

[19] 中华人民共和国国家知识产权局



[12] 发明专利申请公开说明书

[21] 申请号 200480006722.2

[51] Int. Cl.

G06Q 10/00 (2006.01)

G06F 17/30 (2006.01)

[43] 公开日 2006 年 4 月 12 日

[11] 公开号 CN 1759413A

[22] 申请日 2004.2.26

[21] 申请号 200480006722.2

[30] 优先权

[32] 2003.3.13 [33] US [31] 60/454,278

[32] 2004.2.24 [33] US [31] 10/785,305

[86] 国际申请 PCT/US2004/005679 2004.2.26

[87] 国际公布 WO2004/084003 英 2004.9.30

[85] 进入国家阶段日期 2005.9.12

[71] 申请人 西门子医疗健康服务公司

地址 美国宾夕法尼亚州

[72] 发明人 J·R·扎莱斯基 B·劳伦斯

[74] 专利代理机构 中国专利代理(香港)有限公司

代理人 吴立明 王忠忠

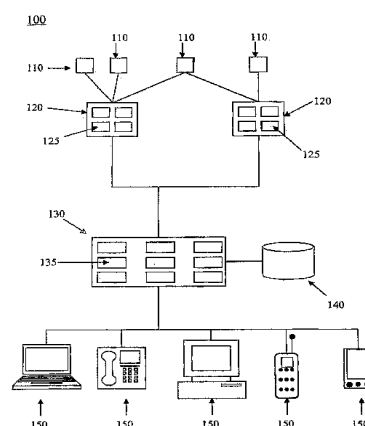
权利要求书 3 页 说明书 24 页 附图 8 页

[54] 发明名称

访问患者信息的系统

[57] 摘要

某些示例性实施例提供一种用于在网络中访问患者医疗信息的系统。所述网络包含多个服务器。网络的某些示例性实施例还包含储存库，所述储存库包括患者标识符和用于标识出一个或者多个存储与特定患者相关联的医疗信息的服务器的相关联的服务器标识符。为了响应接收到的显示特定患者的医疗信息的命令或者请求，网络的某些示例性实施例包含搜索处理器，用于发起所述储存库的搜索，以定位与特定患者的标识符相关联的特定服务器标识符。为了响应用户命令，网络的某些示例性实施例包含接口处理器，用于产生将所定位的特定服务器标识符结合在数据字段中的统一资源定位符(URL)地址，并且用于发起访问在由服务器代管的所产生的 URL 地址处的所存储的特定患者的医疗信息的请求。



1、一种用于在包括多个服务器的网络中访问患者医疗信息的系统，其包含：

5 储存库，所述储存库包含患者标识符和用于标识出存储特定患者的医疗信息的特定服务器的相关联的服务器标识符；

搜索处理器，用于响应于接收到的命令启动所述储存库的搜索，以定位与所述特定患者的标识符相关联的特定服务器标识符；和

10 接口处理器，用于产生将所述已定位的特定服务器标识符结合在数据字段中的 URL 地址，并且用于响应于用户命令启动访问在由所述特定服务器代管的所述生成的 URL 地址处的所述存储的特定患者的医疗信息的请求。

2、根据权利要求 1 所述的系统，其中

所述储存库包含链接所述患者标识符和用于标识出代管特定患者的医疗信息的服务器的所述相关联的服务器标识符的映射。

15 3、根据权利要求 1 所述的系统，其中

所述接口处理器通过将所述特定患者标识符结合在数据字段中来产生所述 URL 地址。

4、根据权利要求 1 所述的系统，其中

20 所述搜索处理器判断以下情况中的至少一个：（a）所述储存库是否含有多个所述特定患者标识符和（b）所述储存库是否不含有与所述特定服务器标识符相匹配的标识符，

响应于所述判断，所述接口处理器启动标识（a）和（b）中的至少一个的消息的生成并且

25 响应于（a）和（b）中的至少一个的判断，所述接口处理器禁止启动访问所述存储的所述特定患者的医疗信息的请求。

5、根据权利要求 1 所述的系统，还包含

采集处理器，用于查询多个不同的服务器，以编辑指示患者标识符和相关联的服务器标识符的数据以存储在所述储存库中并且

30 所述采集处理器响应于标识出所述多个不同服务器的记录来周期地查询所述多个不同的服务器。

6、根据权利要求 1 所述的系统，还包含

显示处理器，用于启动代表所述访问过的所存储的所述特定患者

的医疗信息的数据的生成并且

所述访问过的所存储的所述特定患者的医疗信息包含以下中的至少一个：（a）血压参数、（b）换气参数、（c）生命指征参数、（d）代表血氧浓度的参数、（e）与给液相关联的输液泵参数、（f）点滴医药相关的参数、（g）血气参数、和（h）涉及所述特定患者与保健机构的交互作用的财务信息。

7、根据权利要求 1 所述的系统，还包含

授权处理器，用于响应于所述用户命令验证用户被授权访问所述存储的所述特定患者的医疗信息，并且用于响应于不成功的验证禁止所述用户访问所述存储的所述特定患者的医疗信息。

8. 一种用于在包括多个服务器的因特网协议兼容的网络中访问患者医疗信息的系统，其包含：

支持经因特网兼容的用户接口访问患者医疗信息的可执行的应用程序；

15 搜索处理器，用于响应于使用所述用户接口输入的用户命令来启动至少一个数据源的搜索，以找出与特定患者的标识符相关联的特定服务器标识符；和

接口处理器，用于：

20 产生将由所述搜索处理器找到的所述特定服务器标识符结合在数据字段中的 URL 地址，

通过所述生成的 URL 地址启动访问由所述特定服务器代管的所述存储的所述特定患者的医疗信息的请求；并且

传送所述访问过的所存储的医疗信息，以使用所述因特网兼容的用户接口显示给用户。

25 9、根据权利要求 8 所述的系统，其中

所述至少一个数据源包含以下中的至少一个：（a）包括患者标识符和用于标识出存储特定患者的医疗信息的特定服务器的相关联的服务器标识符的储存库和（b）多个不同的服务器。

30 10. 一种用于在包括多个服务器的因特网协议兼容的网络中访问患者医疗信息的系统，其包含：

支持经因特网兼容的用户接口访问患者医疗信息的可执行的应用程序；

包括患者标识符和用于标识出存储特定患者的医疗信息的特定服务器的相关联的服务器标识符的储存库；

搜索处理器，用于响应于接收到的命令启动所述储存库的搜索，以定位与所述特定患者的标识符相关联的特定服务器标识符；和

- 5 接口处理器，用于响应于用户命令来启动访问在响应于所述已定位的特定服务器标识符和所述特定患者标识符得到的 URL 地址处的所述存储的特定患者的医疗信息的请求。

11、根据权利要求 10 所述的系统，还包含

- 10 采集处理器，用于查询多个不同服务器以编辑指示患者标识符和相关联的服务器标识符的数据，以存储在所述储存库中。

12. 一种用于在包括多个服务器的因特网协议兼容的网络中访问患者医疗信息的方法，所述方法包含以下动作：

响应于使用用户接口输入的用户命令，启动至少一个数据源的搜索，以找到与特定患者的标识符相关联的特定服务器标识符；

- 15 产生将通过所述搜索找到的所述特定服务器标识符结合到 URL 数据字段中的 URL 地址；

启动访问在由所述特定服务器代管的所述生成的 URL 地址处的所存储的特定患者的医疗信息的请求；和

- 20 通过所述因特网协议兼容的网络传送所述访问过的所存储的医疗信息以显示到所述用户接口上。

访问患者信息的系统

相关申请的交叉参考

- 5 本申请要求于2003年3月13日提交的待批的临时申请序列号为60/454,278（申请人档案号为03P03690US）的优先权。

背景技术

- 公知的产品提供一种针对患者生命数据的网络观察器。为了经所述网络观察器显示特定患者的生命数据，用户典型地需要提供一系列标识符，该标识符包括用户ID、用户密码、患者姓名、患者标识码、服务器和/或患者生命数据驻留的位置等等。然而，往往用户不知道一个或者多个用于访问这样的生命数据的所需要的标识符，因为这样的信息一般被至少部分地维持在就医时的临床环境内。此外，这样的数据的暂时流通也是成问题的。在某些公知的产品中，用户得到的数据
- 10 和/或病历不能准确地反映实时的患者信息。

发明内容

- 某些示例性实施例提供一种用于在网络中访问患者医疗信息的系统。所述网络包含多个服务器。网络的某些示例性实施例还包含储存库，所述储存库包括患者标识符和用于标识一个或者多个存储与特定患者相关联的医疗信息的服务器的相关联的服务器标识符。为了响应接收到的显示特定患者的医疗信息的命令或者请求，网络的某些示例性实施例包含搜索处理器，用于发起所述储存库的搜索，以定位与特定患者的标识符相关联的特定服务器标识符。为了响应用户命令，网络的某些示例性实施例包含接口处理器，用于产生将所定位的特定服务器标识符结合到数据字段中的统一资源定位符（URL）地址，并且用于发起访问由服务器代管的所产生的URL地址处的所存储的特定患者的医疗信息的请求。
- 20
- 25

附图说明

- 通过下面的详细说明和附图可以更好地理解广泛的可能的实施例，在附图中：
- 30

图1是用于访问患者信息的系统的示例性实施例；

图2是用于访问患者信息的信息装置的示例性实施例的方框

图；

图 3 是支持与用于访问患者信息的系统相关联的操作的用户接口的示例性实施例；

5 图 4 是支持与用于访问患者信息的系统相关联的操作的用户接口的示例性实施例；

图 5 是支持与用于访问患者信息的系统相关联的操作的用户接口的示例性实施例；

图 6 是支持与用于访问患者信息的系统相关联的操作的用户接口的示例性实施例；

10 图 7 是针对用于访问患者信息的系统的方法的示例性实施例的流程图；以及

图 8 是针对用于访问患者信息的系统的方法的示例性实施例的流程图。

定义

15 当在此使用以下术语时，适用所附定义：

20 数据库-一个或者多个结构化持久性数据集合，通常与更新和查询数据的软件相关联。简单的数据库可以是含有许多记录的单个文件，所述记录的每个都使用相同的字段组来结构化。数据库可以包含映射，其中根据诸如身份、物理位置、网络上的位置、功能等等的各种因素来组织各种标识符。

25 功能链接-页面上的链接，允许用户通过诸如击键盘或者点鼠标之类的动作激活该功能链接来访问特定的功能。功能链接的激活可以经“单个动作”发生，在此使用的所述单个动作指可以激活功能的任何单个动作、诸如鼠标点击、鼠标击活链接、键盘敲打、笔敲打、手指敲打或者信号、（在预定的屏幕位置开始持续预定时间的）语音信号、和/或其任何等效事物。

标识符-对特定的实体、活动、和/或文件独特的一组符号。例如，标识符可以是病历号。标识符可以是人和/或机器可读的、诸如号码、字母串、条形码、RFID 等等。

30 患者标识符-保健机构的特定患者的标识符。患者标识符可以是社会保险号、纳税人 ID 号、国家 ID 号、医疗号、医疗补助号、医疗保险 ID 号、病历号等等。

服务器标识符-一个或者多个患者监控装置所链接到的特定服务器的标识符。

用户标识符-在此所述的装置和/或系统的特定用户的标识符。

5 信息装置-能够处理信息的装置，诸如任何通用和/或专用计算机，诸如个人计算机、工作站、服务器、小型计算机、主机架、超级计算机、计算机终端、膝上型电脑、电话、和/或者其任何等效事物等等。

接口-两个独立的系统跨越其相接并且相互作用或者通信的边界。为了借助于接口连接或者交互作用。

10 可机读介质-可以由信息装置读取的存储器。

存储器-能够存储模拟或者数字信息的装置，例如非易失性存储器、易失性存储器、随机存取存储器（RAM）、只读存储器（ROM）、闪存、磁介质、硬盘、软盘、磁带、光介质、光盘、压密盘、CD、数字通用盘、DVD、和/或冗余独立磁盘阵列等等。所述存储器可被耦合到处理器并且可以存储适配于由根据在此公开的实施例的处理器执行的指令。

网络-有线和/或无线通信网络。

20 网络接口-电话、蜂窝式电话、蜂窝式调制解调器、电话数据调制解调器、传真调制解调器、无线无线电收发机、以太网卡、电缆调制解调器、数字用户线接口、网桥、集线器、路由器，或者其它类似装置。

患者-因保健目的受监管的人或者其它类型的动物。

25 患者信息-与患者的医疗和/或治疗相关的信息，包括实时生命的、生物学的和/或生理学的数据、与生命的、生物学的和/或生理学的数据相关的接近实时的数据和/或既往史数据、血压参数、换气参数、生命指征参数、代表血氧浓度的参数、与给液相关的输液泵参数、点滴药物相关参数、血气参数、保险信息、保健人员信息、保健机构信息、帐单信息、家庭信息、财务信息、治疗信息、药物信息和/或任何等效事物等。

30 患者监控装置-能够收集、显示和/或中继患者信息的装置。

处理器-用于执行任务的装置和/或可机读的指令组。处理器包含硬件、固件和/或软件中的任何一个或者组合。处理器通过操纵、分

析、修改、变换、传输信息来作用信息，以供可执行的程序和/或信息装置使用、和/或将信息路由到输出装置。处理器可以使用控制器的能力。

5 服务器-针对经网络连接到其的其它信息装置提供某种业务的信息装置。普通的例子是文件服务器，所述文件服务器具有本地盘并且对远程客户机在该盘上读和写文件的请求提供服务。服务器还可以提供对诸如程序、共享装置的资源访问等。

10 客户机-信息装置和/或在其上运行的过程，其使用某种协议请求另一个信息装置或者在其上运行的过程（“服务器”）的业务并且接受服务器的响应。客户机是客户机服务器软件结构的一部分。例如，从文件服务器请求文件的内容的计算机是该文件服务器的客户机。

15 瘦客户机-相对简单的客户机程序和/或硬件装置，对于其多数能力主要依赖于服务器。使用标准网络浏览器显示的还包含或者纯文本、HTML、脚本、或者简单对象（诸如 ActiveX 成分或者 Java 小应用程序）的网页代表此类别的示例性实施例。

统一资源定位符（URL）-在因特网、网络 and/或连接到其上的服务器上指定诸如网页的对象的位置的标准方式。URL 可以包含含有一个或者多个标识符的数据字段。

20 用户接口-用于对用户呈递信息和/或从用户请求信息的装置和/或程序。用户接口可以包括文本的、图形的、音频的、视频的、动画的和/或触觉的元素。

用户-能够利用用于访问患者信息的系统的个体。

25 生命指征-活的机体中的任何生物学的和/或生理学过程的测量。生命的、生物学的和/或生理学的数据的示例性实施例可以包含与患者的心率、体温、血气、红血球计数、白血球状态、呼吸容量、呼吸率，和/或其任何等效事物相关联的患者信息。

具体实施方式

30 某些示例性实施例提供用于在网络中访问患者信息的系统。用于在网络中访问患者信息的系统的某些实施例包含用户接口，所述用户接口与使得能够经基于网络的 URL 链接从临床访问应用程序中启动瘦客户机生命观察器的处理方法相耦合。所述系统的某些示例性实施例接受患者标识符和/或诸如用户姓名和密码之类的用户鉴别信息的

输入。患者标识符的输入触发所述系统自动地启动包含属于与来自位于医院临床信息系统内的主机服务器的所接受的患者标识符相关联的患者的信息的正确的生命观察器。用于访问患者信息的系统的某些示例性实施例自动地核查多个服务器，以便为用户提供患者信息。通过轮询服务器，创建和维持患者的标识符和患者的信息驻留在其上的服务器的标识符的公共数据库。经过用户接口，所述患者信息通过利用标识符启动诸如生命观察器之类的基于浏览器的应用程序来提取。

图 1 是用于访问患者信息的系统 100 的示例性实施例。系统 100 的某些示例性实施例包含多个患者监控装置 110。患者与一个或者多个患者监控装置相耦合。患者监控装置 110 被耦合到一个或者多个服务器 120。服务器 120 包含一个或者多个处理器 125。处理器 125 执行任何功能，诸如从相关联的患者监控装置 110 中收集信息、按照患者和/或患者监控装置标识符组织收集到的信息、接收对存储在服务器 120 内的标识符的请求、发送所请求的信息、执行系统维护、授权用户访问系统 100 和/或其任何等效事物等等。

在系统 100 的某些示例性实施例中，服务器 120 被耦合到储存库服务器 130。储存库服务器 130 的某些示例性实施例包含储存库 140。储存库 140 包含维持所有与其各自的监控装置结合的患者列表的数据库的功能。用于访问患者信息的系统 100 的例子利用存储在储存库 140 中的患者标识符和服务器标识符。储存库服务器 130 包含一个或者多个处理器 135。处理器 135 包含搜索处理器、接口处理器、采集处理器、显示处理器、授权处理器和/或者其任何等效事物，等等。

使用任何适当的访问装置 150，用户经储存库服务器 130、储存库 140 和/或服务器 120 访问患者信息。访问装置 150 是任何通用和/或专用的信息装置。系统 100 的网络连接是有线的和/或无线的连接和/或通信网络。系统 100 的某些示例性实施例是密码保护的和/或使用诸如密码和数据加密、防火墙、病毒保护的标准的网络安全措施，和/或其任何等效事物，等等。

图 2 是信息装置 200 的示例性实施例的方框图，所述信息装置在某些操作实施例中例如代表图 1 的患者监控装置 110、服务器 120、储存库服务器 130、储存库 140 和/或访问装置 150。信息装置 200 包

含任何多个公知的部件、诸如一个或者多个网络接口 210、一个或者多个处理器 220、一个或者多个含有指令 240 的存储器 230、一个或者多个输入/输出 (I/O) 装置 250、和/或一个或者多个耦合到 I/O 装置 250 的用户接口 260 等等。

5 信息装置 200 的某些示例性实施例包括用户接口 260。用户接口 260 显示患者信息。用户接口 260 还呈现用于与信息装置 200 交互作用的指令。在某些示例性实施例中，用户接口 260 与一个或者多个输入/输出 (I/O) 装置 250 协同工作。在用户接口 260 与 I/O 装置 250 之间的交互作用允许用户请求、收集、组织、观察和/或中继患者信息等等。I/O 装置 250 的某些示例性实施例自动地收集、请求、中继、
10 显示和/或组织患者信息等等。在某些示例性实施例中，经一个或者多个用户接口 260、诸如图形的用户接口，用户提供感兴趣的患者监控装置的 URL 和/或接收涉及感兴趣的患者监控装置的当前位置信息。

15 信息装置 200 的某些示例性实施例包含患者信息，所述患者信息包含从诸如患者监控装置及其相关联的服务器的其它信息装置中收集到的实时的、接近实时的、或者既往的患者数据。在某些示例性实施例中，患者信息被存储在存储器 230 内。存储器 230 的某些示例性实施例包含患者标识符及其相关联的服务器标识符的列表。信息装置
20 200 的指令 240 管理存储器 230 内的数据和信息的正确收集和组织。指令 240 被存储在一个或者多个不同类型的存储器中。

信息装置 200 的某些示例性实施例包含一个或者多个处理器 220。处理器 220 的示例性实施例是搜索处理器。响应于接收到的命令，处理器 220 发起存储器 230 的搜索。接收到的命令可以是用户发
25 起的，或者是由用户或者由软件调度的定时的事件。处理器 220 搜索患者标识符和服务器标识符。处理器 220 的某些示例性实施例轮询各个服务器以识别出与被链接到一个或者多个服务器的患者监控装置相连接的患者。轮询过程是自动的事件和/或所调度的事件。可替换地，用户命令处理器 220 发起搜索。在某些实施例中，处理器 220 提
30 供关于特定的患者是否在用户企图访问该患者的医疗信息以前当前正被监控的通知。为了访问患者信息，用户不需要知道特定患者的具体位置。

处理器 220 的示例性实施例是接口处理器。从而，处理器 220 的某些示例性实施例协调对针对患者信息的用户命令的响应。处理器 220 产生用于访问所请求的信息的 URL 地址。处理器 220 的某些示例性实施例利用适当的患者标识符和服务器标识符来产生 URL。当激活 URL 时，处理器 220 发起收集所请求的信息。此外，处理器 220 经用户接口 260 和/或 I/O 装置 250 来传送所收集的信息。在某些示例性实施例中，处理器 220 判断所请求的患者标识符和/或相关联的服务器标识符是否存在于网络内并且发起产生传送所请求的标识符是否可得到的消息。如果在相关联的网络中没有所请求的患者标识符和/或服务器标识符，处理器 220 的某些示例性实施例禁止发起访问患者医疗信息的请求。

处理器 220 的示例性实施例是采集处理器。从而，处理器 220 的某些示例性实施例采集和/或编辑患者标识符和服务器标识符的列表，以存储在存储器 230 内。处理器 220 的某些示例性实施例从多个 I/O 装置 250 和/或用户接口 260 中收集其它形式的信息、诸如患者生命指征数据、患者病史、帐单信息和/或任何适当的患者信息。在某些示例性实施例中，处理器 220 的采集功能是自动的和/或被调度的。可替换地，用户人工地命令处理器 220 执行各种任务。处理器 220 的某些示例性实施例周期地和/或非周期地查询多个不同的服务器，以编辑指示患者标识符和相关联的服务器标识符的数据以存储在所述储存库中。处理器 220 的某些示例性实施例响应于识别出多个不同服务器的输入来周期地和/或非周期地查询多个不同的服务器。输入的示例性实施例是识别出多个标识符的任何数据形式或者记录。

处理器 220 的示例性实施例是显示处理器。从而，处理器 220 的某些示例性实施例发起和/或维持各种形式的显示的数据、诸如用户接口 260 上的诸如 EKG 波形的患者信息的文本的和/或图形的数据显示。

处理器 220 的示例性实施例是授权处理器。从而，处理器 220 的某些示例性实施例验证用户被授权访问存储在一个或者多个信息装置 200 和/或由多个信息装置 200 组成的网络内的患者信息。如果用户没有被授权访问患者信息，则处理器 220 阻止访问信息装置 200，并且在某些示例性实施例中，处理器 220 发起指示禁止该访问的消息。

息。可替换地，如果用户被授权访问信息装置 200，则处理器 220 发起指示成功访问的到用户的通信。

信息装置 200 的某些示例性实施例包含网络接口 210。网络接口 210 允许经有线的和/或无线的网络与其它的信息装置 200 交互作用。

图 3 是支持与用于访问患者信息的系统相关联的操作的用户接口 300 的示例性实施例。用户接口 300 的某些示例性实施例是非基于浏览器的可执行应用程序，所述应用程序被配置来经网络收集信息。可执行的应用程序的某些示例性实施例支持经临床访问应用程序访问患者信息，所述临床访问应用程序包含因特网兼容的用户接口 300 和处理方法，从而使得能够经基于网络的 URL 链接启动瘦客户机生命观察器。用户接口 300 的某些示例性实施例经由 URL 地址 330 标识的浏览器页面来观察和/或呈现。URL 地址 330 包含含有各种标识符的数据字段。

用户接口 300 的某些示例性实施例包含登录屏 310。登录屏 310 包含登录功能 320。登录功能 320 的某些示例性实施例包含标准的用户名和密码系统。从而，登录功能 320 接受用户的姓名和密码，以允许访问用于访问患者信息的系统。可替换地，登录功能 320 接受患者标识符的输入。在某些示例性实施例中，患者标识符的输入导致来自位于医院临床信息系统内的主机服务器的正确的生命观察器的呈现。

登录屏 310 的某些示例性实施例包含一个或者多个用户接口元件、诸如按钮 330、功能链接 340、和/或图标链接 350。激活用户接口元件 330、350 和/或功能链接 340 引起诸如启动独立的窗口、将用户转到另一个窗口、和/或启动新的应用程序等的任何动作。

图 4 是支持与用于访问患者信息的系统相关联的操作的用户接口 400 的示例性实施例。用户接口 400 呈现患者信息观察 410。患者信息观察 410 的某些示例性实施例包含患者列表 450、一个或者多个功能链接 440、和/或一个或者多个滚动菜单 460。通过点击页面标签 430 选择不同的屏幕进行观察。概况观察 410 的某些示例性实施例包含在用户接口 400 内的子屏幕。从而用户接口 400 的某些特性保持静止，诸如用户标识符 420 和/或诸如打印功能链接图标或者注销超级

链接的各种功能链接 470。

患者列表 450 的某些示例性实施例包含患者姓名和相关联的患者信息的列表。患者信息包括姓名、年龄、性别、住址和/或生命指征数据等等。患者姓名的某些示例性实施例包含到另外的患者信息的功能链接。在某些示例性实施例中，患者姓名与图表图标 455 相关联。在某些示例性实施例中，图表图标 455 的用户选择允许访问在传统的患者记录内找到的任何患者信息。

图 5 是支持与用于访问患者信息的系统相关联的操作的用户接口 500 的示例性实施例。用户接口 500 包含患者信息观察 510。在某些示例性实施例中，患者姓名的选择启动详细的患者观察 520。详细的患者观察 520 包含附加的患者信息、诸如 ID 号码、生命指征、生理学数据、当前添加到患者的患者监控装置、既往的患者信息，和/或其任何等效事物，等等。详细的患者观察 520 的某些示例性实施例还包含到用于管理患者监控装置的应用程序的功能链接。例如，用户选择 IV 点滴图标，以便访问使得能够调节 IV 点滴参数的用户接口。详细的患者观察 520 的某些示例性实施例包含各种滚动功能 530，以使得能够观察更多的数据。

图 6 是支持与用于访问患者信息的系统相关联的操作的用户接口 600 的示例性实施例。用户接口 600 的某些示例性实施例包含患者生命观察器 610。患者生命观察器 610 包含患者的任何生命指征和/或生理学信息 620、650。患者生命观察器 610 包含一个或者多个子屏幕 630、640，所述子屏幕包含滚动条。这样的结构允许用户在单个患者生命观察器 610 内访问更多的信息。

信息 620 的某些示例性实施例包含实时的和/或接近实时的生理学信息。患者生命观察器 610 内呈现的信息 620、650 是从患者监控装置中收集的数据的结果和/或由保健提供者在护理时输入的数据的结果。信息 620 用文本表示给用户。信息 650 的某些示例性实施例包含图形信息、诸如指示大脑的电活动的痕迹。信息 620、650 还包含以前的文本的和/或图形的患者数据和/或信息。

图 7 是针对用于访问患者信息的系统的方法 700 的示例性实施例的流程图。在动作 710，接收命令。命令的某些示例性实施例包含用于收集与特定患者相关联的患者标识符和/或服务器标识符的指令。

命令被自动地产生或者由用户人工地发起。

在动作 720，收集标识符以存储在储存库内。在某些示例性实施例中，患者标识符指定具有存储在患者信息管理系统内的患者信息的患者。相关联的服务器标识符被用于指定患者的信息被存储在其上的一个或者多个服务器。在动作 730，存储标识符。根据任何机构的系统收集和/或组织标识符。在某些示例性实施例中，储存库服务器使用处理器、诸如采集、搜索、网络、显示、授权和/或接口处理器，以采集和/或组织患者标识符和其相关联的服务器标识符。所述标识符被存储在一个或者多个服务器和/或储存库内。在某些示例性实施例中，储存库包含将患者标识符与其相关联的服务器标识符链接的映射，以便标识代管（host）针对特定患者的医疗信息的服务器。从而，各种轮询过程检索附属于与服务器链接的患者监控装置的有效患者的列表，并且从此收集中创建主列表和/或映射。不断地和自动地更新所述映射，以便可以结合任何所监控的患者参数中的变化，从而消除过时的患者信息。

在动作 740，产生 URL。所述 URL 将患者标识符和/或服务器标识符结合到 URL 数据字段内。在动作 750，处理访问信息的请求。在某些示例性实施例中，结合患者标识符和一个或者多个相关联的服务器标识符的 URL 地址允许经浏览器在网络内检索患者信息。

图 8 示出针对用于访问患者信息的系统的方法 800 的示例性实施例的流程图。在动作 810，执行至少一个数据源的搜索，以便定位患者标识符和任何相关联的服务器标识符。在某些示例性实施例中，所述搜索是自动的指令的结果。可替换地，用户经通过用户接口输入的命令发起搜索。在动作 820，从所述搜索中产生 URL 地址并且将服务器标识符和/或患者标识符结合到其数据字段内。

在动作 830，接收访问 URL 地址处的患者信息的请求。在某些示例性实施例中，所述请求由用户点击功能链接来产生。可替换地，用户输入用户标识符，以便产生与该用户相关联的患者的列表。在某些示例性实施例中，用户还输入患者标识符，以便产生访问与该特定患者相关联的信息的请求。

在动作 840，传送该患者信息，以显示在用户接口上。患者信息的传送允许用户请求实时的和/或接近实时的患者信息。在动作 850，

经用户接口把患者信息显示给用户。在某些示例性实施例中，如果患者标识符利用储存库和/或服务器找到，就启动生命观察器并且向用户在网页中呈现，在该网页中用户观察患者信息。如果患者标识符被复制在多个服务器上，则用户接口将该信息显示给所述用户。如果没有找到等同的患者标识符，则将适当的消息显示给用户。为了访问患者信息，用户不必输入患者位置、诸如物理位置或者在网络内的位置。

用于访问患者信息的系统的某些示例性实施例包含在临床客户机应用程序内的 URL 调用：http ://<host-server-name-or-IP-Address>/WinViewFrontEnd/WVBootAgent.asp。所述 URL 调用被物理映射到网关服务器上的 C:\Inetpub\wwwRoot\WinViewFrontEnd 目录内的文件 WVBootAgent.asp 上。

用于访问患者信息的系统的某些示例性实施例包含 WVBootAgent.asp。该页面利用以下参数来调用：

http ://<host-server-name-or-IP-Address>/WinViewFrontEnd/WVBootAgent.asp ? Login=guest&PID=xxxxxxxxwvyz&Pwd=winview。
针对此的代码包含：

```
<% @ Language=VBScript% >
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
</HEAD>
<BODY      bgcolor="black" text="white" onload="javascript:
close () ">
<%
    dim urlParameter1
    dim urlParameter2
    dim urlParameter3
    dim urlParameter4
    urlParameter1=trim(request("PID"))
    urlParameter2=trim(request("Login"))
```

```

        urlParameter3=trim(request("Pwd"))
        URLValue="checkPID.asp?PID="
        URLValue=URLValue & urlParameter1
        URLValue=URLValue & "&Login=" & urlParameter2 &
5    "&Pwd=" & urlParameter3
        Response.Write("<SCRIPT language='JavaScript'>")
        Response.Write("top.open ('&URLValue&')")
        Response.Write("<SCRIPT>")
    %>
10    <CENTER><h3>WinView Boot Agent</h3></CENTER>
    </BODY>
    </HTML>

    用于访问患者信息的系统的某些示例性实施例可以包含
    checkPID.asp。从调用页面:WVBootAgent.asp 中提取 PID、Login 和
15    Pwd。针对 checkPID.asp 的代码包含:
    <HTML>
    <% @ Language=VBScript %>
    <BODY>
    <%
20        dim gatewayAmount
        dim gatewayArray (10)
        dim URLValue
        dim temp
        dim temp2
25        dim pidString
        dim urlPID
        dim urlUser
        dim urlLogin
        dim urlPwd
30        urlPID =trim (request ("PID"))
        urlLogin =trim (request ("Login"))
        urlPwd =trim (request ("Pwd"))

```

```

Rem*****
Rem* 针对读和写创建文件系统对象。
Rem*****
    et fs=CreateObject("Scripting.FileSystemObject")
5  Rem*****
Rem*针对读、写、添加数据定义常数。
Rem*****
    Const ForReading=1, ForWriting=2, ForAppending=8
Rem*****
10 Rem*与 Pids 的列表比较当前的 PID
Rem*****

    在 checkPID.asp 内，pid-info.inf 是包含患者 Id 及其相关联的
    网关服务器的列表的人工创建的文件。典型地针对每个网关维持（也
    就是说更新）该文件。checkPID.asp 的代码还包含：
15 Response. Write ("<BR>")
    set f=fs.OpenTextFile
    ("C:\SecureFiles\WinViewFE\pid-info.inf", ForReading,
    false )
    IF f.ReadLine = "PID-INFO FILE == DO NOT MODIFY"THEN
20     gatewayAmount = 0
    WHILE NOT f.AtEndOfStream
        temp = f. ReadLine
        pos1 = Instr(1,TEMP, "=", 0)
        IF pos1 > 0 THEN
25         temp2 = Mid (temp, pos1+1, len(temp))
        temp = Mid (temp, 1, pos1)
        pos2 = Instr(1,temp, "\\ ", 0)
        IF pos2 > 0 THEN
25         temp = Mid (temp, pos2+2, len(temp))
        pos3 = Instr(1,temp, "\", 0)
30         IF pos3 > 0 THEN
            temp = Mid (temp, 1, pos3-1)
        END IF
    END IF
END IF

```

```

        IF temp2 = urlPID THEN
            gatewayArray(gatewayAmount) = temp
            gatewayAmount = gatewayAmount+1
        END IF
5         END IF
        WEND
    END IF

    在 checkPID.asp 的某些示例性实施例中，如果利用该患者 ID
    (PID) 只找到一个网关服务器，那么所有都良好，并且下个步骤是
10 调用启动 winwebviewer 的实际的 ActiveX 页面。checkPID.asp 的代码
    还包含：

    Rem*****
    Rem* 如果多于一个 PID 被匹配，则给出选择，
    Rem* 否则，利用新的 PID 和服务器来启动
15  Rem*
    Rem*****
    IF gatewayAmount = 1 THEN
        URLValue = "http://" & gatewayArray(0) &
        "/zeus4panel/index1.htm?Serv="
20         URLValue = URLValue & gatewayArray(0) & "&Login=" &
        urlLogin
        URLValue = URLValue & "&Pwd=" & urlPwd & "&PatID=" &
        urlPID
        Response.Redirect URLValue
25         Response.end
    ELSEIF gatewayAmount > 1 THEN
        Response.Write("<B><FONT FACE=COURIER SIZE=2>PID
        found on more than one gateway server<BR>")
        Response.Write("Please Choose the gateway Server to
30 use</FONT></B><BR><BR>")
        For i=0 to gatewayAmonunt-1
            Response.Write("<A href= 'http://' & gatewayArray(0)
            & "/zeus4panel/index1.htm?Serv=" & gatewayArray(1) &

```

```

        "&Login=" & urlLogin &"&Pwd=" & urlPwd &
        "&PatID=" & urlPID & ">Server:" & gatewayArray(1) &
        "</A><BR>")

```

```

    Next

```

```

5    ELSE

```

```

        Response.Write("<B><FONT                FACE=COURIER
SIZE=2>PID NOT found on any gateway server
        Server</FONT><BR></B>")

```

```

    END IF

```

```

10    %>

```

```

        </BODY>

```

```

        </HTML>

```

用于访问医疗信息的系统的某些示例性实施例包含 GatewayPIDListener ReadMe 文件，所述文件包含：

```

15        WinView Boot Agent
        WinViewBootReadMe.txt

```

```

=====
=====

```

```

20    引言

```

```

=====
=====

```

请求：

```

25    一个或多个网关服务器创建 ptlist.txt

```

共享的目录包含由 GatewayPidListener java 应用程序、网络服务器(推荐 IIS)、WinView 观察器可访问的 ptlist.txt

推荐：

```

30    java sdk (如果需要则重新编辑，推荐 java 版本 1.4.1 )

```

目的：为了如果可得到患者生命值就允许从临床信息访问应用程序中访问 WinWeb 生命观察器。

=====

=====

=====

解压

5

=====

=====

=====

在你的 C 盘上创建称为 SecureFiles 的目录。

将 WinViewBoot.zip 解压到这个目录下。

10

将创建两个目录 (WinViewFE 和 WebRoot)

你将需要或者通过使其成为虚拟目录 (IIS) 或者将其复制到网络可访问的目录下使得 WebRoot 可从网络服务器中得到。

注意：如果你将 WinViewBoot.zip 解压到任何其他目录下，则你通常要修改下列 .ASP 页面中的下列代码行：

15

C:\SecureFiles\WinViewFE\pid_info.inf ->checkPID.asp 中的第 35 行

C:\SecureFiles\WinViewFE\wvpassword.txt -
>ProcessPassword.asp 中的第 29 行

20

编辑这些，以将适当的路径包含到这两个文件中。

=====

=====

=====

25

安装

=====

=====

=====

GateWayList.txt

30

这个文件位于 WinViewFE 目录下。你利用你想要轮询的每个网关服务器上的 ptlist.txt 文件的正确位置来修改这个文件。针对我们有权访问的 java 应用程序，ptlist.txt 可从网络驱动器中来访问。

35

请不要编辑第一行：

针对 ptlist.txt 位置使用下列格式:

\\[ip 地址/服务器名]\\[包含 ptlist.txt 的目录名]\\ptlist.txt

例如:

5 \\Gateway1\\vs files\\ptlist.txt

wvpassword.txt

10 这个文件位于 WinViewFE 目录下。你利用你想要与 WVLogin.asp 页面一起使用的正确的用户名和密码来修改这个文件。

这个文件包含下列格式的用户名和密码:

[用户名][密码]

例如:

15 MyName MyPassword

(注意: 此时姓名不含有任何空格)

=====

====

使用 WinViewBoot Agent

20 =====

====

来自临床信息访问应用程序

WVBootAgent.asp 可从网络服务器中来访问。

25 当调用时, 到来自临床信息访问应用程序的 WVBootAgent.asp 的链接包含下列参数: PID、 USER

另外注意: 也将利用其他临床信息访问应用程序操作。

PID 将包含尝试被观察的患者的 PID。

30 USER 将包含将被用来访问观察器的用户名。

来自包含 GatewayPidListener java 应用程序的服务器

35 为了运行 GatewayPidListener, 确定 GateWayList.txt 与 GatewayListener.bat 文件和 GatewayListener.class 文件在相同的目录下。(注意: 这些文件应位于 WinViewFE 目录下)

执行 GatewayListener.bat 文件，这将启动 java 应用程序并且该应用程序将轮询具体的网关服务器。

用于访问医疗信息的系统的某些示例性实施例包含 GatewayPIDListener.java。该代码包含：

```
5      import java.io.*;
      import java.awt.*;
      import java.net.*;
      import java.lang.*;
      import java.util.*;
10     import java.lang.Math;
      import java.util.Random;
      import java.text.*;
      import java.util.Vector;

15     public class GatewayPidListener extends Thread implements
Runnable
    {
        private      String      ListFileName      =
“GateWayList.txt”; //GateWayList.txt 的位置和文件名
20     private String outputFileName = “pid-info.inf”; //输
出文件的位置和文件名
        private int PollDelay = 30; //以秒为单位的轮询的延迟

        boolean keepRunning = true; //真地连续以秒为单位运行
25     boolean debug = false; //调试输出
        Vector serverArray = new Vector(); //保留所有网关服务器
        private int serverAmount = 0;

        File ListFile = new File(ListFileName);
30     File outputFile = new File(outputFileName);

        public GatewayPidListener ()
```

```
{
    this.start();
} //结束构造程序

5
public void run()
{
    System.out.println("Polling gateway servers.....");
    do
10    {
        ListGrabber();
        CreateOutput();
        //将所有数据复位
        serverArray.removeAllElements();
15        serverAmount = 0

        System.out.print(">");
        try // 休眠计量
        {
20            sleep( PollDelay * 1000);
        } catch ( InterruptedException ie ) {
            System.err.println("Problem with the sleep
thread: "+ie);
        } // 结束 catch 程序
25    } while(keepRunning); //在 keepRunning 期间循环
}

public static void main(String argv[])
{
30    GatewayPidListener MV = new GatewayPidListener();
} //结束主程序
public boolean ListGrabber()
```

```
        {
            try
            {
                BufferedReader listInput = new BufferedReader(new
5  FileReader(ListFile));
                String dummy =
listInput.readLine(); //System.out.println(dummy);
                if(dummy.equals("***GateWay List Path***"))
                {
10  int counter = 0;
                while ((dummy = listInput.readLine())!=null)
                {
                    if(debug)
                        System.out.println("ADDED to
15  serverARRAY["+serverAmount+"]: "+dummy);
                        serverArray.addElement(dummy);
                        serverAmount++;
                } //结束 while
                } //结束 if dummy
20  } catch ( Exception e ) {
                System.err.println("Error in loadList(): "+e );
                return false;
            }
            return true;
25  } //结束 ListGrabber
            public void CreateOutput()
            {
                try
                {
30  PrintWriter FOut = new PrintWriter(new FileWriter
(outputFile));
                    FOut.println("PID-INFO FILE ==DO NOT MODIFY");
```

```
//打开每个 ptlist.txt 来确定 PIDS
for (int i=0; i<serverAmount; i++)
{
    5      try
        {

            File temp = new File ((String)serverArray.elementAt(i));
            BufferedReader Fin = new
10      BufferedReader(new FileReader(temp));
            String dummy;
            for (int j=0; j<9; j++) //跳转到 pids
                dummy = Fin.readLine();
            while ((dummy = Fin.readLine())!=null )
15      {
                if (!dummy.equals("[End List]"))
                {
                    if (debug)
                        System.out.println(dummy);
20      //解析出 pid

                    int pos1 = 0; int pos2 = 0;
                    String thepid="";
                    pos1 = dummy.indexOf("1");
                    if (pos1>=0)
25      thepid
                        dummy.substring(pos1+1, dummy.length());
                    pos2 = thepid.indexOf("|");
                    if (pos2>=0)
                        thepid
30      thepid.substring(0, pos2);
                    if (debug)
                        System.out.println("PID: "+thepid
```

```

);

                                //输出到文件
                                FOut.println((String)serverArray.elementAt
(i)+ "="+thepid);
5                                } //结束 if Idummy
                                } //结束 while
                                } catch (Exception e ) {
                                System.err.println("Error in read ptlist:"+e );
                                }

```

```

10    } //针对 i 结束
        FOut.close();
        } catch (Exception e ) {
        System.err.println("Could not open the output file for
writing:"+e);
15    } //结束 catch
        } //结束 ExamineList
        } //结束 GatewayPidListener

```

用于访问患者信息的系统的某些示例性实施例包含以下的用于定位服务器的指令:

```

20    C:
        Inetpub
            wwwRoot
                WinViewFrontEnd
                    WVBootAgent.asp
25                    checkPID.asp

        C:
            SecureFiles
                WinViewFE
30                    GatewayList.txt
                    pid-info.inf
                    GatewayPidListener.bat

```

GatewayPidListener.java

GatewayPidListener.class

用于访问患者信息的系统的某些示例性实施例包含自动启动网关服务器上的 GatewayPidListener。

- 5 GatewayPidListener-auto.bat 定位在一个网关的 C: 盘上。该文件的内容:

```
cd c:\securefiles\winviewfe
```

```
GatewayPidListener.bat
```

- 然后将下列信息输入到服务器上的 Autoexec.bat 文件中 (使用
10 来自 Start->Run 命令行的 Sysedit 命令来访问):

```
c: \GatewayPidListener-auto.bat
```

- 如果出于某种原因如通过在任务管理操作内检查过程验证出一个过程没有运行, 那么用户既可以访问 c: 盘内的 GatewayPidListener-auto.bat 文件也可以访问
15 securefiles\winviewfe 内的 GatewayPidListener.bat 文件。这还可以通过将 GatewayPidListener-auto.bat 文件置于 Startup 文件夹内完成。这引起在服务器启动时自动地启动该过程。

- 用于访问患者信息的系统的某些示例性实施例包含用户接口和
处理方法, 所述用户接口和处理方法使得能够经基于网络的 URL 链接
20 从临床访问应用程序中启动瘦客户机生命观察器。该系统接受患者 ID 并且自动地从位于医院临床信息系统内的主机网关服务器中启动正确的生命观察器并且将该患者的结果 (如果在监控器上) 经通过其自己的网络窗口启动的网络呈现出。尽管生命观察器本身不要辅助地自行完成此任务, 但是该系统具有自动地 (不需要经临床访问应用程序
25 来指定地) 校验多个现存的服务器以向用户提供正确的患者的生命波形的能力。完成此任务的方法的系统涉及轮询过程, 所述轮询过程在每个现存的服务器上的生命监控器上检索有效患者的列表, 并且创建由基于网络的观察器检索的主列表, 用于接近实时地判断患者是否在生命监控器上。如果发现患者在服务器的相关联的数据库内, 则启动
30 患者的生命观察器并且以网页呈现给用户, 在此网页上用户观察接近实时的生命信息结果。如果患者 ID 被复制在多个服务器上, 则这种方法被显示给用户。如果没有找到等同的患者标识符, 则将该消息显

示给该用户。

当前的服务器产品组提供针对患者生命数据的瘦客户机网络观察器。为了经瘦客户机网络观察器显示特定患者的生命参数必需指定用户 ID、用户密码、患者驻留在其上的特定服务器的名称、和患者标识符。然而，特定服务器的知识典型地对健康信息系统用户是得不到的，因为该信息一般是被维持在就医时的临床环境中的。通过维持所有网关患者列表文件（一般由服务器写入）的公共数据库并且结合启动页使用这些文件以提取网关服务器和服务器的服务器上相关联的患者的标识符，用于访问患者信息的系统去除用户知道和指定特定的网关服务器的需要。

用于访问患者信息的系统能有利地提供启动利用产品的网关服务器组提供的瘦客户机基于网络的生命观察器而不必事前指定患者在特定网关服务器上的位置。在可替换的示例性实施例中，特定服务器标识符被编码成对瘦客户机网络观察器的 URL 调用。该系统可以由需要在企业内的实体的位置（就是说，将特定患者定位在特定医院的临床领域内）的具体知识的其它应用程序来使用。

通过阅读以上的详细说明和某些示例性实施例的附图，其他的实施例对本领域内的普通技术人员将是显而易见的。

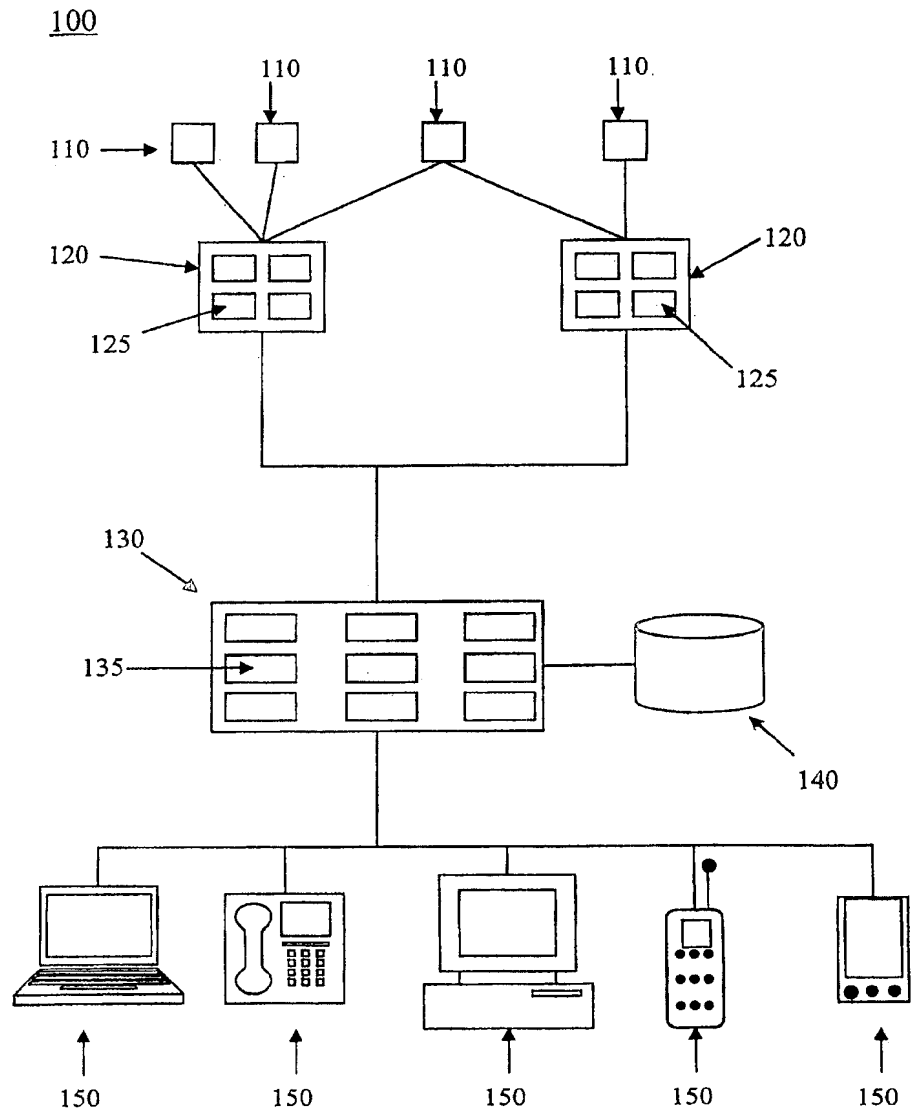


图 1

200

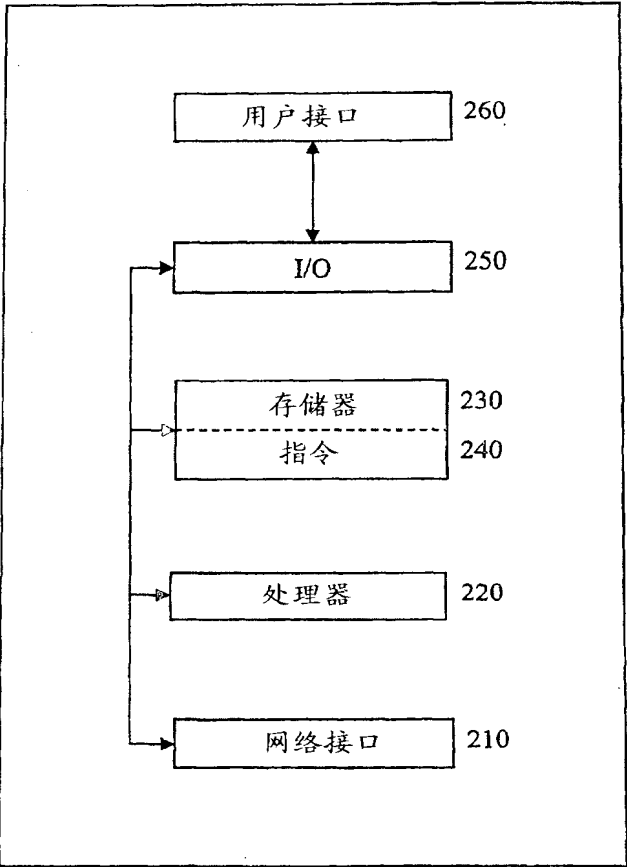


图 2

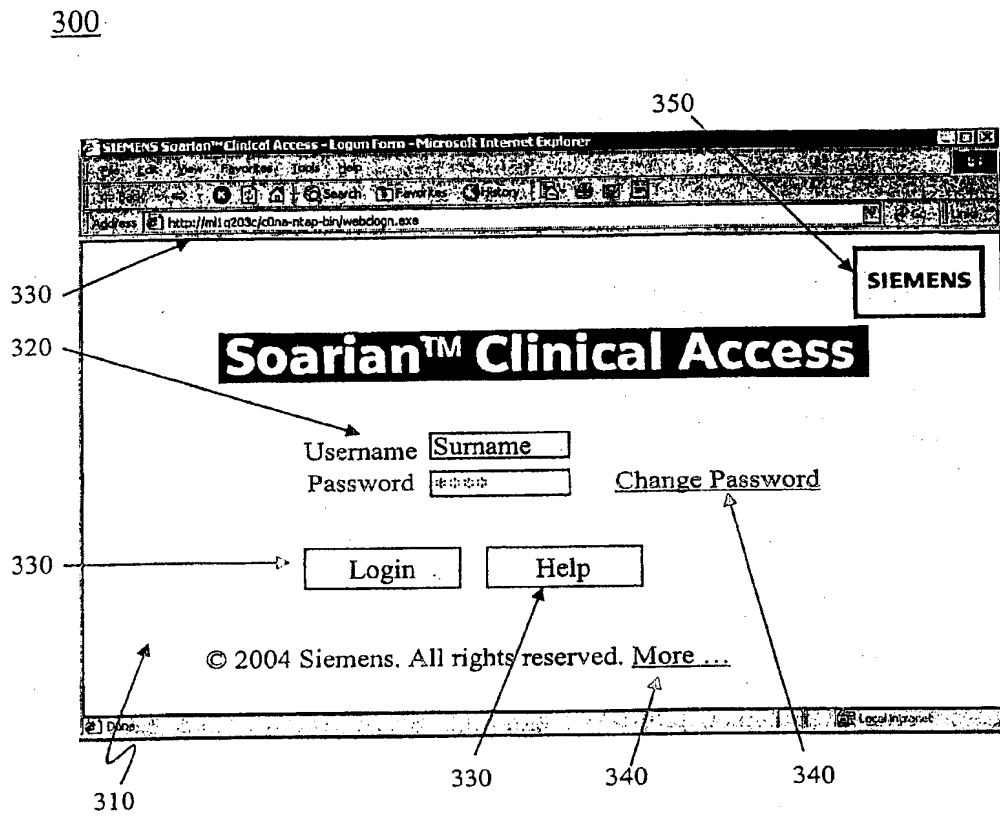


图 3

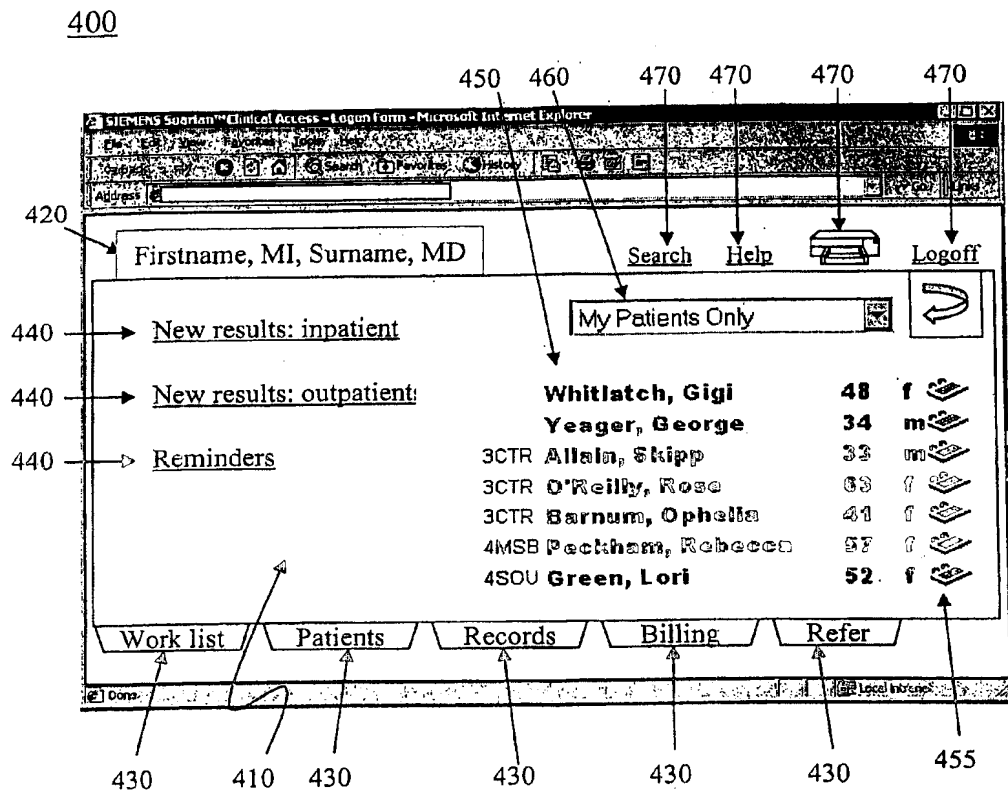


图 4

500

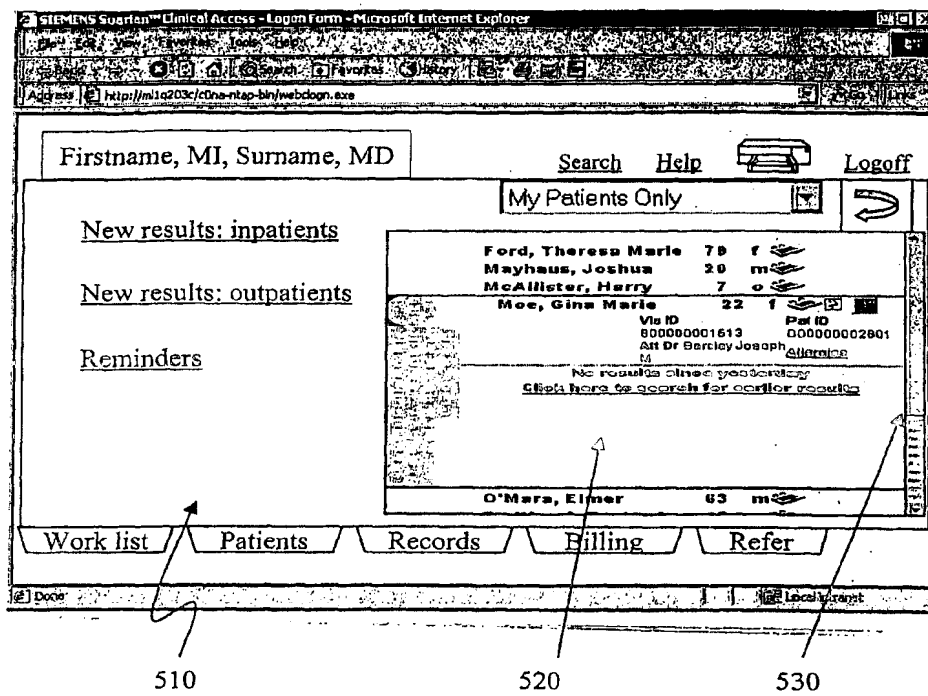


图 5

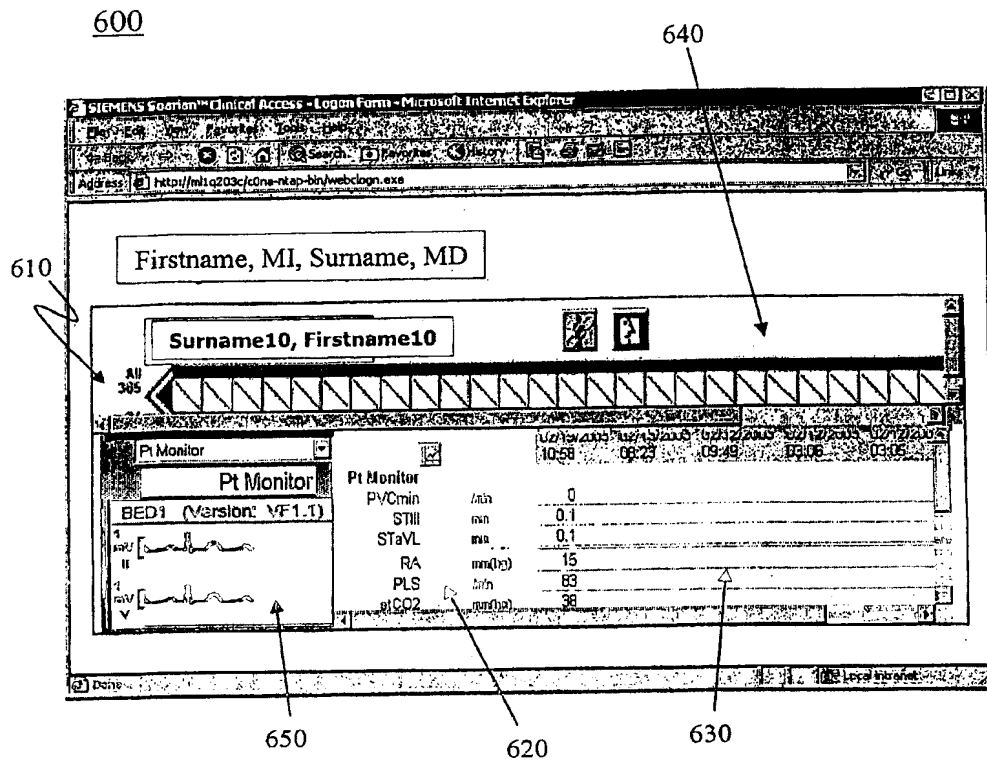


图 6

700

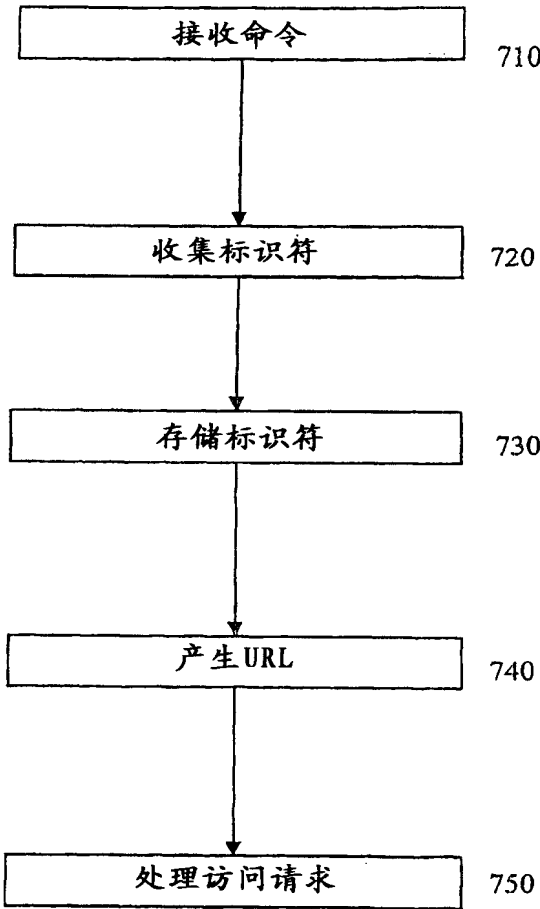


图 7

800

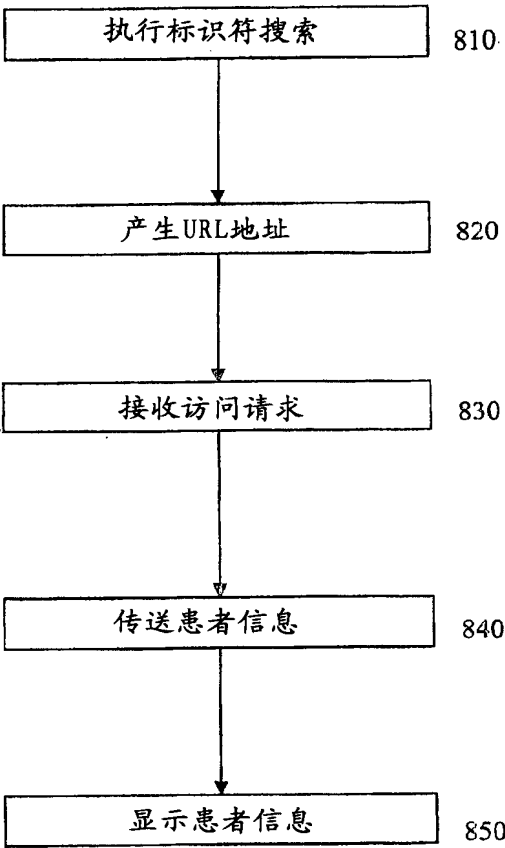


图 8