



(12) 发明专利

(10) 授权公告号 CN 1792076 B

(45) 授权公告日 2010.08.18

(21) 申请号 200480013537.6

(22) 申请日 2004.04.19

(30) 优先权数据

0308991.9 2003.04.17 GB

(85) PCT申请进入国家阶段日

2005.11.17

(86) PCT申请的申请数据

PCT/GB2004/001685 2004.04.19

(87) PCT申请的公布数据

W02004/095806 EN 2004.11.04

(73) 专利权人 维斯托公司

地址 英国伦敦

(72) 发明人 莱昂内尔·沃洛威兹 马克·格里顿

鲍伯·斯坦登

(74) 专利代理机构 中国国际贸易促进委员会专利商标事务所 11038

代理人 董莘

(51) Int. Cl.

H04L 29/08(2006.01)

H04L 29/06(2006.01)

H04L 12/58(2006.01)

G06F 9/45(2006.01)

(56) 对比文件

WO 0177815 A2, 2001.10.18, 说明书第1页第15-20行, 第1页第29-35行, 第3页第12-15行, 第19页第21-28行.

WO 0193033 A, 2001.12.06, 全文.

WO 9746939 A, 1997.12.11, 全文.

审查员 贺利良

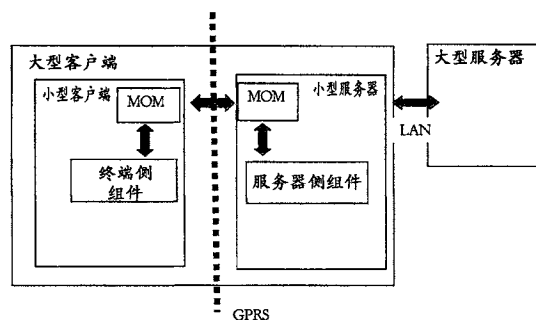
权利要求书 3 页 说明书 23 页 附图 6 页

(54) 发明名称

包含分布式软件应用的数据访问、复制或通信系统

(57) 摘要

本发明设想了一种数据访问、复制或通信系统,该系统包含分布于运行在终端上的终端侧组件和服务器侧组件的软件应用;其中所述终端侧组件和所述服务器侧组件(i)共同构建到服务器的客户端,并且(ii)通过使用消息列队系统通过网络发送消息来协作。从而,我们就将客户端-服务器构架中充当客户端的应用的功能划分(即,分布)为运行在两个或更多物理装置上的部件,这些装置使用诸如面向消息的中间件的消息列队系统通过网络连接相互通信。组成部件在大型的客户端-服务器配置中充当客户端,而服务器为例如邮件服务器。我们称此为‘分布式客户端’模型。分布式客户端模型的核心优势在于其允许终端,如具有有限处理能力、功率和连接性的移动装置,通过将通常与客户端相关的功能中的一部分分布到服务器侧而在移动装置上使用最少资源,来享受功能完备的客户端访问服务器环境的功能,服务器侧并没有如此地受资源限制。



1. 一种数据访问、复制或通信系统,包括分布于服务器侧组件和运行在终端上的终端侧组件的软件应用,

其中所述终端侧组件和所述服务器侧组件共同构建到服务器的客户端,并且

所述终端侧组件和所述服务器侧组件通过使用消息列队系统在网络上发送消息来经由该网络的连接协作,

其中列队的每个消息定义了部分或全部的事件,其中事件足够详细地描述了存储在所述终端或服务器上的数据的更改,以实现数据复制,而无需同步引擎;数据复制由发送事件实现,而不是由用于同步的存储数据的完整数据集或数据集的子集实现。

2. 根据权利要求 1 的系统,其中所述消息列队系统是面向消息的中间件。

3. 根据权利要求 1 的系统,其中在所述网络的连接被中断的情况下,所述终端侧组件使终端程序隔离,使所述终端程序不会被作好准备使所述连接被重新建立的队列消息所影响,从而使所述终端程序进入其下一个任务。

4. 根据权利要求 1 的系统,其中在所述网络的连接被中断的情况下,所述服务器侧组件使服务器程序隔离,使所述服务器程序不会被作好准备使所述连接被重新建立的队列消息所影响,使所述服务器程序进入其下一个任务。

5. 根据权利要求 1 的系统,其中所述终端侧组件可以创建事件并将这些事件列队,或所述终端侧组件可以创建事件并在消息列队系统中将这些事件列队,使所述终端侧组件进入其下一个任务,即使所述网络连接被中断。

6. 根据权利要求 1 的系统,其中所述服务器侧组件可以创建事件并将这些事件列队,或所述服务器侧组件可以创建事件并在消息列队系统中将这些事件列队,使所述服务器侧组件进入其下一个任务,即使所述网络连接被中断。

7. 根据权利要求 5 的系统,其中甚至在所述终端被关闭时,列队的事件也存留在非易失存储器中。

8. 根据权利要求 5 的系统,其中甚至在所述服务器被关闭时,列队的事件也存留在非易失存储器中。

9. 根据权利要求 1 的系统,其中所述终端侧组件和所述服务器侧组件共同地在运行于无线终端之上的终端程序和运行于服务器上的服务器程序之间构建中间件。

10. 根据权利要求 5 的系统,其中在所述终端侧上列队的消息是对保留在所述服务器上的数据的引用。

11. 根据权利要求 9 的系统,其中通过自动地发送终端侧队列中的下一个消息,如果通过网络连接被重新建立,则在所述终端侧上的消息队列系统使所述终端程序不受影响。

12. 根据权利要求 9 的系统,其中通过自动地发送服务器侧队列中的下一个消息,如果通过网络连接被重新建立,则在所述服务器侧上的消息队列系统使所述服务器程序不受影响。

13. 根据权利要求 1 的系统,其中所述终端侧组件处理来自终端程序的事件,所述来自终端程序的事件包括电子邮件应用、联系应用和日程表应用中的一个。

14. 根据权利要求 1 的系统,其中所述服务器侧组件处理来自服务器程序,即邮件服务器程序的事件。

15. 根据权利要求 1 的系统,其中所述终端是无线终端。

16. 根据权利要求 1 的系统,其中所述网络是无线 WAN 网络。
17. 根据权利要求 1 的系统,其中所述服务器侧组件存储从所述终端侧组件发送的登录密码,并且可以使用这个登录来访问服务器程序。
18. 根据权利要求 1 的系统,其中为了避免数据需要通过网络从所述终端发送,所述服务器侧组件可以通过使用保留在所述服务器上的数据来组装所述终端侧组件希望发送的消息。
19. 根据权利要求 1 的系统,其中所述终端侧组件监测所述终端上的可用内存,并自动删除存储在所述终端上的满足包括年龄、使用或大小中的一个或多个的预定标准的数据,而不影响存储在所述终端上的对应数据。
20. 根据权利要求 19 的系统,其中用于删除存储在所述终端上的数据、而且不影响存储在所述服务器上的对应数据的用户选项与用于删除存储在所述终端上的数据和存储在所述服务器上的对应数据的选项,在所述终端上显示的菜单层的同一级上显示。
21. 根据权利要求 19 的系统,其中所述数据为消息数据,并且如果用户请求的话,所述终端侧组件保留允许所述消息数据由服务器重新提供的数据。
22. 根据权利要求 19 的系统,其中如果数据标为未读,并且打开用于用户浏览或操作,或者存在涉及数据从大型服务器请求其他数据的待处理动作,则数据就不从内存中释放。
23. 根据权利要求 1 的系统,其中所述终端侧组件使得文件附件以所述文件在所述服务器上存储的原始格式,或者从原始格式转换而来的更为可用的格式发送到所述无线终端。
24. 根据权利要求 1 的系统,其中所述终端侧组件使得用户能够 (a) 选择释放选项来删除存储在所述终端上的消息,而不删除存储在所述服务器上的对应消息,也能够 (b) 选择删除选项来删除存储在所述终端上的消息,也删除存储在所述服务器上的对应消息,所述释放和删除选项出现在所述终端上显示的菜单层内的同一级。
25. 根据权利要求 1 的系统,其中所述应用通过将 ID 映射到需要抵达所述终端的真实 IP 地址,实现寻址到由所述 ID 所识别的终端的消息的正确路由。
26. 根据权利要求 25 的系统,其中所述地址是由 NAT box 分配的动态 IP 地址。
27. 根据权利要求 26 的系统,其中如果存在有效映射的话,所述应用仅发起消息传送。
28. 根据权利要求 27 的系统,其中无论什么时候从所述终端接收到了特定类型的小的专用消息,映射都被刷新。
29. 根据权利要求 1 的系统,其中所述终端侧组件允许服务器管理员锁定所述终端上的应用,而不影响所述终端上的其他应用。
30. 根据权利要求 1 的系统,其中所述终端组件向任何被怀疑试图拒绝所述终端上的服务攻击的第三方发送询问,而且所述服务攻击的拒绝不会导致任何到所述终端的其他数据业务。
31. 根据权利要求 1 的系统,其中所述应用包括调用分布式通信平台的分布式应用平台。
32. 根据权利要求 31 的系统,其中即使使用了不可靠的传输协议,所述通信平台也能使通过网络传送消息变得可靠,其中所述平台以与会话无关的方式运行。
33. 一种数据访问、复制或通信的方法,包括以下步骤:

(a) 运行分布于终端侧组件和服务器侧组件的软件应用,其中所述终端侧组件和所述服务器侧组件共同构建到服务器的客户端,以及

(b) 在所述终端侧组件和所述服务器侧组件之间使用消息列队系统在网络上发送消息,

其中列队的每个消息定义了部分或全部的事件,其中事件足够详细地描述了存储在所述终端或服务器上的数据的更改,以实现数据复制,而无需同步引擎;数据复制由发送事件实现,而不是由用于同步的存储数据的完整数据集或数据集的子集实现。

包含分布式软件应用的数据访问、复制或通信系统

技术领域

[0001] 本发明涉及包含分布式软件应用的数据访问、复制或通信系统。

背景技术

[0002] 诸如智能电话、移动电话、无线 PDA 等的无线终端通常受到严格的存储器、处理能力和电池的限制。其结果是，使用这一类型的终端来访问、复制或传递数据就表现出巨大的挑战。例如，如果一个人使用无线终端来存储其联系和日程表，还发送和接收电子邮件，那么，通常就出现将数据集与远程服务器上的主数据集同步的需求。服务器与移动装置之间的同步通常通过使用相对的高带宽、低延迟、非计量的连接（例如，USB 或 IR）而进行。其结果是，同步系统经常使用传输大量数据的方法，并且当数据在传输过程中丢失或当底层的通信被中断时并不是十分的健壮。例如，基于服务器的数据集同步通常要求所有连接的装置通过单个会话将它们的整个数据集（例如，所有的电子邮件，所有的联系等）下载到服务器，这样就能够执行与上一次完整同步数据集的主拷贝的比较，以便更新主拷贝并由此更新所有其他数据集。这种方法对于无线装置来说并无吸引力，原因是其强加的功率消耗、潜在的长连接时间和高昂的数据传送。

发明内容

[0003] 本发明设想了一种包含软件应用的数据访问、复制或通信系统，该软件应用跨越终端侧组件而分布，运行于终端和服务器侧组件之上。

[0004] 其中所述终端侧组件和所述服务器侧组件：(i) 共同构成到服务器的客户端并；(ii) 通过使用消息队列系统通过网络发送消息来协作。

[0005] 从而，我们就将客户端-服务器构架（例如，在 Microsoft Exchange 电子邮件环境中）中充当客户端的应用的功能划分（即，分布）为运行在两个或更多物理装置上的部件，这些装置使用诸如面向消息的中间件的消息队列系统通过网络连接相互通信。组成部件在大型的客户端-服务器配置中充当客户端，而服务器为例如 Exchange 邮件服务器。我们称此为‘分布式客户端’模型。分布式客户端模型的核心优势在于其允许终端，如具有有限处理能力、功率和连接性的无线装置，通过将通常与客户端相关的功能中的一部分分布到服务器侧而使用移动装置上的最少资源来享受功能完备的客户端访问服务器环境的功能，服务器侧不是如此地受资源限制。（同时本发明因此在当所述终端是无线终端（例如智能电话）而所述网络是无线网络（例如 GPRS）时特别实用，但并非要受限于这些实施例。从而，本发明也覆盖了例如通过有线网络与远程服务器通信的桌面 PC 的终端）。

[0006] 与其他分布式模型不同，在本发明的一种实现中，每个部件都能够提供与其它组件无关的功能（仅超过访问缓存/存储的数据），甚至是在不具备网络连接的时候。例如，在常规的分布式客户端中，例如使用 web 浏览器和访问 Microsoft Exchange 邮件服务器的 web 服务器分布式客户端的电子邮件 web 访问，如果你中断了浏览器与 web 服务器之间的连接，那么该浏览器除了继续显示任何缓存的电子邮件之外将而不具备任何功能。但是使

用本发明的一种实现,如果通过网络连接中断的话,通过同样排队等待为了将要重新创建的连接而准备就绪的消息,终端侧组件也可同样使终端程序(例如,联系、日程表、电子邮件等)免受其影响而使终端程序继续其下一个任务。由此,用户在终端(例如智能电话、PC)上使用电子邮件或PIM应用的经历就会完全相同,而无论是否有连接。用户可以继续在其终端上创建电子邮件,编辑联系等。用户从网络连接的状态中隔离开是因为终端可以使用消息队列系统排队等待消息,直到当连接被重新建立时接着就可以自动将它们发送出去。同样地,如果通过网络连接中断的话,服务器侧组件同样通过排队等待为了将要重新创建的连接而准备就绪的消息,可以使服务器程序免受其影响而使其继续其下一个任务。

[0007] 由于这种方法通过提出跨越网络的消息队列平台而使终端侧不受各种网络性能的影响,当所述网络是具有高度不可预知的带宽、覆盖率和可用性的无线网络时效果更好。从而,分布式客户端模型直接致力于解决在终端上提供良好的用户体验的问题(例如,最终用户总是能够与其PIM、日程表或电子邮件应用交互,而不管网络的覆盖率如何),不管所述终端具有较大的资源限制,也不管所述连接具有较大的限制。

[0008] 针对分布式客户端优化的一个实现细节是,每个排队的消息定义了‘事件’的部分或全部,其中事件足够详细地描述了存储在终端或服务器的数据的改变以使数据复制无需常规的同步引擎而进行;数据复制由此就通过发送‘事件’完成,而不是通过同步所存储数据的整个数据集(或数据集的子集)。终端侧组件能够创建事件并自己和/或在消息队列系统中排队这些事件,使得终端侧组件即使是在网络连接中断的情况下也能够继续其下一个任务。同样,服务器侧组件能够创建事件并自己和/或在消息队列系统中排队这些事件,使得服务器侧组件即使是在网络连接中断的情况下也能够继续其下一个任务。所排列的事件可以保留在终端上的非易失存储器中,即使是在无线终端被关闭的时候。同样,所排列的事件可以保留在运行服务器侧组件的主机上的非易失存储器中,即使是在服务器被关闭的时候(应该注意运行服务器侧组件的主机未必是运行服务器的相同主机,即使有可能是同一主机)。

[0009] 终端侧组件和服务器侧组件可以在运行于无线终端上的终端程序和运行于服务器上的服务器程序之间共同构成中间件。

[0010] 该应用也可以包括分布式应用平台,其调用到提供消息队列功能的分布式通信平台,其中所述通信平台使通过网络的消息的分发变得可靠,即使是使用不可靠的传输协议,其中所述通信平台以“会话无关”的方式运行。

具体实施方式

[0011] 1、引言

[0012] 本发明将参考来自Visto公司的实现而描述。该实现包括称为MobileMQ™的中间件通信平台和称为Transcend Mail™的分布式应用层。

[0013] Transcend Mail是运行在(即,也分布在)SymbianOS™智能电话无线终端和Windows 2000服务器上的端到端的GPRS-连接应用。它允许SymbianOS智能电话上的电子邮件、联系和日程表应用使用MobileMQ平台来通过GPRS与Microsoft Exchange™邮件服务器一起执行自动数据复制。MobileMQ是再次分布于智能电话和服务器的面向消息的中间件平台。它提供了实现了Transcend Mail用户体验的很多方面的GPRS-有效、可靠的和安

全的通信方法。无论是否具有通过网络（无论是有线还是无线）进行远程数据访问、复制或通信的需求，都可以使用 MobileMQ。MobileMQ 使用会话无关的不可靠的底层传输协议。

[0014] Transcend Mail 允许移动的员工可以从他们的基于 Symbian 的 GPRS 智能电话访问公司的邮件、联系和日程表条目。其设计用来满足移动员工的三方面需求：

[0015] 1. 其使得他们能够在离开其办公桌的同时保持“联系中”，使得他们对顾客、市场和业务需求保持及时反应；

[0016] 2. 其确保他们的联系和日程表保持“最新状态”以便他们可以与同事共同有效地协作；

[0017] 3. 其使得生产性使用能够同时在约会、等待运输或在旅行期间由“停滞时间”组成。

[0018] Transcend Mail 实现了以下特征：

[0019] ● 允许终端装置用户界面与本地数据一起工作，从而 GPRS 延迟，较低且可变的带宽和断断续续的覆盖率并不会阻止电子邮件、联系和日程表应用做到总是可用的并有响应；

[0020] ● 以及时的方式并且在后台自动复制本地数据和服务器侧数据之间的改变，而并不打扰用户；

[0021] ● 极少使用 GPRS 以便顾客从网络运营商的 GPRS 价目表得到合适的价格，并且不会得到有时被称为“bill shock”的价格；

[0022] ● 以顾客组织将他们的员工的 GPRS 移动电话连接到公司的局域网的方式提供了灵活性；

[0023] ● 尽可能无缝地与移动电话应用程序组结合以使用户不用必须学习新的系统；

[0024] ● 尽可能无缝地与后台最终邮件服务器结合以便 IT 管理员感觉舒适并且也具有良好的用户体验；

[0025] ● 终端装置上的不同应用可以独立发送 / 接收，从而即使是下载非常大的电子邮件附件的用户仍然可以更新其日程表。

[0026] 2、核心设计原理

[0027] 在本发明的一种实现中，我们将在客户端 - 服务器构架（例如，Microsoft Exchange 环境中）中充当客户端的 Transcend Mail 应用的功能划分（即，分布）为运行在两个或多个物理装置上的部件，这些装置使用 MobileMQ 面向消息的中间件通过广域连接相互通信。这些部件在大型的客户端 - 服务器配置中共同充当客户端，而服务器为 Exchange 邮件服务器。我们称此为“分布式客户端”模型。分布式客户端模型的核心优势在于其允许终端，如具有有限处理能力、功率和连接性的无线装置，通过将通常与客户端相关的功能中的一部分分布到服务器侧而在移动装置上使用最少资源，来享受功能完备的客户端访问服务器环境的功能，服务器侧不是如此地受资源限制。

[0028] 但是与其他分布式模型不同，在 Transcend Mail 中，每个部件都能够提供与其它部件无关的功能（仅超过访问缓存 / 存储的数据），甚至是在不具备网络连接的时候。例如，在常规的分布式客户端中，例如使用 web 浏览器和访问 Microsoft Exchange 邮件服务器的 web 服务器分布式客户端的电子邮件 web 访问，如果你中断了浏览器与 web 服务器之间的连接，那么该浏览器除了继续显示任何缓存的电子邮件之外将而不具备任何功能。但

是使用 Transcend Mail,用户在智能电话上使用电子邮件或 PIM(联系,日程表)应用的经历就完全相同,而无论是否有连接。该用户可以继续在其智能电话上创建电子邮件,编辑联系等。用户从网络连接的状态中隔离开是因为终端可以使用 MobileMQ(如果 MobileMQ 队列已满的话也使用其自己的队列)排列消息,直到当连接被重新建立时接着就可以自动发送下一个排队的消息。同样地,当重新建立连接时就能够自动发送存储在服务器侧的下一个排队的消息。

[0029] 当网络是具有高度不可预知的带宽、覆盖率和可用性的无线网络时该方法也更加有效,因为它通过提出跨越网络的消息队列平台(即,MobileMQ)而使得终端侧免受网络性能难以预测的变化的影响。从而,分布式客户端模型不仅仅是直接致力于解决在终端上提供优良的用户体验的问题(例如,最终用户总是能够与其联系、日程表和电子邮件应用交互,而不管网络的覆盖率如何),不管该终端具有较大的资源限制,也不管该连接具有较大的限制。这在图 1 中示意性地示意了。

[0030] MobileMQ 面向消息的中间件(MOM)通信平台使得部件能够有效并高效地存储将要在部件之间发送的消息并实际上可靠地将它们发送出去。MobileMQ MOM使得通过网络的消息传递可靠,无论底层使用的传输协议是否可靠;并且与任何会话无关。该传输协议因此被称为‘会话无关’的协议。这又标志着与诸如基于会话的 WAP 的常规网络协议的主要的不同。因此‘会话无关’(如先前所注明)就与基于会话相对立。当数据传送率降低到低于极限值时或在超时之后基于会话的协议就会取消会话。协议就会发起一次要求交换数据业务的新的会话,该会话(a)在用户为所传送的数据量付费的无线系统中开销巨大,并且(b)由于与无线网络有关的高度可变的带宽而时常发生。然而会话无关的协议不会有意取消连接。(我们在此区分不同的情况,其中基础 PDP 环境仅仅为了由其他应用使用而释放或再分配,或者当前基础 PDP 环境丢失或否则超时。在这些类型的情况下,更高层的会话无关的协议仅仅暂停通信活动,挂起 PDP 环境的重新取得-因此它无需重新创建“会话”)。

[0031] MobileMQ MOM 和会话无关的组合致力于解决很多受连接约束的问题,如图 2 所示。

[0032] 另外,使用会话无关的协议,因此同样没有会话的重新创建;相反,发送组件一旦知道怎样将其消息定位到分布式客户端的接收组件就简单地重新启动发送。这在无线环境中尤其有用,因为无线装置根本就没有(除非并且直到可使用 IPv6)持久的 IP 地址,而是仅能够通过频繁改变的任意选择的 IP 地址接收基于 IP 的消息。从而,会话无关很适合使用临时地址向无线装置发送消息的需求,具有应付这些变化的地址所需的最小化开销,因为它避免了重新创建会话的开销。

[0033] Transcend Mail 也采用基于‘事件’的数据复制-即保留了所有新的事件的日志,这些事件定义了存储在客户端或服务器侧的数据的改变,并且仅有这些事件在日志中排队。当分布式客户端的每一方之间的连接建立的时候,那么每个排队的事件(根据事件的复杂程度由一个或更多消息表示)就被发送出去,同时还有由一个或更多分组传送的单个消息。由于 MobileMQ 允许消息的高效和可靠传输,尤其与基于事件的数据复制系统相配,其本身只是需要消息的高效和可靠传输。

[0034] 使用 MOM 的更高层应用并不了解连接的状态-即,MobileMQ 为 MOM 层,它将使用 MobileMQ 的应用从需要了解底层连接存在或不存在的状况的应用(如活动的 PDP 环境)分

离开。在应用层,系统是“连接无关”的。因此术语“会话无关”在应用层在概念上等价于一个系统,其中具有在某种意义上保持着独立于无论实际的连接是建立与否的单独会话,即当连接经过中断后重新建立时,应用能够正好在其停止的同一地点重新发起消息的传输 – 即,下一个由应用发送的消息是在没有连接中断的情况下应该已经被发送的同一个消息。在基于会话的系统中,这些就不会发生。相反,首先考虑的是重新创建会话的相当大的开销。一旦会话重新创建,消息传输就重新发起,即使重新发起消息传输可能也包含前一个传输的消息数据的重新传输的巨大部分 – 还是更加非生产性的数据传送。

[0035] 在 MobileMQ 中,获得可靠性(即,能够保证消息已经被接收)无需求助于基于会话协议的高开销。反之,我们要求在发送装置收到到个别消息已经被接收的确认,并在允许另外一个消息被发送之前在接收装置进行了适当的处理。这个“消息级别”可靠性更加具有数据效率,因为它与基于会话的系统不同,如果连接丢失的话,它只需要重新启动通信的最小开销。这是个非常重要的优势,尤其在无线网络中,其中连接丢失的现象并不罕见。

[0036] 在现有技术中,在会话启动时使用高的数据传输开销而实现验证。因为 MobileMQ MOM 是会话无关的,其转而为每个消息提供验证:“消息级别”的验证。这可以基于无论装置是否重新启动都增加的会话数量而与会话级别的验证组合。

[0037] 当 MobileMQ 与 Trascend Mail 组合时对有效处理终端资源和连接约束这两个问题来说是一种全面而高效的方案,如图 3 所示。

[0038] 图 4 示出了组合的设计是怎样允许由 MOM 提供通常是由基于会话的系统提供的会话无关的特征以适合诸如智能电话等的受资源限制的终端的目的方式部署的。这些特征包括:

[0039] (a) 消息传递的可靠性;

[0040] (b) 发送方验证;

[0041] (c) 消息安全性;

[0042] (d) 数据率流量控制;

[0043] (e) 分组路由。

[0044] 同样,图 4 示出了组合的设计是怎样允许特征被设计以适合于分布式客户端的目的并怎样解决连接限制的问题。主要的特征是使用基于‘事件’的数据,并且由于这个原因得到了作为会话无关的特性。

[0045] 3、更进一步的核心设计原理

[0046] 3.1 分布式客户端

[0047] 分布式客户端的目的是允许具有有限的处理容量,功率和连通性的移动装置通过将通常与客户端相关的一些功能分布到服务器侧而在移动装置上使用最小化资源,享受完全功能的客户端到服务器环境的访问的功能,这样就不受资源限制。该客户端应用为 PIM(日程表,地址簿等)和消息传递(电子邮件, MMS, 传真, SMS 等)。

[0048] 实际上,我们具有在大型的客户端 – 服务器应用中共同充当客户端的单独客户端 – 服务器应用。典型的配置是在诸如智能电话的移动装置(小型客户端)上安装软件和在服务器装置(小型服务器)上安装对应的软件。这两个软件与对方通信……通常是通过高延迟,低带宽,计量的无线连通性,如 GPRS 或 UMTS,经由 MOM,如 MobileMQ。小型客户端和小型服务器在传统的客户端 – 服务器环境中共同充当客户端(“大型客户端”或“分布

式客户端”)。大型客户端就像平常一样与相关的服务器如 Microsoft 的 Exchange (大型服务器) 通信。这样,大型服务器就仅仅知道它在与大型客户端通信。大型客户端对大型服务器来说就像其他与之正常通信的客户端,并且大型客户端的分布式特性对大型服务器是不可见的。(在一种典型的配置中,假设小型服务器在大型服务器所驻留的同一装置之上或者附近。)图 5 示意性地例举了这一点。

[0049] 分布式客户端内部的功能在小型服务器和小型客户端之间被划分。这种划分已经被优化以利用移动装置的“智能”特性,同时限制了对装置的有限资源的影响。

[0050] 这样,驻留在移动装置上的小型客户端(即,Transcend Mail,在这种情况下与本地的电子邮件应用联合工作)通常给终端用户用作用户界面,用作本地数据仓库,并承担了本地的特定数据处理的任务。在其它特征中,小型服务器承担以下功能:

[0051] ●在移动的收件箱中显示邮件列表

[0052] ●用作邮件文本主体的浏览器

[0053] ●接收用户请求以转发,创建或回复邮件

[0054] ●接收用户新邮件文本的输入

[0055] ●响应用户输入,将邮件仅从移动装置存储器中释放(释放动作)

[0056] ●响应用户输入,将邮件从移动装置存储器中释放并产生将相同邮件从大型服务器中释放的通知(删除请求)

[0057] ●接收最终用户用于大型服务器登录密码的输入并将其传送到小型服务器(查看下面关于“分布式登录”的描述)

[0058] ●监测用于那些数据(如新建的,修改的或删除的条目)的变化的本地数据仓库,创建详细叙述该变化的事件,并将这些发送给小型服务器。

[0059] ●从小型服务器接收事件,并使用更改的数据的细节来更新本地数据仓库。

[0060] 等价或类似的 PIM 和非电子邮件的功能将在小型客户端处理/集成到 PIM 和非电子邮件(例如,其它类型的消息)的地方处理。

[0061] 与之类似,小型服务器位于其它媒体(典型为连接到大型服务器的 LAN)上,通过贯通无线设施(如 GPRS)的数据网络连接与移动装置通信,通常充当到大型服务器(如 Microsoft 的 Exchange)的直接接口并承担很多通常与大型客户端相关的数据处理任务。在其它功能中,小型服务器承担以下功能:

[0062] ●按照大型服务器的 API 完成由移动用户所请求的电子邮件的结构,获取从小型客户端,小型服务器,和如果必要的话从大型服务器接收的组件

[0063] ●获取来自大型服务器的电子邮件并将这些划分为部件,仅将这些被认为是严格必要的部分发送到小型服务器

[0064] ●响应来自小型服务器的请求以传递其它电子邮件内容(如长的电子邮件和/或附件的其它文本)

[0065] ●获取登录密码数据(由最终用户通过小型服务器提供)并将这些数据存储在本地存储器中,从而实现到大型服务器的扩展登录(查看以下分布式登录的描述)

[0066] ●监测大型服务器上用于那些数据(如新建的,修改的或删除的条目)更改的本地数据仓库,创建详细叙述该变化的事件,并将这些发送给小型服务器。

[0067] ●从小型服务器接收事件,并在大型服务器上使用变化数据的细节来处理本地数

据仓库的更新

[0068] 等价或类似的 PIM 和非电子邮件的功能可能在小型客户端运行 / 集成到 PIM 和非电子邮件 (例如, 其它类型的消息) 功能的地方得到运行。

[0069] 这种方法的一个结果是, 在大型服务器上的邮件不能被说成是以常规推动电子邮件系统的方式“转发”或“重寄”到设备; 而是与小型服务器一起, 该设备 (小型服务器所位于的) 仅是到邮件服务器的另外一个客户端 (并且也不仅仅是无线设备客户端)。

[0070] Transcend Mail 小型服务器和小型客户端通过 MobileMQ MOM 相互通信。如上所注, 这使得小型服务器和小型客户端 (在分布式客户端服务器系统中一起构建大型客户端) 的与众不同的特征相互之间异步运行。

[0071] 小型服务器有各种排列方式: 例如, 如图 6 所示, 小型服务器可以作为终端侧组件实现, 该组件包含客户端侧 MOM 或与客户端侧 MOM 通信; 小型服务器就可以作为服务器侧组件实现, 该组件包含服务器侧 MOM 或与服务器侧 MOM 通信。

[0072] 图 7 展示了小型客户端是怎样能够包含程序—例如, 电子邮件应用, 和将其连接到终端侧组件的插件。

[0073] 图 8 展示了小型客户端也可以不包含程序, 例如联系程序。终端侧组件就通过数据库与联系程序通信 (使用来自发送到终端侧组件的数据库的事件触发)。图 9 展示了这在概念上是怎样等价于中间件结构的。

[0074] 3.2 分布式登录

[0075] 分布式客户端将小型客户端和小型服务器之间的登录功能划分开。小型客户端从最终用户的输入获取用户密码。接收到输入的密码后, 小型客户端将这些数据发送到小型服务器。小型服务器将这些数据本地保留在存储器中, 然后将该登录传递到大型服务器。

[0076] 这样, Transcend Mail 在服务器上作为无线终端用户的代理; 它隐藏密码并执行其它登录相关的动作, 这些动作正常情况下是由用户在与邮件服务器交互操作的时候执行。仅有当密码被邮件服务器认为过期时, 用户才因此需要亲自使用新密码再次登录。这在安全性方面, 比起其它要求作为能够访问所有邮件记录的超级用户登录的邮件重定向方法来说是一个大的提高。

[0077] 以这种方式分布式登录的另外一个好处是, 这样允许分布式客户端可以不需其它的登录活动 (和相关的无线数据业务) 而继续与大型服务器通信, 即使在小型服务器和小型客户端之间的通信中有中断。这也允许持有和使用移动装置 (可以拥有自己的安全协议) 作为大型服务器登录活动的替代。一旦小型服务器具有了密码信息, 就有可能复制到大型服务器的登录, 这是在移动装置的控制是作为登录信息替代的假设的基础上的。这允许了分布式客户端在服务中断之后无需提示用户其它密码数据而访问大型服务器。这样通过减少与登录活动相关的数据业务而提供了很多优势, 并且也加速了最终用户对大型服务器的访问, 尤其是在具有便于输入密码数据的小键盘的移动装置上。

[0078] 3.3 远程消息结构

[0079] Transcend Mail 也通过应用远程构建消息的方法将无线通信的使用最小化。当分布式客户端从大型服务器“重新获取”消息时, 数据中的一些在小型服务器上列队, 而小型服务器就判断这些数据中有多少要发送到小型客户端。要发送到小型客户端的指定数量的数据会受到最终用户配置和特定要求的影响。例如, 最终用户可以指定发送到小型客户端

的电子邮件文本行数的最大数目,而最终用户就可以要求小型服务器发送其它的文本和/或附件。远程消息结构在最终用户执行如回复或转发电子邮件的操作的时候起作用。当转发带有未修改附件的电子邮件时,小型服务器能够在本地构建电子邮件消息的大部分而不需要来自小型客户端的完整通信。列队的消息只是参考由大型服务器具有的相关未修改的数据(在本例中,附件)。实际上,小型服务器就只是被要求用来发送已经在小型客户端装置上由用户修改的那部分消息。相同的规则应用于邮件回复。小型客户端不需要发送原始消息的整个主体—小型服务器通过获取从小型客户端发送到小型服务器的新文本并将其与原始电子邮件文本组合来构建全部回复—由于其存储在大型服务器上而早已为小型服务器所知。结果,小型客户端发布指示,该指示最终导致小型服务器不但从小型客户端还从大型服务器获取部分消息而构建整个消息。避免不必要的键输入对于小键盘智能电话来说是很有价值的。这些是有效减少小型客户端和小型服务器之间的数据业务量的两种方法。

[0080] 3.4 整理处理

[0081] Transcend Mail 也帮助在移动装置上使用自动内存“整理”功能保存存储器。小型客户端监测移动装置上非易失存储器的使用情况。当使用中的非易失存储器超过了触发量(例如,使用中存储器容量的 80%)时,存储器就在该装置上自动开始“整理”涉及电子邮件信息的本地数据。这包括选择某些已经很长时间没有在本本地访问的电子邮件,将与这些电子邮件相关的很多数据(例如,附件和消息文本)从本地存储器中释放,同时在本本地存储器中保留邮件头信息。使用预先设定的标准比如相关邮件消息的年龄(最老的放在第一位)或用户访问这个邮件的历史来选择从本地存储器中释放的数据;很长时间没有访问的邮件消息在较新的或新近访问的邮件消息之前从本地存储器中移除。

[0082] 如果相关消息标为未读,正在打开被用户浏览或其它动作,或者还有从大型服务器要求其它数据的涉及邮件的未完成的动作的话,消息数据就不从本地存储器中释放。

[0083] 这些移除处理一直持续,直至小型客户端检测到使用中的非易失存储器已经下降到预先设定的“安全”级(例如,使用中存储器的 70%)以下为止。由于大型服务器上保留了所有的邮件数据而小型服务器上保留了邮件头数据,邮件数据没有被“删除”。

[0084] 移除的数据可以通过在用户的请求(“重新获取”的动作)下将其再一次从服务器上下载到移动装置上而重新放置到本地存储器中。用户在这种重新获取发生之前受到明确的警告。这就允许了用户有机会判断重新获取该消息是否成本有效。

[0085] 整理功能可包括处理一些情况的安全机构,其中特别大的邮件或者附件可以无需要回顾所下载内容而将移动装置推到整理的触发点。换言之,该设计意味着避免移动装置可能在邮件被读取之前就移除邮件的情况。在这种情况下,系统临时调整存储器的触发器级别和安全级别以允许最终用户有重新浏览大的入境邮件的机会。

[0086] 3.5 转换的附件下载选项

[0087] 在与服务器共享所复制数据的移动装置中长期具有的困难是用户怎样能够最好的利用有限的内存和处理移动装置上的容量。在移动电子邮件应用中,这个问题常规(在 MS ActiveSync 和其他应用中)是通过限制与该移动装置共享的邮件数据的数量来解决。这种限制通常采取仅复制所限制数量天数的邮件数据,并限制共享邮件数据的大小的方式。限制所复制数据数量的一个通用的方法是拒绝在移动装置上复制邮件附件。通常,如果最终用户愿意忍受传送时间和所涉及的内存开销的话就可以要求下载附件。其他系统已

经生成了给用户仅发送所限制格式的附件文件的方法,该文件就可以在简单的浏览器程序上使用。

[0088] Transcend Mail 给最终用户提供了两个用于将邮件附件下载到移动装置上的选项。如果用户希望下载附件的话,他或她就可以在以下下载选项中进行选择:

[0089] (1) 从文件的原始格式转化而来的较小版本(例如,MS Word 或 PDF 文件的纯文本版本),为了使用在无线终端上的浏览器程序来单纯的浏览。

[0090] (2) 未改变的原始附件。

[0091] 在 Transcend Mail 中这些选项在菜单层的相同级上呈现给用户。

[0092] 3.6 一触释放

[0093] 当常规的移动装置开始受到旧邮件的阻碍时产生了新的问题。经常是当用户试图从移动装置上“删除”这些邮件时,这些邮件也就在下一个同步从服务器上删除。在面对用户需要的时候这可以直接执行(fly)-这些用户仅仅希望释放移动装置上的内存,并且仍在服务器上保留邮件。为了与之相适应,一些同步系统给最终用户提供了选项,或者:

[0094] (1) 在移动装置和服务器上都将整个邮件删除(我们称为“删除”指示),或者

[0095] (2) 仅删除保留在移动装置上的邮件拷贝—服务器上保持未修改(我们称为“释放”指示)

[0096] 在 Transcend Mail 中这些选项在菜单层的相同级别上呈现—其他的系统或者在一系列不同的菜单级内隐藏了这些选项,或者通过在每次请求删除时询问每一个用户是否最终用户实际上是想完全删除还是仅仅清除本地内存来执行这种选项。这两种方案都不是对用户友好的,因为它们都降低了管理移动邮箱的灵活性并且要求输入多个键来完成。

[0097] 3.7 会话无关

[0098] 使用诸如 GPRS 的技术与移动通信装置进行的数据通信由于高延迟,断续而中断的覆盖率,和计量带宽的花费而变得极端困难。常规的通信方法和协议并不能很好的适应与这类环境。例如,通过诸如 GPRS 的无线链路要求网络连接的应用通常使用 TCP 连接,因为这种连接提供可靠的基于会话的连接。然而, TCP 规则是开发用于有线和相对低延迟的连接,并不是“对无线友好”的。在覆盖范围很有限的地区,可用带宽降低,而假设网络拥堵并且““补偿”的 TCP 又加剧了这种情况。这种现象的直接结果就是通过蜂窝无线网络的 TCP 连接并不是有效的传输。这导致重新发送的分组的巨大开销,该开销导致缓慢而昂贵的数据传送。另外,类如 TCP 的协议依赖于与服务器通信“会话”的概念。如果在已定义的时期内没有业务量流通(超时)的话,会话通常会过期。建立每一个新的会话都要求另外的数据业务量的使用,而且也都是费时的。

[0099] MobileMQ 设想了仅使用原始 UDP 分组传送的对 TCP 的无线优化代替方案。MobileMQ 在 (1) 使用无线网络的移动装置和 (2) 与该装置通信(无论是否直接连接到无线网络)的服务器之间发送基于 UDP 的消息,从而使得作为高延迟和断续连接的结果而被浪费的数据业务量最小化。MobileMQ 集中于在消息发送过程中提供高水平的适应力,由此有效保证了消息传送。

[0100] 这使用了管理数据通信的系统而完成,该系统不依赖于常规的“会话”概念—它是“会话无关”的。另外,本发明提供保证消息不但很好的传递到目的装置,而且传递到目的应用,同时将所发送的数据业务量最小化的方法。这增强了保证高度适应力同时具备最小数

据业务量的优点。

[0101] MobileMQ 是分布式的是因为其不但位于发送装置上而且位于接收装置上,通常推动了来自同一硬件平台上的其他系统的消息业务。

[0102] 发送装置从发送应用(例如,电子邮件程序)获取消息—核心通信单元。每个消息限定在意图优化数据业务量使用的最大大小范围内。当发送应用请求系统发送消息时,系统首先在位于该发送装置上的本地非易失存储器中保留(存储)该消息。这就保证了即使该发送装置被重启了消息也能保留下来。该消息就被压缩并可以选择被加密。

[0103] 该消息于是被分段用于发送。每个段都放置在 UDP 分组中,每个分组都不超过相对较短的字节长度,这些涉及底层的低带宽高延迟的传输协议。在 GPRS 环境内的典型实现中,UDP 分组长度可以限制在 1500 字节,因为 1500 字节是 GPRS 分组中的典型最大有效载荷。另外如果 UDP 分组要占据,就说,2GPRS 片段,那么一个 GPRS 分组的失败将意味着两个 GPRS 分组都必须重新发送。MobileMQ 通过将消息缩放以匹配承载分组的消息来避免这一点。

[0104] 3.8 流量控制

[0105] 分片大小和传送率都是由流量控制系统控制,该系统分析发往和来自发送装置的业务量并努力在传输速度和所传输的总比特数之间寻找平衡以保持传输成本有效。UDP 分组可以由接收装置以任何顺序接收,而该接收装置在接收到每个分组后发分组确认。在发送装置没有成功接收到分组确认的地方,流量控制系统就延迟分组重传直至已经经过了一个合理的时间间隔。延迟的确切长度依赖于由流量控制系统在发送装置上检测到的网络响应时间。延迟时间间隔和分组大小都基于所检测的返回时间中的更改而不断重新计算。如果分组确认在预定时间内接收到了,流量控制就渐渐增加数据率直至达到最大值。

[0106] 流量控制系统也作为经常用于数据发送装置中的常规“会话”或“超时”概念的替代。如果通信装置中的一个发生了严重的连接性错误(这可能由于,例如,移出了无线基站的范围或在漫游的网络之间移动而引发),流量控制装置就将这理解为越来越慢的网络响应,并从而降低传输速率。如果服务损耗继续的话,流量控制就继续延长“重试”之间的时间间隔。直接的影响就是这次传输努力就几乎完全停止,直至发送装置开始从接收装置接收返回的业务。当更多的回复分组从接收装置返回到发送装置时该过程就颠倒了:流量控制系统开始“唤醒”并在其意愿中变得更加愿意发送分组。当连接变成更加坚固的重新建立时,传输速率就再次升高直至其达到合理的理想水平—与严格限制包丢失的需求一起平衡整体速度(因为丢失的分组仍可以招致网络传输负荷)。这样 MobileMQ 就不依赖于“会话”的概念,也不识别“超时”的概念。

[0107] 在所有包含消息的分组的接收之后,接收装置发送整个消息都已接收的简要确认。一旦该消息确认到达发送装置,发送装置就不再试图进行任何进一步的组成消息的分组的重发—即使个别的分组确认没有被发送到发送装置或由发送装置接收。这主要是意图限制数据业务的数量。消息传递序列还不完善。

[0108] 在发送了所接收的消息确认之后,接收装置就将消息传递到相关的目的应用(例如电子邮件程序)。接收装置就等候来自目的应用关于相关信息已经被接收并按照由接收应用所使用的任何设定的规则进行了处理的信号。其目的在于接收应用处理接收到的消息以便其在接收装置上经过崩溃—例如系统重启之后仍能保留下来。一旦接收应用对

其已经接收到不可逆的来自 MobileMQ 的消息感到满意,接收应用就用该应用已经“消耗掉 (consume)”该消息的最终确认来响应。来自接收应用的这个最终确认就触发 MobileMQ 的接收装置侧,以发送简要的“消息已消耗掉”的确认到发送装置。

[0109] 一旦发送装置接收到了“消息已消耗掉”的确认,就将该信息转发到发送应用并准备发送来自发送应用的下一个可用消息。以这种方式,MobileMQ 确保了在从发送应用接收任何其它消息业务之前的整个消息的传递。

[0110] MobileMQ 能够同时处理来自多个应用的消息,但是不能够同时处理来自同一应用的多于一个的消息。

[0111] 3.9 基于事件的数据复制

[0112] 服务器和移动装置之间的同步常规是使用相对较高的带宽,较低的延迟,无计量的连接性(如 USB 或 IR)而进行。其结果是,同步系统经常使用发送大量数据而且当数据在发送过程中丢失或者底层的通信被中断时并不健壮的方法。例如,基于服务器的数据集同步通常要求所有相连的装置通过单独的会话将其整个数据集(例如,所有的电子邮件,所有的联系等)下载到服务器,这就会执行与上一次完整同步的数据集的主拷贝的比较以更新主数据集并从而更新其它的数据集。由于其强加的功率损耗,潜在的长连接时间和昂贵的数据传送代价,这种方法对于同步无线装置来说并不吸引人。

[0113] 在 Transcend Mail 中,代替无线装置下载其整个数据集的是,它仅将数据集的更改(或新的“事件”)记录进日志(优选但不必须为以时间顺序),并在连接到服务器时将这些事件的日志发送到服务器。事件提供足够的细节以使数据复制无需同步引擎的需求而发生;数据复制(与真正的同步相对)通过发送事件而不是发送在小型服务器中由同步引擎所存储的用于同步的数据的整个数据集(或数据集的子集)而更加简单的完成。

[0114] 只要在装置上的记录发生了改变(例如,创建了新的邮件并从设备上发送;老的邮件被删除;新的联系被创建等),定义正好这个事件或者更改的条目就在以时间顺序的日志里存储在服务器上;这个事件日志就一直存储直至连接出现,在这时日志内容就被发送到服务器,从而更新了有关数据集的主拷贝。例如,时间可以是‘删除记录 no. x’或在记录‘z’中删除字段‘y’。这些对于接收方来说为了在产生该事件的发送方复制该更改是足够的信息。

[0115] 对设备来说不需要传递整个数据集来判断已经更改了的记录或保证单个会话的维持,同时数据集被发送和接收。在服务器上数据集的任何改变(例如,新邮件的接收)也作为事件日志存储并且该日志使用 MobileMQ 发送到无线装置。因为只生成和交换了相对较小的事件日志,CPU 和数据传送开销要远远小于常规的同步机制。

[0116] 从而,当要复制的数据被输入,修改或删除(在大型服务器或小型客户端上)时,发送装置就在发送装置上创建“事件”并将“事件”记录到日志。

[0117] 事件就作为一个或更多的消息发送到接收装置;消息使用 UDP 分组传送而并非更常规的 TCP 发送。同时 TCP 提供可靠的连接并通常成为商业活动的焦点,它对于无线来说不是(如上所释)有效的传输,因为在感知到网络拥堵的时间内其有明显的补偿,这种现象在无线覆盖率很有限的时候并不罕见,并会导致重新发送分组的巨大开销,导致慢而昂贵的数据传递。为了高效性,UDP 将 UDP 分组大小限制在 1400 字节—恰好在 GPRS 中可用的传输分组大小之下(见上面)。

[0118] 接收装置将定义了事件的个别信息传递到相关目的应用（如电子邮件程序）。发送装置就等待来自目的应用关于定义了事件的相关信息已经被接收到并按照由接收应用使用的任何设定的规则进行了处理的信号。其意图在于接收应用处理接收到的消息以便其在接收装置内的崩溃—例如系统重启之后还能保留下来。一旦接收应用对其已经不可逆接收到消息感到满意，接收应用就响应该应用已经“消耗掉”该消息的最终确认。来自接收应用的这个最终确认就触发接收装置方以发送简要的“消息已消耗掉”的确认到接收装置。

[0119] 一旦发送装置接收到了“消息已消耗掉”的确认，它就对所有定义了事件的消息重复该处理，直至消息已经在接收装置上被安全接收并‘消耗掉’为止。然后它就从发送装置内存中删除 [事件指示信息？] 来终结“事件”处理。它对事件日志或列队里的所有其它事件都重复这种处理。

[0120] 以这种方式，Transcend Mail 确保了从发送应用接收任何其它消息业务之前的整个消息的传递。同时处理来自多个应用的消息是可能的，但是不能够同时处理来自同一应用的多于一个的消息。

[0121] 因此整个的事件可以通过无线链路可靠发送，尽管是使用了不可靠的 UDP。

[0122] 3.10 A/B/X 标志

[0123] 系统也会通过使用带有三个状态选项：A, B 或 X 的标志给每个消息编码而预防重复的消息发送。在常规的操作中，每个消息 [来自应用] 由发送装置使用可选的 A 或 B 标志发送。当接收装置开始接收消息时，就将 A 或 B 标志写入到本地存储器中。当接收来自发送装置的完整消息和来自发送应用的已消耗掉信号时，接收装置就将恰好在发送消息完成 / 消息已消耗掉确认之前处理的消息的标志识别写入到本地存储器。如果接收装置在发送消息完成 / 已消耗掉确认信号之后，但是在确认被接收回来之前重启，那么它就不会知道所消耗的消息是否已经被完整的接收。但是如果对关于给定消息的确认序列使用一类标志做出了标记的话，那么它就会知道任何返回的确认为了要与之相关都必须符合该标志。具有不同标志的确认必须涉及下一个消息并且从而不应该被处理到。

[0124] X 标志意味着接收装置忽略该标志，并且没有标志被写入到接收装置的存储器。其目的在于如果发送应用不关心重复消息发送的风险则给发送应用使用 X 标志。

[0125] 3.11 客户端装置定位与网络更新

[0126] 在很多现有的移动数据网络（如那些使用 GPRS 的）上发送数据传输到移动装置非常困难，因为移动装置没有固定的 IP 地址。相反，当移动装置连接到网络上时（在本地网络或漫游网络），网络运营商就给这些设备动态的分配 IP 地址。此外，动态分配的地址通常是私有 IP 地址，不能在公共因特网上直接使用。而是，来自该设备的数据业务量被网络运营商发送到网络地址转换 (NAT)，而 NAT 将私有 IP 地址映射到公共 IP 地址和从非常有限的公用地址（有时只是两个）列表中抽出的临时端口号和一大组由网络运营商使用的可用临时端口号（通常是几千）。

[0127] 这样即使移动装置将同一个动态分配的 IP 地址和临时端口号保留了很长时间（例如几个小时），也可以利用由网络运营商分配的多个“公用”IP 地址和临时端口号。此外，虽然移动装置知道由网络运营商给它分配的私有 IP 地址，它也没有由 NAT 分配的公用 IP 地址和临时端口号的记录。从任何一个与移动装置通信的装置的观点来看，他们都只能“看到”由 NAT 分配的公用 IP 地址和临时端口号。这就给任何一个希望发起并发送一个数

据消息用于传输到使用这样一个网络的移动装置的人带来了巨大的挑战,因为不能确保最后了解的公用 IP 地址和与给定装置相关的临时端口号在多于几分钟的时间内都还是有效的。

[0128] MobileMQ 有规律地给小型服务器提供网络地址数据以实现消息从小型服务器到移动装置的例行发送。实现的例子将包括使办公室邮件服务器发送邮件业务到移动用户而无需移动用户的介入。

[0129] 该方法包括基于以下任何事件的发生而从移动装置发送非常短的消息(‘网络更新’)到小型服务器:

[0130] ●当移动装置第一次打开并从移动网络运营商获取地址时;

[0131] ●当移动装置从网络运营商接收新的地址时(可能作为从本地网络覆盖到漫游网络或者在漫游网络之间移动该装置的结果);

[0132] ●无论是否分配新的地址到移动装置,以规律的时间努力获得可能由介入的 NAT 分配的新的公用地址和临时端口号,并将这个新的公用地址和临时端口号通知到小型服务器

[0133] 通过接收短的网络更新消息,小型服务器记录分组的原始 IP 地址和临时端口号(将是来自 NAT 分配的公用 IP 地址和临时端口号,假设有关的一方不直接连接到相同的私有 IP 网络)并将这种信息输入到反向查询表。

[0134] 由于假设以计量为基础对数据业务收费,网络更新消息特意很短。典型的实现可能包括只有 17 字节的由移动装置发送的数据和在每个网络更新消息周期中返回的 5 字节(假设没有分组丢失)。

[0135] 这些消息中的每个是用于确认:

[0136] (1) 公用网络地址的持续有效性和

[0137] (2) 移动装置可以用来接收业务量。

[0138] 小型服务器就能够通过给在反向查询表中发现的装置使用最近的地址而尝试其自己—未提示的一到移动装置上的小型客户端的数据传输,假设分配的公用 IP 地址和临时端口号从网络更新消息的时间开始没有再次分配。于是,在小型服务器上来自该装置的网络更新消息的接收作为启动发送在事件日志中列队的任何事件的触发器(查看基于事件的数据复制部分 3.9)。可以配置系统以便只有在网络更新被接收到的时间上出现在日志中的事件才被发送;任何晚一点的事件都必须等待,直至收到了下一个网络更新。这与连续推动电子邮件不同。

[0139] 反向查询表中的登录也是限时的,并且如果多于一定数量的时间已经流逝了的话小型服务器就假设不再可能发送消息到小型客户端,直至接收到新的网络更新消息。在这种情况下,来自小型服务器超出范围的消息就保持在队列里,直到接收到新的网络更新消息。时间被设置为基本上少于 NAT 使用来重新分配公共 IP 地址到移动装置的正常时间间隔(例如,如果 NAT 重新分配时间间隔为 20 分钟的话就是 5 分钟)。系统可以动态调整该时间以便当具有很高的网络使用率时,相关具有短得多的 NAT 重新分配时间间隔,那么该时间就可以缩短。

[0140] 总的来说,这意味着小型服务器和小型客户端能够建立时间窗口,在此期间对小型服务器来说可能发送业务到小型客户端。该窗口在网络更新消息的时间上开始,并在预

先编入程序的空闲时间过期的时候结束。例如,如果网络更新消息每 60 分钟由小型客户端发送,而空闲时间设置为 10 分钟,这就导致不超过 60 分钟的期间内复现的 10 分钟通信窗口。通过提高网络更新消息的频率,小型客户端也可以给小型服务器创建更多或更少的连续通信传输机会。

[0141] 3.12 安全性

[0142] 现存的通过非安全数据通信设施(如 SSL)的端对端通信保证安全的安全性方法由于很多因素,包括高处理器开销,高带宽开销,低延迟,和动态分配地址信息到移动装置等,不能够很好的适应移动远程通信环境。

[0143] MobileMQ 使用专门设计用于移动电信装置的加密实现来在移动装置和服务器之间提供安全的端对端消息传递。

[0144] 该过程在系统第一次安装到移动电信装置上时开始。移动装置(例如,连接到无线网络上的移动邮件阅读器)和服务器(例如,连接到固定有线因特网服务的共同邮件服务器)都使用共享的私密信息加载。为了在它们之间安全的通信消息,发送装置(移动电信装置或是服务器)首先通过使用哈希函数从下述输入计算哈希(hash)来计算消息密钥:

[0145] ●对相关移动装置唯一的编码,例如 GSM 电话听筒的 IMEI 编码(如果移动装置是发送装置的话就使用它自己的唯一编码,如果服务器是发送装置的话就使用有意图接收装置的唯一编码)

[0146] ●共享的秘密信息,和

[0147] ●可以由发送和接收装置二者独立计算的其他涉及(但是不必需要唯一)每个消息(即,增加的消息号,应用/端口号,和会话号)的数据

[0148] 这个密钥就在对称加密算法中用于加密消息。这样每个消息使用密钥序列加密,该序列在算法上涉及各自的移动装置识别码,安装在移动装置和服务器上的共享秘密,以及其他可以由发送和接收装置独立获取的在算法上涉及每个消息的数据。

[0149] 为确保所加密消息的正确性和完整性,发送装置就会使用用密码写的哈希函数计算消息身份验证码('MAC'),其中输入为消息本身和用来加密消息的密钥。作为结果的 MAC 值就被附加到加密消息中。

[0150] 接收装置给相关消息计算第一哈希函数(密钥)(基于移动装置唯一编码号,共享秘密,和其他业务数据的确认),并使用这个密钥来解密该消息。最终,接收装置获取解密的消息和该密钥并使用这些来计算第二哈希值用于与附加到信息上的 MAC 值比较。如果第二哈希值与和消息一起接收的 MAC 值相同的话,那么就会假设消息可信且没有发生改变。另一方面,如果由接收装置计算的第二哈希值不符合与消息一起接收的 MAC 值的话,那么接收装置就努力向发送方发出询问来重新创建安全通信。一旦安全性已经建立,就会触发消息的重传。这样验证系统作为备份的角色来确保消息的数据完整性—传输中任何细小的错误都会导致 MAC 值的失败,安全性询问和消息的重传。

[0151] 在保证了消息本身的机密性,正确性和完整性之外,如果第三方试图假扮为合法的移动装置用户,安全性系统也用来降低成本和对移动装置用户的性能影响。

[0152] 小型服务器保持具有分配给该移动装置的最新报告的地址(和临时端口号)的反向查询表。(查看在 3.11 部分中网络更新的描述)小型服务器假设所有来自移动装置的范围内的分组应该是来自与在该小型服务器反向查询表中所列的装置的当前合法的地址和

端口号相符合的地址和端口号,该小型服务器基于此假设而运行。

[0153] 如果接收到声称来自同一移动装置,但是具有新的返回地址(和/或新的临时端口号)的数据,小型服务器就使用上面所列出的相同加密机制发布安全性询问到位于新地址的该装置上。如果新的地址对该询问返回正确的答案,那么小型服务器就继续处理来自新地址的范围内的业务,并使用新地址更新反向查询表。常规来说这只有在移动装置以该移动装置不了解更改的方式而被分配了新地址或临时端口号的时候才会发生(例如,如果网络运营商 NAT box 做了这样的分配的话),因为通知到移动装置的更改已经触发了网络更新消息。(查看 3.11 部分中网络更新的描述。)

[0154] 另一方面,如果新地址不能够正确地响应安全性询问,那么小型服务器就不更新反向查询表而只是忽略从新地址接收到的数据;(这可能会在,例如,恶意的第三方试图通过发送欺瞒数据业务到小型服务器而中断到合法移动装置的通信的时候发生)与合法装置的通信(在旧地址上与旧的临时端口号)保持不中断并且不需要使用到旧地址的其他询问来重新创建安全性。没有不被希望的数据业务从小型服务器上产生发送到小型客户端;这很重要,因为,使用很多 GPRS 和 UMTS 价目表,用户的花费依赖于所接收数据业务的量,所以能够在小型服务器上禁止拒绝服务进攻就很有价值。

[0155] 由于安全性询问和应答使用相对大量的处理时间和数据传递,这个系统对合法用户将具有巨大的花费和性能优势。

[0156] 3.13 终端应用锁

[0157] 移动通信终端(例如使用 GPRS 的“智能”电话)对作为其主要通信服务器(例如雇主操作的商业邮件服务器)的移动用户和组织呈现出一定的安全性风险。装置的丢失可以导致到通信应用(例如邮件客户端应用)的未授权的访问,就有可能对提供后台服务器容量的最终用户和组织二者都产生负面的结果。此时,安全性风险由各种系统解决(1)给到移动装置本身的访问加锁—通常是在最终用户的请求之下,(2)依赖于移动运营商来拒绝与该装置通信,或(3)拒绝到主通信服务器的远程访问。第一种方法对最终用户来说不方便,因为给装置加锁可能让用户不能访问位于该装置上的其他应用。第二种方法依赖于移动运营商分块指示的经常变化并且适合的实现方法。在与移动装置无关地进行无线网络验证的情况(如使用 SIM 卡的网络)下,该方法进一步受限—拥有该装置的人仍可以访问本地驻留的数据。第三种方法有缺陷,因为它只是阻止了到组织服务器的访问,并不能阻止到本地驻留在移动装置上的数据的访问。

[0158] Transcend Mail 提供了一套系统在预先描述的环境下“锁住”整个通信应用的操作以保护能够对相应的通信服务器做出反应的最终用户和组织,而不需通常从移动装置发送业务的无线网络运营商的干涉。

[0159] 在移动装置上到相关通信应用的访问可以仅在最终用户已经在移动装置上输入了合适的加锁编码之后进行。系统可以指定最小编码长度,但是另外允许最终用户改变该加锁编码。组织管理员(也管理着 Transcend Mail 服务器)也可以在远程基础上更改加锁密码,从而如果最终用户忘记编码或者该装置丢失或被偷了的话可以实现编码的重新设定。这就保护了位于该装置上的电子邮件,也保护了邮件服务器,但是是在组织的控制之下,而不是最终用户或网络运营商—有严格的区分。

[0160] 在加锁状态下,应用锁可以通过将合适的加锁编码输入到移动装置上而不被激

活。

[0161] 如果加锁编码本地存储在移动装置上,就可以基于以下输入使用加密的哈希值来存储:

[0162] ●用于移动装置的唯一装置 ID(例如 GSM 手机上的 IMEI 码)

[0163] ●密钥

[0164] ●解锁码

[0165] 这样,锁就不会仅仅通过从一个装置上获取哈希计算的编码值并在另外一台装置上替换存储的哈希值而被轻易的破解。

[0166] 在解锁之后,应用锁就由一些的不同事件触发,如:

[0167] ●经过了一段时间没有访问该应用(预先设定的空闲时间)

[0168] ●由系统管理员所做的加锁码的远程更改

[0169] ●最终用户请求应用加锁

[0170] ●移动装置或有关应用由于任何原因重新启动或者重新开始

[0171] ●引导应用进入加锁状态的远程消息

[0172] 如果应用处于加锁状态,最终用户就不能访问本地数据并且也不能够访问远程服务器。

[0173] 附录 I

[0174] 附录 1 列举并讨论了与 MobileMQ 和 Transcend Mail 系统关联的各种其他观点。Vega 是 Transcend Mail 的早期名称。

[0175] 1. 来自不同类别的消息的独立传送

[0176] 描述

[0177] 对不同类型的消息的不同传输‘信道’的使用,以确保某个消息类型以更加及时的方式发送,例如:

[0178] ●新的消息信道

[0179] ●其他主体文本信道

[0180] ●附件信道

[0181] 这些信道的使用意味着新的邮件可以部分传送,无论附件和主体文本是否被重新获取。

[0182] 分析

[0183] 这个概念已经用于分离属于不同应用的数据的传送—例如,邮件具有区别于由联系或日程表所使用的信道,并且是通用的复用技术。

[0184] 然而,在现有技术中,使用这种技术来加速单个应用内的通信并不是显而易见的。

[0185] 使用这种技术的优势在于发送或接收大的邮件可能不会阻止与大型消息的后台传送同时发生的与较短消息的更加交互式的经历。超过一定大小的消息可以通过不同的消息信道传送,并且会与用于较短消息的单独信道公平的竞争带宽。这样在发送大邮件之后开始发送的较小邮件可以在大邮件之前完成传送。对于特别大的邮件,这可能意味着对小邮件的响应在大邮件完成传送之前就被接收到。这在很多并发邮件系统中是不可能的,所有的这些系统中邮件无论大小都要排队。

[0186] 一些邮件系统可以与邮件中继(mail relay)并发连接—给每一个邮件建立一

个—但是还是没有基于大小给邮件‘分类’。

[0187] 2. ‘魔力 (magic)’ 邮件地址

[0188] 描述

[0189] 将由服务器转换成确定的电子邮件地址的识别符字符串的使用。现在这具有两个方面：

[0190] ●在‘回复’或‘转发’邮件中的字符串 <a11> 由 Vega 服务器解释为要去到原始邮件并对新消息的‘Cc’域增加消息所有的接收者。

[0191] ●将‘好友名称’增加到终端上的邮件中。当邮件被发送的时候，没有确认为有效邮件地址的所有文本字符串就与交换器中名单的名称比较，如果找到了相符的，该名称就被翻译成那个用户的邮件地址并发送过去。

[0192] 分析

[0193] 使用固定字符串代替邮件地址使其具有依赖于以前邮件内容的含义，这还是可能较新的东西。

[0194] 这样的字符串的使用有效的组成了直接发往服务器来完成新邮件寻址的‘命令’，基于一些服务器所知道的以前邮件的内容。在这种情况下，该命令是‘回复所有’，该‘内容’是在以前的邮件中的邮件地址列表，而‘以前的邮件’是客户端正在做出反应的那些邮件。地址列表或邮件本身都不需要因为这个概念而保留在客户端上工作。所需要的只是对以前邮件的引用（如，邮件头），以使客户端对其具有某类句柄。

[0195] 允许使用短的（或‘友好的’）名称代替在传递给接收方之前由客户端或服务解析的邮件地址是常用的惯例。然而，在无线，分布式客户端的领域内这还是新的概念。

[0196] 3. 启用 / 禁用

[0197] 描述

[0198] 不影响远程另外一端的列队而禁用本地列队的的能力。

[0199] 分析

[0200] 这个功能的目的是允许同级在一个或两个方向上暂时挂起对事件同步的预定。这可能应用于一个，更多或所有的信道。

[0201] 这个功能的例子包括：

[0202] ●用户挂起联系同步，同时执行可能另外产生过度业务量而无法保持远程端同步的本地维护（例如，删除不需要的条目；在本地，与 PC 电缆同步）

[0203] ●在每个用户的基础上管理员挂起服务，以控制成本或阻止滥用

[0204] 4. 预定的队列加法

[0205] 描述

[0206] 执行将导致条目列队用于传输的的动作的能力，不需要立即发送该条目。例如，当 Vega 禁用时创建邮件。该软件不会试图发送邮件直至 Vega 重新启用。

[0207] 分析

[0208] 这个功能在邮件客户端中不是新的了，无论是有线或无线。然而，当应用于无线，分布式客户端领域内时这还是新的功能。

[0209] 5. 自动 / 手动模式操作

[0210] 描述

[0211] 在自动和手动连接模式之间切换的能力。

[0212] 分析

[0213] 表面上,这个功能在邮件客户端中不是新的了,无论是有线或无线。然而,还存在这个特征的确保调查的方面。

[0214] 概念上与Outlook Express所提供的类似,用户可以以固定的时间间隔检查邮件,或者只是对用户界面上按下一个按钮作出响应。

[0215] 然而,这样的系统是轮询系统,其中在定时器超时或由用户界面激活时执行明确的轮询。在Vega中该功能应用于一种推动解决方案,其中定时选项或手动激活指的是‘打开门’了一段允许条目推入的预定时间周期。这个概念在无线,分布式客户端领域内还是新的。

[0216] 6. 在线附件置换

[0217] 描述

[0218] 使用声明真正的附件不能被重新获取的原因的管理消息代替附件的‘占位符’的能力。

[0219] 分析

[0220] 这个功能在邮件服务器中不是新的一例如,一些服务器侧的病毒扫描器会在邮件字符串的主体文本内部呈现管理消息,该邮件声明由于病毒感染没有发送附件的原因。

[0221] 然而,还有很多原因导致附件没有发送到无线,分布式客户端。

[0222] 可以选出一些原因:

[0223] ●大小超过了管理员设定的大小

[0224] ●用户不打算下载附件(例如由于成本控制的原因)

[0225] 一些可能是因情而定:

[0226] ●附件不与终端兼容,并且不能由服务器转换

[0227] ●附件感染了病毒

[0228] ●附件在以前已经被删除

[0229] 7. 远程附件表示

[0230] 描述

[0231] 显示附件表示,以示出附件没有物理性地附加到邮件上,但是将在最终传送之前附加。也在发送文件夹中,附件被发送的表示,不需要保持该附件的本地拷贝。

[0232] 分析

[0233] 表面上,这个功能在邮件客户端中不是新的了,无论是有线或无线。

[0234] 对一些邮件客户端的特征来说显示纸夹子或其他图标来表示具有附件的方法都是类似的。在IMAP4客户端中,附件可以或不可以实际出现。

[0235] 然而,这个特征在无线,分布式客户端领域内是新的—尤其如果与指示组合在一起的话,通过不同的图标或通过本地包含的说明性主体文本来表明附件存在或者其他方面。

[0236] 8. 邮件接收方列表切断

[0237] 描述

[0238] 邮件消息的其他接收方的数量的显示,不需要给每个接收方发送解析的邮件地

址。

[0239] 分析

[0240] 这个功能在无线,分布式客户端领域内是新的。

[0241] 其目的在于当展开的列表中的用户数目超过了一定数时(例如,10),邮件的接收方可能不关心列表中实际的个别用户。这样,数据量就通过不在邮件头中包含其他接收方的姓名和邮件地址而得到减小。用一个更加简单的文本条(如,“38个其他接收方”)来代替。接收方的整个列表可以应要求而下载。

[0242] 特定的文本不会影响用户‘回复所有’,由于真正的邮件中服务器还是知道真实的地址。

[0243] 9. 调用安全性时的内容隐藏

[0244] 描述

[0245] 通过用户使用可选的视图隐藏屏幕上用户数据的功能。这是用来当终端进入‘密码询问’或‘应用锁住’状态时隐藏所有用户数据的。

[0246] 分析

[0247] 这个功能在无线,分布式客户端领域内是新的。

[0248] 这个功能的目的是隐藏特权信息直至获得批准的人完成了安全性询问。其目的在于防止屏幕上信息的被窃。

[0249] 10. 智能用户创建列表

[0250] 描述

[0251] 给管理员提供适合 Vega 帐户创建的用户列表的功能。实际上这意味着去到现行目录,根据用户所在的服务器存储用户,无论该用户的 Vega 帐户是否早已存在。合成的列表(在‘新用户’UI中显示)仅显示可以具有 Vega 帐户的用户。

[0252] 分析

[0253] 使用适合于用户中的每个或所有的特征而将用户列表分类的能力不是新的了。然而,在无线领域内这个功能的应用还是新的,尤其在该列表代指没有与分布式客户端一起配置的用户的情况下。

[0254] 11. 定制的事件日志

[0255] 描述

[0256] 来自1个或更多远程服务器的日志信息的显示,已过滤以便只有与产品相关的事件才被显示。

[0257] 分析

[0258] 远程显示远程服务器的已过滤日志信息的能力不是新的了。然而,这种情况下的‘服务器’是在大型客户端/服务器系统中包含客户端的分布式系统的组件。远程显示不是该客户端/服务器系统的部分,也不是分布式客户端的部分。这是新的技术。

[0259] 12. 文件级的 OEM 定制

[0260] 描述

[0261] 不必须重新编译任何核心二进制文件而给新的 OEM 合作者定制产品的能力。“OEM”文件的代替品是被要求定制产品的文件。

[0262] 分析

[0263] 定制软件的能力—在预先定义的范围内—而不需重新建立二进制文件且不需访问源代码,这不是新的概念。

[0264] 然而,应用于无线,分布式客户端 / 服务器系统的一个或更多组件的这项技术还是新的。

[0265] 13. ‘智能’ 列队

[0266] 描述

[0267] 根据使用用于分析队列内容的 ‘智能’ 算法增加,记录,替换和 / 或从未完成的发送队列中移除条目的能力。

[0268] 分析

[0269] 为了降低所需的数据传送量而对发送队列的内容合并不是新的概念了。例如,如果队列包含后面有立即删除的新联系,简单的算法就可以检测到这种情况并将新联系和其删除选项一起移除,因为二者有效地互相抵消。

[0270] 然而,应用于无线,分布式客户端 / 服务器系统的一个或更多组件的这项技术还是新的,尤其是由于其目的在于减少通过计量的网络传送的数据量。

[0271] 14. 在发送之前重新获取删除

[0272] 描述

[0273] 在数据传送开始之前取消重新获取请求的能力。

[0274] 分析

[0275] 这与从发信箱中移除条目的能力相类似,并且可以应用于邮件,联系和议程条目。当然这不是新的概念了,因为很多邮件和组件客户端都提供这种功能。

[0276] 然而,应用于无线,分布式客户端 / 服务器系统的一个或更多组件的这项技术还是新的—尤其是由于在 ‘发信箱’ 中的条目的目的地不是邮件或组件服务器,但另外一个组件是分布式客户端。只有在发送到远程组件分布式客户端之后,条目才真正的进入到一些其他目的地的 ‘发信箱’ 未完成发送 (例如在因特网上)。这是可以由这个功能取消的初始传送。

[0277] 15. 轮询和推动 (push) 用户

[0278] 描述

[0279] 在单个的组织或装置内创建 ‘轮询’ 和 ‘推动’ 操作用户帐户的能力。

[0280] 分析

[0281] 这项功能允许一些用户在真正的推动模式下操作,其他用户在严格的轮询模式下操作,而其他用户根据轮询的频率在 ‘有效推动’ 的模式下操作。

[0282] 这个概念在普通的邮件客户端不是新的了—用户对邮件传送的偏好的选择通常不对用户所做的选择有任何影响。

[0283] 然而,

[0284] ● 这项功能应用于尤其是在无线领域内运行的 Vega。这种系统通常如适用于该系统的设计哲学所指示,是 “全部推动” 或 “全部轮询”。

[0285] • 术语 “推动” 和 “轮询” 应用于分布式客户端内部的通信—而不是在大型客户端 / 服务器模型内部,在这种系统中总是 ‘推动’。

[0286] 从而,这是全新的概念。

[0287] 16. 预定 (provisioning) 逻辑

[0288] 描述

[0289] 基于关于用户所选帐户和连接手机的信息而修改, 否决或拒绝手机预定处理的能力。

[0290] 分析

[0291] 这项发明的目的在于基于由现存的帐户状态和所连接手机的状态直接给用户提供打开给他看的可能选项的列表。

[0292] 这个概念可能不是新的了, 然而, 这项发明的应用在于分布式客户端模型的两个组件和它们之间链路的锻造和中断:

[0293] ●如果所选择的用户帐户被允许使用 Vega 但是不具有分配的手机

[0294] ○如果该手机正分配给另外一个用户

[0295] ■没有可能的配置: 管理员必须移除该分配

[0296] ○如果当时没有提供手机

[0297] ■安装该软件并执行基本的同步

[0298] ●如果所选择的用户帐户被允许使用 Vega 并且具有分配的手机

[0299] ○如果该手机当前分配给该用户

[0300] ■如果必要的话修复手机软件安装

[0301] ■如果希望的话保持手机不被修改

[0302] ○如果该手机当前分配给另外一个用户

[0303] ■没有可能的配置: 管理员必须移除该分配

[0304] ○如果当前没有提供该手机

[0305] ■修改该分配; 该用户的当前手机被擦除

[0306] ■安装了该软件并执行基本的同步

[0307] ●如果所选择的用户帐户不被允许使用 Vega

[0308] ○该用户的进一步操作不被允许

[0309] 17. 擦除手机

[0310] 描述

[0311] 发送将会导致所有数据从用户装备上移除的命令给该用户装备的能力。

[0312] 分析

[0313] 这项命令的目的是鼓励装备上的所有数据的远程删除。管理员在该手机丢失或者被偷的情况下会更愿意采取这类行为。用户很少会这么做以防止其装备变成毫无功能的。

[0314] 这项特征是新的, 尤其在无线, 分布式客户端领域。

[0315] 18. 数据传送时间戳

[0316] 描述

[0317] 上一次数据被发送或从用户终端接收的时间戳的存储和显示。

[0318] 分析

[0319] 这项特征的目的在于记录上一次从分布式客户端的远程组件接收的数据传送时间和日期。这给用户对系统在任何应用数据不具备的条件下工作提供了方便的检查方式。

[0320] 这项特征的大的单元对能胜任的人来说相当明显。然而, 由于底层传输的不可靠

本质,不可能决定上一次从终端到服务器的传送的确切时间,然而,由于任何方向上的所有通信都导致在另一方的多种确认中的至少一个,记录最后引入的数据就提供了至少一些流出的数据被接收到的证据。

[0321] 19. 软件安装

[0322] 描述

[0323] 无需使用软件安装的标准(SIS文件)方法而从Symbian OS产品安装(和随后卸载)软件的能力。

[0324] 分析

[0325] Symbian SIS文件的目的在于通过InstallShield的类似产品和其它产品以与其他软件环境中频繁使用的大概相同的方式简化安装过程。

[0326] 然而,由Symbian OS装置所使用的连接性平台不再允许SIS文件的自动执行,从而对管理员来说不可能鼓励软件不需要与该装置的物理性交互而进行软件的安装(即,按下按钮,与收件箱交互等)。

[0327] 这项特征的本质是为了避免安装系统一起通过将文件拷贝进某个文件夹,以便在重启该装置时,某些完成安装和对该装置配置的‘运行一次’的组件可以执行。这允许了可能还会要求复杂的SIS文件重新编译的对每个装置的安装。

[0328] 安装者所要求行为的唯一过程是在将装置交付给用户之前关闭该装置。这确保了安装流程的完成。

[0329] 20. 域内的安全性更新

[0330] 描述

[0331] 在域内管理员在用户终端上更新终端安全性方针的能力(例如,提高数字的最小数或应用锁的超时时间间隔)。

[0332] 分析

[0333] 本发明的目的在于允许管理员远程更改装置的某些安全性保证而无需装置预先提供或返还到管理员来执行该更改。

[0334] 可以包含这样一些更改:

[0335] ●要求用户选择的应用锁编码的最小强度的策略的变化—这可能对一些用户会强迫其更改编码

[0336] ●在由装置所使用的共享私密信息中的预定的更改

[0337] ●用户更改其登录密码的要求

[0338] 这种特征在无线领域内是新的,尤其在重新配置的目的地是分布式客户端组件的情况下。

[0339] 21. 在域内轮询/推动更新

[0340] 描述

[0341] 管理员远程更改微轮询频率的能力,或者给个别终端或用户从轮询到推动(反过来也是这样)改变连接性特征的能力。

[0342] 分析

[0343] 管理员可以对某些用户要求改变轮询的频率以响应:

[0344] ●关于某些用户所允许的某种类型的支出的方针更改

[0345] ●对特定用户改变音量习惯

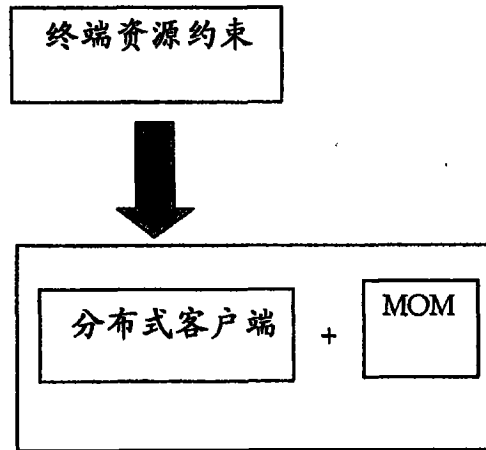
[0346] 这项发明的特征允许管理员在每个用户的基础上更改任何或所有的以下方面：

[0347] ●更改微轮询的频率以在缺乏任何应用数据的情况下改变后台数据量

[0348] ●更改微轮询的频率以通过参考任何网络运营商对 NAT 映射超时来使‘有效推动’可行或不可行

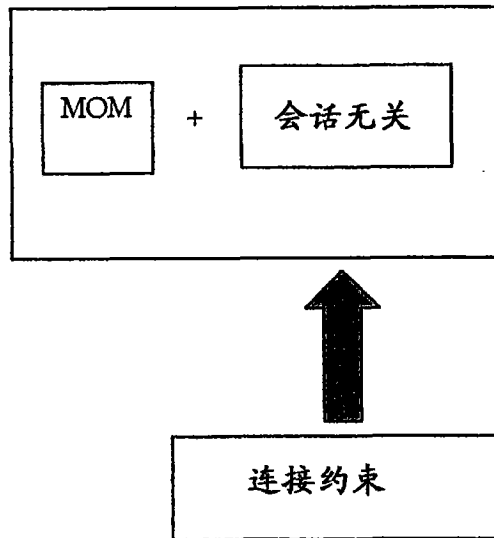
[0349] ●改变成或构成真实的‘推动’模式，在这种情况下没有使用微轮询

[0350] 这个特征对于无线领域内分布式客户端的组件的内部通信来说是特别的。在一种实现中，用户不具备对以上的控制；管理员是这些设置的唯一特权者。



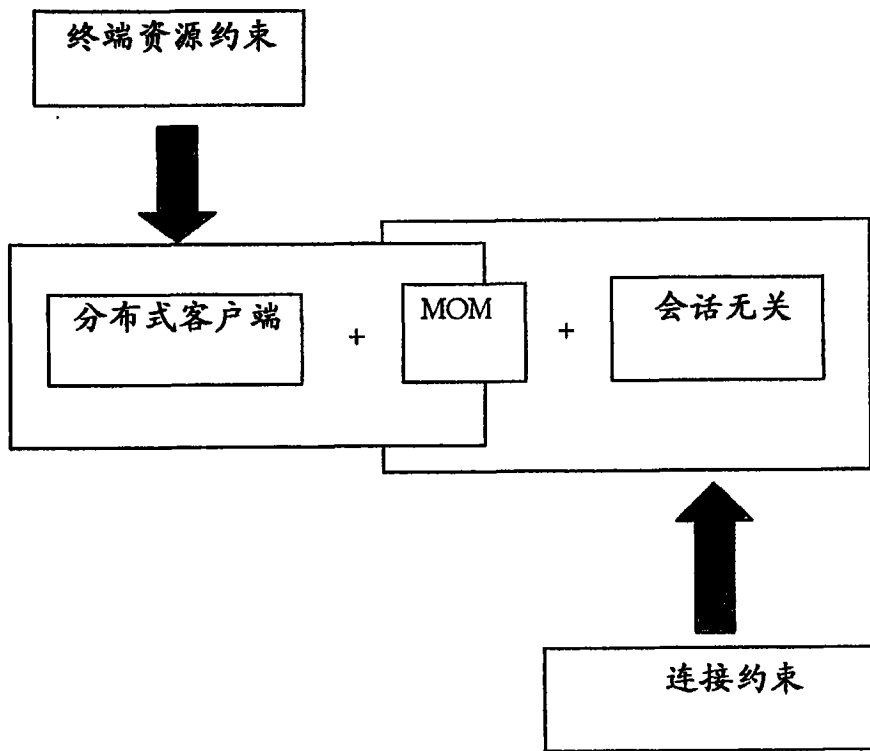
终端资源约束本质上是通过在诸如MOM的消息队列系统上协作的分布式客户端的组合来满足的

图 1



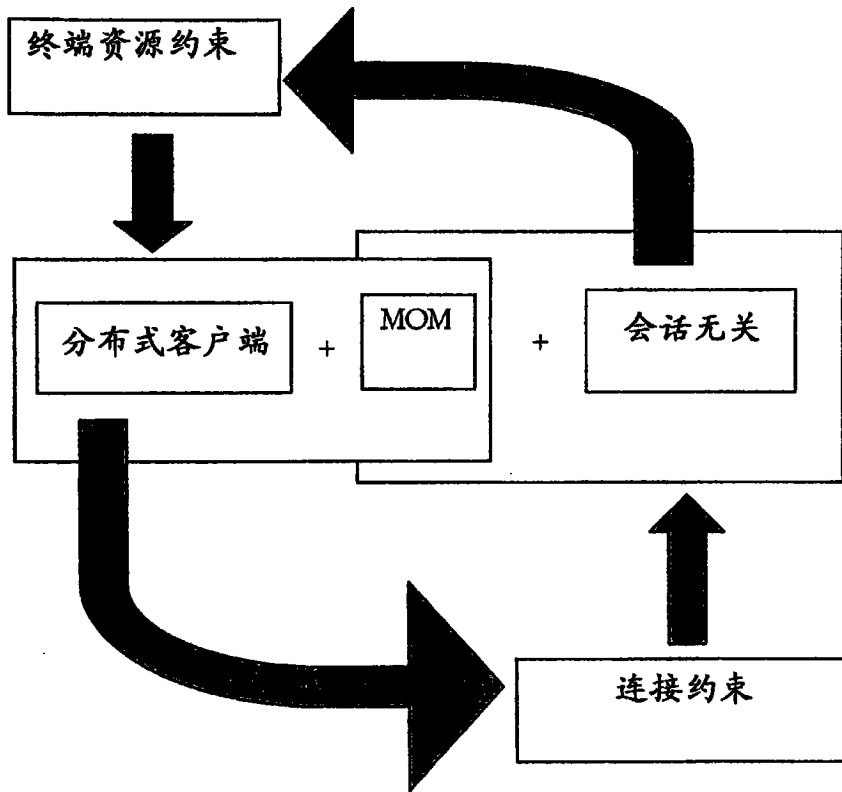
连接约束本质上是通过对以‘会话无关’的方式运行的平台所使用的、诸如MOM的消息队列系统的组合来满足的

图 2



transcend Mail和MbileMQ系统不但致力于终端资源约束，而且致力于连接约束

图 3



通过分布式客户端方案而产生的基于事件的数据复制也致力于连接约束，并且本质上就是会话无关的

强加会话无关使得通常以会话相关方式传送而不一定适合于资源约束装置的功能以现在适合意图的方式部署。这些功能包括消息传递的可靠性、发送方认证、消息安全性、数据率流量控制和分组路由

图 4

‘分布式客户端’模型

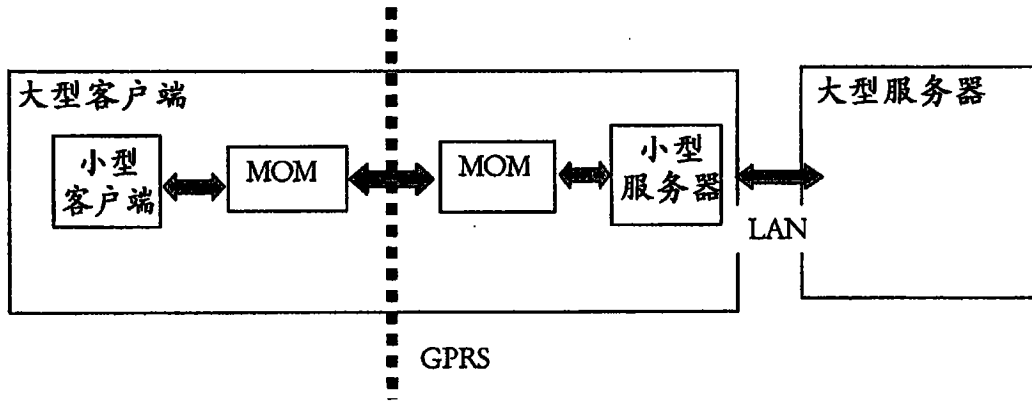


图 5

小型客户端可以含终端侧组件，加上客户端侧MOM；小型服务器也可包含服务器侧组件，加上服务器侧MOM：

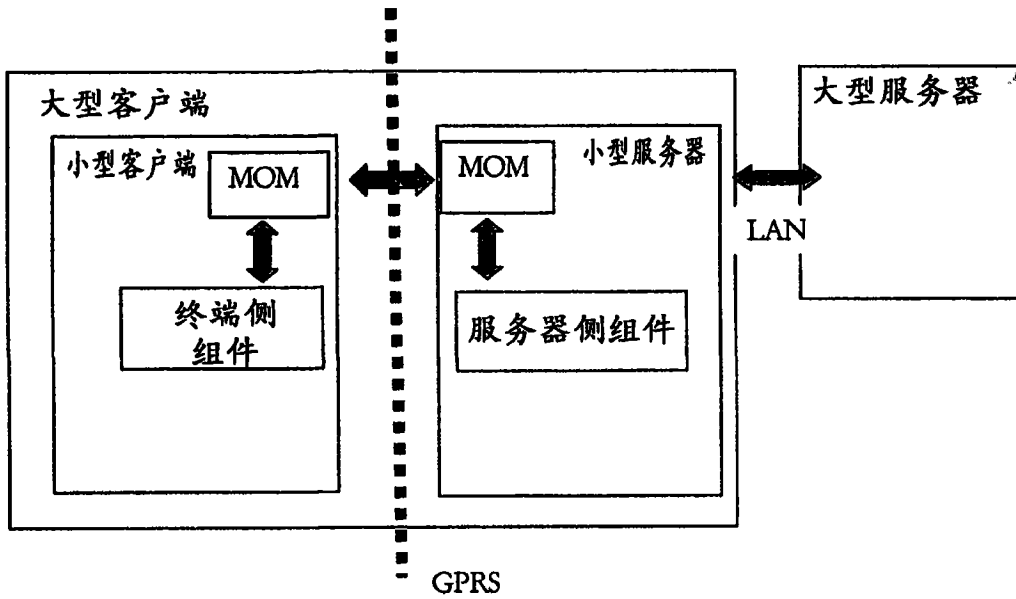


图 6

小型客户端可包含程序—例如电子邮件应用，加上将其连接到终端侧组件的插件

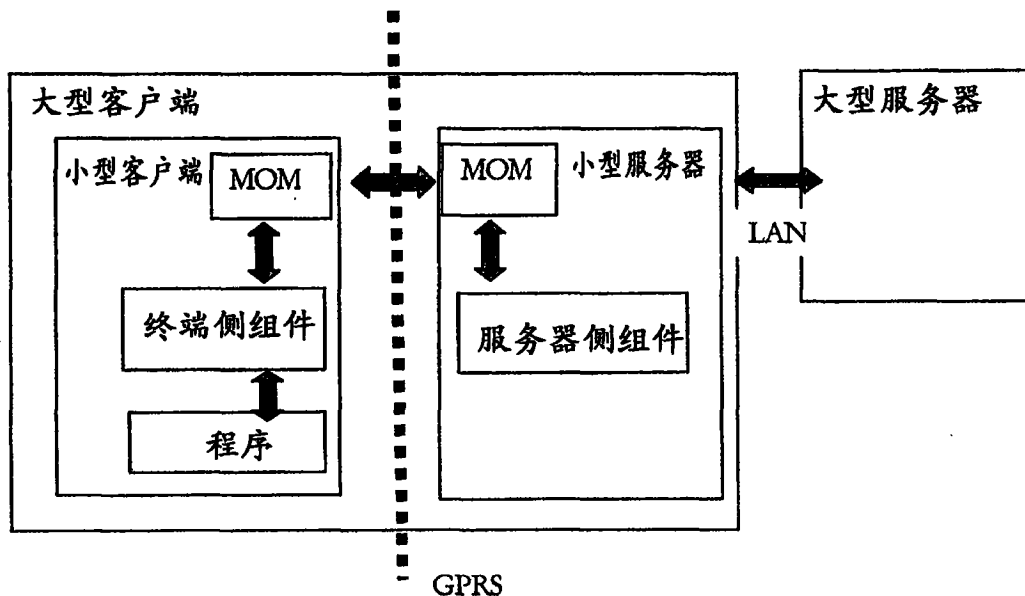


图 7

小型客户端也可不包含程序，例如联系程序。终端侧组件然后通过联系数据库与联系程序通信（使用来自将要被发送到终端侧组件的那个数据库和事件触发器）：

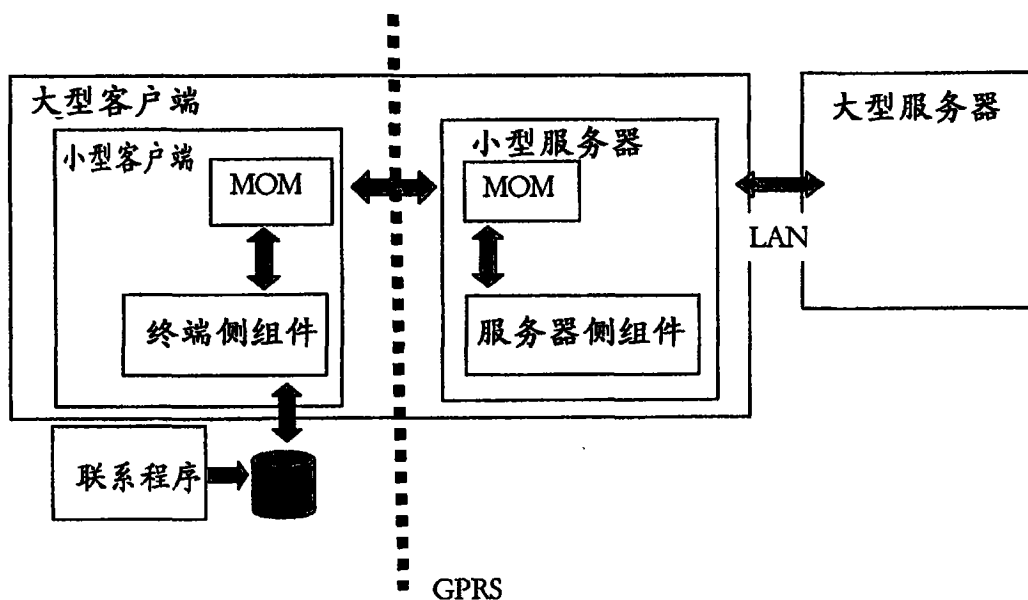


图 8

这在概念上等同于以下中间件体系结构：

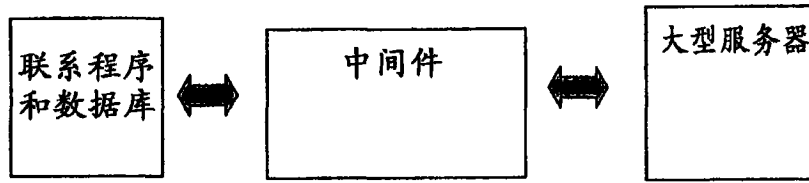


图 9