

移动代理技术的讨论

北京理工大学计算机科学与工程系 盛成 徐进

本文介绍移动代理技术的必然性、要点及应用前景。针对关键难点，提出相应的解决策略。并以IBM AGLETS系统为例，阐述移动代理的技术实现。

移动代理技术的必然性

个性化发展必然性: 随着Internet网上信息量及用户数的急剧增长,需求差异性也越来越大,多样性的用户群不再满足于统一的信息界面形式。他们要求更个性化的访问方式。这种个性化需求带来得却是从简便服务形式到复杂服务实现的技术问题。个性化的需求所带来的复杂服务,要求直接同服务源进行交互的操作越来越多,因此用户的服务软件必须出现在信息的源头站点上,即网络站点上的代理。

网络技术发展的必然性: 另一方面,网络技术的发展趋势是主干网速度越来越快,而连接的分支网速度却没有明显的改善。造成主干网与众多用户连接网络之间的带宽差距越来越大。这样大量用户的主机为避免网络拥塞而构想的种种方案便出现了。

首选解决方案: 移动代理正是上述必然发展应用的解决框架,它将发出请求的客户端代码动态地移动到服务器端执行,无需经过网络传输这一中间环节,而直接同服务源进行交互,显著提高了效率,降低了对分支网络带宽的要求。并且由于网站主机直接与主干网相连,大量信息源获取及筛选的任务集中于网站主机,而经过筛选的最终用户信息相对只是极少量的,这样便进一步缓解由于缓慢分支网速度带来的传递压力。

用户的角度: 从用户的角度看,网络传递速度的提高

以及简单的委托,可以减少时间和金钱的浪费,无疑地是一幸事。

必要的条件: 当然,这一方案的必要条件是快速增长的主干网带宽和高性能的服务器通信技术,因为网站代理要为用户收集、整理、缓存及筛选信息,由此带来的大量开销是肯定的。

移动代理技术要点

移动代理的发展: 移动代理(Mobile Agent)技术受相关的网络技术影响,很长时间只停留在实验室里。随着Internet技术的迅猛发展,这项技术开始逐渐显现出它的应用价值和发展潜力。移动代理技术的研究起源于人工智能领域,它是指模拟人类行为和关系,具有一定智能并能够自主运行和提供相应服务的程序。

主要特征: 移动代理可在异构网络的主机间移动;可受理委托;并具有自我决定权。所谓自我决定权即是可决定自身的行动去向;可决定自己的服务对象;甚至有自身复制的权利,并将复制的代理同样地发送到网上。其技术特征可归纳为:动态执行,异步计算,并行处理,智能化。

智能化和知识库: 移动代理的重要特性是搜索与过滤,其处理依据于智能化技术和知识库系统,待搜索信息的标准与路线作为基本的知识体系。信息标准是指关键字表和选取规则,路线则是指代理在网上所需的移动策略。移动代理在用户授权下,即给定相应的知识,它以用户代表的身份被散发到网上。通过对大量网上信息的智能化搜索、过滤,在规定时间内返回必要的信息结果。

普遍、开放、综合性构架: 实际上移动代理技术并没

有带来新一代的软件,也不比传统的技术具有更好的性能,它真正意义在于:用单一的、普通的结构体系,解决编程设计平分的问题,将系统平分给信息提供者、中介者及客户端。换句话说,移动代理技术给设计者们更多的时间及灵活度,为用户作出更实用的软件。正如一份研究报告中阐述的:虽然移动代理技术没有单独的、足够强的优势,但它所集中的优势却是无人能及的。技术的重要点不在于本身的功能,因为它的任一部分都可被传统的分布式处理所替代。而技术重要点是潜在的大量应用。它为网络服务的发展与个性化环境提供了一个普遍的、开放的、综合的构架。

动态规划:移动代理应该是根据任务的目标、网络负载以及服务器负载等外界环境,动态地规划进行转移操作的。也就是必须对网络资源进行优化,解决负载均衡,并避免盲目的资源访问。

面临的技术难题

Internet的发展将会导致移动代理技术的广泛应用,但是目前移动代理技术还存在着若干急待解决的技术难题。

性能问题:虽然多代理系统能够弥补网络带宽不足的缺陷,但同时带来加重机器负担的负面影响。主要原因在于,代理系统出于对安全性及简便灵活的考虑,通常由运行速度相对慢的解释性语言编写制成,并且必须被嵌入到适当的执行环境中。因此,在排除网络断线的前提下,移动代理要比传统的程序付出更多的代价,包括时间和效率。人们不得不提出:缓慢的代理程序以及移动处理所消耗的时间能比网络阻塞的时间快多少?

幸运的是,类似JAVA的编程语言的编译速度正在加快,与软件相关的其他技术也正在发展之中,移动代理要以接近于本地执行代码的速度运行才能适应需求。

此外,减少移动负担的技术也正在积极探索之中。例如,如果服务器能将移动代理的功能函数写入操作系统的内核中(如WINDOWS的动态连接库中),移动代理的运行及移动的效率将会大大提高。

标准化问题:几乎所有的移动代理系统都希望程序代码在异种机器间自由移动。例如,代码被编译成象JAVA的字节码一样,与平台无关,然后再根据其到达的平台被编译成可执行码,或者在解释器中执行。因为移动代理的性质决定了它的代码可移植(portable)性。然而,要求在所有的计算机上安装同一种移动代理接受系统几乎是不可能的。要使代码在不同系统环境下可真正灵活移植

需要有标准化工作的努力。OMG MASIF的标准已迈出了第一步。下一步的工作即是在特殊的执行环境中进行类似虚拟机的标准化解决方案,其中必须包含对移动代理代码及状态的标准化定义。

安全性问题:一旦给予移动代理访问本地资源的权限,如何避免可能的恶意系统破坏?可以利用类似JAVA APPLES技术,借助其提供的安全机制解决。然而,安全问题是相互的,保护进驻代理不受主机上的恶意攻击者破坏却是不能忽视的。例如:将信用卡号写入授权的移动代理,发送代理到某一网站主机,由于主机将移动代理的源代码及状态上载,该主机上的窥探者将能读到委托人的私人信息,可以修改它们。这种安全隐患是不能容忍的。移动代理所带来的安全问题包括:

① 没设代理访问权限的网络站点,其安全问题如何保证。

② 代理在进驻到授权访问的网络站点后,怎样保护自身的安全。

③ 不在同一管理控制下的机器组,如何保证不受代理的破坏,以及代理不受破坏。

目前,正在研制的诸多网络安全技术,都涉及到以上的问题。但彻底解决以上安全问题还存在很多技术难点。

IBM Aglets 技术的实现

IBM Aglets是一种建立于JAVA技术上的移动代理。在开发这项新的技术中,Aglets有了新的含义:基于JAVA的自治的软件代理,这里的软件代理,是移动代理技术的一种技术实现。通过对比IBM Aglets与JAVA Applets的共性和个性,可以说明这一代理技术的特征。

特性:Aglets扩展了以JAVA Applets著称的网络移动代码模型。区别于JAVA Applets,Aglets的CLASS代码文件可以在网络间移动并同时携带其状态移动。Applet(JAVA小程序)只是一段在网络中由服务器向客户端移动的代码,它不存在状态。而Aglet则是一段运行着的JAVA程序,他带着自身的代码与状态从一台主机移向另一台主机。无论移动到什么位置,它都携带自身的状态。状态的连性决定了Aglet顺序游走在各节点机上的路径,并且最终返回它的出发地。

共性:JAVA Aglet与Applet相似之处在于,它们都以一个或多个线程的形式运行于接受端的JAVA环境中。对于JAVA Applets,WEB浏览器为了运行它,在内部必须启动JAVA应用。并在应用中安装了安全管理

机制,对 Applets 的活动监测潜在着的危险。WEB 浏览器的应用利用加载器类处理服务器端的源代码,即下载 Applets 的源代码。

同样地,Aglets也需要在每台运行该移动代理的主机上有一个 JAVA 应用,称为 Tahiti,当 Aglets 在网间移动时,是从一个装有 Tahiti 的主机移向另一个装有 Tahiti 的主机之间的移动。Tahiti 上载 Aglets,接收希望移动进入的 Aglets 的申请。同时,它也以加载器方式处理上载,从而实现 Agelts 的网间移动。Tahiti 也设有安全管理器,对 Agelts 的活动监测潜在的危险。

框架结构: IBM Aglets 的运行框架结构如图 1 所示。

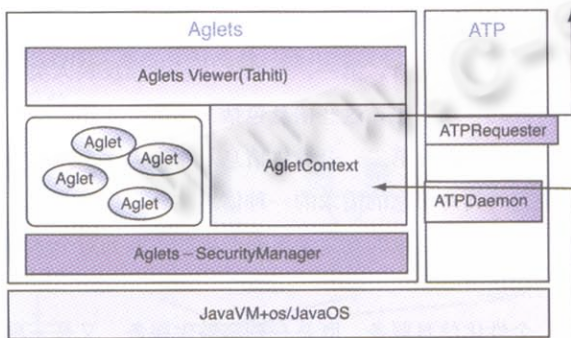


图 1 Aglets 的运行结构框架

框架结构的主要组成包括:

- ① Aglet: 用 JAVA 编写的移动代码
- ② AgletContext: Aglet 的工作台
- ③ ATP: 即 Agent Transfer Protocol, 代理传输协议
- ④ Aglets-Security Manager: Aglets 安全管理器
- ⑤ Java VM+OS: 捆绑有 JAVA 虚拟机的操作系统

生命周期: 系统的生命周期如图 2 所示。其中包括:

- ① Created: 标志着 Aglet 的产生以及状态的初始化, 主线程开始执行;
- ② Cloned: 复制一个 Aglet, 包括复制当前状态;
- ③ Dispatched: 一个 Aglet 被派遣到新的主机, 该 Aglet 的状态也随之发送;
- ④ Retracted: 一个先前被派遣的 Aglet 从远端主机上返回其初始所在的主机;
- ⑤ Deactivated: 一个活动着的 Aglet 进入睡眠状态;
- ⑥ Activated: 一个睡眠的 Aglet 重新被激活;
- ⑦ Disposed of: 一个 Aglet 死亡了, 其状态随之永远消失。

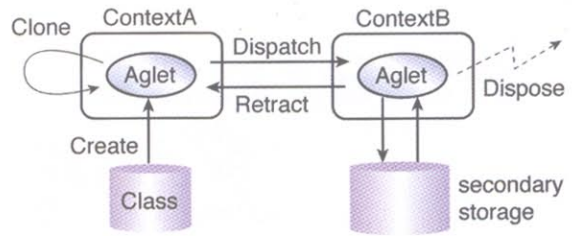


图 2 Aglets 的生命周期

Danny Lange 曾对他的设计这样评价说:“当我发明了 Aglets 时,我正在寻找一种更高级网络计算的抽象。对我来说,Aglets 解决了 15 年来困扰我的一个问题,即在面向对象的编程技术之后,谁是替代者? Aglets 是我对“network computing”的一个关键的提取,而不是 Applet、Servlet 及 RPC。Aglets 允许我以网络计算机的形式来思维,而不是 NC,是所有的 NCs、PCs、workstations 以及大型机。我并不关注 Aglets 在何处运行,Aglet 本身也不关注。就如我们所说的那样“Create once, go anywhere。”移动代理技术为网络环境下解决软件问题提供了办法,而这种技术的实现更接近环境。就象在现实世界中,任何人或任一物体,他(它)们能自由地从 一个地方到另一个地方;去完成任一工作,处理任一事务;有时可打电话解决问题,有时可派助手去,有时也可亲自去。

移动代理的应用前景

由于移动代理的自我智能性质,决定了它的广泛应用前景。其主要应用包括:

信息的搜索与过滤: 由于 Internet 上信息量的迅速增长,从繁杂的信息中找出对你有价值的信息是非常困难的。又由于网络带宽的限制,网络阻塞现象经常发生。用户要花大量的时间和金钱,可能仍然一无所获。利用移动代理在 Internet 上雇用代理,或称特派员、记者等等,代表你访问许多站点,在遍历这些站点的信息后,建立你所需要的信息链接索引。

监视: 网上信息不是广泛散播的,而是随时间动态变化的。移动代理可作为监视者,等待某一特定消息的出现,然后获取。例如,可以发放移动代理去某证券交易所的主机,等某一股票接近某一价格;也可以对指定的新闻内容进行捕获。监视性应用充分体现了移动代理的异步特性,发送代理后不必等待结果回送。

信息分发：可利用移动代理将指定的新闻或商业信息周期性或者一次性发送到指定的用户。

模拟谈判：除了从文件和数据库中搜索信息外，移动代理之间也可以产生信息交流，即所谓模拟谈判。例如，进行聚会的时间商定，可通过发送各自的代理与其他代理交涉，每个代理有各自的用户的时间表，以及作为仲裁者的权衡策略，最终确定时间的登记表。

实物交易：电子商务作为移动代理技术的典型应用。可以帮助购物、定货及付款。例如，移动代理可以被指定访问不同航线的时间表与价格，用以确定最理想的价格与

时间，帮助订票和付款。

并行处理：由于移动代理可以在网络接点间移动并且复制自身产生更多的代理，实际上这种功能可用来管理一项并行处理的工作。将需要多 CPU 系统的计算转为多代理系统的运算。

娱乐：游戏者可用代理来替代自己，赋予他策略，发送它到游戏主机上进行竞争。

可以预言，移动代理技术会以它独特的性能优势，在不远的将来为所有主要的 Internet 网站所接受，并成为网上服务的重要组成部分。■