



(12)发明专利

(10)授权公告号 CN 103947140 B

(45)授权公告日 2017.09.29

(21)申请号 201280056941.6

(72)发明人 R·F·麦茨尼斯

(22)申请日 2012.09.19

(74)专利代理机构 上海专利商标事务所有限公司 31100

(65)同一申请的已公布的文献号  
申请公布号 CN 103947140 A

代理人 张东梅

(43)申请公布日 2014.07.23

(51)Int.Cl.  
H04J 1/00(2006.01)

(30)优先权数据  
61/536,826 2011.09.20 US

(56)对比文件

(85)PCT国际申请进入国家阶段日  
2014.05.20

US 2006111880 A1,2006.05.25,  
US 2004064548 A1,2004.04.01,  
US 2009300057 A1,2009.12.03,  
CN 1325217 A,2001.12.05,  
CN 1380613 A,2002.11.20,

(86)PCT国际申请的申请数据  
PCT/US2012/056018 2012.09.19

审查员 杨文君

(87)PCT国际申请的公布数据  
W02013/043665 EN 2013.03.28

(73)专利权人 以太杰作有限责任公司  
地址 美国纽约州

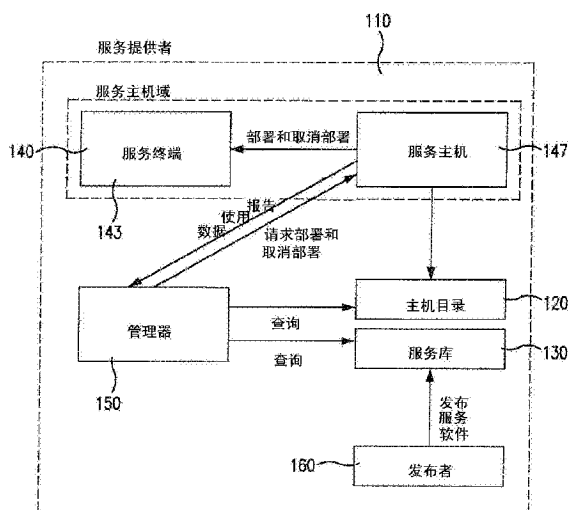
权利要求书3页 说明书11页 附图17页

(54)发明名称

用于位置无关软件的需求驱动的部署的系统和方法

(57)摘要

描述了用于提供和消费web服务的技术,包括被配置为存储一种或多种web服务的服务库和连接到各服务主机的、被配置为存储与各服务主机相关的数据的主机目录。各服务主机是网络,且适于通过实例化web服务中的一种的一个或多个终端来接收和履行对被存储在服务库中的web服务的部署请求。管理器被配置为查询主机目录和服务库,生成部署计划,并把部署请求传送给一个或多个服务主机。



1. 一种用于提供web服务的系统,包括:
  - 服务库,被配置为存储一种或多种web服务;
  - 一个或多个服务主机,其被连接到网络,且适于接收和履行对所述被存储在所述服务库中的web服务的部署请求,其中,履行部署请求包括实例化所述web服务中的一种的一个或多个终端;
  - 主机目录,其被连接到所述服务主机,且被配置为存储与所述一个或多个服务主机相关的数据;以及
  - 管理器,其被配置为查询所述主机目录以基于所存储的与所述一个或多个服务主机相关的数据来检索一响应、查询所述服务库以基于所存储的web服务来检索一响应、基于检索到的响应来生成部署计划、并基于所生成的部署计划把部署请求传送给所述一个或多个服务主机。
2. 如权利要求1所述的系统,其特征在于,所述服务库还被配置为从发布者接收所述一种或多种web服务。
3. 如权利要求1所述的系统,其特征在于,所述一个或多个服务主机还被配置为把关于所述web服务的使用的信息传送给所述管理器。
4. 如权利要求1所述的系统,其特征在于,所述一个或多个服务主机还被配置为把与所述一个或多个服务主机相关的数据发送给所述主机目录,且其中,与所述一个或多个服务主机相关的数据包括可用资源。
5. 如权利要求1所述的系统,进一步包括发现服务,所述发现服务包括目录,所述目录被配置为注册和维护每一服务主机的所述终端的条目。
6. 如权利要求5所述的系统,其特征在于,所述目录服务还被配置为存储在所述web服务和在其上部署所述web服务的一组相应终端之间的映射。
7. 如权利要求6所述的系统,其特征在于,所述目录服务还被配置为接收对应于所述web服务中的一种的URI、并且请求并解析对应于在其上部署所述web服务中的一种的所述终端中的一个的URL。
8. 如权利要求7所述的系统,进一步包括服务使用者,其包括:
  - 使用者代理;以及
  - 入网点,所述使用者代理绑定到所述入网点,所述入网点被配置为与所述发现服务通信,且被配置为绑定到所述一个或多个服务终端。
9. 如权利要求8所述的系统,其特征在于,所述入网点被配置为对服务的位置进行抽象,同时保持所述使用者代理的观点,由此充当供所述使用者代理进入到所述web服务的实例化的网关。
10. 一种提供web服务的方法,包括:
  - 把一种或多种web服务存储在服务库中;
  - 把与一个或多个服务主机相关的数据存储在主机目录中;
  - 借助于管理器控制所述一个或多个服务主机中的每一个上的一个或多个服务终端上的所述一种或多种web服务的部署,其中,控制部署包括:
    - 查询所述服务库以基于所存储的web服务来检索一响应;
    - 查询所述主机目录以基于所存储的与所述一个或多个服务主机相关的数据来检索一

响应；

基于检索到的响应来生成部署计划；

基于所生成的部署计划把部署请求传送给所述一个或多个服务主机；以及  
响应于所述部署请求，实例化所述web服务中的一种的所述一个或多个终端。

11. 如权利要求10所述的方法，进一步包括在所述服务库处从发布者接收所述一种或多种web服务。

12. 如权利要求10所述的方法，进一步包括将关于所述web服务的使用的信息从所述一个或多个服务主机传送到所述管理器。

13. 如权利要求10所述的方法，进一步包括将与所述一个或多个服务主机相关的数据从所述一个或多个服务主机发送到所述主机目录，且其中，与所述一个或多个服务主机相关的数据包括可用资源。

14. 如权利要求10所述的方法，进一步包括在发现服务中注册和维护每一服务主机的所述终端的条目，所述发现服务包括目录。

15. 如权利要求14所述的方法，进一步包括借助于所述发现服务存储在所述web服务和在其上部署所述web服务的一组相应终端之间的映射。

16. 如权利要求15所述的方法，进一步包括在所述发现服务处接收对应于所述web服务中的一种的URI，并且请求并解析对应于在其上部署所述web服务中的一种的所述终端中的一个的URL。

17. 一种用于消费web服务的方法，包括：

把使用者代理绑定到入网点；

经由所述入网点与发现服务通信，以便解析对应于服务终端的URL，所述服务终端对应于期望的web服务URI；以及

把所述入网点绑定到所述服务终端；

其中，所述发现服务被配置为与管理器通信，所述管理器被配置为查询主机目录以基于存储于其中的与一个或多个服务主机相关的数据来检索一响应、查询服务库以基于存储于其中的web服务来检索一响应、基于检索到的响应来生成部署计划、并基于所生成的部署计划把部署和取消部署请求传送给一个或多个服务主机，所述一个或多个服务主机中的一个被配置为实例化所述服务终端。

18. 如权利要求17所述的方法，其特征在于，所述入网点被配置为对服务的位置进行抽象，同时保持所述使用者代理的观点，由此充当供所述使用者代理进入到所述web服务的实例化的网关。

19. 如权利要求17所述的方法，所述主机目录被配置为存储与所述一个或多个服务主机相关的数据。

20. 如权利要求17所述的方法，所述服务目录被配置为至少存储所期望的web服务。

21. 一种非暂态计算机可读介质，其包含计算机可执行指令，在被执行时，所述计算机可执行指令引起一个或多个计算机设备执行一种提供web服务的方法，所述方法包括：

把一种或多种web服务存储在服务库中；

把与一个或多个服务主机相关的数据存储在主机目录中；

借助于管理器控制所述一个或多个服务主机中的每一个上的一个或多个服务终端上

的所述一种或多种web服务的部署,其中,控制部署包括:

查询所述服务库以基于所存储的web服务来检索一响应;

查询所述主机目录以基于所存储的与所述一个或多个服务主机相关的数据来检索一响应;

基于检索到的响应来生成部署计划;

基于所生成的部署计划把部署请求传送给所述一个或多个服务主机;以及

响应于所述部署请求,实例化所述web服务中的一种的所述一个或多个终端。

22. 如权利要求21所述的非暂态计算机可读介质,还包括在所述服务库处从发布者接收所述一种或多种web服务。

23. 如权利要求21所述的非暂态计算机可读介质,还包括将关于所述web服务的使用的信息从所述一个或多个服务主机传送给所述管理器。

24. 如权利要求21所述的非暂态计算机可读介质,还包括将与所述一个或多个服务主机相关的数据从所述一个或多个服务主机发送给所述主机目录,且其中,与所述一个或多个服务主机相关的数据包括可用资源。

25. 如权利要求21所述的非暂态计算机可读介质,还包括在发现服务中注册和维护每一服务主机的所述终端的条目,所述发现服务包括目录。

26. 如权利要求25所述的非暂态计算机可读介质,还包括借助于所述发现服务存储在所述web服务和在其上部署所述web服务的一组相应终端之间的映射。

27. 如权利要求26所述的非暂态计算机可读介质,还包括在所述发现服务处接收对应于所述web服务中的一种的URI,并且请求并解析对应于在其上部署所述web服务中的一种的所述终端中的一个的URL。

28. 一种非暂态计算机可读介质,其包含计算机可执行指令,在被执行时,所述计算机可执行指令引起一个或多个计算机设备执行一种用于消费web服务的方法,所述方法包括:

把使用者代理绑定到入网点;

经由所述入网点与发现服务通信以便解析对应于服务终端的URL,所述服务终端对应于所期望的web服务URI;以及

把所述入网点绑定到所述服务终端;

其中,所述发现服务被配置为与管理器通信,所述管理器被配置为查询主机目录以基于存储于其中的与一个或多个服务主机相关的数据来检索一响应、查询服务库以基于存储于其中的web服务来检索一响应、基于检索到的响应来生成部署计划、并基于所生成的部署计划把部署和取消部署请求传送给一个或多个服务主机,所述一个或多个服务主机中的一个被配置为实例化所述服务终端。

29. 如权利要求28所述的非暂态计算机可读介质,其特征在于,所述入网点被配置为对服务的位置进行抽象,同时保持所述使用者代理的观点,由此充当供所述使用者代理进入到所述web服务的实例化的网关。

30. 如权利要求28所述的非暂态计算机可读介质,所述主机目录被配置为存储与所述一个或多个服务主机相关的数据。

31. 如权利要求28所述的非暂态计算机可读介质,所述服务目录被配置为至少存储所期望的web服务。

## 用于位置无关软件的需求驱动的部署的系统和方法

[0001] 对相关申请的交叉引用

[0002] 本申请涉及于2011年9月20日提交的美国临时申请第61/536,826号,该临时申请通过引用整体合并于此,并从中保护优先权。

[0003] 背景

[0004] 所公开的主题涉及用于面向服务的计算,尤其而非限制,涉及web(网)服务体系结构。

[0005] 面向服务的计算(SOC)促进了把应用组件组装到服务的网络中,服务的网络可以松散地耦合,以便创建跨越组织和计算平台的灵活的、动态的业务流程和灵巧的应用。面向服务的计算已经成为使得紧密耦合的、基于组件的分布式系统进化成可以使用统一的技术来定址服务操作、绑定到服务操作和调用服务操作的更广泛的服务网络的方法。

[0006] 面向服务的计算可以提供创建反映向自主性和异质性发展的趋势的新体系结构的方式。允许面向服务的计算的灵活的且松散耦合的流程的分布式体系结构可以被称为面向服务的体系结构(SOA),其软件服务是主要组件。SOA的主要特性是在松散地耦合的自主服务之间的互操作、促进在宏(服务)级别的代码重用和体系结构的组合性。

[0007] SOA可以由三个主要实体组成——提供者、使用者和注册表。提供者可以在注册表中发布服务位置;使用者可以使用注册表来定位服务以便对其编程并调用。SOA实现可以指定用来描述服务的语言、提供发布和发现它们的方式并规定用来与它们交互的协议和通信机制。日益增加地使用共同称为“web服务”的这样一组标准来实现软件服务在因特网上的部署。

[0008] Web服务可以基于平台无关的标准,例如通过W3C工作组的努力开发的那些标准。这些标准可以定义允许在跨越因特网的异构计算平台上的客户机和服务之间交互的协议、消息格式和服务描述语言。

[0009] 由web服务引入的标准可以提供使得分布式系统从紧密耦合的分布式应用进化成松散耦合的服务系统的手段。通过使用良好定义的接口交换消息,各标准可以允许在异构计算平台之间的互操作。通过对实现和主控技术进行抽象(abstracting),平台无关的web服务标准可以提供这样的手段:使得对现有的异构服务的访问匀质化,同时促进开发新的服务,而无需强制任何人采用特定的实现或主控技术。尽管web服务模型已经被广泛采纳,但其前景受到至少两种限制的削弱。

[0010] 首先,web服务寻址模型可能是刚性的,且仅适用于具有高度可靠的主机的高度可靠的联网环境。它可能无法考虑因特网上的主机的内在动态和不可靠性,并且,目标为能够在主机故障中表现稳健的应用可能变得杂乱分布故障恢复代码。其次,负担过重的和受到指定的服务提供者角色可以引起开发其中web服务的使用仅是偶发的自有部署系统和封闭世界环境。背负着这两个缺点,web服务模型的广泛采用可以引起公开web服务接口的应用高度密集、但在它们所要求的部署系统和主控环境方面很大程度上不兼容的软件服务的前景。

[0011] 因此,存在对改善的web服务体系结构的需求。

[0012] 概述

[0013] 在所公开的本主题的一个方面中,用于提供web服务的系统包括被配置为存储一种或多种web服务的服务库。

[0014] 连接到网络的一个或多个服务主机适于接收和履行对存储在服务库中的web服务的部署和取消部署请求。履行部署和取消部署请求可以包括实例化web服务的一个或多个终端。连接到服务主机的主机目录被配置为存储与服务主机相关的数据。管理器被配置为查询主机目录和服务库,生成部署计划,并向服务主机传送部署和取消部署请求。

[0015] 在一种实施例中,服务库可以被配置为从发布者接收web服务。服务主机可以把关于web服务使用的信息传送给管理器。服务主机可以把与一个或多个服务主机相关的数据(例如可用资源)发送给主机目录。

[0016] 在一种实施例中,系统可以包括发现服务,其包括可以被配置为注册和维护对应于每一服务主机的终端的条目的目录。目录服务可以存储在web服务和web服务部署在其上的相应一组终端之间的映射。目录服务可以被配置为接收对应于web服务中的一种的统一资源标识符(URI)请求,并且请求并解析对应于web服务部署在其上的终端中的一个的URL。

[0017] 在所公开的本主题的另一方面中,用于提供web服务的方法包括把一种或多种web服务存储在服务库中,并把与一个或多个服务主机相关的数据存储在主机目录中。该方法包括借助于管理器控制一个或多个服务主机中的每一个上的一个或多个服务终端上的一种或多种web服务的部署和取消部署。控制可以包括查询服务库和主机目录,以便检索关于web服务和发布者的信息,使用该信息和数据来生成部署计划,把部署和取消部署请求传送给服务主机,并响应于部署和取消部署请求实例化web服务的一个或多个终端。

[0018] 在所公开的本主题的另一方面中,用于消费web服务的方法包括把使用者代理绑定到入网点(point of presence)。入网点可以与发现服务通信以便解析对应于与所期望的web服务URL对应的服务终端的URL。入网点可以绑定到服务终端。发现服务与管理器通信,该管理器被配置为查询主机目录和服务库,生成部署计划,并把部署和取消部署请求传送给一个或多个服务主机。

[0019] 附图简述

[0020] 图1是根据所公开的本主题用于提供web服务的系统的实施例的服务提供者部分的示意图。

[0021] 图2是用于根据所公开的本主题提供web服务的系统的实施例的发现服务和提供者部分的示意图。

[0022] 图3是用于根据所公开的本主题提供和消费web服务的系统的实施例的服务使用者、发现服务和提供者部分的示意图。

[0023] 图4是阐释用于根据所公开的本主题提供和消费web服务的方法的实施例的流程图。

[0024] 图5是根据所公开的本主题的实施例在服务使用者和本地入网点之间的示例性交互的示意图。

[0025] 图6是根据所公开的本主题的实施例在发布者和发布库之间的示例性交互的示意图。

[0026] 图7是根据所公开的本主题的实施例在服务主机和主机目录之间的示例性交互的

示意图。

[0027] 图8是根据所公开的本主题的实施例在入网点和活动服务目录之间的示例性交互的示意图。

[0028] 图9是根据所公开的本主题的实施例在活动服务目录和管理器之间的示例性交互的示意图。

[0029] 图10a是根据所公开的本主题的实施例在管理器和服务库之间的示例性交互的示意图。

[0030] 图10b是根据所公开的本主题的实施例在管理器 and 主机目录之间的示例性交互的示意图。

[0031] 图10c和图10d是根据所公开的本主题的实施例在管理器和服务主机之间的示例性交互的示意图。

[0032] 图10e是根据所公开的本主题的实施例在管理器 and 活动服务目录之间的示例性交互的示意图。

[0033] 图11是根据所公开的本主题的实施例的终端部署的示意图。

[0034] 图12是根据所公开的本主题的实施例的服务调用的示意图。

[0035] 图13是根据所公开的本主题的实施例的需求驱动的动态部署的示意图。

[0036] 贯穿各图,除非由上下文以另外方式说明或指示,否则相同的标号和字符用来指示所阐释的实施例的类似的特征、元素、组件或部分。此外,尽管现在将参照各图详细描述所公开的本主题,但也结合预期是非限制性的说明性实施例来详细描述所公开的本主题。

[0037] 详细描述

[0038] 在此所使用的术语“动作者(actor)”可以指负责采取动作的实体。例如,动作者可以包括自动化计算机程序。替代地,“动作者”可以指带有适于受到计算机的操作者控制的用户接口的计算机程序。

[0039] 在此所使用的术语“基础设施”可以共同地指独立于任何特定实现存在于一种操作状态的所描述的体系结构的组件。

[0040] 在此所使用的术语“web服务”可以指在网络上部署软件服务。在此所使用的术语“web服务”不预期限于在因特网上部署软件服务,本领域中的普通技术人员将明白,这样的服务也可以被部署在其他网络例如局域网LAN或类似物上。

[0041] 出于阐释而非限制的目的,描述现在将由结合所公开的本主题用于描述(web服务定义语言(WSDL))、广告和发现(通用描述、发现和集成(UDDI))以及与web服务通信(简单对象访问协议(SOAP))的某些标准组成。然而,本领域中的普通技术人员将明白,可以使用其他合适的标准,例如代表性状态传输(REST),且下列描述不预期限制。

[0042] Web服务可以通过使用SOAP交换消息来进行交互。SOAP定义web服务的通信协议,其独立于编程语言和平台,且可以在广泛的传输协议上使用。例如,在HTTP上的SOAP可以被用作绑定web服务定义的引用协议。

[0043] SOAP消息可以包括外部XML元素(其可以被称为“包封(envelope)”,“包封”定义消息的命名空间)、可选的首部元素(其包括对消息收发框架的任何相关扩展)和所要求的主体元素。主体元素可以提供用于把信息传送给最终的SOAP接收器的机制,但在这一角色之外尚未指定,且既不具有已定义的结构或解释、也不具有指定要进行的任何处理的方式。

[0044] WSDL是用于以XML描述web服务的语言。服务的组件接口描述可以被实现为WSDL文档,该WSDL文档可以包括定位web服务终端、绑定到web服务终端并与其交互所要求的信息。从在它们的操作、每一操作(包括类型定义)的参数和返回值以及用于通信的协议和数据绑定方面描述web服务终端的XML文档元素构建WSDL文档。

[0045] 可以针对终端的WSDL文档编写请求者代理软件,例如借助于自动代码产生工具来以具体语言生成代理或“存根(stub)”代码。这些可以是独立的工具,例如用于Java的“WSDL-Java”,也可以是基于IDE的工具,例如在Visual Studio和Eclipse中,其为以Perl、C++、C#、PHP及其他产生代码的生成器提供可插入框架。

[0046] 出于阐释而非限制的目的,下面参见图1、图2、图3和图4描述所公开的本主题的示例性实施例。

[0047] 在一个示例性实施例中,且参见图1,用于提供web服务的系统可以包括服务提供者组件110。服务提供者组件110可以例如在计算机程序中实现。计算机程序可以被存储在计算机可读介质上,例如CD-ROM、DVD、磁盘、ROM、RAM等等。可以把程序的指令读取到诸如例如连接到网络的服务器或分布式计算机系统等的的一个或多个计算设备中所包括的一个或多个处理器的存储器中。在执行时,程序可以指示处理器控制计算设备的各种组件。尽管程序中的指令序列的执行引起处理器执行在此描述的某些功能,但硬布线电路可以用来代替用于实现目前所公开的本主题的软件指令或者与这些软件指令组合使用。因而,本发明的各实施例不限于硬件和软件的任何特定组合。

[0048] 与示例性实施例结合,服务提供者组件101可以包括例如服务库130,服务库130可以被配置为存储一种或多种web服务。例如,服务库130可以连接到发布者160,发布者160可以把web服务“发布”给服务库130以供存储。连接到网络的服务器147可以适于接收和履行对存储在服务库130中的web服务的部署和取消部署请求。例如,履行部署和取消部署请求可以包括实例化web服务中的一种的一个或多个终端143。主机目录120可以连接到服务器147并被配置为存储与一个或多个服务器147相关的数据。管理器150可以被配置为查询主机目录120和服务库130,生成部署计划,并把部署和取消部署请求传送给服务器147。

[0049] 服务器147可以被配置为把关于web服务的使用的信息传送给管理器150。服务器147也可以被配置为把与它们的可用资源相关的数据发送给主机目录120。

[0050] 通过把提供web服务的责任划分成多个独立的角色、减少对每一动作者所要求的领域特定的知识量并降低参与提供web服务的壁垒,这一示例性实施例可以减少参与者在web服务生命周期的复杂性。例如发布、部署和主控web服务的任务被看作是不同的独立活动。因而,出于阐释而非限制的目的,服务提供者组件101可以被看作是不同的三个“动作者”:发布者160、管理器150和服务器147。另外,两个体系结构实体:被称为服务库130和主机目录120的储存库。

[0051] 不是明确地部署,web服务提供者代理实现可以改为由发布者160描述,然后发布者160通过把它存储在被称为服务库130的储存库中来把它“发布”到基础设施中。这种方法通过把服务实体化与实际实现分开来表示web服务部署的方法——即,web服务的生命周期可以是在它被发布时开始,而不是在它被部署时开始。

[0052] 为了参与基础设施,服务器147可以通过描述它们的可用资源并向被称为主机



目录120的目录注册来注册它们主控web服务的意愿。服务主机147可以指示它们的可用web服务部署容器,并指定发布者160的列表(它们愿意部署这些发布者160的web服务提供者代理实现)。不是通过广告它们的静态部署的服务而是通过广告它们的主控能力、加入可以由管理器150根据需要动态地使用(且回收)以便满足改变的需求水平的共享的潜在主控资源池,服务主机147因而可以参与基础设施。

[0053] 管理器160可以负责管理单种web服务(对于该web服务,在任何给定的时刻,可以存在零个或更多个终端140)的提供水平。为了制定(enact)部署,管理器160可以首先查询服务库130和主机目录120,然后通过把web服务实现与有适当能力的服务主机147配对来创建部署计划。管理器150可以把部署请求(433)发送给服务主机147,然后,服务主机147可以负责实例化web服务的终端140(或拒绝请求)。服务主机147可以把关于在它们的域内部署的每一web服务终端的使用的信息提供给每一web服务的管理器150。管理器150可以使用这种使用数据来做出关于它们管理的必要的web服务的提供水平的判决。

[0054] 在另一示例性实施例中,且参见图2,用于提供web服务的系统可以包括发现服务组件210。例如且以与服务提供者组件110类似的方式,发现服务组件210可以在计算设备上执行的计算机程序中实现。这种计算机设备可以是与服务提供者组件110物理上分离的设备,经由诸如因特网等的网络耦合到服务提供者组件110。替代地,计算机设备主控的发现服务组件210可以是相同的计算机设备主控的服务提供者组件110。值得注意的是,这些计算设备和在此描述的所有其他计算设备在本发明的精神和范围内都可以被实现为单个设备、设备的群集、网络等等。

[0055] 发现服务组件210可以包括活动服务目录220。例如,活动服务目录220可以被配置为存储在被存储在服务库130中的web服务和web服务部署在其上的相应一组终端143之间的映射。

[0056] 活动服务目录220可以保存在单种web服务和web服务的一组活动终端143之间的映射。体系结构中的中央实体,活动服务目录220中所保存的集合映射可以表示从所有参与者视角来看的基础设施的当前状态。因为所有基础设施参与者可以依靠于活动服务目录220来定位它们所期望的web服务的终端143,对于不具有当前所部署的终端的web服务,这种目录可以促进自动部署过程。

[0057] 活动服务目录220可以提供添加、移除和定位(450)web服务的活动终端143的操作。如果活动服务目录220接收对具体的web服务的活动终端的查找请求,但不存在这样的条目,则活动服务目录220可以主动定位所请求的web服务并告知管理器150。管理器可以使用先前所描述的部署过程来动态地发起新的终端143的部署。管理器150可以负责插入和维护它管理的web服务的任何新近部署的终端143的所有活动服务目录220条目。例如,如果,对其web服务的需求下降到零,则管理器150可以决定取消部署已部署终端143中的一个(或全部)。如果终端143被取消部署,则管理器150可以从活动服务目录220移除该终端条目,确保目录保持最新,且尽可能精确地反映基础设施的当前状态。

[0058] 在另一示例性的实施例,且参见图3,用于提供web服务的系统可以包括服务使用者组件310。例如且以与服务提供者组件110类似的方式,服务使用者组件310可以在计算设备上执行的计算机程序中实现。这种计算机设备可以是与服务提供者组件110物理上分离的设备,经由诸如因特网等的网络耦合到服务提供者组件110。替代地,计算机设备主控的

服务使用者组件310可以是相同的计算机设备主控的服务提供者组件110的一部分。

[0059] 服务使用者组件310可以包括使用者代理320和入网点330。例如,发现服务210可以被配置为从服务使用者组件310接收到对应于服务库130中的web服务中的一种或多个的URI,并且请求并解析对应于已部署的web服务的终端143(如果有的话)的一个或多个URL。结合某些实施例,具体的web服务的多个实现可以被发布给服务库130,其中的每一个都对应于相同的URI。例如,具体的web服务可以使用多种不同的语言来实现和/或要求一个或多个不同的部署环境。每一实现可以被存储在服务库130中,且可以对应于相同的URI,且可以实现相同的接口,以使得对特定实现中的任何一个的调用可以产生相同的返回。使用者代理320可以被绑定到入网点330。

[0060] 入网点330可以与发现服务通信,且可以使用URL来绑定到服务终端143。入网点330可以被配置为对服务的位置进行抽象,同时保持使用者代理320的观点,由此充当供使用者代理进入到web服务的实例化的网关。

[0061] 结合这一示例性实施例,所公开的技术不要求使用URL来描述web服务,这是由于URL可以随时间变得无效。Web服务替代地可以用URI标识,抽象地描述在任何时刻可以具有零个或更多个活动终端的服务。通过引入代替它们执行定位web服务终端143并绑定到web服务终端143的机制,服务使用者组件310可以免除这些任务。

[0062] 这种技术可以提出用于使用web服务有限中介框架,其可以从调用请求中所包含的URI透明地解析活动终端URL。这种框架可以被实现为驻留在服务使用者组件310处的代理机制,该代理机制可以充当进入到体系结构的实例化的网关——“入网点”,如上面所描述且在图3中所叙述的。使用者代理320软件可以被编写为绑定到这一本地入网点330,并使用所期望的web服务的URI来调用web服务操作。通过从活动服务目录220检索web服务的活动终端143的列表、选择终端以供使用、代替服务使用者310调用所请求的操作并返回任何结果,入网点330可以负责根据URI透明地解析URL。它也可以负责透明地检测web服务和它们部署在其上的服务主机147的故障并从中恢复,并且负责通过(根据本地策略)再次尝试替代的终端143来主动从这些故障恢复。不可恢复的错误可以被返回给服务使用者310,指示给定基础设施可用资源,当前不可能履行请求。

[0063] 入网点330可以对服务的位置进行抽象,同时仍然保持当前的服务使用者组件310对系统的观点。它可以简化使用者代理320应用的创建,例如通过允许开发者针对web服务执行什么而不是它处于何处或当前是否部署来编程。进一步,因为它使用要求与发现服务210交互的所有任务,入网点330可以提供在用于基础设施的具体标准版本上的抽象层(例如,UDDI版本)。这可以提供在面对不断进化的标准时针对过时的屏障,同时允许使用者代理应用可以在使用不同的标准的环境之间迁移。

[0064] 在一种示例性实施例中,且参见图4,用于提供web服务的方法可以包括把一种或多种web服务存储在服务库中(410)。例如,服务库可以从发布者接收(411)所发布的web服务。与一个或多个服务主机相关的数据可以被存储(420)在主机目录中。例如,主机目录可以接收(422)与服务主机相关的数据,例如关于可用资源和服务主机乐意主控来源于哪些发布者的web服务的数据。管理器可以控制(430)在服务主机中的每一个上的服务终端上的web服务的部署和取消部署。例如,管理器可以查询(431)服务库和主机目录,以便检索关于web服务的信息以及与服务主机相关的数据。然后,使用关于web服务的信息以及与服务主

机相关的数据,管理器可以生成(432)部署计划。然后,管理器可以把部署和取消部署请求传送(433)给服务主机。然后,服务主机可以响应于部署请求实例化(434)web服务,或响应于取消部署请求去实例化web服务。可以从服务主机把使用数据及其他信息反馈440回到管理器。

[0065] 管理器可以把关于web服务被部署在其上的终端的信息(包括活动服务目录)发送(450)给发现服务。例如,通过本地入网点,服务使用者可以把URI请求发送(460)给目录服务。结合管理器,响应于URI请求,发现服务可以提供URL响应(470),且服务使用者可以绑定到主控所期望的web服务的终端。

[0066] 出于阐释而非限制的目的,参见图5到图10,现在将对在此公开的系统的某些组件之间的示例性交互做出描述。

[0067] 根据所公开的本主题的实施例,且参见图5,服务使用者310不需要直接与web服务终端交互。例如,服务使用者310可以绑定到本地入网点330并对其调用操作。POP可以与服务使用者310共同定位;使用者代理320可以被编写为静态地绑定到web服务的URI,例如在前面加上本地的URI的协议、主机名和端口。

[0068] 例如,图5中阐释调用由URI“ServiceA(服务A)”标识的web服务的操作“x”。使用者代理应用可以绑定到POP并对其调用操作,然后,POP可以定位所请求的web服务的终端。POP可以通过调用“findAll(查找全部)”操作且用URI“ServiceA”作为参数来把查找请求发送给活动服务目录220,且可以接收回描述ServiceA的当前活动终端540和550的ServiceInstanceDescriptor(服务实例描述符)的列表(541、551和552)。POP可以应用本地策略以便选择在连接到终端并代表服务使用者调用操作“x”之前使用哪一终端。操作结果可以被返回到POP且最终被返回给使用者代理320应用。

[0069] 根据所公开的本主题的实施例,且参见图6,发布者160可以是(尽管不必是)与服务库130交互以便发布和取消发布web服务的实现的服务使用者。对于其他服务使用者(例如,310),发布者160可以包括使用者代理610和入网点620,且不需要直接地与任何外部的web服务交互,代替地经由本地入网点620调用所期望的操作。

[0070] 例如,为了发布web服务的实现(例如用ServiceImplementationDescriptor(服务实现描述符)描述的),发布者160可以绑定到本地入网点610并调用服务库130的带有所期望的ServiceImplementationDescriptor的“publishServiceImplementation(发布服务实现)”操作。一旦已经发布了实现,则它可以变得立即可用、可以被按需部署在有能力的服务主机上,以便满足需求。

[0071] 根据所公开的本主题的实施例,且参见图7,服务主机147可以有助于服务提供者的角色并且也充当服务使用者两者(尽管不必如此)。对于其他服务使用者(例如,310),服务主机147可以包括使用者代理710和本地入网点720。服务主机147可以例如通过使用HostDescriptor(主机描述符)描述它们自身来向主机目录130注册自身、绑定到本地入网点720并调用主机管理者130的“addAuthorizedPublisher(添加已授权发布者)”操作。通过本地入网点720进行操作,因而可以借助于由基础设施提供给所有服务使用者的相同的调用过程来执行服务主机147注册过程。一旦注册,服务主机的HostDescriptor就可以被添加到共享的主控资源池,准备好按需进行使用以便满足需求。

[0072] 根据所公开的本主题的实施例,且参见图8,与任何服务使用者310相关联的入网

点330可以是透明终端解析和故障恢复机制,其可以代表服务使用者310定位web服务、绑定到web服务并对其调用操作。例如,为了定位web服务的活动实例,入网点可以与被称为“活动服务目录”810的web服务交互,以所期望的web服务的URI作为参数调用其“findOne(查找一个)”或“findAll”操作。

[0073] 为了参与体系结构的实例化,本地入网点可以具有至少一个活动服务目录终端810的先验知识。可以在活动服务目录220中在web服务URI“活动服务目录”下列出任何替代的活动服务目录终端,且可以使用“findOne”和“findAll”操作来定位任何替代的活动服务目录终端。POP可以周期性地检索和存储替代的活动服务目录终端的列表,以便在故障时使用。

[0074] 根据所公开的本主题的实施例,且参见图9,在活动服务目录220接收到对具有零个活动终端的web服务的查找请求时,它可以定位该web服务的管理器150的实例,使得可以部署终端。通过绑定到其本地入网点,并且使用负责最初所请求的web服务的管理实体920的URI调用活动服务目录“findOne”或“findAll”操作,活动服务目录220可以利用其自己的查找设施。例如,通过以模式“WebServiceURI\_ManagerURL(web服务URI\_管理器URL)”,把“管理器”URI联系到所请求的web服务的URI,可以构建这一URI。可以把活动服务目录返回给描述实现“管理器”接口的web服务终端的ServiceInstanceDescriptor。

[0075] 由于管理器150可以负责维护它们管理的web服务的目录记录,管理器150可以把它们部署的任何终端的记录插入到活动服务目录220。管理器150可以例如借助于“requestFirstInstance(请求第一实例)”调用来带内执行这一操作——如果成功地完成了整个部署和注册过程就返回“true(真)”,否则返回“false(假)”。返回“false”的“requestFirstInstance”操作可以指示,给定系统的当前状态,不可能部署web服务的新实例。这种情况可以表示可以被返回给请求实体(即,服务使用者)的不可恢复的错误。如果操作返回“true”,则活动服务目录220可以再次执行最初所请求的查找操作并把新近插入的ServiceInstanceDescriptor返回给请求实体。

[0076] 根据所公开的本主题的实施例,且参见图10,根据后续的示例性和非限制性描述,管理器150可以与包括服务库130、主机目录120、一个或多个服务主机147和活动服务目录220的各种组件通信。例如,参见图10a,管理器150可以调用服务库130的操作以便检索正在部署的web服务的一组实现。这些实现都可以用ServiceImplementationDescriptor来描述;稍后可以由管理器150使用这一描述符的元素来为部署选择合适的候选实现。

[0077] 以类似的方式,参见图10b,管理器150可以在部署过程期间调用主机目录120的操作。管理器150可以使用主机目录120来检索标识服务主机147部署由所指示的发布者160(例如用唯一PublisherID(发布者ID)标识)发布的web服务实现的意愿的一组HostDescriptor。

[0078] 管理器150可以把从服务库130中检索的ServiceImplementationDescriptor与从主机目录120检索的HostDescriptor进行比较,以便基于本地策略制定合适的候选部署计划。一旦选择了计划,管理器150就可以联系所选择的服务主机147并发起部署过程。

[0079] 参见图10c,在选择合适的web服务实现(用ServiceImplementationDescriptor描述)以便部署在所选择的服务主机147(用HostDescriptor描述)上之后,管理器150可以绑定到HostDescriptor中所包括的服务主机147,且可以直接地调用“部署”操作。一旦成功部

署,服务主机147就可以给管理器150返回描述新近部署的web服务终端的ServiceInstanceDescriptor。然后,管理器150可以在活动服务目录220中注册这一ServiceInstanceDescriptor,完成部署过程。

[0080] 相反,参见图10d,对于在它们的域(例如,在它们控制下的容器)内部署的每一web服务终端25,服务主机147可以向web服务的管理实体报告使用数据。服务主机147可以绑定到本地入网点720,且可以调用通过把“管理器”URI联系到报告其数据的web服务的URI来标识(例如,“WebServiceURI\_ManagerURI”)的web服务的“reportUsageData(报告使用数据)”操作。“reportUsageData”操作可以返回指示在下次报告这一具体web服务的使用数据之前服务主机147应等待的时间长度的数值。服务主机147的本地策略可以规定,使用数据返回得比所请求的更早(例如,由于诸如工作存储器等的本地资源约束);比所请求的周期显著更晚地返回数据可以向管理器150指示,服务主机147存在问题——管理器150可以对其采取动作以便有效地管理其服务的可用性的信息。

[0081] 参见图10e,在web服务的管理器150成功地部署或取消部署该web服务的终端时,它可以在活动服务目录220中添加或移除终端引用。管理器150可以绑定到本地入网点,并使用相关的ServiceInstanceDescriptor来调用活动服务目录220web服务的“addActiveService(添加活动服务)”或“removeActiveService(移除活动服务)”操作。

[0082] 例如,管理器150可以负责维护它们管理的web服务的活动服务目录条目。在履行服务提供者职责的所有实体中,管理器150可以被视为对维护它们各自管理的web服务的精确公共记录最感兴趣,这是由于它给它们提供了精确使用数据,在履行它们管理web服务的提供水平的职责时,它们可以这些精确使用数据做出更明智的判决。

[0083] 示例

[0084] 还通过下面呈现的示例来描述本申请。这样的示例的使用仅仅是说明性的,且无论如何不限制本发明或任何所例示的术语的范围和含义。同样地,本申请不限于在此描述的任何具体优选实施例。实际上,在阅读本说明书之后,本领域中的技术人员将明显看出本发明的多种修改和变更。应当通过所附权利要求以及权利要求所赋予的等效物的全部范围来理解本发明

[0085] 在这一示例中,且参见图11、图12和图13,利用先前所描述的示例性组件交互作为终端部署、服务调用和需求驱动的动态部署的单个步骤用于被称为“DayTime(日时)”的web服务的提供和调用,来描述各个过程。

[0086] 如图11中,新的web服务终端的部署可以由管理器150执行。管理器150可以与服务库130、主机目录120、一个或多个服务主机147和活动服务目录220交互,以便使得新的“DayTime”web服务1100终端1120可用。

[0087] 出于阐释而非限制的目的,为了部署新的web服务终端,管理器150可以首先创建和选择有效的部署计划(部署计划包括,例如,由HostDescriptor中所描述的“能力”匹配其“要求”的ServiceImplementationDescriptor)。管理器150可以从联系服务库130和检索一组所有所发布的“DayTime”web服务实现来开始。可以用ServiceImplementationDescriptor1130来描述所返回的每一实现,每一ServiceImplementationDescriptor1130可以具有“PublisherID”元素1131。对于每一实现,管理器150可以使用PublisherID1131来联系主机目录120,并检索想要部署由所指定的

发布者编写的提供者代理应用的一组服务主机。然后,管理器150可以创建一组有效的部署计划1140,并基于本地策略从候选物中选择一个。接下来,管理器150可以执行部署计划,联系由所选择的HostDescriptor(主机描述符)1150描述的服务主机140并用所选择的ServiceImplementationDescriptor1130来调用其“部署”操作。一旦成功部署,目标服务主机147就可以给管理器150返回描述终端1120的ServiceInstanceDescriptor1160。管理器150可以通过把ServiceInstanceDescriptor1160插入到活动服务目录220来完成部署。

[0088] 一旦在活动服务目录220中列出新近部署的终端,它可以被视为已经部署,准备好由服务使用者定位并由服务使用者调用其操作。

[0089] 参见图12,服务使用者310的服务调用可以包括在本地入网点330上绑定和调用web服务操作。一旦接收到调用请求,本地入网点330就可以负责定位目标web服务的活动终端并代表服务使用者调用所请求的操作,例如“DayTime”web服务1110的“getTime(获取时间)”操作。

[0090] 一旦从使用者代理320应用接收到调用请求,本地入网点330就可以首先从传入的请求提取所期望的web服务URI1210。如果入网点330实现终端高速缓存,则它可以首先本地检查先前所检索的终端列表。如果不存在这样的条目,则可以使用活动服务目录220检索新的列表。一旦检索到终端的列表,则入网点可以选择一个以便根据本地的选择策略使用。然后,入网点330可以准备调用请求(包括可能的对消息的修改)、绑定到所选择的web服务终端1120并转发调用请求。如果由于任何原因调用失败,则可以对活动终端的剩余部分尝试相同的过程,并且如果所有终端都被证明是不可用,则可以把一般性错误(generic error)返回给服务使用者310。如果调用成功,则可以首先准备所得到的响应(再次,可能涉及对消息的修改),然后返回给服务使用者310并结束调用过程。

[0091] 参见图13,在动态需求驱动的系统,终端部署可以响应于现有的或所预测的事件(例如包括调用当前不存在活动终端的web服务的操作)而发生。在这样的环境中,可以执行上面参见图11和图12所公开的本主题。例如,在活动服务目录220接收到对“DayTime”web服务1110的所有终端1120的请求时,它可以发现当前都还没有部署。以此方式,通过联系该web服务的管理器150并调用“requestFirstInstance”操作,活动服务目录220可以发起(尽管并不必然制定)所请求的web服务的第一实例的部署。在这一点上,可以如上所述地执行部署过程;当管理器150从“requestFirstInstance”操作时返回,活动服务目录220可以再次执行查找并返回对新近部署的终端1120的引用或一个错误,该错误指示,给定基础设施中的当前可用资源,不可能在给定时刻履行请求。

[0092] 如在此描述的,为了使得web服务是可部署的,不需要当前就部署web服务的管理器150。如果被实现为web服务,且借助于服务库130中所发布的一般和/或定制的服务专用的管理器实现,则可以使用与管理器用来管理它们的具体web服务的那些部署机制相同的部署机制来部署和管理web服务的管理器。这种递归的、可伸缩的模型可以允许基础设施“逐渐增加(wound up)”以提供足够的web服务终端以便满足需求,且然后,在需求下降到零时“逐渐减少(wound down)”到零资源消耗的状态。

[0093] \*\*\*

[0094] 如上面结合某些实施例所描述的,某些组件例如110、210和310,可以包括计算机或多个计算机、处理器、网络、移动设备、群集或执行各种功能的其他硬件。此外,所公开的

本主题的某些元素可以以计算机可读的代码实现,该计算机可读的代码可以被存储在计算机可读介质上,且在被执行时,引起处理器执行某些功能。在这些实施例中,计算机扮演准许用于提供和/或消费web服务的系统和方法的重要角色,以便在网络例如因特网上提供这样的服务。例如,计算机、处理器、存储器、存储和联网硬件的存在提供了部署、消费和管理web服务环境并且基于所测量的使用者需求动态地应用和回收资源的能力。

[0095] 另外,如上面结合某些实施例所描述的,某些组件可以例如经由网络(例如因特网)与某些其他组件通信。倘若上面没有明确说明,所公开的本主题预期包含每一事务的双方,包括发送和接收。本领域中的普通技术人员将容易地理解,对于以上所描述的特征,如果一个组件传送、发送或以另外方式使其对另一组件可用,则无论是否明确地说明,其他组件都将接收或获取。

[0096] 目前,所公开的本主题的范围不受在此的特定实施例的限制。事实上,本领域中的技术人员从前面的描述和附图将明显看出除了在此描述的那些之外的所公开的本主题的各种修改。这样的修改预期落在所附权利要求的范围内。

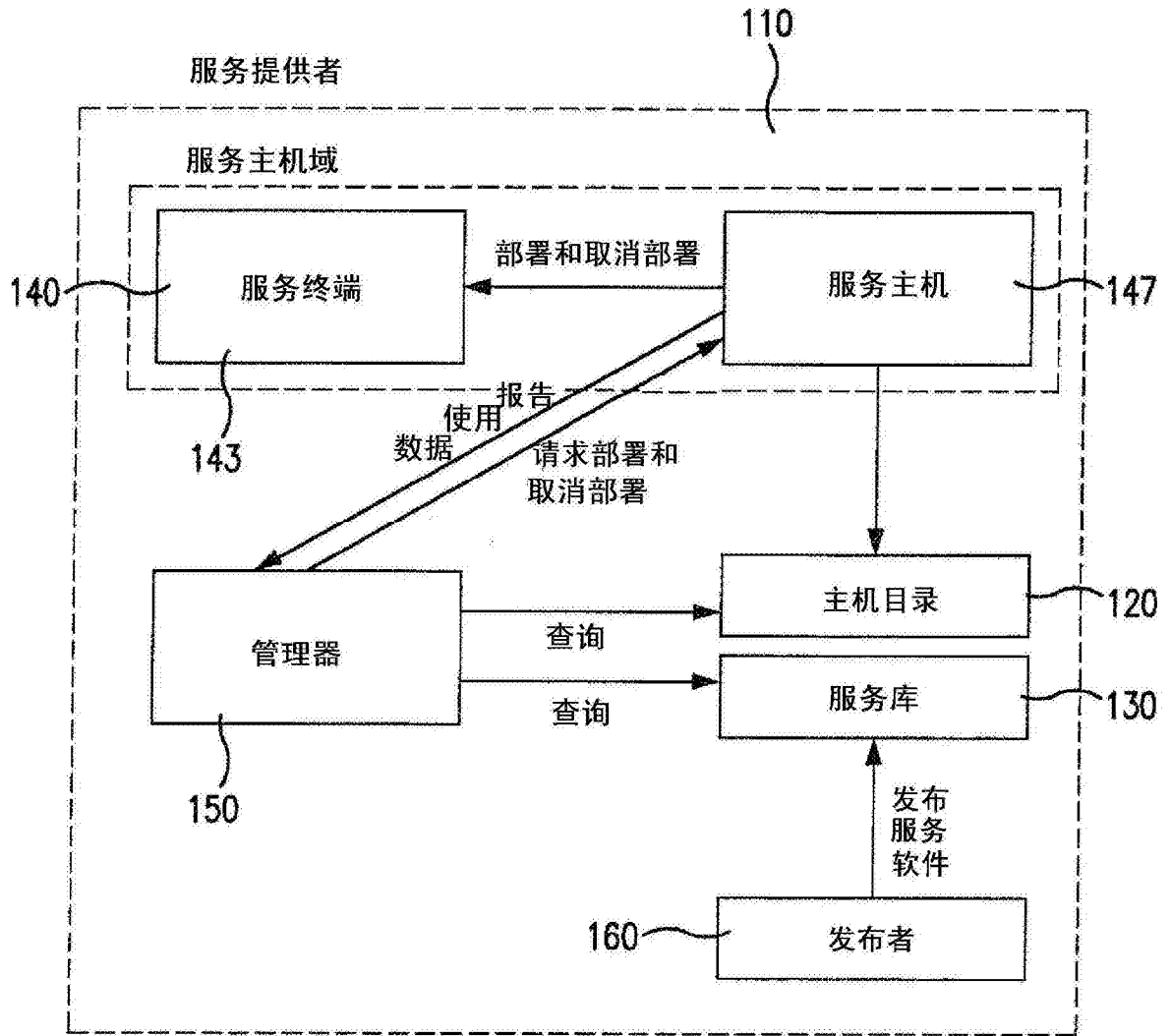


图1



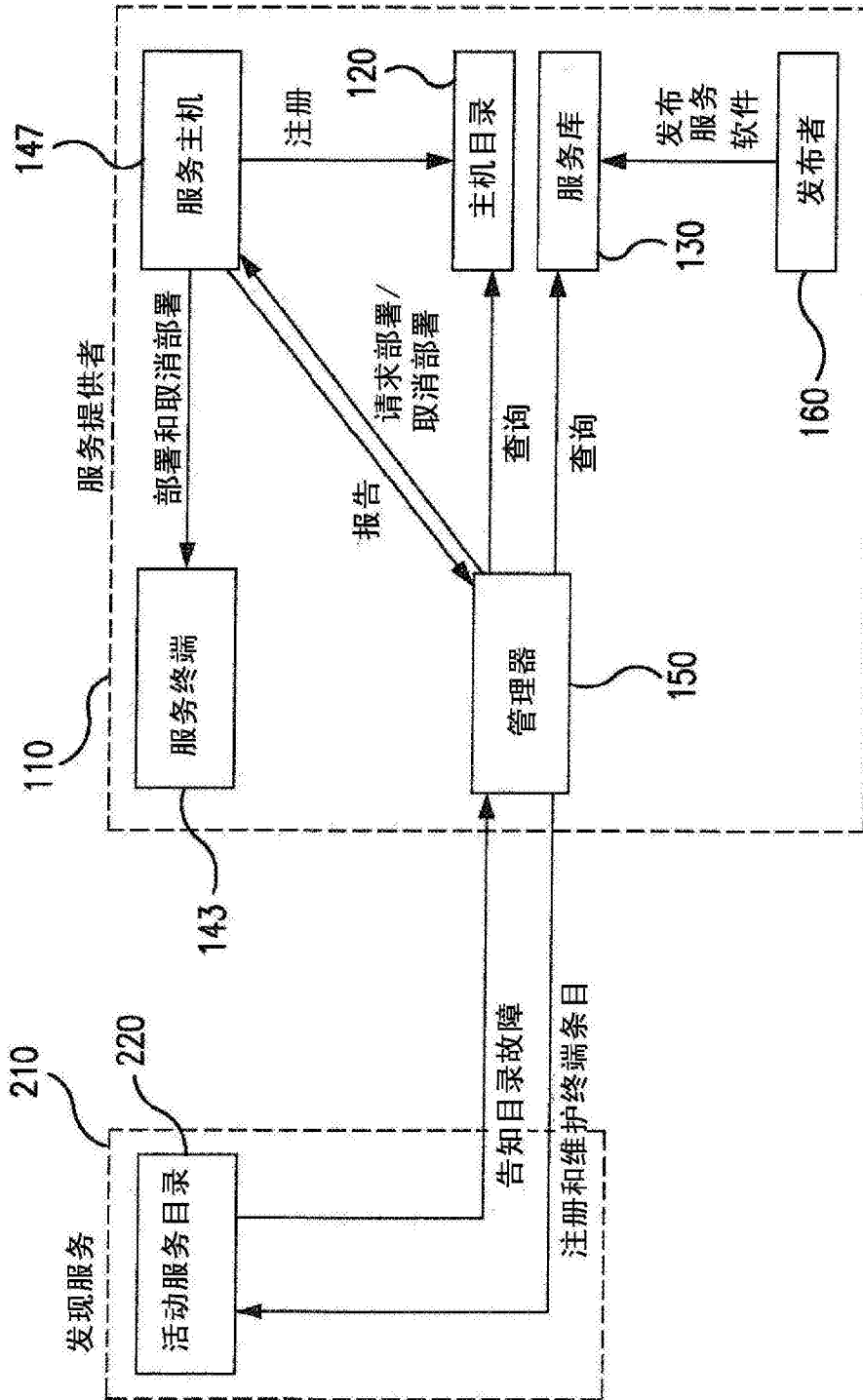


图2

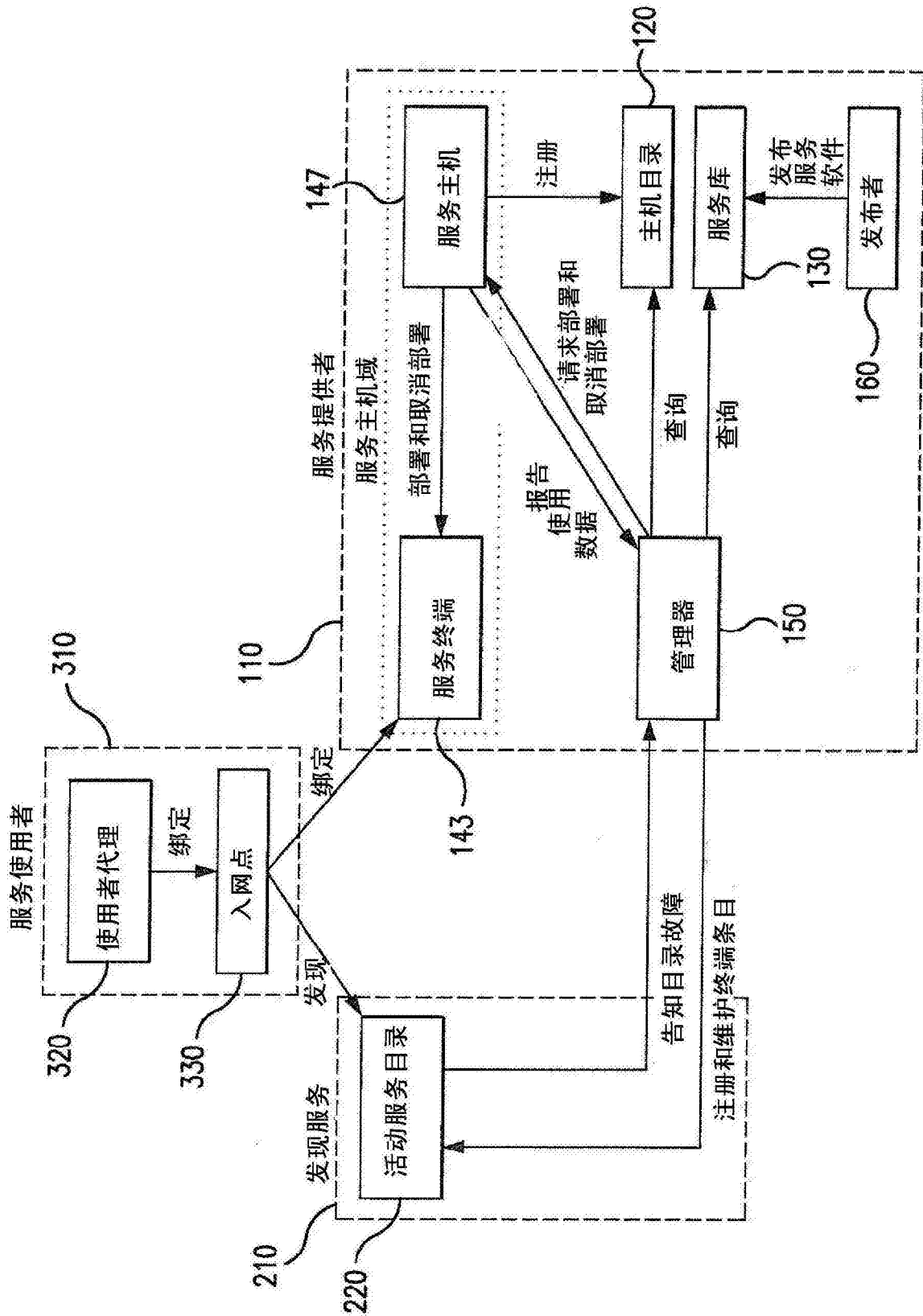


图3

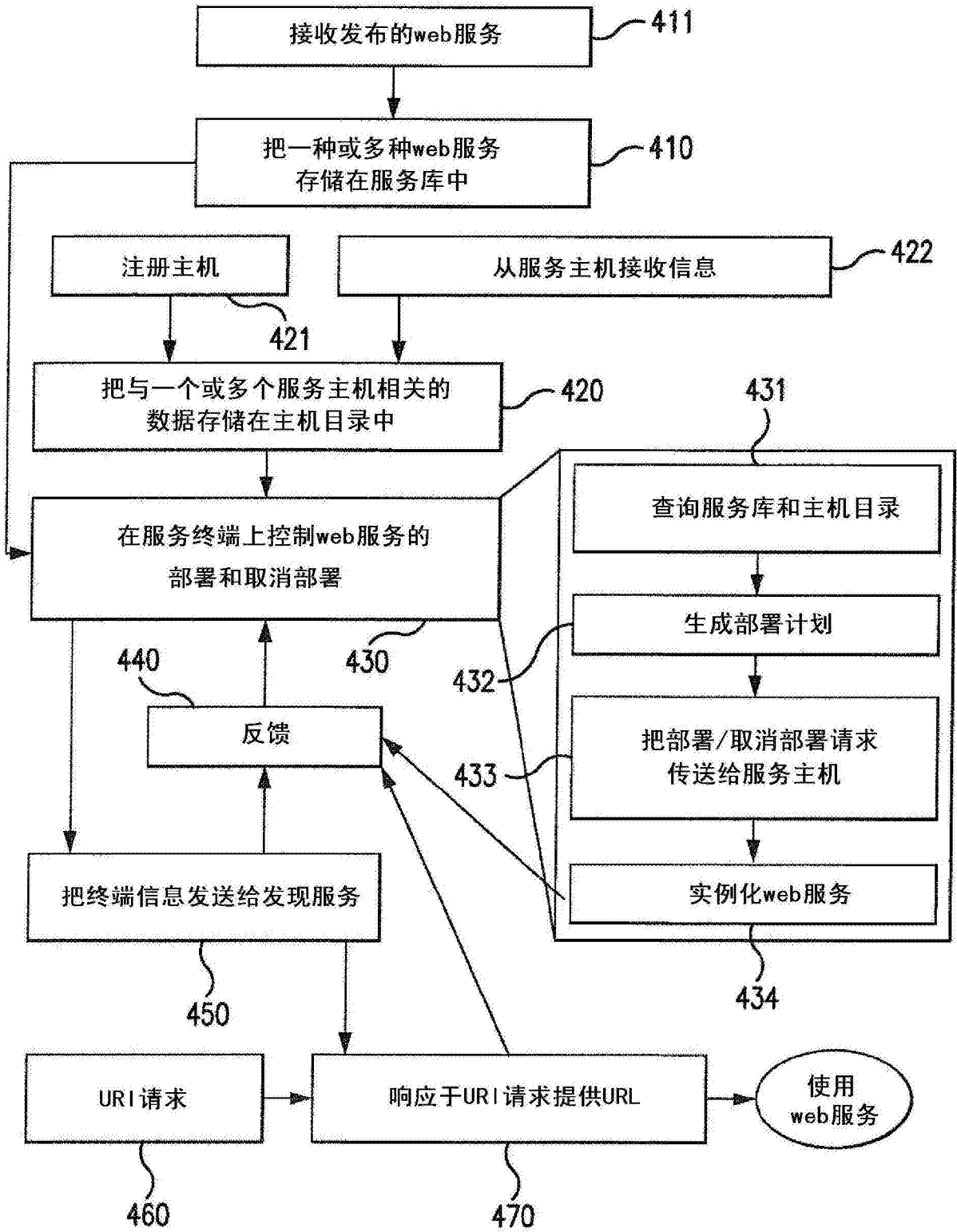


图4

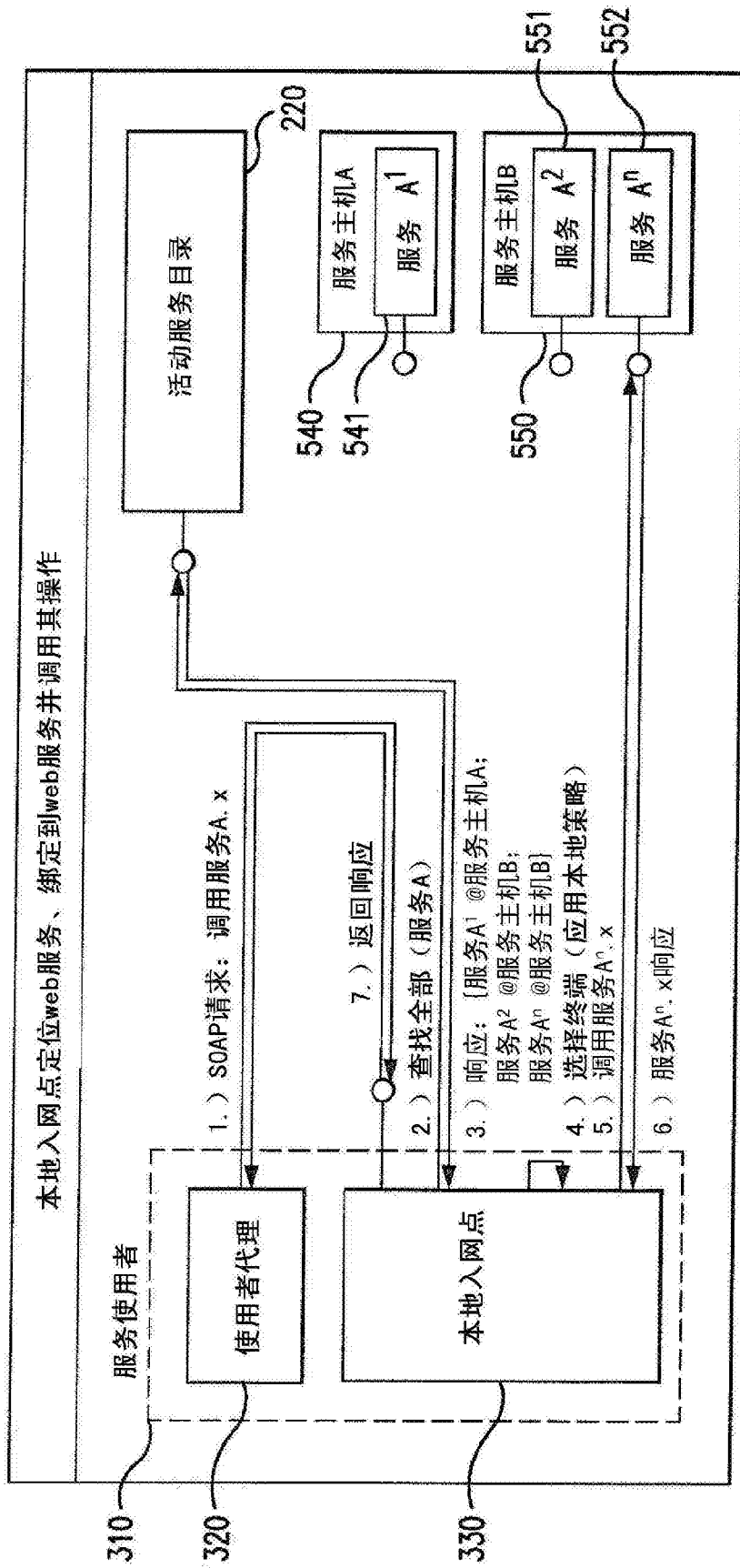


图5

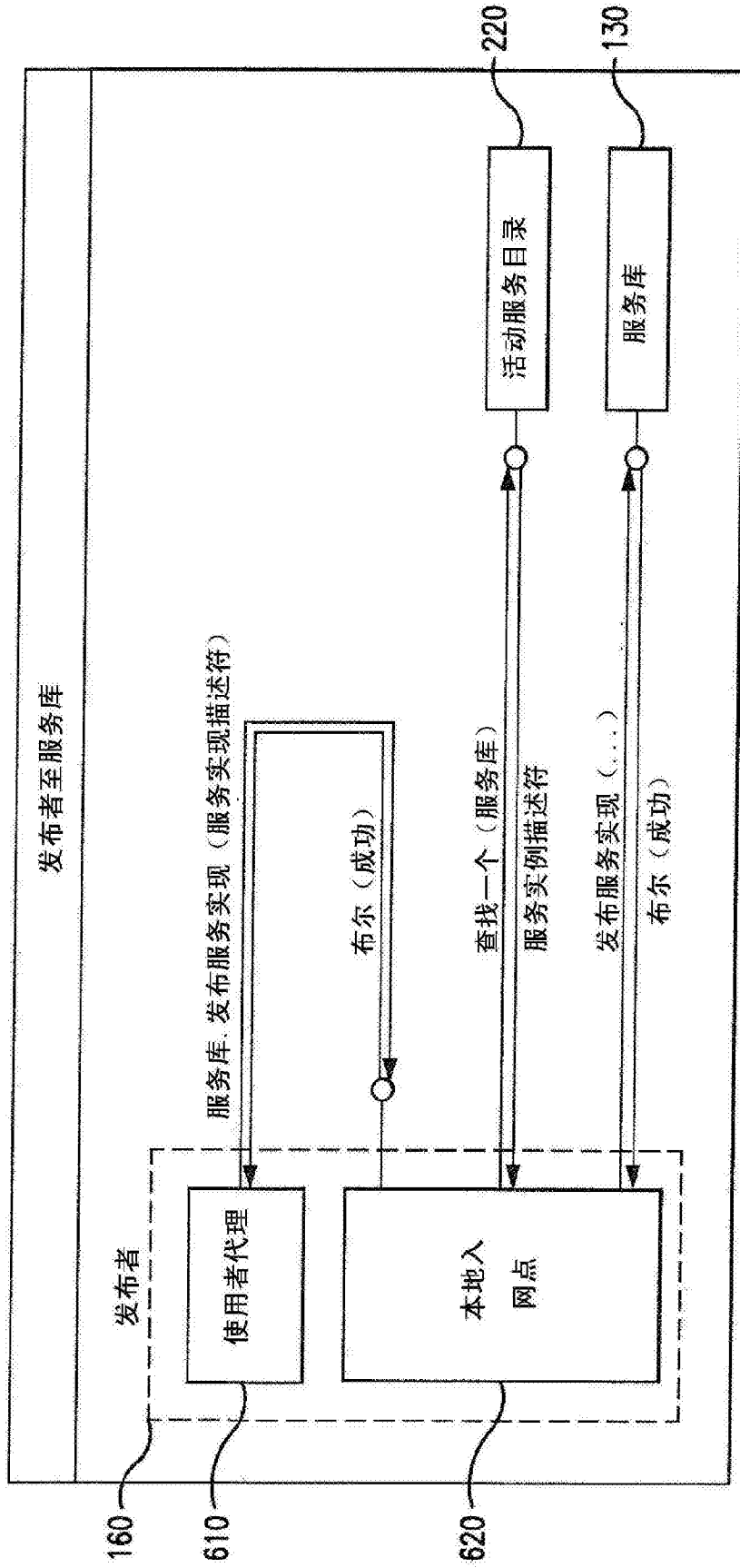


图6

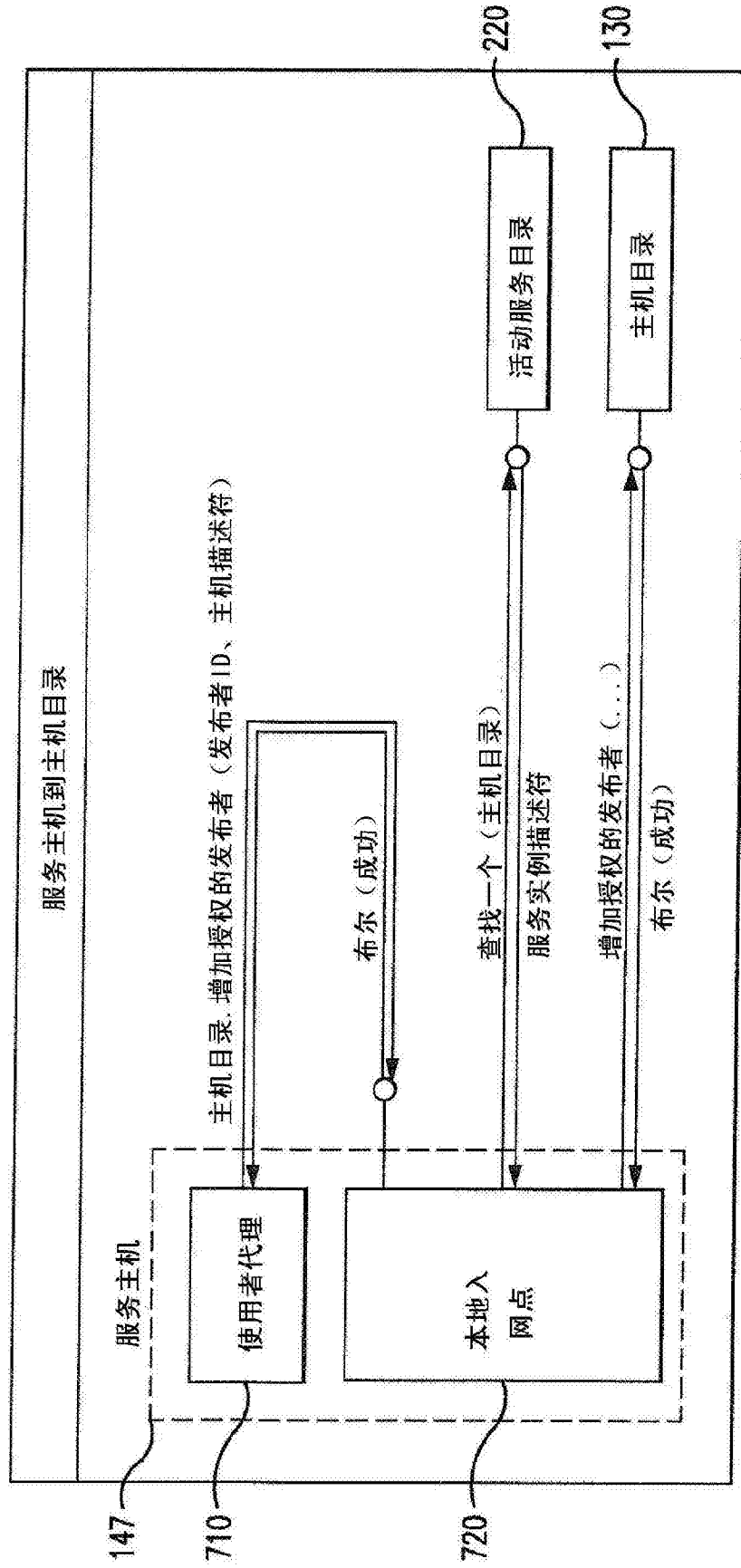


图7

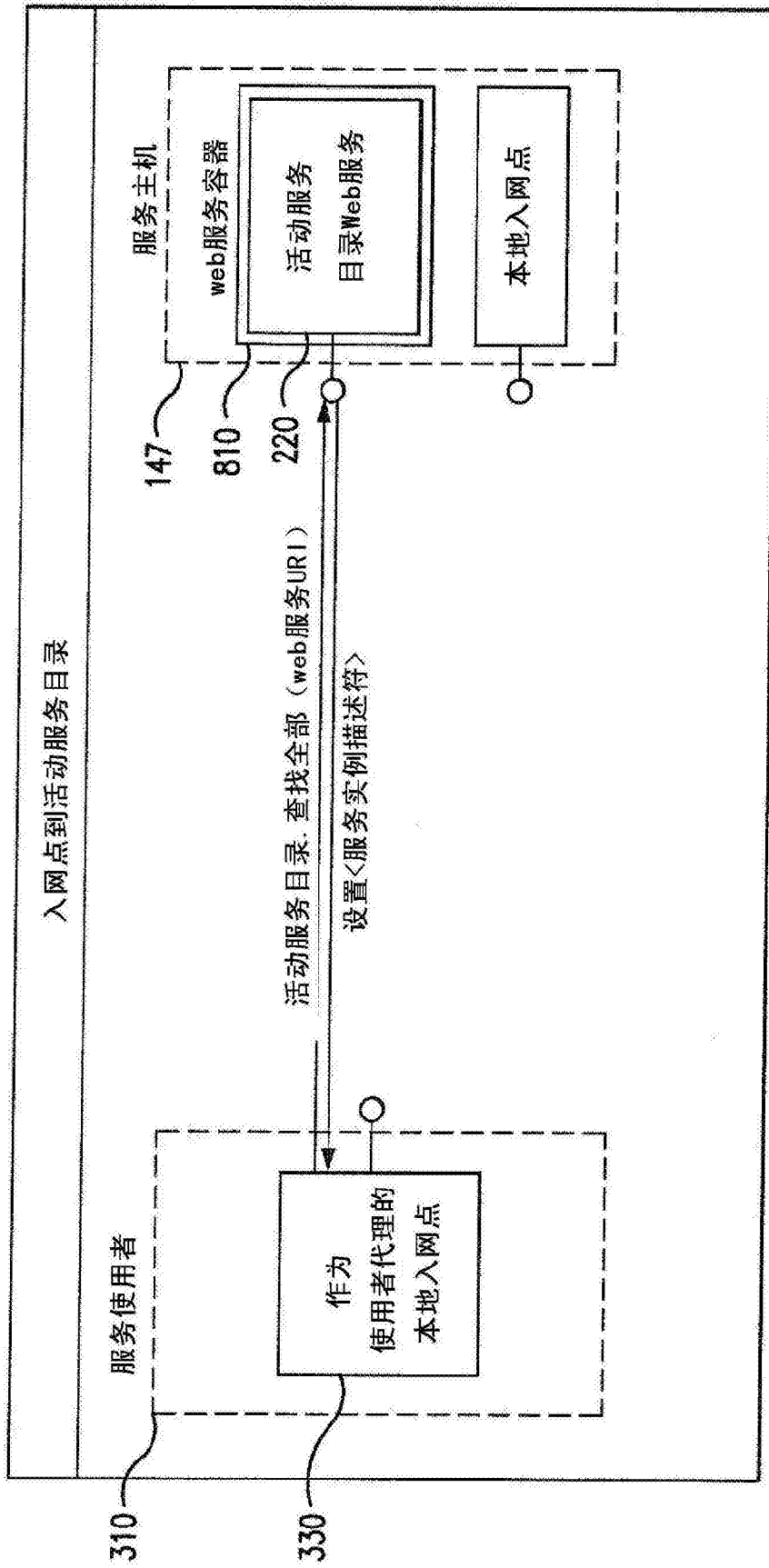


图8

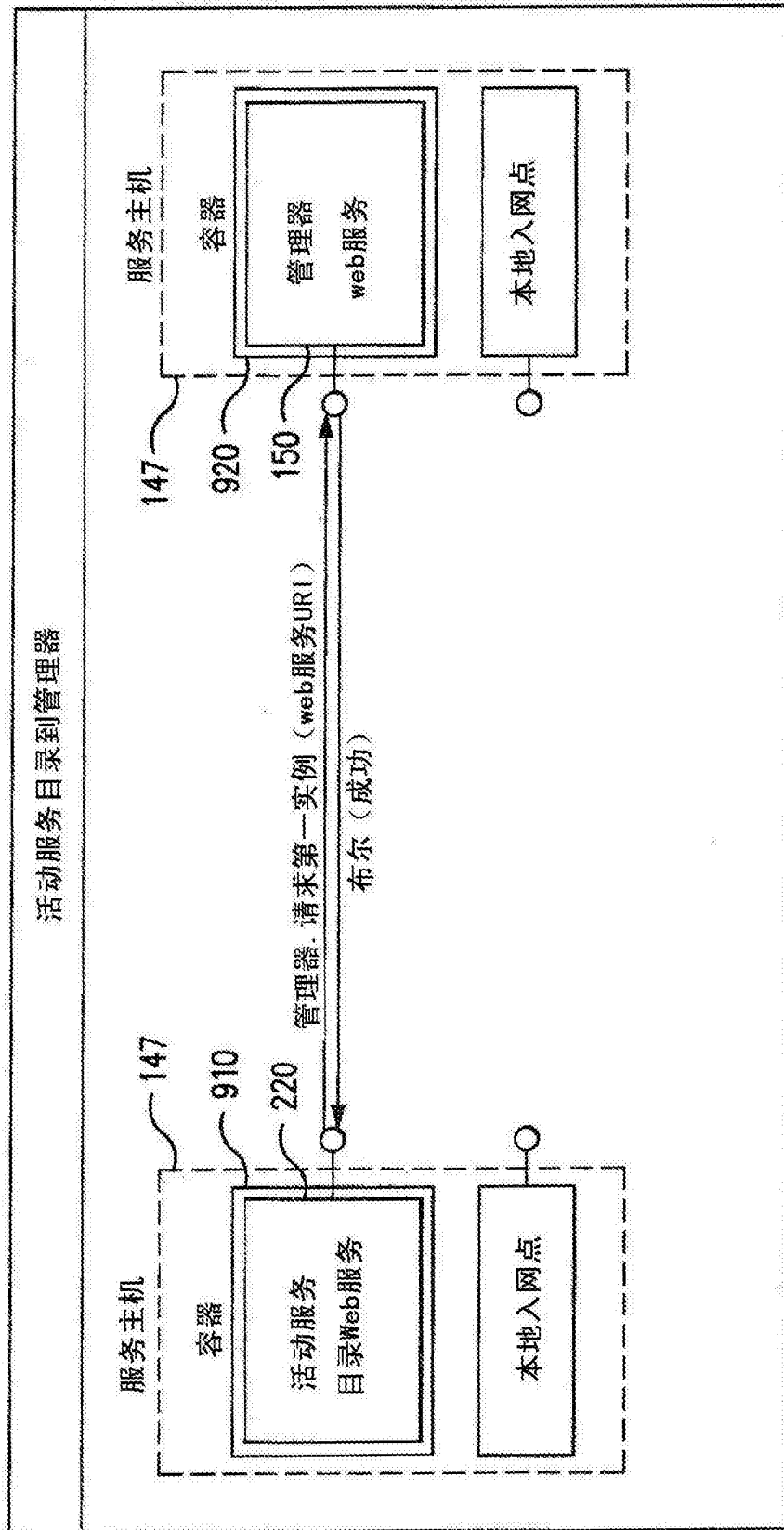


图9



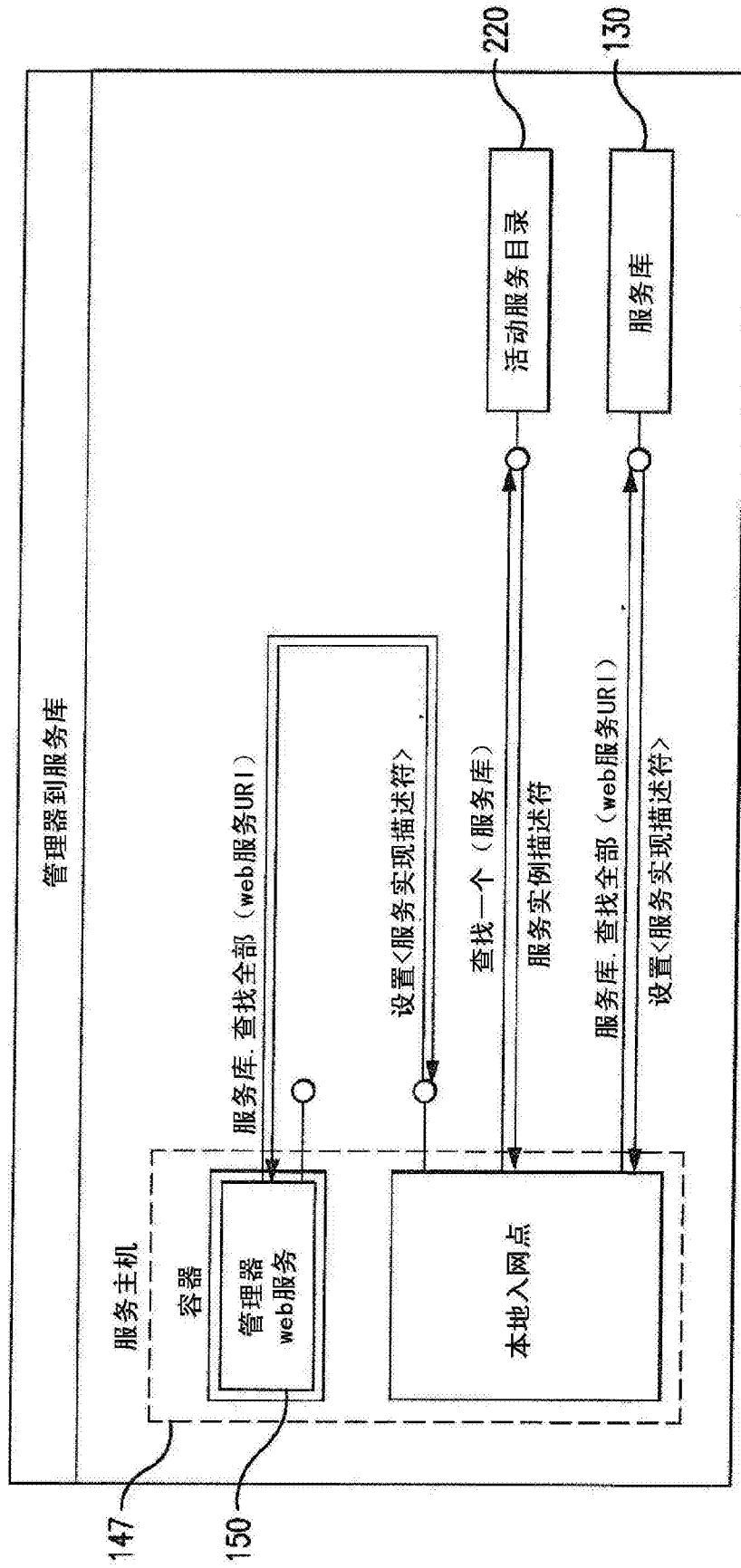


图10a

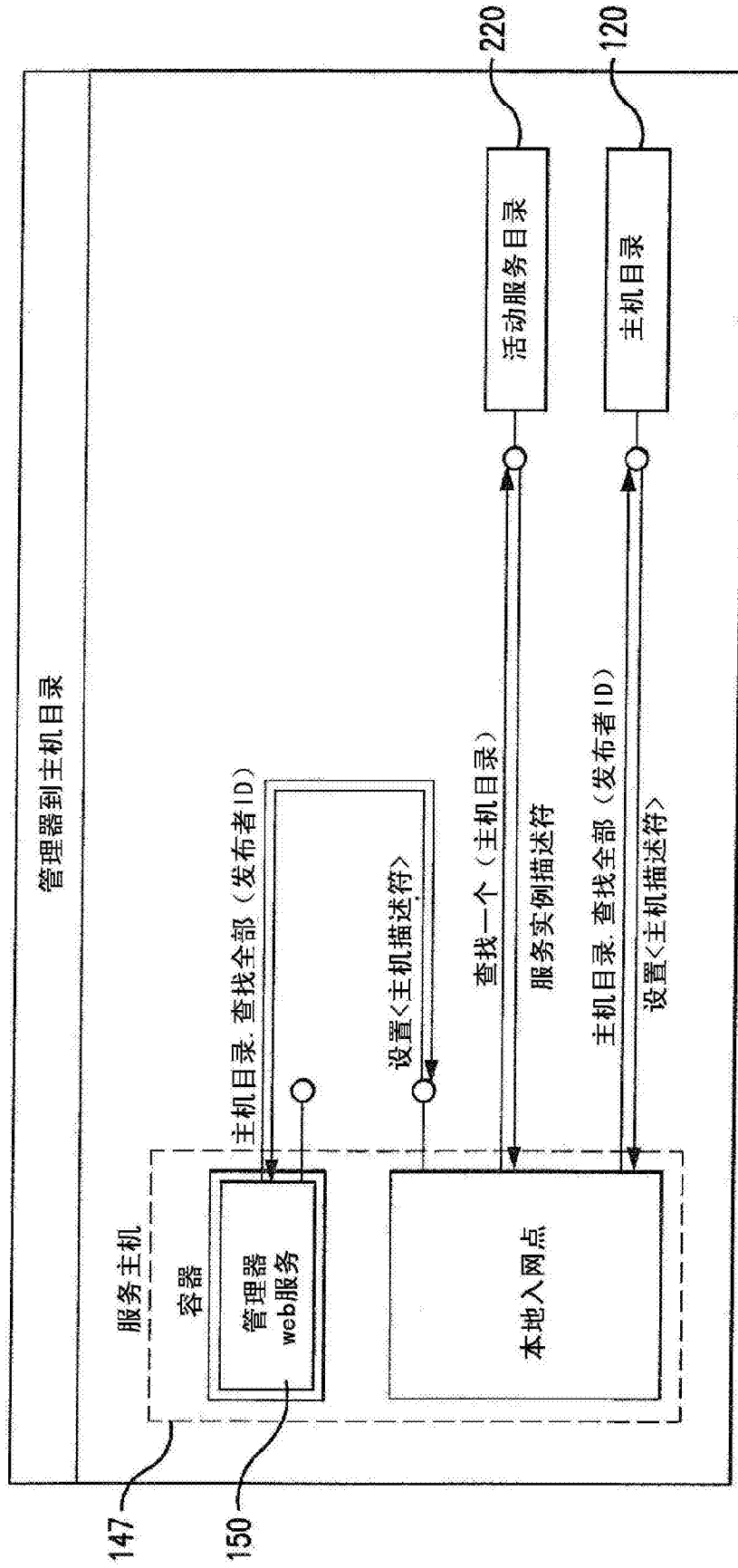


图10b

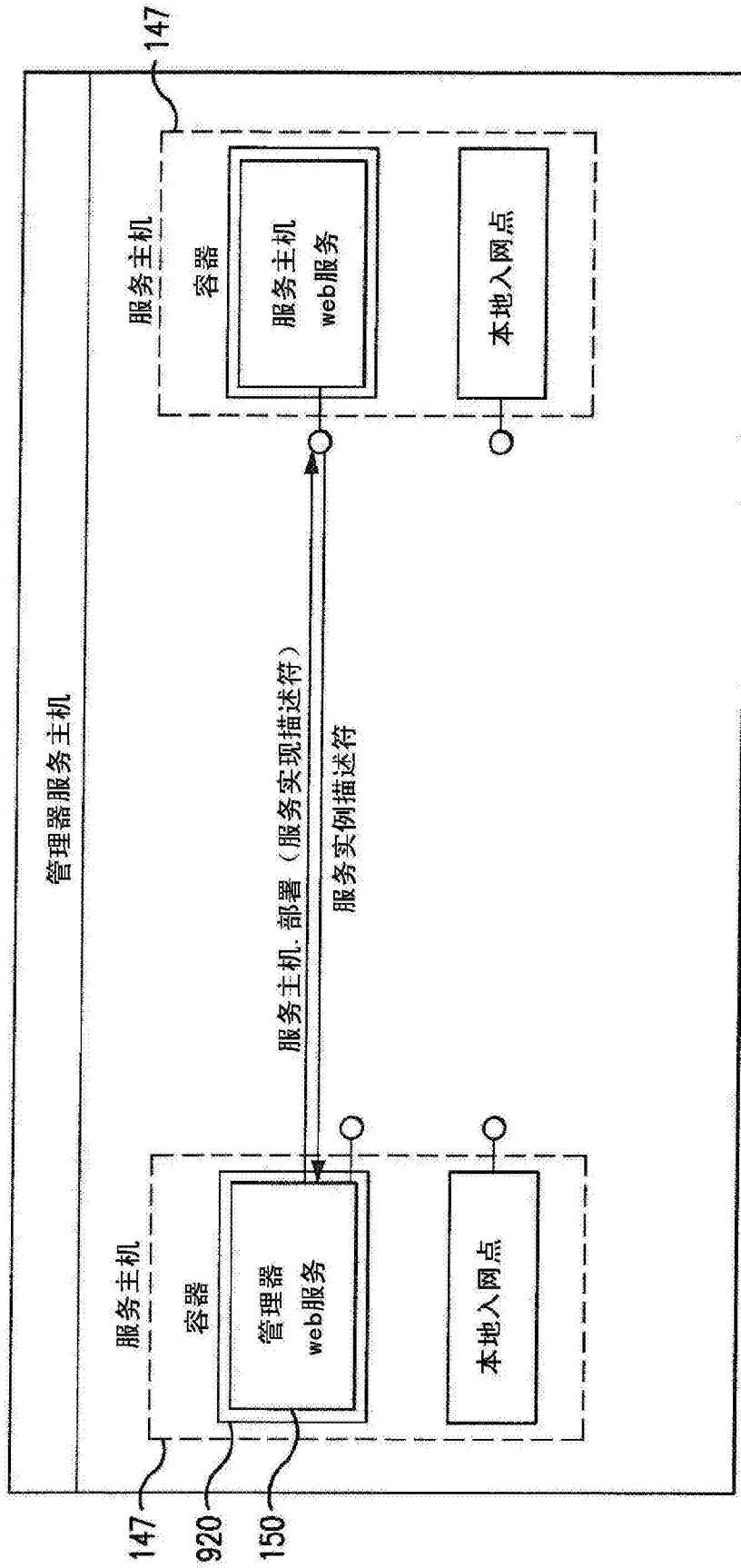


图10c

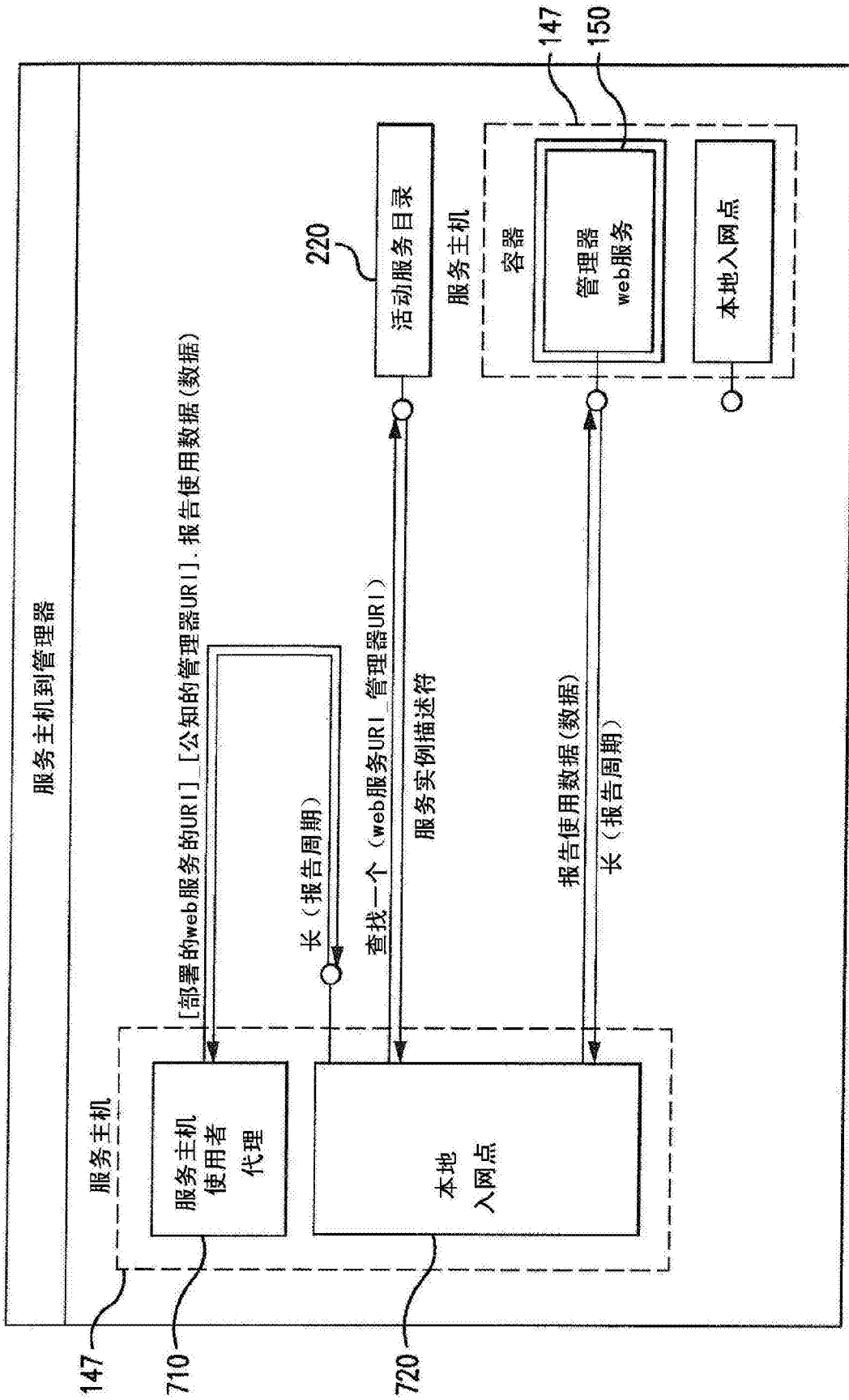


图10d

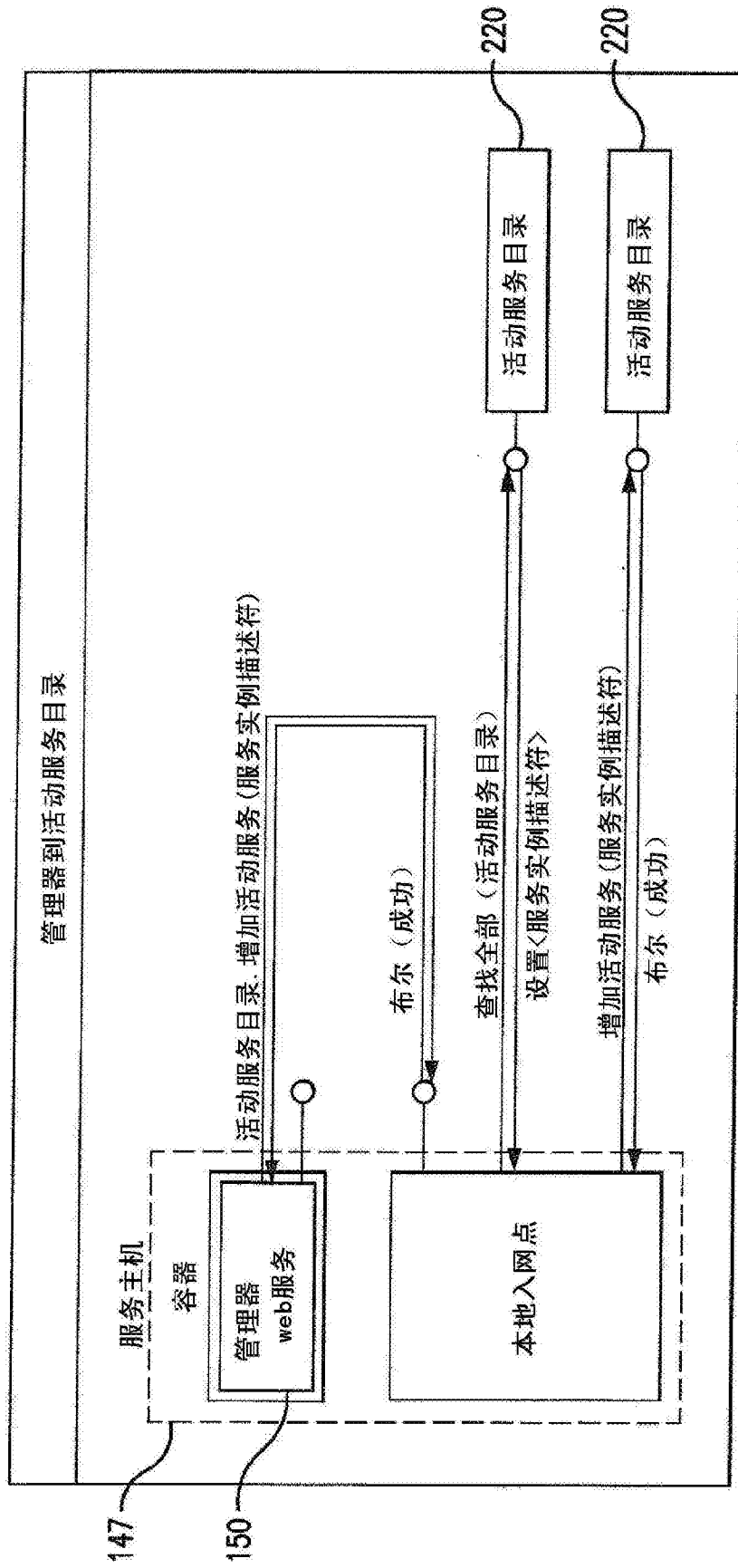


图10e

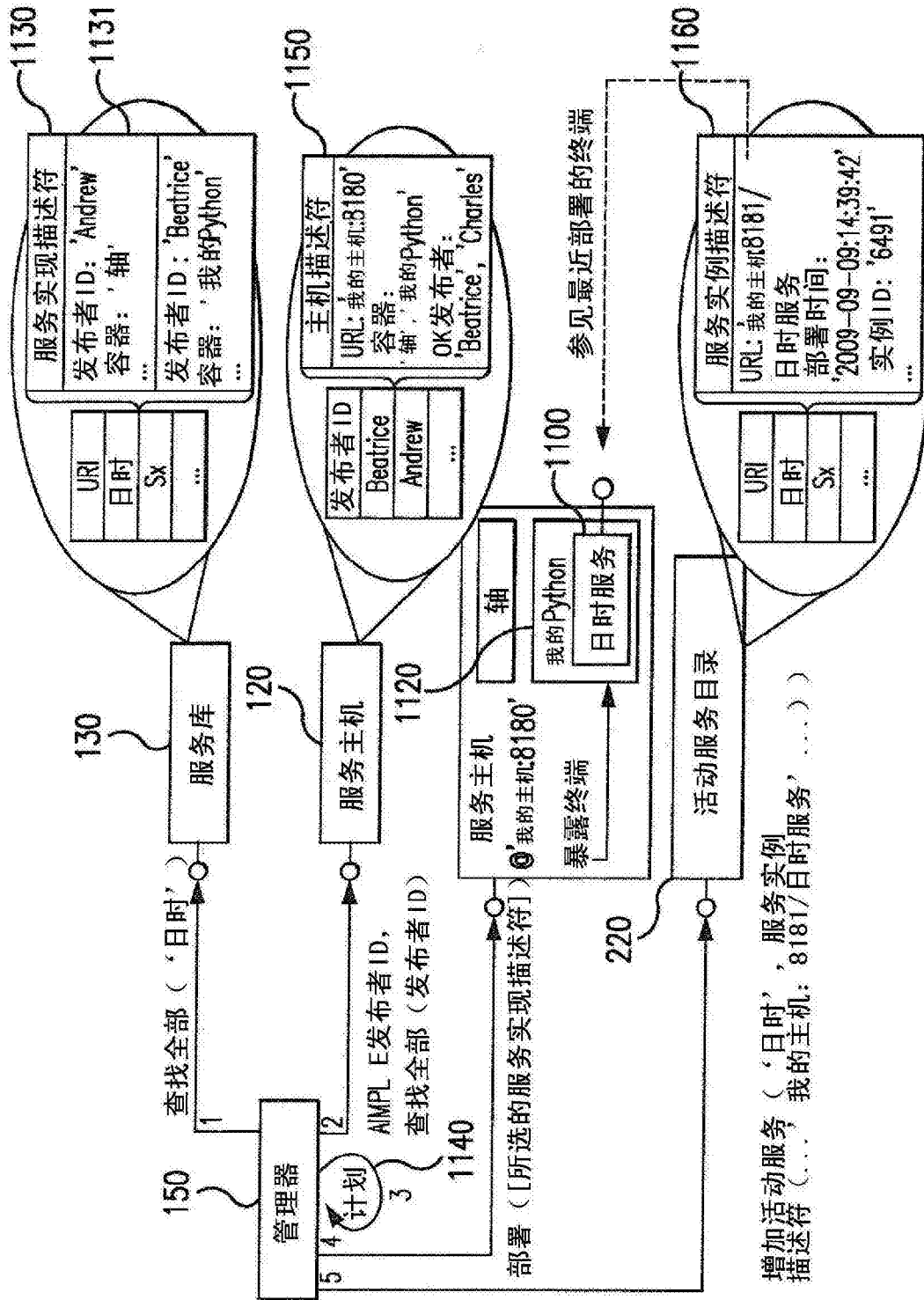


图11

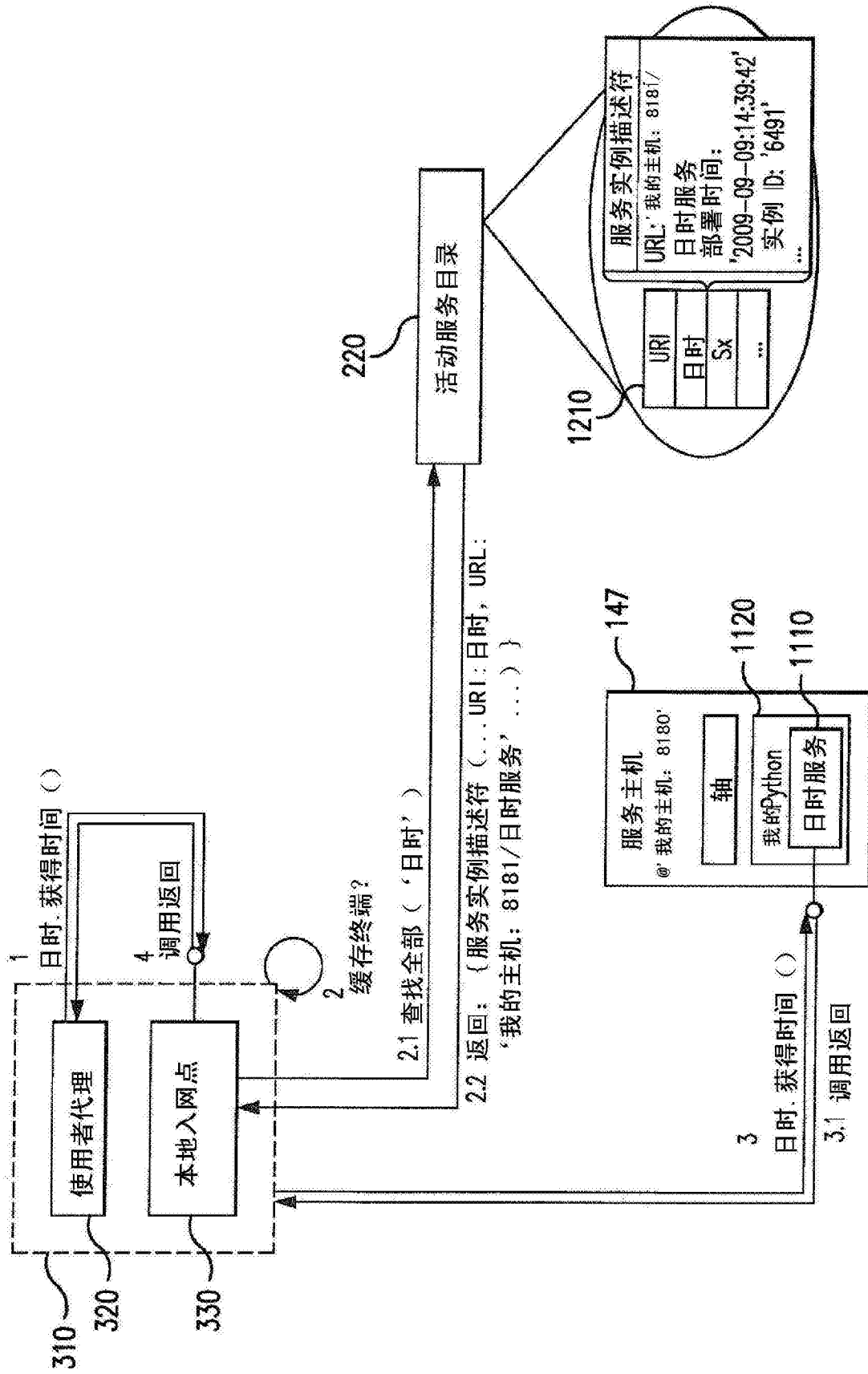


图12

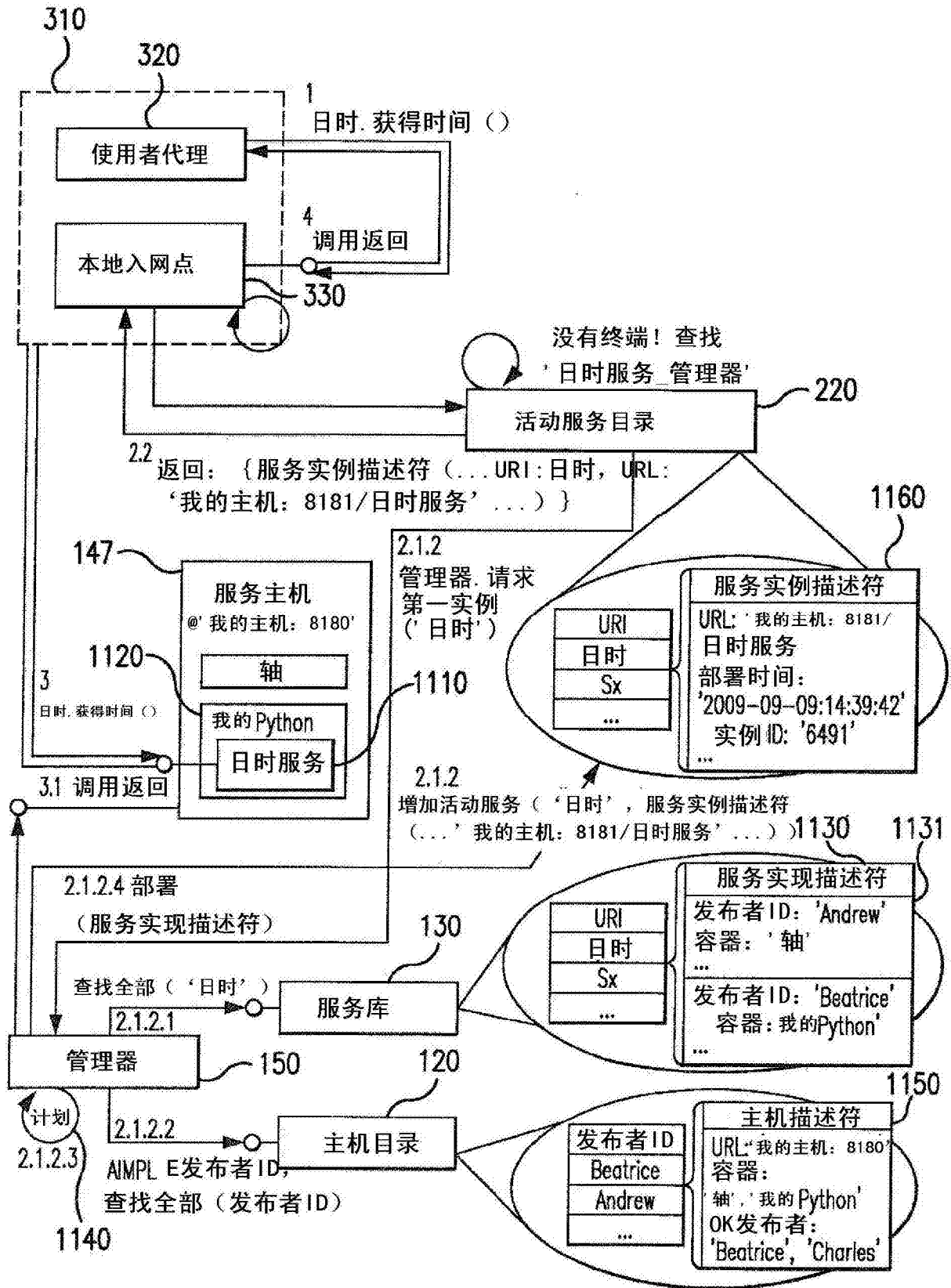


图13