

(12) 发明专利

(10) 授权公告号 CN 101416178 B

(45) 授权公告日 2010.10.27

(21) 申请号 200780012232.7

(51) Int. Cl.

(22) 申请日 2007.01.24

G06F 17/00(2006.01)

(30) 优先权数据

(56) 对比文件

11/278,364 2006.03.31 US

US 2002/0075306 A1, 2002.06.20, 全文.

(85) PCT申请进入国家阶段日

EP 1143363 A1, 2001.10.10, 全文.

2008.10.06

US 5995492 A, 1999.11.30, 全文.

(86) PCT申请的申请数据

审查员 李锋

PCT/US2007/001985 2007.01.24

(87) PCT申请的公布数据

W02007/114884 EN 2007.10.11

(73) 专利权人 微软公司

地址 美国华盛顿州

(72) 发明人 S·D·考克斯

(74) 专利代理机构 上海专利商标事务所有限公

司 31100

代理人 陈斌

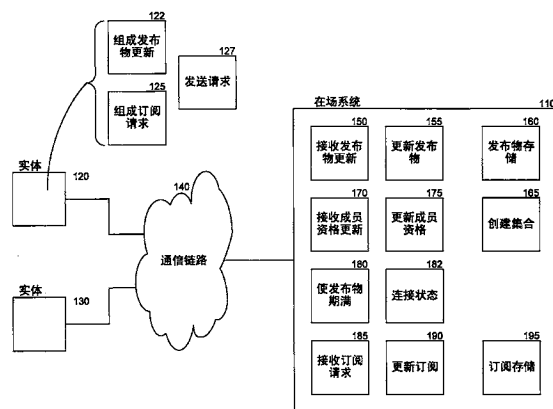
权利要求书 2 页 说明书 9 页 附图 8 页

(54) 发明名称

管理丰富的在场集合

(57) 摘要

提供了用于增加具有发布物的集合的在场服务器的计算和网络效率的方法和系统。该在场系统使用允许在场服务器提供丰富的在场信息而无需昂贵处理的若干技术。首先,该在场系统在单个在场更新请求中接受批量发布物更新。类似地,该在场系统接受批量订阅请求。接着,该在场系统支持消除发布用户持续刷新还未改变的在场信息的需求的发布物更新请求的新期满类型。最后,该在场系统接受包含能访问特定在场集合的成员资格组而非个别用户列表的访问列表。



1. 一种用于修改在场信息的访问权的系统,所述系统包括:

发布物存储组件,所述发布物存储组件存储有发布者的发布物,其中发布物被组织到多个容器中,每个已存储的容器包括一个包含能够访问该容器的发布物的成员的成员列表;

接收成员资格更新组件,它通过用于即时消息通信和在场利用扩展的会话发起协议 SIMPLE 来接收请求消息以更新多个已存储的容器中的一个的成员列表,所述请求消息包括:

用可扩展标记语言 XML 表示的容器部分,所述容器部分包括:

用 XML 表示的容器标识符,所述容器标识符标识已存储的容器中的一个;以及

用 XML 表示的容器版本号,所述容器版本号标识已存储的容器中的一个的版本号;以及

用 XML 表示的成员资格组,所述成员资格组将被授予对存储在多个已存储的容器中的一个中的发布物的访问权,所述成员组包括指定用户资格组类型的属性;

更新成员资格组件,响应于接收到所述请求,通过将成员资格组加入到所述多个已存储的容器中的一个的成员列表中来修改所述多个已存储的容器中的一个的成员列表。

2. 如权利要求 1 所述的系统,其特征在于,所述成员资格组指定域。

3. 如权利要求 1 所述的系统,其特征在于,所述成员资格组指定所述多个已存储的容器中的一个的所有者的联系人列表。

4. 如权利要求 1 所述的系统,其特征在于,所述成员资格组指定联合成员服务器。

5. 如权利要求 1 所述的系统,其特征在于,所述成员资格组指定所述系统外的实体所定义的成员资格组。

6. 如权利要求 1 所述的系统,其特征在于,所述请求包含要对其修改访问权的一个以上的成员资格组。

7. 如权利要求 1 所述的系统,其特征在于,所述请求包含要对其修改访问权的一个以上的容器标识符。

8. 如权利要求 7 所述的系统,其特征在于,所述发布物属于不同的发布用户。

9. 如权利要求 1 所述的系统,其特征在于,还包括接收订阅请求组件,它接收订阅所述多个已存储的容器中的一个的发布物的请求。

10. 如权利要求 9 所述的系统,其特征在于,所述已存储的容器和成员列表创建订阅用户可用的、具有对不同订阅用户而有所不同的详细程度的在场信息的分层结构。

11. 如权利要求 1 所述的系统,其特征在于,所述请求使用 SIP 协议的扩展。

12. 如权利要求 1 所述的系统,其特征在于,所述系统的某些组件位于在场服务器处,而其他组件位于在场客户机处。

13. 一种用于修改在场信息的访问权的方法,所述方法包括:

接收请求消息以更新容器的成员列表,所述请求消息通过用于即时消息通信和在场利用扩展的会话发起协议 SIMPLE 来发送,所述请求消息包括:

用可扩展标记语言 XML 表示的容器部分,所述容器部分包括:

用 XML 表示的容器标识符,所述容器标识符标识已存储的容器中的一个;

用 XML 表示的容器版本号,所述容器版本号标识已存储的容器中的一个的版本号;以

及

用 XML 表示的要对其访问权进行修改的成员资格组,所述成员资格组包括一个类型,类型包括:

联合类型,联合类型表示在该成员资格组中,由多个实体定义对在场信息具有类似访问权的用户列表;

公共云类型,公共云类型表示在该成员资格组中,由不同组织提供的外部可用的用户列表;以及

同一企业类型,同一企业类型表示同一企业内的所有用户能够访问特定的容器;响应于接收到所述请求,更新所述容器的成员列表以修改成员资格组的访问权。

14. 如权利要求 13 所述的方法,其特征在于,所述请求包含要对其修改访问权的一个以上的成员资格组。

15. 一种用于修改成员列表中的成员对于在场信息的访问权的方法,所述方法包括:

向在场服务发送请求以发布在场信息,所述请求通过用于即时消息通信和在场利用扩展的会话发起协议 SIMPLE 来发送,所述请求包括:

用可扩展标记语言 XML 表示的容器部分,所述容器部分包括:

用 XML 表示的容器标识符,所述容器标识符标识容器,该容器的成员列表将要被修改,所述容器存储有在场信息;

用 XML 表示的容器版本号,所述容器版本号标识容器的版本号,该容器的成员列表将要被修改;以及

用 XML 表示的要对其访问权进行修改的成员资格组,所述成员资格组包括一个表示组的类型的属性

其中所述请求指示所述在场服务修改成员资格组内的用户对所述在场信息的访问权。

16. 如权利要求 15 所述的方法,其特征在于,所述成员资格组指定容器的所有者的联系人列表。

17. 如权利要求 15 所述的方法,其特征在于,所述请求包含一个以上的容器部分,所述容器部分标识要对其修改访问权的一个以上的容器。

管理丰富的在场集合

[0001] 背景

[0002] 在场服务器 (presence server) 越来越多地用于提供诸如用户的可用状态等在场信息。用户的在场信息标识该用户的当前“在场状态”。用户可使其在场信息可用,这样其他用户可以决定如何最好地与他们通信。例如,在场信息可以指示用户是登录 (“在线”)到实时消息通信服务器还是注销 (“离线”)。在场信息也可提供有关用户有空与否的更详细信息。例如,即使用户在线,用户也可能在会议中离开其计算机。在这种情形下,在场状态会指示“在线”且“在会议中”。

[0003] 在即时消息通信上下文中,发布用户 (“发布者”) 可向在场服务提供其在场信息,在场服务接着将在场信息提供给订阅用户 (“订阅者”)。因此,在场服务可使用订阅者 / 发布者模型来提供关于在场服务的发布和订阅用户的在场信息。只要用户的在场信息发生改变,就通过该用户的计算机系统向在场服务通知该改变,并进而向订阅用户通知该改变。订阅用户然后可以基于预期参与者的在场信息来决定如何最好地联系发布用户。例如,如果在场信息指示发布用户当前处于会议电话呼叫中,那么订阅用户可决定向发布用户发送即时消息,而非进行电话呼叫。然而如果订阅用户需要呼叫并与发布用户讲话,那么订阅用户可监视发布用户的在场信息,以便知道何时可以进行呼叫。当订阅用户注意到发布用户的在场信息指示电话会议已结束,订阅用户于是可以进行电话呼叫。

[0004] 特定的用户也可从多个设备发布在场信息。例如,用户可同时从膝上型计算机、台式计算机和手机登录到在场服务。在场信息可指示用户在每一设备上的状态。这例如对用户正带着手机前往会议并从其台式计算机离开尤其有用。如果订阅用户想要联系发布用户,则在场信息可指示对手机的呼叫可能是比用户台式计算机处接收的即时消息更有效的联系到用户的方式。每一用户设备也订阅该用户的其它设备的在场信息,例如以允许该设备确定发布用户当前在哪一设备处最活跃。

[0005] 由于即时消息通信系统和其它实时通信系统越来越受欢迎,在场服务需要支持越来越多的用户。此外,这些系统需要支持日益复杂的“在场信息”定义。例如,在场信息可包括指定发布者在接下来数月期间有空与否的日历信息。因此,在场服务器一般被开发成以高效方式提供在场服务。不幸的是,典型的在场模型在更新在场信息时需要密集的计算。因此,当添加了更多的发布者和订阅者时,需要额外的在场服务器来执行必需的计算。

[0006] 早期的在场系统通过将所有的在场信息置于单个 XML 文档中来处理在场信息的发布。当用户的在场信息的一部分改变时,发布者必须指向文档中的一行,并用新的在场信息来替换它,或指向 XML 子树内的节点并用另一 XML 子树替换它。这种更新要求在场服务器执行计算上昂贵的 XML 解析。随着用户和在场更新的数目的增加,XML 解析对在场服务器的负担可能变得过大。这种发布在场信息的模型也不允许订阅者仅订阅在场文档的一部分。此外,文档的各个部分未被版本化,版本化使得订阅者或在场服务器能够容易地告知文档的哪些部分近来被改变。该模型也不允许向不同的订阅者展示在场文档的不同版本,或对文档的各个部分指定不同的授权信息。

[0007] 在相关申请,2005年7月26日提交的题为“ORGANIZING PRESENCE INFORMATION

INTO COLLECTIONS OF PUBLICATIONS(将在场信息组织成发布物集合)”的美国专利申请第 11/190,503 号(代理编号 418268225US)中描述了计算上较不昂贵的更高效的在场系统,该申请通过引用其整体被包含在此。该相关申请描述用于基于发布物集合来管理在场信息的方法和系统。在场集合系统为发布者创建发布物集合。发布物包括类型和值,且可伴有诸如发布物何时期满等属性。例如,类型为“状态”而值为“在线”的发布物可向订阅用户指示发布用户当前在线。发布者可定义要对某些订阅者可用的一个发布物,以及要对其他订阅者可用的另一发布物集合。例如,一个集合可对公众可见,而另一个集合可仅对发布用户的同事可见。每一集合包含被允许订阅该集合内的信息的用户的列表。该在场集合系统可允许发布者更新集合内的各个发布物。当接收更新时,该在场服务器可快速更新在场信息,而无需依赖于计算上昂贵的 XML 文档解析。每一发布物集合也可包含信息的类目。类目是可横跨多个集合的一种发布物。例如,类目“状态”可具有在对公众可用的集合中的值为“不可用”的发布物,以及在对同事可用的集合中的值为“正在与 John 的会议中”的发布物。发布物也可指定情况标识符。情况标识符区分同一集合内的相似信息。例如,用户可具有情况为“膝上型”的发布物“状态”,其指示用户在其膝上型计算机处的状态;以及情况为“手机”的另一发布物“状态”,其指示用户在手机处的状态。

[0008] 尽管该在场集合系统是对现有技术的改进,但它可能对网络资源和在场服务器施加了较高的负担。例如,向在场服务器发送对众多集合内的众多发布物情况的单独改变的发布用户导致对在场服务器的大量网络通信量。类似地,接收每一更新的发布物的单独通知的订阅用户造成了从在场服务器到订阅用户所登录的一个或多个设备的大量网络通信量。一般,在场发布物被设置为在固定时间量内期满,且必须由发布用户定期刷新以保持发布物被发布。刷新未改变的众多发布物的需求导致了对服务器不必要的负担。最后,添加对不同订阅用户具有不同程度的可用细节的集合创造了对更广泛的订阅者组展示在场信息的机会。能够访问集合的用户的列表可能增长得相当大,且检查特定订阅者应订阅哪一集合的用户的列表的计算成本随着该列表的增长而增加,导致对在场服务器的额外负担。

[0009] 概述

[0010] 提供了用于增加具有发布物集合的在场服务器的计算和网络效率的方法和系统。该在场系统使用允许在场服务器提供丰富的在场信息而无需昂贵处理的若干技术。首先,该在场系统在单个在场更新请求中接受批量发布物更新。类似地,该在场系统接受批量订阅请求。接着,该在场系统支持消除发布用户持续刷新还未改变的在场信息的需求的发布物更新请求的新期满类型。最后,该在场系统接受包含能访问特定在场集合的成员资格组而非个别用户列表的访问列表。

[0011] 提供本概述以便以简化的形式介绍将在以下详细描述中进一步描述的一些概念。本概述并不旨在标识出所要求保护的的主题的关键特征或必要特征,也不旨在用于限制所要求保护的的主题的范围。

[0012] 附图简述

[0013] 图 1 是示出在一个实施例中在场系统的组件的框图。

[0014] 图 2 是示出一个实施例中用于发送发布物更新的 SIP 协议扩展的协议列表。

[0015] 图 3 是示出一个实施例中在场系统的更新发布物组件的处理的流程图。

[0016] 图 4 是示出一个实施例中用于发送订阅请求的 SIP 协议扩展的协议列表。

[0017] 图 5 是示出一个实施例中在场系统的更新订阅组件的处理的流程图。

[0018] 图 6 是示出一个实施例中用于更新集合成员资格的 SIP 协议扩展的协议列表。

[0019] 图 7 是示出一个实施例中在场系统的更新成员资格组件的处理的流程图。

[0020] 图 8 是示出一个实施例中在场系统的使发布物期满组件的处理的流程图。

[0021] 详细描述

[0022] 提供了用于增加具有发布物集合的在场服务器的计算和网络效率的方法和系统。该在场系统使用允许该在场服务器提供丰富的在场信息而无需昂贵处理的若干技术。首先,该在场系统在单个在场更新请求中接受批量发布物更新。例如,发布用户可通过向在场服务器发送单个更新请求以在多个集合中发布多个信息类目。类似地,该在场系统接受批量订阅请求。例如,订阅用户可通过向在场服务器发送单个订阅请求来请求订阅多个集合中的多个信息类目。接着,该在场系统支持消除发布用户持续刷新还未改变的在场信息的需求的发布物更新请求的新期满类型。例如,发布用户的家庭电话号码很少改变,因此该在场系统允许用户发布该信息一次并且除非该信息改变否则不更新它,而非每小时重新发布该信息一次。最后,该在场系统接受包含能访问特定在场集合的成员资格组而非个别用户列表的访问列表。例如,发布用户可通过对集合应用成员资格组而非个别地列出每一同事来对其所有同事授予访问权。

[0023] 发布批处理

[0024] 在一个实施例中,在场系统在单个请求中接受多个在场发布物。发布物可横跨多个类目、情况和集合。例如,具有发布物类目“状态”和“位置”的发布用户可在公众可访问的集合中发布状态“不可用”和位置“微软总部”,并在同事可访问的集合中发布状态“在与 John 的会议中”以及位置“40 号楼第 5 会议室”。发布用户也可在每一集合内发布同一信息的多个情况。例如,类目“电话号码”可具有用于家庭电话号码的“家庭”情况,用于工作电话号码的“工作”情况,以及用于手机电话号码的“移动”情况。用户可在同一发布物请求中发布这些情况中的每一个的值。

[0025] 在一个实施例中,在场系统在单个请求中接受对发布物取消发布和进行发布。例如,正在前往会议的发布用户可能注销其台式计算机,导致与台式机有关的发布物情况的取消发布,且用户可能在其手机处变为活跃,导致需要为手机添加新情况。在场系统可在单个更新请求中接受台式机发布物的取消发布和手机发布物的进行发布两者。例如,企业的网络在场服务可在单个请求中接收来自这两个设备的发布物更新,并将更新转发给全局在场服务。

[0026] 在一个实施例中,在场系统使用对话发起协议 (SIP) 以及用于即时消息通信和在场利用扩展的 SIP (SIMPLE) 协议的扩展来发送多个在场更新。SIP 是设备可使用来彼此发现并在设备之间建立、修改和终止会话的应用层控制协议,且是在“RFC3261”中描述的因特网建议的标准。RFC3261(在因特网 www.ietf.org/rfc/rfc3261.txt 可用)通过引用其整体被包含在此。SIP 标准允许实现者添加扩展来定义自定义行为。在场系统添加包含指定要进行发布的在场信息的 <publication>(发布物) 标签列表的标签 <publish>(发布),并添加类似的包含指定要取消发布的在场信息的 <publication> 标签的列表的 <unpublish>(取消发布) 标签。每一发布物标识其要应用的类目、情况和集合。在某些实施例中,在场系统对 <publish> 标签内的发布物接受期满值零作为取消发布信息的指示,

而非具有单独的 <unpublish> 标签。

[0027] 在一个实施例中,在场系统在单个请求中接受多个发布用户的更新。例如,当使用 SIP 协议时,每一发布物可包含发布物所应用的用户的 URI。这允许服务代表大量用户发布信息。例如,Exchange 电子邮件服务器可为使用该服务器的所有用户发布日历信息。作为另一示例,手机运营商可将关于其手机顾客的位置的信息向在场服务器发布。在这两个示例中,之前可能生成过度网络通信量的信息可在向在场服务器的单个请求中为大量用户发布。

[0028] 在一个实施例中,在场系统在单个请求中接受具有不同属性的发布物请求。发布用户可为一个集合中的发布物指定不同于另一个集合中的发布物的期满策略。例如,在具有发布物“地址”的详细信息的集合中,详细状态信息(例如,“在会议室 2 中”)可频繁改变,且具有比另一集合的较不详细的信息(例如,“在西雅图”)短的期限。作为另一示例,发布用户可在同一请求中为一类发布物指定不同于另一类的通知策略。例如,通知策略可指示某些类目中的更新不应导致任何通知。

[0029] 订阅批处理

[0030] 在一个实施例中,在场系统在单个请求中接受来自订阅用户对多个类目的订阅。例如,对发布用户的状态和位置感兴趣的订阅用户可组成在类目列表中包含“状态”和“位置”的单个订阅请求。

[0031] 在一个实施例中,在场系统在单个请求中接受来自订阅用户对多个发布用户的订阅。例如,订阅用户可在单个请求中订阅用户 A 和用户 B 的多个类目。当订阅用户正订阅其联系人列表上的每一用户的在场信息时,这可导致大量的资源节约,因为所有订阅可在单个请求中作出。类似地,在场系统也可在单个请求中接受不同用户的不同类目列表。例如,订阅用户可在单个请求中订阅用户 A 的类目 1 和 2 以及用户 B 的类目 3 和 4。

[0032] 在一个实施例中,在场系统允许订阅请求在不创建订阅的情况下检索在场信息。例如,用户可能想要请求发布用户的在场信息一次而不在信息改变时接收通知。用户可发送订阅请求以及该请求是对所指定类目中在场信息的当前值的查询而非创建订阅的请求的指示。

[0033] 在一个实施例中,在场系统在与新订阅添加的同一请求中接受对现有订阅的移除。例如,如果订阅用户不再对知道发布用户的位置感兴趣,且可能想要跟踪发布用户的当前电话号码,则订阅用户可组成在对其要取消订阅的类目列表中包含类目“状态”且在对其要订阅的类目列表中包含类目“电话号码”的单个请求。

[0034] 在一个实施例中,在场系统在单个请求中接受代表多个订阅用户的订阅请求。例如,企业可具有从企业内的订阅用户接收个别订阅请求并将请求作为单个批量订阅请求转发给全局在场服务器的本地在场服务器。

[0035] 在一个实施例中,在场系统使用对 SIP 和 SIMPLE 协议的扩展来发送多个在场订阅。在场系统添加包含指定要订阅的类目列表的 <categoryList>(类目列表) 标签的标签 <batchSub>(批量订阅)。<batchSub> 标签也可指定 <adhocList>(特设列表) 标签,<adhocList> 标签指定要订阅其类目的多个用户的列表。在场系统也可添加也包含指定正在取消订阅的用户及其类目列表的标签的标签 <batchUnsub>(批量取消订阅)。系统也可允许具有空正文以指示应取消订阅所有用户和类目的订阅请求,而不要求订阅用户列出他

们已经订阅的每个类目和用户。

[0036] 在一个实施例中,在场系统通过要求对订阅用户感兴趣的每一类目的显式订阅来过滤掉订阅用户不想要的在场信息。例如,发布用户可发布数百个信息类目,但订阅用户可能仅想要知道发布用户的当前状态。订阅用户可发送指示对类目“状态”的订阅的订阅请求,且该用户将仅接收到该类目的通知。这防止订阅用户由于在场服务器提供的信息量增长而被用户不感兴趣的通知淹没。

[0037] 期满模型

[0038] 在一个实施例中,在场系统接受在发布设备从在场服务器中注销时期满的发布物。在以往的系统,发布用户将信息发布固定的持续时间(例如,一小时),且在发布物接近期满时必须刷新或重新发布信息以保持信息被发布。在场系统定义新的一类期满,它保持在场信息被发布,直到发布信息的设备下线。这例如当诸如手机等设备发布了仅当手机在线时有效的信息的情况时使用。当手机关机或下线时,在场服务器可检测到并移除仅在设备在线时有关的信息。当设备在线时,不再要求持续更新未改变的在场信息以保持其不会期满。

[0039] 在一个实施例中,在场系统接受当发布用户不再在线时期满的发布物。这种期满类型对只要用户在线就应用于多个设备但当用户下线就不再相关的在场信息有用。在场系统可在用户不再通过任何设备登录时检测到用户离线。在场系统可在设备从在场服务器断开或当设备向在场服务器告知用户不再使用该设备时检测到该用户不再登录到该设备上。例如,公共因特网网吧可允许用户通过插入标识用户的 USB 道尔芯片来访问因特网。当移除道尔芯片时,因特网网吧可通知诸如在场服务等用户曾连接到的任何服务,该用户不再使用该设备。当用户在线时,该信息保持被发布,而无需来自用户的刷新请求,从而使在场服务器无需处理附加的更新请求。

[0040] 在一个实施例中,在场系统接受未期满的发布物。这种称之为静态的期满类型对无论用户是否在线都相关的信息有用,诸如用户的电话号码或地址。这种期满类型也可由代表用户发布信息的设备使用。例如,发布多个用户的日历信息的 Exchange 服务器不需要在发布用户的日历信息之前知道用户是否在线,且无论用户的当前登录状态是什么信息都应不会期满。从而,这样的服务可通过为发布物指定静态期满类型来发布信息,且所发布的信息不会被取消发布,除非在场服务器接收显式的取消发布请求。

[0041] 成员资格组

[0042] 在一个实施例中,在场系统接受包含用户组的集合成员资格列表。以往,能够访问特定在场信息集合的用户被显式列出,且默认集合包含应用于不在任何列表上的用户的信息。在场系统允许指定用户组,这减少了对大型用户组确定用户是否能够访问一集合的计算难度。集合成员资格列表可在发布在场信息时指定,或者可在单独的请求中独立指定。一类成员资格组“联系人列表”指定位于发布用户的联系人列表上的任何订阅用户可查看该组所分配到的集合中的信息。

[0043] 在一个实施例中,在场系统接受包含同一企业内的用户的成员资格组作为发布用户。该成员资格组允许用户集合成员资格列表中的单个条目指定同一企业中的任何订阅用户应能够访问特定的集合。例如,对如微软的大型公司,这允许以包含成员资格列表中指定“同一企业”成员资格组的单个条目的同事的附加细节来创建集合。在场服务器通过检查用

户的企业是否与发布用户相同,诸如通过查询外部域控制器来确定特定的订阅用户是否能访问该集合。这防止在场服务器复制外部域控制器已经包含的同一用户列表。

[0044] 在一个实施例中,在场系统接受包含公共云内的用户的成员资格组。公共云被定义为不同组织所提供的外部可用的用户列表。公共云成员资格组当在场服务器的操作员之一与其他在场系统的操作员合作时有用。例如,使用微软的即时消息通信软件的在场服务器可使用公共云成员资格组与 AOL、Yahoo 和其他即时消息平台交互。通过在集合成员资格列表中指定公共云成员资格组,发布用户可定义其他在场系统的成员可订阅并查看的在场发布物而无需在集合成员资格列表中显式列出每一成员。

[0045] 在一个实施例中,在场系统接受联合成员资格组。联合成员资格组是其中多个实体定义应对在场信息具有类似访问权的用户列表的成员资格组。例如,诸如微软和英特尔等两个公司可使其雇员列表对联合成员资格组可用。接收具有联合成员资格组类型的发布物的在场服务器查阅指定的联合成员资格服务器以确定哪些用户是联合组的成员。发布用户然后可指定联合组可订阅的发布物。

[0046] 在一个实施例中,在场系统在单个请求中接受批量成员资格列表改变。例如,发布用户可在单个请求中向同一企业成员资格组内的订阅用户授予对一个集合的访问权,并将公共云成员资格组中的用户从集合中移除。发布用户可在单个请求中指定多个集合的成员资格列表以及对一特定集合的多个成员资格组和一成员资格列表中的个别用户。

[0047] 通过组合集合模型与成员资格组,发布用户可建立在场信息访问的分层结构。例如,可使用用户的日历信息可为同一企业内的同事详细使用,而对一般公众可作为空闲/忙碌信息(仅指示用户空闲的时间以及用户忙碌的时间)使用。这允许一般公众的成员安排与发布用户进行的不与用户的日程安排的其余部分冲突的约会,而不展示关于该用户正在做什么的个人信息。另一方面,向同事提供用户正在做什么的更详细视图,例如使得同事能够知道用户在当天稍晚将会参加与同事的会议。

[0048] 图 1 是示出在一个实施例中在场系统的组件的框图。在场系统 110 经由诸如因特网等通信链路 140 连接至实体设备 120 和 130。实体设备对应于可以是发布者或订阅者的实体。实体设备包括组成发布物更新组件 122、组成订阅请求组件 125 以及发送请求组件 127。组成发布物更新组件 122 组成对可能包括批量个别发布物更新的所发布在场信息的更新请求。组成订阅请求组件 125 组成可能包括对多个发布用户的多个在场信息类目的批量订阅的订阅请求。发送请求组件 127 向在场系统 110 发送发布物更新请求和订阅请求。

[0049] 在场系统包括接收发布物更新组件 150、更新发布物组件 155、接收成员资格更新组件 170、更新成员资格组件 175、使发布物期满组件 180、检测连接状态组件 182、接收订阅请求组件 185、更新订阅组件 190、创建集合组件 165、发布物存储组件 160 以及订阅存储组件 195。发布物存储 160 包含发布者的发布物。发布物被组织到由创建集合组件 165 所创建的集合中。当从发布者接收更新一个或多个发布物的请求时,调用接收发布物更新组件 150。接收发布物更新组件 150 调用更新发布物组件 155 来按请求所指定地添加、移除或更新发布物存储 160 中的发布物。在从发布者接收对更新发布物存储 160 中的一个或多个集合的成员资格列表的请求时,调用接收成员资格更新组件 170。该请求可作为独立消息或可作为发布物更新请求的一部分接收。接收成员资格更新组件 170 调用更新成员资格组件 175 来添加、移除和更新发布物存储 160 中集合上的成员资格。在从一实体接收订阅发布者

的一类或多类发布物的请求时,调用接收订阅请求组件 185。接收订阅请求组件 185 调用更新订阅组件 190 来将使该实体订阅所请求类型的发布物。订阅被存储在订阅存储 195 中,订阅存储可与发布物存储 160 集成。在场系统周期性地调用使发布物期满组件 180 以清除发布物存储 160 中的期满发布物。尽管未在图 1 中示出,但实体设备包括在场系统的定义集合及其成员资格、发送发布物更新、发送订阅请求以及接收对发布物更新的通知的各组件。

[0050] 在其上实现该系统的计算设备可包括中央处理单元、存储器、输入设备(例如,键盘和定点设备)、输出设备(例如显示设备)和存储设备(例如,盘驱动器)。存储器和存储设备是包含实现该系统的指令的计算机可读介质。此外,数据结构和消息结构可被存储或经由诸如通信链路上的信号之类的数据传送介质发送。可以使用各种通信链路,诸如因特网、局域网、广域网、点对点拨号连接、蜂窝电话网络等。

[0051] 该系统的实施例可以在各种操作环境中实现,这些操作环境包括个人计算机、服务器计算机、手持式或膝上型设备、多处理器系统、基于微处理器的系统、可编程消费者电子产品、数码相机、网络 PC、小型计算机、大型计算机、包括任何上述系统或设备中任一种的分布式计算环境等。计算机系统可以是手机、个人数字助理、智能电话、个人计算机、可编程消费者电子产品、数码照相机等。

[0052] 该系统可以在由一个或多个计算机或其他设备执行的诸如程序模块等计算机可执行指令的通用上下文中描述。一般而言,程序模块包括执行特定的任务或实现特定的抽象数据类型的例程、程序、对象、组件、数据结构等等。通常,程序模块的功能可以在各个实施例中按需进行组合或分布。

[0053] 图 2 是示出一个实施例中用于发送发布物更新的 SIP 协议扩展的协议列表。该协议以 SIP 动词“PUBLISH(发布)”210 开始,它发出发布信息的请求的信号。该协议包含进一步描述请求的头部 220,诸如其内容类型。该协议包含“<publish>”部分 230,它包含“<publication>”部分 232。“<publications>”部分列出要发布的发布物 235 和 240 以及要取消发布的发布物 250。发布物 235 指定类目(category)名为“status(状态)”、类目的具体情况(instance)名为“laptop(膝上型)”、容器(container,这是一类集合)标识为“1”、版本(version)为“0”、发布用户由 SIP 地址“sip:foo@bar.com”标识且发布物的期满类型(expiretype)为“user(用户)”(这指示发布物在用户下线时应被移除)的在场信息。尽管为清楚起见对一些字段显示了文本,但也可使用数字。例如,“instance”字段的值可包含被分配给设备的唯一标识符而非文本“laptop”。另一发布物 240 在对同一发布用户的不同容器“2”中指定同样的类目“status”,但不同的情况“desktop(台式机)”。该协议还包含要从发布物存储移除的发布物 250,如在本示例中由期满类型 0 指示。尽管在此示例中,“expireType(期满类型)”字段被重载以指示发布物要被取消发布,但也可使用协议内的单独字段或单独的 XML 部分。由类目名、情况标识符、容器标识符以及发布用户地址标识正被移除的发布物 250。在发布物中指定发布用户的地址允许除该用户以外的实体代表用户发布和取消发布在场信息,诸如上述的 Exchange 服务器。

[0054] 图 3 是示出一个实施例中在场系统的更新发布物组件的处理的流程图。该组件在接收到诸如图 2 中的发布请求时由接收发布物更新组件调用。在框 310,该组件从接收发布物更新组件接收更新发布物请求。在框 320,该组件从请求中选择下一个发布物。在框 330,该组件定位由发布物存储中的所选发布物标识的发布用户的集合。在判定框 340,如

果该发布物存在,则该组件在框 360 继续,否则该组件就在框 350 继续。在框 350,之前不存在的发布物被添加到发布物存储。在框 360,用请求中的信息更新发布物存储中现有的发布物。在判定框 370,如果在请求中还有发布物,则该组件循环到框 320 以选择下一发布物,否则该组件完成。

[0055] 图 4 是示出一个实施例中用于发送订阅请求的 SIP 协议扩展的协议列表。该协议以 SIP 动词“SUBSCRIBE(订阅)”410 开始,它发出订阅发布信息的请求的信号。该协议包含进一步描述请求的头部 420,诸如其内容类型。该协议包含“<batchSub>”部分 440,它包含“<action>”(动作)部分 442。动作部分包含“<adHocList>”部分 445 和“<categoryList>”部分 450。特设列表部分 445 指定订阅用户想要对其发布物接收通信的发布用户的标识符。类目列表部分 450 指定对每一发布用户订阅的类目。订阅请求可使订阅用户对每一指定用户的每一指定类目接收通知。尽管未在图 4 中示出,但该请求也可包含与订阅部分 440 相似的取消订阅的类目和用户的列表。

[0056] 图 5 是示出一个实施例中在场系统的更新订阅组件的处理的流程图。该组件在接收到诸如图 4 中的订阅请求时由接收订阅请求组件调用。在框 510,该组件从接收订阅请求组件接收订阅请求。在判定框 520,如果在请求中包含更多类目过滤器,则该组件在框 530 继续,否则该组件就在框 550 继续。在框 530,该组件从请求中选择下一类目过滤器。在框 540,该组件更新订阅存储中的类目过滤器。组件然后循环至框 520 以确定在请求中是否还有类目过滤器。在判定框 550,如果在请求中还有订阅,则该组件在框 560 继续,否则该组件就在框 580 继续。在框 560,该组件从订阅请求中选择下一订阅。在框 570,该组件将该订阅添加到订阅存储中。订阅可采用用户列表和类目列表的形式,使得可在单个订阅部分中为多个用户订阅多个类目。该组件然后循环回框 550,以确定在请求中是否还有其他订阅。在判定框 580,如果在请求中还有取消订阅,则该组件在框 590 继续,否则该组件完成。在框 590,该组件选择请求中的下一取消订阅。在框 595,该组件将该所指定的订阅从订阅存储中移除。在处理了所有的取消订阅之后,该组件完成。

[0057] 图 6 是示出一个实施例中用于更新集合成员资格的 SIP 协议扩展的协议列表。所示协议的部分可被包括在如图 2 所示的 SIP 发布请求的主体中,或它可以是专用于更新成员资格的独立消息类型的一部分。该协议以包含标识要修改其成员资格的集合的一个或多个“<container>”部分 615 的“<setContainerMembers>”(设置容器成员资格)部分 610 开始。每一容器部分包含一个或多个成员条目 620、630 和 640。图 6 中的第一成员条目 620 指定指示该值是特定用户的 SIP URI 的类型“uri”。动作(action)类型“add(添加)”指示所指定的用户要作为能访问指定集合的成员添加。第二成员条目 630 指定类型“sameDomain(同一域)”,指示与发布用户来自同一域的用户成员资格被修改,而动作类型“remove(移除)”指示对集合的访问权被移除。第三成员条目 640 指定类型“contactList(联系人列表)”以及动作类型“add”,指示对发布用户的联系人列表的任何成员授予对指定集合的访问权。

[0058] 图 7 是示出一个实施例中在场系统的更新成员资格组件的处理的流程图。该组件在接收到诸如图 6 中的成员更新请求时由接收成员资格更新组件调用。在框 710,该组件从接收成员资格更新组件接收更新成员资格请求。在框 720,该组件从请求中选择下一个容器部分。在框 730,该组件选择容器部分内的下一成员更新。在判定框 740,如果该成员更新

包含动作类型“add”，则该组件在框 750 继续，否则该组件就在框 760 继续。在框 750，将在成员更新中指定的用户或组添加到发布物存储中的所选容器的成员列表。在框 760，将在成员更新中指定的用户或组从发布物存储中的所选容器的成员列表中移除。在判定框 770，如果在所选容器部分中还有成员更新，则该组件循环至框 730 以选择下一成员更新，否则该组件继续至框 780。在判定框 780，如果在成员资格更新请求中还有容器部分，则该组件循环到框 720 以选择下一容器部分，否则该组件完成。

[0059] 图 8 是示出一个实施例中在场系统的使发布物期满组件的处理的流程图。在场系统周期性地调用该组件以从发布物存储中移除过时发布物。在框 810，该组件从发布物存储中选择下一个发布物。在判定框 820，如果所选发布物的期满类型为“static(静态)”，则该组件在框 860 继续，否则该组件就在框 830 继续。在判定框 830，如果所选发布物的期满类型为“device(设备)”，则该组件在框 833 继续，否则该组件就在框 840 继续。在判定框 833，如果发布所选发布物的设备离线，则该组件在框 836 继续，否则该组件就在框 860 继续。在框 836，该组件从发布物存储中移除期满发布物。在判定框 840，如果所选发布物的期满类型为“user”，则该组件在框 843 继续，否则该组件就在框 850 继续。在判定框 843，如果发布所选发布物的用户未通过任何设备登录，则该组件在框 846 继续，否则该组件就在框 860 继续。在框 846，该组件从发布物存储中移除期满发布物。在判定框 850，如果所选发布物的期满类型为“duration(持续时间)”，则该组件在框 853 继续，否则该组件就在框 860 继续。在判定框 853，如果为所选发布物指定的时间量期满，则该组件在框 856 继续，否则该组件就在框 860 继续。在框 856，该组件从发布物存储中移除期满发布物。在判定框 860，如果在发布物存储中还有发布物，则该组件循环到框 810 以选择下一发布物，否则该组件完成。尽管所述方法示出了检查和移除期满发布物的轮询方法，但本领域的普通技术人员将理解，可使用其他方法，诸如其中用户或设备从在线改为离线状态的事件使得期满发布物被直接移除的事件驱动模型。

[0060] 尽管用对结构特征和 / 或方法动作专用的语言描述了本主题，但可以理解，所附权利要求书中定义的主题不必限于上述具体特征或动作。相反，上述具体特征和动作是作为实现权利要求的示例形式公开的。例如，使用 SPI 协议描述了在场系统所处理的请求，但可使用其他协议，诸如对传输控制协议 (TCP) 的自定义协议。在某些情形中，步骤被描述为由在场服务器的组件执行，但它们也可由与在场服务器交互的实体处的组件执行。系统也可被实现为其中没有单个中央在场服务器，相反可由对等体分发和存储在信息的对等网络。因此，本发明只由所附权利要求来限制。

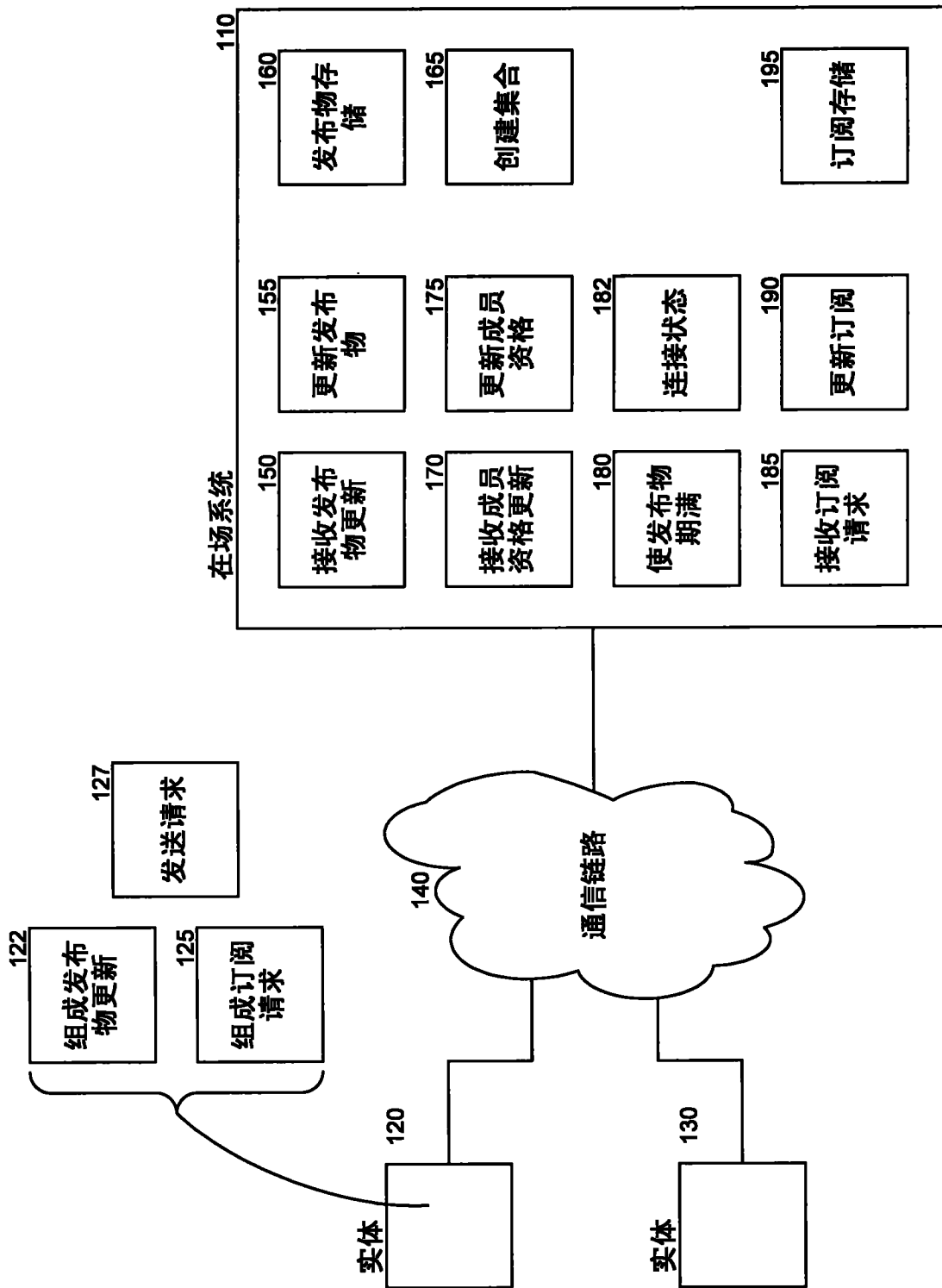


图 1

```
210 { PUBLISH
      {
220 { Event: presence
      { Content-type: application/msrtc-presence-publish
      { ...
232 { <publish xmlns="http://schemas.microsoft.com/2006/09/sip/rich-presence">
      { <publications>
235 { <publication categoryName="status" instance="laptop" container="1"
      { version="0" uri="sip:foo@bar.com" expireType="user"/>
240 { <publication categoryName="status" instance="desktop" container="2"
      { version="0" uri="sip:foo@bar.com" expireType="device"/>
250 { <publication categoryName="location" instance="laptop" container="2"
      { uri="sip:foo@bar.com" expireType="0"/>
      { </publications>
      { </publish>
```

图 2

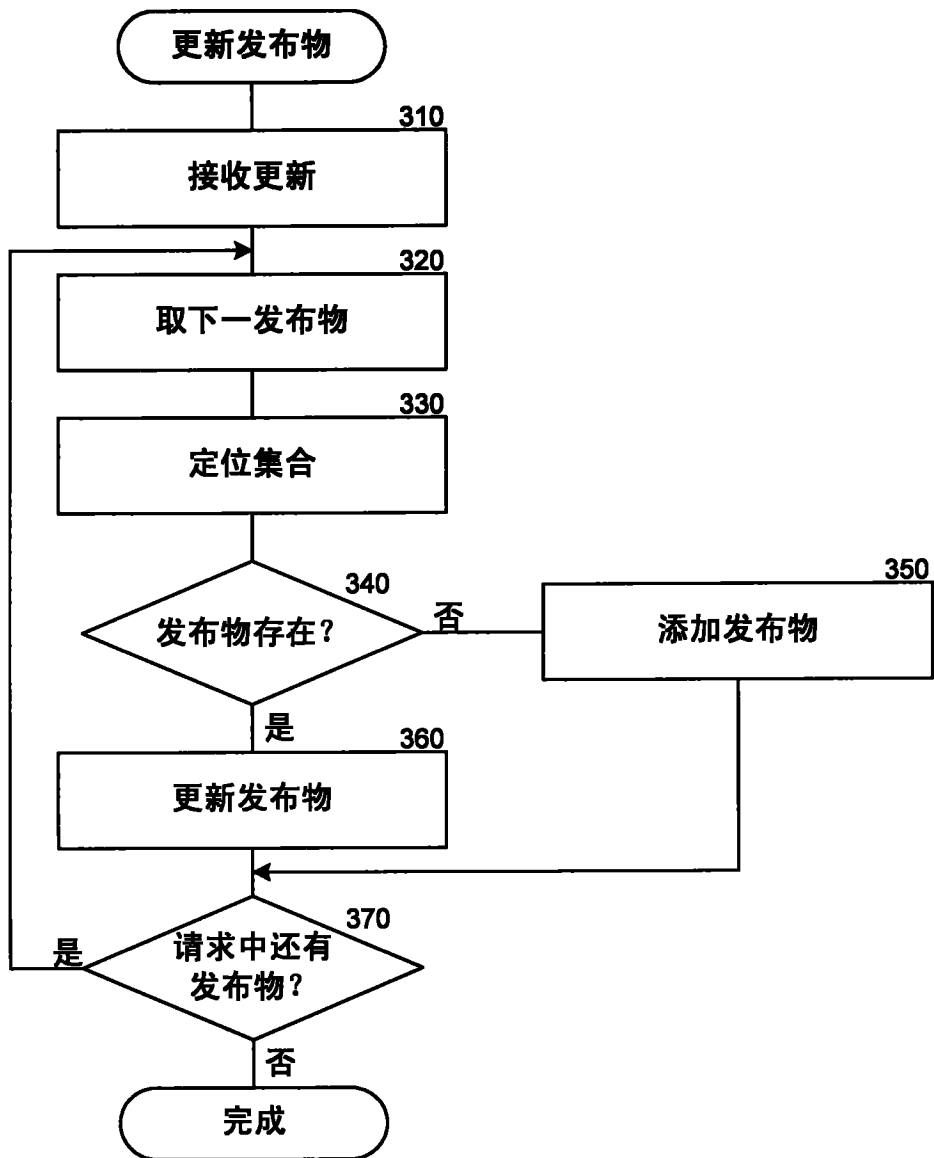


图 3

410 { SUBSCRIBE

420 { Event: presence
Content-type: application/msrtc-event-category-filter+xml
...

442 { <batchSub xmlns= "http://schemas.microsoft.com/2006/01/sip/batch-subscribe">
<action name="subscribe">

445 { <adhocList>
<resource uri="sip:userA@bar.com"/>
<resource uri="sip:userB@bar.com"/>
</adhocList>

440 { <categoryList>
450 { <category name="location"/>
<category name="status"/>
<category name="enterprise.location"/>
</categoryList>

</action>
</batchSub>

图 4

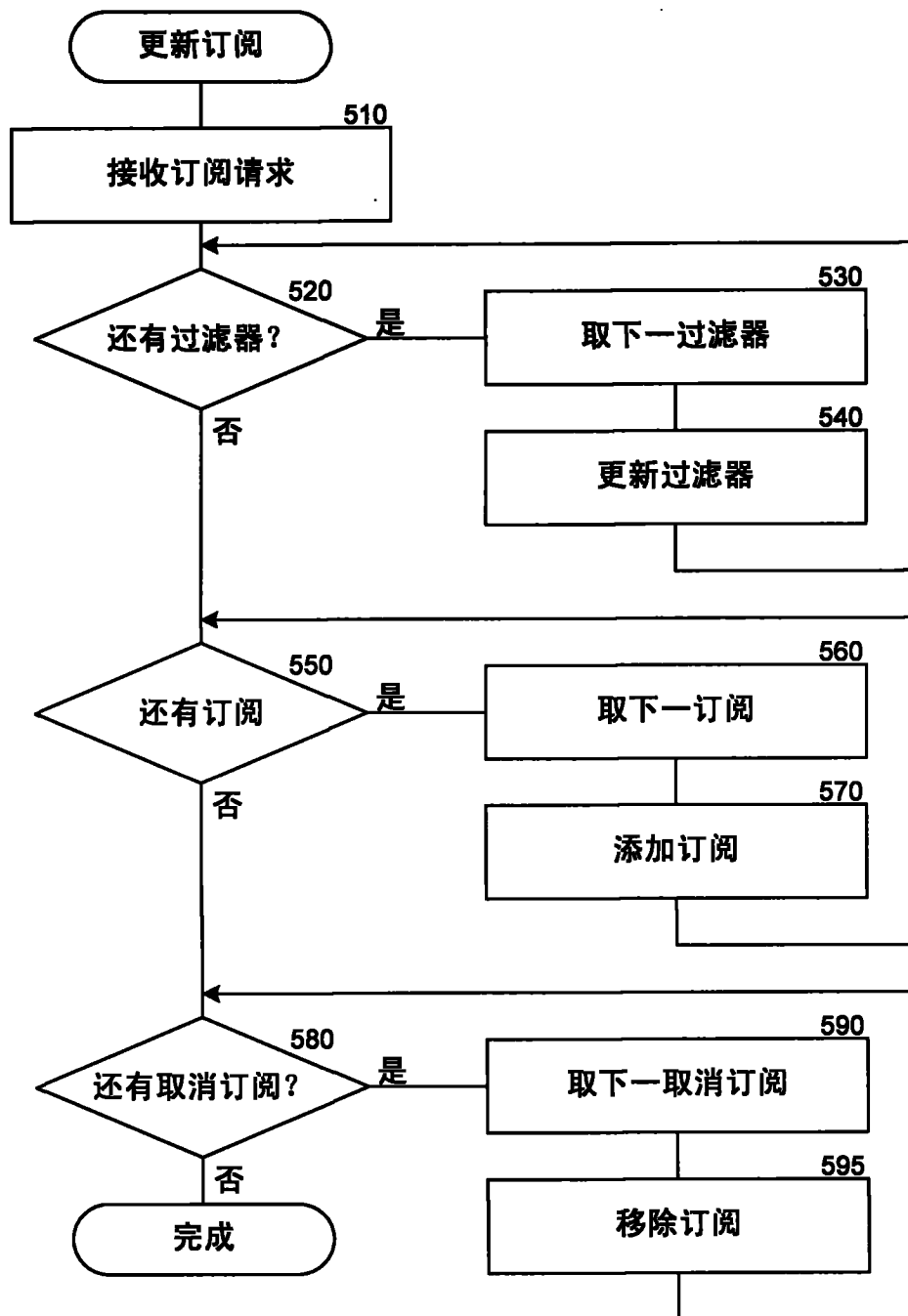


图 5

```
...  
615 { <setContainerMembers xmlns="http://schemas.microsoft.com/2006/09/sip/container-management">  
      <container id="1" version="0">  
620 ~~~~~ <member type="uri" value="user1@domain.com" action="add"/>  
630 ~~~~~ <member type="sameDomain" action="remove"/>  
640 ~~~~~ <member type="contactList" action="add"/>  
      </container>  
    </setContainerMembers>
```

图 6

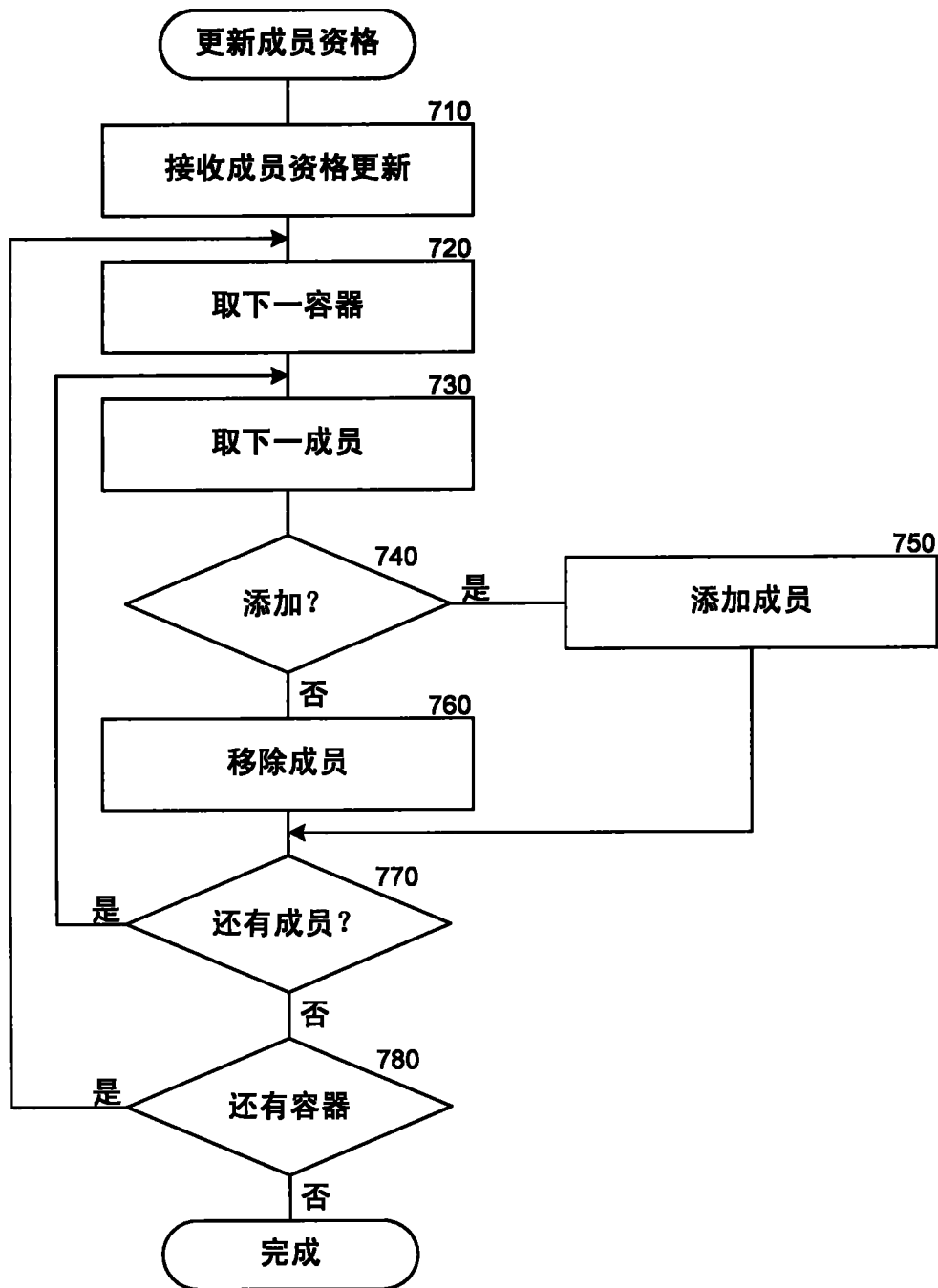


图 7

