样可以使访问同一个 TCE 的每个用户可以根据选择的自然语言对 TCE 具有不同“查看”。

[0587] SSD 高速缓存

[0588] 该 SSD 高速缓存保持 SSD 用于公知的服务、远程服务、或宏(先前保存)服务。用于公知服务的 SSD 是用于不支持提供其 SSD 的公知服务的 SSD。用于远程服务的 SSD 是用于用户一次性或经常使用的远程服务的 SSD。用于宏服务的 SSD 是用于在保存窗格保存的或者从其他用户或系统信息中心库获得的宏的 SSD。

[0589] SDS 实施例

[0590] 接着,我们将给出语义描述服务(SDS)实施例。服务是用户感兴趣功能性的抽象概念。这种功能性通常至少有 3 个不同来源:设备、应用程序以及通过 web 的 e 服务。这 3 个来源是松定义类别,因为这些类别的边界非常容易变化。一般的说,起源于设备的服务与该设备用于传送的核心功能性有关;例如,电话(设备)的主要功能性是进行电话调用(服务)。同样,起源于应用程序的功能性与在计算设备上执行的计算应用程序有关;例如,个人信息管理(PIM)应用程序的功能性包括存储和检索人们的联系人信息。最后,e 服务功能性与在某个远程服务器上执行并通过访问 web、在用户的局域网络的边界之外传送该功能性的 web 服务有关。这种大致分类仅是用于区别各服务的性质的一种粗略方式。通常情况是,接着描述的服务属于多种类别。例如,在用户在家期间使用与用户的家庭立体声音响设备有关的视频流服务时,它通常是设备服务,而在用户在工作期间使用它以将歌曲从他的计算机传送到他家的立体声音响设备供其同房间的人监听时,还可以将它看作 e 服务。

[0591] 还应该注意这些区别如何不固定;制造商提供智能消费设备,通常具有网络接入能力(例如,网络电器),软件公司开发试图提供消费设备型功能性的应用程序(MICROSOFT 开发的用于观看 DVD 或听 CD 的 MediaPlayer)。值得注意的是,在任务计算过程中,这种功能性的起源与用户无关,因为用户仅对为了确定任务而组合这些功能感兴趣。

[0592] 基于设备的服务

[0593] 通过显示器观看

[0594] 该服务将 web 页面或各种文档显示在诸如投影仪、等离子体显示器、LCD 显示器的设备上。它接收 URL 作为输入,并对用户提供控制窗格。在利用指向网页或文档的 URL 调用它时,显示设备显示该网页或文档。通过控制窗格,用户可以控制显示设备。通过其控制窗格,她可以上移和下移该网页、跟踪到新网页的链接、装载新 URL 等。

[0595] 通过多种设备观看

[0596] 该服务将网页或各种文档显示在诸如投影仪、等离子体显示器、LCD 显示器的多种设备上。各设备显示完全相同的内容。各设备可以是在地理位置上分散开的,只要它们通过具有服务运行位置的网络连接在一起即可。服务接收指向网页或文档的 URL 作为输入,并对用户提供控制窗格。在利用 URL 调用它时,多种设备显示网页或文档。同时,用户可以通过

过控制窗格控制该设备。通过控制窗格,她可以上移和下移该网页、跟踪到新网页的链接、装载新 URL 等。

[0597] 地图观看器

[0598] (A) 该服务利用标记的位置在其显示器上显示用户指定位置附近的地图。它接收位置或地址作为输入,并对用户提供控制窗格。(B) 该服务利用标记的位置在其显示器上显示固定位置或地址附近的地图。它不接收输入,而对用户提供控制窗格。

[0599] 地图路线选择器

[0600] 该服务示出其显示器上两个位置之间的路线。该服务接收一个或者两个位置或地址,或者位置列表或地址列表,并对用户提供控制窗格。该服务提供 (A) 服务的固定位置或地址与用户指定位置或地址之间的路线,(B) 两个用户指定位置或地址之间的路线,或 (C) 显示这种表中所有位置或地址之间的路线(任选将固定位置用作开始点或结束点)。有时,对于服务提供商,将开始位置或目标位置限定到服务的固定位置非常重要,因为它禁止无限制地将该服务用作一般绘制地图服务。

[0601] 楼层 / 校园 / 建筑 / 博物馆 / 商场 / 医院地图观看器

[0602] 除了该地图在其地图中具有的结构比公路地图具有的结构多之外,该服务与“地图观看器”基本相同。

[0603] 实例提供服务

[0604] 该服务提供语义实例。(A) 它接收输入,并产生指向该语义实例的 URL 作为输出(同一个机器上的 web 服务器通常又对该语义实例提供服务)。(B) 它不接收输入,而产生语义实例(例如,OWL/DAML 实例)作为输出。该服务可以用于提供与诸如在楼层上联系人人们或预定特殊房间的遍布环境有关的语义实例。该服务还可以用于提供同一种语义实例列表,例如,联系人或日程安排。

[0605] 打印

[0606] 该服务打印 URL 指定的文件。(A) 它接收 URL,而且没有输出。在利用 URL 调用它时,它采用固定格式打印出由给定 URL 指定文件的固定数量的副本。(B) 它接收 URL,而且没有输出。在利用 URL 调用它时,该显示器显示打印选项框,以使用户设置打印参数,然后打印出给定 URL 指定的文件。(C) 它接收 URL,并产生控制窗格作为输出。在利用 URL 调用它时,它返回控制窗格,以使用户设置打印参数,然后,打印出给定 URL 指定的文件。

[0607] 打印名片

[0608] 该服务利用给定联系人打印一张或者多张名片。(A) 它接收联系人,而没有输出。在利用联系人调用它时,它以固定格式印出该联系人的固定数量的名片。(B) 它接收联系人,而没有输出。在利用联系人调用它时,显示器显示选项框,以使用户对名片打印设置参数,然后印出相应名片。(C) 它接收联系人,并产生控制窗格作为输出。在利用联系人调用它时,它返回控制窗格,以使用户设置参数,然后印出相应联系人的名片。

[0609] 传真

[0610] 该服务将 URL 指定的文件传真到指定的电话号码。它接收一个或者多个 URL 和一个电话号码作为输入,而不产生输出,或者产生用户用于控制该服务的控制窗格作为输出。在利用一个或者多个 URL 和电话号码调用它时,它将印出的图像传真到指定的电话号码。在返回控制窗格时,用户可以控制如何发送该传真,或者对该传真的指定封面。

[0611] 播放视频服务

[0612] 该服务允许用户播放视频。它接收一个 URL 作为输入,它或者不产生输出,或者产生控制窗格作为输出,以使用户控制该服务。在调用它时,它播放 URL 指定的视频文件。在返回控制窗格时,用户可以控制如何在设备上播放。

[0613] 播放音频服务

[0614] 该服务允许用户播放音频。它接收一个 URL 作为输入,它或者不产生输出,或者产生控制窗格作为输出,以使用户控制该服务。在调用它时,它播放位于 URL 的音频文件。在返回控制窗格时,用户可以控制如何在设备上播放。

[0615] 数码相片帧

[0616] 数码照片帧接收图像文件作为输入,并返回控制窗格作为输出。它在其显示器上显示图像文件(通常以照片帧形式),并允许用户利用控制窗格控制取向、图像缩放比、打印图像等。

[0617] 位置确定

[0618] 该服务提供设备的当前位置(因此,提供使用该设备的用户的当前位置)。它接收特定服务参数,并产生位置。参数可以是 WLAN 卡的 MAC 地址,或者是接入点 MAC 地址和其信号强度对的表。在调用它时,它产生位置,例如到位置本体的线路。

[0619] 目录发布程序

[0620] 该服务发布一组对象:文件、联系人、日程安排、地址等。作为语义服务,它不要求输入,而返回对象(例如,文件、联系人、日程安排、地址等)作为相应输出。在调用时,它允许用户从对象组中选择一个对象,然后返回用户的选择。

[0621] 库(事务/服务)

[0622] 该服务允许用户根据对象或作为其输入的给定的 SSD 动态建立 SDS,以与其他 TCE 用户共享,或者供之后它自己使用。在调用该服务时,用户可以使用控制窗格设置将由库服务新设置的口令、有效期、调用数的限制、使用的发现机制以及 SDS 的其他特性。

[0623] 通常,库服务在 3 种不同方式下有帮助。

[0624] 首先,在用户建立涉及遍布环境的特定任务,然后希望在该环境下使用语义服务组成表示的任务时,用于服务的库非常有用。任务本身是服务。利用本地服务选择器(Local Service Selector)和本地拷贝服务(Copy Service Locally)以及服务库(Service Bank),可以容易地实现这种情况。通过使服务仅接收具有诸如收费/免费、估计执行时间以及服务提供商的特定特性的服务,可以设计服务库,。

[0625] 使用库服务的第二种方式是,通过利用选择的或缺省服务发现机制提供 SSD 作为输入,建立 SDS。

[0626] 最后,用户通过库服务可以保留文件、URL、联系人、日程安排、服务以及涉及该环境的其他语义实例。建立库以利用单独服务提供每个实例,或者提供一个或者多个列表中的同样类型的所有实例。在服务可以具有多个列表时,用户可以建立用户附加的实例的列表,或者用户对现有列表之一附加实例。利用这种类型的服务,例如,用户可以在会议室内保留表示文件,在该会议室内经常要使用该表示文件。或者,在一个会议结束并将下一个会议的日程安排插入日历程序时,在建立下一个会议之后,用户可以在该会议室内对进入该会议室的其他用户保留该日程安排。(在这种情况下,如果所使用的日程安排库被连接到

该会议室内的日程安排管理系统,将提高使用率,该日程安排管理系统具有可远程访问界面。)

[0627] 在输入是“文件”对象时,可以建立库服务,以便在本地拷贝该文件,并由运行库服务的同一个机器提供该文件。

[0628] 为了具有更高的使用率,库可以与某些相关后端系统,例如,文件库的文件系统和联系人 / 日程安排库的联系人 / 日程安排管理软件组合在一起。

[0629] 该服务还是 SDSCM。对于库服务的 SDSCM 方面,可以参考“SDSCM 实施例”中的库描述。

[0630] 图 65 示出库服务的示例 6500。

[0631] 电话拨号器服务

[0632] 它接收电话号码作为其输入,而什么也不返回。在被调用时,它将通过用户的电话设备(可以是模拟 / 数字 / VoIP) 拨号。

[0633] 无线 / 有线 TC 存储器

[0634] 最好以当前的小型 USB 存储器形式因数实现它,但是可以以任意形式实现它。它包括 (1) 网络接口 (2) [任选] 文件夹发布服务 (3) [任选] 在文件夹服务中保存 (可能是诸如图像或音频的特定文件类型) (4) [任选] 库服务。服务 (1) 至 (3) 可以具有诸如图像或音频的特定文件类型,或者它们可以对不同文件类型提供单独服务。

[0635] 网络接口可以是以太网 (有线) 网络接口、无线 LAN 网络接口、蓝牙网络接口或红外线网络接口。对于无线 LAN 网络接口,它可以用作接入点,或者它可以处于 ad-hoc 模式,以便其他设备可以连接到该网络上 (以信息基础设施模式或 ad-hoc 模式),从而与该设备相连。对于无线 LAN 网络接口,它可以尝试连接到它发现的网络,或者可以具有简单接口 (在设备本身上或者通过无线),以对假定该设备所连接的网络的 SSID 和 / 或 WEP 密钥进行配置。

[0636] 拷贝到可移动存储器上

[0637] 该服务允许用户将文件拷贝到诸如软盘、CD-R/CD-RW、DVD-R/-RW/-RAM/+RW、存储卡、USB 存储器的可移动存储器上 (通常是收费的)。它接收一个或者多个 URL 作为输入,而不产生输出,或者产生控制窗格作为输出,以使用户控制该服务。在调用它时,它将 URL 指定的文件拷贝到可移动存储器上。在返回控制窗格时,用户可以控制如何进行拷贝。(例如制备音频 CD、照片 CD、简单数据光盘)。

[0638] 电子邮件处理器 (Emailer) (遍布)

[0639] 该服务允许用户将电子邮件发送到指定的电子邮件地址。它接收电子邮件地址作为输入,而没有输出。在利用电子邮件地址调用它时,它弹出对话框,以允许用户输入 / 编辑消息,并将该消息发送到该电子邮件地址。

[0640] 以下是电子邮件处理器 (Emailer) (遍布) 的另外两个可能实施例: (A) 它接收电子邮件地址作为输入,并产生 URL 作为输出。在利用电子邮件地址调用它时,它将 URL 返回允许用户编辑并发送电子邮件的网页。 (B) 该服务具有它自己的显示器和诸如键盘和鼠标的用户输入设备。它接收电子邮件地址作为输入,而没有输出。在利用电子邮件地址调用它时,该显示器在其显示中显示允许用户编辑并发送电子邮件的窗口。

[0641] 基于应用的服务

[0642] 内部服务

[0643] 为了实现最高效率,对于 TCC I,在内部实现以下服务。

[0644] 列表选择器程序

[0645] 该服务允许用户从同一种实例的列表中选择项目。该服务还可以使用户根据一个或者多个属性对实例进行排序。

[0646] 属性选择器程序

[0647] 该服务用户从实例中选择一个属性(或特征)。还可以利用属性调用该服务,以从其中进行选择。在后者情况下,无需用户干预,就可以执行该服务。

[0648] 你的输入

[0649] 根据任意对象的本体,该服务允许用户输入该对象的属性值,并利用这些值产生实例。可以执行该服务,以便它通常与其他实例提供服务一起用作提供同一种实例的服务。

[0650] 实例观看器

[0651] 该服务允许用户检验并修改该实例。它还可以用作调试工具。由于它具有用于输入和输出的同一个签名,所以可以将该服务插在任意服务之间,或者将该服务附加到尾部。

[0652] 本地服务

[0653] 下面的服务是为了实现最高效率与 TCC I 运行在同一个机器上的应用程序。作为 Window OS 中的环境菜单、任务菜单中的项目的一切均可以构成任务计算环境中的本地服务。下面列出一些本地服务。

[0654] 本地实例选择器

[0655] 本地实例选择器是允许用户选择计算设备的一个或者多个本地实例,并使这些实例可以用作语义描述实例。本地实例选择器包括但并不局限于类属本地实例选择器、本地文件选择器、收藏 URL 选择器、联系人选择器、日程安排选择器、地址选择器以及本地服务选择器。

[0656] 类属本地实例选择器

[0657] 类属本地实例选择器允许用户选择计算设备本地的任意本地可用语义实例。通过将计算设备的所有其他可用本地实例选择器组合在一起,可以实现该服务。

[0658] 本地文件选择器

[0659] 该服务允许用户选择本地文件,并通过本地 web 服务器,全局显示该服务。它不接收输入,而产生 URL 作为输出。在调用它时,它弹出一个对话框,以允许用户选择本地文件、将该文件拷贝到 web 服务器上,将该 URL 返回该文件。

[0660] 收藏 URL 选择器

[0661] 该服务允许用户从收藏文件夹(或任意文件夹)中选择 URL,并提供 URL。它不接收输入,而产生 URL 作为输出。在调用它时,它弹出一个对话框,以允许用户从收藏文件夹(或任意文件夹)中选择 URL,并返回 URL。

[0662] 地址选择器

[0663] 该服务允许用户选择地址,并通过本地 web 服务器,全局显示该服务。它不接收输入,而产生 URL 作为输出。在调用它时,它弹出一个对话框,以允许用户从例如你的联系人管理系统中选择地址,将该信息作为语义实例拷贝到 web 文件夹,将该 URL 直接返回语义实例或实例。

[0664] 联系人选择器

[0665] 该服务允许用户选择联系人,通过本地 web 服务器,全局显示该服务。它不接收输入,而产生 URL 作为输出。在调用它时,它弹出一个对话框,以允许用户从例如你的联系人管理系统中选择联系人,将该信息作为语义实例拷贝到 web 文件夹,将该 URL 直接返回语义实例或该实例。

[0666] 日程安排选择器

[0667] 该服务允许用户选择安排的事件,并通过本地 web 服务器,全局显示它。它不接收输入,而产生 URL 作为输出。在调用它时,它弹出一个对话框,以允许用户从例如你的日程安排管理系统中选择日程安排,将该信息作为语义实例拷贝到 web 文件夹,将该 URL 直接返回语义实例或该实例。

[0668] 本地服务选择器

[0669] 该服务允许用户从高速缓存、本地数据库、文件系统或计算设备本地的任何位置中选择 SSD,并使 SSD 描述的服务可以用作语义实例。

[0670] 语义实例拷贝器

[0671] 语义实例拷贝器获得其他服务产生的语义实例,并在本地将它们拷贝到适于这种类型的语义实例的应用程序。在用户干预情况下,或者在没有用户干预情况下,它均可以执行其功能。在前者情况下,弹出的窗口或某个其他用户界面允许用户修改对象、确定如何和在何处拷贝该实例,或者取消拷贝过程。语义实例拷贝器包括但并不局限于:类属语义实例拷贝器、本地拷贝、对收藏附加 URL,插入联系人表、插入日程安排表以及在本地拷贝服务。

[0672] 类属语义实例拷贝器

[0673] 类属语义实例拷贝器允许用户将任意语义实例拷贝到正确的应用程序。通过将计算设备上的所有其他可用语义实例拷贝器组合在一起,可以实现该服务。

[0674] 本地拷贝

[0675] 该服务允许用户保存 URL 指定的文件。它接收 URL 作为输入,而没有输出。在调用它时,它弹出一个对话框,以允许用户选择地方保存 URL 指定的文件。

[0676] 对收藏添加 URL

[0677] 该服务允许用户将 URL 保存到收藏文件夹内。它接收 URL 作为输入,而没有输出。在调用它时,它弹出一个对话框,以允许用户利用用户指定的名称将 URL 保存到收藏文件夹内。

[0678] 插入联系人列表中

[0679] 该服务允许用户将联系人插入联系人管理软件(例如,MICROSOFTOUTLOOK)。它接收联系人作为输入,而没有输出。在利用联系人调用它时,它将该联系人插入联系人管理软件中。它可以选择弹出一个对话框,以允许用户检验并修改联系人信息。

[0680] 插入日程安排列表中

[0681] 该服务允许用户将日程安排插入日程安排管理软件中(例如,MICROSOFTOUTLOOK)。它接收日程安排作为输入,而没有输出。在利用日程安排调用它时,它将该日程安排插入日程安排管理软件中。它可以选择弹出一个对话框,以允许用户检验并修改日程安排信息。

[0682] 在本地拷贝服务

[0683] 该服务允许用户将作为语义实例的其他服务提供的 SSD 拷贝到高速缓存、数据库、文件系统或计算设备本地的任何位置。

[0684] 电子邮件处理器（本地）

[0685] 该服务允许用户将电子邮件发送到指定的电子邮件地址。它接收电子邮件地址作为输入，而没有输出。在利用电子邮件地址调用它时，它弹出一个对话框以允许用户编辑并发送电子邮件。

[0686] URL 观看器（本地）

[0687] 该服务允许用户浏览 URL 指定的网页。它接收 URL 作为输入，而没有输出。在利用 URL 调用它时，它弹出对该网页设置的浏览器，并允许用户在本机浏览它。当前，我们利用为该服务指定的 URL，调用 MICROSOFTINTERNET EXPLORER。

[0688] 位置确定（本地）

[0689] 该服务提供计算设备的当前位置（因此，提供使用它的用户的当前位置）。它不接收输入，而产生位置。在调用它时，它产生位置，该位置是根据小信标（mini-beacon）、GPS 等链接到位置本体的链接。（客户机侧位置确定技术）。在被调用时，根据用户输入的位置确定服务弹出一个对话框或地图，以允许用户指定位置。

[0690] 数据库发布程序（从表格、项目）

[0691] 通过在数据库中指定表格的一部分或整个表格，根据该表格的模式，数据库发布程序自动建立本体，并发布允许用户从数据库中该表格的一部分或整个表格中选择对象的语义服务，然后，输出选择的对象作为输出。

[0692] 或者，通过指定数据库中表格的项目，根据表格的模式，该数据库发布程序自动建立本体，并发布语义对象提供服务，在执行该语义对象提供服务时，该语义对象提供服务产生选择的对象作为其输出。

[0693] 基于 web 的服务（E- 服务）

[0694] 地点信息

[0695] 该服务提供关于给定位置的信息，例如位于房间内的设备、最靠近浴室，谁占据该房间。它接收该位置作为输入，并利用该信息产生到该网页的 URL 作为输出。

[0696] 宿主服务

[0697] 宿主服务是对 TCC II 特殊设计的服务。它们包括“上载文件”，其诸如“上载音频文件”、“指向输入的 URL”、“通过浏览器观看”的变体，其诸如“通过浏览器播放”、“上载文件”以及“指向输入的 URL”的变体允许用户从用户设备和 URL 上载文件，并分别返回它们作为输出。宿主服务使用服务控制 UI 实现它们的功能性。

[0698] “通过浏览器观看”接收“文件”，而什么也不返回。与 TCC II 相同，“通过浏览器观看”简单允许用户通过同一个浏览器观看文件。利用 UI，将“通过浏览器观看”实现为同步 web 服务。它重定向服务控制 UI 以访问输入的 URL。

[0699] 远程服务

[0700] 通常，通过进行语义服务搜索，将远程服务（在此，利用该远程服务，我们指利用 WSDL/SOAP 界面实现的 web 服务）引入任务计算环境。但是，在某些情况下，无需任何语义服务描述 (SSD)，就可以建议远程服务。在这些情况下，通过经过改变其 WSDL URL 的扩展名，或者搜索其 SSD 的公知网点，尝试下载 SSD，TCC I 试图获得其 SSD。

[0701] 公知服务

[0702] 对于公知的服务,我们可以在本地或某个已知网点保持其语义服务描述。在我们遇到这些服务(UPnP、WSDL/SOAP 等)时,我们可以使用这些 SSD,以将这些服务引入我们的任务计算环境中。

[0703] TCC I 可以在其高速缓存具有诸如 AMAZON.COM 或 GOOGLE 的公知远程服务的 SSD,并允许用户将这些服务用作其他 SDS。或者,如果 TCC I 遇到没有 SSD 的 WSDL/SOAP 服务,则通过动态建立 SSD,它仍可以使用该服务。

[0704] SDSCM 实施例

[0705] 在该章中,我们将给出语义描述服务控制机制(SDSCM)实施例。语义描述服务控制机制(SDSCM)是通过包括但并不局限于以下内容,允许某个实体(可以是 TCC 的用户或任意一个软件)管理该服务的机制:

[0706] 建立服务

[0707] 改变用于要使用的服务的发现机制

[0708] 保持服务

[0709] 删除服务

[0710] 修改服务参数包括:

[0711] 设置服务超时

[0712] 设置对服务的访问控制

[0713] 设置服务的可用性(本地、全局、验证用户等)

[0714] 可以以包括但并不局限于:通过用户与 SDSCM 提供的用户界面进行交互,通过本地 API 以及通过服务调用机制的多种方式,来访问 SDSCM。与 SDSCM 交互的最后一种方法使得 SDSCM 被处理为具有适当 SSD 的 SDS。

[0715] 以下列出 SDSCM 的一些实施例。

[0716] PIPE(遍布实例提供环境)

[0717] PIPE 是用于发布和管理语义实例和语义服务的工具。可以将 PIPE 本身看作 SDSCM,因为为了建立并管理 SDS,它提供 web 服务 API。它具有两个主要功能。

[0718] 首先,作为管理工具,PIPE 对用户提供的实例和服务进行综合控制。

[0719] 图 66 示出 PIPE 管理界面的屏幕画面 6600。用户可以将服务/实例从遍布变更为本地,反之亦然。或者,保持一些服务/实例。如果服务/实例处于保持状态,则没有用户可以发现它,除非之后恢复该服务/实例。在 PIPE 中,可以对实例/服务分配任选有效期。PIPE 周期性地删除期满的实例/服务。最后,可以对各实例确定任选调用限制。如果将该限制赋予实例,则在其他服务每次检索该实例时,该数值减 1。在该数值到达 0 时,PIPE 将删除该实例。

[0720] 其次,它起作为其他 SDSCM 的核心的作用。在 PIPE 以及诸如白洞(White Hole)或通过该界面的图像服务、发布服务/实例的其他应用程序中,确定 Web 服务界面。

[0721] 在许多服务中,需要动态改变其 SSD 内的 IP 地址。另一方面,如果 TCE 部件可以适应自动变化的 IP 地址,则用户可以具有零配置 TCE。

[0722] 对于提供许多语义对象提供服务的 PIPE,这非常重要。PIPE 运行用于监视可能 IP 地址变化的单独线程。例如,利用其中利用令牌表示文件中出现的 IP 地址的模板文件表示

SSD。在初始化 PIPE 时,利用实际 IP 地址代替令牌。在检测到 IP 地址发生变化时,再装载窗格,并利用新 IP 地址产生最新 SSD。

[0723] 另一种方式是仅具有模板。仅在接收对 SSD 的请求时,它才检验 IP 地址,然后,利用 IP 地址产生新 SSD。

[0724] 应用程序的语义实例剪贴 (Scraper) 程序

[0725] 该服务对在用户当前使用的应用程序内建立的语义实例进行剪贴,并使它们在本地和全局可用 (或者使它们可以用于限制的人们 / 设备)。可以根据包括但并不局限于: 一些语义描述,或文本和对象的句法分析,剪贴语义实例。可以进行剪贴,以便即使在用户继续其他活动时,它仍可以将该语义实例高速缓存一段时间 (例如,一天)。在一些应用程序中,对于包括但并不局限于以下内容,实现剪贴程序:

[0726] web 浏览器

[0727] 电子邮件处理器

[0728] 电子表格

[0729] 文字处理器

[0730] 数据库

[0731] 表示

[0732] 剪贴程序将调用 PIPE 的 API 以发布被剪贴的语义实例。

[0733] 白洞 (事务、服务) (white hole (Thing, Service))

[0734] 白洞服务利用拖放界面发布对象和服务。可以将 OS 和应用程序中的对象,例如联系人管理软件中的文件、URL、联系人项目、日程安排管理软件中的日程安排项目动态地放到白洞上。该白洞将分析放入的对象类型,然后产生相应语义实例。在放入的对象是 SSD 文件或到 SSD 的 URL 的情况下,它将分别使用 SSD。

[0735] 它具有自己的管理界面 (基本上是 PIPE),以便发布并管理服务。除了其他参数外,它还允许用户设置使项目无效和 / 或被移除的缺省时间。在其中该服务使用户能够动态交换信息的 P2P (对等) 环境下,白洞非常有用。

[0736] 图 67 示出白洞拖放界面 6700。

[0737] 公共联系人 / 日程安排 / 文件目录

[0738] 通过显示服务,该 SDSCM 允许用户设置全局可用的联系人或日程安排或文件目录。在公共场所内,SDSCM 是白洞 SCSCM 以及联系人 / 日程安排 / 文件管理软件的特殊用途。缺省情况下,服务在特定时间周期之后消失。该服务最适于诸如商店街、高大建筑以及校园的公共场所。

[0739] 语义实例库服务

[0740] 文件库

[0741] URL 库

[0742] 联系人库

[0743] 日程安排库

[0744] 地址库

[0745] 服务库

[0746] 这种服务允许用户存储该环境下的一个或者多个文件、URL、联系人、日程安排、地

址、服务或任何其他语义实例。它接收一个或者多个 URL、URL、联系人、日程安排、地址、服务或其他语义实例作为输入，而不产生输出，或者产生控制窗格作为输出，以便由用户控制服务。在调用它时，它拷贝该实例，并建立提供该实例的服务。它利用 PIPE 发布这些实例提供服务，并允许其他人使用 PIPE 管理这些新建立的服务。

[0747] 该服务还是 SDS。请参考对库服务的 SDS 方面的“SDS 实施例”中的库所做的说明。

[0748] 通用库

[0749] 该通用库服务是所有语义实例库服务中的最一般形式。该服务匹配产生任何语义实例的任何一个服务。（该服务接受作为所有对象的超类的“事务”）。在调用它时，它拷贝该实例，并提供该实例以及用于实例提供服务的 SDS。在作为内部服务或本地服务运行它时，它可以起语义实例高速缓存的作用。

[0750] 该通用库并没有消除对诸如文件库、日程安排库的更专用库的需要，因为在许多环境下，并不始终要求允许用户保留该环境下的任意类型的对象。

[0751] 图像 / 音频 / 视频服务控制

[0752] 这种类型的 SDSCM 允许用户建立多媒体内容，而 SDSCM 可以自动使该多媒体内容用作这些多媒体内容的语义对象提供服务。或者，SDSCM 允许用户使该多媒体内容可用，并对如何使它们可用进行控制。

[0753] 例如，扫描 SDSCM 允许用户扫描他或她的材料，并将扫描的数据显示为图像实例提供服务。该显示允许用户设置扫描参数并尝试进行扫描直到他或她满意。在该结果满意时，该用户可以使该数据普遍可以用于例如在本地拷贝数据（通过与“本地拷贝”服务组成在一起）。通过该显示，用户可以设置临时口令。该服务可以根据页数或数据量对用户计费。

[0754] 作为扫描 SDSCM，数码相机和摄像机 SDSCM 可以建立图像提供服务和视频实例提供服务。或者，数字录音机 SDSCM 可以建立音频实例提供服务。

[0755] 可拆卸介质发布者

[0756] 在用户将可拆卸介质插入或连接到计算设备上时，该可拆卸介质发布者询问用户，该用户是否希望通过发布可拆卸介质上的文件。例如，用户将音频 CD 插入计算机，然后，可拆卸介质发布者询问该用户，她是否希望发布音频 CD 上的音频文件作为音频文件提供服务，允许其他人在音频 CD 上选择音频文件并产生选择的音频文件作为输出。其他可能性包括：(1) 用于在用户将 USB 数码相机插入计算机时，发布图像提供服务的可拆卸介质发布者；(2) 用于在用户将存储卡连接到计算机时，发布特定类型文件提供服务的可拆卸介质发布者。对于 (2)，可以建立该可拆卸介质发布者，以便如果在存储卡内发现多种类型的文件，则该可拆卸介质发布者发布多种类型的文件提供服务。

[0757] 可以建立可拆卸介质发布者，以便在它检测到某个可拆卸介质插入或者连接到计算机时，它自动产生（各）发布服务。

[0758] 传真接收机

[0759] 在接收传真时，传真 SDSCM 产生图像、文件或传真数据实例提供服务。它建立用于多页的多图像实例提供服务，或者建立用于例如 PDF 文件的文件实例提供服务，以包括所有传真页，或建立传真数据专用的传真数据实例提供服务。它可以与 OCR（光学字符识别）服务组成在一起，以对包括发送者的信息、头衔、备忘录等的传真数据产生更结构化的

对象。

[0760] 控制窗格切换

[0761] 控制窗格切换是用于将服务控制用户界面从一个用户切换到另一个用户的 SDSCM。这可以应用于具有控制窗格的任何 SDS, 例如“通过显示器观看”、“地图观看器”、“地图路线选择器”。该服务在其控制窗格上设置“切换”按钮。在用户点击该按钮时, 建立用于对控制窗格提供 URL 的新服务 (或多个新服务)。其他 (各) 用户可以检验新建立的服务, 并将它与某些观看服务组成在一起, 以显示该控制窗格, 然后, 控制原始 SDS。

[0762] 可以设计新建立的服务, 以便在经过特定调用次数之后, 就不出现它, 从而限制访问该控制窗格的人数。(通常设置为 1)。还可以将服务设置为经过特定时间周期后期满, 以便它仅在给定时间周期内有效。可以建立它, 以便原始用户使用的控制窗格自动关闭, 或者询问该用户她是否希望关闭它, 而且如果应答为是, 则关闭它, 或者如果应答为否, 则保持它。

[0763] 使用情况的预排工作 (Walkthroughs of Use Cases)

[0764] 接着, 我们将利用 TCC I 描述任务计算使用情况。

[0765] 在预排工作中, 存在文件提供服务、日程安排提供服务以及联系人提供服务。它们是本地或遍布实例提供服务, 它们分别提供文件、日程安排以及联系人。

[0766] 办公室

[0767] 让我们假定读者是一位销售人员, 并在它们的会议室第一次遇到 Bob 及其小组。允许你连接到对访问者设置的无线网上。你的 TCC I 发现该会议室内存在许多服务。你去操作“组成窗格”。

[0768] 商务卡交换

[0769] 商务卡交换=联系人提供服务+插入联系人列表

[0770] TCC I 发现运行在 Bob 的 PDA 上的 Bob 的“联系人提供服务”和运行在你的 PDA 上的“插入联系人列表”的组成。你认为 Bob 对你的商务非常重要, 因此你就按下“执行”按钮以执行该组成。将执行 Bob 的“联系人提供服务”, TCC I 获得其联系人。然后, TCC I 利用他的联系人执行在你的 PDA 本机上运行的“插入联系人列表”。就在将“插入联系人列表”插入 Bob 的联系人之前, 该服务对你弹出一个确认对话框, 以检验服务将插入你的联系人管理软件中的信息。对于该服务, 你按下“OK”按钮继续前进, 然后, 将它插入你的联系人管理软件。

[0771] 表示

[0772] 投影该表示=本地文件选择器+通过投影仪观看 (URL 观看器)

[0773] 你启动你的产品表示。你使你的表示文件安装在你的 PDA 上。在会议室内, 你使用“本地文件选择器”服务与“通过投影观看 (URL 观看器)”服务的组成。因此, 你使你的表示投影到投影屏幕上, 而使控制页面位于你的 PDA 上。利用该控制页面, 你利用无线方法控制你的表示。不硬连接到 VGA 电缆, 并按下正确的功能键。

[0774] 留下该表示=本地文件选择器+文件库

[0775] 如果没有问题, Bob 要求你留下你的表示。当然, 没有问题。你刚想, 在上次你执行“本地文件选择器”时, 应该建立了该表示文件的“文件提供服务”。无论如何, 你要在会议室内执行“本地文件选择器”与“文件库”服务的组成。现在, 可以从文件库提供的文件

提供服务获得该表示文件。

[0776] 打印使用说明书

[0777] 打印使用说明书=收藏 URL+ 打印（遍布）

[0778] Bob 要求你对你们的产品提供更详细材料。你记着你们公司网站上有产品使用说明书,而且在你的收藏文件夹内,你有到该使用说明书的 URL。你在会议室内执行本地“收藏 URL”与“打印（遍布）”的组成。你取下在会议室内的打印机上印出的材料,将它交给 Bob。

[0779] 成员买书

[0780] 买书=书提供服务+AMAZON.COM(众所周知的服务)

[0781] 在开会期间, Bob 向你介绍了一本新书。你在会议室内发现“书提供服务”,“书提供服务”对 Bob 的书提供书语义实例。你执行“书提供服务”与“AMAZON.COM web 服务”的组成。即使 AMZON.COM web 服务不提供其 SSD, TCC I 仍具有该众所周知服务的 SSD, 而 TCC I 可以将 AMAZON.COM web 服务用作一个 SDS。通过执行该组成, 你可以检验关于该书的信息, 而且可以购买它。

[0782] 在该会议室内召开会议的日程安排

[0783] Bob 和你安排在同个会议室内召开下一次会议。Bob 在其 PDA 的日程安排管理系统上建立日程安排项目。然后, 他将该项目拖到“白洞服务”。

[0784] 日程安排提供服务（白洞服务）+ 插入日程安排列表

[0785] 你执行 Bob 的 PDA 上的“白洞服务”提供的“日程安排提供服务”与你的本地“插入日程安排列表”服务的组成。

[0786] 日程安排提供服务（白洞服务）+ 日程安排库

[0787] 与此同时, 对于在该会议室内将“日程安排提供服务”用于下一次会议的其他人, Bob 执行“白洞服务”提供的“日程安排提供服务”与该会议室内的“日程安排库”的组成, 在执行后, “日程安排库”将提供“日程安排提供服务”。

[0788] 利用白洞服务管理窗口, Bob 停止“日程安排提供服务”。

[0789] 获得附近机场的路线指南

[0790] 地址提供服务（遍布）+ 地图路线选择器

[0791] 现在, 我们准备好要走。Bob 建议你执行你要去的机场的“地址提供服务”与“地图路线选择器”服务的组成。为了方便访问者, 该会议室提供该地区内感兴趣地点的“地址提供服务”。你在该会议室内、提供地图路线选择器服务的 LCD 上看到从 Bob 的办公室到机场的路线。你利用你 PDA 的浏览器上的控制页面控制地图路线选择器。

[0792] 以后表示

[0793] Bob 做的表示=文件提供服务（文件库）+ 通过投影仪观看（URL 观看器）

[0794] 在 Bob 之后对他公司内的其他成员说明你们的产品时, 他使用你保留在会议室内的表示文件。他执行文件库提供的你的表示文件的“文件提供服务”与会议室内的“通过投影观看（URL 观看器）”服务的组成, 以给出该表示。

[0795] 公共场所

[0796] 下面的情况可以应用于诸如商店街、机场以及火车站的公共场所。在此, 我们选择商店街情况。

- [0797] 你计划与朋友一起在节日进行购物。
- [0798] 保留备忘录
- [0799] 位置确定 (本地 / 遍布)+L- 便条
- [0800] 你已经等了你朋友 15 分钟。你确信你朋友最终会来的,但是你想找一个座位,然后喝一杯咖啡。因此,你执行该商店街提供的“位置确定”服务以及该商店街提供的“L- 便条”服务,从而对你朋友留个便条。“位置确定”提供你的当前位置,将该当前位置送到“L- 便条”服务以产生你的 URL,从而在你的当前位置留下备忘录。
- [0801] 你留下用于告诉你朋友,你将在附近的咖啡馆等他的备忘录。
- [0802] 发现该备忘录
- [0803] 位置确定 +(本地 / 遍布)+L- 便条
- [0804] 之后,你朋友来了。但是你朋友未发现你在那里。你朋友执行“位置确定”服务和“L- 便条”服务,以发现你留的消息并阅读它。
- [0805] 因特网服务台 (internet kiosk)
- [0806] 阅读新闻=收藏 URL+ 通过因特网服务台观看 (URL 观看器)
- [0807] 你到达该商店街附近的咖啡馆,要了一杯咖啡,坐在桌边。你发现在你所在的桌子上有一个因特网服务台。因特网服务台实际上是具有计算机的 LCD。计算机被隐藏,而且为了不让人们把它搞乱,它既不设置键盘,又不设置鼠标。应该通过提供“通过因特网服务台观看 (URL 观看器)”的控制页面使用它。
- [0808] 你决定通过因特网服务台阅读新闻来消磨时间。你执行你的本地“收藏 URL”与设置在咖啡馆内的“通过因特网服务台观看 (URL 观看器)”的组合。你从对话框中选择你收藏的新闻网站,在对话框中弹出“收藏 URL”。然后,新闻网站出现在因特网服务台上。你利用你 PDA 上的浏览器上的控制页面控制该因特网服务台。
- [0809] 欣赏照片=本地文件选择器 / 白洞服务 + 通过因特网服务台观看 (URL 观看器)
- [0810] 经过一段时间之后,你朋友出现了,连声道歉。你接受他的道歉。然后,你和你朋友开始谈论你上次旅游的情况。通过执行“本地文件选择器”与“通过因特网服务台 (URL 观看器)”的组成,你通过因特网服务台显示几张你的最佳照片。你在对话框内选择照片文件,然后,照片显示在因特网服务台上。
- [0811] 公用目录
- [0812] 然后,你和你朋友逛商店街。
- [0813] 公用联系人 / 日程安排目录 + 插入联系人 / 日程安排列表
- [0814] 你发现公用联系人 / 日程安排目录。你在目录中发现你希望进入的商场的联系人。你将它拖到白洞内。然后,联系人提供服务出现在 TCC I 上。你执行该服务和“插入联系人列表”服务,以将商场联系人插入你的 PDA 上的联系人管理系统中。
- [0815] 在该目录上你还发现一件感兴趣的事件。利用同样的步骤,你将该事件日程安排插入你的 PDA 上的日程安排管理系统。
- [0816] 10 分钟之后 (商店街经理已经设置缺省超时期间),服务消失。
- [0817] 商场联系人提供服务列表 + 列表选择器 + 插入联系人列表
- [0818] 另一种可能情况是与下面的情况类似。TCC I 发现“商场联系人提供服务列表”。TCC I 将“商场联系人提供服务列表”、“列表选择器”以及“插入联系人列表”的组成显示为

“插入联系人列表”“商场联系人之一”。通过执行该组成,你得到弹出的对话框,以便你从该表中选择一个商场联系人。选择之后,你将该联系人插入你 PDA 上的联系人管理系统中。

[0819] 楼层地图

[0820] 位置确定(本地/遍布)+楼层地图

[0821] 通过执行“位置确定”服务与“楼层地图”服务的组成,你发现你在该商店街内的位置。你获得对你的位置加了亮点的该商店街的楼层地图。

[0822] 根据位置过滤服务

[0823] 根据楼层,或者你所在的区域,你可以过滤服务。TCC I 利用“位置确定”之一来过滤服务。TCC I 将“位置确定”产生的“位置”的本地用于该过滤器所使用的级(例如楼层或区域)。你可以选择 TCC I 上对服务进行过滤的级。经过进行过滤,你在 TCC I 上就有合理数量的服务了。如果你选择区域级过滤,则你仅在你的 PDA 上获得区域或本地服务。

[0824] 获得路线指南

[0825] 地址选择器+地图路线选择器

[0826] 购物之后,你和你朋友决定去你几个月前发现的一家餐馆。在你的联系人管理系统中有这家餐馆的地址,但是你不知道如何从商店街到该餐馆。幸运的是,你发现在其上具有“地图路线选择器”的因特网服务台,该因特网服务台是商店街为了方便访问者设置的。通过执行“地址选择器”和“地图路线选择器”的组成并在弹出的“地址选择器”对话框上选择该餐馆的地址,你在因特网服务台上发现该路线。

[0827] P2P

[0828] 在该小节,我们将说明一些 P2P 情况。下面的情况不必进行因特网连接。它们应该通过 WLAN 和蓝牙的特定模式工作,而不会有任何问题。不访问本地的一个可能限制是,服务的组成仅局限于输入签名与输出签名完全匹配。它们可以通过把曾经使用过、经常使用和/或众所周知的本地本地保持在计算设备上防止。

[0829] 共享餐馆信息

[0830] 白洞服务+插入联系人列表

[0831] 你想与你朋友共享关于你最近发现的餐馆的信息。你将联系人项目从你的联系人管理系统拖放到你的 PDA 上的白洞服务上。然后,你朋友执行你的 PDA 上的联系人提供服务与你朋友的 PDA 上的“插入联系人列表”的组成,以将联系人插入他或她的联系人管理系统。

[0832] 共享上次旅游中的照片

[0833] 白洞服务+本地拷贝

[0834] 你希望与你朋友共享上次旅游中的照片。在你的 PDA 上,你将照片文件从你的文件管理系统拖放到白洞服务上。然后,你朋友执行你的 PDA 上的文件提供服务与你朋友的 PDA 上的“本地拷贝”的组成,以将照片文件拷贝到他或她的文件系统中。

[0835] 共享收藏的音乐

[0836] 白洞服务+本地拷贝

[0837] 你希望与你朋友共享你喜好的新发布的音乐。在你的 PDA 上,你将音乐文件从你的文件管理系统拖放到白洞服务。然后,你朋友执行你的 PDA 上的文件提供服务与你朋友的 PDA 上的“本地拷贝”的组成,以将音乐文件拷贝到他或她的文件系统中。

[0838] 益处

[0839] 任务计算对最终用户和开发商均提供了实质性好处。

[0840] 最终用户

[0841] 任务计算的主要受益者是最终用户。最终用户从任务计算受益,因为:(a) 任务计算减少了使用当今的复杂环境所需的专业知识,(b) 在执行复杂任务时,它们可以节省时间,以及(c) 对于用户与遍布计算环境交互,任务计算是一种更自然、有效的方式。

[0842] 任务计算试图重新定义用户如何与计算环境交互以及如何使用计算环境。任务计算的基本前提是,对用户可能处于用户当前环境中的任务,以帮助用户利用较简单任务,象搭积木一样,建立更复杂任务,并指导用户执行该复杂任务。一旦用户(或者由其他用户)定义了复杂任务,就可以再使用这种任务,非常象当今的流行应用程序中的宏。与宏(至少 MICROSOFT OFFICE 用户熟悉的宏)不同,积木任务以及获得的复杂任务可以跨越多个应用程序和多个计算平台(不是单个设备)。实际上,任务计算的一个目标是,用户不需要知道,也不需要关心所执行的实际组成任务,或者在何处(在哪个机器上)执行它们。任务计算的最终目标是,对用户显示用户可以特殊方式使用的资源(设备和服务)的简要说明(abstract view),以执行任意复杂性的任务。

[0843] 以下我们可以对任务计算对最终用户的优点做个总结:

[0844] 每天用于用户任务计算的计算不需要用户深刻理解她的计算环境和其内的可用资源。任务计算对用户显示用户当前环境中可能的和可行的可能性(要做的事务),并在建立并执行更复杂(而且可以再使用的)任务的过程中指导用户。

[0845] 对于所有用户更节约时间

[0846] 对于新手和高级用户,任务计算均可以节省时间,因为该系统(而不是用户)以与用户环境和对象一致的方式对应用程序之间的工作流和数据传送进行管理。此外,由于任务一旦被建立就可以再使用,所以用户只需一次性定义复杂任务。

[0847] 遍布环境解决方案

[0848] 所知晓的计算环境是遍布计算环境,其中充满了事先不被最终用户所知的设备和服务。此外,这种环境下的用户设备(用户使用它访问遍布环境的功能性的设备,例如,PDA、移动电话等)具有在物理上和功能上受限的用户界面,这意味着,作为“程序员”的用户个人计算环境作法是不可行的。通过仅对用户显示可行的功能性,而且通过在这种环境下部分地自动执行复杂任务,同时通过减少用户监视任务执行过程并根据需要提供输入,任务计算以个人计算作法不能实现的方式,解决了遍布计算环境的特质问题。此外,因为任务计算的实时性意味着用户可以在他们规定和执行任务的时间“查看”什么可行,所以任务计算解决了遍布计算环境的动态性,其中设备和服务可以随时工作,而且最终用户通常事先对它们知之甚少。

[0849] 开发商和商业机构

[0850] 但是,最终用户不是从任务计算受益的唯一受影响方。商业机构和为他们开发应用程序的开发商是其他受益者。在过期的 10 年中,我们感受到 web 进步的革命性影响。Web 包括各用户可用的大量系统和应用程序,同时,拥有并控制这些系统的商业机构必须通过开发商的劳动将这些(通常是后端)系统“引入”web。这种变革改变了用户与计算系统交互的方式,但是随着变革的进步,这是不完整的。用户与计算系统交互,进行搜索、输入、剪切

和粘贴、处理和理解这些应用程序提供的任何输出。具有常识的用户理解如何对拖到任务或目标使用特定应用程序 (web 网站), 或者正如通常发生的那样, 为了实现一个目标, 如何与多个 web 网站交互。该方法存在两个基本缺点: (a) 用户被过度地涉及和依赖, 以便执行复杂任务, 如前面在本文件中描述的, 以及 (b) 商业机构不能从将他们的应用程序引入 web 的投资中获得资金收益。因为呈现给用户的所有功能性被仔细地、但是不可改变地提供到 web 应用程序 (用户面对着观看应用程序), 使应用程序或系统成为 web 应用程序将产生一个静态系统。此外, 为了建立新感兴趣的应用程序, 该功能性不易与其他 web 应用程序 (可能由其他商业机构拥有) 的功能性组合在一起。最后, 该设计时间 (design-time) 构建耗时, 易出错并因此而非常昂贵。

[0851] 最后一点是重点, 因此需要对其做进一步详细说明。推广 web 服务的广泛努力部分地是因为当前 web 应用程序的局限性导致的。成功的 web 服务使得利用现有 web 应用程序, 特别是将现有 web 应用程序的功能性表示为 web 服务, 然后将它们用作新应用程序的积木, 来开发新应用程序变得更加容易了。此外, 从这些新“元”应用程序仍变成具有设计者将其设计到其实现过程中的功能性和可行性的静态应用程序意义上说, web 服务是设计时间技术。

[0852] 任务计算的目的在于, 使功能性与提供它们的系统脱离关系, 并丰富它们, 以便对于更复杂的任务, 用户以他们希望的任何方式将它们组合在一起。该新范例具有深远的效果, 而且对于计算机工业提供最终用户功能性的过程有益。

[0853] 图 68 示出任务计算之前的能力金字塔 (Capability Pyramid)。通常, 如图 68 中的金字塔所示, 最终用户能力由建立在计算机 (硬件和软件) 平台顶部的应用程序提供。因为每个应用程序是对特定用途设计的, 而因为它提供通过设计编程提供的用户功能性, 所以扩展最终用户能力需要新应用程序 (或设备), 或者在这种应用程序与设备之间的定制连接。因此, 少量提高最终用户能力要求在下面的层进行实质性投资 (劳动力、金钱、时间), 因为对用户提供的功能性与应用程序和这种应用程序运行的平台紧密相连。

[0854] 该任务计算范例可以将大量附加能力 (用户可以做的事务) 传送到用户, 而在对下面的层, 即, 设备和应用程序以及支持它们的平台提供的功能性进行很少量投资。

[0855] 图 69 示出进行任务计算之后, 相反的能力金字塔范例。将传统金字塔倒置的效果是, 由于任务计算, 所以利用给定的一组设备和应用程序增加用户能力的成本非常低。

[0856] 任务计算的又一个益处在于, 显著提高诸如 web 服务和 UPnP 服务的 e 服务的价值, 因为它增加它们的用户基础和潜在用途。程序员 / 开发商建立 Web 服务, 而且开发商将使用该 web 服务。为了利用它们建立应用程序, 而且还因为缺少与这些 e 服务的功能性有关的较深语义并且其内部工作需要人检查, 该约束与需要实质诀窍的程序设计界面的复杂性有关。此外, 甚至简单发现 (或者仅知道) 这些 e 服务也是开发商驱动的过程, 因为缺少语义。因为任务计算可以使这些 e 服务对于进行网上访问的所有人 (以及任务计算客户机) 中的非常广泛的用户基础可用, 而且因为每个这种用户均可以以各种方式将这些 e 服务中的每个 e 服务组成在一起, 所以与对这种 e 服务的开发商设计的 e 服务的任务不同的任务通常接受多个用户, 并使用它们开发的 e 服务。

[0857] 再参考图 13 和 14, 因为在现有 web 服务的顶部引入 web 服务的语义层抽象概念, 而且因为构思、定义用户任务, 而且就用户而论, 在语义层执行, 所以, 升级提供该服务的实

际服务 (web 服务), 或者修改它, 或者替换仅是以用户不察觉的方式, 将 web 服务的语义描述对手头的新服务的连接进行变更; 这种方式不存在与改变所采用的 web 服务相关联的成本。

[0858] 这就是任务计算引起兴趣之处。与静态元 - 应用程序相反, 通过实时建立并执行他们自己的任务, 用户可以实时建立它们自己的“应用程序”。对于开发商和商业机构的益处在于, 可以以各用户随心所欲的各种未计划方式, 采用 (使用) 商业机构使其可以用作 web 服务的功能性。

[0859] 应用程序

[0860] 任务计算取决于语义描述服务和信息内容的能力。根据服务的语义描述, 该系统使用户通过密集交互执行任务。在这个意义上, 任务计算可以被看作与被称为语义 web 的 web 的下次重复大致相关的语义启动应用程序。即使语义 web 的广泛成功取决于广泛采用关于任何事情“向那边 (out there)”的本体, 任务计算仍具有更适中的要求, 因为不想使它成为完全自动的系统。

[0861] 此外, 任务计算还是一种解决新遍布 (或普遍存在的) 计算环境的复杂性的方法, 这种计算环境由使其功能性可以用作服务的大量设备构成。

[0862] 即使用户客户机设备的因特网连通性可以最完全采用任务计算, 但是对于任务计算, 不要求全时因特网连通性。受限连通性导致对系统产生受限本体可用性, 并因此导致系统能力受限。但是, 即使本体不可用, 该系统仍提供某种功能性, 因此, 每当网络连通性不可用时, 可以将经常使用的本体高速缓存或存储到用户设备内, 并被 TCC 使用。

[0863] 任务计算用途的一大类是遍布环境, 因此利用无线装置的网络连通性非常有益 (无线 LAN 特殊模式、无线 LAN 信息基础设施模式、蓝牙 (PAN)、IR 等)。

[0864] 接着, 我们将提供一些正在使用的任务计算的例子。我们假定, 用户正在使用运行诸如 TCC I 的任务计算客户机的计算设备, 或者正在使用可以与诸如 TCC II 的任务计算客户机交互的 web 浏览器。我们将对在不同环境和场所使用的任务计算例子进行说明。

[0865] 家庭环境

[0866] 许多家用电器提供可以利用较清楚、简单语义描述的功能性, 因为这种功能性具有良好定义的域和有限范围。因此, 家庭环境是任务计算的良好应用领域。随着家用电器变得更小, 连接的家用电器的数量更多、更遍布 (换句话说, “更不可视”), 而且这些家用电器的数量不断增长, 家庭环境下的用户发现 (或找到) 设备 (或这些设备提供的服务)、以容易理解的方式对用户显示它们以及允许用户利用较少努力控制设备 (或执行服务) 变得非常难, 这些就是任务计算的目标。

[0867] 图 70 示出应用程序 eHome 7000 的例子。除了通常可能存在于家庭 7002 内的诸如具有 TCE 7004 的 PC、家庭网关路由器 (HGR) 以及 PDA 7006 的传统计算设备外, 其他多功能设备也连接到有线或无线网上。“家庭监视设备”7008、“视频适配器”7010 以及“音频适配器”7012 是 3 种这种设备。任务计算服务模块安装在它们内, 这样将该设备变换为语义服务提供者。模块既可以是软件包, 又可以是硬件附属装置。HGR 可以是允许家庭用户互相共享文件, 或者在所有者允许的情况下, 与外部用户共享文件的共享文件服务器。在这种环境下, PC、PDA 或移动电话 7014 可以用于进行任务计算。

[0868] 在防火墙 7016 之外, 存在遍布服务 7018 和动态服务入口 7020, 它们提供可以与用

于任务计算用途的 eHome 服务组成在一起的附加服务。

[0869] 家庭用户可以利用该任务计算环境执行各种任务。它们可以使用目录发布器以在其 PC 上层发布视频、音频以及图像目录,并在诸如 TV 和录音机的传统设备上播放它们。在家里,他们可以利用他们的 PDA 或移动电话遥控该设备。如果他们不在家,为了利用他们的 PDA 或移动电话遥控该设备,他们仍可以通过 SSL 反代理 (reverse proxy) 实现。他们可以将家庭监视设备视频作为视频文件存储到其 PC 内,并在之后,在家时,或者外出时,重新观看它。他们可以将约定或联系人存储到动态服务入口,以便之后随处使用它们。利用以上说明的任何一种任务计算发布机制,他们可以与他们的朋友共享图片。这些仅是家庭用户可以做的一些事务的例子。重要的是应该注意,在该环境下,未对所有这些任务进行编程。一旦该功能性可以用作服务,则该任务计算环境就可以使能所有这些任务。从用户的观点出发,利用用户使用的、设置在台式计算机、膝上型计算机、PDA 或移动电话(无论位于家里还是位于家庭外部)上的任务计算客户机的用户界面,所有这些任务是可用的。

[0870] 办公室环境

[0871] 办公室环境是任务计算的另一个感兴趣应用。设想访问者(或雇员)在办公室中。我们假定访问者携带某种计算设备(或用于该事务的各种计算设备),例如膝上型计算机或 PDA。在访问者在办公室内移动时,她的任务计算环境为她显示她可以根据其环境执行的事务表以及其他设备和服务的实时可用性。

[0872] 可以利用各种服务“装备”该办公室环境本身,许多服务起源于办公室内的设备。

[0873] 图 71 示出 eOffice 7100 的例子。在会议室 7102 内,运行“通过投影仪观看”、“音频播放器”、“数码照片像框”、“目录发布器”、“拨号器”、“通过因特网服务台观看”、“地图观看器和路线选择器”、“遍布打印”、“视频播放器”、“图像服务控制”以及许多“其他遍布服务”。在用户个人的设备上,安装 TCE 和某些“本地服务”。借助 S- 备忘录和白洞,将远程服务加入该环境。

[0874] 再参考图 71,通过网络 7104,会议室 7102 连接到地图观看与路线选择器 7106、通过服务台观看服务 7108、拨号器 7110、目录发布器服务 7112、数码照片像框 7114、音频播放器服务 7122、视频播放器服务 7124、遍布打印服务 7126 以及其他遍布服务 7128。本地服务 7120 还与远程服务 7130 通信。

[0875] 从用户的观点出发,重要的是请注意,用户事先不知道所有这些服务的可用性,而且她也不知道如何访问它们和使用它们。用户的任务计算环境负责自动识别这些服务、理解它们提供的功能、而且仅负责访问它们并使用它们。给定可用资源的情况下,用户负责定义她的任务。对于她,在特定环境下,资源与她可以执行的动作有关。办公室内的这些(原子)动作(作为服务执行的)包括:

[0876] 使用观看服务

[0877] 通过公用显示器

[0878] 通过投影仪

[0879] 查看对会议室安排的会议

[0880] 将会议添加到会议室日程安排中

[0881] 对于特定会议室对未来的访问者留下便条

[0882] 将其他雇员或访问者的联系人信息附加到用户设备上的 PIM 应用程序中

- [0883] 查看当前位置附近的宾馆列表
- [0884] 查看当前位置附近的餐馆列表
- [0885] 在地图上观看选择的位置
- [0886] 观看本地机场的列表
- [0887] 为了对其执行某些动作,从用户设备选择文件或 URL。
- [0888] 观看任意两个位置之间的路线
- [0889] 尽管这些动作(功能性或服务)中的一些动作的兴趣可能在它们自己的权利,但是任务计算的真正价值是,用户可以将这些服务的输入与输出组合在一起,以定义并执行更复杂动作(任务)。一旦用户准备好这样做,运行在用户设备上的任务计算环境负责对用户显示,可用动作(服务)可以被组合在一起(考虑“积木”比喻)并执行复合任务的可能方式。这些组成动作(任务)的例子包括:
- [0890] 通过公用显示器观看特定餐馆在地图上的位置,或者从当前位置到该餐馆或到机场的路线。
- [0891] 从用户的本地设备上选择要显示在公众可访问的显示器或投影仪上的 URL 或文件(例如,使用投影仪进行显示,而不必将电缆或导线连接到投影仪)
- [0892] 以类似于交换商务卡的方式,将会议室内共同参与人的联系人信息插入运行在她的 PDA 上的用户的 PIM 内。
- [0893] 将下一次会议的日程安排留在会议室内
- [0894] 重要的是请注意,这些更复杂任务都没有被设计到该系统内。用户负责利用她自己的设备上的用户界面“建立”这些复杂任务,同时与运行在她的设备上的任务计算环境交互。该环境负责执行用户定义的任务。该遍布环境下的可用服务的提供者(程序员)未考虑到他们可用的这些服务的特定用途。在取决于用户的希望和环境的实时(执行时间)中,这些服务用作实现更复杂任务的积木,用户将永远不必直接与这些服务交互。
- [0895] 公用设施(机场、火车站等)
- [0896] 在遍布环境下任务计算的另一个例子中,我们说明带着她的 PDA 或移动电话绕机场转的旅行者的任务计算经验。如图 72 所示,在机场内建立“遍布服务”,并将该服务连接到“网络”。让我们假定在该场所内下面的服务可用。
- [0897] 使用通过公用显示器的观看服务
- [0898] 查看在机场或周围区域内发生的事件列表
- [0899] 查看机场内的设施列表(休息室、餐馆、咖啡馆、吸烟区)
- [0900] 购买报纸或杂志
- [0901] 粘贴分类附加
- [0902] 为了对其执行某个动作,从用户设备选择文件或 URL
- [0903] 在地图上观看选择的位置
- [0904] 观看两个位置之间的路线
- [0905] 观看航班信息(起飞、到达、入口)
- [0906] 关于地面运输的信息
- [0907] 如图 72 所示,遍布服务 7018 与网络 7204 交互,网络 7024 还与 PDA7206(或移动电话)通信。

[0908] 一旦用户或者通过 web 连接,或者利用诸如蓝牙或 WLN 的本地网络连接连接到网络,她就可以执行下面的任何一项任务:

[0909] 通过公用显示器(旅行者选择的,或者由她的任务计算系统根据她的位置选择的)观看到旅行者的连接入口的方向指示(利用她的 PDA 上的、诸如 MICROSOFT OUTLOOK 的个人信息管理应用程序,或 PIM 中的旅行者的航班信息),

[0910] 同样,观看到最近的吸烟区或休息室或咖啡馆或餐馆的方向指示,

[0911] 通过公用显示器观看从机场到机场附近发生事件的位置的方向指示,并对她的 PDA 上的、旅行者的 PIM 应用程序附加正确条目。

[0912] 图 72 示出应用机场 7200 的例子。图 72 示出客户机设备的屏幕画面 7208。如果客户机设备含有 TCE 模块,则它们可以将它们的本地服务与遍布服务组成在一起。然而,这不是强制性的。

[0913] 还应该注意,这些复杂任务均未被预编译,但是它们是用户利用该环境下的可用服务以及用户动态建立、而被用户的任务计算环境执行的服务,实时(利用可用资源)构建的。

[0914] 零售区(咖啡馆、商店街等)

[0915] 图 73 示出客户机设备的屏幕画面 7308。此外,图 73 示出与网络 7304 通信的遍布服务 7108,网络 7304 还与 PDA 7306(或移动电话)通信。

[0916] 与对机场的方式类似,在诸如咖啡馆、商店街等的遍布计算环境下,利用在上述例子中描述的概念,任务计算可以帮助最终用户完成复杂任务。例如,可以对咖啡馆应用同样的设置(图 73)。单独、组成服务的提供者仍仅提供可以被用作积木的服务,以便用户构建并执行复杂任务。

[0917] 携带 PDA(或移动电话)的客户进入咖啡馆。咖啡馆提供的一些服务可以包括:

[0918] 通过安装在座台上的显示器,或公用等离子体显示器,使用观看服务,

[0919] 查看发生在特定咖啡馆或周围区域内的事件列表

[0920] 对可能到该商场的的朋友留下个人便条,

[0921] 查看菜单,

[0922] 购买报纸或杂志

[0923] 粘贴分类附加

[0924] 为了对其执行某个动作,从用户设备选择文件或 URL

[0925] 在地图上观看选择的位置

[0926] 观看任何两个位置之间的路线

[0927] 利用运行在她的 PDA 上的任务计算环境,客户可以尝试做以下事情:

[0928] 浏览菜单并下订单,在她所在的桌子上传送该订单(因为 TCE 确定她的位置),

[0929] 确定执行顺序、付款以及在订货准备好可以取时发出通知

[0930] 浏览可用供应,并选择电子报纸以购买,然后在用户最终坐在任何位置时,在台式安装的 LCD 上显示该报纸。

[0931] 发现某种事件的通告、根据判定参加一些事件,根据日程安排和指示检查该事件,而且最终将该事件附加到她的日程安排中,并将一个便条发送到朋友与她一起去,

[0932] 对在她离开后可能到该咖啡馆的的朋友留个便条,例如“我去‘United Artists’电影

院看 ‘Minority Report’ 了,如果你喜好请与我一起看”。她可以包括到电影院的方向和关于 “Minority Report” 日程安排。

[0933] 在通过台式安装的显示器观看时,可以与朋友共享照片 (存储在客户的 PDA 上),

[0934] 在该咖啡馆内,为之后要到该咖啡馆的其他朋友保留照片。在最后的例子中,用户建立在他或她的 PDA 上提供照片文件的新服务,这样,他或她的朋友将执行该新服务以将它们拷贝到他们的 PDA 中。

[0935] 为了这些任务中的一些任务,消费者可能必须访问在更广阔的因特网上提供的服务,但是仍可以从该咖啡馆位置进行访问。与以前相同,用户可以以以前未曾设计到系统内的方式,将可用服务用作复杂任务的积木。

[0936] 台式机 / 膝上型机环境

[0937] 遍布计算环境不是任务计算的唯一应用领域。任务计算还将用户在当今的个人计算环境下最终执行复杂任务的方式变换为用户使用她的膝上型机或台式机时的情况。

[0938] 让我们对以下情况进行研究。正在浏览因特网的用户遇到感兴趣事件,说了一句她有兴趣参加。此发现之后,她手动执行以下操作:

[0939] 在搜索引擎上输入位置名称 (在这种情况下为宾馆),以查找该事件的准确地址。搜索引擎的一个例子是 GOOGLE™。

[0940] 在浏览了返回的结果后,用户遇到具有关于该宾馆的进一步信息的网页,而在访问该宾馆的网页后,她可能发现该宾馆的准确街道地址。

[0941] 掌握了该地址,她可以将该地址输入到因特网映射服务的搜索窗口内,以查看地图上的位置。

[0942] 如果用户判定该位置便于参加,则她可以检查是否与她的台式机上的 PIM 应用程序内的日程安排冲突,然后,在判定参加该事件后,她可以回到映射服务,输入她的家庭地址或工作地址,并获得到该事件位置的消息指导。

[0943] 该步骤之后,她可能将该事件附加到她的 PIM 应用程序,再输入事件信息,或许利用驾驶指南和 / 或报告通告。

[0944] 最后,她希望使感兴趣加入该事件的同事注意该事件,并给他发送电子邮件,再附上该事件的描述以及到该事件的指南或地图。

[0945] 甚至对于知道如何执行这一系列步骤的熟练用户,执行上述任务仍需要在她的机器上调用应用程序以及将信息从一个应用程序传送到另一个应用程序方面花费相当长的时间 (或许 15-20 分钟)。此外,在下一次用户遇到感兴趣事件时,她必须通过同样 (或类似) 的步骤,而且在执行复杂任务方面,花费同样长的不必要时间。最后,该手动过程易出错,这本身可能导致附加时延而且令用户不满。

[0946] 让我们研究使用任务计算用户的经验如何不同 (参考图 74 至 76)。用户的任务计算客户机 (TCC) 被动运行,而用户忙于她的日常任务。

[0947] 用户遇到感兴趣对话的网页。

[0948] 在用户遇到一个事件的网页时, TCC 识别网页上信息的性质 (意义或语义),即这是一个特定类型的事件,识别事件的发起人、地址、时间等,对用户显示 TCC 可以代表用户执行的可能并可行的动作,如图 74 所示。

[0949] 图 75 示出 TCC 识别在网页上描述了什么,并对用户建议一些提供当前内容和环境

的可能动作。

[0950] 用户可以选择发现该位置的地址的动作,这样支持 TCC 搜索因特网上含有该位置的地址的网页(宾馆名称)。

[0951] 然后,TCC 更新给出新信息的用户可用动作表,并动作给出新信息的、修改的动作表 7500,如图 75 所示。

[0952] 图 75 示出在用户首先进行选择以发现对话位置的地址后,TCC 发现该地址并对用户显示新可能动作表。

[0953] 然后,用户选择查看从她家到该位置的指南的动作,TCC 实际显示有关位置表,以便如果可用,用户从诸如家庭地址、工作地址或用户的当前位置中进行选择。

[0954] 如果用户判定该位置方便,则用户可以选择此时,在 TCC 显示的修改的屏幕上,将事件附加到她的 PIM(如果与安排的其他活动不冲突)。此外,她可以选择包括将电子邮件发送到同事的动作(再从用户显示的表中选择)以及将所有相关信息附加到电子邮件 7600,如图 76 所示。

[0955] 图 76 示出在用户在地图上进行选择以检验对话位置的地址的位置后,TCC 显示具有选项的修改的动作表以保存此时之前的用户动作作为未来(再)使用的新任务。

[0956] 在该处理过程的任何一部分,用户均可以进行选择以保存序列步骤和动作,作为未来使用的任务,因此,在下次用户在同样或类似的情况下发现她自己时,组成任务(序列动作)可以变成 TCC 显示的可能并可行的动作表上的可用动作。

[0957] TCC 仍负责执行每个这种动作,并在用户引导工作流的任何点对用户显示更新的可能并可行的动作。由于 TCC 能够“理解”网页的信息和可能动作以及它们的输入和输出的意义,所以结果处理过程比手动执行 workflow 快得多。

[0958] 请注意,利用用户动作自动、动态改变该环境的情况。(新语义对象变成可用,因为用户导航 web。)这些环境的变化触发了上述序列。

[0959] 虚拟社区

[0960] 利用任务计算,各个体可以构成虚拟社区,其成员可以共享与该社区有关的服务以及个人拥有的服务,例如,与拥有的设备有关的服务,和/或他们中的每个个体运行的服务,利用其他任意一般可用服务,每个成员可以为他们自己或者为该社区编写这些服务。

[0961] 对该社区(或者由社区的成员,或者由提供或者支持该虚拟社区功能的实体)准备“社区服务发现”目录,而对其成员提供用于注册或去注册与该社区有关的服务以及个人拥有的服务,或者用于发现社区服务的界面。上述说明的发布与管理任务计算环境机制(SDSCM)可以用于共享社区内的服务。特别是,白洞可以用作用于注册诸如为社区提供服务的实例的图形用户界面。

[0962] 用户可以编写并执行包括社区服务和每个单独用户可用的其他服务的任意组成。

[0963] 企业应用程序集成(EAI)

[0964] 即使它可能花费一段时间,但是我们仍相信任务计算可以逐渐应用于企业应用程序集成(EAI)应用领域。任务计算对 EAI 的一个可能应用是通过交互和使用服务语义,支持 EAI 应用的开发商。开发商具有较短的开发时间,而且对任务计算 EAI 应用的维护支持不易出错的服务集成过程和有效试错过程。

[0965] 动态服务入口

[0966] “动态服务入口”基本上是一个语义 web 服务宿主。这可以是为了对其他免费商务提供附加值设置的、或者是为某个组织设置的室内服务等收费服务。

[0967] 动态服务入口的提供者准备一组在需要时可以利用某些访问控制的情况下,可以从网络上的某些区域访问的 web 服务。当然,用户可以访问这些服务的区域和情况越多,该用户就对用户越有用。

[0968] 例如,动态服务入口的个人版可以是提供这种入口网站的功能性、被显示为语义 web 服务的“MY YAHOO!”的语义 web 服务版。在这种例子中,动态服务入口提供的服务可以包括:

[0969] ●类属文档文件的远程文件夹服务

[0970] ●对诸如音乐、图片、视频等的特定类型文件规定的远程文件夹服务

[0971] ●用于联系人、日程安排等的个人信息管理 (PIM) 服务

[0972] ●诸如天气、地图、金融信息、新闻等的信息提供服务

[0973] 多媒体远程文件夹服务可以与在线购买数字内容组合在一起。例如,在用户购买一个音乐作品时,该音乐文件出现在音乐文件的远程文件夹上。

[0974] 动态服务入口的提供者允许其用户具有这些 web 服务的语义服务描述 (SDS)。提供者可以将附加在电子邮件上的这些 SDS 发送到用户,或者利用电子邮件或者通过 web 网页,将到这些 SDS 的 URL 通知用户。

[0975] 通过利用诸如白洞的 SDSCM,使这些 SDS 在本地或者遍布可用,用户可以在任务计算环境 (TCE) 下使用这些 web 服务。在用户使 SDS 在本地可用时,相应服务变得仅对该用户可用。在用户使 SDS 遍布可用时,相应服务变得不仅对该用户可用,而且对同一个环境下作为用户的其他人可用。在 TCE 的一个实施例中,用户可以将 SDS 文件本身或到该 SDS 文件的 URL 拖到白洞,以使它在本地或遍布可用。利用 PIPE,用户可以将服务的可用性变更为本地可用或遍布可用,也可以临时保持该 SDS。

[0976] 在用户使动态服务入口提供的特定 web 服务的 SDS 对她自己可用时,该用户可以在 TCE 内使用该 web 服务。这意味着,用户可以使用与本地、遍布或其他远程服务组成在一起的 web 服务。这就是诸如“MY YAHOO!”的动态服务入口网站的优点。一个入口网站的各服务的互操作可以直到进入入口网站。结果某些入口网站对位于入口网站的数据和用户计算设备上的应用程序中的数据提供同步软件,但是该软件是独占的,而且互操作在此结束。如果你超过它,你必须在 web 网页与应用程序之间进行剪切和粘贴。可以在 TCE 内实现动态服务入口提供的语义 web 服务,以便不仅与入口网站内的服务,而且与计算设备、遍布环境下的设备本地的服务以及远程服务(一些服务可能是由其他动态服务入口提供的)互操作。这样显著提高了这种服务的利用率,而且对于用户容易做的事情,提高用户能力。

[0977] 用户可以在任何位置使用与任务计算兼容的任何设备访问这些服务。用户可以使用具有 web 浏览器的任何设备通过 TCC II 访问这些服务。

[0978] 用户可以在任何位置(在家、在办公室、在移动中等)使用与本地/遍布/(其他)远程服务组成在一起的这些服务。

[0979] 可以将社团或室内动态服务入口看作一种新式 ASP(应用程序服务提供者)。除了上述提到的个人版的服务外,它还可以提供社团、组织、雇员 web 服务。

[0980] 该系统还包括永久或可拆卸存储器,例如磁盘和光盘、RAM、ROM 等,它们上面可以

存储和分布本发明的方法和数据结构。还可以通过经过例如诸如因特网的网络下载,分销这些方法。

[0981] 根据详细说明,本发明的许多特征和优点将变得更加明显,因此,所附权利要求试图覆盖属于本发明实质范围的本发明的所有这些特征和优点。此外,由于本技术领域内的熟练技术人员容易设想出许多修改和变更,不要求本发明完全局限于所说明和描述的构建和运行过程,因此,所有修改和等效物均被再划分在本发明范围内。

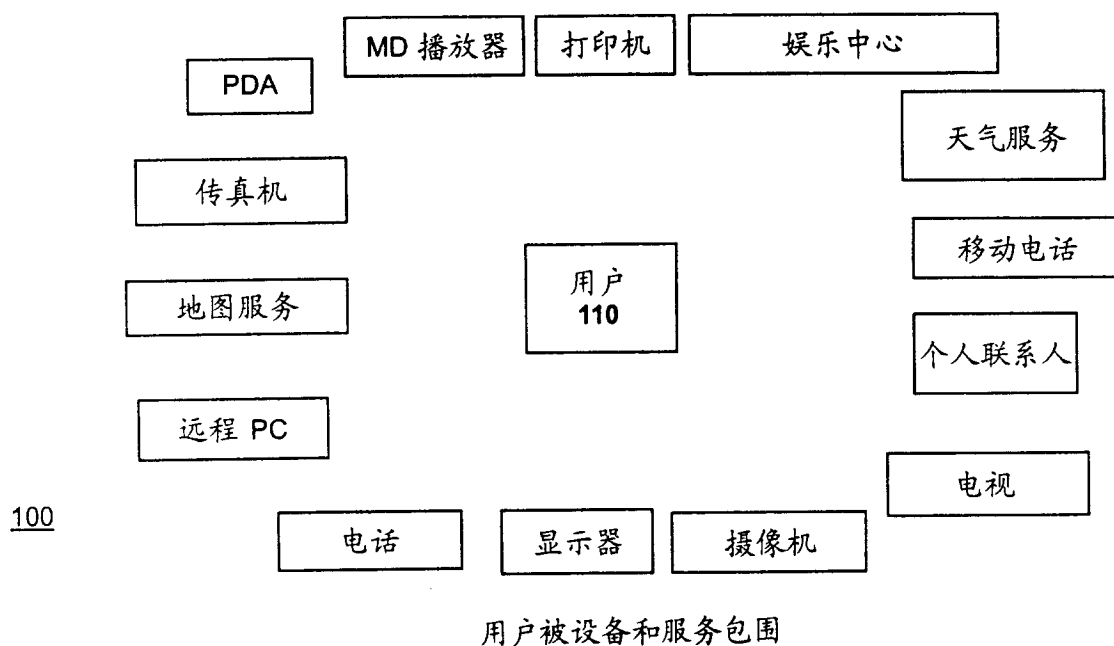
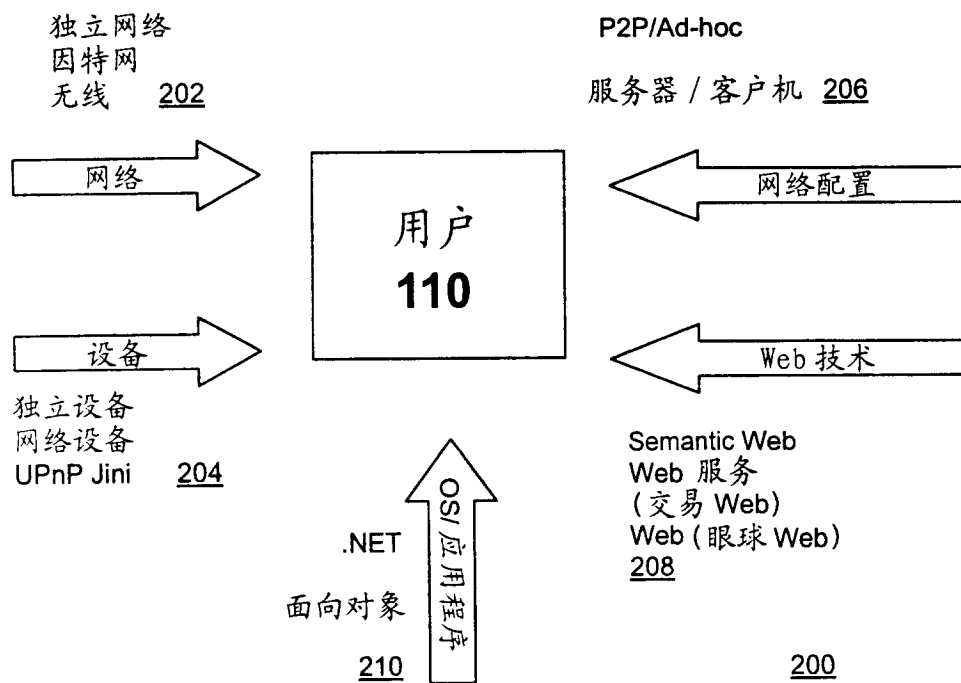
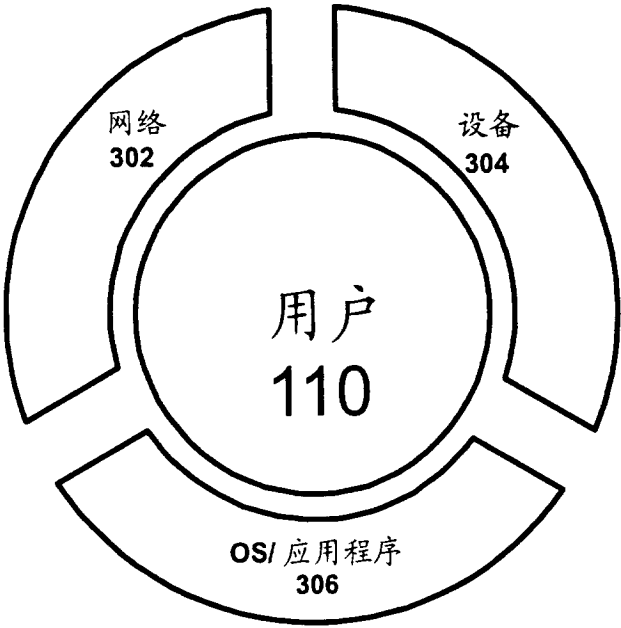


图 1



以用户为中心的视图 (I)

图 2



以用户为中心的视图 (II)

300

图 3

400

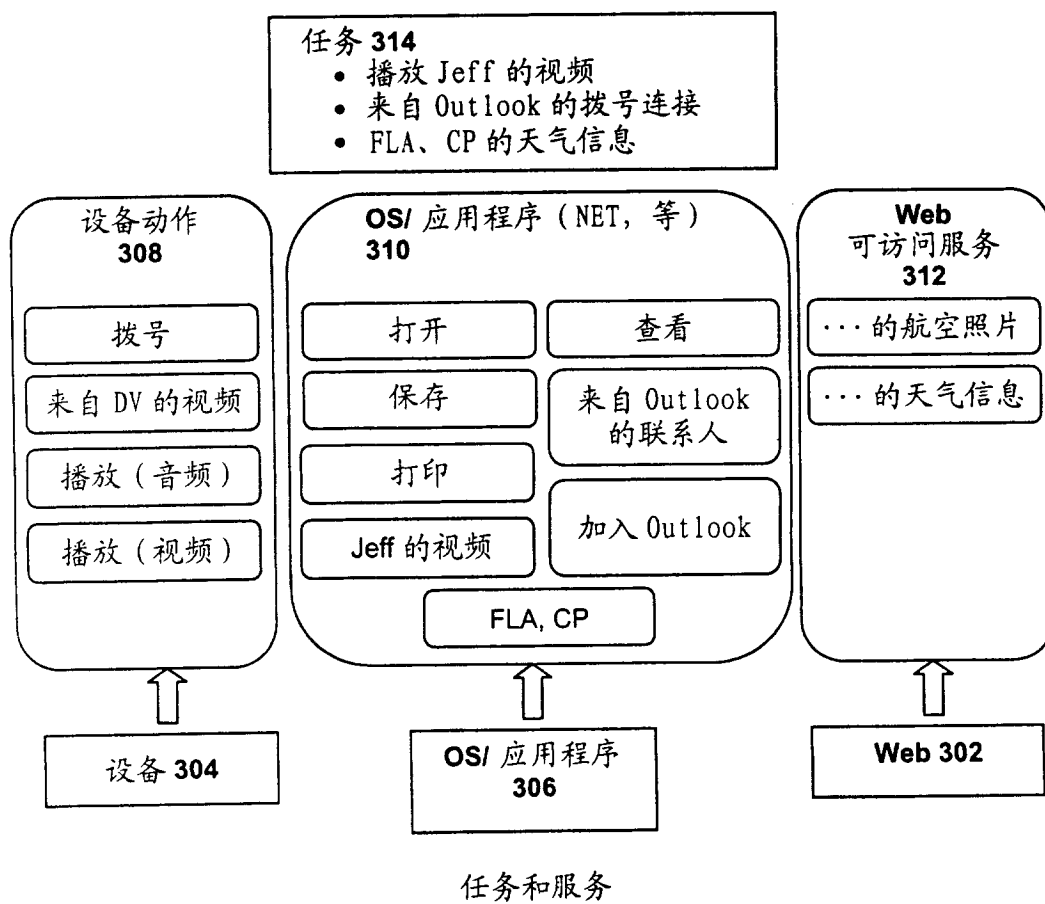
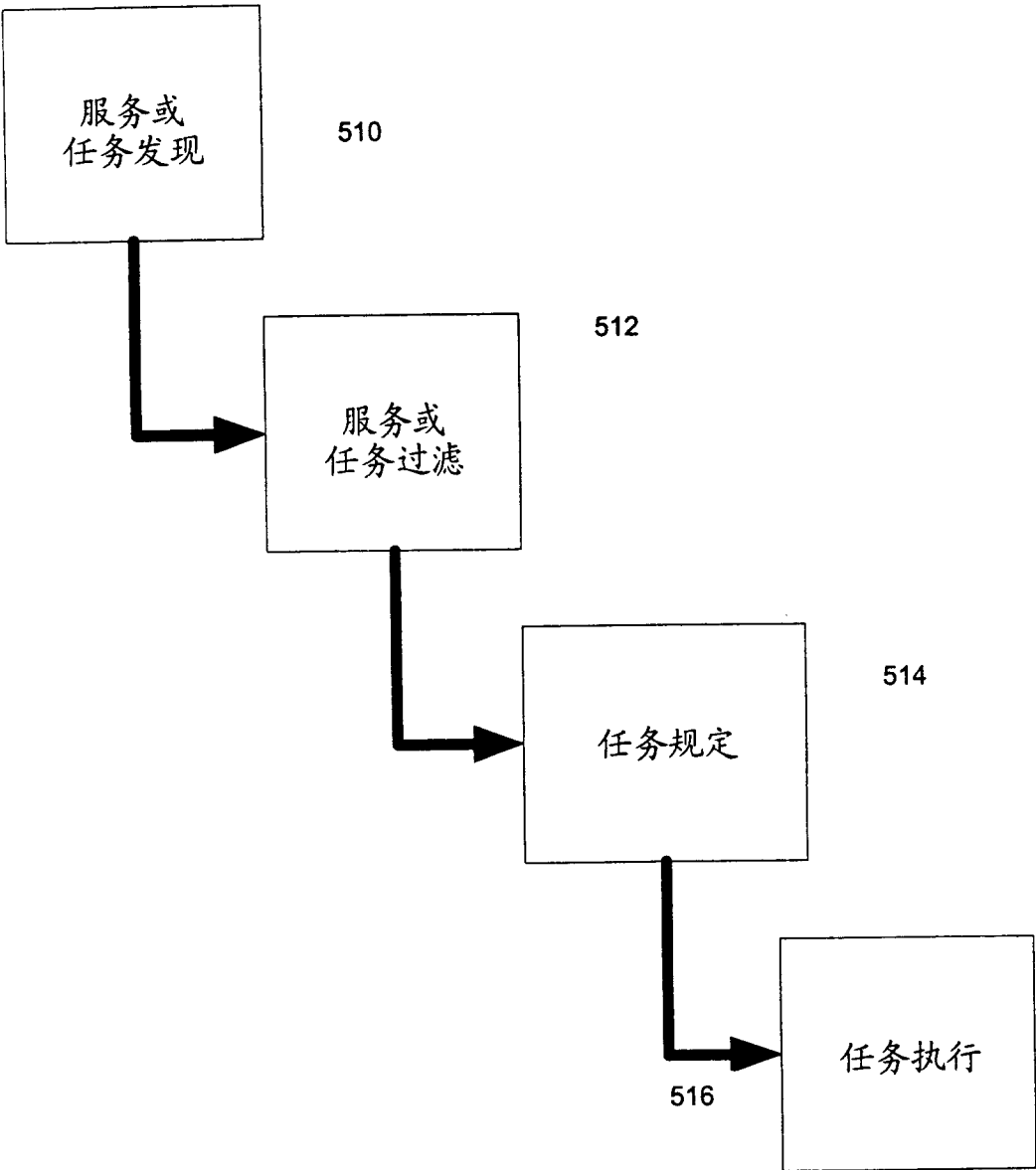


图 4



任务计算 workflow 变型 (I) 500

图 5

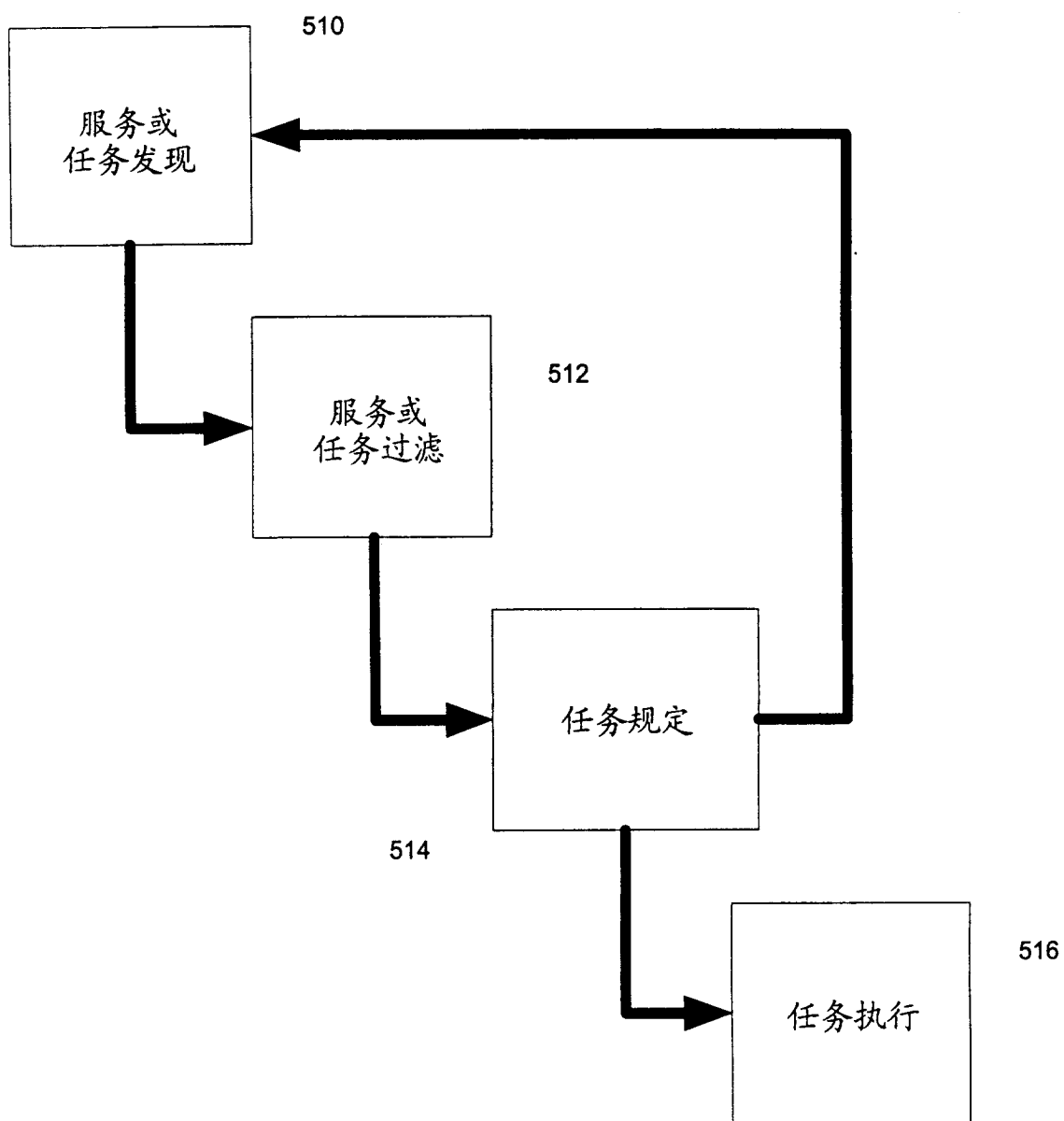
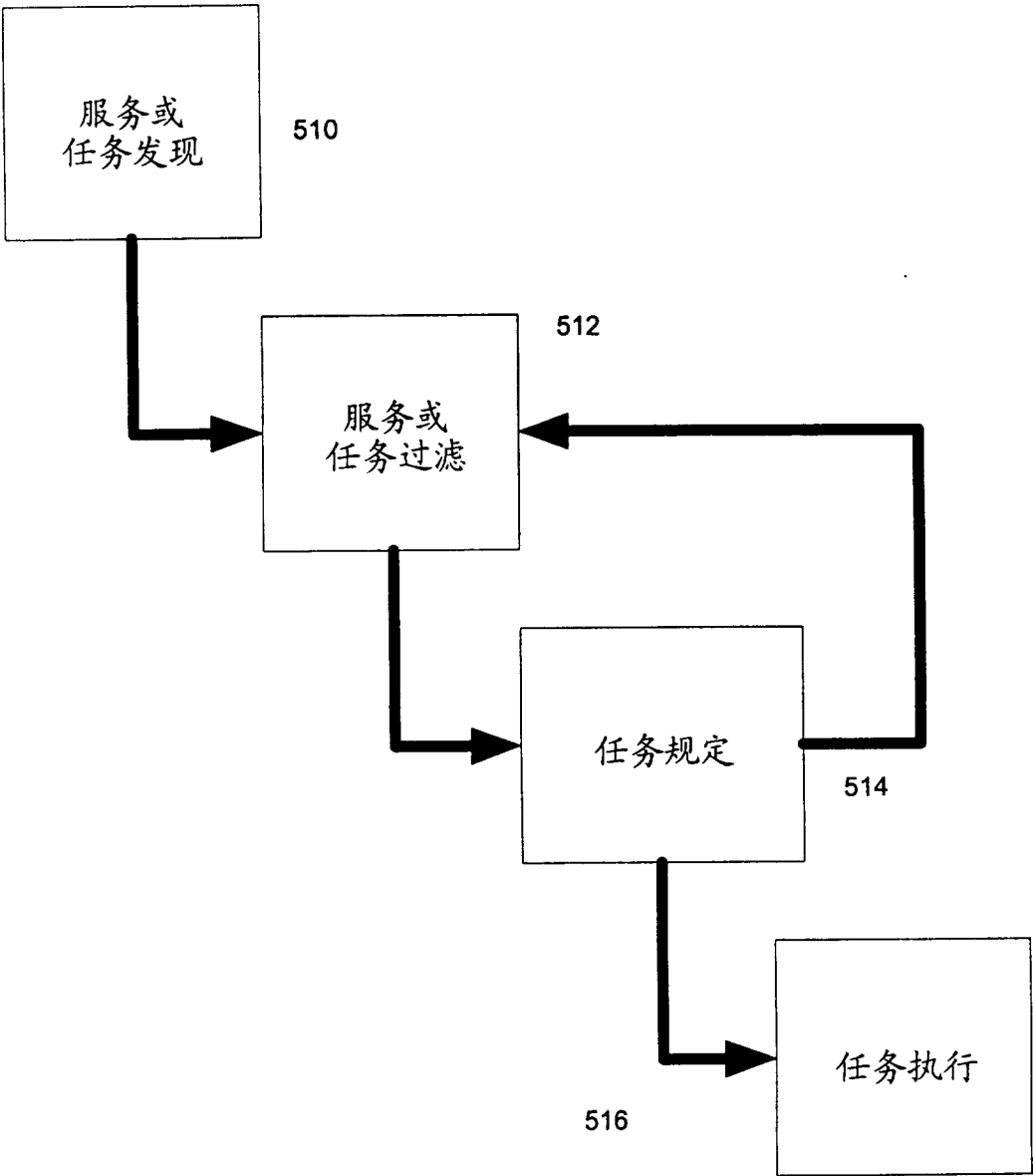
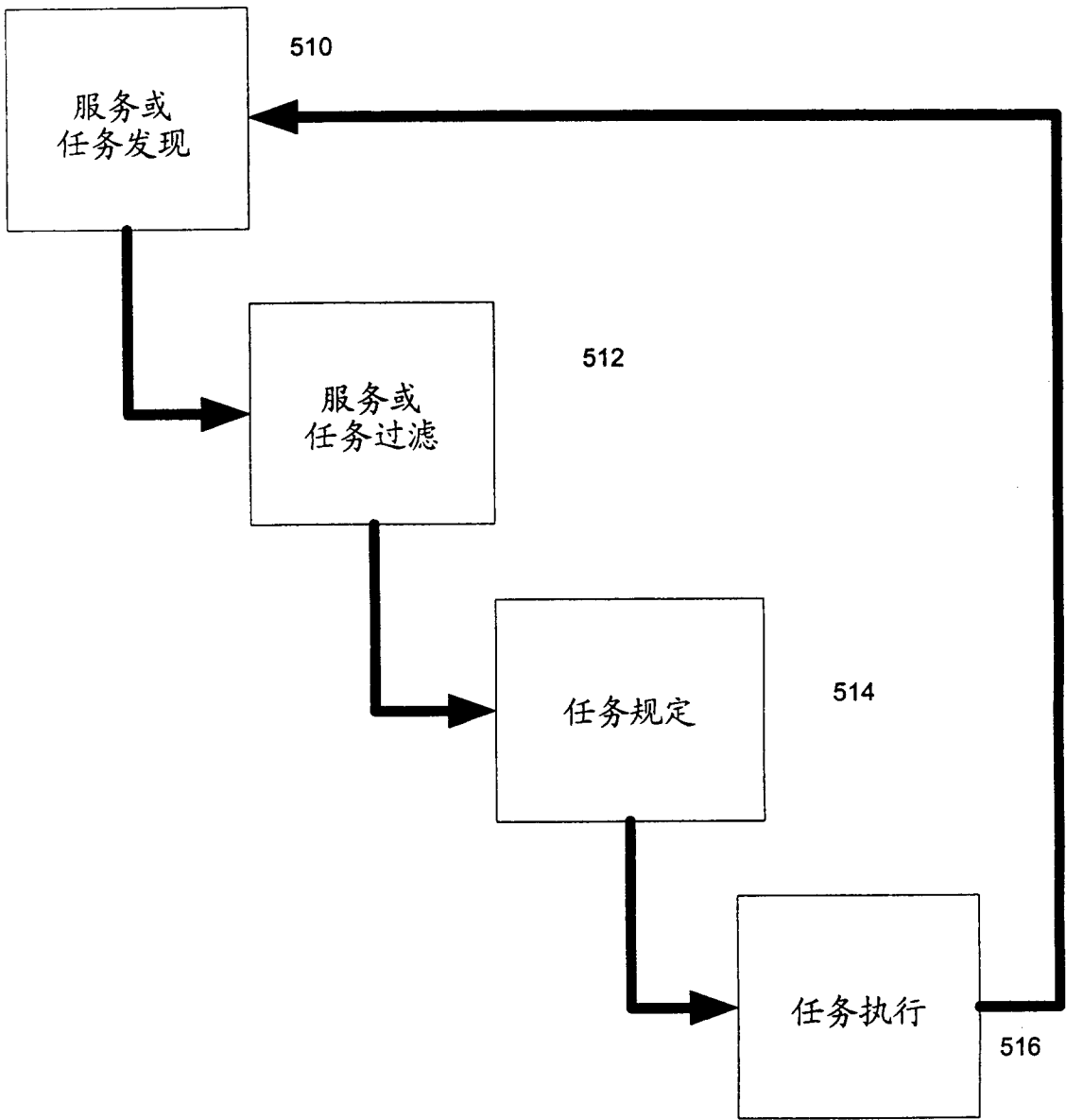
任务计算工作流变型 (II) 600

图 6



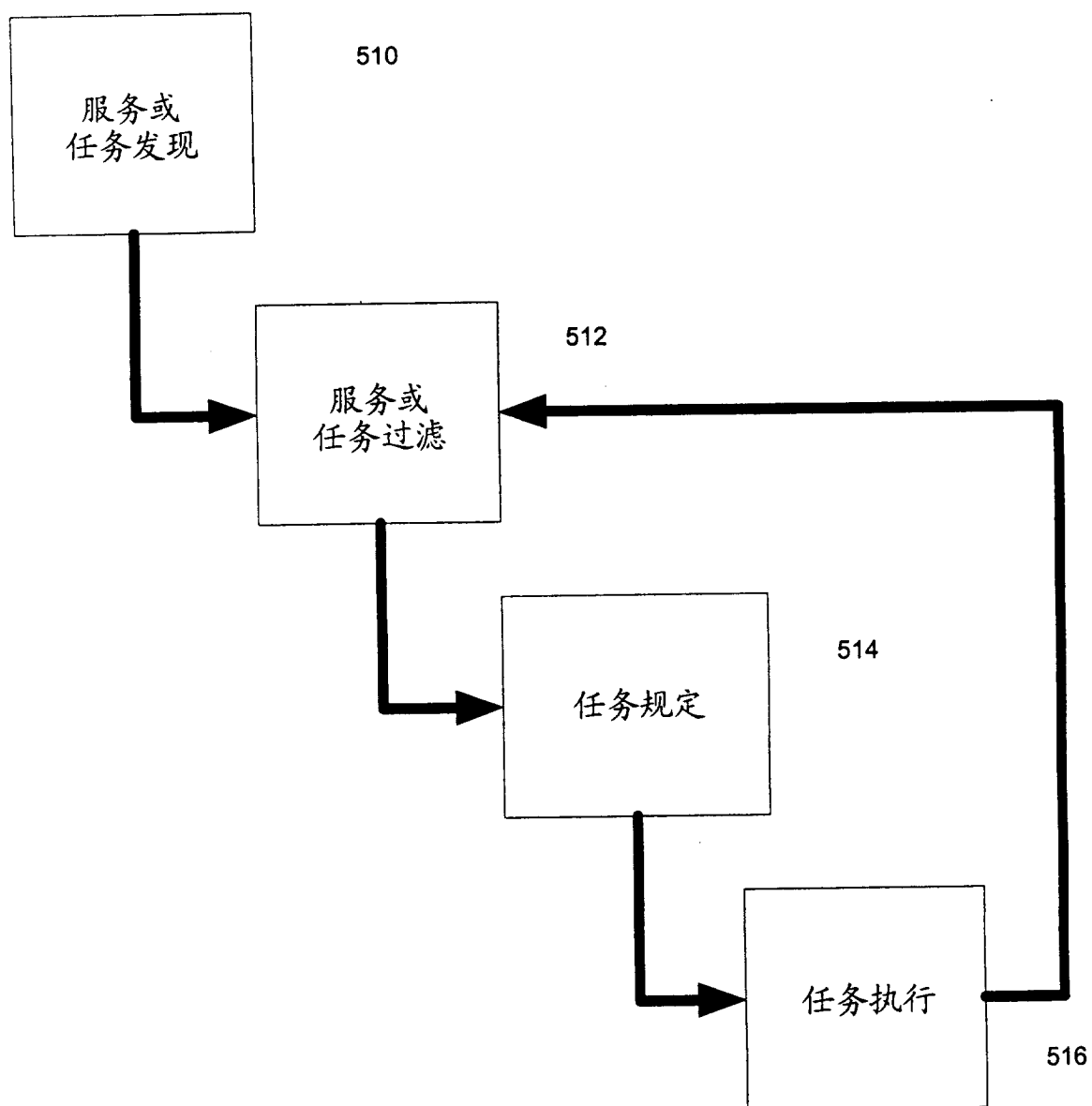
任务计算 workflow 变型 (III) 700

图 7



任务计算 workflow 变型 (IV) 800

图 8



任务计算 workflow 变型 (V) 900

图 9

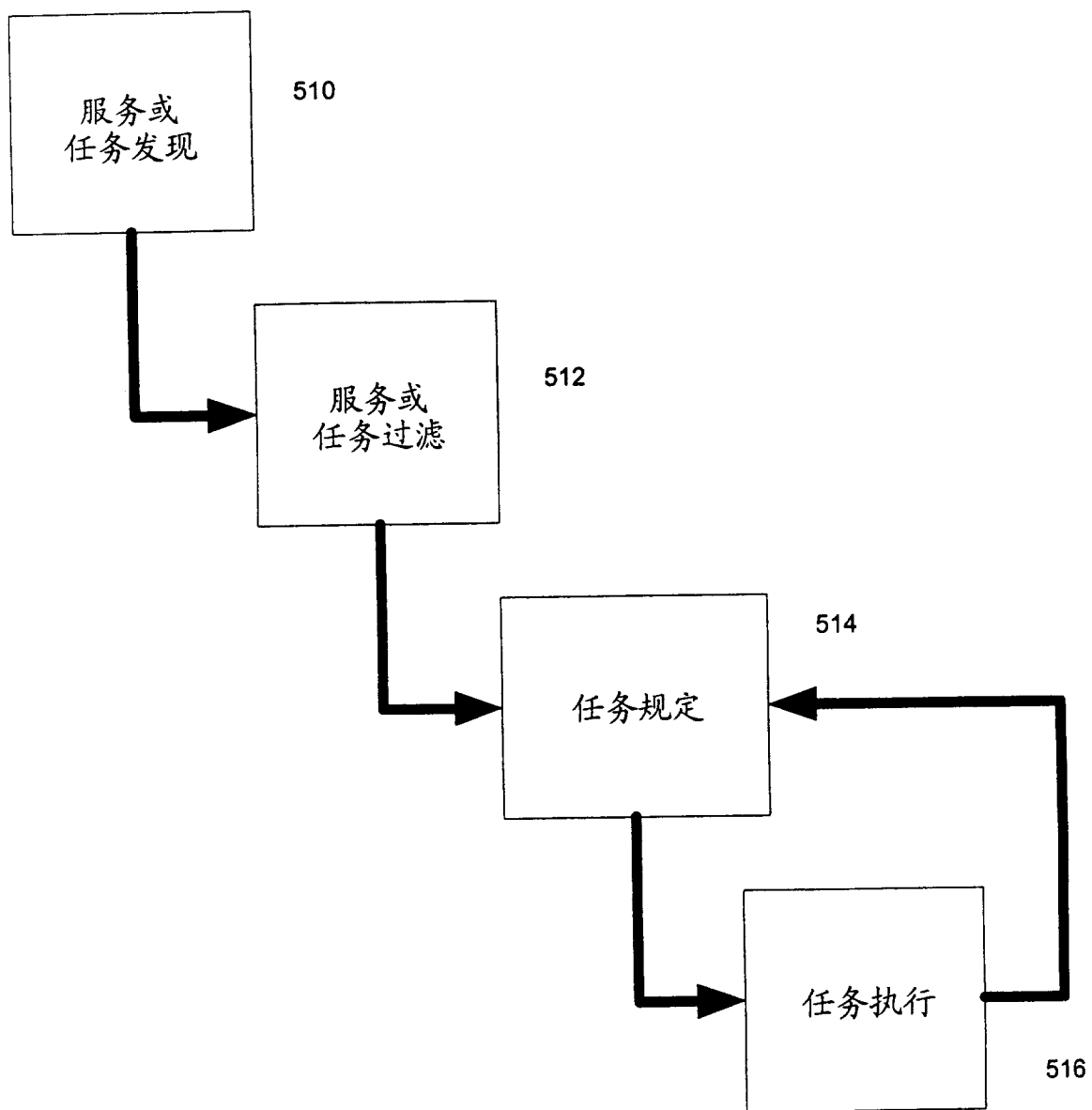
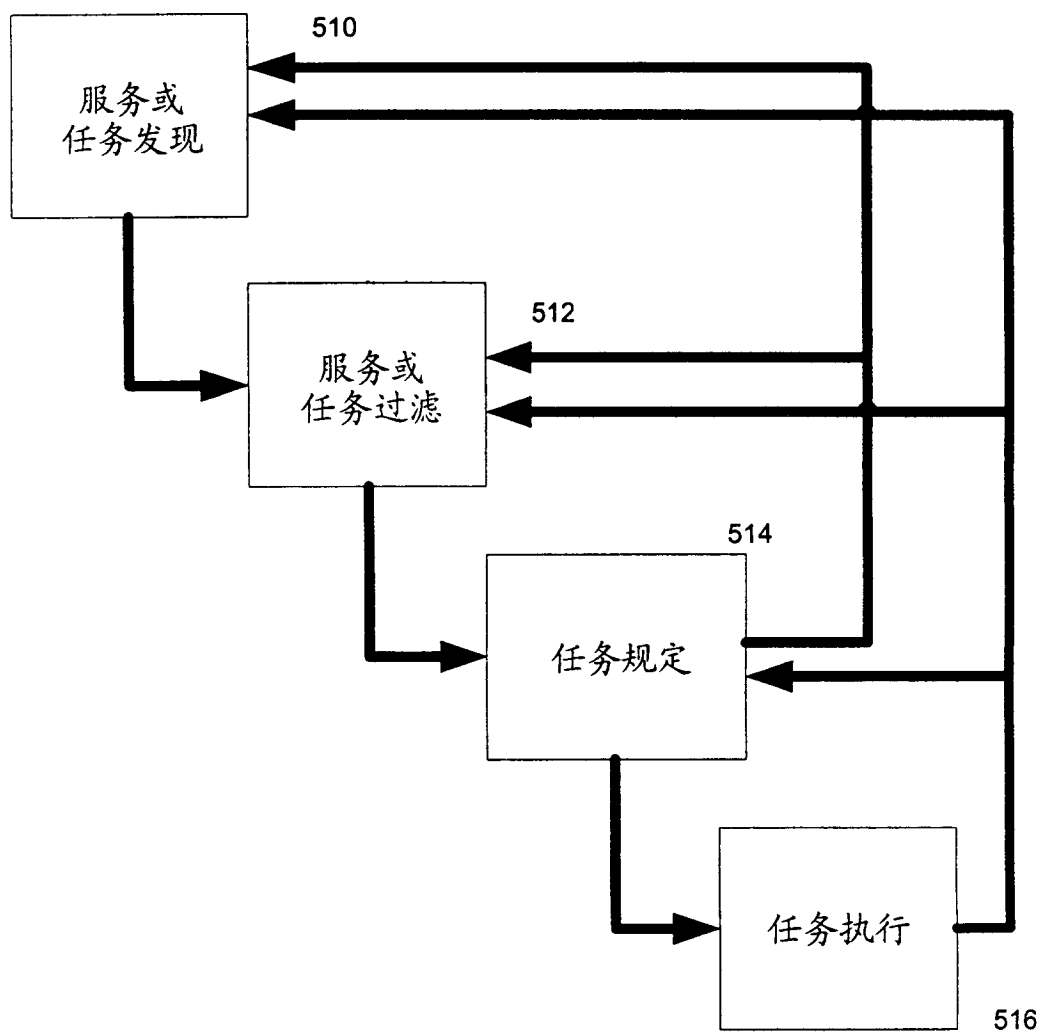
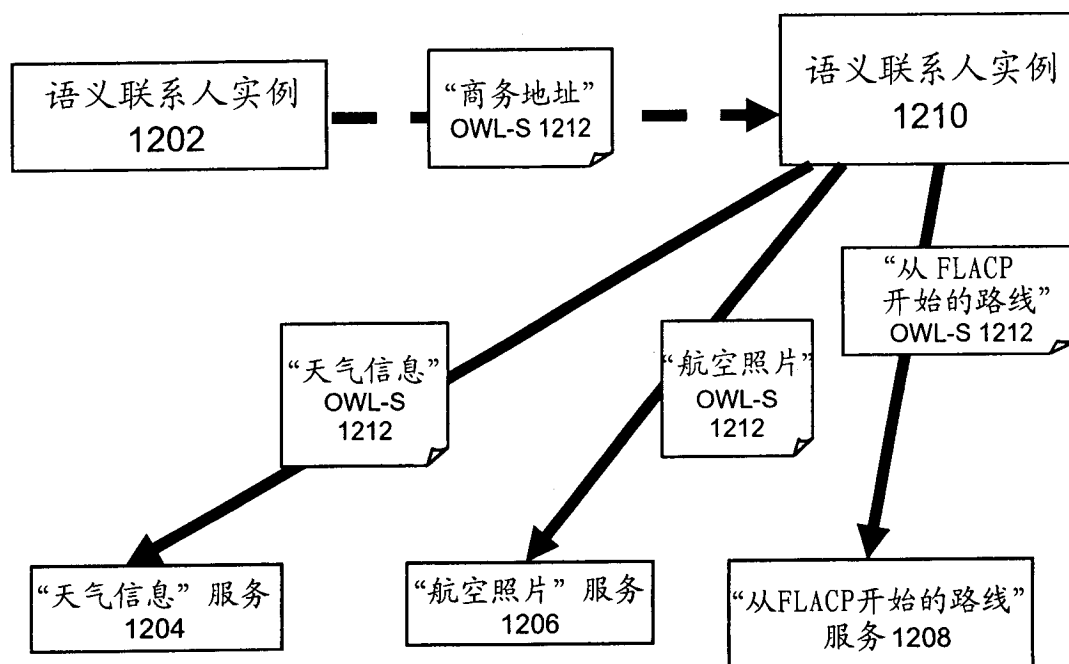
任务计算 workflow 变型 1000

图 10



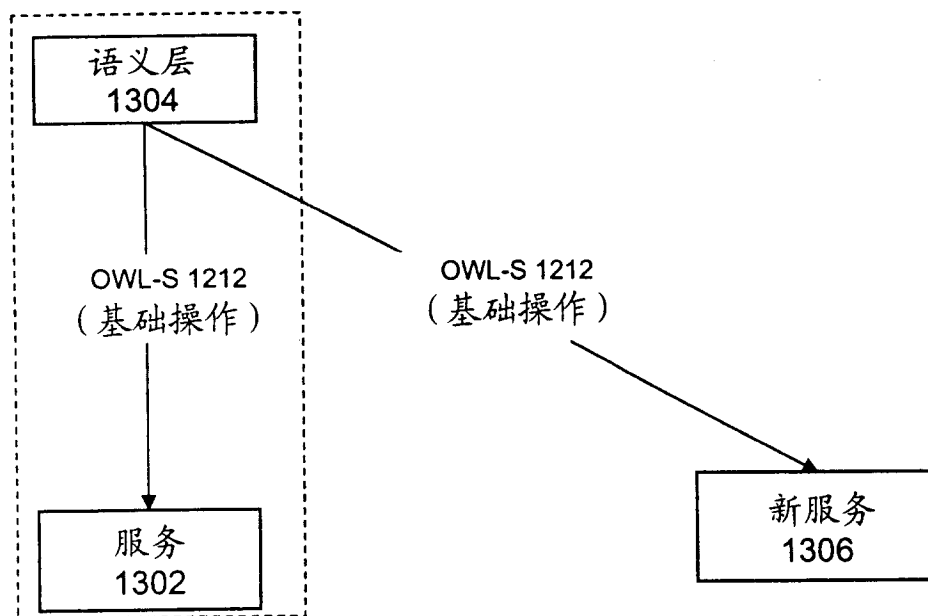
复合任务计算 workflow 1100

图 11



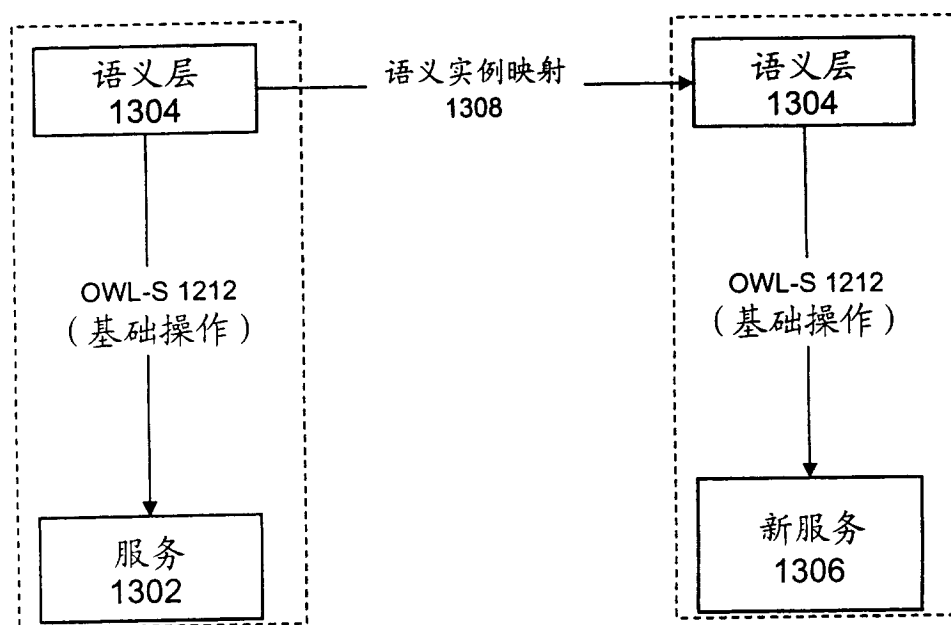
语义服务实例映射和基础 1200

图 12



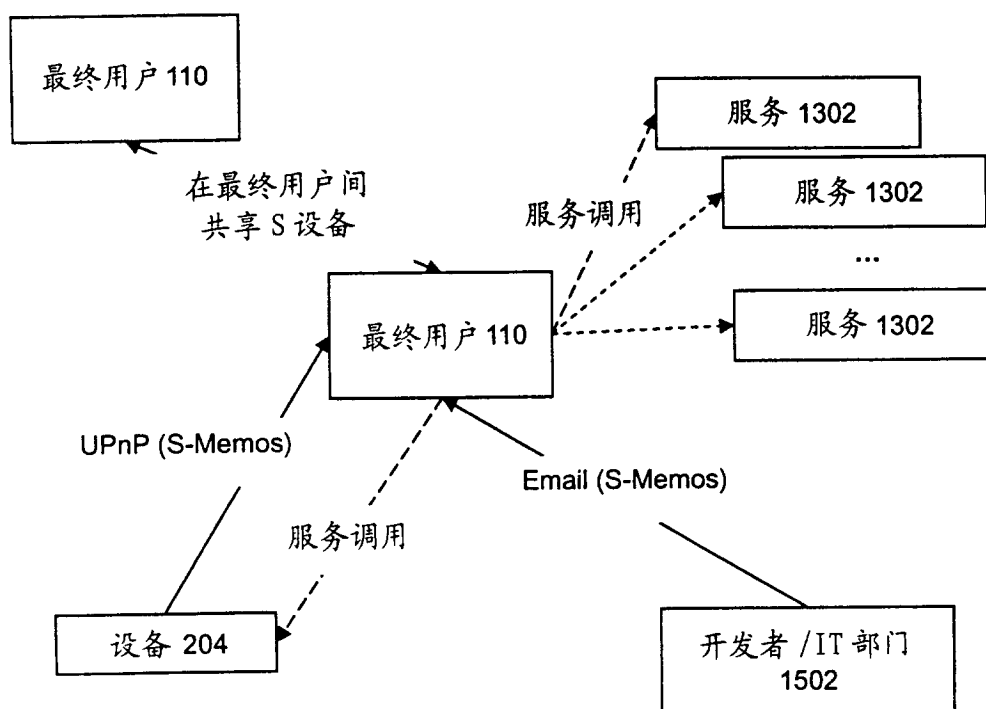
并入新服务 (I) 1300

图 13



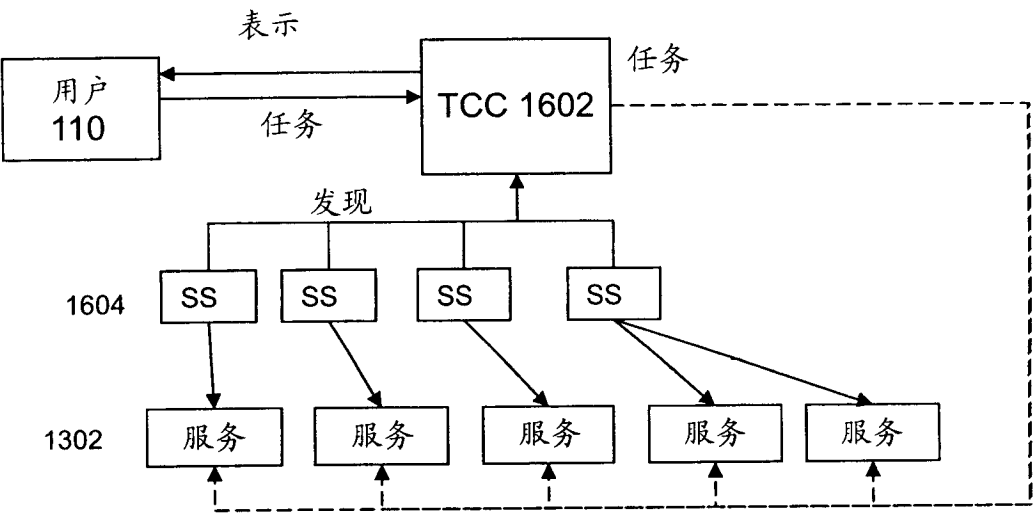
并入新服务 (II) 1400

图 14



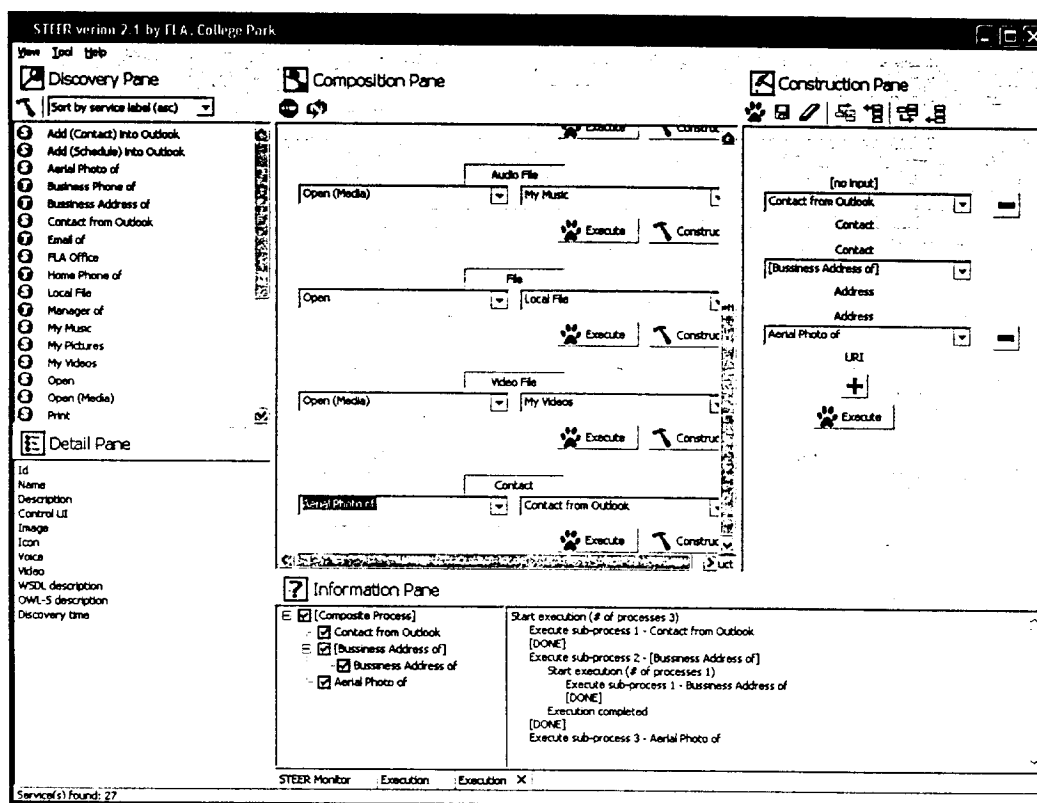
基于 S 备注的最终用户系统开发 / 部署 **1500**

图 15



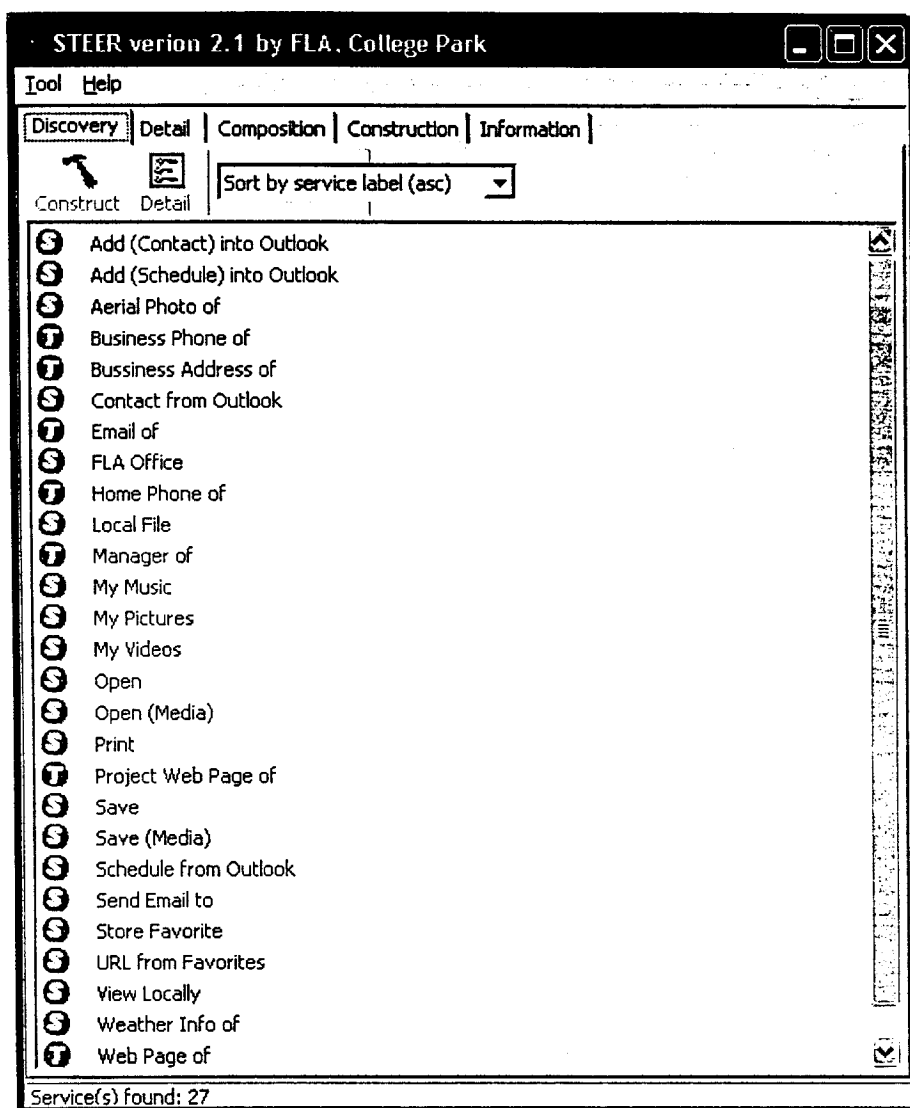
TCE 实施例的一般体系 1600

图 16



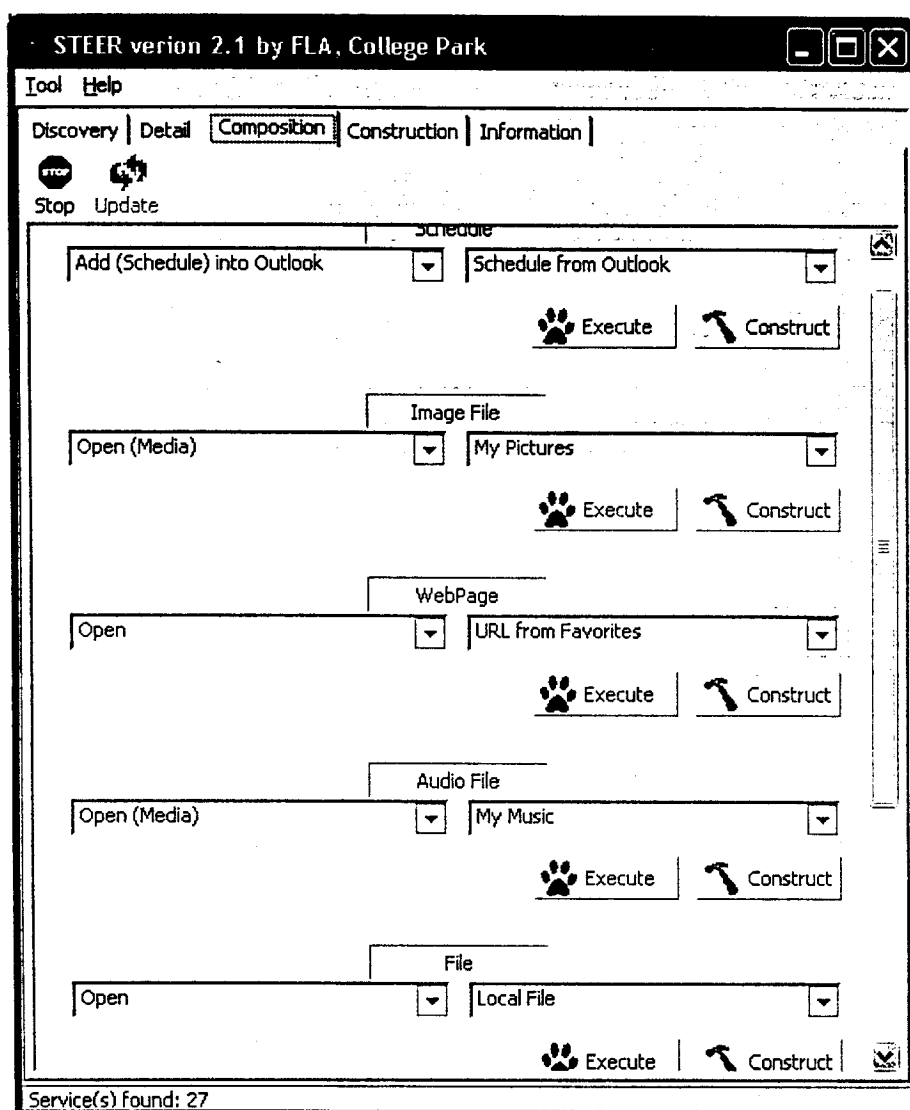
用户界面完全版本 1700

图 17



用户界面选项卡版本 (I) 1800

图 18



用户界面选项卡版本 (II) 1900

图 19

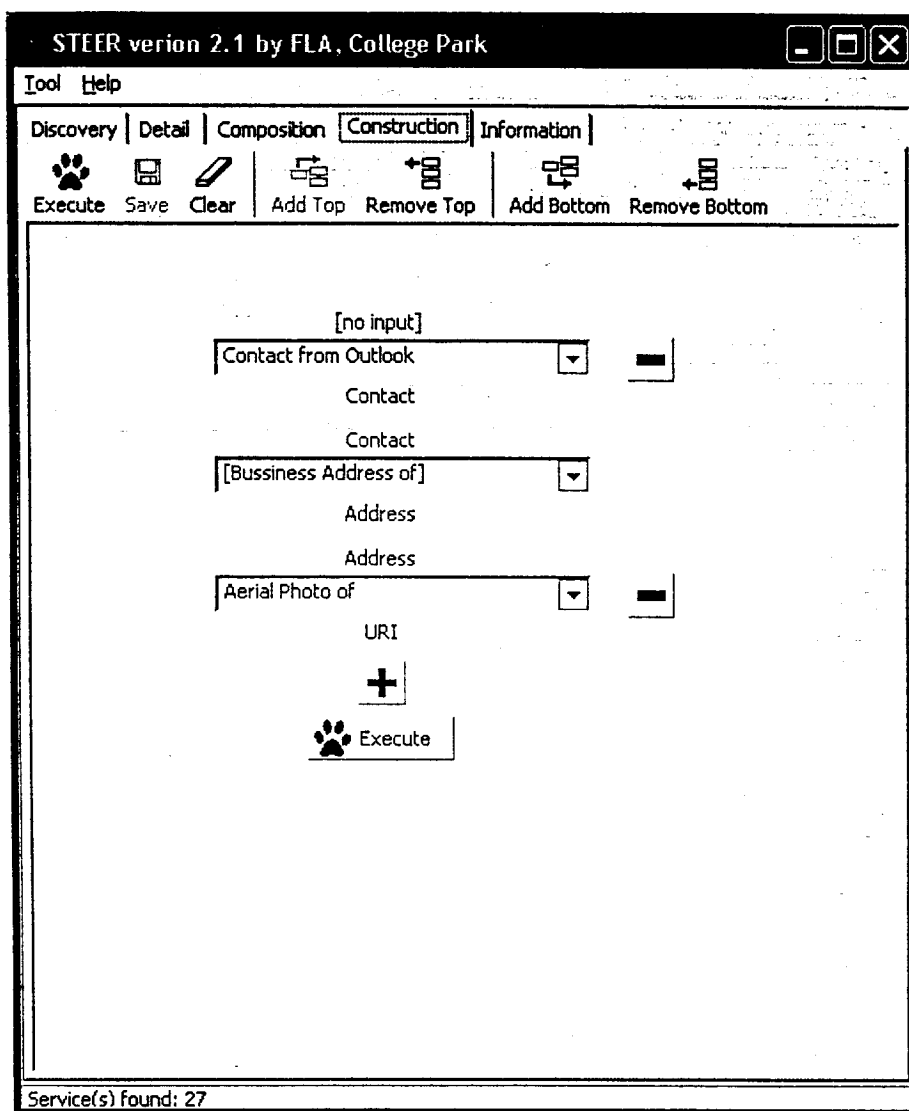
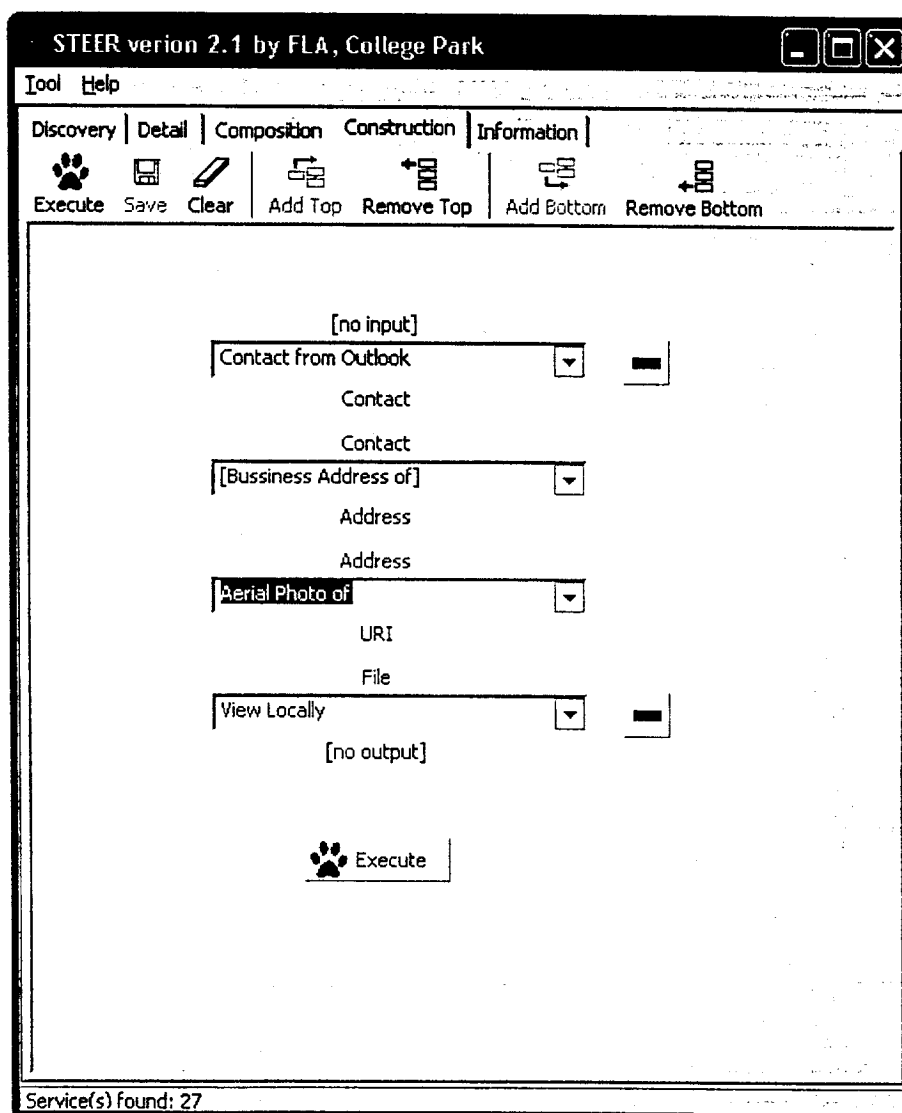
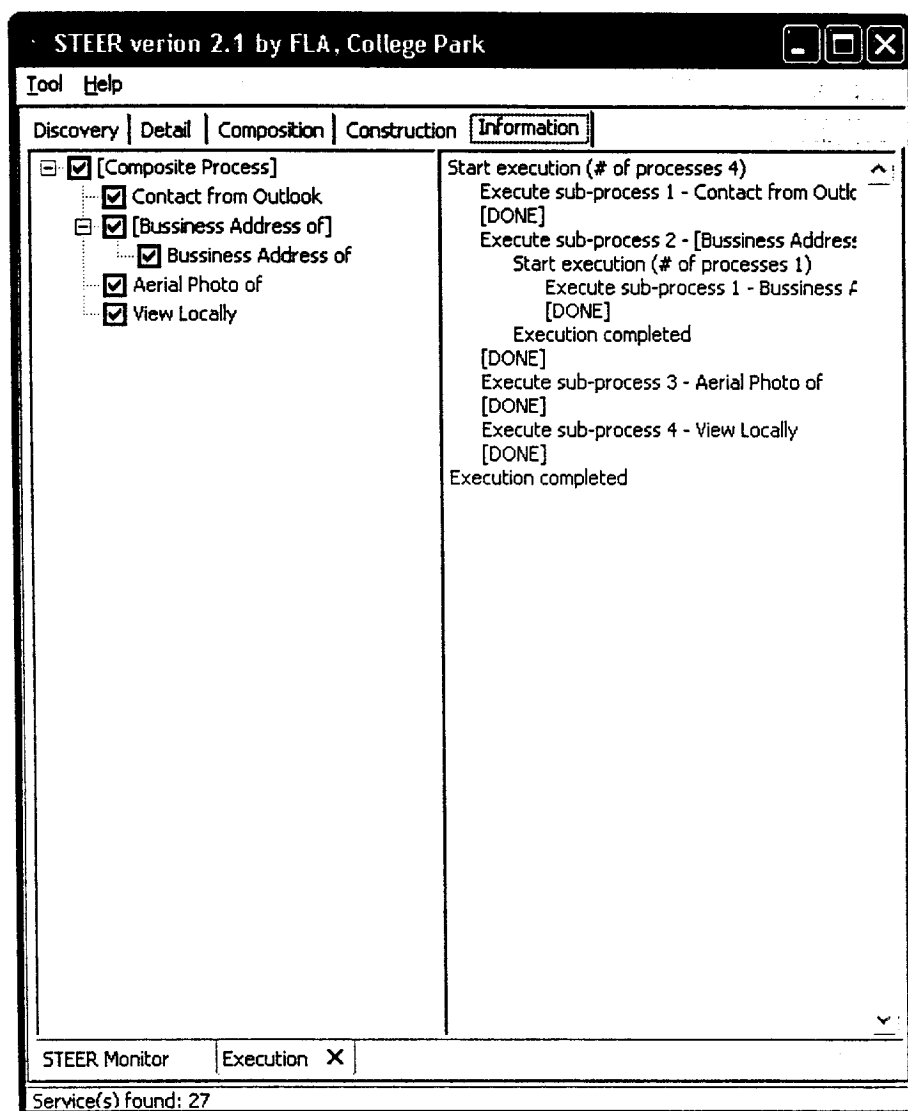
用户界面选项卡版本 (III) 2000

图 20



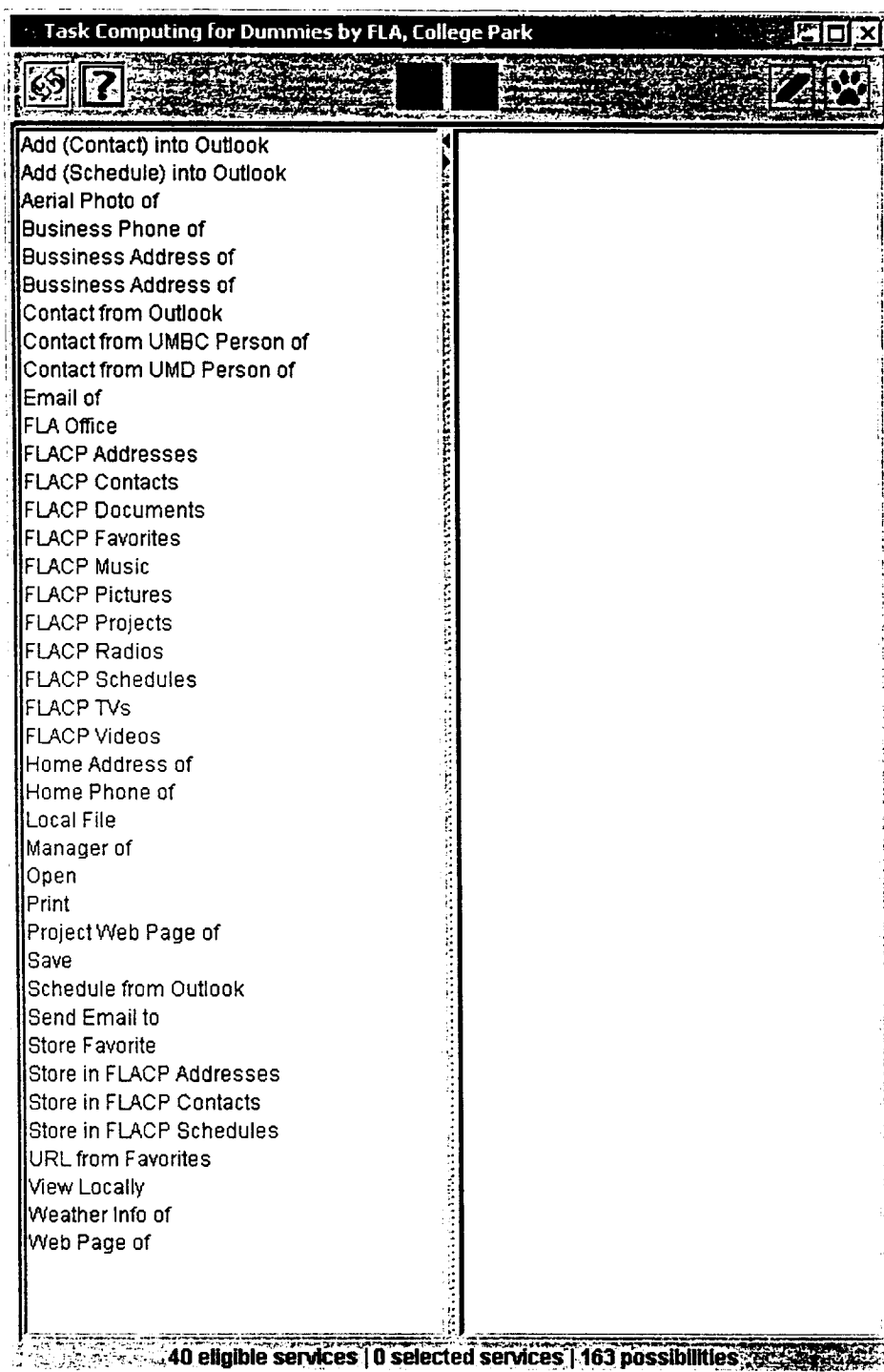
用户界面选项卡版本 (IV) 2100

图 21



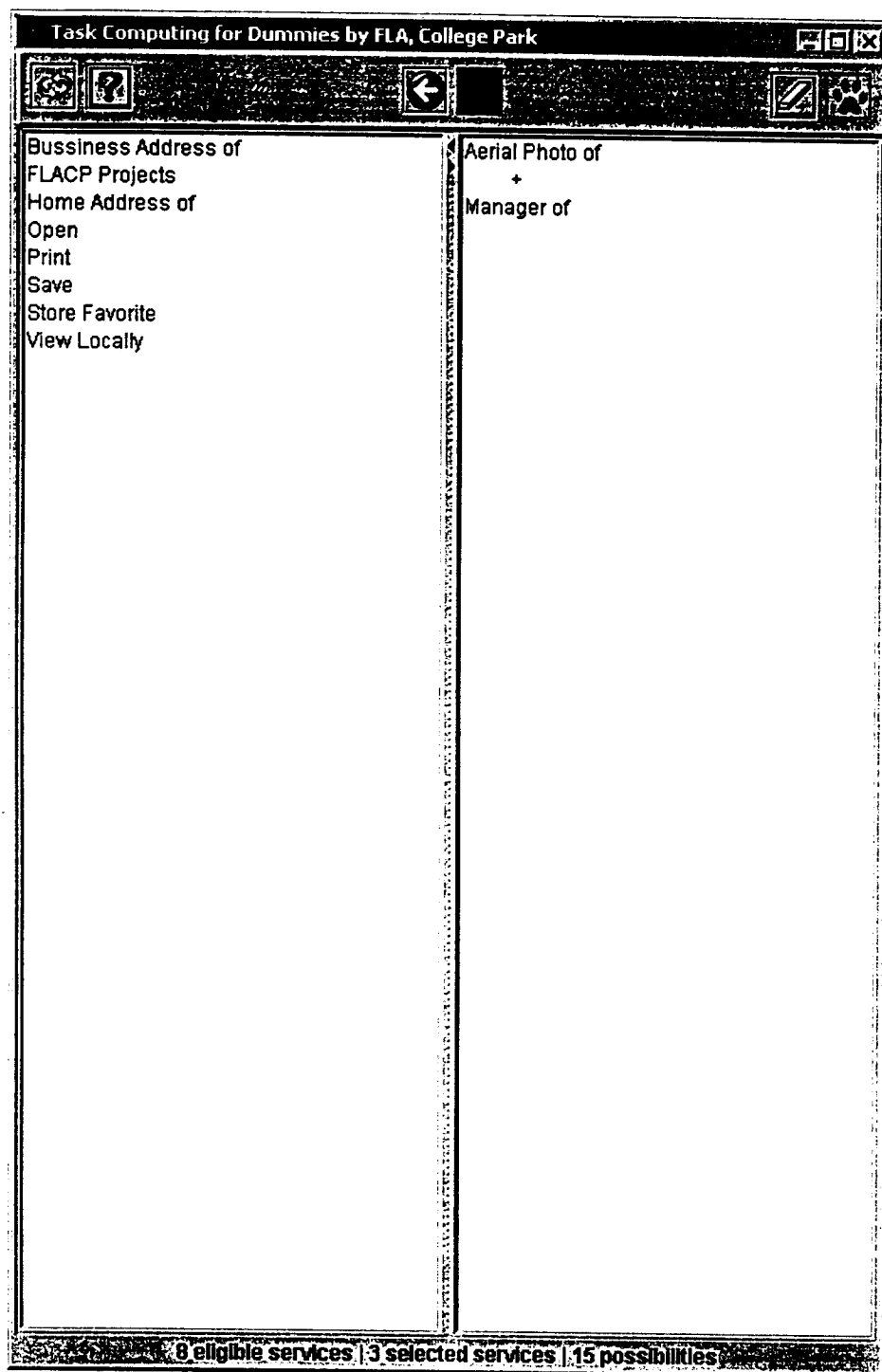
用户界面选项卡版本(V) 2200

图 22



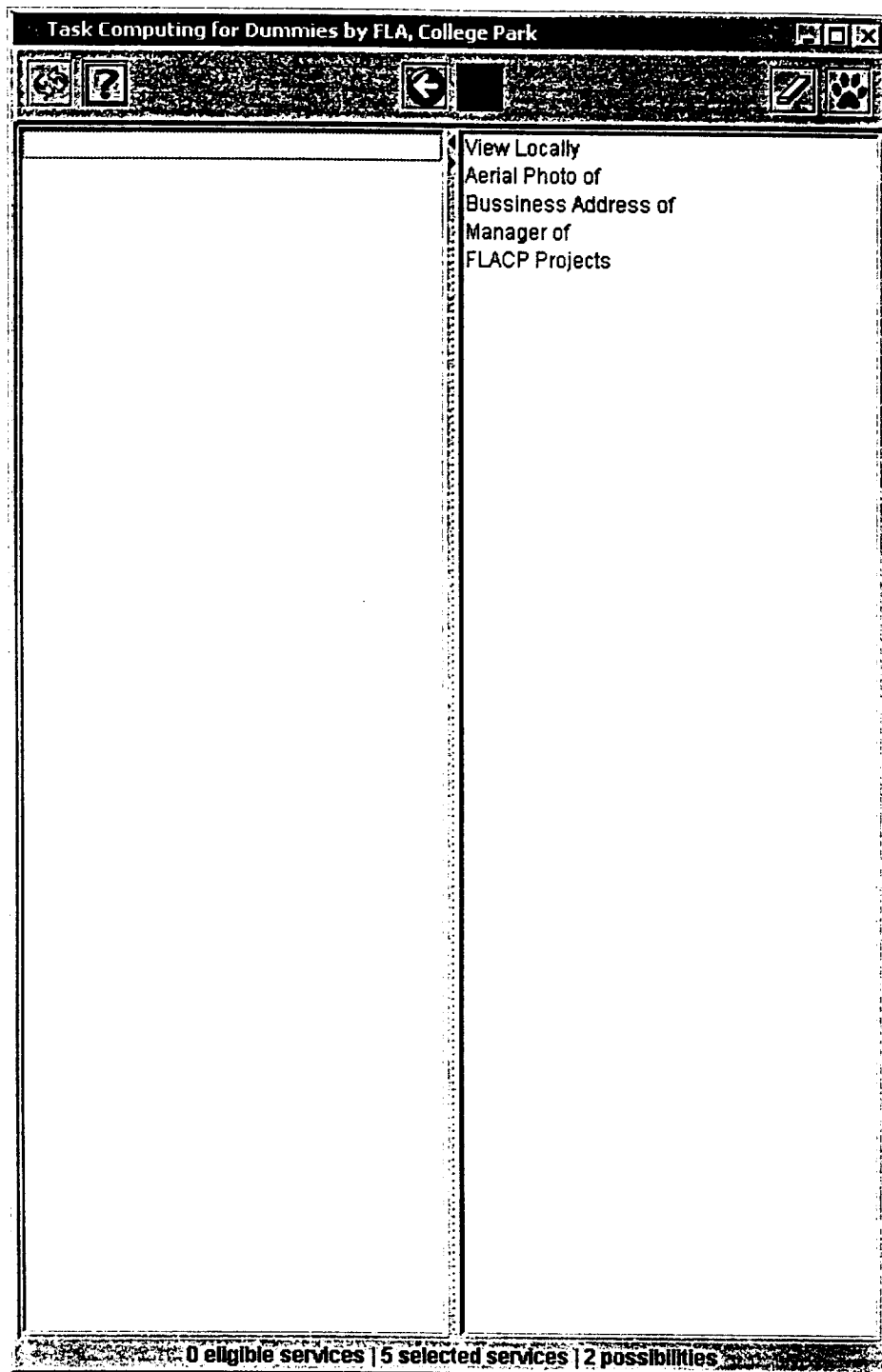
用户界面流版本 (I) 2300

图 23



用户界面流版本 (II) 2400

图 24



用户界面流版本 (III) 2500

图 25

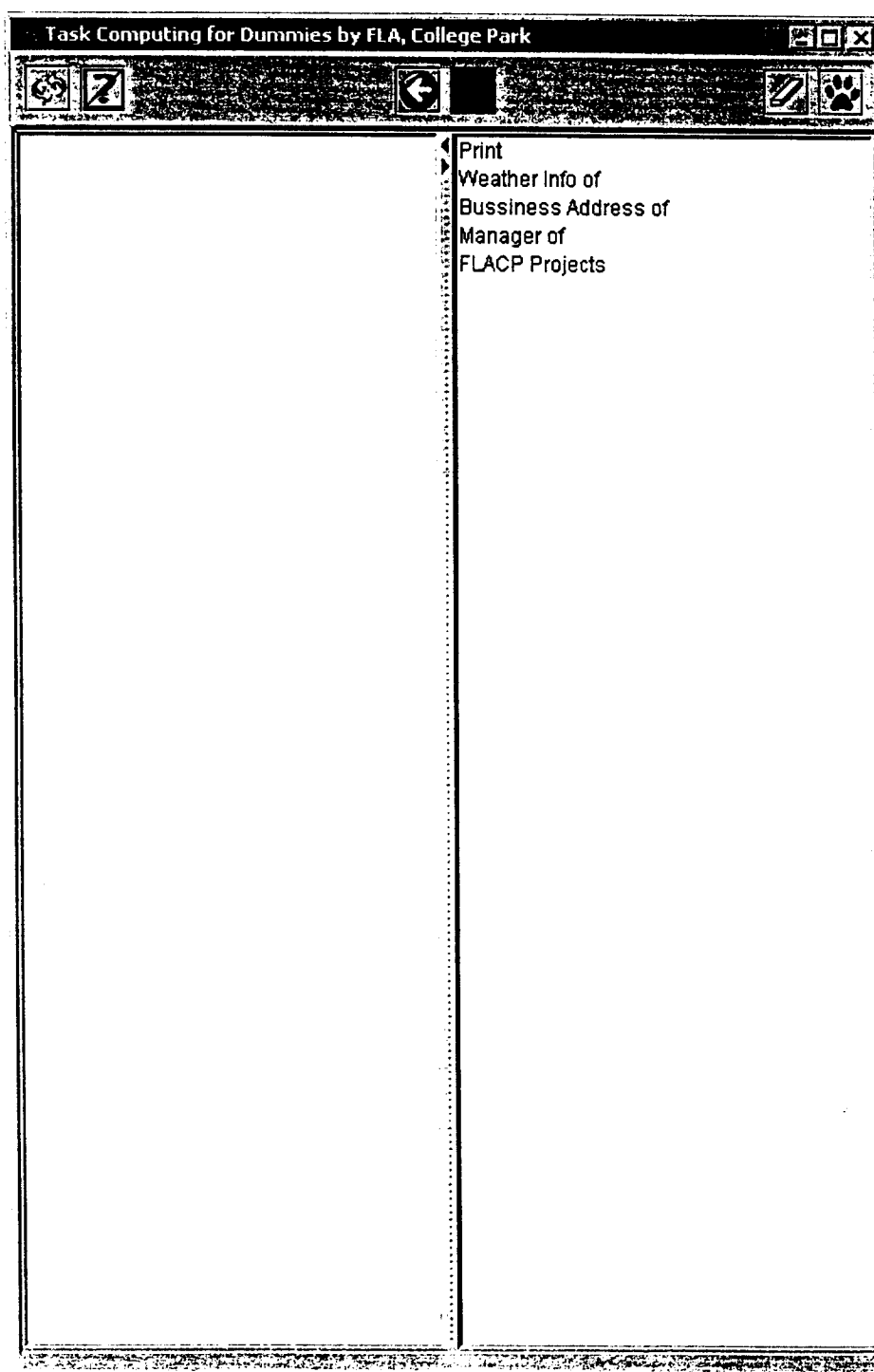
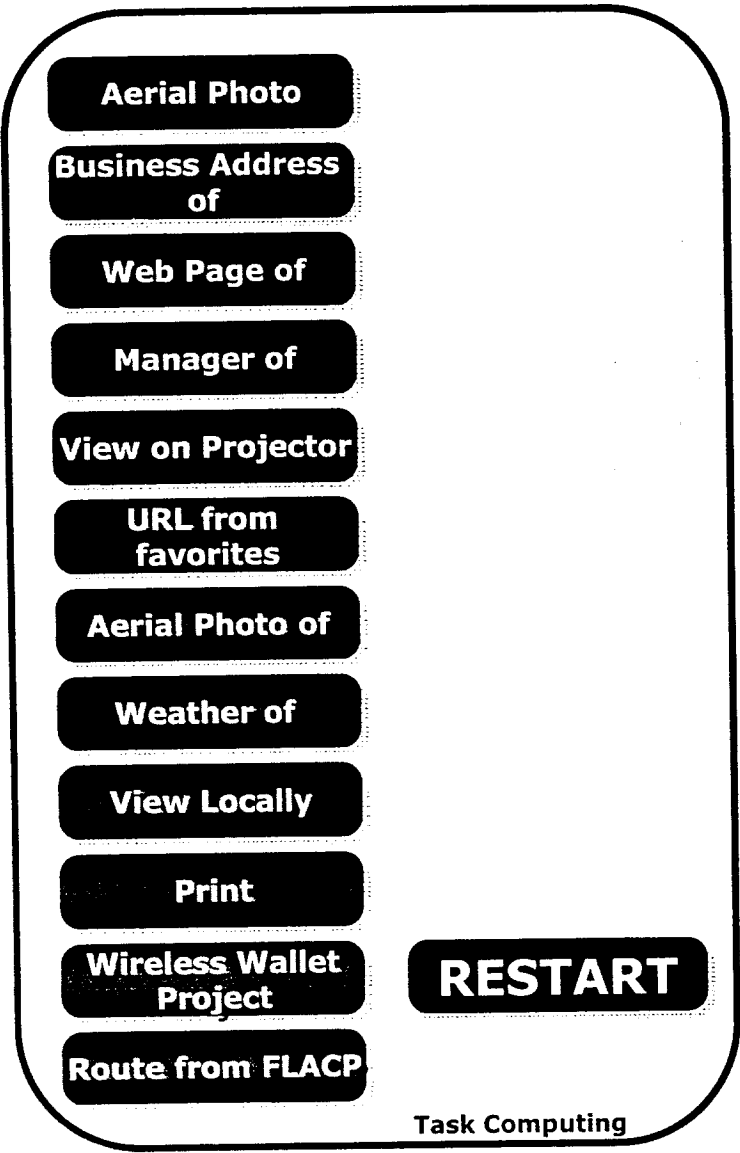
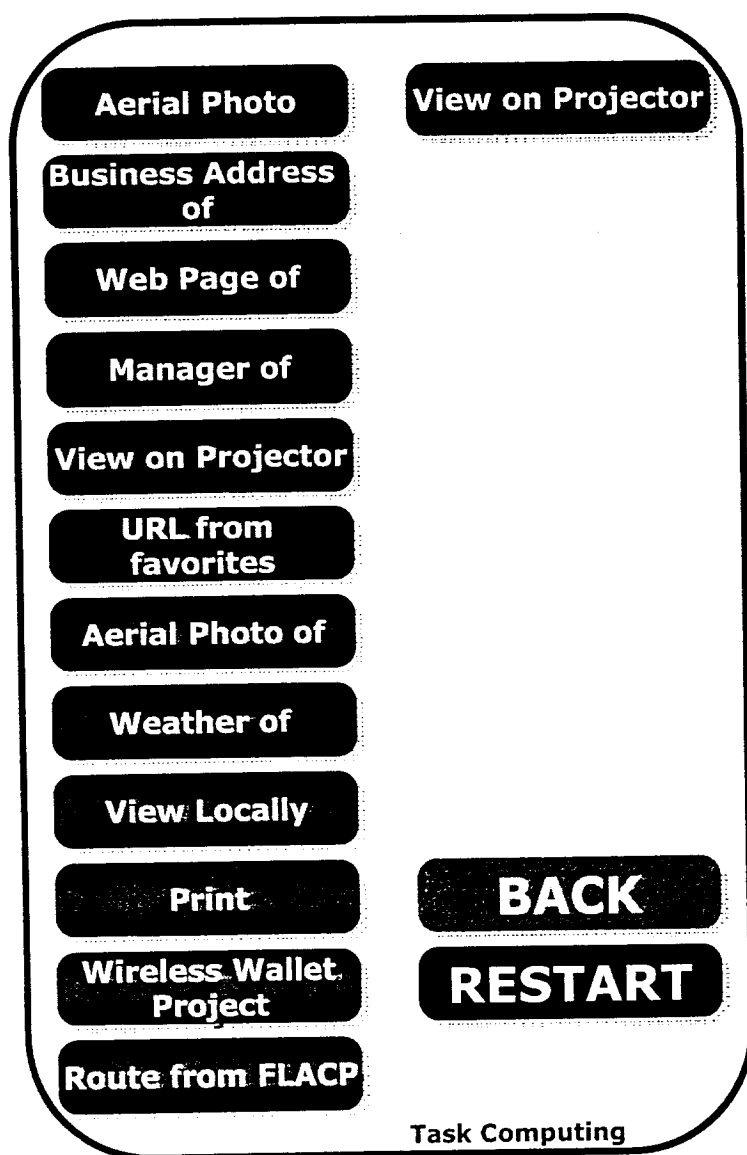
用户界面流版本 (IV) 2600

图 26



启动时 2700

图 27



在用户选择后 2800

图 28



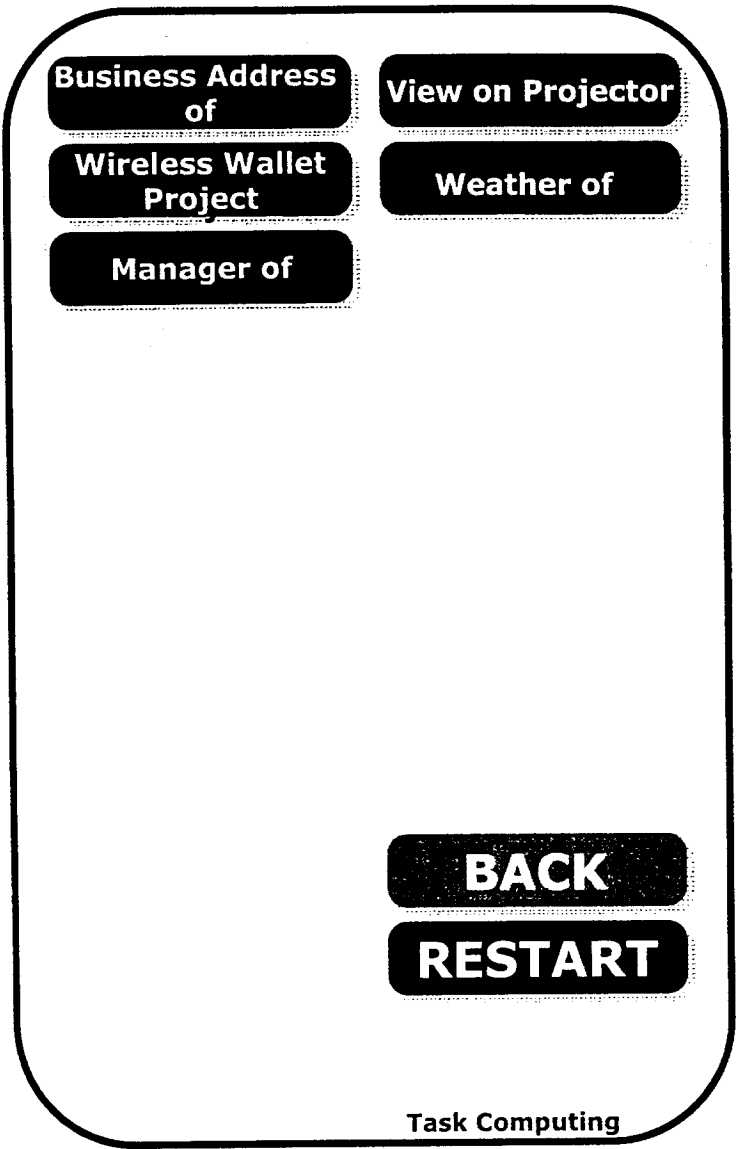
在刷新左窗格后 2900

图 29



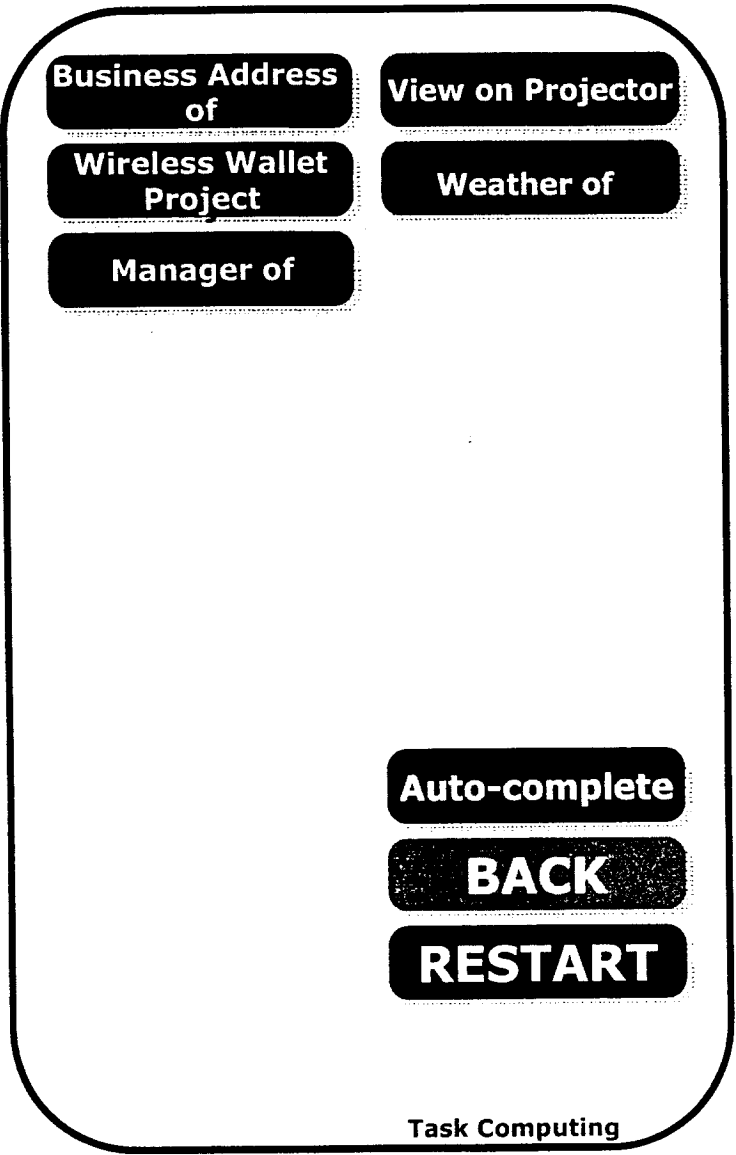
在用户选择后 3000

图 30



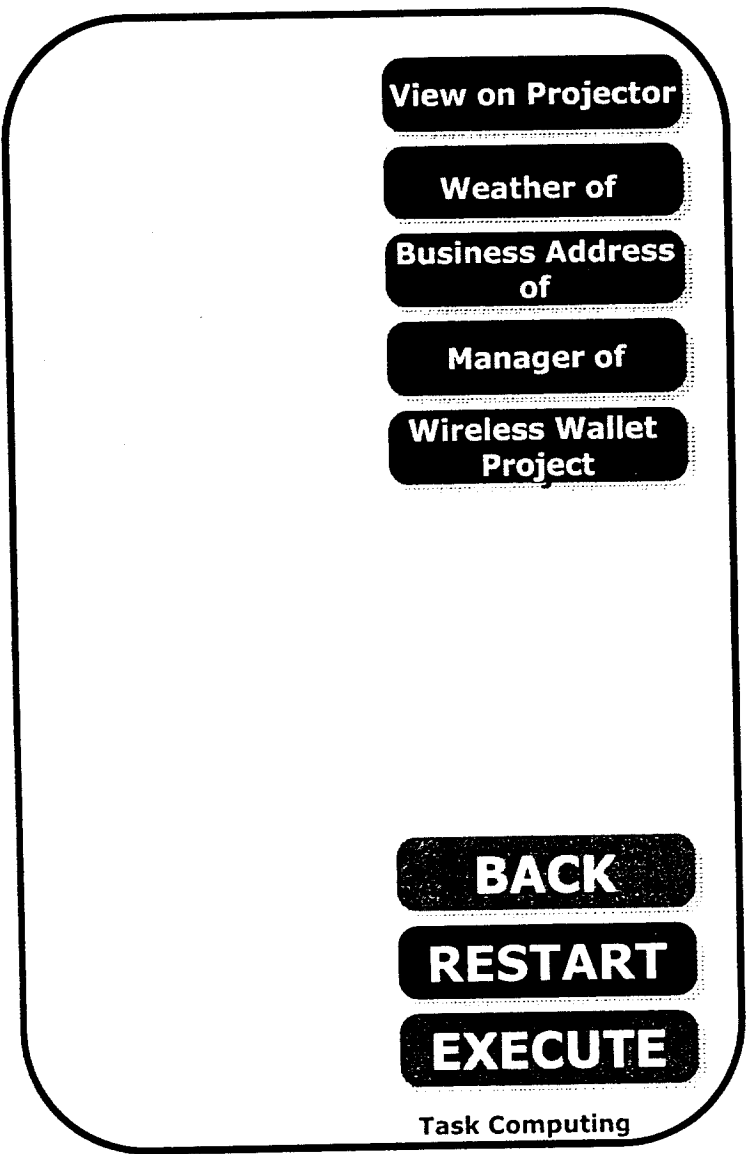
在刷新左窗格后 3100

图 31



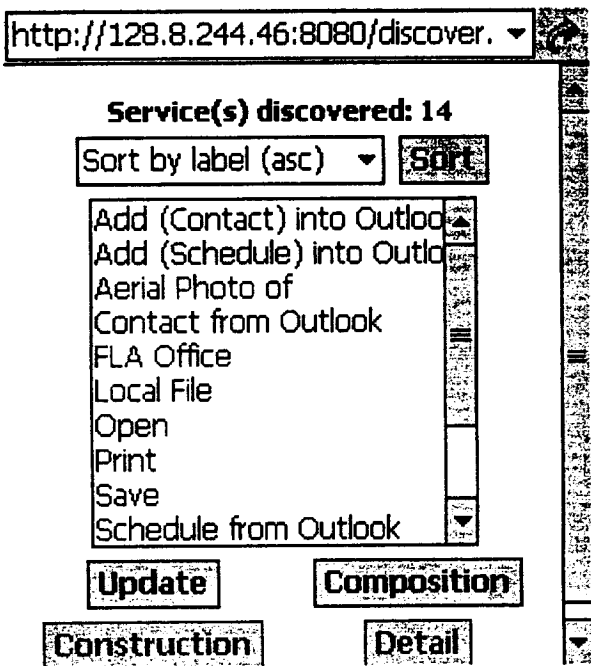
自动完成现在是可能的 3200

图 32



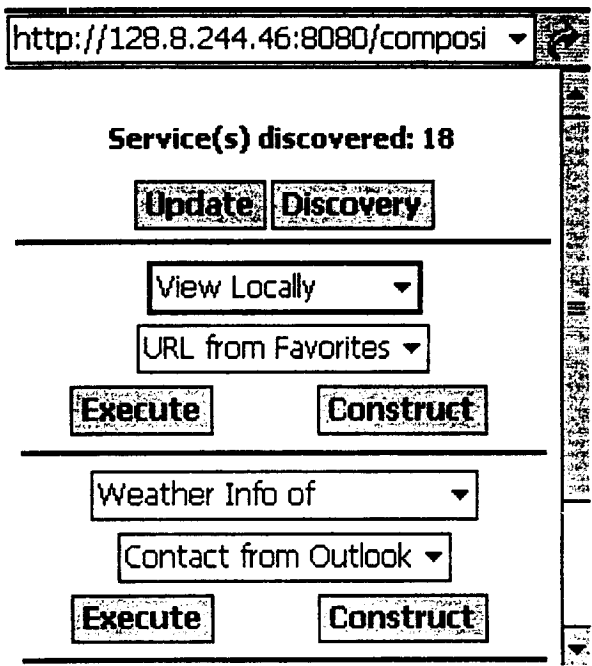
在用户选择自动完成后，或者如果自动完成是可能的，其可以自动进行 3300

图 33



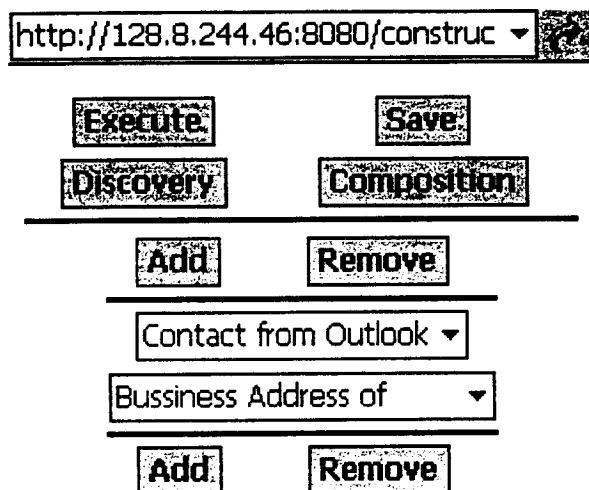
用户界面 Web 版本 (I) 3400

图 34



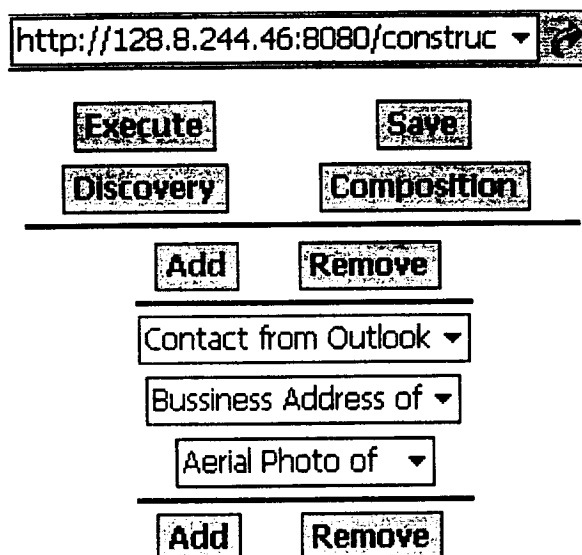
用户界面 Web 版本 (II) 3500

图 35



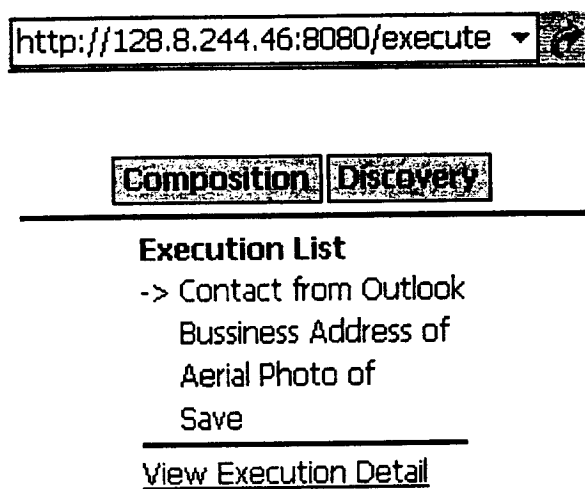
用户界面 Web 版本 (III) 3600

图 36



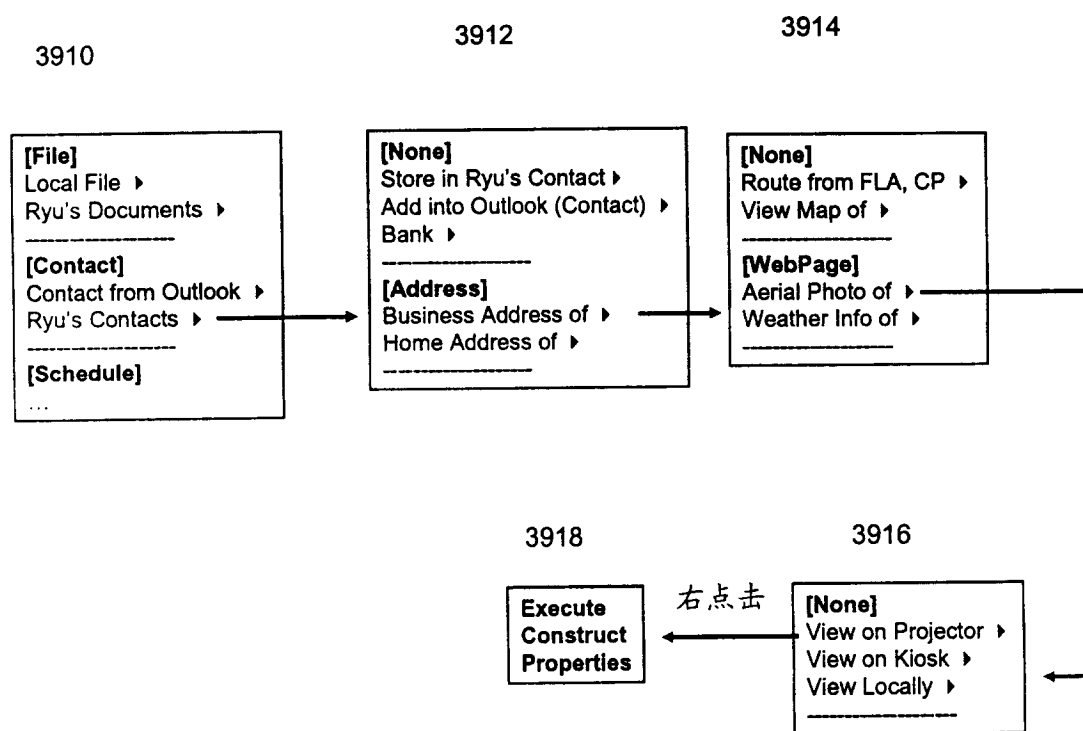
用户界面 Web 版本 (IV) 3700

图 37



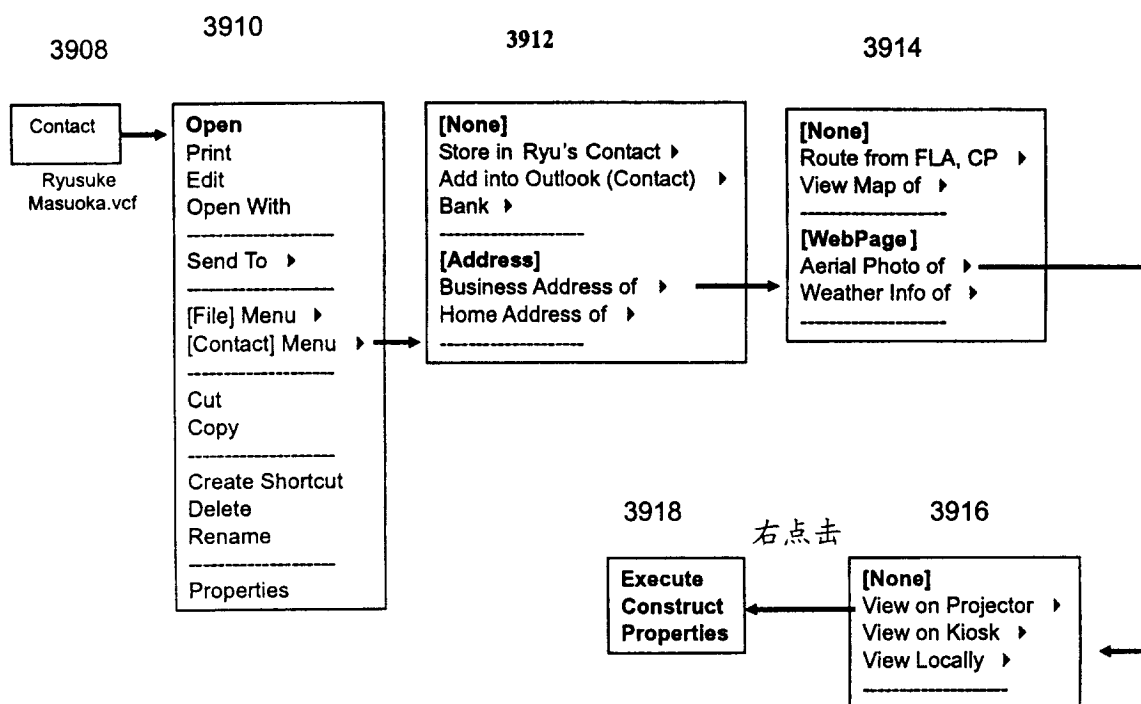
用户界面 Web 版本 (V) 3800

图 38



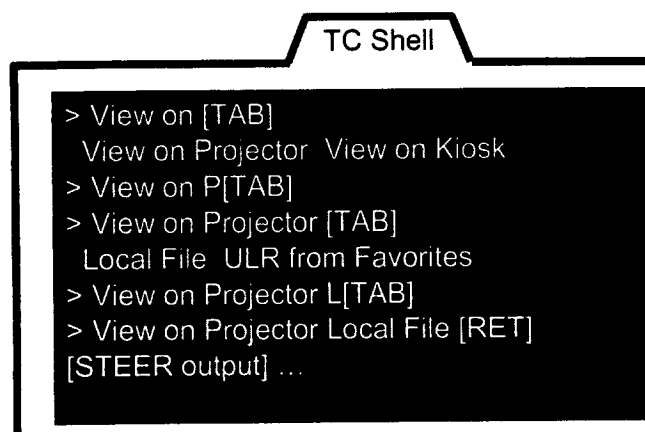
用户界面嵌套菜单版本 (I) 3900

图 39



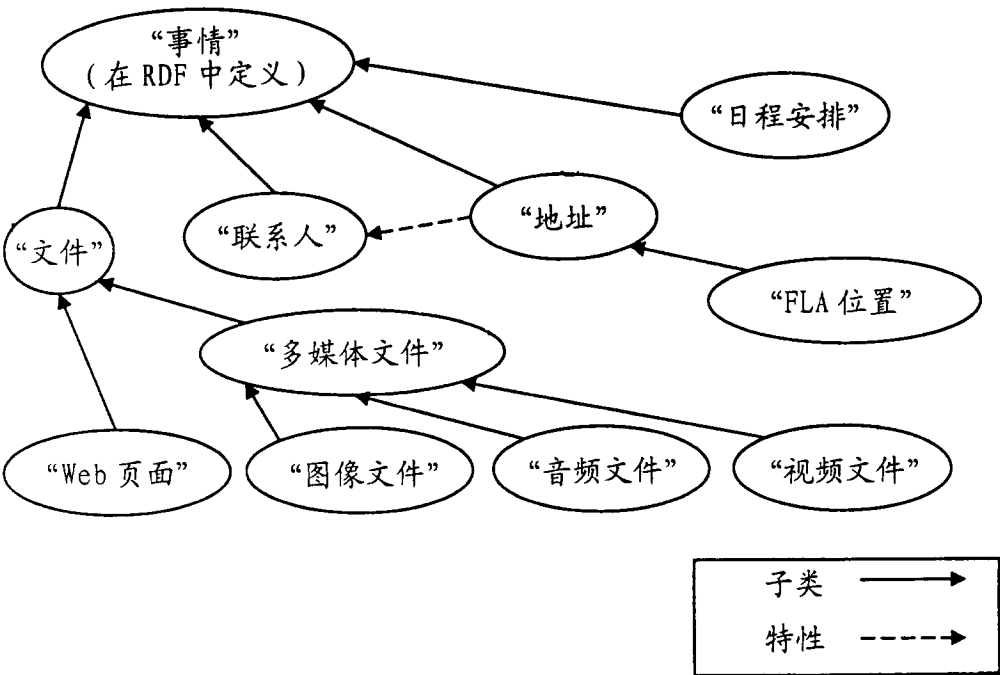
用户界面嵌套菜单版本 (II) 4000

图 40



用户界面命令行版本 **4100**

图 41



语义实例类型之间的关系 4200

图 42

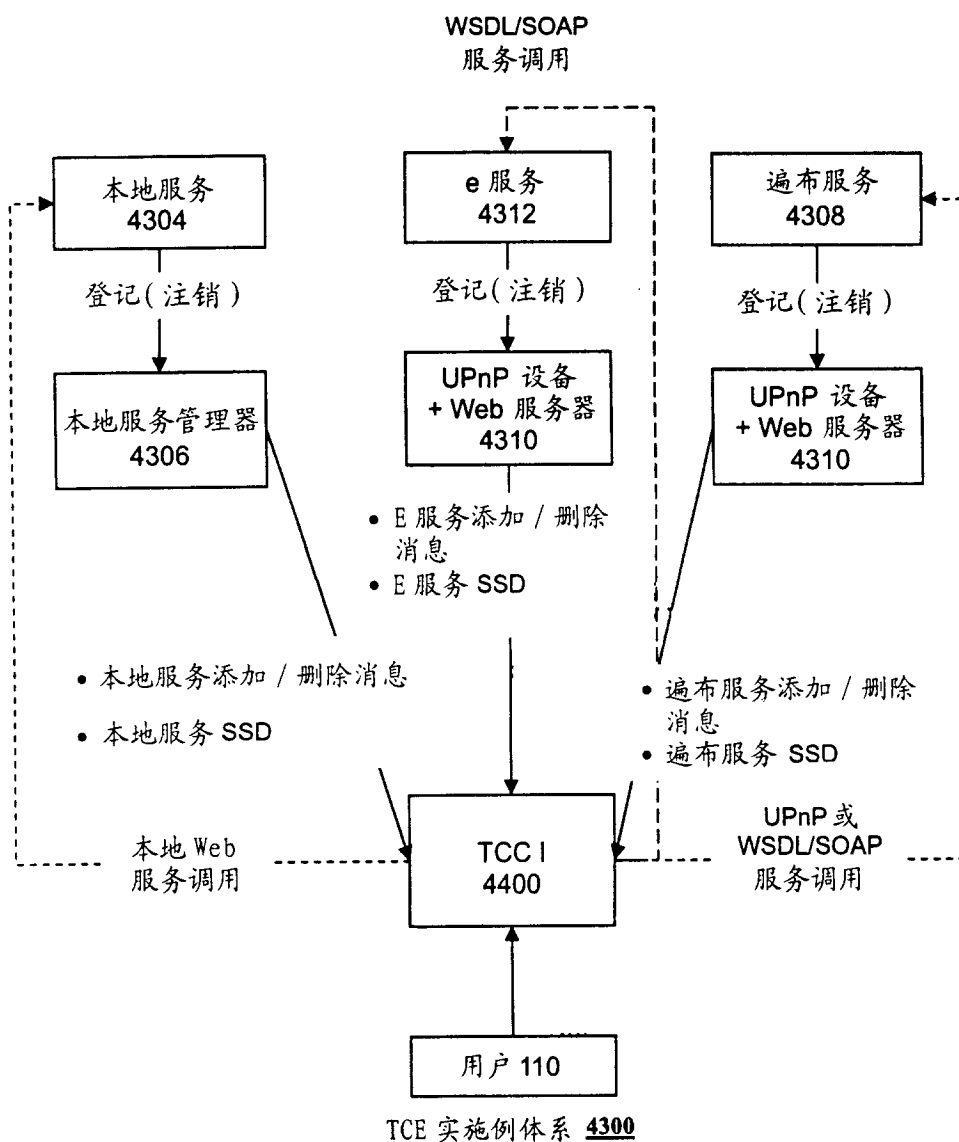


图 43

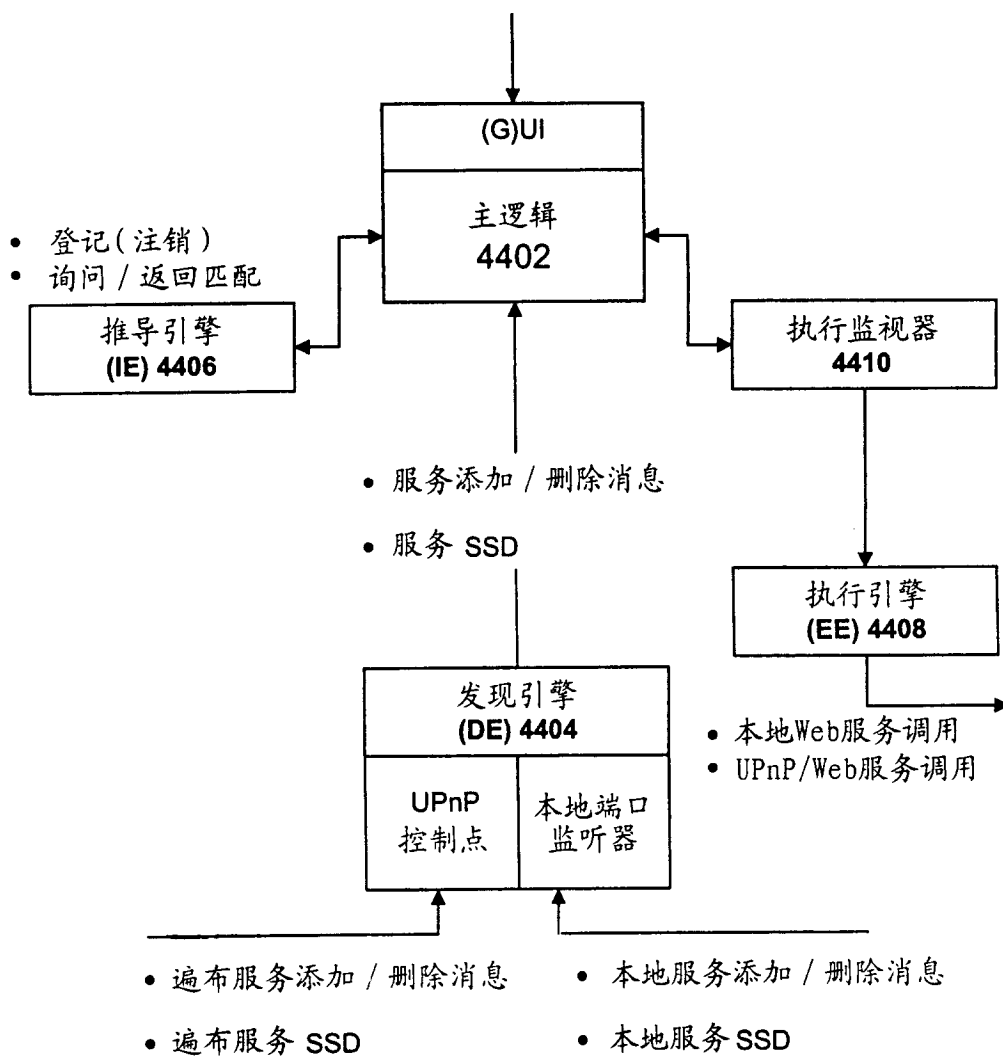
TCC I 内部机制 4400

图 44

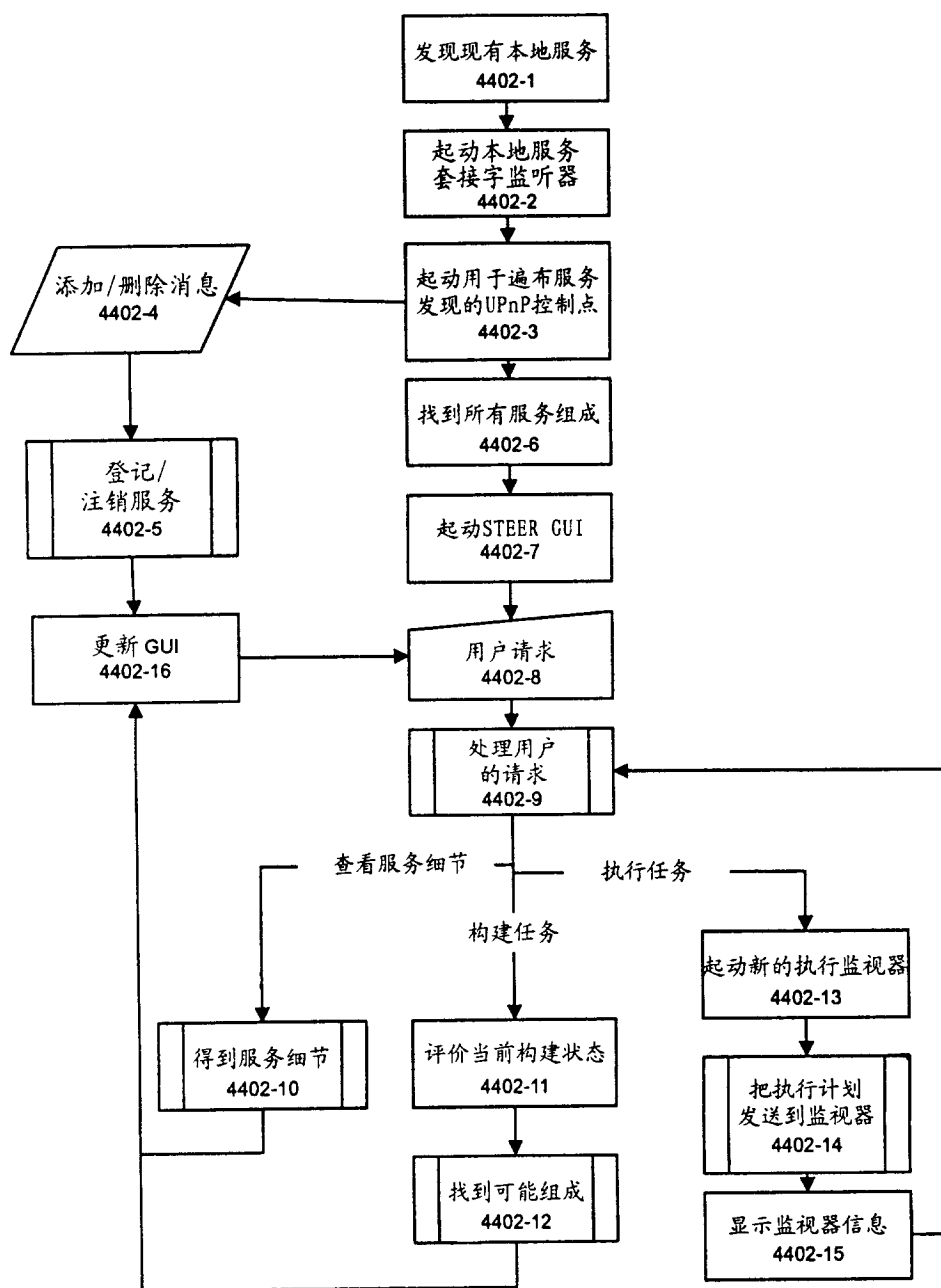
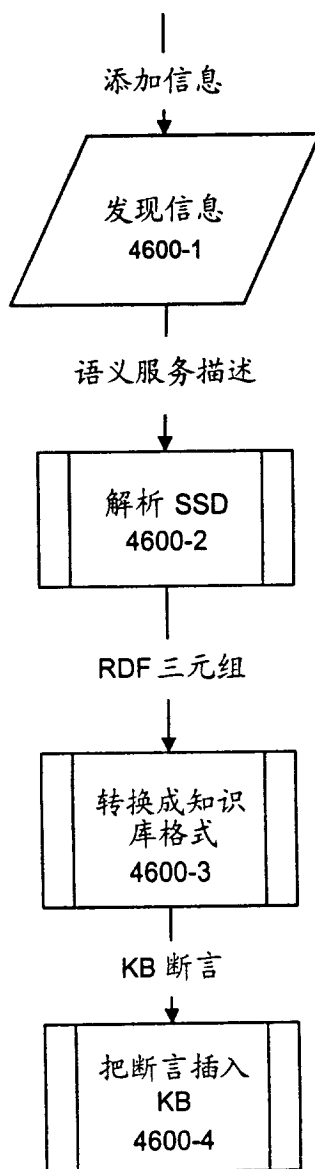
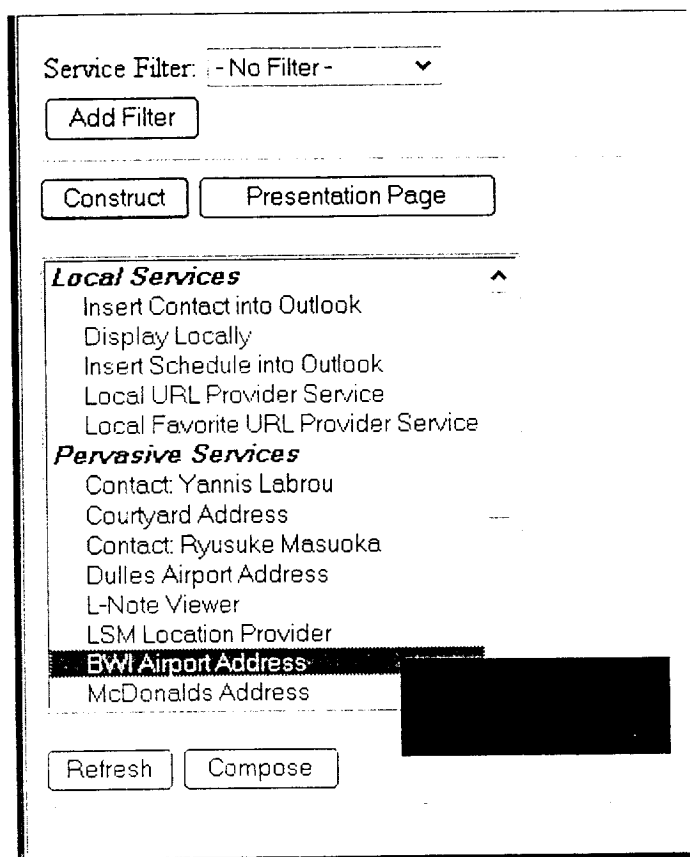
TCC I 主逻辑 **4402**

图 45



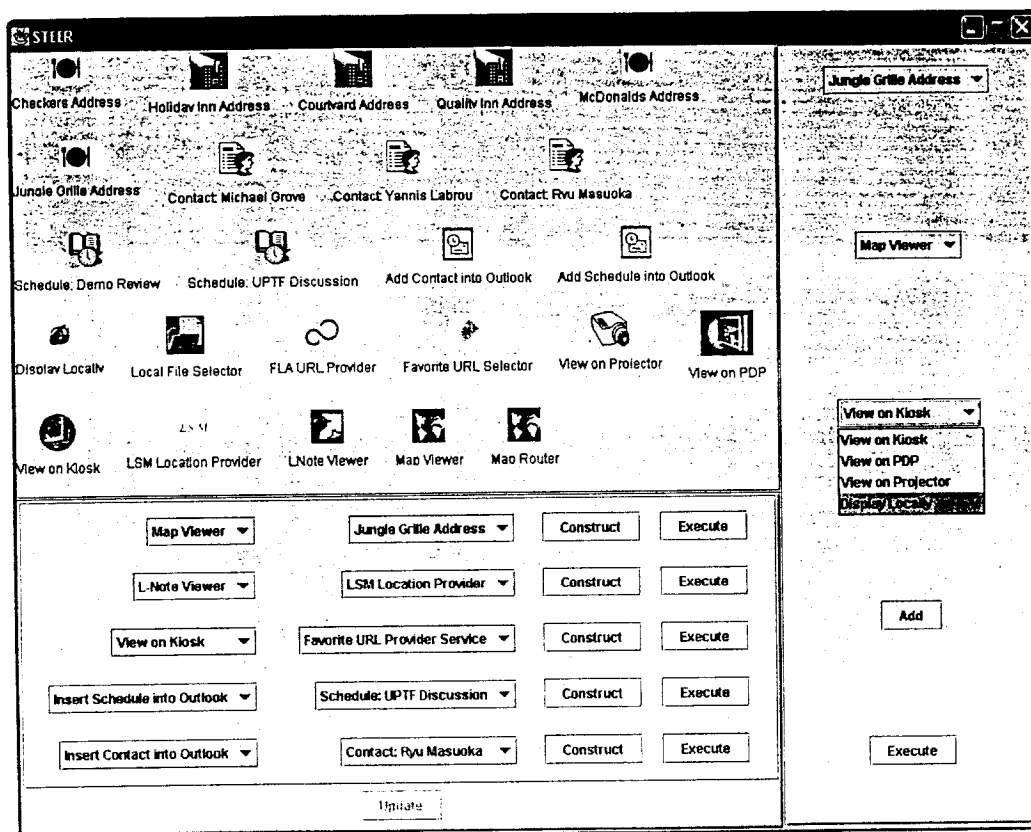
使用 IE4406 发现和登记 4600

图 46



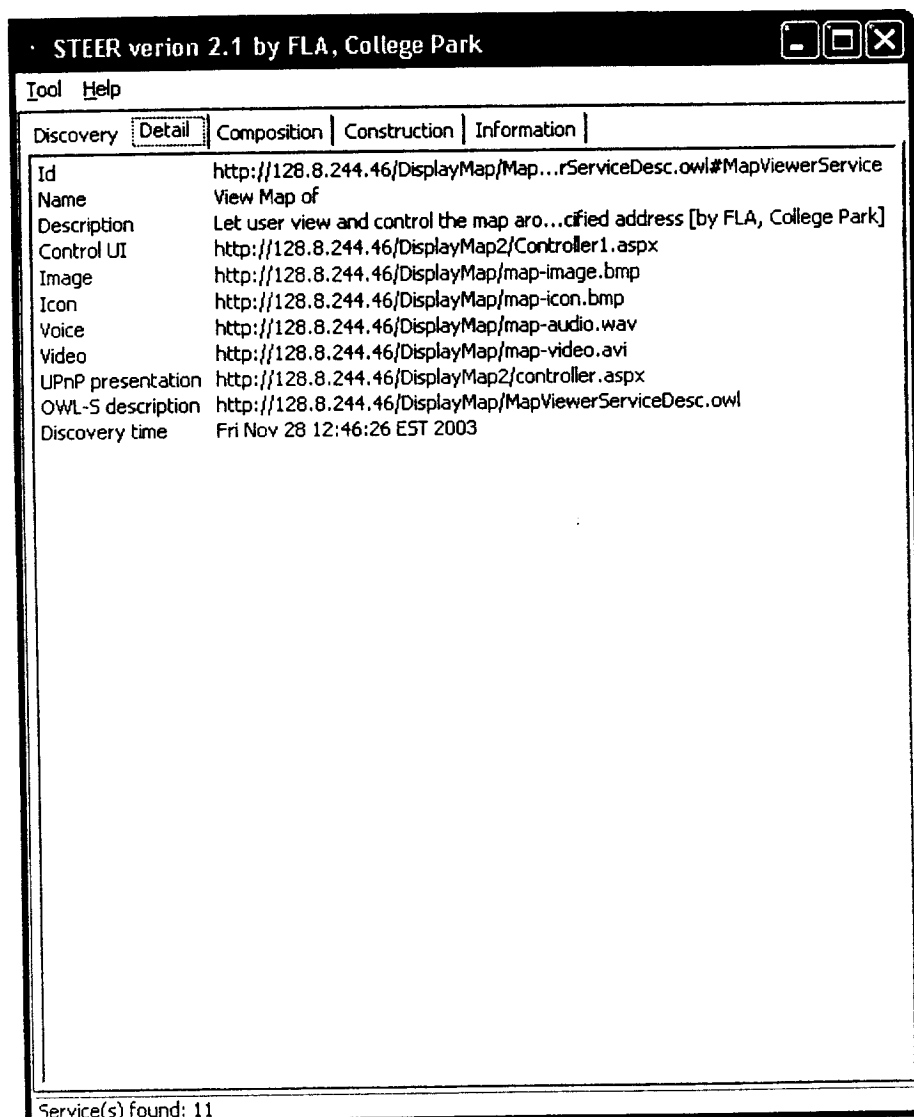
发现窗格 4700

图 47



具有各服务的图标的发现窗格 4800

图 48



详情窗格 4900

图 49

DiscoverUpdate

Service Filter: Filter by LocationRoomCurrent Location: Conference RoomAdd Filter

Map ViewerDulles Airport AddressExecuteConstruct

服务过滤 5000

图 50

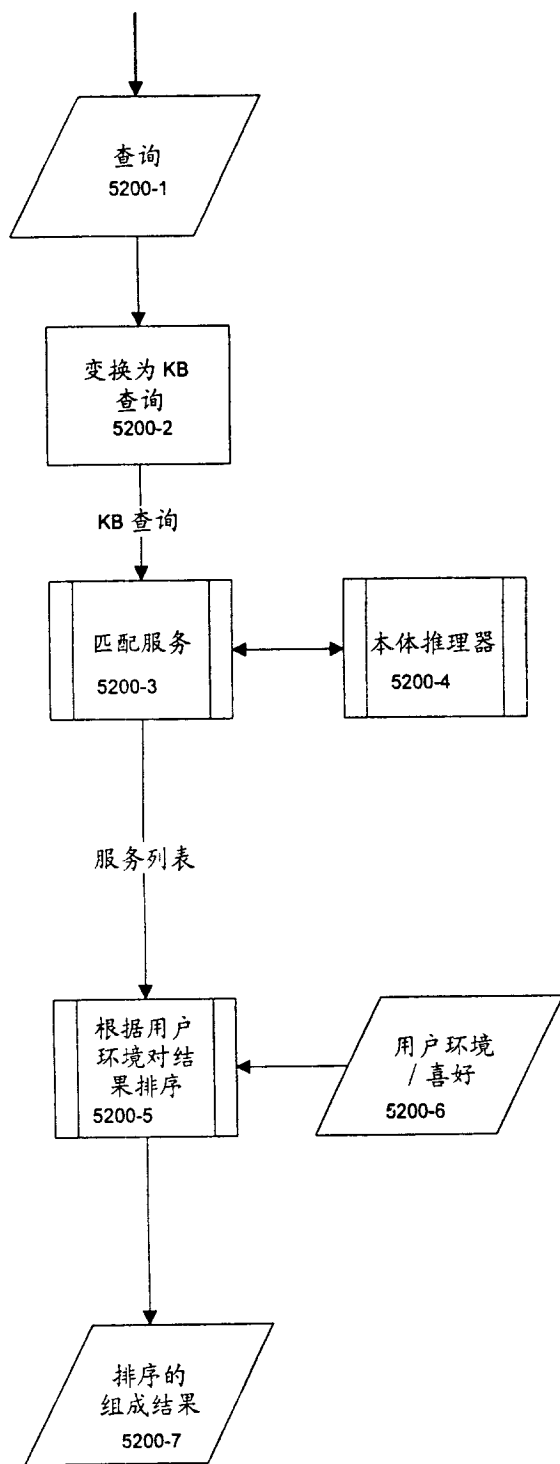
DiscoverUpdate

Service Filter: - No Filter -Add Filter

Dulles Airport Address- Search Web Services -Map ViewerMap Router- Search Web Services -

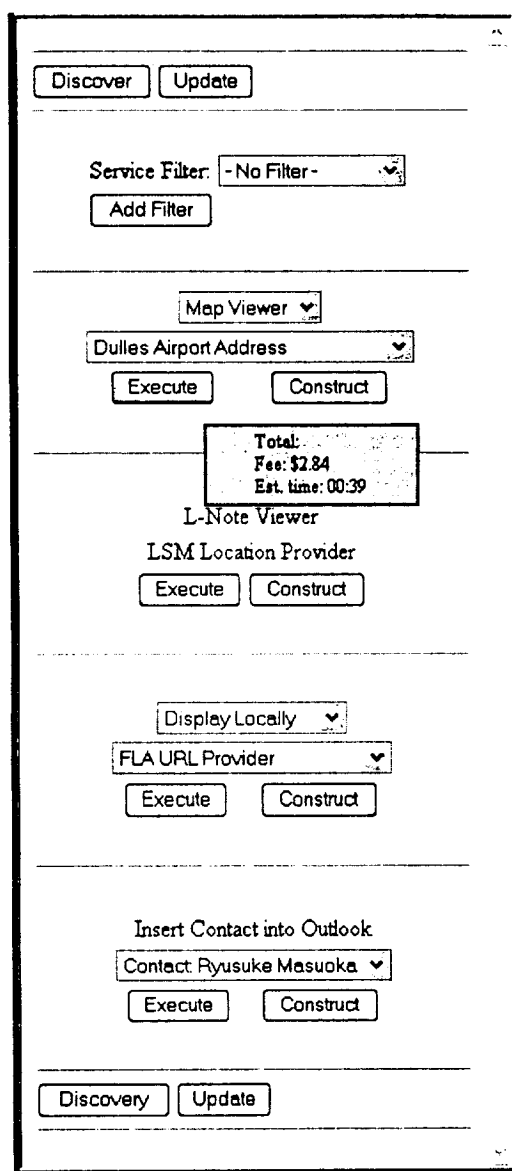
示出 “Search Web Services” 的发现窗格 5100

图 51



利用 IE4406 找到组成 **5200**

图 52



组成窗格 **5300**

图 53

Service Filter: - No Filter - ▼

Add Filter

Add Service

[- None -]

FLA URL Provider ▼ Remove Service

[- URL -] Add Service

Display Locally ▼ Remove Service

[- Control URL -]

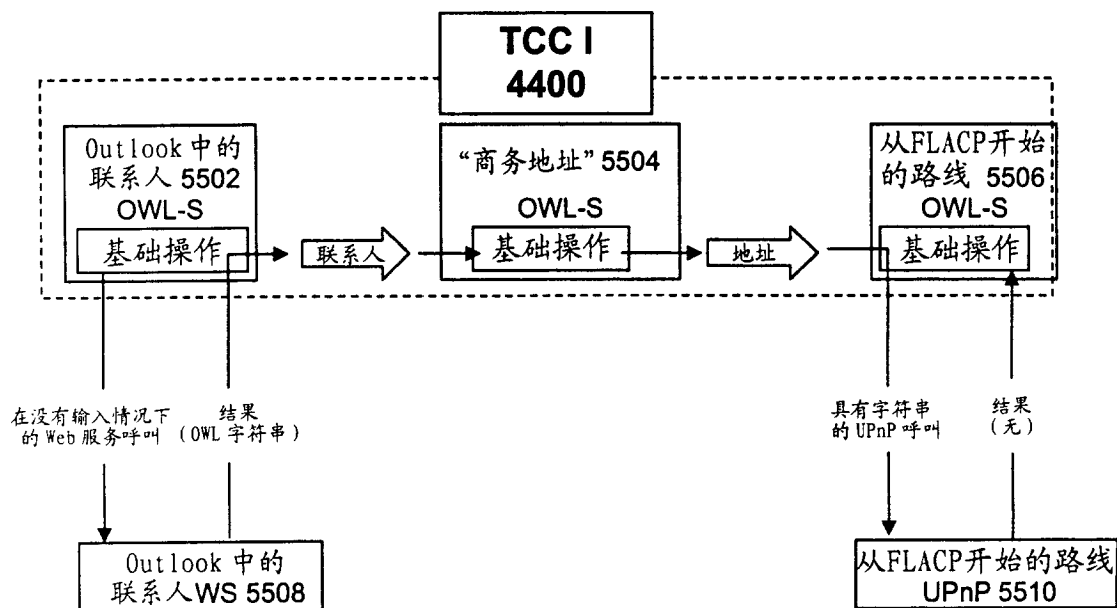
Add Service

Execute Save

Discovery Compose

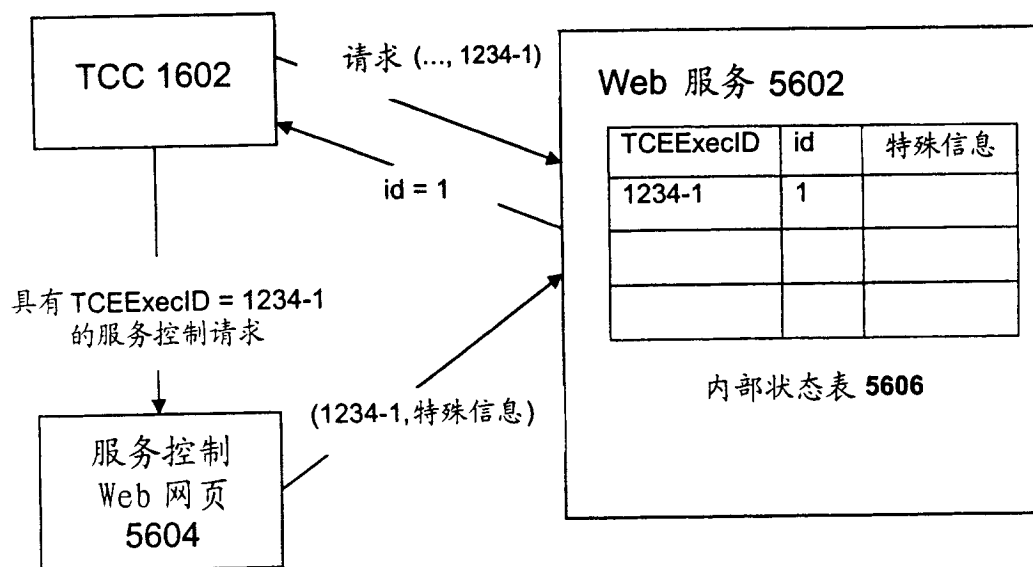
构建窗格 5400

图 54



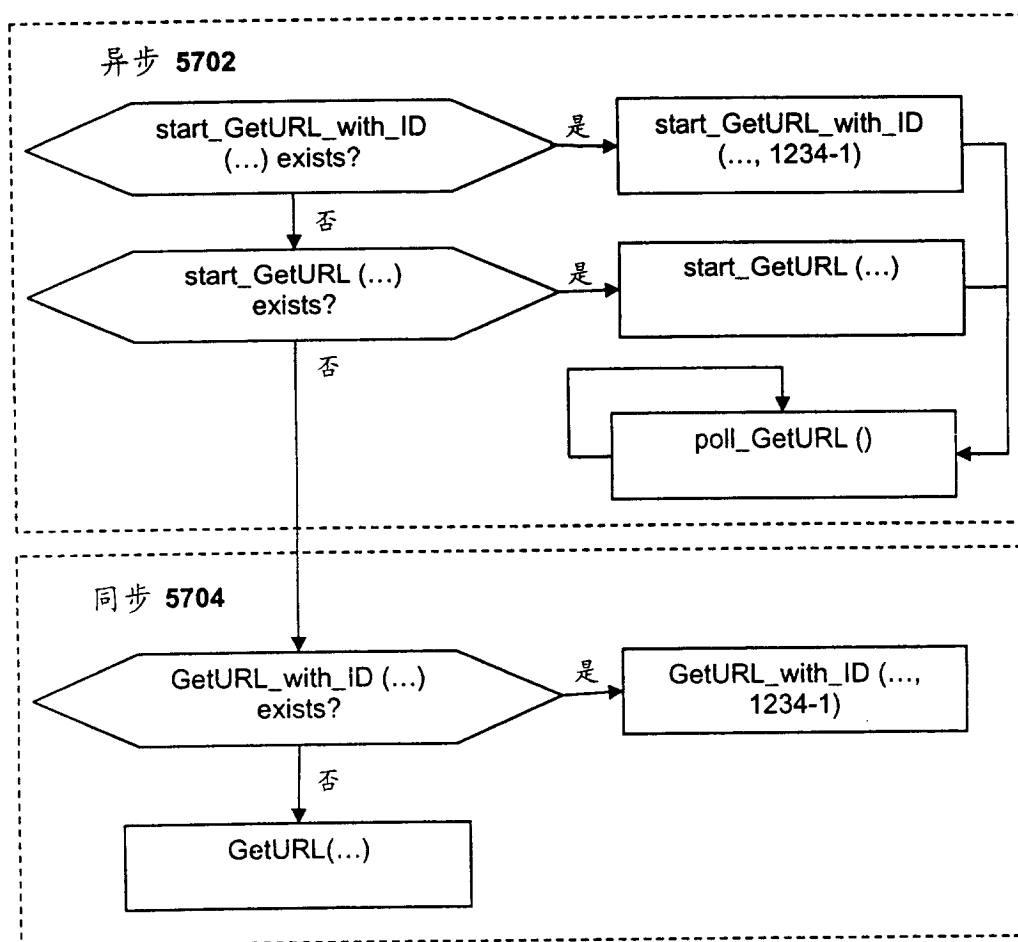
语义层和语法层的任务执行过程 5500

图 55



利用 Web 服务器支持调用 Web 服务 5600

图 56



TCC I 4400 的服务调用流程图 5700

图 57

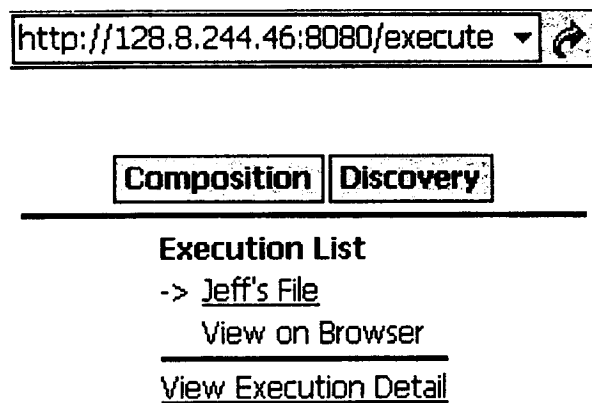
具有服务控制 UI (I) 的基于 Web 的 UI **5800**

图 58

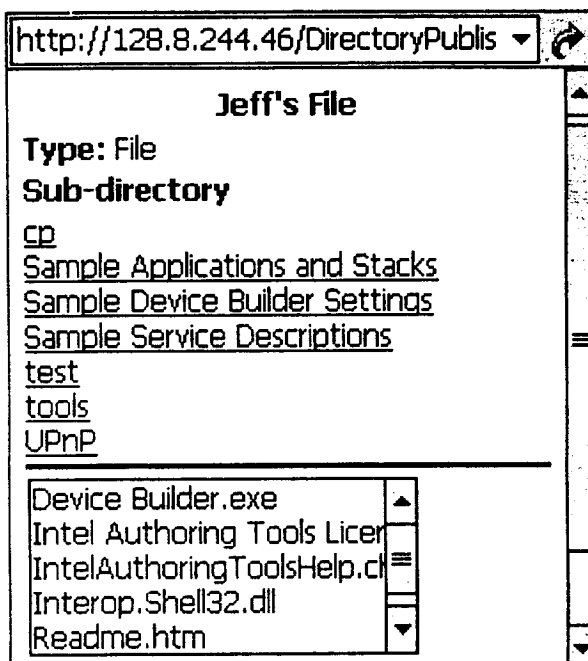
具有服务控制 UI (II) 的基于 Web 的 UI **5900**

图 59

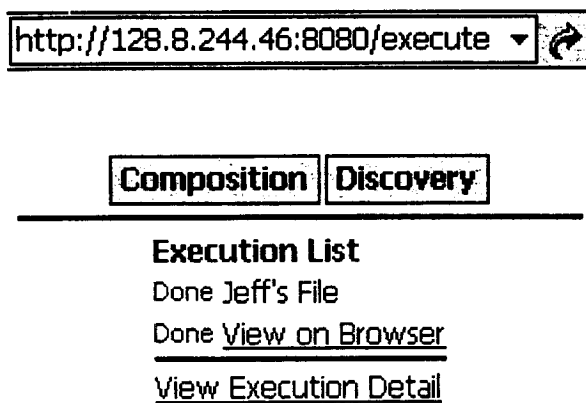
具有服务控制 UI (III) 的基于 Web 的 UI **6000**

图 60

Done Executing:

User Input: Enter Book Title

Currently Executing: Amazon Book Search

Still to Execute:

Verify Credit Card
Confirm Order
Complete Order

Update Save

Reconstruct Discovery Compose

监视器窗格 6100

图 61

Inputs:
[- None -]

Add Service

[- None -]

Local File Selector Exact Match: ☒

[- Local File (File) -] Add Service

Min Resolution: 1024 x 768

View On Projector Exact Match: ☐ Max Resolution: x

Brand [- Any -]

[- Control URL (URL) -] Add Service

Display Locally Exact Match: ☒

[- None -]

Add Service

Outputs:
[- None -]

Name: Present slides

Text Description:
Present the
PowerPoint slides
on the projector
available in the
room.

Save Cancel

Discovery Compose

保存窗格 6200

图 62

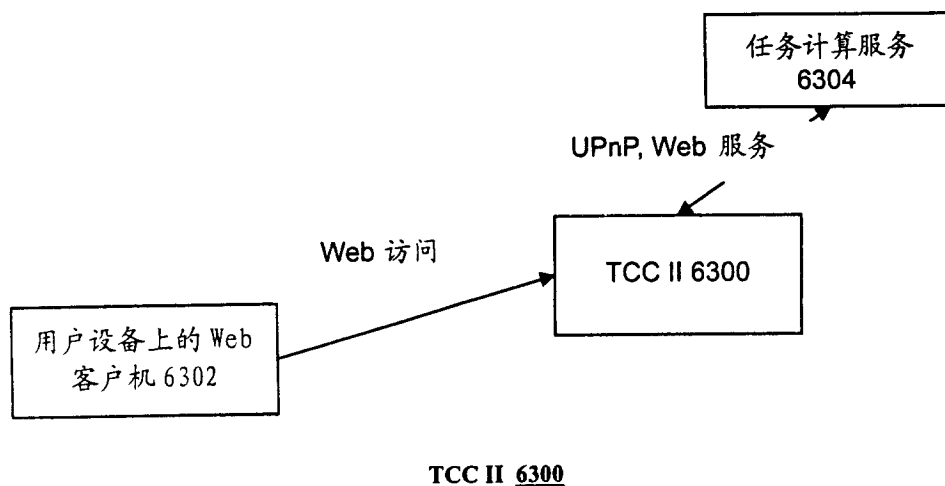


图 63

```
...
<!-- Profile description -->

<profile:ServiceProfile rdf:ID="ProjectorProfile">
  <service:isPresentedBy rdf:resource="#ProjectorService"/>

  <profile:serviceName encoding="en">
    View on Projector</profile:serviceName>
  <profile:serviceName encoding="jp">
    プロジェクタで表示する</profile:serviceName>

  <profile:textDescription encoding="en">
    Lets you view the specified URL on the Projector
    in the conference room</profile:textDescription>
  <profile:textDescription encoding="jp">
    会議室のプロジェクタで指定された URL を表示し、
    コントロールする
  </profile:textDescription>

  ...
```

SSD 中的多种自然语言支持 **6400**

图 64

FLA Bank Service

For Deposit

File Type:

Target URL:

Service Name:

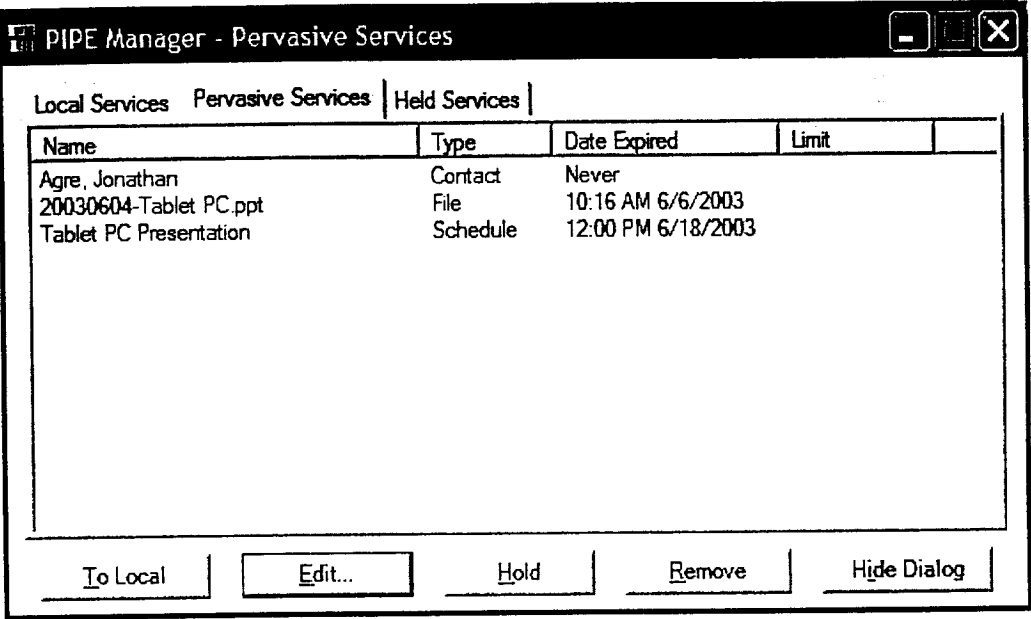
Time Out:

Current Available Services

Actions	File Type	Service Name	Original URL	Deposit Time	Time out	Unit
☐ ☐	File	Photo (Mexico Trip)	http://128.8.244.13/P237.jpg	12/13/2002 6:29:23 PM	10	Day (s)
☐ ☐	Favorite	Yahoo Home	http://www.yahoo.com	12/13/2002 6:29:41 PM	365	Day (s)
☐ ☐	Contact	Contact for Sam Chen	http://128.8.244.37/Sam.daml	12/13/2002 6:30:05 PM	30	Day (s)
☐ ☐	Schedule	Office Admin Meeting (12/17/02)	http://128.8.244.33/OAM.daml	12/13/2002 6:31:25 PM	5	Day (s)

库服务 6500

图 65



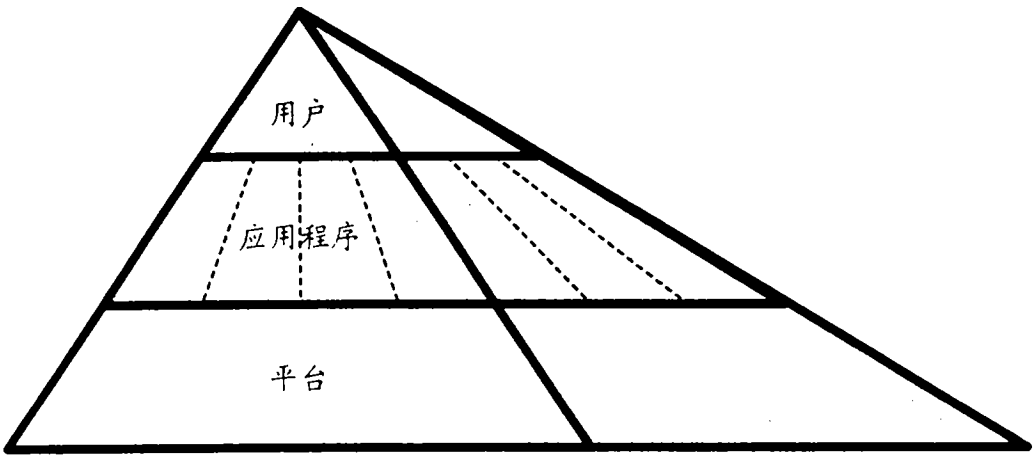
PIPE 管理器 6600

图 66



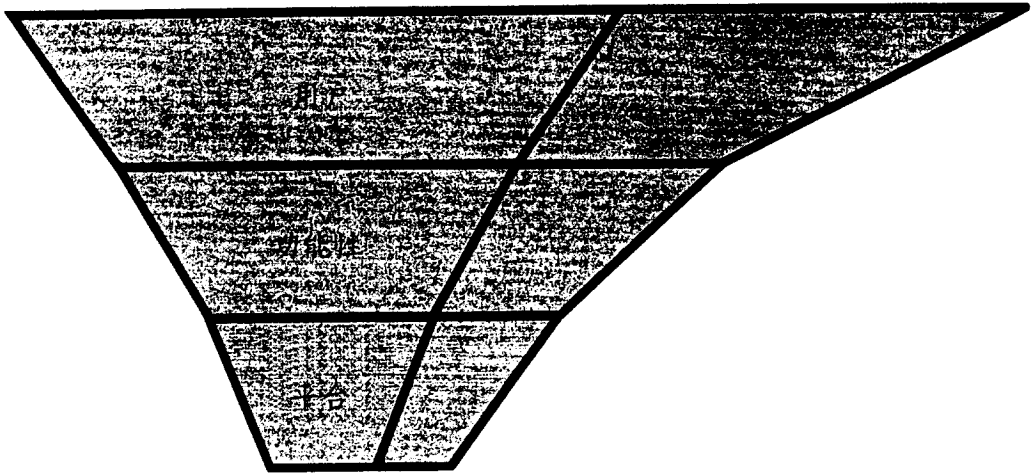
白洞拖放界面 6700

图 67



任务计算前的能力金字塔 **6800**

图 68



任务计算后的倒转能力金字塔 **6900**

图 69

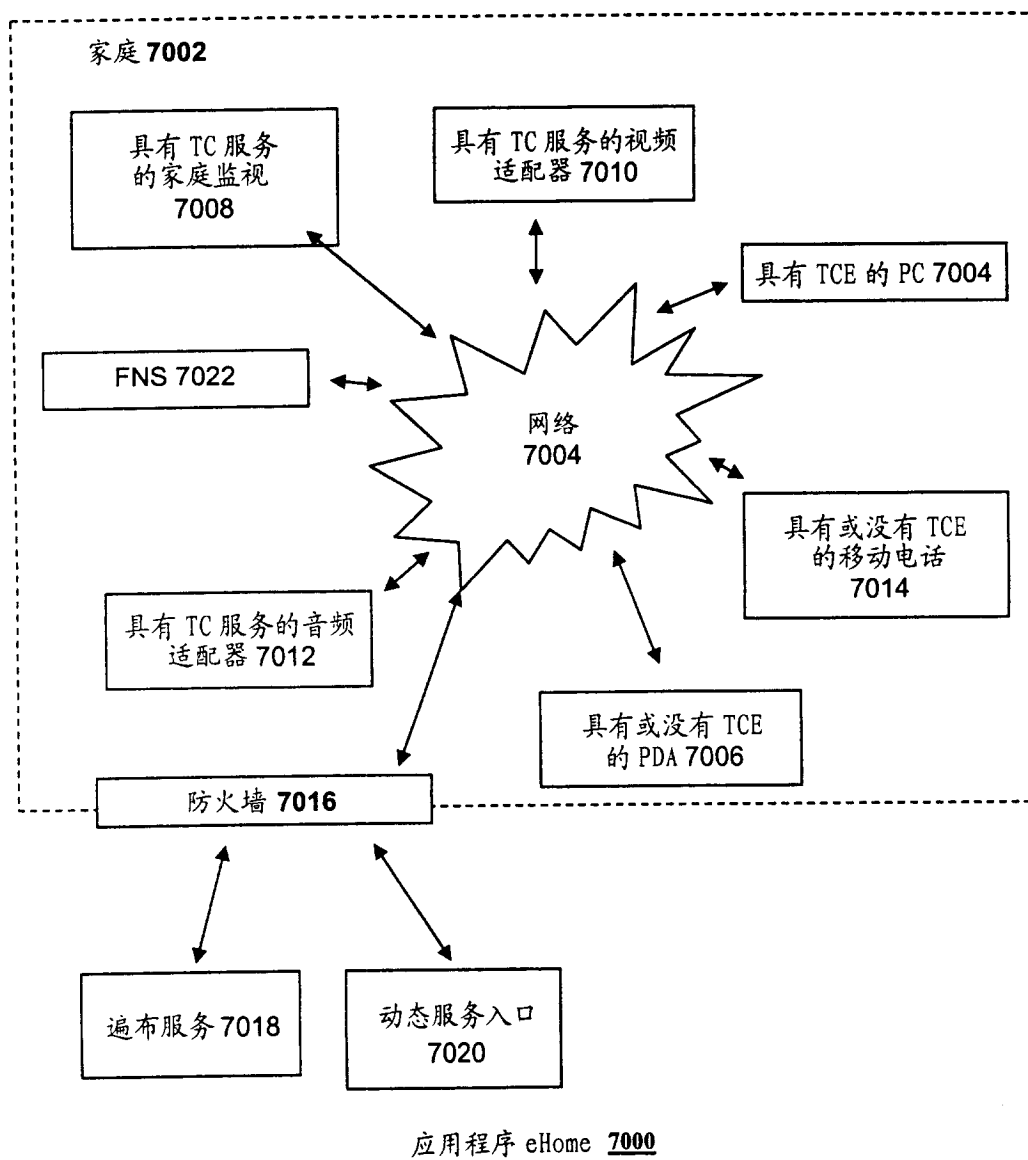
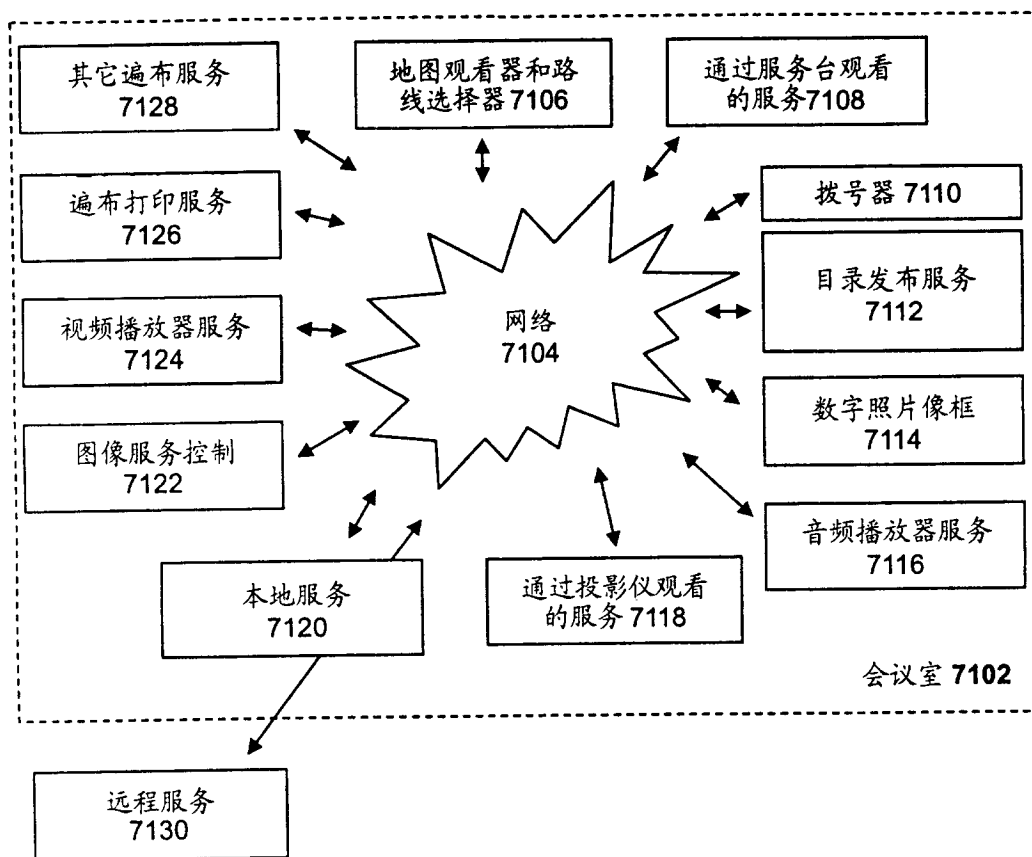
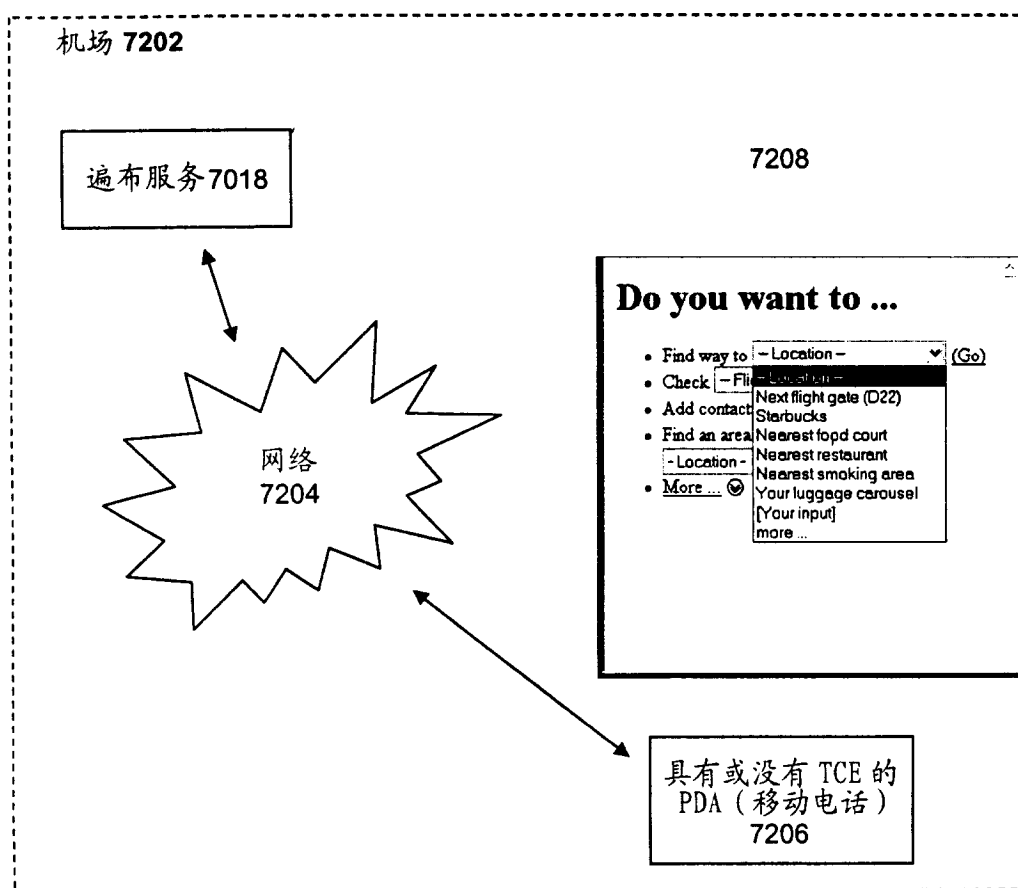


图 70



应用程序 eOffice 7100

图 71



应用程序机场 7200

图 72

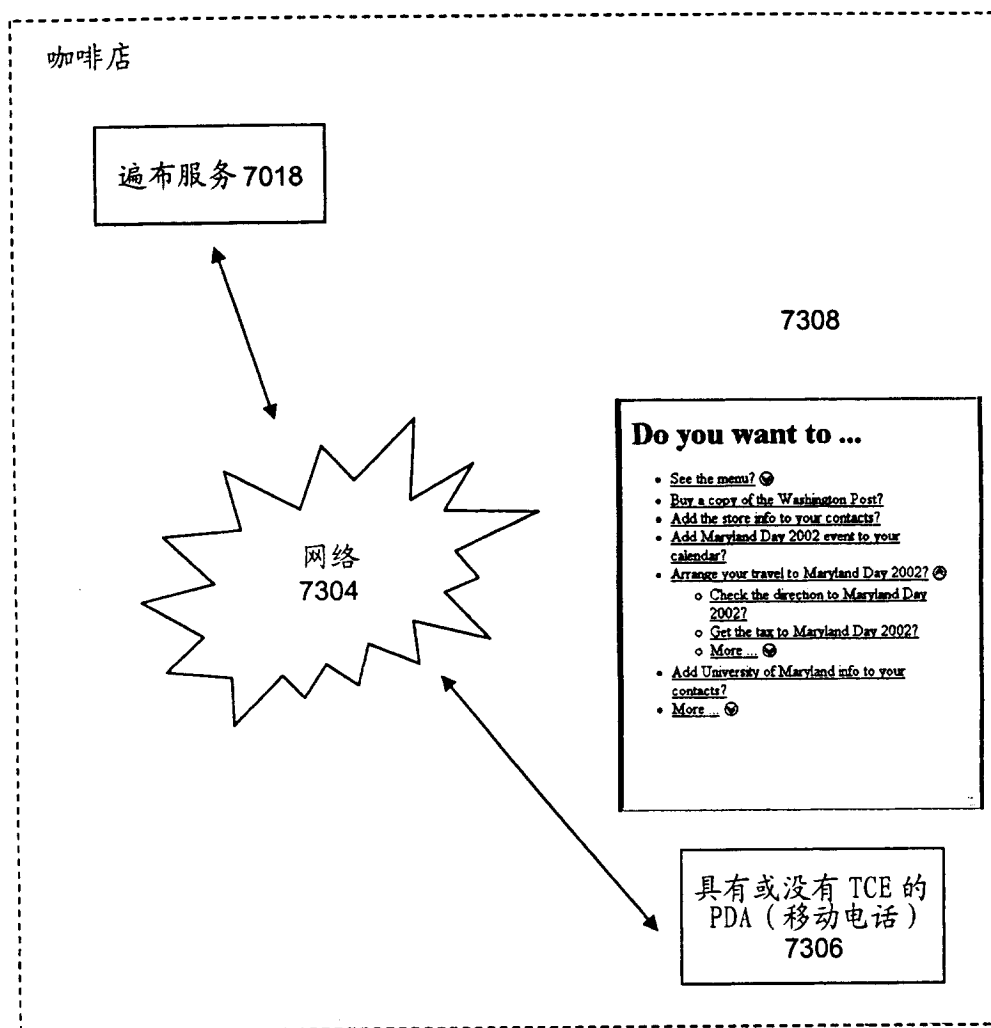
应用程序咖啡店 7300

图 73

In this page you can find the following:

an *Event* of type *Talk* with
Title: **Signs of the Times: Information Technology in A Cautious Economy**
Speaker: **Kathleen Hayes**
on **Date:** **Tuesday, May 14, 2002**
with **Starting Time** of **08:00 AM**
and **Ending Time** of **10:00 AM**
at **Location** **Baltimore Marriott Waterfront**

☒ Would you like to *find the address* of **Baltimore Marriott Waterfront**

☐ Would you like to *add event to*

☐ Would you like to *send a note about this event to*

☐ Would you like to *find events of type talk by speaker* **Kathleen Hayes**

☐ Would you like to *find events of type talk at location* **Baltimore Marriott Waterfront**

在给定当前内容和环境的情况下，TCE 识别在网页中描述了什么，并向用户建议某些可能的动作 **7400**

图 74

In this page you can find the following:

an Event of type *Talk* with
Title: Signs of the Times: Information Technology in A Cautious Economy
Speaker: Kathleen Hayes
on *Date: Tuesday, May 14, 2002*
with *Starting Time* of 08:00 AM
and *Ending Time* of 10:00 AM
at *Location* **Baltimore Marriott Waterfront**
with *address* 700 Aliceanna Street, Baltimore, MD 21202

☒ Would you like to *see directions* to **Baltimore Marriott Waterfront** from

☐ Would you like to *find* *close to* **Baltimore Marriott Waterfront**

☐ Would you like to *add event to*

☐ Would you like to *send a note about this event to*

☐ Would you like to *find events* of type *talk* by *speaker* **Kathleen Hayes**

☐ Would you like to find events of type *talk* at *location* **Baltimore Marriott Waterfront**

You can save your actions so far as a task. Choose a name for this task

在用户决定首先找到谈话的地点的地址后，TCE 找到地址并向用户
呈现可能动作的新列表 **7500**

图 75

In this page you can find the following:

an Event of type *Talk* with
Title: *Signs of the Times: Information Technology in A Cautious Economy*
Speaker: *Kathleen Hayes*
on **Date:** *Tuesday, May 14, 2002*
with **Starting Time** of *08:00 AM*
and **Ending Time** of *10:00 AM*
at **Location** *Baltimore Marriott Waterfront*
with address *700 Aliceanna Street, Baltimore, MD 21202*

☒ Would you like to add add event to
☒ include directions to *Baltimore Marriott Waterfront* from

☒ Would you like to send a note about this event to
☒ include directions to *Baltimore Marriott Waterfront* from

☐ Would you like to find events of type *talk* by speaker *Kathleen Hayes*

☐ Would you like to find events of type *talk* at location *Baltimore Marriott Waterfront*

☐ Would you like to just see directions to *Baltimore Marriott Waterfront* from

☐ Would you like to find close to *Baltimore Marriott Waterfront*

You can save your actions so far as a task. Choose a name for this task

在用户选择检查地图上谈话的地址的位置后，TCE 呈现动作的修改列表，其具有的选项可把到此时的用户动作保存为用于将来（再）使用的新任务 **7600**

图 76