

[19] 中华人民共和国国家知识产权局

[51] Int. Cl⁷

G06F 15/16

G06F 15/163



[12] 发明专利申请公开说明书

[21] 申请号 02822116.8

[43] 公开日 2005 年 2 月 16 日

[11] 公开号 CN 1582437A

[22] 申请日 2002.10.28 [21] 申请号 02822116.8

[30] 优先权

[32] 2001.11.1 [33] US [31] 10/015,816

[32] 2001.11.16 [33] US [31] 60/331,565

[86] 国际申请 PCT/US2002/034510 2002.10.28

[87] 国际公布 WO2003/038640 英 2003.5.8

[85] 进入国家阶段日期 2004.5.8

[71] 申请人 雅虎公司

地址 美国加利福尼亚州

[72] 发明人 陆 奇 阿希什·巴尔杜亚

[74] 专利代理机构 中原信达知识产权代理有限责
任公司

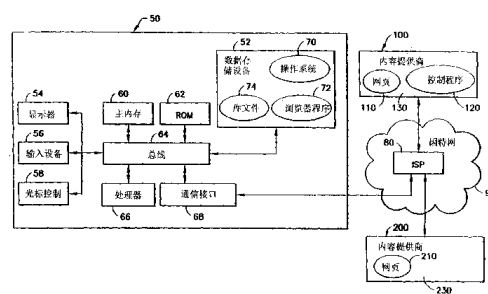
代理人 樊卫民 钟 强

权利要求书 10 页 说明书 52 页 附图 21 页

[54] 发明名称 便利使用因特网浏览器自动登录网
站的方法与系统

[57] 摘要

本发明公开一种方法与系统，其向因特网浏览器界面(20)添加功能。在本发明的一个实施例中，添加的功能可以便利使用因特网浏览器(72)自动登录到网站上。在另一个实施例中，添加的功能可以使得用户使用因特网浏览器(72)执行各种任务，例如，但不限于，执行各种为浏览一个或多个网页，或为从一个或多个网页或网站获取用户想要的信息所必需的任务。



1. 一种便利用户自动登录到网站的方法，其使用因特网浏览器与电脑，所述方法包括这样的步骤，将电脑代码传输到所述电脑，以
5 向所述浏览器增加自动登录功能，所述功能使用用户数据来自动地将所述用户登录到所述网站。

2. 如权利要求 1 所述的方法，其中，所述传输步骤包括传输电脑代码，以添加按钮，所述按钮向所述浏览器添加自动登录功能，使
10 得用户能够访问商户列表并通过从所述商户列表中选择所述商户来自动地登录到所述商户列表上的商户的网站。

3. 如权利要求 2 所述的方法，其中，所述按钮通过下拉菜单提供到所述商户列表的访问。
15

4. 如权利要求 2 所述的方法，其中，当用户从所述商户列表中选择商户时，所述电脑代码便利了到服务器的请求的自动传输。

5. 如权利要求 4 所述的方法，其中，所述请求包括安全 cookie。
20

6. 如权利要求 4 所述的方法，其中，所述请求包括第一请求与第二请求，其包括用户标识符与商户标识符。

7. 如权利要求 6 所述的方法，其中，所述第一请求包括安全
25 cookie。

8. 如权利要求 1 所述的方法，其中，所述电脑代码通过截获促使所述浏览器导航到的每个因特网站点的因特网地址，来监测所述浏览器的因特网导航。
30

9. 如权利要求 2 所述的方法，其中，所述电脑代码判定商户是否是所支持的商户。

5 10. 如权利要求 1 所述的方法，其中，当促使所述浏览器导航到所支持的商户网站上时，所述电脑代码提供指示符。

11. 如权利要求 2 所述的方法，其进一步包括以下步骤：
从用户接收第一请求；
判定所述第一请求是否是安全请求；
10 如果所述第一请求是安全请求，则向所述用户传输响应；和
如果所述第一请求不是安全请求，则向所述用户传输安全登录网页，以供所述因特网浏览器显示。

12. 如权利要求 11 所述的方法，其中，所述判定步骤包括判定
15 所述第一请求是否包含安全 cookie。

13. 如权利要求 11 所述的方法，其进一步包括以下步骤：
接收第二请求，其要求用户证书；和
向所述用户传输所述用户证书。
20

14. 如权利要求 13 所述的方法，其中，所述用户证书被存储在服务器上，并且其中，所述用户证书至少是登录标识符与密码之一，并且其中，所述用户证书特定于商户及所述商户的网站。

25 15. 如权利要求 13 所述的方法，其中，所述第二请求包括用户标识符与商户标识符中的至少一个。

16. 如权利要求 14 所述的方法，其中，所述用户证书包括用于所选商户的商户规则文件。
30

17. 一种因特网浏览器界面，其可由因特网浏览器显示于电脑显示器上，所述浏览器的用户可修改所述浏览器界面，以添加功能，从而使得用户能够使用用户数据自动地将所述用户登录到网站。

5 18. 如权利要求 17 所述的因特网浏览器界面，其中，所述功能由电脑代码提供，所述电脑代码可结合所述电脑的处理器进行操作。

10 19. 如权利要求 18 所述的因特网浏览器界面，其中，所述功能由电脑代码提供，所述电脑代码可结合所述电脑的处理器进行操作，用于向所述浏览器界面添加自动登录按钮，所述按钮向所述浏览器添加自动登录功能，使得用户能够访问商户列表并通过从所述商户列表中选择所述商户来自动地登录到所述商户列表上的商户的网站，并且其中所述按钮提供到所述商户列表的访问。

15 20. 如权利要求 19 所述的因特网浏览器界面，其中，其当用户从所述商户列表中选择商户时，所述电脑代码便利了到服务器的请求的自动传输。

20 21. 如权利要求 20 所述的因特网浏览器界面，其中，所述请求包括安全 cookie。

22. 如权利要求 20 所述的因特网浏览器界面，其中，所述请求包括第一请求与第二请求，其包括用户标识符与商户标识符。

25 23. 如权利要求 22 所述的因特网浏览器界面，其中，所述第一请求包含安全 cookie。

30 24. 如权利要求 18 所述的因特网浏览器界面，其中，所述电脑代码可被进一步地结合所述处理器进行操作，以通过截获促使所述浏览器导航到的每个因特网站点的因特网地址，来监测所述浏览器的因

特网导航。

25. 如权利要求 18 所述的因特网浏览器界面，其中，所述电脑
代码可被进一步地结合所述处理器进行操作，以判定商户网站是否是
所支持的商户网站。

26. 如权利要求 19 所述的因特网浏览器界面，其中，所述电脑
代码可被进一步地结合所述电脑的所述处理器进行操作，用于：
传输第一请求；
传输第二请求，其要求用户证书；和
接收所述用户证书。

27. 如权利要求 26 所述的因特网浏览器界面，其中，所述用户
证书被存储在服务器上，并且其中，所述用户证书至少是登录标识符
与密码之一，并且其中，所述用户证书特定于商户及所述商户的网站。

28. 如权利要求 26 所述的因特网浏览器界面，其中，所述第一
请求包含安全 cookie。

29. 如权利要求 28 所述的因特网浏览器界面，其中，所述第二
请求包括用户标识符与商户标识符中的至少一个。

30. 如权利要求 29 所述的因特网浏览器界面，其中，所述用户
证书包括用于所选商户的商户规则文件。

31. 一种用于便利用户自动登录到网站的系统，其使用因特网浏
览器与电脑，其中，用户数据是登录到所述网站所必需的，所述系统
包括：

服务器，其可连接到因特网，其包括：
数据存储设备；和

处理器，其可结合存储在所述数据存储设备上的软件进行操作，用于向所述用户的电脑传输电脑代码，以向所述浏览器增加自动登录功能，其中，所述电脑代码向所述浏览器增加自动登录功能，其使用用户数据以自动地使所述用户登录到所述网站。

5

32. 如权利要求 31 所述的系统，其中，所述电脑代码向所述浏览器添加按钮，所述按钮向所述浏览器添加自动登录功能，使得用户能够访问商户列表并通过从所述商户列表中选择所述商户来自动地登录到所述商户列表上的商户的网站。

10

33. 如权利要求 32 所述的系统，其中，所述按钮通过下拉菜单提供到所述商户列表的访问。

34. 如权利要求 32 所述的系统，其中，当用户从所述商户列表中选择商户时，所述电脑代码便利了到服务器的请求的自动传输。

15

35. 如权利要求 34 所述的系统，其中，所述请求包括安全 cookie。

36. 如权利要求 35 所述的系统，其中，所述请求包括第一请求与第二请求，其包括用户标识符与商户标识符。

20

37. 如权利要求 36 所述的系统，其中，所述第一请求包括安全 cookie。

38. 如权利要求 34 所述的系统，其中，所述处理器可被进一步地结合软件进行操作，以判定所述请求是否包括安全 cookie。

25

39. 如权利要求 31 所述的系统，其中，所述电脑代码通过截获促使所述浏览器导航到的每个因特网站点的因特网地址，来监测所述浏览器的因特网导航。

30

40. 如权利要求 39 所述的系统，其中，所述电脑代码判定商户网站是否是所支持的商户网站。

5 41. 如权利要求 31 所述的系统，其中，所述服务器是安全服务器。

42. 如权利要求 32 所述的系统，其中，所述处理器可被进一步地结合软件进行操作，用于：

10 从用户接收第一请求；
判定所述第一请求是否是安全请求；
如果所述第一请求是安全请求，则向所述用户传输响应；和
如果所述第一请求不是安全请求，则向所述用户传输安全登录网页，以供所述因特网浏览器显示。

15 43. 如权利要求 42 所述的系统，其中，所述处理器可被进一步地结合软件进行操作，以判定所述第一请求是否包含安全 cookie。

20 44. 如权利要求 42 所述的系统，其中，所述处理器可被进一步地结合软件进行操作，用于：
接收第二请求，其要求用户证书；和
向所述用户传输所述用户证书。

25 45. 如权利要求 44 所述的系统，其中，用户证书被存储在服务器上，并且其中，所述用户证书是登录标识符与密码中的一个或多个，并且其中，所述用户证书特定于商户及所述商户的网站。

30 46. 如权利要求 44 所述的系统，其中，所述第二请求包括用户标识符与商户标识符中的至少一个。

47. 如权利要求 45 所述的系统，其中，所述用户证书包括用于所选商户的商户规则文件。

5 48. 一种控制因特网浏览器界面的方法，所述界面可由因特网浏览器显示在电脑的显示器上，所述因特网浏览器使得所述电脑与因特网浏览器的用户能够访问与浏览因特网，以及接收并在所述电脑显示器上显示来自一个或多个因特网站点的一个或多个网页，其包括显示来自预先确定的因特网站点的网页，所述因特网浏览器具有至少一个因特网浏览器工具条，所述工具条具有至少一个工具条按钮，其向所述电脑与因特网浏览器的所述用户提供预先确定的功能，所述方法包
10 括以下步骤：

(a) 在所述预先确定的因特网站点，提供到程序的访问，用于控制所述因特网浏览器界面；和

(b) 使得所述预先确定的因特网站点能够下载文件，用于将用户工具条作为所述因特网浏览器界面的一部分来显示，所述用户工具条向所述用户开放附加的功能，使得一旦显示所述用户工具条，无论所述因特网浏览器被促使导航到哪个因特网站点，所述用户工具条保持被显示，并且所述附加的功能保持对所述用户开放，其中，所述文件向所述浏览器添加自动登录功能，其使用用户数据，以自动地使所述
15 用户登录到所述网站。
20

49. 如权利要求 48 所述的方法，其中，所述用户工具条包括界面对象，并且可由所述用户定制，以在所述用户工具条内提供用户选择的功能。
25

50. 如权利要求 48 所述的方法，其中，所述文件包括 ActiveX 控件。
30

51. 如权利要求 48 所述的方法，其中，所述文件包括插件。

52. 如权利要求 49 所述的方法，其中，所述界面对象是工具条按钮。

5 53. 如权利要求 49 所述的方法，其中，所述界面对象是搜索窗口，其使得所述用户能够在所述预先确定的因特网站点上发起搜索，无论在发起所述搜索时所述电脑通过所述浏览器连接到哪个因特网站点。

10 54. 如权利要求 48 所述的方法，其中，所述预先确定的因特网站点为多个用户维护用户特定的信息，包括所述电脑与因特网浏览器的用户，所述方法进一步包括这样的步骤：使得所述预先确定的因特网站点能够下载信息，所述信息特定于所述电脑与因特网浏览器的所述用户，其用于定义所述用户工具条的全部或部分，并且其中，所述用户工具条的显示的全部或部分依赖于为所述预先确定的因特网站点
15 所下载、特定于所述电脑与因特网浏览器的所述用户的所述信息。

55. 如权利要求 54 所述的方法，其进一步包括这样的步骤：使得所述预先确定的因特网站点能够下载附加的信息，所述附加的信息特定于所述电脑与因特网浏览器的所述用户，其用于定义所述用户工具条的全部或部分，并且其中，所述用户工具条的显示的全部或部分
20 依赖于所述被下载的信息。

56. 一种因特网浏览器界面，其可由因特网浏览器显示在电脑的显示器上，所述因特网浏览器便利了所述电脑与一个或多个因特网站点之间的连接，所述一个或多个因特网站点包括预先确定的因特网站点，所述因特网浏览器具有至少一个因特网浏览器工具条，其向所述电脑与因特网浏览器的用户开放预先确定的功能，所述因特网浏览器进一步地便利了在所述电脑显示器上显示来自一个或多个因特网站点的一个或多个网页，其包括显示来自所述预先确定的因特网站点的网页，所述因特网浏览器界面包括用户工具条，其在所述因特网浏览器
25
30

被激活时，无论所述电脑通过所述浏览器连接到哪个因特网站点，都作为所述因特网浏览器界面的一部分并与所述因特网浏览器工具条一起来显示，其中，所述用户工具条向所述用户开放附加的功能，作为所述因特网浏览器的一部分，使得一旦显示所述用户工具条，无论所述因特网浏览器被促使导航到哪个因特网站点，所述用户工具条保持被显示，并且所述附加的功能保持对所述用户开放，其中，所述附加的功能向所述浏览器添加自动登录功能，其使用用户数据，以自动地使所述用户登录到所述网站。

57. 如权利要求 56 所述的因特网浏览器界面，其中，所述用户工具条包括界面对象，并且可由所述用户定制。

58. 如权利要求 57 所述的因特网浏览器界面，其中，所述界面对象是工具条按钮。

59. 如权利要求 57 所述的因特网浏览器界面，其中，所述界面对象是搜索窗口，其使得所述用户能够在所述预先确定的因特网站点上发起搜索，无论在发起所述搜索时所述电脑通过所述因特网浏览器连接到哪个因特网站点。

60. 如权利要求 57 所述的因特网浏览器界面，其中，所述界面对象包括 ActiveX 控件，其允许所述界面对象的用户定制。

61. 如权利要求 57 所述的因特网浏览器界面，其中，所述界面对象包括插件控件，其允许所述界面对象的用户定制。

62. 如权利要求 57 所述的因特网浏览器界面，其中，所述预先确定的因特网站点为多个用户维护用户特定的信息，包括所述电脑与因特网浏览器的用户，其中，当所述因特网浏览器被第一次激活时，所述界面对象促使所述因特网浏览器建立到所述预先确定的因特网站

点的连接，并且从所述预先确定的因特网站点接收信息，所述信息特定于所述电脑与因特网浏览器的所述用户，并且其中，所述用户工具条的全部或部分依赖于从所述预先确定的因特网站点接收、特定于所述电脑与因特网浏览器的所述用户的所述信息。

5

63. 如权利要求 62 所述的因特网浏览器界面，其中，当所述因特网浏览器被激活时，所述界面对象促使所述因特网浏览器周期性地重新建立到所述预先确定的因特网站点的连接，以从所述预先确定的因特网站点接收附加的信息，所述附加的信息特定于所述电脑与因特网浏览器的用户，并且其中所述用户工具条的全部或部分依赖于所述被接收的信息与附加的信息。

10

便利使用因特网浏览器自动登录网站的方法与系统

5 相关申请交叉引用

本申请是于 1999 年 10 月 28 日提交的美国专利申请 09/429,585 的部分继续申请 (Continuation-In-Part), 且为于 2001 年 11 月 1 日提交的美国专利申请 10/015,816 的部分继续申请, 它们题为“Method And System Of Facilitating On-Line Shopping Using An Internet Browser (便利使用因特网浏览器在线购物的方法与系统)”, 两者现在均为未决 (pending)。本申请还要求于 2001 年 11 月 16 日提交美国临时专利申请 60/331,565 的优先权益。

技术领域

15 本发明涉及一种方法与系统, 其用于向因特网浏览器界面添加功能。

背景技术

当访问因特网 (也就是万维网、网络等等) 时, 典型地, 因特网用户通过电脑启动或激活浏览器软件程序, 例如, Netscape Navigator™ 或 Microsoft Internet Explorer™。浏览器程序 (这里也称作浏览器) 在用户的电脑与因特网之间建立链路 (例如, 通过调制解调器与因特网服务提供商 (ISP)), 并且提供文本与图形用户界面, 也就是浏览器界面, 其具有预先确定的外观与功能, 两者目前均不能被因特网用户显著地改变。因此, 当因特网用户浏览因特网, 从网站移动到网站, 从应用程序移动到应用程序, 或者从 HTML (超文本标记语言) 页面移动到 HTML 页面时, 浏览器界面保持相对静止。

目前可获得对浏览器界面的有限控制, 其通过可执行软件程序, 其可以, 比方说, 向浏览器界面添加功能按钮。然而, 附加功能在浏

浏览器被初始激活时添加到浏览器界面，并且在其后保持静止。因此，显示如刚才所述被修改的浏览器界面的浏览器不可能在浏览器被初始化或用户浏览因特网时，动态地从因特网站点下载信息并定制。这样的被修改的浏览器界面也不向插件与界面提供对各种浏览器应用编程接口（API）的访问。因此，存在这样的需求：提供方法，用于修改浏览器界面，并且提供浏览器界面，其克服了上面所描述的现有技术的缺点。

当访问特定网站时，在被允许通过特定于该用户的网站访问数据之前，用户可能必须输入用户标识符与密码。例如，大多数金融机构，投资公司，以及其它服务提供实体，包括，但不限于在线购物网站，允许用户（或该实体的客户）通过因特网访问他的或她的账号。出于显而易见的原因，在获得对该用户的账号的访问之前，要求该用户输入某些用户特定的信息。例如，为建立在线账号，用户可能被要求提供某些用户特定的信息，例如，用户标识符与密码。一旦提供该用户特定的信息，建立在线账号，用户可能只能访问他的或她的在线账号，其提供用户特定的信息，作为登录过程的一部分。

对于用户而言并非不常见的是，维护多个在线账号（涉及多个金融机构，投资公司，以及其它服务提供实体），其可通过多个网站访问。虽然可以想见的是，对于多个网站中的每一个，用户可使用同样的登录标识符与密码，这样的实践是不被推荐的。事实上，更可取、被推荐的实践是，对于每个在线账号使用不同的登录标识符与密码。如果某人碰巧获得用户的登录标识符与密码中的一个的话，这将增加用户的在线账号的安全性，减小对这些账号的未授权访问的可能性。然而，对于每个在线账号使用不同的登录标识符与密码存在自己的问题；其中并非无关紧要的是，记住每个登录标识符与密码。

用户也存在这样的需求，从一个或多个网站获取信息，而不必导航到每个网站。进一步地，存在这样的需求，简单地、快速地使用因

特网浏览器，以获得这样的信息。

发明内容

5 本发明涉及一种方法与系统，其向因特网浏览器界面添加功能。在本发明的一个实施例中，添加的功能可便利使用因特网浏览器自动登录到网站。在另一个实施例中，添加的功能可允许用户使用因特网浏览器执行各种任务，例如，但不限于，执行为浏览一个或多个网页，或为从一个或多个网页或网站获取用户想要的信息所必需的各种任务。

10

遵照本发明的实施例，电脑代码从服务器传输到用户的电脑，其向用户的因特网浏览器添加特定的功能。这样，本发明提供对因特网浏览器界面的动态控制，其允许用户选择性地向浏览器界面添加按钮和/或工具条，并向浏览器添加特性与功能。在优选实施例中，电脑代码允许用户修改浏览器界面，其选择性地向因特网浏览器界面添加界面对象。界面对象可包括按钮，其可被添加到浏览器界面，作为现存工具条的一部分，或者作为添加到浏览器界面的新工具条的一部分。该按钮允许用户访问商户列表，并且也向浏览器增加了自动登录功能。对商户列表的访问允许用户添加商户，编辑商户，和选择商户，其用于自动登录。一旦该按钮被添加到浏览器界面，用户可点击添加的按钮，这里一般地称为自动登录按钮，并且取得对下拉菜单的访问，其显示一个或多个商户，作为商户列表。通过选择商户列表上的商户之一，遵照本发明添加到浏览器上的功能将促使用户自动登录到所选商户，并促使用户的浏览器导航到该商户的网站。该过程对用户本质上是透明的，当用户从商户列表中选择商户时启动。

15

20

25

优选地，遵照本发明提供的自动登录功能使用安全登录过程，其要求用户在对所选商户的自动登录过程完成之前输入安全钥匙。如果在特定因特网会话期间用户尚未输入安全钥匙，或者如果在会话期间用户输入了安全钥匙，但自输入安全钥匙时起已过去预先确定量的时

30

间，将提示用户输入安全钥匙，作为自动登录过程的一部分。仅在输入正确的安全钥匙后，才会进行自动登录过程。如上所述的安全登录过程可由安全服务器或用户的电脑执行。

5 如果在因特网会话期间用户已安全地登录到安全服务器，可从该安全服务器传输安全 cookie，并存储到用户的电脑。如果安全 cookie 存在于用户的电脑上，并且有效（也就是，比方说，其尚未过期），当用户从商户列表上选择商户时，电脑代码将该安全 cookie 传输到安全服务器，作为第一请求的一部分。安全服务器接收该请求并判定其是否包含安全 cookie，该安全 cookie 是否有效。如果第一请求包含有效的安全 cookie，安全服务器将确认传输到电脑代码，依次地，其传输第二请求到供给服务器（其可与安全服务器相同或不同，这取决于设计选择）。第二请求包含用户标识符与商户标识符。当供给服务器接收到第二请求时，其将用户标识符和商户标识符与数据库条目进行比较，以判定是否存在匹配。如果找到匹配，供给服务器将证书请求传输到另一服务器，其可为安全服务器。所述另一服务器包含安全数据库，其具有多个用户的多个数据。每个用户的数据通过用户标识符标识，并且可包含登录标识符与密码，其用于多个商户中的每一个，多个商户中的每一个具有相关联的商户规则文件。每个数据库条目包含登录标识符、密码与商户规则文件，该条目这里也可称为证书。一旦确定与用户选择的商户相关联的证书，所述另一服务器将该证书传输到用户的电脑。一旦接收到证书，遵照本发明提供的电脑代码促使用户的浏览器导航到所选商户的预先确定的网页，例如登录网页，并且自动地填写该网页上为用户登录并访问该用户的账号所必需的任何域。此外，该过程对用户是透明的。如果自动登录成功，商户网页（其典型地跟随成功的登录）将通过商户的服务器传输，并显示在用户的浏览器窗口。

10

15

20

25

因此，通过从商户列表选择商户，用户能够自动地导航和登录到该商户的网页，并访问该用户的账号。

30

如这里所使用的，术语“控制”与“可控的”是指，比如，但不限于，添加、去除和修改因特网浏览器界面。因特网浏览器界面，如这里所称的，是指呈现给浏览器用户的可视的或可听的表达，用户通过其
5 与其与浏览器交互。术语“定制”（以及其变形）也可使用在这里，以描述遵照本发明提供的可控性。如这里所使用的，术语“动态控制”（以及其变形）是指一种方法，通过这种方法，因特网浏览器界面的一部分（也就是界面对象）可被选择性地添加，显示和周期性地改变或更新。

10

浏览器，如这里所使用的，被给予其通常的、业界承认的含义，一般指软件程序，其提供到万维网或因特网的界面。浏览器允许因特网用户浏览因特网，建立到各个因特网站点（这里也称作网站）的连接，阅览内容提供商在各个因特网站点提供的网页。特别地，浏览器有能力通过因特网站点调用 ActiveX 控件或插件。浏览器也允许因特网
15 用户在因特网站点之间导航，也就是网上冲浪。浏览器向因特网用户提供浏览器界面，其格式由浏览器软件程序的提供商确定。典型地，浏览器界面定义浏览器提供给因特网用户的布局（例如颜色、大小、数量与位置）与功能。浏览器界面可包括第一父窗口，典型地，其定义浏览器界面的一般大小、颜色与布局，并包含窗口控制按钮（例如，
20 最小化、关闭等等）。浏览器界面还可包括第二父窗口（其可为第一父窗口的子窗口），以及一个或多个依赖于第二父窗口的窗口。第二父窗口及依赖于其的窗口可向因特网用户提供，比如，各种信息（广告、优待券、新闻、HTML 链接等等）与功能（工具条、下拉菜单、
25 插件、应用程序等等）。

ActiveX 控件，如这里所使用的，指一种工具，其用于将桌面应用程序链接到因特网，其基于业界承认、Microsoft 开发的规范。插件，如这里所使用的，指一种类型的程序，其被开发以用于 Netscape 浏览器，并与更大的应用程序（例如浏览器软件程序）集成，以向该更大
30

的应用程序添加特定功能或能力。如上所述的、这里所称的 **ActiveX** 控件与插件可与任何因特网浏览器一起使用。

5 如这里所使用的，术语“因特网站点（或网站）”是指因特网上的地点，其为因特网地址或 URL（统一或通用资源定位符）所定义。如这里所使用的，术语“因特网网页”是指在因特网站上提供的超文本标记语言（HTML）命令的集合，其提供文本、图像与功能的格式信息，以创建超文本文档，其可通过浏览器在因特网用户的电脑显示器上显示。例如，因特网用户输入 URL，以建立到因特网站点的连接，该因特网站点向用户的浏览器提供 HTML 命令，使得该因特网站点的网页可以显示在用户的电脑显示器上。浏览器解释内嵌于网页内的 HTML 命令，并使用 HTML 命令来为该网页格式化文本与图像。

15 本发明向因特网用户、因特网内容提供商与因特网服务提供商（ISP）提供了有利之处。对于因特网用户，本发明提供一种方法，其动态地控制或定制该用户的因特网浏览器界面。因特网用户现在可以定制浏览器界面，使得每次用户使用浏览器访问因特网时，用户定义的信息和/或功能（这里统称为信息）将显示于浏览器界面。例如，用户可包括书签、地址簿与电话簿、个人金融信息、个性化新闻以及各种功能，例如可通过 **ActiveX** 控件与插件获得。

25 另外，如果因特网用户有内容提供商的账号，浏览器现在可以在浏览器界面上动态地显示该用户的特定账号信息。目前，因特网用户只能在连接到该内容提供商的因特网站点时访问该用户的特定账号信息。用户必须返回该内容提供商的站点，以接收更新的帐户信息。一旦用户离开该因特网站点，浏览器界面上不再显示账号信息。本发明提供动态控制的方法，以及可动态地控制的浏览器界面，其允许因特网用户在浏览器界面上显示并持续地更新特定于该用户的信息和/或功能。

30

对于内容提供商，本发明保证每次用户使用浏览器访问因特网时，该因特网用户（通过浏览器）自动地建立到该内容提供商的因特网站点的连接。因此，一旦因特网用户促使浏览器执行（例如通过选择浏览器图标），浏览器就自动地建立到内容提供商的因特网站点的连接，以加载该用户的定制的浏览器界面信息。本发明也可在浏览器活动时和用户网上冲浪时，周期性地、自动地促使用户的浏览器连接到内容提供商的因特网站点。在一个实施例中，内容提供商可向因特网用户提供到程序的访问，该程序用于控制浏览器界面。一旦因特网用户访问该控制程序以定制该用户的浏览器界面，每次用户访问因特网时，用户的浏览器可自动地建立到该内容提供商的连接。因此，与目前可获得的浏览器（其仅当用户输入 URL（或其它实际地行动，以促使连接建立，比方说，通过选择链接或广告条）时建立到因特网站点的连接）相反，本发明在浏览器初始化时自动地建立到内容提供商的连接，其独立于预编程到浏览器的任何主页选择，无论这是因特网用户还是浏览器销售商的选择。一旦该初始连接建立，内容提供商可向因特网用户的电脑装载用户特定的信息和/或功能，以显示到该用户的浏览器界面。

另外，内容提供商也可周期性地促使浏览器自动地重连到该内容提供商的因特网站点，以更新、下载新的信息和/或功能、或以其它方式传输信息和/或功能，用于因特网用户的浏览器界面。例如，如果因特网用户订阅了内容提供商的电子邮件服务，该内容提供商接收的该因特网用户的电子邮件消息可自动地传输给该因特网用户，即使该用户正在其它地方“冲浪”。当用户的浏览器执行时，用户的浏览器初始地建立到内容提供商的因特网站点的连接，该内容提供商传输给该因特网用户的信息包括对该浏览器的指令，以周期性地重连到该内容提供商的因特网站点。因此，无论用户访问的因特网站点的数量，也无论用户当前正访问哪个特定因特网站点，回到内容提供商的因特网站点的连接将以内容提供商确定的间隔自动地建立；这些重连对于因特网用户而言是透明的，但用户从内容提供商接收通知（也就是新邮

件到达)时除外。因此,在因特网用户网上冲浪时,浏览器界面可被动态地控制。对于ISP,好处至少与内容提供商一样大。

5 初始地,因特网用户的浏览器界面必须使用软件程序定制,其可由内容提供商或ISP提供,或可在用户的电脑上获得。该软件程序,此后称之为电脑代码、控制程序或用于控制的程序(以及其变形),改变用户的浏览器工作的方式。更明确地,控制程序在因特网用户的电脑上下载或创建库文件。库文件可以是,比如说,动态链接库(DLL)(用于Windows操作系统),其在浏览器内创建一个外壳(shell)(或多个外壳),各种信息和/或功能可作为ActiveX控件或插件装载到其中。库文件包含ActiveX控件或插件功能,其遵照本发明定义添加到浏览器界面的界面对象。当因特网用户启动或激活浏览器,库文件被打开,包含于其内的ActiveX控件或插件代码可为浏览器得到,并集成到浏览器界面,从而促使界面对象显示到浏览器界面上。只要浏览器激活,一般地只要用户网上冲浪,库文件,并且随之地,外壳(或多个外壳),就保持打开。因此,甚至在用户从因特网站点移动到因特网站点时,用于定制浏览器界面并装载到外壳的信息和/或功能保持活动。当在这里的上下文内使用时,术语“信息和功能”是指任何信息、数据和/或软件驱动的功能,其可为库文件所包含,或者是库文件的一部分。

10 20

当用户初始地激活浏览器时,库文件也促使浏览器建立到内容提供商的因特网站点的连接。内容提供商的因特网站点将会把用于界面对象的信息和/或功能装载到用户的电脑,以用于浏览器,并显示到浏览器界面。如果,比方说,用户有内容提供商的账号,内容提供商装载的信息和/或功能可以是特定于因特网用户的。作为另一种可供选择的方案,如果,比方说,用户没有内容提供商的账号(也就是,其为客人(guest)),内容提供商可以装载一般的信息和/或功能。

25

30 本发明使用对象链接与嵌入(OLE)进程内(in-process)服务器,

以控制浏览器界面的信息和/或功能。使用包含于库文件之内的 ActiveX 控件或浏览器插件（每一个在这里被称为浏览器界面覆盖（BIO）库），事实上 ActiveX 控件或插件可获得的任何信息和/或功能均可使用本发明被添加到浏览器界面。库文件（通过 BIO 库）从而包含了为定制，
5 也就是添加、删除和/或修改浏览器界面所必需的代码。

一旦因特网用户访问控制程序，并定制该用户的浏览器的操作，当浏览器被激活时，显示被定制的浏览器界面。与现有技术 的浏览器修改方法相反，本发明提供一种方法与浏览器界面，其可被动态地控制。当因特网用户网上冲浪时，更新的或变化的信息和/或功能可传输
10 到浏览器并显示在浏览器界面上，同时维持浏览器界面被定制的信息和/或功能。因此，只要用户正在使用浏览器网上冲浪，在该用户的浏览器界面上，在任何因特网站点上，因特网用户可自动地接收最新的信息，比如说，股票行情、电子邮件、新的头条新闻。

15 本发明也提供一种方法与系统，其便利了网上购物，其修改浏览器界面，以提供购物助理按钮与功能。遵照本发明的本实施例提供的被修改的浏览器界面简化了在线购物，其为每个购物者或用户创建钱包。钱包存储于安全服务器上的数据库内，仅可被该用户访问、使用、
20 或修改，且仅与用户提供的安全钥匙相联系。钱包包含用户提供的信息，其可被直接移植（port）到所支持的商户结账网页。因此，当用户正在所支持的商户的网站上在线购物，并且想从该商户购买特定商品和/或服务时，本发明向该用户提供到他的/她的钱包的访问途径，并便利了钱包内包含的数据的移植，以快速地、不费力地填写该商户
25 的结账网页上提供的该商户的订单表格。

在本发明的又一个实施例中，可向因特网浏览器添加自动登录功能。向浏览器界面添加自动登录界面对象的步骤与上面所描述的关于本发明的各种其它的实施例的步骤相同。因此，不必为本实施例重复
30 这些步骤。自动登录界面对象向浏览器添加各种功能，例如，但不限

于，对商户列表的访问和从列表中修订、添加和/或去除商户的能力，以及自动地登录到所选商户网站的能力。需要注意的是，尽管“商户站点”在这里被联系本实施例指代，本发明适用于用户可在其上建立在线账号的任何网站，其可要求用户输入登录账号、密码和/或其它用户数据，以通过该网站访问该用户的账号。

从后面的详细描述，联系所附的绘图来考虑，本发明的其它目标与特性将变得显而易见。然而，需要理解的是，这些绘图仅为阐述的目的而设计，而非作为对本发明的界限的定义，本发明的界限应参照所附的权利要求书。

附图说明

附图仅为阐述性的，且其中所有视图中相似的参考字符表示相似的组件：

图 1 描述现有技术 的因特网浏览器界面；

图 2 描述遵照本发明的一个实施例的因特网浏览器界面的视图，其包括浏览器工具条内的界面对象；

图 3 描述遵照本发明的一个实施例的因特网浏览器界面的视图，其具有界面工具条，其包括界面对象；

图 4 描述遵照本发明的一个实施例的因特网浏览器界面的视图，其具有界面工具条，其包括多个界面对象；

图 5 是一种方法的流程图，其遵照本发明的一个实施例控制因特网浏览器界面；

图 6-9 是一种方法的流程图，其遵照本发明的各个实施例控制和显示因特网浏览器界面；

图 10 是一种电脑的示意框图，其连接到因特网上，本发明可在其上实现；

图 11 是遵照本发明的一个实施例的因特网浏览器界面的示例性的屏幕截图，其具有购物助理按钮；

图 12 是遵照本发明的一个实施例的钱包建立网页的示例性的屏

幕截图；

图 13 是遵照本发明的一个实施例的钱包数据条目网页的示例性的屏幕截图；

5 图 14 是遵照本发明的一个实施例的钱包小结网页的示例性的屏幕截图；

图 15 是遵照本发明的一个实施例的安全钥匙窗口的示例性的屏幕截图；

图 16 是遵照本发明的一个实施例的钱包信息窗口的示例性的屏幕截图；

10 图 17 是遵照本发明的一个实施例的所支持的商户结账网页的示例性的屏幕截图；

图 18A-18B 是一种方法的流程图，其遵照本发明的一个实施例自动地将用户登录到商户网站上；和

15 图 19 是一种系统的示意图，其用于遵照本发明的一个实施例便利利用户自动登录到商户网站上。

具体实施方式

本发明涉及一种方法与系统，其便利了使用因特网浏览器自动登录到网站上。遵照本发明的一个实施例，电脑代码从服务器传输到用户的电脑，其向用户的因特网浏览器添加特定的功能。本发明从而提供了对因特网浏览器界面的动态控制，其使得用户能够选择性地向浏览器界面添加按钮和/或工具条，并且向浏览器添加特性与功能。在优选实施例中，电脑代码使得用户能够修改浏览器界面，其选择性地向因特网浏览器界面添加界面对象。界面对象可包括按钮，其可被添加到浏览器界面，作为现存的工具条的一部分，或者作为添加到浏览器界面的新的工具条的一部分。该按钮允许用户访问商户列表，并且也向浏览器增加了自动登录功能。对商户列表的访问允许用户添加商户，编辑商户，和选择商户，用于自动登录。一旦该按钮被添加到浏览器界面，用户可点击添加的按钮，这里一般地称为自动登录按钮，并且取得对下拉菜单的访问，下拉菜单显示一个或多个商户，作为商

20

25

30

户列表。通过选择商户列表上的商户之一，遵照本发明添加到浏览器上的功能将促使用户自动登录到所选商户，并促使用户的浏览器导航到该商户的网站。该过程对用户本质上是透明的，当用户从商户列表中选择商户时启动。

5

本发明也提供一种方法与系统，其便利了网上购物，其修改浏览器界面，以提供购物助理按钮与功能。遵照本发明的本实施例提供的被修改的浏览器界面简化了在线购物，其为每个购物者或用户创建钱包。钱包存储于安全服务器上的数据库内，仅可被该用户访问、使用或修改，且仅与用户提供的安全钥匙相联系。钱包包含用户提供的信息，其可被直接移植到所支持的商户结账网页。因此，当用户正在所支持的商户的网站上在线购物，并且想从该商户购买特定商品和/或服务时，本发明向该用户提供到他的/她的钱包的访问途经，并便利了钱包内包含的数据的移植，以快速地、不费力地填写该商户的结账网页上提供的该商户的订单表格。

10

15

现在详细地参照附图，图 10 是电脑 50 的框图，电脑 50 连接到因特网 90，本发明可于其上实现。电脑 50 包含内部总线 64，其便利了电脑 50 的各个设备之间的信息通信，也便利了电脑通过通信接口 68 与外部设备和系统的通信。处理器 66 连接到总线 64，在电脑 50 内处理信息。电脑 50 也包括主内存 60，例如，随机存取内存（RAM）或其它等价的动态内存存储设备，其连接到总线 64，用于接收和存储从处理器 66 上传输的指令。主内存 60 还可被用于在处理器 66 执行指令时暂时地存储变量或其它中间信息。只读内存（ROM）62 也连接到总线 64，用于存储静态数据与指令，其为处理器 66 使用。作为电脑 50 的一部分，提供各种输入与输出设备，包括，但不限于，显示器 54（例如阴极射线管（CRT）、液晶显示器（LCD）等等），输入设备 56，例如键盘，以及光标控制设备 58，例如鼠标或跟踪球。数据存储设备 52，例如，磁盘驱动器与磁盘、CD-ROM 驱动器与 CD-ROM 或其它等价的设备与数据存储介质，连接到总线 64，以与处理器 66、

20

25

30

主内存 60 和通信接口 68 通信。优选地，存储设备 52 上存储有操作系统 70 和因特网浏览器软件程序 72。如下面将更详细地讨论的，库文件 74 也可存储于数据存储设备 52 上。如这里所使用的，术语“电脑”意欲取其最广泛的涵义，而非受限于任何物理大小或构造，或任何体系结构配置。更确切地，术语“电脑”，如这里所使用的，意欲包含有能力发送和/或接收数字数据，存储和/或获取数字数据，处理数字数据，并且可通过任何方式、使用任何手段（例如因特网）来连接到网络的任何设备。电脑可以是任何样式、大小与配置的电脑，其包括，但不限于，服务器、工作站、桌面电脑、膝上电脑、因特网设备、笔记本电脑、个人数字助理（PDA）、因特网使能蜂窝电话或其他现在已知或此后开发的设备。电脑包括如下部件中的一些或全部：中央处理单元（CPU 或处理器），其可结合软件（例如，操作系统、应用程序等等）来操作，硬盘单元（HDD），永久内存（例如，ROM），临时内存（例如，RAM），抽取式数据存储设备（例如，CD-ROM、软盘驱动器等等），输入设备（例如，键盘、鼠标、跟踪球等等）输出设备（例如，监视器或显示器），以及 I/O 设备（例如，调制解调器、红外发送器/接收器、无线（蜂窝式）发送器接收器等等）。本领域技术人员知道，除未列出的部件之外，电脑可包含这些部件中的一些或全部。

20

电脑 50 可通过基于传输介质之上的通信接口 68 连接到因特网 90，该传输介质包括，但不限于，同轴电缆、铜线与光纤电缆。电脑 50 与因特网 90 之间的通信也可通过无线或蜂窝式接口。通信接口 68 便利了电脑 50 与另一电子设备或系统（例如内容提供商 100、200 提供的服务器电脑（未显示））之间的双向通信。

25

使用电脑 50 的因特网用户（未显示）可取得对因特网 90 的访问，其促使浏览器 72 执行，从而通过因特网服务提供商（ISP）80 在电脑 50 的通信接口 68 与内容提供商 100 的因特网站点 130 之间打开通信链路。浏览器 72 向电脑显示器 54 提供浏览器界面 20（参见，比如

30

说，图 1)，其具有窗口 30 的布局（例如，颜色、大小、数量、位置）与功能，其为浏览器销售商在浏览器 72 内预先确定。内容提供商 100 将因特网内容传输到电脑 50，以显示到浏览器接口 20 的内容窗口 32 中。

5

遵照本发明的一个实施例，第一互联网内容提供商 100 可向因特网用户提供对程序 120 的访问，以控制浏览器 72 与浏览器界面 20。当被用户执行时，控制程序 120 在因特网用户的电脑 50 的数据存储设备 52 上下载或创建库文件 74，例如，动态链接库（DLL）。优选地，库文件 74 包含 ActiveX 控件或插件功能。其后，当因特网用户使用浏览器 72 访问因特网时，浏览器 72 打开库文件 74，并且，优选地，自动地建立到内容提供商的因特网站点 130 的连接。作为对浏览器 72 建立的连接的响应，内容提供商将信息和/或功能数据装载到外壳，该外壳操作在浏览器之内，由库文件 74 创建。例如，如果用户具有内容提供商 100 的账号，可装载定制的信息和/或功能到库文件 74。如果用户不具有账号，可装载更一般的信息和/或功能。

10

15

20

遵照本发明的各种实施例，如这里更详细地描述的，数据可在电脑之间（例如，在服务器与用户或客户端电脑之间）传输。本发明预期这样的数据传输（或发送/接收、下载/上载等等）可以以任何方式，使用任何介质，使用任何现在已知或今后开发的介质或手段来实现。

25

30

基本地，库文件 74 在浏览器 72 之内打开外壳（或多个外壳），其包含 ActiveX 控件或插件代码，可控制因特网浏览器 72 与浏览器界面 20。当装载 ActiveX 控件或插件后，优选地，库文件 74 包含函数、对象、数据与其它软件，这里一般地称之为信息，其可用于控制浏览器 72 与浏览器界面 20。当因特网用户从因特网站点 130 移动到因特网站点 230 时，本发明确保库文件 74（与外壳）不关闭。因此，当因特网用户从装载了 ActiveX 控件或插件的因特网站点断开连接，并连接到另一因特网站点时，通过 ActiveX 控件或插件提供的信息和/或功

能没有丢失。

接着参照图 1，描述了现有技术 的因特网浏览器界面 20，其具有多个窗口，每个窗口向因特网用户提供各种功能。浏览器界面 20 可包括第一父窗口 30，典型地，其定义浏览器界面的一般大小、颜色与布局，并包含用于该窗口 30 的窗口控制按钮 28（例如，最小化、关闭等等）。浏览器界面 20 还可在第一父窗口 30 之内包括第二父窗口 36（第一父窗口的子窗口），以及一个或多个子窗口 38，其依赖于第二父窗口。典型地，第二父窗口 36 及子窗口 38 定义信息和/或功能，当访问与浏览因特网时，其将协助因特网用户。例如，第二父窗口 36 及依赖于其的窗口 38 可提供工具条、下拉菜单、插件、应用程序等等。

例如，在界面 20 的顶部（在图中）提供的三个窗口 38 定义了三个工具条 22，其可包括多个界面控件 24，例如，下拉菜单、功能按钮（例如，停止、后退、前进、主页等等）以及功能按钮与窗口的组合（例如，搜索按钮与窗口）。最上面的工具条 22 提供多个下拉菜单 24；中间的工具条 22 提供多个功能按钮 24；而最下面的工具条 22 提供下拉菜单与窗口 26（URL 地址窗口）。内容窗口 32 也作为界面 20 的一部分提供，来自因特网内容提供商 100（参看，例如，图 10）的内容可在窗口内显示。因特网用户可开启和关闭较低的两个（在图上）工具条 22 中的任何一个，其使用第二工具条 22 内提供的查看（View）工具条对象 24（下拉菜单）。然而，因特网用户目前不可以添加、删除或修改浏览器界面 20。

25

图 2-4 描述了遵照本发明的各种实施例配置的因特网浏览器 20。在图 2 中，浏览器界面 20 包括界面对象 40，其为 ActiveX 控件或插件定义，该 ActiveX 控件或插件由内容提供商 100 装载到库文件 74。界面对象 40 包括下拉菜单 44，并与浏览器 72 提供的界面控件（也就是浏览器工具条对象）24 一起显示在浏览器工具条 22 内。在图 3 中，

30

界面对象 40 包括界面对象工具条 42 与下拉菜单 44，其作为单独的窗口 48 显示于浏览器界面 20 内。在图 4 中，界面对象 40 包括界面工具条 42，其包含多个下拉菜单 44 与搜索窗口 46，显示于浏览器界面 20 内的单独的 48 内。界面对象 40，遵照本发明的各种实施例，基本上可包括通过浏览器可获得的任何类型的信息和/或功能。这样，但不
5 限于，界面对象 40 可包括下拉菜单，工具条与下拉菜单，文本信息（例如，广告、优待券等等），文本和/或伴音信息（例如，文本广告与伴随的声音），文本、伴音、和/或图像（动画或非动画）信息，视频，视频和音频，音频，等等。

10

图 2-4 描述的所发明的因特网浏览器界面 20 的多种实施例仅仅是本发明的阐述性的、非限制性的范例。可遵照这里提供的教导对所描述的浏览器界面 20 作出变形。

15

接着参照图 5，那里描述了一种控制因特网浏览器界面 20 的方法，其遵照本发明的一个实施例，一般地被指示为 500。在步骤 510，因特网用户通过因特网网站 130、ISP 80 或该用户的电脑 50 访问控制程序 120。在被因特网用户执行时，控制程序 120 向该用户提供其后控制该用户的因特网浏览器界面 20 的能力，如下面更详细地讨论的。

20

在步骤 520，控制程序 120 在因特网用户的电脑 50 上下载或创建库文件 74，其包含 ActiveX 控件或插件代码，定义界面对象。每次因特网用户启动或激活浏览器 72，库文件 74 被打开，到预先确定的因特网网站 110 的连接被自动建立，优选地，该因特网网站 110 被配置以与该因特网用户的电脑 50、界面对象的 ActiveX 控件或插件代码进行通信，如步骤 530 与 540 所指示的那样。基本地，打开的库文件 74 在
25 浏览器 72?) 内提供外壳，ActiveX 控件或插件提供的功能可于其内添加到浏览器界面 20。在浏览器 72 被关闭之前，库文件 74 或外壳均不会关闭。在步骤 550，创建界面对象 40，其将被显示在浏览器界面 20 内；界面对象 40 的信息和/或功能由 ActiveX 控件或插件定义。遵照本发明的各种实施例，只要用户继续网上冲浪或者浏览器 72 被激

25

30

活，浏览器 72 在浏览器界面 20 内保持显示界面对象 40，如步骤 560 所指示的那样。因此，在因特网用户网上冲浪时，遵照本发明添加到浏览器 72 与浏览器界面 20 的功能不会丢失。

5 本发明提供多种实施例，用于控制浏览器 72 与浏览器界面 20，上面参照图 5 一般地进行了描述，现在将详细地描述每一个实施例。

10 接着参照图 6，并继续参照图 10，现在将提供一种方法的一种实施例的描述，该方法遵照本发明控制和显示浏览器 72 和浏览器界面 20，一般地指示为 600。出于图 6 的目的，并为了下面涉及的讨论，已在因特网用户的电脑 50 上下载或创建了库文件 74，如这里所讨论的那样。在步骤 610，因特网用户启动或激活浏览器 72，以初始化到因特网 90 的访问。在步骤 620，库文件 74 被打开，并且到预先确定的内容提供商的连接被自动建立，如步骤 630 所指示的那样。内容提
15 供商将 BIO 库的 ActiveX 控件或插件代码定义的功能传输到用户的电脑 50 上，以创建界面对象 40，其可被添加到浏览器界面 20 上。界面对象 40 显示于因特网浏览器界面 20，如步骤 640 所指示的那样。当因特网用户漫游因特网 90 时，界面对象 40 的功能，如 ActiveX 控件或插件代码所定义的，保持在浏览器浏览器界面 20 内，无论用户访问的因特网站点的数量或类型，并且在用户使用浏览器软件程序 72
20 访问因特网 90 期间始终持续。当因特网用户从一个因特网站点移动到另一个时，如步骤 650 所指示的那样，本发明判定界面对象 40 是否在该移动后仍存活；浏览器 72 是否仍然在浏览器界面 20 上显示它，如步骤 652 所指示的那样。如果界面对象 40 不再显示于浏览器界面 20
25 内（换言之，它已被从浏览器界面 20 上删除，或者被终止），界面对象 40 被重画，如步骤 660 所指示的那样。如果用户从一个因特网站点移动到另一个时，界面对象 40 仍然存活，并继续显示在浏览器界面 20 内，在步骤 654，本发明还判定浏览器 72 是否活动；这是因为仅在浏览器 72 运作时，界面对象 40 显示于浏览器界面 20 内。如果
30 浏览器 72 不活动，界面对象 40 被终止，库文件 74 被关闭，如步骤 670

所指示的那样。如果浏览器 72 保持活动，在步骤 652，本发明继续确保界面对象 40 显示在浏览器界面 20 内。当因特网用户在因特网站点之间或之中移动时，本发明监控界面对象 40 的状态，并确保在用户漫游因特网 90 期间，浏览器 72 在浏览器界面 20 上显示它。

5

根据图 6 描述的实施例，本发明确保当因特网用户离开因特网网站 130 时，浏览器 72 在浏览器界面 20 上显示界面对象 40，BIO 库即于其上被初始地调用并装载到用户的电脑 50 上。例如，当用户启动或激活浏览器 72 时，自动创建到第一、预先确定的因特网站点 130 10 的连接，该站点由第一内容提供商 100 维护，并且 BIO 库被装载到用户的电脑 50 上。之后，浏览器 72 可通过库文件 74 创建的外壳来获得 BIO 库提供的功能。当因特网用户连接到第二内容提供商 200 维护的第二因特网站点 230 时，界面对象 40 的功能将继续呈现在浏览器界面 20 内。在此实施例中，当到第一因特网站点 130 的连接被终止 15 时，本发明防止电脑 50 的浏览器 72 或操作系统 70 通过卸载库文件 74，使 BIO 库的功能失效。

例如，当浏览器 72 初始地连接到因特网站点 130 时，该站点 130 以 ActiveX 控件软件代码的形式将功能信息作为 BIO 库传输到因特网 20 用户的电脑 50，其被装载到库文件 74。当到第一因特网站点 130 的连接被终止时，如果操作系统 70 或浏览器 72 没有显式地指令库文件 74 关闭或卸载，甚至在到第一因特网站点 130 的连接被关闭之后，库文件 74 将保持被装载，向浏览器界面 20 内的界面对象 40 提供所需要的功能。当因特网用户在多个因特网站点之间移动时，保持库文件 25 74 被装载，使得能够从被装载 ActiveX 控件（其仅由第一因特网站点 130 传输给因特网用户 50）之外、库文件 74 之内装载数据、函数与对象。只要库文件 74 保持被装载，库文件 74 之内、ActiveX 控件之外创建的任何数据或对象将保持被装载，并继续在浏览器界面 20 内工作。

30

为在冲浪期间保持库文件 74 打开，在浏览器 72 连接到因特网站点 130 之后和该连接终止之前，一个全局对象，对象 A，被创建于因特网用户的电脑 50 的程序堆中，而非调用函数的堆中。因此，当调用函数完成时，该全局对象仍存活。该全局对象可以用 C++ 的 new 操作符创建，或通过在全局声明中声明一个全局对象。在任一种情形下，当浏览器 72 与第一因特网站点 130 之间的连接终止时，该全局对象将仍存活。

在第一因特网站点 130 提供的 ActiveX 控件关闭后，也就是终止到第一因特网站点 130 的初始连接后，这样定义的全局变量保持工作。一旦创建该全局对象，即使用该全局对象创建一个界面。该界面将服务于，比方说，向浏览器 72 与浏览器界面 20 删除、替代和/或添加功能。该界面可作为该全局对象的一部分创建，或由该全局对象分配一个新的界面对象 40。例如，可创建界面对象 40，其通过，比方说，在浏览器窗口 38（参见，例如，图 3 与 4）之内创建界面对象窗口 48，并将其作为子窗口添加到浏览器界面 20。作为另一种可供选择的方案，可直接修改浏览器界面 20，其通过，比方说，添加或修改浏览器界面 20 内的浏览器工具条 22 或浏览器工具条对象 24。还有一种可供选择的方案是创建界面工具条 42，其分离于浏览器界面 20，如图 4 所描述的那样。

另外，需要一个指针，其用于控制浏览器 72 并指令浏览器 72 建立到预先确定的因特网站点 130 的连接。优选地，该指针被全局地存储，使得库文件 74 内任何向浏览器 72 发送命令的函数或对象均可访问它。在 Microsoft Internet Explorer 中，比方说，可使用 IWebBrowser、IWebBrowser2 或 IWebBrowserApp 对象链接与嵌入（OLE）接口命令来创建该指针。使用 Microsoft 基本类库（MFC），比方说，可使用 COleControl 类的 GetClientSite 成员来创建该指针，以从第一因特网站点 130，也就是装载 BIO 库的因特网站点，获取指针。GetClientSite 作为浏览器 72 的进入点，与 BIO 库进行通信。可使用前一步骤返回

的 IOleClientSite 类的 GetContainer 成员来取得 BIO 库的容器 (container) 的指针。BIO 库的容器是这样的容器, ActiveX 控件于其内被装载。一般地, 因特网浏览器界面 20 包含多个部分, 包括浏览器工具条 22 与内容窗口 32。浏览器 72 为因特网用户访问的每一个网页创建一个文档对象, 该文档对象包含出现在一个特定网页上的所有数据。该文档对象也是 BIO 库的容器。因此, 也可称文档对象为 BIO 库的容器。

可使用前一步骤返回的 IOleContainer 类的 QueryInterface 成员来取得 IServiceProvider 接口的指针, 可使用其来定位浏览器 72 呈现的任何其它接口。最后, 可使用前一步骤返回的 IServiceProvider 类的 QueryService 成员来取得 IWebBrowserApp、IWebBrowser 或 IWebBrowser2 接口的指针, 其取决于调用 BIO 库的浏览器 72 提供的特定接口。

在另一种可供选择的实施例, 可使用本发明来修改整个浏览器窗口 30。如果与将界面对象 40 集成到现存的浏览器窗口 38 相反, 整个浏览器窗口 30 被修改, 被修改的浏览器窗口 30 的整个类可被子类化, 或者, 作为另一种可供选择的方案, 该特定的浏览器窗口 38 可被子类化。如这里所使用的, 术语子类化窗口, 也称为勾住窗口, 是指以用户定义的窗口消息处理过程替代原先的浏览器窗口消息处理过程, 以处理发送给窗口的所有消息。例如, 可使用 Microsoft 基本类库 (MFC) 的 CWnd::SubclassWindow() 函数将窗口子类化。作为另一种可供选择的方案, 可使用调用 SetWindowLong 函数 (一个 Microsoft Windows 函数) 并传递 GWK_WNDPROC 参数 (一个 Microsoft Windows 常数) 将窗口子类化。SetWindowLong 函数调用返回的指针可保存到用于被子类化的窗口的原浏览器窗口消息处理过程。这使得 BIO 库能够截获传递给窗口 30 或 38 的所有消息, 并且 BIO 库可以解释来自界面控件 24 的命令, 该界面控件 24 包括按钮、菜单等等, 其为浏览器 72 提供, 或来自界面对象 40, 为 BIO 库遵照本发明添加。

用户定义的窗口消息处理过程，其为 BIO 库提供，并且其替代原浏览器窗口消息处理过程，这里称之为 BIO 过程。对浏览器 72 使用 BIO 过程，可以截获和修改打算送给浏览器 72 的消息（例如，命令），或者 BIO 过程可以将替代或新消息（例如，界面对象 40 的消息处理器）传输到浏览器 72。

本发明还确保界面对象 40 不会从浏览器界面 20 删除。例如，当因特网用户访问新网站时，一些因特网浏览器 72 重画整个浏览器界面 20。在用户进行这样的因特网移动之后，尽管全局变量可能仍然工作，界面对象 40 将被从浏览器界面 20 上删除，因此将不会显示在浏览器界面 20 上。为防止这样的情况发生，BIO 过程可截获来自浏览器 72、重画浏览器界面 20 的消息，并在重画浏览器界面 20 之后立即重画界面对象 40。作为另一种可供选择的方案，可周期性地测试界面对象 40 在浏览器界面 20 内是否存在，如果在浏览器界面 20 内不存在，可重画界面对象 40。优选地，这样的周期性的测试以小于大约一秒的间隔发生。

作为上面所描述的实施例（在图 6 中描述）的一个范例，ActiveX 控件被作为 BIO 库装载，并向浏览器界面 20 添加菜单项（与功能）。本发明创建 ActiveX 控件，其动态地创建一个新的全局对象，对象 A，其创建新的菜单对象（其可以是界面对象 40），具有所需要的、要添加到浏览器界面 20 的功能。可使用（比如说）指令：AfxGetMainWnd() -> GetMenu() -> AppendMenu()，来将菜单对象 40 添加到浏览器界面 20，这里浏览器界面 20 的所附加的菜单将包括弹出式菜单，指向菜单对象 40。浏览器界面窗口之后将被子类化，并处理 BIO 过程所处理的菜单对象 40 的消息，以及被传递到浏览器 72 的消息处理器的浏览器界面 20 的消息。当因特网用户从第一因特网站点 130 上断开连接时（例如，离开包含 ActiveX 控件的网页），ActiveX 控件将关闭，但全局对象将保持在程序堆中，并继续向浏览器界面 20 提供所需要的功能。

作为对子类化浏览器窗口（界面对象 40 被添加到其中）的替代，对象 A，或者其后代之一，可保留对界面对象 40 的所有权。这样，可创建界面对象 40 的消息处理器。例如，除了使用 Microsoft 基本类库的 SetOwner 函数将界面对象 40 的所有权设置为对象 A 或其后代之一以外，遵照上面所描述的本发明的实施例，可向浏览器工具条 22 添加界面对象 40。

接着参照图 7，并继续参照图 10，现在将提供一种方法的另一种可供选择的实施例的描述，该方法控制和显示浏览器界面 20，其遵照本发明，一般地指示为 700。出于图 7 的目的，并为了下面进行的讨论，已在因特网用户的电脑 50 上创建了库文件 74，如这里所讨论的那样。在步骤 710，因特网用户启动或激活浏览器 72，以访问因特网 90。在步骤 720，库文件 74 在用户的电脑 50 上被打开，并且到预先确定的因特网站点 130 的连接被自动建立，如步骤 730 所指示的那样。在步骤 740，内容提供商将 BIO 库的 ActiveX 控件或插件代码定义的功能传输到用户的电脑 50（也就是到库文件 74）上，以创建界面对象 40，可使用持续循环将其显示在浏览器界面 20 上，以控制界面对象 40 的显示。仅当持续循环被终止时，可删除界面对象 40（也就是终止其功能）。当因特网用户漫游因特网 90 时，界面对象 40 的功能，如 ActiveX 控件或插件代码所定义的，保持在浏览器浏览器界面 20 内，无论用户访问的因特网网站的数量或类型，并且只要浏览器 72 保持被激活，以及因特网用户使用该浏览器软件程序访问因特网，就始终持续。当因特网用户从一个网站移动到另一个时，如步骤 750 所指示的那样，本发明判定界面对象 40 是否在该移动后仍存活，换言之，浏览器界面 20 是否仍然显示它，如步骤 752 所指示的那样。如果浏览器 72 不再在浏览器界面 20 内显示界面对象 40，界面对象 40 被重画，如步骤 760 所指示的那样。如果用户从一个网站移动到另一个后，界面对象 40 仍然存活，并继续显示在浏览器界面 20 内，在步骤 754，本发明还判定浏览器 72 是否活动；这是因为仅在浏览器 72 运作时，

浏览器 72 在浏览器界面 20 内显示界面对象 40。如果浏览器 72 不活动，界面对象 40 被终止，库文件 74 被关闭，如步骤 770 所指示的那样。如果浏览器 72 保持活动，在步骤 756，本发明判定持续循环是否已被终止。如果循环已被终止，界面对象 40 也被终止，如步骤 770 所指示的那样。如果循环仍在执行，本发明判定界面对象 40 是否仍显示于浏览器界面 20 之内，如步骤 752 所指示的那样。

图 7 中描述的本发明的实施例防止 ActiveX 控件卸载，其冻结库文件 74（ActiveX 控件代码被装载于其内）的操作，甚至在操作系统 70 或浏览器 72 生成卸载或终止库文件 74 的指令时也是如此。由于库文件 74 被冻结，并且永远不结束卸载，在库文件 74 被指令卸载之后，ActiveX 控件在库文件 74 内创建的所有数据、函数与对象将继续存在和工作。

在一些操作系统 70 和/或浏览器 72 中，当 ActiveX 控件不再出现在因特网站点时，BIO 库将被关闭。当因特网用户从具有 ActiveX 控件的因特网站点，例如，图 10 中的因特网站点 130，移动到另一个不具有 ActiveX 控件的因特网站点，例如图 10 中的因特网站点 230 时，这也可能发生。为使得界面对象 40 在不存在 ActiveX 控件的情况下继续操作，必须防止 BIO 库（和库文件 74）关闭。

为实现这一点，创建一个指针，其被用于控制浏览器 72。优选地，该指针被全局地存储，使得库文件 74 内任何可能需要向浏览器 72 发送命令的函数或对象均可访问它。在 Microsoft Internet Explorer 中，比方说，可使用 IWebBrower、IWebBrower2 或 IWebBrowerApp OLE 接口命令来创建该指针。为使用 Microsoft 基本类库来做到这一点，可使用 COleControl 类的 GetClientSite 成员（用于与 BIO 库通信）来获取 BIO 库的因特网站点（也就是提供 ActiveX 控件的因特网站点）的指针。可使用前一步骤返回的 IOleClientSite 类的 GetContainer 成员来取得 BIO 库的容器的指针。可使用前一步骤返回的 IOleContainer

类的 `QueryInterface` 成员来取得 `IServiceProvider` 接口的指针。使用 `IServiceProvider` 接口来简便地查找浏览器 72 呈现的任何其它接口。可使用前一步骤返回的 `IServiceProvider` 类的 `QueryService` 成员来取得 `IWebBrowserApp`、`IWebBrowser` 或 `IWebBrowser2` 接口的指针，其取决于调用 BIO 库的浏览器 72 的版本所呈现的接口。

为防止库文件 74 关闭，在其能够终止之前，其操作被停止。为冻结或停止库文件 74 的操作，可创建和执行持续的程序循环，其仅当 BIO 库将被卸载时终止，此时，该程序循环也泵取消息队列。这里称该消息循环为消息泵，其可在循环中使用例如 `PeekMessage`、`GetMessage`、`TranslateMessage` 与 `DispatchMessage` 命令来创建。下面提供了示例性的 C++ 代码，以执行消息泵：

```
while (m_Continue)
{
    if (PeekMessage(&msg, NULL, 0, 0, PM_NOREMOVE) != 0)
    {
        GetMessage(&msg, NULL, 0, 0);
        TranslateMessage(&msg);
        DispatchMessage(&msg);
    }
}
```

其中 `m_Continue` 是布尔变量，其被用于指示循环何时停止，而 `PeekMessage`、`GetMessage`、`TranslateMessage` 与 `DispatchMessage` 都是 Windows 函数。如果 `m_Continue` 等于假，循环将结束，从而结束消息泵。`msg` 参数是对 Windows MSG 结构的引用。

消息泵优选操作的方式是其使用 `PeekMessage` 函数检查消息队列中是否有任何消息在等待。如果有消息，消息泵使用 `GetMessage` 函数从消息队列中攫取该消息，并使用 `TranslateMessage` 函数将其从虚键消息译为字符消息。最后，消息泵使用 `DispatchMessage` 函数将该

消息发送给应当接收该消息的原窗口消息处理过程。

也可使用操作系统 70 的内建能力来构建消息泵，以泵取消息队列。例如，使用诸如 MFC 命令 `CWnd::MessageBox("my modal", MB_OK)` 等命令，模式对话框或消息框很好地服务于这个目的，可被用于提供所需要的对库文件 74 的操作的冻结或停止。

只要消息泵执行持续循环，甚至在因特网用户访问其它因特网站点时，ActiveX 控件将不会终止。

10

图 7 所描述的本发明的实施例以与上面所描述的图 6 的实施例相同的方式（例如，子类化、消息处理等等）执行。

对于（图 6 与 7 的）两个上面所描述的实施例而言，为界面对象 40 提供退出函数，使得对象 A 及其所有后代将被关闭，这是优选的，尽管不是必需的。调用退出函数的可能的示例性的场景包括在 BIO 过程中截获到关闭浏览器窗口的消息，或者周期性地寻找浏览器窗口并且，如果未找到它，终止 ActiveX 控件。

15

接着参照图 8，描述了本发明的方法的又一种实施例，其一般地指示为 800。出于图 8 的目的，并为了下面涉及的讨论，已在因特网用户的电脑 50 上创建了库文件 74，如这里所讨论的那样。在步骤 810，因特网用户启动或激活浏览器 72，以访问因特网 90。在步骤 820，库文件 74 在用户的电脑 50 上被打开，并且到预先确定的因特网站点 130（参看，例如，图 10）的连接被自动建立，如步骤 830 所指示的那样。在步骤 832，创建新的浏览器界面，其为浏览器 72 提供的初始浏览器界面的副本。在步骤 840，创建函数窗口（Function Window），其代表原浏览器界面，插件的功能于其中被初始地装载。

25

图 8A 描述了遵照本发明的本实施例创建函数窗口的步骤，其一

30

般地指示为 840。在步骤 842，确定初始浏览器界面窗口的句柄。初始浏览器界面窗口，现在是函数窗口，在步骤 844 被隐藏和/或使之失效，使得其不能被关闭，否则将导致 BIO 库崩溃。在步骤 846，创建一个指针，其被用于控制浏览器 72。最后，在步骤 848，创建新的浏览器界面窗口，因特网用户可使用其来继续漫游因特网。

再次参照图 8，并从步骤 850 开始，现在可控制浏览器界面，其首先子类化任何浏览器窗口，或浏览器使用的任何窗口，其将被控制，之后添加、删除和/或修改窗口，如下面更详细地描述的那样。初始浏览器窗口消息处理过程由 BIO 过程所代替。在步骤 860，本发明判定是否已打开任何新的浏览器窗口，也就是这样的窗口，其可能尚未遵照本发明被子类化，或者可能尚未被控制。如果已打开新的窗口，在步骤 870，本发明判定是否要将界面对象 40 添加到这些新窗口。如果不是，本发明判定因特网用户是否想要关闭浏览器 72，如步骤 880 所指示的那样。如果是这样，在步骤 890，本发明关闭所有浏览器窗口，包括函数窗口。如果用户不想要关闭浏览器 72，如步骤 880 所判定的那样，本发明返回到步骤 860，并再次判定是否已打开新的浏览器窗口。

继续参照图 8、8A 与 10，现在将更详细地讨论本发明的上面所述的实施例。当内容提供商 100 通过因特网站点 130 装载 ActiveX 控件时，典型地，作为对浏览器 72 建立到该网站 130 的连接并调用 ActiveX 控件的响应，使位于用户的电脑上的库文件 74 打开，并在浏览器 72 内创建外壳，ActiveX 控件的代码可于其内装载。如果当 ActiveX 控件被关闭时（当用户终止到因特网站点 130 的连接时），操作系统 70 或浏览器 72 显式地指令包含 ActiveX 控件的库文件 74 卸载或关闭，在库文件 74 被卸载时，创建于 ActiveX 控件之外、库文件 74 之内的任何数据、函数或对象将被摧毁。为防止库文件 74 卸载，防止浏览器 72 关闭 ActiveX 控件，直到被指令这样做。如果永远不指令 ActiveX 控件关闭，就永远不卸载库文件 74。

一种实现这一点的优选的方法是隐藏装载 ActiveX 控件的初始浏览器界面窗口和/或使之失效，并创建该窗口的新的拷贝，因特网用户可于其内继续工作和漫游因特网。由于优选地，初始浏览器界面被隐藏和/或使之失效，不能关闭 ActiveX 控件，直到库文件 74 显示、使之有效或关闭初始浏览器界面窗口，也就是装载库文件 74 的窗口。

优选地，图 8 的上面所描述的方法在浏览器 72 内作为标准 ActiveX 控件装载 BIO 库，其使用，比如说，<Object>标签，其典型地包含在网页 130 内，或作为 Band 对象，如 Microsoft Internet Explorer 4.x 软件开发工具包所描述的那样。这指令浏览器 72 初始化并运行包含 BIO 库的代码的库文件 74。

第一次初始化和调用 BIO 库时，创建函数窗口，当因特网用户访问其它因特网网站 210，也就是其它网页时，其通过保持与 ActiveX 控件的会话本身打开而保持 BIO 库打开。函数窗口也使得不具有打开的 BIO 库的拷贝的浏览器窗口有可能访问浏览器 72 的 OLE 接口。

为创建函数窗口，优选地，初始浏览器界面窗口（也就是装载 BIO 库的窗口）被隐藏和/或使之失效。这可通过确定初始浏览器界面窗口的句柄来完成，其从 BIO 库的句柄开始。为从 BIO 库的句柄获取初始浏览器界面窗口句柄，持续调用 GetParent 函数（Windows 函数），直到该函数调用的当前值代表桌面窗口的下一级。例如，在循环中执行诸如“m_Handle = GetParent(m_Handle)”的语句，可提供想要的功能与结果，其中 m_Handle 的值初始地等于 BIO 库的句柄的值，并且将最终返回初始浏览器界面窗口的句柄。

下一步骤是隐藏和/或使初始浏览器界面窗口（现在称之为函数窗口）失效，使得因特网用户不能关闭该函数窗口（比如说，通过关闭浏览器 72），从而促使 BIO 库从浏览器界面 20 卸载和去除 BIO 库

提供的功能。为了对于用户隐藏初始浏览器界面窗口和/或使该浏览器窗口失效于用户驱动的事件，可向初始浏览器界面窗口发送 WM_SHOWWINDOW 和/或 WM_ENABLE 消息（两者均为 Windows 常量），取值为隐藏和/或使浏览器窗口失效。例如，可使用 PostMessage 或 SendMessage 函数（现有的 Windows 函数）来向初始浏览器界面窗口发送消息，其使用浏览器窗口句柄。作为另一种可供选择的方案，可使用 ShowWindow 与 EnableWindow 函数（现有的 Windows 函数）来获得同样的结果。

10 创建一个指针，以控制浏览器 72。优选地，该指针被全局地存储，使得库文件 74 内任何向浏览器 72 发送命令的函数或对象均可访问它。在 Microsoft Internet Explorer 中，比方说，可使用 IWebBrowser、IWebBrowser2 或 IWebBrowserApp OLE 接口来创建该指针。为使用 Microsoft 基本类库来做到这一点，可使用 COleControl 类的 GetClientSite 成员，其作为浏览器 72 的进入点，来与 BIO 库通信，并获取 BIO 库的因特网站点，也就是装载 ActiveX 控件的因特网站点的指针。可使用前一步骤返回的 IOleClientSite 类的 GetContainer 成员来取得 BIO 库的指针。可使用前一步骤返回的 IOleContainer 类的 QueryInterface 成员来取得 IServiceProvider 接口的指针；优选地，使用 IServiceProvider 接口来查找浏览器 72 呈现的任何其它接口。可使用前一步骤返回的 IServiceProvider 类的 QueryService 成员来取得 IWebBrowserApp、IWebBrowser 或 IWebBrowser2 接口的指针，其取决于调用 BIO 库的浏览器 72 版本呈现的接口。

25 最后，由于先前使用来创建函数窗口的浏览器窗口已被隐藏和/或使之失效，创建新的浏览器窗口，因特网用户可以使用其来继续网上冲浪，并继续访问各种不同的因特网站点。优选地，使用 IWebBrowser、IWebBrowser2 或 IWebBrowserApp OLE 接口中的任何一个来创建新的浏览器窗口，比如说，使用该 OLE 接口的 Navigate 或 Navigate2 成员。作为另一种可供选择的方案，可向浏览器 72 发送

WM_COMMAND 消息，其响应任何这样的命令，浏览器 72 可使用该命令来打开新的浏览器窗口，例如新窗口命令或在新窗口中打开命令，等等。也可使用浏览器 72 提供的动态数据交换（DDE）支持来打开新窗口。

5

BIO 库现在必须控制各种浏览器界面特性与功能。第一步骤是子类化任何浏览器窗口，或浏览器使用的任何窗口（也就是子窗口），其将被遵照本发明进行控制。使用 BIO 窗口消息处理过程来替代原窗口消息处理过程，下文中称之为 BIO 过程。

10

一旦已子类化浏览器窗口，或其任何子窗口，有可能向子类添加菜单，其使用 GetMenu 函数获得浏览器窗口菜单的指针。一旦获得菜单的句柄的指针，可使用菜单函数，例如 ModifyMenu、AppendMenu、InsertMenu 等等，来向浏览器窗口添加任何想要的菜单。用来子类化 BIO 窗口的 BIO 过程必须处理任何赋予菜单的命令。不得使用与包含在浏览器中的相同的命令标识符来创建菜单。

15

可向浏览器界面 20 添加界面对象工具条 42，其获取窗口的句柄（下文中称为帧句柄），界面对象工具条 42 将使用标准 Windows 函数将其添加到该窗口。典型地，该窗口将是 BIO 窗口，或者帧窗口，其为 BIO 窗口的子窗口。之后，以帧句柄为父窗口创建一个窗口。例如，要将对话框条（其为一种形式的工具条）作为界面对象 40 添加，创建一个 CDialogBar 类型（一个 Microsoft 基本类）或自该类型派生的对象，调用其 Create 方法时使用帧句柄。如果使用资源，例如，图像、工具条、对话框等等，并且浏览器 72 与 BIO 库不共享同样的资源，在可从 BIO 库资源装载任何数据之前，用 BIO 库资源临时性地替换浏览器的资源。之后，可用浏览器原先的资源替换 BIO 库资源。

20

25

当打开新的浏览器窗口后，可能想要向这些新窗口添加界面对象 40。可创建定时器，其使用 SetTimer Windows 函数，其将调用用户定

30

义的函数，该函数将使用 FindWindowEx 函数（Windows 函数）来检查桌面窗口的每一个子窗口，以查找与函数窗口具有同样的类名的窗口。对于尚未被修改，也就是没有界面对象 40 的浏览器窗口而言，可获取必需的句柄，并进行与原 BIO 窗口相同的改变。

5

最后，当因特网用户想要关闭浏览器 72 时，必须判定是否所有的浏览器窗口均被关闭，函数窗口除外，并且如果所有其它浏览器窗口被关闭，函数窗口也必须被关闭。可通过在 BIO 消息处理过程中监听 WM_CLOSE 消息（Windows 常量），或通过设置定时器来周期性地判定有多少个浏览器窗口是打开的，来实现这一点。要关闭原函数窗口，可向该窗口发送 WM_CLOSE 消息。

10

接着参照图 9，描述了一种方法的另一种可供选择的实施例，该方法遵照本发明控制和显示因特网浏览器界面 20，一般地指示为 900。

15

步骤 910、920 和 930 基本上与上面所描述的图 6-8 的实施例相同。在步骤 940，创建新的浏览器界面窗口，初始浏览器界面窗口被隐藏和/或使之失效，其称之为函数窗口。插件确定初始浏览器界面窗口的句柄，将该窗口隐藏和/或使之失效，并且创建新的浏览器界面窗口，其可被因特网用户使用。在步骤 950，子类化所有浏览器窗口，之后可控制浏览器界面，如步骤 960 所指示的那样，以用于所有打开的窗口。在步骤 962，本发明判定是否已打开任何新的浏览器窗口，在此情形下本发明返回步骤 960。如果没有打开新的浏览器界面窗口，步骤 962 进行到步骤 964，以判定因特网用户是否想要关闭浏览器。在关闭函数窗口前，必须先关闭所有窗口，其为步骤 966 判定。如果所有窗口被关闭，则关闭函数窗口，如步骤 970 所指示的那样。否则，步骤 966 返回到步骤 964。

20

25

在本发明的又一种可供选择的实施例中，本发明提供一种方法来控制因特网浏览器界面，其使用浏览器插件，以控制调用的浏览器 72

30

的功能，并在用户离开装载该插件的因特网站点 130 之后，保留该插件的功能。

5 当作为对浏览器 72 建立到因特网站点 130 的连接并调用插件的响应，将浏览器插件装载到因特网用户的电脑 50 时，库文件 74 在浏览器 72 内创建外壳，插件的代码可于其内装载。如果当插件被关闭时，操作系统 70 或浏览器 72 显式地指令库文件 74 卸载，在库文件 74 被卸载时，创建于插件之外、库文件 74 之内的任何数据、函数或对象将被摧毁。为防止库文件 74 卸载，防止浏览器 72 关闭插件，直到
10 浏览器 72 接收到关闭插件的指令。如果永远不指令插件关闭，也就永远不指令库文件 74 卸载。可通过隐藏装载插件的初始浏览器窗口和/或使之失效，并通过创建该窗口的新的拷贝，因特网用户可继续使用其访问和漫游因特网，来实现这一点。由于优选地，初始浏览器窗口被隐藏和/或使之失效，不能关闭插件，直到库文件 74 选择显示、
15 使之有效或关闭装载插件的初始浏览器窗口。

例如，在浏览器 72 内作为标准插件装载浏览器插件，优选地，其在网页 110（参见例如图 10）使用<Embed>标签，其指令浏览器 72 初始化并装载包含插件代码（也就是 BIO 库）的库文件 74。

20

第一次初始化和调用 BIO 库时，创建函数窗口，其隐藏和/或使原浏览器窗口失效，从而防止 BIO 库卸载，其保持与插件的会话本身打开。函数窗口也使得不具有打开的 BIO 库的拷贝的浏览器窗口有可能访问用于插件的内建的应用编程接口（API）（也就是通过插件功能
25 实现的因特网用户可经由浏览器界面 20 获得的功能），例如由 Netscape Navigator 与 Microsoft Internet Explorer 所提供的。

优选地，创建函数窗口，其在浏览器 72 第一次调用插件时，隐藏和/或使初始浏览器窗口（插件处于其内）失效。为实现这一点，插件
30 必须首先确定初始浏览器窗口的句柄。将 NPWindow 结构的窗口成

员作为 `NPP_SetWindow` 函数的第二个参数从浏览器 72 传递给 BIO 库，该成员为插件窗口的句柄（`NPWindow` 和 `NPP_SetWindow` 是 Netscape 和 Internet Explorer 用于插件的 API 的一部分）。为从该窗口成员获取初始浏览器窗口句柄，持续调用 `GetParent` 函数（Windows 函数），直到该函数调用的当前值代表桌面窗口的下一级。例如，在循环中执行诸如“`m_Handle = GetParent(m_Handle)`”的语句，可提供想要的功能与结果，其中 `m_Handle` 的值初始地等于 `NPWindow` 结构的句柄的值，并且将最终返回初始浏览器窗口的句柄。

之后，隐藏和/或使初始浏览器窗口失效，使得因特网用户不能关闭该函数窗口，从而导致 BIO 库崩溃。为了对用户隐藏初始浏览器窗口和/或使该窗口失效于用户驱动的事件，可向初始浏览器窗口发送 `WM_SHOWWINDOW` 和/或 `WM_ENABLE` 消息（两者均为 Windows 常量），取值为隐藏和/或使浏览器窗口失效。可以通过，例如，使用 `PostMessage` 或 `SendMessage` 函数（Windows 函数）来向初始浏览器窗口发送消息，其使用浏览器窗口句柄，来实现这一点。作为另一种可供选择的方案，可使用 `ShowWindow` 与 `EnableWindow` 函数（Windows 函数）来获得同样的结果。

最后的步骤是创建新的浏览器窗口，在初始浏览器窗口已被隐藏和/或使之失效后，因特网用户可以使用其来继续网上冲浪。可以通过，比如说，调用下列 Netscape 与 Internet Explorer 插件 API 函数中的任何一个：`NPN_GetURL`、`NPN_PostURL`、`NPN_GetURLNotify`、`NPN_PostURLNotify`，其中目标参数设置为 `_new`、`_blank` 或尚未存在的任何窗口名，来实现这一点。上面函数的 `NPP` 参数为 `NPP` 结构，浏览器 72 将其提供给函数窗口的插件。做到这一点的另一种方法是向浏览器 72 发送 `WM_COMMAND` 消息，其响应任何这样的命令，浏览器 72 可使用该命令来打开新的浏览器窗口，例如新窗口命令或在新窗口中打开命令。也可使用浏览器提供的动态数据交换（DDE）支持来打开新窗口。`NPN_GetURL`、`NPN_PostURL`、

NPN_GetURLNotify、NPN_PostURLNotify 和 NPP 结构是 Netscape 和 Internet Explorer 用于插件的 API 的一部分，WM_COMMAND 是 Windows 常量。

- 5 BIO 库现在可控制浏览器 72 的特性与功能。第一步骤是子类化任何浏览器窗口，或浏览器使用的任何窗口（统称 BIO 窗口），其将被遵照本发明进行控制。

一旦已子类化浏览器窗口，或其任何子窗口，有可能向浏览器界面 20 添加菜单，其使用 GetMenu 函数（Windows 函数）获得浏览器窗口菜单的指针。一旦获得菜单的句柄的指针，可使用菜单函数，例如 ModifyMenu、AppendMenu、InsertMenu 等等（Windows 函数），来向浏览器窗口添加任何想要的菜单。用来子类化 BIO 窗口的 BIO 过程必须处理任何赋予菜单的命令，不得使用与包含在浏览器 72 中的相同的命令标识符来创建菜单。

作为另一种可供选择的方案，或者附加地，可向浏览器界面 20 添加界面对象工具条 42，其获取窗口的句柄（下文中称为帧句柄），工具条 42 将使用标准 Windows 函数添加到该窗口；典型地，该窗口将是 BIO 窗口，或者帧窗口，其为 BIO 窗口的子窗口。之后，以帧句柄为父窗口创建一个窗口。例如，要将对话框条（其为一种形式的工具条）作为界面对象 40 添加，可创建一个 CDialogBar 类型（一个 Microsoft 基本类）或自该类型派生的对象，调用其 Create 方法时使用帧句柄。如果使用资源，例如，图像、工具条、对话框等等，并且浏览器 72 与 BIO 库不共享同样的资源，在可从 BIO 库资源装载任何数据之前，用 BIO 库资源临时性地替换浏览器的资源。之后，可用浏览器原先的资源替换 BIO 库资源。

当打开新的浏览器窗口后，可向这些新窗口添加界面对象 40。可通过创建定时器，其使用 SetTimer（Windows 函数），其将调用用

户定义的函数，该函数将使用 FindWindowEx 函数（Windows 函数）来检查桌面窗口的每一个子窗口，以查找与函数窗口具有同样的类名的窗口，来实现这一点。对于尚未具有被修改的界面，也就是不包含界面对象 40 的新窗口而言，可获取必需的句柄，并对这些窗口进行
5 与原 BIO 窗口相同的改变。

最后，当因特网用户希望关闭浏览器 72 时，必须判定是否所有的浏览器窗口均被关闭，函数窗口除外，并且如果它们是这样，函数窗口可被关闭。可以，比如说，通过在 BIO 窗口消息处理过程中监听
10 WM_CLOSE 消息（Windows 常量），或通过设置定时器，周期性地检查打开的浏览器窗口的数量，来实现这一点。可通过向其发送 WM_CLOSE 消息来关闭原函数窗口。

遵照本发明，可装载 BIO 库（也就是插件），其功能自动地，也
15 就是不要求用户主动地访问特定因特网站点，也就是浏览调用该插件的网页，提供于浏览器界面 20 内。例如，Netscape 在其 Windows 注册表内有一个标识为自动启动（Automation Startup）的键。在激活时，Netscape 装载列于自动启动键内的所有 OLE 控件。通过在自动启动键内放置对库文件 74（从而对 BIO 库与插件，其定义界面对象 40）
20 的引用或调用，可在每次用户启动或激活 Netscape 浏览器时装载该库文件。可在该库文件 74 内包含在浏览器界面 20 内创建界面对象 40 的实例的指令，以及让浏览器 72 建立到预先确定的因特网站点 130 的连接指令。使用此技术，用户不必选择访问特定因特网站点 130 以装载 BIO 库。需要保持库文件 74 打开，至少直到可在浏览器 72 内
25 装载插件，以通过浏览器界面 20 显示和访问。做到这一点的一种方法是增加与库文件 74 相关联的引用计数，使得当 Netscape 卸载自动启动键内列出的 OLE 控件时，将不会卸载库文件 74，因为其具有更高的引用数。

30 可将库文件 74 作为插件装载，使用 DDE 周期性地查找 Netscape

DDE 服务器，其使用定时器或循环。当浏览器 72 从 Netscape DDE 服务器接收到返回时，Netscape 准备接收命令，并可装载插件。之后，可使用 DDE 来向浏览器 72 发送命令，例如 WWW_OPENURL，其将促使插件装载，如这里所讨论的那样。

5

另一种用于 BIO 库的隐藏 Netscape 插件窗口的方法是将其从任务条（也就是 Windows “开始” 按钮所处的地方）去除，并将其放置于可视的屏幕之外。一种将其从任务条去除的方法是调用 SetWindowLong，并将插件窗口的窗口样式改为工具箱窗口。工具箱窗口不显示在工具条内。可通过调用 MoveWindow，并提供不在用户桌面的可视范围之内的坐标，来将 Netscape 插件窗口放置于屏幕之外。

10

用于替代原浏览器消息处理过程的 BIO 窗口消息处理过程必须知道一个消息准备到达哪个窗口，以及 BIO 窗口消息处理过程接收到消息后如何动作。做到这一点的一种优选的方法是创建映射表，其将一条信息链接到另一条。对于本发明而言，优选地使用这样的映射表，其将窗口句柄链接到包含该窗口的重要信息的结构。例如，当 BIO 库将界面对象 40 添加到新的浏览器窗口时，在映射表内创建新条目，其将该 BIO 窗口的句柄链接到包含对该 BIO 窗口有用的信息的结构。优选地，包含在该结构内的信息中的一条是用于该 BIO 窗口的浏览器的原窗口消息处理过程。有必要维护浏览器的原窗口消息处理过程，使得如果 BIO 窗口消息处理过程不知道如何处理一条消息，它可以将该消息传递到浏览器的原窗口消息处理过程。

15

20

当 BIO 窗口消息处理过程接收到一条消息时，传递给该过程的第一参数是接收该消息的窗口的句柄。为获取包含所有该窗口特定的数据的结构，使用该窗口句柄作为键，在映射表内执行查找。返回的结构将包含所有保存的窗口特定的信息，例如原窗口消息处理过程。

25

当控制浏览器界面 20 时，本发明改变浏览器 72 工作的方式。因

30

特网用户使用浏览器可以做的几乎所有事情都通过向浏览器的窗口或子窗口发送消息来工作。发送消息的对象或窗口包括菜单、按钮、组合框以及几乎所有其它的东西，因特网用户能够直接与其交互，例如界面控件。例如，消息可以分为四个部件：1)准备接收该消息的窗口的句柄；2)该消息的 `msg` 值；3)`wParam`，其使用通常依赖于 `msg` 值；和 4)`lParam`，其值通常也依赖于 `msg` 值。

例如，点击浏览器的窗口内的按钮可以向浏览器窗口的窗口消息处理过程发送这样的消息，其包含 `WM_COMMAND`，其为 Windows 常量，`msg` 值。该消息内的 `wParam` 变量的两个低字节将是一个数，其用于标识哪个按钮被按下。

通过子类化浏览器的窗口或子窗口，如上面所描述的那样，可截获用户与任何界面控件交互式发送的任何消息。一旦截获到消息，`BIO` 窗口消息处理过程可以解释它并对它做出反应。如果界面控件的功能保持不变（也就是本发明不添加、删除或修改），可将该消息传回原窗口消息处理过程。这样，基本上可以控制所有存在于浏览器 72 内的界面控件。另外，可将界面控件添加到浏览器界面 20，并向其赋予命令标识符（其在 `wParam` 内传递）。之后，`BIO` 窗口消息处理过程可以解释 `wParam` 并提供要被装载的界面控件的功能。另外，可以去除功能，其在接收到与要从浏览器上去除的界面控件相关联的命令标识符时，简单地让发明的窗口消息处理过程不做任何事情。这样，可防止该命令传递到浏览器窗口的原消息处理过程。

使用本发明的各种实施例，如上面详细地讨论的那样，因特网用户可创建具有用户定义的界面控件的浏览器界面 20。之后，通过将该浏览器界面 20 的窗口的父窗口设置为已被子类化的窗口，任何来自该新（也就是受控的）窗口的消息将在 `BIO` 窗口消息处理过程中处理。其可用来向浏览器界面 20 添加任何界面对象 40，例如工具条、对话框条、浮动对话框等等。

提供下面的阐述性的、非限制性的应用范例，以进一步描述本发明。当因特网用户漫游因特网时保持持久的插件或 ActiveX 控件可向浏览器界面 20 添加界面对象 40，其使得用户能够从位于因特网上的数据库下载他们的“书签”或“收藏（favorites）”。界面将直接添加到浏览器界面，并将允许用户访问他们下载的“书签”或“收藏”链接，其使用插件或 ActiveX 控件提供的界面对象 40。该界面对象 40 将提供与现有浏览器上的目前的“收藏”或“书签”菜单项和工具条类似的功能。有利的区别在于，由于书签将从因特网上的数据库下载，用户可在任何电脑的有能力装载插件或 ActiveX 控件的浏览器上访问他们的书签。

也可用本发明生成收入，其基于通过插件或 ActiveX 控件在浏览器界面 20 上放置包含于“收藏”或“书签”之内的广告“链接”。消费者的选择（consumer targeting）可基于，比如说，存储于数据库内的信息，例如姓名、年龄、性别、收入、种族、教育与地理位置，以及偏好，例如收藏、书签或者存储于数据库上或存在于浏览器 72 上的其它偏好。

可使用当用户漫游因特网时保持持久的插件或 ActiveX 控件来向浏览器界面 20 添加界面对象 40，其允许用户从位于因特网上的数据库下载他们的“地址簿”。这样的界面对象 40 可直接添加到浏览器界面 20 的界面，并且将允许用户发送电子邮件，以及获取列于他们的“地址簿”之内的保存的联系人信息。

也可用本发明赚取收入，其基于在插件或 ActiveX 控件的界面对象 40 上放置包含于“地址簿”之内的广告“链接”。消费者的选择（consumer targeting）可基于，比如说，存储于数据库内的信息，例如姓名、年龄、性别、收入、种族、教育与地理位置，以及偏好，例如收藏与书签或者存储于数据库上或存在于浏览器 72 上的其它偏好。

本发明可使用插件或 ActiveX 控件来在浏览器界面 20 上添加编辑框，其允许用户在浏览器内直接输入搜索，而不必访问允许用户进行搜索的网页。

5

另外，甚至在用户正在浏览不同的网站时，保持持久的插件或 ActiveX 控件可以从优选的网站轮询，或者周期性地搜索（以用户、网站或程序选择的间隔）信息。当轮询优选的网站时，该优选的网站可以向用户的浏览器上的界面对象发送更新后的信息，例如接收邮件的接近实时的通知，股票价格的持续更新，或其它时间敏感的信息，
10 例如，新闻头条（news feed headline）、所选的喜爱的运动队的比赛得分等等。如果需要的话，优选的网站可以控制轮询的定时，从而在使用高峰期间通过延长轮询之间的时间间隔来控制该优选的网站上的流量。

15

由于库文件创建的外壳，如这里所描述的那样，是这样一个环境，可于其内运行应用程序，或者显示信息，可使用本发明向浏览器的界面添加任何信息或程序。该外壳独立于浏览器界面，浏览器的内容，甚至该外壳本身的内容。简单地说，该外壳是可适配的功能片，极端
20 地，它甚至不必是用户可见的。因此，在使用中，该外壳可以为空，并从网站接收其内容，或者该外壳可取得插件，或者该外壳甚至可以取得新的库文件并且在该外壳从网站或用户接收新内容时“在飞行中”学习解析新信息。这样，本发明提供显著的机会，以将来自优选的站点的所需要的信息指引给用户，甚至在用户访问其它站点时也是如此。当然，网站用本发明的可订制的界面提供的用户特定的功能越
25 多，该网站可产生的用户忠诚度也越大。

在本发明的另一个实施例中，并且现在参照图 11-17，可向因特网浏览器界面的工具条添加购物助理按钮，以便利在所支持的商户网站上在线购物。该实施例的功能，包括定义购物助理按钮，由从服务
30

器传输并存储在用户的电脑上的电脑代码来提供。电脑代码监控因特网浏览器的因特网导航，以判定该浏览器是否在所支持的商户网站上，当该因特网浏览器在所支持的商户网站上时为购物助理按钮提供指示，并填写所支持的商户结账网页。

5

如这里所使用的，术语“在线购物”是指这样的过程，因特网浏览器的用户可通过该过程使用电脑在因特网上购买商品和/或服务。在线商户（例如因特网上的商品和/或服务的销售者）可提供网站，其建立在商户服务器上，用户可使用因特网浏览器通过该网站访问一个或多个网页。商户可通过一个或多个网页提供各种商品和/或服务的销售。购物者可查看商品，服务的描述，选择各种商品和/或服务以放置到电子购物车，查看该电子购物车的内容，并且结账（如下面更详细地描述的那样）。

图 11 中描述的浏览器界面 20 包含界面对象 40，其包括界面对象工具条 42，以及购物助理按钮 144，其向浏览器界面 20 提供各种功能，包括，但不限于，浏览器界面 20 的用户（术语用户与购物者在这里可互换使用）创建和编辑电子钱包的能力（下面更详细地讨论），使购物助理按钮 144（以及浏览器界面 20 的购物功能）失效，罗列所支持的商户，查询关于购物助理按钮 144、下拉菜单 44、与购物助理按钮 144 提供的功能，以及查询关于购物助理按钮 144 的使用及提供的功能的帮助。购物助理电脑代码或软件（例如，.dll 或.exe 文件、javascript 等等）由服务器提供，并存储在用户的电脑 50 上，并可结合浏览器与浏览器界面 20 进行操作，以定义和提供购物助理按钮 144 和遵照本发明提供的其它功能，如下面更详细地描述的那样。

在优选实施例中，当将界面对象 40 添加到浏览器界面 20 时，购物助理按钮 144 及提供的功能被提供，如上面结合本发明的各种实施例所详细地描述的那样。也就是说，当将界面对象 40 添加到浏览器界面 20 时，购物助理按钮 144 及相关联的电脑代码被传输或下载到

用户的电脑 50 上。作为另一种可供选择的方案，可按照用户的请求，也就是在添加界面对象 40 之前或之后，将购物助理按钮 144 及功能选择性地添加到浏览器界面 20。

5 至少在第一个实例中，服务器 102（其可包括一台或多台电脑，其位置在物理上相互靠近，或在物理上相互分离）（参见，例如，图 10）遵照本发明的本实施例提供购物助理功能，当用户在浏览器界面 20 的 URL 地址窗口 26 提供预先确定的因特网地址或 URL 时，可访问该服务器。优选地，服务器 102 是安全服务器（也就是仅可通过安全套接字层（SSL）连接访问）。在该预先确定的因特网地址上，用户可访问一个或多个网页，其中至少一个使得用户能够遵照本发明的本实施例建立钱包。图 12-14 描述了示例性的钱包建立网页，下面更详细地讨论之。

15 服务器 102 下载到用户的电脑 50 上的购物助理电脑代码或软件也提供购物助理功能，其可结合浏览器界面 20 与购物助理按钮 144 进行操作。购物助理代码可以是.dll 或.exe 文件，比如说，javascript，或其它已知类型的电脑代码或软件文件。购物助理代码监测因特网浏览器的因特网导航，其在每次用户促使浏览器导航到不同的因特网地址时，截获因特网地址（或者域或 URL）。购物助理代码也对截获的因特网地址和所支持的商户的因特网地址进行比较，其对截获的地址与包含所支持的商户的因特网地址的所支持的商户文件进行比较。优选地，所支持的商户文件由服务器 102 下载，并存储在用户的电脑 50 上。每次用户登录服务器 102，服务器 102 将其所支持的商户文件的版本与存储在用户的电脑 50 上的版本（例如，用户可在登录过程中传输该版本信息）进行比较。如果有必要，服务器 102 将更新版本的所支持的商户文件下载到用户的电脑。购物助理代码也截获浏览器接收的每个网页，并通过 HTML 代码与 http 请求报头来判定网页的类型（例如，账单网页、商户主页、商品页等等）。

30

购物助理代码还定义购物助理按钮 144 与下拉菜单 44，并控制浏览器、服务器 102 与商户服务器之间的通信，以进行在线购物。

5 购物助理按钮 144 可作为界面对象 40 的一部分提供，其遵照本发明的各种实施例。然而，用户为利用和访问购物助理功能，该用户必须首先建立钱包。如这里所使用的那样，术语“钱包”是指针对用户的文件（其可以是编码的或加密的），其包括为该用户定制的信息（这里也称为数据），其由该用户提供，并存储于服务器 102（参见，例如，图 10）的数据存储设备（例如，硬盘驱动器、光盘等等）上提供的钱包数据库 104 内。组织钱包数据库 106104 的结构，使得用户
10 在建立钱包时提供的信息存储于预定义的字段或字段名内。类似地，使用同样的字段或字段名来组织所支持的商户文件的结构，使得商户结账网页要求与在钱包建立过程中提供的相同的信息。

15 为建立钱包，用户必须提供特定信息，作为对服务器 102 提供的特定钱包建立网页的响应。本领域技术人员知道，因特网浏览器从服务器接收数据，典型地为 HTML 数据，用于浏览器界面 20 的显示。因此，这里的对服务器提供的网页的引用是指从服务器下载或传输数据，用于因特网浏览器的显示。

20

图 12 描述了示例性的钱包建立网页，其一般地指示为 1000。可以，比如说，通过购物助理按钮 144 的下拉菜单 44 上的“编辑钱包”选项来访问该网页 1000。通过在钱包建立网页 1000 上选择预先确定的链接，比如说，“建立你的钱包”链接 1002，服务器 102 下载钱包数据条目网页 1010，例如如图 13 所描述的那样。在钱包数据条目网页 1010，用户在多个字段 1012（下拉菜单或字母数字条目）内输入
25 特定信息。该信息可以是，但不限于，信用卡类型、号码、过期日期、用户的名和姓、账单地址、电话号码、用户 ID、用户密码和针对该用户特定的各种其它信息。当建立钱包时，除用户的 ID 和密码之外，
30 还要求每个用户输入安全钥匙（参见，例如，图 16），其为必需项，

以访问和使用该用户的钱包。当用户完全填写钱包数据条目网页 1010 上的所有必需的字段 1012 时，用户可选择“结束”按钮（未显示），其向服务器 102 发送用户输入的信息。作为响应，服务器 102 下载钱包小结网页 1020，其包括用户的新近输入的账号信息的小结 1022，
5 以在最后建立用户的钱包之前供用户回顾，如图 14 所描述的那样。如果用户信息是正确的，用户可选择钱包小结网页 1020 上的链接，例如，“点击此处继续……”链接 1024，其促使用户信息（数据）被发送到服务器 102 并存储到数据存储设备 106 上的钱包数据库 104 内。在方才描述的钱包建立过程期间，数据在服务器 102 与用户的电脑 50
10 之间传输（参见，例如，图 10）。例如，当用户选择“建立你的钱包”链接 1002 时，网页 1000（图 12）和 1010（图 11）均可被服务器 102 发送，以供用户的浏览器界面 20 显示。当用户已完成钱包数据条目网页 1010 上必需的信息并选择“结束”按钮（未显示）时，可将用户信息或数据发送到服务器 102。当从用户的电脑 50 接收到用户信息
15 时，服务器 102 可发送钱包小结网页 1020。作为另一种可供选择的方案，服务器 102 可分别发送每个上面所描述的网页，作为对用户发起的行动的响应（例如在用户选择链接或按钮时）。

服务器 102 可属于内容提供商 100 或其它因特网服务提供商，例如，
20 搜索引擎提供商或 ISP，这里统称为提供商 100。提供商 100 选择在线商户，本发明的购物功能可结合其使用。所支持的商户文件包含所有所支持的商户的列表，包括它们各自的因特网地址。购物助理电脑代码可使用该文件来判定用户是否已导航到所支持的商户网站，其比较输入到 URL 地址窗口 26 的 URL（或由链接提供）与所支持的
25 商户文件内包含的 URL。

一旦已选择商户（在图 10 中商户被一般地指示为 200），提供商 100 判定该商户的结账过程，包括结账网页的布局（参见，例如，图 17），以及完成该商户的结账过程与结账网页所必需的信息。提供商 100
30 为每个选中的商户（这里也称为所支持的商户）创建所支持的商户规

则与映射文件，其定义多个商户字段名，其对应于商户在其结账过程与其结账网页期间所要求的用户数据。例如，并且参照图 17，所支持的商户结账网页 2100 要求购物者提供账单信息，包括，比如说，支付方式、信用卡号、过期日期以及该购物者的个人信息（例如，姓名、地址与电话号码）。商户字段名对应于定义在用户的钱包内的钱包字段名。由于每个商户在其结账过程期间可能要求不同的信息，每个所支持的商户规则与映射文件可包含不同的商户字段名。然而，所有商户字段名对应于钱包字段名。

所支持的商户规则与映射文件将钱包数据（也就是钱包字段名）映射到必需的商户数据（也就是商户字段名）。例如，可组织商户规则与映射文件内定义名字段与姓字段的条目的结构如下：

```
rmi_pageFields[1] = new rmi_fieldMap("contactFirstName", "text",  
"wallet_b_fname_first");  
rmi_pageFields[2] = new rmi_fieldMap("contactLastName", "text",  
"wallet_b_fname_last");
```

其中“wallet_b_fname_first”是用于用户的名的钱包数据库字段名。然而，在所支持的商户文件中，用户的名被标识为“contactFirstName”。因此，本发明将钱包数据库内的字段映射到所支持的商户文件内的字段，使得当用户在所支持的商户网站上购物时，每个商户订单表格可被自动填写。

现在参照图 15-17，现在将描述示例性的结账过程。由于不同的商户可能有不同的结账过程，并提供不同的结账网页，下面的描述仅是阐述性的，而无意限制本发明的范围，或以其它方式定义本发明。另外，下面的描述不追踪整个在线购物过程（其也可能在不同商户之间有所不同），而仅是结账过程，一般地，其开始于用户在商户网页上选择“结账”或其它类似的按钮之后。

当用户启动他们的因特网浏览器时，界面对象 40 在用户的电脑

50 与服务器 102 之间自动建立连接。服务器 102 将各种用户数据下载到用户的电脑 50。用户可通过在 URL 地址窗口（参见，例如，图 11）内输入地址，或通过选择网页上的链接，促使因特网浏览器导航到任何因特网地址。无论如何促使浏览器导航因特网，购物助理代码截获
5 用户导航的每一个因特网站点的 URL，并将该 URL 与所支持的商户数据文件进行比较。如果找到匹配，购物助理代码在购物助理按钮 144 上提供用户可察觉的指示符，例如黄色的圆圈，其指示用户已导航到所支持的商户网站。购物助理代码也请求服务器 102 下载这样的文件，其包含该商户的规则与映射数据，并提供该商户的结账过程的细节。

10

购物助理代码也截获浏览器接收的每个网页，以通过查看 HTML 代码与 http 请求报头来判定网页的类型，例如，账单网页、电子邮件地址请求、信用卡信息请求等等。该信息在提供商选择商户以及创建所支持的商户数据文件期间获得。使用商户的规则与映射文件，购物
15 助理代码能够判定特定页面被提供，以及应该如何在该页面内提供用户的信息。

20

如果购物助理代码判定用户的钱包对特定页面是必需的，它在用户的浏览器界面 20 内打开安全钥匙窗口 3000，请求用户在数据条目窗口 3020 内输入安全钥匙（SK），如图 15 所描述的那样。一旦用户
_ 20 输入 SK，并选择“继续”按钮 3030，用户的电脑 50 向服务器 102 安全地发送该安全钥匙，其优选地使用 https。服务器 102 将用户数据（例如，ID 与 SK）与先前存储在钱包数据库 106 上的该用户的钱包内的数据进行比较。如果数据匹配，则 SK 被验证通过，并且服务器 102
25 向购物助理代码发送认证；如果数据不匹配，则发送拒绝。认证可作为钱包信息窗口 3100 来提供，例如如图 16 所描述的那样。当服务器 102 成功地验证通过 SK 时，通过 SSL 向安全服务器发送“安全 cookie”，并且当该“安全 cookie”被设置时，该用户使用他的/她的钱包的在线购物会话开始。“安全 cookie”会主动地过期（aggressively time out），
30 优选地，在不超过一（1）小时内有效。

如果用户想要使用本发明的购物助理功能继续结账过程，该用户可选择“继续”按钮 3130。如果用户想要使用商户的结账过程结账，该用户可选择“取消”按钮 3140。当用户选择“继续”按钮 3130 时，购物助理代码从服务器 102 上的钱包数据库 104 获取用户的钱包，并且自动地填写商户的结账网页 2100，如图 17 所描述的那样。服务器 102 安全地发送用户的钱包，其优选地使用 https。用户的钱包，以及用户的个人数据，无需存储在用户的数据存储设备 52 上，而只需存储在临时性的内存（这里也称为主内存或 RAM）60 内，通过并结合本发明的本实施例的购物助理代码与功能使用。用户的数据可自动地过期，或者可在用户关闭浏览器时将其从内存 60 清除。

依赖于浏览器显示的网页的类型，购物助理代码从用户的钱包（RAM 内的临时性的数据）向该网页提供适当的用户数据。例如，如果购物助理代码判定网页是“账单地址”网页，将从 RAM 拷贝该用户的账单地址数据，以填写该账单地址网页内的字段，并最终发送给商户的服务器。在结账过程期间，购物助理代码结合商户规则与映射文件和用户的钱包进行操作，以自动地提供各种商户结账网页所要求的用户数据。

钱包数据库 106 可以仅包含钱包数据，或者作为另一种可供选择的方案，数据库内可包含钱包数据和其它数据。虽然优选地，钱包是通过并结合购物助理按钮 144 与功能使用的，钱包也可由购物助理之外的其它应用程序，例如，账单支付，无购物助理的购物，并结合基本上任何要求用户输入个人数据的在线交易来使用。钱包提供安全传输和对用户的数据的处理，并使得用户可结合各种在线交易来使用该数据，而无需不必要地危及该数据的安全性。

一旦用户建立钱包，遵照本发明的本实施例提供的功能现在作为用户在线商店对用户开放。优选地，无论用户（通过因特网浏览器）

连接到哪个因特网站点，在线购物功能对结合所支持的商户的使用开放。因此，用户无需连接到提供商的因特网站点，甚至所支持的商户的网站。购物助理代码处理用户（用户的电脑 50）、服务器 102、钱包数据库 104 与商户的因特网站点和结账网页之间的所有通信。

5

遵照本发明的另一个实施例，提供商 102 之外的一方可将按钮 44（也就是界面对象）添加到浏览器界面工具条，例如界面对象工具条 42（参见，例如，图 4）。例如，可在公司网站上提供链接，使得该网站的访问者能够将按钮 44 添加到他们的界面对象工具条 42，该按钮将提供从浏览器界面 20 到该公司网站的单击链接。由于将公司按钮添加到界面对象工具条 42 使得用户能够通过单击鼠标从任何其它网站返回到该公司的网站，公司或维护网站的任何实体，可赞助提供商 100。这样，将公司按钮添加到浏览器界面（也就是工具条 42），在用户和公司网站之间建立起亲和力。公司赞助商设计按钮（例如，外观（单击按钮或下拉菜单）、图标等等），并将按钮设计与 URL 数据提供给提供商 100，以集成到该提供商维护的数据库，并与浏览器界面相关联。这样，公司赞助商可拥有定义于提供商数据库中的按钮。

15

如果公司网站的访问者想在他们的浏览器界面 20 内安装公司按钮，但是没有安装浏览器界面工具条 42，本发明也允许在浏览器界面 20 内安装浏览器界面工具条 42 与公司按钮 44。另一方面，公司按钮 44 可添加到浏览器界面 20 的工具条 22，或者可作为界面对象工具条 42 的按钮 44 添加。

20

本发明还可作为教导工具使用。例如，界面对象工具条 42 可包括多个按钮 44，其可被促使改变颜色、闪烁、高亮等等，以将用户的注意力指引到该按钮或该按钮提供的功能。

25

本发明还可允许提供商或公司赞助商在用户访问竞争网站时，将该提供商或公司赞助商的特定属性通知给用户。例如，当用户访问汽

30

车网站时，相应的提供商属性（例如，提供商 汽车）将高亮，以提醒用户该特性可在该提供商网站上获得。可在用户的电脑上提供软件，以在每次用户促使浏览器导航到网站时，截获 URL。截获的 URL 可与包含多个 URL 的文件进行比较，以用于提供商属性。

5

重要地，用户的电脑单独判定在任何特定时刻用户导航到哪个网站。因此，没有向任何服务器发送关于用户在因特网上的活动的信息。

一般地，需要两个步骤来向用户的商户列表添加商户。第一，用户必须通过商户的网页建立商户的在线账号。这样做时，用户可能被请求输入用户登录标识符、密码与账号编号。一旦商户接受该数据，即为用户建立起在线账号。第二，用户必须将在线商户账号链接到安全数据库内用于该用户的数据。例如，用户可访问这样的网页，可通过其来选择商户，并输入对该用户和商户特定的数据（例如，该用户的登录标识符、密码、账号号码等等）。之后，该数据可被验证（通过与商户网站通信）并为该用户存储到安全数据库。这一（这些）步骤可使用另一网页（也就是可以不与该商户相关联的网页）执行。一旦完成这两个步骤，用户可向商户列表添加商户，并通过简单地从商户列表选择商户来实现自动登录到该用户在特定商户上的账号。

10
15
20

每次用户激活（启动）他的/她的浏览器，遵照本发明提供的电脑代码将自动地与预先确定的因特网站点通信，以获取特定于该用户的数据。可全部地或部分地使用该获取的数据，以向浏览器提供功能，并向浏览器界面添加界面对象。这样，获取的数据将定义自动登录界面对象，包括最新的商户列表。

25

接着参照图 18A-18B，现在将详细地讨论本发明的自动登录方法的优选实施例。图 18A-18B 描述的、并在下面详细地讨论的方法，认为自动登录界面对象已被添加到浏览器界面，如上面所详细地描述的那样。该方法也认为遵照本发明提供自动登录（与其它）功能的电脑

30

代码已被传输到用户的电脑，并且可结合其处理器和结合用户的因特网浏览器进行操作。在步骤 1810，用户从通过自动登录界面对象提供的商户列表中选择商户，该对象可包括按钮与下拉菜单。在步骤 1820，电脑代码（也称为客户端）促使浏览器打开隐藏窗口，自动登录过程

5 将于其内执行。电脑代码也向服务器，优选地为安全服务器，传输第一请求，其包括商户标识符与 cookie，如图 19 描述的、并在下面更详细地讨论的那样。在步骤 1822，安全服务器接收第一请求，并判定 cookie 是否是安全 cookie。如果 cookie 不是安全 cookie，则用户尚未为当前的因特网会话进行安全地登录，或者自用户安全地登录后已过去预先

10 确定的量的时间。在任一种情形下，在步骤 1830，安全服务器向用户的浏览器传输一个或多个网页（其作为新的浏览器窗口在用户的浏览器内打开），提示用户输入特定数据，以安全地登录。用户必须输入的数据之一是安全钥匙，在步骤 1832，安全服务器接收其并判定其是否是该用户的正确的和/或有效的安全钥匙。如果输入不正确的安全钥匙，在步骤 1834，安全服务器判定用户尝试重新输入安全钥匙的次数

15 是否已超过预先确定的次数。如果用户已超过允许的重试次数，安全登录过程将终止（由安全服务器终止），并且用户的浏览器将返回显示在用户从自动登录界面对象（按钮）的商户列表中选择商户之前显示的网页，如步骤 1850。如果用户尚未超过允许的重试次数，并且尚未选择取消安全登录过程，在步骤 1836，用户可尝试重新输入安全钥匙；在步骤 1832，安全服务器再次判定是否已输入正确的安全钥匙。如果，经过预先确定的尝试次数之后仍未输入正确的安全钥匙，或者用户通过选择取消按钮来选择终止安全登录过程，安全登录过程将终止（由安全服务器终止），并且用户的浏览器将返回显示在用户从自

20 动登录界面对象（按钮）的商户列表中选择商户之前显示的网页，如步骤 1850。

25

如果用户成功地输入正确的安全钥匙，安全服务器向电脑代码传输批准消息，步骤 1840，该电脑代码之后将向供给服务器传输第二请求 1842，其请求特定于所选商户的用户的证书。第二请求可包含用户

30

标识符与商户标识符，其可由供给服务器进行解释，以在数据库内定位用户和用户的证书。如果供给服务器将第二请求匹配到数据库条目，则供给服务器将向用户的电脑传输用户的证书（例如，用于该商户的登录标识符与密码，用于该商户的网站标识符（例如 URL），和
5 商户规则文件）。电脑代码接收证书，并促使用户的浏览器导航到该商户的网站（如 URL 所确定的那样），并自动地用用户的登录标识符和密码并根据商户规则文件来填写登录字段，从而实现了用户到商户网站的自动登录。

10 遵照本发明提供的电脑代码以透明的方式执行上面所描述的自动登录过程。另外，本发明提供这样的能力，其用于控制浏览器界面，从而控制用户的导航，其以这样一种方式，用于控制导航（在此情形中，于自动登录到商户网站）的规则，可在服务器端改变，而无需在客户端改变。本发明可有利地利用浏览器可获得的文档对象模型
15 （DOM），以实现如方才所述的规则变化。那就是说，在用户从商户列表选择商户之后，电脑代码执行自动登录过程，而无需来自用户的进一步的输入（用户没有安全地登录的情形除外）。继续参照图 18A-18B，将更详细的讨论遵照本发明提供的电脑代码执行的自动登录。在步骤 1844，电脑代码判定其是否已接收到证书，其作为对第二请求
20 的响应。如果没有，电脑代码生成错误消息，其由用户的浏览器显示，步骤 1860。之后，电脑代码促使隐藏的窗口（在步骤 1820 打开）关闭，并且使自动登录过程终止。浏览器内的显示之后返回到在用户从商户列表中选择商户之前显示的网页。如果电脑代码已接收到证书，如在步骤 1844 所判定的那样，在步骤 1870，电脑代码启动另一个隐
25 藏窗口，以执行自动登录过程的剩余部分。在自动登录过程完成之前的任何时刻，用户可取消该过程，其选择电脑代码显示的合适的按钮，例如当电脑代码执行自动登录过程时显示的取消按钮。如果用户选择取消自动登录过程，在步骤 1872，电脑代码将关闭在该过程期间打开的所有隐藏窗口，并且浏览器的显示返回到在用户从商户列表中选择
30 商户之前显示的网页。如果用户没有手动地终止自动登录过程，如果

在该过程完成之前已过去预先确定的量的时间，电脑代码可能取消该过程。例如，在接收到证书之后，由于设备或网络问题，电脑代码可能不能促使浏览器导航到商户网站。在此情形下，电脑代码将终止自动登录过程，步骤 1874，并向用户显示错误消息：不能完成自动登录，
5 步骤 1878。在这一点，电脑代码可向用户提供机会，以再次尝试自动登录过程，步骤 1878，在该点过程返回到步骤 1844，并且如这里所描述的那样进行。

如果电脑代码判定自动登录过程尚未自动地超时，步骤 1874，
10 则电脑代码判定其是否已从商户网站接收到登录确认，步骤 1876。为做到这一点，电脑代码评估隐藏窗口内接收的 HTML 数据，并寻找预先确定的一个词、多个词、消息或其它指示符，其指示到商户网站上的登录已经成功。例如，在用户成功登录后，商户网站可显示一条消息。该消息可作为商户规则文件的一部分被包含，该文件作为证书的一部分由供给服务器传输到电脑代码。这样，电脑代码能够判定自动
15 登录是否已经成功。如果成功，电脑代码在主浏览器窗口内显示来自商户网站的网页，步骤 1890，从而有效地促使用户的浏览器导航到商户网站，并且直接到用于该用户的账户的网页。

接着参照图 19，那里描述了一种系统，其用于便利用户到网站的自动登录。该系统包括服务器，其可以是单独的服务器，或者多个
20 服务器 1900、1920、1940，其作为设计选择的日常事务。在优选实施例中，该系统包括服务器，其具有数据存储设备，并且其具有存储在那上面的软件。该软件将电脑代码传输到用户的电脑，以向用户的因特网浏览器添加自动登录功能，如这里详细地描述的那样。
25

在图 19 描述的实施例中，安全服务器 1900 结合电脑代码执行安全登录过程。安全服务器 1900 具有数据库，其具有多个用户数据及相应的安全钥匙信息。安全服务器 1900 可从用户接收 cookie，并判定该 cookie 是否是具有有效的安全钥匙的安全 cookie。如果是这样，
30

安全服务器 1900 可向用户的电脑上的用户代码传输确认或认证。

供给服务器 1920 具有数据库，其具有多个用户数据，包括商户登录与密码数据，和商户规则数据，其用于每个商户，由用户提供。

5 当用户电脑上的电脑代码从安全服务器接收到确认或认证时，电脑代码向供给服务器发送第二请求，其包括 cookie，具有用户标识符与商户标识符。供给服务器判定在用户标识符与该供给服务器上的数据库内确定的商户之间是否存在匹配，并且，如果是这样，将证书传输到用户的电脑。在优选实施例中，证书包括用户登录标识符、密码以及

10 商户规则文件，其均特定于特定商户。电脑代码使用该证书来促使用户的浏览器导航到商户的网站，并且到该站点上的登录网页，以自动地填写登录字段（登录标识符与密码），并且根据商户规则文件实现登录（例如选择“登录”或其它类似的按钮）。在这一点，用户的电脑可通过因特网连接到商户的服务器 1940。一旦电脑代码判定自动登录

15 已经成功，浏览器显示用于该用户的账号的商户网页。

对于本领域技术人员而言，并且根据这里提供的公开信息，显然的是，可使用在用户的电脑上和各种服务器上提供的软件来执行本发明的各种实施例。

20

需要知道的是，可利用任何数量的电脑技术来实现本发明。例如，尽管结合因特网来公开呈现各个实施例，可以在任何电脑网络上使用本发明，包括，比如说，广域网。类似地，用户电脑 50 是可以连接到网络的任何设备，包括，比如说，个人数字助理、支持上网的

25 蜂窝电话、拨号到网络的固定电话、移动电脑、个人电脑、因特网设备等等。进一步地，这里描述的服务器可以是任何类型，运行任何软件，并且这里描述的软件模块、对象与插件可以用任何编程语言写成。最后地，这里描述的数据库与存储设备可使用任何存储技术，包括，比如说，本地电脑内存，网络存储，以及任何已知的存储介质，例如

30 磁（介质）或光（介质）。

这样，尽管显示、描述和指出了本发明的基本的新奇的特性，如适用于优选实施例的那样，需要知道的是，本领域技术人员可对所公开的发明的形式与细节进行各种删节与替代，而不偏离本发明的精神。因此，其仅受所附的权利要求书的范围的限制。

5

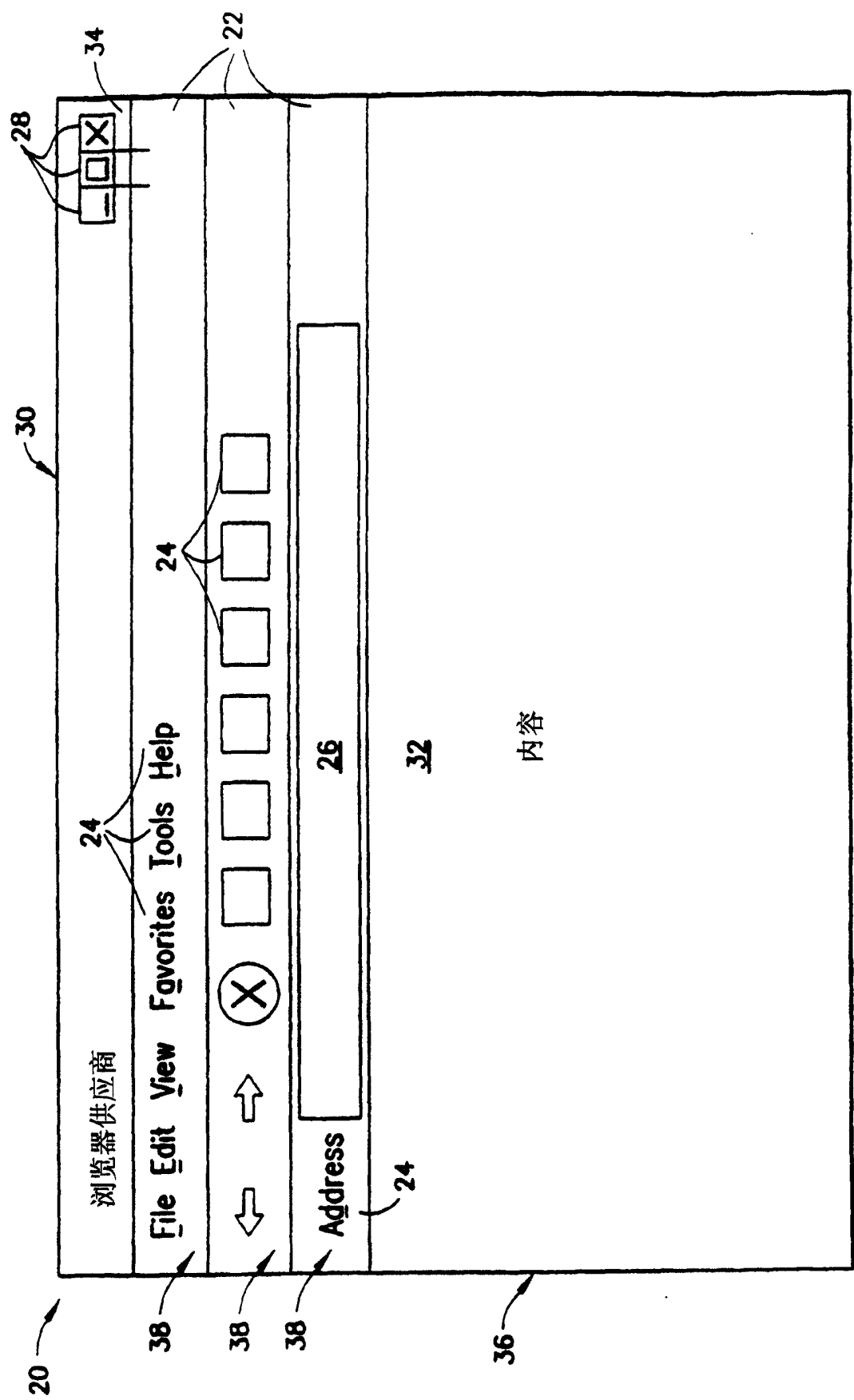


图1
现有技术

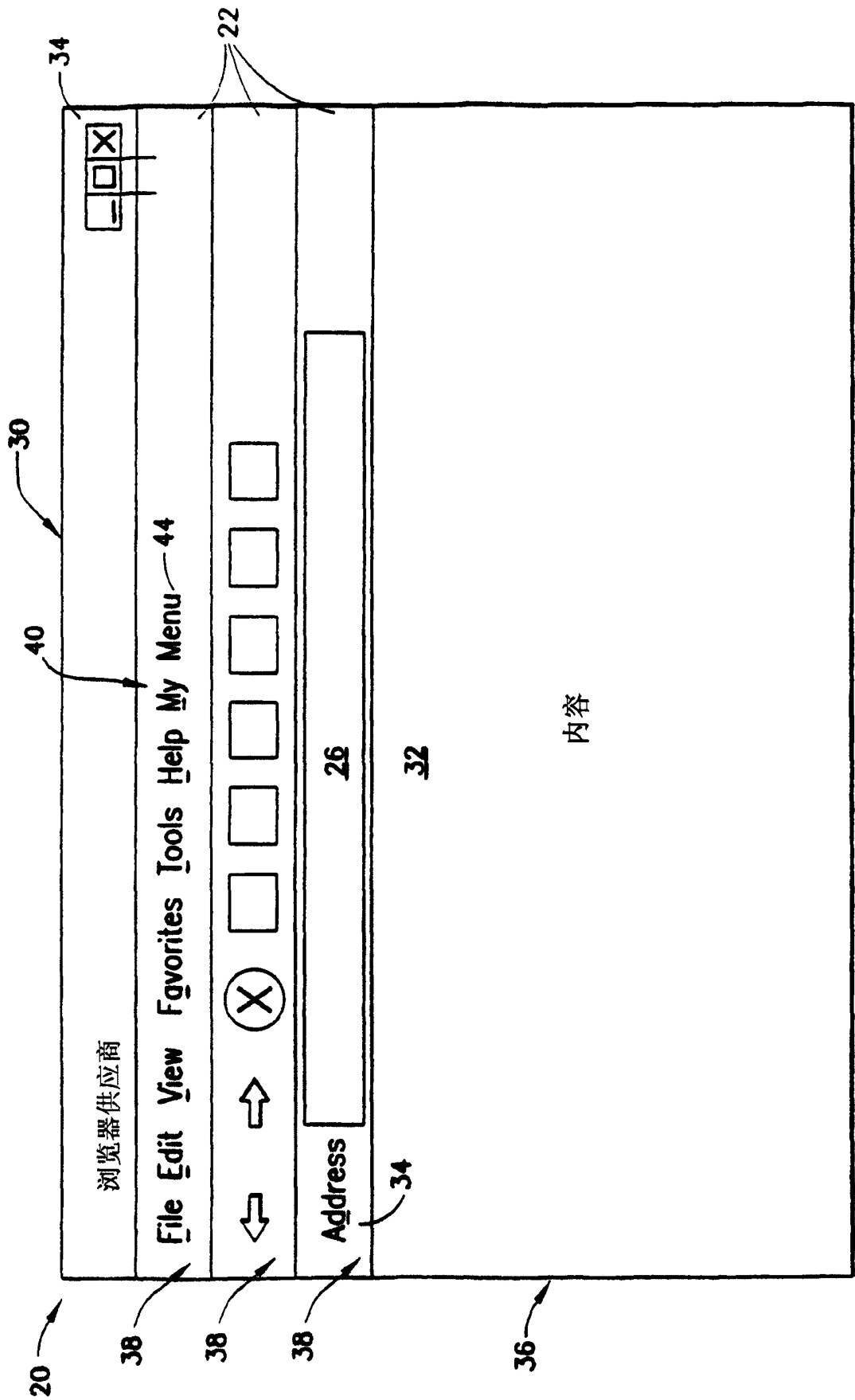


图2

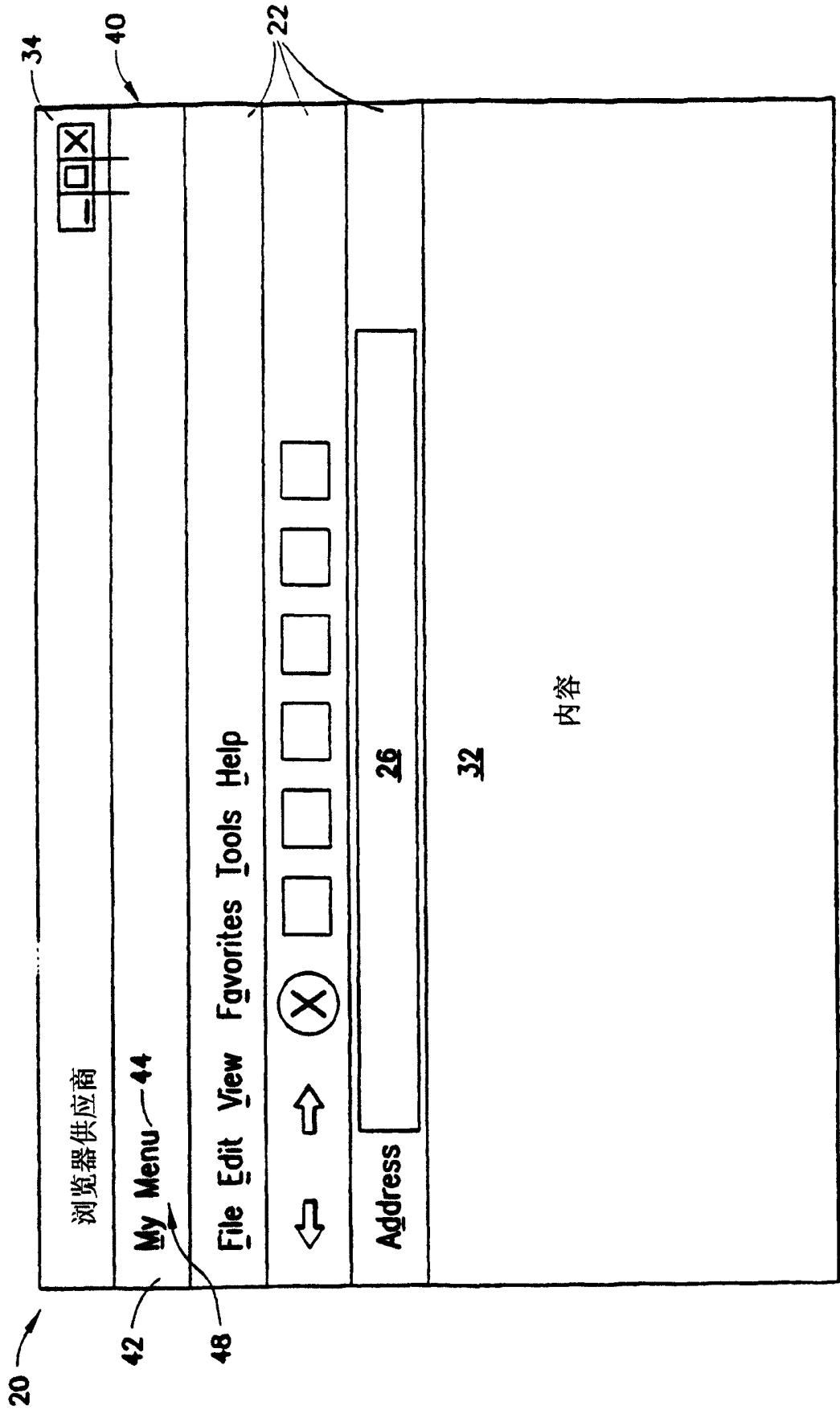


图3

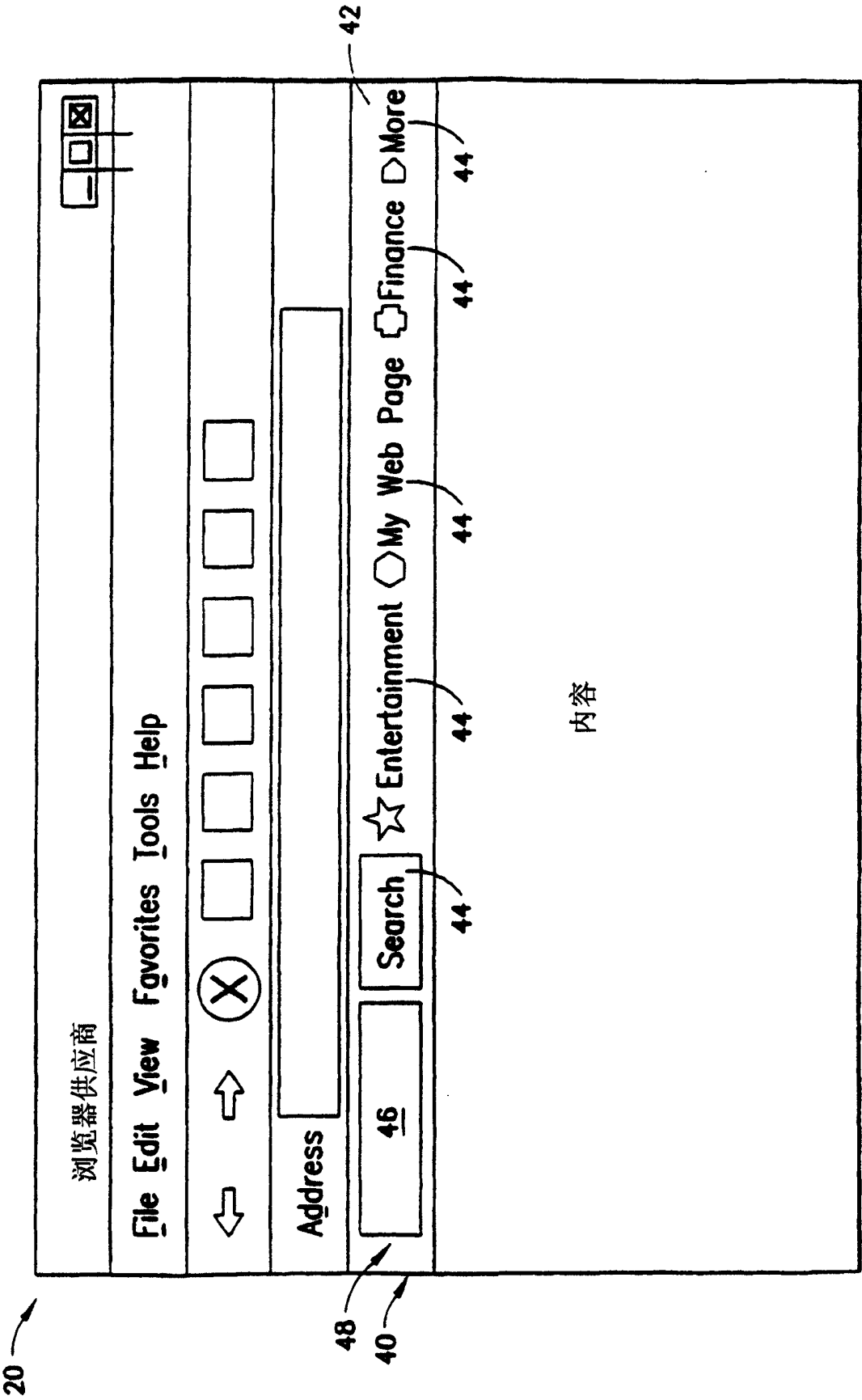


图4

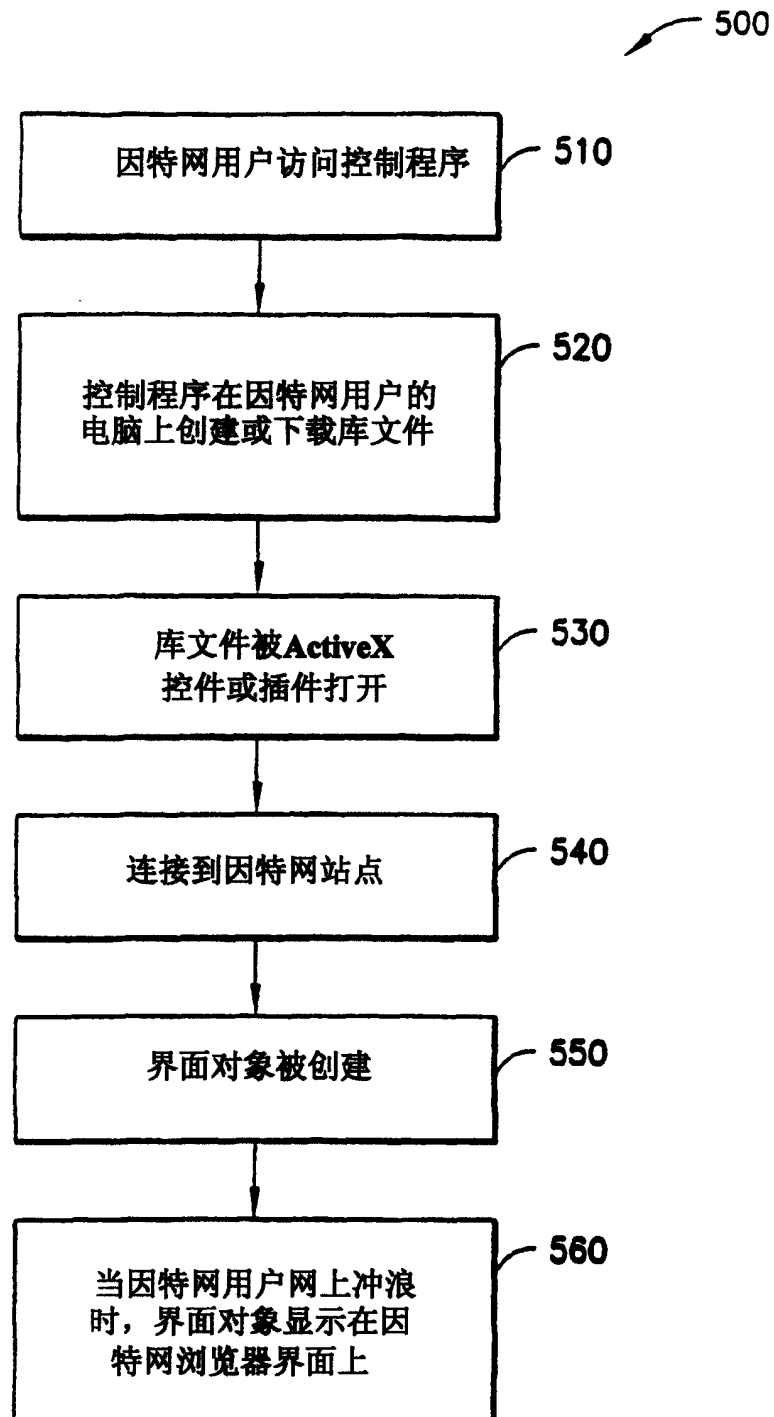


图5

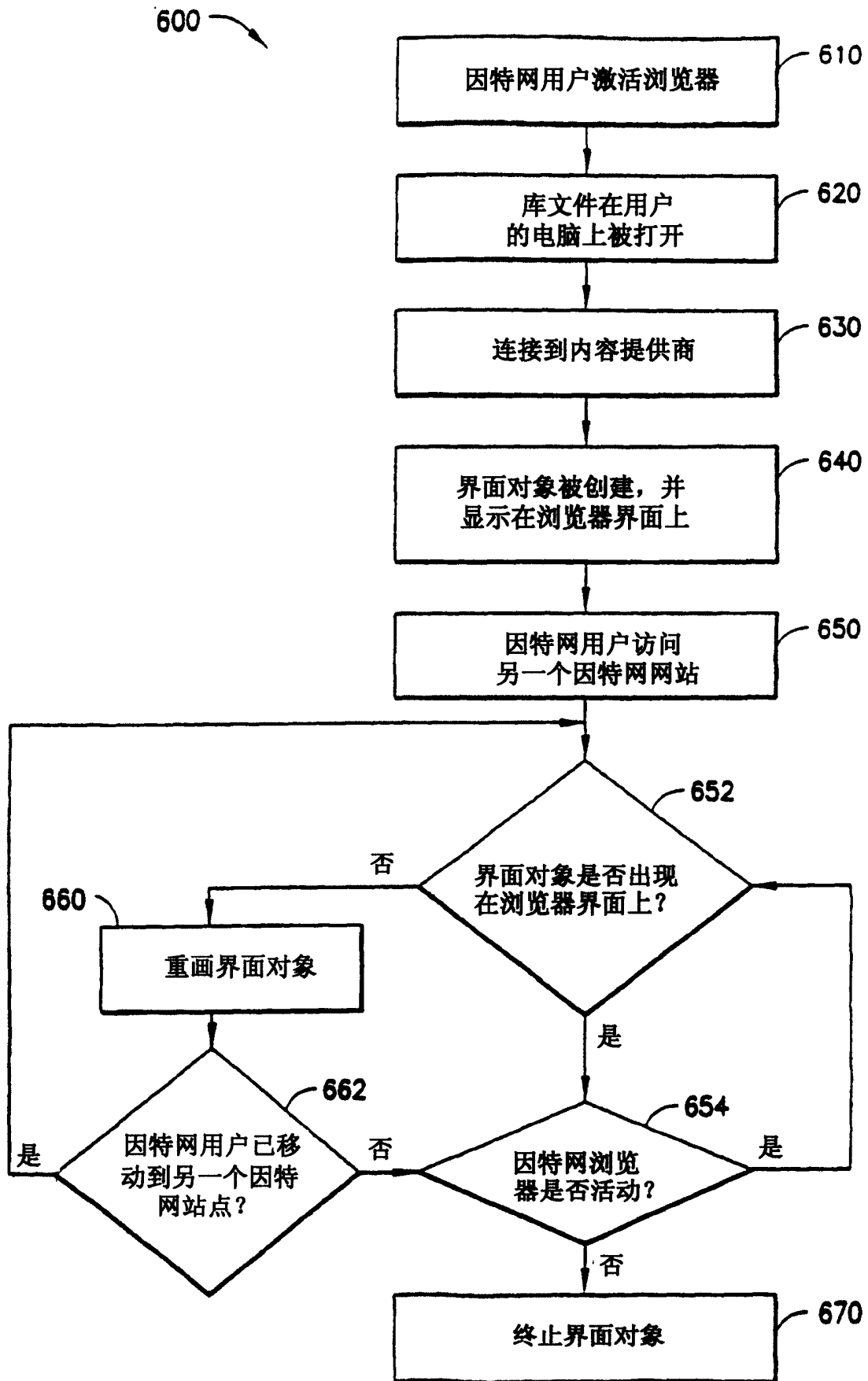


图6

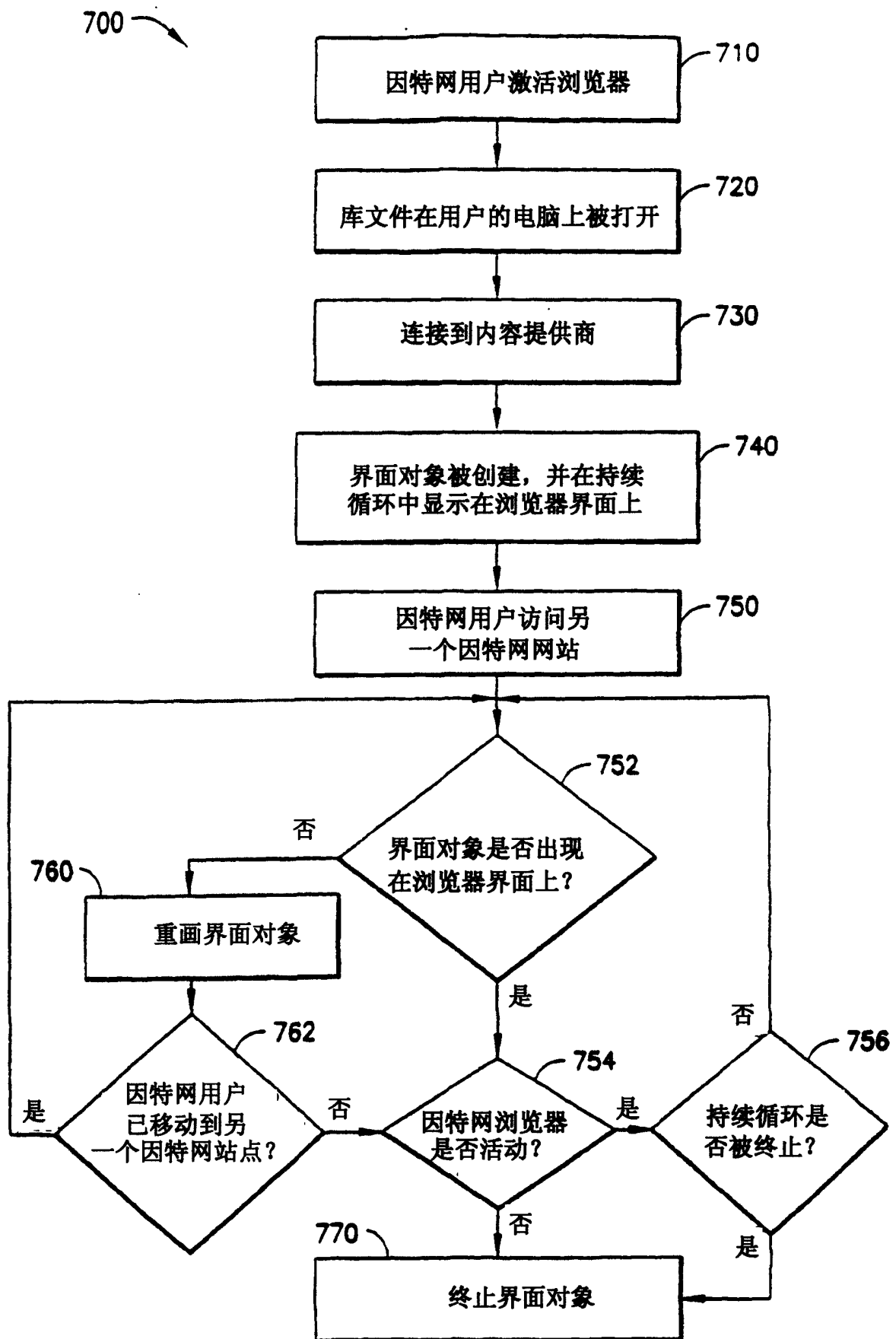
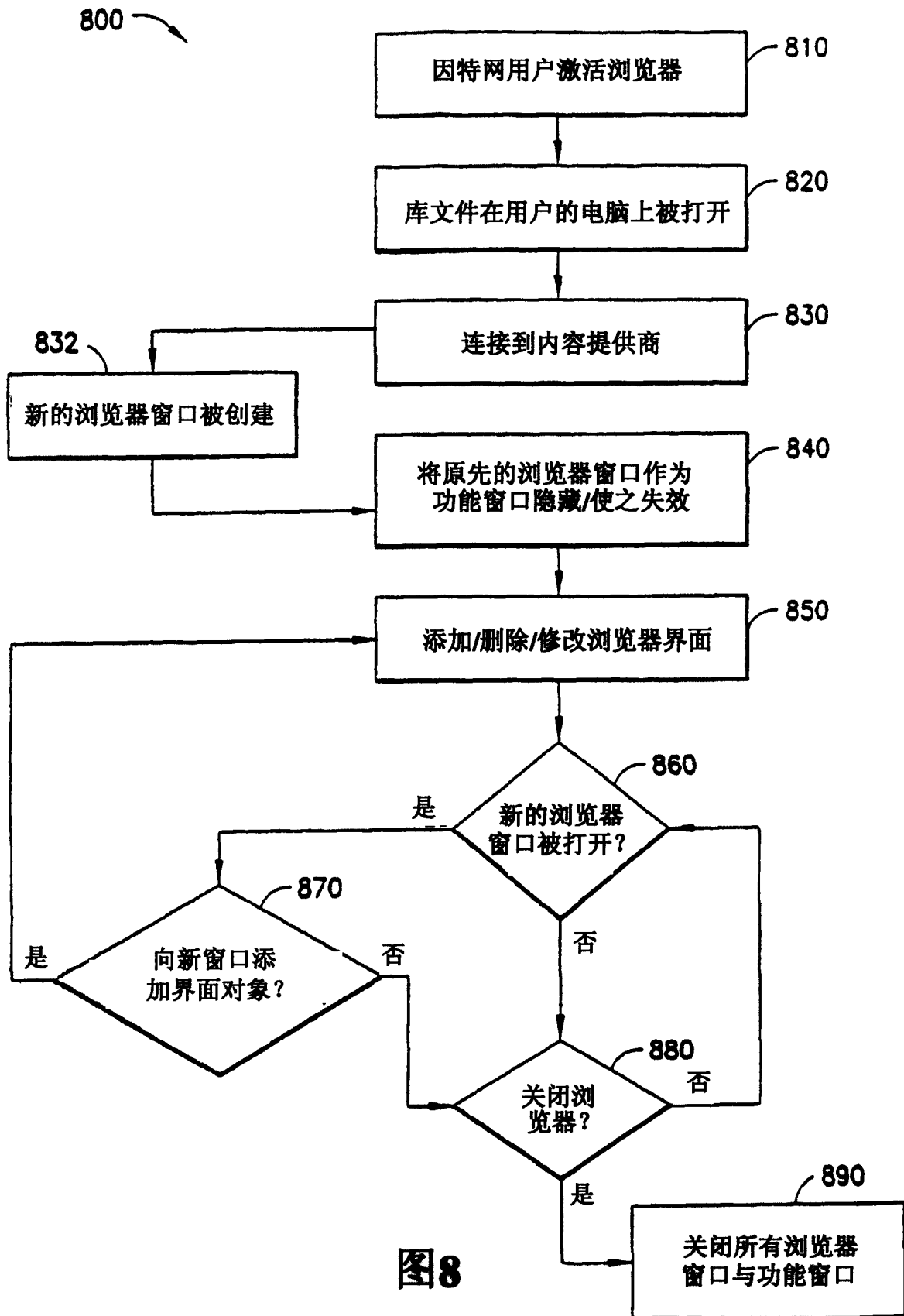
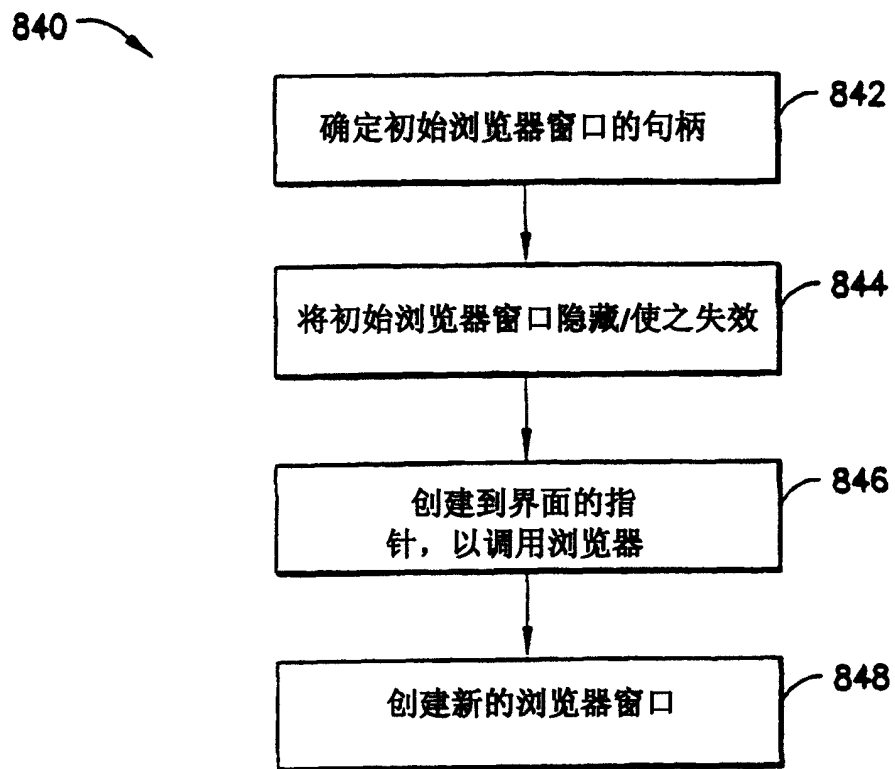


图7



**图8A**

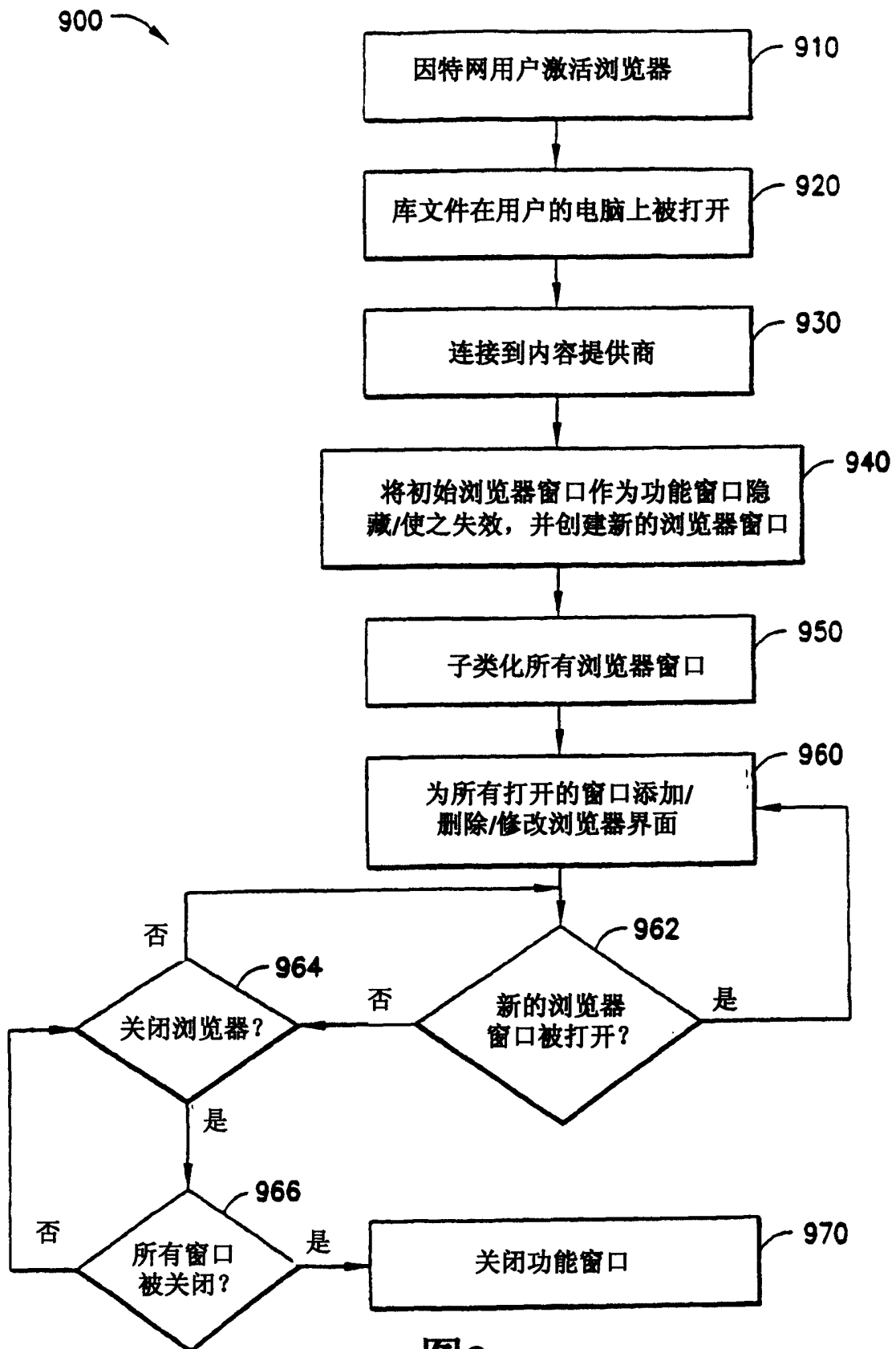


图9

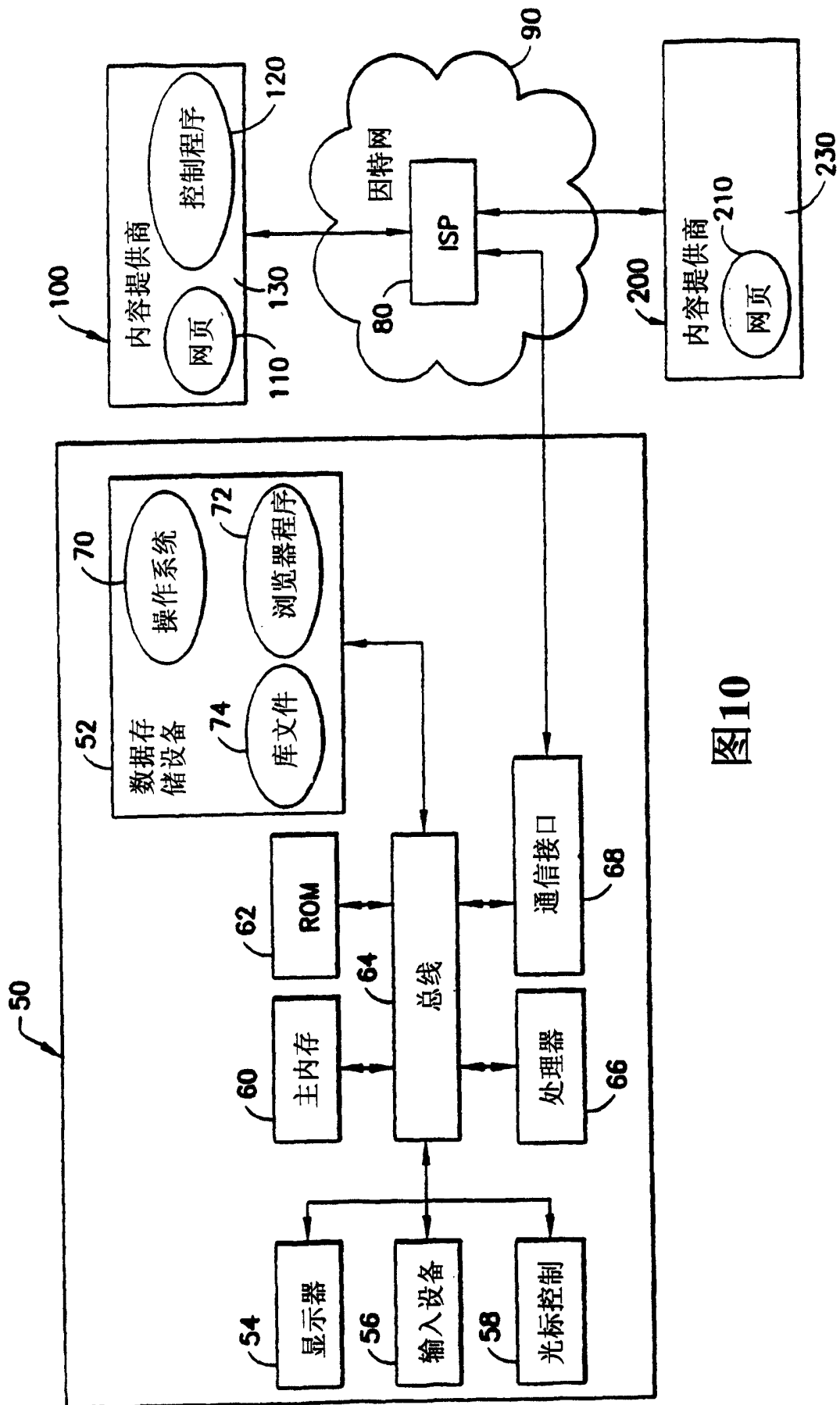
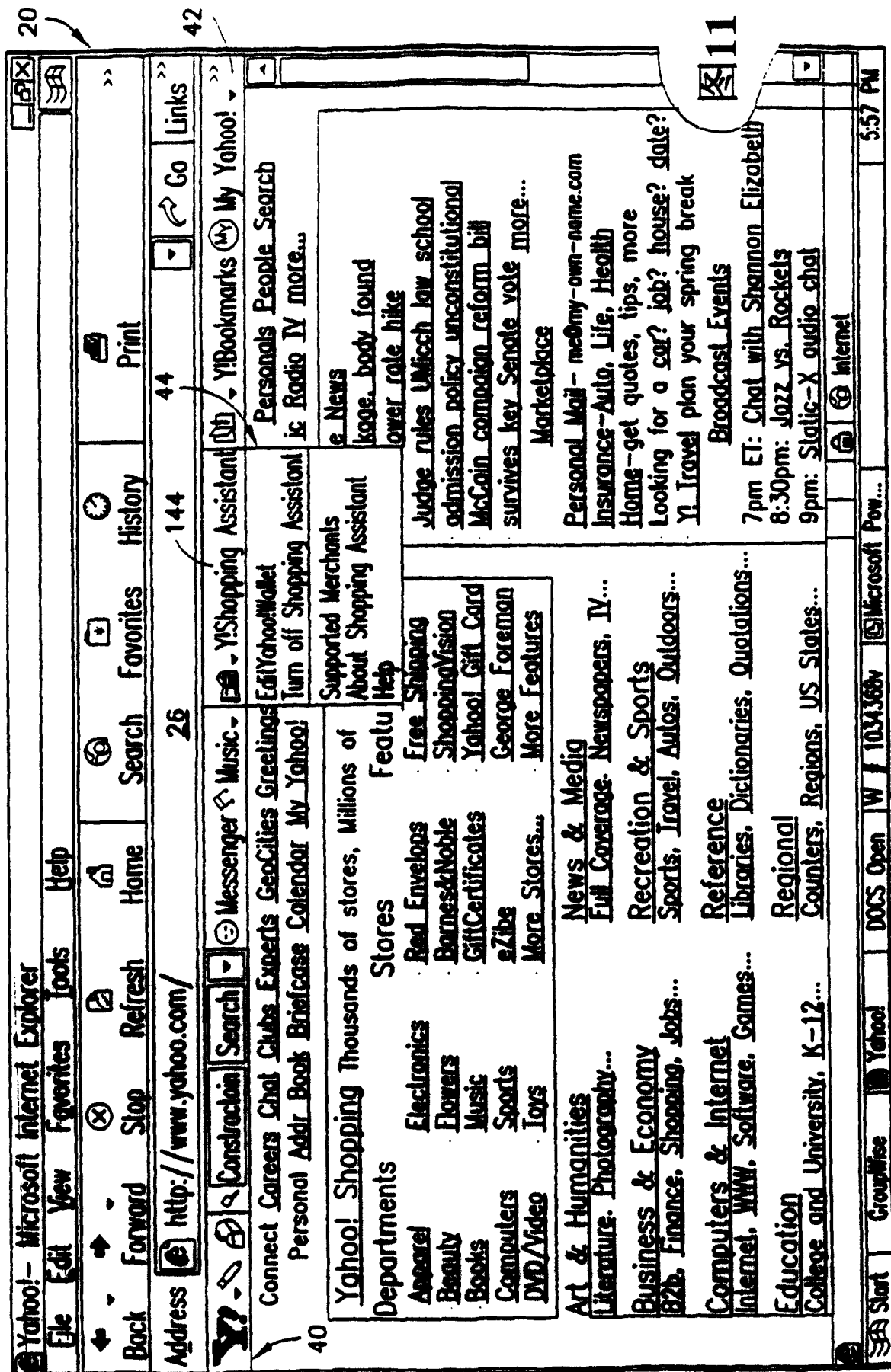


图10



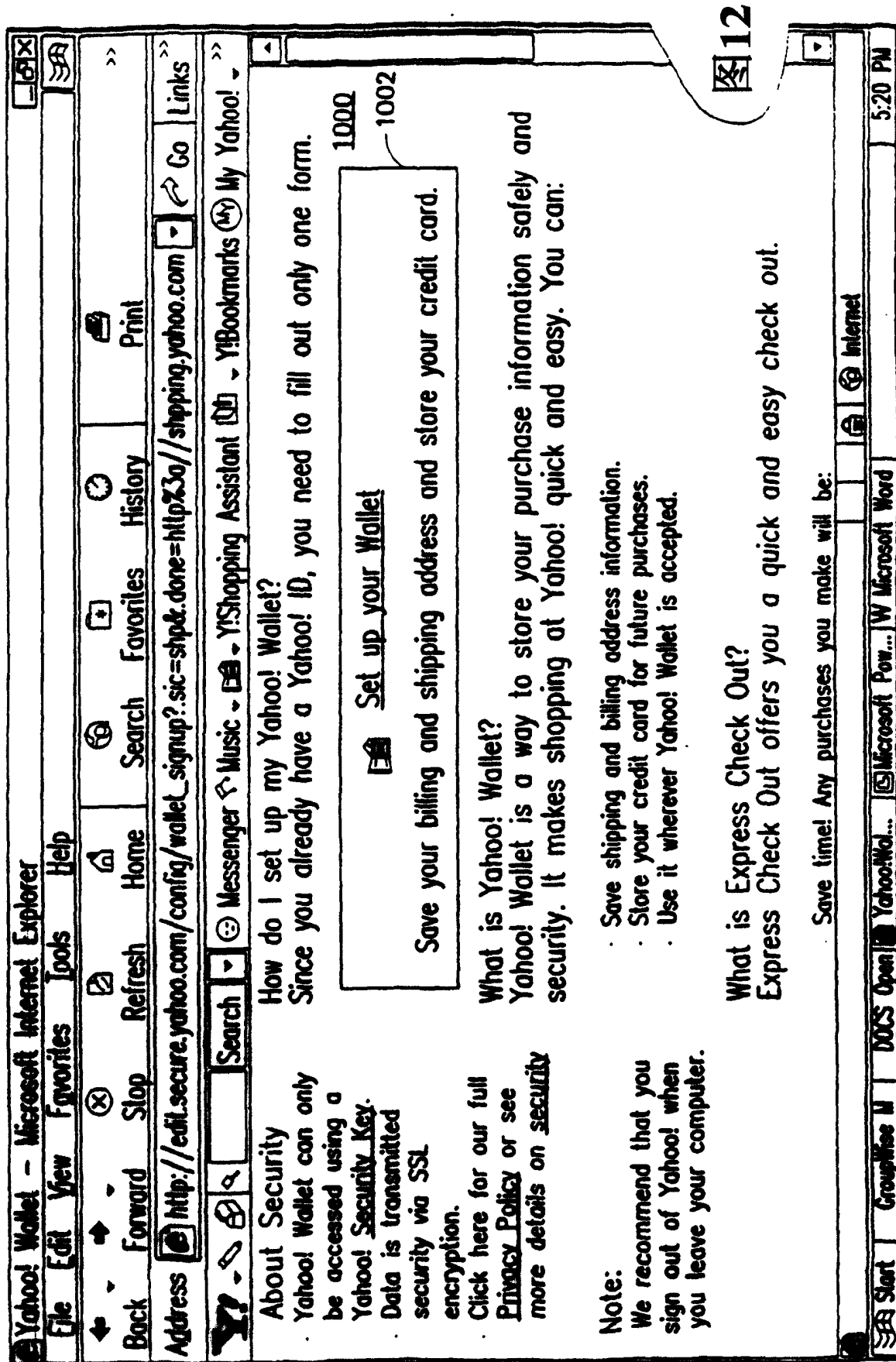


图12

Welcome to Yahoo! Wallet - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Print

Address http://edit.secure.yahoo.com/config/wallet_register?sc=shp&done=http%3a//shopping.yahoo.com=0&t... Go Links

Search Messenger Music Yahoo! Shopping Assistant Yahoo! Bookmarks My Yahoo!

provide here. All fields are required.

This information will be securely stored in your Yahoo! Wallet.

Yahoo! Wallet

We'll securely save your credit card information so you won't have to type it every time you use at Yahoo!. Your billing address must be exactly the same as the address for your credit card. Don't worry, your credit card will not be billed until you confirm a transaction.

Credit Card

Credit Card Number

Expiration Date

First Name Last Name

Billing Address

Address Line 2

City

State/Province Zip/Post Code

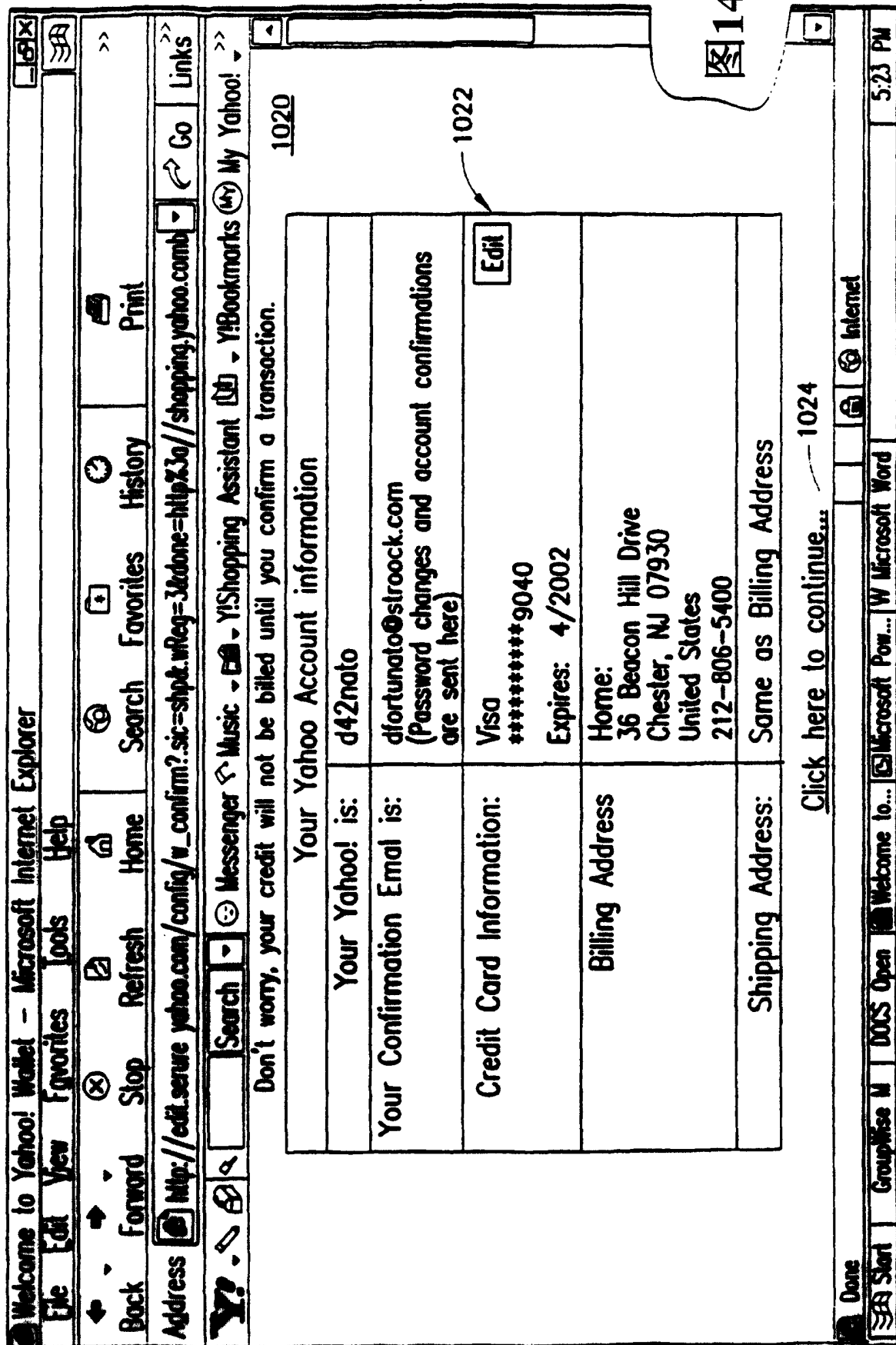
Country/Territory

Phone Number

This is: ☐ my Home Address

Done Start GroupWise M DOCS Open Welcome to... Microsoft Pow... W Microsoft Word 5:21 PM

图13



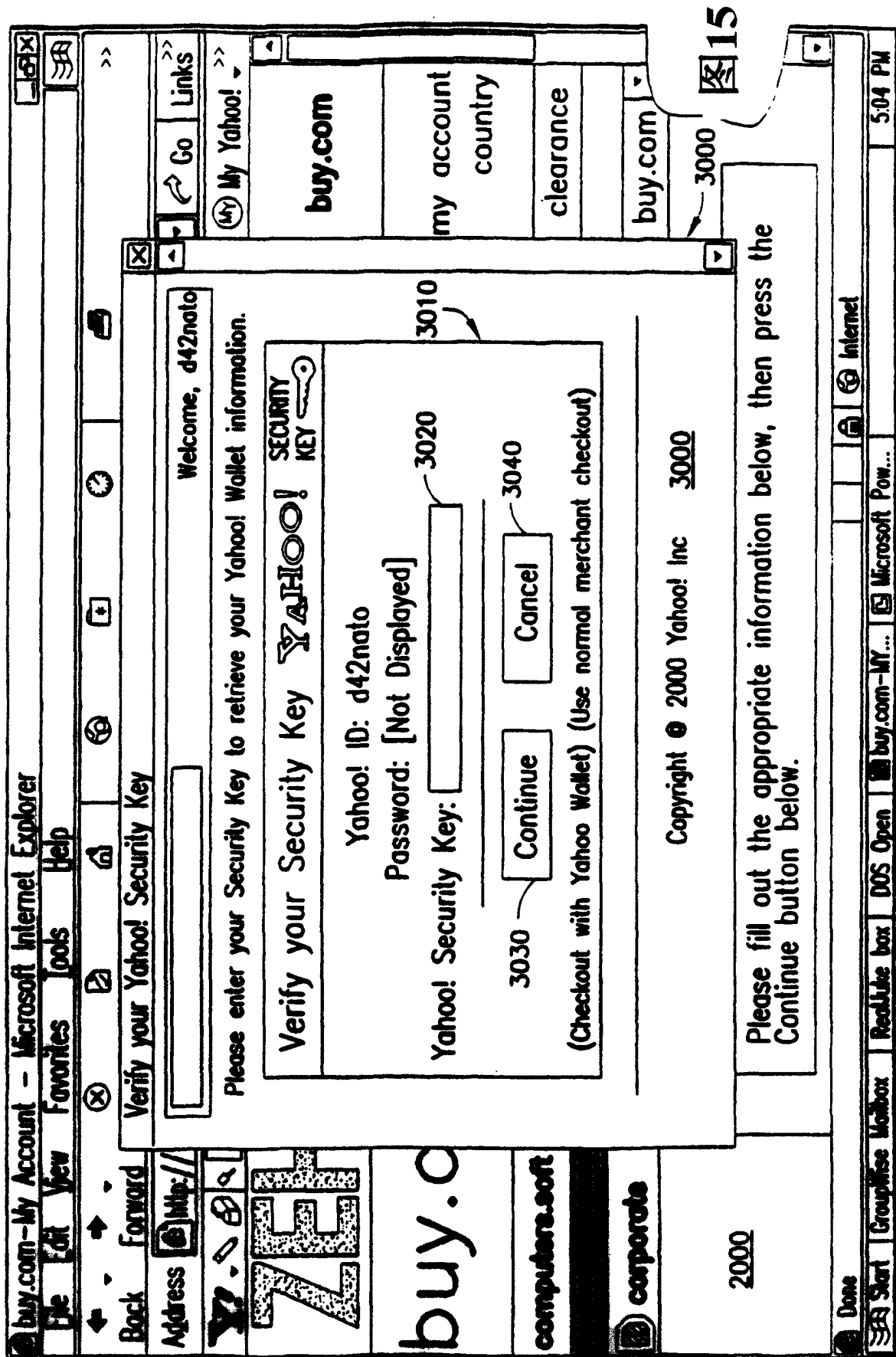
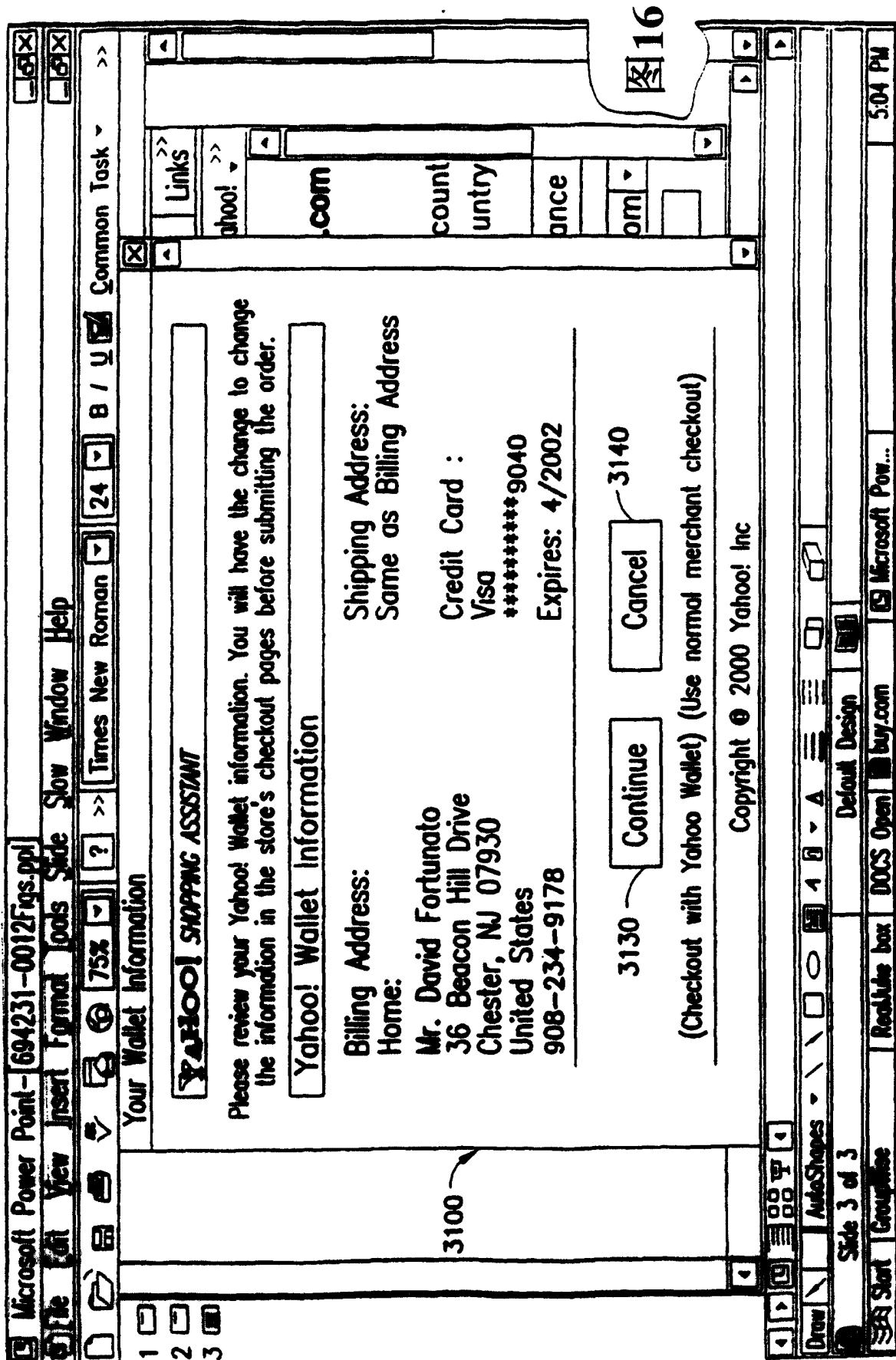


图15



http://www.us.buy.com/basket/shopper.asp?section1&fic=6776 - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Print

Address http://www.us.buy.com/basket/shopper.asp?section1&fic=6776

Search Messenger Music Shopping Assistant Bookmarks My Yahoo! Links

Go

Internet

Billing Information

Please note that paying by check or money order requires extra processing time.
More information on payment options.

Payment Method

Visa (Webcam) Credit Card Number *****9040 Expiration Date 04/2002

Your billing address must be entered exactly as it appears on your credit card statement.
Please check your statement for accuracy to avoid delays in processing your order.

*First Name David *Last Name Fortunato

Company Name

*Address - Line 1 36 Beacon Hill Drive

*Address - Line 2

*City Chester *State NJ *Zip Code 07930

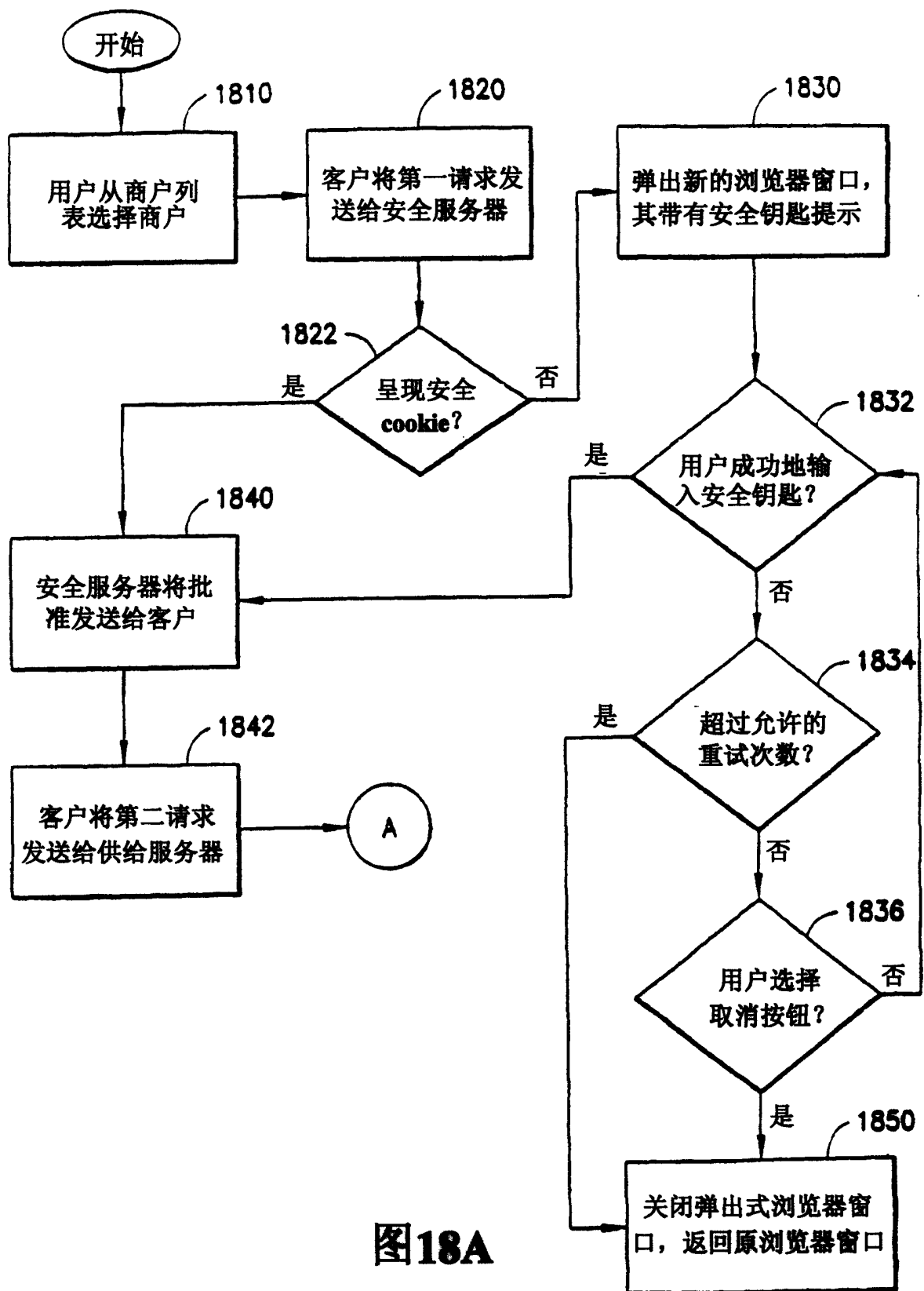
*Home Phone Number (908) 234 - 9178

*Company Phone Number - Ext: -

Fax Number -

Zero Liability THE ONLINE CURRENCY USD

图17



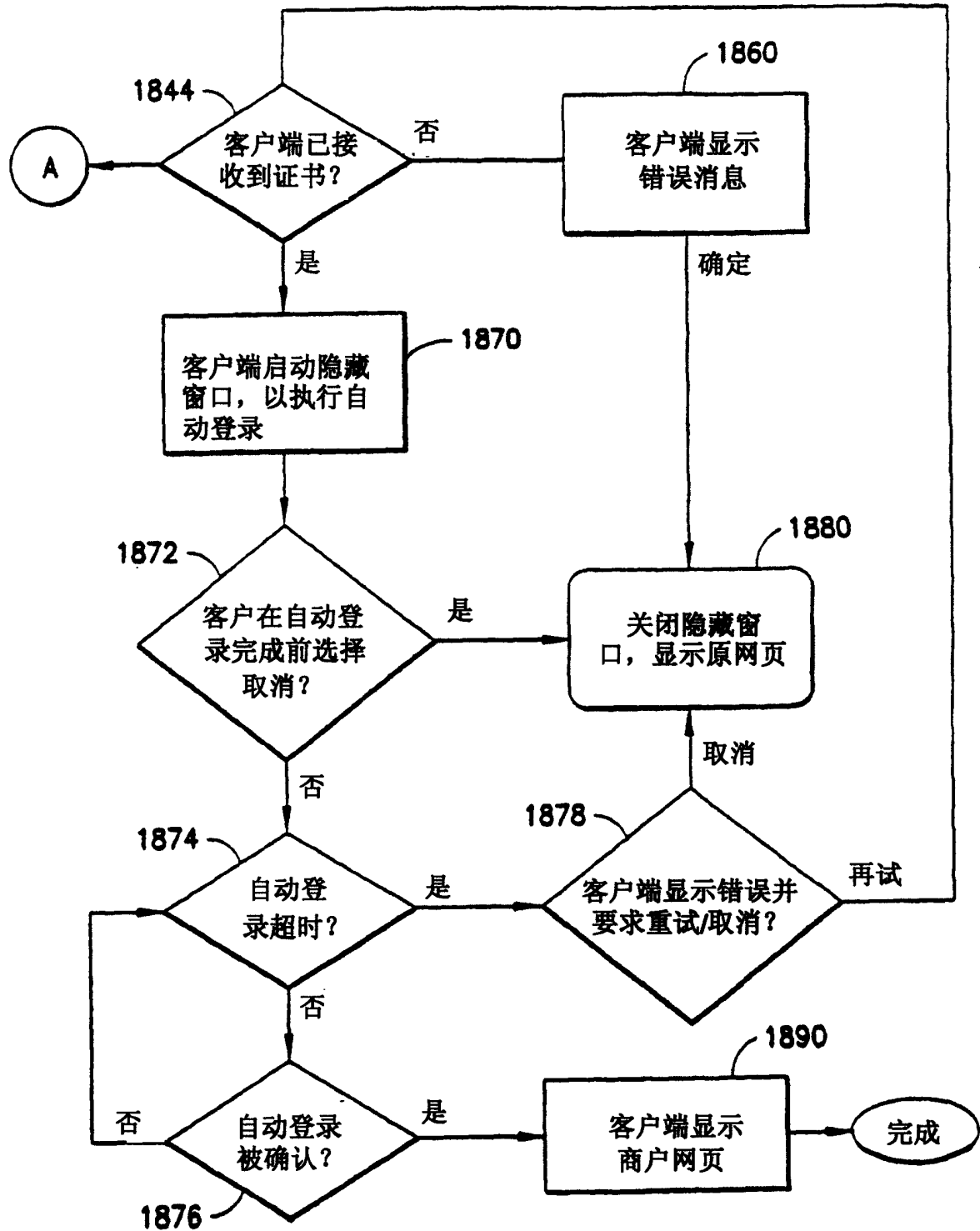


图18B

