



## [12] 发明专利申请公开说明书

[21] 申请号 00120241.3

[43] 公开日 2004 年 5 月 19 日

[11] 公开号 CN 1497465A

[22] 申请日 1999.10.27 [21] 申请号 00120241.3  
分案原申请号 99801737.X

[30] 优先权

[32] 1998.10.28 [33] US [31] 60/105917  
[32] 1999.5.17 [33] US [31] 60/134395

[71] 申请人 维迪科隆有限公司  
地址 美国乔治亚州

[72] 发明人 R·布尔森 D·乌尔博格  
G·弗雷斯塔特

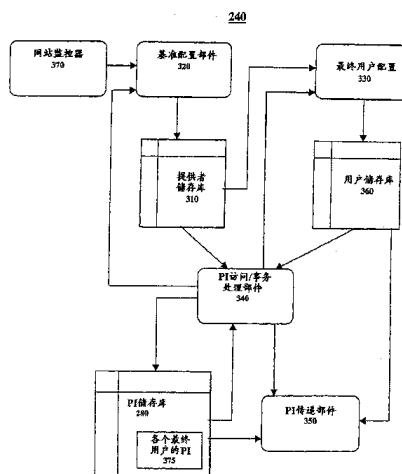
[74] 专利代理机构 中国专利代理(香港)有限公司  
代理人 王 勇 王忠忠

权利要求书 2 页 说明书 25 页 附图 11 页

[54] 发明名称 用于自动访问个人信息的系统与方法

## [57] 摘要

按照本发明的用于传递个人信息的系统，具有包含最终用户数据的用户储存库、一个包含信息提供者数据的提供者储存库、一个包含个人信息的个人信息储存库和一个与这些储存库通信的处理器。处理器选择一个最终用户来进行个人信息聚集。处理器与一个或更多的信息提供者连接。然后，处理器就开始从所连接的信息提供者为选定最终用户检索个人信息。这种检索所根据的是与选定最终用户关联的最终用户数据和与所连接的信息提供者关联的提供者数据。检索出的个人信息被存储到个人信息储存库。



1. 一种自动访问与最终用户关联的个人信息的方法，其中，个人信息在个人信息提供者上存储，该方法包含的步骤有：

5 (a) 将与存储在个人信息提供者上的个人信息对应的一个个人信息的表示和一个链接显示在与最终用户关联的客户计算机上；

(b) 当启动所显示的链接时，向客户机下载一个应用程序，其中，下载的应用程序在客户计算机上执行时执行下述步骤：

- 10 (i) 连接到个人信息提供者；  
(ii) 漫游到个人信息提供者上的个人信息；以及  
(iii) 将个人信息提交给客户计算机的用户。

2. 权利要求 1 的方法，进一步包含的步骤为：

- (a) 将与最终用户关联的最终用户数据传输到客户计算机；以及  
(b) 将与个人信息提供者关联的个人信息提供者数据传输到客户计算机。

15 3. 权利要求 2 的方法，其中，传输与个人信息提供者关联的个人信息提供者数据的步骤包含传输一个与个人信息对应的导航脚本。

20 4. 权利要求 3 的方法，其中，传输与最终用户关联的最终用户数据的步骤包含根据所传输的导航脚本来传输与最终用户关联的最终用户数据。

5. 权利要求 1 的方法，进一步包含的步骤是，根据与个人信息提供者关联的个人信息提供者数据、根据个人信息和根据与最终用户关联的最终用户数据，生成一个向客户计算机下载的一个应用。

25 6. 一种自动访问与最终用户关联的个人信息的方法，其中，个人信息在个人信息提供者上存储，该方法包含的步骤为：

(a) 将与存储在个人信息提供者上的个人信息对应的一个个人信息的表示和一个链接在与最终用户关联的客户计算机上表示出来；

30 (b) 当启动所表示的链接时，传输一个含有表格的页，表格中包括登录信息，当客户计算机打开登录信息时，登录信息将客户计算机重定向到个人信息提供者上的一个后登录页；

7. 一种自动访问与最终用户关联的个人信息的方法，其中，个人信息在个人信息提供者上存储，该方法包含的步骤为：

(a) 将与存储在个人信息提供者上的个人信息对应的一个个人信息的表示和一个链接在与最终用户关联的客户计算机上表示出来；

(b) 当启动所表示的链接时，通过执行下述步骤，将客户计算机驱动到个人信息提供者上存储的个人信息：

- 5 (i) 连接到个人信息提供者；
- (ii) 漫游到个人信息提供者上的个人信息；
- (iii) 将个人信息提交给客户计算机的用户；以及
- (iv) 代理客户计算机随后向个人信息提供者提出的请求。

8. 一种存储指令的计算机可读的存储器，它存储的指令在执行  
10 时使处理器自动地访问与最终用户关联的个人信息，其中，个人信息在个人信息提供者上存储，所执行的步骤有：

(a) 在与最终用户关联的客户计算机上显示与存储在个人信息提供者上的个人信息对应的一个个人信息的表示和一个链接；  
15 (b) 当启动所显示的链接时，向客户机下载一个应用程序，其中，  
下载的应用程序在客户计算机上执行时执行下述步骤：

- (i) 连接到个人信息提供者；
- (ii) 漫游到个人信息提供者上的个人信息；以及
- (iii) 将个人信息显示给客户计算机的用户。

9. 一种用于自动地访问与最终用户关联的个人信息的系统，其  
20 中，个人信息在个人信息提供者上存储，该系统包含：

(a) 一个用于存储与最终用户相关联的数据的用户储存库；  
(b) 一个用于存储与个人信息提供者相关联的数据的个人信息提供者储存库；  
25 (c) 一个与用户储存库及个人信息提供者储存库通信的处理器，  
处理器执行下列步骤：

(i) 在与最终用户关联的客户计算机上显示与存储在个人信息提供者上的个人信息对应的一个个人信息的表示和一个链接；

(ii) 当启动所显示的链接时，向客户机下载一个应用程序，其中，  
下载的应用程序在客户计算机上执行时执行下述步骤：

30 (A) 连接到个人信息提供者；  
(B) 漫游到个人信息提供者上的个人信息；以及  
(C) 将个人信息显示给客户计算机的用户。

## 用于自动访问个人信息的系统与方法

5 本申请是 CN99801737.X 的分案申请，原案申请日为 1999 年 10 月 27 日，其发明名称为“用于自动聚集和投递电子个人信息或数据以及涉及电子个人信息或数据的事务处理的装置与方法”。

10 本申请依照 35U.S.C. § 119(e)，要求申请人的临时美国专利申请系列号 60/105,917（提交日：1998 年 10 月 28 日，名称：用于自动聚集和投递电子个人信息或数据以及涉及电子个人信息或数据的事务处理的装置与方法）和申请人的临时美国专利申请系列号 60/134,395（提交日：1999 年 5 月 17 日，名称：用于自动聚集和投递电子个人信息或数据以及涉及电子个人信息或数据的事务处理的装置与方法）的权益。

15 本发明涉及一种自动聚集和投递电子个人信息或数据（PI）的装置与过程。本发明进一步涉及对涉及电子 PI 的事务处理的自动化。

20 回顾过去的五年，显然，随着因特网的迅猛发展，消费者需要能使他们的在线体验更加简单、易用和令人满意的应用和服务。成功的因特网站点的发展适应了在过去若干年中发展起来的许多主题。仔细分析的结论是，这种演变是正在出现的数字经济的合乎逻辑的发展。

25 1994 年之前，因特网并不是一种大众媒体，部分原因是当时业已存在的技术（FTP、Archie、Usenet 和 Gopher）并不是用户友好的，要求最终用户做所有工作（例如，最终用户要了解现有数据源，寻找地址，漫游到目的地，然后下载信息）。随着越来越多的消费者开始访问因特网，人们开发了搜索引擎来解决这种实用性问题。由于商业搜索引擎的出现，可以容易地将更多的内容添加到因特网，最终用户也获得一种寻找和访问这种信息的工具。消费者们要求有比搜索引擎更好的工具来组织和访问这种通用内容资源。人们探索过 PUSH 技术，最后，成功地采用了门户策略，作为一种消费者用来容易地访问格式单一、易用的各种内容资源的有效方法。随着可用在线内容的数量指 30 级地持续增长，门户现在面临着根据消费者的特定偏好和趣味向不同的消费者提供不同类型的内容的需要。

因特网门户（portal）和目的地站点的非凡成功，体现了创造性地、智能地对网上可用的海量信息进行聚集、组织和表示的重要性。35 搜索引擎、门户和目的地站点有基于最终用户对它们站点访问的频

率、持续时间和品位的因特网策略。由于这个原因，目的地站点和门户不断地寻求能驱使优质业务量来到它们站点并作停留的内容和/或技术。近来的趋势表明，如果按照个人偏好来组织信息，则因特网用户回访站点的可能性会增长 25 倍。

5 图 1 显示当前的获取 PI 100 的过程。在步骤 110，最终用户首先选择一个信息提供者站点。最终用户继续到步骤 120，找出并输入选定的信息提供者的因特网地址。这个步骤可以用若干种复杂程度不同的方式来完成。完成这个步骤的一个简单方法是使用书签或爱好，尽管第一次找出信息提供者时会费很多时间和努力来进行在线搜索。10 在步骤 130，最终用户用选定的信息提供者的网站的专用登录协议在该网站登录。这个协议一般要用用户名和口令或其它验证手段来验证最终用户的身15 份，从最终用户的系统上驻留的 cookies (饼干) 获取验证数据或者所请求数据与 cookie 数据的组合。最终用户继续在步骤 140 在信息提供者网站上的网页中漫游，直到找出所希望的信息。在这个过程期间，经常会要求最终用户访问其目的只是获取网站上驻留的特定 PI 而对最终用户来说用处不大或毫无用处的网页。最后在步骤 150，向最终用户提交所希望 PI。对最终用户所希望的各条 PI，重复整个过程 100。按照这个 PI 访问模型，最终用户必须访问各独立的信息提供者，为每个信息提供者留下可能不同的身份验证数20 据，使用各站点的不同用户界面，还可能在数量可观的填充网页间奔忙。

图 4 用图形表示了当前这种访问过程的体系结构。最终用户 210 用客户计算机 220 在因特网上访问各 PI 网站 250。当前这种模型有一些重大缺陷。最终用户必须单独登录到各站点。各独立站点有其自己的图形用户界面。各站点都希望最终用户停留和再来访问；各被访问站点都希望尽可能长时间地保持用户的注意。不存在真正的 PI 的聚集；多处访问只是能够顺序访问 PI 的各特定部分。

近来研究出了一种对这些问题的部分解决方案，其形式是门户站点。通用门户站点将聚集的资源分成门类，并提供向涉及这些门类包含的主题的站点的链接。Yahoo 和 Excite 就是这种通用门户站点的例子。这些站点方便了通用内容的横向聚集 (horizontal aggregation) - 横向聚集指的是对在特定信息提供者类别 (诸如银

行或公用事业公司)内 PI 访问的聚集。有的门户站点有限制地让个别最终用户能选择和配置种类不同的通用 PI (generic PI)。通用 PI 指的是对特定最终用户有价值的、不要求特定的身份验证就能获得的 PI。例如，某最终用户可能对其本地区的天气预报有兴趣。可以将这种信息综合到一个门户页中，而无需对接收这个 PI 的特定最终用户的身份验证。这种个人化的门户页向寻求聚集通用 PI 的用户提供了极大的好处。然而，当前的门户页一般不提供要求身份验证的 PI，诸如用户的证券财产目录或银行帐户余额。此外，这些页不能简化采用 PI 的事务处理。

在当前技术条件下，在因特网上聚集可用的 PI，对时间、精力和学习曲线 (learning curve) 方面等方面的要求负担很重。希望访问其 PI 的最终用户需要个别地访问许多各有自己的要求、图形用户界面和登录协议的信息提供者站点。

在本发明中，用一个连网的计算机来方便最终用户对涉及与该特定最终用户关联的电子 PI (诸如证券财产目录、本地天气、体育比赛比分、银行帐户余额或其它相关信息或数据) 的访问、操作及涉及该 PI 的事务处理。按照本发明，在连网的计算机上聚集与特定最终用户相关的 PI。由各种可选择的传递平台，诸如传真、客户计算机、电话、常规邮件、电子邮件、寻呼机、其它无线设备、网页或频道或其它传递载体，将这种信息或数据以统一的方式传递到最终用户。本发明还能促进各种涉及 PI 的电子事务诸如证券交易、零售买卖、帐单支付、银行帐户资金转移或其它事务。

按照本发明的一个传递个人信息的系统包括一个包含最终用户数据的用户储存库、一个包含信息提供者数据的提供者储存库、一个包含个人信息的个人信息储存库和一个与这些数据储存库通信的处理器。处理器支持个人信息的聚集。处理器选择一个最终用户来进行个人信息聚集。一旦选定了最终用户，处理器就连接一个或多个信息提供者。处理器然后开始从所连接的信息提供者为选定的最终用户检索个人信息。这种检索根据的是与选定最终用户关联的最终用户数据和与所连接信息提供者关联的提供者数据。检索出的数据被存入个人信息储存库。

在本发明的一个方面，用网络计算机 - 或者叫宿主计算机 - 来发

布、存储和检索与特定最终用户关联的电子信息。在特定实施例中，信息是诸如证券财产目录、本地天气、体育比赛比分、银行帐户余额或其它相关信息或数据的个人信息。按照这个方面，在宿主计算机上聚集与特定最终用户相关的 PI。宿主计算机将所聚集的数据传输到与为其而聚集数据的特定最终用户相关联的客户计算机。最好将所聚集的数据以 cookie 数据的形式传输到客户计算机并以 cookie 数据的形式由客户计算机存储。在有些实施例中，在传输之前将所聚集的数据加密。宿主计算机接收关于聚集数据的请求。请求的源最好是客户计算机，然而在本发明范围内也可以考虑是其它合适的设备源。宿主计算机从客户计算机接收 - 最好是 cookie 数据形式的 - 聚集数据。如果所聚集数据是加密的，就解密。宿主计算机继续服务该请求以生成请求结果。可以将请求结果传递到各种平台，最好是网页。另一方面，也可以将结果传递到电话、电子邮件目的地、传真、或其它打印设备，直接传递到 Web 浏览器、第三者计算机、无线设备或其它合适的传递平台。

在这个方面的另一个实施例中，在客户计算机上可存在专用软件，用它来在客户计算机上服务关于由宿主计算机传送的聚集数据的请求。这种专用软件要包含适当的解密软件。

在本发明的另一个方面，将可能来自不同的源的电子信息与从可能不同的源确定的式样首选 (style preferences) 动态地组合起来，生成一致的电子文档。在这个方面的一个实施例中，将与特定最终用户相关联的电子 PI (诸如证券财产目录、本地天气、体育比赛比分、银行帐户余额或其它相关信息或数据) 与分销商或提供者内容组合，产生所生成文档的内容。式样信息是从最终用户的首选项和发布者及提供者的式样信息收集的。通过对组合的内容应用组合的式样信息，生成适应性一致的 (adaptably compliant) 电子文档。可以由各种可选择的传递平台，诸如传真、客户计算机、无线设备、个人组织器 (organizer)、电话、寻呼机、、网页或频道或其它传递载体，以统一的方式将生成的这种文档传递给最终用户。

按照本发明这个方面的一种用于生成适应地一致的电子文档的系统包含式样合并器单元 (style merger unit)、内容合并器单元 (content merger unit) 和处理器，它们可以被包含在本发明的网

络计算机中。式样合并器单元从一个或多个式样提供者收集式样信息并动态地合并所收集的式样信息。内容合并器单元从一个或多个内容提供者收集内容信息并动态地合并所收集的内容信息。处理器接收合并的式样和内容信息，并通过将所接收的式样信息应用到所接收的内容信息来生成适应地一致的电子文档。所生成的页可以输出到各种传递平台。  
5

在另一个方面，用宿主计算机调度从一个或多个信息提供者对与一个或多个最终用户的信息的收获。宿主计算机与一个存储用户的关联数据的用户数据储存库和一个存储信息提供者的关联数据的提供者储存库通信，并且包含一个处理器。  
10

对于每个最终用户，在用户数据储存库中保存一个过去访问时间、登录时间的轮廓（profile）。对于每个信息提供者，在信息提供者储存库中保存一个以更新时间和标准为内容的轮廓。更新时间和标准可以按每个信息提供者所提供的所有信息来存储，或者，更新时间  
15 和标准可以按每个信息提供者所提供的每条信息来存储。

对于选定的信息提供者，宿主计算机处理器确定该选定信息提供者所存储信息的更新时间和可以在更新时间由更新而修改其信息的最终用户集合。宿主计算机处理器生成所确定最终用户集合中每个最终用户的预测登录时间，及生成的每个登录时间倒退预定的时间间隔。宿主计算机处理器按预测登录时间或变动的登录时间将所确定的最终用户集合分类，根据每个最终用户的变动的或预测的登录时间为每个最终用户分配一个收获时间。在这个方面的一个实施例中，宿主计算机处理器进一步可以在分配给各最终用户的收获时间从选定信息提供者为所确定集合中每个最终用户收获信息。  
20

在这个方面的另一个实施例中，宿主计算机处理器确定可以在所确定更新时间由更新修改其信息的最终用户集合的方法是，首先选择被配置成从所选定的信息提供者接收信息的最终用户，剔除那些不是配置成从所选定的信息提供者接收在所确定的更新时间受更新的信息的最终用户。宿主计算机处理器可以进一步从该集合中剔除不符合与信息提供者关联的更新或在确定的更新时间受更新的信息的更新标准或条件的最终用户。  
30

宿主计算机处理器可以根据用户储存库中存储的登录时间轮

廓，为所确定集合中每个最终用户生成一个预测登录时间。对于所确定集合中每个最终用户，判断最终用户的登录时间轮廓是否符合预定信用阀值。如果轮廓符合这个阀值，就根据轮廓分配一个预测登录时间。如果轮廓不符合这个阀值，就分配一个相当于当日当时的预测登录时间。

宿主计算机处理器根据每个最终用户的预测登录时间为每个最终用户分配一个收获时间。在这个方面的一个实施例中，每个最终用户所分配的收获时间相当于其生成的预测登录时间倒退预定时间间隔。

在这个方面的另一个实施例中，宿主计算机处理器为每个最终用户分配收获时间所根据的不仅是其预测登录时间，还要根据预期的网络活动。宿主计算机处理器首先执行一个时间上的分布拟合 (distribution fit across time)，以生成一个能确定在特定时间段内须收获的最终用户的数目的多项式函数。下一步，宿主计算机处理器确定与其及选定信息提供者关联的网络活动的网络活动曲线。生成所确定的网络活动曲线的逆。然后，它用所生成的多项式函数和网络活动曲线的逆执行一个整体匹配算法。最后，它为每个最终用户分配收获时间，将高峰收获时间重新朝时间零分布，以整平时间上的分布拟合 (distribution fit across time)。

在本发明的另一个方面，涉及与最终用户关联的个人信息 (PI) 的电子操作是自动为最终用户执行的。最终用户将具有各种与其关联的电子 PI，诸如证券财产目录、本地天气、体育比赛比分、银行帐户余额或其它相关信息或数据。在这个方面的一个实施例中，最终用户储存库含有与最终用户关联的最终用户数据和与最终用户的应答相关的触发事件的记录。宿主计算机处理器访问与最终用户关联的记录。对于每个被访问记录，宿主计算机判断记录中的触发事件是否发生，如果已经发生，就执行与确定发生的触发事件关联的应答。

应答的执行可能涉及将触发事件发生的通知传递给特定传递平台，诸如无线设备、传真、电话、打印设备、寻呼机、Web 服务器上驻留的网页、电子邮件系统或其它合适的传递载体。宿主计算机可以不传递这种通知就 - 或者除传递这种通知外还 - 自动执行一个涉及与最终用户相关联的个人信息的事务处理。

在这个方面的另一个实施例中，自动执行这种事务处理，包含宿主计算机根据在确定触发事件发生所在的被访问最终用户记录中指示的应答，访问某信息提供者的关联记录。宿主计算机与由被访问的与该信息提供者关联的数据指示的某个信息提供者计算机连接。宿主  
5 计算机然后根据被访问的与该信息提供者关联的数据、在确定触发事件发生所在的被访问最终用户记录中指示的应答、及最终用户储存库中的最终用户数据，在所连接的信息提供者计算机上执行一个事务处理脚本。

在本发明的另一个方面，宿主计算机监控最终用户与个人信息提供者之间通过中介计算机的交互作用。交互作用一般不外乎有两个类别：请求传递个人信息和请求涉及个人信息的事务处理。宿主计算机与存储最终用户的关联个人信息的个人信息储存库以及存储中介计算机的关联记帐数据 (accounting data) 的记帐储存库通信。宿主计算机包含一个处理器。  
10

宿主计算机处理器从中介计算机接收关于最终用户的关联个人信息的请求并根据个人信息储存库中该最终用户的关联个人信息服务于该请求。宿主计算机处理器更新中介计算机的关联记帐数据。宿主计算机处理器可用多种方式更新记帐数据。第一，宿主计算机处理器可以为在选定时间段内中介计算机的每个新最终用户递增用户计数。其次，宿主计算机处理器可以为通过中介计算机进行的交互作用计数。第三，在服务请求是请求进行事务处理的情况下，宿主计算机处理器可以根据所服务的请求递增服务费总额。最后，宿主计算机处理器可以用这些方式的任何组合来更新中介计算机的关联记帐数据。  
20

宿主计算机处理器根据更新的记帐数据生成给中介计算机的发票。宿主计算机处理器可在定期的基础上生成这些发票。在这个方面的另一个实施例中，宿主计算机处理器可以将所生成的发票传递到选定的目的地，诸如电子邮件目的地、打印设备、Web 服务器上驻留的 web 页、因特网客户机、电话和传真。  
25

在这个方面的另一个实施例中，中介计算机有与之关联的帐户。宿主计算机可以在生成发票之前计入该帐户的借方，使得所生成的发票仅反映中介计算机帐户表明的数额以外的其它收入。计入借方的数  
30

额可以根据用上述任何更新方式更新过的中介计算机的关联记帐数据来得出。

本发明的另一个方面是一种自动访问最终用户的关联个人信息的系统与方法，其中，在个人信息提供者上存储个人信息。将个人信息提供者上存储的个人信息对应的一个个人信息的表示和一个链接通过客户计算机提交给最终用户。当启动该链接时，客户计算机被自动驱动到通过客户程序向用户提交个人信息提供者上的一页的个人信息提供者。

在这个方面的一个实施例中，向客户机下载一个应用程序。下载的应用程序启动客户计算机与个人信息提供者之间的连接。应用程序在个人信息提供者上的各页之间漫游，直到到达该个人信息。最后，应用程序在客户计算机上将该个人信息提交给用户。应用程序可以与任何必要的与最终用户关联的和与个人信息提供者关联的数据一起生成，或者可以将这种数据传输给应用程序。与个人信息提供者关联的数据包含一个导航脚本（navigation script），用于将应用程序引导到该个人信息。与最终用户关联的数据可以包含通过导航脚本启动漫游所必需的任何数据。

在这个方面的另一个实施例中，将一个包含任何必需的用户信息和个人信息提供者数据的消息传送给客户计算机，使客户计算机自动地将该最终用户登录到个人信息提供者，由此让最终用户留在一个后登录页（post login page）上。在这个方面的一个最佳实施例中，该消息包含的一页中含有一个表格，表格中包含登录信息，当客户计算机上的软件打开登录信息时，登录信息将客户计算机重定向到一个后登录页。

在这个方面的另一个实施例中，将客户计算机驱动到个人信息的方法是，连接到个人信息提供者，漫游到个人信息提供者上的个人信息，将个人信息通过客户计算机提交给用户，代理随后在与个人信息提供者的给定会话期内客户计算机与个人信息提供者之间的交互作用。

在以下结合个附图的说明中，本发明的以上和其它的目的与优点将更加显而易见。

图 1 是最终用户执行的访问因特网可用 PI 的当前过程的过程

图。

图 2 是可用来实现本发明的部件的框图。

图 3 是 PI 引擎的部件的框图。

图 4 是当前 PI 访问体系结构的图。

5 图 5 是支持用中介网站的 PI 访问的体系结构的图。

图 6 是 Cookie/客户机高速缓存体系结构的图。

图 7 是通过图 1 的传统过程和通过跳板技术 (springboard technology) 访问特定 PI 的基础页的流程图。

图 8 表示 HTML 页的动态生成的集成模型。

10 图 9 表示 HTML 页的动态生成的运行时过程。

图 10 表示采用改进的 Java 虚拟机的自动化小应用程序交互作用的过程。

图 11 是举例说明中间网站事务处理结构 (transaction structure) 的流程图。

15 现在详细说明本发明的最佳实施例。参看各附图，各视图中相同的数字指示相同的部分。

最终用户很快将不得不登录到众多不同的网站 - 每个网站有独立的口令、安全、规则、软件和“观感” - 在一连串的操作最后，就是要通过检查邮箱，取出当前获得的信息。因特网将在根本上改变最终用户访问个人信息 (PI) 的方式，将使电子商务成为与使用 ATM 一样为人们熟悉的东西。“个人信息”是公司即信息提供者具有的、特定于各人或各人独有的所有数据，诸如每月帐单、银行帐户余额、投资信息、保健津贴、电子邮件、语音和传真消息、401(k) 保有 (holdings) 或与特定最终用户相关的可能的任何其它信息。

25 本发明通过自动聚集 PI - 不仅有由门户所聚集的通用 PI，而且有特定于需要身份验证才能访问的最终用户的 PI，缓和了当前一些 PI 获取方法具有的一些问题。在一个实施例中，本发明将 PI 获取和传递过程自动化。图 2 提供了可用来实现本发明的各成分的框图。最终用户 210 访问一个运行客户机软件 270 的客户计算机 220。在特定实施例中，客户机软件可以是一个通用网络浏览器，诸如 (Netscape 公司的)Navigator 或 Communicator。客户计算机 220 利用因特网 230 访问在 PI 宿主机 290 上运行的 PI 引擎 240。PI 引擎 240 检查所存储

PI 的新旧。如果有陈旧的 PI 项，就将其更新，方法是从在因特网 230 上访问到的在特定信息提供者的计算机系统 260 上运行的信息提供者网站 250 直接重新获取该 PI。PI 引擎 240 将新版 PI 存储在其储存库 280 中并将该 PI 传递到选定目的地 - 在本实例中是通过因特网 230 传递到客户计算机 220，后者用客户机软件 270 向最终用户 210 显示该信息。PI 引擎 240 在将所聚集 PI 传递到储存库 280 和投递目的地 - 本实例中是客户计算机 220 - 之前，要以同样的方式刷新所有陈旧的 PI。PI 引擎 240 可以顺序地或并行地刷新 PI。例如，最终用户的检查帐户余额要通过其银行的网站更新，其电子邮件从其特定电子邮件站点更新，其证券财产目录信息从其经纪人的站点更新，其电费帐单从其电力公司的站点更新。

图 3 显示的是 PI 引擎 240 的各部件的框图。PI 引擎 240 由存储和处理两种部件组成。三个主要存储部件是 PI 储存库 280、PI 提供者储存库 310 和用户储存库 360。PI 引擎 240 的第一个存储部件是 PI 储存库 280、PI 储存库 280 含有各个 PI 记录 375；与特定最终用户关联的 PI 是与所有其它最终用户的 PI 分开的。PI 引擎也利用提供者储存库 310，它保存着与特定 PI 提供者关联的通用参数。PI 提供者的通用参数定义了欲接入该 PI 提供者时要遵守的程序和必要的验证数据的类型。每个 PI 提供者记录也包含该提供者所提供的 PI 的类型和该提供者所支持的事务处理的类型。与 PI 或事务处理的类型一起包含在记录中的还有其它数据类型和访问 PI 或执行事务处理的必要程序。用户储存库 360 也是维护关于特定最终用户的配置和验证信息所必需的。对于每个最终用户来说，用户选定的 PI 提供者、PI 和事务处理，是连同从该 PI 提供者获取该 PI 或执行该事务处理所必需的验证数据一起登记的。

PI 储存库 280 可以以各种方式实现。参看图 2，PI 储存库 280 可包含一个在 PI 宿主机 290 上驻留的数据库。按照这种方式，各个最终用户 210 的 PI 是作为独立的记录或对象 375 在数据库中存储的。在另一个实施例中，各个最终用户 210 的 PI 可存储在单独的文件 375 中，所以在文件层上执行分离不同用户的 PI 的任务。

此外，或者作为替代，采用 cookie 技术，与每个最终用户 210 关联的 PI 可以驻留在其客户计算机 220 上。这种 cookie 技术登载于

D. Kristol 和 L. Montulli 的《HTTP 状态管理机制》意见征询 (RFC) 2109 (1997 年 2 月) (访问网址: <http://www.ietf.org/rfc/rfc2109.txt>) , 本文明确地全文引用。与每个最终用户 210 关联的 PI 要被存储为 PI cookies 375。这种实现机制为将一个最终用户的关联 PI 375 与所有其它最终用户的关联 PI 的分离提供了内在的支持。采用这种方法来代替中央化储存库, 提供了一个防非授权访问的安全层。进一步的措施是, 可以将 cookies 中存储的 PI 数据以加密格式存储。

图 6 是采用 cookies 技术的 PI 储存库 208 的典型实现的示意图, 10 关于 PI 引擎 240 的内部操作, 也要参考上述对图 3 的说明。如果用户试图直接地或通过中介 Web 服务器访问 PI, PI 引擎 240 的访问/事务处理部件 340 就从 PI 储存库 280 中检索所存储的 PI 375。按照这种方式, 该存储的 PI 375 将被直接从最终用户 210 的客户计算机 220 发送的 cookies 接收。PI 访问/事务处理部件 340 将进行任何必要的解密。所要求的任何更新都由 PI 提供者 250 的直接访问所获得。PI 传递部件 350 将提供机制来更新 PI 储存库 280 以及将所请求的 PI 直接地或通过中间网站传输到最终用户 210。PI 传递部件 350 通过替换客户计算机 220 上存储的过时的 PI cookies 375 来将更新过的 PI 放入 PI 储存库 280。PI 传递部件 350 也要进行任何必要的加密。PI 20 传递部件 350 还负责传输所请求的 PI。在最佳实施例中, PI 储存库 280 要用这种基于 cookies 的体系结构来实现。

用户储存库 360 可以以各种方式实现。参看图 2, 用户储存库 360 可以包含一个驻留在 PI 宿主机 290 上的数据库。按照这种方式, 各个最终用户 210 的个人配置数据都是以独立的记录或对象在数据库中存储的。此外或者作为替代, 最终用户数据可以按类似于以上就 PI 储存库 280 所述的 cookie/高速缓存体系结构的方式来分布。

在最佳实施例中, 用户储存库 360 可以通过个人信息配置 (PIC) 30 文件来实现。PIC 文件为每个最终用户存储一个安全加密的个人轮廓, 内容诸如是姓名、地址、社会保障号。PIC 文件便于最终用户通过最终用户配置部件 330 自动向信息提供者注册。这个部件将读取 PIC 文件并用所提取的个人信息预先填充选定各提供者的注册模板。然后, 如果需要, 它将提示用户输入要求输入的但在轮廓中没有的信

息。如果信息是完整的，注册就自动完成。下一步，最终用户配置部件 330 完成任何提供者表格，得到应答并更新最终用户的 PIC.

这四个主要处理部件访问并操作三个储存库中的数据。处理部件可以在单一处理器上或多处理器上执行，前者例如基于奔腾级（MMX、  
5 PRO、II、III 等）中央处理单元的文件服务器计算机系统或同等系统。如图 3 所示，这四个处理部件是基准配置部件 320、用户配置部件 330、PI 访问/事务处理部件 340 和 PI 传递部件 350。基准配置部件 320 提供通过其向系统添加新用户可选择的 PI 提供者的界面。这个部件 320 可以以各种方式实现，包括试探出错后由人工输入配置信息，  
10 半自动试探出错（自动的超文本标记语言（HTML）<FORM>元素、Javascript 函数和 Java 小应用程序的定位）后由人工输入配置信息，或者最好举例配置（在模拟的 Web 客户机中执行协议，模拟的 Web 客户机自动生成一系列所需数据和访问过程中的一系列步骤）。这些过程要在两个层次被使用：第一层是对特定 PI 提供者的一般访问  
15 所需的数据和步骤的集合，第二层是在 PI 提供者站点访问各特定 PI 项时所需的另外的数据和步骤的集合。基准配置部件 320 可以在有新的 PI 提供者加入系统时独立启动，也可以由于 PI 访问/事务处理部件 340 的故障、可能表明对失败访问的访问要求有变化而导致启动。后一种警告更可能发生的情形是：PI 访问/事务处理部件 340 应  
20 最终用户的请求来验证前面通过最终用户配置部件 330 输入的必要访问数据，而将提供者储存库 310 提供的对访问 PI 提供者的一般要求和对访问 PI 或事务处理的特定要求二者与用户储存库 360 提供的最终用户数据比较，最后发现不一致。如果确定不一致，就对提供者储存库 320 作更新，使提供者数据与当前的访问/事务处理的要求一致。  
25

最终用户配置部件 330 允许最终用户选择和配置特定用户感兴趣的 PI 和事务处理。这种配置信息保存在用户储存库 360 中。当最终用户向按照本发明的系统预订时，系统允许用户选择所希望的 PI 和/或事务处理的类型和源。首先，系统请求最终用户允许它代表用户获得任何选定的 PI 来执行任何授权的事务处理。然后，系统从提供者储存库 320 向用户提供一系列已知的信息供给者和特定 PI 提供者所提供的 PI 的类型及所支持的事务处理的类型。系统请求由 PI 提

供者要求的、访问各 PI 提供者所必需的验证数据以及特定 PI 和/或事务处理所要求的其它数据。假定最终用户是所选定的 PI 提供者的已注册用户，或特定 PI 提供者不要求在先注册，则将最终用户提供 的数据放入用户储存库 360。

5 一种获得任何 cookie 数据的方法是最终用户用 PI 引擎 240 作为代理服务器访问各个以前访问过的 PI. PI 引擎 240 将 cookie 数据与适当的网页请求传送给 PI 提供者，以获得 PI 或执行事务处理，并在最终用户的准许下在用户储存库 360 中的该用户的记录中保留 cookie 数据的副本。另一种获得 cookie 数据的方法是从最终用户的计算机直接上载 cookie 信息。在最佳实施例中，如果用户已经在提供者登录过，就不需要 cookie 数据。所需的只是用于登录的验证数据。  
10

如果最终用户因为不是选定 PI 提供者的注册用户而没有必需的信息，用户配置部件 330 就提示用户提供将最终用户向 PI 提供者注册所必需的信息，并执行 PI 提供者所要求的注册程序。模拟的 Web 客户机可以通过提供所需访问数据和发送任何必需的 cookie 数据而自动地执行这个过程。这种模拟客户机为最终用户注册的方式主要取决于 PI 提供者网站所用的交互方法。如果该网站使用 HTML 表格和公共网关接口（CGI）应用程序，则最终用户配置部件 330 可以构建一个能模仿表格实际使用效果的统一资源定位器（URL）并将该 URL 提交给模拟的 Web 客户机。用 URL 模仿表格相当于人工地向 Web<表格>（Web<FORM>）元素输入数据。参看 Kerven、Foust、Zakour 的《HTML 15 3.2 及问题解答》（HTML 3.2 Plus How-To Waite Group Press 出版，1997，559 - 569 页）。如果网站使用 HTML 表格与 Javascript 20 函数的组合，则用带改进的 Javascript 解释程序的模拟的 Web 客户机遵循特定 PI 提供者的最终用户注册过程就能有效地为用户注册。要遵循的注册过程可从提供者储存库 320 内该特定 PI 提供者的记录中获得。模拟的 Web 客户机中的 Javascript 解释程序将按照这个过程并提供由最终用户提供的数据。如果 PI 提供者网站上的注册过程 25 采用 Java 小应用程序，则也可以使用类似的过程。带 Java 字节码解释程序的网络客户机通过遵循储存库 320 内特定 PI 提供者的最终用户注册过程就能有效地为用户注册。字节码解释程序会提供最终用户  
30

以前输入过的数据，而不要求由最终用户进行交互式输入。如果 PI 提供者采用表格、脚本和小应用程序的组合，则可以将上述各个过程组合起来完成所希望的注册。

参看图 2 和图 3，用改进的 Java 虚拟机（VM）可以实现 PI 引擎 5 240 的各种功能部件与可通过提供者网络服务器 250 得到的 Java 小应用程序之间的自动交互。用于与特定小应用程序交互的模板可驻留在提供者储存库 310 上。这种模板所使用的具体输入数据可存储在用户储存库 360 中。当诸如最终用户配置部件 330 或访问/事务处理部件 340 的功能部件要求与提供者网络服务器计算机 250 上的 Java 小 10 应用程序自动通信时，改进的 Java 虚拟机就为这种交互提供条件。

图 10 表示一个用这种改进的 Java 虚拟机实现这种自动交互作用的过程。在步骤 1010，要求交互作用的功能部件标识提供者及部件需要与之交互作用的提供者上的特定小应用程序。在步骤 1020，部件访问提供者储存库 310 中与小应用程序交互作用所必需的模板。继续到步骤 1030，部件访问用户储存库 360，以获得模板所要求的数据。改进的 Java 虚拟机在步骤 1040 解释小应用程序，并且并不像标准 Java 小应用程序执行时那样要求来自用户的交互式输入，而是等待来自 PI 引擎的交互功能部件的输入或是输出到 PI 引擎的交互功能部件。在步骤 1050，功能部件按照所访问的模板向改进的 Java 虚拟机 20 提供输入数据并按照所访问的模板检索数据和接收输出数据。只要小应用程序继续输入或输出其它数据，步骤 1040 和 1050 就重复执行。小应用程序结束时，功能部件在步骤 1060 继续其自己的处理。

注册成功后会向最终用户显示注册信息供将来参考。此外，最终用户配置部件 330 还在用户储存库 360 中存储 PI 提供者必要的访问验证数据和访问选定 PI 或事务处理所要求的额外数据。 25

在这种自动注册的最佳实施例中，任何必需的 cookie 数据都为最终用户配置部件 330 接受并按需存储起来。在许多情况中，cookie 数据是通话特有的，因此长期用途不大。在注册期间生成的 cookie 仅在注册期间使用，一旦注册完成就丢弃。

30 注册失败的原因有几种情况。首先，试图向 PI 提供者注册的最终用户不符合注册条件，例如试图向银行注册的最终用户在该银行没有帐户，而银行只对帐户持有者开放。其次，最终用户可能提供了不

适当或不正确的数据，例如银行注册过程可能要求社会保险号、口令、银行帐号和最终用户母亲的父姓，如果用户输入了不正确的社会保险号，注册过程就会失败。最后，PI 提供者可能已经修改了其站点的注册程序。在这种情况下，遵循由提供者储存库 320 提供的过程会导致注册的失败。在任何注册失败的情况下，都会将最初向系统提供的注册数据向最终用户表示。系统然后会请最终用户复查所提供的信息的准确性，如果发现错误就更正，然后再提交数据。如果由于提交相同的必需数据而导致第二次失败，就会生成一个向最终用户表示的出错消息，表示要么用户没有从所选定的 PI 提供者访问所选定 PI 的资格，要么由于 PI 提供者所作的修改导致了注册的出错。这第二次失败也会触发一个警告，建议可能需要重新配置提供者储存库 320 中该 PI 提供者的记录。

最终，用户储存库 360 就含有每个最终用户的记录。前文说过，这个记录可以是一个数据库项、一个或多个 cookies 或者是一个诸如 PIC 文件的文件。每个记录标识所选定的各 PI 提供者及所需的一般访问验证数据并且在每个 PI 提供者下标识该最终用户感兴趣的、该特定 PI 提供者所提供的 PI 和所支持事务处理以及访问该 PI 或执行该事务处理所需的任何额外数据的表。准确地说，诸如最终用户名的重复信息只在记录中集中存储一次。

最终用户配置部件 330 也允许最终用户选择一个或多个投递目的地。一个目的地可能会是最终用户的计算机 - 其代表是图 2 中所示的运行客户机软件 270 的客户计算机 220。然而，计算机并不是本发明所设想的唯一目的地。PI 传递的目的地可以包括传真、电子邮件、电话、常规邮件、寻呼机、诸如 Palm Pilot(3 Com) 的其它无线设备、网页或频道、网络浏览器或其它传递机构。本发明也设想由最终用户用网站作为中介间接地访问 PI，然而，这种间接访问不要求最终用户指定传递目的地，除非希望有另外的传递选择。

此外，可以考虑通过由图 2 中所示的运行客户机软件 270 的客户计算机 220 经因特网直接访问 PI 引擎来访问最终用户配置部件 330。然而，其它访问方法也是同样可行的。例如用户可通过使用中介网站来间接访问 PI 引擎。另一种选择方案是用电话接口来实现对最终用户配置部件的访问。

参考图 3, PI 访问/事务处理部件 340 支持 PI 引擎 240 的更新、获取和事务处理功能。PI 访问/事务处理部件 340 负责访问和存储用户 PI 并执行最终用户授权的事务处理。当需要为选定的最终用户进行访问或更新时, PI 访问/事务处理部件 340 就综合提供者储存库 320 和用户储存库 360 的信息, 以更新 PI 储存库 280 中的最终用户 PI。  
5 对于每条要求访问或更新的 PI, PI 访问/事务处理部件 340 在提供者储存库 320 中查找该特定 PI 所需的访问过程和信息。在用户储存库 360 中找出验证与访问数据。PI 访问/事务处理部件 340 用这个信息在因特网上连接该 PI 提供者的网站, 以访问该 PI。如果有多条 PI  
10 要求更新或访问, 则访问可以串行地或并行地进行。

被请求的事务处理也得到类似的支持。对于每个事务处理, PI 访问/事务处理部件 340 综合提供者储存库 320 和用户储存库 360 的信息, 来执行所请求的事务处理。PI 访问/事务处理部件 340 在提供者储存库 320 中查找该特定事务处理所需的事务处理过程和信息。在用户储存库 360 中找到验证和访问数据。PI 访问/事务处理部件 340  
15 用这个信息从该 PI 提供者的网站在因特网上执行该事务处理。

模拟的 Web 客户机可以自动执行访问或事务处理过程, 同时提供必要的访问和验证数据。这种模拟客户机访问 PI 或执行事务处理的方式, 极大地依赖于 PI 提供者网站上所用的交互方法。如果网站采用 HTML 表格和公共网关接口 (CGI) 应用程序, 则 PI 访问/事务处理部件 340 可以构建一个统一资源定位器 (URL) 来复制表格实际使用效果, 并将该 URL 提交给模拟的 Web 客户机。用 URL 模仿 HTML 表格相当于人工向 Web<FORM>元素输入数据。参看 Kerven, Foust, Zakour 的《HTML 3.2 plus How-To》(Waite Group Press 出版, 1997,  
20 559 - 569 页)。如果网站使用 HTML 表格与 Javascript 函数的组合, 则带改进的 Javascript 解释程序的模拟的 Web 客户机只要分别按照特定 PI 或事务处理的 PI 访问/事务处理过程就能有效地访问 PI 或执行事务处理。要遵守的访问/事务处理过程可从提供者储存库 320 中该特定 PI 或事务处理的记录中获得。模拟的 Web 客户机中的  
25 Javascript 解释程序将按照这个程序及提供在用户储存库 360 中找到的数据。如果 PI 提供者网站采用 Java 小应用程序, 则也可以使用一个类似过程。带 Java 字节码解释程序的网络客户机通过遵循提供  
30

者储存库 320 内存储的该特定 PI 或事务处理的过程就能有效地访问 PI 或执行事务处理。字节码解释程序会提供来自最终用户储存库 360 的数据，而不要求由最终用户进行交互式输入。如果 PI 提供者 Web 站点采用表格、脚本和小应用程序的组合，则可以将上述各个过程组合起来完成所希望的访问。

在这种自动访问或事务处理的一个最佳实施例中，任何必需的 cookie 数据都为 PI 访问/事务处理部件 340 接受并按需存储起来。在许多情况中，cookie 数据是通话特有的，因此长期用途不大。所生成的 cookie 仅在这些功能期间使用，一旦挖掘 (mining) 或事务处理操作完成就被丢弃。

为了在登录后迅速向最终用户提供个人信息，PI 访问/处理部件 340 有必要在最终用户登录之前就选择数据收获的最终用户。一种解决方法是，每当某最终用户直接或通过中介网站请求访问其 PI 时就更新最终用户的所有 PI。另一种方法是，每当向特定提供者请求 PI 时，就更新该提供者所提供的某最终用户的所有 PI。所以，由最终用户登录到系统的操作，实际上为立即 PI 更新选择了该最终用户。然而，这种方法可能导致对 PI 引擎 240 资源的低效使用。

鉴于潜在用户和提供者的庞大数目以及提供尽可能最新数据的目标，另一个实施例包括一个为优化选择从提供者收集数据的最终用户的进度表 (Schedule) 而开发的算法。这个算法分解是提供者的更新策略、用户的登录习惯和用户-提供者帐户特点。适当应用该算法，会保证对给定用户来说要尽可能不频繁进行 PI 的收获，由此尽量减少系统资源的消耗。

如果能准确预测下一个提供者更新时间和下一个期待的用户登录，就能创建一个能更聪明收获的模型。并非在提供者更新其网站时就立即收获提供者所有用户的数据，而是可以根据用户的预期登录时间和网络活动轮廓 (profiles) 随着时间的延续而分散地收获。例如，如果提供者 A 在星期五夜间更新其网站，并且预计该提供者的一大批用户在星期一早晨之前不会登录，则收获负荷就可以分布在若干天中。这种作法的优点是将 PI 引擎 240 的负荷峰值以及由 PI 引擎 240 对提供者带宽的耗费都最小化。要取得这种优化，PI 引擎 240 就必须维护并改进每个提供者和用户的模型。可以将这种数据分别维护在

提供者储存库 310 和用户储存库 360 中。

每当用户使用 PI 引擎 240 时，都可以捕获时间和日期。一旦积累了足够的登录次数，就可以就每月哪些天、每周哪些天、每天哪些时间来分析登录时间。在一个模型中用这些分析结果来预测预期的下一次用户登录。然后用以后的登录来测试并改进该模型，直到建立了一定程度的可信度。一旦确定了较高的可信度，就将该用户模型并入到自适应收集调度程序中。在特定最终用户的模型达到较高的可信度之前，可以使用上述的收获方法之一。

每个提供者根据因其独特资源和商务模型而产生的策略更新其站点。要使任何自适应的调度程序 (adaptive scheduler) 工作，必须使每个提供者的策略模型化。在有些情形中，策略是不言自明的。在另外一些情形中，必须根据经验来确定策略。提供者的策略最可能是下述各类型之一：

- 类型 I. 为所有用户定期更新的
- 15      · 类型 II. 相对于每个用户定期更新的
- 类型 III. 以伪随机 (pseudo-random) 方式更新的

根据提供者的类型可以使用以下方法。

#### 第 I 类提供者策略调度算法

1. 假设具有“无可信度”模型的用户有一个立即登录时间。
- 20      2. 根据用户的预测登录时间对用户按时间顺序排序。
3. 将所有用户的预期登录时间后推 1 小时。
4. 执行一个沿时间边界的密度曲线拟合 (density curve fit)，以得到一个多项式函数，可用多项式函数来确定给定期期要求收获的用户帐户的数目。
- 25      5. 用讨论中的时间段的网络活动曲线的逆执行一个累计匹配算法 (integral matching algorithm)，以调节分布曲线。
6. 可能的话，朝时间零的方向重新分配高峰收获时间，以整平分布曲线。
7. 按照分布曲线向排序的用户分配收获时间。
- 30      8. 监控时间并在适当时收获用户帐户。

#### 第 II 类提供者策略调度算法

对于这类的每个提供者来说，必须标识一个确定个人信息何时被

更新的用户属性。在有些情况中，可能需要向用户询问这种信息。其它情况下，可以从所收获的信息中确定这种信息。如果用这些手段的哪一种都不能确定用户的这个属性，可以每天监控该提供者网站上个人信息的变化，直到确立了一个模式。

5 由于给定的一天中提供者所更新的帐户是自然、均匀分布的，所以可以在用户预期登录时间之前一小时收获用户的帐户。如同第 I 类算法一样，具有“无可信度”模型的用户应当被立即收获。

### 第 III 类提供者策略调度算法

这种策略是最困难的策略。由于提供者更新用户帐户的方式不是  
10 确定不变的，所以每个提供者要决定信息对用户的要害程度。对于那些非常关键的提供者来说，应当每天、甚至更频繁地收获每个用户帐户。对于那些不太关键的提供者来说，应当次数较少地、可能在总体的系统活动程度较低时收获用户帐户。

PI 传递部件 350 负责将 PI 格式化并传递到最终用户。传递一般  
15 是在对所有过时的 PI 更新之后进行的。除通过中间网站访问的 PI 外，PI 将被传递到如在用户储存库 360 中所指定一个或多个目的地  
（例如传真、电话、寻呼机、网络浏览器、电子邮件等等）。如果目的地不是中间网站，PI 传递部件就执行将 PI 传递到适当的目的地所必须的所有格式化。例如，如果目的地是网络浏览器，就将 PI 按 HTML  
20 文档进行格式化；如果目的地是电话，就将 PI 提交去进行语音合成和传输。

就目的地是中间网站的情况而言，PI 是以一种能被中间网站配置的格式传递的。图 5 是本发明采用中间网站的一个可能的实施例的示意图。最终用户 210 用客户计算机 220 在因特网 230 上访问中间网站 510。最终用户 210 登录到中间网站 510。中间网站 510 在因特网  
25 230 上联系 PI 引擎 240，并从 PI 提供者网站 250 直接接收按要求更新过的该最终用户的 PI。中间网站 510 接收 PI，按照其特定的风格和图形用户界面，将其并入若干页，并将这些页传递给最终用户 210。PI 引擎 240 的使用对最终用户 210 是透明的。此外，起着给最  
30 终用户 210 聚集 PI 的作用的中间网站 510，可以—并极其可能—同时起 PI 提供者的作用。

在另一个实施例中，这种格式化是通过一个结合各种来源的样式

和布局信息的动态 HTML 生成系统而发生的。PI 传递部件 350 动态地生成定制的 HTML 页。这些页是根据来自各种源的若干样式要素（诸如背景颜色、前景颜色、字体大小、颜色和样式、页面布局等等）和来自各种源的内容而定制的。信息提供者、发布者、最终用户、PI 传递部件 350 或这些源的任何组合、或者其它相关源，都可以提供用于页面生成的定制要素。最后，每个 HTML 页必须用数据填充。这种页中使用的数据，可以来自例如信息提供者、发布者、最终用户、PI 传递部件 350 或这些源的任何组合、或者其它相关源。所要求的解决方案是一个代表在运行时执行这种 HTML 生成的通用算法的系统。样式和内容可以以任何适当的格式提供，例如可扩展样式表语言（XSL-Extensible Stylesheet Language，由 W3C 在 <http://www.w3.org/TR/WD-xsl> 中规定，本文全部采用作为参考）和/或可扩展标记语言（XSL-Extensible Markup Language，由 W3C 在 <http://www.w3.org/TR/REC-xsl> 中规定，本文全部采用作为参考）或者其它合适的格式化标准。对这种系统的关键要求是问题域的完全封装和运行效率。

在最佳实施例中，解决方案根据的是如图 8 所示的以下基本模型：

1. 确定 6 组定制要素：发布者内容 810、提供者内容 820、发布者样式规范 830、提供者样式规范 840、用户特定的内容 850 和用户特定的样式 860。
2. 每组定制要素 810-860 被视为是向执行动态页生成的运行时系统 870 的一个单独、独立和必要的输入。
3. 每个输入 810-860 都将采用 XML 流的形式。
4. 输出 880 将采用 HTML 流的形式。
5. 动态页生成系统 870 针对每组 6 个有效输入 810-860 生成有效输出 880。

图 9 表示了由这种系统 870 实际执行的运行时输入处理序列：

1. 由内容合并器单元 910 将发布者内容 810 与提供者内容 820 及用户特定的内容 850 组合，以便产生一个完整的内容说明 930。
2. 由样式合并器单元 920 将发布者样式 810 与提供者样式 840 及用户特定的样式 860 组合，以便产生一个完整的样式说明 940。

3. 由样式应用器 950 将样式说明 940 应用到内容说明 930, 以便产生生成页 880.

为了完全地封装问题域, 必须对系统 870 作以下要求:

1. 每个 XML 输入 810-860 都是有效的 XML 流。
- 5 2. 内容说明 810、820 和 850 对于同一个文档类型定义来说全部都是有效的。

3. 样式说明 830、840 和 860 对于同一个文档类型定义 (诸如 XSL DTD 标准) 来说全部都是有效的。

4. 以接受两个或更多 XML 流并产生组合的 XML 输出为任务的合  
10 并单元 910 和 920 必须能够为任何一组有效的 XML 输入生成这种输出。

另一个执行这种任务的方法是将 PI 格式化或带有预定义的类 (CLASS) 属性的 HTML 单元。接收这些单元的中间 Web 网站可以动态地将它们附入向 PI 的最终用户传递的页中。并入这种单元的页可以包括与预定义的类 (CLASS) 集相关的不同样式信息。可以用第 1 层  
15 级联样式表来实现这种可配置性。参看 Kerven、Foust、Zakour 的《HTML 3.2 Plus How-To》(Waite Group Press 1997 年出版, 651-693 页) 和 Walsh 的《An Introduction to Cascading Style Sheets》  
20 (World Wide Web Journal 杂志, 1997 年冬季, 147-156 页)。这个选择对中间网站的程序支持要求最小, 但是对中间网站向最终用户提交 PI 的灵活性有一定程度的限制。

另一方面, 中间网站可以用标准化应用程序设计接口 (API) 开发一种应用程序来直接访问 PI 数据。在这种情况下, 可以不用 PI 传递部件 350, 也可能要将 PI 传递部件 350 用作负责服务于对数据的 API 请求的部件。按照这个模型, 中间网站要负责作出对原始 PI 数据格式化的全部决定。这个实施选择要求中间网站更多的程序支持, 但是允许中间网站在使用原始 PI 时有更大的灵活性。

能利用中间网站来传递 PI 是有十分重要的用途的。这种功能使已经熟悉某个现有 PI 提供者的最终用户不仅能访问该特定 PI 提供者  
30 的相关 PI, 也能方便地在熟悉的用户界面—即该现有 PI 提供者网站中访问其它 PI 提供者的所有 PI。在这种情况下, 对 PI 的请求直接发端于中间 PI 提供者网站而间接发端于最终用户。安全措施会限制

对授权的中间网站的访问。这些措施可能包括对最终用户和中间网站的验证。此外，为了更加安全，可能还要求验证最终用户与特定中间网站之间的关联。

此外，中间网站的使用也支持一种新型的事务处理模型。在这个事务处理模型中，  
5 中间网站补充或全面补偿 PI 引擎管理员向最终用户提供服务。这些事务处理由于 PI 引擎的审计和跟踪功能而变得更加方便。这些功能允许对按人的费用、按交易的费用、按访问的费用或它们某种组合的计算进行评估。可以直  
10 接向中间网站要求支付各评估值。或者，可将这些值从向中间 Web 站点收取的最低月费中记帐，而超出最低收费的费用则直接向中间 Web 站点收取。

图 11 表示按所述模型的典型过程的流程图。在步骤 1110，中间网站支付最低月费。在步骤 1120，PI 引擎审计和跟踪用户通过中间网站的使用情况。所审计的使用情况用于按用户、按访问、按交易或它们的组合评估费用。在步骤 1130，将所审计的数额从在步骤 1110 支付的费用计入借方。在步骤 1140 向中间网站收取超出所支付的最低费用的任何费用。  
15

最终用户经常需要访问由特定 PI 的提供者生成的基础网页。传递部件可以不仅传递 PI，也传递直通提供该 PI 的提供者的网页的访问点。访问点的形式可以是链接、表格按钮或其它某种交互式访问机制。  
20

这种访问点大大地提高了最终用户访问基础网页的效率，如图 7 所示。在访问 PI 的传统过程 100 中，最终用户必须经历许多中间网页才能到达所需的网页，这期间要求进行各种经常是繁琐的交互作用。

最终用户首先必须标识提供者（110）。下一步，最终用户必须定位该提供者的网址（120）。然后，用户请求提供者的登录页面（130）。如果最终用户忘记了登录所必需的信息，就必须找出这种信息，否则就不能通过万维网访问所需的信息。最终用户然后漫游提供者的网站（140）。这通常必然要访问提供者的主页（710），然后在提供者网站上浏览各种中间网页（720）。最终用户可能不得不数次返回主页（710），也可能偶然地完全离开系统，只好再次登录（140），直到最终定位所需的信息（150）。  
25

如果采用跳板技术，整个过程被精简成只要点击一次访问点。PI 引擎的传递部件连同 PI 一起传递一个对提供者的基础网页的访问点。结果，最终用户只要与 PI 表示页进行一次交互作用（760）。这种交互作用立即执行为将用户带到所需基础网页（150）而必需的与提供者网站的交互作用。

在一个实施例中，可以用 Java 小应用程序来实现这个跳板技术。参看图 2，小应用程序要由最终用户的客户机软件 270—通常是网络浏览器—从 PI 宿主机 290 下载，并由最终用户的计算机 220 在本地执行。小应用程序会将客户机软件 270 驱动到所需的页。这种小应用程序能从提供者储存库 310 和用户储存库 360 提取用于驱动客户机软件的程序和数据。

在另外一个实施例中，PI 引擎 240 可以作为一个按需直接访问提供者储存库 310 和用户储存库 360 的代理服务器工作。当 PI 引擎 240 接到向特定信息的源跳转的请求时，该引擎就执行必要的操作，以漫游到所希望的页，并将所希望的页传送到最终用户的计算机 220。与该页的进一步交互作用可能要求由 PI 引擎 240 进行其它的代理操作，因为累积的 cookie 数据可能驻留在 PI 宿主机 290 上。这个实施例局限于在处理标准 HTTP 通讯而不是安全 HTTP 通讯中使用。

在最佳实施例中，跳板为最终用户提供向 PI 提供者网站 250 的自动登录并允许最终用户 210 通过客户机软件 270 漫游。这种自动登录可以通过使用超文本传输协议（HTTP）重定向来完成。当接收到最终用户 210 通过客户机软件 270 发出的跳板访问请求时，PI 宿主机 290 就向该跳板访问的目标 PI 提供者网站 250 请求登录页。在 PI 宿主机 290 上运行的 PI 引擎 240 接收这个登录页并通过访问提供者储存库 310 和用户储存库 360 中的适当数据来构造一个登录请求。该登录请求被内置在向客户机软件 270 发送的 HTTP 重定向中。客户机软件 270 被重定向到目标 PI 提供者网站 250，然后将最终用户 210 自动登录到该网站。

另一方面，这个功能也能通过如上所述的 Java 小应用程序来实现。此外，PI 引擎 240 还能生成一个含有有关登录请求而不是 HTTP 重定向的 Javascript 页。该 Javascript 页可以被返回到客户机软件 270。然后由客户机软件 270 执行该页，以完成自动登录。

图 3 的 PI 引擎 240 也可以包含一个网站监控器 370 处理部件。这个部件有系统地监控所支持的 PI 提供者网站的变化。这个部件加强了系统识别 PI 提供者网站过程、数据要求和 cookies 要求中的变化的能力。这个部件由于通过来自 PI 访问/事务处理部件 340 的反馈来补充或替代变化标识而提高系统的效率。  
5

本发明的另外一个实施例可支持 PI 的本地化操作。这在图 2 的客户计算机 220 上运行的客户机软件 270 是一个专用网络客户软件而不是诸如 Netscape 的通用网络客户程序时可以实现。这个专用客户程序可以用 Web 频道技术来使本地 PI 下载和更新过程自动化。如果  
10 PI 储存库是通过前文所述的 cookie 体系结构实现的，这个专用客户程序就可以提供对所存储的 PI 的直接本地访问。

在另一个实施例中，图 3 的 PI 引擎 240 可以既支持系统支持的 PI 提供者也支持特定于特定最终用户群的 PI 提供者。在这个实施例中，最终用户不限于只能使用存在于提供者储存库 310 中的 PI 提供者的 PI。如果最终用户要增加由不受支持的 PI 提供者提供的数据，  
15 则最终用户要访问基准配置部件 320 并为该不受支持的 PI 提供者创建一个配置。该提供者和 PI 配置连同验证和访问数据，要与用户的记录一起存储在用户储存库 360 中。

本发明的另一个实施例支持在图 3 的提供者储存库 310 中加入 PI  
20 事务处理过程和访问要求。实现这种事务处理所需的最终用户特定的信息要与用户记录一起驻留在用户储存库 360 中。PI 访问/事务处理部件 340 的功能要扩展到支持执行事务处理。这个额外的功能可以以与前文针对用模拟的 Web 客户机执行访问所述的过程类似的方式得到支持。这个实施例的另外一个特点是包含通过提供自动启动事务处理的触发事件而实现的自动化或半自动化的帐户管理。  
25

例如，图 2 的最终用户 210 能通过 PI 引擎 240 保持其帐户的在线状态。如果某信息提供者能够接受在线支付，PI 引擎 240 就能支持这种事务处理的完全活部分自动化。如果某个信息提供者有帐单到期日期，PI 引擎 240 就能设置该信息标志并向最终用户 210 发电子邮件，通知帐单到期。所以，用户就不必逐个地向每个提供者查询到期日期信息。PI 引擎 240 也能为允许通过网络服务器 260 支付的提供者自动实现有限帐单额度内的支付，然后用电子邮件将支付通知发

送给用户。

到期日期的获取可以用图3所示的PI访问/事务处理部件340来完成。通过任何由PI传递部件350支持的传递都能将到期日期信息提供给最终用户。PI访问/事务处理部件340会用标准电子商务帐单支付方法来向提供者支付用户的帐单（如果用户选择这种方式的话）。一旦支付了帐单，就向用户发送电子邮件通知，内有提供者信息和付帐信息。用户能在用户储存库360中规定自动支付的数额范围。如果帐单超出用户规定的数额，PI引擎就向用户发出电子邮件通知，而不是自动地支付帐单。

以上所述的各实施例都是仅作为示例性的例子给出的。非常显然，可以对本说明书中披露的具体实施例作出许多改动而不偏离本发明。因此，本发明的范围应当由下面的各项权利要求来确定而不是局限于以上具体描述的实施例。

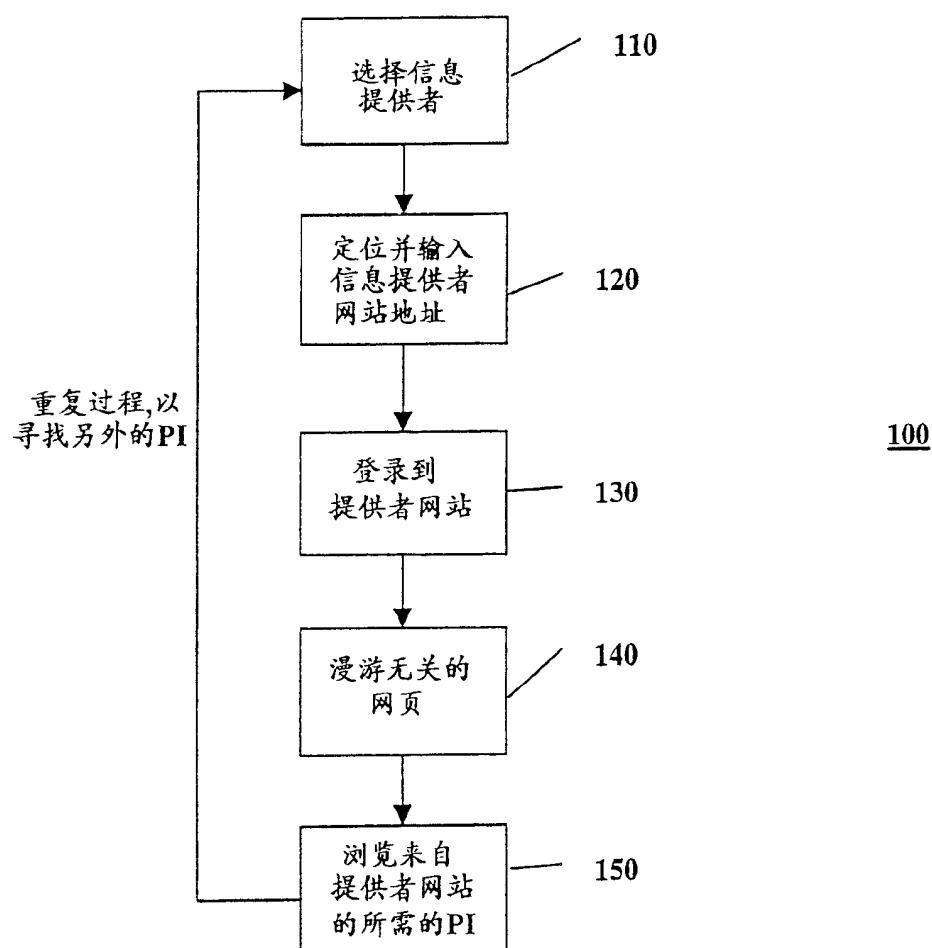


图 1  
(现有技术)

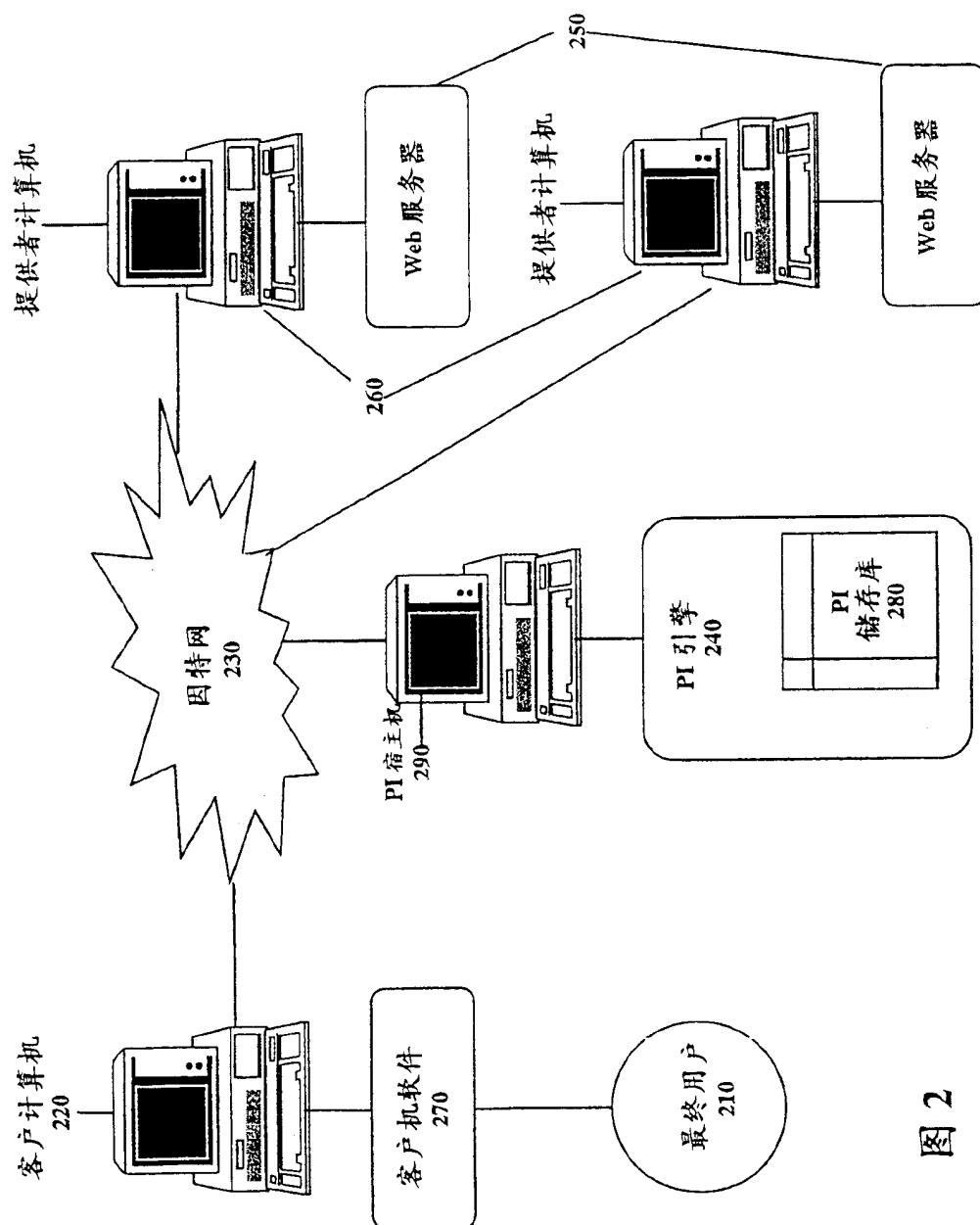


图 2

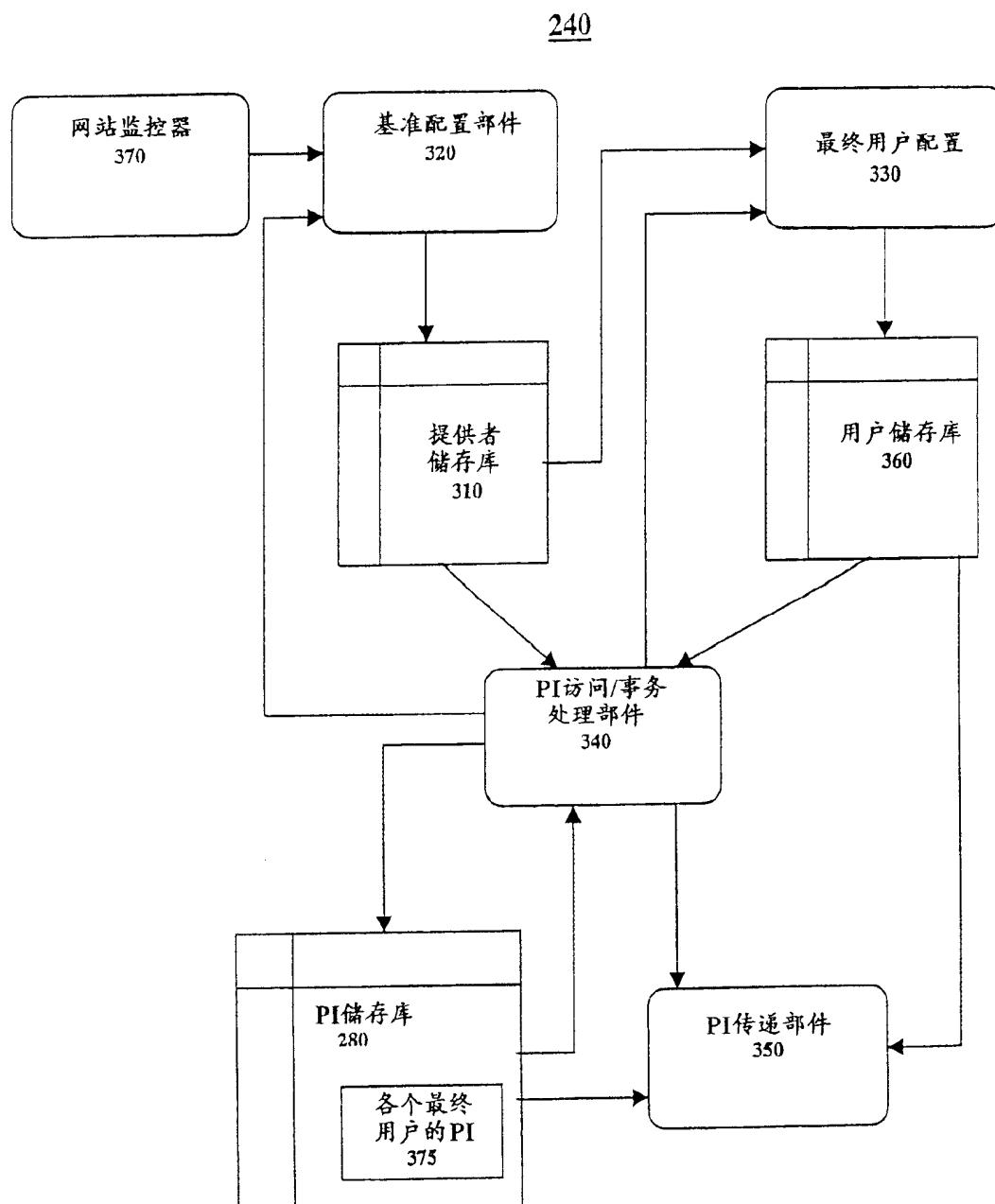
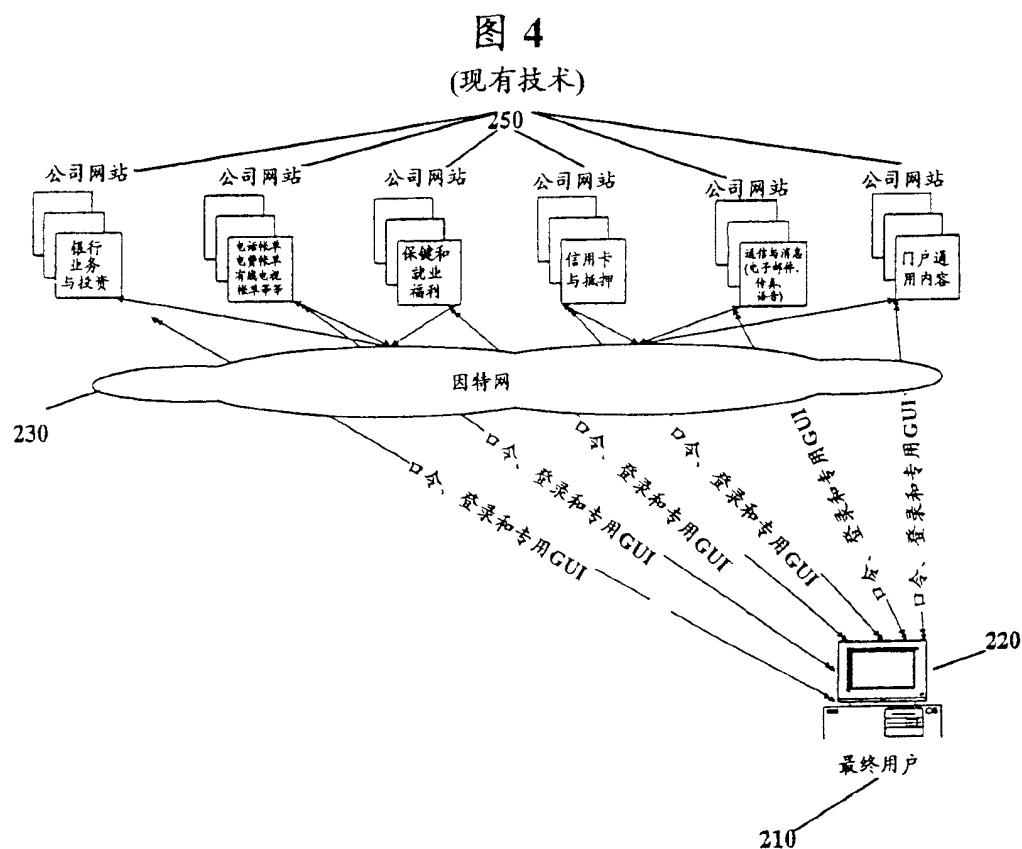


图 3



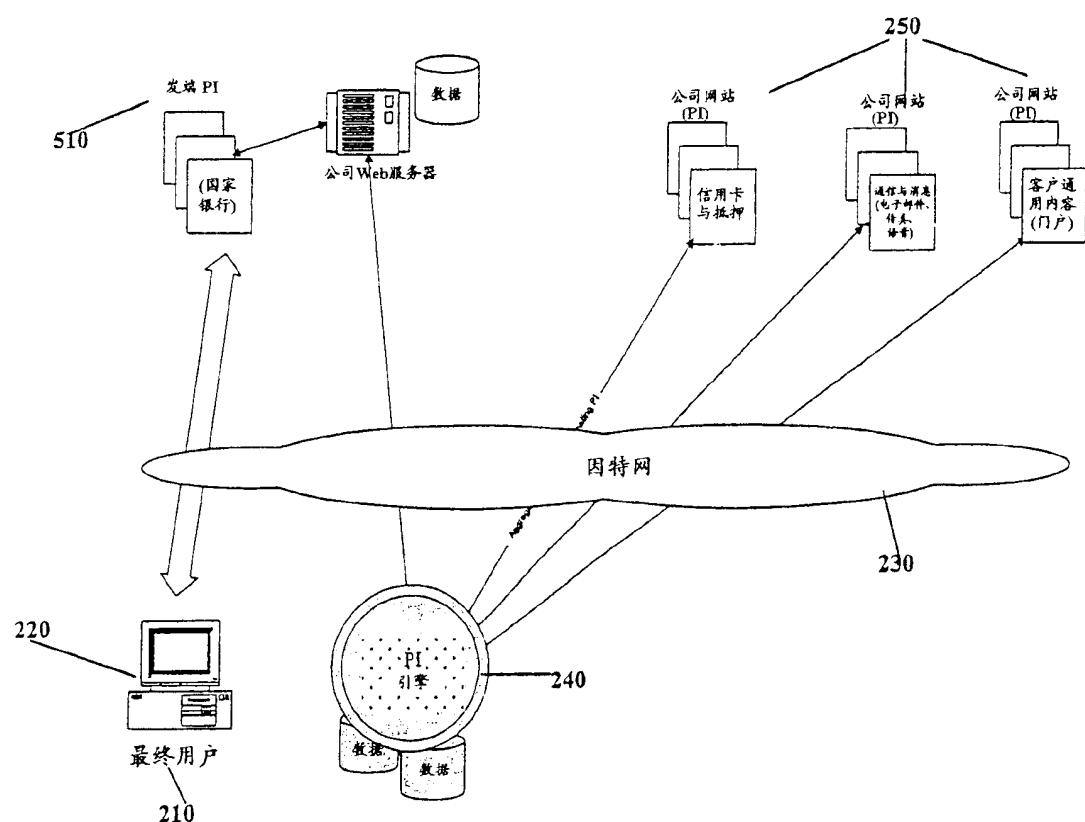


图 5

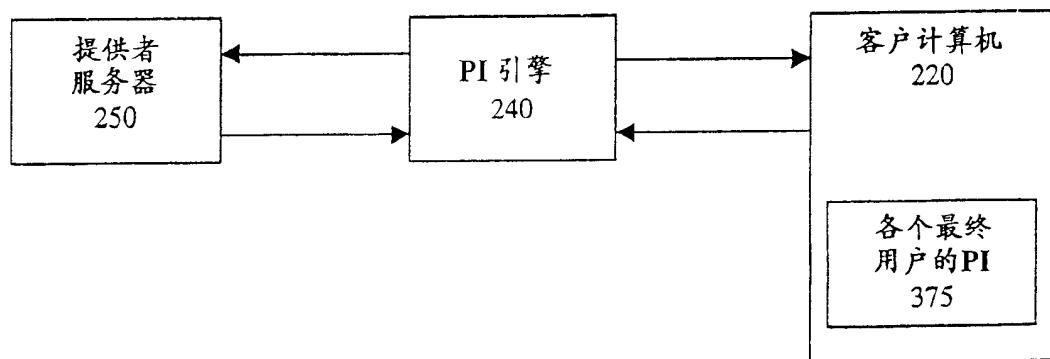


图 6

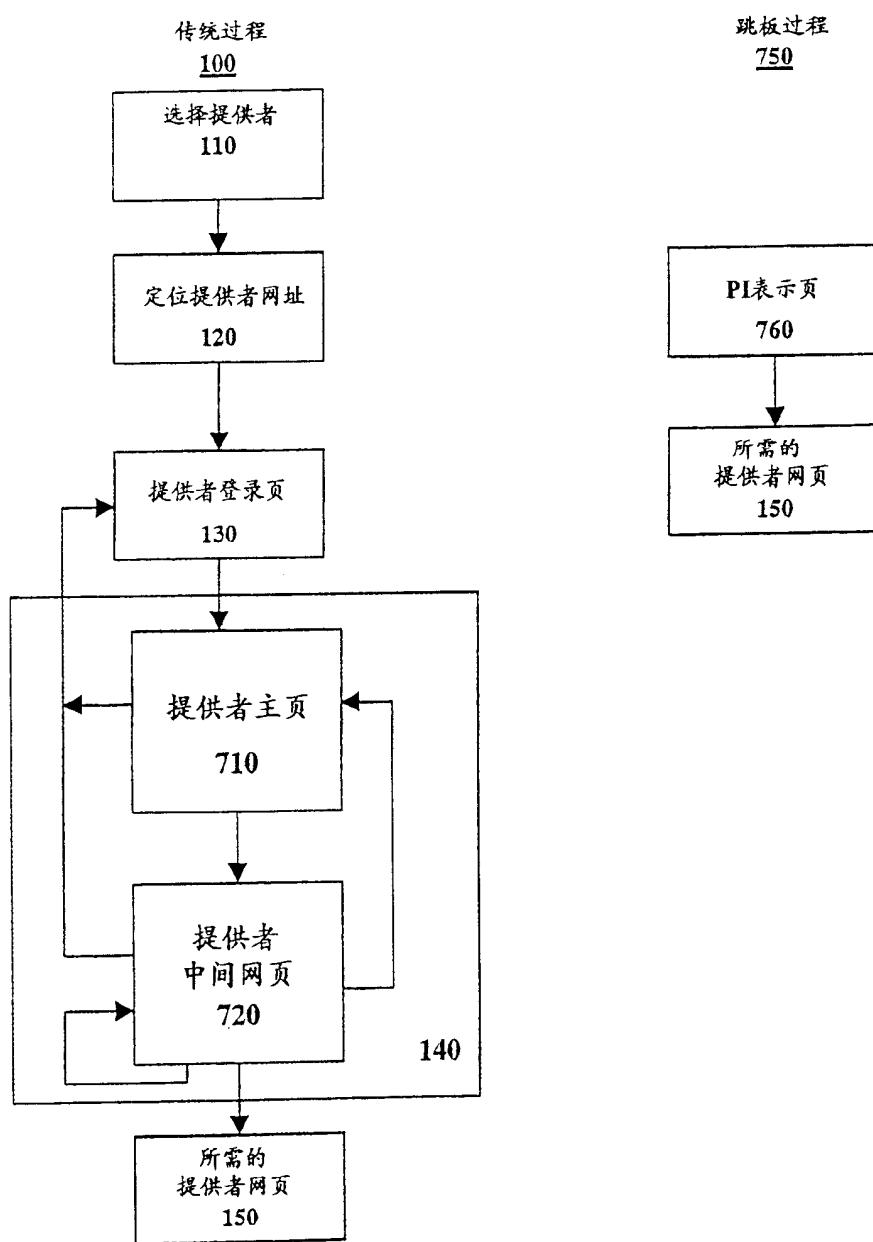


图 7

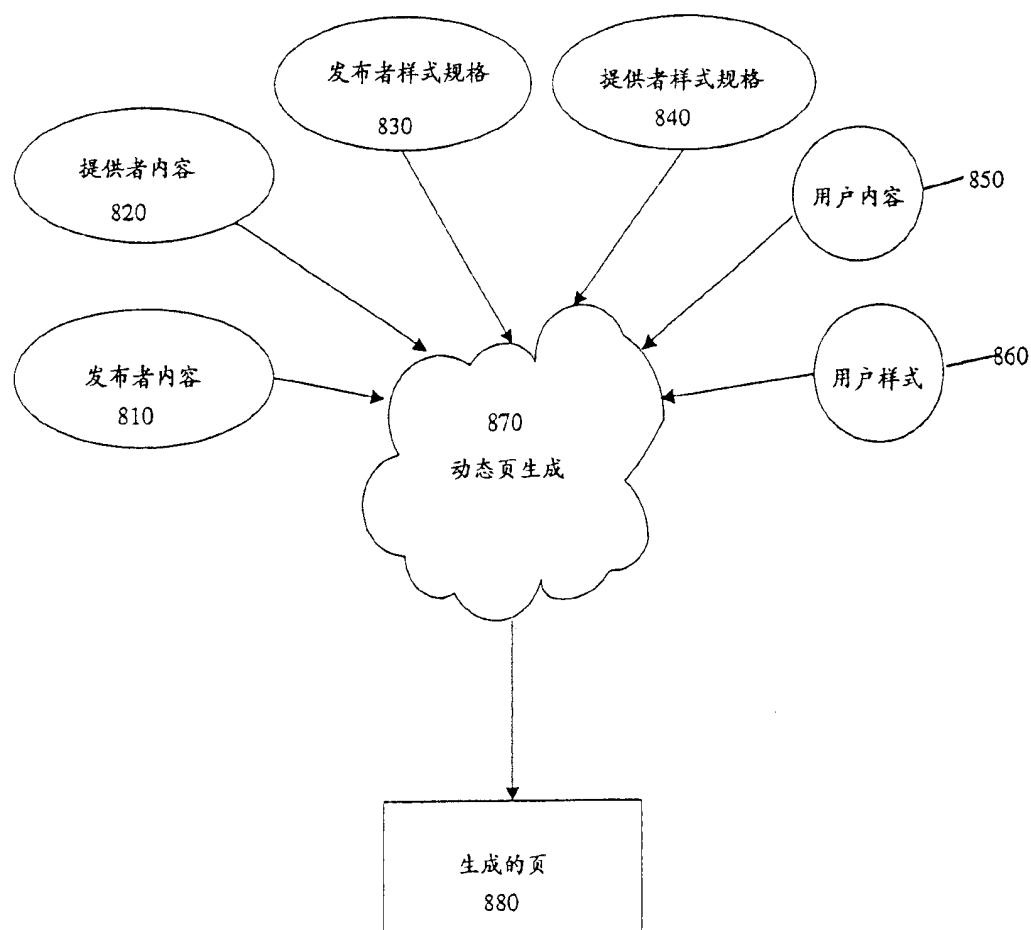


图 8

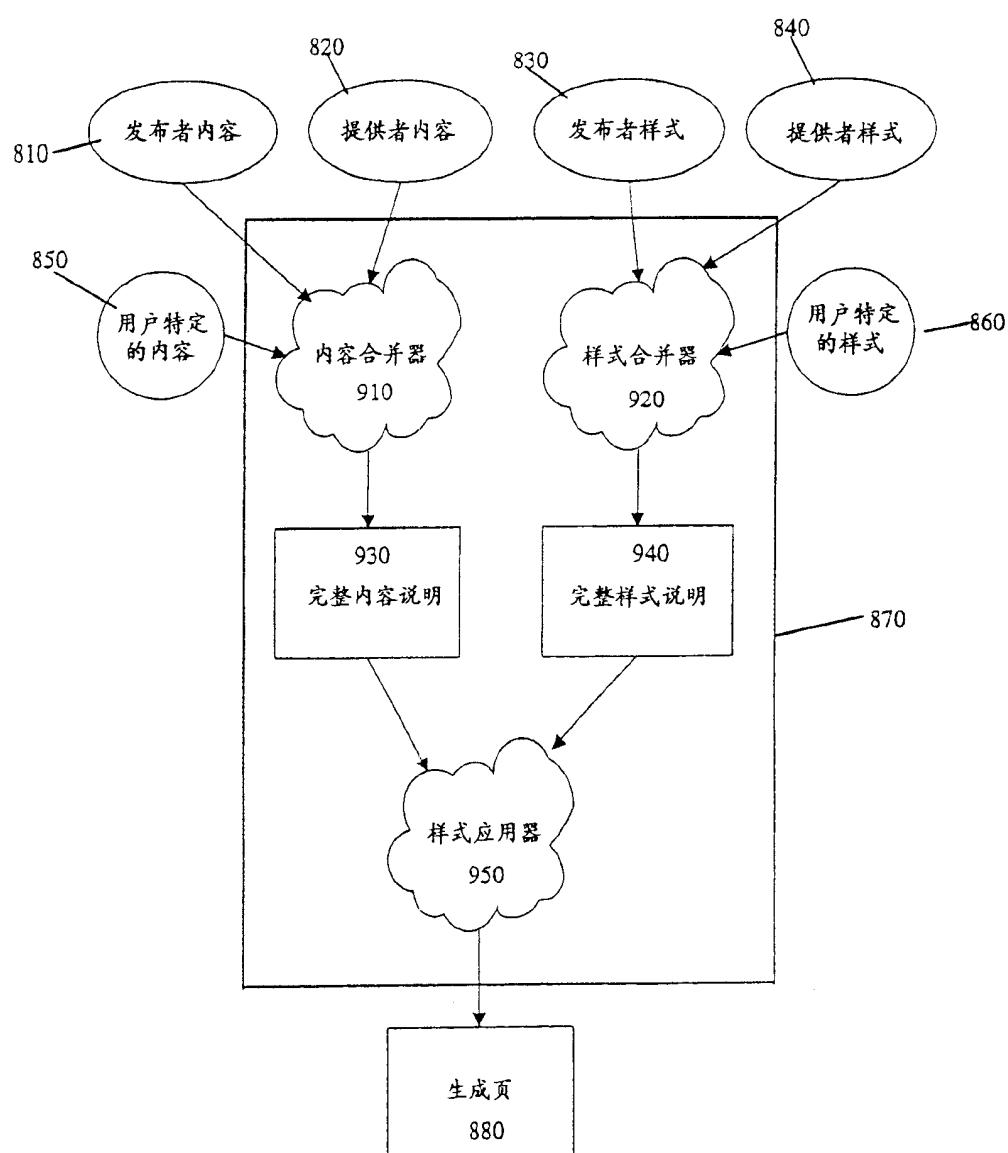


图 9

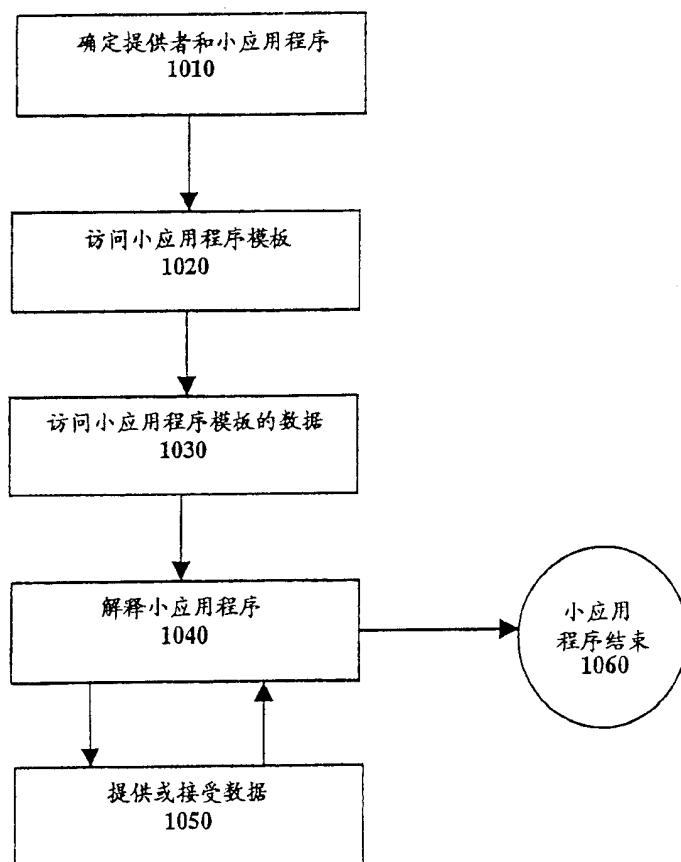


图 10

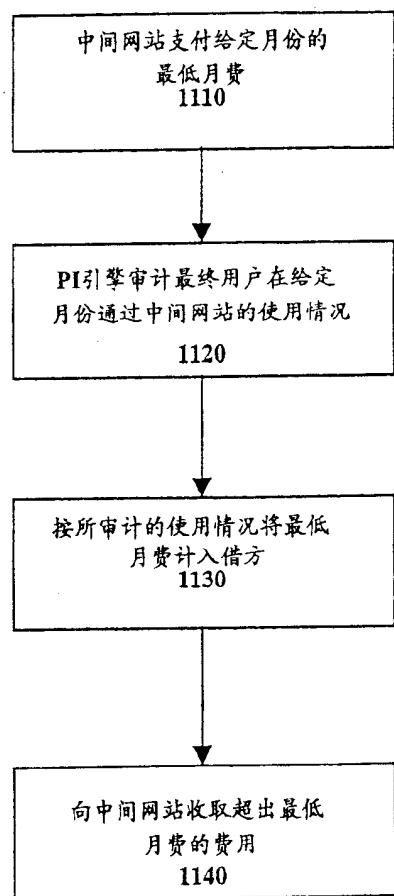


图 11