22764 U.S. PTO 033006

PTO/SB/05 (05-05)

UTILITY PATENT APPLICATION TRANSMITTAL

(Only for new nonprovisional applications under 37 CFR 1.53(b))

Attorney Docket No.	022193-090614US
First Inventor	Tuttle, Timothy
Title	ASYNCHRONOUS MESSAGING USING A NODE SPECIALIZATION ARCHITECTURE IN THE DYNAMIC ROUTING NETWORK
Express Mail Label No.	EV 724262889 US

APPLICATION ELEMENTS

See MPEP chapter 600 concerning utility patent application contents.

1. ☐ **Fee Transmittal Form** (e.g., PTO/SB/17)
   (Submit an original and a duplicate for fee processing)
2. ☐ **Applicant claims small entity status.**
   See 37 CFR 1.27.
3. ☒ **Specification** [*Total Pages* 54]
   Both the claims and abstract must start on a new page
   (For information on the preferred arrangement, see MPEP 608.01(a))
4. ☒ **Drawing(s)** *(35 U.S.C.113)* [*Total Sheets* 12]
5. **Oath or Declaration** [*Total Sheets* 1]
   a. ☐ Newly executed (original or copy)
   b. ☒ A copy from a prior application (37 CFR 1.63 (d))
   (for a continuation/divisional with Box 18 completed)
   i. ☐ DELETION OF INVENTOR(S)
   Signed statement attached deleting inventor(s) named in the prior application, see 37 CFR 1.63(d)(2) and 1.33(b).
6. ☒ **Application Data Sheet.** See 37 CFR 1.76
7. ☐ **CD-ROM** or **CD-R** in duplicate, large table or Computer Program *(Appendix)*
   ☐ Landscape Table on CD
8. **Nucleotide and/or Amino Acid Sequence Submission**
   (if applicable, items a. - c. are required)
   a. Computer Readable Form (CRF)
      i. ☐ Computer Readable Form (CRF)
      ii. ☐ Transfer Request (37 CFR 1.821(e))
   b. Specification Sequence Listing on:
      i. ☐ CD-ROM or CD-R (2 copies); or
      ii. ☐ Paper
   c. ☐ Statements verifying identity of above copies

ADDRESS TO: **Commissioner for Patents**
P.O. Box 1450
Alexandria, VA 22313-1450

112948 U.S. PTO 11/396251 033006

ACCOMPANYING APPLICATION PARTS

9. ☐ **Assignment Papers** (cover sheet & document(s))
   Name of Assignee _____
10. ☐ **37 CFR 3.73(b) Statement** *(when there is an assignee)* ☒ **Power of Attorney**
11. ☐ **English Translation Document** *(if applicable)*
12. ☐ **Information Disclosure Statement** (PTO/SB/08 or PTO-1449)
    ☐ Copies of foreign patent documents, publications, & other information
13. ☐ **Preliminary Amendment**
14. ☒ **Return Receipt Postcard** (MPEP 503)
    (Should be specifically itemized)
15. ☐ **Certified Copy of Priority Document(s)**
    (if foreign priority is claimed)
16. ☐ **Nonpublication Request** under 35 U.S.C. 122 (b)(2)(B)(i). Applicant must attach form PTO/SB/35 or its equivalent.
17. ☐ Other: _____

18. If a CONTINUING APPLICATION, *check appropriate box, and supply the requisite information below and in the first sentence of the specification following the title, or in an Application Data Sheet under 37 CFR 1.76:*

☒ Continuation ☐ Divisional ☐ Continuation-in-part (CIP) of prior application No: 10/105,018

Prior application information: *Examiner* Bharat Barot *Art Unit:* 2155

19. CORRESPONDENCE ADDRESS

☒ The address associated with Customer Number: **20350** *OR* ☐ Correspondence address below

Name					
Address					
City		State		Zip Code	
Country		Telephone		Email Address	

Signature	C. Bart Sullivan	Date	March 30, 2006
Name (Print/Type)	C. Bart Sullivan	Registration No. (Attorney/Agent)	41,516

60735697 v1

REVOCATION OF POWER OF ATTORNEY WITH NEW POWER OF ATTORNEY AND CHANGE OF CORRESPONDENCE ADDRESS		
	Application Number	10/105,018
	Filing Date	March 21, 2002
	First Named Inventor	Tuttle, Timothy
	Art Unit	2151
	Examiner Name	Unassigned
	Attorney Docket Number	022193-090611US

I hereby revoke all previous powers of attorney given in the above-identified application:

☐ A Power of Attorney is submitted herewith.

OR

☒ I hereby appoint the practitioners associated with the Customer Number: **20350**

☒ Please change the correspondence address for the above-identified application to:

☒ The address associated with Customer Number: 20350

OR

☐ Firm *or* Individual Name					
Address					
Address					
City		State		ZIP	
Country					
Telephone		Fax			

I am the:

☐ Applicant/Inventor.

☒ Assignee of record of the entire interest. See 37 CFR 3.71.
Statement under 37 CFR 3.73(b) is enclosed. (Form PTO/SB/96)

SIGNATURE of Applicant or Assignee of Record

Name	Julia Ceffalo		
Signature			
Date	14 May 2003	Telephone	

NOTE: Signatures of all the inventors or assignees of record of the entire interest or their representative(s) are required. Submit multiple forms if more than one signature is required, see below*.

☐ *Total of _____ forms are submitted.

60260349 v1

Attorney Docket No.: 022193-090614US

PATENT APPLICATION

ASYNCHRONOUS MESSAGING USING A NODE SPECIALIZATION ARCHITECTURE IN THE DYNAMIC ROUTING NETWORK

Inventor(s): Timothy Tuttle, a citizen of The United States, residing at
3449 22nd Street, Apt. 8
San Francisco, CA 94110

Karl E. Rumelhart, a citizen of The United States, residing at
555 Washington Avenue
Palo Alto, CA 94301

Entity: Large

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, Eighth Floor
San Francisco, California 94111-3834
Tel: 415-576-0200

ASYNCHRONOUS MESSAGING USING A NODE SPECIALIZATION ARCHITECTURE IN THE DYNAMIC ROUTING NETWORK

INVENTORS

TIMOTHY TUTTLE
KARL E. RUMELHART

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application is a continuation-in-part of U.S. Application No. 10/017,182 filed December 14, 2001, which claims the benefit of U.S. Provisional Application No. 60/256,613, filed December 18, 2000, U.S. Provisional Application No. 60/276,847, filed March 16, 2001, U.S. Provisional Application No. 60/278,303, filed March 21, 2001, U.S. Provisional Application No. 60/279,608, filed March 28, 2001, and U.S. Provisional Application No. 60/280,627, filed March 29, 2001, all of which are hereby incorporated by reference herein.

BACKGROUND

FIELD OF THE INVENTION

[0002] This invention pertains in general to transferring information through digital networks and in particular to transferring information for remotely updating content at client devices through the digital networks.

BACKGROUND ART

[0003] The Internet is a digital network of computers. An individual computer on the Internet is typically identified by an internet protocol (IP) address. A computer on the

Internet sends a packet of information to another computer by routing the packet to a logical port at the destination computer's IP address. The destination computer interprets the packet according to one of several possible protocols determined by the port to which the packet was sent.

[0004] The World Wide Web (the "Web") is a collection of technology and content available on the Internet that allows the content to be routed from server computers to particular destination computers. The Web includes a large number of web pages residing on many different servers. Web pages contain one or more files, or references to one or more files, specifying instructions for presenting the web page and content, such as text, images, applets, video, and/or audio.

[0005] Web pages use a variety of definitional and programming languages to control how information is presented. The most fundamental of these is the Hypertext Markup Language (HTML). HTML uses a system of "tags" to specify how content should be displayed. Recent advances in HTML introduce "style sheets" which help separate content information from display information. HTML has also been modified and extended to provide new capabilities. For example, Extensible Markup Language (XML) adds semantic content to web pages. In addition, Dynamic HTML (DHTML) adds some dynamic content to web pages.

[0006] A web page may also include one or more programs for controlling how the web page is displayed. For example, JAVA® applets and JAVASCRIPT® scripts may be used to control the display of a web page. In addition, DHTML uses scripts to control the dynamic content. Thus, a web page designer can use applets and scripts to produce

animation effects or modify the display based on user interaction. For example, the designer can write a script that changes the color of a piece of text when a user clicks on a button.

[0007] Devices that display/execute web pages are often called "client devices" or simply "clients." Client devices include personal computers, web-enabled set-top boxes and televisions, cellular telephones, personal digital assistants and other handheld devices, and special-purpose web-browsing appliances. Client devices typically employ a program called a "web browser" for interpreting the HTML or other display instructions in the web page and displaying the content accordingly. Most web browsers include special functionality, such as a Java Virtual Machine, for executing JAVA® applets and/or other applets or scripts embedded in the web pages.

[0008] A client device specifies a web page or other document on the web using a Uniform Resource Locator (URL). A URL has the form "service://server/path/file." Here "service" refers to the protocol to be used, such as the file transfer protocol (FTP) or the hypertext transport protocol (HTTP). "Server" is the IP address of the server containing the page, and "path/file" specifies the particular web page on the server.

[0009] The Web suffers from a substantial limitation with respect to dynamically updating content in a web page at a client device. The Web's only mode of operation is for a client device to first request a page from a server and then for the server to send the requested page to the client device. Once the server delivers the page to the client, it typically terminates its connection to the client, and does not retain any information about the client or the page that was sent. For this reason, servers are typically "stateless." As

a result, client devices drive and control the flow of information around the Web. While client-side control is appropriate in some situations, it does not permit efficient updating of data at the client devices. For example, if a web page contains information that may change, such as the score of a baseball game or a stock quote, the server has no way to inform the client devices that are viewing the page of the change. Instead, the client devices must ask the server for the updated information. However, the client devices do not know when the information on the web page has changed, and thus do not know to ask for the update.

[0010] There are some simple web programming techniques that attempt to update content on client device-side web pages. One approach that web designers use is to rely on the client devices to periodically re-request web pages. This updating can be performed as the result of user action (such as pressing the "refresh" button) or can be automated to occur on a particular schedule (such as by using the HTML Meta Refresh tag to cause the client device to request the page every 'X' seconds). Although this technique provides client devices with more up-to-date information, it is very wasteful of resources. In particular, the web server must resend the page even if nothing has changed, and, even when something has changed, it must resend the entire web page rather than just the updated information, which may be only a very small part of the page. Further, attempting to reduce unnecessary requests by decreasing the request rate results in decreasing the currency of the data. This is an unalterable trade off in a client-driven approach.

[0011] The performance of automatic refreshing can be improved somewhat by putting information that may change in a separate frame from information that is less

likely to change, and only refreshing the separate frame. A few web designers even write custom JAVA applets to limit refreshing to individual components on a page, such as the score of a soccer game. A willingness to go to such effort illustrates the serious drain of resources caused by frequent refreshing. Nevertheless, even custom JAVA applets are not a meaningful attack on this problem. Custom applets require a large separate development effort for each item on each page that might need to be updated. More importantly, most custom applets still update content based upon client-driven requests, although it is possible to design an applet that accepts "pushed" messages. This solution is not scalable to provide updated information for large numbers of client devices and for large numbers of web pages.

[0012] Therefore, there is a need in the art for an efficient way to provide dynamic content to a web page at a client device.

DISCLOSURE OF THE INVENTION

[0013] The above need is met by a dynamic content routing network that routes messages containing data for updating properties of live objects to clients displaying web pages or other representations of data containing the live objects. The web server that initially provides the web pages to the clients does not need to track which clients are currently displaying the live objects. Instead, the information provider or a dynamic content provider (generically referred to as an "input source") that provided the live object simply sends an update message to the routing network. This routing utilizes bandwidth efficiently because the update messages are provided to the clients only when the live objects change.

[0014] The routing network is adapted to selectively send messages to the nodes in the network. In one embodiment, a hierarchy of registrations is used. Each gateway in the routing network maintains the mappings between the live objects and the nodes that have registered for the live objects. Each node in the routing network, in turn, maintains the mappings between the live objects and the clients that display them. An input source provides a message to a gateway in each cluster in the routing network. Each gateway forwards to each node only messages that reference the objects for which it has registered. Each node forwards to each client only messages that reference the objects for which it has registered. Adding node functionality to the gateway and client functionality to the node advantageously allows the routing network to decide which nodes should receive an update message. As a result, messages are sent to only nodes that have registered for the messages. Furthermore, each node receives all the messages that the clients connected to that node are interested in.

[0015] In another embodiment, all messages from an input source are assigned to one or more of N categories. Also, the nodes are assigned to one or more of M types, and mappings are created between message categories and node types. Each gateway keeps track of these mappings. When a gateway receives messages from input sources, the gateway identifies the categories of the messages and routes the messages to the nodes of the types to which the categories are mapped. To ensure that clients have access to the messages they need, clients are allowed to communicate with nodes of several types using client proxies connected between the clients and the nodes. There are at least two ways to implement the client proxy embodiment. The implementations differ primarily in where the client registration information is stored. In the first implementation, client registration information is stored at the nodes, and the client proxy merely passes messages among the clients and nodes. When a node directs a message to the client, the node passes the message to the client proxy along with a pointer to the client socket. When the client proxy receives the message, it simply pushes it to the client socket.

[0016] In the second implementation of the client proxy embodiment, the client proxy stores client registration information. In this embodiment, each node stores the object IDs registered by client proxies connected to the node. The client proxy registers with the nodes for all objects for which it needs to receive updates. The nodes receive the messages from the input sources, determine which client proxies have registered for the messages, and send the messages to the appropriate client proxies. The client proxies, in turn, transmit the messages to the clients that have registered for the messages.

[0017] The features and advantages described in this summary and the following detailed description are not all-inclusive, and particularly, many additional features and

advantages will be apparent to one of ordinary skill in the art in view of the drawings, specification, and claims hereof.

BRIEF DESCRIPTION OF THE DRAWINGS

[0018] FIG. 1 is a high-level block diagram illustrating an environment containing a dynamic content routing network;

[0019] FIG. 2 is an interaction diagram illustrating interactions among a server, information provider, dynamic content provider, client, and routing network to update a property of a live object on a web page;

[0020] FIG. 3 is a high-level diagram graphically indicating the many-to-many mapping performed by the routing network;

[0021] FIG. 4 illustrates two different web pages containing sports scores;

[0022] FIG. 5 is a block diagram illustrating an input source and the tools available to it for generating the update messages;

[0023] FIG. 6 is a flow chart illustrating the steps performed by an embodiment of an activation module;

[0024] FIG. 7 is a block diagram illustrating a lower-level view of the routing network according to an embodiment of the present invention;

[0025] FIG. 8 is a flow chart illustrating steps performed by a gateway and a node in a cluster to perform object-based routing of a message received from an input source in an embodiment using a hierarchy of registries;

[0026] FIG. 9 is a block diagram illustrating a high-level view of the routing network in an embodiment adapted to use message categories, node types, and client proxies;

[0027] FIG. 10 is a flow chart illustrating steps performed by a gateway, a node that stores client registration information, and a pass-through client proxy to perform object-based routing of a message received from an input source;

[0028] FIG. 11 is a block diagram illustrating a high-level view of the routing network for an embodiment in which the client proxy stores the client registration information; and

[0029] FIG. 12 is a block diagram illustrating a high-level view of the routing network for an embodiment in which the nodes adopt client proxy functionality.

[0030] The figures depict an embodiment of the present invention for purposes of illustration only. One skilled in the art will readily recognize from the following description that alternative embodiments of the structures and methods illustrated herein may be employed without departing from the principles of the invention described herein.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0031] FIG. 1 is a high-level block diagram illustrating an environment 100 containing a dynamic content routing network 110 (hereafter referred to as the "routing

network"). The environment 100 also contains a server 112 in communication with a client 114, an information provider 108, and a dynamic content provider 116. Although a typical environment 100 will have hundreds of servers 112 and information providers 108, thousands (or even millions) of clients 114, and multiple dynamic content providers 116, FIG. 1 illustrates only one of each of these entities in order to enhance the clarity of this description.

[0032] The server 112, client 114, information provider 108, dynamic content provider 116, and routing network 110 are preferably in communication via conventional communications links 117 such as those comprising the Internet. The communications links 117 include known wired communications media, such as dedicated or shared data, cable television or telephone lines, and/or known wireless communications media, such as communications over the cellular telephone network using protocols such as the global system for mobile communications (GSM), code division multiple access (CDMA), time division multiple access (TDMA), etc.

[0033] In one embodiment, the entities may each be in communication with one or more Internet Service Providers (ISPs) (not shown) that provide each entity with access to other computers on the Internet. In addition, the server 112, client 114, information provider 108, dynamic content provider 116, and routing network 110 are preferably each identified by at least one Internet Protocol (IP) address such as "66.35.209.224." The IP address may also have one or more domain names associated with it, such as "bangnetworks.com." Alternative embodiments of the present invention may use alternative addressing schemes and/or naming conventions instead of, or in addition to, those described herein. For example, embodiments wherein one or more of the clients

are cellular telephones or other portable devices may rely on different addressing schemes.

[0034] Preferably, the information provider 108 provides web pages or other representations of data to the server 112. The web pages contain one or more "live objects," which are designated to be real-time dynamically-updateable objects. Each live object is identified by an object identifier, or object ID. Preferably, the server 112 provides the web pages 118 to multiple clients 114. The clients 114 contact the routing network 110 and register for update messages for the object IDs on the web page. The routing network 110, in turn, preferably maintains a registry indicating which clients have registered for which object IDs.

[0035] The information provider 108 and/or dynamic content provider 116 send update messages to the routing network 110. These messages can be sent any time the information provider 108 or dynamic content provider 116 wants to update a property of a live object. Each update message preferably identifies a live object and contains data for updating a property of the identified live object. The routing network 110 accesses the registry and determines which clients have registered for the identified object. Then, the routing network 110 routes the update message to the appropriate clients. Upon receipt of an update message, the clients 114 update the specified property of the live object.

[0036] The routing network 110 provides an efficient one-to-many mapping of objects to clients (and by inference of information, a many-to-many mapping of information providers 108/dynamic content providers 116 to clients) through object-

based routing. Messages provided by the information provider 108 and/or dynamic content provider 116 to the routing network 110 are not routed to the clients 114 based entirely on a specified destination; more specifically, they are not routed based on the IP address of the client, as in conventional IP routing schemes. Instead, the messages are routed based on the live objects referenced by the message.

[0037] The mapping and object-based routing provided by the routing network 110 allow the information provider 108 and dynamic content provider 116 to update properties of live objects at a dynamically changing cross-section of clients in real-time, without requiring the information provider or dynamic content provider to track the clients or web pages being viewed by the clients. The clients 114, in turn, do not need to have any a priori knowledge of object IDs – they "discover" which IDs they should register when they receives the web pages 118 from the server 112.

[0038] Object-based routing also allows information providers 108 to dynamically update content on web pages without requiring the clients 114 to re-request the content, and without requiring the information providers 108 or servers 112 to maintain connections with the clients. In this manner, significantly more clients can receive updated content from a given information provider 108 than would be possible utilizing conventional client-side request-driven transmission control protocol/Internet Protocol (TCP/IP) connections between the clients and the server 112.

[0039] Turning now to the individual entities illustrated in Fig. 1, the server 112 is preferably a conventional computer system configured to act as a web server and serves

web pages 118 and other data representations to clients 114. The web pages 118 provided by the server 112 are associated with one or more information providers 108.

[0040] An information provider 108 is an entity providing one or more web pages 118, information contained in web pages, and/or other representations of data served by the server 112. The information provider 108 preferably has a conventional computer system coupled to the Internet. In one embodiment, the server 112 is directly controlled by the information provider 108 (e.g., the server is physically located at the information provider and/or is dedicated to serving only the information provider's web pages). In this embodiment, the server 112 and information provider 108 can be treated as the same entity. In an alternative embodiment, the server 112 serves web pages from multiple information providers.

[0041] As is known in the art, the web pages 118 and other content on the server 112 are specified by uniform resource locators (URLs) having the form "service://server/path/web page." Typically, web pages 118 are obtained via the hypertext transport protocol (HTTP) and thus an exemplary URL for retrieving the web page "b1.html" from the web server having the domain name "www.bangnetworks.com" is "http://www.bangnetworks.com/news/b1.html."

[0042] As used herein, a "web page" is a block of data available from the server 112. In the simplest case, a web page is a file written in the hypertext markup language (HTML). The web page may also contain or refer to one or more other blocks of data, such as other files, text, images, applets, video, and/or audio. In addition, the web page may contain instructions for presenting the web page and its content, such as HTML tags

and style sheets. The instructions may also be in the Extensible Markup Language (XML), which is related to HTML and adds semantic content to web pages or the Dynamic HTML (DHTML), which adds some dynamic content to web pages. Additionally, the instructions may take the form of one or more programs such as JAVA® applets and JAVASCRIPT® and/or DHTML scripts.

[0043] As used herein, the phrase "web page" also refers to other representations of data served by the server 112 regardless of whether these data representations include characteristics of conventional web pages. These data representations include, for example, application programs and data intended for the web browser 120 or other application programs residing at the clients 114 or elsewhere, such as spreadsheet or textual (e.g., word processing) data, etc.

[0044] In a preferred embodiment, objects at the client, such as web pages and elements of web pages, can be designated as "live" by the information provider 108. Properties of a live object can be dynamically updated in real-time at the client 114 by the information provider 108 or another entity acting on behalf of the information provider. As used herein, an "object" is any datum or data at the client 114 that can be individually identified or accessed. Examples of objects include elements of web pages such as text characters and strings, images, frames, tables, audio, video, applets, scripts, HTML, XML, and other code forming the web page, variables and other information used by applets, scripts and/or code, URLs embedded in the web page, etc. Application and operating system constructs are also objects. For example, cells of spreadsheets, text in word processor documents, and title bars and messages displayed by the operating system or applications are objects. Preferably, multiple objects can be grouped together into a

single, logical object. Thus, an object can be defined at any desired or useful level of granularity.

[0045] Since content on a web page is conceptualized and organized by "object," the present invention essentially abstracts web pages and web page content, and other modules and/or functionality at the client 114, away from the HTML code or other conventional representation. This abstraction allows the information provider 108 to update a property of an object without concern for the location, display format, or other specifics of how the data is being represented at the client 114.

[0046] Live objects have associated "properties" which include any modifiable data related to the object or referenced with respect to the object. The information provider 108 typically, but not necessarily, provides initial settings for the properties of live objects provided to the client 114. The properties may or may not affect the visual representation of the object in the web page or other data representation. A property may affect an internal aspect of the object and, thus, a change to the property may not have any direct effect on a web page containing the object. For example, the property may affect whether particular aspects of the object are modifiable, how the object responds to user input or other stimuli, etc. Additionally, a property may also have a direct effect on how the object is displayed at the client 114. For example, the property may affect the content, color, typeface, size, formatting, or other attribute of text, images, or other data displayed by the object. Other properties may occupy parts of the spectrum between having no effect on the visible representation of the object and having a direct effect on the visible representation of the object. For example, a web page showing scores of football games may include a list of games and the current scores of the games as of the

time the server 112 serves the web page. The list of games, subset of games to be displayed, and the scores of the games can be designated as live objects (or properties of a single live object) and updated as necessary or desired.

[0047] A property can also preferably include instantiating an instance of the object or invoking functionality of the object. For example, a property of a browser window object may include functionality for instantiating another browser window. This function can be invoked as a logical change to a property of the object. The second browser window can be referenced through the original browser window (i.e., object) or designated as a new live object.

[0048] An information provider 108 or other entity preferably updates a live object at a client 114 via an update message. In general, an update message identifies the live object and, if necessary, the property of the live object, and contains data for updating the property. In one embodiment, the data may be the actual value for the property or executable code for causing the object's property to be updated. For example, the data may be a simple numerical or textual value, e.g., "4," to which the property should be set, and/or the data may be JAVASCRIPT® code or a call to a JAVASCRIPT® function at the client that effects the desired change to the property of the object.

[0049] The update message preferably implicitly or explicitly identifies a handler at the client 114 for use in updating the live object's property. In one embodiment, the client 114 utilizes a default handler when the message implicitly specifies the handler (e.g. when the message does not identify a specific handler). In one embodiment, if the update message specifies the actual value for the property, a default handler generates

JAVASCRIPT® code for changing the property to the specified value. If the data in the update message are JAVASCRIPT® code, the default handler does not perform any processing of the code. In either case, the default handlers preferably use LiveConnect to execute the JAVASCRIPT® code in a Java Virtual Machine (JVM) 122 at the client 114 and thereby update the property of the live object.

[0050] For certain objects and/or data types, the default handlers are not appropriate. In these cases, the message preferably explicitly identifies a handler for performing the update. For example, the message may explicitly specify a function to call on the data or the message may explicitly identify the environment in which the data should be executed. For example, the data in the update message may include code for execution by a software "plug-in" such as MACROMEDIA FLASH® and the message may explicitly identify FLASH as the handler.

[0051] The information provider 108 preferably designates an object as "live" by including a unique identifier for the object, the object ID, in the web page or other data representation provided to the client 114. In one embodiment, the information provider 108 encodes the object ID in an object's corresponding HTML "ID" attribute using the following HTML expression:

ID = "elementIdentifier,"

where "elementIdentifier" is the object ID and is preferably a string. The string can encode any information desired by the information provider 108 or other entity establishing the object ID and in one embodiment is a simple textual and/or numeric identifier. In one embodiment, the information provider 108 begins the object ID with a

predefined token, such as "Bang$," in order to distinguish live objects from other objects that happen to have defined ID attributes. For example, an object can have the object ID "Bang$elementIdentifier."

[0052] In the preferred embodiment, each information provider 108 optionally encodes a unique information provider ID in its object IDs in order to prevent naming collisions between the object IDs of different information providers. In one embodiment, the information provider ID is a textual and/or numeric identifier. The information provider 108 may specify the information provider ID and the object ID as part of a hierarchical namespace. For example, in one embodiment objects are named as follows: "$namespace1$[namespace2$...$namespaceN$]objectId," where "$namespace1" is the information provider ID and the "$" operates as the name separator and defines additional optional levels of a namespace hierarchy. One embodiment of the system 100 supports typical directory services functionality. For example, two dollar sign characters appearing together, "$$," refers to the top level of the namespace hierarchy.

[0053] Thus, the object ID for a live object is preferably formed from a combination of the predefined token, the information provider ID namespace, and a value assigned by the information provider 108. For example, the object ID for a live object representing the real time price of a stock having the symbol "BANG" might be: "Bang$$informationProviderID$equities$realtime$bang." In this example, "Bang$" is the predefined token that signifies a live object, "$informationProviderID" is the ID identifying the information provider, "$equities$realtime$" defines levels of a namespace hierarchy, and "bang" identifies the specific object.

[0054] In some embodiments and situations, the object ID utilizes relative names. For example, an information provider 108 referring to its own object IDs is implicitly in its own namespace. Accordingly, the information provider 108 does not need to include the information Provider ID in the object IDs it utilizes internally. In one embodiment, the information provider ID is not explicitly encoded into the object ID. Instead, the information provider ID is encoded elsewhere in the web page in order to provide scope to the page's object IDs.

[0055] In one embodiment, the object ID identifies a point (i.e., a node in a tree) in a Document Object Model (DOM) representation of a web page or other document at the client 114. The DOM is a platform- and language-neutral interface that represents a document as a hierarchy of objects. The DOM also provides an interface that allows programs and scripts to dynamically access and update properties of the objects. Object properties can be inherited by descendent objects.

[0056] In this embodiment, the client 114 preferably executes an update message in the context of the specified point in the DOM representation. The update may specify a change to a property of the object at the identified point. The update also may specify a change to a parent or descendent of the object at the identified point. In each case, the update is executed relative to the specified point in the DOM representation. In one embodiment, points in the DOM representation specify how to update properties of live objects located at those points. Thus, the same update may be interpreted differently depending upon the identified live object's location in the DOM representation.

[0057] For example, assume there is an object in the DOM representation identified as "window.document.frame[3].ObjectID." Also assume that the object has an "innerText" property located at "window.document.frame[3].ObjectID.innerText" that specifies the text displayed by the object. An update message can change the text displayed by the object by specifying "ObjectID" and the new value for the innerText property.

[0058] An advantage of utilizing object IDs to specify objects is that the information provider 108 or other entity providing the update message can access and change properties of objects without knowing the object's actual location in the DOM representation. Indeed, the object may be in different locations in different DOM representations and/or in multiple locations in the same DOM representation. In any of these cases, the update message will change the specified properties of all of the objects having the given object ID.

[0059] Depending upon the particular embodiment of the environment 100, the information provider 108 and/or the dynamic content provider 116 provides update messages to the routing network 110. The dynamic content provider 116 is preferably a conventional computer system operated by an entity that provides real-time information, such as stock prices and/or sports scores. In one embodiment, the information provider 108 receives updated properties for the live objects from the dynamic content provider 116 or another source (or generates the updated properties internally). Then, the information provider 108 sends an update message specifying the object ID and the change to the object property to the routing network 110. In this embodiment, the dynamic content provider 116 may be absent from the environment 100.

[0060] In another embodiment, the dynamic content provider 116 provides the object IDs for live objects to one or more information providers 108 and the information providers 108 distribute the live objects to the clients 114. Then, the dynamic content provider 116 sends messages specifying the changes to the properties of the live objects to the routing network 110. For example, the dynamic content provider 116 distributes an object ID associated with the score of a particular baseball game to the information providers 108. Then, the dynamic content provider 116 sends a message specifying the object ID and an update to a property of the object that controls the displayed score of the particular baseball game to the routing network 110. These two embodiments are not mutually exclusive and, therefore, some updates may be provided to the routing network 110 by the information provider 108 while others are provided by the dynamic content provider 116.

[0061] The client 114 is a device that retrieves web pages 118 and/or other information from the server 112. In one embodiment, the client 114 is a conventional personal computer used by a person to access information on the Internet. In alternative embodiments, the client 114 is a different consumer electronic device having Internet connectivity, such as an Internet-enabled television, a cellular telephone, a personal digital assistant (PDA), a web browsing appliance, etc. The client 114 preferably, but not necessarily, has an associated display device.

[0062] The client 114 preferably executes a web browser 120, such as MICROSOFT INTERNET EXPLORER®, for retrieving web pages and displaying them on the display device. In embodiments where the client receives data representations from the server 112 other than conventional web pages, the web browser 120 does not necessarily share

similarities with conventional web browsers. Preferably, the web browser 120 contains a JVM 122 for executing JAVA® applets and/or scripts. The web browser 120 also preferably contains Dynamic HTML capabilities, such as support for JAVASCRIPT® (or another scripting language, such as VBScript) and the Document Object Model (DOM), and enables communications between JAVA® and the scripting languages. In one embodiment, the web browser 120 supports the LiveConnect standard for enabling communication between JAVA® applets and scripts written in the supported scripting languages. The web browser 120 can also be extended through software plug-ins such as MACROMEDIA FLASH®, REAL NETWORKS REALPLAYER®, and/or APPLE QUICKTIME®. In alternative embodiments, the functionality of the JVM 122 and/or other aspects of the web browser 120 are provided by one or more other functional units within the client 114. The term "module" is used herein to refer to software computer program code and/or any hardware or circuitry utilized to provide the functionality attributed to the module. The web browser 120 and JVM 122 are examples of modules in the client 114.

[0063] In some embodiments, the client 114 does not necessarily have a display device, web browser 120 and/or other components associated with a typical consumer device. The client 114, for example, may be a dedicated purpose device having certain aspects of web connectivity such as an embedded HTTP client in a web-enabled appliance or in a controller for an automobile, audio-visual equipment, or some other device.

[0064] A web page 118 provided from the server 112 to the client 114 preferably includes instructions for enabling the live objects on the web page. The instructions

cause the client 114 to automatically and transparently (i.e., without user interaction) contact the routing network 110 and download an activation module 124 for activating the live objects. In one embodiment, the instructions comprise a URL specifying the location of the activation module 124 at the routing network 110. In an alternative embodiment, the client 114 obtains the activation module 124 from the server 112 or another source.

[0065] The activation module 124 preferably contains JAVA® instructions for execution by the JVM 122. However, alternative embodiments of the module 124 may encode the instructions in the web page 118 and/or the activation module 124 using different languages and/or techniques. For example, the instructions and/or activation module 124 can be embedded in the web browser 120 or operating system, either as native code or as plug-ins. In these alternative embodiments, the web browser 120 does not have to download the activation module 124 from an external source.

[0066] The activation module 124 preferably registers object IDs from the web page 118 downloaded by the client 114 with the routing network 110 and updates the live objects in response to update messages received from the network. The routing network 110 records the registrations in the registry 125. The client's registrations preferably remain in effect as long as the client is displaying the associated web page 118, although other embodiments of the system 100 may use different criteria for determining when to terminate the client's registrations.

[0067] FIG. 2 is an interaction diagram illustrating interactions among the server 112, information provider 108/dynamic content provider 116 (generically referred to as an

"input source 210"), client 114, and the routing network 110 to update a property of a live object. Initially, the client 114 sends 212 a web page request to the server 112. In response, the server 112 provides 214 to the client 114 the web page containing or otherwise identifying the one or more live objects. Instructions encoded in the web page preferably cause the client 114 to transparently request 216 the activation module 124 from the routing network 110. In response, the routing network 110 sends 218 the activation module 124. The client 114 executes 220 the activation module 124, which identifies the object IDs of the live objects at the client and registers 222 the object IDs with the routing network 110. The routing network 110 updates 223 its registry to identify the object IDs for which the client 114 has registered.

[0068] At some point, the input source 210 sends 224 an update message to the routing network 110 in order to change a property of a live object at the client 114. In one embodiment, the message from the input source 210 to the routing network 110 contains only a single object ID and an update to a property of the identified object. In another embodiment, the message contains multiple object IDs and the corresponding property updates. In this latter embodiment, the message may have an associated "Batch ID" that identifies the message as having multiple object IDs and updates. Preferably, the information provider 108 can include a batch ID in a web page 118 in the same manner as including an object ID. Likewise, the client 114 can preferably register for a batch ID with the routing network 110 in the same manner as an object ID. In fact, the batch ID can be the same as the object ID so that the client 114 registers for both batch and non-batch messages by registering one ID. Alternatively, separate procedures can be

established for registering batch messages. The client 114 preferably processes the component messages of a batch as if each message were delivered separately.

[0069] The routing network 110, in turn, routes 226 the message to each client 114 that has registered for the specified object ID, preferably by utilizing standard Internet communications protocols, such as IP addresses, etc. The activation module 124 at the client 114 processes the message and updates 228 the property of the identified live object. If live objects having the same object ID appear in multiple locations at the client 114 (e.g., at multiple locations on a web page being displayed at the client), the activation module 124 preferably updates each of the live objects having the specified ID. As a result, the routing network 110 allows live objects at the client 114 to be dynamically updated. Preferably, this routing and updating happens quickly enough to be considered "real-time" for the purposes of the input source 210.

[0070] This update process, indicated within the dashed box 230 in FIG. 2, can repeat an indefinite number of times and is fully asynchronous as to the information provider 210 and client 114. For example, the input source 210 may send regular update messages to the routing network 110 as the score of a sporting event changes or a stock price fluctuates, but may stop sending update messages once the sporting event ends or stock market closes. When the client 114 ends the display of a web page containing the live object, or otherwise no longer desires to receive update messages, the client preferably closes 232 the connection with the routing network 110. The routing network 110, in turn, updates 234 the registry 125 to remove the client's object registrations. In another embodiment, the client 114 sends messages to the routing network 110 that selectively

register and/or de-register the client from one or more objects yet leaves the connection open in order to receive update messages pertaining to other objects.

[0071] FIG. 3 is a high-level diagram graphically indicating the many-to-many mapping performed by the routing network 110. Multiple input sources (labeled 210A-C) send update messages to the routing network 110. Each update message preferably specifies at least one object ID and an update to a property of the identified object. The routing network 110, in turn, selectively routes the update messages to the clients 114 that have registered for the given object ID from the given input source 210. In FIG. 3, assume for example that clients 312A and 312B have registered for a given object ID while the other clients have not registered for the object ID. Accordingly, the routing network 110 routes the update message to clients 312A and 312B, but does not route the message to clients 312C-312H.

[0072] FIG. 4 illustrates an example of the capabilities of the dynamic content routing network 110. FIG. 4 illustrates two different web pages 410, 412 containing sports scores. Although the web pages are formatted differently, each page contains the same scores for two professional football games and two professional baseball games. Web page 410 contains all four games under the heading "Local Sports Scores" while web page 412 contains the baseball games under the heading "Baseball Scores" and the football games under the heading "Football Scores."

[0073] There are various ways to internally represent the games and scores in the web pages using live objects. In one embodiment, a "game" object is defined having properties for the two teams involved in the game and the score associated with each

team. The game object is placed at a selected position in the web page and the properties of the object cause the information about the game to be displayed on the page. In another embodiment, "team" and "score" objects are defined, with the team object having a property defining the name of a team and the score object having a property defining a score. In this second embodiment, the team and score objects are placed at selected locations on the page so that the proper teams and scores are aligned when the page is rendered. In yet another embodiment, an object is defined having properties for the name of one team and a score associated with that team. Then, pairs of the objects are placed in the page in the proper alignment to indicate the games and scores. In another embodiment, an object is defined having properties specifying names of two teams and a separate object is defined having properties specifying two scores. In this last embodiment, the two objects are placed in the page so that the names of the teams align with the associated scores. Obviously, additional variations of these representations are possible.

[0074] Assume for the example of FIG. 4 that the names of teams in a game are specified by a "names" object having properties for the two team names and the scores in the game are specified by a "scores" object having properties for two scores. In web page 410, a names object 414 having properties set to identify the "SF 49ers" and the "STL Rams" is located directly under the "Local Sports Scores" heading. A scores object 416 having a property set to identify the score of the game as "42" to "7" is directly to the right of the names object 414. In web page 412, the properties of the second names object 418 identify the same game using slightly different terminology:

"SF" and "STL." However, this names object 418 is aligned with the same scores object 416 as is utilized in web page 410.

[0075] Thus, the same scores object 416 is utilized in different positions in each web page 410, 412. In order to update the score of the San Francisco 49ers vs. St. Louis Rams football game on both web pages, the input source 210 simply sends an update message to the routing network 110 specifying the object ID for the scores object 416 and the update to the score property. The routing network 110 routes the update message to the appropriate clients 114, and the clients update the appropriate score regardless of the particular page layout.

[0076] The input source 210, i.e., the information provider 108 and/or dynamic content provider 116 can use a variety of tools to generate the update messages. FIG. 5 is a block diagram illustrating an input source 210 and the tools available to it for generating the update messages. Other tools can be utilized in addition to or instead of the ones described herein.

[0077] Preferably, the tools allow the input source 210 to access an application programming interface (API) provided by the routing network 110 for accepting messages. In one embodiment, the messages sent by the input source 210 are in the same format as utilized by the activation module 124 at the client 114. In an alternative embodiment, the messages provided to the routing network 110 are in a different format and the routing network translates the messages into the format utilized by the activation module 124.

[0078] In one embodiment, the input source 210 utilizes a data pump module 510 to access the API. The data pump module 510 reads an extensible markup language (XML) file containing one or more object IDs and the new values for the identified objects at regular intervals and automatically generates API calls that send messages representing changes to object properties to the routing network 110. In another embodiment, the data pump module 510 is event-driven and reads the XML file in response to a change in the file or some other occurrence.

[0079] In another embodiment, the input source 210 utilizes a director console module 512 to access the API. Preferably, the director console module 512 presents an administrator with a graphical interface displaying the contents of the web page 118. For example, the administrator may use the director console 512 to edit textual data, images, and/or any objects or properties of objects on the web page. After editing, the administrator uses a "send update" button or similar technique to cause the director console module 512 to send messages for the changed objects and properties to the routing network 110 via the API.

[0080] In another embodiment, the information provider 108 and dynamic content provider 116 work together as the input source 210 by using a content management system module 514 to access the API. Preferably, the content management system module 514 resides at the information provider 108 and receives object property updates from the dynamic content provider 116. The content management system module 514 preferably updates the properties of the live objects in the web page 118 stored at the server 112 and also sends messages for the changed properties to the routing network 110. In this manner, the web page 118 at the server 112 and the web page displayed at

the client 114 are updated almost simultaneously. In one embodiment, the dynamic content provider 116 sends the update messages to the routing network 110 instead of to the information provider 108. Embodiments of the system 100 can also utilize any combination of the content management techniques described herein.

[0081] For example, the tools described above can generate a message having the following code for updating the text displayed by a score object to "2":

```
LiveObject score = new LiveObject("Bang$homeScoreID");
score.setProperty("innerText", "2").
```

This code sets the innerText property of the object having object ID "Bang$homeScoreID" to "2." The tools use the API to pass this message to the routing network 110.

[0082] Turning now to the actions performed at the client 114, FIG. 6 is a flow chart illustrating the steps performed by an embodiment of the activation module 124. Those of skill in the art will recognize that different embodiments may perform the steps of FIG. 6 in different orders. The activation module 124 generally performs three functions: register object IDs with the routing network 110, handle messages received by the client 114 from the network in order to update the properties of live objects, and control communications between the client and the network.

[0083] In order to register object IDs, the activation module 124 preferably parses 610 the web page 118 received from the server 112 and identifies the object IDs of the live objects. In an alternative embodiment, the activation module 124 identifies only a subset of the object IDs, such as the IDs of only live objects that are currently being displayed by the web browser 120. Alternatively, a list of object IDs may be pre-encoded

in the web page in addition to the objects themselves, thereby enabling easy identification by the activation module 124. In yet another embodiment, a user of the client 114 selects the object IDs to register.

[0084] The activation module 124 preferably opens 612 a connection between the client 114 and the routing network 110. The activation module 124 can open 612 this connection before or after the activation module receives and/or parses the web page 118. In some cases, the client 114 is located behind a firewall that puts a restriction on the types of connection requests the client can make. A firewall might, for example, block all non-HTTP traffic. For this reason, the activation module 124 preferably wraps the connection request in an HTTP header in order to get the request to the routing network 110 through the firewall.

[0085] The activation module 124 uses the connection between the client 114 and routing network 110 to register 614 the object IDs by communicating to the routing network 116 a vector (e.g., a list or array) containing the identified object IDs. In order to accomplish this task through the firewall, the activation module 124 preferably puts the vector into a string, referred to as "object data," and then preferably creates an HTTP message to communicate the object data to the routing network 110. A schematic example is as follows:

POST / HTTP/1.1\r\n
Content-Length: <length of object data> \r\n
\r\n
<object data>

IPR2025-00200 / 722 Patent Page 34 of 1171 EX1002

where <object data> is the object ID list. When the routing network 110 receives such an HTTP request, it extracts the object data and updates the registry 125 to indicate that the client 114 has registered for the identified objects.

[0086] If the web browser 120 loads 616 a new page, or otherwise terminates display of the objects on the initial page, the activation module 124 associated with the initial web page preferably terminates 618 the client's connection with the routing network 110. Those of skill in the art will recognize that this termination 618 can occur asynchronously with the other steps illustrated in FIG. 6. Thus, the location of steps 616 and 618 represents only one possible place in the sequence of steps where the termination may occur.

[0087] If the connection is not terminated, the activation module 124 preferably waits until it receives 618 a message from the routing network 110 specifying an object ID and an update to a property of the identified object. In one embodiment, this message is received as HTTP data. Upon receipt of the message, the activation module 124 preferably extracts 620 the object ID and update from the HTTP data. Then, the activation module 124 updates 622 the property of the identified object, or causes the object to be updated, as specified by the message.

[0088] The sequence of receiving messages 618, extracting data 620, and updating objects 622 is preferably repeated until a new page is loaded 616 or the connection with the routing network 110 is otherwise terminated. Although not shown in FIG. 6, in certain circumstances, such as when a user action with respect to the web page 118 activates a new live object, the activation module 124 may register new object IDs with

the routing network 110 without first downloading and parsing a new page. In one embodiment, if the newly-loaded page contains live objects, then the process of downloading the activation module 124 and updating the objects as described by FIG. 6 is repeated. In an alternative embodiment, the activation module 124 remains active at the client 114 and, therefore, the client does not re-download the activation module from the routing network 110. Instead, the already-present activation module 124 performs the live-enabling process on the new page.

[0089] FIG. 7 is a block diagram illustrating a lower-level view of the routing network 110 according to one embodiment of the present invention. FIG. 7 illustrates multiple input sources (labeled 710A-D) representative of sources providing messages to the routing network 110, such as an information provider 710A and a dynamic content provider 710B. FIG. 7 also illustrates multiple clients (labeled 712A-F) representative of the many clients in communication with the routing network 110 at any given instant.

[0090] Internally, the routing network 110 is preferably divided into one or more clusters 714. In FIG. 7, the routing network 110 has three clusters 714A, 714B, 714C, although the number of clusters can vary depending upon the processing needs of the network. An input-side global load balancer 716 preferably routes messages from the input sources 710 to the clusters 714. Similarly, a client-side global load balancer 718 preferably routes connection requests from the clients 712 to the clusters 714. The load balancers 716, 718 are designed to ensure that load is distributed among the clusters 714 according to a predetermined heuristic. For example, the load may be distributed evenly among the clusters 714 or a more powerful cluster may be distributed a majority of the load. In one embodiment, one load balancer performs the functions of the input-side 716

and client-side 718 load balancers and utilizes conventional Domain Name System-(DNS-) based load balancing.

[0091] Each cluster 714, of which cluster 714A is representative, preferably contains an input-side cluster load balancer 720A and a client-side cluster load balancer 722A. The cluster load balancers 720A, 722A function similarly to the corresponding global load balancers 716, 718 in that the input-side cluster load balancer 720A balances and routes incoming messages among one or more gateways 724A and the client-side cluster load balancer 722A balances and routes incoming connection requests among one or more nodes 726A and application servers 728A. The gateways 724A are connected to the nodes 726A. In one embodiment every gateway 724A is connected to every node 726A and in another embodiment certain gateways are connected to only certain nodes.

[0092] Preferably, the routing network 110 utilizes conventional single-processor computer systems executing the Linux operating system (OS). Preferably, each component of the routing network 110 is implemented by a separate, dedicated computer system in order to enable the separate optimization of the components. The input/output (I/O) functionality of the OS is preferably enhanced through the use of a non-blocking OS package such as NBIO available from the University of California, Berkeley, California. Based on the assumption that connections with the nodes 728 are long-lived, the OS is preferably configured to not allocate resources toward monitoring idle connections. Instead, the well-known /dev/poll patch is preferably applied to the OS in order to provide advanced socket polling capabilities.

[0093] Those skilled in the art will recognize that there are many ways to use the functionality of the routing network 110 to route update messages to clients 710. For example, in one embodiment, every message is distributed to every node 726. In another embodiment, the routing network 110 selectively sends messages to the nodes 726 in the routing network 110. Selectively sending messages to the nodes in the routing network presents at least the two difficulties. First, for a given message, a decision needs to be made as to which nodes should receive it. Second, a client must receive all messages in which it is interested. There are at least two approaches meeting these difficulties. The first approach uses a hierarchy of registries at the gateways and nodes to respectively keep track which messages to send to the nodes and clients. The second approach assigns messages to one or more categories, assigns nodes to one or more types, and maintains mappings between categories and types. This latter approach also uses client proxies to allow clients 712 to communicate with multiple nodes of different types.

[0094] FIG. 7 illustrates the embodiment using the hierarchy of registries. A node 726 registers with each gateway 724 in all of the clusters and indicates which messages it needs and the clients do the same with the nodes. Each gateway 724 preferably maintains a registry 734 containing the object IDs registered by the nodes 726 connected to the gateway 724. In one embodiment, the gateway registry 734 associates each object ID with a linked list containing one entry for each node 726 that has registered for that object ID. In another embodiment, the gateway registry 734 is a hash table containing the object ID registered by the nodes 726 connected to the gateway 724. A node 726 preferably maintains a node registry 732 containing the object IDs registered by clients 712 connected to the node. The gateways 724 in each cluster 714 receive the messages

from the input sources 710 and direct the messages to the appropriate node or nodes 726. The nodes 726 preferably transmit messages received from the gateways 724 to the clients 712 that have registered in the object IDs identified by the messages.

[0095] In one embodiment, the node registry 732 associates each object ID with a linked list containing one entry for each client 712 that has registered for that object ID. In another embodiment, the node registry 732 is a hash table containing the object ID registered by the clients 712 connected to the nodes 726. Each entry in the linked list or hash table preferably contains a pointer to a socket representing the connection to the corresponding client 712. As is known in the art, the pointer to the socket, typically called a "file descriptor," represents an address to which the node can write in order to send the message to the corresponding client. Gateways 724 can also use file descriptors in this manner to store node addresses. Alternative embodiments of the present invention utilize other data structures in addition to, or instead of, the hash table and linked list, and/or may utilize different data within the data structures.

[0096] Preferably, the node 726 adds an entry to its registry 732 every time a client 712 registers an interest in an object and deletes the corresponding entry from the registry when the client 712 disconnects from the node or otherwise indicates that it is no longer interested in a particular object. If the node 726 determines that the client 712 registered for an object ID that was not previously registered on that node, the node preferably registers that object ID with the gateways 724 to which it is connected. Similarly, if the node 726 determines that the client 712 deregistered an object ID for which it was the last interested client, the node 726 deregisters that object ID with the gateways 724 to which

it is connected. The gateways 724 update their registries 734 in response to the communication from the node 726.

[0097] In alternative embodiments of the present invention, when the client 712 disconnects from the node 726 or otherwise indicates that it is no longer interested in a particular object, the node 726 waits for a period of time or until some event occurs before deregistering the object ID with the gateway 724. For example, the node 726 could wait until it receives a message associated with that object ID to do the deregistration. Alternatively, the wait time can be fixed, random, or based on the frequency of registrations for that object ID, or registrations for the same input source 710 as the object being deregistered. This latter approach keeps the object IDs from more frequently used input sources 710 registered for a longer period of time. This waiting advantageously reduces the number of registration changes required between nodes and gateways in cases where it is likely that another client will soon register for the same object ID.

[0098] Since a gateway 724 does not control the rate at which it receives messages from input sources 710, it is possible for the gateway to receive messages faster than it can process them (i.e., send the messages to the nodes). Therefore, each gateway 724 preferably maintains a queue 730 of messages that have been received but not yet processed in order to avoid losing messages. In one embodiment, the gateway 724 drops messages if the queue 730 becomes too long. In another embodiment, the gateway 724 utilizes priorities assigned to certain messages or input sources to determine which messages to drop.

[0099] The application server 728 within each node 714 preferably serves the activation module 124 to the clients 712 in response to client requests. In addition, the application server 728 serves any other modules that may be required or desired to support the environment 100. In an alternative embodiment of the routing network, a single application server 728 fulfills all of the client requests. This application server 728 may be within a certain cluster 714 or independent of the clusters. However, this single-application-server embodiment is less desirable because it lacks redundancy.

[0100] FIG. 8 is a flow chart illustrating steps performed by a gateway 724 and a node 726 in a cluster 714 to perform object-based routing of a message received from an input source 710 in the embodiment using a hierarchy of registries. Initially, the gateway 724 receives 810 the message from the input source 710. The gateway 724 extracts 820 the object ID from the message. The gateway 724 examines its registry 734 to determine the nodes 726 that have registered in the object ID. The gateway 724 transmits 840 the message to each of the registered nodes 726. Each node 726 that receives the message uses its registry 732 to determine 850 which clients 712 have registered for the message. Each node 726 then forwards 860 the message to the registered clients 712.

[0101] Adding node functionality to the gateway and client functionality to the node advantageously allows the routing network 110 to solve the difficulties identified above. For example, it allows the routing network 110 to decide which nodes should receive an updated message. As a result, messages are sent to only nodes that have registered for the message. Further, no matter which node a client connects to, that node will receive all messages that the client wants. One skilled in the art would understand that while the

present invention allows registration at both a gateway level and node level, registration could be extended to any number of levels.

[0102] In the second approach for meeting the difficulties described above, all messages in the routing network 110 are assigned to one or more of N categories, and all of the nodes are assigned to one or more of M types. Mappings are created that specify which categories of messages are forwarded to which types of nodes. The mappings allow control over the amount of traffic processed by the nodes.

[0103] FIG. 9 is a block diagram illustrating a high-level view of the routing network 110 of FIG. 7 in an embodiment adapted to use the approach having message categories and node types. Although FIG. 9 illustrates only two node types—nodes of type 1 726 and nodes of type 2 726—for purposes of simplicity, this embodiment of the network 110 can have any number of node types. FIG. 9 also illustrates a client proxy 740 for reasons described below. However, an embodiment of the present invention using message categories and node types does not necessarily utilize a client proxy 740.

[0104] There are many different possible mappings between message categories and node types. In the simple case, there is one-to-one mapping between message categories and node types. For example, if the message is of category 1, it is forwarded to the nodes of type 1. In more complicated mappings, messages of one category are mapped to nodes of multiple types. For example, messages of category 1 are mapped to nodes of types 1, 2, and 3, whereas messages of category 2 are mapped to nodes of types 2, 3 and 4. In short, any possible mapping of message categories to node types is possible and the

number of message categories, N, does not have to be the same as the number of node types, M.

[0105] There are multiple ways to assign the messages into categories. One way is to assign all messages from a given input source 710 into a certain category. Another way is to explicitly specify the category in the object ID for the message. Yet another way is to utilize a hashing function or lookup table to partition messages into categories based on object IDs or other values. For example, in one embodiment a hash function is applied to the object ID to generate an integer between 1 and N, and this integer is the message category.

[0106] Nodes are preferably assigned to types based on information stored in the gateways 724. In one embodiment, each gateway 724 holds a lookup table or other data structure that specifies the types to which each node is assigned. The lookup table also preferably stores the mappings between message categories and node types. When a gateway 724 receives a message from an input source, the gateway preferably determines the category of the message using one of the techniques described above. Then, the gateway 724 determines the node type (or types) to which the message category maps, and determines which nodes are of the given type. The gateway 724 routes the message to the appropriate nodes.

[0107] In one embodiment, the gateway 724 uses a combination of multiple techniques to determine the message categories, node types, and/or mappings. For example, a lookup table can be used to encode a priori knowledge about categories, types, and/or mappings and a hash table can be used to route messages for which there is

no a priori knowledge. Continuing this example, assume that certain messages are assigned to a given category based on a table lookup, while other messages are assigned to categories based on a hashing function. In this example, the gateway 724 looks up the object ID (or other information, such as an input source ID) of an arriving message in a lookup table to determine if it has a specified category. If the object ID is stored in the lookup table, the gateway 724 determines the mappings for the category and routes the messages to the nodes of the appropriate types. If the object ID is not stored in the lookup table, the gateway 724 utilizes a hash function on the object ID (or other information) to determine the message category.

[0108] Message categorization advantageously allows the routing network 110 to decide which nodes get which messages. However, message categorization does not ensure that clients 712 have access to the messages they need. Assume a client 712 connects to a node of type 1 and the client 712 wants to receive messages of category 3. If there is a simple one-to-one mapping of message categories to node types, nodes of type 1 will never receive messages of category 3, and neither will the client 712. To ensure that clients have access to the messages they need, the embodiment shown in FIG. 9 uses client proxies to allow clients 712 to communicate with multiple nodes of different types. As shown in FIG. 9, the clients 712 connect to a client proxy 740 instead of the nodes 726. Each client proxy 740 is connected to at least one node of each type. For example, as shown in FIG. 9, the client proxy 740 connects to a node of type 1 726 and a node of type 2 726.

[0109] There are at least two ways to implement the embodiment that uses client proxies to ensure that clients connect to multiple nodes of different types. The

implementations vary primarily in where the client registration information is stored. In the first variation, client registration information is stored at the nodes, and the client proxy is adapted to simply pass update messages and registration information among the clients and nodes. In the second variation, the client proxy 740 is responsible for keeping track of client registrations.

[0110] FIG. 9 illustrates the variation where the nodes store the client registration information, and the client proxy 740 is adapted to pass messages and registration information among the clients and nodes. When a client sends registration information to the client proxy 740, the client proxy passes it to the appropriate node or nodes. In one embodiment, the client proxy 740 sends client registration information to at least one node of every type, and the nodes ignore irrelevant registration information (e.g., registrations for messages of categories not handled by the node). In another embodiment, the client proxy 740 itself analyzes the registrations, and passes only registrations relevant for a given node type to a node of that type. In a third embodiment, the client contains functionality for determining which node types handle which registration requests, and the client tells the proxy 740 to which node types to pass the registration information.

[0111] The client proxy 740 preferably stores an identifier of its connection to each client 712, such as a pointer to the socket for the client connection, and sends this identifier to the nodes along with the registration information. Each node 726, in turn, maintains a registry 732 storing the relevant registration information it receives from the client proxy 740. Preferably, at most one node of each type will contain registration information for a given client for messages of a category mapped to the node type. In

addition, a node preferably does not store registration information for messages of categories not mapped to its node type.

[0112] The registry 732 for each node 726 preferably indicates the object IDs registered by the clients, the client proxy 740 to which the client is connected (e.g., a pointer to the socket at the node to which the client proxy is connected), and the identifier indicating the client's connection to the client proxy (e.g., the pointer to the socket at the client proxy to which the client is connected). When the node 726 receives an update message, it uses the registry to identify the clients to which it should forward the message, the client proxies to which the clients are connected, and the specific connections between the client proxies and the clients. The node routes the update message to the identified client proxies and includes the identifier (e.g., the pointer to the client socket) telling the client proxy where to send the update message. When the client proxy 740 receives the update message, it uses the identifier to send the message to the client or clients.

[0113] In one embodiment, the node registry 732 contains, for each object ID, a list with one entry for each client registered for that ID. Each entry contains a (name of client proxy, name of client) pair where each "name" is a socket identifier or some other information for identifying how to route the message to the named entity. In another embodiment, the registry 732 is adapted to more efficiently handle the case where multiple clients at a given client proxy are registered for the same object ID by placing a list of clients registered for the object ID in each entry. Thus, each entry in the list for a given object ID contains a (name of client proxy, list of names of clients on that proxy) pair. When a message having a particular object ID arrives at the node, the node walks

down the corresponding entries in the registry for that ID, pushing one copy of the message to each listed client proxy. Along with the message, the node includes the list of clients on that proxy (e.g., a list of pointers to sockets for the clients) who have registered for the object ID. When the client proxy receives the message and the list, it simply pushes a copy of the message to each client.

[0114] The approach using the pass-through client proxy 740 described above is advantageous because it reduces memory loads because the client proxy stores only minimal state, and each node has a restricted list of object IDs for which it is responsible. It also expedites message routing because the client proxy does very little processing to forward a message. Plus, this approach separates different potential stress points in the network into different components, allowing the network to be tuned to provide good performance. In particular, the nodes handle a large amount of data and store a large amount of state, but do not have to hold open a large number of connections. The client proxies do not store a large amount of data or state, but hold open a large number of connections to the clients. In different embodiments, the ratios of nodes to client proxies can be varied to match the network to its requirements.

[0115] In the second variation of client proxies, the client proxy 740 stores client registration information. FIG. 11 illustrates a high-level diagram of the routing network 110 in which the client proxy 740 stores the client registration information. In this embodiment, the node registry 732 stores the object IDs registered by client proxies 740 connected to the node. Each client proxy 740 preferably maintains a client proxy registry 736 containing the object IDs registered by clients 712 connected to the client proxy 740. The client proxy 740 registers with the nodes for all objects for which it needs to receive

updates. Thus, this variation is similar to the hierarchical registry embodiment described above.

[0116] An alternative to maintaining a separate client proxy as described above is to have the nodes themselves provide the client proxy functionality. In this alternative, clients connect to a node of a first type. When a client registers for messages of a category not handled by the node, the node passes the registration to a node of the appropriate type along with an identifier of the client connection. In this case, the node connected to the client adopts the functionality of the pass-through client proxy. The node can also adopt the functionality of the client proxy that stores client registration information, thereby creating a hierarchy of registrations among the nodes themselves.

[0117] FIG. 12 is a high-level block diagram of the routing network 110 of the embodiment in which the nodes themselves adopt client proxy functionality. There are at least two ways to implement this embodiment. In one implementation there is a preferred node type, such as node of type 3, that always serves as a client proxy. This node type may be designated to handle the most common categories of messages. As a result, the client proxy functionality would be utilized very rarely. As shown in FIG. 12, client 1 712 and client 2 712 are connected to the node of type 3 726, which in turn is connected to a node of type 1 726 and a node of type 2 726.

[0118] In another implementation, instead of having one preferred node type that always serves as a client proxy, each node type can serve as a node and as a client proxy. For example, when a client 712 connects to a node 726, it passes the node all of the object IDs the client wishes to register. For object IDs of message categories handled by

that node, the node stores the registrations in its registry. For object IDs of messages in other categories, the node acts as a client proxy and passes the registration requests to nodes of the appropriate types. This implementation can be made more efficient by providing functionality in the activation module for identifying the most common categories of messages sought by the client, and causing the client to connect to a node of a type that receives those categories of messages.

[0119] An alternative to connecting clients 712 to nodes 726 using a client proxy 740 is having a client 712 maintain multiple connections to the routing network 110. This can be accomplished, for example, by attaching message-categorization functionality to the activation module 124. The activation module 124 preferably determines which categories of messages it needs and to which nodes types it should connect. The activation module 124 then makes a connection to one node of each relevant type and registers the appropriate object IDs with the appropriate nodes.

[0120] An alternative to using client proxies is to allow clients to register for only categories of messages handled by one type of node. For example, if each node type handles messages from only one input source, each client 712 can be required to register for messages from only one input source. This can be done, for example, by configuring the load balancer 718 to ensure that clients registering for messages from one input source connect to a node of the type that receives messages from that input source.

[0121] FIG. 10 is a flow chart illustrating steps performed by a gateway 724, a node 726 that stores client registration information, and a pass-through client proxy 740 to perform object-based routing of a message received from an input source 710. Initially,

the gateway 724 receives 1010 the message from an input source 710. The gateway 724 determines 1020 to which category the message belongs using one or more of the techniques described above. After the gateway 724 has determined to which category the message belongs, the gateway 724 identifies 1030 node types to which the message category is mapped and routes 1040 the message to the nodes of those types. Each node 726 that receives the message extracts 1050 the object ID from the message. Each node 726 examines its registry 732 to determine which clients have registered for the message, the client proxies to which the clients are connected, and the connections between the client proxies and the clients. Each node 726 then forwards 1060 the message to the identified client proxies 740 along with the identifiers for the client connections (e.g., the client socket or a list of client sockets). When a client proxy 740 receives the message from a node 726, it pushes 1070 the message to the identified client socket or sockets.

[0122] The above description is included to illustrate the operation of the preferred embodiments and is not meant to limit the scope of the invention. The scope of the invention is to be limited only by the following claims. From the above discussion, many variations will be apparent to one skilled in the relevant art that would yet be encompassed by the spirit and scope of the invention.

CLAIMS

1. A method for dynamically updating a property of a live object at remote clients, comprising:

receiving a first message from a remote input source, the first message identifying the live object and containing data for updating a property of the live object;

identifying nodes that have registered for updates to the live object, wherein the nodes are distributed on a network;

routing the first message via the network to the registered nodes;

identifying remote clients that have registered for updates to the live object, wherein the remote clients are distributed on the network; and

routing a second message via the network from the nodes to the registered clients, the second message identifying the live object and containing the data for updating the property of the live object;

wherein the registered clients are adapted to process the data to cause the update to the property of the live object.

2. The method of claim 1, wherein the live object is identified by an object ID.

3. The method of claim 1, further comprising:

receiving a registration request from a node, the registration request identifying the live object with an object ID.

4. The method of claim 1, further comprising:

receiving a registration request from a client, the registration request identifying the live object with an object ID.

5. The method of claim 1, further comprising:

receiving a request from a client for an activation module, the activation module adapted to identify any live objects at the client; and

providing the activation module to the client.

6. The method of claim 1, further comprising:

receiving, by a node, a registration request registering for updates to properties of live objects at the client.

7. The method of claim 1, further comprising:

processing the first message from a first format to a second format to produce the second message.

8. The method of claim 1, wherein the first message and the second message are identical.

9. A dynamic content routing network for enabling updating of a property of a live object at a client coupled to the network, comprising:

a gateway for receiving a message from a remote input source, the message identifying the live object and containing data for updating a property of the live object, for maintaining a registry of nodes coupled to the gateway that have registered to receive updates to properties of the live object, and for routing the message to the registered nodes; and

a node in communication with the gateway, for receiving the message from the gateway, for maintaining a registry of remote clients coupled to the network that have registered to receive updates to properties of the live object, and for routing the message to the registered clients.

10. The routing network of claim 9, wherein the node is further adapted to receive a registration request from a client, the registration request identifying the live object with an object ID, to update the node's registry in response to the registration

request, and to send a corresponding registration request to the gateway if the node is not registered for the object ID.

11. The routing network of claim 9, wherein the node is further adapted to receive a deregistration request from a client, the deregistration request identifying the live object with an object ID, and to send a corresponding deregistration request to the gateway if no other clients connected to the node are registered for the object ID.

12. The routing network of claim 9, wherein the registry maintained by the node comprises:

a data structure identifying live objects for which clients have registered, and an address of each registered client.

13. The routing network of claim 9, wherein the registry maintained by the gateway comprises:

a data structure identifying live objects for which nodes have registered, and an address of each registered node.

14. A dynamic content routing network for enabling updating properties of live objects at a client coupled to the network, comprising:

a gateway for receiving messages from a remote input source, the messages identifying the live objects and containing data for updating properties of the live objects, for identifying categories of the messages, for determining node types to which the message categories map, and for routing the messages responsive to the mappings; and

a plurality of nodes, each node assigned to at least one node type, the nodes in communication with the gateway and adapted to receive messages of categories mapped to the nodes' types, the nodes for maintaining a registry of clients coupled to the network that have registered to receive updates to properties of the live objects, and for routing the messages to the registered clients.

15. The routing network of claim 14, wherein a node includes client identifiers with the messages routed to the clients, the network further comprising:

a client proxy in communication with a node of each type, for receiving the messages routed by the nodes and the client identifiers and for using the client identifiers to route the messages to the registered clients.

16. The routing network of claim 14, wherein the registry maintained by a node comprises:

a data structure identifying clients that have registered for live objects, client proxy identifiers identifying the client proxies to which the clients are connected, and client identifiers identifying connections between the client proxies and clients.

17. The routing network of claim 14, wherein the registry maintained by the node comprises:

a data structure identifying clients that have registered for live objects, client proxy identifiers identifying the client proxies to which the clients are connected, and a list of client identifiers identifying connections between the clients and client proxies.

18. The routing network of claim 15, wherein the client proxy is further adapted to receive a registration request for update messages for a live object from a client and to route the registration request to a node.

19. The routing network of claim 18, wherein the client proxy routes the registration request to at least one node of each type.

20. The routing network of claim 18, wherein the client proxy routes the registration request to a node of a type to which the update message category is mapped.

21. A dynamic content routing network for enabling updating properties of live objects at a client coupled to the network, comprising:

a gateway for receiving messages from a remote input source, the messages identifying the live objects and containing data for updating properties of the live objects, for identifying categories of the messages, for determining node types to which the message categories map, and for routing the messages responsive to the mappings; and

a plurality of nodes, each node assigned to at least one node type, the nodes in communication with the gateway and adapted to receive messages of categories mapped to the nodes' types, the nodes for determining a client proxy that is registered to receive updates to properties of the live object, and for routing the message to the registered client proxy.

22. The routing network of claim 21, wherein the registry maintained by the client proxy comprises:

a data structure identifying live objects for which clients have registered, and an address of each registered client.

23. A method for dynamically updating properties of live objects at remote clients, the method comprising the steps:

receiving a message from a remote input source, the message identifying a live object and containing data for updating a property of the live object;

identifying a category of the message;

identifying a mapping to a node type for the identified category of the message; and

routing the message via the network to a node of the identified node type;

wherein the node is adapted to identify clients that have registered for updates to properties of the live object and to route the message to those clients, and wherein the clients are adapted to process the message to cause the update to the property of the live object.

24. The method of claim 23, wherein the step of identifying a category of the message comprises the step of:

looking up the message in a lookup table.

25. The method of claim 23, wherein the step of identifying a category of the message comprises the step of:

applying a hash function to an ID of the live object to generate the category.

IPR2025-00200 / 722 Patent Page 56 of 1171 EX1002

ASYNCHRONOUS MESSAGING USING A NODE SPECIALIZATION ARCHITECTURE IN THE DYNAMIC ROUTING NETWORK

ABSTRACT OF THE DISCLOSURE

[0123] A network routes update messages containing updates to properties of live objects from input sources to clients having the objects. When the clients receive live objects, the clients identify the object IDs associated with the objects and register the object IDs with the routing network. The routing network is adapted to selectively send update messages to nodes in the network and the nodes forward the messages to the clients. One implementation uses a hierarchy of registries to indicate which nodes and clients receive which update messages. Another implementation assigns update messages to one or more of N categories and nodes to one or more of M types, and the gateways maintain mapping between categories and types. To ensure that clients receive all of the update messages for which they register, the clients connect to client proxies that in turn connect to at least one node of each type.

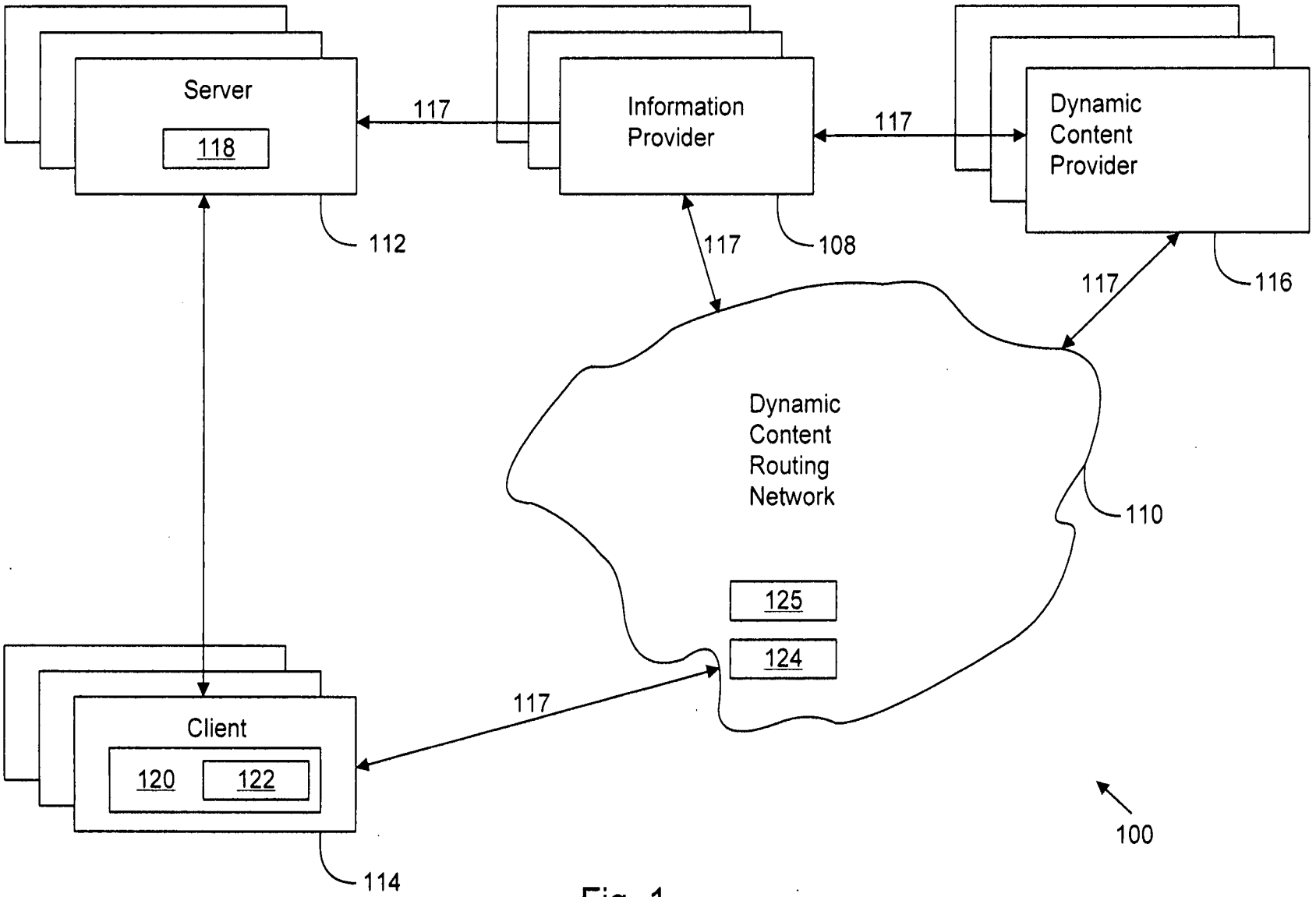

Server
118
112
117
Information Provider
108
117
Dynamic Content Provider
116
117
117
Dynamic Content Routing Network
110
125
124
117
Client
120
122
114
100

Fig. 1

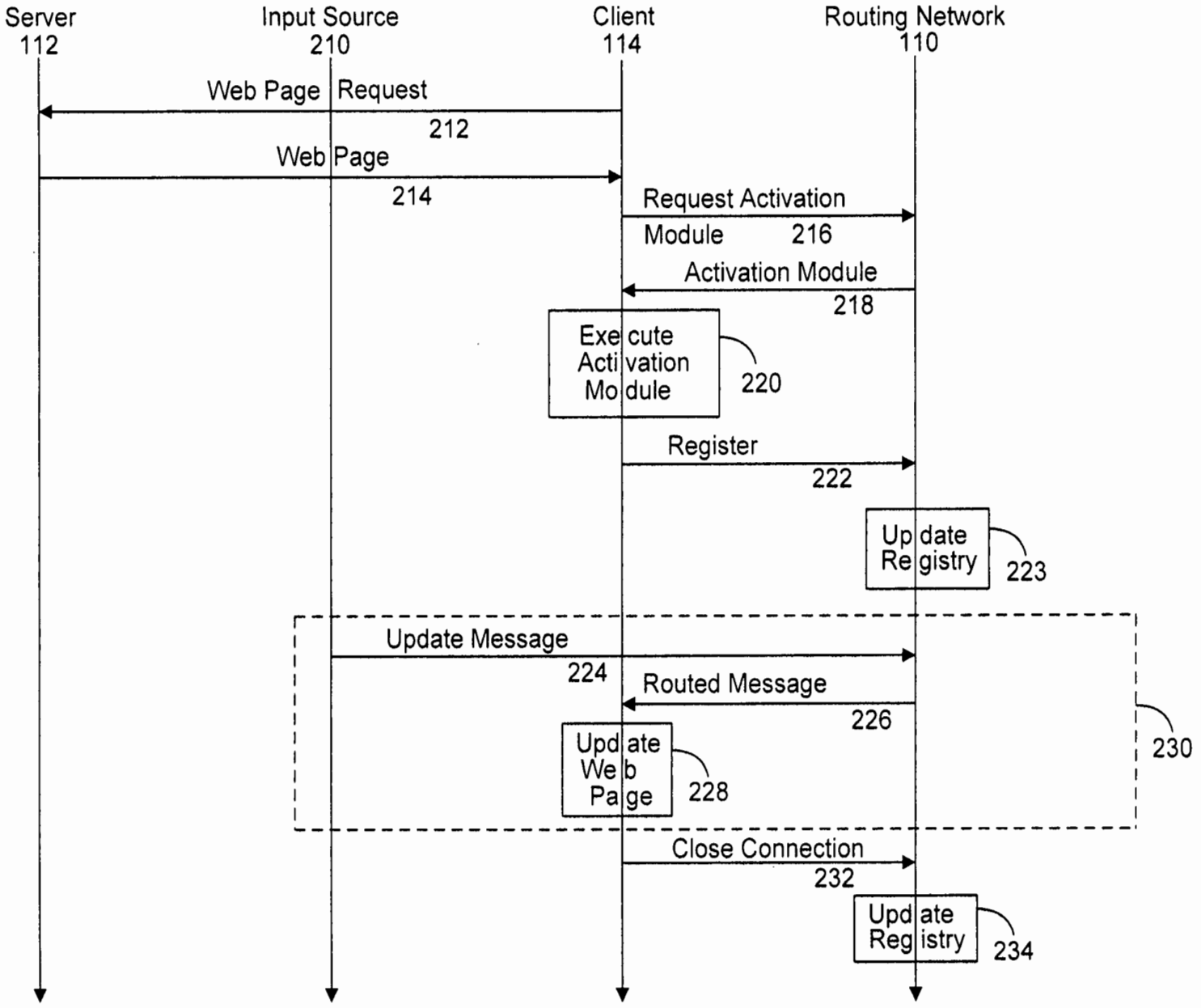

Server
112
Input Source
210
Client
114
Routing Network
110
Web Page Request
212
Web Page
214
Request Activation Module 216
Activation Module
218
Execute Activation Module
220
Register
222
Update Registry
223
Update Message
224
Routed Message
226
Update Web Page
228
230
Close Connection
232
Update Registry
234

Fig. 2

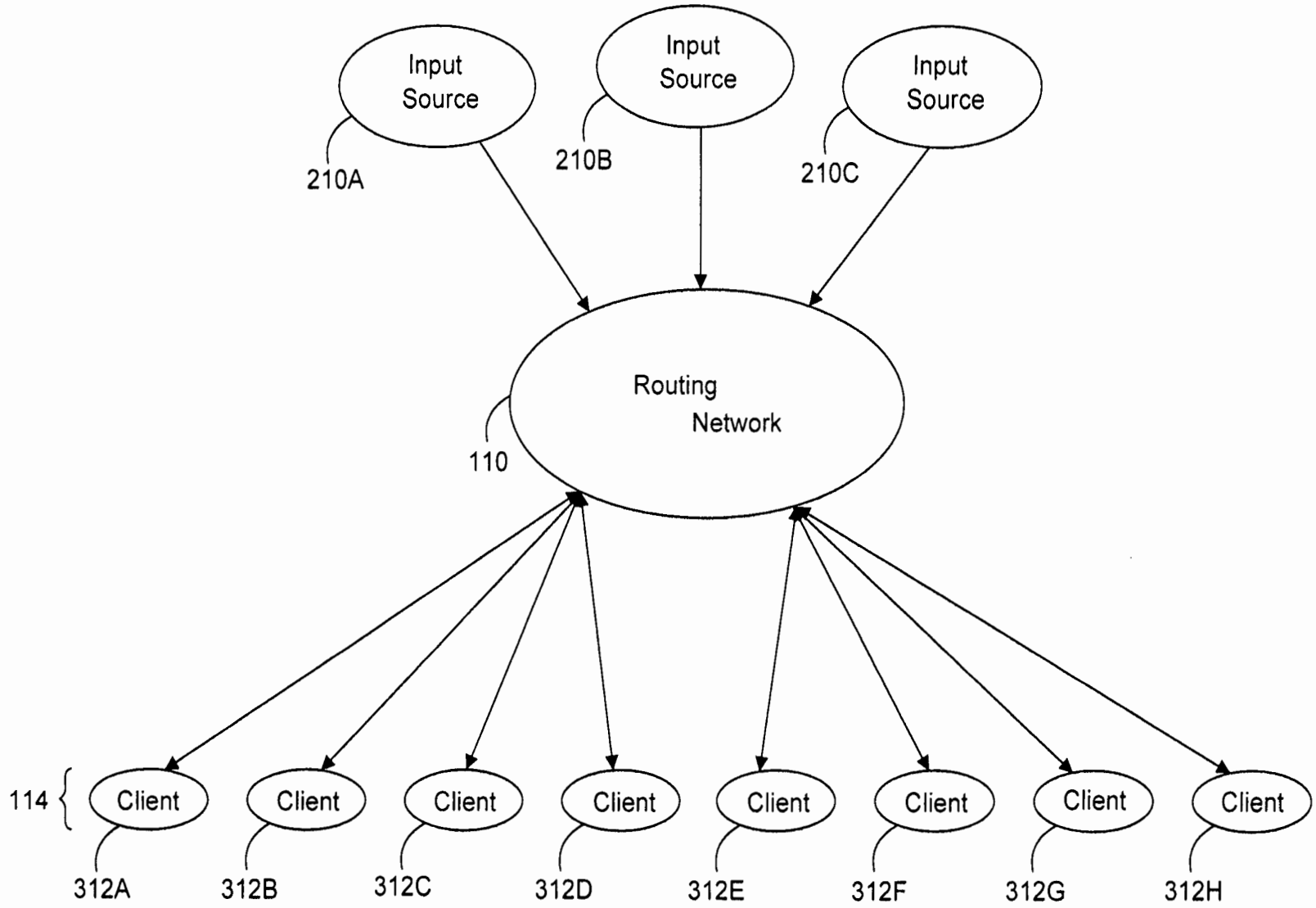
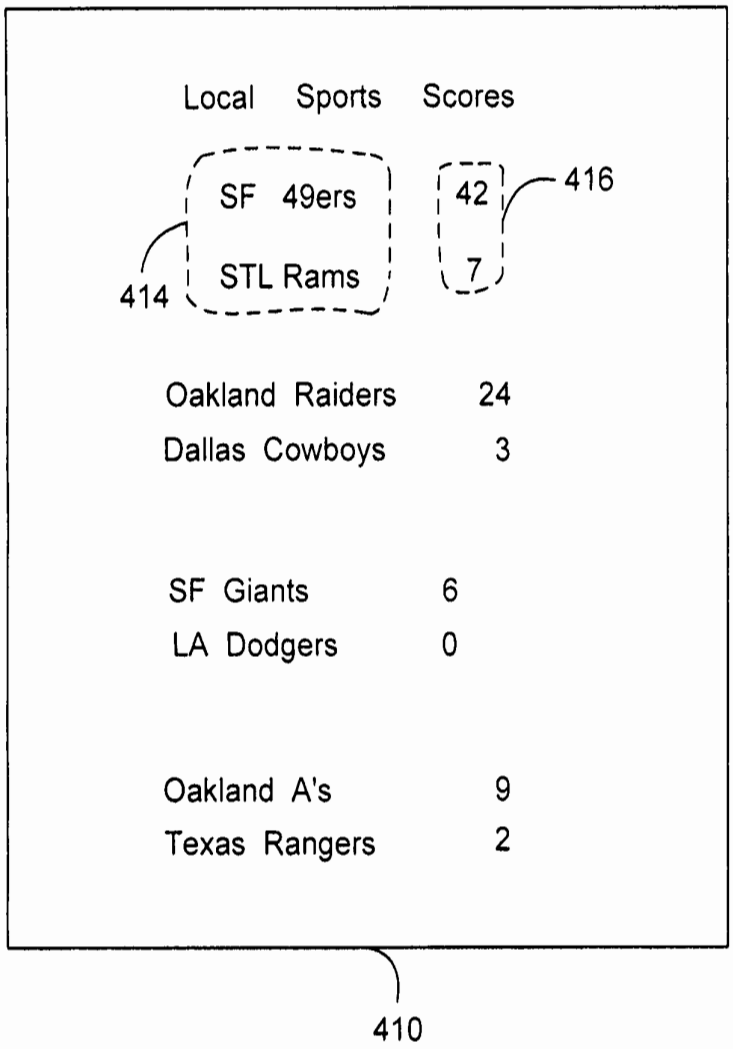

Input Source
Input Source
Input Source
210A
210B
210C
Routing Network
110
114
Client
Client
Client
Client
Client
Client
Client
Client
312A
312B
312C
312D
312E
312F
312G
312H

Fig. 3


Local Sports Scores
SF 49ers
STL Rams
42
7
414
416
Oakland Raiders 24
Dallas Cowboys 3
SF Giants 6
LA Dodgers 0
Oakland A's 9
Texas Rangers 2
410

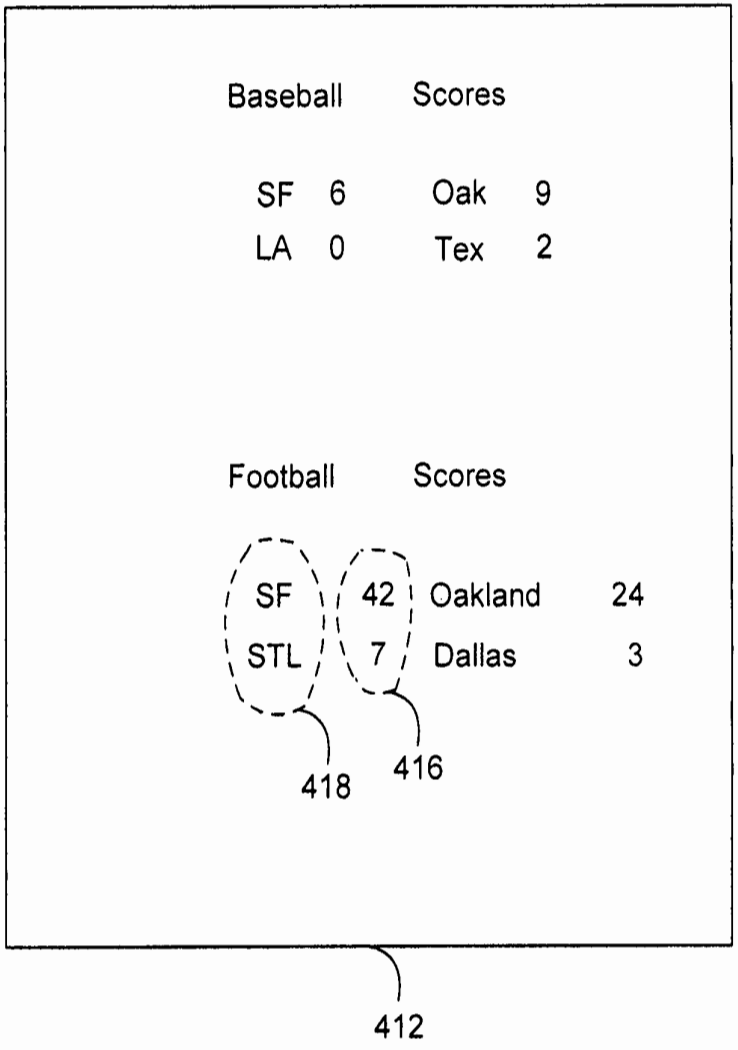

Baseball Scores
SF 6 Oak 9
LA 0 Tex 2
Football Scores
SF 42 Oakland 24
STL 7 Dallas 3
418
416
412

Fig. 4

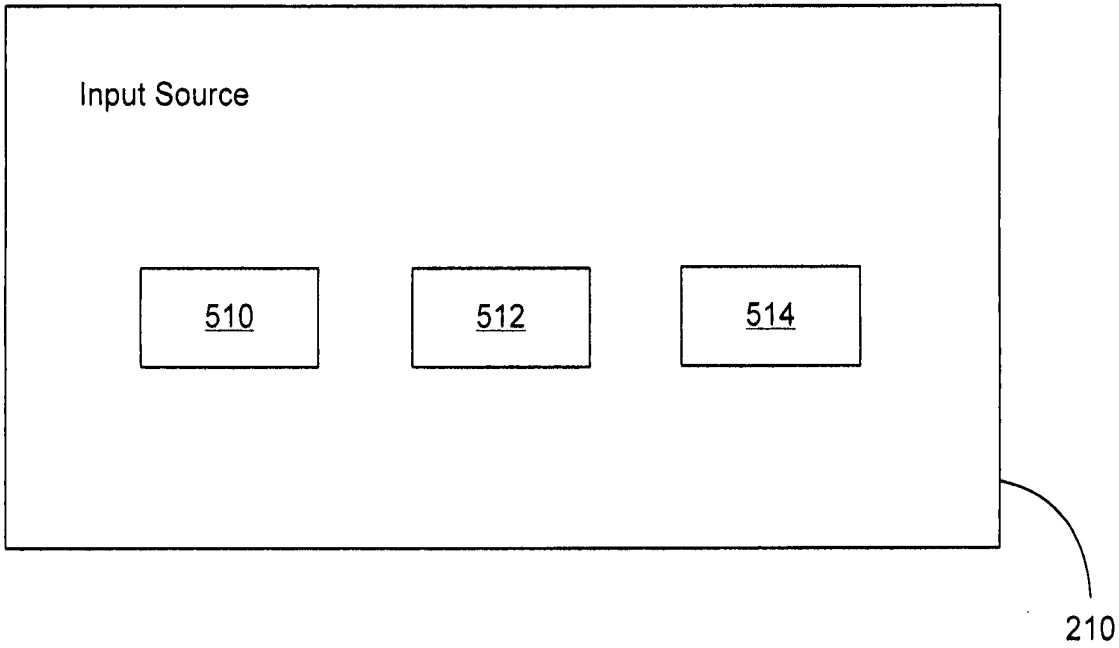


Fig. 5

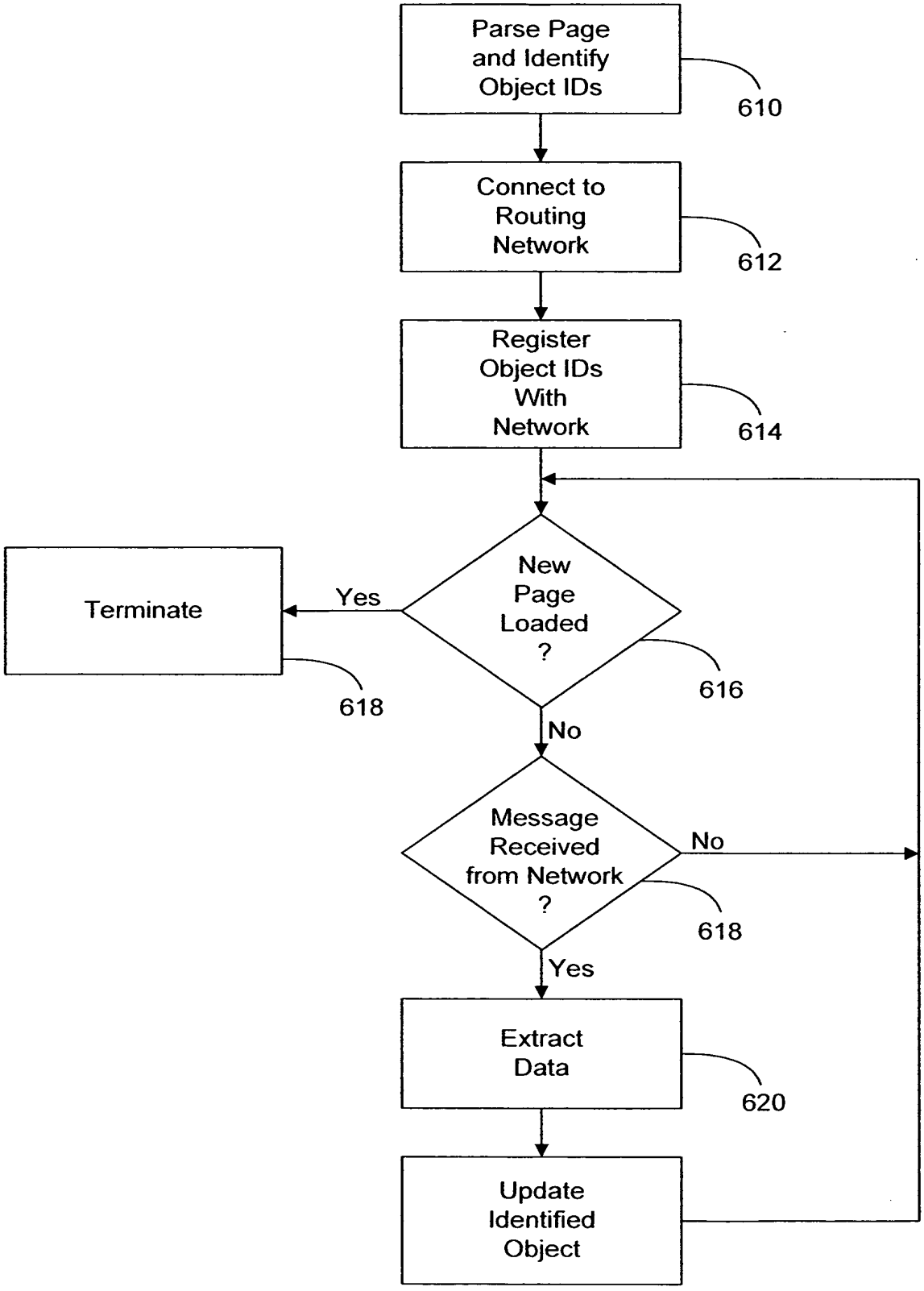

Parse Page and Identify Object IDs
610
Connect to Routing Network
612
Register Object IDs With Network
614
New Page Loaded ?
616
Yes
Terminate
618
No
Message Received from Network ?
618
No
Yes
Extract Data
620
Update Identified Object

Fig. 6

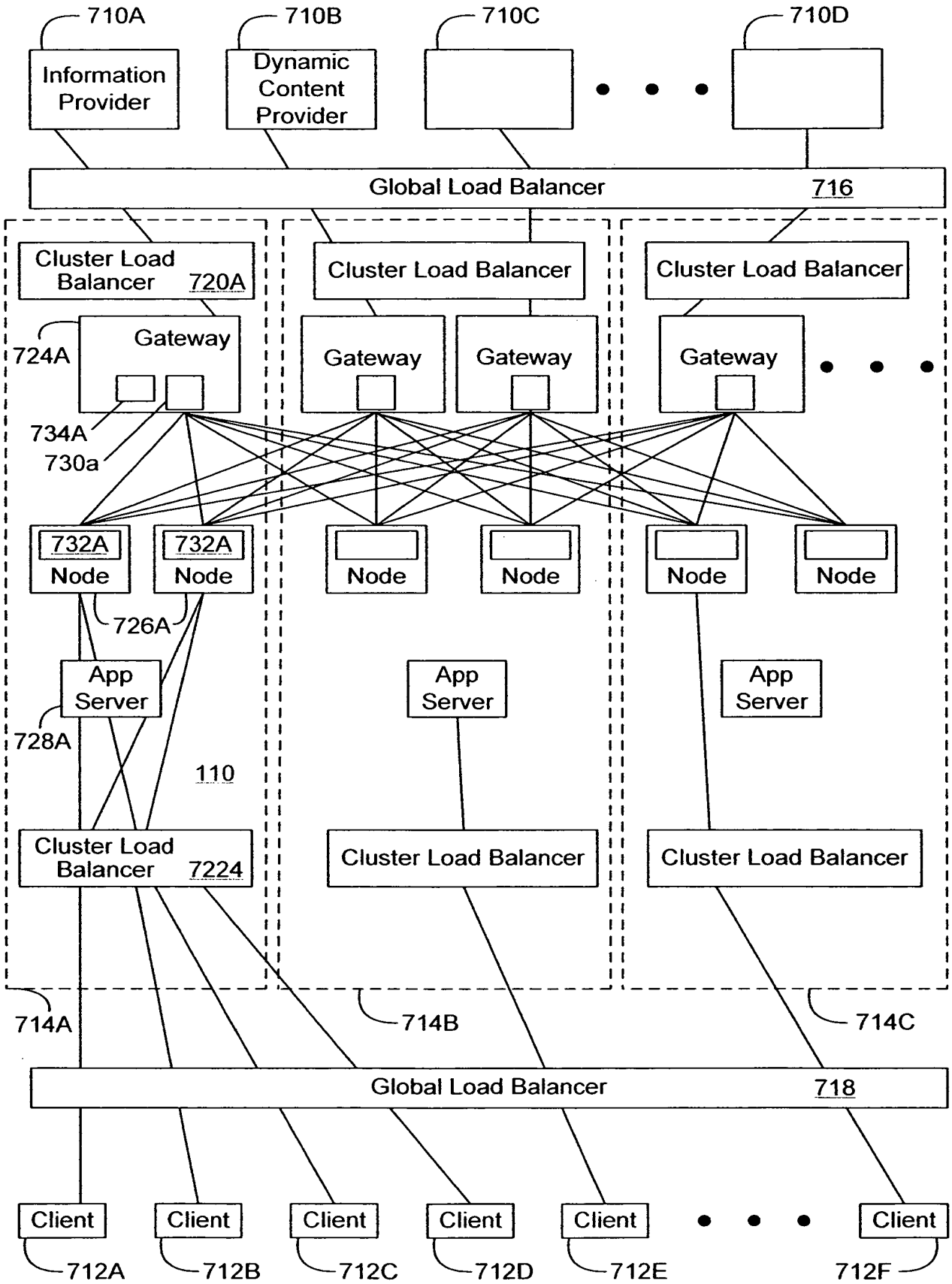

710A
710B
710C
710D
Information Provider
Dynamic Content Provider
Global Load Balancer
716
Cluster Load Balancer
720A
Cluster Load Balancer
Cluster Load Balancer
724A
Gateway
Gateway
Gateway
Gateway
734A
730a
732A
Node
732A
Node
Node
Node
Node
Node
726A
App Server
App Server
App Server
728A
110
Cluster Load Balancer
7224
Cluster Load Balancer
Cluster Load Balancer
714A
714B
714C
Global Load Balancer
718
Client
Client
Client
Client
Client
Client
712A
712B
712C
712D
712E
712F

Fig. 7

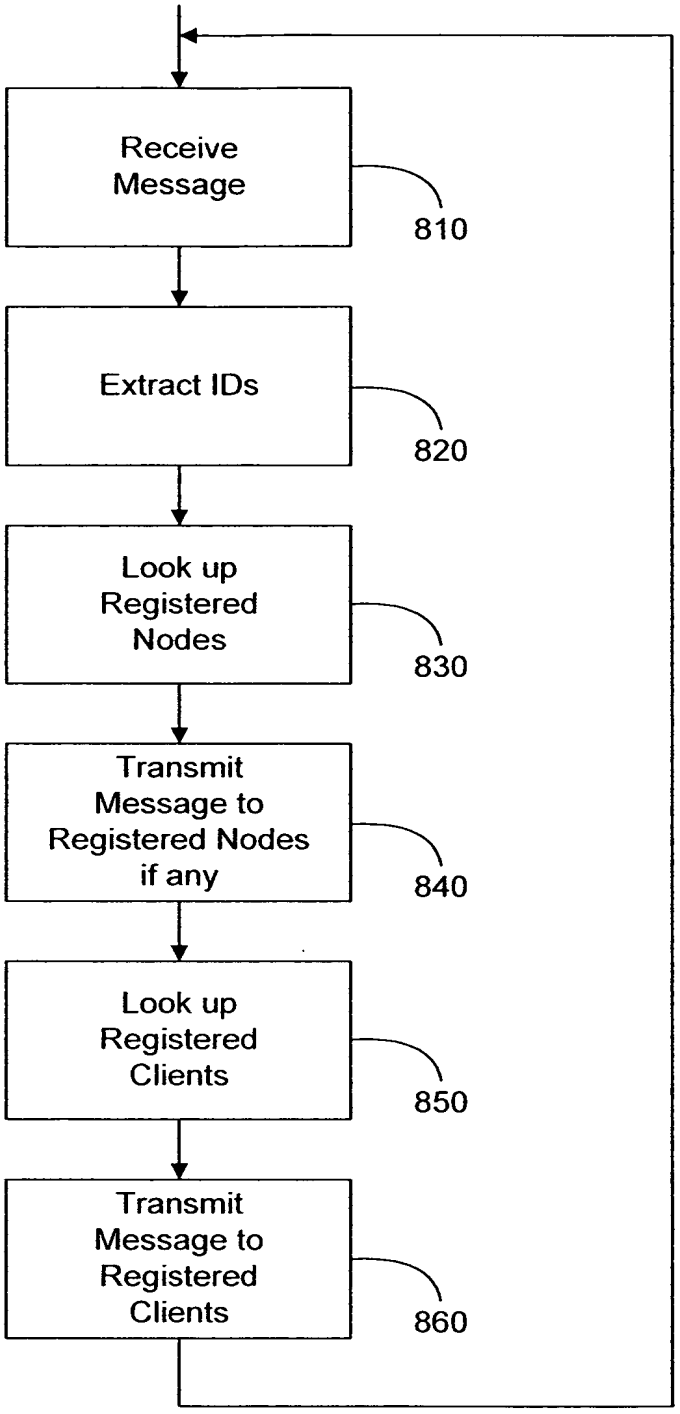

Receive Message
810
Extract IDs
820
Look up Registered Nodes
830
Transmit Message to Registered Nodes if any
840
Look up Registered Clients
850
Transmit Message to Registered Clients
860

Fig. 8

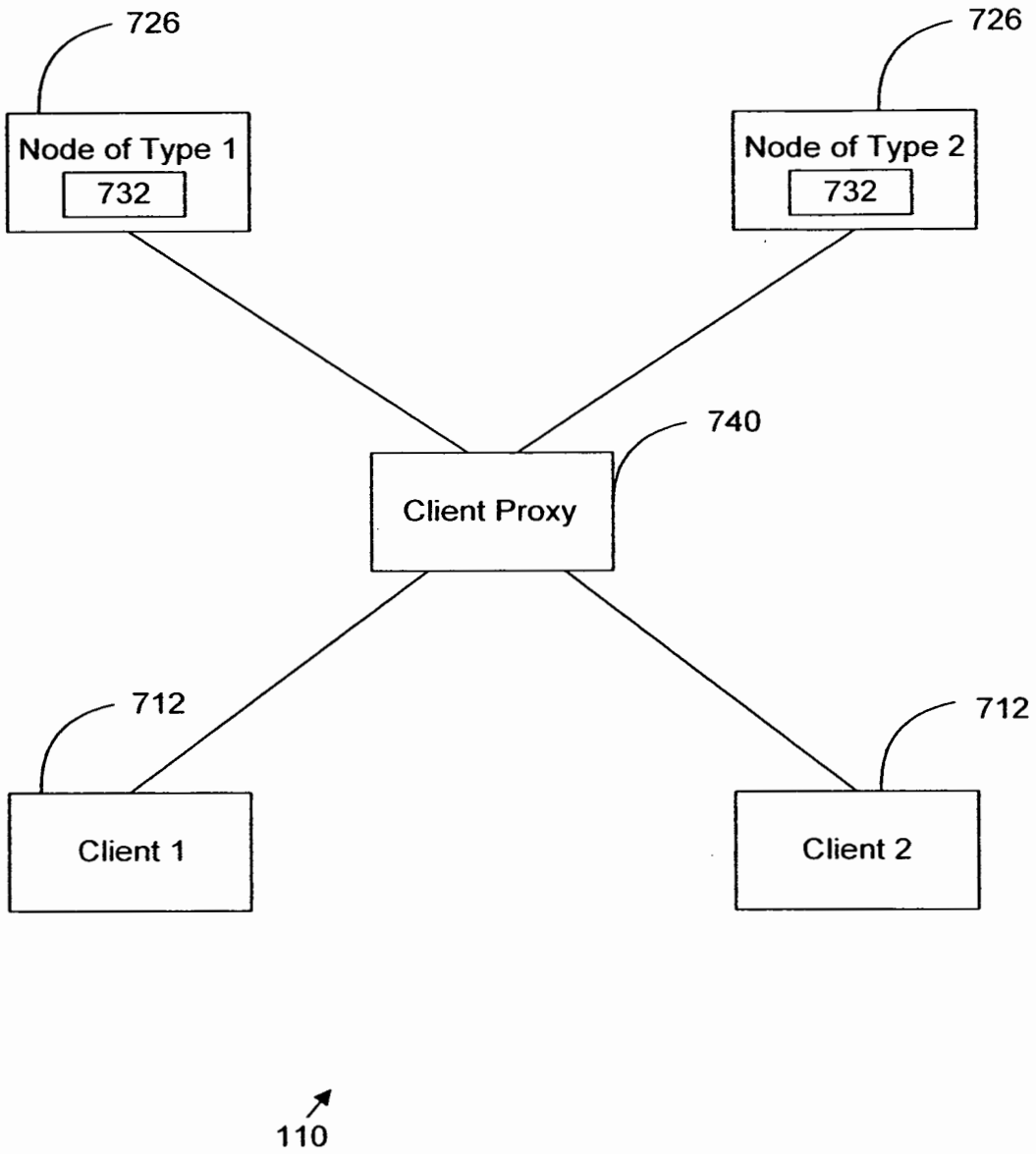

726
726
Node of Type 1
732
Node of Type 2
732
740
Client Proxy
712
712
Client 1
Client 2
110

Fig. 9

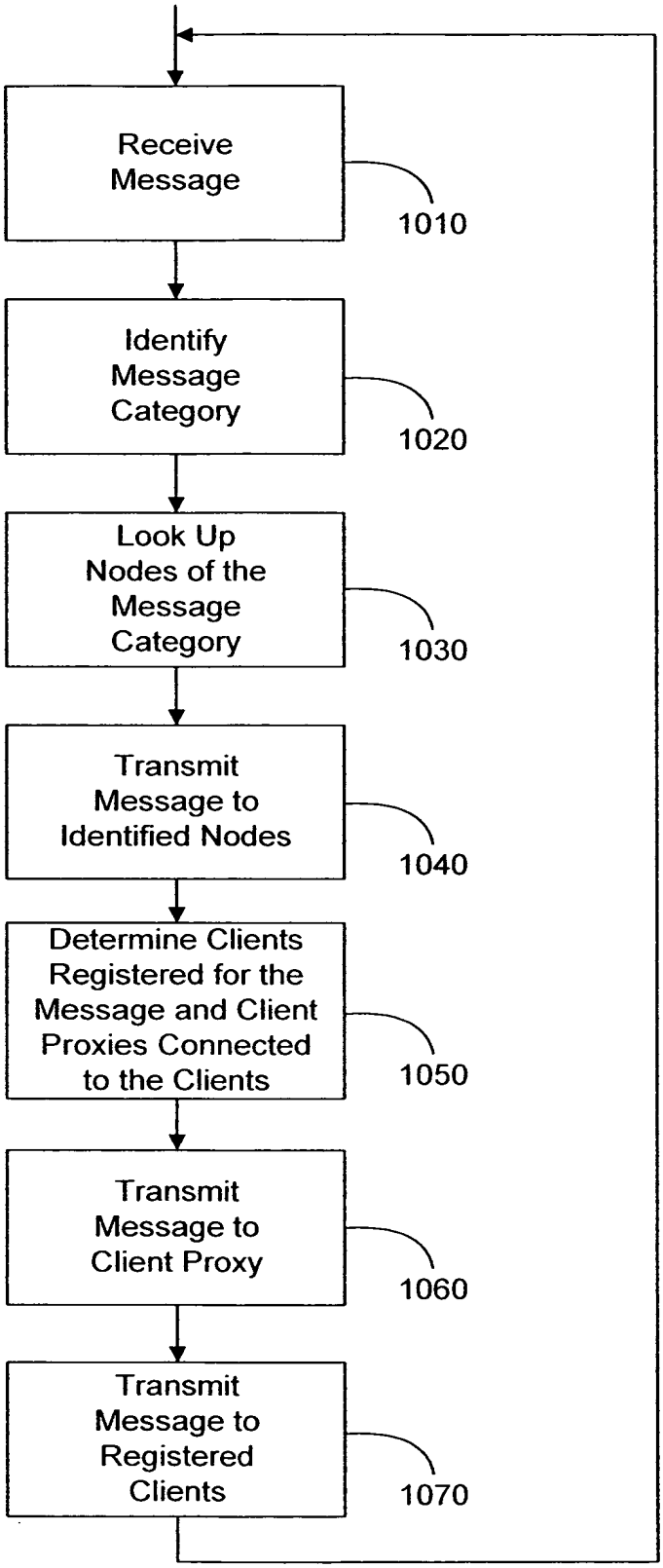

Receive Message
1010
Identify Message Category
1020
Look Up Nodes of the Message Category
1030
Transmit Message to Identified Nodes
1040
Determine Clients Registered for the Message and Client Proxies Connected to the Clients
1050
Transmit Message to Client Proxy
1060
Transmit Message to Registered Clients
1070

Fig. 10

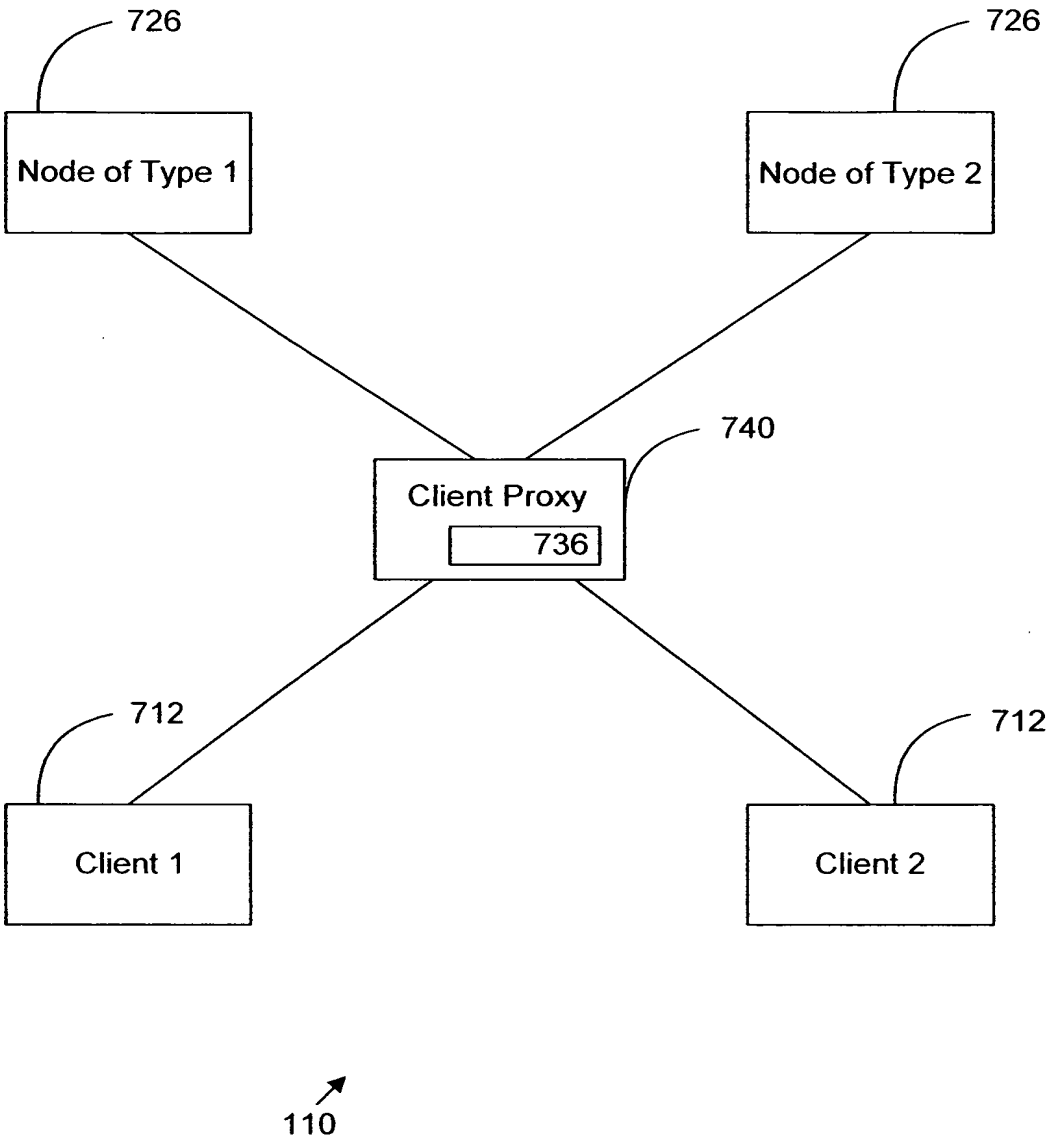

726
726
Node of Type 1
Node of Type 2
740
Client Proxy
736
712
712
Client 1
Client 2
110

Fig. 11

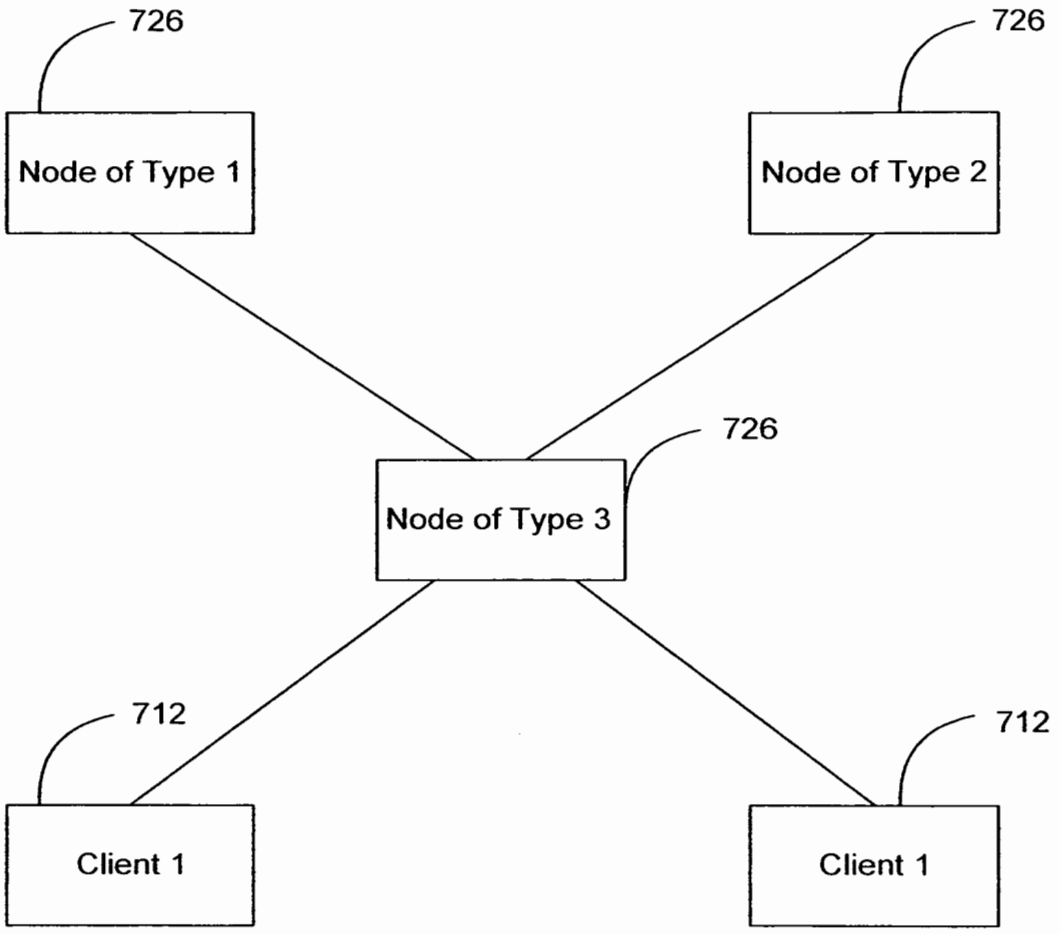

726
726
Node of Type 1
Node of Type 2
726
Node of Type 3
712
712
Client 1
Client 1
110

Fig. 12

ATTORNEY DOCKET NO. 22726-05909

PTO/SB/01A (10-00)
Approved for use through 10/31/2002. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

DECLARATION (37 CFR 1.63) FOR UTILITY OR DESIGN APPLICATION USING AN APPLICATION DATA SHEET (37 CFR 1.76)

As the below named inventor(s), I/we declare that:

This declaration is directed to:

☐ The attached application or

☒ Application No. 10/105,018, filed on March 21, 2002,

☐ as amended on _____ (if applicable);

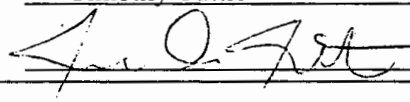
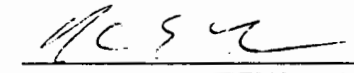
I/we believe that I/we am/are the original and first inventor(s) of the subject matter which is claimed and for which a patent is sought;

I/we have reviewed and understand the contents of the above-identified application, including the claims, as amended by any amendment specifically referred to above;

I/we acknowledge the duty to disclose to the United States Patent and Trademark Office all information known to me/us to be material to patentability as defined in 37 CFR 1.56, including material information which became available between the filing date of the prior application and the National or PCT International filing date of the continuation-in-part application, if applicable; and

All statements made herein of my/our knowledge are true, all statements made herein on information and belief are believed to be true, and further that these statements were made with the knowledge that willful false statements and the like are punishable by fine or imprisonment, or both, under 18 U.S.C. 1001, and may jeopardize the validity of the application or any patent issuing thereon.

FULL NAME OF INVENTOR(S)

Inventor one:	Timothy Tuttle	Citizen of:	United States
Signature:		Date:	5/29/02
Inventor two:	Karl E. Rumelhart	Citizen of:	United States
Signature:		Date:	5/29/02
Inventor three:		Citizen of:	
Signature:		Date:	
Inventor four:		Citizen of:	
Signature:		Date:	

☐ Additional inventors are being named on one additional form attached hereto.

Burden Hour Statement: This collection of information is required by 35 U.S.C. and 37 CFR 1.63. The information is used by the public to file (and the PTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This form is estimated to take 1 minute to complete. This time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, Washington, DC 20231.

BEST AVAILABLE COPY

Application Data Sheet

Application Information

Application number::	
Filing Date::	Herewith
Application Type::	Regular
Subject Matter::	Utility
Suggested classification::	
Suggested Group Art Unit::	
CD-ROM or CD-R??::	
Number of CD disks::	
Number of copies of CDs::	
Sequence Submission::	
Computer Readable Form (CRF)?::	
Number of copies of CRF::	
Title::	ASYNCHRONOUS MESSAGING USING A NODE SPECIALIZATION ARCHITECTURE IN THE DYNAMIC ROUTING NETWORK
Attorney Docket Number::	022193-090614US
Request for Early Publication::	No
Request for Non-Publication::	No
Suggested Drawing Figure::	7
Total Drawing Sheets::	12
Small Entity?::	No
Latin name::	
Variety denomination name::	
Petition included?::	No
Petition Type::	
Licensed US Govt. Agency::	
Contract or Grant Numbers One::	

Secrecy Order in Parent Appl.::	No

Applicant Information

Applicant Authority Type::	Inventor
Primary Citizenship Country::	US
Status::	Full Capacity
Given Name::	Timothy
Middle Name::	
Family Name::	Tuttle
Name Suffix::	
City of Residence::	San Francisco
State or Province of Residence::	CA
Country of Residence::	US
Street of Mailing Address::	3449 22nd Street, Apt. 8
City of Mailing Address::	San Francisco
State or Province of mailing address::	CA
Country of mailing address::	US
Postal or Zip Code of mailing address::	94110

Applicant Authority Type::	Inventor
Primary Citizenship Country::	US
Status::	Full Capacity
Given Name::	Karl
Middle Name::	E.
Family Name::	Rumelhart
Name Suffix::	
City of Residence::	Palo Alto
State or Province of Residence::	CA
Country of Residence::	US
Street of Mailing Address::	555 Washington Avenue
City of Mailing Address::	Palo Alto

State or Province of mailing address::	CA
Country of mailing address::	US
Postal or Zip Code of mailing address::	94301

Correspondence Information

Correspondence Customer Number:: 20350

Representative Information

Representative Customer Number:: 20350

Domestic Priority Information

Application::	Continuity Type::	Parent Application::	Parent Filing Date::
This Application	Continuation of	10/105,018	03/21/02
10/105,018	Continuation-in-part of	10/017,182	12/14/01
10/017,182	An Appn claiming benefit under 35 USC 119(e) of	60/278,303	06/27/01
10/017,182	An Appn claiming benefit under 35 USC 119(e) of	60/280,627	03/29/01
10/017,182	An Appn claiming benefit under 35 USC 119(e) of	60/279,608	03/28/01
10/017,182	An Appn claiming benefit under 35 USC 119(e) of	60/276,847	03/16/01
10/017,182	An Appn claiming benefit under 35 USC 119(e) of	60/256,613	12/18/00

Foreign Priority Information

Country::	Application number::	Filing Date::

Assignee Information

Assignee Name::

Street of mailing address::

City of mailing address::

State or Province of mailing address::

Country of mailing address::

Postal or Zip Code of mailing address::

PTO/SB/06 (12-04)
Approved for use through 7/31/2006. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PATENT APPLICATION FEE DETERMINATION RECORD
Substitute for Form PTO-875

Application or Docket Number
11,396,251

APPLICATION AS FILED – PART I

FOR	(Column 1) NUMBER FILED	(Column 2) NUMBER EXTRA	SMALL ENTITY RATE ($)	SMALL ENTITY FEE ($)	OR	OTHER THAN SMALL ENTITY RATE ($)	OTHER THAN SMALL ENTITY FEE ($)
BASIC FEE (37 CFR 1.16(a), (b), or (c))				150			300
SEARCH FEE (37 CFR 1.16(k), (i), or (m))				250			500
EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))				100			200
TOTAL CLAIMS (37 CFR 1.16(i))	25 minus 20 =	5	X$ 25		OR	X$50	250
INDEPENDENT CLAIMS (37 CFR 1.16(h))	5 minus 3 =	* 2	X$100			X$200	400
APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is $250 ($125 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR						
MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))			180			360	
* If the difference in column 1 is less than zero, enter "0" in column 2.			TOTAL			TOTAL	1650

APPLICATION AS AMENDED – PART II

		(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA	SMALL ENTITY RATE ($)	SMALL ENTITY ADDITIONAL FEE ($)	OR	OTHER THAN SMALL ENTITY RATE ($)	OTHER THAN SMALL ENTITY ADDITIONAL FEE ($)
AMENDMENT A	Total (37 CFR 1.16(i))	*	Minus	**	=	X =		OR	X =	
	Independent (37 CFR 1.16(h))	*	Minus	***	=	X =		OR	X =	
	Application Size Fee (37 CFR 1.16(s))									
	FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))					180		OR	360	
						TOTAL ADD'T FEE		OR	TOTAL ADD'T FEE	

		(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA	RATE ($)	ADDITIONAL FEE ($)	OR	RATE ($)	ADDITIONAL FEE ($)
AMENDMENT B	Total (37 CFR 1.16(i))	*	Minus	**	=	X =		OR	X =	
	Independent (37 CFR 1.16(h))	*	Minus	***	=	X =		OR	X =	
	Application Size Fee (37 CFR 1.16(s))									
	FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))					N/A		OR	N/A	
						TOTAL ADD'T FEE		OR	TOTAL ADD'T FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".
*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.

This collection of information is required by 37 CFR 1.16. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING OR 371 (c) DATE	FIRST NAMED APPLICANT	ATTORNEY DOCKET NUMBER
11/396,251	03/30/2006	Timothy Tuttle	022193-090614US

CONFIRMATION NO. 1129

FORMALITIES LETTER

20350
TOWNSEND AND TOWNSEND AND CREW, LLP
TWO EMBARCADERO CENTER
EIGHTH FLOOR
SAN FRANCISCO, CA 94111-3834

Date Mailed: 04/25/2006

NOTICE TO FILE MISSING PARTS OF NONPROVISIONAL APPLICATION

FILED UNDER 37 CFR 1.53(b)

Filing Date Granted

Items Required To Avoid Abandonment:

An application number and filing date have been accorded to this application. The item(s) indicated below, however, are missing. Applicant is given **TWO MONTHS** from the date of this Notice within which to file all required items and pay any fees required below to avoid abandonment. Extensions of time may be obtained by filing a petition accompanied by the extension fee under the provisions of 37 CFR 1.136(a).

- The statutory basic filing fee is missing.
  Applicant must submit $ **300** *to complete the basic filing fee for a non-small entity. If appropriate, applicant may make a written assertion of entitlement to small entity status and pay the small entity filing fee (37 CFR 1.27).*

The applicant needs to satisfy supplemental fees problems indicated below.

The required item(s) identified below must be timely submitted to avoid abandonment:

- Additional claim fees of **$650** as a non-small entity, including any required multiple dependent claim fee, are required. Applicant must submit the additional claim fees or cancel the additional claims for which fees are due.
- To avoid abandonment, a surcharge (for late submission of filing fee, search fee, examination fee or oath or declaration) as set forth in 37 CFR 1.16(f) of $130 for a non-small entity, must be submitted with the missing items identified in this letter.

SUMMARY OF FEES DUE:

Total additional fee(s) required for this application is **$1780** for a Large Entity

- **$300** Statutory basic filing fee.
- **$130** Surcharge.

- The application search fee has not been paid. Applicant must submit **$500** to complete the search fee.

- The application examination fee has not been paid. Applicant must submit **$200** to complete the examination fee for a large entity

- Total additional claim fee(s) for this application is **$650**
  - **$400** for **2** independent claims over 3.
  - **$250** for **5** total claims over 20.

Replies should be mailed to: Mail Stop Missing Parts
Commissioner for Patents
P.O. Box 1450
Alexandria VA 22313-1450

A copy of this notice MUST be returned with the reply.

R. B

Office of Initial Patent Examination (571) 272-4000, or 1-800-PTO-9199, or 1-800-972-6382
PART 3 - OFFICE COPY

I hereby certify that this correspondence is being filed via EFS-Web with the United States Patent and Trademark Office on July 17, 2006.

TOWNSEND and TOWNSEND and CREW LLP

By: Julianne Kinsler

PATENT
Docket No.: 022193-090614US

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	Confirmation No.: 1129
Timothy Tuttle, et al.	Examiner: Not known
Application No.: 11/396,251	Art Unit: Not known
Filed: March 30, 2006	COMMUNICATION
For: ASYNCHRONOUS MESSAGING USING A NODE SPECIALIZATION ARCHITECTURE IN THE DYNAMIC ROUTING NETWORK	

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This Communication is in response to the Notice to File Missing Parts of Nonprovisional Application mailed April 25, 2006 for the above-captioned matter.

The Commissioner is hereby authorized to charge the summary of fees listed below to Deposit Account No. 20-1430:

1.	Statutory basic filing fee	$ 300.00
2.	Additional claims fee	650.00
3.	Surcharge for late filing	130.00
4.	Search Fee	500.00
5.	Examination Fee	200.00
6.	One month extension fee	120.00
	Total	**$1900.00**

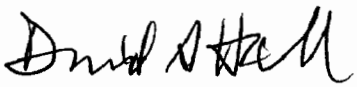
CONCLUSION

Also attached is a copy of the Notice to File Missing Parts of Nonprovisional Application. In view of the foregoing, Applicants have provided all missing fees.

The Commissioner is hereby authorized to charge any additional fees due or credit any overpayments associated with this matter to Deposit Account 20-1430.

If the Examiner believes a telephone conference would expedite prosecution of this application, please telephone the undersigned at 858-350-6100.

Respectfully submitted,



David A. Hall
Reg. No. 32,233

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, Eighth Floor
San Francisco, California 94111-3834
Tel: (858) 350-6100
Fax: (415) 576-0300
DAH:dah

60822954 v1

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING OR 371 (c) DATE	FIRST NAMED APPLICANT	ATTORNEY DOCKET NUMBER
11/396,251	03/30/2006	Timothy Tuttle	022193-090614US

CONFIRMATION NO. 1129
FORMALITIES LETTER

20350
TOWNSEND AND TOWNSEND AND CREW, LLP
TWO EMBARCADERO CENTER
EIGHTH FLOOR
SAN FRANCISCO, CA 94111-3834

Date Mailed: 04/25/2006

NOTICE TO FILE MISSING PARTS OF NONPROVISIONAL APPLICATION

FILED UNDER 37 CFR 1.53(b)

Filing Date Granted

Items Required To Avoid Abandonment:

An application number and filing date have been accorded to this application. The item(s) indicated below, however, are missing. Applicant is given **TWO MONTHS** from the date of this Notice within which to file all required items and pay any fees required below to avoid abandonment. Extensions of time may be obtained by filing a petition accompanied by the extension fee under the provisions of 37 CFR 1.136(a).

- The statutory basic filing fee is missing.
  Applicant must submit $ **300** *to complete the basic filing fee for a non-small entity. If appropriate, applicant may make a written assertion of entitlement to small entity status and pay the small entity filing fee (37 CFR 1.27).*

The applicant needs to satisfy supplemental fees problems indicated below.

The required item(s) identified below must be timely submitted to avoid abandonment:

- Additional claim fees of **$650** as a non-small entity, including any required multiple dependent claim fee, are required. Applicant must submit the additional claim fees or cancel the additional claims for which fees are due.
- To avoid abandonment, a surcharge (for late submission of filing fee, search fee, examination fee or oath or declaration) as set forth in 37 CFR 1.16(f) of $130 for a non-small entity, must be submitted with the missing items identified in this letter.

SUMMARY OF FEES DUE:

Total additional fee(s) required for this application is **$1780** for a Large Entity

- **$300** Statutory basic filing fee.
- **$130** Surcharge.

- The application search fee has not been paid. Applicant must submit **$500** to complete the search fee.

- The application examination fee has not been paid. Applicant must submit **$200** to complete the examination fee for a large entity

- Total additional claim fee(s) for this application is **$650**
  - **$400** for **2** independent claims over 3.
  - **$250** for **5** total claims over 20.

Replies should be mailed to: Mail Stop Missing Parts
Commissioner for Patents
P.O. Box 1450
Alexandria VA 22313-1450

A copy of this notice MUST be returned with the reply.

RCB

Office of Initial Patent Examination (571) 272-4000, or 1-800-PTO-9199, or 1-800-972-6382
PART 2 - COPY TO BE RETURNED WITH RESPONSE

Electronic Acknowledgement Receipt

EFS ID:	1115589
Application Number:	11396251
Confirmation Number:	1129
Title of Invention:	Asynchronous messaging using a node specialization architecture in the dynamic routing network
First Named Inventor:	Timothy Tuttle
Customer Number:	20350
Filer:	David A Hall/Leeanne Lawlor
Filer Authorized By:	David A Hall
Attorney Docket Number:	022193-090614US
Receipt Date:	17-JUL-2006
Filing Date:	30-MAR-2006
Time Stamp:	15:33:31
Application Type:	Utility
International Application Number:	

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)	Multi Part	Pages
1		022193-090614MissingParts .pdf	212821	yes	4

	Multipart Description		
	Doc Desc	**Start**	**End**
	Miscellaneous Incoming Letter	1	2
	Miscellaneous Incoming Letter	3	4

Warnings:	
Information:	
Total Files Size (in bytes):	212821

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

I hereby certify that this correspondence is being filed via EFS-Web with the United States Patent and Trademark Office on July 17, 2006.

TOWNSEND and TOWNSEND and CREW LLP

By: Liliane Krivulka

PATENT
Docket No.: 022193-090614US

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	Confirmation No.: 1129
Timothy Tuttle, et al.	Examiner: Not known
Application No.: 11/396,251	Art Unit: Not known
Filed: March 30, 2006	COMMUNICATION
For: ASYNCHRONOUS MESSAGING USING A NODE SPECIALIZATION ARCHITECTURE IN THE DYNAMIC ROUTING NETWORK	

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This Communication is in response to the Notice to File Missing Parts of Nonprovisional Application mailed April 25, 2006 for the above-captioned matter.

The Commissioner is hereby authorized to charge the summary of fees listed below to Deposit Account No. 20-1430:

1. Statutory basic filing fee	$ 300.00
2. Additional claims fee	650.00
3. Surcharge for late filing	130.00
4. Search Fee	500.00
5. Examination Fee	200.00
6. One month extension fee	120.00
Total	**$1900.00**

07/31/2006 VWAN1 00000007 201430 11396251
01 FC:1251 120.00 DA

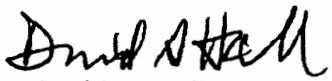
CONCLUSION

Also attached is a copy of the Notice to File Missing Parts of Nonprovisional Application. In view of the foregoing, Applicants have provided all missing fees.

The Commissioner is hereby authorized to charge any additional fees due or credit any overpayments associated with this matter to Deposit Account 20-1430.

If the Examiner believes a telephone conference would expedite prosecution of this application, please telephone the undersigned at 858-350-6100.

Respectfully submitted,



David A. Hall
Reg. No. 32,233

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, Eighth Floor
San Francisco, California 94111-3834
Tel: (858) 350-6100
Fax: (415) 576-0300
DAH:dah

60822954 v1

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING OR 371 (c) DATE	FIRST NAMED APPLICANT	ATTORNEY DOCKET NUMBER
11/396,251	03/30/2006	Timothy Tuttle	022193-090614US

CONFIRMATION NO. 1129
FORMALITIES LETTER

20350
TOWNSEND AND TOWNSEND AND CREW, LLP
TWO EMBARCADERO CENTER
EIGHTH FLOOR
SAN FRANCISCO, CA 94111-3834

Date Mailed: 04/25/2006

NOTICE TO FILE MISSING PARTS OF NONPROVISIONAL APPLICATION

FILED UNDER 37 CFR 1.53(b)

Filing Date Granted

Items Required To Avoid Abandonment:

An application number and filing date have been accorded to this application. The item(s) indicated below, however, are missing. Applicant is given **TWO MONTHS** from the date of this Notice within which to file all required items and pay any fees required below to avoid abandonment. Extensions of time may be obtained by filing a petition accompanied by the extension fee under the provisions of 37 CFR 1.136(a).

- The statutory basic filing fee is missing.
  Applicant must submit $ **300** *to complete the basic filing fee for a non-small entity. If appropriate, applicant may make a written assertion of entitlement to small entity status and pay the small entity filing fee (37 CFR 1.27).*

The applicant needs to satisfy supplemental fees problems indicated below.

The required item(s) identified below must be timely submitted to avoid abandonment:

- Additional claim fees of **$650** as a non-small entity, including any required multiple dependent claim fee, are required. Applicant must submit the additional claim fees or cancel the additional claims for which fees are due.
- To avoid abandonment, a surcharge (for late submission of filing fee, search fee, examination fee or oath or declaration) as set forth in 37 CFR 1.16(f) of $130 for a non-small entity, must be submitted with the missing items identified in this letter.

SUMMARY OF FEES DUE:

Total additional fee(s) required for this application is **$1780** for a Large Entity

- **$300** Statutory basic filing fee.
- **$130** Surcharge.

- The application search fee has not been paid. Applicant must submit **$500** to complete the search fee.

07/31/2006 WGANI1 00000019 201430 11396251

01 FC:2051 300.00 DA
02 FC:2052 130.00 DA
03 FC:2053 [illegible] DA
04 FC:2054 [illegible] DA
05 FC:2055 [illegible] DA
06 FC:2056 [illegible] DA
07 FC:2057 [illegible] DA

- The application examination fee has not been paid. Applicant must submit **$200** to complete the examination fee for a large entity

- Total additional claim fee(s) for this application is **$650**
  - **$400** for **2** independent claims over 3.
  - **$250** for **5** total claims over 20.

Replies should be mailed to: Mail Stop Missing Parts
Commissioner for Patents
P.O. Box 1450
Alexandria VA 22313-1450

A copy of this notice MUST be returned with the reply.

RCB

Office of Initial Patent Examination (571) 272-4000, or 1-800-PTO-9199, or 1-800-972-6382
PART 2 - COPY TO BE RETURNED WITH RESPONSE

I hereby certify that this correspondence is being filed via EFS-Web with the United States Patent and Trademark Office on 8-30-06.

TOWNSEND and TOWNSEND and CREW LLP

By: Lillian Montano

PATENT

Attorney Docket No.: 022193-090614US

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	Confirmation No.
Timothy Tuttle, et al.	Examiner:
Application No.: 11/396,251	Technology Center/Art Unit:
Filed: March 30, 2006	PRELIMINARY AMENDMENT
For: ASYNCHRONOUS MESSAGING USING A NODE SPECIALIZATION ARCHITECTURE IN THE DYNAMIC ROUTING NETWORK	
Customer No.: 20350	

Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Prior to examination of the above-referenced application, please enter the following amendments and remarks:

Amendments to the Specification begin on page 2 of this paper.

Amendments to the Claims are reflected in the listing of claims which begins on page 3 of this paper.

Amendments to the Drawings begin on page 10 of this paper and include both an attached replacement sheet and an annotated sheet showing changes.

Remarks/Arguments begin on page 11 of this paper.

An **Appendix** including amended drawing figures is attached following page 11 of this paper.

IPR2025-00200 / 722 Patent Page 88 of 1171 EX1002

Amendments to the Specification:

Please replace paragraph **[0001]** with the following amended paragraph, in which added text is indicated by underlining, and deleted text is indicated by ~~strikethrough~~, and changes are identified by a vertical bar in the margin:

[0001] This application is a continuation of U.S. Application No. 10/105,018 filed March 21, 2002 (now U.S. Patent No. 7,051,070), which is a continuation-in-part of U.S. application Ser. No. 10/017,182 filed Dec. 14, 2001, which claims the benefit of U.S. Provisional Application No. 60/256,613, filed Dec. 18, 2000, U.S. Provisional Application No. 60/276,847, filed Mar. 16, 2001, U.S. Provisional Application No. 60/278,303, filed Mar. 21, 2001, U.S. Provisional Application No. 60/279,608, filed Mar. 28, 2001, and U.S. Provisional Application No. 60/280,627, filed Mar. 29, 2001, all of which are hereby incorporated by reference herein.

Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in this application. Added text is indicated by underlining, and deleted text is indicated by ~~strikethrough~~. Changes are identified by a vertical bar in the margin.

Listing of Claims:

1-25. (canceled).

26. (new) A method for providing dynamic content over a network, the method comprising:

receiving an update message from an input source, the update message identifying a live object and containing data for updating a property of the live object;

identifying a category of the update message;

identifying a mapping of the category to a node type;

routing the update message to a node having the mapped node type;

determining a client that has registered for updates of the live object and routing the data to the client, wherein the client is adapted to process the update data and update the property of the live object.

27. (new) The method of claim 26, wherein determining a client and routing the data to the client are performed at the node.

28. (new) The method of claim 26, wherein determining a client comprises extracting an object ID from the update message and determining a connection to the client.

29. (new) The method of claim 28, wherein determining a connection comprises determining at least one client proxy with which the client communicates and routing the data to the client proxy, and then routing the data from the client proxy to the client.

30. (new) The method of claim 29, wherein client registration information concerning the client connection is maintained at the node.

IPR2025-00200 / 722 Patent Page 90 of 1171 EX1002

31. (new) The method of claim 29, wherein client registration information concerning the client connection is maintained at the client proxy.

32. (new) The method of claim 28, wherein:

routing the update message comprises routing the update message to a proxy node comprising a node that is adapted to receive messages of more than one message category; and

determining a connection comprises determining at least one corresponding node having a node type that is mapped to the message category, with which the registered client communicates, and routing the data to the corresponding node.

33. (new) A routing network for enabling dynamic updating of a property of a live object at a client coupled to the network, the routing network comprising:

a gateway that receives an update message from an input source, the update message identifying a live object and containing data for updating a property of the live object, wherein the gateway identifies a category of the update message, identifies a mapping of the category to a node type, and routes the update message in accordance with the identified mapping; and

routing means for determining a client that has registered for updates of the live object and routing the data from a node that has received the routed update message to the registered client, wherein the registered client is adapted to process the data and update the property of the live object.

34. (new) The routing network of claim 33, wherein the routing means extracts an object ID from the update message and determines a connection to the registered client for routing the data to the registered client.

35. (new) The routing network of claim 34, wherein the routing means determines a connection by determining at least one client proxy with which the registered client communicates, and then routing the data to the client proxy.

36. (new) The routing network of claim 35, wherein client registration information concerning the client connection is maintained at the node.

37. (new) The routing network of claim 35, wherein client registration information concerning the client connection is maintained at the client proxy.

38. (new) The routing network of claim 33, wherein:

the gateway routs the update message by routing the update message to a proxy node comprising a node that is adapted to receive messages of more than one message category; and

the routing means determines at least one corresponding node that has a node type mapped to the message category and with which the registered client communicates, and then routing the data to the corresponding node.

39. (new) A computer method for providing dynamic content over a network, the method comprising:

providing a data representation to a client device coupled to the network, wherein the data representation includes at least one live object that is recognized by the client device, and wherein the client responds to the live object of the data representation by determining an object ID of the live object to register for updates of the live object with a routing network, such that registering the client with the routing network provides client connection information to the routing network;

sending an update message to the routing network, wherein the update message identifies the live object and contains update data that updates a property of the live object such that the routing network has sufficient information to identify the client device as a registered device and send a routed message containing the update data to the client device, such that the client device processes the routed message upon receipt to update the property of the live object at the client device.

40. (new) The method of claim 39, wherein the live object of the data representation causes the client device to register with a client proxy of the network.

41. (new) The method of claim 39, wherein the live object of the data representation causes the client device to register with a node of the network.

42. (new) The method of claim 39, wherein the received data representation includes an activation module that is executed by the client device and is adapted to register the live object with the routing network.

43. (new) The method of claim 42, wherein the activation module determines a node type that handles registration and causes the client device to register with a node of the determined registration node type.

44. (new) The method of claim 42, wherein the activation module determines a message category of the data representation and causes the client device to register with a node having a node type corresponding to the message category.

45. (new) An apparatus for providing dynamic content over a network, the apparatus comprising:

a content provider that provides a data representation to a client device coupled to the network, wherein the data representation includes at least one live object that is recognized by the client device, and wherein the client responds to the live object of the data representation by determining an object ID of the live object to register for updates of the live object with a routing network, such that registering the client with the routing network provides client connection information to the routing network;

an information provider that sends an update message to the routing network, wherein the update message identifies the live object and contains update data that updates a property of the live object such that the routing network has sufficient information to identify the client device as a registered device and send a routed message containing the update data to the

client device, such that the client device processes the routed message upon receipt to update the property of the live object at the client device.

46. (new) The apparatus of claim 45, wherein the live object of the data representation causes the client device to register with a client proxy of the network.

47. (new) The apparatus of claim 45, wherein the live object of the data representation causes the client device to register with a node of the network.

48. (new) The apparatus of claim 45, wherein the received data representation includes an activation module that is executed by the client device and is adapted to register the live object with the routing network.

49. (new) The apparatus of claim 48, wherein the activation module determines a node type that handles registration and causes the client device to register with a node of the determined registration node type.

50. (new) The apparatus of claim 48, wherein the activation module determines a message category of the data representation and causes the client device to register with a node having a node type corresponding to the message category.

51. (new) A computer-readable medium having computer code embodied therein for providing dynamic content over a network, the computer program code configured to cause a computing device to perform operations comprising:

providing a data representation to a client device coupled to the network, wherein the data representation includes at least one live object that is recognized by the client device, and wherein the client responds to the live object of the data representation by determining an object ID of the live object to register for updates of the live object with a routing network, such that registering the client with the routing network provides client connection information to the routing network;

sending an update message to the routing network, wherein the update message identifies the live object and contains update data that updates a property of the live object such that the routing network has sufficient information to identify the client device as a registered device and send a routed message containing the update data to the client device, such that the client device processes the routed message upon receipt to update the property of the live object at the client device.

52. (new) The computer-readable medium of claim 51, wherein the live object of the data representation causes the client device to register with a client proxy of the network.

53. (new) The computer-readable medium of claim 51, wherein the live object of the data representation causes the client device to register with a node of the network.

54. (new) The computer-readable medium of claim 51, wherein the received data representation includes an activation module that is executed by the client device and is adapted to register the live object with the routing network.

55. (new) The computer-readable medium of claim 54, wherein the activation module determines a node type that handles registration and causes the client device to register with a node of the determined registration node type.

56. (new) The computer-readable medium of claim 54, wherein the activation module determines a message category of the data representation and causes the client device to register with a node having a node type corresponding to the message category.

57. (new) A device for providing dynamic content over a network, the device comprising:

logic configured to provide a data representation to a client device coupled to the network, wherein the data representation includes at least one live object that is recognized by the client device, and wherein the client responds to the live object of the data representation by

determining an object ID of the live object to register for updates of the live object with a routing network, such that registering the client with the routing network provides client connection information to the routing network;

logic configured to provide an update message to the routing network, wherein the update message identifies the live object and contains update data that updates a property of the live object such that the routing network has sufficient information to identify the client device as a registered device and send a routed message containing the update data to the client device, such that the client device processes the routed message upon receipt to update the property of the live object at the client device.

58. (new) The device of claim 57, wherein the live object of the data representation causes the client device to register with a client proxy of the network.

59. (new) The device of claim 57, wherein the live object of the data representation causes the client device to register with a node of the network.

60. (new) The device of claim 57, wherein the received data representation includes an activation module that is executed by the client device and is adapted to register the live object with the routing network.

61. (new) The device of claim 60, wherein the activation module determines a node type that handles registration and causes the client device to register with a node of the determined registration node type.

62. (new) The device of claim 60, wherein the activation module determines a message category of the data representation and causes the client device to register with a node having a node type corresponding to the message category.

Amendments to the Drawings:

The attached drawing sheet includes changes to Fig. 7. This sheet replaces the original drawing sheet for Fig. 7 and comprises changing reference numeral "7224" to --722A--.

Attachment: Replacement Sheet
Annotated Sheet Showing Changes

REMARKS/ARGUMENTS

Entry of this Preliminary Amendment is requested prior to a first Office Action in this Continuation application. New claims 26-62 have been added to the application. Original claims 1-25 have been canceled. Claims 26-62 are pending after entry of this preliminary amendment. No new matter has been added.

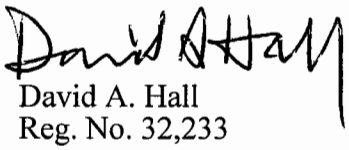
Drawing corrections comprise a correction to "7224" in Figure 7 so it correctly reads --722A--.

CONCLUSION

In view of the foregoing, Applicants believe all claims now pending in this Application are in condition for allowance. The issuance of a formal Notice of Allowance at an early date is respectfully requested.

If the Examiner believes a telephone conference would expedite prosecution of this application, please telephone the undersigned at 858-350-6100.

Respectfully submitted,



David A. Hall
Reg. No. 32,233

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, Eighth Floor
San Francisco, California 94111-3834
Tel: 858-350-6100
Fax: 415-576-0300
Attachments
DAH:dah
60855839 v3

ANNOTATED SHEET

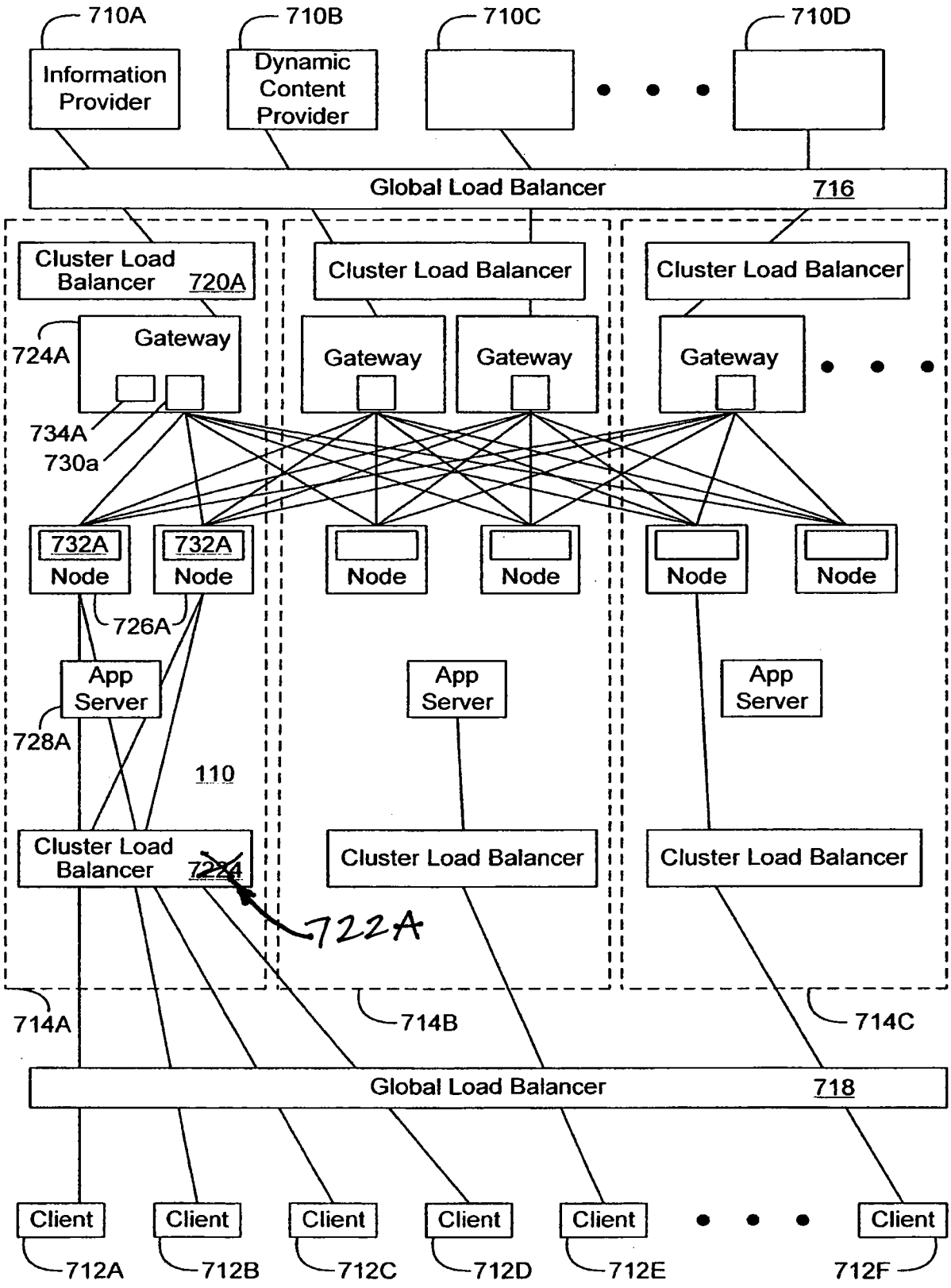


Fig. 7

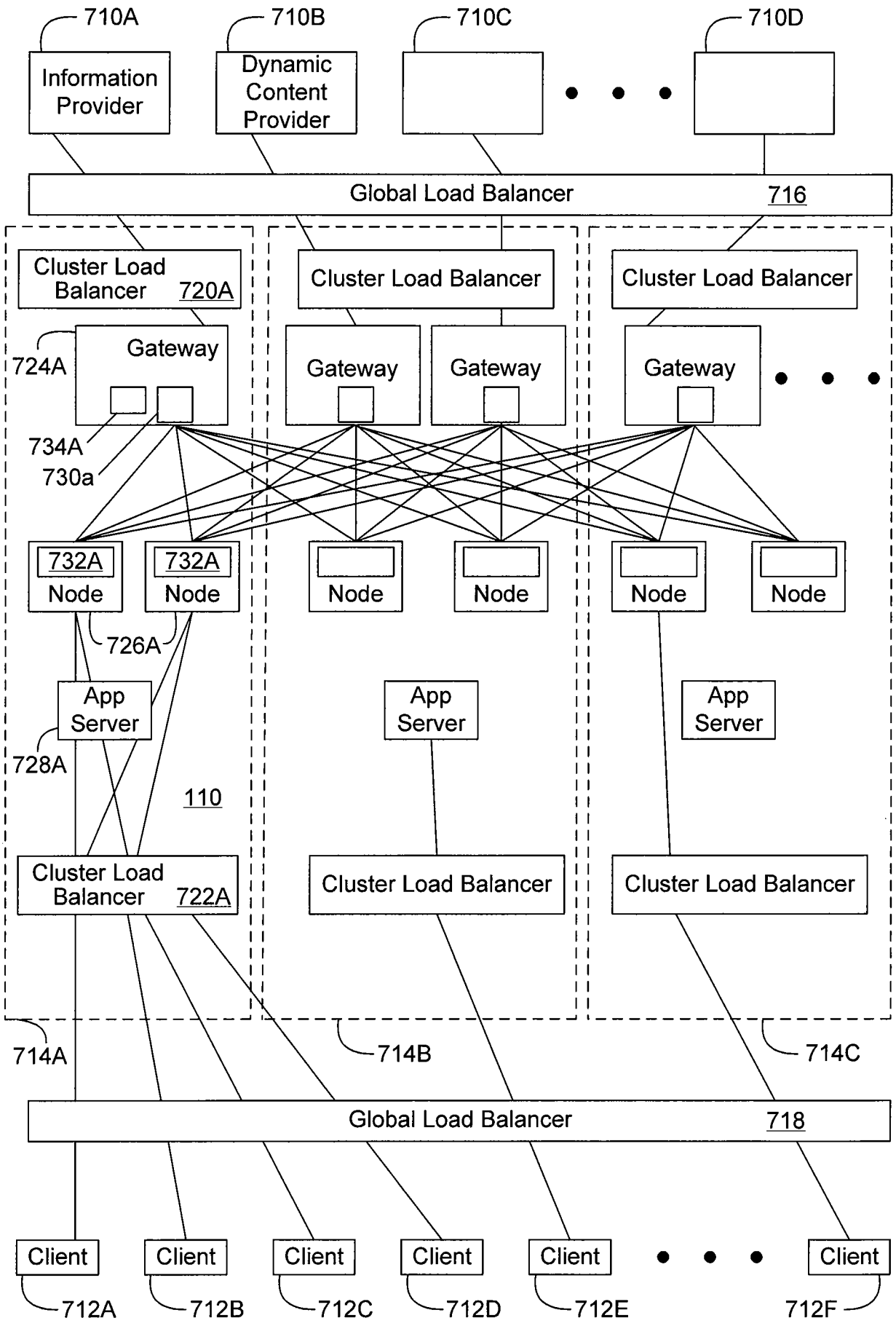

710A
710B
710C
710D
Information Provider
Dynamic Content Provider
Global Load Balancer
716
Cluster Load Balancer
720A
Cluster Load Balancer
Cluster Load Balancer
724A
Gateway
Gateway
Gateway
Gateway
734A
730a
732A
Node
732A
Node
Node
Node
Node
Node
726A
App Server
App Server
App Server
728A
110
Cluster Load Balancer
722A
Cluster Load Balancer
Cluster Load Balancer
714A
714B
714C
Global Load Balancer
718
Client
Client
Client
Client
Client
Client
712A
712B
712C
712D
712E
712F

Fig. 7

Electronic Acknowledgement Receipt

EFS ID:	1179526
Application Number:	11396251
Confirmation Number:	1129
Title of Invention:	Asynchronous messaging using a node specialization architecture in the dynamic routing network
First Named Inventor:	Timothy Tuttle
Customer Number:	20350
Filer:	David A Hall/Lillian Montano
Filer Authorized By:	David A Hall
Attorney Docket Number:	022193-090614US
Receipt Date:	30-AUG-2006
Filing Date:	30-MAR-2006
Time Stamp:	18:48:24
Application Type:	Utility
International Application Number:	

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)	Multi Part	Pages
1		PrelimAmend.pdf	526546	yes	14

	Multipart Description		
	Doc Desc	**Start**	**End**
	Transmittal letter	1	1
	Preliminary Amendment	2	12
	Drawings	13	14

Warnings:	
Information:	
Total Files Size (in bytes):	526546

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

PTO/SB/21 (09-04)

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Application Number	11/396,251
Filing Date	March 30, 2006
First Named Inventor	Tuttle, Timothy
Art Unit	
Examiner Name	
Attorney Docket Number	022193-090614US

Total Number of Pages in This Submission

ENCLOSURES *(Check all that apply)*

- ☐ Fee Transmittal Form
  - ☐ Fee Attached
- ☒ Amendment/Reply
  - ☐ After Final
  - ☐ Affidavits/declaration(s)
- ☐ Extension of Time Request
- ☐ Express Abandonment Request
- ☐ Information Disclosure Statement
- ☐ Certified Copy of Priority Document(s)
- ☐ Reply to Missing Parts/ Incomplete Application
  - ☐ Reply to Missing Parts under 37 CFR 1.52 or 1.53
- ☐ Drawing(s)
- ☐ Licensing-related Papers
- ☐ Petition
- ☐ Petition to Convert to a Provisional Application
- ☐ Power of Attorney, Revocation Change of Correspondence Address
- ☐ Terminal Disclaimer
- ☐ Request for Refund
- ☐ CD, Number of CD(s) _____
  - ☐ Landscape Table on CD
- ☐ After Allowance Communication to TC
- ☐ Appeal Communication to Board of Appeals and Interferences
- ☐ Appeal Communication to TC **(Appeal Notice, Brief, Reply Brief)**
- ☐ Proprietary Information
- ☐ Status Letter
- ☒ Other Enclosure(s) (please identify below):

Replacement Drawing Figure 7

Remarks: The Commissioner is authorized to charge any additional fees to Deposit Account 20-1430.

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm Name	Townsend and Townsend and Crew LLP		
Signature	David A Hall		
Printed name	David A. Hall		
Date	August 30, 2006	Reg. No.	32,233

CERTIFICATE OF TRANSMISSION/MAILING

I hereby certify that this correspondence is being filed via EFS-Web with the United States Patent and Trademark Office on the date shown below.

Signature	Lillian Montano		
Typed or printed name	Lillian M. Montano	Date	8-30-06

60858875 v1

PTO/SB/80 (01-06)
Approved for use through 12/31/2008. OMB 0651-0035
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

POWER OF ATTORNEY TO PROSECUTE APPLICATIONS BEFORE THE USPTO

I hereby revoke all previous powers of attorney given in the application identified in the attached statement under 37 CFR 3.73(b).

I hereby appoint:

[X] Practitioners associated with the Customer Number: 26111

OR

[ ] Practitioner(s) named below (if more than ten patent practitioners are to be named, then a customer number must be used):

Name	Registration Number		Name	Registration Number

as attorney(s) or agent(s) to represent the undersigned before the United States Patent and Trademark Office (USPTO) in connection with any and all patent applications assigned only to the undersigned according to the USPTO assignment records or assignment documents attached to this form in accordance with 37 CFR 3.73(b).

Please change the correspondence address for the application identified in the attached statement under 37 CFR 3.73(b) to:

[X] The address associated with Customer Number: 26111

OR

[ ] Firm or Individual Name

Address					
City		State		Zip	
Country		Telephone		Email	

Assignee Name and Address:

Shaw Parsing LLC
2215-B Renaissance Drive, Suite 5
Las Vegas, NV 89119

A copy of this form, together with a statement under 37 CFR 3.73(b) (Form PTO/SB/96 or equivalent) is required to be filed in each application in which this form is used. The statement under 37 CFR 3.73(b) may be completed by one of the practitioners appointed in this form if the appointed practitioner is authorized to act on behalf of the assignee, and must identify the application in which this Power of Attorney is to be filed.

SIGNATURE of Assignee of Record

The individual whose signature and title is supplied below is authorized to act on behalf of the assignee

Signature		Date	24 MAY 2007
Name	Julia Ceffalo	Telephone	
Title	Authorized Person for Shaw Parsing LLC		

PTO/SB/96 (04-07)
Approved for use through 09/30/2007. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

STATEMENT UNDER 37 CFR 3.73(b)

2222.775000E

Applicant/Patent Owner: Timothy TUTTLE et al.

Application No./Patent No.: 11/396,251 Filed/Issue Date: March 20, 2006

Entitled: Asynchronous Messaging Using a Node Specialization Architecture in the Dynamic Routing Network

Shaw Parsing LLC, a Limited Liability Company
(Name of Assignee) (Type of Assignee, e.g., corporation, partnership, university, government agency, etc.)

states that it is:

1. [X] the assignee of the entire right, title, and interest; or

2. [ ] an assignee of less than the entire right, title and interest
(The extent (by percentage) of its ownership interest is _____%)

in the patent application/patent identified above by virtue of either:

A. [ ] An assignment from the inventor(s) of the patent application/patent identified above. The assignment was recorded in the United States Patent and Trademark Office at Reel _____, Frame _____, or for which a copy thereof is attached.

OR

B. [X] A chain of title from the inventor(s), of the patent application/patent identified above, to the current assignee as follows:

1. From: Timothy TUTTLE et al. To: Bang Networks, Inc.
The document was recorded in the United States Patent and Trademark Office at
Reel 019462, Frame 0914, or for which a copy thereof is attached.

2. From: Bang Networks, Inc. To: Shaw Parsing LLC
The document was recorded in the United States Patent and Trademark Office at
Reel 019462, Frame 0936, or for which a copy thereof is attached.

3. From: _____ To: _____
The document was recorded in the United States Patent and Trademark Office at
Reel _____, Frame _____, or for which a copy thereof is attached.

[ ] Additional documents in the chain of title are listed on a supplemental sheet.

[X] As required by 37 CFR 3.73(b)(1)(i), the documentary evidence of the chain of title from the original owner to the assignee was, or concurrently is being, submitted for recordation pursuant to 37 CFR 3.11.

[NOTE: A separate copy (*i.e.*, a true copy of the original assignment document(s)) must be submitted to Assignment Division in accordance with 37 CFR Part 3, to record the assignment in the records of the USPTO. See MPEP 302.08]

The undersigned (whose title is supplied below) is authorized to act on behalf of the assignee.

Signature	Date
Glenn J. Perry, Registration No. 28,458	202.371.2600
Printed or Typed Name	Telephone Number
Authorized Person	
Title	

This collection of information is required by 37 CFR 3.73(b). The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

Electronic Acknowledgement Receipt

EFS ID:	1906957
Application Number:	11396251
International Application Number:	
Confirmation Number:	1129
Title of Invention:	Asynchronous messaging using a node specialization architecture in the dynamic routing network
First Named Inventor/Applicant Name:	Timothy Tuttle
Customer Number:	20350
Filer:	Glenn J. Perry/Nihad Heto
Filer Authorized By:	Glenn J. Perry
Attorney Docket Number:	022193-090614US
Receipt Date:	25-JUN-2007
Filing Date:	30-MAR-2006
Time Stamp:	15:59:46
Application Type:	Utility

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)	Multi Part /.zip	Pages (if appl.)
1		2222775000Epoaandstatement under373b.pdf	101533	yes	2

	Multipart Description/PDF files in .zip description		
	Document Description	**Start**	**End**
	Power of Attorney	1	1
	Assignee showing of ownership per 37 CFR 3.73(b).	2	2
Warnings:			
Information:			
	Total Files Size (in bytes):	101533	

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING OR 371 (c) DATE	FIRST NAMED APPLICANT	ATTY. DOCKET NO./TITLE
11/396,251	03/30/2006	Timothy Tuttle	022193-090614US

CONFIRMATION NO. 1129

OC000000024616239

20350
TOWNSEND AND TOWNSEND AND CREW, LLP
TWO EMBARCADERO CENTER
EIGHTH FLOOR
SAN FRANCISCO, CA 94111-3834

Date Mailed: 06/29/2007

NOTICE REGARDING CHANGE OF POWER OF ATTORNEY

This is in response to the Power of Attorney filed 06/25/2007.

- The Power of Attorney to you in this application has been revoked by the assignee who has intervened as provided by 37 CFR 3.71. Future correspondence will be mailed to the new address of record(37 CFR 1.33).

Boedy

Office of Initial Patent Examination (571) 272-4000, or 1-800-PTO-9199
OFFICE COPY

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING OR 371 (c) DATE	FIRST NAMED APPLICANT	ATTY. DOCKET NO./TITLE
11/396,251	03/30/2006	Timothy Tuttle	022193-090614US

CONFIRMATION NO. 1129

26111
STERNE, KESSLER, GOLDSTEIN & FOX P.L.L.C.
1100 NEW YORK AVENUE, N.W.
WASHINGTON, DC 20005

OC000000024616286

Date Mailed: 06/29/2007

NOTICE OF ACCEPTANCE OF POWER OF ATTORNEY

This is in response to the Power of Attorney filed 06/25/2007.

The Power of Attorney in this application is accepted. Correspondence in this application will be mailed to the above address as provided by 37 CFR 1.33.

Badiy

Office of Initial Patent Examination (571) 272-4000, or 1-800-PTO-9199
OFFICE COPY

I hereby certify that this correspondence is being filed via EFS-Web with the United States Patent and Trademark Office on 8-30-06.

TOWNSEND and TOWNSEND and CREW LLP
By: Gillian Montano

PATENT
Attorney Docket No.: 022193-090614US

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	Confirmation No.
Timothy Tuttle, et al.	Examiner:
Application No.: 11/396,251	Technology Center/Art Unit:
Filed: March 30, 2006	PRELIMINARY AMENDMENT
For: ASYNCHRONOUS MESSAGING USING A NODE SPECIALIZATION ARCHITECTURE IN THE DYNAMIC ROUTING NETWORK	
Customer No.: 20350	

Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Prior to examination of the above-referenced application, please enter the following amendments and remarks:

Amendments to the Specification begin on page 2 of this paper.

Amendments to the Claims are reflected in the listing of claims which begins on page 3 of this paper.

Amendments to the Drawings begin on page 10 of this paper and include both an attached replacement sheet and an annotated sheet showing changes.

Remarks/Arguments begin on page 11 of this paper.

An **Appendix** including amended drawing figures is attached following page 11 of this paper.

07/05/2007 MTEKLEMI 00000012 201430 11396251
01 FC:1201 200.00 DA
02 FC:1202 600.00 DA

IPR2025-00200 / 722 Patent Page 110 of 1171 EX1002

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING or 371(c) DATE	GRP ART UNIT	FIL FEE REC'D	ATTY.DOCKET.NO	TOT CLAIMS	IND CLAIMS
11/396,251	03/30/2006	2155	2580	022193-090614US	37	6

CONFIRMATION NO. 1129

UPDATED FILING RECEIPT

26111
STERNE, KESSLER, GOLDSTEIN & FOX P.L.L.C.
1100 NEW YORK AVENUE, N.W.
WASHINGTON, DC20005

Date Mailed: 07/06/2007

Receipt is acknowledged of this regular Patent Application. It will be considered in its order and you will be notified as to the results of the examination. Be sure to provide the U.S. APPLICATION NUMBER, FILING DATE, NAME OF APPLICANT, and TITLE OF INVENTION when inquiring about this application. Fees transmitted by check or draft are subject to collection. Please verify the accuracy of the data presented on this receipt. **If an error is noted on this Filing Receipt, please mail to the Commissioner for Patents P.O. Box 1450 Alexandria Va 22313-1450. Please provide a copy of this Filing Receipt with the changes noted thereon. If you received a "Notice to File Missing Parts" for this application, please submit any corrections to this Filing Receipt with your reply to the Notice. When the USPTO processes the reply to the Notice, the USPTO will generate another Filing Receipt incorporating the requested corrections (if appropriate).**

Applicant(s)

Timothy Tuttle, San Francisco, CA;
Karl E. Rumelhart, Palo Alto, CA;

Power of Attorney: The patent practitioners associated with Customer Number 26111

Domestic Priority data as claimed by applicant

This application is a CON of 10/105,018 03/21/2002 PAT 7,051,070
which is a CIP of 10/017,182 12/14/2001 PAT 7,043,525
which claims benefit of 60/278,303 03/21/2001
and claims benefit of 60/280,627 03/29/2001
and claims benefit of 60/279,608 03/28/2001
and claims benefit of 60/276,847 03/16/2001
and claims benefit of 60/256,613 12/18/2000

Foreign Applications

If Required, Foreign Filing License Granted: 04/25/2006

The country code and number of your priority application, to be used for filing abroad under the Paris Convention, is **US11/396,251**

Projected Publication Date: 10/11/2007

Non-Publication Request: No

Early Publication Request: No

Title

Asynchronous messaging using a node specialization architecture in the dynamic routing network

Preliminary Class

709

PROTECTING YOUR INVENTION OUTSIDE THE UNITED STATES

Since the rights granted by a U.S. patent extend only throughout the territory of the United States and have no effect in a foreign country, an inventor who wishes patent protection in another country must apply for a patent in a specific country or in regional patent offices. Applicants may wish to consider the filing of an international application under the Patent Cooperation Treaty (PCT). An international (PCT) application generally has the same effect as a regular national patent application in each PCT-member country. The PCT process **simplifies** the filing of patent applications on the same invention in member countries, but **does not result** in a grant of "an international patent" and does not eliminate the need of applicants to file additional documents and fees in countries where patent protection is desired.

Almost every country has its own patent law, and a person desiring a patent in a particular country must make an application for patent in that country in accordance with its particular laws. Since the laws of many countries differ in various respects from the patent law of the United States, applicants are advised to seek guidance from specific foreign countries to ensure that patent rights are not lost prematurely.

Applicants also are advised that in the case of inventions made in the United States, the Director of the USPTO must issue a license before applicants can apply for a patent in a foreign country. The filing of a U.S. patent application serves as a request for a foreign filing license. The application's filing receipt contains further information and guidance as to the status of applicant's license for foreign filing.

Applicants may wish to consult the USPTO booklet, "General Information Concerning Patents" (specifically, the section entitled "Treaties and Foreign Patents") for more information on timeframes and deadlines for filing foreign patent applications. The guide is available either by contacting the USPTO Contact Center at 800-786-9199, or it can be viewed on the USPTO website at http://www.uspto.gov/web/offices/pac/doc/general/index.html.

For information on preventing theft of your intellectual property (patents, trademarks and copyrights), you may wish to consult the U.S. Government website, http://www.stopfakes.gov. Part of a Department of Commerce initiative, this website includes self-help "toolkits" giving innovators guidance on how to protect intellectual property in specific countries such as China, Korea and Mexico. For questions regarding patent enforcement issues, applicants may call the U.S. Government hotline at 1-866-999-HALT (1-866-999-4158).

LICENSE FOR FOREIGN FILING UNDER

Title 35, United States Code, Section 184

Title 37, Code of Federal Regulations, 5.11 & 5.15

GRANTED

The applicant has been granted a license under 35 U.S.C. 184, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" followed by a date appears on this form. Such licenses are issued in all applications where the conditions for issuance of a license have been met, regardless of whether or not a license may be required as set forth in 37 CFR 5.15. The scope and limitations of this license are set forth in 37 CFR 5.15(a) unless an earlier license has been issued under 37 CFR 5.15(b). The license is subject to revocation upon written notification. The date indicated is the effective date of the license, unless an earlier license of similar scope has been granted under 37 CFR 5.13 or 5.14.

This license is to be retained by the licensee and may be used at any time on or after the effective date

thereof unless it is revoked. This license is automatically transferred to any related applications(s) filed under 37 CFR 1.53(d). This license is not retroactive.

The grant of a license does not in any way lessen the responsibility of a licensee for the security of the subject matter as imposed by any Government contract or the provisions of existing laws relating to espionage and the national security or the export of technical data. Licensees should apprise themselves of current regulations especially with respect to certain countries, of other agencies, particularly the Office of Defense Trade Controls, Department of State (with respect to Arms, Munitions and Implements of War (22 CFR 121-128)); the Bureau of Industry and Security, Department of Commerce (15 CFR parts 730-774); the Office of Foreign AssetsControl, Department of Treasury (31 CFR Parts 500+) and the Department of Energy.

NOT GRANTED

No license under 35 U.S.C. 184 has been granted at this time, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" DOES NOT appear on this form. Applicant may still petition for a license under 37 CFR 5.12, if a license is desired before the expiration of 6 months from the filing date of the application. If 6 months has lapsed from the filing date of this application and the licensee has not received any indication of a secrecy order under 35 U.S.C. 181, the licensee may foreign file the application pursuant to 37 CFR 5.15(b).

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING OR 371(c) DATE	FIRST NAMED APPLICANT	ATTY. DOCKET NO./TITLE
11/396,251	03/30/2006	Timothy Tuttle	2222.775000E

CONFIRMATION NO. 1129

26111
STERNE, KESSLER, GOLDSTEIN & FOX P.L.L.C.
1100 NEW YORK AVENUE, N.W.
WASHINGTON, DC20005

Title: Asynchronous messaging using a node specialization architecture in the dynamic routing network

Publication No. US-2007-0239822-A1
Publication Date: 10/11/2007

NOTICE OF PUBLICATION OF APPLICATION

The above-identified application will be electronically published as a patent application publication pursuant to 37 CFR 1.211, et seq. The patent application publication number and publication date are set forth above.

The publication may be accessed through the USPTO's publically available Searchable Databases via the Internet at www.uspto.gov. The direct link to access the publication is currently http://www.uspto.gov/patft/.

The publication process established by the Office does not provide for mailing a copy of the publication to applicant. A copy of the publication may be obtained from the Office upon payment of the appropriate fee set forth in 37 CFR 1.19(a)(1). Orders for copies of patent application publications are handled by the USPTO's Office of Public Records. The Office of Public Records can be reached by telephone at (703) 308-9726 or (800) 972-6382, by facsimile at (703) 305-8759, by mail addressed to the United States Patent and Trademark Office, Office of Public Records, Alexandria, VA 22313-1450 or via the Internet.

In addition, information on the status of the application, including the mailing date of Office actions and the dates of receipt of correspondence filed in the Office, may also be accessed via the Internet through the Patent Electronic Business Center at www.uspto.gov using the public side of the Patent Application Information and Retrieval (PAIR) system. The direct link to access this status information is currently http://pair.uspto.gov/. Prior to publication, such status information is confidential and may only be obtained by applicant using the private side of PAIR.

Further assistance in electronically accessing the publication, or about PAIR, is available by calling the Patent Electronic Business Center at 1-866-217-9197.

Pre-Grant Publication Division, 703-605-4283

Sterne Kessler Goldstein Fox
ATTORNEYS AT LAW

Robert Greene Sterne
Jorge A. Goldstein
David K.S. Cornwell
Robert W. Esmond
Tracy-Gene G. Durkin
Michele A. Cimbala
Michael B. Ray
Robert E. Sokohl
Eric K. Steffe
Michael Q. Lee
John M. Covert
Robert C. Millonig
Donald J. Featherstone
Timothy J. Shea, Jr
Michael V. Messinger
Judith U. Kim
Jeffrey T. Helvey
Eldora L. Ellison

Donald R. Banowit
Peter A. Jackman
Brian J. Del Buono
Mark Fox Evens
Vincent L. Capuano
Elizabeth J. Haanes
Michael D. Specht
Kevin W. McCabe
Glenn J. Perry
Edward W. Yee
Grant E. Reed
Virgil Lee Beaston
Theodore A. Wood
Joseph S. Ostroff
Jason D. Eisenberg
Tracy L. Muller
Jon E. Wright
LuAnne M. DeSantis

Ann E. Summerfield
Helene C. Carlson
Cynthia M. Bouchez
Timothy A. Doyle
Gaby L. Longsworth
Lori A. Gordon
Laura A. Vogel
Bashir M.S. Ali
Shannon A. Carroll
Anbar F. Khal
Michelle K. Holoubek
Marsha A. Rose
Scott A. Schaller
Lei Zhou
W. Blake Coblentz
James J. Pohl
John T. Haran
Mark W. Rygiel

Michael R. Malek*
Carla Ji-Eun Kim
Doyle A. Siever*
Ulrike Winkler Jenks
Paul A. Calvo
Robert A. Schwartzman
C. Matthew Rozier+
Shameek Ghose
Randall K. Baldwin
Daniel J. Nevivy

Registered Patent Agents•
Karen R. Markowicz
Matthew J. Dowd
Mita Mukherjee
Scott M. Woodhouse
Peter A. Socarras
Jeffrey K. Mills

Danielle L. Letting
Lori Brandes
Steven C. Oppenheimer
Aaron S. Lukas
Gaurav Asthana
Robert E. Bakin
Salvador M. Bezos

Of Counsel
Edward J. Kessler
Kenneth C. Bass III
Marvin C. Guthrie
Christopher P. Wrist

*Admitted only in Maryland
+Admitted only in Virginia
•Practice Limited to Federal Agencies

October 23, 2007

WRITER'S DIRECT NUMBER:
(202) 772-8703
INTERNET ADDRESS:
GJPERRY@SKGF.COM

Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

Art Unit 2194

Attn: Mail Stop Amendment

Re: U.S. Utility Patent Application
Application No. 11/396,251; Filed: March 30, 2006
For: **Asynchronous Messaging Using a Node Specialization Architecture in the Dynamic Routing Network**
Inventors: Tuttle *et al.*
Our Ref: 2222.775000E

Sir:

Transmitted herewith for appropriate action are the following documents:

1. Information Disclosure Statement;

2. Form PTO/SB/08A (3 sheets) listing sixty-two (62) documents (US1-US56 and FP1-FP6);

3. Form PTO/SB/08B (3 sheets) listing twenty-nine (29) documents (NPL1-NPL29); and

4. Copies of cited documents (FP1-FP6 and NPL1-NPL29).

The above-listed documents are filed electronically through EFS-Web.

In the event that extensions of time are necessary to prevent abandonment of this patent application, then such extensions of time are hereby petitioned.

The U.S. Patent and Trademark Office is hereby authorized to charge any fee deficiency, or credit any overpayment, to our Deposit Account No. 19-0036.

Respectfully submitted,

STERNE, KESSLER, GOLDSTEIN & FOX P.L.L.C.

Glenn J. Perry
Attorney for Applicants
Registration No. 28,458

GJP/ahw
Enclosures

DOC#736527_1.DOC

IPR2025-00200 / 722 Patent Page 116 of 1171 EX1002

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	Confirmation No.: 1129
Tuttle *et al.*	Art Unit: 2194
Appl. No.: 11/396,251	Examiner: FATEHI, PARHAM R.
Filed: March 30, 2006	Atty. Docket: 2222.775000E
For: **Asynchronous Messaging Using a Node Specialization Architecture in the Dynamic Routing Network**	

Information Disclosure Statement

Mail Stop Amendment

Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

Sir:

Listed on accompanying IDS Forms are documents that may be considered material to the examination of this application, in compliance with the duty of disclosure requirements of 37 C.F.R. §§ 1.56, 1.97 and 1.98.

Where the publication date of a listed document does not provide a month of publication, the year of publication of the listed document is sufficiently earlier than the effective U.S. filing date and any foreign priority date so that the month of publication is not in issue. Applicants have listed publication dates on the attached IDS Forms based on information presently available to the undersigned. However, the listed publication dates should not be construed as an admission that the information was actually published on the date indicated.

Applicants reserve the right to establish the patentability of the claimed invention over any of the information provided herewith, and/or to prove that this information may

not be prior art, and/or to prove that this information may not be enabling for the teachings purportedly offered.

This statement should not be construed as a representation that a search has been made, or that information more material to the examination of the present patent application does not exist. The Examiner is specifically requested not to rely solely on the material submitted herewith.

Applicants have checked the appropriate boxes below.

☐ 1. Statement under 37 C.F.R. 1.704(d). Each item of information contained in this Information Disclosure Statement was first cited in a communication from a foreign patent office in a counterpart application and this communication was not received by any individual designated in 37 C.F.R. § 1.56(c) more than thirty days prior to the filing of this information disclosure statement.

☒ 2. Filing under 37 C.F.R. § 1.97(b). This Information Disclosure Statement is being filed before the mailing date of a first Office Action on the merits. No statement or fee is required.

☐ 3. Filing under 37 C.F.R. § 1.97(c). This Information Disclosure Statement is being filed more than three months after the U.S. filing date AND after the mailing date of the first Office Action on the merits, but before the mailing date of a Final Rejection, or Notice of Allowance, or an action that otherwise closes prosecution in the application.

☐ a. Statement under 37 C.F.R. § 1.97(e)(1). I hereby state that each item of information contained in this Information Disclosure Statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of this Information Disclosure Statement. 37 C.F.R. § 1.97(e)(1).

☐ b. Statement under 37 C.F.R. § 1.97(e)(2). I hereby state that no item of information in this Information Disclosure Statement was cited in a communication from a foreign patent office in a counterpart foreign application and, to my knowledge after making reasonable inquiry, was known to any individual designated in 37 C.F.R. § 1.56(c) more than three months prior to the filing of this Information Disclosure Statement. 37 C.F.R. § 1.97(e)(2).

☐ c. The required fee is provided through online credit card payment authorization in the amount of $0.00 in payment of the fee under 37 C.F.R. § 1.17(p).

☐ 4. Filing under 37 C.F.R. § 1.97(d) This Information Disclosure Statement is being filed more than three months after the U.S. filing date and after the mailing date of a Final Rejection or Notice of Allowance, but before payment of the Issue Fee. The required fee is provided through online credit card payment authorization in the amount of $0.00 in payment of the fee under 37 C.F.R. § 1.17(p); in addition:

☐ a. Statement under 37 C.F.R. § 1.97(e)(1). I hereby state that each item of information contained in this Information Disclosure Statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of this Information Disclosure Statement. 37 C.F.R. § 1.97(e)(1).

☐ b. Statement under 37 C.F.R. § 1.97(e)(2). I hereby state that no item of information in this Information Disclosure Statement was cited in a communication from a foreign patent office in a counterpart foreign application and, to my knowledge after making reasonable inquiry, was known to any individual designated in 37 C.F.R. § 1.56(c) more than three months prior to the filing of this Information Disclosure Statement. 37 C.F.R. § 1.97(e)(2).

☐ 5. The document(s) was/were cited in a search report by a foreign patent office in a counterpart foreign application. Submission of an English language version of the search report that indicates the degree of relevance found by the foreign office is provided in satisfaction of the requirement for a concise explanation of relevance. 1138 OG 37, 38.

☐ 6. A concise explanation of the relevance of the non-English language document(s) appears below in accordance with 37 C.F.R. § 1.98(a)(3).

☒ 7. Copies of documents FP1-FP6 and NPL1-NPL29 are submitted. However, in accordance with 37 C.F.R. § 1.98(a)(2), no copies of U.S. patents and patent application publications cited on the attached IDS Forms are submitted.

☐ 8. Copies of the documents were cited by or submitted to the Office in an IDS that complies with 37 C.F.R. § 1.98(a)-(c) in Application No._____, filed _____, which is relied upon for an earlier filing date under 35 U.S.C. § 120. Thus, copies of these documents are not attached. 37 C.F.R. § 1.98(d).

☒ 9. It is expected that the examiner will review the prosecution and cited art in the parent application nos. 10/105,018 filed March 21, 2002 (now U.S. Pat. No. 7,051,070) and 10/017,182 filed December 14, 2001 (now U.S. Pat. No. 7,043,525) in accordance with MPEP 2001.06(b) and requested that the examiner will review the prosecution and cited art in the sibling application nos. 09/901,582 filed July 9, 2001, 10/213,269 filed August 5, 2002 (now U.S. Pat. No. 7,127,720), 11/205,263 filed August 15, 2005, 11/205,233 filed August 15, 2005, 11/205,237 filed August 15, 2005, 11/347,802 filed February 3, 2006 (now U.S. Pat. No. 7,277,917), 11/469,325 filed August 31, 2006, 11/515,233 filed August 31, 2006, and 11/515,366 filed August 31, 2006, and indicate in the next communication from the office that the art cited in the earlier prosecution history has been reviewed in connection with the present application.

It is respectfully requested that the Examiner initial and return a copy of the enclosed IDS Forms, and indicate in the official file wrapper of this patent application that the documents have been considered.

Atty. Dkt. No. 2222.775000E

The U.S. Patent and Trademark Office is hereby authorized to charge any fee deficiency, or credit any overpayment, to our Deposit Account No. 19-0036.

Respectfully submitted,

STERNE, KESSLER, GOLDSTEIN & FOX P.L.L.C.

Glenn J. Perry
Attorney for Applicants
Registration No. 28,458

Date: Oct. 23, 2007

1100 New York Avenue, N.W.
Washington, D.C. 20005-3934
(202) 371-2600

DOC#736526_1.DOC

IPR2025-00200 / 722 Patent Page 122 of 1171 EX1002

Equivalent of Form PTO/SB/08A (04-07)
Approved for use through 09/30/2007. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Substitute for form 1449/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Use as many sheets as necessary)

Sheet 1 of 3

Complete if Known	
Application Number	11/396,251
Filing Date	March 30, 2006
First Named Inventor	Timothy Tuttle
Art Unit	2194
Examiner Name	FATEHI, PARHAM R.
Attorney Docket Number	2222.775000E

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.[1]	Document Number Number-Kind Code[2] (If Known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
	US1	5,535,335	07-09-1996	Cox et al.	
	US2	5,706,516	01-06-1998	Chang et al.	
	US3	5,822,543	10-13-1998	Dunn et al.	
	US4	5,878,420	03-02-1999	de la Salle	
	US5	5,886,643	03-23-1999	Diebboll et al.	
	US6	5,964,839	10-12-1999	Johnson et al.	
	US7	5,974,457	10-26-1999	Waclawsky et al.	
	US8	6,018,619	01-25-2000	Allard et al.	
	US9	6,029,175	02-22-2000	Chow et al.	
	US10	6,052,447	04-18-2000	Golden et al.	
	US11	6,055,493	04-25-2000	Ries et al.	
	US12	6,112,240	08-29-2000	Pogue et al.	
	US13	6,173,406 B1	01-09-2001	Wang et al.	
	US14	6,233,600 B1	05-15-2001	Salas et al.	
	US15	6,240,451 B1	05-29-2001	Campbell et al.	
	US16	6,253,167 B1	06-26-2001	Matsuda et al.	
	US17	6,308,209 B1	10-23-2001	Lecheler	
	US18	6,324,587 B1	11-27-2001	Trenbeath et al.	
	US19	6,363,421 B2	03-26-2002	Barker et al.	
	US20	6,405,245 B1	06-11-2002	Burson et al.	

FOREIGN PATENT DOCUMENTS

Examiner Initials*	Cite No.[1]	Foreign Patent Document Country Code[3] Number[4] Kind Code[5] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear	T[6]
	FP1	EP 0 733 983 A2	09-25-1996	AT&T Corp.		
	FP2	EP 0 749 081 A1	12-18-1996	Pointcast Inc.		
	FP3	EP 0 889 421 A1	01-07-1999	Yahoo Inc.		
	FP4	WO 97/16796 A1	05-09-1997	Herz Frederick et al.		
	FP5	WO 01/63837 A2	08-30-2001	Navic Systems Inc.		
	FP6	WO 2005/046184 A1	05-19-2005	Dotworlds Ltd.		
	FP7					
	FP8					
	FP9					

Examiner Signature		Date Considered	

DOC#736524_1.DOC

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant. [1] Applicant's unique citation designation number (optional). [2] See Kinds Codes of USPTO Patent Documents at www.uspto.gov or MPEP 901.04. [3] Enter Office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] Applicant is to place a check mark here if English language Translation is attached.

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 2 hours to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 (1-800-786-9199) and select option 2.

Equivalent of Form PTO/SB/08A (04-07)
Approved for use through 09/30/2007. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Substitute for form 1449/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Use as many sheets as necessary)

Sheet 2 of 3

Complete if Known	
Application Number	11/396,251
Filing Date	March 30, 2006
First Named Inventor	Timothy Tuttle
Art Unit	2194
Examiner Name	FATEHI, PARHAM R.
Attorney Docket Number	2222.775000E

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.[1]	Document Number Number-Kind Code[2] (If Known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
	US21	6,418,448 B1	07-09-2002	Sarkar	
	US22	6,418,467 B1	07-09-2002	Schweitzer et al.	
	US23	6,446,257 B1	09-03-2002	Pradhan et al.	
	US24	6,484,143 B1	11-19-2002	Swildens et al.	
	US25	6,502,131 B1	12-31-2002	Vaid et al.	
	US26	6,539,427 B1	03-25-2003	Natarajan et al.	
	US27	6,560,611 B1	05-06-2003	Nine et al.	
	US28	6,567,411 B2	05-20-2003	Dahlen	
	US29	6,577,328 B2	06-10-2003	Matsuda et al.	
	US30	6,606,643 B1	08-12-2003	Michael et al.	
	US31	6,609,138 B1	08-19-2003	Merriam	
	US32	6,654,804 B1	11-25-2003	Fleming	
	US33	6,658,652 B1	12-02-2003	Alexander et al.	
	US34	6,687,729 B1	02-03-2004	Sievert et al.	
	US35	6,691,165 B1	02-10-2004	Bruck et al.	
	US36	6,751,663 B1	06-15-2004	Farrell et al.	
	US37	6,789,115 B1	09-07-2004	Singer et al.	
	US38	6,792,458 B1	09-14-2004	Muret et al.	
	US39	6,829,642 B1	12-07-2004	Didier et al.	
	US40	6,836,886 B2	12-28-2004	Tuerke et al.	

FOREIGN PATENT DOCUMENTS

Examiner Initials*	Cite No.[1]	Foreign Patent Document Country Code[3] Number[4] Kind Code[5] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear	T[6]
	FP10					
	FP11					
	FP12					
	FP13					
	FP14					
	FP15					
	FP16					
	FP17					
	FP18					

Examiner Signature		Date Considered	

DOC#736524_1.DOC

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant. [1] Applicant's unique citation designation number (optional). [2] See Kinds Codes of USPTO Patent Documents at www.uspto.gov or MPEP 901.04. [3] Enter Office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] Applicant is to place a check mark here if English language Translation is attached.

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 2 hours to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 (1-800-786-9199) and select option 2.

Equivalent of Form PTO/SB/08A (04-07)
Approved for use through 09/30/2007. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Substitute for form 1449/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Use as many sheets as necessary)

Sheet 3 of 3

Complete if Known	
Application Number	11/396,251
Filing Date	March 30, 2006
First Named Inventor	Timothy Tuttle
Art Unit	2194
Examiner Name	FATEHI, PARHAM R.
Attorney Docket Number	2222.775000E

U.S. PATENT DOCUMENTS

Examiner Initials*	Cite No.[1]	Document Number Number-Kind Code[2] (If Known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
	US41	6,871,346 B1	03-22-2005	Kumbalimutt et al.	
	US42	6,970,924 B1	11-29-2005	Chu et al.	
	US43	7,020,082 B2	03-28-2006	Bhagavath et al.	
	US44	7,043,525 B2	05-09-2006	Tuttle et al.	
	US45	7,051,070 B2	05-23-2006	Tuttle et al.	
	US46	7,127,720 B2	10-24-2006	Cano et al.	
	US47	7,207,043 B2	04-17-2007	Blythe et al.	
	US48	7,277,917 B2	10-02-2007	Tuttle et al.	
	US49	2003/0120817 A1	06-26-2003	Ott et al.	
	US50	2005/0033841 A1	02-10-2005	McCarthy et al.	
	US51	2006/0117318 A1	06-01-2006	Rumelhart et al.	
	US52	2006/0041681 A1	02-23-2006	Rumelhart	
	US53	2006/0075279 A1	04-06-2006	Cameros et al.	
	US54	2007/0033293 A1	02-08-2007	Rumelhart	
	US55	2007/0050519 A1	03-01-2007	Cano et al.	
	US56	2007/0061811 A1	03-15-2007	Rumelhart et al.	
	US57				
	US58				
	US59				
	US60				

FOREIGN PATENT DOCUMENTS

Examiner Initials*	Cite No.[1]	Foreign Patent Document Country Code[3] Number[4] Kind Code[5] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear	T[6]
	FP19					
	FP20					
	FP21					
	FP22					
	FP23					
	FP24					
	FP25					
	FP26					
	FP27					

Examiner Signature		Date Considered	

DOC#736524_1.DOC

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant. [1] Applicant's unique citation designation number (optional). [2] See Kinds Codes of USPTO Patent Documents at www.uspto.gov or MPEP 901.04. [3] Enter Office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] Applicant is to place a check mark here if English language Translation is attached.

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 2 hours to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 (1-800-786-9199) and select option 2.

Equivalent of Form PTO/SB/08B (04-07)
Approved for use through 09/30/2007. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Substitute for form 1449/PTO				***Complete if Known***	
INFORMATION DISCLOSURE STATEMENT BY APPLICANT *(Use as many sheets as necessary)*				*Application Number*	11/396,251
				Filing Date	March 30, 2006
				First Named Inventor	Timothy Tuttle
				Art Unit	2194
				Examiner Name	FATEHI, PARHAM R.
Sheet	1	of	3	*Attorney Docket Number*	2222.775000E

NON PATENT LITERATURE DOCUMENTS			
Examiner Initials*	Cite No.[1]	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published	T[2]
	NPL1	Franklin et al., "Dissemination-Based Information Systems," IEEE Data Engineering Bulletin, vol. 19, No. 3, Sep. 1996, 9 pages.	
	NPL2	Nagami et al., "Toshiba's Flow Attribute Notification Protocol (FANP) Specification," Apr. 1997, RFC 2129, Internet RFC/STD/FYI/BCP Archives [online], [retrieved on May 16, 2002]. Retrived from the Internet: <landfield.com/rfcs/rfc2129.html>, 16 pages.	
	NPL3	Strom et al., "Gryphon: An Information Flow Based Approach to Message Brokering," International Symposium on Software Reliability Engineering '98, 1998, 2 pages.	
	NPL4	Sturman et al., "Reflection in the Gryphon Message Brokering System," Reflection Workshop of the 13.sup.th ACM Conference on Object Oriented Program Systems, Languages and Applications (OOPSLA '98), 1998, 5 pages.	
	NPL5	International DOI Foundation, "Introduction to the Digital Object Identifier," [online]. Apr. 1998 [retrieved on May 16, 2002]. Retrieved from the Internet: <doi.org/introduction.html.>, 4 pages.	
	NPL6	Aksoy et al., "Research in Data Broadcast and Dissemination", Proc. 1st Int'l Conf. on Advanced Multimedia Content Processing, Osaka University, Osaka, Japan, Nov., 1998.	
	NPL7	Banavar et al., "An Efficient Multicast Protocol for Content-Based Publish-Subscribe Systems," Proc. of the 19th International Conference on Distributed Computing Systems, 1999, 9 pages.	
	NPL8	Banavar et al., "Information Flow Based Event Distribution Middleware," Proceedings of the 1999 ICDCS Workshop on Electronic Commerce and Web-Based Applications, 1999, 8 pages.	
	NPL9	Aguilera et al., "Matching Events in a Content-based Subscription System," Eighteenth ACM Symposium on Principles of Distributed Computing (PODC '99), Atlanta, GA, May 4-6, 1999, 9 pages.	
	NPL10	Banavar et al., "A Case for Message Oriented Middleware," Distributed Computing, 13.sup.th International Symposium, Bratislava, Slavak Republic, Sep. 27-29, 1999, 18 pages.	

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] Applicant's unique citation designation number (optional). [2] Applicant is to place a check mark here if English language Translation is attached.

This collection of information is required by 37 CFR 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 2 hours to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: **Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Equivalent of Form PTO/SB/08B (04-07)
Approved for use through 09/30/2007. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Substitute for form 1449/PTO				*Complete if Known*	
INFORMATION DISCLOSURE STATEMENT BY APPLICANT *(Use as many sheets as necessary)*				*Application Number*	11/396,251
				Filing Date	March 30, 2006
				First Named Inventor	Timothy Tuttle
				Art Unit	2194
				Examiner Name	FATEHI, PARHAM R.
Sheet	2	of	3	*Attorney Docket Number*	2222.775000E

NON PATENT LITERATURE DOCUMENTS			
Examiner Initials*	Cite No.[1]	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume number, publisher, city and/or country where published	T[2]
	NPL11	Aguilera et al., "Efficient Atomic Broadcast Using Deterministic Merge," Symposium on Principles of Distributed Computing, 2000, 10 pages.	
	NPL12	Opyrchal et al., "Exploiting IP Multicast in Content-Based Publish-Subscribe Systems," Proceedings of the IFIP/ACM International Conference on Distributed Systems Platforms (Middleware 2000), Apr. 2000, 23 pages.	
	NPL13	Caplin Systems Ltd., White Paper "Real Time Text Protocol (RTTP)," Version 1.1, Sep. 2000, 11 pages.	
	NPL14	Reuters, "Reuters Market Data Systems and the Trading Solutions Architecture," Version 1.0, Jan. 12, 2001, 51 pages.	
	NPL15	Ramamrithan et al., "Dissemination of Dynamic Data on the Internet," [online]. Powerpoint Presentation, Spring 2001, [retrieved on Feb. 6, 2002], 5 pages. Retrieved from the Internet <.cs.umbc.edu/courses/graduate/CMSC691T/spring.sub.--2001/rlist/amit.ppt>.	
	NPL16	ComputerLetter, vol. 17, No. 23, Jul. 16, 2001, pp. 1-8.	
	NPL17	ComputerLetter, vol. 17, No. 31, Sep. 24, 2001, pp. 1-6.	
	NPL18	ComputerLetter, vol. 17, No. 35, Nov. 5, 2001, pp. 1-6.	
	NPL19	Tuttle et al., "Upstream Delivery of Information in a Digital Network", U.S. Appl. No. 09/901,582 filed July 9, 2001.	
	NPL20	"Repackaging the Net", ComputerLetter, vol. 17, No. 35, Nov. 5, 2001, pp. 1-5.	

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] Applicant's unique citation designation number (optional). [2] Applicant is to place a check mark here if English language Translation is attached.

This collection of information is required by 37 CFR 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 2 hours to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

Equivalent of Form PTO/SB/08B (04-07)
Approved for use through 09/30/2007. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Substitute for form 1449/PTO				*Complete if Known*	
INFORMATION DISCLOSURE STATEMENT BY APPLICANT *(Use as many sheets as necessary)*				*Application Number*	11/396,251
				Filing Date	March 30, 2006
				First Named Inventor	Timothy Tuttle
				Art Unit	2194
				Examiner Name	FATEHI, PARHAM R.
Sheet	3	of	3	*Attorney Docket Number*	2222.775000E

NON PATENT LITERATURE DOCUMENTS			
Examiner Initials*	Cite No.[1]	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume number, publisher, city and/or country where published	T[2]
	NPL21	"Reckoning with IP", ComputerLetter, vol. 17, No. 37, Nov. 19, 2001, pp. 1-6.	
	NPL22	"Persistence Counts", ComputerLetter, vol. 17, No. 23, Jul. 16, 2001, pp. 1, 5-7.	
	NPL23	Zhao et al.; "A Workflow-centric Study of Organizational Knowledge Distribution;" Proceedings of the 33rd Hawaii International Conference on System Sciences;" 2000; pp.1-10; IEEE.	
	NPL24	Gribble, et al.; "The Ninja Architecture for Robust Internet-scale Systems and Services;" Computer Networks; 2001; pp.473-497; Volume 35; Elsevier Science B.V.	
	NPL25	Carmona, David; "Programming the Thread Pool in the .NET Framework"; 'Online' June 2002, pp.1-17, XP002357234; retrieved on 12/01/2005 from the Internet: URL:http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dndotnet/hmtl/progthrepool.asp>, pages 1-17.	
	NPL26	Welsh, Matthew D.; "An Architecture for Highly Concurrent, Well-Conditioned Internet Services"; URL: http://www.eecs.harvard.edu/{mdw/papers/mdw-phdthesis/pdf>, 2005, pages 48-54, 101, and 113-114.	
	NPL27	Written Opinion of the International Searching Authority for International Application No. PCT/US2005/029162, recorded January 17, 2006, 7 pages.	
	NPL28	Written Opinion of the International Searching Authority for International Application No. PCT/US2005/029021, recorded December 14, 2005, 10 pages.	
	NPL29	Written Opinion of the International Searching Authority for International Application No. PCT/US2005/029158, recorded January 25, 2006, 7 pages.	
	NPL30		

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] Applicant's unique citation designation number (optional). [2] Applicant is to place a check mark here if English language Translation is attached.

This collection of information is required by 37 CFR 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 2 hours to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: **Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 733 983 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
25.09.1996 Bulletin 1996/39

(51) Int Cl.[6]: **G06F 17/30**

(21) Application number: **96301706.6**

(22) Date of filing: **13.03.1996**

(84) Designated Contracting States:
DE ES FR GB

(30) Priority: **24.03.1995 US 409579**

(71) Applicant: **AT&T IPM Corp.**
Coral Gables, Florida 33134 (US)

(72) Inventors:
- **Farber, James M.**
  Rumson, New Jersey 07760 (US)
- **Huber, Kenneth M.**
  Red Bank, New Jersey 07701 (US)
- **Hanson, Bruce Lowell**
  Little Silver, New Jersey 07739 (US)
- **Morehead, David Richard**
  Morganville, New Jersey 07751 (US)
- **Roesler, Marina L.**
  Westfield, New Jersey 07090 (US)

(74) Representative: **Johnston, Kenneth Graham et al**
Lucent Technologies (UK) Ltd,
5 Mornington Road
Woodford Green Essex, IG8 OTU (GB)

(54) Personalized real time information display

(57) Information "feeds" received from multiple information sources are aggregated, re-formatted and stored in a personalized real time information display system. When the display system receives a request from a user having a personal computer or other similar display device, the system is arranged to respond with information needed to display a screen saver that includes personalized, up to date information, such as traffic, weather and sports, that is of interest to that particular user. Each user has a personalized user profile indicating (a) the categories or types of information that the user desires to receive, such as sports information, weather, investment advisories, and son on, and (b) parameters that specify, for that user, the exact information desired to be received in each category. If the user s personal computer display has remained inactive for a predetermined period of time, the screen saver is activated, causing the personal computer to access the service node, and after interchange of prestored user identification and password data, retrieve information, obtained by the system from a variety of sources, based on the user's individual profile. The retrieved information is displayed on the screen without interrupting the screen saver function, and the displayed personalized data is thereafter updated in the same manner, at user-specified time intervals.

EP 0 733 983 A2

Printed by Jouve, 75001 PARIS (FR)

Description

Field of the Invention

This invention relates generally to the provision and display of information, and, in particular, to a system and method for personalizing a screen saver display on a personal computer (PC) or other visual display device, so that the screen saver display is periodically changed to include current up-to-date information related to items that interest the user of the PC.

Background of the Invention

A user having a display device such as a personal computer (PC) (or a similar visual display device such as a screen telephone), often will employ a screen saver to display pictorial information on the device when the device has remained inactive for a predetermined period of time. While the screen saver can be personalized to cause various time-varying patterns to be displayed, thereby avoiding burn-in on the screen surface, the display does not itself convey useful information to the user.

Summary of the Invention

In accordance with the present invention, information "feeds" received from multiple information sources are aggregated, re-formatted and stored in a service node that may be disposed in, and thus be a part of, a telecommunications network, or, alternatively, that may be outside of the network, but that is accessed through the telecommunications network. When the service node receives a request from a user having a personal computer or other similar display device, the service node is arranged to respond with information needed to display a screen saver that includes personalized, up to date, information, such as traffic, weather and sports, that is of interest to that particular user.

When a user initially arranges to use the present invention, a personalized user profile is established, indicating (a) the categories or types of information that the user desires to receive, such as sports information, weather, investment advisories, and so on, and (b) parameters that specify, for that user, the exact information desired to be received n each category. If the user's personal computer display has remained inactive for a predetermined period of time, the screen saver is activated, causing the personal computer to access the service node. The screen saver application transmits user identification and password data, previously stored in the screen saver application, to the service node. The PC requests information from the service node, which information was obtained by the service node from a variety of sources, based on the user's individual profile. The service node formats the information for display on the user's device, the information is transmitted from the service node to the PC, and the retrieved information is displayed on the device without interrupting the screen saver function. The displayed personalized data is thereafter updated in the same manner, by arranging the personal computer to make periodic requests for information. Accordingly, the information that is displayed as a screen saver is always "current".

In accordance with one aspect of the present invention, the service node includes several clients in an information feed interface that, in effect, insulates information providers from direct connection to users, and assures that any necessary formatting and "publishing" changes required for display by the users' information obtained from providers is digested and stored/arranged in individual databases within the service node, so that the information is available for efficient user access. In accordance with another aspect of the present invention, pre-established user profiles are maintained in the service node, in order to enable retrieval and display of the desired information. However, the system is arranged to make intelligent selection of "default" profiles, based on factors that can be ascertained even about a new user, such as the user's location from which a request is initiated. This allows the system to customize information even for users who have not pre-specified a profile.

Brief Description of the Drawings

The present invention will be more fully appreciated by consideration of the following detailed description which should be read in light of the accompanying drawing in which:

Fig. 1 is a block diagram illustrating the overall arrangement of the present invention and the environment in which the invention operates;
Fig. 2 is a block diagram illustrating the arrangement of database 135 within server 130 of Fig. 1;
Fig. 3 is a flow diagram illustrating one embodiment of the process used to provide customized, up to date information to a personal computer end user device; and
Fig. 4 is an example of an "At-A-Glance" screen saver display which integrates personalized information obtained from a variety of user-specified services using the present invention.

Detailed Description

Referring first to Fig. 1, there is shown a block diagram illustrating the overall arrangement of the present invention and the environment in which the invention operates. A service node, designated generally as 120, is arranged to provide a plurality of users having personal computers 101 (or other terminals with visual display capability), with personalized, up to date information, such as traffic, weather and sports, that is of interest to each particular user, that can be displayed as a screen saver

IPR2025-00200 / 722 Patent Page 130 of 1171 EX1002

when the personal computer remains idle for a predetermined period of time. Representative multimedia terminals other than the personal computers 101 shown in Fig. 1 can include a screen phone, such as a Picturephone 2500 available from AT&T Corp., a television equipped with a set top box (such as the TV Information Center available from AT&T Corp.) providing an interactive capability, or a personal digital assistant, such as a MagicLink device available from Sony.

Personal computers 101 are connected to service node 120 through a telecommunications network 110, which may include switches and other elements in local exchange carrier networks as well as interexchange carrier networks. In some embodiments of the present invention, service node 120 may be disposed within telecommunications network 110, and thus be part of either a local exchange carrier network or an interexchange carrier network.

As shown in Fig. 1, service node 120 includes a server 130 and an information feed interface 140. Server 130 is connected to personal computers 101 through telecommunications network 110, and provides the information needed to provide a display on the PC's of the type illustrated in Fig. 4 that is described in more detail below. Information feed interface 140 is connected to server 130 as well as to a plurality of information providers that may be at different remote locations. As explained below, these information providers may include, for example, a weather provider 150, a traffic provider 152 and a financial information provider 154. Commercial TCP/IP networking software may be used to provide the communications interface between server 130 and information feed interface 140.

Server 130, which may be a workstation such as the Sparc 20 available from Sun MicroSystems, is arranged to perform various processes, including parsing the information received from information providers and storing the parsed data in a database, as well as other functions described below, using a microprocessor 139 operating under the control of programs stored in a file system 137. Microprocessor 139 has access to information stored in a database 135, which stores information of the types illustrated in Fig. 2, discussed below. Information and instructions are communicated between microprocessor 139 and personal computers 101 using a first communications manager 131, which acts as a communications interface and protocol converter. Communications manager 131 includes usage profile management module 136, that provides system usage information needed for administration and billing purposes for each user. Communications manager 131 also includes database query routines 138 that are used to gain access to user data and stored information pertinent to each user. A second communication manager 133 provides a communications interface between server 130 and information feed interface 140. Communication manager 133 includes parsing routines 134 in order to break-down the information received from information feed interfaces 140 into addressable data fields (e.g., date, time, location, traffic report) and then store the data fields in the appropriate portion of database 135. Operations, administration, and maintenance (OA&M) routines 132 are also provided in server 130. These routines operate, administer, and maintain service node 120. This allows a system operator to monitor the status of service node 120, perform software/data backups and restorations, resolve operating alarms, provide usage data required for billing, and so on.

Communications manager 131 implements a data link communications protocol and an applications protocol for communication with personal computers 101 or any other multimedia terminal that is employed as an end user device. The data link communications protocol defines how "bits" are packaged, while the applications protocol defines how to interpret the bits (i.e., what is the information). This includes provision of user applications that format user application generated data so that it can be understood and displayed by the personal computer or other end user device. Formatting may be accomplished using a language such as HyperText Markup Language (HTML) commonly used by the World Wide Web.

Server 130 also includes a file system 137, which contains information and programs necessary to control and operate service node 120.

Information feed interface 140 includes a plurality of clients 142-144, each of which provide an interface to a corresponding one of the information providers 150, 152 and 154. Thus, weather client 143 is connected to and interfaces with weather provider 150, traffic client 142 is connected to and interfaces with traffic provider 152, and financial client 144 is connected to and interfaces with financial information provider 154. Each of the clients 142-144 include local processors that may be configured in one of several ways. First, the clients may periodically connect to the associated information provider and download information that is available from the provider. In this configuration, the clients can also answer calls from an information provider and receive information sent by the provider. In an alternative configuration, the clients may maintain a continuous communications link to the information provider, and either download information (service node initiated) or receive information (provider initiated). Clients 142-144 may be implemented in commercially available personal computers using commercially available communications software, such as Procomm-Plus. In either event, the client arrangement in effect insulates information providers from direct connection to users, and assures that any necessary formatting and "publishing" changes required for display by the user's display system are made.

It is to be noted here that the arrangement of 110 shown in Fig. 1 is illustrative only, and that the precise connection between the personal computers 101 or other multimedia terminal equipment used by any user and

IPR2025-00200 / 722 Patent Page 131 of 1171 EX1002

service node 120 will also depend upon the communications facility that is available to interconnect that user with service node 120. What is essential to the present invention is that information "feeds" received from multiple information sources are aggregated, re-formatted and stored in the service node, for later access by a user. For example, if a particular user has an ISDN terminal, the user may be connected directly to server 130 via an ISDN line.

Referring now to Fig. 2, there is shown a block diagram illustrating the arrangement of database 135 within server 130 of Fig. 1. Generally speaking, server 130 includes a plurality of individual databases such as traffic database 171, weather database 172, and financial database 173, containing information obtained from the various information providers. Other information databases 175 may also be provided, depending upon the types of information that are available from information providers and thus can be offered to users of the system. Database 135 also contains other databases such as a user profile database 174 and usage records 176. User profile database 174 contains information for each user of the system, specifying (a) the categories or types of information services that are to be provided to that user, and (b) for those information services, the parameters that are associated with the desired information. For example, a first user may desire traffic, financial and sports information, a second user may desire weather and news information, and a third user may desire traffic, news and weather. For each of these three users, the detailed information desired may be different. Thus, the first user may desire traffic information for certain roadways, financial information for certain securities, and sports information for particular teams. The second user may desire to obtain some of the same types of information, but the details will be different. In this example, the second user may desire local news and weather for City A, while the third user may desire news and weather for a different location, City B. The database management system in server 130 may be implemented using a commercially available relational database management system, such as Informix®.

Referring now to Fig. 3, there is shown a flow diagram illustrating one embodiment of the process used to provide customized, up to date information to a personal computer end user device 101 that accesses service node 120. Each block in Fig. 3 identifies the operations performed by personal computer 101 and by service node 120 to provide the functionality contemplated by the present invention. The steps in the process assure that when the system receives a request from a screen saver application running on a personal computer or other similar display device, the system can respond with information needed to display a screen saver that includes personalized, up to date information, such as traffic, weather and sports, that is of interest to that particular user.

The process beings in step 300, in which a determination is made as to whether the screen saver is activated, such that it requires information from the present invention. If a NO result is obtained, indicating that the screen saver is not activated, the process loops, such that step 300 is repeated until a YES result is obtained. This loop will generally continue until a user-specified period of PC inactivity has elapsed (e.g., five minutes).

When a YES result occurs in step 300, the screen saver is activated and the process for retrieving current up-to-date data is initiated by proceeding to step 301, in which a communication session is established between personal computer 101 and service node 120. This step may also be triggered after the screen saver has been active for a user-specified period of time (e.g., fifteen minutes). In both cases personal computer 101 is arranged to generate a query or transmit a message to service node 120, typically by dialing a predefined telephone number. Service node 120 is arranged to answer the call and send a "challenge" message to personal computer 101. In turn, personal computer 101 responds with an acknowledgment message, which is sent back to service node 120.

Once a communication session is established in step 301, a determination is made in step 303 as to whether two-way communications are enabled between a personal computer 101 and service node 120. If not, an error condition exists, and the process proceeds to step 305, in which personal computer 101 terminates the connection to service node 120 and displays an appropriate error message to the user. Concurrently, service node 120 terminates the partially established communication session. At this point, the PC screen saver may attempt to re-initiate communications, as by repeating the process of step 301.

If a YES result is obtained in step 303, the process proceeds to step 307, in which the user identification number (ID) is validated, or another validation process is performed. In this step, personal computer 101 sends the prestored user ID to service node 120, whereupon service node 120 checks the ID against stored information in database 135 to determine its validity. Depending upon the outcome, service node 120 sends a valid ID or invalid ID message to personal computer 101.

It is to be noted here that information is stored in database 135 when a user initially arranges to use the display system of the present invention. At that time, a personalized user profile is established, indicating (a) the categories or types of information that the user desires to receive, such as sports information, weather, investment advisories, and so on, and (b) parameters that specify, for that user, the exact information desired to be received in each category. The profile may be created in a number of ways, such as by communicating with an interactive voice platform such as a Conversant® voice response system available from AT&T Corp., or by using a PC application to select desired service options.

If the ID transmitted by the PC screen saver is not valid, the process continues with a NO result in step 309

IPR2025-00200 / 722 Patent Page 132 of 1171 EX1002

and proceeds to step 311, where the error condition is dealt with. In that step, both personal computer 101 and service node 120 terminate the existing communications session, and personal computer 101 displays an appropriate message to the user..

If the ID transmitted by the PC screen is valid, the process continues with a YES result in step 309 and proceeds to step 313, in which the user's personalized information is retrieved from data base 135. In this step, personal computer 101 requests the personalized information from service node 120, service node 120 accesses the user's profile, and queries data base 135 for the specific information desired by this user, based upon the profile information.

Next, in step 315, the user's personalized information is appropriately formatted. In this step, the information retrieved in step 313 is actually transmitted from service node 120 to personal computer 101.

In step 317, a determination is then made as to whether an error condition has occurred with respect to the retrieval, formatting and transmission of information to the user. If a YES result is obtained, indicating an error, the process proceeds to step 319, which, like step 311, results in both personal computer 101 and service node 120 terminating the existing communications session, and personal computer 101 displaying an appropriate message to the user.

Following step 317, if an error condition has not occurred, the process proceeds to step 321, in which the communication session between service node 120 and personal computer 101 is terminated. In this step, personal computer 101 sends a terminate session message to service node 120, whereupon service node 120 and personal computer 101 both terminate the communication session. In step 323, the information retrieved in step 313 and formatted in step 315 is displayed on the user's personal computer 101, in the context of the PC screen saver without interrupting the screen saver function.

It is important to note that the screen saver function remains active throughout all the steps illustrated in Fig. 3 and that all these steps take place automatically, that is without user input.

Referring now to Fig. 4, there is shown an example of an "At-A-Glance" screen saver display which integrates personalized information obtained from a variety of user-specified services using the present invention. The display integrates information from a variety of user-specified services, and presents the user with valuable information without being required to navigate through a series of menus. As shown, a series of screen areas 401-406 "float" or randomly move throughout the display area 400, preferably without interfering with or covering each other. Each of the display areas contain different information, relating to topics selected by the user, including, in this example, weather (areas 402 and 403), traffic (area 401), news (area 405), financial (area 404) and other information (area 406), in this case pertaining to communications messages waiting for the user. The information displayed in areas 401-406 is up-to-date, since it was recently obtained from service node 120, and it is customized, in that the information within each of the areas 401-406 was selected by the user and set forth in his or her profile. As the data is updated, the screen saver remains active and the screen saver motion is not interrupted. Numerous variations of the screen saver user presentation are possible.

Various modifications and adaptations of the present invention will be apparent to persons skilled in the art. For example, instead of pre-stored profiles for individual users, the system can be arranged to make intelligent selection of "default" profiles, based on factors that can be ascertained even about a new user, such as the user's location from which a request is initiated. Thus, a user calling service node 120 from a telephone located in New Jersey will, unless different information is requested, automatically receive information about traffic and weather conditions in New Jersey. This allows the system to customize information even for users who have not stored or pre-specified a profile. Accordingly, it is intended that the invention be limited only by the appended claims.

Claims

1. A system for displaying personalized time-varying information on a visual display device, comprising

   a service node including a database,
   an information feed interface in said service node arranged to obtain information from one or more information providers and to store said information in said database,
   a server in said service node arranged to maintain a profile for each user of said system, said profile indicating particular stored information in said database to be provided to said each user, and
   a communications manager for downloading said particular stored information to said visual display device.

2. The invention defined in claim 1 wherein said visual display device is arranged to display said particular stored information as a screen saver, whenever the screen saver is activated.

3. The invention defined in claim 1 wherein said service node is disposed in a telecommunications network.

4. The invention defined in claim 1 wherein said communications manager is arranged to perform said downloading in response to a command from said visual display device.

IPR2025-00200 / 722 Patent Page 133 of 1171 EX1002

5. The invention defined in claim 1 wherein said visual display device is a personal computer.

6. A method for displaying personalized time varying information as a screen saver on a visual display device, comprising the steps of

obtaining information from one or more information providers and storing said information in said database,
maintaining a profile for each user of said system, said profile indicating particular stored information in said database to be provided to said each user,
downloading said particular stored information to said visual display device, and
displaying said particular stored information on said visual display device whenever the screen saver is activated.

IPR2025-00200 / 722 Patent Page 134 of 1171 EX1002

FIG. 1

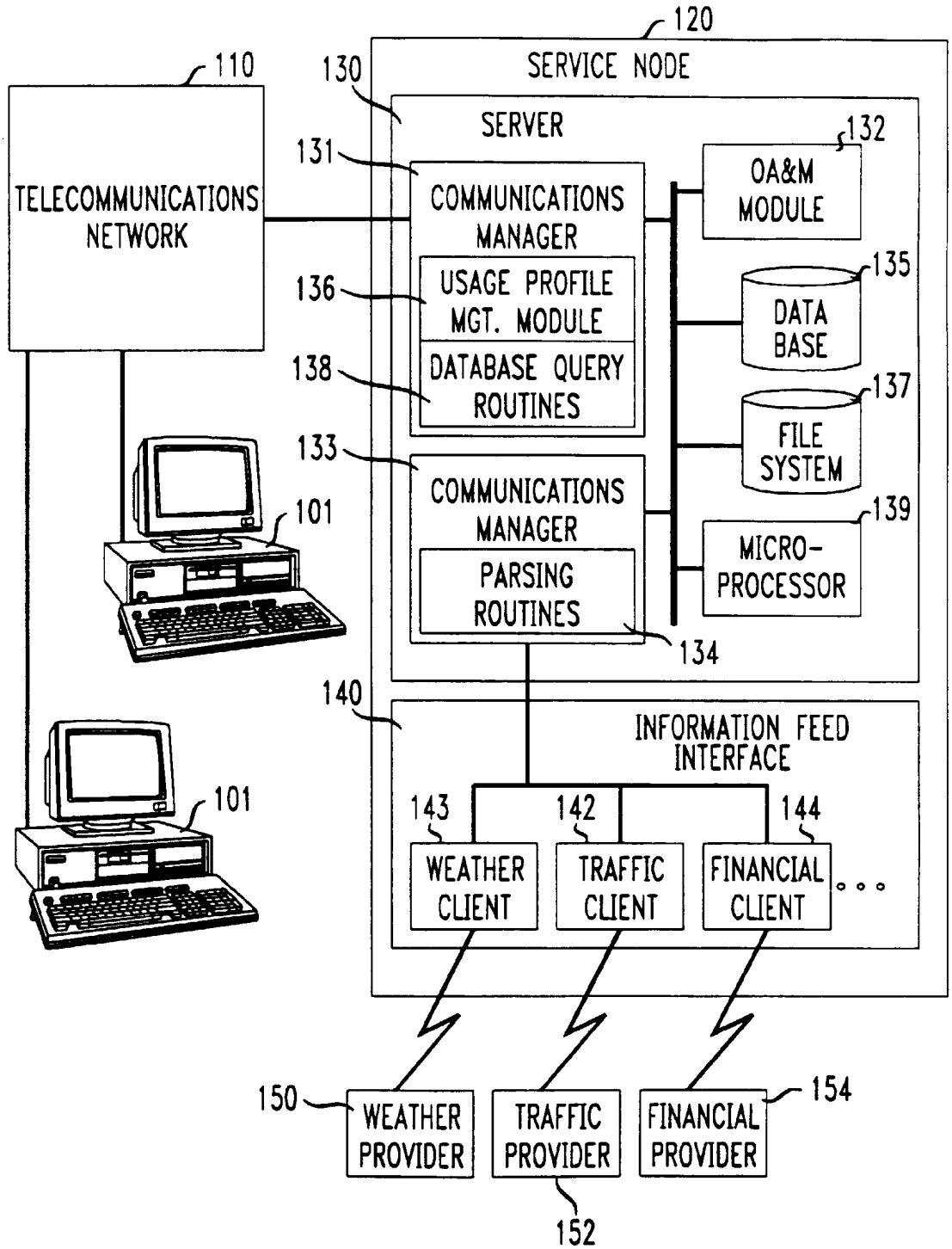
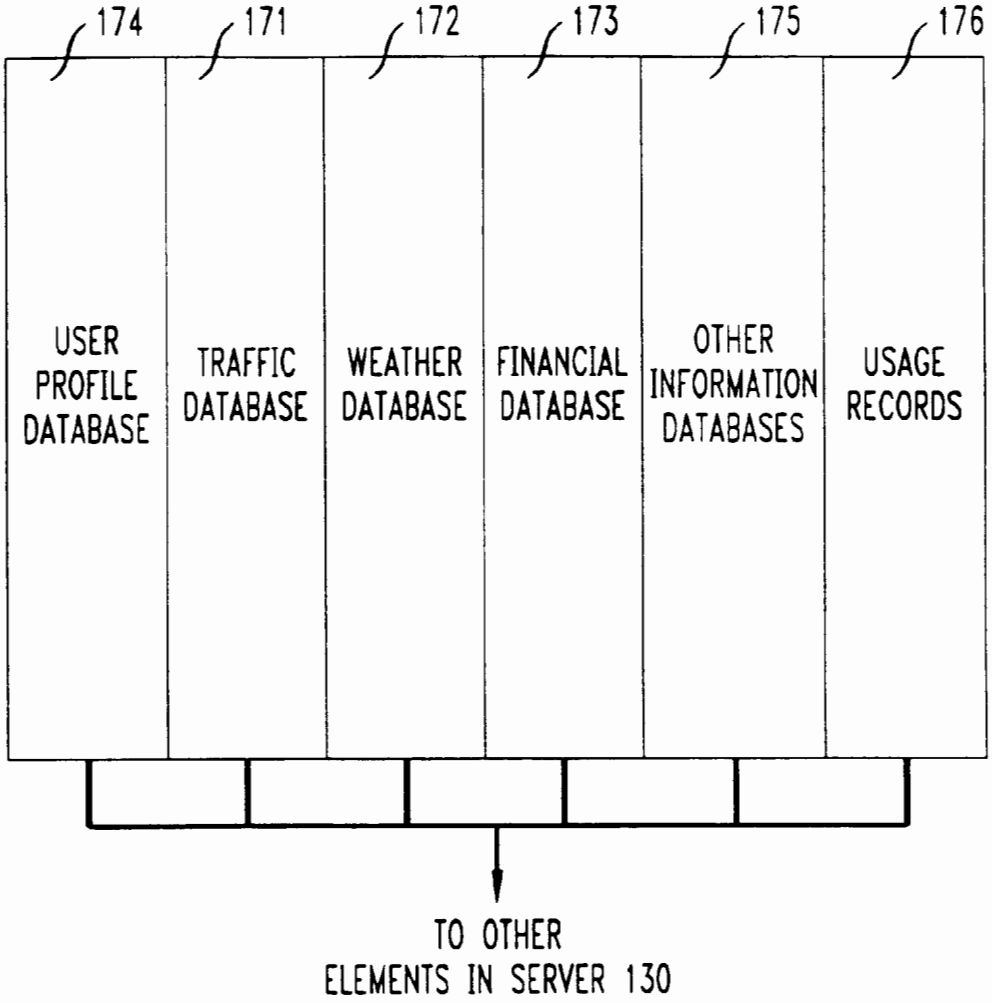

120
SERVICE NODE
110
130
SERVER
131
TELECOMMUNICATIONS NETWORK
COMMUNICATIONS MANAGER
132
OA&M MODULE
136
USAGE PROFILE MGT. MODULE
135
DATA BASE
138
DATABASE QUERY ROUTINES
137
FILE SYSTEM
133
COMMUNICATIONS MANAGER
101
139
MICRO-PROCESSOR
PARSING ROUTINES
134
140
INFORMATION FEED INTERFACE
101
143
142
144
WEATHER CLIENT
TRAFFIC CLIENT
FINANCIAL CLIENT
150
WEATHER PROVIDER
TRAFFIC PROVIDER
FINANCIAL PROVIDER
154
152

FIG. 2


174
171
172
173
175
176
USER PROFILE DATABASE
TRAFFIC DATABASE
WEATHER DATABASE
FINANCIAL DATABASE
OTHER INFORMATION DATABASES
USAGE RECORDS
TO OTHER ELEMENTS IN SERVER 130

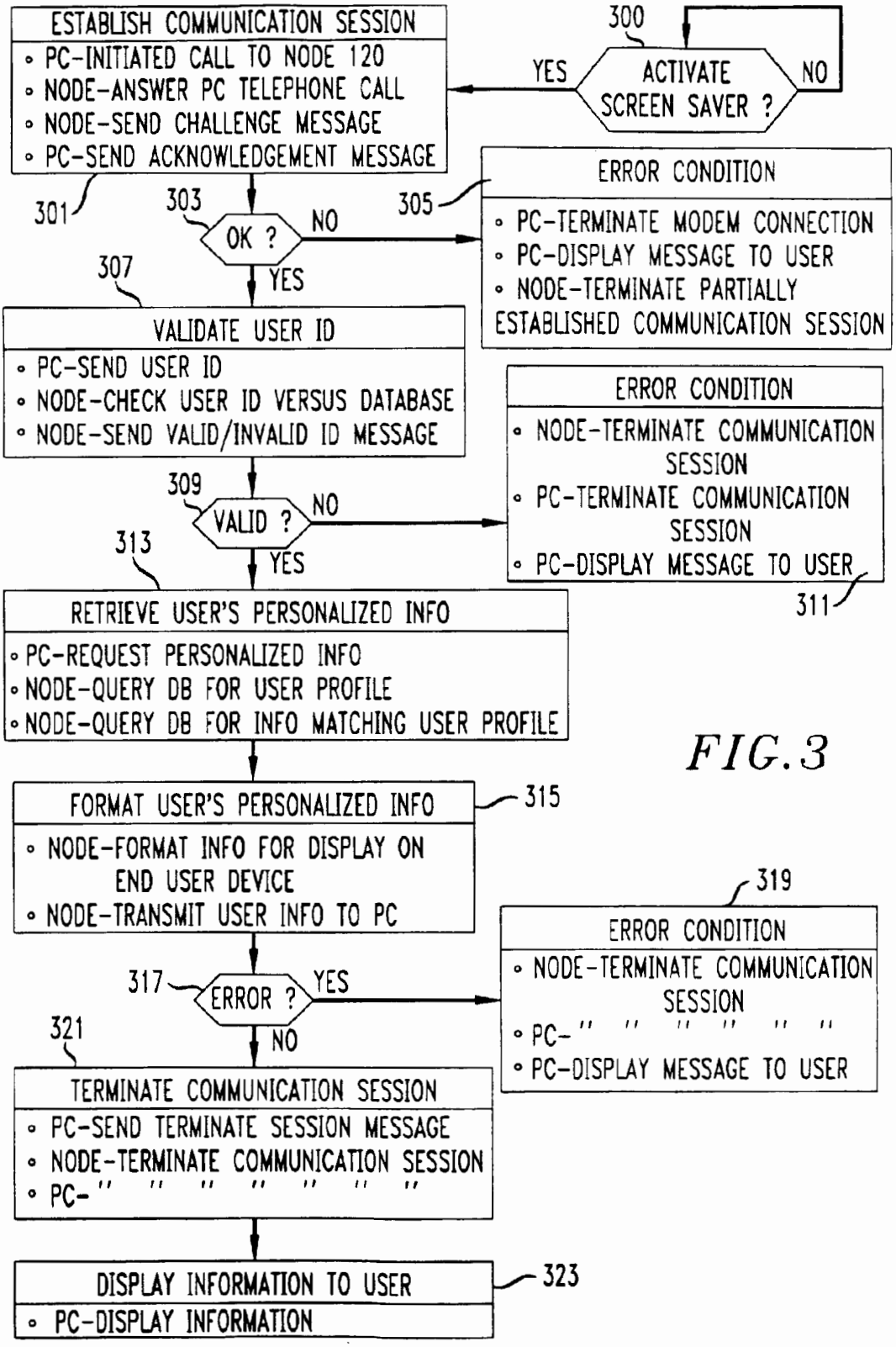
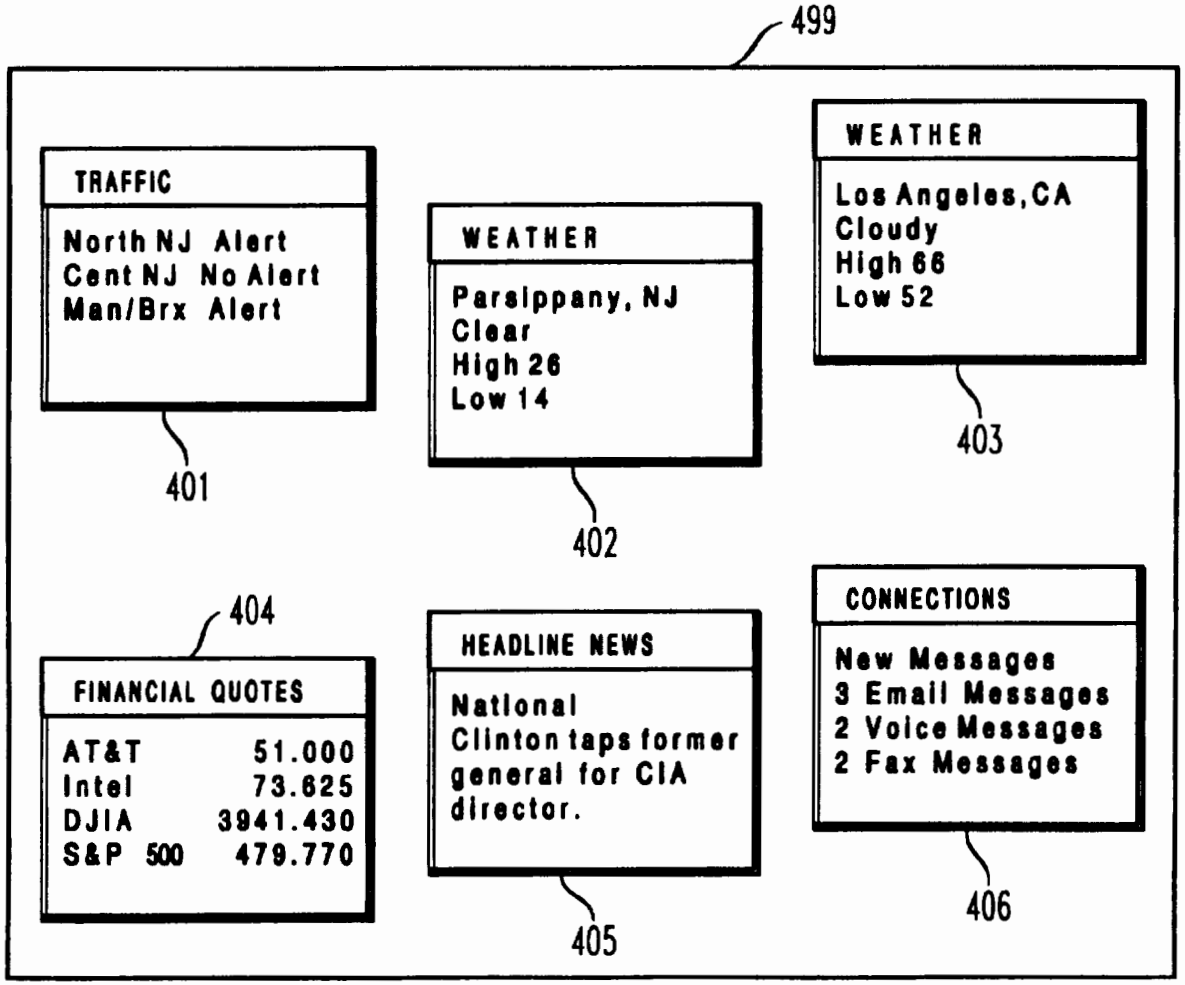

ESTABLISH COMMUNICATION SESSION
• PC-INITIATED CALL TO NODE 120
• NODE-ANSWER PC TELEPHONE CALL
• NODE-SEND CHALLENGE MESSAGE
• PC-SEND ACKNOWLEDGEMENT MESSAGE
301
300
ACTIVATE SCREEN SAVER ?
YES
NO
303
OK ?
NO
YES
305
ERROR CONDITION
• PC-TERMINATE MODEM CONNECTION
• PC-DISPLAY MESSAGE TO USER
• NODE-TERMINATE PARTIALLY ESTABLISHED COMMUNICATION SESSION
307
VALIDATE USER ID
• PC-SEND USER ID
• NODE-CHECK USER ID VERSUS DATABASE
• NODE-SEND VALID/INVALID ID MESSAGE
309
VALID ?
NO
YES
ERROR CONDITION
• NODE-TERMINATE COMMUNICATION SESSION
• PC-TERMINATE COMMUNICATION SESSION
• PC-DISPLAY MESSAGE TO USER
311
313
RETRIEVE USER'S PERSONALIZED INFO
• PC-REQUEST PERSONALIZED INFO
• NODE-QUERY DB FOR USER PROFILE
• NODE-QUERY DB FOR INFO MATCHING USER PROFILE
315
FORMAT USER'S PERSONALIZED INFO
• NODE-FORMAT INFO FOR DISPLAY ON END USER DEVICE
• NODE-TRANSMIT USER INFO TO PC
319
ERROR CONDITION
• NODE-TERMINATE COMMUNICATION SESSION
• PC-
• PC-DISPLAY MESSAGE TO USER
317
ERROR ?
YES
NO
321
TERMINATE COMMUNICATION SESSION
• PC-SEND TERMINATE SESSION MESSAGE
• NODE-TERMINATE COMMUNICATION SESSION
• PC-
323
DISPLAY INFORMATION TO USER
• PC-DISPLAY INFORMATION

FIG.3

FIG. 4


499
TRAFFIC
North NJ Alert
Cent NJ No Alert
Man/Brx Alert
401
WEATHER
Parsippany, NJ
Clear
High 26
Low 14
402
WEATHER
Los Angeles, CA
Cloudy
High 66
Low 52
403
404
FINANCIAL QUOTES
AT&T 51.000
Intel 73.625
DJIA 3941.430
S&P 500 479.770
HEADLINE NEWS
National
Clinton taps former general for CIA director.
405
CONNECTIONS
New Messages
3 Email Messages
2 Voice Messages
2 Fax Messages
406

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 749 081 A1

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
18.12.1996 Bulletin 1996/51

(51) Int. Cl.[6]: **G06F 17/60**

(21) Application number: **96303727.0**

(22) Date of filing: **24.05.1996**

(84) Designated Contracting States:
AT DE FR GB IE NL

(30) Priority: **12.06.1995 US 489591**

(71) Applicant: **POINTCAST INC.**
Cupertino, California 95014 (US)

(72) Inventors:
- **Reilly, James P.**
  San Francisco, California 95014 (US)
- **Hassett, Gregory P.**
  Cupertino, California 95014 (US)

(74) Representative: **Cross, Rupert Edward Blount et al**
BOULT WADE TENNANT
27 Furnival Street
London EC4A 1PQ (GB)

(54) Information and advertising distribution system and method

(57) In summary, the present invention is an information and advertising distribution system. A data server stores and updates a database of information items and advertisements. The information items and advertisements are each categorized so that each has an associated information category. Workstations remotely located from the data server each include a display device, a communication interlace for receiving at least a subset of the information items and advertisements in the data server's database and local memory for storing the information items and advertisements received from the data server. An information administrator in each workstation establishes communication with the data server from time to time so as to update the information items and advertisements stored in local memory with at least a subset of the information items and advertisements stored by the data server. An information display controller in each workstation displays on the workstation's display device at least a subset of the information items and advertisements stored in local memory when the workstation meets predefined idleness criteria. At least a subset of the workstations include a profiler for storing subscriber profile data. The subscriber profile data represents subscriber information viewing preferences, indicating information categories for which the subscriber does and does not want to view information items. The information display controller includes a filter for excluding from the information items displayed on the display device those information items inconsistent with the subscriber profile data.

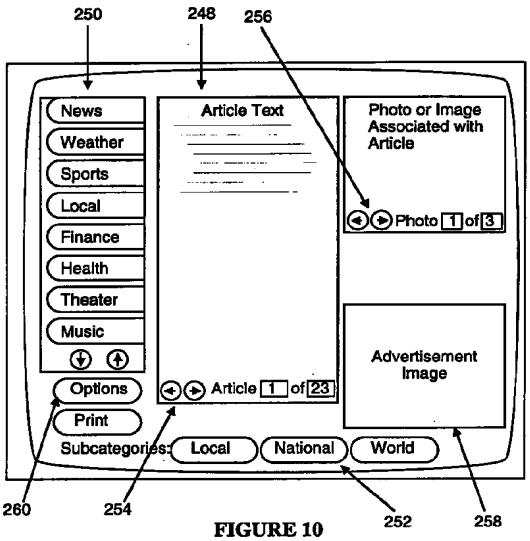


FIGURE 10

EP 0 749 081 A1

Printed by Rank Xerox (UK) Business Services
2.13.9/3.4

Description

The present invention relates generally to computer based information distribution systems, and particularly to a system for distributing to a set of subscribers' computers information matching each subscriber's interests as well as advertising, and for distributing the information and advertising to each subscriber's computer during time periods in which the subscriber's computer is otherwise inactive.

BACKGROUND OF THE INVENTION

The use of advertising revenues to pay for information dissemination is well established in domains such as television and radio in which end users are tuned to a continuous signal over a lengthy period of time. In such systems, due to the continuous nature of the signal being monitored by the end users, the end users are sufficiently similar to a "captive audience" that many or most end users remain tuned to the same signal even when the main program to which they are listening or viewing is interrupted by advertisements.

Another example of advertising mixed with information dissemination is the use of scrolled text at the bottom of a television of computer screen, where the main program occupies most of the end user's visual field and a smaller portion is occupied by advertisements and the like on a "scroll bar" or similar visual device along the periphery of the screen. In some contexts, such as cable television channels that display a "stock ticker tape," this relationship is reversed: the information portion of the screen occupies a small part of the screen, such as horizontally scrolling image region at the top or bottom of the display and the remainder of the screen is occupied by advertisements, "infomercials" and the like.

Yet another example of mixing advertisements with information dissemination are newspapers and magazines.

Most, and perhaps all such examples of mixing advertisements with information content are based on systems in which the end user has actively elected to view or listen to a program or to otherwise receive information. Furthermore, in virtually all such systems or media, the juxtaposition or placement of advertisements and information content is explicitly programmed or determined by human beings working as "editors" or in a similar content and/or presentation editing capacity.

Up until the present, distributing information via the Internet or other publicly accessible computer communication networks has been largely unsupported by advertising revenues due to the lack of good mechanisms for mixing advertising and information content in such a way as to be acceptable to both end users and advertisers. There are, of course, some exceptions where advertising/content mixtures from other contexts, such as newspapers and television, have been simply replicated on the Internet. For instance, some newspapers have been "published" at least in part on the Internet, and include advertisements along with information content. In fact, some newspapers sell advertising space on an associated World Wide Web (WWW) site, which often includes extensive listings of certain types of advertisements such as real estate advertisements, personal advertisements, and so on. Similarly, the scroll bar type advertisement at the bottom of a computer screen is based on similar advertising techniques used in cable television and other television contexts.

There are also examples of computer programs which contain advertisements. In all such examples known to the inventors, the advertisements are either permanently embedded in the computer programs or reside permanently with computer programs such that they cannot be easily updated.

Embodiments of the present invention address a problem prevalent in electronic information distribution systems. In particular, "on line" newspapers and magazines are notoriously difficult and tedious to read. Graphics and animation and full motion video, all techniques widely used in television news programs, require substantial data transmission bandwidth. Such data transmission is expensive both in terms of communications bandwidth (capacity) and time. In non-computer publishing such as printed magazines and newspapers, graphics are often used to make reading less difficult and tedious. In television the majority of information is delivered with movement (animation), although graphics are also often used.

The use of large bandwidth data transmissions is not economically practical in the context of data dissemination via the Internet and other computer networks, although the cost of such data transmissions will undoubtedly continue to decrease. As a result, graphics and animation have typically received relatively little use in computer network based information dissemination systems.

Embodiments of the present invention mix advertising and information content dissemination in a manner unlike the examples mentioned above.

Embodiments of the present invention disseminate information and advertisements to subscribers' computers in a system where the information and advertisements are automatically displayed when the subscriber's computer is on but meets predefined idleness criteria. For example, the predefined idleness criteria could be the failure to receive any input for a period of at least five minutes.

Embodiments of the present invention automatically update each subscriber's local database of news stories at least once per day, and preferably multiple times per day so as to present subscribers with timely information.

IPR2025-00200 / 722 Patent Page 140 of 1171 EX1002

Embodiments of the present invention present news stories and advertisements in a dynamic and easy to read manner.

Embodiments of the present invention categorize news stories and advertisements, and to display advertisements associated with each category at the same time that new stories associated with same category are displayed, thereby providing a "targeted" audience for advertisers.

Embodiments of the present invention provide each subscriber with the ability to set up and change a user profile indicating categories and subcategories of topics which are of interest and not of interest to the subscriber, and to select the news stories displayed on the subscriber's computer accordingly.

Embodiments of the present invention divide news stories into at least two portions, a preliminary portion and a secondary portion, where the preliminary portions of news stories are automatically displayed during idle periods, and the secondary portions are displayed only upon subscriber request.

SUMMARY OF THE INVENTION

In summary, the present invention is an information and advertising distribution system. A information server stores and updates a database of information items and advertisements. The information items and advertisements are each categorized so that each has an associated information category. Workstations remotely located from the information server each include a display device, a communication interface for receiving at least a subset of the information items and advertisements in the information server's database and local memory for storing the information items and advertisements received from the information server. An information administrator in each workstation establishes communication with the information server from time to time so as to update the information items and advertisements stored in local memory with at least a subset of the information items and advertisements stored by the information server. An information display controller in each workstation displays on the workstation's display device at least a subset of the information items and advertisements stored in local memory when the workstation meets predefined idleness criteria.

At least a some of the workstations include a profiler for storing subscriber profile data. The subscriber profile data represents subscriber information viewing preferences, indicating information categories for which a subscriber associated with the workstation does and does not want to view information items. The information display controller includes a filter for excluding from the information items displayed on the display device those information items inconsistent with the subscriber profile data.

BRIEF DESCRIPTION OF THE DRAWINGS

Additional objects and features of embodiments of the invention will be more readily apparent from the following detailed description and appended claims when taken in conjunction with the drawings, in which:

Figure 1 is a block diagram of an information and advertising distribution system in accordance with an embodiments of the present invention.

Figure 2 is a block diagram of a subscriber's computer in the information and advertising distribution system of Figure 1.

Figure 3 schematically depicts the procedures and data structures in a set of category managers.

Figure 4 schematically depicts a user profile data structure stored in a subscriber's computer to store status and configuration information for a particular subscriber and workstation.

Figure 5 schematically depicts the dialog box used to define the user profile for one information category.

Figure 6 schematically depicts a display generated on a subscriber's display device using the screen saver procedure in a preferred embodiment of the present invention.

Figures 7A and 7B schematically depicts the dialog box used to define a display script and the resulting data structure.

Figures 8 and 9 schematically depict data structures stored in a subscriber's computer to indicate advertisements and news stories available for display in various information categories.

Figure 10 schematically depicts a display generated on a subscriber's display device using a data viewer procedure in a preferred embodiment of the present invention.

Figure 11 depicts the relationships between various processes in the information server.

Figure 12 is a flow chart depicting the procedure for updating the local database and software modules of a subscriber's computer.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

Referring to Figure 1, there is shown a computer based information and advertising distribution system 100 having many client computers 102 and at least one information server computer 104. Client computers are often called "subscribers' computers" in the present document, and the terms "subscriber computer" and "client computer" will be used synonymously. In many instances, a set of subscribers 102 will be located within a common local area network (LAN) 106, and are connected to a LAN server 108.

In the preferred embodiment, each subscriber's computer 102 is connected to the information server 104 via the Internet 119 for a small fraction of each day. Other forms of electronic communication connections, including private wide area networks similar to CompuServe, America OnLine or Prodigy, can be used to connect subscribers' computers to the information server 104 in alternate embodiments of the invention.

While most client computers are desktop computers, such as IBM compatible computers and Macintosh computers, virtually any type of computer can be a client computer so long as it can support the "screen saver" mode of operation of the preferred embodiment.

Information Server

The information server 104 includes a central processing unit 110, primary memory 112 (i.e., fast random access memory) and secondary memory 114 (typically disk storage), a user interface 116, an Internet interface 118 for communication with the client computers 102 via the Internet 119, and one or more news wire interfaces 120 for receiving news feeds from information transmission services such as the AP news feed, the DOW news feed and various sports news feeds. An information editor 130 is used, typically under the direction of a person using the user interface 116, to select news stories received from the new feeds and to edit and format the news stories into a form suitable to dissemination to subscribers' computers using the present invention. The selected and edited news stories 132 are stored in an information database 134 in the information server 104.

In the preferred embodiment, the information editor 130 is used to assign each news story to an information category and, where appropriate, to also assign the news story to one or more sub-categories. The information editor maintains a list of the currently defined categories and sub-categories. The category list can be updated by the personnel operating the information server, typically to add and delete special new categories associated with major news events such as a famous trial or event which generates many news stories. The category to which each news story is assigned is represented in one or more Data Access Tables 136.

The information editor 130 is also used to divide most news stories into two components or portions: a primary component or portion and a secondary component or portion. The primary component is what is displayed on a subscriber's workstation when the subscriber's workstation is turned on but has been idle, while the secondary component is what is displayed, along with the primary components only upon a subscriber's request. For instance, as will be described below, there are number of ways in which a subscriber can request the display of the "full text" of a news item (which may include photographs and the like). For convenience, the primary component of each news story is sometimes herein called the "headline", even though it will often contain more information than just the headline of the news item, and the secondary component of each news story will sometimes be called the "body."

Advertisements 138 are also stored in the information database 134 and each advertisement is assigned to at least one of the predefined information categories. Each advertisement is displayed on subscribers' workstations simultaneously with news items assigned to the same category as the advertisement. When an advertisement is assigned to multiple categories, it is treated in most respects as several advertisements each assigned to one category, except that only one copy of the advertisement is actually stored.

Next, the information database in the server computer includes a set of images 140 used during the display of news items and advertisements. For instance, different "wallpaper" or background images may be useful when displaying news items in various types of information categories. As an example, the images 140 include three fixed images for indicating that the stock market has risen, fallen or stayed largely unchanged. Then, depending on what has happened to the stock market on any particular day, information concerning the amount of change in the stock market during the relevant time period, and sometimes other associated information, is superimposed on a selected one of those fixed images. Other images stored in the information database include various "actors" that can be moved around the display with the news items when the system is in screen saver mode.

The information database 134 also stores a set of "display scripts" 142. A script controls the display of news items and advertisements, typically displaying a selected number of news items and one advertisement for a period of 30 sec-

IPR2025-00200 / 722 Patent Page 142 of 1171 EX1002

onds. A script determines the number of news items displayed, determines the positions of the news items and advertisement on the display, determines any movement of the news items around the displayed image, and determines what background image or images are displayed in conjunction with the news items.

An important concept associated with the present embodiment is that constantly varying the presentation of news items and advertisements, through the use of a rotating set of scripts, makes it easier for subscribers to read the news headlines and advertisements being presented. In a preferred embodiment, at least two distinct scripts, and preferably three or more distinct scripts are provided for most information categories, with a total of at least ten different scripts being used. Most scripts can be used with multiple categories of news items. The procedure for defining display scripts and the associated data structure are described below with reference to Figure 7A.

The information database 134 also stores software modules 144 for downloading to subscribers' computers. The information administration management procedures and information viewing procedures in subscribers' computers will need updating and upgrading from time to time. The new versions of these software procedures are stored in the information server's information database 134 for downloading into the computers of subscribers at the same time that the information items or advertisements in the subscriber computers' information database 184 is updated. Since numerous types of subscriber computers are supported, the server's information database 134 will typically store a set of updated software modules for each of the supported types of computers.

Finally, the information database 134 includes advertising display statistics 148 and news item display statistics 149. The display statistics are collected from the subscribers' computers when the subscribers' computers call in for updated news stories and the like. Advertising display statistics indicate how many times each advertisement has been displayed on subscribers' computers. In a preferred embodiment, display statistics for each advertisement are divided into a display count for displaying during data viewer usage, a display count for other display instances, and an indication of each advertisement the user has interacted with, such as by "clicking" on the advertisement to connect to the advertiser's World Wide Web page. News item display statistics 149 concern how much time the subscriber spent viewing each non-advertising item in the data viewer as well as the amount of time the screen saver was active for each information category.

Other procedures stored in the information server's secondary memory are a router procedure 150, application server procedures 152, and data server procedures 154. The utility of these procedures is explained below with reference to Figures 8 and 9.

Subscriber's Workstation

Figure 2 is a schematic representation of the subscriber's workstation or computer 102 that is not connected to the information server 104 via a LAN server. For subscribers' workstations connected to the information server 104 via a LAN server 108, Figure 2 is representative of the LAN server, but the display device used by each such subscriber's computer to view news items and advertisements is part of the subscriber's workstation rather than the LAN server 108.

The subscriber workstation 102 includes a central processing unit 170, primary memory 172 (i.e., fast random access memory) and secondary memory 174 (typically disk storage), a user interface 176, and an Internet interface 178 for communication with the information server 104 via the Internet 119. In this document, whenever the phrase "clicking on X" is used, that phrase means a subscriber selecting the X image on a display device by positioning a pointer image over the X image, using the subscriber computer's mouse or trackball device, and then depressing a button or key to indicate selection of the X image.

An administration manager 180 schedules and controls all communications with the information server 104. The administration manager 180 includes a connection scheduler 181 that initiates the execution of a connection manager 182 that handles communications with the information server as well as the integration of information and software procedures received from the information server into the information and software procedures stored in the client computer.

The workstation's secondary memory is used to store a local information database 184 that includes news stories 183, advertisements 188, images 190 and display scripts 192. In each case the workstation's secondary memory stores at least a subset of the corresponding items stored in the information server 104. The amount of information stored in the workstation's secondary memory depends on the amount of secondary memory available for storing such information, as well as a user profile 194 for the subscriber that indicates which categories and subcategories of news stories are of interest to the subscriber.

Data Access Tables 186, which are discussed in more detail below with reference to Figures 5 and 6, are used to access news stories, advertisements and display scripts associated with each of the categories of news items that are to be displayed on the subscriber's workstation.

Screen Saver and Viewer Procedures 200 are a set of procedures for controlling the display of news stories and advertisements. These procedures include a main screen saver procedure 201, category managers 202, an animation engine 204, a profiler 206, a data viewer 208 and an advertisement display statistics generator 210.

Each of the category managers 202 is a collection of programs and data associated with particular information cat-

IPR2025-00200 / 722 Patent Page 143 of 1171 EX1002

egories. In the preferred embodiment there is a separate category manager for each information category, although in some cases it may be more efficient to use the same category manager for two or more information categories.

Referring to Figure 3, each category manager 202 includes a category profiler 202A, a category profile data structure 202B, one or more display drivers 202C for viewing items in the corresponding information category with the data viewer, a sprite generator 202D generating images displayed by the screen saver procedure, and an update manager 202E.

The category profiler in each category presents a category profile dialog to the subscriber to determine the subscriber's interest in receiving information relating to particular subcategories. Subcategories may relate to specific companies, geographic regions, specific sports and sports teams, and so on, depending on the category. The result of the decisions made by the subscriber during the category profile dialog is stored as a category profile data structure.

The update manager in each category handles the process of updating the local information database with new items from the information server for that information category as well as the deletion of all items and the rebuilding of the portion of the data access tables used to control access to the information items, advertisements and display scripts in that information category.

The display drivers in each category manager are customized to generate images specifically needed in the corresponding categories. For instance, in the category manager for the sports category, the display driver includes instructions for generating a simulated scoreboard which is automatically updated every few seconds to show a sequence of game scores or contest outcomes in various sporting events. In another example, the display driver for the weather category includes instructions specifically designed for efficiently displaying weather maps and other weather information.

Referring once again to Figure 2, the animation engine 204 interprets a currently selected display script and controls the display of a selected set of news stories and an advertisement in accordance with the instructions in the currently selected display script.

The profiler 206 is actually a set of procedures that define and update the subscriber's user profile 194. Referring to Figure 4, in the preferred embodiment, the user profile 194 includes:

a subscriber identifier 212;
a connection password 213 used in conjunction with the subscriber identifier when connecting to the information server to identify the calling computer as a registered subscriber;
subscriber hardware and software configuration information 214 that identifies for the information server hardware and software information needed to determine the type of software and image files that are compatible with the subscriber's computer;
a connection schedule 215 that specifies to the connection scheduler 181 within the administrative manager 180 how often the subscriber's computer should connect to the information server 104 to update its information database 184;
category and subcategory preferences information 216 that identifies categories and subcategories of news stories that the subscriber does not want to view, as well as a list of "special categories" of news stories of special interest to the subscriber which override any categories noted as not being of interest to the subscriber;
timestamps 217a - 217c indicating the time of the last updates to the subscriber computer's locally stored set of news stories, advertisements and administrative files (including scripts, images and software modules);
advertising and news item display statistics 218;
screen saver information 219 indicating the last displayed information category and the last displayed advertisement and news items in each information category are stored in a portion of the user profile 194 not transmitted to the information server; and
a screen saver exit mode indicator 220, indicating what actions cause the screen saver procedure to terminate and what actions cause the data viewer 208 to be executed.

The default connection schedule is for the subscriber's computer to initiate a connection to the information server once during the middle of the night (e.g., a randomly selected time between 11 p.m. and 7 a.m. local time) for an "administrative update," and once every four hours during the rest of the day for "news story updates." During the administrative update connection, the set of advertisements, scripts and images in the subscriber computer's local information database are updated as necessary, and any software upgrades are also downloaded onto the subscriber's computer. During both "administrative update" and "news story update" connections, the news stories in the subscriber computer's local information database are updated. At the option of the information server's system operator, script and/or software updates can be made during "news story update" connections, especially when a malfunction has been detected in previously distributed scripts or software.

In one preferred embodiment, the profiler 206 can be used to specify a connection schedule other than the default schedule. For instance, if the subscriber's computer is typically turned off at night, the administrative update connection may be scheduled to occur (A) during the subscriber's typical lunch time, or (B) once per day when the subscriber's computer has not received any user input for a specified minimum period of time (e.g., ten minutes) that indicates the

IPR2025-00200 / 722 Patent Page 144 of 1171 EX1002

subscriber is away from his/her computer.

The downloading of advertisements (which are typically images), fixed images used by display scripts, and software modules is preferably performed during the night or long periods of user inactivity because images and software modules are typically much larger than the news items, which are primarily text data. Images, including advertisements, and software modules are compressed using well known data compression techniques to make the download transmissions as time efficient as possible. Even so, downloading images is a time consuming process. For instance, downloading two high resolution advertisement images having pixel sizes of, say, 400 x 300 pixels each, even when using data compression, will typically take over two minutes using conventional 14.4K baud modems. By way of contrast, downloading a dozen news stories and corresponding database base update instructions will typically take less than fifteen seconds of connection time using conventional 14.4K baud modems. Therefore, updating the local database's set of news items can be accomplished relatively unobtrusively even while the subscriber is using his/her workstation, while updates to the advertisements and fixed images in the local database take longer and are therefore more intrusive.

It is noted that the secondary portions of news items can also include images, such as photographs that accompany the text of a news story. The transmission of such news story images can significantly increase the amount of connection time required for news item updates, and thus most news stories in the preferred embodiment do not use images, and every effort is made to transmit those news stories that have images to subscribers' computers during the overnight administrative update rather than during the daytime news item updates.

The data viewer 208 is a program for viewing news items that the subscriber specifically wants to read. The data viewer 208 can be executed at the subscriber's explicit command, and can also be launched from the screen saver if the user indicates he/she wants to read a news story shown in the screen saver display. This is explained in more detail below.

The display statistics generator 210 keeps tracks of how many times each advertisement in the local information database has been displayed since the last time advertisement display statistics have been transferred to the information server. The display statistics generator 210 also keeps track of how many times each news item has been displayed in the same time period. These display statistics are stored in the user profile 194 at 218. In the preferred embodiment, the advertisement display statistics, and news items display statistics, are transferred to the information server once per day during a connection also used to update the subscriber computer's information database. In alternate embodiments, the advertisement display statistics could be transferred more often (e.g., every time the subscriber's computer connects to the information server) or less often (e.g., once per week).

Category Profiler Dialog

As mentioned earlier, each of the category managers includes a profiler procedure for defining the subscriber's interest in receiving news items within each information category. An example of the profile definition dialog generated by a category profiler, for the Sports category, is shown in Figure 5. In this example, the Sports Definition Profile dialog box 222 includes, on the left side, a scroll box 223 in which the user can select and deselect subcategories of sports information by clicking on boxes next to the listed subcategories. A "Select All" button in the dialog box can be used (i.e., by clicking the subscriber computer's mouse or trackball device on the image of the box) to select all subcategories, and a "Deselect All" button can be used to indicate that the subscriber does not want to receive any news items for the Sports category. For each subcategory, either an "include only" or an "exclude" filter (but not both) can be defined where the user types in key words to be used to select (for the include only) or deselect news items within that subcategory. For instance, if the subscriber types in the words "49ers, Rams" in the box for the include only filter for the "football news" subcategory, only news items using either of those words will be shown to the subscriber.

The category manager profile procedure generates a category profile data structure 202B that represents the subcategories of interest to the subscriber as well as any associated filters that have been defined.

Display Script Definition Procedure

Referring to Figure 6, there is shown in outline form a snapshot of typical display generated by the screen saver procedure of the present embodiment. On this particular exemplary display are shown three news story "headlines" 230a-232c and one advertisement image 232. Each of the headlines 230 is an image representing the text of the "primary component " of a news items, as explained above. While the image shown in Figure 6 appears static, in the preferred embodiment the display script that controls the display of the headlines and advertisement can and most often does contain instructions for continuously moving the headline images around the screen.

The display scripts also mix fixed images with the headline images to create varied and interesting displays. In one example of a display script, cartoon characters appear to move the headlines around. In another example of display script, the background behind and surrounding the headlines is a sequence of fixed images such as pictures of peaceful landscapes, while the headlines gently float around the portions of the display not occupied by the advertisement image 232.

IPR2025-00200 / 722 Patent Page 145 of 1171 EX1002

Referring to Figure 7A, the preferred embodiment provides an easy to use dialog 234 for display script definition. A display script consists of definitions for two or more actors, plus an optional definition of a background image, called the wallpaper image. Each "actor" represents a sprite, which is a displayable image, that can move around the screen and whose size can vary dynamically. An new actor is initially defined by selecting the "new actor" command in the Actor menu, as shown in Figure 7A, and then entering a text string (shown in box 235) that specifies (A) the sprite generator procedure to use to generate the image for the actor, (B) the source of the information to be displayed, (C) the nominal width and height of the sprite (e.g., in units of pixels), and (D) any optional parameters that are specific to the specified sprite generator (e.g., a font may be specified for the News information category's sprite generator, whereas a font designation parameter may be meaningless for other ones of the sprite generators).

In the preferred embodiment, the specified sprite generator must be either the static sprite generator that is part of the animation engine 204, or is any specified one of the sprite generators 202D in the category managers 202. In an alternate embodiment, additional sprite generators may be provided by the animation engine 204, such as an animated sprite generator for successively displaying a sequence of images to simulate a motion. The source of information to be displayed is either a static image, in the case of the static sprite generator, or information items in a specified information category. For instance, the parameter "NextHL" in an actor definition indicates that the information to be displayed in the corresponding sprite is the next headline in the information category corresponding to the specified sprite generator for the actor. In another example, the parameter "NextAd" in an actor definition indicates that the information to be displayed in the corresponding sprite is the next advertisement image for the information category corresponding to the specified sprite generator for the actor.

The second stage of defining a sprite is to define its position and size at one second intervals, for 30 seconds in the preferred embodiment. The position of the sprite for a particular time can be defined by either typing in an X,Y, or by selecting a box representing the sprite with the user interface and then moving it to a position on a simulated display screen 236. The size specification for the sprite at each time is a percentage of the sprite's nominal size (e.g., "size=120" indicates the sprite is to be displayed at 120% of its nominal size). The full definition for a sprite includes thirty X,Y,size tuples for a thirty second screen saver display period. In a typical display script, nor more than one advertisement, three news items and two static images are used because the resulting display will be excessively busy, although the display script definition procedure allows a virtually unlimited number of sprites to be specified.

The data structure 237 representing each display script is shown in Figure 7A: a header specifying the script's name, the number of actors defined in the script, an optional Wallpaper definition, and a list of all static images referenced by the script; plus a set of Actor definition arrays.

The screen save procedures interpret each display script and generate an animated display for 30 seconds based on the script. During display, the image corresponding to each actor is moved and sized in a virtually continuous manner, where the position and size of each sprite is linearly interpolated between the instantaneous position and size specifications for each second. During the display definition process, the sequence X,Y,size parameters for a currently selected actor can be smoothed, to produce more fluid movement and size changes of the actor by selecting the "smooth path" command in the Actor menu.

Referring to Figures 7A and 7B, the person preparing a display script using the display script definition dialog 234 can see the movement and sizing of the actors in the simulated display screen 236 by selecting the simulate command in the File menu, which cause the boxes in the simulated display screen 236 to move and be sized in accordance with the sequence of X,Y,size parameters for each specified actor.

While in the preferred embodiment advertisements are always simultaneously displayed with news items, in other embodiments advertisements and news items could be displayed sequentially. Computer programmers of ordinary skill in the art could modify the script definition dialog of the preferred embodiment, as described above, to define display scripts with sequential display of advertisements and news items.

Screen Saver Procedures

In the preferred embodiment, the screen saver procedures for displaying news items and advertisements are invoked using the same types of criteria as are used by other types of screen saver procedures. Generally, whenever the system detects a lack of user inputs via either keyboard or pointer device (e.g., a mouse or trackball) for a user configurable or otherwise specified length of time (e.g., 5 minutes), the screen saver procedures of the present invention begin the display of news items and advertisements from the local information database. In the preferred embodiment, the screen saver procedures display news items and advertisements for a sequence of information categories in a sequence of 30 second time slots.

More specifically, under the control of the screen saver procedures, news stories and an advertisement assigned to a first information category are displayed using a first display script for 30 seconds, then news stories and an advertisement assigned to a second information category are displayed using a second display script for the next 30 seconds, and so on until news stories and an advertisement have been displayed in all the information categories indicated in the subscriber's user profile 194 as being of interest to the subscriber, at which point the process repeats with the first infor-

IPR2025-00200 / 722 Patent Page 146 of 1171 EX1002

mation category.

Referring to Figure 8, news stories, advertisements and display scripts are stored in files or similar data structures which have assigned unique file names. Each news story (herein usually called a news item) is usually assigned to a single information category, although nothing in the system of the preferred embodiment would prevent a news story from being assigned to multiple information categories. Advertisements can be assigned to multiple information categories as can display scripts.

As shown in Figures 8 and 9, the advertisements assigned to each information category are organized, through the use of a set of data access tables 186, in a separate linked list so as to create a separate "queue" of advertisements for each information category. Similarly the news items and display scripts assigned to each information category are organized in separate linked lists so as to generate separate queues of news items and display scripts for each information category.

Figure 8 includes an example of an advertisement (A001) assigned to two information categories (News and Sports). This advertisement is stored only once in the workstation's local hard disk, but is included in two of the linked lists of advertisements.

The basic procedure for determining what display script, advertisement and news stories to display during each 30 second time slot is shown in pseudocode form in Table 1.

TABLE 1

Pseudocode Representation of Screen Saver Procedure

```
Store, indication of last information category displayed, and for each
    category an indication of the last advertisement, news story and
    display script used.
Do Until Screen Saver Mode is exited:
    {
    Select next information category (SIC).
    Select next display script (SDS) from queue of display scripts and
        next advertisement (SA) from queue of advertisements for the
        selected information category.
    Inspect selected display script to determine NN, the number of news
        items to be displayed.  Select the NN next news items (SNI)
        from queue of news items for the selected information category.
    Update User Profile to indicate the last selected information
        category, and to indicate for the selected information category,
        the selected display script, advertisement and last selected news
        story.
    Call Animation Engine (SDS, SA, SNI) to display for the next 30
        seconds the selected advertisement (SA) and news items (SNI)
        under the direction of the selected display script (SDS).
    Call Ad Display Statistics Generator to update displayed
        advertisement statistics to include the advertisement displayed
        during current screen saver display period.
    }
```


IPR2025-00200 / 722 Patent Page 147 of 1171 EX1002

Each time the Screen Saver procedure 201 is invoked, it starts with the next information category after the last one to have been used, and starts with the next advertisement and news stories after the last ones used in that information category. The screen saver status information 219 indicating the last displayed information category and the last displayed advertisement and news items in each information category are stored in a portion of the user profile 194 not transmitted to the information server.

Execution of the Screen Saver procedure 201, like other screen savers, is terminated and the subscriber's computer's display is returned to whatever was being displayed before the Screen Saver was executed, upon detection of certain types of user input. In the preferred embodiment, the user can use the profiler to select one of at least two exit modes: in a first mode, the Screen Saver procedure is terminated by hitting any key on the subscriber computer's user interface keyboard or by moving the user interface's mouse or trackball; in a second mode, the Screen Saver procedure is terminated by hitting any key on the subscriber computer's user interface keyboard, but movement of the mouse or trackball does not cause the Screen Saver procedure to terminate. Rather, in the second screen saver exit mode, the subscriber can use the mouse or trackball to point to any of the news items being displayed and upon clicking one of the mouse or trackball's buttons, the data viewer 208 is executed with the news item selected by the subscriber being displayed.

When using the second screen saver exit mode, if subscriber user clicks on an advertisement, the subscriber's computer is automatically connected to the an associated World Wide Web page on the Internet that provides additional information from the advertiser. This is accomplished by World Wide Web connection and viewer procedures 211 (see Figure 2) stored on subscriber's computer. Each advertisement is stored on both the information server and subscriber computers as a C++ data structure that includes (A) an image data array, typically representing a "GIF" format image, as well as (B) a list of static images (such as corporate logos and legends), if any, incorporated into the advertisement, and (C) a Web site address that is used by the World Wide Web connection and viewer procedures 211 to connect the subscriber to the advertiser's specified Web page when the subscriber clicks on the image of the associated advertisement.

Data Viewer

Referring to Figure 10, the data viewer 208 is a program for viewing news items that the subscriber specifically wants to read. The data viewer 208 can be executed at the subscriber's explicit command, and as just described in the immediately preceding section of this document, the data viewer can also be launched from the screen saver when the subscriber indicates that he/she wants to read a news story shown in the screen saver display by "clicking" the subscriber's computer's mouse or trackball on that news story.

The news stories shown in the center section 248 of the data viewer's display is selected by first selecting an information category by clicking on any of the category buttons 250 on the left margin of the display, and a subcategory button 252, if any, on the bottom margin of the display, and then clicking on the article advance backward and forward buttons 254 to scroll through the news items in the selected information category. When a news item has more than one photo image associated with it, the subscriber can click on the photo advance backward and forward buttons 256 to scroll through the photos.

Each news item displayed in the center section 248 of the data viewer's display includes both the primary and secondary portions of the news item, thereby providing the subscriber in most instances with access to a fuller version of the news item than was shown by the screen saver. In the case of very short news items, the entire news item may be contained in its primary component. Furthermore, in client computers with very limited hard disk space available for storing news items, as indicated by the user profile 194 for the client computer, the secondary component of news items may not be stored in the local information database in order to conserve disk space.

A portion of the data viewer screen is always occupied by an advertisement image 258. The advertisement image shown is selected on the basis of the information category associated with the news item being viewed. In a preferred embodiment, the advertisement shown in the data viewer screen is changed (A) every time the subscriber clicks on a category button 250 so as to select a different information category than the one previously selected, and (B) every 30 seconds when subscriber continues to view news items in a single information category for more than 30 seconds. The advertisements are selected in rotating order among the advertisements assigned to each information category, as described above for the screen saver procedure.

When using the data viewer, if subscriber user clicks on the displayed advertisement, the subscriber's computer is automatically connected to the an associated World Wide Web page on the Internet that provides additional information from the advertiser.

The Options button 260 is used to invoke dialog procedures in which the subscriber specifies general preferences, such as how quickly data scrolls in the scrolling windows, and which mode of screen-saver termination the subscriber prefers.

IPR2025-00200 / 722 Patent Page 148 of 1171 EX1002

Connecting the Subscriber's Computer to the Information Server

Referring to Figures 11 and 12, the information server is preferably a set of computers interconnected by a local area network that each operate under a multi-tasking, multi-threading operating system such as Microsoft's Windows NT. The information server 104 has multiple "application servers" 272, which are processes run on one or more computers. Each application server 272 preferably has multiple threads, each of which can service one connection with a client computer at any one time.

A primary concern with the architecture of the information server is that the information be able to handle a very large volume of connection requests from client computers. The information server may need to service thousands of connection requests per hour, and thus efficient handling of each connection request is important.

In a preferred embodiment, during each connection of a subscriber computer to the information server, the information server sends a "next recommended download time" to the subscriber computer along with the other information being downloaded onto the subscriber computer. The server computer selects the next recommended download times sent to the various subscriber computers so as to spread their connection requests fairly evenly over time. In an alternate embodiment, connection requests are spread over time by having the subscriber computers randomly select connection times within the general boundaries of a specified schedule of connections (e.g., a randomly selected time anywhere within a half hour, plus or minus, of each scheduled connection time).

When a client computer first initiates a connection to the information server, it sends a first message to the Internet address associated with a router process 270 in the information server. The router selects an application server 272 with at least one available thread and passes back to the client computer an Internet address associated with that application server.

The client computer then sends a portion of its user profile to the assigned application server. If an administrative update is being requested, the locally accumulated advertising display statistics 218 (see Figure 4) are also sent to the application server.

Based on the time of day and the information in the transmitted user profile, the application server determines (A) what type of update is to be performed (i.e., a news item update or an administrative update), and (B) what new information needs to be downloaded to the client computer and what items in the client computer's local information database should be deleted. The application server 272 then makes calls to one or more data servers 274 to collect all the information that needs to be sent to the client computer and then sends those items to the client computer, along with instructions on what items, if any, should be deleted from the client computer's local information database.

The client computer then loads the received information into its local database, and replaces software modules with received software modules, if any. It also deletes the items, if any, specified for deletion by the information server. Finally, it updates its data access tables 186 to incorporate all the changes to the information database so that the client computer is ready to display news items and advertisements in each information category.

A more detailed explanation of the local database update process is provided by a pseudocode representation of that process in Table 2.

In one preferred embodiment, when the "client" that is connected to the information server for an update is itself a local area network server, the client downloads all news items into its local database. In a second preferred embodiment, the client/LAN server generates a group profile that represents the union of all news category and subcategory preferences of the subscribers connected to the client computer, and news items are downloaded into the client's local database based on that union group profile. In either embodiment, the screen saver procedures filter out news items in the LAN server's local information database that are not consistent with each subscriber's user profile, thereby showing each subscriber only the subset of news items corresponding to the subscriber's user profile. In the preferred embodiments, the subscriber level news item filtering is accomplished by setting up the subscriber's data access tables 186 to include only news items corresponding to the subscriber's user profile. In the computers of stand alone subscribers, the filtering of news stories is handled during the data download process, by only downloading news items corresponding to the subscriber's user profile.

The subscriber level news item filtering function is also used to enable the information server to instruct the subscribers' computers to "black out" an advertisement, without deleting it from the local database. For example, a company may want to suspend its advertisements for a few days after a disaster involving the company. The black out function is achieved by simply removing the corresponding advertisement(s) from the advertisement queues in the data access tables. For this purpose, the information server and subscriber computers may temporarily define a "non-use" information category and a corresponding advertisement queue for keeping track of blacked out items.

IPR2025-00200 / 722 Patent Page 149 of 1171 EX1002

TABLE 2

Pseudocode Representation of Database Update Procedure

```
Connect to Information Server
If Update Type=Administrative  /* i.e., not a news story only update */
    {
    Client sends display statistics to server, and clears display statistics
        upon confirmation that server has successfully received them

    /* Pool Synchronization */
    Server Sends list of items (i.e., advertisement and scripts) that should be
        included in the client's advertisement and script pools
    Client deletes items in its advertisement and script pools that are not
        included in the list received from the Server
    Client determines what items are missing from its advertisement and
        script pools
    Client sends requests to Server for advertisements and scripts
        determined to be missing from local pools
```


IPR2025-00200 / 722 Patent Page 150 of 1171 EX1002

```
        Server sends requested items to Client
        Client stores received advertisements and scripts in their respective disk
            directories
        Client opens all advertisement and script files to determine the static
            images referenced by those files, but not included in the local static
            image pool.
        Client sends requests to Server for static images determined to be
            missing from local pool
        Server sends requested items to Client
        Client stores received static images in their assigned disk directory

        /* Software Module Synchronization */
        Client sends message indicate it is ready for software synchronization,
            including date and time of last administrative update
        Server sends new software modules, if any, based on date and time of
            last administrative update
        }

    For each Category Manager (CMx)
        {
        /*  CMx.Fetch Procedure: */
        Client (CMx.Fetch procedure) sends profile data for CMx to Server,
            including subcategory data and filter data, if any
        Server sends items consistent with profile data
        Client (CMx.Fetch procedure) stores received items in data structures
            and files for that category
        Client (CMx.Fetch procedure) deletes items, in FIFO order, for current
            category which (A) exceed data storage limit in date, (B) exceed
            item count limit, or (C) exceed specified age limit
            /*  Item storage limits 221 for each category are defined in a portion
                of the user profile 194 (see Figure 4)  */
        }

    Client updates data access tables
    Return
```

ALTERNATE EMBODIMENTS AND EXTENSIONS

In an alternate embodiment of the present invention, the server's information database 134 also includes a client

IPR2025-00200 / 722 Patent Page 151 of 1171 EX1002

catalog which lists all subscribers authorized to receive news items and advertisements from the server, including a connection password that is checked whenever the subscriber's computer calls the information server for an update, and status information included the last time that each subscriber's computer received updated news items, advertisements, scripts, and software modules.

In another alternate embodiment, the information server broadcasts information updates to all the subscriber computers, for example by sending an e-mail message or a sequence of e-mail messages containing all news item, advertising, display script and software updates to all the computers of registered subscribers.

Claims

1. An information and advertising distribution system, comprising:

   a data server for storing and updating a database of information items and advertisements, at least a subset of said information items having an associated information category;
   a plurality of workstations remotely located from said data server for use by a corresponding set of subscribers; each workstation including:

   a display device;
   a communication interface for establishing a communication connection with said data server and for receiving at least a subset of said information items and advertisements in said data server's database;
   local memory for storing the information items and advertisements received from said data server;
   an information administrator that establishes communication with said data server from time to time so as to update the information items and advertisements stored in said local memory with at least a subset of said information items and advertisements stored by said data server; and
   an information display controller for displaying on said display device at least a subset of said information items and advertisements stored in said local memory when said workstation meets predefined idleness criteria.

2. The information and advertising distribution system of claim 1, wherein said information display controller also displays said information items and advertisements on said display device upon request by one of said subscribers using said workstation.

3. The information and advertising distribution system of claim 1, wherein

   said data server includes means for receiving new information, for updating said information items stored by said data server with new information items corresponding to said new information and for deleting older ones of said stored information items;
   said information administrator in at least a subset of said workstations includes a scheduler that prompts said information administrator to download said information items from said data server so as to update the information items stored in those workstations' local memory more often than said information administrator downloads said advertisements from said data server so as to update the advertisements stored in those workstations' local memory.

4. The information and advertising distribution system of claim 1, wherein

   at least a subset of said information items stored in said local memory and in said data server each have associated therewith at least one information category selected from a defined set of information categories;
   at least a subset of said workstations include a profiler for storing subscriber profile data, said subscriber profile data including data representing subscriber information viewing preferences for viewing information items associated with various ones of said defined set of information categories; and
   said information display controller includes a filter for excluding from the information items displayed on said display device those of said information items inconsistent with said subscriber information viewing preferences.

5. The information and advertising distribution system of claim 1, wherein

   at least a subset of said information items stored in said local memory and in said data server each have associated therewith at least one information category selected from a defined set of information categories;
   at least a subset of said workstations include a profiler for storing subscriber profile data, said subscriber profile

IPR2025-00200 / 722 Patent Page 152 of 1171 EX1002

data including data representing subscriber information viewing preferences for viewing information items associated with various ones of said defined set of information categories; and

said information administrator includes a filter for excluding from the information items stored in said local memory those of said information items inconsistent with said subscriber information viewing preferences.

6. The information and advertising distribution system of claim 1, wherein

at least a subset of said information items stored in said local memory and in said data server each have associated therewith at least one information category selected from a defined set of information categories;

said information display controller displays on said display device, in successive time periods, ones of said information items stored in said local memory associated with successively selected ones of said information categories.

7. The information and advertising distribution system of claim 1, wherein

said local memory of each said workstation stores a plurality of display scripts, each display script for controlling display of at least one information item in conjunction with display of at least one advertisement; and

said information display controller displays on said display device, in successive time periods, said information items stored in said local memory in conjunction with advertisements stored in said local memory, utilizing successive ones of said display scripts stored in said local memory during said successive time periods.

8. The information and advertising distribution system of claim 7, wherein

at least a subset of said information items stored in said local memory and in said data server each have associated therewith at least one information category selected from a defined set of information categories;

at least a subset of said advertisements stored in said local memory and in said data server each have associated therewith at least one information category selected from a defined set of information categories; and

said information display controller displays on said display device, during each of at least a subset of said successive time periods, ones of said information items and advertisements associated with a same one of said defined set of information categories.

9. The information and advertising distribution system of claim 7, wherein

said information items stored in said local memory and in said data server each have associated therewith an information category selected from a defined set of information categories;

said advertisements stored in said local memory and in said data server each have associated therewith at least one information category selected from a defined set of information categories; and

said information display controller displays on said display device, during each said successive time period, at least one of said locally stored advertisements and ones of the locally stored information items associated with a selected one of said defined set of information categories, wherein said selected one of said defined set of information categories changes during successive time periods.

10. The information and advertising distribution system of claim 1, wherein

said data server stores a set of display scripts, each display script for controlling display of at least one information item in conjunction with display of at least one advertisement;

said local memory of each said workstation stores at least a subset of said display scripts;

said information administrator updates the display scripts stored in said local memory with at least a subset of said display scripts stored by said data server; and

said information display controller displays on said display device, in successive time periods, said information items stored in said local memory in conjunction with advertisements stored in said local memory, utilizing successive ones of said display scripts stored in said local memory during said successive time periods.

11. The information and advertising distribution system of claim 10, wherein

at least a subset of said information items stored in said local memory and in said data server each have associated therewith at least one information category selected from a defined set of information categories;

at least a subset of said advertisements stored in said local memory and in said data server each have associated therewith at least one information category selected from a defined set of information categories; and

IPR2025-00200 / 722 Patent Page 153 of 1171 EX1002

said information display controller displays on said display device, during each of at least a subset of said successive time periods, ones of said information items and advertisements associated with a same one of said defined set of information categories.

12. The information and advertising distribution system of claim 10, wherein

said information items stored in said local memory and in said data server each have associated therewith an information category selected from a defined set of information categories;
said advertisements stored in said local memory and in said data server each have associated therewith at least one information category selected from a defined set of information categories; and
said information display controller displays on said display device, during each said successive time period, at least one of said locally stored advertisements and ones of the locally stored information items associated with a selected one of said defined set of information categories, wherein said selected one of said defined set of information categories changes during successive time periods.

13. An information and advertising distribution system, comprising:

a data server for storing and updating a database of information items and advertisements, at least a subset of said information items having an associated information category;
a plurality of workstations remotely located from said data server for use by a corresponding set of subscribers; each workstation including:

a display device;
a communication interface for establishing a communication connection with said data server and for receiving at least a subset of said information items and advertisements in said data server's database;
local memory for storing the information items and advertisements received from said data server;
an information administrator that updates the information items and advertisements stored in said local memory from time to time with at least a subset of said information items and advertisements stored by said data server; and
an information display controller for displaying on said display device at least a subset of said information items and advertisements stored in said local memory when said workstation meets predefined idleness criteria.

14. An information and advertising display system, comprising:

a display device;
local memory for storing information items and advertisements;
a plurality of display scripts, each display script for controlling display of at least one information item in conjunction with display of at least one advertisement; and
an information display controller for displaying on said display device at least a subset of said information items and advertisements stored in said local memory when said workstation meets predefined idleness criteria;
wherein said information display controller displays on said display device, in successive time periods, said information items stored in said local memory in conjunction with advertisements stored in said local memory, utilizing successive ones of said display scripts stored in said local memory during said successive time periods.

IPR2025-00200 / 722 Patent Page 154 of 1171 EX1002

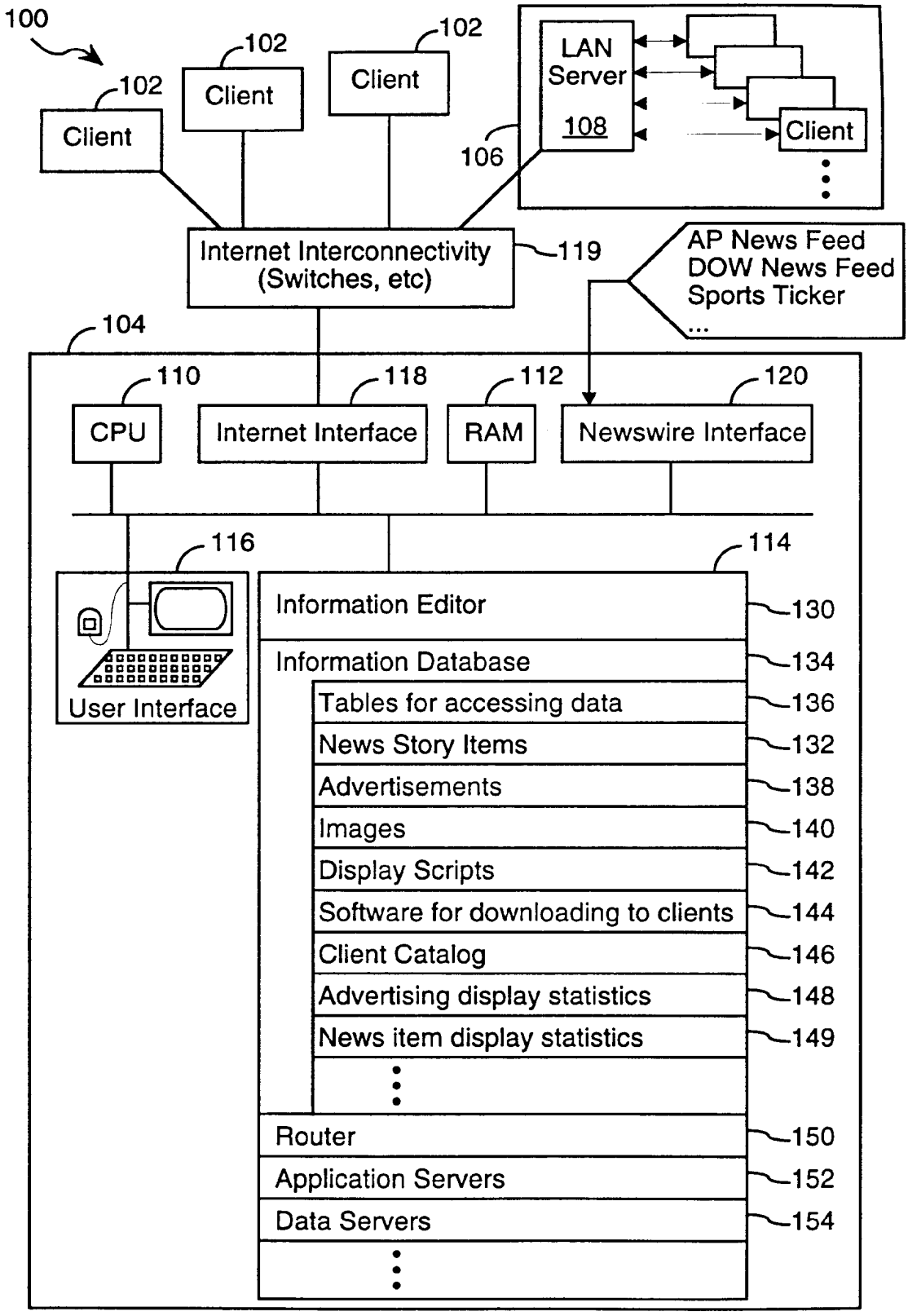

100
102
Client
102
Client
102
Client
LAN Server
108
106
Client
Internet Interconnectivity (Switches, etc)
119
AP News Feed
DOW News Feed
Sports Ticker
...
104
110
CPU
118
Internet Interface
112
RAM
120
Newswire Interface
116
User Interface
114
Information Editor
130
Information Database
134
Tables for accessing data
136
News Story Items
132
Advertisements
138
Images
140
Display Scripts
142
Software for downloading to clients
144
Client Catalog
146
Advertising display statistics
148
News item display statistics
149
Router
150
Application Servers
152
Data Servers
154

FIGURE 1

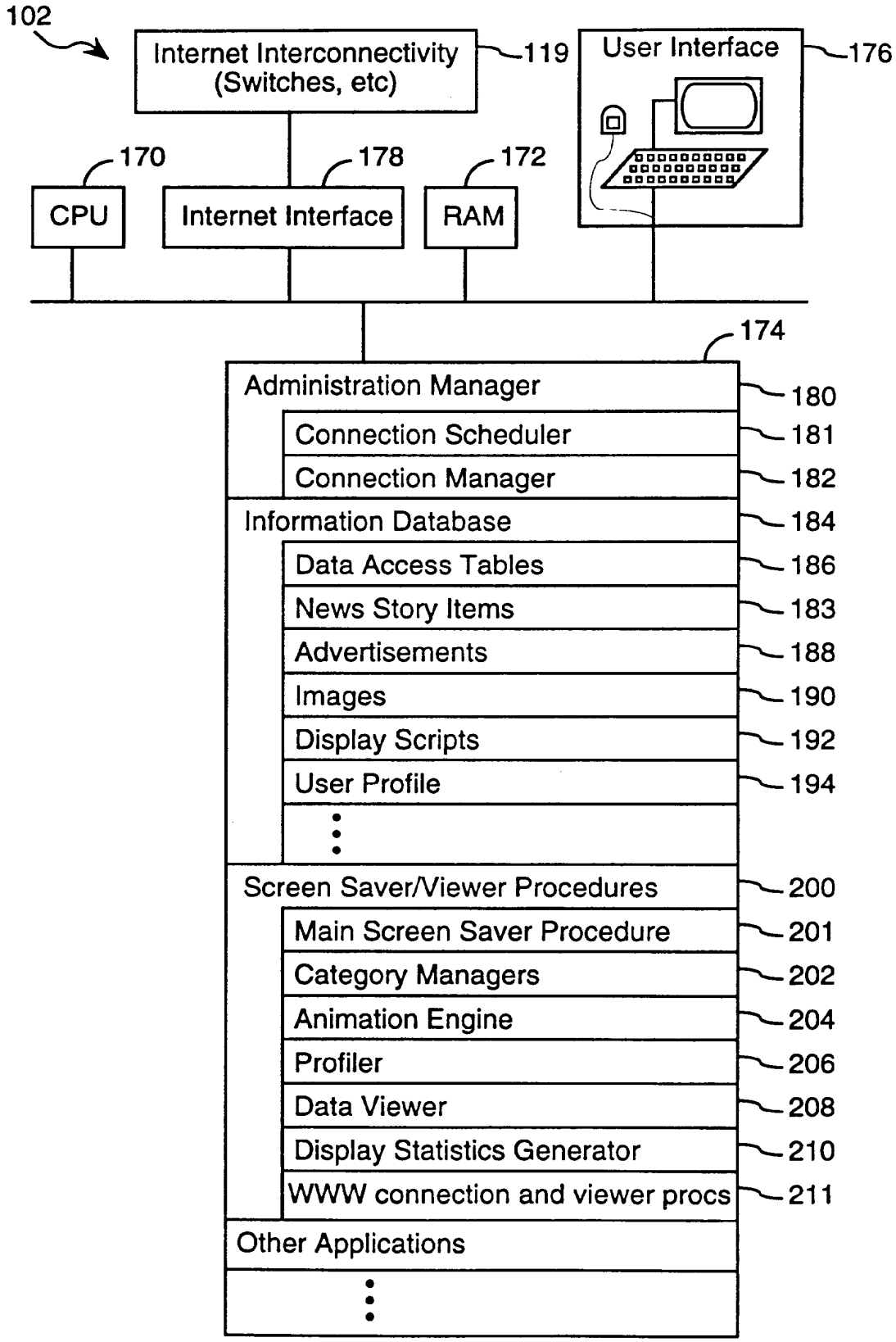

102
Internet Interconnectivity (Switches, etc)
119
User Interface
176
170
178
172
CPU
Internet Interface
RAM
174
Administration Manager
180
Connection Scheduler
181
Connection Manager
182
Information Database
184
Data Access Tables
186
News Story Items
183
Advertisements
188
Images
190
Display Scripts
192
User Profile
194
Screen Saver/Viewer Procedures
200
Main Screen Saver Procedure
201
Category Managers
202
Animation Engine
204
Profiler
206
Data Viewer
208
Display Statistics Generator
210
WWW connection and viewer procs
211
Other Applications

FIGURE 2

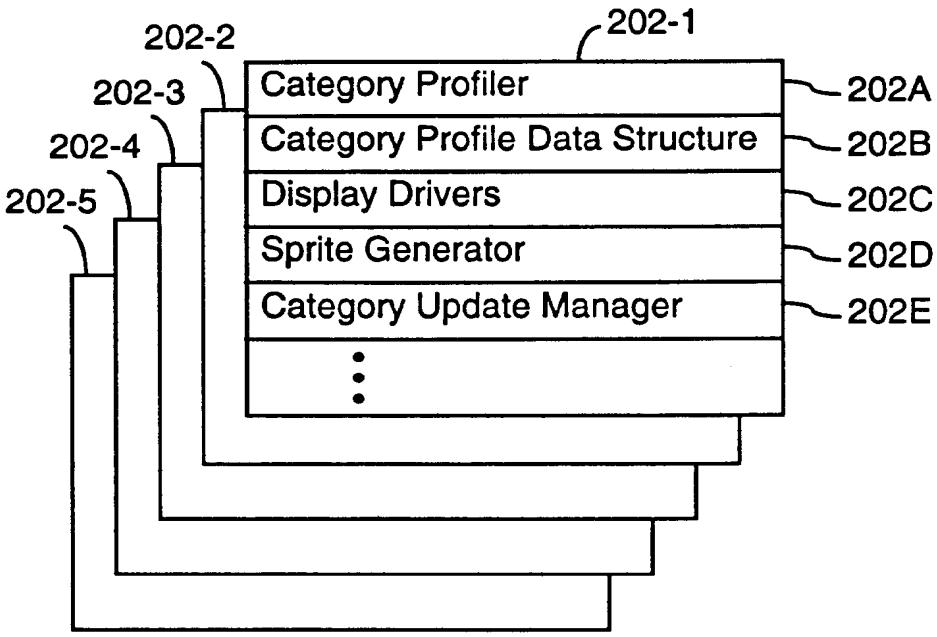


FIGURE 3

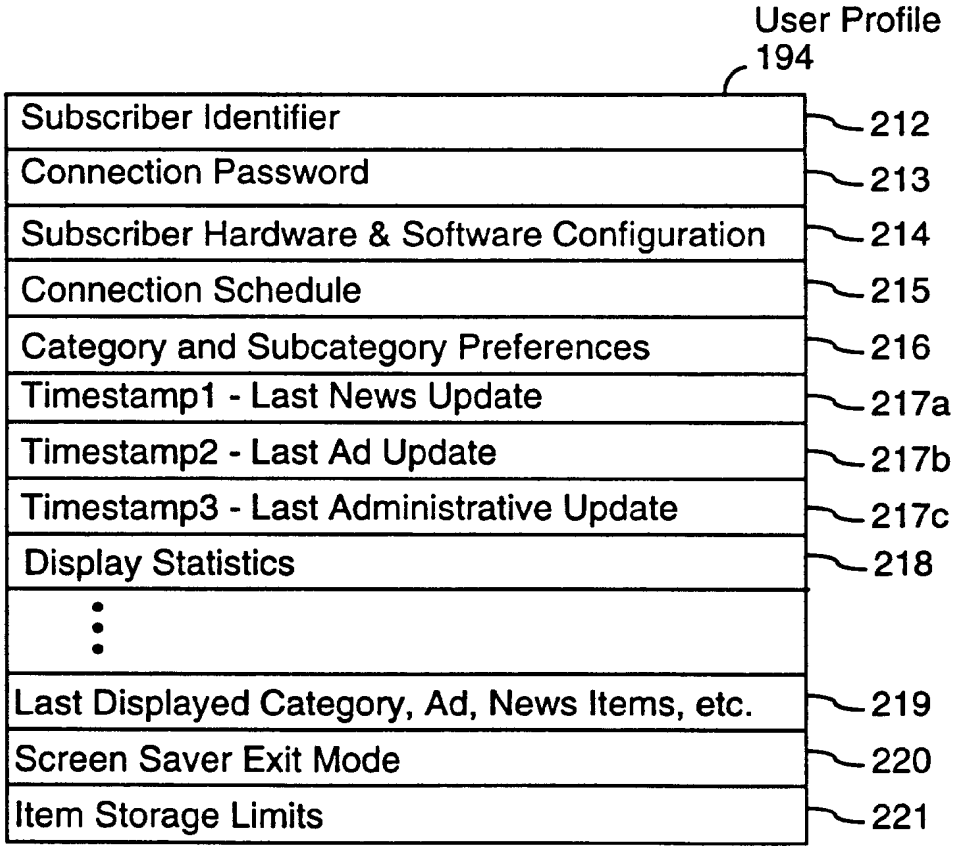


FIGURE 4

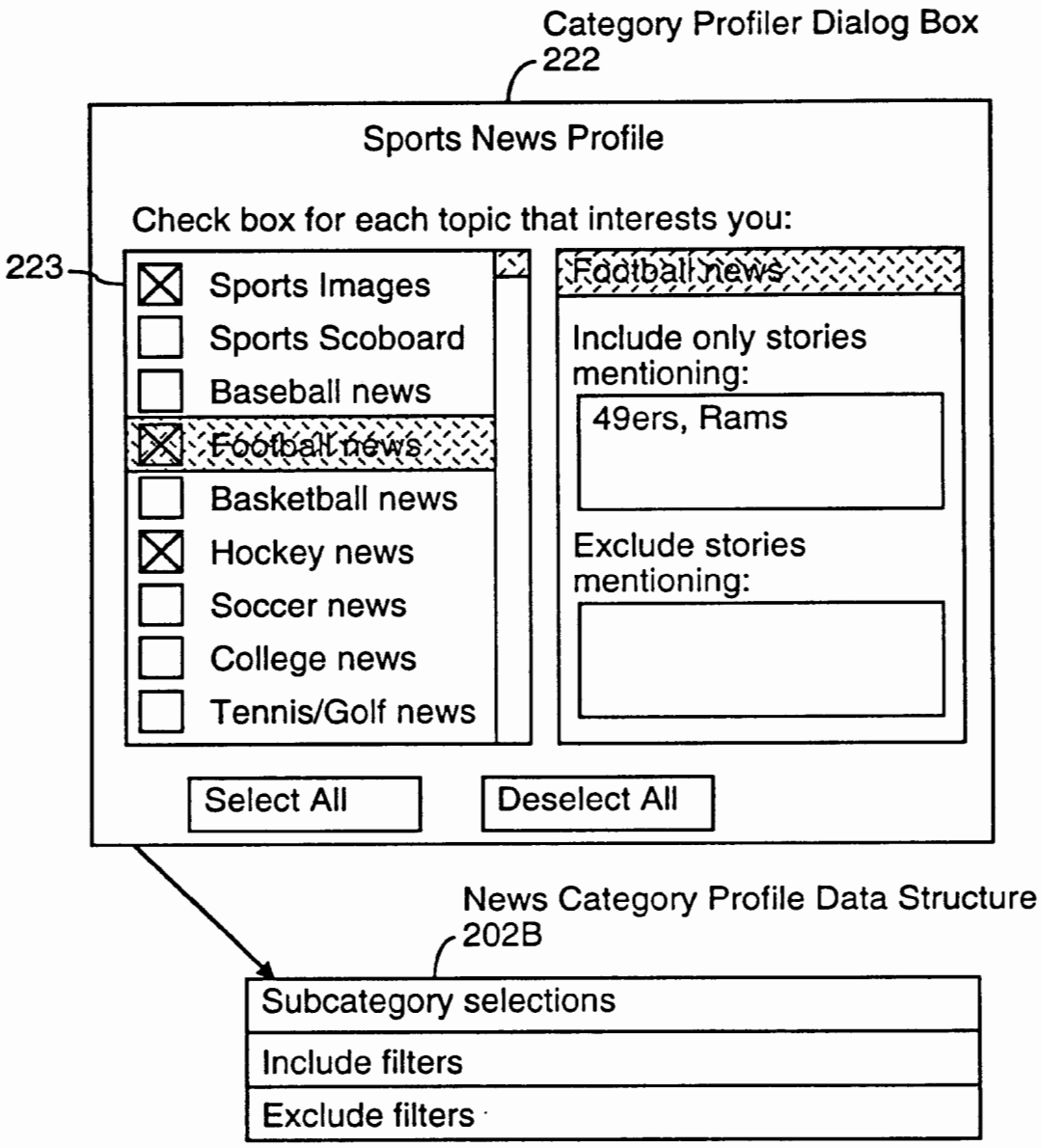

Category Profiler Dialog Box
222
Sports News Profile
Check box for each topic that interests you:
223
Sports Images
Sports Scoboard
Baseball news
Football news
Basketball news
Hockey news
Soccer news
College news
Tennis/Golf news
Football news
Include only stories mentioning:
49ers, Rams
Exclude stories mentioning:
Select All
Deselect All
News Category Profile Data Structure
202B
Subcategory selections
Include filters
Exclude filters

FIGURE 5

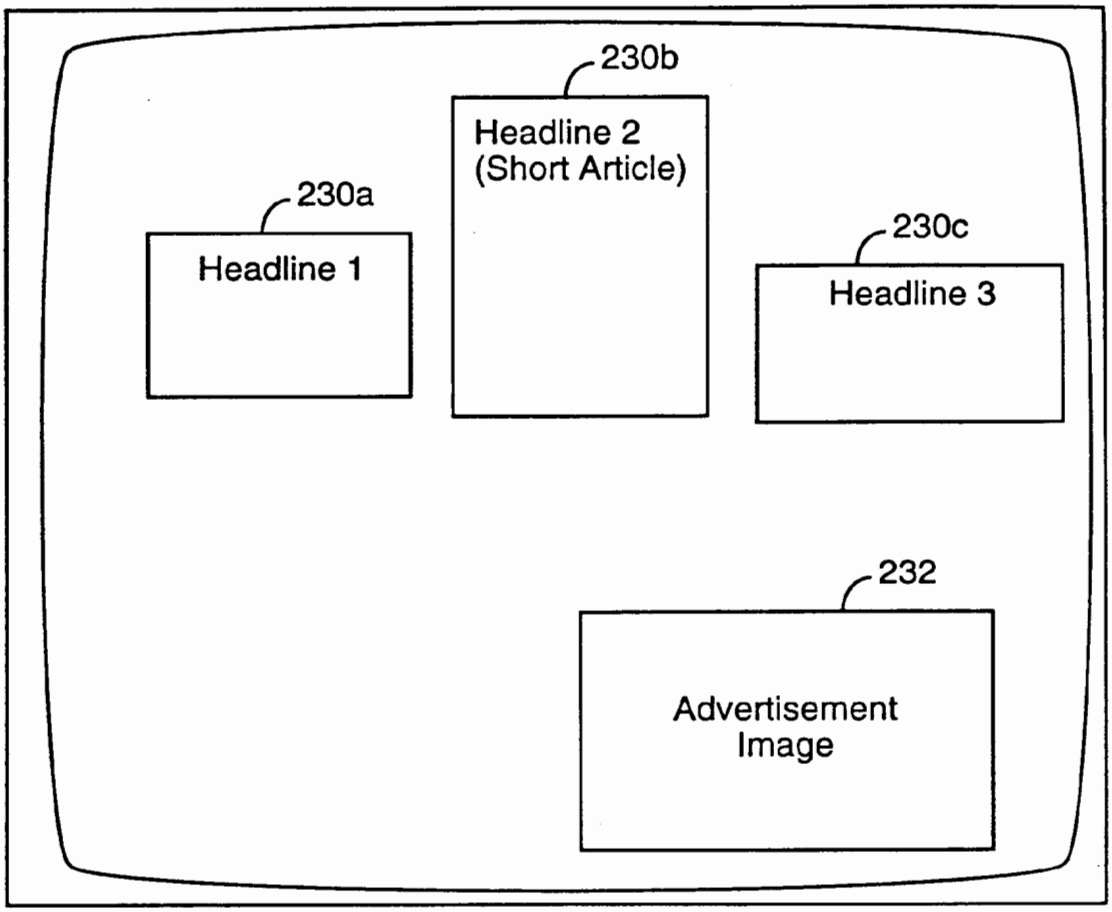

230b
Headline 2
(Short Article)
230a
Headline 1
230c
Headline 3
232
Advertisement
Image

FIGURE 6

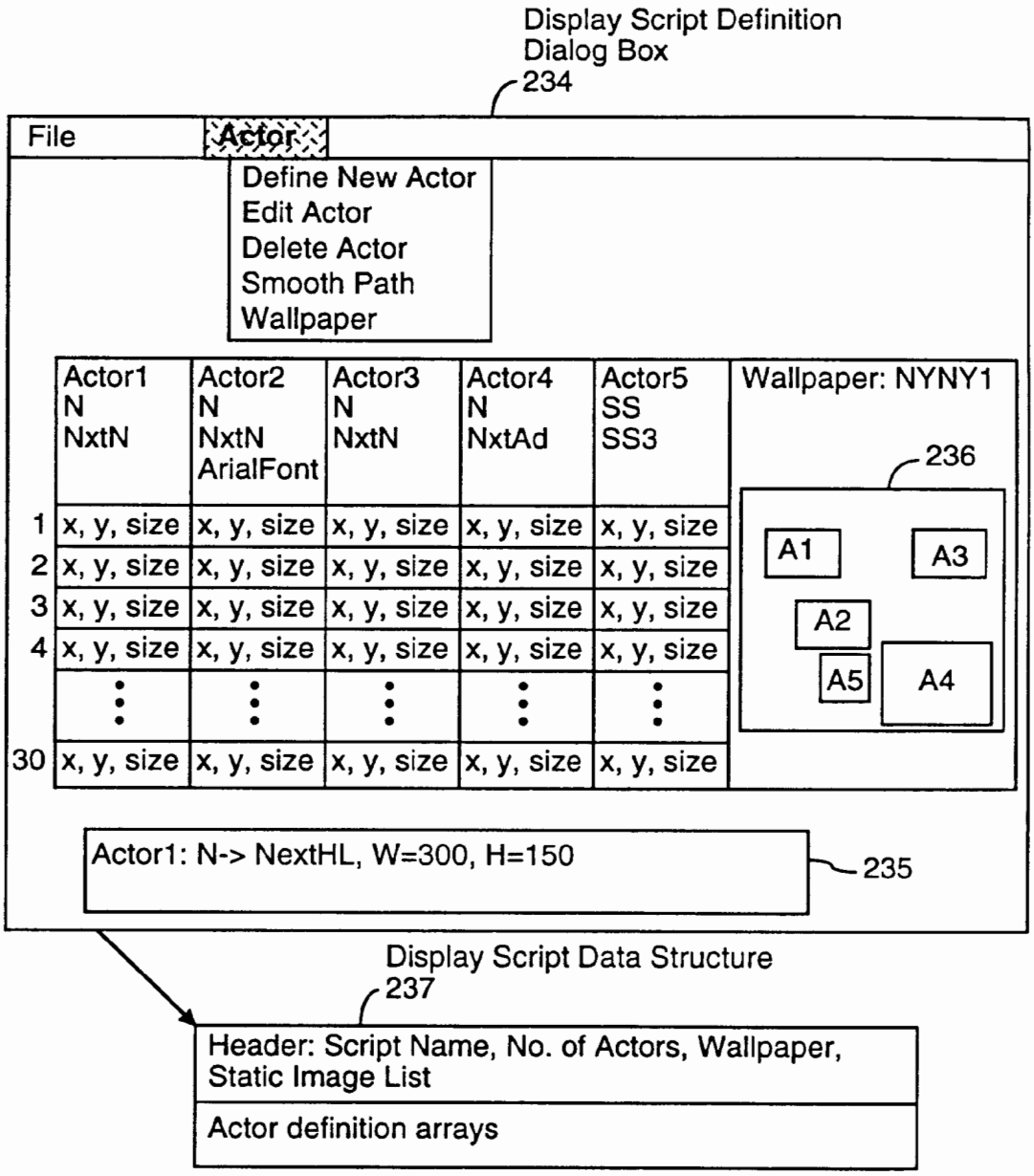


FIGURE 7A

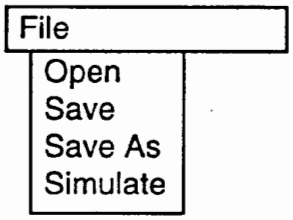


FIGURE 7B

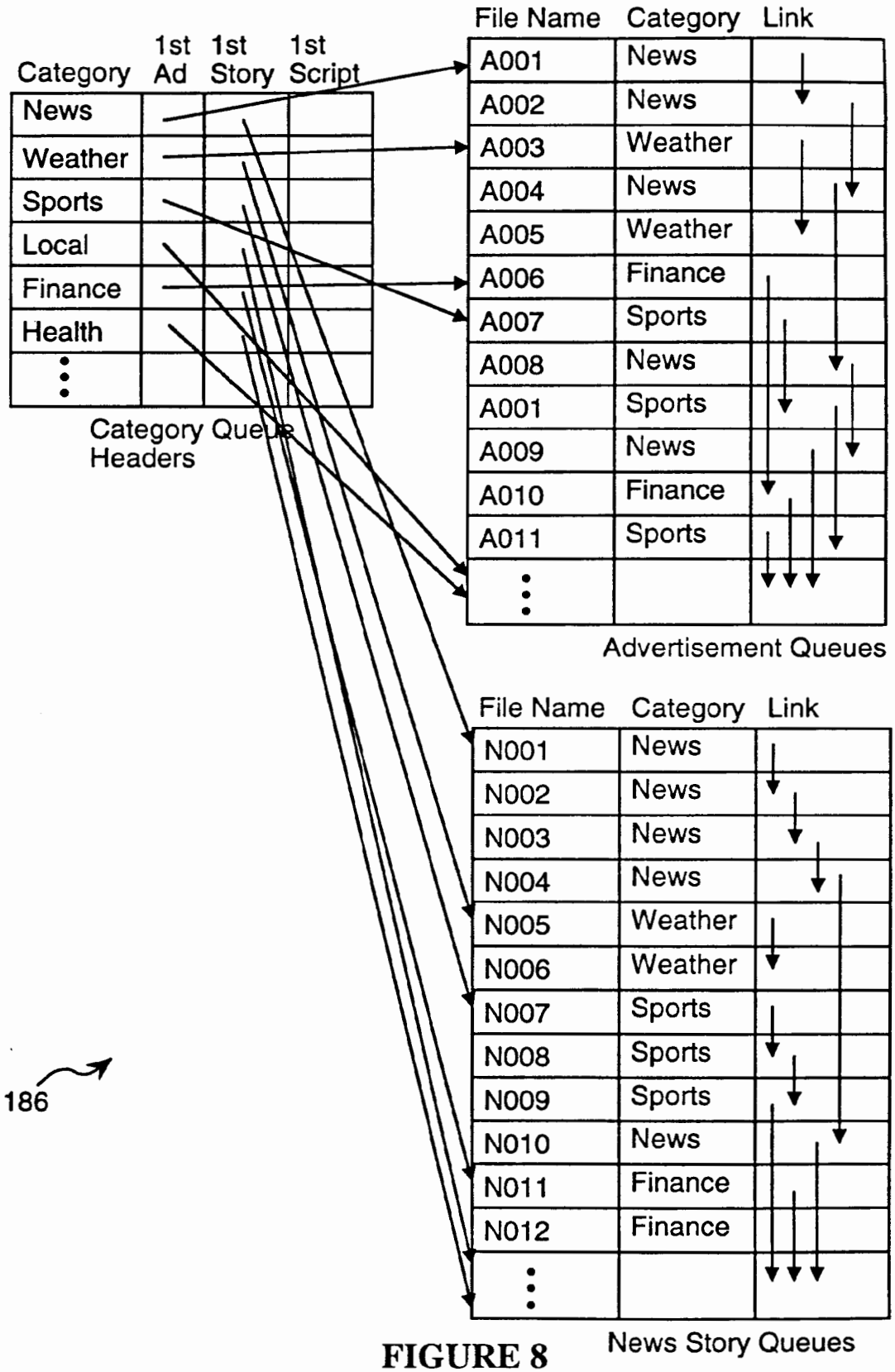

Category
1st Ad
1st Story
1st Script
News
Weather
Sports
Local
Finance
Health
Category Queue Headers
File Name
Category
Link
A001 News
A002 News
A003 Weather
A004 News
A005 Weather
A006 Finance
A007 Sports
A008 News
A001 Sports
A009 News
A010 Finance
A011 Sports
Advertisement Queues
File Name
Category
Link
N001 News
N002 News
N003 News
N004 News
N005 Weather
N006 Weather
N007 Sports
N008 Sports
N009 Sports
N010 News
N011 Finance
N012 Finance
News Story Queues
186

FIGURE 8

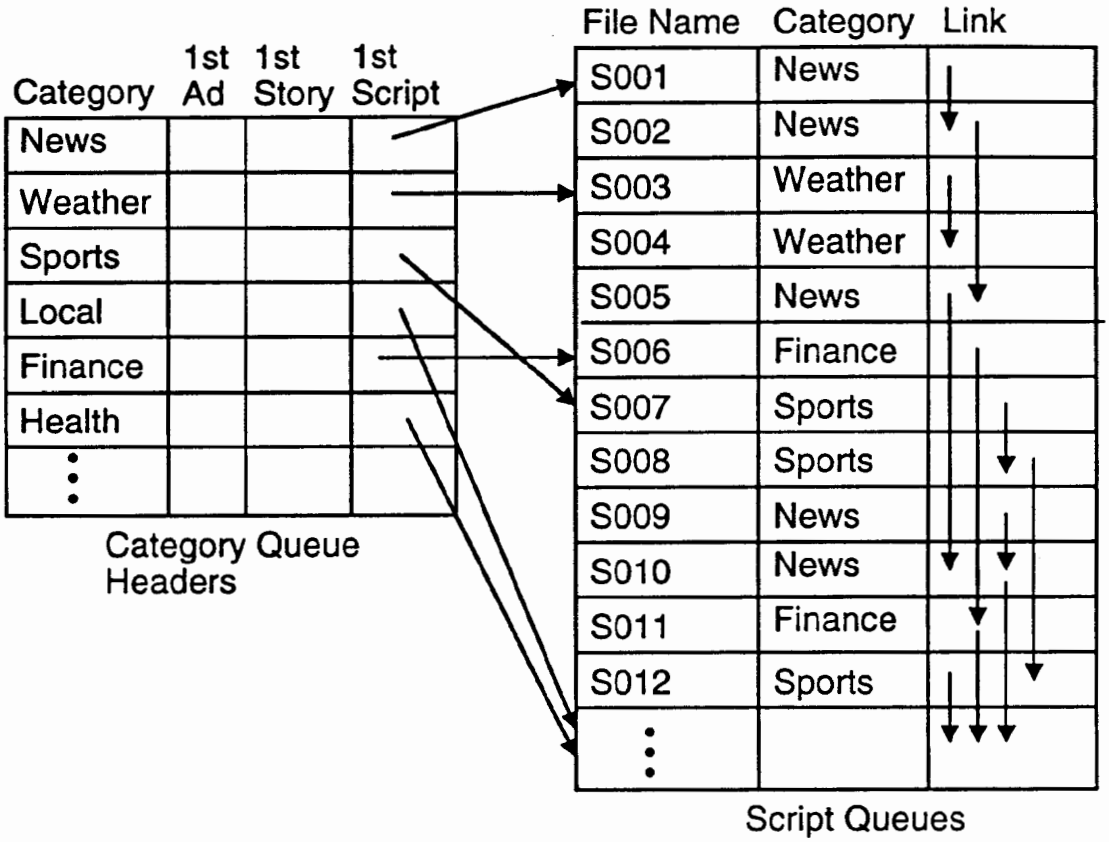

Category
1st Ad
1st Story
1st Script
News
Weather
Sports
Local
Finance
Health
Category Queue Headers
File Name
Category
Link
S001 News
S002 News
S003 Weather
S004 Weather
S005 News
S006 Finance
S007 Sports
S008 Sports
S009 News
S010 News
S011 Finance
S012 Sports
Script Queues

FIGURE 9

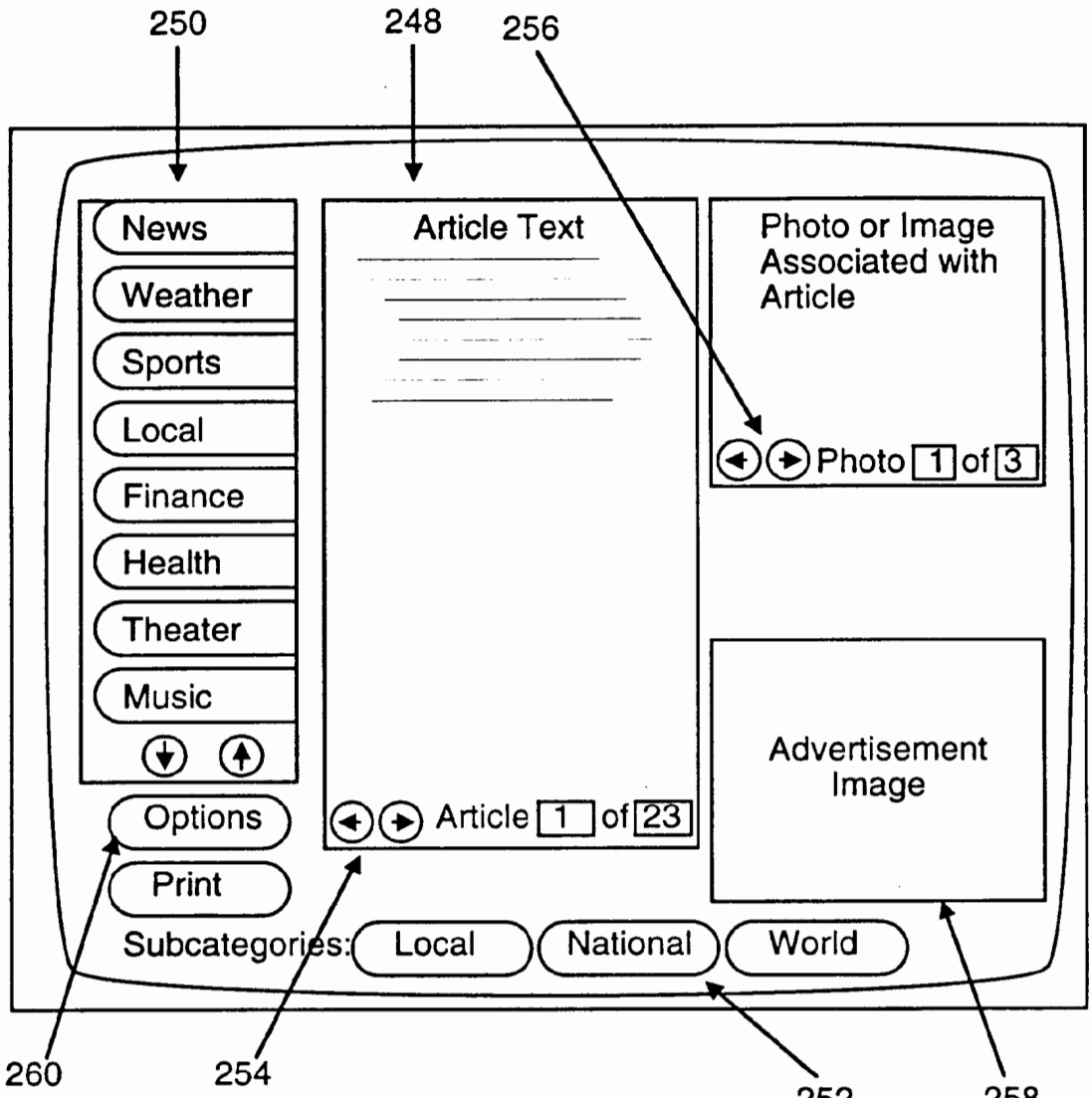

250
248
256
News
Weather
Sports
Local
Finance
Health
Theater
Music
Options
Print
Article Text
Article 1 of 23
Photo or Image Associated with Article
Photo 1 of 3
Advertisement Image
Subcategories:
Local
National
World
260
254
252
258

FIGURE 10

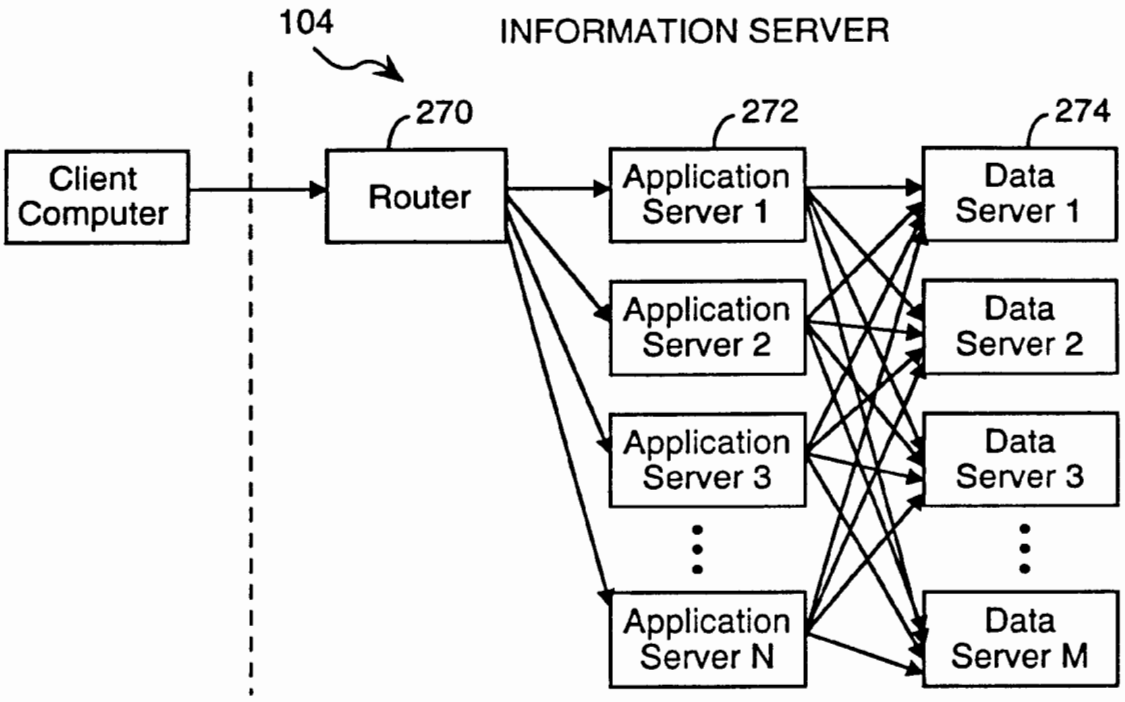


FIGURE 11

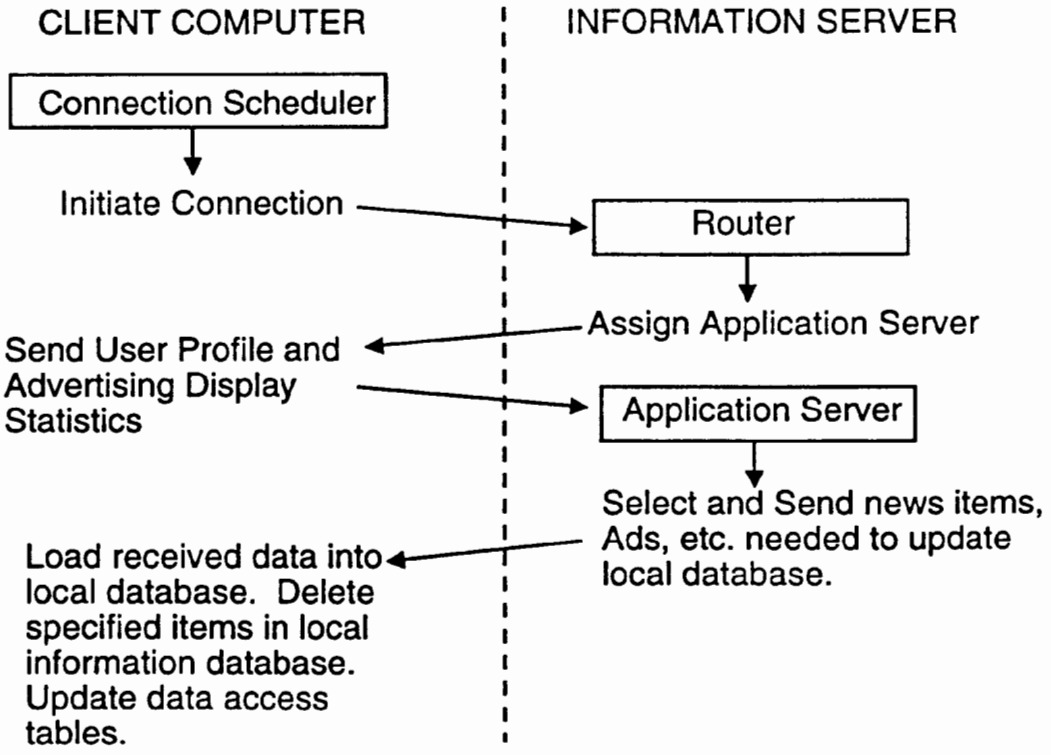


FIGURE 12

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 96 30 3727

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.6)
X	EP-A-0 113 022 (INT STANDARD ELECTRIC CORP ;STANDARD ELEKTRIK LORENZ AG (DE)) 11 July 1984	1,13,14	G06F17/60
Y	* page 2, line 3 - line 16; figure 1 *	4	
A	* page 5, line 23 - page 7, line 6 * * abstract *	2,3	
X	PATENT ABSTRACTS OF JAPAN vol. 015, no. 476 (E-1140), 4 December 1991 & JP-A-03 204259 (NIPPON TELEGR & TELEPH CORP), 5 September 1991,	1,13,14	
Y	* abstract *	4	
A		2,3	
X	PATENT ABSTRACTS OF JAPAN vol. 95, no. 2, 31 March 1995 & JP-A-06 314184 (KOKUSAI ELECTRIC CO LTD), 8 November 1994,	14	
A	* abstract *	1,7,8,13	TECHNICAL FIELDS SEARCHED (Int.Cl.6)
X	GB-A-2 281 434 (YAMAHA CORP) 1 March 1995	1-3,13, 14	G06F G09F
Y	* page 6, line 22 - page 11, line 6; figures 1,2 * * abstract *	4,5	
X	WO-A-93 19427 (SINGER GERALD ;MURPHY ARTHUR J (US)) 30 September 1993	1-3	
Y	* page 7, line 12 - page 8, line 16; figure 1 * * abstract *	4,5	

-/--

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
BERLIN	29 August 1996	Deane, E

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 96 30 3727

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.6)
X	PATENT ABSTRACTS OF JAPAN vol. 017, no. 029 (P-1472), 20 January 1993 & JP-A-04 250591 (NEC ENG LTD), 7 September 1992, * abstract *	14	
Y	US-A-5 283 731 (LALONDE JAMES E ET AL) 1 February 1994 * abstract *	4,5	
A	GB-A-2 141 907 (GILMORE MICHAEL) 3 January 1985 * abstract; figure 2 *	14	
Y	DATABASE INSPEC INSTITUTE OF ELECTRICAL ENGINEERS, STEVENAGE, GB Inspec No. AN564074, XP002011881 * abstract * & NAUCHO-TEKH. INF. 2, no. 7, 1973, USSR, pages 25-27, ALEKPEROV V.P.: "An adaptation method for SDI"	4	

TECHNICAL FIELDS SEARCHED (Int.Cl.6)

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
BERLIN	29 August 1996	Deane, E

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

IPR2025-00200 / 722 Patent Page 166 of 1171 EX1002

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 889 421 A1

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
07.01.1999 Bulletin 1999/01

(51) Int Cl.[6]: **G06F 17/30**

(21) Application number: **98304651.7**

(22) Date of filing: **12.06.1998**

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: **12.06.1997 US 873975**

(71) Applicant: **Yahoo, Inc.**
Santa Clara, California 95051 (US)

(72) Inventors:
- **Nazem, Farzad**
  Redwood City, California 94061 (US)
- **Patel, Ashvinkumar P.**
  Milpitas, California 95035 (US)

(74) Representative:
Cross, Rupert Edward Blount et al
BOULT WADE TENNANT,
27 Furnival Street
London EC4A 1PQ (GB)

(54) Dynamic page generator

(57) An custom page server is provided with user preferences organized into templates stored in compact data structures and the live data used to fill the templates stored local to the page server which is handing user requests for custom pages. One process is executed on the page server for every request. The process is provided a user template for the user making the request, where the user template is either generated from user preferences or retrieved from a cache of recently used user templates. Each user process is provided access to a large region of shared memory which contains all of the live data needed to fill any user template. Typically, the pages served are news pages, giving the user a custom selection of stock quotes, news headlines, sports scores, weather, and the like. With the live data stored in a local, shared memory, any custom page can be built within the page server, eliminating the need to make requests from other servers for portions of the live data. While the shared memory might include RAM (random access memory) and disk storage, in many computer systems, it is faster to store all the live data in RAM.

EP 0 889 421 A1

Printed by Jouve, 75001 PARIS (FR)

Description

BACKGROUND OF THE INVENTION

The present invention relates to the field of customized information presentation. More specifically, one embodiment of the invention provides a custom page server which can quickly serve custom pages and is scalable to handle many users simultaneously.

Web servers for serving static documents ("Web pages") over the global Internet are known. While static documents are useful in many applications where the information to be presented to each requesting user is the same, some applications require customization to appeal to users. For example, in presenting news to users, custom pages present news which is more relevant to the requesting users than static pages. With static pages, a user will often have to scroll through many topics not of interest to that user to get to the information of interest. With custom pages, the information is filtered according to each user's interest.

Customizing a server response based on the requestor is known, however known systems do not scale well. One method of serving custom pages is to execute a script, such as a CGI (Common Gateway Interface) script, or other program to collect the information necessary to generate the custom page. For example, if the custom page is a news page containing stock quotes, sports scores and weather, the script might poll a quote server to obtain the quotes of interest, poll a sports score server to obtain the scores of interest and poll a weather server to obtain the weather. With this information, the server generates the custom page and returns it to the user. This approach is useful where there are not many requestors and where the attendant delay is acceptable to users. While it may be the case that current users are willing to wait while pages load in their browsers, growing impatience with waiting will turn users away from such servers, especially as use increases.

One approach to avoiding long waits is to transfer the custom information in non-real-time, so that the information is stored local to the user as it arrives and is presented to the user on request. A disadvantage of such a system is that the networks used by the user become clogged with data continually streaming to the user and require large amounts of local storage. Another disadvantage is that the locally stored information will become out of date as the server receives new data.

From the above it is seen that an improved system for delivering custom pages is needed.

SUMMARY OF THE INVENTION

The present invention provides an improved custom page server. In one embodiment, user preferences are organized into templates stored in compact data structures and the live data used to fill the templates is stored local to the page server which is handing user requests for custom pages. One process is executed on the page server for every request. The process is provided a user template for the user making the request, where the user template is either generated from user preferences or retrieved from a cache of recently used user templates. Each user process is provided access to a large region of shared memory which contains all of the live data needed to fill any user template. Typically, the pages served are news pages, giving the user a custom selection of stock quotes, news headlines, sports scores, weather, and the like. With the live data stored in a local, shared memory, any custom page can be built within the page server, eliminating the need to make requests from other servers for portions of the live data. While the shared memory might include RAM (random access memory) and disk storage, in many computer systems, it is faster to store all the live data in RAM.

If the volume of requests becomes too great for one page server to handle, the system is easily scaled by adding additional page servers. Each page server maintains its own copy of the live data in its shared memory, and needs to maintain only the user templates for the requests it is handling, so no communication between page servers is needed.

Examples of embodiments of the inventions herein may be realized by reference to the remaining portions of the specification and the attached drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram of a client-server architecture according to one embodiment of the present invention;

FIG. 2 is a schematic diagram showing how a user's custom page is generated from a user configuration, a global template and live data.

FIG. 3 is an illustration of a global front page template as might be used to generate user templates.

FIG. 4 is an illustration of a user template generated from the global front page template of FIG. 3 as might be used to generate a custom user page.

FIG. 5 is an illustration of a user page generated using the global template of FIG. 4.

FIG. 6 is an illustration of how intelligent defaults are selected.

Two appendices are included at the end of this description. Appendix A is a full listing of the user template shown in part in FIG.4. Appendix B is an HTML source code listing of the HTML page used to generate the browser display shown in FIG. 5.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

FIG. 1 shows a client-server system 100 which is used to display custom news pages. A custom news page is displayed on a browser 102 which obtains the page from a page server 104 via Internet 106. While only one browser 102 is shown, a typical system will have many browsers connecting and disconnecting to the system.

The art of client-server systems using HTTP (HyperText Transport Protocol) messaging or other protocols is well known and will not be addressed in detail here. Essentially, browser 102 makes a request for a particular page by specifying a Uniform Resource Locator ("URL") for the page. In the example shown in FIG. 1, the request is directed to the URL "http:/my.yahoo.com/". Normally, this URL is directed to the root directory of a machine named my.yahoo.com. As is the convention in Internet communications, browser 102 submits the domain portion ("my.yahoo.com") of the URL to a name server, such as name server 108, to determine an actual address for the page server 104. Name server 108 returns an IP (Internet Protocol) address to browser 102 directing it to a page server 104. Where multiple page servers 104 are used, name server 108 returns IP addresses in a round-robin fashion to distribute the load over multiple page servers. Alternatively, name server 108 might distribute the load more deterministic by tracking browser addresses and hashing the browser address to select a page server 104. It is deterministic in that any given browser always accesses the same page server 104. This allows for more efficient caching of user templates, since more cache hits are likely where a given browser always returns to one page server.

When a page server receives the URL for its root directory, it interprets that as a request for the user's custom summary page. The user is determined not from the URL, but from a "cookie" provided by browser 102 with the URL. Cookies are strings of data stored by browsers and sent along with any request to a URL having a domain associated with the cookie.

Page servers 104 obtain the live data from many disparate sources and reformat the data into a form suitable for use by the page server. Page servers 104 are coupled, via a network, to edit servers 112, which are used when a user changes his or her user template. The user templates are stored in a user configuration database 116 and are stored and provided to edit servers by a network appliance 114 written for this purpose. Network appliance 114 is a process tuned to quickly locate files in large directories (N400 files/directory) and return them to the edit servers or page servers. One embodiment of network appliance 114 is the F330 fault-tolerant scalable server supplied by Network Appliance, of Mountain View, California.

In a specific embodiment, page servers 104 are microcomputers running the Unix® operating system with 64 to 128 megabytes of shared memory, page servers 104 and edit servers 112 are coupled using TCP/IP (Transport Control Protocol/Internet Protocol) and the user configuration database 116 is a Unix file structure which stores each user configuration in a text file. The particular file used by a user is a combination of the user's user name and a hash result, to allow for quick access when many user configurations are stored. For example, the user configuration for summary "front" page for a user "ash802" might be stored at /de/13/y.ash802, where "de" and "13" are hash results of a hash of the user name "ash802".

FIG. 2 shows in more detail the generation of a custom page for a user, using a front page generator 200 and page server 104. Front page generator 200 generates a user template 202 from a global front page template 204 and a user configuration record 206. FIG. 3 shows an example of a global front page template. User configuration record 206 is a record selected from user configuration database 116. The record might have been obtained from a cache, but in the preferred embodiment, the records are not cached, the user templates are.

Page server 104 is shown comprising a page generator 210, a shared memory 212 for storing live data and a cache 214 for caching user templates such as user template 202. Page generator 210 generates a custom front page 218 from a user template and the live data stored in shared memory 212. Although not shown, custom pages other than the front page can be generated in a similar fashion. Using user templates and a shared memory for the live data, page server 104 can quickly build custom pages in response to a user request. Where the user template is cached, the page can be generated entirely within page server 104.

Shared memory 212 is organized as a set of memory mapped files. With memory mapped files, the operating system maintains the data in permanent storage, but permanently caches the files in shared memory 212. This allows for quick recovery should a page server crash, since all of the shared memory can simply be reloaded from the permanent storage. This is a great feature from a user convenience standpoint, since the user will never be faced with a situation where they have to wait for a server to rebuild a page for them by querying the various data providing servers, such as sports server 230, stock server 232 and news server 234. As should be apparent from this description and FIG. 2, page generator 210 can generate custom front page 218 much more quickly using shared memory 212 as compared with using servers 230, 232, 234 and page template 202. One reason for this is that the time it takes to

retrieve data from shared memory 212 does not appreciably increase relative to the bandwidth delay time when more data is retrieved. For example, if stock server 232 were queried for each individual stock quote, a page with fifty stock quotes might take ten times as long to generate as a page with five stock quotes.

One aspect of the present invention is the realization that every piece of information a person can request on a page is storable in a shared memory closely coupled to a page generator. For example, page server 104 shown in FIG. 2 can accommodate 65,000 different symbols for which quotes are provided. In one embodiment, all of the stock information for all 65,000 symbols is stored in a 13 to 14 megabytes section of the shared memory. Where shared memory 212 is a 64MB or 128MB memory, this leaves sufficient room for other data, such as news headlines, sport scores, and memory used by the operating system for each process running on page server 104. In some embodiments, shared memory 212 is large enough to also accommodate more than just news headlines. For example, news summaries (as described further in connection with FIG. 5) can be stored in shared memory 212 for quick access.

As shown in FIG. 2, the user's front page template 202 does not need to be generated each time, but rather is stored in cache 214. In a preferred embodiment, user templates are stored in cache 214 for long enough to be reused. Some users might choose to access their front page only infrequently, while others might choose to access their front page hourly. Since the pages are customized and dynamic, the user would see different information each time, but the same user template would be used each time. Of course, when the user edits his or her template, any cached copy of a user template is flushed. A garbage-collection process may also flush the cache of user pages which have been inactive for several days. In one implementation, cache 214 would accommodate 60,000 to 70,000 user templates. Where a particular page server is assigned on a random round robin basis, multiple page servers may cache their own copy of a given user template, but where a user is directed always to a particular server (except in the case where the particular server fails and a secondary server is used), that page server will be the only one which needs to cache that users user template. Even where the round robin name server scheme is used, some browsers may cache IP addresses, even longer than the specified "time to live" variable associated with the IP address, in order to save the time required to obtain an IP address each time. With such a browser, the user is effectively directed to the same page server each time and the server side of the page serving system does not need to direct users to particular page servers. With newer browsers, however, the "time to live" variable is honored and new requests are made for IP addresses after the "time to live" has expired. In these cases, if the assignment of a user to a single page server is desired, name server 108 (see FIG. 1) will use the user name from the provided cookie or the user's IP address to assign a page server based on a hash of the user name or IP address.

FIG. 3 is an illustration of global user template 204. Global user template 204 is an HTML (HyperText Markup Language) document with additional tags as placeholders for live data. Several placeholders 302 are shown in FIG. 3.

FIG. 4 is an illustration of user template 202 as might be generated from global user template 204 (see Fig. 3) and a user configuration record 206. A full listing of user template 202 is included herewith in Appendix A. User template 202 is determined by the user configuration and is independent of the live data, therefore it can be cached without needing to be updated, unless the user chooses to edit the configuration information. Preferably, the user templates are cached rather than the user configuration, to save a step and reduce the time to respond to a request for the page. Caching is more effective where the typical user makes several requests in a short time span and then doesn't make any requests for a long period of time.

Essentially, user template 202 contains the information about the user which does not change until the user changes his or her preferences. Of course, the system operator could choose to make changes to how the system operates, thus requiring changes to the user preferences and user templates. User template 202 is shown comprising internal variables such as a time zone and demographic information. The demographic information, on the second line in FIG. 4 is used for selection of an advertisement which will be part of the custom page. In this example, the advertisement is targeted by the demographic information in the user template ":M,85,95035,T,*" indicating that a suitable ad should be targeted to a male user, age 85, located in zip code 95035, etc. As shown, the portfolio section contains selected stock symbols, the scoreboard section contains selected team symbols, and the weather section contains selected weather cities/zip codes.

The selections of stock quote symbols, team scores, and weather cities are set by the user. In a preferred embodiment, intelligent defaults are selected by the system prior to user selection, so that users unfamiliar with the customization process will nonetheless be able to view non empty custom pages. This is described in further detail below in connection with FIG. 6.

FIG. 5 is an illustration of a user front page 218 returned by page server 104. User front page 218 as shown in FIG. 5 includes many elements, some of which are described here in further detail. User front page 218 is built according to a user template and live data. The user template specifies, for example which quotes are shown in the portfolio module, which cities are displayed in the weather module, etc. Each of the modules 504 can be customized by a user and moved about front page 218. The modules 504 are also reusable, in that any customized module which appears on multiple pages can be edited from any one of those pages and the edits will be reflected on each of the pages. Other custom pages for the user can be viewed by selecting one of the page buttons 502 appearing below the header.

IPR2025-00200 / 722 Patent Page 170 of 1171 EX1002

Other pages and utilities can be selected using the buttons 508 which are part of the header.

In addition to all of the live date shown in FIG. 5 being stored in the shared memory, summaries from each of the major news topics can also be stored in the shared memory and viewed by pressing on the news topic header, such as news topic header 506. As should be noted, the page generator can also intelligently display dates 510 customized for a particular user, using a time zone variable in the user template. This time zone variable is shown as the first line in user template 202 in FIG. 4. In addition to being able to modify each of the modules, in many cases the order of appearance of the modules is customizable. For example, the order of the various sections of user template 202 shown in FIG. 4 is not fixed.

The preference editing process can be initiated by the user pressing the appropriate edit button 512. As explained above, once the editing process is complete, the user template is flushed from the cache and regenerated. Since each of the news stories is essentially a static page linked to a headline shown in the news section, these can simply be linked to static pages on a news server.

Referring now to FIG. 6, an illustration of intelligent defaulting for populating a user template, and consequently a user summary page. As part of a registration process, a user indicates, among other things, his or her zip code. This zip code is used to locate an approximate longitude and latitude for the user using a zip code lookup table 602. This allows the user's location to be located on a map 604. Map 604 provides city boundaries and, with team location table 606, also provides locations for various sports teams which can be selected in a sports module. In selecting a default predetermined number of cities and sports teams for inclusion as initial selections for a particular user, a circle is drawn around the user and increased in diameter until the circle envelopes a suitable predetermined number of cities and sports teams. In this way, each user is guaranteed a default number of nearby teams and cities for sports and weather, respectively. While this assumes that the user is interested in only the teams nearest the user, the system can be arranged to provide intelligent defaults where geographic anomalies are known to exist. Geographic anomalies occur in communities which have more loyalty to distant teams than nearby teams, such as might occur when the distant team is much better than the nearby team or when the nearby team recently moved to a distant location. In any case, the user is allowed customize his or her pages beginning with the intelligent defaults selected.

Other intelligent defaults can be provided in other contexts. For example, the header of user front page 218 includes a button 508 labelled "myweb" which, when pressed, would lead the user to a custom listing of web sites. The initial defaults for that custom listing of web sites might be generated based on the keywords of interest to that user or based on the news topics, sports teams or weather cities selected by the user.

IPR2025-00200 / 722 Patent Page 171 of 1171 EX1002

Appendix A. Listing of a User Template

```
<!-- timezone:-8,PT 0 -->
<!-- ad:M,85,95035,T,* 792 -->
<!--
portfolio:Quotes,pf_1,1,^DJI,^NYA,^IXIC,^SPX,^XAX,YHOO,NSCP,IOM,NSCP,YHOO
2836 -->
<!--
scoreboard:NCAAFSSC,NHLSAN,MLSSAN,NCAAFSSS,ALOAK,NBAGSW,NFLOAK,NCAAFCCD,NF
LSFO,NLSFO 3803 -->
<!-- weather:f,30901,uk_londo,94601,95101 4368 -->
<!--
mode_bar:"FRONT_PAGE",.hier=News+Summary%3aEdit&.done=http://my.yahoo.com/
news/summary.html?v 4538 -->
<!-- summary2:3,rt,rw,z0000,mlb,re,vf 4597 -->
<!-- motd:us_motn.html 4696 -->
<html>
<head>
<title>My Yahoo! news summary for ash802</title>
</head>
<body>

<center>
<map name="top"><area shape=rect ccords="0,0,45,66"
href="/news/summary.html?v"><area shape=rect coorcs="45,0,90,66"
href="/internet/directory.html?v"><area shape=rect coords="90,0,135,66"
href="/contacts/white_pages.html?v"><area shape=rect coords="135,0,346,66"
href="/?myHome"><area shape=rect coords="346,0,389,66"
href="/help/news/summary.html"><area shape=rect coords="389,0,431,66"
href="http://edit.my.yahoo.com/config/eval_profile"><area shape=rect
coords="431,0,477,66" href="http://edit.my.yahoo.com/config/toffly">
</map>
<center><a href="/bin/top_menu"><img border=0 height=62 width=477 ismap
usemap="#top" src="http://www.yahoo.com/images/my/banner/news1.gif"
alt="My Yahoo!"></a></center>

<p>
<center><form action="http://search.my.yahoo.com/search/my"> <font
size="-2">[ <a
href="http://edit.my.yahoo.com/config/set_attr?attr=sloc&value=b&.done=htt
p://my.yahoo.com/news/summary.html?v">move to bottom</a> ]  </font><input
size=25 name=p> <font size="-1">  <input type=submit
value=Search><input type=hidden name=n value="25">   <a
href="http://search.my.yahoo.com/search/my/options">Options</a></font></fo
rm></center><p>
```

```
<center>
<table border=0 cellspacing=0 cellpadding=2>
<tr>
<td align=center><font
color="#000000"><b>Sections:</b>  </font></td>
<td align=center bgcolor="#dcdcdc"><font
color="#000000"><b>Front Page</b></font></td>
<td> | </td>
<td align=center><a href="/news/business/summary.html?v">Business</a></td>
<td> | </td>
<td align=center><a
href="/news/portfolio_summary.html?v">Portfolios</a></td>
<td> | </td>
<td align=center><a
href="/news/entertainment/summary.html?v">Entertainment</a></td>
<td> | </td>
<td align=center><a href="/news/sports/summary.html?v">Sports</a></td>
<td> | </td>
<td align=center><a
href="/news/technology/summary.html?v">Technology</a></td>
</tr>
</table>
</center>
</center>
<table border=1 cellpadding=4 cellspacing=0 width="100%">
<tr><td align=center valign=top width="1%">
<a
href="http://edit.my.yahoo.com/config/eval_left?mode=left_nsum&.done=http%
3a%2f%2fmy%2eyahoo%2ecom%2fnews%2fsummary%2ehtml%3fv"><img border=0
src="http://www.yahoo.com/images/my/left.gif" alt="edit left"></a>
<table border=0 cellpadding=1 cellspacing=0 width="100%">
<tr bgcolor="#dcdcdc"><td colspan=4 width="100%"><table border=0
cellpadding=2 cellspacing=0 width="100%"><tr><td width="100%"><b><font
size="+1">P<font size="+0">ORTFOLIOS</font></font></b></td><td
align=right><a
href="http://edit.my.yahoo.com/config/eval_quotes?.done=http://my.yahoo.co
m/news/summary.html?v"><img align=absmiddle border=0 height=13 width=25
src="http://www.yahoo.com/images/my/personalize.gif"
alt="edit"></a></td></tr></table></td></tr>
<tr><td align=center colspan=4><form method=get
action="http://quote.yahoo.com/quotes"><font size="-1"><input type=text
size=10 name=symbols><input type=submit value="Get Quotes"><br><font
size="0">quotes delayed 20 minutes - <a
href="/fin_disclaimer.html">disclaimer</a><br>click on symbol for detailed
quote + news<br><b>*</b> indicates new news during last
24hrs<font></form></td></tr></table>
```


IPR2025-00200 / 722 Patent Page 173 of 1171 EX1002

```
<hr size=0><table border=0 cellpadding=1 cellspacing=0 width="100%">
<tr bgcolor="#dcdcdc"><td colspan=5 width="100%"><table border=0
cellpadding=2 cellspacing=0 width="100%"><tr><td width="100%"><a
href="/sum/?http://sports.yahoo.com/"><b><font size="+1">S<font
size="+0">COREBOARD</font></font></b></a></td><td align=right><a
href="http://edit.my.yahoo.com/config/set_sports?.done=http://my.yahoo.com
/news/summary.html?v"><img align=absmiddle border=0 height=13 width=25
src="http://www.yahoo.com/images/my/personalize.gif"
alt="edit"></a></td></tr></table></td></tr>

<hr size=0><table border=0 cellpadding=2 cellspacing=0 width="100%">
<tr bgcolor="#dcdcdc"><td colspan=3 width="100%"><table border=0
cellpadding=2 cellspacing=0 width="100%"><tr><td width="100%"><a
href="/my/?http://weather.yahoo.com/"><b><font size="+1">W<font
size="+0">EATHER</font></font></b></a></td><td align=right><a
href="http://edit.my.yahoo.com/config/set_wni?.done=http://my.yahoo.com/ne
ws/summary.html?v"><img align=absmiddle border=0 height=13 width=25
src="http://www.yahoo.com/images/my/personalize.gif"
alt="edit"></a></td></tr></table></td></tr>
<tr><td align=center colspan=3><font size=0>click on city for extended
forecast</font></td></tr>
</table>

<hr size=0>
<p></td>
<td align=center valign=top width="100%">

<table border=1 cellpadding=2 cellspacing=0 width="100%">
</table>


</td></tr>
</table>

<center>
<table border=1 cellpadding=2 cellspacing=0 width="100%">

</table>
</center>

<p>
<hr>
<center>
<table border=0 cellpadding=0 cellspacing=0 width="100%">
<tr>
<td align=left valign=top><table border=0 cellpadding=0 cellspacing=0>
```

```
<tr><td align=left valign=bottom><a href="/my/?http://www.yahoo.com/"><img
border=0 height=47 width=47 src="http://www.yahoo.com/images/my/yahoo.gif"
alt="Yahoo!"></a></td>
<td>   </td>
<td align=left valign=bottom width="100%"><font size="-1"><i>Copyright
&copy; 1994-1997 Yahoo! Inc. All rights reserved.</i><br>Portions
copyright &copy; 1997 Reuters Ltd. All rights reserved.<br>
Portions copyright &copy; 1997 SportsTicker Enterprises LP. All rights
reserved.<br>
Portions copyright &copy; 1997 Weathernews Inc. All rights reserved.<br>
Portions copyright &copy; 1997 Business Wire. All rights reserved.<br>
Portions copyright &copy; 1997 PR Newswire. All rights reserved.<br>
Questions, comments, or suggestions? Send us <a
href="http://my.yahoo.com/feedback.html">feedback</a>.</font></td>
</tr>
</table>
</td>
<td>   </td>
<td align=right><a href="/bin/logout_menu"><img border=0 height=68
width=100 ismap usemap="#logout"
src="http://www.yahoo.com/images/my/logout.gif" alt="Change
User"></a></td>
</tr>
</table>
</center>
<map name=logout>
<area coords="0,0,50,68" href="/my/?http://www.yahoo.com/">
<area coords="51,0,100,68"
href="http://edit.my.yahoo.com/config/login?logout=1">
</map>

</body>
</html>
```

Appendix B. Listing of a User Page

```
<html>
<head>
<title>My Yahoo! news summary for ash802</title>
</head>
<body>

<center>
<map name="top"><area shape=rect coords="0,0,45,66"
href="/news/summary.html?v"><area shape=rect coords="45,0,90,66"
href="/internet/directory.html?v"><area shape=rect coords="90,0,135,66"
href="/contacts/white_pages.html?v"><area shape=rect coords="135,0,346,66"
href="/?myHome"><area shape=rect coords="346,0,389,66"
href="/help/news/summary.html"><area shape=rect coords="389,0,431,66"
href="http://edit.my.yahoo.com/config/eval_profile"><area shape=rect
coords="431,0,477,66" href="http://edit.my.yahoo.com/config/toffly">
</map>
<center><a href="/bin/top_menu"><img border=0 height=62 width=477 ismap
usemap="#top" src="http://www.yahoo.com/images/my/banner/news1.gif" alt="My
Yahoo!"></a></center>
<font size="-5"><p></font><!-- SpaceID=0 noad -->

<font size="-5"><p></font>
<p>
<center><form action="http://search.my.yahoo.com/search/my"> <font size="-2">[ <a
href="http://ecit.my.yahoo.com/config/set_attr?attr=sloc&value=b&.done=http://my.y
ahoo.com/news/summary.html?v">move to bottom</a> ]  </font><input size=25 name=p>
<font size="-1">  <input type=submit value=Search><input type=hidden name=n
value="25">   <a href="http://search.my.yahoo.com/search/my/options"
>Options</a></font></form></center><p>
<center>
<table border=0 cellspacing=0 cellpadding=2>
<tr>
<td align=center><font color="#000000"><b>Sections:</b>  </font></td>
<td align=center bgcolor="#dcdcdc"><font
color="#000000"><b>Front Page</b></font></td>
<td> | </td>
<td align=center><a href="/news/business/summary.html?v">Business</a></td>
<td> | </td>
<td align=center><a href="/news/portfolio_summary.html?v">Portfolios</a></td>
<td> | </td>
<td align=center><a
href="/news/entertainment/summary.html?v">Entertainment</a></td>
<td> | </td>
<td align=center><a href="/news/sports/summary.html?v">Sports</a></td>
<td> | </td>
<td align=center><a href="/news/technology/summary.html?v">Technology</a></td>
</tr>
</table>
```


IPR2025-00200 / 722 Patent Page 176 of 1171 EX1002

```
</center>

</center>

<table border=1 cellpadding=4 cellspacing=0 width="100%">
<tr><td align=center valign=top width="1%">
<a
href="http://edit.my.yahoo.com/config/eval_left?mode=left_nsum&.done=http%3a%2f%2f
my%2eyahoo%2ecom%2fnews%2fsummary%2ehtml%3fv"><img border=0
src="http://www.yahoo.com/images/my/left.gif" alt="edit left"></a>
<table border=0 cellpadding=1 cellspacing=0 width="100%">
<tr bgcolor="#dcdcdc"><td colspan=4 width="100%"><table border=0 cellpadding=2
cellspacing=0 width="100%"><tr><td width="100%"><b><font size="+1">P<font
size="+0">ORTFOLIOS</font></font></b></td><td align=right><a
href="http://edit.my.yahoo.com/config/eval_quotes?.done=http://my.yahoo.com/news/s
ummary.html?v"><img align=absmiddle border=0 height=13 width=25
src="http://www.yahoo.com/images/my/personalize.gif"
alt="edit"></a></td></tr></table></td></tr>
<tr><td valign=top width="10%"><nobr><a
href="http://edit.my.yahoo.com/config/set_portf_expand?port=Quotes&expand=0&.done=
http://my.yahoo.com/news/summary.html?v"><img align=absmiddle border=0 height=11
width=11 src="http://www.yahoo.com/images/my/arrowdown.gif"
alt="edit"></a></td><td valign=top width="30%"><nobr><b><a
href="http://quote.yahoo.com/portfolio?view&key=pf_1">Quotes</a></b></nobr></td><t
d width="30%"></td><td width="30%"><big></td></tr>
<tr><td align=left valign=top width="10%"><nobr><font size="-
1"> <br> <br> <br> <br> <br>*<br>*<br>*<br>*<br>*<br></fo
nt></nobr></td><td valign=top width="30%"><nobr><font size="-1"><a
href="http://quote.yahoo.com/quotes?SYMBOLS=^DJI&detailed=t">DJIA</a><br>
<a href="http://quote.yahoo.com/quotes?SYMBOLS=^NYA&detailed=t">NYSE</a><br>
<a href="http://quote.yahoo.com/quotes?SYMBOLS=^IXIC&detailed=t">NASDAQ</a><br>
<a href="http://quote.yahoo.com/quotes?SYMBOLS=^SPX&detailed=t">S&P 500</a><br>
<a href="http://quote.yahoo.com/quotes?SYMBOLS=^XAX&detailed=t">AMEX</a><br>
<a href="http://quote.yahoo.com/quotes?SYMBOLS=YHOO&detailed=t">YHOO</a><br>
<a href="http://quote.yahoo.com/quotes?SYMBOLS=NSCP&detailed=t">NSCP</a><br>
<a href="http://quote.yahoo.com/quotes?SYMBOLS=IOM&detailed=t">IOM</a><br>
<a href="http://quote.yahoo.com/quotes?SYMBOLS=NSCP&detailed=t">NSCP</a><br>
<a href="http://quote.yahoo.com/quotes?SYMBOLS=YHOO&detailed=t">YHOO</a><br>
</font></nobr></td><td align=right valign=top width="30%"><nobr><font size="-
1">7312.15<br>442.36<br>1384.91<br>845.48<br>609.02<br>32 1/2<br>29 1/4<br>19
1/4<br>29 1/4<br>32 1/2<br></font></nobr></td><td align=right valign=top
width="30%"><nobr><font size="-1"><font size="-1"
color="#007f00">+22.75</font><br><font size="-1"
color="#007f00">+0.82</font><br><font size="-1" color="#7f0000">-
19.88</font><br><font size="-1" color="#7f0000">-0.88</font><br><font size="-1"
color="#007f00">+1.85</font><br><font size="-1" color="#7f0000">-
5/8</font><br><font size="-1" color="#7f0000">-5/8</font><br><font size="-1"
color="#007f00">+1/2</font><br><font size="-1" color="#7f0000">-
5/8</font><br><font size="-1" color="#7f0000">-
5/8</font><br></font></nobr></td></tr>
<tr><td align=center colspan=4><form method=get
```

```
action="http://quote.yahoo.com/quotes"><font size="-1"><input type=text size=10
name=symbols><input type=submit value="Get Quotes"><br><font size="0">quotes
delayed 20 minutes - <a href="/fin_disclaimer.html">disclaimer</a><br>click on
symbol for detailed quote + news<br><b>*</b> indicates new news during last
24hrs<font></form></td></tr></table>
<hr size=0><table border=0 cellpadding=1 cellspacing=0 width="100%">
<tr bgcolor="#dcdcdc"><td colspan=5 width="100%"><table border=0 cellpadding=2
cellspacing=0 width="100%"><tr><td width="100%"><a
href="/sum/?http://sports.yahoo.com/"><b><font size="+1">S<font
size="+0">COREBOARD</font></font></b></a></td><td align=right><a
href="http://edit.my.yahoo.com/config/set_sports?.done=http://my.yahoo.com/news/su
mmary.html?v"><img align=absmiddle border=0 height=13 width=25
src="http://www.yahoo.com/images/my/personalize.gif"
alt="edit"></a></td></tr></table></td></tr>
<tr><td colspan=5><b><font size="-1">TODAY</font></b></td></tr>
<tr><td valign=top><font size=-1><a
href="http://sports.yahoo.com/al/">AL</a></font></td><td valign=top><font size=-
1>Detroit<br><b>Oakland</b></font></td><td align=right valign=top><font size=-
1>8<br>9</font></td><td valign=top> </td><td valign=top><font size=-1><a
href="http://sports.yahoo.com/al/rcps/970603/detoak.html">F</a></font></td></tr>
<tr><td valign=top><font size=-1><a
href="http://sports.yahoo.com/nl/">NL</a></font></td><td valign=top><font size=-
1>San Francisco<br><b>Florida</b></font></td><td align=right valign=top><font
size=-1>9<br>1</font></td><td valign=top> </td><td valign=top><font size=-
1><a
href="http://sports.yahoo.com/nl/rcps/970603/sfofla.html">F</a></font></td></tr>
<tr><td colspan=5><b><font size="-1">YESTERDAY</font></b></td></tr>
<tr><td valign=top><font size=-1><a
href="http://sports.yahoo.com/al/">AL</a></font></td><td valign=top><font size=-
1>Detroit<br><b>Oakland</b></font></td><td align=right valign=top><font size=-
1>8<br>7</font></td><td valign=top> </td><td valign=top><font size=-1><a
href="http://sports.yahoo.com/al/rcps/970602/detoak.html">F</a></font></td></tr>
<tr><td valign=top><font size=-1><a
href="http://sports.yahoo.com/nl/">NL</a></font></td><td valign=top><font size=-
1>San Francisco<br><b>Florida</b></font></td><td align=right valign=top><font
size=-1>2<br>4</font></td><td valign=top> </td><td valign=top><font size=-
1><a
href="http://sports.yahoo.com/nl/rcps/970602/sfofla.html">F</a></font></td></tr>
</table>

<hr size=0><table border=0 cellpadding=2 cellspacing=0 width="100%">
<tr bgcolor="#dcdcdc"><td colspan=3 width="100%"><table border=0 cellpadding=2
cellspacing=0 width="100%"><tr><td width="100%"><a
href="/my/?http://weather.yahoo.com/"><b><font size="+1">W<font
size="+0">EATHER</font></font></b></a></td><td align=right><a
href="http://edit.my.yahoo.com/config/set_wni?.done=http://my.yahoo.com/news/summa
ry.html?v"><img align=absmiddle border=0 height=13 width=25
src="http://www.yahoo.com/images/my/personalize.gif"
alt="edit"></a></td></tr></table></td></tr>
<tr><td><a href="http://weather.yahoo.com/forecast/Augusta_GA_US_f.html"><font
size="-1">Augusta, GA</font></a></td><td align=right><font size="-1">63...78
```

```
F</font></td><td align=right><font size="-1"><img height=24 width=32
src="http://www.yahoo.com/images/my/wni/87.gif" alt="Showers"></font></td></tr>
<tr><td><a href="http://weather.yahoo.com/forecast/London_UK_f.html"><font size="-
1">London, UK</font></a></td><td align=right><font size="-1">50...73
F</font></td><td align=right><font size="-1"><img height=24 width=32
src="http://www.yahoo.com/images/my/wni/87.gif" alt="Showers"></font></td></tr>
<tr><td><a href="http://weather.yahoo.com/forecast/Oakland_CA_US_f.html"><font
size="-1">Oakland, CA</font></a></td><td align=right><font size="-1">59...70
F</font></td><td align=right><font size="-1"><img height=24 width=32
src="http://www.yahoo.com/images/my/wni/66.gif" alt="Partly
Cloudy"></font></td></tr>
<tr><td><a href="http://weather.yahoo.com/forecast/San_Jose_CA_US_f.html"><font
size="-1">San Jose, CA</font></a></td><td align=right><font size="-1">54...75
F</font></td><td align=right><font size="-1"><img height=24 width=32
src="http://www.yahoo.com/images/my/wni/82.gif" alt="Rain"></font></td></tr>
<tr><td align=center colspan=3><font size=0>click on city for extended
forecast</font></td></tr>
</table>

<hr size=0>
<p></td>
<td align=center valign=top width="100%">
<center>
<table border=1 cellpadding=4 cellspacing=0 width="100%">
<tr>
<td bgcolor="#000000">
<table border=0 cellpadding=0 cellspacing=0 width="100%"><tr>
<td align=left valign=bottom><font color="#ffffff"
size="+1"><nobr><b>F</b></nobr></font><font
color="#ffffff"><b><nobr>RONT</nobr>  <nobr>PAGE</nobr></b> - Jun 3
7:40pm</font></td>
<td align=right><a
href="http://edit.my.yahoo.com/config/edit_news?.hier=News+Summary%3aEdit&.done=ht
tp://my.yahoo.com/news/summary.html?v"><img align=right border=0 height=15
width=66 vspace=2 src="http://www.yahoo.com/images/my/edit_topics.gif" alt="edit
topics"></a></td></tr>
</table>
</td>
</tr>
</table>
</center>

<table border=1 cellpadding=2 cellspacing=0 width="100%">
<tr><td bgcolor="#dcdcdc" colspan=2><a href="/news/top_stories.html?v"><b><font
size="+1">B</font>reaking News</b></a>  <i>Jun 3 5:03pm</i></td></tr>
<tr><td colspan=2><ul>
<li><a href="http://www.yahoo.com/headlines/970603/news/stories/bombing_61.html"
>Judge Moves to Avoid McVeigh 'Lynching'</a>
<li><a href=
"http://www.yahoo.com/headlines/970603/news/stories/bombingappeals_1.html">McVeigh
's Expected Appeals May Last Years</a>
```

```
<li><a
href="http://www.yahoo.com/headlines/970603/news/stories/bombingpenalty_1.html">Ex
perts: Plea for Mercy is McVeigh's Best Hope</a>
</ul></td></tr>
<tr><td bgcolor="#dcdcdc" colspan=2><a href="/news/world.html?v"><b><font
size="+1">W</font>orld</b></a>  <i>Jun 3 2:09pm</i></td></tr>
<tr><td colspan=2><ul>
<li><a
href="http://www.yahoo.com/headlines/970603/international/stories/canada_6.html">C
anadian Liberals Face Thorny Issues</a>
<li><a
href="http://www.yahoo.com/headlines/970603/international/stories/nato_10.html">U.
S. Advocates Ready for NATO Expansion Fight</a>
<li><a
href="http://www.yahoo.com/headlines/970603/international/stories/leone_11.html">U
.S. Ferries More Foreigners from Sierra Leone</a>
</ul></td></tr>
<tr><td bgcolor="#dcdcdc" colspan=2><a
href="http://biz.yahoo.com/news/z0000.html"><b><font size="+1">U</font>.S. Stock
News and Report</b></a>  <i></i></td></tr>
<tr><td colspan=2><ul>
<li><a href="http://biz.yahoo.com/finance/97/06/03/crdm_y001_1.html">Cardima Inc
IPO range reduced - underwriter</a>
<li><a href="http://biz.yahoo.com/finance/97/06/03/luk_wtsc_1.html">AMEX to trade
Leucadia, West TeleServices options</a>
<li><a href="http://biz.yahoo.com/finance/97/06/03/acv_acva_1.html">CBOE to trade
options on Alberto-Culver Class A</a>
</ul></td></tr>
<tr><td bgcolor="#dcdcdc" colspan=2><a href="/news/sports/mlb.html?v"><b><font
size="+1">M</font>ajor League Baseball</b></a>  <i>Jun 3 5:15pm</i></td></tr>
<tr><td colspan=2><ul>
<li><a href="http://sports.yahoo.com/mlb/lgns/970603/draftlist.html">1997 baseball
free agent draft first round selections</a>
<li><a href="http://sports.yahoo.com/nl/lgns/970603/nlstars.html">Mike Piazza
leads balloting in NL All-Star voting</a>
<li><a href="http://sports.yahoo.com/nl/lgns/970603/nlvotes.html">National League
All-Star voting results</a>
</ul></td></tr>
<tr><td bgcolor="#dcdcdc" colspan=2><a
href="/news/entertainment/top_stories.html?v"><b><font
size="+1">E</font>ntertainment Top Stories</b></a>  <i>Jun 3
6:35pm</i></td></tr>
<tr><td colspan=2><ul>
<li><a href="http://www.yahoo.com/headlines/970603
/entertainment/summary_1.html#headline_1">Clinton Reflects on Music </a>
<li><a href="http://www.yahoo.com/headlines/970603
/entertainment/summary_1.html#headline_2">Gifford: Marriage is Solid </a>
<li><a href="http://www.yahoo.com/headlines/970603
/entertainment/summary_1.html#headline_3">Rockabilly Legend Hospitalized </a>
</ul></td></tr>
<tr><td bgcolor="#dcdcdc" colspan=2><a
```

```
href="/news/entertainment/movies.html?v"><b><font size="+1">M</font>ovies +
Film</b></a>  <i>Jun 3 10:13am</i></td></tr>
<tr><td colspan=2><ul>
<li><a href="http://www.yahoo.com/headlines/970603/entertainment
/stories/film_cage_1.html">FEATURE: Nicolas Cage Asks, Where Are The Heroes?</a>
<li><a href="http://www.yahoo.com/headlines/970603/entertainment
/stories/film_cassavetes_1.html">Nick Cassavetes Updates Father's 'Bookie'</a>
<li><a href="http://www.yahoo.com/headlines/970603/entertainment
/stories/film_boxoffice_9.html">'Lost World' Props Up Slow Box Office</a>
</ul></td></tr>
</table>

</td></tr>
</table>

<center>
<table border=1 cellpadding=2 cellspacing=0 width="100%">
<tr bgcolor="#f0f8ff">
<td align=center colspan=2><table border=0 cellpadding=2 cellspacing=0
width="100%">
<tr><td align=left width="60%"><font size="-1"><b>TIP </b> clicking on the My
Yahoo! logo at the top will reload and update your default page.</font></td><td
align=center width="28%" nowrap><font size="-1"><a
href="http://bwguide.yahoo.com/bwguide/start.html">Yahoo! BackWeb Channel
Guide</a><br><a href="http://www.yahoo.com/docs/info/outreach/">Yahoo! Community
Relations</a></font><br><br><font size="-1"><a
href="/motn/?http://local.yahoo.com/local/">Get Local</a> &#183; <a
href="/motn/?http://my.yahoo.com/ticker.html">News Ticker</a> &#183; <a
href="/motn/?http://fool.yahoo.com/fool">Motley Fool</a></font></td></tr>
</table></td>
</tr>
</table>
</center>
<p>
<hr>
<center>
<table border=0 cellpadding=0 cellspacing=0 width="100%">
<tr>
<td align=left valign=top><table border=0 cellpadding=0 cellspacing=0>
<tr><td align=left valign=bottom><a href="/my/?http://www.yahoo.com/"><img
border=0 height=47 width=47 src="http://www.yahoo.com/images/my/yahoo.gif"
alt="Yahoo!"></a></td>
<td>   </td>
<td align=left valign=bottom width="100%"><font size="-1"><i>Copyright &copy;
1994-1997 Yahoo! Inc. All rights reserved.</i><br>Portions copyright &copy; 1997
Reuters Ltd. All rights reserved.<br>
Portions copyright &copy; 1997 SportsTicker Enterprises LP. All rights
reserved.<br>
Portions copyright &copy; 1997 Weathernews Inc. All rights reserved.<br>
Portions copyright &copy; 1997 Business Wire. All rights reserved.<br>
Portions copyright &copy; 1997 PR Newswire. All rights reserved.<br>
```

```
Questions, comments, or suggestions? Send us <a
href="http://my.yahoo.com/feedback.html">feedback</a>.</font></td>
</tr>
</table>
</td>
<td>   </td>
<td align=right><a href="/bin/logout_menu"><img border=0 height=68 width=100 ismap
usemap="#logout" src="http://www.yahoo.com/images/my/logout.gif" alt="Change
User"></a></td>
</tr>
</table>
</center>
<map name=logout>
<area coords="0,0,50,68" href="/my/?http://www.yahoo.com/">
<area coords="51,0,100,68" href="http://edit.my.yahoo.com/config/login?logout=1">
</map>

</body>
</html>
```

Claims

1. A server system for serving a customized information page to a user at a browser on request, comprising:

   a database of user configurations;
   means for obtaining information from the user referencing a record in the database of user configurations, the record being a record for the user;
   means for generating a template based on the record for the user;
   a plurality of user processes, one process per user accessing the server system;
   a shared memory;
   a data structure within the shared memory for storing the dynamic information needed to generate the customized information page; and
   response means for responding to a request with an information page containing dynamic information from the data structure formatted according to the template generated for the user making the request.

2. The server system of claim 1, wherein the plurality of user processes execute on a plurality of servers, the server system further comprising a means for assigning one of the plurality of servers to users who send requests to the server system but do not already have an associated user process.

3. The server system of claim 2, wherein the means for assigning servers is a means for assigning a specific server to a particular user, such that the particular user is assigned the specific server for each request.

4. The server system of claim 3, further comprising a cache for caching templates for recently serviced users on each of the specific servers assigned to particular users.

5. The server system of any of claims 1 to 4, further comprising a cache for caching templates for recently serviced users.

6. The server system of any of claims 1 to 5, further comprising means for generating default user configurations based on user demographic information.

7. The server system of claim 6, further comprising means for determining a default list of cities for a weather report and sports teams for sports reporting based on user demographic information.

IPR2025-00200 / 722 Patent Page 182 of 1171 EX1002

8. The server system of claim 7, wherein the means for determining the default list of cities and default lists of sports teams uses user postal code information, the server system further comprising:

means for translating postal code information to user geographic position;
means for comparing the user geographic position to geographic positions assigned to each city and sports team; and
means for determining a threshold distance from the user geographic position which is greater than or equal to a distance to a predetermined nonzero number of cities and a predetermined nonzero number of sports team geographic positions.

9. A method of serving custom news pages, comprising the steps of:

collecting current security quotes for each of a plurality of selectable securities;
collecting current sports news and scores for each of a plurality of selectable sports events;
collecting current news headlines which reference news articles;
formatting the current security quotes, current sports news and current news headlines into a data structure stored in a shared memory;
accepting a request for a custom page from a user;
in response to the request, reading a user template for the user from a template database, wherein the user template describes a layout of a custom news page customized for the user;
populating a page based on the read user template and the current data stored in the data structure in the shared memory to form a custom news page; and
providing the custom news page to the user in realtime response to the request.

IPR2025-00200 / 722 Patent Page 183 of 1171 EX1002

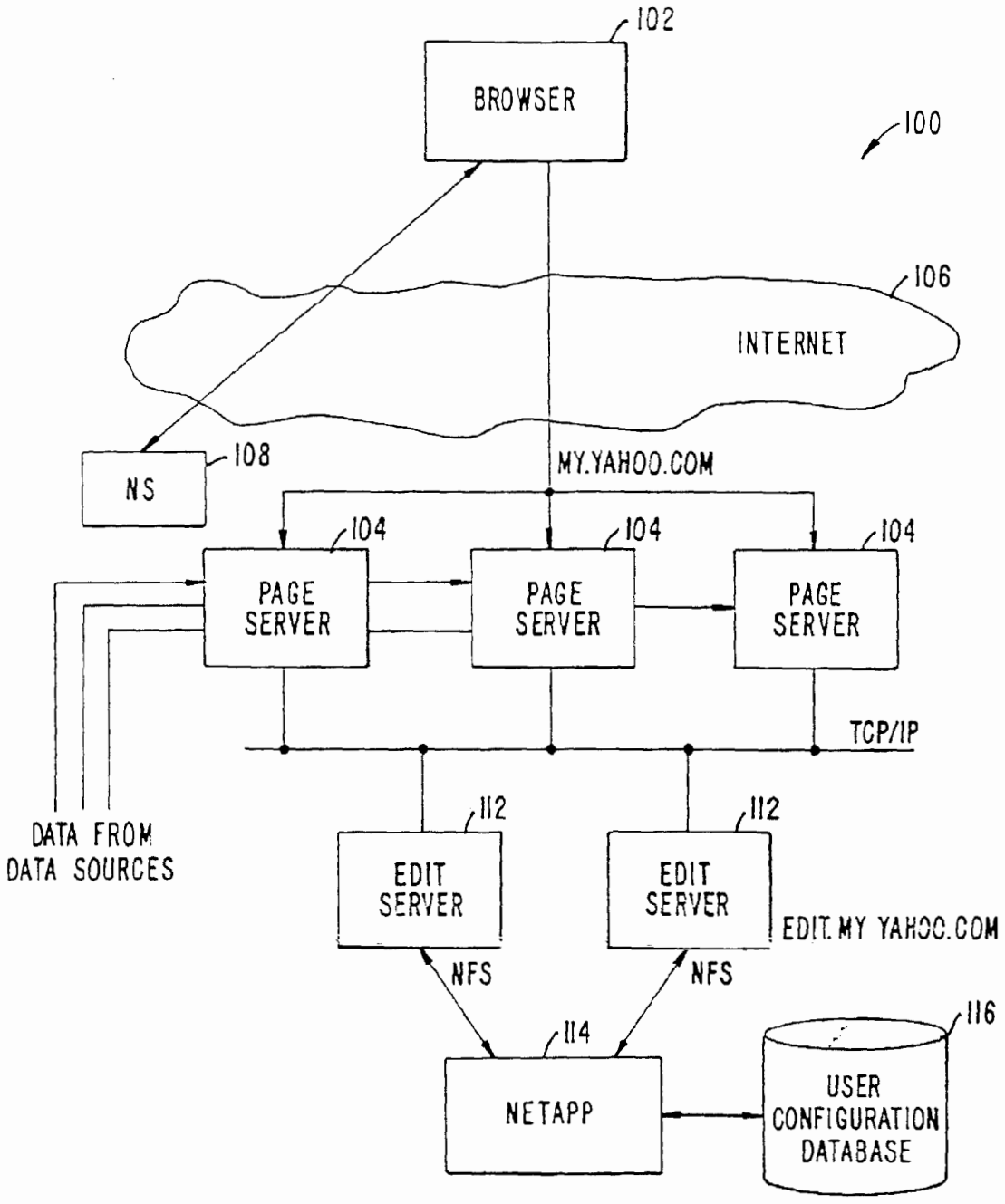

102
BROWSER
100
106
INTERNET
108
NS
MY.YAHOO.COM
104
104
104
PAGE SERVER
PAGE SERVER
PAGE SERVER
TCP/IP
DATA FROM DATA SOURCES
112
112
EDIT SERVER
EDIT SERVER
EDIT.MY YAHOO.COM
NFS
NFS
114
116
NETAPP
USER CONFIGURATION DATABASE

FIG. 1.

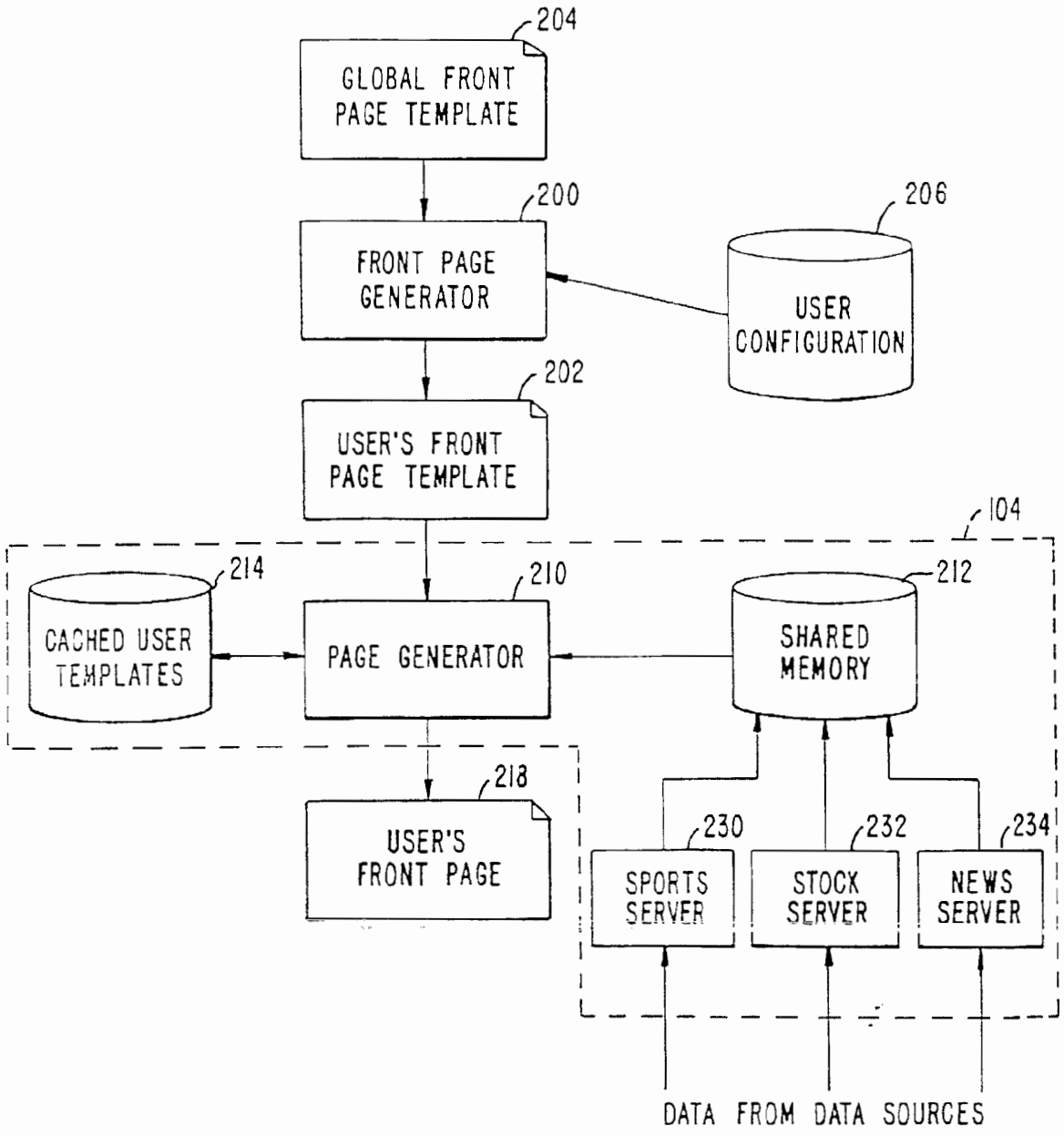

204
GLOBAL FRONT PAGE TEMPLATE
200
FRONT PAGE GENERATOR
206
USER CONFIGURATION
202
USER'S FRONT PAGE TEMPLATE
104
214
CACHED USER TEMPLATES
210
PAGE GENERATOR
212
SHARED MEMORY
218
USER'S FRONT PAGE
230
SPORTS SERVER
232
STOCK SERVER
234
NEWS SERVER
DATA FROM DATA SOURCES

FIG. 2.

```
<html>
<head>
<title>My Yahoo! news summary for <!-- login --></title>
</head>
<body>

<center>
<!--banner:sum -->

<!-- ad -->
<!-- nav_bar -->  302
</center>

<table border=1 cellpadding=4 cellspacing=0 width=100%">
<tr><td align=center valign=top width="1%">
<!-- leftside:nsum -->  302

<p></td>
<td align=center valign=top width="100%">
<!-- mode bar:"FRONT_PAGE" -->
<!-- channel:nsum -->  302

</td></tr>
</table>

<center>
<table border=1 cellpadding=2 cellspacing=0 width=100%">
<!-- motd:motn.html -->
</table>
</center>
<!-- search -->

<!-- copyright:sum -->

</body>
</html>
```

204

FIG. 3.

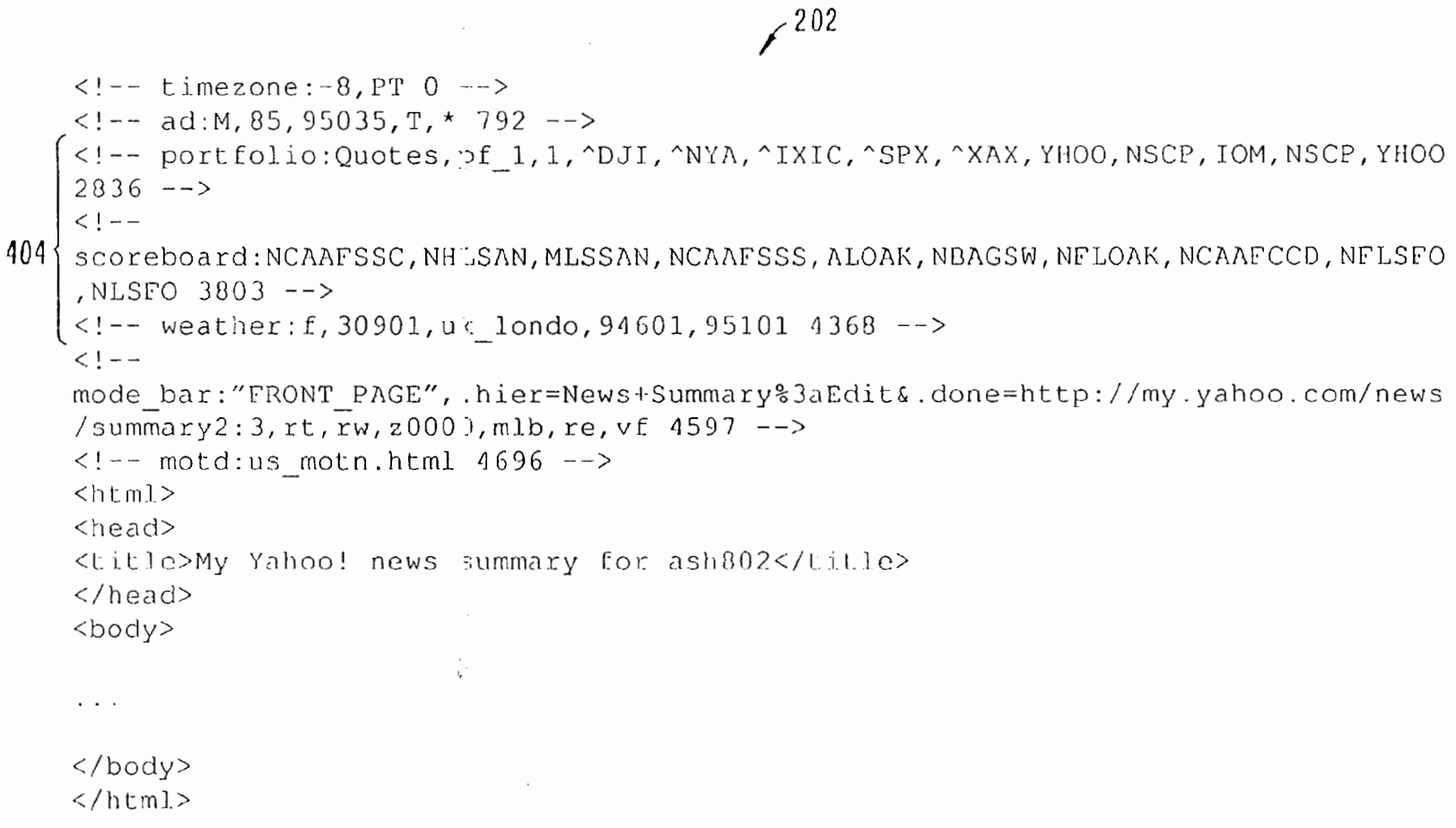

202
<!-- timezone:-8,PT 0 -->
<!-- ad:M,85,95035,T,* 792 -->
<!-- portfolio:Quotes,pf_1,1,^DJI,^NYA,^IXIC,^SPX,^XAX,YHOO,NSCP,IOM,NSCP,YHOO
2836 -->
<!--
404
scoreboard:NCAAFSSC,NHLSAN,MLSSAN,NCAAFSSS,ALOAK,NBAGSW,NFLOAK,NCAAFCCD,NFLSFO
,NLSFO 3803 -->
<!-- weather:f,30901,uk_londo,94601,95101 4368 -->
<!--
mode_bar:"FRONT_PAGE",.hier=News+Summary%3aEdit&.done=http://my.yahoo.com/news
/summary2:3,rt,rw,z000),mlb,re,vf 4597 -->
<!-- motd:us_motn.html 4696 -->
<html>
<head>
<title>My Yahoo! news summary for ash802</title>
</head>
<body>
...
</body>
</html>

FIG. 4.

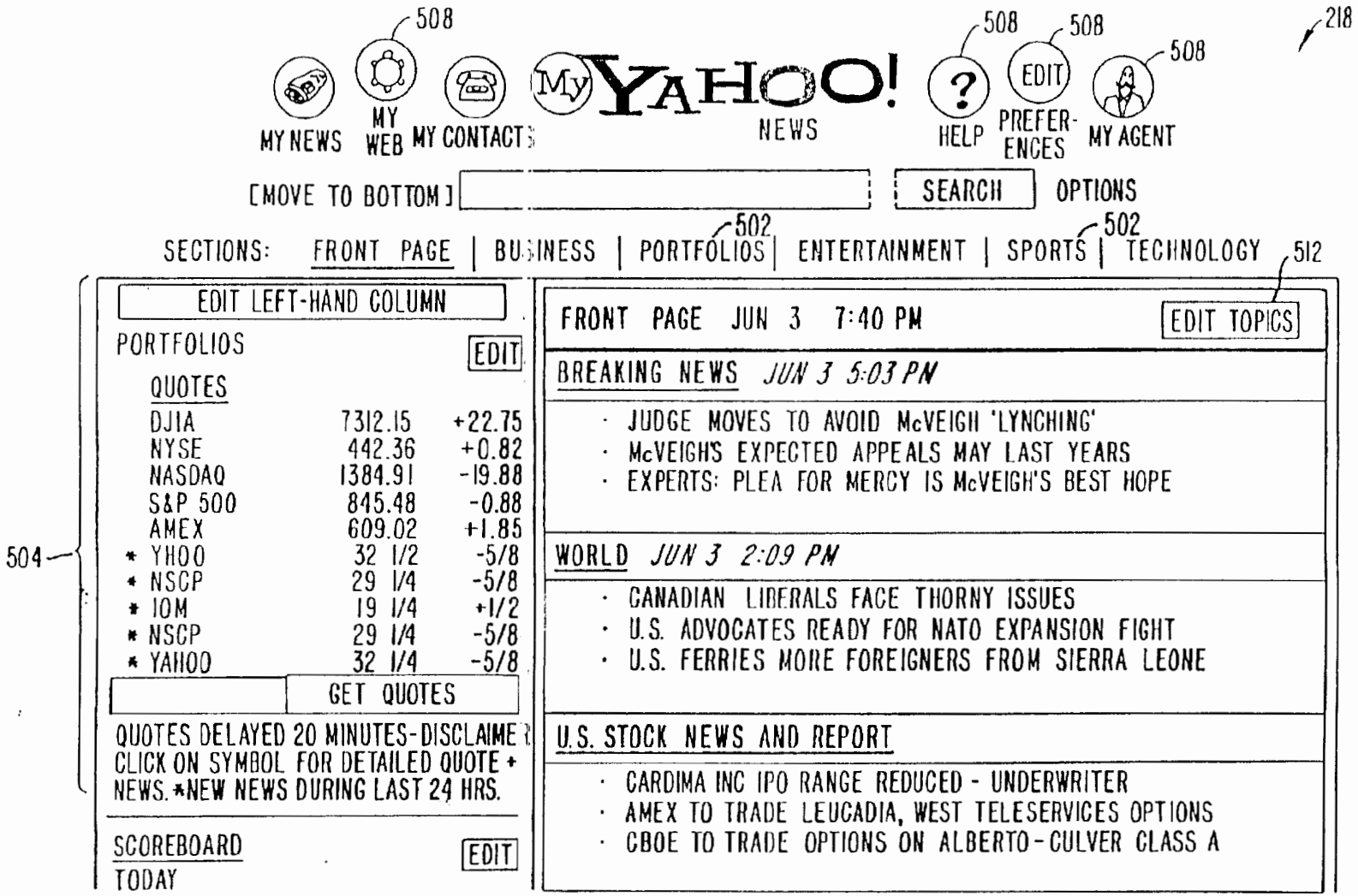


FIG. 5A.

EP 0 889 421 A1

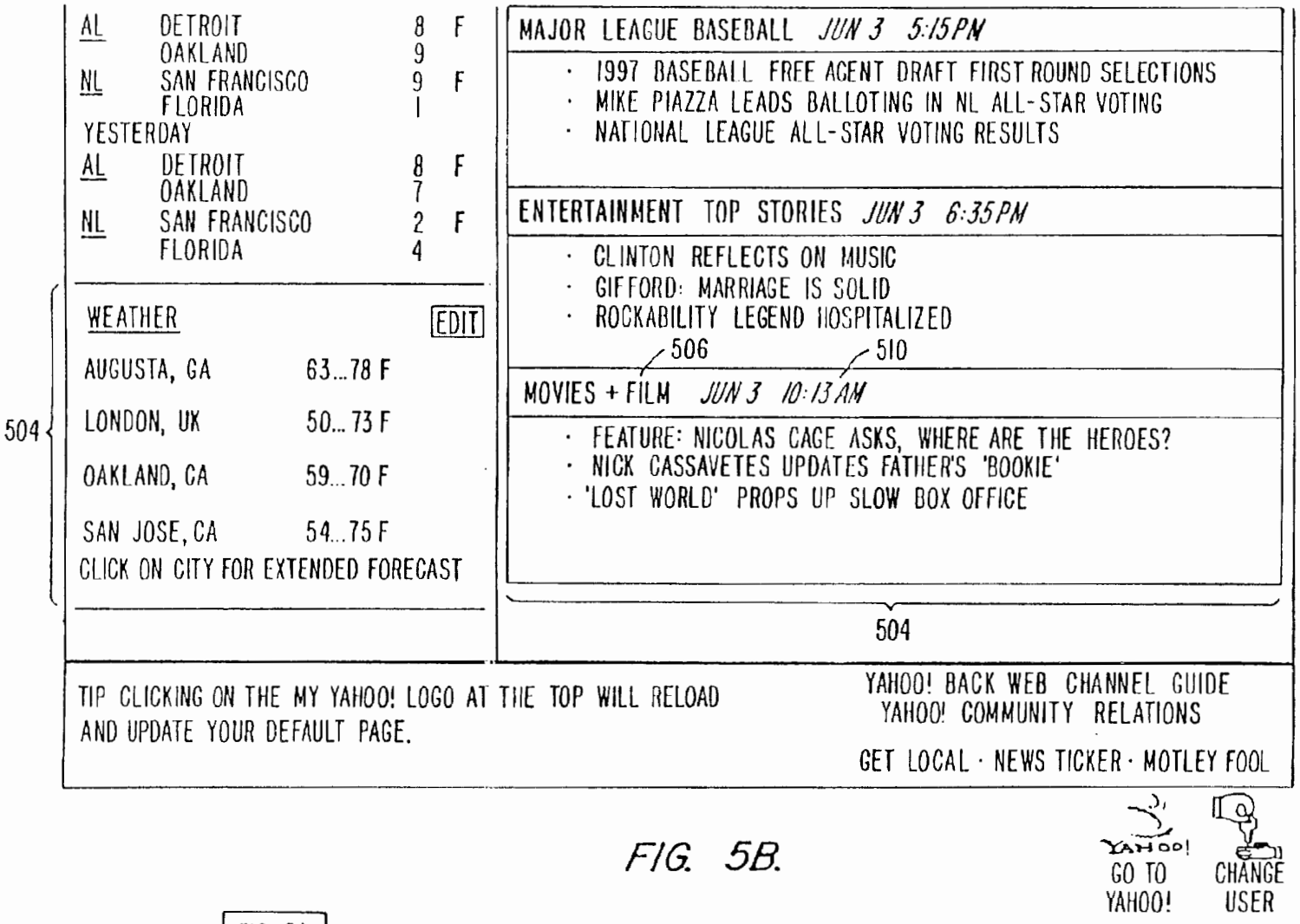


FIG. 5B.

FIG. 5. FIG. 5A. FIG. 5B.

EP 0 889 421 A1

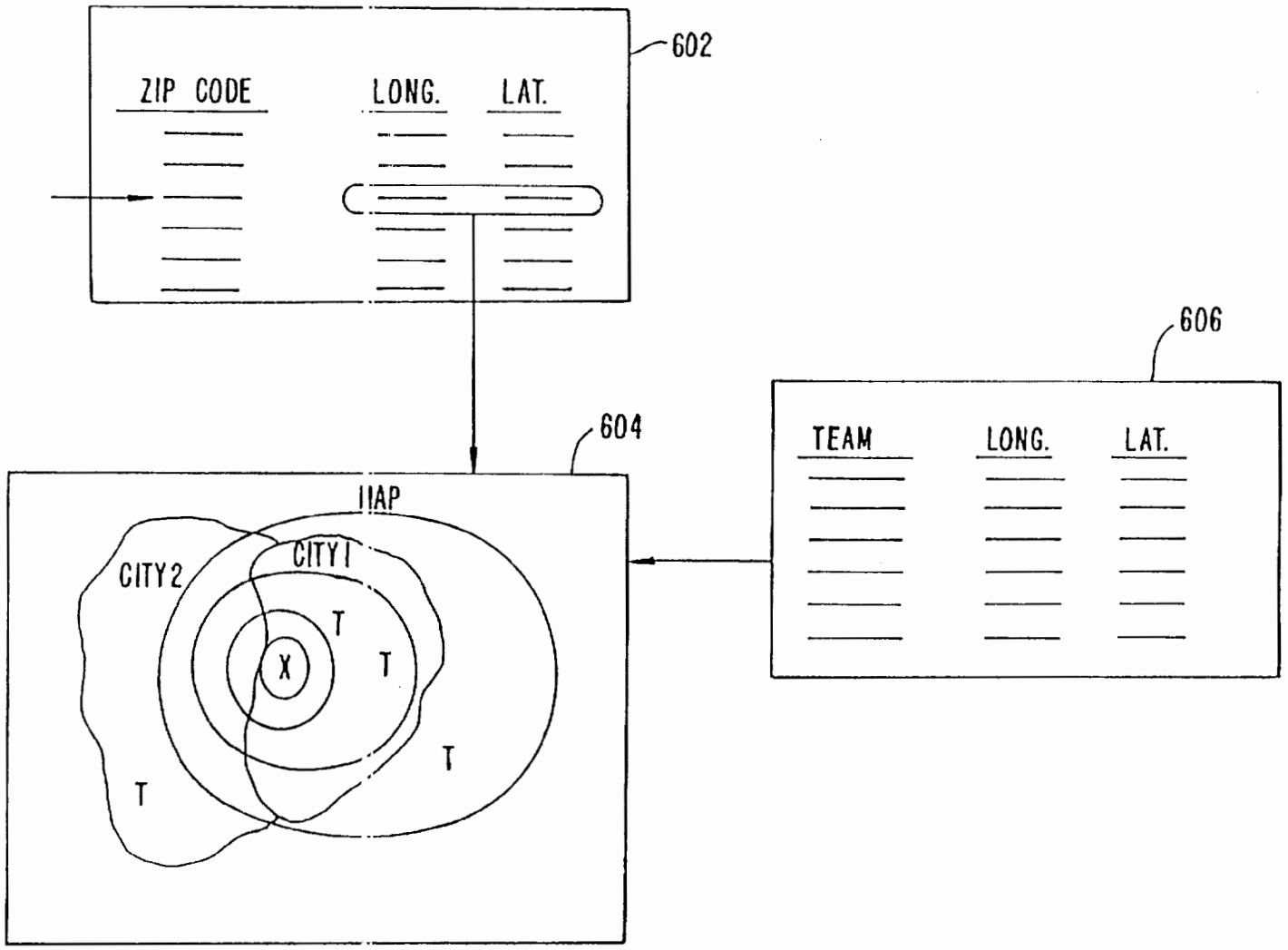

602
ZIP CODE
LONG.
LAT.
606
604
TEAM
LONG.
LAT.
IIAP
CITY2
CITY1
T
T
X
T
T
T

FIG. 6.

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 98 30 4651

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.6)
X	QUINTANA Y: "KNOWLEDGE-BASED INFORMATION FILTERING OF FINANCIAL INFORMATION" PROCEEDINGS OF THE NATIONAL ONLINE MEETING, 13 May 1997, pages 279-285, XP002057953 * the whole document *	1,6,7,9	G06F17/30
A	---	2-5,8	
X	EP 0 749 081 A (POINTCAST INC) 18 December 1996 * abstract * * page 3, line 15 - page 3, line 29 * * page 6, line 7 - page 6, line 19 * * page 11, line 3 - page 11, line 47 * * claims *	1-3,9	
Y	---	4,5	
A	WO 97 17662 A (C NET INC) 15 May 1997 * abstract * * page 2, line 21 - page 3, line 12 *	1-3	
Y	---	4,5	TECHNICAL FIELDS SEARCHED (Int.Cl.6)
X	CHESNAIS, P.R., MUCKLO, M.J., SHEENA, J.A.: "The Fishwrap personalized news system" PROCEEDINGS OF THE 2ND INTERNATIONAL WORKSHOP ON COMMUNITY NETWORKING INTEGRATED MULTIMEDIA SERVICES IN THE HOME, 20 - 22 June 1995, pages 275-282, XP002074902 Princeton, NJ, USA * page 275, column 1, line 42 - page 276, column 1, line 11 * * page 277, column 2, line 29 - page 279, column 1, line 6 *	1,6-9	G06F

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	19 August 1998	Abbing, R

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : **G06F 17/30**	**A1**	**(11) International Publication Number:** **WO 97/16796** **(43) International Publication Date:** 9 May 1997 (09.05.97)

(21) International Application Number: PCT/US96/17981

(22) International Filing Date: 29 October 1996 (29.10.96)

(30) Priority Data:
08/551,198 31 October 1995 (31.10.95) US

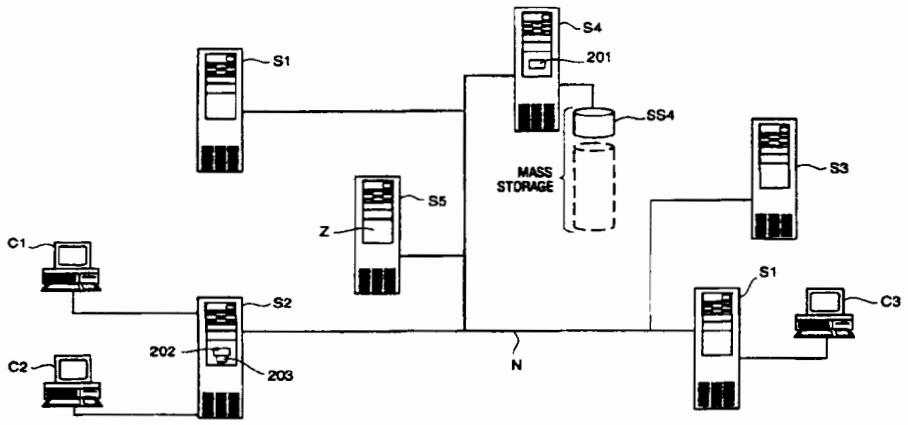
(71)(72) Applicants and Inventors: HERZ, Frederick, S., M. [US/US]; Box 625 Canaan Valley, Davis, WV 26260 (US). EISNER, Jason, M. [US/US]; 1015 Spruce Street, Philadelphia, PA 19107 (US). SMITH, Jonathan, M. [US/US]; 771 Princeton-Kingston Road, Princeton, NJ 08540-4165 (US). SALZBERG, Steven, L. [US/US]; 4409 Wickford Road, Baltimore, MD 21210 (US).

(74) Agents: GRAZIANO, James, M. et al.; Duft, Graziano & Forest, P.C., Suite 140, 1790-30th Street, Boulder, CO 80301-1018 (US).

(81) Designated States: AM, AU, BR, BY, CA, CN, EE, IL, IS, JP, KP, KR, KZ, LV, MN, MX, NZ, RU, SG, TM, TR, UA, UZ, VN, European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.

(54) Title: SYSTEM FOR CUSTOMIZED ELECTRONIC IDENTIFICATION OF DESIRABLE OBJECTS



(57) Abstract

This invention relates to customized electronic identification of desirable objects, such as news articles, in an electronic media environment, and in particular to a system that automatically constructs both a "target profile" for each target object in the electronic media based, for example, on the frequency with which each word appears in an article relative to its overall frequency of use in all articles, as well as a "target profile interest summary" for each user, which target profile interest summary describes the user's interest level in various types of target objects. The system then evaluates the target profiles against the users' target profile interest summaries to generate a user-customized rank ordered listing of target objects most likely to be of interest to each user so that the user can select from among these potentially relevant target objects, which were automatically selected by this system from the plethora of target objects that are profiled on the electronic media. Users' target profile interest summaries can be used to efficiently organize the distribution of information in a large scale system consisting of many users interconnected by means of a communication network. Additionally, a cryptographically-based pseudonym proxy server is provided to ensure the privacy of a user's target profile interest summary, by giving the user control over the ability of third parties to access this summary and to identify or contact the user.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AM	Armenia	**GB**	United Kingdom	**MW**	Malawi
AT	Austria	**GE**	Georgia	**MX**	Mexico
AU	Australia	**GN**	Guinea	**NE**	Niger
BB	Barbados	**GR**	Greece	**NL**	Netherlands
BE	Belgium	**HU**	Hungary	**NO**	Norway
BF	Burkina Faso	**IE**	Ireland	**NZ**	New Zealand
BG	Bulgaria	**IT**	Italy	**PL**	Poland
BJ	Benin	**JP**	Japan	**PT**	Portugal
BR	Brazil	**KE**	Kenya	**RO**	Romania
BY	Belarus	**KG**	Kyrgystan	**RU**	Russian Federation
CA	Canada	**KP**	Democratic People's Republic of Korea	**SD**	Sudan
CF	Central African Republic			**SE**	Sweden
CG	Congo	**KR**	Republic of Korea	**SG**	Singapore
CH	Switzerland	**KZ**	Kazakhstan	**SI**	Slovenia
CI	Côte d'Ivoire	**LI**	Liechtenstein	**SK**	Slovakia
CM	Cameroon	**LK**	Sri Lanka	**SN**	Senegal
CN	China	**LR**	Liberia	**SZ**	Swaziland
CS	Czechoslovakia	**LT**	Lithuania	**TD**	Chad
CZ	Czech Republic	**LU**	Luxembourg	**TG**	Togo
DE	Germany	**LV**	Latvia	**TJ**	Tajikistan
DK	Denmark	**MC**	Monaco	**TT**	Trinidad and Tobago
EE	Estonia	**MD**	Republic of Moldova	**UA**	Ukraine
ES	Spain	**MG**	Madagascar	**UG**	Uganda
FI	Finland	**ML**	Mali	**US**	United States of America
FR	France	**MN**	Mongolia	**UZ**	Uzbekistan
GA	Gabon	**MR**	Mauritania	**VN**	Viet Nam

SYSTEM FOR CUSTOMIZED ELECTRONIC IDENTIFICATION OF DESIRABLE OBJECTS

CROSS-REFERENCE TO RELATED APPLICATIONS

This patent application is a continuation-in-part of U.S. Patent Application Serial No. 08/346,425, filed November 28, 1994 and titled "SYSTEM AND METHOD FOR SCHEDULING BROADCAST OF AND ACCESS TO VIDEO PROGRAMS AND OTHER DATA USING CUSTOMER PROFILES", which application is assigned to the same assignee as the present application.

FIELD OF INVENTION

This invention relates to customized electronic identification of desirable objects, such as news articles, in an electronic media environment, and in particular to a system that automatically constructs both a "target profile" for each target object in the electronic media based, for example, on the frequency with which each word appears in an article relative to its overall frequency of use in all articles, as well as a "target profile interest summary" for each user, which target profile interest summary describes the user's interest level in various types of tar get objects. The system then evaluates the target profiles against the users' target profile interest summaries to generate a user-customized rank ordered listing of target objects most likely to be of interest to each user so that the user can select from among these potentially relevant target objects, which were automatically selected by this system from the plethora of target objects that are profiled on the electronic media. Users' target profile interest summaries can be used to efficiently organize the distribution of information in a large scale system consisting of many users interconnected by means of a communication network. Additionally, a cryptographically based proxy server is provided to ensure the privacy of a user's target profile interest summary, by giving the user control over the ability of third parties to access this summary and to identify or contact the user.

PROBLEM

It is a problem in the field of electronic media to enable a user to access information of relevance and interest to the user without requiring the user to expend an excessive amount of time and energy searching for the information. Electronic media, such as on-line information sources, provide a vast amount of information to users, typically in the form of "articles," each of which comprises a publication item or document that relates to a specific topic. The difficulty with electronic media is that the amount of information available to the user is overwhelming and the article repository systems that are connected on-line are not

organized in a manner that sufficiently simplifies access to only the articles of interest to the user. Presently, a user either fails to access relevant articles because they are not easily identified or expends a significant amount of time and energy to conduct an exhaustive search of all articles to identify those most likely to be of interest to the user. Furthermore, even if the user conducts an exhaustive search, present information searching techniques do not necessarily accurately extract only the most relevant articles, but also present articles of marginal relevance due to the functional limitations of the information searching techniques. There is also no existing system which automatically estimates the inherent quality of a n article or other target object to distinguish among a number of articles or target objects identified as of possible interest to a user.

Therefore, in the field of information retrieval, there is a long-standing need for a system which enables users to navigate through the plethora of information. With commercialization of communication networks, such as the Internet, the growth of available information has increased. Customization of the information delivery process to the user's unique tastes and interests is the ultimate solution to this problem. However, the techniques which have been proposed to date either only address the user's interests on a superficial level or provide greater depth and intelligence at the cost of unwanted demands on the user's time and energy. While many researchers have agreed that traditional methods have been lacking in this regard, no one to date has successfully addressed these problems in a holistic manner and provided a system that can fully learn and reflect the user's tastes and interests. This is particularly true in a practical commercial context, such as on-line services available on the Internet. There is a need for an information retrieval system that is largely or entirely passive, unobtrusive, undemanding o f the user, and yet both precise and comprehensive in its ability to learn and truly represent the user's tastes and interests. Present information retrieval systems require the user to specify the desired information retrieval behavior through cumbersome interfaces.

Users may receive information on a computer network either by actively retrieving the information or by passively receiving information that is sent to them. Just as users of information retrieval systems face the problem of too much information, so do users who are targeted with electronic junk mail by individuals and organizations. An ideal system would protect the user from unsolicited advertising, both by automatically extracting only the most relevant messages received by electronic ma il, and by preserving the

IPR2025-00200 / 722 Patent Page 195 of 1171 EX1002

confidentiality of the user's preferences, which should not be freely available to others on the network.

Researchers in the field of published article information retrieval have devoted considerable effort to finding efficient and accurate methods of allowing users to select articles of interest from a large set of articles. The most widely used methods of information retrieval are based on keyword matching: the user specifies a set of keywords which the user thinks are exclusively found in the desired articles and the information retrieval computer retrieves all articles which contain those keywords. Such methods are fast, but are notoriously unreliable, as users may not think of the right keywords, or the keywords may be used in unwanted articles in an irrelevant or unexpected context. As a result, the information retrieval computers retrieve many articles which are unwanted by the user. The logical combination of keywords and the use of wild-card search parameters help improve the accuracy of keyword searching but do not completely solve the problem of inaccurate search results.

Starting in the 1960's, an alternate approach to information retrieval was developed: users were presented with an article and asked if it contained the information they wanted, or to quantify how close the information contained in the article was to what they wanted. Each article was described by a profile which comprised either a list of the words in the article or, in more advanced systems, a table of word frequencies in the article. Since a measure of similarity between articles is the distance between their profiles, the measured similarity of article profiles can be used in article retrieval. For example, a user searching for information on a subject can write a short description of the desired information. The information retrieval computer generates an article profile for the request and then retrieves articles with profiles similar to the profile generated for the request. These requests can then be refined using "relevance feedback", where the user actively or passively rates the articles retrieved as to how close the information contained therein is to what is desired. The information retrieval computer then uses this relevance feedback information to refine the request profile and the process is repeated until the user either finds enough articles or tires of the search.

A number of researchers have looked at methods for selecting articles of most interest to users. An article titled "Social Information filtering: algorithms for automating 'word of mouth'" was published at the CHi-95 Proceedings by Patti Maes et al and describes the Ringo information retrieval system which recommends musical selections. The Ringo

IPR2025-00200 / 722 Patent Page 196 of 1171 EX1002

system requires active feedback from the users -- users must manually specify how much they like or dislike each musical selection. The Ringo system maintains a complete list of users ratings of music selections and makes recommendations by finding which selections were liked by multiple people. However, the Ringo system does not take advantage of any available descriptions of the music, such as structured descriptions in a data base, or free text, such as that contained in music reviews. An article titled "Evolving agents for personalized information filtering", published at the Proc. 9th IEEE Conf. on AI for Applications by Sheth and Maes, described the use of agents for information filtering which use genetic algorithms to learn to categorize Usenet news articles. In this system, users must define news categories and the users actively indicate their opinion of the selected articles. Their system uses a list of keywords to represent sets of articles and the records of users' interests are updated using genetic algorithms.

A number of other research groups have looked at the automatic generation and labeling of clusters of articles for the purpose of browsing through the articles. A group at Xerox Parc published a paper titled "Scatter/gather: a cluster-based approach to browsing large article collections" at the 15 Ann. Int'l SIGIR '92, ACM 318-329 (Cutting et al. 1992). This group developed a method they call "scatter/gather" for performing information retrieval searches. In this method, a collection of articles is "scattered" into a small number of clusters, the user then chooses one or more of these clusters based on short summaries of the cluster. The selected clusters are then "gathered" into a subcollection, and then the process is repeated. Each iteration of this process is expected to produce a small, more focused collection. The cluster "summaries" are generated by picking those words which appear most frequently in the cluster and the titles of those articles closest to the center of the cluster. However, no feedback from users is collected or stored, so no performance improvement occurs over time.

Apple's Advanced Technology Group has developed an interface based on the concept of a "pile of articles". This interface is described in an article titled "A 'pile' metaphor for supporting casual organization of information in Human factors in computer systems" published in CHI '92 Conf. Proc. 627-634 by Mander, R. G. Salomon and Y. Wong. 1992. Another article titled "Content awareness in a file system interface: implementing the 'pile' metaphor for organizing information" was published in 16 Ann. Int'l SIGIR '93, ACM 260-269 by Rose E. D. et al. The Apple interface uses word frequencies to automatically file articles by picking the pile most similar to the article being filed. This

IPR2025-00200 / 722 Patent Page 197 of 1171 EX1002

system functions to cluster articles into subpiles, determine key words for indexing by picking the words with the largest TF/IDF (where TF is term (word) frequency and IDF is the inverse document frequency) and label piles by using the determined key words.

Numerous patents address information retrieval methods, but none develop records of a user's interest based on passive monitoring of which articles the user accesses. None of the systems described in these patents pre sent computer architectures to allow fast retrieval of articles distributed across many computers. None of the systems described in these patents address issues of using such article retrieval and matching methods for purposes of commerce or of matching users with common interests or developing records of users' interests. U.S. Patent No. 5,321,833 issued to Chang et al. teaches a method in which users choose terms to use in an information retrieval query, and specify the relative weightings of the different terms. The Chang system then calculates multiple levels of weighting criteria. U.S. Patent No. 5,301,109 issued to Landauer et al. teaches a method for retrieving articles in a multiplicity of languages by constructing "latent vectors" (SVD or PCA vectors) which represent correlations between the different words. U.S. Patent No. 5,331,554 issued to Graham et al. discloses a method for retrieving segments of a manual by comparing a query with nodes in a decision tree. U.S. Patent No. 5,331,556 addresses techniques for deriving morphological part-of-speech information and thus to make use of the similarities of different forms of the same word (e.g. "article" and "articles").

Therefore, there presently is no information retrieval and delivery system operable in an electronic media environment that enables a user to access information of relevance and interest to the user without requiring the user to expend an excessive amount of time and energy.

SOLUTION

The above-described problems are solved and a technical advance achieved in the field by the system for customized electronic identification of desirable objects in an electronic media environment, which system enables a user to access target objects of relevance and interest to the user without requiring the user to expend an excessive amount of time and energy. Profiles of the target objects are stored on electronic media and are accessible via a data communication network. In many applications, the target objects are informational I n nature, and so may themselves be stored on electronic media and be accessible via a data communication network.

IPR2025-00200 / 722 Patent Page 198 of 1171 EX1002

Relevant definitions of terms for the purpose of this description include: (a.) an object available for access by the user, which may be either physical or electronic in nature, is termed a "target object", (b.) a digitally represented profile indicating t hat target object's attributes is termed a "target profile", (c.) the user looking for the target object is termed a "user", (d.) a profile holding that user's attributes, including age/zip code/etc. is termed a "user profile", (e.) a summary of digital profiles of target objects that a user likes and/or dislikes, is termed the "target profile interest summary" of that user, (f.) a profile consisting of a collection of attributes, such that a user likes target objects whose profiles are similar to this collection of attributes, is termed a "search profile" or in some contexts a "query" or "query profile," (g.) a specific embodiment of the target profile interest summary which comprises a set of search profiles is termed the "search profile set" of a user, (h.) a collection of target objects with similar profiles, is termed a "cluster," (i.) an aggregate profile formed by averaging the attributes of all tar get objects in a cluster, termed a "cluster profile," (j.) a real number determined by calculating the statistical variance of the profiles of all target objects in a cluster, is termed a "cluster variance," (k.) a real number determined by calculating the maximum distance between the profiles of any two target objects in a cluster, is termed a "cluster diameter."

The system for electronic identification of desirable objects of the present invention automatically constructs both a target profile for each target object in the electronic media based, for example, on the frequency with which each word appears in an article relative to its overall frequency of use in all articles, as well as a "target profile interest summary" for each user, which target profile interest summary describes the user's interest level in various types of target objects. The system then evaluates the target profiles against the users' target profile interest summaries to generate a user-customized rank ordered listing of tar get objects most likely to be of interest to each user so that the user can select from among these potentially relevant target objects, which were automatically selected by this system from the plethora of target objects available on the electronic media.

Because people have multiple interests, a target profile interest summary for a single user must represent multiple areas of interest, for example, by consisting of a set of individual search profiles, each of which identifies one of the user's areas of interest. Each user is presented with those target objects whose profiles most closely match the user's interests as described by the user's target profile interest summary. Users' target profile interest summaries are automatically updated on a continuing basis to reflect each user's

IPR2025-00200 / 722 Patent Page 199 of 1171 EX1002

changing interests. In addition, target objects can be grouped into clusters based on their similarity to each other, for example, based on similarity of their topics in the case where the target objects are published articles, and menus automatically generated for each cluster of target objects to allow users to navigate throughout the clusters and manually locate target objects of interest. For reasons of confidentiality and privacy, a particular user may not wish to make public all of the interests recorded in the user's target profile interest summary, particularly when these interests are determined by the user's purchasing patterns. The user may desire that all or part of the target profile interest summary be kept confidential, such as information relating to the user's political, religious, financial or purchasing behavior; indeed, confidentiality with respect to purchasing behavior is the user's legal right in many states. It is therefore necessary that data in a user's target profile interest summary be protected from unwanted disclosure except with the user's agreement. At the same time, the user's target profile interest summaries must be accessible to the relevant servers that perform the matching of target objects to the users, if the benefit of this matching is desired by both providers and consumers of the target objects. The disclosed system provides a solution to the privacy problem by using a proxy server which acts as an intermediary between the information provider and the user. The proxy server dissociates the user's true identity from the pseudonym by the use of cryptographic techniques. The proxy server also permits users to control access to their target profile interest summaries and/or user profiles, including provision of this information to marketers and advertisers if they so desire, possibly in exchange for cash or other considerations. Marketers may purchase these profiles in order to target advertisements to particular users, or they may purchase partial user profiles, which do not include enough information to identify the individual users in question, in order to carry out standard kinds of demographic analysis and market research on the resulting database of partial user profiles.

In the preferred embodiment of the invention, the system for customized electronic identification of desirable objects uses a fundamental methodology for accurately and efficiently matching users and target objects by automatically calculating, using and updating profile information that describes both the users' interests and the target objects' characteristics. The target objects may be published articles, purchasable items, or even other people, and their properties are stored, and/or represented and /or denoted on the electronic media as (digital) data. Examples of target objects can include, but are not limited to: a newspaper story of potential interest, a movie to watch, an item to buy, e-mail

to receive, or another person to correspond with. In all these cases, the information delivery process in the preferred embodiment is based on determining the similarity between a profile for the target object and the profiles of target objects for which the user (or a similar user) has provided positive feedback in the past. The individual data that describe a target object and constitute the target object's profile are herein termed "attributes" of the target object. Attributes may include, but are not limited to, the following: (1) long pieces of text (a newspaper story, a movie review, a product description or an advertisement), (2) short pieces of text (name of a movie's director, name of town from which an advertisement was placed, name of the language in which an article was written), (3) numeric measurements (price of a product, rating given to a movie, reading level of a book), (4) associations with other types of objects (list of actors in a movie, list of persons who have read a document). Any of these attributes, but especially the numeric ones, may correlate with the quality of the target object, such as measures of its popularity (how often it is accessed) or of user satisfaction (number of complaints received).

The preferred embodiment of the system for customized electronic identification of desirable objects operates in an electronic media environment for accessing these target objects, which may be news, electronic mail, other published documents, or product descriptions. The system in its broadest construction comprises three conceptual modules, which may be separate entities distributed across many implementing systems, or combined into a lesser subset of physical entities. The specific embodiment of this system disclosed herein illustrates the use of a first module which automatically constructs a "target profile" for each target object in the electronic media based on various descriptive attributes of the target object. A second module uses interest feedback from users to construct a "target profile interest summary" for each user, for example in the form of a "search profile set" consisting of a plurality of search profiles, each of which corresponds to a single topic of high interest for the user. The system further includes a profile processing module which estimates each us er's interest in various target objects by reference to the users' target profile interest summaries, for example by comparing the target profiles of these target objects against the search profiles in users' search profile sets, and generates for each us er a customized rank-ordered listing of target objects most likely to be of interest to that user. Each user's target profile interest summary is automatically updated on a continuing basis to reflect the user's changing interests.

IPR2025-00200 / 722 Patent Page 201 of 1171 EX1002

Target objects may be of various sorts, and it is sometimes advantageous to use a single system that delivers and/or clusters target objects of several distinct sorts at once, in a unified framework. For example, users who exhibit a strong interest in certain novels may also show an interest in certain movies, presumably of a similar nature. A system in which some target objects are novels and other target objects are movies can discover such a correlation and exploit it in order to group particular novels with particular movies, e.g., for clustering purposes, or to recommend the movies to a user who has demonstrated interest in the novels. Similarly, if users who exhibit an interest in certain World Wide Web sites also exhibit an interest in certain products, the system can match the products with the sites and thereby recommend to the marketers of those products that they place advertisements at those sites, e.g., in the form of hypertext links to their own sites.

The ability to measure the similarity of profiles describing target objects and a user's interests can be applied in two basic ways: filtering and browsing. Filtering is useful when large numbers of target objects are described in the electronic media s pace. These target objects can for example be articles that are received or potentially received by a user, who only has time to read a small fraction of them. For example, one might potentially receive all items on the AP news wire service, all items posted to a number of news groups, all advertisements in a set of newspapers, or all unsolicited electronic mail, but few people have the time or inclination to read so many articles. A filtering system in the system for customized electronic identification of desirable objects automatically selects a set of articles that the user is likely to wish to read. The accuracy of this filtering system improves over time by noting which articles the user reads and by generating a measurement of the depth to which the user reads each article. This information is then us ed to update the user's target profile interest summary. Browsing provides an alternate method of selecting a small subset of a large number of target objects, such as articles. Articles are organized so that users can actively navigate among groups of articles by moving from one group to a larger, more general group, to a smaller, more specific group, or to a closely related group. Each individual article forms a one-member group of its own, so that the user can navigate to and from individual article s as well as larger groups. The methods used by the system for customized electronic identification of desirable objects allow articles to be grouped into clusters and the clusters to be grouped and merged into larger and larger clusters. These hierarchies of clusters then form the basis for menuing and navigational systems to allow

IPR2025-00200 / 722 Patent Page 202 of 1171 EX1002

the rapid searching of large numbers of articles. This same clustering technique is applicable to any type of target objects that can be profiled on the electronic media.

There are a number of variations on the theme of developing and using profiles for article retrieval, with the basic implementation of an on-line news clipping service representing the preferred embodiment of the invention. Variations of this basic system are disclosed and comprise a system to filter electronic mail, an extension for retrieval of target objects such as purchasable items which may have more complex descriptions, a system to automatically build and alter menuing systems for browsing and searching through large numbers of target objects, and a system to construct virtual communities of people with common interests. These intelligent filters and browsers are necessary to provide a truly passive, intelligent system interface. A user interface that permits intuitive browsing and filtering represents for the first time an intelligent system for determining the affinities between users and target objects. The detailed, comprehensive target profiles and user-specific target profile interest summaries enable the system to provide responsive routing of specific queries for user information access. The information maps so produced and the application of users' target profile interest summaries to predict the information consumption patterns of a user allows for pre-caching of data at locations on the data communication network and at times that minimize the traffic flow in the communication network to thereby efficiently provide the desired information to the user and/or conserve valuable storage space by only storing those target objects (or segments thereof) which are relevant to the user's interests.

BRIEF DESCRIPTION OF THE DRAWING

Figure 1 illustrates in block diagram form a typical architecture of an electronic media system in which the system for customized electronic identification of desirable objects of the present invention can be implemented as part of a user server system;

Figure 2 illustrates in block diagram form one embodiment of the system for customized electronic identification of desirable objects;

Figures 3 and 4 illustrate typical network trees;

Figure 5 illustrates in flow diagram form a method for automatically generating article profiles and an associated hierarchical menu system;

Figures 6-9 illustrate examples of menu generating process;

Figure 10 illustrates in flow diagram form the operational steps taken by the system for customized electronic identification of desirable objects to screen articles for a user;

IPR2025-00200 / 722 Patent Page 203 of 1171 EX1002

Figure 11 illustrates a hierarchical cluster tree example;

Figure 12 illustrates in flow diagram form the process for determination of likelihood of interest by a specific user in a selected target object;

Figures 13A-B illustrate in flow diagram form the automatic clustering process;

Figure 14 illustrates in flow diagram form the use of the pseudonymous server;

Figure 15 illustrates in flow diagram form the use of the system for accessing information in response to a user query; and

Figure 16 illustrates in flow diagram form the use of the system for accessing information in response to a user query when the system is a distributed network implementation.

DETAILED DESCRIPTION

MEASURING SIMILARITY

This section describes a general procedure for automatically measuring the similarity between two target objects, or, more precisely, between target profiles that are automatically generated for each of the two target objects. This similarity determination process is applicable to target objects in a wide variety of contexts. Target objects being compared can be, as an example but not limited to: textual documents, human beings, movies, or mutual funds. It is assumed that the target profiles which describe the target objects are stored at one or more locations in a data communication network on data storage media associated with a computer system. The computed similarity measurements serve as input to additional processes, which function to enable human users to locate desired target objects using a large computer system. These additional processes estimate a human user's interest in various target objects, or else cluster a plurality of target objects in to logically coherent groups. The methods used by these additional processes might in principle be implemented on either a single computer or on a computer network. Jointly or separately, they form the underpinning for various sorts of database systems and information retrieval systems.

Target Objects and Attributes

In classical Information Retrieval (IR) technology, the user is a literate human and the target objects in question are textual documents stored on data storage devices interconnected to the user via a computer network. That is, the target objects consist entirely of text, and so are digitally stored on the data storage devices within the computer network. However, there are other target object domains that present related retrieval problems that are not capable of being solved by present information retrieval technology:

(a.) the user is a film buff and the target objects are movies available on videotape.

(b.) the user is a consumer and the target objects are used cars being sold.

(c.) the user is a consumer and the target objects are products being sold through promotional deals.

(d.) the user is an investor and the target objects are publicly traded stocks, mutual funds and/or real estate properties.

(e.) the user is a student and the target objects are classes being offered.

(f.) the user is an activist and the target objects are Congressional bills of potential concern. (g.) the user is a direct-mail marketer and the target objects are potential customers.

(h.) the user is a net-surfer and the target objects are pages, servers, or newsgroups available on the World Wide Web.

(i.) the user is a philanthropist and the target objects are charities.

(j.) the user is ill and the target objects are medical specialists.

(k.) the user is an employee and the target objects are potential employers.

(l.) the user is an employer and the target objects are potential employees.

(m.) the user is an beleaguered executive and the target objects are electronic mail messages addressed to the user.

(n.) the user is a lonely heart and the target objects are potential conversation partners. (o.) the user is in search of an expert and the target objects are users, with known retrieval habits, of an document retrieval system.

(p.) the user is a social worker and the target objects are families that may need extra visits.

(q.) the user is an oncologist and the target objects are women for whom mammograms may be advisable.

(r.) the user is an auto insurance company and the target objects are potential customers

In all these cases, the user wishes to locate some small subset of the target objects -- such as the target objects that the user most desires to rent, buy, investigate, meet, read, give mammograms to, insure, and so forth. The task is to help the user identify the most interesting target objects, where the user's interest in a target object is defined to be a numerical measurement of the user's relative desire to locate that object rather than others.

IPR2025-00200 / 722 Patent Page 205 of 1171 EX1002

The generality of this problem motivates a general approach to solving the information retrieval problems noted above. It is assumed that many target objects are known to the system for customized electronic identification of desirable objects, and that specifically, the system stores (or has the ability to reconstruct) several pieces of information about each target object. These pieces of information are termed "attributes": collectively, they are said to form a profile of the target object, or a "target profile." For example, where the system for customized electronic identification of desirable objects is activated to identify movies of interest, the system is likely be concerned with the values of attributes such as these:

(a.) title of movie,

(b.) name of director,

(c.) Motion Picture Association of America (MPAA) child-appropriateness rating (0=G, 1=PG, ...),

(d.) date of release,

(e.) number of stars granted by a particular critic,

(f.) number of stars granted by a second critic,

(g.) number of stars granted by a third critic,

(h.). full text of review by the third critic,

(i.). list of customers who have previously rented this movie,

(j.) list of actors.

Each movie has a different set of values for these attributes. This example conveniently illustrates three kinds of attributes. Attributes c-g are *numeric attributes*, of the sort that might be found in a database record. It is evident that they can be used to help the user identify target objects (movies) of interest. For example, the user might previously have rented many Parental Guidance (PG) films, and many films made in the 1970's. This generalization is useful: new films with values for one or both attributes that are numerically similar to these (such as MPAA rating of 1, release date of 1975) are judged similar to the films the user already likes, and therefore of probable interest. Attributes a-b and h are *textual attributes*. They too are important for helping the user locate desired films. For example, perhaps the user has shown a past interest in films whose review text (attribute h) contains words like "chase," "explosion," "explosions," "hero," "gripping," and "superb." This generalization is again useful in identifying new films of interest. Attribute i is an *associative attribute*. It records associations between the target objects in this domain,

IPR2025-00200 / 722 Patent Page 206 of 1171 EX1002

namely movies, and ancillary target objects of an entirely different sort, namely humans. A good indication that the user wants to rent a particular movie is that the user has previously rented other movies with similar attribute values, and this holds for attribute I just as it does for attributes a-h. For example, if the user has often liked movies that customer C_{17} and customer C_{190} have rented, then the user may like other such movies, which have similar values for attribute i. Attribute j is another example of an associative attribute, recording associations between target objects and actors. Notice that any of these attributes can be made subject to authentication when the profile is constructed, through the use of digital signatures; for example, the target object could be accompanied by a digitally signed note from the MPAA, which note names the target object and specifies its authentic value for attribute c .

These three kinds of attributes are common: numeric, textual, and associative. In the classical information retrieval problem, where the target objects are documents (or more generally, coherent document sections extracted by a text segmentation method), the system might only consider a single, textual attribute when measuring similarity: the full text of the target object. However, a more sophisticated system would consider a longer target profile, including numeric and associative attributes:

(a.) full text of document (textual),

(b.) title (textual),

(c.) author (textual),

(d.) language in which document is written (textual),

(e.) date of creation (numeric),

(f.) date of last update (numeric),

(g.) length in words (numeric),

(h.) reading level (numeric),

(i.) quality of document as rated by a third_party editorial agency (numeric),

(j.) list of other readers who have retrieved this document (associative).

As another domain example, consider a domain where the user is an advertiser and the target objects are potential customers. The system might store the following attributes for each target object (potential customer):

(a.) first two digits of zip code (textual),

(b.) first three digits of zip code (textual),

(c.) entire five-digit zip code (textual),

IPR2025-00200 / 722 Patent Page 207 of 1171 EX1002

(d.) distance of residence from advertiser's nearest physical storefront (numeric),

(e.) annual family income (numeric),

(f.) number of children (numeric),

(g.) list of previous items purchased by this potential customer (associative),

list of filenames stored on this potential customer's client computer (associative),

list of movies rented by this potential customer (associative),

list of investments in this potential customer's investment portfolio (associative),

list of documents retrieved by this potential customer (associative),

written response to Rorschach inkblot test (textual),

multiple-choice responses by this customer to 20 self-image questions (20 textual attributes).

As always, the notion is that similar consumers buy similar products. It should be noted that diverse sorts of information are being used here to characterize consumers, from their consumption patterns to their literary taste s and psychological peculiarities, and that this fact illustrates both the flexibility and power of the system for customized electronic identification of desirable objects of the present invention. Diverse sorts of information can be used as attributes in other domains as well (as when physical, economic, psychological and interest-related questions are used to profile the applicants to a dating service, which is indeed a possible domain for the present system), and the advertiser domain is simply an example.

As a final domain example, consider a domain where the user is an stock market investor and the target objects are publicly traded corporations. A great many attributes might be used to characterize each corporation, including but not limited to the following:

(a.) type of business (textual),

(b.) corporate mission statement (textual),

(c.) number of employees during each of the last 10 years (ten separate numeric attributes),

(d.) percentage growth in number of employees during each of the last 10 years,

(e.) dividend payment issued in each of the last 40 quarters, as a percentage of current share price,

(f.) percentage appreciation of stock value during each of the last 40 quarters, list of shareholders (associative),

(g.) composite text of recent articles about the corporation in the financial press (textual). It is worth noting some additional attributes that are of interest in some

IPR2025-00200 / 722 Patent Page 208 of 1171 EX1002

domains. In the case of documents and certain other domains, it is useful to know the source of each target object (for example, refereed journal article vs. UPI newswire article vs. Usenet newsgroup posting vs. question-answer pair from a question-and-answer list vs. tabloid newspaper article vs. ...); the source may be represented as a single-term textual attribute. Important associative attributes for a hypertext document are the list of documents that it links to, and the list of documents that link to it. Documents with similar citations are similar with respect to the former attribute, and documents that are cited in the same places are similar with respect to the latter. A convention may optionally be adopted that any document also links to itself. Especially in systems where users can choose whether or not to retrieve a target object, a target object's popularity (or circulation) can be usefully measured as a numeric attribute specifying the number of users who have retrieved that object. Related measurable numeric attributes that also indicate a kind of popularity include the number of replies to a target object, in the domain where target objects are messages posted to an electronic community such as an computer bulletin board or newsgroup, and the number of links leading to a target object, in the domain where target objects are interlinked hypertext documents on the World Wide Web or a similar system. A target object may also receive explicit numeric evaluations (another kind of numeric attribute) from various groups, such as the Motion Picture Association of America (MPAA), as above, which rates movies' appropriateness for children, or the American Medical Association, which might rate the accuracy and novelty of medical research papers, or a random survey sample of users (chosen from all users or a selected set of experts), who could be asked to rate nearly anything. Certain other types of evaluation, which also yield numeric attributes, may be carried out mechanically. For example, the difficulty of reading a text can be assessed by standard procedures that count word and sentence lengths, while the vulgarity of a text could be defined as (say) the number of vulgar words it contains, and the expertise of a text could be crudely assessed by counting the number of similar texts its author had previously retrieved and read using the invention, perhaps confining this count to texts that have high approval ratings from critics. Finally, it is possible to synthesize certain textual attributes mechanically, for example to reconstruct the script of a movie by applying speech recognition techniques to its soundtrack or by applying optical character recognition techniques to its closed-caption subtitles.

IPR2025-00200 / 722 Patent Page 209 of 1171 EX1002

Decomposing Complex Attributes

Although textual and associative attributes are large and complex pieces of data, for information retrieval purposes they can be decomposed into smaller, simpler numeric attributes. This means that any set of attributes can be replaced by a (usually larger) set of numeric attributes, and hence that any profile can be represented as a vector of numbers denoting the values of these numeric attributes. In particular, a textual attribute, such as the full text of a movie review, can be replaced by a collection of numeric attributes that represent scores to denote the presence and significance of the words "aardvark," "aback," "abacus," and so on through "zymurgy" in that text. The score of a word in a text may be defined in numerous ways. The simplest definition is that the score is the rate of the word in the text, which is computed by computing the number of times the word occurs in the text, an d dividing this number by the total number of words in the text. This sort of score is often called the "term frequency" (TF) of the word. The definition of term frequency may optionally be modified to weight different portions of the text unequally: for example, any occurrence of a word in the text's title might be counted as a 3-fold or more generally k-fold occurrence (as if the title had been repeated k times within the text), in order to reflect a heuristic assumption that the words in the title are particularly important indicators of the text's content or topic.

However, for lengthy textual attributes, such as the text of an entire document, the score of a word is typically defined to be not merely its term frequency, but its term frequency multiplied by the negated logarithm of the word's "global frequency," as measured with respect to the textual attribute in question. The global frequency of a word, which effectively measures the word's uninformativeness, is a fraction between 0 and 1, defined to be the fraction of all target objects for which the textual attribute in question contains this word. This adjusted score is often known in the art as TF/IDF ("term frequency times inverse document frequency"). When global frequency of a word is taken into account in this way, the common, uninformative words have scores comparatively close to zero, no matter how often or rarely they appear in the text. Thus, their rate has little influence on the object's target profile. Alternative methods of calculating word scores include latent semantic indexing or probabilistic models.

Instead of breaking the text into its component words, one could alternatively break the text into overlapping word bigrams (sequences of 2 adjacent words), or more generally, word n-grams. These word n-grams may be scored in the same way as individual words.

IPR2025-00200 / 722 Patent Page 210 of 1171 EX1002

Another possibility is to use character n-grams. For example, this sentence contains a sequence of overlapping character 5-grams which starts "for e", "or ex", "r exa", " exam", "examp", etc. The sentence may be characterized, imprecisely but usefully, by the score of each possible character 5-gram ("aaaaa", "aaaab", ... "zzzzz") in the sentence. Conceptually speaking, in the character 5-gram case, the textual attribute would be decomposed into at least $26^5 = 11,881,376$ numeric attributes. Of course, for a given target object, most of these numeric attributes have values of 0, since most 5-grams do not appear in the target object attributes. These zero values need not be stored anywhere. For purposes of digital storage, the value of a textual attribute could be characterized by storing the set of character 5-grams that actually do appear in the text, together with the nonzero score of each one. Any 5-gram that is no t included in the set can be assumed to have a score of zero. The decomposition of textual attributes is not limited to attributes whose values are expected to be long texts. A simple, one-term textual attribute can be replaced by a collection of numeric attributes in exactly the same way. Consider again the case where the target objects are movies. The "name of director" attribute, which is textual, can be replaced by numeric attributes giving the scores for "Federico-Fellini," "Woody-Allen," "Terence- Davies," and so forth, in that attribute. For these one-term textual attributes, the score of a word is usually defined to be its rate in the text, without any consideration of global frequency. Note that under these conditions, one of the scores is 1, while the other scores are 0 and need not be stored. For example, if Davies did direct the film, then it is "Terence-Davies" whose score is 1, since "Terence-Davies" constitutes 100% of the words in the textual value of the "name of director" attribute. It might seem that nothing has been gained over simply regarding the textual attribute as having the string value "Terence-Davies." However, the trick of decomposing every non-numeric attribute into a collection of numeric attributes proves useful for the clustering and decision tree methods described later, which require the attribute values of different objects to be averaged and/or ordinally ranked. Only numeric attributes can be averaged or ranked in this way. Just as a textual attribute may be decomposed into a number of component terms (letter or word n-grams), an associative attribute may be decomposed into a number of component associations. For instance, in a domain where the target objects are movies, a typical associative attribute used in profiling a movie would be a list of customers who have rented that movie. This list can be replaced by a collection of numeric attributes, which give the "association scores" between the movie and each of the customers known to the system. For example, the 165th such numeric

IPR2025-00200 / 722 Patent Page 211 of 1171 EX1002

attribute would be the association score between the movie and customer #165, where the association score is defined to be 1 if customer #165 has previously rented the movie, and 0 otherwise. In a subtler refinement, this association s core could be defined to be the degree of interest, possibly zero, that customer #165 exhibited in the movie, as determined by relevance feedback (as described below). As another example, in a domain where target objects are companies, an associative attribute indicating the major shareholders of the company would be decomposed into a collection of association scores, each of which would indicate the percentage of the company (possibly zero) owned by some particular individual or corporate body. Just as with the term scores used in decomposing lengthy textual attributes, each association score may optionally be adjusted by a multiplicative factor: for example, the association score between a movie and customer #165 might be multiplied by the negated logarithm of the "global frequency" of customer #165, i.e., the fraction of all movies that have been rented by customer #165. Just as with the term scores used in decomposing textual attributes, most association scores found when decomposing a particular value of an associative attribute are zero, and a similar economy of storage may be gained in exactly the same manner by storing a list of only those ancillary objects with which the target object has a nonzero association score, together with their respective association scores.

Similarity Measures

What does it mean for two target objects to be similar? More precisely, how should one measure the degree of similarity? Many approaches are possible and any reasonable metric that can be computed over the set of target object profiles can be used, where target objects are considered to be similar if the distance between their profiles is small according to this metric. Thus, the following preferred embodiment of a target object similarity measurement system has m any variations.

First, define the distance between two values of a given attribute according to whether the attribute is a numeric, associative, or textual attribute. If the attribute is numeric, then the distance between two values of the attribute is the absolute value of the difference between the two values. (Other definitions are also possible: for example, the distance between prices p1 and p2 might be defined by $|(p1-p2)|/(\max(p1,p2)+1)$, to recognize that when it comes to customer interest, $50 00 and $5020 are very similar, whereas $3 and $23 are not.) If the attribute is associative, then its value V may be decomposed as described above into a collection of real numbers, representing the

association scores between the target object in question and various ancillary objects. V may therefore be regarded as a vector with components V_1, V_2, V_3, etc., representing the association scores between the object and ancillary objects 1, 2, 3, etc., respectively. The distance between two vector values V and U of an associative attribute is then computed using the angle distance measure, arccos $(VU^t / \text{sqrt}((Vv^t)(UU^t))$. (Note that the three inner products in this expression have the form $XY^t = X_1 Y_1 + X_2 Y_2 + X_3 Y_3 + \ldots$, and that for efficient computation, terms of the form $X_i Y_i$ may be omitted from this sum if either of the scores X_i and Y_i is zero.) Finally, if the attribute is textual, then its value V may be decomposed as described above into a collection of real numbers, representing the scores of various word n-grams or character n-grams in the text. Then the value V may again be regarded as a vector, and the distance between two values is again defined via the angle distance measure. Other similarity metrics between two vectors, such as the dice measure, may be used instead. It happens that the obvious alternative metric, Euclidean distance, does not work well: even similar texts tend not to overlap substantially in the content words they use, so that texts encountered in practice are all substantially orthogonal to each other, assuming that TF/IDF scores are used to reduce the influence of non-content words. The scores of two words in a textual attribute vector may be correlated; for example, "Kennedy" and "JFK" tend to appear in the same documents. Thus it may be advisable to alter the text somewhat before computing the scores of terms in the text, by using a synonym dictionary that groups together similar words. The effect of this optional pre-alteration is that two texts using related words are measured to be as similar as if they had actually used the same words. One technique is to augment the set of words actually found in the article with a set of synonyms or other words which tend to co-occur with the words in the article, so that "Kennedy" could be added to every article that mentions "JFK." Alternatively, words found in the article may be wholly replaced by synonyms, so that "JFK" might be replaced by "Kennedy" or by "John F. Kennedy" wherever it appears. In either case, the result is that documents about Kennedy and documents about JFK are adjudged similar. The synonym dictionary may be sensitive to the topic of the document as a whole; for example, it may recognize that "crane" is likely to have a different synonym in a document that mentions birds than in a document that mentions construction. A related technique is to replace each word by its morphological stem, so that "staple", "stapler", and "staples" are all replaced by "staple." Common function words ("a", "and", "the" ...) c an influence the calculated similarity of texts without regard to their topics, and so are typically removed from the text

IPR2025-00200 / 722 Patent Page 213 of 1171 EX1002

before the scores of terms in the text are computed. A more general approach to recognizing synonyms is to use a revised measure of the distance between textual attribute vectors V and U, namely arccos(AV(AU)^t /sqrt (AV(AV)^t AU(AU)^t), where the matrix A is the dimensionality-reducing linear transformation (or an approximation thereto) determined by collecting the vector values of the textual attribute, for all target objects known to the system, and applying singular value decomposition to the resulting collection. The same approach can be applied to the vector values of associative attributes. The above definitions allow us to determine how close together two target objects are with respect to a single attribute, whether numeric, associative, or textual. The distance between two target objects X and Y with respect to their entire multi-attribute profiles P_X and P_Y is then denoted d(X,Y) or d(P_X, P_Y) and defined as:

(((distance with respect to attribute a)(weight of attribute a))^k + ((distance with respect to attribute b)(weight of attribute b))^k + ((distance with respect to attribute c)(weight of attribute c))^k + ...)^k

where k is a fixed positive real number, typically 2, and the weights are non- negative real numbers indicating the relative importance of the various attributes. For example, if the target objects are consumer goods, and the weight of the "color" attribute is comparatively very small, then price is not a consideration in determining similarity: a user who likes a brown massage cushion is predicted to show equal interest in the same cushion manufactured in blue, and vice-versa. On the other hand, if the weight of the "color" attribute is comparatively very high, then users are predicted to show interest primarily in products whose colors they have liked in the past: a brown massage cushion and a blue massage cushion are not at all the same kind of target object, however similar in other attributes, and a good experience with one does not by itself inspire much interest in the other. Target objects may be of various sorts, and it is sometimes advantageous to use a single system that is able to compare tar get objects of distinct sorts. For example, in a system where some target objects are novels while other target objects are movies, it is desirable to judge a novel and a movie similar if their profiles show that similar users like them (an associative attribute). However, it is important to note that certain attributes specified in the movie's target profile are undefined in the novel's target profile, and vice versa: a novel has no "cast list" associative attribute and a movie has no "reading level" numeric attribute. In general, a system in which target objects fall into distinct sorts may sometimes have to measure the similarity of two target objects for which somewhat different

IPR2025-00200 / 722 Patent Page 214 of 1171 EX1002

sets of attributes are defined. This requires an extension to the distance metric d(*,*) defined above. In certain applications, it is sufficient when carrying out such a comparison simply to disregard attributes that are not defined for both target objects: this allows a cluster of novels to be matched with the most similar cluster of movies, for example, by considering only those attributes that novels and movies have in common. However, while this method allows comparisons between (say) novels and movies, it does not define a proper metric over the combined space of novel s and movies and therefore does not allow clustering to be applied to the set of all target objects. When necessary for clustering or other purposes, a metric that allows comparison of any two target objects (whether of the same or different sorts) can b e defined as follows. If a is an attribute, then let Max(a) be an upper bound on the distance between two values of attribute a; notice that if attribute a is an associative or textual attribute, this distance is an angle determined by arccos, so that Max(a) may be chosen to be 180 degrees, while if attribute a is a numeric attribute, a sufficiently large number must be selected by the system designers. The distance between two values of attribute a is given as before in the case where both values are defined; the distance between two undefined values is taken to be zero; finally, the distance between a defined value and an undefined value is always taken to be Max(a)/2. This allows us to determine how close together two target objects are with respect to an attribute a, even if attribute a does not have a defined value for both target objects. The distance d(*,*) between two target objects with respect to their entire multi-attribute profiles is then given in terms of these individual attribute distances exactly as before. It is assumed that one attribute in such a system specifies the sort of target object ("movie", "novel", etc.), and that this attribute may be highly weighted if target objects of different sorts are considered to be very different despite any attributes they may have in common.

UTILIZING THE SIMILARITY MEASUREMENT

Matching Buyers and Sellers

A simple application of the similarity measurement is a system to match buyers with sellers in small-volume markets, such as used cars and other used goods, artwork, or employment. Sellers submit profiles of the goods (target objects) they want to sell, and buyers submit profiles of the goods (target objects) they want to buy. Participants may submit or withdraw these profiles at any time. The system for customized electronic identification of desirable objects computes the similarities between seller-submitted profiles and buyer-submitted profiles, and when two profiles match closely (i.e., the

IPR2025-00200 / 722 Patent Page 215 of 1171 EX1002

similarity is above a threshold), the corresponding seller and buyer are notified of each other's identities. To prevent users from being flooded with responses, it may be desirable to limit the number of notifications each user receives to a fixed number, such as ten per day.

Filtering: Relevance Feedback

A filtering system is a device that can search through many target objects and estimate a given user's interest in each target object, so as to identify those that are of greatest interest to the user. The filtering system uses relevance feed back to refine its knowledge of the user's interests: whenever the filtering system identifies a target object as potentially interesting to a user, the user (if an on-line user) provides feedback as to whether or not that target object really is of interest. Such feedback is stored long-term in summarized form, as part of a database of user feedback information, and may be provided either actively or passively. In active feedback, the user explicitly indicates his or her interest, for instance, on a scale of -2 (active distaste) through 0 (no special interest) to 10 (great interest). In passive feedback, the system infers the user's interest from the user's behavior. For example, if target objects are textual documents, the system might monitor which documents the user chooses to read, or not to read, and how much time the user spends reading them. A typical formula for assessing interest in a document via passive feedback, in this domain, on a scale of 0 to 10, might be:

+ 2 if the second page is viewed,

+ 2 if all pages are viewed,

+ 2 if more than 30 seconds was spent viewing the document,

+ 2 if more than one minute was spent viewing the document,

+ 2 if the minutes spent viewing the document are greater than half the number of pages.

If the target objects are electronic mail messages, interest points might also be added in the case of a particularly lengthy or particularly prompt reply. If the target objects are purchasable goods, interest points might be added for target objects that the user actually purchases, with further points in the case of a large-quantity or high-price purchase. In any domain, further points might be added for target objects that the user accesses early in a session, on the grounds that users access the object s that most interest them first. Other potential sources of passive feedback include an electronic measurement of the extent to which the user's pupils dilate while the user views the target object or a description of the target object. It is possible to combine active and passive feedback. One option is to take

IPR2025-00200 / 722 Patent Page 216 of 1171 EX1002

a weighted average of the two ratings. Another option is to use passive feedback by default, but to allow the user to examine and actively modify the passive feedback score. In the scenario abo ve, for instance, an uninteresting article may sometimes remain on the display device for a long period while the user is engaged in unrelated business; the passive feedback score is then inappropriately high, and the user may wish to correct it before continuing. In the preferred embodiment of the invention, a visual indicator, such as a sliding bar or indicator needle on the user's screen, can be used to continuously display the passive feedback score estimated by the system for the target object being viewed, unless the user has manually adjusted the indicator by a mouse operation or other means in order to reflect a different score for this target object, after which the indicator displays the active feedback score selected by the user, and this active feedback score is used by the system instead of the passive feedback score. In a variation, the user cannot see or adjust the indicator until just after the user has finished viewing the target object. Regardless how a user's feedback is computed, it is stored long-term as part of that user's target profile interest summary.

Filtering: Determining Topical Interest Through Similarity

Relevance feedback only determines the user's interest in certain target objects: namely, the target objects that the user has actually had the opportunity to evaluate (whether actively or passively). For target objects that the user has not yet seen, the filtering system must estimate the user's interest. This estimation task is the heart of the filtering problem, and the reason that the similarity measurement is important. More concretely, the preferred embodiment of the filtering system is a news clipping service that periodically presents the user with news articles of potential interest. The user provides active and/or passive feedback to the system relating to these presented articles. However, the system does not have feedback information from the user for articles that have never been presented to the user, such as new articles that have just been added to the database, or old articles that the system chose not to present to the user. Similarly, in the dating service domain where target objects are prospective romantic partners, the system has only received feedback on old flames, not on prospective new loves.

As shown in flow diagram form in Figure 12, the evaluation of the likelihood of interest in a particular target object for a specific user can automatically be computed. The interest that a given target object X holds for a user U is assumed to be a sum of two quantities: q(U, X), the intrinsic "quality" of X, plus f(U, X), the "topical interest" that users

like U have in target objects like X. For any target object X, the intrinsic quality measure q(U, X) is easily estimated at steps 1201-1203 directly from numeric attributes of the target object X. The computation process begins at step 1201, where certain designated numeric attributes of target object X are specifically selected, which attributes by their very nature should be positively or negatively correlated with users' interest. Such attributes, termed "quality attributes," have the normative property that the higher (or in some cases lower) their value, the more interesting a user is expected to find them. Quality attributes of target object X may include, but are not limited to, target object X's popularity among users in general, the rating a particular reviewer has given target object X, the age (time since authorship -- also known as outdatedness) of target object X, the number of vulgar words used in target object X, the price of target object X, and the amount of money that the company selling target object X has donated to the user's favorite charity. At step 1202, each of the selected attributes is multiplied by a positive or negative weight indicative of the strength of user U's preference for those target objects that have high values for this attribute, which weight must be retrieved from a data file storing quality attribute weights for the selected user. At step 1203, a weighted sum of the identified weighted selected attributes is computed to determine the intrinsic quality measure q(U, X). At step 1204, the summarized weighted relevance feedback data is retrieved, wherein some relevance feedback points are weighted more heavily than others and the stored relevance data can be summarized to some degree, for example by the use of search profile sets. The more difficult part of determining user U's interest in target object X is to find or compute at step 1205 the value of f(U, X), which denotes the topical interest that users like U generally have in target objects like X. The method of determining a user's interest relies on the following heuristic: when X and Y are similar target objects (have similar attributes), and U and V are similar users (have similar attributes), then topical interest f(U, X) is predicted to have a similar value to the value of topical interest f(V, Y). This heuristic leads to an effective method because estimated values of the topical interest function f(*, *) are actually know n for certain arguments to that function: specifically, if user V has provided a relevance-feedback rating of r(V, Y) for target object Y, then insofar as that rating represents user V's true interest in target object Y, we have r(V, Y) = q(V, Y) + f(V, Y) and can estimate f(V, Y) as r(V, Y) - q(V, Y). Thus, the problem of estimating topical interest at all points becomes a problem of interpolating among these estimates of topical interest at selected points, such as the feedback estimate of f(V, Y) a s r(V, Y) - q(V, Y). This

interpolation can be accomplished with any standard smoothing technique, using as input the known point estimates of the value of the topical interest function f(*, *), and determining as output a function that approximates the entire topical interest function f(*, *).

Not all point estimates of the topical interest function f(*, *) should be given equal weight as inputs to the smoothing algorithm. Since passive relevance feedback is less reliable than active relevance feedback, point estimates made from passive relevance feedback should be weighted less heavily than point estimates made from active relevance feedback, or even not used at all. In most domains, a user's interests may change over time and, therefore, estimates of topic al interest that derive from more recent feedback should also be weighted more heavily. A user's interests may vary according to mood, so estimates of topical interest that derive from the current session should be weighted more heavily for the duration of the current session, and past estimates of topical interest made at approximately the current time of day or on the current weekday should be weighted more heavily. Finally, in domains where users are trying to locate target objects of long-term interest (investments, romantic partners, pen pals, employers, employees, suppliers, service providers) from the possibly meager information provided by the target profiles, the users are usually not in a position to provide reliable immediate feedback on a target object, but can provide reliable feedback at a later date. An estimate of topical interest f(V, Y) should be weighted more heavily if user V has had more experience with target object Y. Indeed, a useful strategy is for the system to track long-term feedback for such target objects. For example, if target profile Y was created in 1990 to describe a particular investment that was available in 1990, and that was purchased in 1990 by user V, then the system solicits relevance feedback from user V in the years 1990, 1991, 1992, 1993, 1994, 1995, etc., and treats these as successively stronger indications of user V's true interest in target profile Y, and thus as indications of user V's likely interest in new investments whose current profiles resemble the original 1990 investment profile Y. In particular, if in 1994 and 1995 user V is well-disposed toward his or her 1990 purchase of the investment described by target profile Y, then in those years and later, the system tends to recommend additional investments when they have profiles like target profile Y, on the grounds that they too will turn out to be satisfactory in 4 to 5 years. It makes these recommendations both to user V and to users whose investment portfolios and other attributes are similar to user V's. The relevance feedback provided by user V in this case may be either active (feedback =

IPR2025-00200 / 722 Patent Page 219 of 1171 EX1002

satisfaction ratings provided by the investor V) or passive (feedback = difference between average annual return of the investment and average annual return o f the Dow Jones index portfolio since purchase of the investment, for example).

To effectively apply the smoothing technique, it is necessary to have a definition of the similarity distance between (U, X) and (V, Y), for any users U and V and any target objects X and Y. We have already seen how to define the distance d(X, Y) between two target objects X and Y, given their attributes. We may regard a pair such as (U, X) as an extended object that bears all the attributes of target X and all the attributes of user U; then the distance between (U, X) and (V, Y) may be computed in exactly the same way. This approach requires user U, user V, and all other users to have some attributes of their own stored in the system: for example, age (numeric), social security number (textual), and list of documents previously retrieved (associative). It is these attributes that determine the notion of "similar users." Thus it is desirable to generate profiles of users (termed "user profiles") as well as profiles of target objects (termed "target profiles"). Some attributes employed for profiling users may be related to the attributes employed for profiling target objects: for example, using associative attributes, it is possible to characterize target objects such as X by the interest that various users have shown in them, and simultaneously to characterize users such as U by the interest that they have shown in various target objects. In addition, user profiles may make use of any attributes that are useful in characterizing humans, such as those suggested in the example domain above where target objects are potential consumers. Notice that user U's interest can be estimated even if user U is a new user or an off-line user who has never provided any feedback, because the relevance feedback of users whose attributes are similar to U's attributes is taken into account.

For some uses of filtering systems, when estimating topical interest, it is appropriate to make an additional "presumption of no topical interest" (or "bias toward zero"). To understand the usefulness of such a presumption, suppose the system needs to determine whether target object X is topically interesting to the user U, but that users like user U have never provided feedback on target objects even remotely like target object X. The presumption of no topical interest says that if this is so, it is because users like user U are simply not interested in such target objects and therefore do not seek them out and interact with them. On this presumption, the system should estimate topical interest f(U, X) to be low. Formally, this example has the characteristic that (U, X) is far away from all the points (V, Y) where feedback is available. In such a case, topical interest f(U, X) is presumed to

IPR2025-00200 / 722 Patent Page 220 of 1171 EX1002

be close to zero, even if the value of the topical interest function f(*, *) is high at all the faraway surrounding points at which its value is known. When a smoothing technique is used, such a presumption of no topical interest can be introduced, if appropriate, by manipulating t he input to the smoothing technique. In addition to using observed values of the topical interest function f(*, *) as input, the trick is to also introduce fake observations of the form topical interest f(V, Y) = 0 for a lattice of points (V, Y) distributed throughout the multidimensional space. These fake observations should be given relatively low weight as inputs to the smoothing algorithm. The more strongly they are weighted, the stronger the presumption of no interest.

The following provides another simple example of an estimation technique that has a presumption of no interest. Let g be a decreasing function from non-negative real numbers to non-negative real numbers, such as $g(x) = e^x$ or $g(x) = \min(1, x^{-k})$ where $k > 1$. Estimate topical interest f(U, X) with the following g-weighted average:

$$f(U,X) = \frac{\sum ((r(V,Y) - q(V,Y)) * g(distance(U,X),(V,Y))}{\sum g(distance(U,V),(V,Y))}$$

Here the summations are over all pairs (V, Y) such that user V has provided feedback r(V, Y) on target object Y, i.e., all pairs (V, Y) such that relevance feedback r(V, Y) is defined. Note that both with this technique and with conventional smoothing techniques, the estimate of the topical interest f(U, X) is not necessarily equal to r(U, X) - q(U, X), even when r(U, X) is defined.

Filtering: Adjusting Weights and Residue Feedback

The method described above requires the filtering system to measure distances between (user, target object) pairs, such as the distance between (U, X) and (V, Y). Given the means described earlier for measuring the distance between two multi-attribute profiles, the method must therefore associate a weight with each attribute used in the profile of (user, target object) pairs, that is, with each attribute used to profile either users or target objects. These weights specify the relative importance of the attributes in establishing similarity or difference, and therefore, in determining how topical interest is generalized from one (user, target object) pair to another. Additional weights determine which attributes of a target object contribute to the quality function q, and by how much. It is possible and often desirable for a filtering system to store a different set of weights for each user. For example, a user who thinks of two-star films as having materially different topic and style from

IPR2025-00200 / 722 Patent Page 221 of 1171 EX1002

four-star films wants to assign a high weight to "number of stars" for purposes of the similarity distance measure d(*, *); this means that interest in a two-star film does not necessarily signal interest in an otherwise similar four-star film, or vice-versa. If the user also agrees with the critics, and actually prefers four-star films, the user also wants to assign "number of stars" a high positive weight in the determination of the quality function q. In the same way, a user who dislikes vulgarity wants to assign the "vulgarity score" attribute a high negative weight in the determination of the quality function q, although the "vulgarity score" attribute does not necessarily have a high weight in determining the topical similarity of two films.

Attribute weights (of both sorts) may be set or adjusted by the system administrator or the individual user, on either a temporary or a permanent basis. However, it is often desirable for the filtering system to learn attribute weights automatically, based on relevance feedback. The optimal attribute weights for a user U are those that allow the most accurate prediction of user U's interests. That is, with the distance measure and quality function defined by these attribute weights, user U's interest in target object X, q(U, X) + f(U, X), can be accurately estimated by the techniques above. The effectiveness of a particular set of attribute weights for user U can therefore be gauged by seeing how well it predicts user U's known interests.

Formally, suppose that user U has previously provided feedback on target objects $X_1, X_2, X_3, \ldots X_n$, and that the feedback ratings are $r(U, X_1), r(U, X_2), r(U, X_3), \ldots r(U, X_n)$. Values of feedback ratings r(*,*) for other users and other target objects may also be known. The system may use the following procedure to gauge the effectiveness of the set of attribute weights it currently stores for user U: (I) For each $1 <= I <= n$, use the estimation techniques to estimate $q(U, X_1) + f(U, X_i)$ from all known values of feedback ratings r. Call this estimate a_i. (ii) Repeat step (i), but this time make the estimate for each $1 <= i <= n$ without using the feedback ratings $r(U, X_j)$ as input, for any j such that the distance $d(X_i, X_j)$ is smaller than a fixed threshold. That is, estimate each $q(U, X_i) + f(U, X_i)$ from other values of feedback rating r only; in particular, do not use $r(U, X_i)$ itself. Call this estimate b_i. The difference $a_i - b_i$ is herein termed the "residue feedback $r_{res}(U, X_i)$ of user U on target object X_i." (iii) Compute user U's error measure, $(a_1 - b_1)^2 + (a_2 - b_2)^2 + (a_3 - b_3)^2 + \ldots + (a_n - b_n)^2$.

A gradient-descent or other numerical optimization method may be used to adjust user U's attribute weights so that this error measure reaches a (local) minimum. This

IPR2025-00200 / 722 Patent Page 222 of 1171 EX1002

approach tends to work best if the smoothing technique used in estimation is such that the value of f(V, Y) is strongly affected by the point estimate r(V, Y) - q(V, Y) when the latter value is provided as input. Otherwise, the presence or absence of the single input feedback rating r(U, X_i), in steps (i)-(ii) may not make a_i and b_i very different from each other. A slight variation of this learning technique adjusts a single global set of at tribute weights for all users, by adjusting the weights so as to minimize not a particular user's error measure but rather the total error measure of all users. These global weights are used as a default initial setting for a new user who has not yet provided any feedback. Gradient descent can then be employed to adjust this user's individual weights over time. Even when the attribute weights are chosen to minimize the error measure for user U, the error measure is generally still positive, meaning that residue feedback from user U has not been reduced to 0 on all target objects. It is useful to note that high residue feedback from a user U on a target object X indicates that user U liked target object X unexpectedly well given its profile, that is, better than the smoothing model could predict from user U's opinions on target objects with similar profiles. Similarly, low residue feedback indicates that user U liked target object X less than was expected. By definition, this unexplained preference or dispreference cannot be the result of topical similarity, and therefore must be regarded as an indication of the intrinsic quality of target object X. It follows that a useful quality attribute for a target object X is the average amount of residue feedback r_{res}(V, X) from users on that target object, averaged over all users V who have provided relevance feedback on the target object. In a variation of this idea, residue feedback is never averaged indiscriminately over all users to form a new attribute, but instead is smoothed to consider users' similarity to each other. Recall that the quality measure q(U, X) depends on the user U as well as the target object X, so that a given target object X may be perceived by different users to have different quality. In this variation, as before, q(U, X) is calculated as a weighted sum of various quality attributes that are dependent only on X, but then an additional term is added, namely an estimate of r_{res} (U, X) found by applying a smoothing algorithm to known values of r_{res} (V, X). Here V ranges over all users who have provided relevance feedback on target object X, and the smoothing algorithm is sensitive to the distances d(U, V) from each such user V to user U.

Using the Similarity Computation for Clustering

A method for defining the distance between any pair of target objects was disclosed above. Given this distance measure, it is simple to apply a standard clustering algorithm,

IPR2025-00200 / 722 Patent Page 223 of 1171 EX1002

such as k- means, to group the target objects into a number of clusters, in such a way that similar target objects tend to be grouped in the same cluster. It is clear that the resulting clusters can be used to improve the efficiency of matching buyers and sellers in the application described in section "Matching Buyers and Sellers" above: it is not necessary to compare every buy profile to every sell profile, but only to compare buy profiles and sell profiles that are similar enough to appear in the same cluster. As explained below, the results of the clustering procedure can also be used to make filtering more efficient, and in the service of querying and browsing tasks.

The k-means clustering method is familiar to those skilled in the art. Briefly put, it finds a grouping of points (target profiles, in this case, whose numeric coordinates are given by numeric decomposition of their attributes as described above) to minimize the distance between points in the clusters and the centers of the clusters in which they are located. This is done by alternating between assigning each point to the cluster which has the nearest center and then, once the points have been assigned, computing the (new) center of each cluster by averaging the coordinates of the points (target profiles) located in this cluster. Other clustering methods can be used, such as "soft" or "fuzzy" k-means clustering, in which objects are allowed to belong to more than one cluster. This can be cast as a clustering problem similar to the k-means problem, but now the criterion being optimized is a little different:

$$\sum_i \sum_c i_{ic}\, d(x_i, \bar{x}_c)$$

where C ranges over cluster numbers, i ranges over target objects, x_i is the numeric vector corresponding to the profile of target object number i, $\bar{x}_C$ is the mean of all the numeric vectors corresponding to target profiles of target objects in cluster number C, termed the "cluster profile" of cluster C, d(*, *) is the metric used to measure distance between two target profiles, and i_{iC} is a value between 0 and 1 that indicates how much target object number i is associated with cluster number C, where i is an indicator matrix with the property that for each i, SUM SUB C I SUB iC = 1. For k-means clustering, i_{iC} is either 0 or 1.

Any of these basic types of clustering might be used by the system:

1) Association-based clustering, in which profiles contain only associative attributes, and thus distance is defined entirely by associations. This kind of

clustering generally (a) clusters target objects based on the similarity of the users who like them or (b) clusters users based on the similarity of the target objects they like. In this approach, the system does not need any information about target objects or users, except for their history of interaction with each other.

2) Content-based clustering, in which profiles contain only non-associative attributes. This kind of clustering (a) clusters target objects based on the similarity of their non-associative attributes (such as word frequencies) or (b) clusters users base d on the similarity of their non-associative attributes (such as demographics and psychographics). In this approach, the system does not need to record any information about users' historical patterns of information access, but it does need information about the intrinsic properties of users and/or target objects.

3) Uniform hybrid method, in which profiles may contain both associative and non-associative attributes. This method combines 1a and 2a, or 1b and 2b. The distance $d(P_X, P_Y)$ between two profiles P_X and P_Y may be computed by the general similarity-measurement methods described earlier.

4) Sequential hybrid method. First apply the k-means procedure to do 1a, so that articles are labeled by cluster based on which user read them, then use supervised clustering (maximum likelihood discriminant methods) using the word frequencies to do the process of method 2a described above. This tries to use knowledge of who read what to do a better job of clustering based on word frequencies. One could similarly combine the methods 1b and 2b described above.

Hierarchical clustering of target objects is often useful. Hierarchical clustering produces a tree which divides the target objects first into two large clusters of roughly similar objects; each of these clusters is in turn divided into two or more smaller clusters, which in turn are ea ch divided into yet smaller clusters until the collection of target objects has been entirely divided into "clusters" consisting of a single object each, as diagrammed in Figure 8 In this diagram, the node d denotes a particular target object d, or equivalently, a single-member cluster consisting of this target object. Target object d is a member of the cluster (a, b, d), which is a subset of the cluster (a, b, c, d, e, f), which in turn is a subset of all target objects. The tree shown in Figure 8 would be produced from a set of target objects such as those shown geometrically in Figure 7. In Figure 7, each letter represents a target object, and axes x1 and x2 represent two of the many numeric attributes on which the target

IPR2025-00200 / 722 Patent Page 225 of 1171 EX1002

objects differ. Such a cluster tree may be created by hand, using human judgment to form clusters and subclusters of similar objects, or may be created automatically in either of two standard ways: top-down or bottom-up. In top-down hierarchical clustering, the set of all target objects in Figure 7 would be divided into the clusters (a, b, c, d, e, f) and (g, h, i, j, k). The clustering algorithm would then be reapplied to the target objects in each cluster, so that the cluster (g, h, i, j, k) is subpartitioned into the clusters (g, k) and (h, i, j), and so on to arrive at the tree shown in Figure 8. In bottom-up hierarchical clustering, the set of all target objects in Figure 7 would be grouped into numerous small clusters, namely (a, b), d, (c, f), e, (g,k), (h, i), and j. These clusters would then themselves be grouped into the larger clusters (a, b, d), (c, e, f), (g, k), and (h, i, j), according to their cluster profiles. These larger clusters would themselves be grouped into (a, b, c, d, e, f) and (g, k, h, i, j), and so on until all target objects had been grouped together, resulting in the tree of Figure 8. Note that for bottom-up clustering to work, it must be possible to apply the clustering algorithm to a set of existing clusters. This requires a notion of the distance between two clusters. The method disclosed above for measuring the distance between target objects can be applied directly, provided that clusters are profiled in the same way as target objects. It is only necessary to adopt the convention that a cluster's profile is the average of the target profiles of all the target objects in the cluster; that is, to determine the cluster's value for a given attribute, take the mean value of that attribute across all the target objects in the cluster. For the mean value to be well-defined, all attributes must be numeric, so it is necessary as usual to replace each textual or associative attribute with its decomposition into numeric attributes (scores), as described earlier. For example, the target profile of a single Woody Allen film would assign "Woody-Allen" a score of 1 in the "name-of-director" field, while giving "Federico-Fellini" and "Terence-Davies" scores of 0. A cluster that consisted of 20 films directed by Allen and 5 directed by Fellini would be profiled with scores of 0.8, 0.2, and 0 respectively, because, for example, 0.8 is the average of 20 ones and 5 zeros.

Searching for Target Objects

Given a target object with target profile P, or alternatively given a search profile P, a hierarchical cluster tree of target objects makes it possible for the system to search efficiently for target objects with target profiles similar to P. It is only necessarily to navigate through the tree, automatically, in search of such target profiles. The system for customized electronic identification of desirable objects begins by considering the largest, top-level clusters, and selects the cluster whose profile is most similar to target profile P.

IPR2025-00200 / 722 Patent Page 226 of 1171 EX1002

In the event of a near-tie, multiple clusters may be selected. Next, the system considers all subclusters of the selected clusters, and this time selects the subcluster or subclusters whose profiles are closest to target profile P. This refinement process is iterated until the clusters selected on a given step are sufficiently small, and these are the desired clusters of target objects with profiles most similar to target profile P. Any hierarchical cluster tree therefore serves as a decision tree for identifying target objects. In pseudo-code form, this process is as follows (and in flow diagram form in Figures 13A and 13B):

1. Initialize list of identified target objects to the empty list at step 13A00
2. Initialize the current tree T to be the hierarchical cluster tree of all objects at step 13A01 and at step 13A02 scan the current cluster tree for target objects similar to P, using the process detailed in Figure 13B. At step 13A03, the list of target objects is returned.
3. At step 13B00, the variable I is set to 1 and for each child subtree Ti of the root of tree T, is retrieved.
4. At step 13B02, calculate $d(P, p_i)$, the similarity distance between P and p_i,
5. At step 13B03, if $d(P, p_i) < t$, a threshold, branch to one of two options
6. If tree Ti contains only one target object at step 13B04, add that target object to list of identified target objects at step 13B05 and advance to step 13B07.
7. If tree Ti contains multiple target objects at step 13B04, scan the ith child subtree for target objects similar to P by invoking the steps of the process of Figure 13B recursively and then recurse to step 3 (step 13A01 in Figure 13A) with T bound for the duration of the recursion to tree Ti, in order to search in tree Ti for target objects with profiles similar to P.

In step 5 of this pseudo-code, smaller thresholds are typically used at lower levels of the tree, for example by making the threshold an affine function or other function of the cluster variance or cluster diameter of the cluster p_i. If the cluster tree is distributed across a plurality of servers, as described in the section of this description titled "Network Context of the Browsing System", this process may be executed in distributed fashion as follows: steps 3-7 are executed by the server that stores the root node of hierarchical cluster tree T, and the recursion in step 7 to a subcluster tree T_i involves the transmission of a search request to the server that stores the root node of tree T_i, which server carries out the recursive step upon receipt of this request. Steps 1-2 are carried out by the processor that

initiates the search, and the server that executes step 6 must send a message identifying the target object to this initiating processor, which adds it to the list.

Assuming that low-level clusters have been already been formed through clustering, there are alternative search methods for identifying the low-level cluster whose profile is most similar to a given target profile P. A standard back-propagation neural net is one such method: it should be trained to take the attributes of a target object as input, and produce as output a unique pattern that can be used to identify the appropriate low-level cluster. For maximum accuracy, low-level clusters that are similar to each other (close together in the cluster tree) should be given similar identifying patterns. Another approach is a standard decision tree that considers the attributes of target profile P one at a time until it can identify the appropriate cluster. If profiles are large, this may be more rapid than considering all attributes. A hybrid approach to searching uses distance measurements as described above to navigate through the top few levels of the hierarchical cluster tree, until it reaches an cluster of intermediate size whose profile is similar to target profile P, and then continues by using a decision tree specialized to search for low-level subclusters of that intermediate cluster.

One use of these searching techniques is to search for target objects that match a search profile from a user's search profile set. This form of searching is used repeatedly in the news clipping service, active navigation, and Virtual Community Service applications, described below. Another use is to add a new target object quickly to the cluster tree. An existing cluster that is similar to the new target object can be located rapidly, and the new target object can be added to this cluster. If the object is beyond a certain threshold distance from the cluster center, then it is advisable to start a new cluster. Several variants of this incremental clustering scheme can be used, and can be built using variants of subroutines available in advanced statistical packages. Note that various methods can be used to locate t he new target objects that must be added to the cluster tree, depending on the architecture used. In one method, a "webcrawler" program running on a central computer periodically scans all servers in search of new target objects, calculates the target profiles of these objects, and adds them to the hierarchical cluster tree by the above method. In another, whenever a new target object is added to any of the servers, a software "agent" at that server calculates the target profile and adds it to the hierarchical cluster tree by the above method.

IPR2025-00200 / 722 Patent Page 228 of 1171 EX1002

Rapid Profiling

In some domains, complete profiles of target objects are not always easy to construct automatically. When target objects are wallpaper patterns, for example, an attribute such as "genre" (a single textual term such as "Art-Deco," "Children's," "Rustic," etc.) may be a matter of judgment and opinion, difficult to determine except by consulting a human. More significantly, if each wallpaper pattern has an associative attribute that records the positive or negative relevance feedback to that pattern from various human users (consumers), then all the association scores of any newly introduced pattern are initially zero, so that it is initially unclear what other patterns are similar to the new pattern with respect to the users who like them. Indeed, if this associative attribute is highly weighted, the initial lack of relevance feedback information may be difficult to remedy, due to a vicious circle in which users of moderate-to-high interest are needed to provide relevance feedback but relevance feedback is needed to identify users of moderate-to-high interest. Fortunately, however, it is often possible in principle to determine certain attributes of a new target object by extraordinary methods, including but not limited to methods that consult a human. For example, the system can in principle determine the genre of a wallpaper pattern by consulting one or more randomly chosen individuals from a set of known human experts, while to determine the numeric association score between a new wallpaper pattern and a particular user, it can in principle show the pattern to the that user and obtain relevance feedback. Since such requests inconvenience people, however, it is important not to determine all difficult attributes this way, but only the ones that are most important for purposes of classifying the document. "Rapid profiling" is a method for selecting those numeric attributes that are most important to determine. (Recall that all attributes can be decomposed into numeric attributes, such as association scores or term scores.) First, a set of existing target objects that already have complete or largely complete profiles are clustered using a k-means algorithm. Next, each of the resulting clusters is assigned a unique identifying number, and each clustered target object is labeled with the identifying number of its cluster. Standard methods then allow construction of a single decision tree that can determine any target object's cluster number, with substantial accuracy, by considering the attributes of the target object, one at a time. Only attributes that can if necessary be determined for any new target object are used in the construction of this decision tree. To profile a new target object, the decision tree is traversed downward from its root as far as is desired. The root of the decision tree considers some attribute of the

IPR2025-00200 / 722 Patent Page 229 of 1171 EX1002

target object. If the value of this attribute is not yet known, it is determined by a method appropriate to that attribute; for example, if the attribute is the association score of the target object with user #4589, then relevance feedback (to be used as the value of this attribute) is solicited from user #4589, perhaps by the ruse of adding the possibly uninteresting target object to a set of objects that the system recommends to the user's attention, in order to find out what the user thinks of it. Once the root attribute is determined, the rapid profiling method descends the decision tree by one level, choosing one of the decision subtrees of the root in accordance with the determined value of the root attribute. The root of this chosen subtree considers another attribute of the target object, whose value is likewise determined by an appropriate method. The process c an be repeated to determine as many attributes as desired, by whatever methods are available, although it is ordinarily stopped after a small number of attributes, to avoid the burden of determining too many attributes.

It should be noted that the rapid profiling method can be used to identify important attributes in any sort of profile, and not just profiles of target objects. In particular, recall that the disclosed method for determining topical interest through similarity requires users as well as target objects to have profiles. New users, like new target objects, may be profiled or partially profiled through the rapid profiling process. For example, when user profiles include an associative attribute that records the user's relevance feedback on a ll target objects in the system, the rapid profiling procedure can rapidly form a rough characterization of a new user's interests by soliciting the user's feedback on a small number of significant target objects, and perhaps also by determining a small n umber of other key attributes of the new user, by on-line queries, telephone surveys, or other means. Once the new user has been partially profiled in this way, the methods disclosed above predict that the new user's interests resemble the known interests of other users with similar profiles. In a variation, each user's user profile is subdivided into a set of long-term attributes, such as demographic characteristics, and a set of short-term attributes that help to identify the user's temporary desires and emotional state, such as the user's textual or multiple-choice answers to questions whose answers reflect the user's mood. A subset of the user's long-term attributes are determined when the user first registers with the system, through the use of a rapid profiling tree of long-term attributes. In addition, each time the user logs on to the system, a subset of the user's short-term attributes are additionally determined, through the use of a separate rapid profiling tree that asks about short-term attributes.

Market Research

IPR2025-00200 / 722 Patent Page 230 of 1171 EX1002

A technique similar to rapid profiling is of interest in market research (or voter research). Suppose that the target objects are consumers. A particular attribute in each target profile indicates whether the consumer described by that target profile h as purchased product X. A decision tree can be built that attempts to determine what value a consumer has for this attribute, by consideration of the other attributes in the consumer's profile. This decision tree may be traversed to determine whether additional users are likely to purchase product X. More generally, the top few levels of the decision tree provide information, valuable to advertisers who are planning mass-market or direct-mail campaigns, about the most significant characteristics of consumers of product X.

Similar information can alternatively be extracted from a collection of consumer profiles without recourse to a decision tree, by considering attributes one at a time, and identifying those attributes on which pro duct X's consumers differ significantly from its non-consumers. These techniques serve to characterize consumers of a particular product; they can be equally well applied to voter research or other survey research, where the objective is to characterize those individuals from a given set of surveyed individuals who favor a particular candidate, hold a particular opinion, belong to a particular demographic group, or have some other set of distinguishing attributes. Researchers may wish to purchase batches of analyzed or unanalyzed user profiles from which personal identifying information has been removed. As with any statistical database, statistical conclusions can be drawn, and relationships between attributes can be elucidated using knowledge discovery techniques which are well known in the art.

SUPPORTING ARCHITECTURE

The following section describes the preferred computer and network architecture for implementing the methods described in this patent.

Electronic Media System Architecture

Figure 1 illustrates in block diagram form the overall architecture of an electronic media system, known in the art, in which the system for customized electronic identification of desirable objects of the present invention can be used to provide user customized access to target objects that are available via the electronic media system. In particular, the electronic media system comprises a data communication facility that interconnects a plurality of users with a number of information servers. The users a re typically individuals, whose personal computers (terminals) T_1-T_n are connected via a data communications link, such as a modem and a telephone connection established in well-known fashion, to a

telecommunication network N. User information access software is resident on the user's personal computer and serves to communicate over the data communications link and the telecommunication network N with one of the plurality of network vendors V_1-V_k (America Online, Prodigy, CompuServe, other private companies or even universities) who provide data interconnection service with selected ones of the information servers I_1 -I_m. The user can, by use of the user information access software, interact with the information servers I_1 -I_m to request and obtain access to data that resides on mass storage systems -SS_m that are part of the information server apparatus. New data is input to this system y users via their personal computers T_1 -T_n and by commercial information services by populating their mass storage systems SS_1 -SS_m with commercial data. Each user terminal T -T and the information servers I_1 -I_m have phone numbers or IP addresses on the network N which enable a data communication link to be established between a particular user terminal T_1 -T_n and the selected information server I_1 -I_m. A user's electronic mail address also uniquely identifies the user and the user's network vendor V_1 -V_k in an industry-standard format such as: username@aol.com or username@netcom.com. The network vendors V_1 -V_k provide access passwords for their subscribers (selected users), through which the users can access the information servers I_1 -I_m. The subscribers pay the network vendors V_1 -V_k for the access services on a fee schedule that typically includes a monthly subscription fee and usage based charges. A difficulty with this system is that there are numerous information servers I_1 -I_m located around the world, each of which provides access to a set of information of differing format, content and topics and via a cataloging system that is typically unique to the particular information server I_1 -I_m. The information is comprised of individual "files," which can contain audio data, video data, graphics data, text data, structured database data and combinations thereof. In the terminology of this patent, each target object is associated with a unique file: for target objects that are informational in nature and can be digitally represented, the file directly stores the informational content of the target object, while for target objects that are not stored electronically, such as purchasable goods, the file contains an identifying description of the target object. Target objects stored electronically as text files can include commercially provided news articles, published documents , letters, user-generated documents, descriptions of physical objects, or combinations of these classes of data. The organization of the files containing the information and the native format of the data contained in files of the same conceptual type may vary by information server I_1 -I_m.

IPR2025-00200 / 722 Patent Page 232 of 1171 EX1002

Thus, a user can have difficulty in locating files that contain the desired information, because the information may be contained in files whose information server cataloging may not enable the user to locate them. Furthermore, there is no standard catalog that defines the presence and services provided by all information servers I_1 -I_m. A user therefore does not have simple access to information but must expend a significant amount of time and energy to excerpt a segment of the information that may be relevant to the user from the plethora of information that is generated and populated on this system. Even if the user commits the necessary resources to this task, existing information retrieval processes lack the accuracy and efficiency to ensure that the user obtains the desired information. It is obvious that within the constructs of this electronic media system, the three modules of the system for customized electronic identification of desirable objects can be implemented in a distributed manner, even with various modules being implemented on and/or by different vendors within the electronic media system. For example, the information servers I_1 -I_m can include the target profile generation module while the network vendors V_1 -V_k may implement the user profile generation module, the target profile interest summary generation module, and/or the profile processing module. A module can itself be implemented in a distributed manner, with numerous nodes being present in the network N, each node serving a population of users in a particular geographic area. The totality of these nodes comprises the functionality of the particular module. Various other partitions of the modules and their functions are possible and the examples provided herein represent illustrative examples and are not intended to limit the scope of the claimed invention. For the purposes of pseudonymous creation and update of users' target profile interest summaries (as described below), the vendors V_1 -V_k may be augmented with some number of proxy servers, which provide a mechanism for ongoing pseudonymous access and profile building through the method described herein. At least one trusted validation server must be in place to administer the creation of pseudonyms in the system.

An important characteristic of this system for customized electronic identification of desirable objects is its responsiveness, since the intended use of the system is in an interactive mode. The system utility grows with the number of the users and this increases the number of possible consumer/product relationships between users and target objects. A system that serves a large group of users must maintain interactive performance and the disclosed method for profiling and clustering target objects and users can in turn be used for

IPR2025-00200 / 722 Patent Page 233 of 1171 EX1002

optimizing the distribution of data among the members of a virtual community and through a data communications network, based on users' target profile interest summaries.

Network Elements and System Characteristics

The various processors interconnected by the data communication network N as shown in Figure 1 can be divided into two classes and grouped as illustrated in Figure 2: clients and servers. The clients C1-Cn are individual user's computer systems which are connected to servers S1-S5 at various times via data communications links. Each of the clients Ci is typically associated with a single server Sj, but these associations can change over time. The clients C1-Cn both interface with users and produce and retrieve files to and from servers. The clients C1-Cn are not necessarily continuously on-line, since they typically serve a single user and can be movable systems, such as laptop computers, which can be connected to the data communications network N at any of a number of locations. Clients could also be a variety of other computers, such as computers and kiosks providing access to customized information as well as targeted advertising to many users, where the users identify themselves with passwords or with smart cards. A server Si is a computer system that is presumed to be continuously on-line and functions to both collect files from various sources on the data communication network N for access by local clients C1-Cn and collect files from local clients C1-Cn for access by remote clients. The server Si is equipped with persistent storage, such as a magnetic disk data storage medium, and are interconnected with other servers via data communications links. The data communications links can be of arbitrary topology and architecture, and are described herein for the purpose of simplicity as point-to-point links or, more precisely, as virtual point-to-point links. The servers S1-S5 comprise the network vendors V1-Vk as well as the information servers I_1 -I_m of Figure 1 and the functions performed by these two classes of modules can be merged to a greater or lesser extent in a single server Si or distributed over a number of servers in the data communication network N. Prior to proceeding with the description of the preferred embodiment of the invention, a number of terms are defined. Figure 3 illustrates in block diagram form a representation of an arbitrarily selected network topology for a plurality of servers A-D, each of which is interconnected to at least one other server and typically also to a plurality of clients p-s. Servers A-D are interconnected by a collection of point to point data communications links, and server A is connected to client r, server B is connected to clients p-q, while server D is connected to client s. Servers transmit encrypted or unencrypted messages amongst themselves: a message typically contains the textual and/or

IPR2025-00200 / 722 Patent Page 234 of 1171 EX1002

graphic information stored in a particular file, and also contains data which describe the type and origin of this file, the name of the server that is supposed to receive the message, and the purpose for which the file contents are being transmitted. Some messages are not associated with any file, but are sent by one server to other servers for control reasons, for example to request transmission of a file or to announce the availability of a new file. Messages can be forwarded by a server to another server, as in the case where server A transmits a message to server D via a relay node of either server C or servers B, C. It is generally preferable to have multiple paths through the network, with each path being characterized by its performance capability and cost to enable the network N to optimize traffic routing.

Proxy Servers and Pseudonymous Transactions

While the method of using target profile interest summaries presents many advantages t o both target object providers and users, there are important privacy issues for both users and providers that must be resolved if the system is to be used freely and without inhibition by users without fear of invasion of privacy. It is likely that user s desire that some, if not all, of the user-specific information in their user profiles and target profile interest summaries remain confidential, to be disclosed only under certain circumstances related to certain types of transactions and according to t heir personal wishes for differing levels of confidentiality regarding their purchases and expressed interests.

However, complete privacy and inaccessibility of user transactions and profile summary information would hinder implementation of the system for customized electronic identification of desirable objects and would deprive the user of many of the ad vantages derived through the system's use of user-specific information. In many cases, complete and total privacy is not desired by all parties to a transaction. For example, a buyer may desire to be targeted for certain mailings that describe products that are related to his or her interests, and a seller may desire to target users who are predicted to be interested in the goods and services that the seller provides. Indeed, the usefulness of the technology described herein is contingent upon the ability of the system to collect and compare data about many users and many target objects. A compromise between total user anonymity and total public disclosure of the user's search profiles or target profile interest summary is a pseudonym. A pseudonym is an artifact that allows a service provider to communicate with users and build and accumulate records of their preferences over time, while at the same time remaining ignorant of the users' true identities, so that users can keep their purchases

or preferences private. A second and equally important requirement of a pseudonym system is that it provide for digital credentials, which are used to guarantee that the user represented by a particular pseudonym has certain properties. These credentials may be granted on the basis of result of activities and transactions conducted by means of the system for customized electronic identification of desirable objects, or on the basis of other activities and transactions conducted on the network N of the present system, on the basis of users' activities outside of network N. For example, a service provider may require proof that the purchaser has sufficient funds on deposit at his/her bank, which might possibly not be on a network, before agreeing to transact business with that user. The user, therefore, must provide the service provider with proof of funds (a credential) from the bank, while still not disclosing the user's true identity to the service provider.

Our method solves the above problems by combining the pseudonym granting and credential transfer methods taught by D. Chaum and J.H. Evertse, in the paper titled "A secure and privacy-protecting protocol for transmitting personal information between organizations," with the implementation of a set of one or more proxy servers distributed throughout the network N. Each proxy server, for example S2 in Figure 2, is a server which communicates with clients and other servers S5 in the network either directly or through anonymizing mix paths as detailed in the paper by D.Chaum titled "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms," published in Communications of the ACM, Volume 24, Number 2, February 1981. Any server in the network N may be configured to act as a proxy server in addition to its other functions. Each proxy server provides service to a set of users, which set is termed the "user base" of that proxy server. A given proxy server provides three sorts of service to each user U in its user base, as follows:

1. The first function of the proxy server is to bidirectionally transfer communications between user U and other entities such as information servers (possibly including the proxy server itself) and/or other users. Specifically, letting S denote the server that is directly associated with user U's client processor, the proxy server communicates with server S (and thence with user U), either through anonymizing mix paths that obscure the identity of server S and user U, in which case the proxy server knows user U only through a secure pseudonym, or else through a conventional virtual point-to-point connection, in which case the proxy

server knows user U by user U's address at server S, which address may be regarded as a non-secure pseudonym for user U.

2. A second function of the proxy server is to record user-specific information associated with user U. This user-specific information includes a user profile and target profile interest summary for user U, as well as a list of access control instructions specified by user U, as described below, and a set of one-time return addresses provided by user U that can be used to send messages to user U without knowing user U's true identity. All of this user-specific information is stored in a database that is keyed by user U's pseudonym (whether secure or non-secure) on the proxy server.

3. A third function of the proxy server is to act as a selective forwarding agent for unsolicited communications that are addressed to user U: the proxy server forwards some such communications to user U and rejects others, in accordance with the access control instructions specified by user U.

Our combined method allows a given user to use either a single pseudonym in all transactions where he or she wishes to remain pseudonymous, or else different pseudonyms for different types of transactions. In the latter case, each service provider might transact with the user under a different pseudonym for the user. More generally, a coalition of service providers, all of whom match users with the same genre of target objects, might agree to transact with the user using a common pseudonym, so that the target profile interest summary associated with that pseudonym would be complete with respect to said genre of target objects. When a user employs several pseudonyms in order to transact with different coalitions of service providers, the user may freely choose a proxy server to service each pseudonym; these proxy servers may be the same or different.

From the service provider's perspective, our system provides security, in that it can guarantee that users of a service are legitimately entitled to the services used and that no user is using multiple pseudonyms to communicate with the same provider. This uniqueness of pseudonyms is important for the purposes of this application, since the transaction information gathered for a given individual must represent a complete and consistent picture of a single user's activities with respect to a given service provider or coalition of service providers; otherwise, a user's target profile interest summary and user profile would not be able to represent the user's interests to other parties as completely and accurately as possible.

IPR2025-00200 / 722 Patent Page 237 of 1171 EX1002

The service provider must have a means of protection from users who violate previously agreed upon terms of service. For example, if a user that uses a given pseudonym engages in activities that violate the terms of service, then the service provider should be able to take action against the user, such as denying the user service and blacklisting the user from transactions with other parties that the user might be tempted to defraud. This type of situation might occur when a user employs a service provider for illegal activities or defaults in payments to the service provider. The method of the paper titled "Security without identification: Transaction systems to make Big-Brother obsolete", published in the Communications of the ACM, 28(10), Oct. 1985; pp.1030-1044, incorporated herein, provides for a mechanism to enforce protection against this type of behavior through the use of resolution credentials, which are credentials that are periodically provided to individuals contingent upon their behaving consistent with the agreed upon terms of service between the user and information provider and network vendor entities (such as regular payment for services rendered, civil conduct, etc.). For the user's safety, if the issuer of a resolution credential refuses to grant this resolution credential to the user, then the refusal may be appealed to an adjudicating third party. The integrity of the user profiles and target profile interest summaries stored on proxy servers is important: if a seller relies on such user-specific information to deliver promotional offers or other material to a particular class of users, but not to other users, then the user-specific information must be accurate and untampered with in any way. The user may likewise wish to ensure that other parties not tamper with the user's user profile and target profile interest summary, since such modification could degrade the system's ability to match the user with the most appropriate target objects. This is done by providing for the user to apply digital signatures to the control messages sent by the user to the proxy server. Each pseudonym is paired with a public cryptographic key and a private cryptographic key, where the private key is known only to the user who holds that pseudonym; when t he user sends a control message to a proxy server under a given pseudonym, the proxy server uses the pseudonym's public key to verify that the message has been digitally signed by someone who knows the pseudonym's private key. This prevents other parties from masquerading as the user.

Our approach, as disclosed in this application, provides an improvement over the prior art in privacy-protected pseudonymy for network subscribers such as taught in U.S. Patent 5,245,656, which provides for a name translator station to act as an intermediary between a service provider and the user. However, while U.S. Patent 5,245,656 provides

IPR2025-00200 / 722 Patent Page 238 of 1171 EX1002

that the information transmitted between the end user U and the service provider be doubly encrypted, the fact that a relationship exist s between user U and the service provider is known to the name translator, and this fact could be used to compromise user U, for example if the service provider specializes in the provision of content that is not deemed acceptable by user U's peers. The method of U.S. Patent 5,245,656 also omits a method for the convenient updating of pseudonymous user profile information, such as is provided in this application, and does not provide for assurance of unique and credentialed registration of pseudonyms from a credentialing agent as is also provided in this application, and does not provide a means of access control to the user based on profile information and conditional access as will be subsequently described. The method described by Loeb et al. also does not describe any provision for credentials, such as might be used for authenticating a user's right to access particular target objects, such as target objects that are intended to be available only upon payment of a subscription fee, or target objects that are intend ed to be unavailable to younger users.

Proxy Server Description

In order that a user may ensure that some or all of the information in the user's user profile and target profile interest summary remain dissociated from the user's true identity, the user employs as an intermediary any one of a number of proxy servers available on the data communication network N of Figure 2 (for example, server S2). The proxy servers function to disguise the true identity of the user from other parties on the data communication network N. The proxy server represents a given us er to either single network vendors and information servers or coalitions thereof. A proxy server, e.g. S2, is a server computer with CPU, main memory, secondary disk storage and network communication function and with a database function which retrieves the target profile interest summary and access control instructions associated with a particular pseudonym P, which represents a particular user U, and performs bi-directional routing of commands, target objects and billing information between the user at a given client (e.g. C3) and other network entities such as network vendors V1-Vk and information servers I1-Im. Each proxy server maintains an encrypted target profile interest summary associated with each allocated pseudonym in its pseudonym database D . The actual user-specific information and the associated pseudonyms need not be stored locally on the proxy server, but may alternatively be stored in a distributed fashion and be remotely addressable from the proxy server via point-to-point connections .

The proxy server supports two types of bi-directional connections: point-to-point connections and pseudonymous connections through mix paths, as taught by D.Chaum in the paper titled "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms", Communications of the ACM, Volume 24, Number 2, February 1981. The normal connections between the proxy server and information servers, for example a connection between proxy server S2 and information server S4 in Figure 2, are accomplished through the point-to-point connection protocols provided by network N as described in the "Electronic Media System Architecture" section of this application. The normal type of point-to-point connections may be used between S2-S4, for example, since the dissociation of the user and the pseudonym need only occur between the client C3 and the proxy server S2, where the pseudonym used by the user is available. Knowing that an information provider such as S4 communicates with a given pseudonym P on proxy server S2 does not compromise the true identity of user U. The bidirectional connection between the user and the proxy server S2 can also be a normal point-to-point connection, but it may instead be made anonymous and secure, if the user desires, though the consistent use of an anonymizing mix protocol as taught by D.Chaum in the paper titled "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms", Communications of the ACM, Volume 24, Number 2, February 1981. This mix procedure provides untraceable secure anonymous mail between to parties with blind return addresses through a set of forwarding and return routing servers termed "mixes". The mix routing protocol, as taught in the Chaum paper, is used with the proxy server S2 to provide a registry of persistent secure pseudonyms that can be employed by users other than user U, by information providers I1-Im, by vendors V1-Vk and by other proxy servers to communicate with the users in the proxy server's user base on a continuing basis. The security provided by this mix path protocol is distributed and resistant to traffic analysis attacks and other known forms of analysis which may be used by malicious parties to try and ascertain the true identity of a pseudonym bearer. Breaking the protocol requires a large number of parties to maliciously collude or be cryptographically compromised. In addition an extension to the method is taught where the user can include a return path definition in the message so the information server S4 can return the requested information to the user's client processor C3 . We utilize this feature in a novel fashion to provide for access and reachability control under user and proxy server control.

Validation and Allocation of a Unique Pseudonym

Chaum's pseudonym and credential issuance system, as described in a publication by D. Chaum and J.H. Evertse, titled "A secure and privacy-protecting protocol for transmitting personal information between organizations," has several desirable properties for use as a component in our system. The system allows for individuals to use different pseudonyms with different organizations (such as banks and coalitions of service providers). The organizations which are presented with a pseudonym have no more information about the individual than the pseudonym itself and a record of previous transactions carried out under that pseudonym. Additionally, credentials, which represent facts about a pseudonym that an organization is willing to certify, can be granted to a particular pseudonym, and transferred to other pseudonyms that the same user employs. For, example, the user can use different pseudonyms with different organizations (or disjoint sets of organizations), yet still present credentials that were granted by one organization, under one pseudonym, in order to transact with another organization under another pseudonym, without revealing that the two pseudonyms correspond to the same user. Credentials may be granted to provide assurances regarding the pseudonym bearer's age, financial status, legal status, and the like. For example, credentials signifying "legal adult" may be issued to a pseudonym based on information known about the corresponding user by the given is suing organization. Then, when the credential is transferred to another pseudonym that represents the user to another disjoint organization, presentation of this credential on the other pseudonym can be taken as proof of legal adulthood, which might satisfy a condition of terms of service. Credential-issuing organizations may also certify particular facts about a user's demographic profile or target profile interest summary, for example by granting a credential that asserts "the bearer of this pseudonym is either well-read or is middle-aged and works for a large company"; by presenting this credential to another entity, the user can prove eligibility for (say) a discount without revealing the user 's personal data to that entity.

Additionally, the method taught by Chaum provides for assurances that no individual may correspond with a given organization or coalition of organizations using more than one pseudonym; that credentials may not be feasibly forged by the user; and t hat credentials may not be transferred from one user's pseudonym to a different user's pseudonym. Finally, the method provides for expiration of credentials and for the issuance of "black marks" against Individuals who do not act according to the terms of service that they are extended. This is done through the resolution credential mechanism as described in Chaum's work,

IPR2025-00200 / 722 Patent Page 241 of 1171 EX1002

in which resolutions are issued periodically by organizations to pseudonyms that are in good standing. If a user is not issued this resolution credential by a particular organization or coalition of organization, then this user cannot have it available to be transferred to other pseudonyms which he uses with other organizations. Therefore, the user cannot convince these other organizations that he has acted accordance with terms of service in other dealings. If this is the case, then the organization can use this lack of resolution credential to infer that the user is not in good standing in his other dealings. In one approach organizations (or other users) may issue a list of quality related credentials based upon the experience of transaction (or interaction) with the user which may act similarly to a letter of recommendation as in a resume. If such a credential is issued from multiple organizations, their values become averaged. In an alternative variation organizations may be issued credentials from users such as customers which may be used to indicate to other future users quality of service which can be expected by subsequent users on the basis of various criteria.

In our implementation, a pseudonym is a data record consisting of two fields. The first field specifies the address of the proxy server at which the pseudonym is registered. The second field contains a unique string of bits (e.g., a random binary number) that is associated with a particular user; credentials take the form of public-key digital signatures computed on this number, and the number itself is issued by a pseudonym administering server Z, as depicted in Figure 2, and detailed I n a generic form in the paper by D. Chaum and J.H. Evertse, titled "A secure and privacy-protecting protocol for transmitting personal information between organizations.". It is possible to send information to the user holding a given pseudonym, by enveloping the information in a control message that specifies the pseudonym and is addressed to the proxy server that is named in the first field of the pseudonym; the proxy server may forward the information to the user upon receipt of the control message.

While the user may use a single pseudonym for all transactions, in the more general case a user has a set of several pseudonyms, each of which represents the user in his or her interactions with a single provider or coalition of service providers. Each pseudonym in the pseudonym set is designated for transactions with a different coalition of related service providers, and the pseudonyms used with o ne provider or coalition of providers cannot be linked to the pseudonyms used with other disjoint coalitions of providers . All of the user's transactions with a given coalition can be linked by virtue of the fact that they are conducted

IPR2025-00200 / 722 Patent Page 242 of 1171 EX1002

under the same pseudonym, and therefore can be combined to define a unified picture, in the form of a user profile and a target profile interest summary, of the user's interests vis-a-vis the service or services provided by said coalition. There are other circumstances for which the use of a pseudonym may be useful and the present description is in no way intended to limit the scope of the claimed invention for example, the previously described rapid profiling tree could be used to pseudonymously acquire information about the user which is considered by the user to be sensitive such as that information which is of interest to such entities as insurance companies, medical specialists, family counselors or dating services.

Detailed Protocol

In our system, the organizations that the user U interacts with are the servers S1-Sn on the network N. However, rather than directly corresponding with each server, the user employs a proxy server, e.g. S2, as an intermediary between the local server of the user' s own client and the information provider or network vendor. Mix paths as described by D.Chaum in the paper titled "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms", Communications of the ACM, Volume 24, Number 2, February 1981 allow for untraceability and security between the client, such as C3, and the proxy server, e.g. S2. Let S(M,K) represent the digital signing of message M by modular exponentiation with key K as detailed in a paper by Rivest, R.L., Shamir, A., and Adleman, L. Titled "A method for obtaining digital signatures and public-key cryptosystems", published in the Comm. ACM 21, 2 Feb.120-126. Once a user applies to server Z for a pseudonym P and is granted a signed pseudonym signed with the private key SK_Z of server Z, the following protocol takes place to establish an entry for the user U in the proxy server S2's database D. 1. The user now sends proxy server S2 the pseudonym, which has been signed by Z to indicate the authenticity and uniqueness of the pseudonym. The user also generates a PK_P, SK_P key pair for use with the granted pseudonym, where is the private key associated with the pseudonym and PK_P is the public key associated with the pseudonym. The user forms a request to establish pseudonym P on proxy server S2, by sending the signed pseudonym $S(P, SK_Z)$ to the proxy server S2 along with a request to create a new database entry, indexed by P, and the public key PK_P. It envelopes the message and transmits it to a proxy server S2 through an anonymizing mix path, along with an anonymous return envelope header. 2. The proxy server S2 receives the database creation entry request and associated certified pseudonym message. The proxy server S2 checks to ensure that the requested pseudonym P is signed

by server Z and if so grants the request and creates a database entry for the pseudonym, as well as storing the user's public key PK_p to ensure that only the user U can make requests in the future using pseudonym P. 3. The structure of the user's database entry consists of a user profile as detailed herein, a target profile interest summary as detailed herein, and a Boolean combination of access control criteria as detailed below, along with the associated public key for the pseudonym P. 4. At any time after database entry for Pseudonym P is established, the user U may provide proxy server S2 with credentials on that pseudonym, provided by third parties, which credentials make certain assertions about that pseudonym. The proxy server may verify those credentials and make appropriate modifications to the user's profile as required by these credentials. such as recording the user's new demographic status as an adult. It may also store those credentials, so that it can present them to service providers on the user's behalf.

The above steps may be repeated, with either the same or a different proxy server, each time user U requires a new pseudonym for use with a new and disjoint coalition of providers. In practice there is an extremely small probability that a given pseudonym may have already been allocated by due to the random nature of the pseudonym generation process carried out by Z. If this highly unlikely event occurs, then the proxy server S2 may reply to the user with a signed message indicating that the generated pseudonym has already been allocated, and asking for a new pseudonym to be generated.

Pseudonymous Control of an Information Server

Once a proxy server S2 has authenticated and registered a user's pseudonym, the user may begin to use the services of the proxy server S2, in interacting with other network entities such as service providers, as exemplified by server S4 in Figure 2, an information service provider node connected to the network. The user controls the proxy server S2 by forming digitally encoded requests that the user subsequently transmits to the proxy server S2 over the network N. The nature and format of these requests will vary, since the proxy server may be used for any of the services described in this application, such as the browsing , querying, and other navigational functions described below.

In a generic scenario, the user wishes to communicate under pseudonym P with a particular information provider or user at address A, where P is a pseudonym allocated to the user and A is either a public network address at a server such as S4, or another pseudonym that is registered on a proxy server such as S4. (In the most common version of this scenario, address A is the address of an information provider, and the user is

IPR2025-00200 / 722 Patent Page 244 of 1171 EX1002

requesting that the in formation provider send target objects of interest.) The user must form a request R to proxy server S2, that requests proxy server S2 to send a message to address A and to forward the response back to the user. The user may thereby communicate with other parties, either non-pseudonymous parties, in the case where address A is a public network address, or pseudonymous parties, in the case where address A is a pseudonym held by, for example, a business or another user who prefers to operate pseudonymously.

In other scenarios, the request R to proxy server S2 formed by the user may have different content. For example, request R may instruct proxy server S2 to use the methods described later in this description to retrieve from the most convenient server a particular piece of information that has been multicast to many servers, and to send this information to the user. Conversely, request R may instruct proxy server S2 to multicast to many servers a file associated with a new target object provided by the user, as described below. If the user is a subscriber to the news clipping service described below, request R may instruct proxy server S2 to forward to the user all target objects that the news clipping service has sent to proxy server S2 for the user's attention. If the user is employing the active navigation service described below, request R may instruct proxy server S2 to select a particular cluster from the hierarchical cluster tree and provide a menu of its subclusters to the user, or to activate a query that temporarily affects proxy server S2's record of the user's target profile interest summary. If the user is a member of a virtual community as described below, request R may instruct proxy server S2 to forward to the user all messages that have been sent to the virtual community.

Regardless of the content of request R, the user, at client C3, initiates a connection to the user's local server S1, and instructs server S1 to send the request R along a secure mix path to the proxy server S2, initiating the following sequence of actions:

1. The user's client processor C3 forms a signed message $S(R, SK_P)$, which is paired with the user's pseudonym P and (if the request R requires a response) a secure one-time set of return envelopes, to form a message M. It protects the message M with an multiply enveloped route for the outgoing path. The enveloped route s provide for secure communication between S1 and the proxy server S2. The message M is enveloped in the most deeply nested message and is therefore difficult to recover should the message be intercepted by an eavesdropper.

IPR2025-00200 / 722 Patent Page 245 of 1171 EX1002

2. The message M is sent by client C3 to its local server S1, and is then routed by the data communication network N from server S1 through a set of mixes as dictated by the outgoing envelope set and arrives at the selected proxy server S2.

3. The proxy server S2 separates the received message M into the request message R, the pseudonym P, and (if included) the set of envelopes for the return path. The proxy server S2 uses pseudonym P to index and retrieve the corresponding record in proxy server S2's database, which record is stored in local storage at the proxy server S2 or on other distributed storage media accessible to proxy server S2 via the network N. This record contains a public key PK_P, user-specific information, and credentials associated with pseudonym P. The proxy server S2 uses the public key PK_P to check that the signed version $S(R, SK_P)$ of request message R is valid.

4. Provided that the signature on request message R is valid, the proxy server S2 acts on the request R. For example, in the generic scenario described above, request message R includes an embedded message M1 and an address A to whom message M1 should be sent; in this case, proxy server S2 sends message M1 to the server named in address A, such as server S4. The communication is done using signed and optionally encrypted messages over the normal point to point connections provided by the data communication network N. When necessary in order to act on embedded message M1, server S4 may exchange or be caused to exchange further signed and optionally encrypted messages with proxy server S2, still over normal point to point connections, in order to negotiate the release of user-specific information and credentials from proxy server S2. In particular, server S4 may require server S2 to supply credentials proving that the user is entitled to the information requested -- for example, proving that the user is a subscriber in good standing to a particular information service, that the user is old enough to legally receive adult material, and that the user has been offered a particular discount (by means of a special discount credential issued to the user's pseudonym).

5. If proxy server S2 has sent a message to a server S4 and server S4 has created a response M2 to message M1 to be sent to the user, then server S4 transmits the response M2 to the proxy server S2 using normal network point-to-point connections.

IPR2025-00200 / 722 Patent Page 246 of 1171 EX1002

6. The proxy server S2, upon receipt of the response M2, creates a return message Mr comprising the response M2 embedded in the return envelope set that was earlier transmitted to proxy server S2 by the user in the original message M. It transmits the return message Mr along the pseudonymous mix path specified by this return envelope set, so that the response M2 reaches the user at the user's client processor C3.

7. The response M2 may contain a request for electronic payment to the information server S4. The user may then respond by means of a message M3 transmitted by the same means as described for message M1 above, which message M3 encloses some form of anonymous payment. Alternatively, the proxy server may respond automatically with such a payment, which is debited from an account maintained by the proxy server for this user.

8. Either the response message M2 from the information server S4 to the user, or a subsequent message sent by the proxy server S2 to the user, may contain advertising material that is related to the user's request and/or is targeted to the user. Typically, if the user has just retrieved a target object X, then (a) either proxy server S2 or information server S4 determines a weighted set of advertisements that are "associated with" target object X, (b) a subset of this set is chosen randomly, where the weight of an advertisement is proportional to the probability that it is included in the subset, and (c) proxy server S2 selects from this subset just those advertisements that the user is most likely to be interested in. In the variation where proxy server S2 determines the set of advertisements associated with target object X, then this set typically consists of all advertisements that the proxy server's owner has been paid to disseminate and whose target profiles are within a threshold similarity distance of the target profile of target object X. In the variation where proxy server S4 determines the set of advertisements associated with target object X, advertisers typically purchase the right to include advertisements in this set. In either case, the weight of an advertisement is determined by the amount that an advertiser is willing to pay. Following step (c), proxy server S2 retrieves the selected advertising material and transmits it to the user's client processor C3, where it will be displayed to the user, within a specified length of time after it is received, by a trusted process running on the user's client processor C3. When proxy server S2 transmits an advertisement, it sends a message to the advertiser, indicating that

the advertisement has been transmitted to a user with a particular predicted level of interest. The message may also indicate the identity of target object X. In return, the advertiser may transmit an electronic payment to proxy server S2; proxy server S2 retains a service fee for itself, optionally forwards a service fee to information server S4, and the balance is forwarded to the user or used to credit the user's account on the proxy server.

9. If the response M2 contains or identifies a target object, the passive and/or active relevance feedback that the user provides on this object is tabulated by a process on the user's client processor C3. A summary of such relevance feedback information, digitally signed by client processor C3 with a proprietary private key SK_{C3}, is periodically transmitted through an a secure mix path to the proxy server S2, whereupon the search profile generation module 202 resident on server S2 updates the appropriate target profile interest summary associated with pseudonym P, provided that the signature on the summary message can be authenticated with the corresponding public key PK_{C3} which is available to all tabulating process that are ensured to have integrity.

When a consumer enters into a financial relationship with a particular information server based on both parties agreeing to terms for the relationship, a particular pseudonym may be extended for the consumer with respect to the given provider as detailed in the previous section. When entering into such a relationship, the consumer and the service provider agree to certain terms. However, if the user violates the terms of this relationship, the service provider may decline to provide service to the pseudonym under which it transacts with the user. In addition, the service provider has the recourse of refusing to provide resolution credentials to the pseudonym, and may choose to do so until the pseudonym bearer returns to good standing.

Pre-Fetching of Target Objects

In some circumstances, a user may request access in sequence to many files, which are stored on one or more information servers. This behavior is common when navigating a hypertext system such as the World Wide Web, or when using the target object browsing system described below.

In general, the user requests access to a particular target object or menu of target objects; once the corresponding file has been transmitted to the user's client processor, the user views its contents and makes another such request, and so on. Each request may take

IPR2025-00200 / 722 Patent Page 248 of 1171 EX1002

many seconds to satisfy, due to retrieval and transmission delays. However, to the extent that the sequence of requests is predictable, the system for customized electronic identification of desirable objects can respond more quickly to each request, by retrieving or starting to retrieve the appropriate files even before the user requests them. This early retrieval is termed "pre-fetching of files."

Pre-fetching of locally stored data has been heavily studied in memory hierarchies, including CPU caches and secondary storage (disks), for several decades. A leader in this area has been A. J. Smith of Berkeley, who identified a variety of schemes and analyzed opportunities using extensive traces in both databases and CPU caches. His conclusion was that general schemes only really paid off where there was some reasonable chance that sequential access was occurring, e.g, in a sequential read of data. As the balances between various latencies in the memory hierarchy shifted during the late 1980's and early 1990's, J. M. Smith and others identified further opportunities for pre-fetching of both locally stored data and network data. In particular, deeper analysis of patterns in work by Blaha showed the possibility of using expert systems for deep pattern analysis that could be used for pre-fetching. Work by J. M. Smith proposed the use of reference history trees to anticipate references in storage hierarchies where there was some historical data. Recent work by Touch and the Berkeley work addressed the case of data on the World-Wide Web, where the large size of images and the long latencies provide extra incentive to pre-fetch; Touch's technique is to pre-send when large bandwidths permit some speculation using HTML storage references embedded in WEB pages, and the Berkeley work uses techniques similar to J. M. Smith's reference histories specialized to the semantics of HTML data.

Successful pre-fetching depends on the ability of the system to predict the next action or actions of the user. In the context of the system for customized electronic identification of desirable objects, it is possible to cluster users into groups according to the similarity of their user profiles. Any of the well-known pre-fetching methods that collect and utilize aggregate statistics on past user behavior, in order to predict future user behavior, may then be implemented in so as to collect and utilize a separate set of statistics for each cluster of users. In this way, the system generalizes its access pattern statistics from each user to similar users, without generalizing among users who have substantially different interests. The system may further collect and utilize a similar set of statistics that describes the aggregate behavior of all users; in cases where the system cannot confidently make a prediction as to what a particular user will do, because the relevant statistics concerning that

IPR2025-00200 / 722 Patent Page 249 of 1171 EX1002

user's user cluster are derived from only a small amount of data, the system may instead make its predictions based on the aggregate statistics for all users, which are derived from a larger amount of data. For the sake of concreteness, we now describe a particular instantiation of a pre-fetching system, that both employs these insights and that makes its pre-fetching decisions through accurate measurement of the expected cost and benefit of each potential pre-fetch.

Pre-fetching exhibits a cost-benefit tradeoff. Let t denote the approximate number of minutes that pre-fetched files are retained in local storage (before they are deleted to make room for other pre-fetched files). If the system elects to pre-fetch a file corresponding to a target object X, then the user benefits from a fast response at no extra cost, provided that the user explicitly requests target object X soon thereafter. However, if the user does not request target object X within t minutes of the pre-fetch, then the pre-fetch was worthless, and its cost is an added cost that must be borne (directly or indirectly) by the user. The first scenario therefore provides benefit at no cost, while the second scenario incurs a cost at no benefit. The system tries to favor the first scenario by pre-fetching only those files that the user will access anyway. Depending on the user's wishes, the system may pre-fetch either conservatively, where it controls costs by pre-fetching only files that the user is extremely likely to request explicitly (and that are relatively cheap to retrieve), or more aggressively, where it also pre-fetches files that the user is only moderately likely to request explicitly, thereby increasing both the total cost and (to a lesser degree) the total benefit to the user.

In the system described herein, pre-fetching for a user U is accomplished by the user's proxy server S. Whenever proxy server S retrieves a user-requested file F from an information server, it uses the identity of this file F and the characteristics of the user, as described below, to identify a group of other files G1...Gk that the user is likely to access soon. The user's request for file F is said to "trigger" files G1...Gk. Proxy server S pre-fetches each of these triggered files Gi as follows:

1. Unless file Gi is already stored locally (e.g., due to previous pre-fetch), proxy server S retrieves file Gi from an appropriate information server and stores it locally.

2. Proxy server S timestamps its local copy of file Gi as having just been pre-fetched, so that file Gi will be retained in local storage for a minimum of approximately t minutes before being deleted.

IPR2025-00200 / 722 Patent Page 250 of 1171 EX1002

Whenever user U (or, in principle, any other user registered with proxy server S) requests proxy server S to retrieve a file that has been pre-fetched and not yet deleted, proxy server S can then retrieve the file from local storage rather than from another server. In a variation on steps 1-2 above, proxy server S pre-fetches a file Gi somewhat differently, so that pre-fetched files are stored on the user's client processor q rather than on server S:

1. If proxy server S has not pre-fetched file Gi in the past t minutes, it retrieves file Gi and transmits it to user U's client processor q.

2. Upon receipt of the message sent in step 1, client q stores a local copy of file Gi if one is not currently stored.

3. Proxy server S notifies client q that client q should timestamp its local copy of file Gi; this notification may be combined with the message transmitted in step 1, if any.

4. Upon receipt of the message sent in step 3, client q timestamps its local copy of file Gi as having just been pre-fetched, so that file Gi will be retained in local storage for a minimum of approximately t minutes before being deleted.

During the period that client q retains file Gi in local storage, client q can respond to any request for file Gi (by user U or, in principle, any other user of client q) immediately and without the assistance of proxy server S.

The difficult task is for proxy server S, each time it retrieves a file F in response to a request, to identify the files G1...Gk that should be triggered by the request for file F and pre-fetched immediately. Proxy server S employs a cost-benefit analysis, performing each pre-fetch whose benefit exceeds a user-determined multiple of its cost; the user may set the multiplier low for aggressive prefetching or high for conservative prefetching. These pre-fetches may be performed in parallel. The benefit of pre-fetching file Gi immediately is defined to be the expected number of seconds saved by such a pre-fetch, as compared to a situation where Gi is left to be retrieved later (either by a later pre-fetch, or by the user's request) if at all. The cost of pre-fetching file Gi immediately is defined to be the expected cost for proxy server S to retrieve file Gi, as determined for example by the network locations of server S and file Gi and by information provider charges, times 1 minus the probability that proxy server S will have to retrieve file Gi within t minutes (to satisfy either a later pre-fetch or the user's explicit request) if it is not pre-fetched now.

The above definitions of cost and benefit have some attractive properties. For example, if users tend to retrieve either file F1 or file F2 (say) after file F, and tend only in

IPR2025-00200 / 722 Patent Page 251 of 1171 EX1002

the former case to subsequently retrieve file G1, then the system will generally not pre-fetch G1 immediately after retrieving file F: for, to the extent that the user is likely to retrieve file F2, the cost of the pre-fetch is high, and to the extent that the user is likely to retrieve file F1 instead, the benefit of the pre-fetch is low, since the system can save as much or nearly as much time by waiting until the user chooses F1 and pre-fetching G1 only then.

The proxy server S may estimate the necessary costs and benefits by adhering to the following discipline:

1. Proxy server S maintains a set of disjoint clusters of the users in its user base, clustered according to their user profiles.

2. Proxy server S maintains an initially empty set PFT of "pre-fetch triples" <C,F,G>, where F and G are files, and where C identifies either a cluster of users or the set of all users in the user base of proxy server S. Each pre-fetch triple in the set PFT is associated with several stored values specific to that triple. Pre-fetch triples and their associated values are maintained according to the rules in 3 and 4.

3. Whenever a user U in the user base of proxy server S makes a request R2 for a file G, or a request R2 that triggers file G, then proxy server S takes the following actions:

a. For C being the user cluster containing user U, and then again for C being the set of all users:

b. For any request R0 for a file, say file F, made by user U during the t minutes strictly prior to the request R2:

c. If the triple <C,F,G> is not currently a member of the set PFT, it is added to the set PFT with a count of 0, a trigger-count of 0, a target-count of 0, a total benefit of 0, and a timestamp whose value is the current date and time.

d. The count of the triple <C,F,G> is increased by one.

e. If file G was not triggered or explicitly retrieved by any request that user U made strictly in between requests R0 and R2, then the target-count of the triple <C,F,G> is increased by one.

f. If request R2 was a request for file G, then the total benefit of triple <C,F,G> is increased either by the time elapsed between request R0 and request R2, or by the expected time to retrieve file G, whichever is less.

g. If request R2 was a request for file G, and G was triggered or explicitly retrieved by one or more requests that user U made strictly in between requests R0 and R2,

IPR2025-00200 / 722 Patent Page 252 of 1171 EX1002

with R1 denoting the earliest such request, then the total benefit of triple <C,F,G> is decreased either by the time elapsed between request R1 and request R2, or by the expected time to retrieve file G, whichever is less.

4. If a user U requests a file F, then the trigger-count is incremented by one for each triple currently in the set PFT such that the triple has form <C,F,G>, where user U is in the set or cluster identified by C.

5. The "age" of a triple <C,F,G> is defined to be the number of days elapsed between its timestamp and the current date and time. If the age of any triple <C,F,G> exceeds a fixed constant number of days, and also exceeds a fixed constant multiple of the triple's count, then the triple may be deleted from the set PFT.

Proxy server S can therefore decide rapidly which files G should be triggered by a request for a given file F from a given user U, as follows.

1. Let C0 be the user cluster containing user U, and C1 be the set of all users.

2. Server S constructs a list L of all triples <C0,F,G> such that <C0,F,G> appears in set PFT with a count exceeding a fixed threshold.

3. Server S adds to list L all triples <C1,F,G> such that <C0,F,G> does not appear on list L and <C1,F,G> appears in set PFT with a count exceeding another fixed threshold. 4. For each triple <C,F,G> on list L:

5. Server S computes the cost of triggering file G to be expected cost of retrieving file Gi, times 1 minus the quotient of the target-count of <C,F,G> by the trigger-count of <C,F,G>.

6. Server S computes the benefit of triggering file G to be the total benefit of <C,F,G> divided by the count of <C,F,G>.

7. Finally, proxy server S uses the computed cost and benefit, as described earlier, to decide whether file G should be triggered. The approach to pre-fetching just described has the advantage that all data storage and manipulation concerning pre-fetching decisions by proxy server S is handled locally at proxy server S. However, this "user-based" approach does lead to duplicated storage and effort across proxy servers, as well as incomplete data at each individual proxy server. That is, the information indicating what files are frequently retrieved after file F is scattered in an uncoordinated way across numerous proxy servers. An alternative, "file-based" approach is to store all such information with file F itself. The difference is as follows. In the user-based approach, a pre-fetch triple <C,F,G> in

server S's set PFT may mention any file F and any file G on the network, but is restricted to clusters C that are subsets of the user base of server S. By contrast, in the file-based approach, a pre-fetch triple <C,F,G> in server S's set PFT may mention any user cluster C and any file G on the network, but is restricted to files F that are stored on server S. (Note that in the file-based approach, user clustering is network wide, and user clusters may include users from different proxy servers.) When a proxy server S2 sends a request to server S to retrieve file F for a user U, server S2 indicates in this message the user U's user cluster C0, as well as the user U's value for the user-determined multiplier that is used in cost-benefit analysis. Server S can use this information, together with all its triples in its set PFT of the form <C0,F,G> and <C1,F,G>, where C1 is the set of all users everywhere on the network, to determine (exactly as in the user-based approach) which files G1...Gk are triggered by the request for file F. When server S sends file F back to proxy server S2, it also sends this list of files G1...Gk, so that proxy server S2 can proceed to pre-fetch files G1...Gk.

The file-based approach requires some additional data transmission. Recall that under the user-based approach, server S must execute steps 3c-3g above for any ordered pair of requests R0 and R2 made within t minutes of each other by a user who employs server S as a proxy server. Under the file-based approach, server S must execute steps 3c-3g above for any ordered pair of requests R0 and R2 made within t minutes of each other, by any user on the network, such that R0 requests a file stored on server S. Therefore, when a user makes a request R2, the user's proxy server must send a notification of request R2 to all servers S such that, during the preceding t minutes (where the variable t may now depend on server S), the user has made a request R0 for a file stored on server S. This notification need not be sent immediately, and it is generally more efficient for each proxy server to buffer up such notifications and send them periodically in groups to the appropriate servers.

Access And Reachability Control of Users and User-Specific Information

Although users' true identities are protected by the use of secure mix paths, pseudonymity does not guarantee complete privacy. In particular, advertisers can in principle employ user-specific data to barrage users with unwanted solicitations. The general solution to this problem is for proxy server S2 to act as a representative on behalf of each user in its user base, permitting access to the user and the user's private data only in

IPR2025-00200 / 722 Patent Page 254 of 1171 EX1002

accordance with criteria that have been set by the user. Proxy server S2 can restrict access in two ways:

1. The proxy server S2 may restrict access by third parties to server S2's pseudonymous database of user-specific information. When a third party such as an advertiser sends a message to server S2 requesting the release of user-specific information for a pseudonym P, server S2 re fuses to honor the request unless the message includes credentials for the accessor adequate to prove that the accessor is entitled to this information. The user associated with pseudonym P may at any time send signed control messages to proxy server S2, specifying the credentials or Boolean combinations of credentials that proxy server S2 should thenceforth consider to be adequate grounds for releasing a specified subset of the information associated with pseudonym P. Proxy server S2 stores these access criteria with its database record for pseudonym P. For example, a user might wish to proxy server S2 to release purchasing information only to selected information providers, to charitable organizations (that is, organizations that can provide a government-issued credential that is issued only to registered charities), and to market researchers who have paid user U for the right to study user U's purchasing habits.

2. The proxy server S2 may restrict the ability of third parties to send electronic messages to the user. When a third party such as an advertiser attempts to send information (such as a textual message or a request to enter into spoken or written real -time communication) to pseudonym P, by sending a message to proxy server S2 requesting proxy server S2 to forward the information to the user at pseudonym P, proxy server S2 will refuse to honor the request, unless the message includes credentials for the accessor adequate to meet the requirements the user has chosen to impose, as above, on third parties who wish to send information to the user. If the message does include adequate credentials, then proxy server S2 removes a single-use pseudonymous return address envelope from it s database record for pseudonym P, and uses the envelope to send a message containing the specified information along a secure mix path to the user of pseudonym P. If the envelope being used is the only envelope stored for pseudonym P, or more generally if the supply of such envelopes is low, proxy server S2 adds a notation to this message before sending it, which notation indicates to the user's local server that it should send additional envelopes to proxy server S2 for future use.

IPR2025-00200 / 722 Patent Page 255 of 1171 EX1002

In a more general variation, the user may instruct the proxy server S2 to impose more complex requirements on the granting of requests by third parties, not simply boolean combinations of required credentials. The user may impose any Boolean combination of simple requirements that may include, but are not limited to, the following:

(a.) the accessor (third party) is a particular party

(b.) the accessor has provided a particular credential

(c.) satisfying the request would involve disclosure to the accessor of a certain fact about the user's user profile

(d.) satisfying the request would involve disclosure to the accessor of the user's target profile interest summary

(e.) satisfying the request would involve disclosure to the accessor of statistical summary data, which data are computed from the user's user profile or target profile interest summary together with the user profiles and target profile interest summaries of at least n other users in the user base of the proxy server

(f.) the content of the request is to send the user a target object, and this target object has a particular attribute (such as high reading level, or low vulgarity, or an authenticated Parental Guidance rating from the MPAA)

(g.) the content of the request is to send the user a target object, and this target object has been digitally signed with a particular private key (such as the private key used by the National Pharmaceutical Association to certify approved documents)

(h.) the content of the request is to send the user a target object, and the target profile has been digitally signed by a profile authentication agency, guaranteeing that the target profile is a true and accurate profile of the target object it claims to describe, with all attributes authenticated.

(i.) the content of the request is to send the user a target object, and the target profile of this target object is within a specified distance of a particular search profile specified by the user

(j.) the content of the request is to send the user a target object, and the proxy server S2,.by using the user's stored target profile interest summary, estimates the user's likely interest in the target object to be above a specified threshold

(k.) the accessor indicates its willingness to make a particular payment to the user in exchange for the fulfillment of the request

IPR2025-00200 / 722 Patent Page 256 of 1171 EX1002

The steps required to create and maintain the user's access-control requirements are as follows:

1. The user composes a boolean combination of predicates that apply to requests; the resulting complex predicate should be true when applied to a request that the user wants proxy server S2 to honor, and false otherwise. The complex predicate may be encoded in another form, for efficiency.

2. The complex predicate is signed with SK_P, and transmitted from the user's client processor C3 to the proxy server S2 through the mix path enclosed in a packet that also contains the user's pseudonym P.

3. The proxy server S2 receives the packet, verifies its authenticity using PK_P and stores the access control instructions specified in the packet as part of its database record for pseudonym P.

The proxy server S2 enforces access control as follows:

1. The third party (accessor) transmits a request to proxy server S2 using the normal point-to-point connections provided by the network N . The request may be to access the target profile interest summaries associated with a set of pseudonyms P1...Pn, or to access the user profiles associated with a set of pseudonyms P1...Pn, or to forward a message to the users associated with pseudonyms P1...Pn. The accessor may explicitly specify the pseudonyms P1...Pn, or may ask that P1...Pn be chosen to be the set of all pseudonyms registered with proxy server S2 that meet specified conditions.

2. The proxy server S2 indexes the database record for each pseudonym Pi ($1 <= I <= n$), retrieves the access requirements provided by the user associated with Pi, and determines whether and how the transmitted request should be satisfied for Pi. If the requirements are satisfied, S2 proceeds with steps 3a-3c.

3a. If the request can be satisfied but only upon payment of a fee, the proxy server S2 transmits a payment request to the accessor, and waits for the accessor to send the payment to the proxy server S2. Proxy server S2 retains a service fee and forward s the balance of the payment to the user associated with pseudonym Pi, via an anonymous return packet that this user has provided.

3b. If the request can be satisfied but only upon provision of a credential, the proxy server S2 transmits a credential request to the accessor, and waits for the accessor to send the credential to the proxy server S2.

IPR2025-00200 / 722 Patent Page 257 of 1171 EX1002

3c. The proxy server S2 satisfies the request by disclosing user-specific information to the accessor, by providing the accessor with a set of single-use envelopes to communicate directly with the user, or by forwarding a message to the user, as requested. 4. Proxy server S2 optionally sends a message to the accessor, indicating why each of the denied requests for P1...Pn was denied, and/or indicating how many requests were satisfied.

5. The active and/or passive relevance feedback provided by any user U with respect to any target object sent by any path from the accessor is tabulated by the above-described tabulating process resident on user U's client processor C3. As described above, a summary of such information is periodically transmitted to the proxy server S2 to enable the proxy server S2 to update that user's target profile interest summary and user profile.

The access control criteria can be applied to solicited as well as unsolicited transmissions. That is, the proxy server can be used to protect the user from inappropriate or misrepresented target objects that the user may request. If the user requests a target object from an information server, but the target object turns out not to meet the access control criteria, then the proxy server will not permit the information server to transmit the target object to the user, or to charge the user for such transmission. For example, to guard against target objects whose profiles have been tampered with, the user may specify an access control criterion that requires the provider to prove the target profile's accuracy by means of a digital signature from a profile authentication agency. As another example, the parents of a child user may instruct the proxy server that only target objects that have been digitally signed by a recognized child protection organization may be transmitted to the user; thus, the proxy server will not let the user retrieve pornography, even from a rogue information server that is willing to provide pornography to users who have not supplied an adulthood credential.

Distribution of Information with Multicast Trees

The graphical representation of the network N presented in Figure 3 shows that at least one of the data communications links can be eliminated, as shown in Figure 4, while still enabling the network N to transmit messages among all the servers A-D. By elimination, we mean that the link is unused in the logical design of the network, rather than a physical disconnection of the link. The graphs that result when all redundant data communications links are eliminated are termed "trees" or "connected acyclic graphs." A

IPR2025-00200 / 722 Patent Page 258 of 1171 EX1002

graph where a message could be transmitted by a server through other servers and then return to the transmitting server over a different originating data communications link is termed a "cycle." A tree is thus an acyclic graph whose edges (links) connect a set of graph "nodes" (servers). The tree can be used to efficiently broadcast any data file to selected servers in a set of interconnected servers.

The tree structure is attractive in a communications network because much information distribution is multicast in nature -- that is, a piece of information available at a single source must be distributed to a multiplicity of points where the information can be accessed. This technique is widely known: for example, "FAX trees" are in common use in political organizations, and multicast trees are widely used in distribution of multimedia data in the Internet; for example, see "Scalable Feedback Control for Multicast Video Distribution in the Internet," (Jean - Chrysostome Bolot, Thierry Turletti, & Ian Wakeman, Computer Communication Review, Vol. 24, # 4, Oct. '94, Proceedings of SIGCOMM'94 , pp. 58 - 67) or "An Architecture For Wide-Area Multicast Routing," (Stephen Deering, Deborah Estrin, Dino Farinacci, Van Jacobson, Ching-Gung Liu, & Liming Wei, Computer Communication Review, Vol. 24, # 4, Oct. '94, Proceedings of SIGCOMM'94 , pp. 126 - 135). While there are many possible trees that can be overlaid on a graph representation of a network, both the nature of the networks (e.g., the cost of transmitting data over a link) and their use (for example, certain nodes may exhibit more frequent intercommunication) can make one choice of tree better than another for use as a multicast tree. One of the most difficult problems in practical network design is the construction of "good" multicast trees, that is, tree choices which exhibit low cost (due to data not traversing links unnecessarily) and good performance (due to data frequently being close to where it is needed)

Constructing a Multicast Tree

Algorithms for constructing multicast trees have either been ad-hoc, as is the case of the Deering, et al. Internet multicast tree, which adds clients as they request service by grafting them into the existing tree, or by construction of a minimum cost spanning tree. A distributed algorithm for creating a spanning tree (defined as a tree that connects, or "spans," all nodes of the graph) on a set of Ethernet bridges was developed by Radia Perlman ("Interconnections: Bridges and Routers," Radia Perlman, Addison-Wesley, 1992). Creating a minimal-cost spanning tree for a graph depends on having a cost model for the arcs of the graph (corresponding to communications l inks in the communications network). In the case of Ethernet bridges, the default cost (more complicated costing models for path costs are

IPR2025-00200 / 722 Patent Page 259 of 1171 EX1002

discussed on pp. 72-73 of Perlman) is calculated as a simple distance measure to the root; thus the spanning tree minimizes the cost to the root by first electing a unique root and then constructing a spanning tree based on the distances from the root. In this algorithm, the root is elected by recourse to a numeric ID contained in "configuration messages": the server w hose ID has minimum numeric value is chosen as the root. Several problems exist with this algorithm in general. First, the method of using an ID does not necessarily select the best root for the nodes interconnected in the tree. Second, the cost model is simplistic.

We first show how to use the similarity-based methods described above to select the servers most interested in a group of target objects, herein termed "core servers" for that group. Next we show how to construct an unrooted multicast tree that can be used to broadcast files to these core servers. Finally, we show how files corresponding to target objects are actually broadcast through the multicast tree at the initiative of a client, and how these files are later retrieved from the core servers when clients request them.

Since the choice of core servers to distribute a file to depends on the set of users who are likely to retrieve the file (that is, the set of users who are likely to be interested in the corresponding target object), a separate set of core servers and hence a separate multicast tree may be used for each topical group of target objects. Throughout the description below, servers may communicate among themselves through any path over which messages can travel; the goal of each multicast tree is to optimize the multicast distribution of files corresponding to target objects of the corresponding topic. Note that this problem is completely distinct from selecting a multiplicity of spanning trees for the complete set of interconnected nodes as disclosed by Sincoskie in U.S. Patent No. 4,706,080 and the publication titled "Extended Bridge Algorithms for Large Networks" by W. D. Sincoskie and C. J. Cotton, published January 1988 in IEEE Network on pages 16-24. The trees in this disclosure are intentionally designed to interconnect a selected *subset* of nodes in the system, and are successful to the degree that this *subset* is relatively small.

Multicast Tree Construction Procedure

A set of topical multicast trees for a set of homogenous target objects may be constructed o r reconstructed at any time, as follows. The set of target objects is grouped into a fixed number of topical clusters C1...Cp with the methods described above, for example, by choosing C1...Cp to be the result of a k-means clustering of the set of target objects, or alternatively a covering set of low-level clusters from a hierarchical cluster tree

of these target objects. A multicast tree MT(c) is then constructed from each cluster C in C1...Cp, by the following procedure:

1. Given a set of proxy servers, S1...Sn, and a topical cluster C. It is assumed that a general multicast tree MT_{full} that contains all the proxy servers S1...Sn has previously been constructed by well-known methods.

2. Each pair <Si, C> is associated with a weight, w(Si, C), which is intended to covary with the expected number of users in the user base of proxy server Si who will subsequently access a target object from cluster C. This weight is computed by proxy server Si in any of several ways, all of which make use of the similarity measurement computation described herein.

One variation makes use of the following steps: (a) Proxy server Si randomly selects a target object T from cluster C. (b) For each pseudonym in its local database, with associated user U, proxy server Si applies the techniques disclosed above to user U's stored user profile and target profile interest summary in order to estimate the interest w(U, T) that user U has in t he selected target object T. The aggregate interest w(Si, T) that the user base of proxy server Si has in the target object T is defined to be the sum of these interest values w(U, T). Alternatively, w(Si, T) may be defined to be the sum of values s(w(U, T)) over all U in the user base. Here s(*) is a sigmoidal function that is close to 0 for small arguments and close to a constant p_{max} for large arguments; thus s(w(U, T)) estimates the probability that user U will access target object T, which probability is assumed to be independent of the probability that any other user will access target object T. In a variation, w(Si, T) is made to estimate the probability that at least one user from the user base of Si will access target object T: then w(Si, T) may be defined as the maximum of values w(U, T), or of 1 minus the product over the users U of the quantity (1 - s(w(U, T))). (c)Proxy server Si repeats steps (a)-(b) for several target objects T selected randomly from cluster C, and averages the several values of w(Si, T) thereby computed in step (b) to determine the desired quantity w(Si, C), which quantity represents the expected aggregate interest by the user base of proxy server Si in the target objects of cluster C.

In another variation, where target profile interest summaries are embodied as search profile sets, the following procedure is followed to compute w(Si, C): (a). For each search profile P_S in the locally stored search profile set of any user in the user base of proxy server Si, proxy server Si computes the distance $d(P_S , P_C)$ between the search profile and the cluster profile P_C of cluster C. (b). w(Si,C) is chosen to be the maximum value of (-d(P_S

IPR2025-00200 / 722 Patent Page 261 of 1171 EX1002

,P_C)/r) across all such search profiles P_S, where r is computed as an affine function of the cluster diameter of cluster C. The slope and/or intercept of this affine function are chosen to be smaller (thereby increasing w(Si, C)) for servers Si for which the target object provider wishes to improve performance, as may be the case if the users in the user base of proxy server Si pay a premium for improved performance, or if performance at Si will otherwise be unacceptably low due to slow network connections.

In another variation, the proxy server Si is modified so that it maintains not only target profile interest summaries for each user in its user base, but also a single aggregate target profile interest summary for the entire user base. This aggregate target profile interest summary is determined in the usual way from relevance feedback, but the relevance feedback on a target object, in this case, is considered to be the frequency with which users in the user base retrieved the target object when it was new. Whenever a user retrieves a target object by means of a request to proxy server Si, the aggregate target profile interest summary for proxy server Si is updated. In this variation, w(Si, C) I s estimated by the following steps:

(a) Proxy server Si randomly selects a target object T from cluster C.

(b) Proxy server Si applies the techniques disclosed above to its stored aggregate target profile interest summary in order to estimate the aggregate interest w(Si, T) that its aggregated user base had in the selected target object T, when new; this may be interpreted as an estimate of the likelihood that at least one member of the user base will retrieve a new target object similar to T.

(c)Proxy server Si repeats steps (a)-(b) for several target objects T selected randomly from cluster C, and averages the several values of w(Si, T) thereby computed in step (b) to determine the desired quantity w(Si, C), which quantity represents the expected aggregate interest by the user base of proxy server Si in the target objects of cluster C..

3. Those servers Si from among S1...Sn with the greatest weights w(Si, C) are designated "core servers" for cluster C. In one variation, where it is desired to select a fixed number of core servers, those servers Si with the greatest values of w(Si, C) are selected. In another variation, the value of w(Si, C) for each server Si is compared against a fixed threshold w_{min}, and those servers Si such that w(Si, C) equals or exceeds w_{min} are selected as core servers. If cluster C represents a narrow and specialized set of target objects, as often happens when the clusters C1...Cp are numerous, it is usually adequate to select only

a small number of core server cluster C, thereby obtaining substantial advantages in computational efficiency in steps 4-5 below

.4. A complete graph G(C) is constructed whose vertices are the designated core servers for cluster C. For each pair of core servers, the cost of transmitting a message between those core servers along the cheapest path is estimated, and the weight of the edge connecting those core servers is taken to be this cost. The cost is determined as a suitable function of average transmission charges, average transmission delay, and worst-case or near-worst-case transmission delay.

5. The multicast tree MT(C) is computed by standard methods to be the minimum spanning tree (or a near-minimum spanning tree) for G(C), where the weight of an edge between two core servers is taken to b e the cost of transmitting a message between those two core servers. Note that MT(C) does not contain as vertices all proxy servers S1...Sn, but only the core servers for cluster C.

6. A message M is formed describing the cluster profile for cluster C, the core servers for cluster C and the topology of the multicast tree MT(C) constructed on those core servers. Message M is broadcast to all proxy servers S1...Sn by means of the general multicast tree MT_{full}. Each proxy server Si, upon receipt of message M, extracts the cluster profile of cluster C, and stores it on a local storage device, together with certain other information that it determines from message M, as follows. If proxy server Si is named in message M as a core server for cluster C, then proxy server Si extracts and stores the subtree of MT(C) induced by all core servers whose path distance from Si in the graph MT(C) is less than or equal to d, where d is a constant positive integer (usually from 1 to 3). If message M does not name proxy server Si as a core server for MT(C), then proxy server Si extracts and stores a list of one or more nearby core servers that can be inexpensively contacted by proxy server Si over virtual point-to-point links.

In the network of Figure 3, to illustrate the use of trees, as applied to the system of the present invention, consider the following simple example where it is assumed that client r provides on-line information for the network, such as an electronic newspaper. This information can be structured by client r into a prearranged form, comprising a number of files, each of which is associated with a different target object. In the case of an electronic newspaper, the files can contain textual representations of stock prices, weather forecasts, editorials, etc. The system determines likely demand for the target objects associated with these files in order to optimize the distribution of the files through the network N of

interconnected clients p-s and proxy servers A-D. Assume that cluster C consists of text articles relating to the aerospace industry; further assume that the target profile interest summaries stored at proxy servers A and B for the users at clients p and r indicate that these users are strongly interested in such articles. Then the proxy servers A and B are selected as core servers for the multicast tree MT(C). The multicast tree MT(C) is then computed to consist of the core servers, A and B, connected by an edge that represents the least costly virtual point-to-point link between A and B (either the direct path A-B or the indirect path A-C-B, depending on the cost).

Global Requests to Multicast Trees

One type of message that may be transmitted to any proxy server S is termed a "global request message." Such a message M triggers the broadcast of an embedded request R to all core servers in a multicast tree MT(C). The content of request R and the identity of cluster C are included in the message M, as is a field indicating that message M is a global request message. In addition, the message M contains a field S_{last} which is unspecified except under certain circumstances described below, when it names a specific core server. A global request message M may be transmitted to proxy server S by a user registered with proxy server S, which transmission may take place along a pseudonymous mix path, or it may be transmitted to proxy server S from another proxy server, along a virtual point-to-point connection.

When a proxy server S receives a message M that is marked as a global request message, it acts as follows: 1. If proxy server S is not a core server for topic C, it retrieves its locally stored list of nearby core servers for topic C, selects from this list a nearby core server S', and transmits a copy of message M over a virtual point-to-point connection to cor e server S'. If this transmission fails, proxy server S repeats the procedure with other core servers on its list. 2. If proxy server S is a core server for topic C, it executes the following steps: (a) Act on the request R that is embedded in message M. (b) Set S_{curr} to be S(C) Retrieve the locally stored subtree of MT(C), and extract from it a list L of all core servers that are directly linked to S_{curr} in this subtree. (d) If the message M specifies a value for S_{last} and S_{last} appears on the list L, remove S_{last} from the list L. Note that list L may be empty before this step, or may become empty as a result of this step. (e) For each server Si in list L, transmit a copy of message M from server S to server Si over a virtual point-to-point connection, where the S_{last} field of the copy of message M has been altered to S_{curr}. If Si cannot be reached in a reasonable amount of time by any virtual point-to-point connection

IPR2025-00200 / 722 Patent Page 264 of 1171 EX1002

(for example, server Si is broken), recurse to step (c) above with S_{orig} bound to S_{curr} and S_{curr} bound to S{\sub I} for the duration of the recursion.

When server S' in step 1 or a server Si in step 2(e) receives a copy of the global request message M, it acts according to exactly the same steps. As a result, all core servers eventually receive a copy of global request message M and act on the embedded request R, unless some core servers cannot be reached. Even if a core server is unreachable, step (e) ensures that the broadcast can continue to other core servers in most circumstances, provided that d > 1; higher values of d provide additional insurance against unreachable core servers.

Multicasting Files

The system for customized electronic information of desirable objects executes the following steps in order to introduce a new target object into the system. These steps are initiated by an entity E, which may be either a user entering commands via a keyboard at a client processor q, as illustrated in Figure 3, or an automatic software process resident on a client or server processor q. 1. Processor q forms a signed request R, which asks the receiver to store a copy of a file F on its local storage device. File F, which is maintained by client q on storage at client q or on storage accessible by client q over the network, contains the informational content of or an identifying description of a target object, as described above. The request R also includes an address at which entity E may be contacted (possibly a pseudonymous address at some proxy server D), and asks the receiver t o store the fact that file F is maintained by an entity at said address. 2. Processor q embeds request R in a message M1, which it pseudonymously transmits to the entity E's proxy server D as described above. Message M1 instructs proxy server D to broadcast request R along an appropriate multicast tree. 3. Upon receipt of message M1, proxy server D examines the doubly embedded file F and computes a target profile P for the corresponding target object. It compares the target profile P to each of the cluster profiles for topical clusters C1...Cp described above, and chooses Ck to be the cluster with the smallest similarity distance to profile P. 4. Proxy server D sends itself a global request message M instructing itself to broadcast request R along the topical multicast tree MT(Ck). 5. Proxy server D notifies entity E through a pseudonymous communication that file F has been multicast along the topical multicast tree for cluster Ck.

As a result of the procedure that server D and other servers follow for acting on global request messages, step 4 eventually causes all core servers for topic Ck to act on

request R and therefore store a local copy of file F. In order to make room for file F on its local storage device, a core server Si may have to delete a less useful file. There are several ways to choose a file to delete. One option, well known in the art, is for Si to choose to delete the least recently accessed file. In another variation, Si deletes a file that it believes few users will access. In this variation, whenever a server Si stores a copy of a file F, it also computes and stores the weight $w(Si, C_F)$, where C_F is a cluster consisting of the single target object associated with file F. Then, when server Si needs to delete a file, it chooses to delete the file F with the lowest weight $w(Si, C_F)$. To reflect the fact that files are accessed less as they age, server Si periodically multiplies its stored value of $w(Si, C_F)$ by a decay factor, such as 0.95, for each file F that it then stores. Alter natively, instead of using a decay factor, server Si may periodically recompute aggregate interest $w(Si, C_F)$ for each file F that it stores; the aggregate interest changes over time because target objects typically have an age attribute that the system considers in estimating user interest, as described above.

If entity E later wishes to remove file F from the network, for example because it has just multicast an updated version, it pseudonymously transmits a digitally signed global request message to proxy server D, requesting all proxy servers in the multicast tree MT(Ck) to delete any local copy of file F that they may be storing.

Queries to Multicast Trees

In addition to global request messages, another type of message that may be transmitted to any proxy server S is termed a "query message." When transmitted to a proxy server, a query message causes a reply to be sent to the originator of the message; this reply will contain an answer to a given query Q if any of the servers in a given multicast tree MT(C) are able to answer it, and will otherwise indicate that no answer is available. The query and the cluster C are named in the query message. In addition, the query message contains a field S_{last} which is unspecified except under certain circumstances described below, when it names a specific core server. When a proxy server S receives a message M that is marked as a query message, it acts as follows:1. Proxy server S sets A_r to be the return address for the client or server that transmitted message M to server S. A_r may be either a network address or a pseudonymous address 2. If proxy server S is not a core server for cluster C, it retrieves its locally stored list of nearby core servers for topic C, selects from this list a nearby core server S', and transmits a copy of the locate message M over a virtual point-to-point connection to core server S'. If this transmission fails, proxy

IPR2025-00200 / 722 Patent Page 266 of 1171 EX1002

server S repeats the procedure with other core servers on its list. Upon receiving a reply, it forwards this reply to address A_r. 3. If proxy server S is a core server for cluster C, and it is able to answer query Q using locally stored information, then it transmits a "positive" reply to A_r containing the answer. 4. If proxy server S is a core server for topic C, but it is unable to answer query Q using locally stored information, then it carries out a parallel depth-first search by executing the following steps: (a) Set L to be the empty list. (b) Retrieve the locally stored subtree of MT(C). For each server Si directly linked to S_{curr} in this subtree, other than S_{last} (if specified), add the ordered pair (Si, S) to the list L. (c) If L is empty, transmit a "negative" reply to address A_r saying that server S cannot locate an answer to query Q, and terminate the execution of step 4; otherwise proceed to step (d). (d) Select a list L1 of one or more server pairs (Ai, Bi) from the list L. For each server pair (Ai, Bi) on the list L1, form a locate message M(Ai, Bi), which is a copy of message M whose S_{last} field has been modified to specify Bi, and transmit this message M(Ai, Bi) to server Ai over a virtual point-to-point connection. (e) For each reply received (by S) to a message sent in step (d), act as follows: (I) If a "positive" reply arrives to a locate message M(Ai, Bi), then forward this reply to A_r and terminate step 4, immediately. (ii) If a "negative" reply arrives to a locate message M(Ai, Bi), then remove the pair (Ai, Bi) from the list L1. (iii) If the message M(Ai, Bi) could not be successfully delivered to Ai, then remove the pair (Ai, Bi) from the list L1, and add the pair (Ci, Ai) to the list L1 for each Ci other than Bi that is directly linked to Ai in the locally stored subtree of MT(C). (f) Once L1 no longer contains any pair (Ai, Bi) for which a message M(Ai, Bi) has been sent, or after a fixed period of time has elapsed, return to step (c).

Retrieving Files from a Multicast Tree

When a processor q in the network wishes to retrieve the file associated with a given target object, it executes the following steps. These steps are initiated by an entity E, which may be either a user entering commands via a keyboard at a client q, as illustrated in Figure 3, or an automatic software process resident on a client or server processor q. 1. Processor q forms a query Q that asks whether the recipient (a core server for cluster C) still stores a file F that was previously multicast to the multicast tree MT(C); if so, the recipient server should reply with its own server name . Note that processor q must already know the name of file F and the identity of cluster C; typically, this information is provided to entity E by a service such as the news clipping service or browsing system described below, which must identify files to the user by (name, multicast topic) pair. 2. Processor q forms a query

IPR2025-00200 / 722 Patent Page 267 of 1171 EX1002

message M that poses query Q to the multicast tree MT(C). 3. Processor q pseudonymously transmits message M to the user's proxy server D, as described above. 4. Processor q receives a response M2 to message M. 5. If the response M2 is "positive," that is, it names a server S that still stores file F, then processor q pseudonymously instructs the user's proxy server D to retrieve file F from server S. If the retrieval fails because server S has deleted file F since it answered the query, then client q returns to step 1. 6. If the response M2 is "negative," that is, it indicates that no server in MT(C) still stores file F, then processor q forms a query Q that asks the recipient for the address A of the entity that maintains file F; this entity will ordinarily maintain a copy of file F indefinitely. All core servers in MT(C) ordinarily retain this information (unless instructed to delete it by the maintaining entity), even if they delete file F for space reasons. Therefore, processor q should receive a response providing address A, whereupon processor q pseudonymously instructs the user's proxy server D to retrieve file F from address A.

When multiple versions of a file F exist on local servers throughout the data communication network N, but are not marked as alternate versions of the same file, the system's ability to rapidly locate files similar to F (by treating them as target objects and applying the methods disclosed in "Searching for Target Objects" above) makes it possible to find all the alternate versions, even if they are stored remotely. These related data files may then be reconciled by any method. In a simple instantiation, all versions of the data file would be replaced with the version that had the latest date or version number. In another instantiation, each version would be automatically annotated with references or pointers to the other versions.

NEWS CLIPPING SERVICE

The system for customized electronic identification of desirable objects of the present invention can be used in the electronic media system of Figure 1 to implement an automatic news clipping service which learns to select (filter) news articles to match a user's interests, based solely on which articles the user chooses to read. The system for customized electronic identification of desirable objects generates a target profile for each article that enters the electronic media system, based on the relative frequency of occurrence of the words contained in the article. The system for customized electronic identification of desirable objects also generates a search profile set for each user, as a function of the target profiles of the articles the user has accessed and the relevance feedback the user has provided on these articles. As new articles are received for storage on the mass storage

IPR2025-00200 / 722 Patent Page 268 of 1171 EX1002

systems SS_1 -SS_m of the information servers I_1 -I_m , the system for customized electronic identification of desirable objects generates their target profiles. The generated target profiles are later compared to the search profiles in the users' search profile sets, and those new articles whose tar get profiles are closest (most similar) to the closest search profile in a user's search profile set are identified to that user for possible reading. The computer program providing the articles to the user monitors how much the user reads (the number of screens of data and the number of minutes spent reading), and adjusts the search profiles in the user's search profile set to more closely match what the user apparently prefers to read. T he details of the method used by this system are disclosed in flow diagram form in Figure 5. This method requires selecting a specific method of calculating user-specific search profile sets, of measuring similarity between two profiles, and of updating a user's search profile set (or more generally target profile interest summary) based on what the user read, and the examples disclosed herein are examples of the many possible implementations that can be used and should not be construed to limit the scope of the system.

Initialize Users' Search Profile Sets

The news clipping service instantiates target profile interest summaries as search profile sets, so that a set of high-interest search profiles is stored for each user. The search profiles associate d with a given user change over time. As in any application involving search profiles, they can be initially determined for a new user (or explicitly altered by an existing user) by any of a number of procedures, including the following preferred methods : (1) asking the user to specify search profiles directly by giving keywords and/or numeric attributes, (2) using copies of the profiles of target objects or target clusters that the user indicates are representative of his or her interest, (3) using a standard set of search profiles copied or otherwise determined from the search profile sets of people who are demographically similar to the user.

Retrieve New Articles from Article Source

Articles are available on-line from a wide variety of sources. In the preferred embodiment, one would use the current days news as supplied by a news source, such as the AP or Reuters news wire. These news articles are input to the electronic media system by being loaded into the mass storage system SS_4 of an information server S_4. The article profile module 201 of the system for customized electronic identification of desirable objects can reside on the information server S_4 and operates pursuant to the steps illustrated in the flow diagram of Figure 5, where, as each article is received at step 501 by the

IPR2025-00200 / 722 Patent Page 269 of 1171 EX1002

information server S_4, the article profile module 201 at step 502 generates a target profile for the article and stores the target profile in an article indexing memory (typically part of mass storage system SS_4 for later use in selectively delivering articles to users. This method is equally useful for selecting which articles to read from electronic news groups and electronic bulletin boards, and can be used as part of a system for screening and organizing electronic mail ("e-mail").

Calculate Article Profiles

A target profile is computed for each new article, as described earlier. The most important attribute of the target profile is a textual attribute that stands for the entire text of the article. This textual attribute is represented as described earlier, as a vector of numbers, which numbers in the preferred embodiment include the relative frequencies (TF/IDF scores) of word occurrences in this article relative to other comparable articles. The server must count the frequency of occurrence of each word in the article in order to compute the TF/IDF scores.

These news articles are then hierarchically clustered in a hierarchical cluster tree at step 503, which serves as a decision tree for determining which news articles are closest to the user's interest. The resulting clusters can be viewed as a tree in which the top of the tree includes all target objects and branches further down the tree represent divisions of the set of target objects into successively smaller subclusters of target objects. Each cluster has a cluster profile, so that at each node of the tree, the average target profile (centroid) of all target objects stored in the subtree rooted at that node is stored. This average of target profiles is computed over the representation of target profiles as vectors of numeric attributes, as described above.

Compare Current Articles' Target Profiles to a User's Search Profiles

The process by which a user employs this apparatus to retrieve news articles of interest is illustrated in flow diagram form in Figure 11. At step 1101, the user logs into the data communication network N via their client processor C_1 and activates the news reading program. This is accomplished by the user establishing a pseudonymous data communications connection as described above to a proxy server S_2, which provides front-end access to the data communication network N. The proxy server S_2 maintains a list of authorized pseudonyms and their corresponding public keys and provides access and billing control. The user has a search profile set stored in the local data storage medium on the proxy server S_2. When the user requests access to "news" at step 1102, the profile

matching module 203 resident on proxy server S_2 sequentially considers each search profile p_k from the user's search profile set to determine which news articles are most likely of interest to the user. The news articles were automatically clustered into a hierarchical cluster tree at an earlier step so that the determination can be made rapidly for each user. The hierarchical cluster tree serves as a decision tree for determining which articles' target profiles are most similar to search profile p_k: the search for relevant articles begins at the top of the tree, and at each level of the tree the branch or branches are selected which have cluster profiles closest to p_k. This process is recursively executed until the leaves of the tree are reached, identifying individual articles of interest to the user, as described in the section "Searching for Target Objects" above.

A variation on this process exploits the fact that many users have similar interests. Rather than carry out steps 5-9 of the above process separately for each search profile of each user, it is possible to achieve added efficiency by carrying out these steps only once for each group of similar search profiles, thereby satisfying many users' needs at once. In this variation, the system begins by non-hierarchically clustering all the search profiles in the search profile sets of a large number of users. For each cluster k of search profiles, with cluster profile p_k, it uses the method described in the section "Searching for Target Objects" to locate articles with target profiles similar to p_k. Each located article is then identified as of interest to each user who has a search profile represented in cluster k of search profiles.

Notice that the above variation attempts to match clusters of search profiles with similar clusters of articles. Since this is a symmetrical problem, it may instead be given a symmetrical solution, as the following more general variation shows. At some point before the matching process commences, all the news articles to be considered are clustered into a hierarchical tree, termed the "target profile cluster tree," and the search profiles of all users to be considered are clustered into a second hierarchical tree, termed the "search profile cluster tree." The following steps serve to find all matches between individual target profiles from any target profile cluster tree and individual search profiles from any search profile cluster tree: 1. For each child subtree S of the root of the search profile cluster tree (or, let S be the entire search profile cluster tree if it contains only one search profile): 2. Compute the cluster profile P_S to be the average of all search profiles in subtree S 3. For each subcluster (child subtree) T of the root of the target profile cluster tree (or, let T be the entire target profile cluster tree if it contains only one target profile): 4. Compute the cluster profile P_T to be the average of all target profiles in subtree T 5. Calculate $d(P_S, P_T,$

IPR2025-00200 / 722 Patent Page 271 of 1171 EX1002

the distance between P_S and P_T 6. If $d(P_S, P_T) < t$, a threshold, 7. If S contains only one search profile and T contains only one target profile, decl are a match between that search profile and that target profile,8. otherwise recurse to step 1 to find all matches between search profiles in tree S and target profiles in tree T.

The threshold used in step 6 is typically an affine function or other function of the greater of the cluster variances (or cluster diameters) of S and T. Whenever a match is declared between a search profile and a target profile, the target object that contributed the target profile is identified as being of interest to the user who contributed the search profile. Notice that the process can be applied even when the set of users to be considered or the set of target objects to be considered is very small. In the case of a single user, the process reduces to the method given for identifying articles of interest to a single user. In the case of a single target object, the process constitutes a method for identifying users to whom that target object is of interest.

Present List of Articles to User

Once the profile correlation step is completed for a selected user or group of users, at step 1104 the profile processing module 203 stores a list of the identified articles for presentation to each user. At a user's request, the profile processing system 203 retrieves the generated list of relevant articles and presents this list of titles of the selected articles to the user, who can then select at step 1105 any article for viewing. (If no titles are available, then the first sentence(s) of each article can be used.) The list of article titles is sorted according to the degree of similarity of the article's target profile to the most similar search profile in the user's search profile set. The resulting sorted list is either transmitted in real time to the user client processor C_1, if the user is present at their client processor C_1, or can be transmitted to a user's mailbox, resident on the user's client processor C_1 or stored within the server S_2 for later retrieval by the user; other methods of transmission include facsimile transmission of the printed list or telephone transmission by means of a text-to-speech system. The user can then transmit a request by computer, facsimile, or telephone to indicate which of the identified articles the user wishes to review, if any. The user can still access all articles in any information server S_4 to which the user has authorized access, however, those lower on the generated list are simply further from the user's interests, as determined by the user's search profile set. The server S_2 retrieves the article from the local data storage medium or from an information server S_4 and presents the article one screen at

IPR2025-00200 / 722 Patent Page 272 of 1171 EX1002

a time to the user's client processor C_1. The user can at any time select another article for reading or exit the process.

Monitor Which Articles Are Read

The user's search profile set generator 202 at step 1107 monitors which articles the user reads, keeping track of how many pages of text are viewed by the user, how much time is spent viewing the article, and whether all pages of the article were viewed. This information can be combined to measure the depth of the user's interest in the article, yielding a passive relevance feedback score, as described earlier. Although the exact details depend on the length and nature of the articles being searched, a typical formula might be: measure of article attractiveness =0.2 if the second page is accessed + 0.2 if all pages are accessed + 0.2 if more than 30 seconds was spent on the article + 0.2 if more than one minute was spent on the article + 0.2 if the minutes spent in the article are greater than half the number of pages.

The computed measure of article attractiveness can then be used as a weighting function to adjust the user's search profile set to thereby more accurately reflect the user's dynamically changing interests.

Update User Profiles

Updating of a user's generated search profile set can be done at step 1108 using the method described in copending U.S. Patent Application Serial No. 08/346,425. When an article is read, the server S_2 shifts each search profile in the set slightly in the direction of the target profiles of those nearby articles for which the computed measure of article attractiveness was high. Given a search profile with attributes u_{ik} from a user's search profile set, and a set of J articles available with attributes d_{jk} (assumed correct for now), where I indexes users, j indexes articles, and k indexes attributes, user I would be predicted to pick a set of P distinct articles to minimize the sum of $d(u_I, b_j)$ over the chosen articles j. The user's desired attributes u_{ik} and an article's attributes d_{jk} would be some form of word frequencies such as TF/IDF and potentially other attributes such as the source, reading level, and length of the article, while $d(u_I, d_j)$ is the distance between these two attribute vectors (profiles) using the similarity measure described above. If the user picks a different set of P articles than was predicted, the user search profile set generation module should try to adjust u and/ord to more accurately predict the articles the user selected. In particular, u_I and/or d_j should be shifted to increase their similarity if user I was predicted not to select article j but did select it, and perhaps also to decrease their similarity if user I was predicted

IPR2025-00200 / 722 Patent Page 273 of 1171 EX1002

to select article j but did not. A preferred method is to shift u for each wrong prediction that user I will not select article j, using the formula: $u_{ik}' = u_{ik} - e(u_{ik} d_{jk})$

Here u_I is chosen to be the search profile from user I's search profile set that is closest to target profile. If e is positive, this adjustment increases the match between user I's search profile set and the target profiles of the articles user I actually selects, by making u_I closer to d_j for the case where the algorithm failed to predict an article that the viewer selected. The size of e determines how many example articles one must see to change the search profile substantially. If e is too large, the algorithm becomes unstable, but for sufficiently small e, it drives u to its correct value. In general, e should be proportional to the measure of article attractiveness; for example, it should be relatively high if user I spends a long time reading article j. One could in theory also use the above formula to decrease the match in the case where the algorithm predicted an article that the user did not read, by making e negative in that case. However, there is no guarantee that u will move in the correct direction in that case. One can also shift the attribute weights w_I of user I by using a similar algorithm: $w_{ik}' = (w_{ik} - e|u_{ik} - d_{jk}|) / \Sigma_k (w_{ik} - e| u_{ik} - d_{jk}|)$ This is particularly important if one is combining word frequencies with other attributes. As before, this increases the match if e is positive -- for the case where the algorithm failed to predict an article that the user read, this time by decreasing t he weights on those characteristics for which the user's target profile u_I differs from the article's profile d_j Again, the size of e determines how many example articles one must see to replace what was originally believed. Unlike the procedure for adjusting u, one also make use of the fact that the above algorithm decreases the match if e is negative -- for the case where the algorithm predicted an article that the user did not read. The denominator of the expression prevents weights from shrinking to zero over time by renormalizing the modified weights w_I' so that they sum to one. Both u and w can be adjusted for each article accessed. When e is small, as it should be, there is no conflict between the two parts of the algorithm. The selected user's search profile set is updated at step 1108.

Further Applications of the Filtering Technology

The news clipping service may deliver news articles (or advertisements and coupons for purchasables) to off-line users as well as to users who are on-line. Although the off-line users may have no way of providing relevance feedback, the user profile of an off-line user U may be similar to the profiles of on-line users, for example because user U is demographically similar to these other users, and the level of user U's interest in

IPR2025-00200 / 722 Patent Page 274 of 1171 EX1002

particular target objects can therefore be estimated via the general interest-estimation methods described earlier. In one application, the news clipping service chooses a set of news articles (respectively, advertisements and coupons) that are predicted to be of interest to user U, thereby determining the content of a customized newspaper (respectively, advertising/coupon circular) that may be printed and physically sent to user U via other methods. In general, the target objects included in the printed document delivered to user U are those with the highest median predicted interest among a group G of users, where group G consists of either the single off-line user U, a set of off-line users who are demographically similar to user U, or a set of off-line users who are in the same geographic area and thus on the same newspaper delivery route. In a variation, user group G is clustered into several subgroups G1...Gk; an average user profile Pi is created from each subgroup Gi; for each article T and each user profile Pi, the interest in T by a hypothetical user with user profile Pi is predicted, and the interest of article T to group G is taken to be the maximum interest in article T by any of these k hypothetical users; finally, the customized newspaper for user group G is constructed from those articles of greatest interest to group G.

The filtering technology of the news clipping service is not limited to news articles provided by a single source, but may be extended to articles or target objects collected from any number of sources. For example, rather than identifying new news articles of interest, the technology may identify new or updated World Wide Web pages of interest. In a second application, termed "broadcast clipping," where individual users desire to broadcast messages to all interested users, the pool of news articles is replaced by a pool of messages to be broadcast, and these messages are sent to the broadcast-clipping-service subscribers most interested in them. In a third application, the system scans the transcripts of all real-time spoken or written discussions on the network that are currently in progress and designated as public, and employs the news-clipping technology to rapidly identify discussions that the user may be interested in joining, or to rapidly identify and notify users who may be interested in joining an ongoing discussion. In a fourth application, the method is used as a post-process that filters and ranks in order of interest the many target objects found by a conventional database search, such as a search for all homes selling for under $200,000 in a given area, for all 1994 news articles about Marcia Clark, or for all Italian-language films. In a fifth application, the method is used to filter and rank the links in a hypertext document by estimating the user's interest in the document or other object

IPR2025-00200 / 722 Patent Page 275 of 1171 EX1002

associated with each link. In a sixth application, paying advertisers, who may be companies or individuals, are the source of advertisements or other messages, which take the place of the news articles in the news clipping service. A consumer who buys a product is deemed to have provided positive relevance feedback on advertisements for that product, and a consumer who buys a product apparently because of a particular advertisement (for example, by using a coupon clipped from that advertisement) is deemed to have provided particularly high relevance feedback on that

advertisement. Such feedback may be communicated to a proxy server by the consumer's client processor (if the consumer is making the purchase electronically), by the retail vendor, or by the credit-card reader (at the vendor's establishment) that the consumer uses to pay for the purchase. Given a database of such relevance feedback, the disclosed technology is then used to match advertisements with those users who are most interested in them; advertisements selected for a user are presented to that user by any one of several means, including electronic mail, automatic display on the users screen, or printing them on a printer at a retail establishment where the consumer is paying for a purchase. The threshold distance used to identify interest may be increased for a particular advertisement, causing the system to present that advertisement to more users, in accordance with the amount that the advertiser is willing to pay.

A further use of the capabilities of this system is to manage a user's investment portfolio. Instead of recommending articles to the user, the system recommends target objects that are investments. As illustrated above by the example of stock market investments, many different attributes can be used together to profile each investment. The user's past investment behavior is characterized in the user's search profile set or target profile interest summary, and this information is used to match the user with stock opportunities (target objects) similar in nature to past investments. The rapid profiling method described above may be used to determine a rough set of preferences for new users. Quality attributes used in this system can include negatively weighted attributes, such as a measurement of fluctuations in dividends historically paid by the investment, a quality attribute that would have a strongly negative weight for a conservative investor dependent on a regular flow of investment income. Furthermore, the user can set filter parameters so that the system can monitor stock prices and automatically take certain actions, such as placing buy or sell orders, or e-mailing or paging the user with a notification, when certain stock performance characteristics are met. Thus, the system can immediately notify the user

IPR2025-00200 / 722 Patent Page 276 of 1171 EX1002

when a selected stock reaches a predetermined price, without the user having to monitor the stock market activity. The user's investments can be profiled in part by a "type of investment" attribute (to be used in conjunction with other attributes), which distinguishes among bonds, mutual funds, growth stocks, income stocks, etc., to thereby segment the user's portfolio according to investment type. Each investment type can then be managed to identify investment opportunities and the user can identify the desired ratio of investment capital for each type.

E-mail Filter

In addition to the news clipping service described above, the system for customized electronic identification of desirable objects functions in an e_mail environment in a similar but slightly different manner. The news clipping service selects and retrieves news information that would not otherwise reach its subscribers. But at the same time, large numbers of e-mail messages do reach users, having been generated and sent by humans or automatic programs. These users need an e-mail filter, which automatically processes the messages received. The necessary processing includes a determination of the action to be taken with each message, including, but not limited to: filing the message, notifying the user of receipt of a high priority message, automatically responding to a message. The e-mail filter system must not require too great an investment on the part of the user to learn and use, and the user must have confidence in the appropriateness of the actions automatically taken by the system. The same filter may be applied to voice mail messages or facsimile messages that have been converted into electronically stored text, whether automatically or at the user's request, via the use of w ell-known techniques for speech recognition or optical character recognition.

The filtering problem can be defined as follows: a message processing function MPF(*) maps from a received message (document) to one or more of a set of actions. The actions, which may be quite specific, may be either predefined or customized by the use r. Each action A has an appropriateness function F_A (*,*) such that F_A (U,D) returns a real number, representing the appropriateness of selecting action A on behalf of user U when us er U is in receipt of message D. For example, if D comes from a credible source and is marked urgent, then discarding the message has a high cost to the user and has low appropriateness, so that $F_{discard}$ (U,D) is small, whereas alerting the user of receipt of the message is highly appropriate, so that F_{alert} (U,D) is large. Given the determined

appropriateness function, the function MPF(D) is used to automatically select the appropriate action or actions. As an example, the following set of actions might be useful:

1. Urgently notify user of receipt of message
2. Insert message into queue for user to read later
3. Insert message into queue for user to read later, and suggest that user reply
4. Insert message into queue for user to read later, and suggest that user forward it to individual R
5. Summarize message and insert summary into queue
6. Forward message to user's secretary
7. File message in directory X
8. File message in directory Y
9. Delete message (i.e., ignore message and do not save)
10. Notify sender that further messages on this subject are unwanted

Notice that actions 8 and 9 in the sample list above are designed to filter out messages that are undesirable to the user or that are received from undesirable sources, such as pesky salespersons, by deleting the unwanted message and/or sending a reply that indicates that messages of this type will not be read. The appropriateness functions must be tailored to describe the appropriateness of carrying out each action given the target profile for a particular document, and then a message processing function MPF can be found which is in some sense optimal with respect to the appropriateness function. One reasonable choice of MPF always picks the action with highest appropriateness, an d in cases where multiple actions are highly appropriate and are also compatible with each other, selects more than one action: for example, it may automatically reply to a message and also file the same message in directory X, so that the value of MPF(D) is the set \{reply, file in directory X\}. In cases where the appropriateness of even the most appropriate action falls below a user-specified threshold, as should happen for messages of an unfamiliar type, the system asks the user for confirmation of the action(s) selected by MPF. In addition, in cases where MPF selects one action over another action that is nearly as appropriate, the system also asks the user for confirmation: for example, mail should not be deleted if it is nearly as appropriate to let the user see it.

It is possible to write appropriateness functions manually, but the time necessary and lack of user expertise render this solution impractical. The automatic training of this system is preferable, using the automatic user profiling system described above. Each received

IPR2025-00200 / 722 Patent Page 278 of 1171 EX1002

document is viewed as a target object whose profile includes such attributes as the entire text of the document (represented as TF/IDF scores), document sender, date sent, document length, date of last document received from this sender, key words, list of other addressees, etc. It was disclosed above how to estimate an interest function on profiled target objects, using relevance feedback together with measured similarities among target objects and among users. In the con text of the e-mail filter, the task is to estimate several appropriateness functions F_A (*,*), one per action. This is handled with exactly the same method as was used earlier to estimate the topical interest function f(*,*). Relevance feedback in this case is provided by the user's observed actions over time: whenever user U chooses action A on document D, either freely or by choosing or confirming an action recommended by the system, this is taken to mean that the appropriateness of action A on document D is high, particularly if the user takes this action A immediately after seeing document D. A presumption of no appropriateness (corresponding to the earlier presumption of no interest) is used so that action A is considered inappropriate on a document unless the user or similar users have taken action A on this document or similar documents. In particular, if no similar document has been seen, no action is considered especially appropriate, and the e-mail filter asks the user to specify the appropriate action or confirm that the action chosen by the e-mail filter is the appropriate one.

Thus, the e-mail filter learns to take particular actions on e-mail messages that have certain attributes or combinations of attributes. For example, messages from John Doe that originate in the (212) area code may prompt the system to forward a copy by fax transmission to a given fax number, or to file the message in directory X on the user's client processor. A variation allows active requests of this form from the us er, such as a request that any message from John Doe be forwarded to a desired fax number until further notice. This active user input requires the use of a natural language or form-based interface for which specific commands are associated with particular attributes and combinations of attributes.

Update Notification

A very important and novel characteristic of the architecture is the ability to identify new or updated target objects that are relevant to the user, as determined by the user's search profile set or target profile interest summary. ("Updated target objects" include revised versions of documents and new models of purchasable goods.) The system may notify the user of these relevant target objects by an electronic notification such as an e-mail message

IPR2025-00200 / 722 Patent Page 279 of 1171 EX1002

or facsimile transmission. In the variation where the system sends an e-mail message, the user's e-mail filter can then respond appropriately to the notification, for instance, by bringing the notification immediately to the user's personal attention, or by automatically submitting an electronic request to purchase the target object named in the notification. A simple example of the latter response is for the e-mail filter to retrieve an on-line document at a nominal or zero charge, or request to buy a purchasable of limited quantity such as a used product or an auctionable.

ACTIVE NAVIGATION (BROWSING)

Browsing by Navigating Through a Cluster Tree

A hierarchical cluster tree imposes a useful organization on a collection of target objects. The tree is of direct use to a user who wishes to browse through all the target objects in the tree. Such a user may be exploring the collection with or without a well-specified goal. The tree's division of target objects into coherent clusters provides an efficient method whereby the user can locate a target object of interest. The user first chooses one of the highest level (largest) clusters from a menu, and is presented with a menu listing the subclusters of said cluster, whereupon the user may select one of these subclusters. The system locates the subcluster, via the appropriate pointer that was stored with the larger cluster, and allows the user to select one of its subclusters from another menu. This process is repeated until the user comes t o a leaf of the tree, which yields the details of an actual target object. Hierarchical trees allow rapid selection of one target object from a large set. In ten menu selections from menus of ten items (subclusters) each, one can reach 10^{10} =10,000,000,000 (ten billion) items. In the preferred embodiment, the user views the menus on a computer screen or terminal screen and selects from them with a keyboard or mouse. However, the user may also make selections over the telephone, with a voice synthesizer reading the menus and the user selecting subclusters via the telephone's touch-tone keypad. In another variation, the user simultaneously maintains two connections to the server, a telephone voice connection and a fax connection; the server sends successive menus to the user by fax, while the user selects choices via the telephone's touch-tone keypad.

Just as user profiles commonly include an associative attribute indicating the user's degree of interest in each target object, it is useful to augment user profiles with an additional associative attribute indicating the user's degree of interest in each cluster in the hierarchical cluster tree. This degree of interest may be estimated numerically as the

IPR2025-00200 / 722 Patent Page 280 of 1171 EX1002

number of subclusters or target objects the user has selected from menus associated with the given cluster or its subclusters, expressed as a proportion of the total number of subclusters or target objects the user has selected. This associative attribute is particularly valuable if the hierarchical tree was built using "soft" or "fuzzy" clustering, which allows a subcluster or target object to appear in multiple clusters: if a target document appears in both the "sports" and the "humor" clusters, and the user selects it from a menu associated with the "humor" cluster, then the system increases its association between the user and the "humor" cluster but not its association between the user and the "sports" cluster.

Labeling Clusters

Since a user who is navigating the cluster tree is repeatedly expected to select one of several subclusters from a menu, these subclusters must be usefully labeled (at step 503), in such a way as to suggest their content to the human user. It is straightforward to include some basic information about each subcluster in its label, such as the number of target objects the subcluster contains (possibly just 1) and the number of these that have been added or updated recently. However, it is also necessary to display additional information that indicates the cluster's content. This content-descriptive information may be provided by a human, particularly for large or frequently accessed clusters, but it may also be generated automatically. The basic automatic technique is simply to display the cluster's "characteristic value" for each of a few highly weighted attributes. With numeric attributes, this may be taken to mean the cluster's average value for that attribute: thus, if the "year of release" attribute is highly weighted in predicting which movies a user will like, then it is useful to display average year of release as part of each cluster's label. Thus the user sees that one cluster consists of movies that were released around 1962, while another consists of movies from around 1982. For short textual attributes, such as "title of movie" or "title of document," the system can display the attribute's value for the cluster member (target object) whose profile is most similar to the cluster's profile (the mean profile for all members of the cluster), for example, the title of the most typical movie in the cluster. For longer textual attributes, a useful technique is to select those terms for which the amount by which the term's average TF/IDF score across members of the cluster exceeds the term's average TF/IDF score across all tar get objects is greatest, either in absolute terms or else as a fraction of the standard deviation of the term's TF/IDF score across all target objects. The selected terms are replaced with their morphological stems, eliminating duplicates (so that if bot h "slept" and "sleeping" were selected, they would be replaced by the single term

IPR2025-00200 / 722 Patent Page 281 of 1171 EX1002

"sleep") and optionally eliminating close synonyms or collocates (so that if both "nurse" and "medical" were selected, they might both be replaced by a single term such as "nurse," "medical," "medicine," or "hospital"). The resulting set of terms is displayed as part of the label. Finally, if freely redistributable thumbnail photographs or other graphical images are associated with some of the target objects in the cluster f or labeling purposes, then the system can display as part of the label the image or images whose associated target objects have target profiles most similar to the cluster profile.

Users' navigational patterns may provide some useful feedback as to the quality of the labels. In particular, if users often select a particular cluster to explore, but then quickly backtrack and try a different cluster, this may signal that the first cluster's label is misleading. Insofar as other terms and attributes can pro vide "next-best" alternative labels for the first cluster, such "next-best" labels can be automatically substituted for the misleading label. In addition, any user can locally relabel a cluster for his or her own convenience. Although a cluster label provided by a user is in general visible only to that user, it is possible to make global use of these labels via a "user labels" textual attribute for target objects, which attribute is defined for a given target object to be the concatenation of all label s provided by any user for any cluster containing that target object. This attribute influences similarity judgments: for example, it may induce the system to regard target articles in a cluster often labeled "Sports News" by users as being mildly similar to articles in an otherwise dissimilar cluster often labeled "International News" by users, precisely because the "user labels" attribute in each cluster profile is strongly associated with the term "News." The "user label" attribute is also used in the automatic generation of labels, just as other textual attributes are, so that if the user-generated labels for a cluster often include "Sports," the term "Sports" may be included in the automatically generated label as well.

It is not necessary for menus to be displayed as simple lists of labeled options; it is possible to display or print a menu in a form that shows in more detail the relation of the different menu options to each other. Thus, in a variation, the menu options are visually laid out in two dimensions or in a perspective drawing of three dimensions. Each option is displayed or printed as a textual or graphical label. The physical coordinates at which the options are displayed or printed are generated by the following sequence of steps: (1) construct for each option the cluster profile of the cluster it represents, (2) construct from each cluster profile its decomposition into a numeric vector, as described above, (3) apply singular value decomposition (SVD) to determine the set of two or three orthogonal linear

IPR2025-00200 / 722 Patent Page 282 of 1171 EX1002

axes along which these numeric vectors are most greatly differentiated, and (4) take the coordinates of each option to be the projected coordinates of that option's numeric vector along said axes. Step (3) may be varied to determine a set of, say, 6 axes, so that step (4) lays out the options in a 6-dimensional space; in this case the user may view the geometric projection of the 6-dimensional layout onto any plane passing through the origin, and may rotate this viewing plane in order to see differing configurations of the options, which emphasize similarity with respect to differing attributes in the profiles of the associated clusters. In the visual representation, the sizes of the cluster labels can be varied according to the number of objects contained in the corresponding clusters. In a further variation, all options from the parent menu are displayed in some number of dimensions, as just described, but with the option corresponding to the current menu replaced by a more prominent subdisplay of the options on the current menu; optionally, the scale of this composite display may be gradually increased over time, thereby increasing the area of the screen devoted to showing the options on the current menu, and giving the visual impression that the user is regarding the parent cluster and "zooming in" on the current cluster and its subclusters.

Further Navigational

It should be appreciated that a hierarchical cluster-tree may be configured with multiple cluster selections branching from each node or the same labeled clusters presented in the form of single branches for multiple nodes ordered in a hierarchy. In one variation, the user is able to perform lateral navigation between neighboring clusters as well, by requesting that the system search for a cluster whose cluster profile resembles the cluster profile of the currently selected cluster. If this type of navigation is performed at the level of individual objects (leaf ends), then automatic hyperlinks may be then created as navigation occurs. This is one way that nearest neighbor clustering navigation may be performed. For example, in a domain where target objects are home pages on the World Wide Web, a collection of such pages could be laterally linked to create a " virtual mall."

The simplest way to use the automatic menuing system described above is for the user to begin browsing at the top of the tree and moving to more specific subclusters. However, in a variation, the user optionally provides a query consisting of textual and/or other attributes, from which query the system constructs a profile in the manner described herein, optionally altering textual attributes as described herein before decomposing them into numeric attributes. Query profiles are similar to the search profiles in a user's search

profile set, except that their attributes are explicitly specified by a user, most often for one-time usage, and unlike search profiles, they are not automatically updated to reflect changing interests. A typical query in the domain of text articles might have "Tell me about the relation between Galileo and the Medici family" as the value of its "text of article" attribute, and 8 as the value of its "reading difficulty" attribute (that is, 8th-grade level). The system uses the method of section "Searching for Target Objects" above to automatically locate a small set of one or more clusters with profiles similar to the query profile, for example, the articles they contain are written at roughly an 8th-grade level and tend to mention Galileo and the Medicis. The user may start browsing at any of these clusters, and can move from it to subclusters, superclusters, and other nearby clusters. For a user who is looking for something in particular, it is generally less efficient to start at the largest cluster and repeatedly select smaller subclusters than it is to write a brief description of what one is looking for and then to move to nearby clusters if the objects initially recommended are not precisely those desired.

Although it is customary in information retrieval systems to match a query to a document, an interesting variation is possible where a query is matched to an already answered question. The relevant domain is a customer service center, electronic newsgroup , or Better Business Bureau where questions are frequently answered. Each new question-answer pair is recorded for future reference as a target object, with a textual attribute that specifies the question together with the answer provided. As explained earlier with reference to document titles, the question should be weighted more heavily than the answer when this textual attribute is decomposed into TF/IDF scores. A query specifying "Tell me about the relation between Galileo and the Medici family" as t he value of this attribute therefore locates a cluster of similar questions together with their answers. In a variation, each question-answer pair may be profiled with two separate textual attributes, one for the question and one for the answer. A query might then locate a cluster by specifying only the question attribute, or for completeness, both the question attribute and the (lower-weighted) answer attribute, to be the text "Tell me about the relation between Galileo and the Medici family."

The filtering technology described earlier can also aid the user in navigating among the target objects. When the system presents the user with a menu of subclusters of a cluster C of target objects, it can simultaneously present an additional menu of the most interesting target objects in cluster C, so that the user has the choice of accessing a subcluster or

IPR2025-00200 / 722 Patent Page 284 of 1171 EX1002

directly accessing one of the target objects. If this additional menu lists n target objects, then for each I between 1 and n inclusive, in increasing ord er, the Ith most prominent choice on this additional menu, which choice is denoted Top(C,i), is found by considering all target objects in cluster C that are further than a threshold distance t from all of Top(C,1), Top(C,2), ... Top(C, I-1), and selecting the one in which the user's interest is estimated to be highest. If the threshold distance t is 0, then the menu resulting from this procedure simply displays the n most interesting objects in cluster C, but the threshold distance may be increased to achieve more variety in the target objects displayed. Generally the threshold distance t is chosen to be an affine function or other function of the cluster variance or cluster diameter of the cluster C.

As a novelty feature, the user U can "masquerade" as another user V, such as a prominent intellectual or a celebrity supermodel; as long as user U is masquerading as user V, the filtering technology will recommend articles not according to user U's preferences, but rather according to user V's preferences. Provided that user U has access to the user-specific data of user V, for example because user V has leased these data to user U for a financial consideration, then user U can masquerade as user V by instructing user U's proxy server S to temporarily substitute user V's user profile and target profile interest summary for user U's. In a variation, user U has access to an average user profile and an composite target profile interest summary for a group G of users; by instructing proxy server S to substitute these for user U's user-specific data, user U can masquerade as a typical member of group G, as is useful in exploring group preferences for sociological, political, or market research. More generally, user U may "partially masquerade" as another user V or group G, by instructing proxy server S to temporarily replace user U's user-specific data with a weighted average of user U's user-specific data and the user-specific data for user V and group G.

Menu Organization

Although the topology of a hierarchical cluster tree is fixed by the techniques that build the tree, the hierarchical menu presented to the user for the user's navigation need not be exactly isomorphic to the cluster tree. The menu is typically a somewhat modified version of the cluster tree, reorganized manually or automatically so that the clusters most interesting to a user are easily accessible by the user. In order to automatically reorganize the menu in a user-specific way, the system first attempts automatically to identify existing clusters that are of interest to the user. The system may identify a cluster as interesting

because the user often accesses target objects in that cluster -- or, in a more sophisticated variation, because the user is predicted to have high interest in the cluster's profile, using the methods disclosed herein for estimating interest from relevance feedback.

Several techniques can then be used to make interesting clusters more easily accessible. The system can at the user's request or at all times display a special list of the most interesting clusters, or the most interesting subclusters of the current cluster, so that the user can select one of these clusters based on its label and jump directly to it. In general, when the system constructs a list of interesting clusters in this way, the Ith most prominent choice on the list, which choice is denoted Top(I), is found by considering all appropriate clusters C that are further than a threshold distance t from all of Top(1), Top(2), ... Top(I-1), and selecting the one in which the user's interest is estimated to be highest. Here the threshold distance t is optionally dependent on the computed cluster variance or cluster diameter of the profiles in the latter cluster. Several techniques that reorganize the hierarchical menu tree are also useful. First, menus can be reorganized so that the most interesting subcluster choices appear earliest on the menu, or are visually marked as interesting; for example, their labels are displayed in a special color or type face, or are displayed together with a number or graphical image indicating the likely level of interest. Second, interesting clusters can be moved to menus higher in the tree, i.e., closer to the root of the tree, so that they are easier to access if the user starts browsing at the root of the tree. Third, uninteresting clusters can be moved to menus lower in the tree, to make room for interesting clusters that are being moved higher. Fourth, clusters with an especially low interest score (representing active dislike) can simply be suppressed from the menus; thus, a user with children may assign an extremely negative weight to the "vulgarity" attribute in the determination of q, so that vulgar clusters and documents will not be available at all. As the interesting clusters and the documents in them migrate toward the top of the tree, a customized tree develops that can be more efficiently navigated by the particular user. If menus are chosen so that each menu item is chosen with approximately equal probability, then the expected number of choices the user has to make is minimized. If, for example, a user frequently accessed target objects whose profiles resembled the cluster profile of cluster (a, b, d) in Figure 8 then the menu in Figure 9 could be modified to show the structure illustrated in Figure 10.

In the variation where the general techniques disclosed herein for estimating a user's interest from relevance feedback are used to identify interesting clusters, it is possible for

IPR2025-00200 / 722 Patent Page 286 of 1171 EX1002

a user U to supply "temporary relevance feedback" to indicate a temporary interest that is added to his or her usual interests. This is done by entering a query as described above, i.e., a set of textual and other attributes that closely match the user's interests of the moment . This query becomes "active," and affects the system's determination of interest in either of two ways. In one approach, an active query is treated as if it were any other target object, and by virtue of being a query, it is taken to have received relevance feedback that indicates especially high interest. In an alternative approach, target objects X whose target profiles are similar to an active query's profile are simply considered to have higher quality q(U, X), in that q(U, X) is incremented by a term that increases with target object X's similarity to the query profile. Either strategy affects the usual interest estimates: clusters that match user U's usual interests (and have high quality q(*)) are still considered to be of interest, and clusters w hose profiles are similar to an active query are adjudged to have especially high interest. Clusters that are similar to both the query and the user's usual interests are most interesting of all. The user may modify or deactivate an active query at any ti me while browsing. In addition, if the user discovers a target object or cluster X of particular interest while browsing, he or she may replace or augment the original (perhaps vague) query profile with the target profile of target object or cluster X, t hereby amplifying or refining the original query to indicate an particular interest in objects similar to X. For example, suppose the user is browsing through documents, and specifies an initial query containing the word "Lloyd's," so that the system predicts documents containing the word "Lloyd's" to be more interesting and makes them more easily accessible, even to the point of listing such documents or clusters of such documents, as described above. In particular, certain articles about insurance containing the phrase "Lloyd's of London" are made more easily accessible, as are certain pieces of Welsh fiction containing phrases like "Lloyd's father." The user browses while this query is active, and hits upon a useful article describing the relation of Lloyd's of London to other British insurance houses; by replacing or augmenting the query with the full text of this article, the user can turn the attention of the system to other documents that resemble this article, such as documents about British insurance houses, rather than Welsh folk tales.

In a system where queries are used, it is useful to include in the target profiles an associative attribute that records the associations between a target object and whatever terms are employed in queries used to find that target object. The association score of target object X with a particular query term T is defined to be the mean relevance feedback on

target object X, averaged over just those accesses of target object X that were made while a query containing term T was active, multiplied by the negated logarithm of term T's global frequency in all queries. The effect of this associative attribute is to increase the measured similarity of two documents if they are good responses to queries that contain the same terms. A further maneuver can be used to improve the accuracy of responses to a query: in the summation used to determine the quality q(U, X) of a target object X, a term is included that is proportional to the sum of association scores between target object X and each term in the active query, if any, so that target objects that are closely associated with terms in an active query are determined to have higher quality and therefore higher interest for the user. To complement the system's automatic reorganization of the hierarchical cluster tree, the user can be given the ability to reorganize the tree manually, as he or she sees fit. Any changes are optionally saved on the user's local storage device so that they will affect the presentation of the tree in future sessions. For example, the user can choose to move or copy menu options to other menus, so that useful clusters can thereafter be chosen directly from the root menu of the tree or from other easily accessed or topically appropriate menus. In an other example, the user can select clusters $C_1, C_2, \ldots C_k$ listed on a particular menu M and choose to remove these clusters from the menu, replacing them on the menu with a single aggregate cluster M' containing all the target objects from clusters $C_1, C_2, \ldots C_k$. In this case, the immediate subclusters of new cluster M' are either taken to be clusters $C_1, C_2, \ldots C_k$ themselves, or else, in a variation similar to the "scatter-gather" method, are automatically computed by clustering the set of all the subclusters of clusters $C_1, C_2, \ldots C_k$ according to the similarity of the cluster profiles of these subclusters.

Electronic Mall

In one application, the browsing techniques described above may be applied to a domain where the target objects are purchasable goods. When shoppers look for goods to purchase over the Internet or other electronic media, it is typically necessary to display thousands or tens of thousands of products in a fashion that helps consumers find the items they are looking for. The current practice is to use hand-crafted menus and sub-menus in which similar items are grouped together. It is possible to use the automated clustering and browsing methods described above to more effectively group and present the items. Purchasable items can be hierarchically clustered using a plurality of different criteria. Useful attributes for a purchasable item include but are not limited to a textual description and predefined category labels (if available), the unit price of the item, and an associative

IPR2025-00200 / 722 Patent Page 288 of 1171 EX1002

attribute listing the users who have bought this item in the past. Also useful is an associative attribute indicating which other items are often bought on the same shopping "trip" as this item; items that are often bought on the same trip will be judged similar with respect to this attribute, so tend to be grouped together. Retailers may be interested in utilizing a similar technique for purposes of predicting both the nature and relative quantity of items which are likely to be popular to their particular clientele. This prediction may be made by using aggregate purchasing records as the search profile set from which a collection of target objects is recommended. Estimated customer demand which is indicative of (relative) inventory quantity for each target object item is determined by measuring the cluster variance of that item compared to another target object item (which is in stock).

As described above, hierarchically clustering the purchasable target objects results in a hierarchical menu system, in which the target objects or clusters of target objects that appear on each menu can be labeled by names or icons and displayed in a two -dimensional or three-dimensional menu in which similar items are displayed physically near each other or on the same graphically represented "shelf." As described above, this grouping occurs both at the level of specific items (such as standard size Ivory soap or large Breck shampoo) and at the level of classes of items (such as soaps and shampoos). When the user selects a class of items (for instance, by clicking on it), then the more specific level of detail is displayed. It is neither necessary nor desirable to limit each item to appearing in one group; customers are more likely to find an object if it is in multiple categories. Non-purchasable objects such as artwork, advertisements, and free samples may also be added to a display of purchasable objects, if they are associated with (liked by) substantially the same users as are the purchasable objects in the display.

Network Context of the Browsing System

The files associated with target objects are typically distributed across a large number of different servers S1-So and clients C1-Cn. Each file has been entered into the data storage medium at some server or client in any one of a number of ways, including, but not limited to: scanning, keyboard input, e-mail, FTP transmission. automatic synthesis from another file under the control of another computer program. While a system to enable users to efficiently locate target objects may store its hierarchical cluster tree on a single centralized machine, greater efficiency can be achieved if the storage of the hierarchical cluster tree is distributed across many machines in the network. Each cluster C, including

IPR2025-00200 / 722 Patent Page 289 of 1171 EX1002

single-member clusters (target objects), is digitally represented by a file F, which is multicast to a topical multicast tree MT(C1); here cluster C1 is either cluster C itself or some supercluster of cluster C. In this way, file F is stored at multiple servers, for redundancy. The file F that represents cluster C contains at least the following data:

1. The cluster profile for cluster C, or data sufficient to reconstruct this cluster profile. 2. The number of target objects contained in cluster C. 3. A human-readable label for cluster C, as described in section "Labeling Clusters" above. 4. If the cluster is divided into subclusters, a list of pointers to files representing the subclusters. Each pointer is an ordered pair containing naming, first, a file, and second, a multicast tree or a specific server where that file is stored. 5. If the cluster consists of a single target object, a pointer to the file corresponding to that target object.

The process by which a client machine can retrieve the file F from the multicast tree MT(C1) is described above in section "Retrieving Files from a Multicast Tree." Once it has retrieved file F, the client can perform further tasks pertaining to this cluster, such as displaying a labeled menu of subclusters, from which the user may select subclusters for the client to retrieve next.

The advantage of this distributed implementation is threefold. First, the system can be scaled to larger cluster sizes and numbers of target objects, since much more searching and data retrieval can be carried out concurrently. Second, the system is fault-tolerant in that partial matching can be achieved even if portions of the system are temporarily unavailable. It is important to note here the robustness due to redundancy inherent in our design - data is replicated at tree sites so that even if a server is down, the data can be located elsewhere.

The distributed hierarchical cluster tree can be created in a distributed fashion, that is, with the participation of many processors. Indeed, in most applications it should be recreated from time to time, because as users interact with target objects, the associative attributes in the target profiles of the target objects change to reflect these interactions; the system's similarity measurements can therefore take these interactions into account when judging similarity, which allows a more perspicuous cluster tree to be built The key technique is the following procedure for merging n disjoint cluster trees, represented respectively by files F1...Fn in distributed fashion as described above, into a combined cluster tree that contains all the target objects from all these trees. The files F1...Fn are described above, except that the cluster labels are not included in the representation. The following steps are executed by a server S1, in response to a request message from another

IPR2025-00200 / 722 Patent Page 290 of 1171 EX1002

server S0, which request message includes pointers to the files F1...Fn. 1. Retrieve files F1...Fn. 2. Let L and M be empty lists. 3. For each file Fi from among F1...Fn: 4. If file Fi contains pointers to subcluster files, add these pointers to list L. 5. If file Fi represents a single target object, add a pointer to file Fi to list L. 6. For each pointer X on list L, retrieve the file that pointer P points to and extract the cluster profile P(X) that this file stores. 7. Apply a clustering algorithm to group the pointers X on list L according to the distances between their respective cluster profiles P(X). 8. For each (nonempty) resulting group C of pointers: 9. If C contains only one pointer, add this pointer to list M; 10. otherwise, if C contains exactly the same subcluster pointers as does one of the files Fi from among F1...Fn, then add a pointer to file Fi to list M; 11. otherwise: 12. Select an arbitrary server S2 on the network, for example by randomly selecting one of the pointers in group C and choosing the server it points to. 13. Send a request message to server S2 that includes the subcluster pointers in group C and requests server S2 to merge the corresponding subcluster trees. 14. Receive a response from server S2, containing a pointer to a file G that represents the merged tree. Add this pointer to list M. 15. For each file Fi from among F1...Fn: 16. If list M does not include a pointer to file Fi, send a message to the server or servers storing Fi instructing them to delete file Fi. 17. Create and store a file F that represents a new cluster, whose subcluster pointers are exactly the subcluster pointers on list M. 18. Send a reply message to server S0, which reply message contains a pointer to file F and indicates that file F represents the merged cluster tree.

With the help of the above procedure, and the multicast tree MT full that includes all proxy servers in the network, the distributed hierarchical cluster tree for a particular domain of target objects is constructed by merging many local hierarchical cluster trees, as follows. 1. One server S (preferably one with good connectivity) is elected from the tree. 2. Server S sends itself a global request message that causes each proxy server in MT_{full} (that is., each proxy server in the network) to ask its clients for files for the cluster tree. 3. The clients of each proxy server transmit to the proxy server any files that they maintain, which files represent target objects from the appropriate domain that should be added to the cluster tree. 4. Server S forms a request R1 that, upon receipt, will cause the recipient server S1 to take the following actions: (a) Build a hierarchical cluster tree of all the files stored on server S1 that are maintained by users in the user base of S1. These files correspond to target objects from the appropriate domain. This cluster tree is typically stored entirely on S1, but may in principle be stored in a distributed fashion.

(b) Wait until all servers to which the server S1 has propagated request R have sent the recipient reply messages containing pointers to cluster trees. (c) Merge together the cluster tree created in step 5(a) and the cluster trees supplied in step 5(b), by sending any server (such as S1 itself) a message requesting such a merge, as described above. (d) Upon receiving a reply to the message sent in (c), which reply includes a pointer to a file representing the merged cluster tree, forward this reply to the sender of request R1, unless this is S1 itself. 5. Server S sends itself a global request message that causes all servers in MT_{full} to act on embedded request R1. 6. Server S receives a reply to the message it sent in 5(c). This reply includes a pointer to a file F that represents the completed hierarchical cluster tree. Server S multicasts file F to all proxy servers in MT_{full}. Once the hierarchical cluster tree has been created as above, server S can send additional messages through the cluster tree, to arrange that multicast trees MT(C) are created for sufficiently large clusters C, and that each file F is multicast to the tree MT(C), where C is the smallest cluster containing file F.

MATCHING USERS FOR VIRTUAL COMMUNITIES

Virtual Communities

Computer users frequently join other users for discussions on computer bulletin boards, newsgroups, mailing lists, and real-time chat sessions over the computer network, which may be typed (as with Internet Relay Chat (IRC)), spoken (as with Internet phone), or videoconferenced. These forums are herein termed "virtual communities." In current practice, each virtual community has a specified topic, and users discover communities of interest by word of mouth or by examining a long list of communities (typically hundreds or thousands). The users then must decide for themselves which of thousands of messages they find interesting from among those posted to the selected virtual communities, that is, made publicly available to members of those communities. If they desire, they may also write additional messages and post them to the virtual communities of their choice. The existence of thousands of Internet bulletin boards (also termed newsgroups) and countless more Internet mailing lists and private bulletin board services (BBS's) demonstrates the very strong interest among members of the electronic community in forums for the discussion of ideas about almost any subject imaginable. Presently, virtual community creation proceeds in a haphazard form, usually instigated by a single individual who decides that a topic is worthy of discussion. There are protocols on the Internet for voting to determine whether

a newsgroup should be created, but there is a large hierarchy of newsgroups (which begin with the prefix "alt.") that do not follow this protocol.

The system for customized electronic identification of desirable objects described herein can of course function as a browser for bulletin boards, where target objects are taken to be bulletin boards, or subtopics of bulletin boards, and each target profile is the cluster profile for a cluster of documents posted on some bulletin board. Thus, a user can locate bulletin boards of interest by all the navigational techniques described above, including browsing and querying. However, this method only serves to locate existing virtual communities. Because people have varied and varying complex interests, it is desirable to automatically locate groups of people with common interests in order to form virtual communities. The Virtual Community Service (VCS) described below is a network-based agent that seeks out users of a network with common interests, dynamically creates bulletin boards or electronic mailing lists for those users, and introduces them to each other electronically via e-mail. It is useful to note that once virtual communities have been created by VCS, the other browsing and filtering technologies described above can subsequently be used to help a user locate particular virtual communities (whether pre-existing or automatically generated by VCS); similarly, since the messages sent to a given virtual community may vary in interest and urgency for a user who has joined that community, these browsing and filtering technologies (such as the e-mail filter) can also be used to alert the user to urgent messages and to screen out uninteresting ones.

The functions of the Virtual Community Service are general functions that could be implemented on any network ranging from an office network in a small company to the World Wide Web or the Internet. The four main steps in the procedure are: 1. Scan postings to existing virtual communities. 2. Identify groups of users with common interests. 3. Match users with virtual communities, creating new virtual communities when necessary. 4. Continue to enroll additional users in the existing virtual communities.

More generally, users may post messages to virtual communities pseudonymously, even employing different pseudonyms for different virtual communities. (Posts not employing a pseudonymous mix path may, as usual, be considered to be posts employing a non-secure pseudonym, namely the user's true network address.) Therefore, the above steps may be expressed more generally as follows 1. Scan pseudonymous postings to existing virtual communities. 2. Identify groups of pseudonyms whose associated users have common interests. 3. Match pseudonymous users with virtual communities,

creating new virtual communities when necessary. 4. Continue to enroll additional pseudonymous users in the existing virtual communities.

Each of these steps can be carried out as described below.

Scanning

Using the technology described above, Virtual Community Service constantly scans all the messages posted to all the newsgroups and electronic mailing lists on a given network, and constructs a target profile for each message found. The network can be the Internet, or a set of bulletin boards maintained by America Online, Prodigy, or CompuServe, or a smaller set of bulletin boards that might be local to a single organization, for example a large company, a law firm, or a university. The scanning activity need not be confined to bulletin boards and mailing lists that were created by Virtual Community Service, but may also be used to scan the activity of communities that predate Virtual Community Service or are otherwise created by means outside the Virtual Community Service system, provided that these communities are public or otherwise grant their permission.

The target profile of each message includes textual attributes specifying the title and body text of the message. In the case of a spoken rather than written message, the latter attribute may b e computed from the acoustic speech data by using a speech recognition system. The target profile also includes an associative attribute listing the author(s) and designated recipient(s) of the message, where the recipients may be individuals and/or entire virtual communities; if this attribute is highly weighted, then the system tends to regard messages among the same set of people as being similar or related, even if the topical similarity of the messages is not clear from their content, as may happen when some of the messages are very short. Other important attributes include the fraction of the message that consists of quoted material from previous messages, as well as attributes that are generally useful in characterizing documents, such as the message's date, length, and reading level.

Virtual Community Identification

Next, Virtual Community Service attempts to identify groups of pseudonymous users with common interests. These groups, herein termed "pre-communities," are represented as sets of pseudonyms. Whenever Virtual Community Service identifies a pre-community, it will subsequently attempt to put the users in said pre-community in contact with each other, as described below. Each pre-community is said to be "determined" by a cluster of messages, pseudonymous users, search profiles, or target objects.

IPR2025-00200 / 722 Patent Page 294 of 1171 EX1002

In the usual method for determining pre-communities, Virtual Community Service clusters the messages that were scanned and profiled in the above step, based on the similarity of those messages ' computed target profiles, thus automatically finding threads of discussion that show common interests among the users. Naturally, discussions in a single virtual community tend to show common interests; however, this method uses all the texts from every available virtual community, including bulletin boards and electronic mailing lists. Indeed, a user who wishes to initiate or join a discussion on some topic may send a "feeler message" on that topic to a special mailing list designated for feeler mess ages; as a consequence of the scanning procedure described above, the feeler message is automatically grouped with any similarly profiled messages that have been sent to this special mailing list, to topical mailing lists, or to topical bulletin boards. The clustering step employs "soft clustering," in which a message may belong to multiple clusters and hence to multiple virtual communities. Each cluster of messages that is found by Virtual Community Service and that is of sufficient size (for example, 10 -20 different messages) determines a pre-community whose members are the pseudonymous authors and recipients of the messages in the cluster. More precisely, the pre-community consists of the various pseudonyms under which the messages in the cluster were sent and received.

Alternative methods for determining a pre-community, which do not require the scanning step above, include the following: 1. Pre-communities can be generated by grouping together users who have similar interests of any sort, not merely Individuals who have already written or received messages about similar topics. If the user profile associated with each pseudonym indicates the user's interests, for example through an associative attribute that indicates the documents or Web sites a user likes, then pseudonyms can be clustered based on the similarity of their associated user profiles, and each of the resulting clusters of pseudonyms determines a pre-community comprising the pseudonyms in the cluster. 2. If each pseudonym has an associated search profile set formed through participation in the news clipping service described above, then all search profiles of all pseudonymous users can be clustered based on their similarity, and each cluster of search profiles determines a pre-community whose members are the pseudonyms from whose search profile sets the search profiles in the cluster are drawn. Such groups of people have been reading about the same topic (or, more generally, accessing similar target objects) and so presumably share an interest. 3. If users participate in a news clipping service or any other filtering or browsing system for target objects, then an individual user can

pseudonymously request the formation of a virtual community to discuss a particular cluster of one or more target objects known to that system. This cluster of target objects determines a pre-community consisting of the pseudonyms of users determined to be most interested in that cluster (for example, users who have search profiles similar to the cluster pro file), together with the pseudonym of the user who requested formation of the virtual community.

Matching Users with Communities

Once Virtual Community Service identifies a cluster C of messages, users, search profiles, or target objects that determines a pre-community M, it attempts to arrange for the members of this pre-community to have the chance to participate in a common virtual community V. In many cases, an existing virtual community V may suit the needs of the pre-community M. Virtual Community Service first attempts to find such an existing community V. In the case where cluster C is a cluster of messages, V may be chosen to be any existing virtual community such that the cluster profile of cluster C is within a threshold distance of the mean profile of the set of messages recently posted to virtual community V; in the case where cluster C is a cluster of users, V may be chosen to be any existing virtual community such that the cluster profile of cluster C is within a threshold distance of the mean user profile of the active members of virtual community V; in the case where the cluster C is a cluster of search profiles, V may be chosen to be any existing virtual community such that the cluster profile of cluster C is within a threshold distance of the cluster profile of the largest cluster resulting from clustering all the search profiles of active members of virtual community V; and in the case where the cluster C is a cluster of one or more target objects chosen from a separate browsing or filtering system, V may be chosen to be any existing virtual community initiated in the same way from a cluster whose cluster profile in that other system is within a threshold distance of the cluster profile of cluster C. The threshold distance used in each case is optionally dependent on the cluster variance or cluster diameter of the profile sets whose means are being compared.

If no existing virtual community V meets these conditions and is also willing to accept all the users in pre-community M as new members, then Virtual Community Service attempts to create a new virtual community V. Regardless of whether virtual community V is an existing community or a newly created community, Virtual Community Service sends an e-mail message to each pseudonym P in pre-community M whose associated user U does not already belong to virtual community V (under pseudonym P) and has not

IPR2025-00200 / 722 Patent Page 296 of 1171 EX1002

previously turned down a request to join virtual community V. The e-mail message informs user U of the existence of virtual community V , and provides instructions which user U may follow in order to join virtual community V if desired; these instructions vary depending on whether virtual community V is an existing community or a new community. The message includes a credential, granted to pseudonym P, which credential must be presented by user U upon joining the virtual community V, as proof that user U was actually invited to join. If user U wishes to join virtual community V under a different pseudonym Q, user U may first transfer the credential from pseudonym P to pseudonym Q, as described above. The e-mail message further provides an indication of the common interests of the community, for example by including a list of titles of messages recently sent to the community, or a charter or introductory message provided by the community (if available), or a label generated by the methods described above that identifies the content of the cluster of messages, user profiles, search profiles, or target objects that was used to identify the pre-community M.

If Virtual Community Service must create a new community V, several methods are available for enabling the members of the new community to communicate with each other. If the pre-community M is large, for example containing more than 50 users, then Virtual Community Service typically establishes either a multicast tree, as described below, or a widely-distributed bulletin board, assigning a name to the new bulletin board. If the pre-community M has fewer members, for example 2-50, Virtual Community Service typically establishes either a multicast tree, as described below, or an e-mail mailing list. If the new virtual community V was determined by a cluster of messages, then Virtual Community Service kicks off the discussion by distributing these messages to all members of virtual community V. In addition to bulletin boards and mailing lists, alternative fora that can be created and in which virtual communities can gather include real-time typed or spoken conversations (or engagement or distributed multi-user applications including video games) over the computer network and physical meetings, any of which can be scheduled by a partly automated process wherein Virtual Community Service requests meeting time preferences from all members of the pre-community M and then notifies these individuals of an appropriate meeting time.

Continued Enrollment

Even after creation of a new virtual community, Virtual Community Service continues to scan other virtual communities for new messages whose target profiles are

IPR2025-00200 / 722 Patent Page 297 of 1171 EX1002

similar to the community's cluster profile (average message profile). Copies of any such messages are sent to the new virtual community, and the pseudonymous authors of these messages, as well as users who show high interest in reading such messages, are informed by Virtual Community Service (as for pre-community members, above) that they may want to join the community. Each such user can then decide whether or not to join the community. In the case of Internet Relay Chat (IRC), if the target profile of messages in a real time dialog are (or become) similar to that of a user, VCS may also send an urgent e-mail message to such user whereby the user may be automatically notified as soon as the dialog appears, if desired.

With these facilities, Virtual Community Service provides automatic creation of new virtual communities in any local or wide- area network, as well as maintenance of all virtual communities on the network, including those not created by Virtual Community Service. The core technology underlying Virtual Community Service is creating a search and clustering mechanism that can find articles that are "similar" in that the users share interests. This is precisely what was described above. One must be sure that Virtual Community Service does not bombard users with notices about communities in which they have no real interest. On a very small network a human could be "in the loop", scanning proposed virtual communities and perhaps even giving them names. But on larger networks Virtual Community Service has to run in fully automatic mode, since it is likely to find a large number of virtual communities.

Delivering Messages to a Virtual Community

Once a virtual community has been identified, it is straightforward for Virtual Community Service to establish a mailing list so that any member of the virtual community may distribute e-mail to all other members. Another method of distribution is to use a conventional network bulletin board or newsgroup to distribute the messages to all servers in the network, where they can be accessed by any member of the virtual community. However, these simple methods do not take into account cost and performance advantages which accrue from optimizing the construction of a multicast tree to carry messages to the virtual community. Unlike a newsgroup, a multicast tree distributes messages to only a selected set of servers, and unlike an e-mail mailing list, it does so efficiently.

A separate multicast tree MT(V) is maintained for each virtual community V, by use of the following four procedures. 1. To construct or reconstruct this multicast tree, the core servers for virtual community V are taken to be those proxy servers that serve at least one

IPR2025-00200 / 722 Patent Page 298 of 1171 EX1002

pseudonymous member of virtual community V. Then the multicast tree MT(V) is established via steps 4 -6 in the section "Multicast Tree Construction Procedure" above. 2. When a new user joins virtual community V, which is an existing virtual community, the user sends a message to the user's proxy server S. If user's proxy server S is not already a core server for V, then it is designated as a core server and is added to the multicast tree MT(V), as follows. If more than k servers have been added since the last time the multicast tree MT(V) was rebuilt, where k is a function of the number of core servers already in the tree, then the entire tree is simply rebuilt via steps 4-6 in the section "Multicast Tree Construction Procedure" above. Otherwise, server S retrieves its locally stored list of nearby core servers for V, and chooses a server S1. Server S sends a control message to S1, indicating that it would like to be added to the multicast tree MT(V). Upon receipt of this message, server S1 retrieves its locally stored subtree G1 of MT(V), and forms a new graph G from G1 by removing all degree-1 vertices other than S1 itself. Server S1 transmits graph G t o server S, which stores it as its locally stored subtree of MT(V). Finally, server S sends a message to itself and to all servers that are vertices of graph G, instructing these servers to modify their locally stored subtrees of MT(V) by adding S as a vertex and adding an edge between S1 and S. 3. When a user at a client q wishes to send a message F to virtual community V, client q embeds message F in a request R instructing the recipient to store message F locally, for a limited time, for access by member s of virtual community V. Request R includes a credential proving that the user is a member of virtual community V or is otherwise entitled to post messages to virtual community V (for example is not "black marked" by that or other virtual community members). Client q then broadcasts request R to all core servers in the multicast tree MT(V), by means of a global request message transmitted to the user's proxy server as described above. The core servers satisfy request R, provided that they can verify the included credential. 4. In order to retrieve a particular message sent to virtual community V, a user U at client q initiates the steps described in section "Retrieving Files from a Multicast Tree," above. If user U does not want to retrieve a particular message, but rather wants to retrieve all new messages sent to virtual community V, then user U pseudonymously instructs its proxy server (which is a core server for V) to send it all messages that were multicast to MT(V) after a certain date. In either case, user U must provide a credential proving user U to be a member of virtual community V, or otherwise entitled to access messages on virtual community V.

IPR2025-00200 / 722 Patent Page 299 of 1171 EX1002

SUMMARY

A method has been presented for automatically selecting articles of interest to a user. The method generates sets of search profiles for the users based on such attributes as the relative frequency of occurrence of words in the articles read by the users, and uses these search profiles to efficiently identify future articles of interest. The methods is characterized by passive monitoring (users do not need to explicitly rate the articles), multiple search profiles per user (reflecting interest in multiple topics) and use of elements of the search profiles which are automatically determined from the data (notably, the TF/IDF measure based on word frequencies and descriptions of purchasable items). A method has also been presented for automatically generating menus to allow users to locate and retrieve articles on topics of interest. This method clusters articles based on their similarity, as measured by the relative frequency of word occurrences. Clusters are labeled either with article titles or with key words extracted from the article. The method can be applied to large sets of articles distributed over many machines.

It has been further shown how to extend the above methods from articles to any class of target objects for which profiles can be generated, including news articles, reference or work articles, electronic mail, product or service descriptions, people (based on the articles they read, demographic data, or the products they buy), and electronic bulletin boards (based on the articles posted to them). A particular consequence of being able to group people by their interests is that one can form virtual communities of people of common interest, who can then correspond with one another via electronic mail.

WE CLAIM:

1. A method for providing a user with access to selected ones of a plurality of target objects and sets of target object characteristics that are accessible via an electronic storage media, where said users are connected via user terminals and data communication connections to a target server system which includes said electronic storage media, said method comprising the steps of:

automatically generating target profiles for target objects and sets of target object characteristics stored in said electronic storage media, each of said target profiles being generated from the contents of an associated one of said target objects and sets of target object characteristics;

automatically generating at least one user target profile interest summary for a user at a user terminal, each said user target profile interest summary being generated from ones of said target objects and sets of target object characteristics accessed by said user; and

enabling access to said plurality of target objects and sets of target object characteristics stored on said electronic storage media by users via said target profiles.

2. The method of claim 1 wherein said step of enabling access comprises:

correlating said user target profile interest summaries, generated for an identified user, with said generated target profiles to identify ones of said plurality of target objects and sets of target object characteristics stored on said electronic storage media that are likely to be of interest to said identified user.

3. The method of claim 2 wherein said step of enabling access further comprises: transmitting a list, that identifies at least one of said identified ones of said plurality of target objects and sets of target object characteristics, to said identified user; and

providing access to a selected one of said plurality of target objects and sets of target object characteristics stored on said electronic storage media in response to said identified user selecting an item from said list.

4. The method of claim 3 wherein said step of providing access comprises:

transmitting data, in response to said identified user activating a one of said user terminals to identify said selected item on said list, indicative of said identified user's selection of said selected item from said one user terminal to said target server via a one of said data communication connections.

5. The method of claim 4 wherein said step of providing access further comprises:

retrieving, in response to receipt of said data from said one user terminal, a target object identified by said selected item from said electronic storage media; and

transmitting said retrieved target object to said one user terminal for display thereon to said identified user.

6. The method of claim 2 wherein said step of enabling access further comprises: transmitting at least one of said identified ones of said plurality of target objects and sets of target object characteristics, to said identified user in advance of said user requesting said at least one of said identified ones of said plurality of target objects and sets of target object characteristics.

7. The method of claim 2 wherein said step of enabling access further comprises: transmitting a list, that identifies at least one of said identified ones of said plurality of target objects and sets of target object characteristics, to said identified user; and

transmitting said identified ones of said plurality of target objects and sets of target object characteristics stored on said electronic storage media from said target server system to a designated server located closer via said electronic communications connections to said user terminal than said target server system.

8. The method of claim 7 wherein said step of providing access comprises:

transmitting data, in response to said identified user activating a one of said user terminals to identify said selected item on said list, indicative of said identified

IPR2025-00200 / 722 Patent Page 302 of 1171 EX1002

user's selection of said selected item from said one user terminal to said designated server via a one of said data communication connections.

9. The method of claim 8 wherein said step of providing access further comprises:

retrieving, in response to receipt of said data from said one user terminal, a target object identified by said selected item from said designated server; and

transmitting said retrieved target object to said one user terminal for display thereon to said identified user.

10. The method of claim 1 wherein said target object is a document having at least one page, said step of enabling access comprises:

automatically generating a user target profile interest summary for an identified user that is indicative of target objects and sets of target object characteristics retrieved by said identified user as well as the number of pages of said retrieved documents accessed by said identified user.

11. The method of claim 10 wherein said automatically generated user target profile interest summaries are also indicative of a length of time said identified user accessed said retrieved target objects and sets of target object characteristics.

12. The method of claim 1 wherein said step of automatically generating target profiles comprises:

automatically generating a hierarchical menu that directs said users to at least a subset of said plurality of target objects and sets of target object characteristics stored on said electronic media, comprising:

sorting all target objects and sets of target object characteristics in said subset into a plurality of clusters of target objects and sets of target object characteristics based on an empirical measure of similarity of content of said target objects and sets of target object characteristics, and

generating a hierarchical menu that identifies a content in common of target objects and sets of target object characteristics sorted into each of said

IPR2025-00200 / 722 Patent Page 303 of 1171 EX1002

plurality of clusters, to enable said identified user to identify ones of said plurality of target objects and sets of target object characteristics stored on said electronic storage media that are likely to be of interest to said identified user.

13. The method of claim 12 wherein said step of automatically generating a hierarchical menu further comprises:

ascribing a cluster profile to each of said plurality of clusters.

14. The method of claim 12 wherein said step of sorting comprises:

dividing said plurality of target objects and sets of target object characteristics into at least two clusters based upon said empirical measure of similarity of content of said target objects and sets of target object characteristics,

subdividing each of said at least two clusters into at least two subclusters based upon said empirical measure of similarity of content of said target objects and sets of target object characteristics, and

repeating said step of subdividing to produce a multi-level hierarchy of identified clusters.

15. The method of claim 14 wherein said step of generating a hierarchical menu comprises:

ascribing a cluster profile to each cluster produced by all steps of dividing and subdividing in said step of sorting.

16. The method of claim 15 wherein said step of ascribing comprises:

identifying at least one term in said generated target profiles produced for ones of said plurality of target objects and sets of target object characteristics sorted into a cluster that is indicative of the target content of said ones of said plurality of target objects and sets of target object characteristics sorted into said cluster.

17. The method of claim 15 wherein said step of ascribing comprises:

selecting at least one target object of said ones of said plurality of target objects and sets of target object characteristics sorted into said cluster that are closest to the center of the cluster, and

ascribing a cluster profile that is indicative of the target content of said ones of said plurality of target objects and sets of target object characteristics sorted into said cluster, said cluster profile comprising elements of at least one of: a title of said selected at least one target object and sets of target object characteristics, and a set of words contained in the target profile of said selected at least one target object cluster which have the highest relative frequency.

18. Apparatus for providing a user with access to selected ones of a plurality of target objects and sets of target object characteristics that are accessible via an electronic storage media, where said users are connected via user terminals and data communication connections to a target server system which includes said electronic storage media, comprising:

means for automatically generating target profiles for target objects and sets of target object characteristics stored in said electronic storage media, each of said target profiles being generated from the contents of an associated one of said target objects and sets of target object characteristics;

means for automatically generating at least one user target profile interest summary for a user at a user terminal, each said user target profile interest summary being generated from ones of said target objects and sets of target object characteristics accessed by said user; and

means for enabling access to said plurality of target objects and sets of target object characteristics stored on said electronic storage media by users via said target profiles.

19. The apparatus of claim 18 wherein said means for enabling access comprises:

means for correlating a user target profile interest summary, generated for an identified user, with said generated target profiles to identify ones of said plurality of target objects and sets of target object characteristics stored on said electronic storage media that are likely to be of interest to said identified user.

IPR2025-00200 / 722 Patent Page 305 of 1171 EX1002

20. The apparatus of claim 19 wherein said means for enabling access further comprises:

means for transmitting a list, that identifies at least one of said identified ones of said plurality of target objects and sets of target object characteristics, to said identified user; and

means for providing access to a selected one of said plurality of target objects and sets of target object characteristics stored on said electronic storage media in response to said identified user selecting an item from said list.

21. The apparatus of claim 20 wherein said means for providing access comprises:

means for transmitting data, in response to said identified user activating a one of said user terminals to identify said selected item on said list, indicative of said identified user's selection of said selected item from said one user terminal to said target server via a one of said data communication connections.

22. The apparatus of claim 21 wherein said means for providing access further comprises:

means for retrieving, in response to receipt of said data from said one user terminal, a target object identified by said selected item from said electronic storage media; and

means for transmitting said retrieved target object to said one user terminal for display thereon to said identified user.

23. The apparatus of claim 19 wherein said means for enabling access further comprises:

means for transmitting at least one of said identified ones of said plurality of target objects and sets of target object characteristics, to said identified user in advance of said user requesting said at least one of said identified ones of said plurality of target objects and sets of target object characteristics.

24. The apparatus of claim 19 wherein said means for enabling access further comprises:

means for transmitting a list, that identifies at least one of said identified ones of said plurality of target objects and sets of target object characteristics, to said identified user; and

means for transmitting said identified ones of said plurality of target objects and sets of target object characteristics stored on said electronic storage media from said target server system to a designated server located closer via said electronic communications connections to said user terminal than said target server system.

25. The apparatus of claim 24 wherein said means for providing access comprises:

means for transmitting data, in response to said identified user activating a one of said user terminals to identify said selected item on said list, indicative of said identified user's selection of said selected item from said one user terminal to said designated server via a one of said data communication connections.

26. The apparatus of claim 25 wherein said means for providing access further comprises:

means for retrieving, in response to receipt of said data from said one user terminal, a target object identified by said selected item from said designated server; and

means for transmitting said retrieved target object to said one user terminal for display thereon to said identified user.

27. The apparatus of claim 20 wherein said target object is a document having at least one page, said means for enabling access comprises:

means for automatically generating a user target profile interest summary for an identified user that is indicative of target objects and sets of target object characteristics retrieved by said identified user as well as the number of pages of said retrieved documents accessed by said identified user.

IPR2025-00200 / 722 Patent Page 307 of 1171 EX1002

28. The apparatus of claim 27 wherein said automatically generated user target profile interest summary is also indicative of a length of time said identified user accessed said retrieved target objects and sets of target object characteristics.

29. The apparatus of claim 22 wherein said means for automatically generating target profiles comprises:

means for automatically generating a hierarchical menu that directs said users to at least a subset of said plurality of target objects and sets of target object characteristics stored on said electronic media, comprising:

means for sorting all target objects and sets of target object characteristics in said subset into a plurality of clusters of target objects and sets of target object characteristics based on an empirical measure of similarity of content of said target objects and sets of target object characteristics, and

means for generating a hierarchical menu that identifies the content in common of target objects and sets of target object characteristics sorted into each of said plurality of clusters, to enable said identified user to identify ones of said plurality of target objects and sets of target object characteristics stored on said electronic storage media that are likely to be of interest to said identified user.

30. The apparatus of claim 29 wherein said means for automatically generating a hierarchical menu further comprises:

means for ascribing a cluster profile to each of said plurality of clusters.

31. The apparatus of claim 29 wherein said means for sorting comprises:

means for dividing said plurality of target objects and sets of target object characteristics into at least two clusters based upon said empirical measure of similarity of content of said target objects and sets of target object characteristics,

means for subdividing each of said at least two clusters into at least two subclusters based upon said empirical measure of similarity of content of said target objects and sets of target object characteristics, and

means for repeating said step of subdividing to produce a multi-level hierarchy of identified clusters.

32. The apparatus of claim 31 wherein said means for generating a hierarchical menu comprises:

means for ascribing a cluster profile to each cluster produced by all said means for dividing and subdividing.

33. The apparatus of claim 32 wherein said means for ascribing comprises:

means for identifying at least one term in said generated target profiles produced for ones of said plurality of target objects and sets of target object characteristics sorted into a cluster that is indicative of the target content of said ones of said plurality of target objects and sets of target object characteristics sorted into said cluster.

34. The apparatus of claim 32 wherein said means for ascribing comprises:

means for selecting at least one target object of said ones of said plurality of target objects and sets of target object characteristics sorted into said cluster that are closest to the center of the cluster; and

means for ascribing a cluster profile that is indicative of the target content of said ones of said plurality of target objects and sets of target object characteristics sorted into said cluster, said cluster profile comprising elements of at least one of: a title of said selected at least one target object, and a set of words contained in the target profile of said selected at least one target object cluster which have the highest relative frequency.

IPR2025-00200 / 722 Patent Page 309 of 1171 EX1002

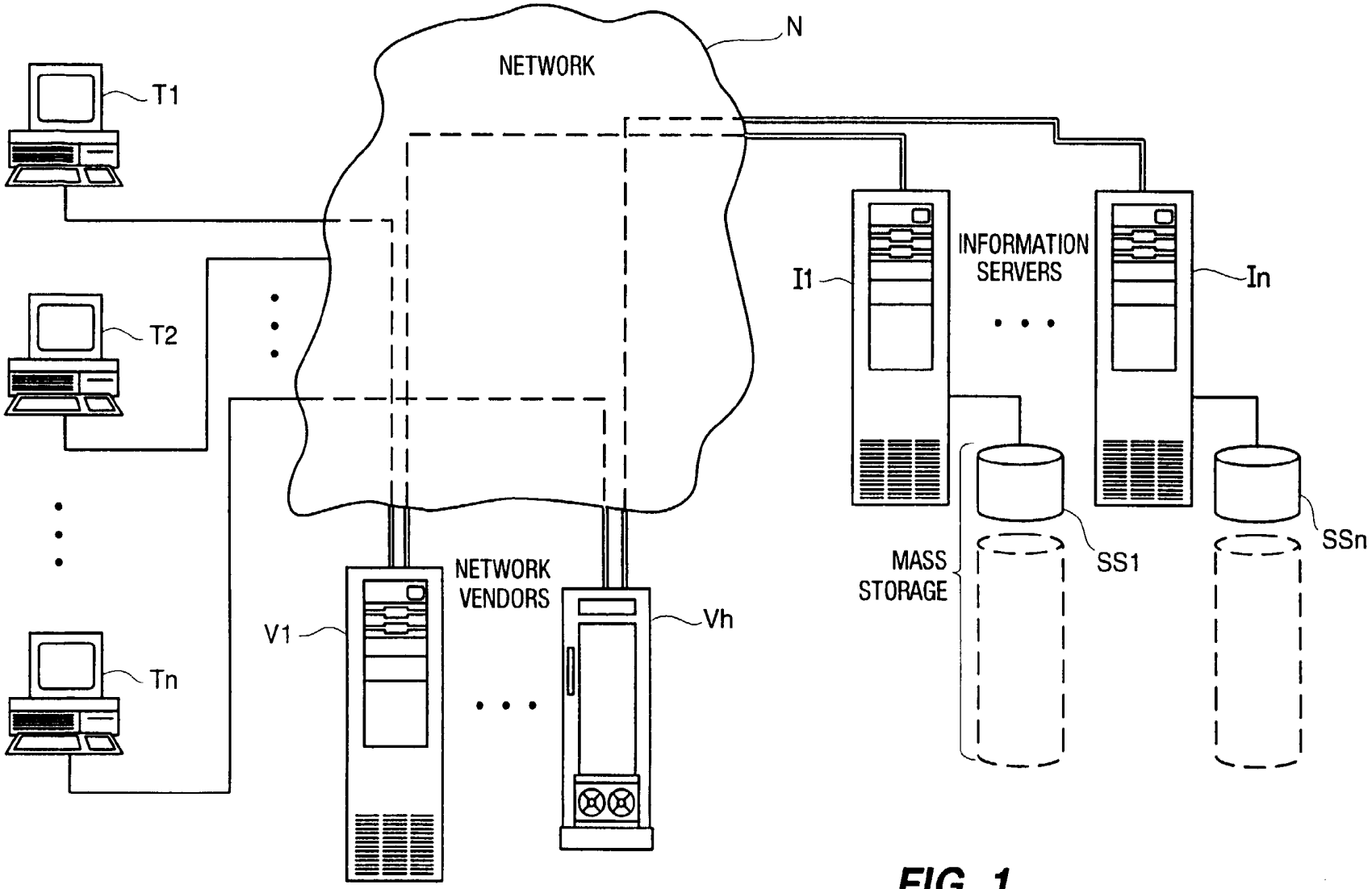


FIG. 1
PRIOR ART

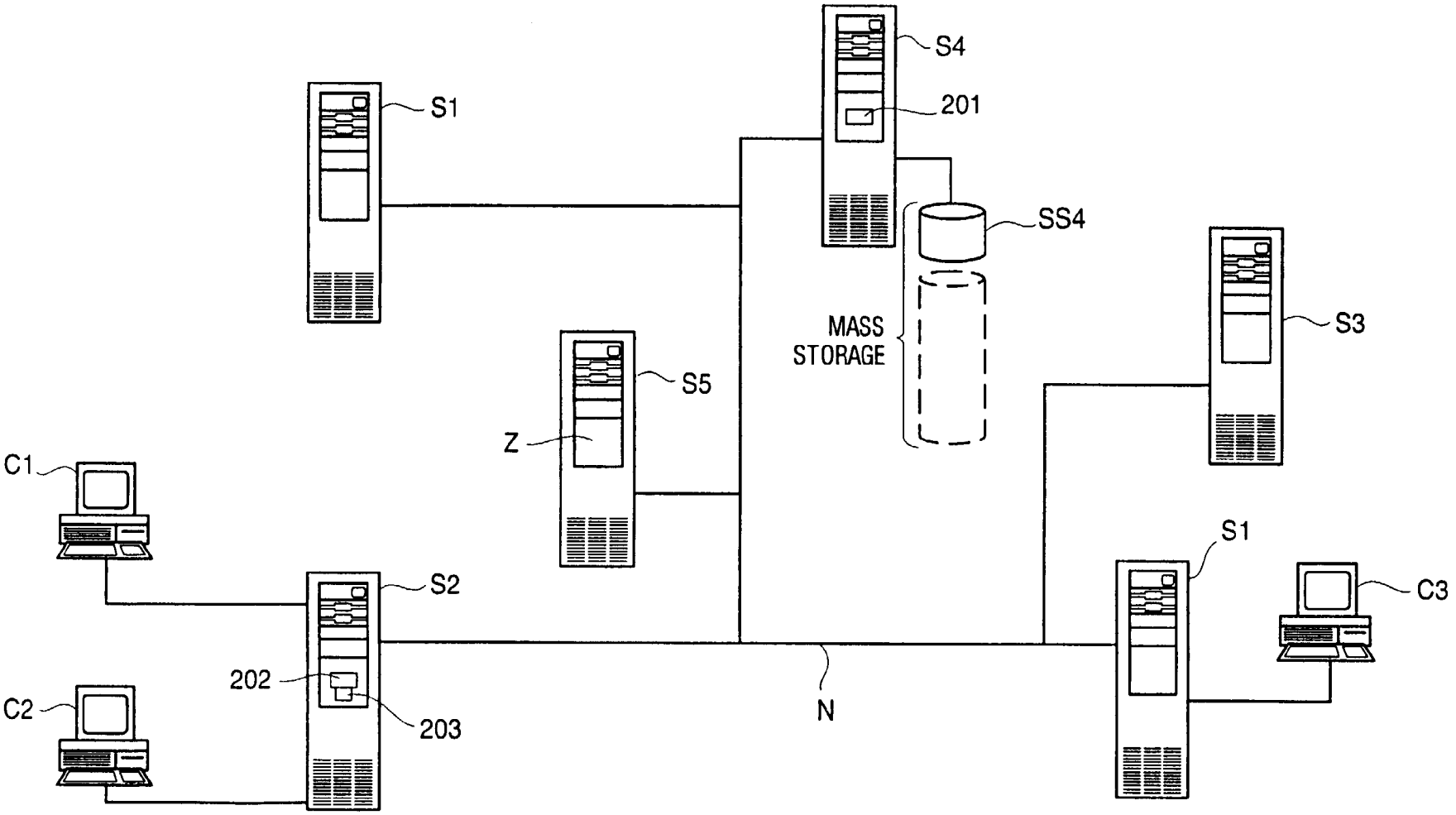


FIG. 2

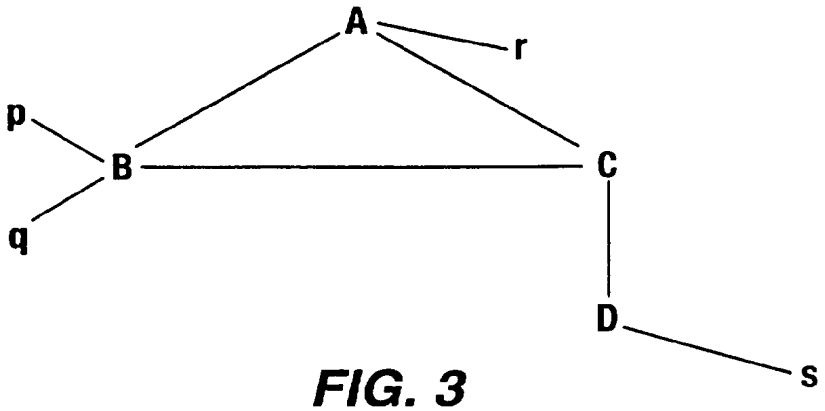


FIG. 3

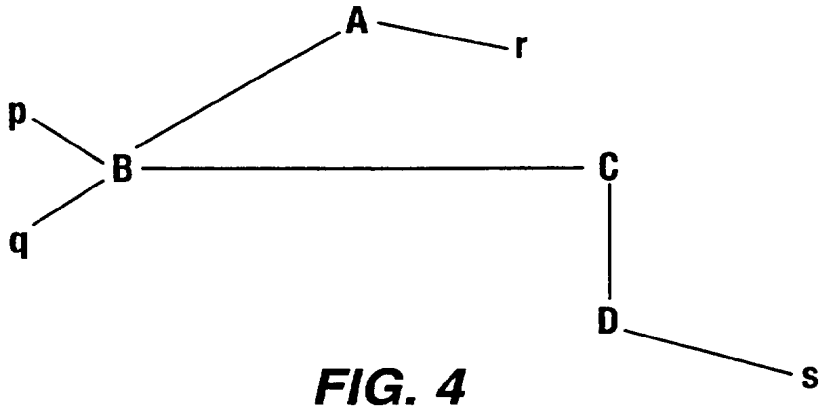


FIG. 4

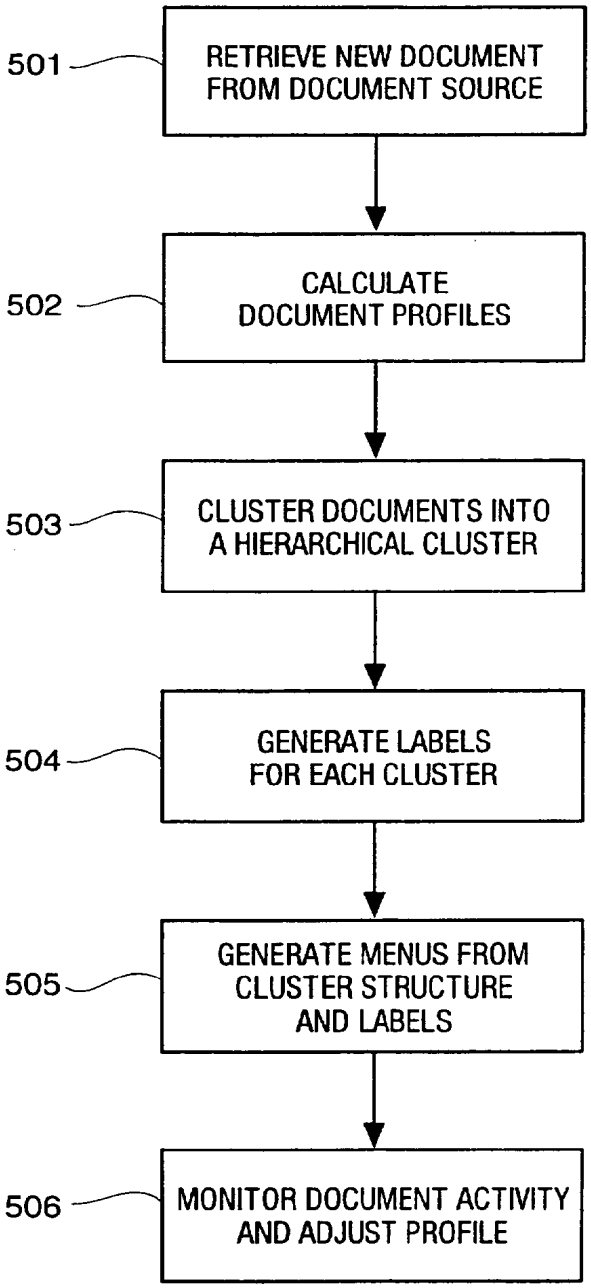
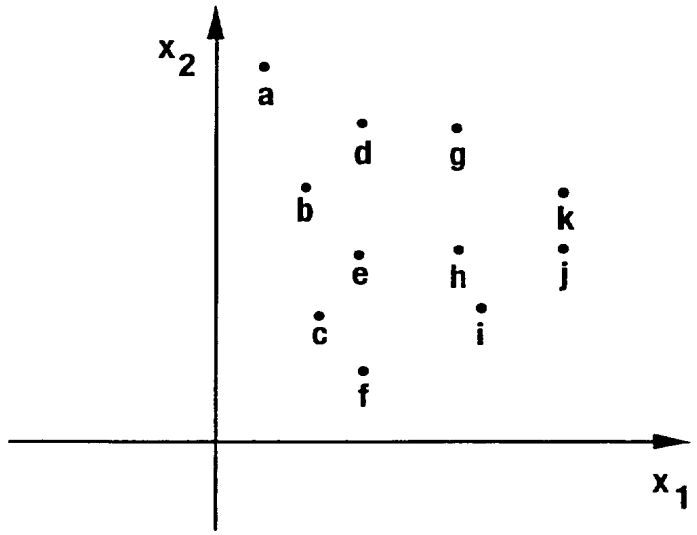

501
RETRIEVE NEW DOCUMENT FROM DOCUMENT SOURCE
502
CALCULATE DOCUMENT PROFILES
503
CLUSTER DOCUMENTS INTO A HIERARCHICAL CLUSTER
504
GENERATE LABELS FOR EACH CLUSTER
505
GENERATE MENUS FROM CLUSTER STRUCTURE AND LABELS
506
MONITOR DOCUMENT ACTIVITY AND ADJUST PROFILE

FIG. 5

FIG. 6


x2
x1
a
b
c
d
e
f
g
h
i
j
k

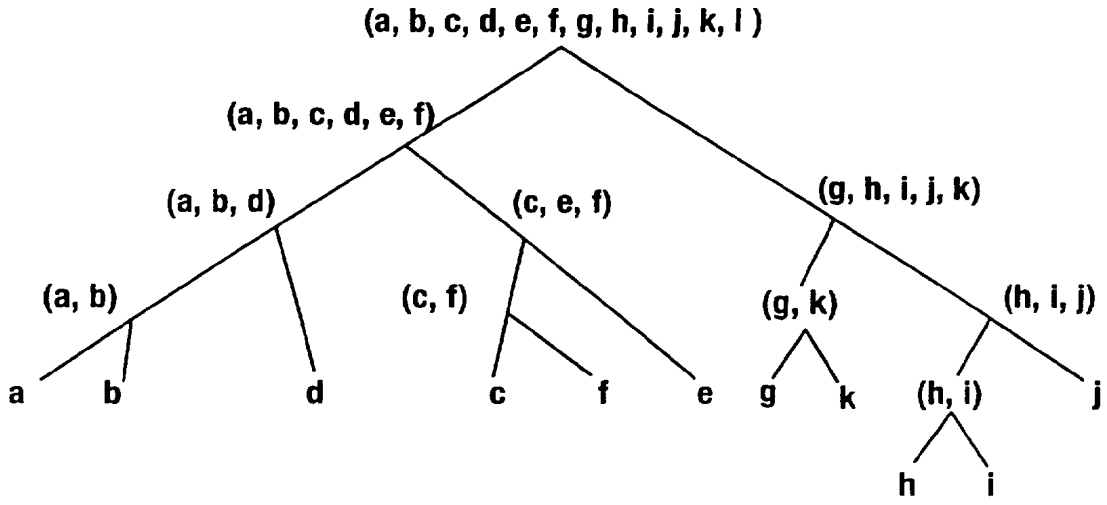

(a, b, c, d, e, f, g, h, i, j, k, l)
(a, b, c, d, e, f)
(a, b, d)
(c, e, f)
(g, h, i, j, k)
(a, b)
(c, f)
(g, k)
(h, i, j)
(h, i)
a
b
d
c
f
e
g
k
h
i
j

FIG. 7

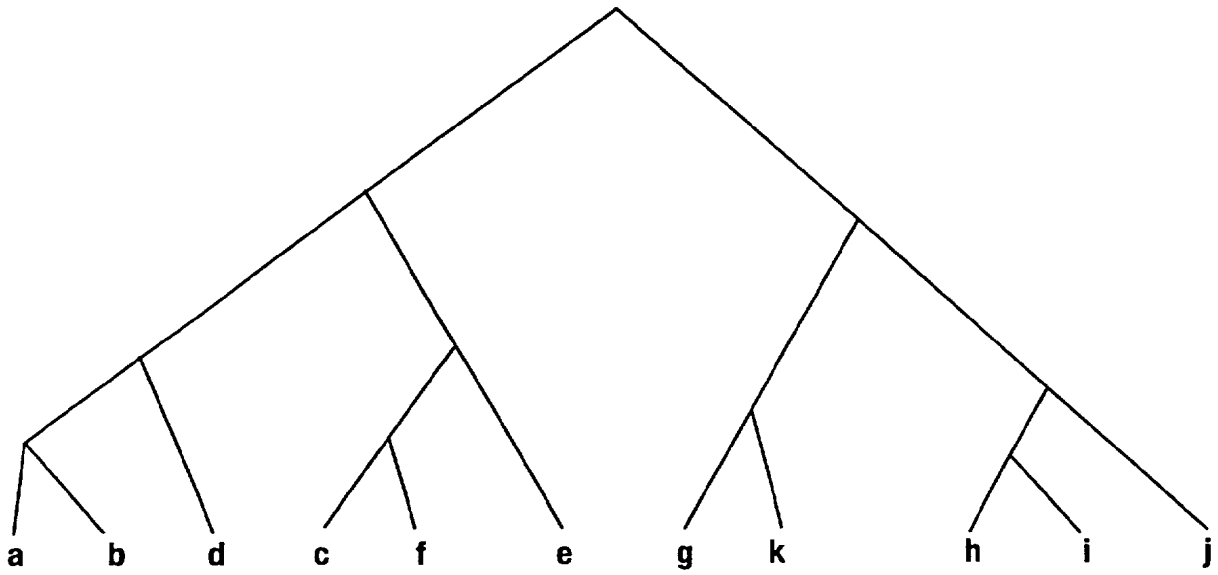


FIG. 8

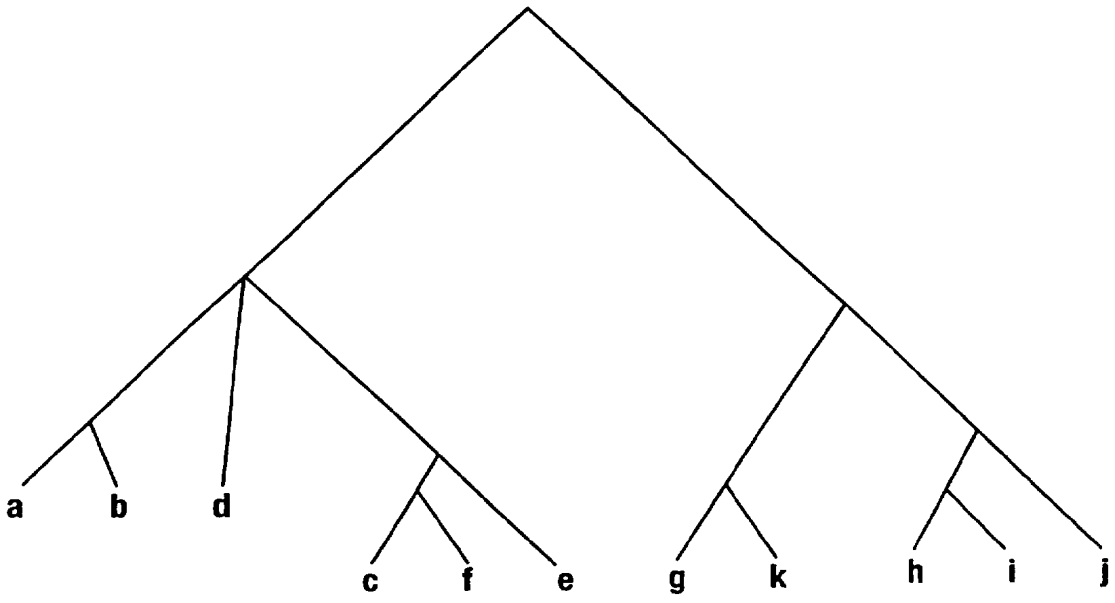


FIG. 9

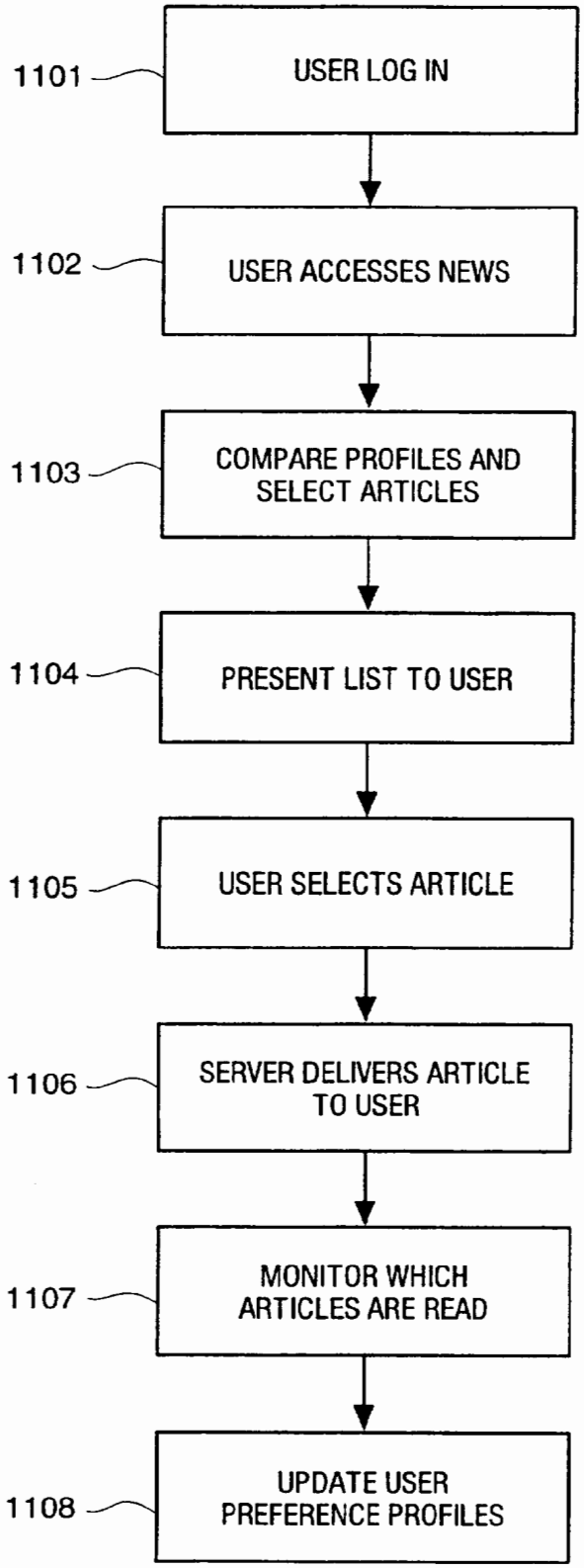

1101
USER LOG IN
1102
USER ACCESSES NEWS
1103
COMPARE PROFILES AND SELECT ARTICLES
1104
PRESENT LIST TO USER
1105
USER SELECTS ARTICLE
1106
SERVER DELIVERS ARTICLE TO USER
1107
MONITOR WHICH ARTICLES ARE READ
1108
UPDATE USER PREFERENCE PROFILES

FIG. 10

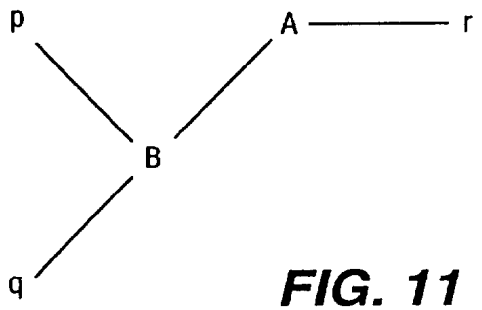

p
A
r
B
q

FIG. 11

FIG. 12

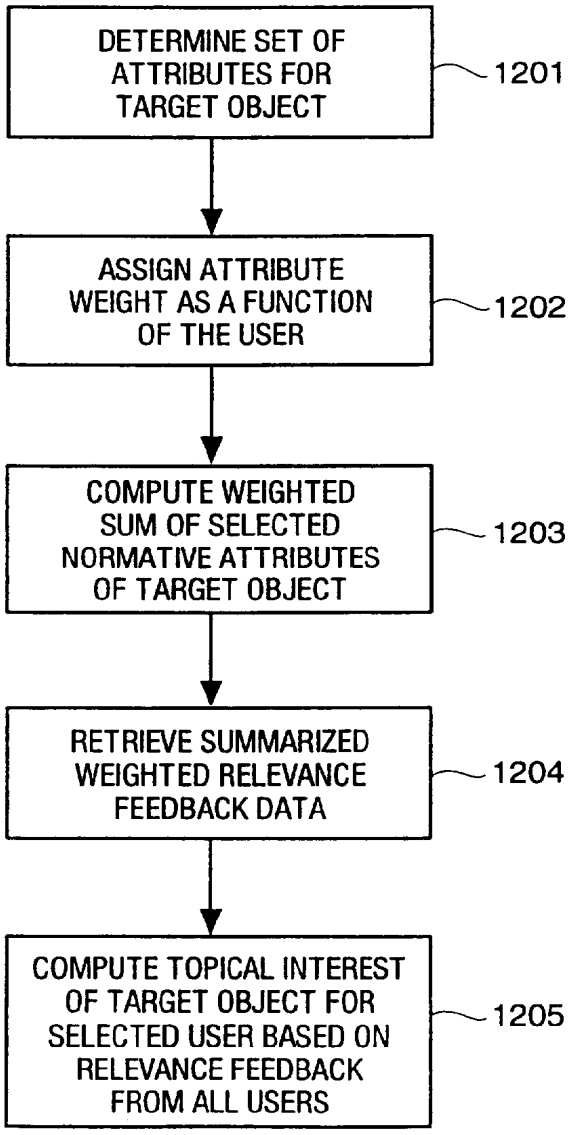

DETERMINE SET OF ATTRIBUTES FOR TARGET OBJECT
1201
ASSIGN ATTRIBUTE WEIGHT AS A FUNCTION OF THE USER
1202
COMPUTE WEIGHTED SUM OF SELECTED NORMATIVE ATTRIBUTES OF TARGET OBJECT
1203
RETRIEVE SUMMARIZED WEIGHTED RELEVANCE FEEDBACK DATA
1204
COMPUTE TOPICAL INTEREST OF TARGET OBJECT FOR SELECTED USER BASED ON RELEVANCE FEEDBACK FROM ALL USERS
1205

FIG. 13A

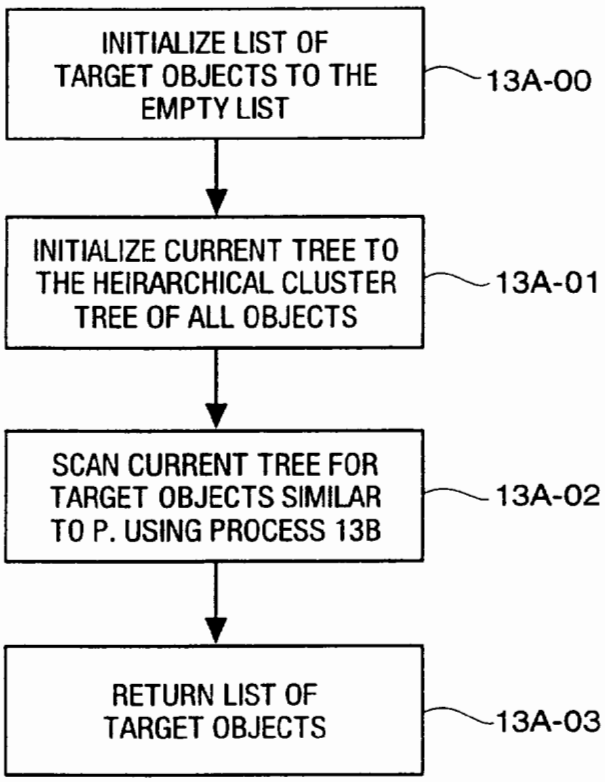

INITIALIZE LIST OF TARGET OBJECTS TO THE EMPTY LIST
13A-00
INITIALIZE CURRENT TREE TO THE HEIRARCHICAL CLUSTER TREE OF ALL OBJECTS
13A-01
SCAN CURRENT TREE FOR TARGET OBJECTS SIMILAR TO P. USING PROCESS 13B
13A-02
RETURN LIST OF TARGET OBJECTS
13A-03

FIG. 13B

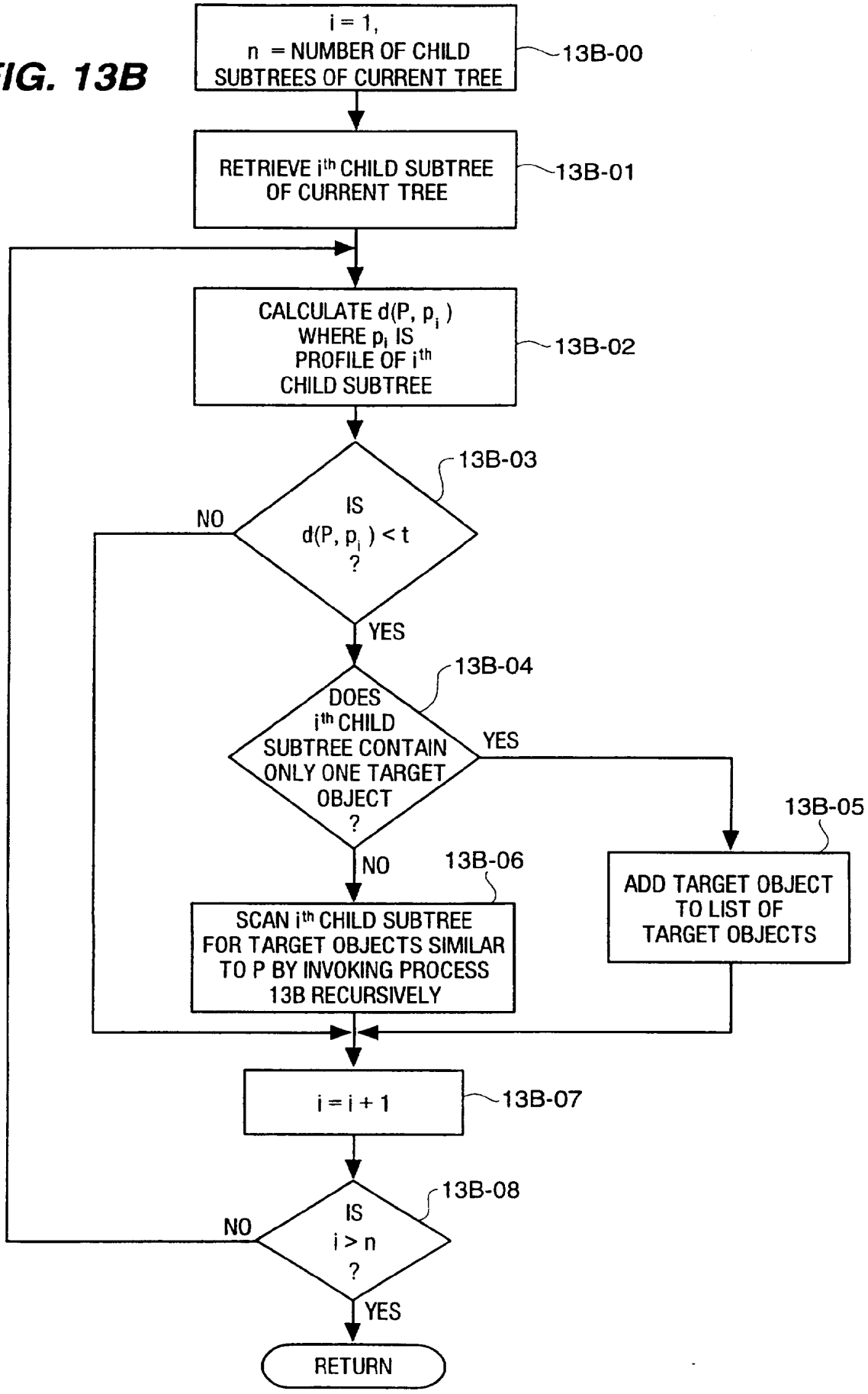

i = 1,
n = NUMBER OF CHILD SUBTREES OF CURRENT TREE
13B-00
RETRIEVE i^th CHILD SUBTREE OF CURRENT TREE
13B-01
CALCULATE d(P, p_i) WHERE p_i IS PROFILE OF ith CHILD SUBTREE
13B-02
IS d(P, p_i) < t ?
13B-03
NO
YES
DOES i^th CHILD SUBTREE CONTAIN ONLY ONE TARGET OBJECT ?
13B-04
YES
NO
ADD TARGET OBJECT TO LIST OF TARGET OBJECTS
13B-05
SCAN i^th CHILD SUBTREE FOR TARGET OBJECTS SIMILAR TO P BY INVOKING PROCESS 13B RECURSIVELY
13B-06
i = i + 1
13B-07
IS i > n ?
13B-08
NO
YES
RETURN

FIG. 14

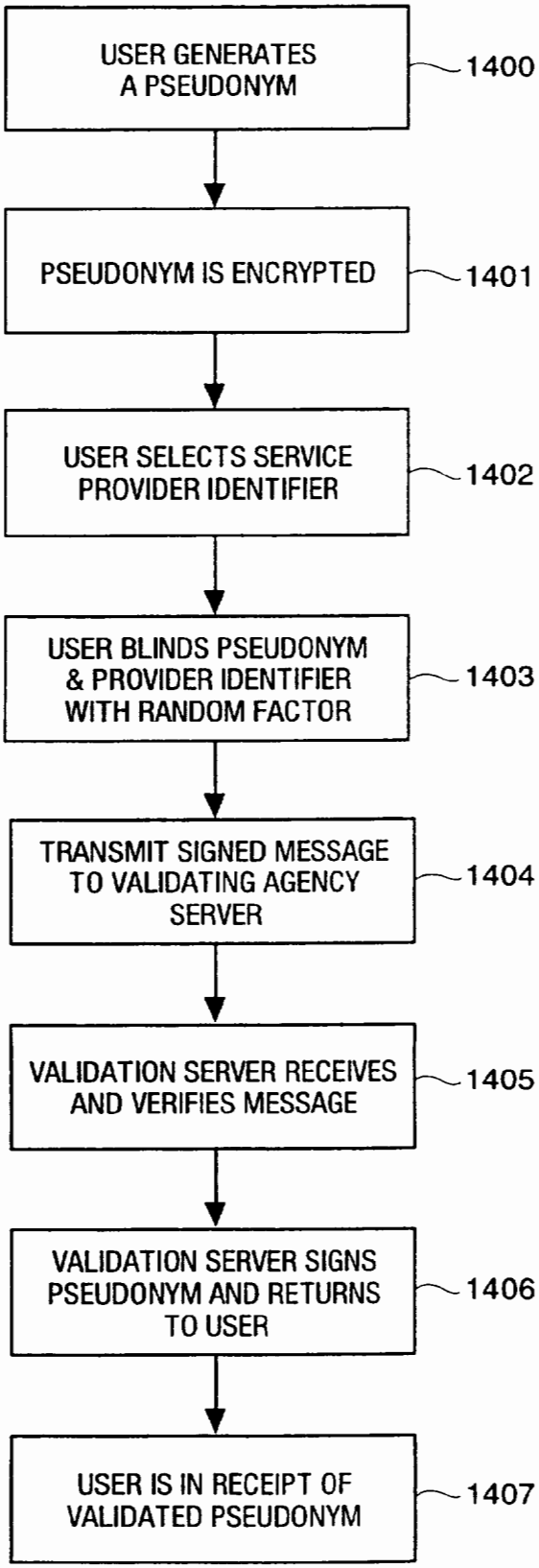

USER GENERATES A PSEUDONYM
1400
PSEUDONYM IS ENCRYPTED
1401
USER SELECTS SERVICE PROVIDER IDENTIFIER
1402
USER BLINDS PSEUDONYM & PROVIDER IDENTIFIER WITH RANDOM FACTOR
1403
TRANSMIT SIGNED MESSAGE TO VALIDATING AGENCY SERVER
1404
VALIDATION SERVER RECEIVES AND VERIFIES MESSAGE
1405
VALIDATION SERVER SIGNS PSEUDONYM AND RETURNS TO USER
1406
USER IS IN RECEIPT OF VALIDATED PSEUDONYM
1407

FIG. 15

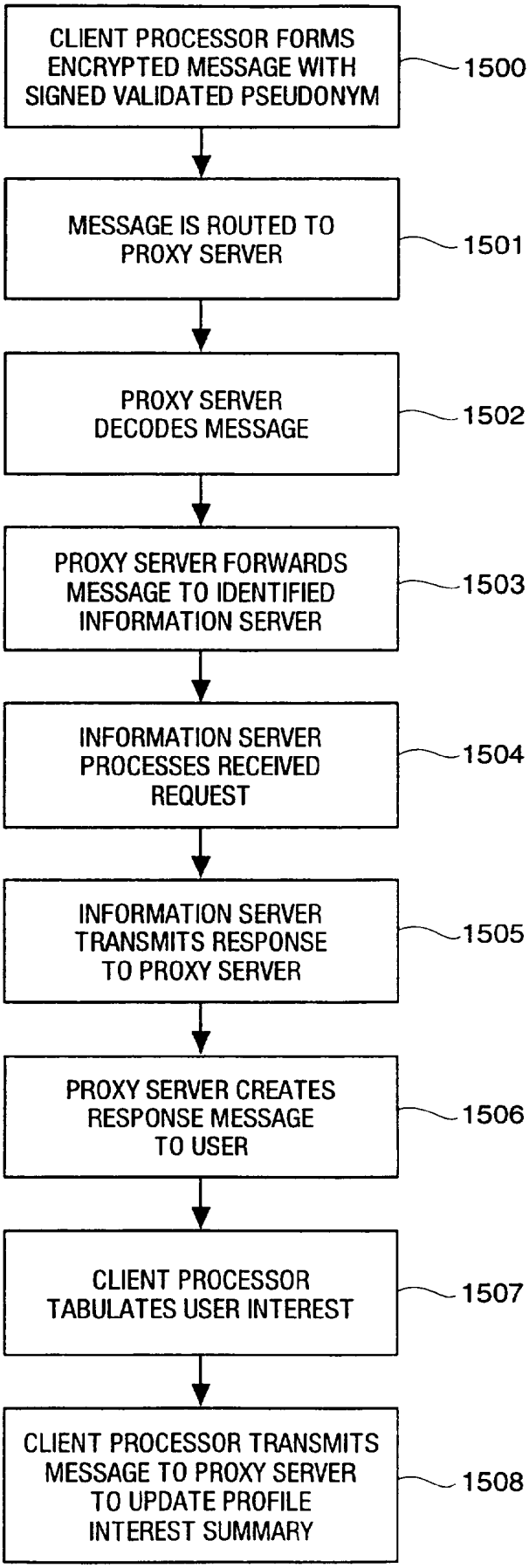
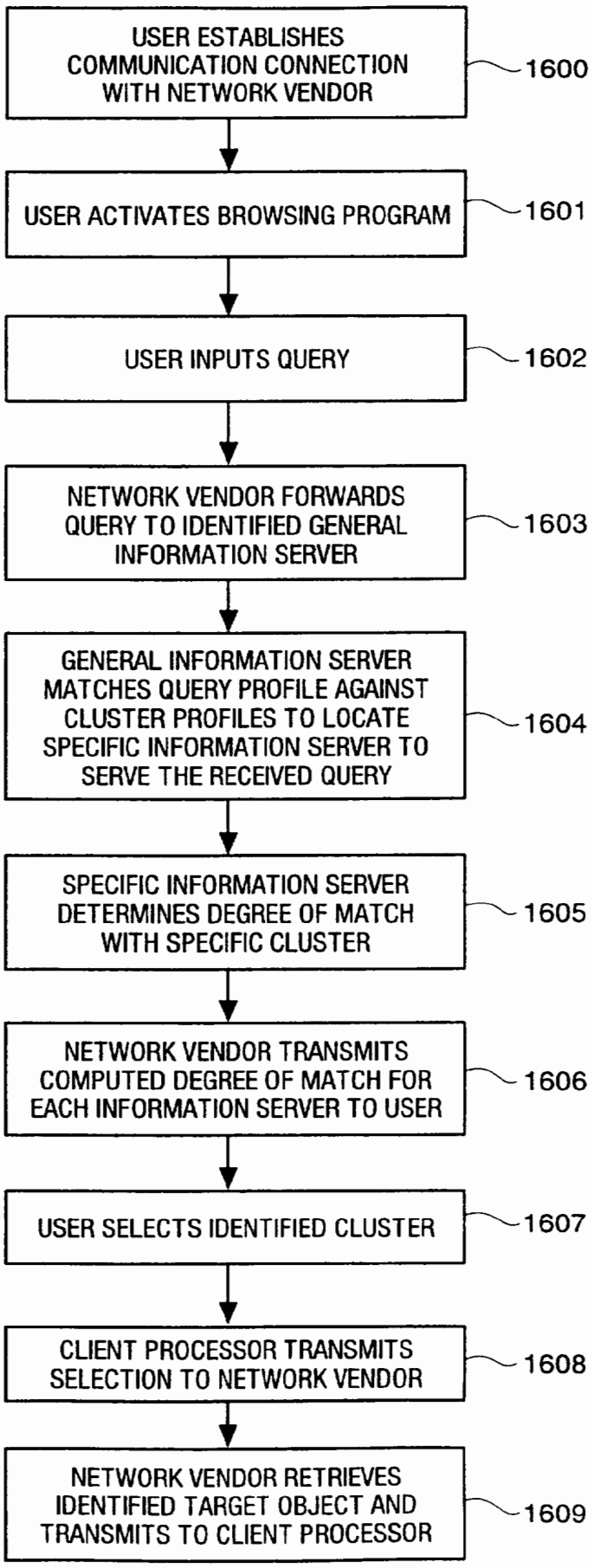

CLIENT PROCESSOR FORMS ENCRYPTED MESSAGE WITH SIGNED VALIDATED PSEUDONYM
1500
MESSAGE IS ROUTED TO PROXY SERVER
1501
PROXY SERVER DECODES MESSAGE
1502
PROXY SERVER FORWARDS MESSAGE TO IDENTIFIED INFORMATION SERVER
1503
INFORMATION SERVER PROCESSES RECEIVED REQUEST
1504
INFORMATION SERVER TRANSMITS RESPONSE TO PROXY SERVER
1505
PROXY SERVER CREATES RESPONSE MESSAGE TO USER
1506
CLIENT PROCESSOR TABULATES USER INTEREST
1507
CLIENT PROCESSOR TRANSMITS MESSAGE TO PROXY SERVER TO UPDATE PROFILE INTEREST SUMMARY
1508

FIG. 16


USER ESTABLISHES COMMUNICATION CONNECTION WITH NETWORK VENDOR
1600
USER ACTIVATES BROWSING PROGRAM
1601
USER INPUTS QUERY
1602
NETWORK VENDOR FORWARDS QUERY TO IDENTIFIED GENERAL INFORMATION SERVER
1603
GENERAL INFORMATION SERVER MATCHES QUERY PROFILE AGAINST CLUSTER PROFILES TO LOCATE SPECIFIC INFORMATION SERVER TO SERVE THE RECEIVED QUERY
1604
SPECIFIC INFORMATION SERVER DETERMINES DEGREE OF MATCH WITH SPECIFIC CLUSTER
1605
NETWORK VENDOR TRANSMITS COMPUTED DEGREE OF MATCH FOR EACH INFORMATION SERVER TO USER
1606
USER SELECTS IDENTIFIED CLUSTER
1607
CLIENT PROCESSOR TRANSMITS SELECTION TO NETWORK VENDOR
1608
NETWORK VENDOR RETRIEVES IDENTIFIED TARGET OBJECT AND TRANSMITS TO CLIENT PROCESSOR
1609

INTERNATIONAL SEARCH REPORT

Inter nal Application No
PCT/US 96/17981

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 G06F17/30

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	SIGIR '94, DUBLIN, JULY 3 - 6, 1994, no. CONF. 17, 3 July 1994, CROFT W B;VAN RIJSBERGEN C J (EDITORS), pages 272-281, XP000475327 MORITA M ET AL: "INFORMATION FILTERING BASED ON USER BEHAVIOR ANALYSIS AND BEST MATCH TEXT RETRIEVAL" see abstract see page 272, line 1 - page 275, line 1 --- -/--	1,2,12, 18,19

[X] Further documents are listed in the continuation of box C.

[ ] Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

1

Date of the actual completion of the international search	Date of mailing of the international search report
5 February 1997	14. 02 97

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax (+31-70) 340-3016	Katerbau, R

Form PCT/ISA/210 (second sheet) (July 1992)

IPR2025-00200 / 722 Patent Page 323 of 1171 EX1002

INTERNATIONAL SEARCH REPORT

Inter ial Application No
PCT/US 96/17981

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	PROCEEDINGS OF THE 1995 USENIX TECHNICAL CONFERENCE, PROCEEDINGS USENIX WINTER 1995 TECHNICAL CONFERENCE, NEW ORLEANS, LA, USA, 16-20 JAN. 1995, 1995, BERKELEY, CA, USA, USENIX ASSOC, USA, pages 177-186, XP000617276 YAN T W ET AL: "SIFT-a tool for wide-area information dissemination" see abstract see page 177, column 1, line 1 - page 179, column 2, line 1 ---	1,2,12, 18,19
A	JOURNAL OF INTELLIGENT INFORMATION SYSTEMS: INTEGRATING ARTIFICIAL INTELLIGENCE AND DATABASE TECHNOLOGIES, SEPT. 1995, NETHERLANDS, vol. 5, no. 2, ISSN 0925-9902, pages 81-99, XP000617268 BINKLEY J ET AL: "Rama: an architecture for Internet information filtering" see abstract -----	1,2,12, 18,19

1

Form PCT/ISA/210 (continuation of second sheet) (July 1992)

IPR2025-00200 / 722 Patent Page 324 of 1171 EX1002

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
30 August 2001 (30.08.2001)

PCT

(10) International Publication Number
WO 01/63837 A2

(51) International Patent Classification[7]: H04L 12/00

(21) International Application Number: PCT/US01/04981

(22) International Filing Date: 15 February 2001 (15.02.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/185,202 25 February 2000 (25.02.2000) US
09/519,221 6 March 2000 (06.03.2000) US

(71) Applicant: NAVIC SYSTEMS, INC. [US/US]; 74 Crescent Road, Needham, MA 02494 (US).

(72) Inventors: KANOJIA, Chaitanya; 69 Homestead Street, Waban, MA 02468 (US). KAMENSTKY, Lee; 32 Dodge Street, Arlington, MA 02474 (US). HALL, Peter; 42 Read Road, Ashtead, Surry KT21 2HS (US).

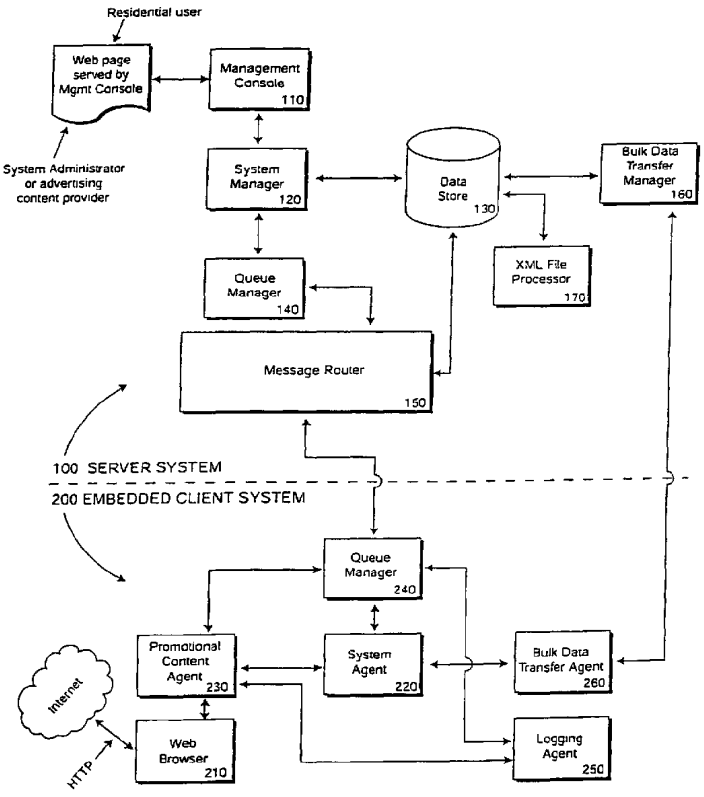
(74) Agents: THIBODEAU, David, J., Jr.__ et al.; Hamilton, Brook, Smith & Reynolds, P.C., Two Militia Drive, Lexington, MA 02421 (US).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE,

[Continued on next page]

(54) Title: MESSAGE ROUTER



(57) Abstract: The present disclosed system is directed toward a communication and management system that dynamically targets network devices for content deployment, such as application programs, device drivers, configuration files, and registry subhives. Moreover, the present system targets users of network devices for promotions, such as advertisements offered by Internet e-commerce sites. Promotions are generally icons or graphic images with links to host web servers overlaying a video display, but also includes audio and video clips or data streams. Network devices and their users are targeted through user profiles. User profiles are created when network devices register with the system server and are continually updated with information provided by user activity and event logs that are periodically uploaded from each device. A scalable messaging system provides for data transmission between the system server and among the network devices such that it is neutral as to the specific hardware platforms on which it is implemented.

WO 01/63837 A2

IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:

— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

MESSAGE ROUTER

BACKGROUND OF THE INVENTION

Generally, under the current state of technology and in the past, television has been delivered to the residential home either through radio-frequency broadcasts, satellite downlink, or over coaxial cable television (CATV) network. Data network communications, such as Internet access, have been delivered via the telephone networks through dial-up connections, ISDN (Integrated Services Digital Network), and DSL (Digital Subscriber Line) lines or over hybrid broadcast/data CATV networks, where a portion of the bandwidth transmitted by the coaxial cable is allocated for shared data network functionality using a CSMA/CD-style transmission protocol. Less commonly, data connections to the home are provided via satellite links where data are downloaded via the satellite link and uploads are handled through land lines, such as the telephone network. Another technique is to transmit data to the home via wireless, CAMA, for example, links.

Almost universally, the clients or network devices in the residences are personal computers. Typically, they execute application programs such as email clients and browsers that utilize the data network connectivity offered by one of the above techniques.

The trend, however, is towards a more ubiquitous computing model where the network devices in the home will be embedded systems designed for a particular function or purpose. This has already occurred to some degree. Today, for example, CATV network set-top boxes typically have limited data communication capabilities. Their main function is to handle channel access issues between residential users and a server on the cable TV network.

In the future, the functionality offered by these set-top boxes or other embedded platforms, such as a game system, will be expanded. For example, they may offer Internet browsing capabilities and e-commerce serving capabilities. Moreover, it is anticipated that common-household appliances will also have network functionality, in which they will be attached to the network to automate various tasks.

The data networks must evolve with deployment of these embedded systems. Where the personal computer can be updated with new network drivers as the network evolves, embedded client systems remain relatively static. Moreover, the process of installation in the residence must be made less complicated so that a network technician is not required every time a new embedded device is connected onto the network.

SUMMARY OF THE INVENTION

One issue that arises in the context of these data networks is a messaging system that is adaptable and can monitor the status of the embedded devices to enable effective communication between the server system and the devices. This issue is particularly critical where communication with and among the embedded devices is sporadic due to the mobile/transient nature of these network devices.

The present disclosed system is directed toward a communication and management system that dynamically targets network devices for content deployment, such as application programs, device drivers, configuration files, and registry subhives.

Moreover, the present system targets users of network devices for promotions, such as advertisements offered by Internet e-commerce sites. Promotions are generally icons or graphic images with links to host web servers overlaying a video display, but also includes audio and video clips or data streams.

Network devices and their users are targeted through user profiles. User profiles are created when network devices register with the system server and are continually updated with information provided by user activity and event logs that are periodically uploaded from each device.

The present invention implements a scalable messaging system for data transmission between the system server and among the network devices such that it is neutral as to the specific hardware platforms on which it is implemented.

According to the present invention, a message router system is provided that transfers messages to the embedded devices over the data network when those embedding devices are accepting messages. A message store, however, is also provided that can temporarily store the messages when the destination embedded device is unavailable. The message router can handle delivery at a later time when the network device is active and able to receive messages (*i.e.* the network device is powered on).

In general, according to one aspect, the invention features a message router system for a server system that communicates with embedded devices over a data network. The router system comprises a router that transfers messages to the embedded devices on the data network when the embedded devices are accepting messages. A message store is also provided that temporarily stores the messages when the embedded devices are not active and stores them until the message devices can accept the messages.

In specific embodiments, the system comprises a system manager, executing on the server system, that tracks that states of the embedded devices on the data network and whether the embedded devices are able to receive messages. A queue manager can also be provided for facilitating the transfer of messages between the message router and the system manager.

According to other aspects of the preferred embodiments, the router retrieves messages from the message store when the system manager indicates that the embedded device, to which the messages are addressed, are able to receive the messages.

Moreover, to handle the transfer of large files, a bulk data transfer manager is provided on the server system with a corresponding bulk data transfer agent on the network device. Typically, the bulk data transfer agent requests a file at a specific location, which the bulk transfer manager then sends. This interchange happens without any intervention from the router.

In general, according to another aspect, the invention also features a method for routing messages from a server system to embedded devices over a data network. This method comprises transferring messages to the embedded devices over the network when the embedded messages are accepting messages. The messages are stored when the embedded device is inactive and remain stored until the embedded device can receive the messages.

The above and other features of the invention including various novel details of construction and combinations of parts, and other advantages, will now be more particularly described with reference to the accompanying drawings and pointed out in the claims. It will be understood that the particular method and device embodying the invention are shown by way of illustration and not as a limitation of the invention. The principles and features of this invention may be employed in various and numerous embodiments without departing from the scope of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

In the accompanying drawings, reference characters refer to the same parts throughout the different views. The drawings are not necessarily to scale; emphasis has instead been placed upon illustrating the principles of the invention.

Of the drawings:

Fig. 1A is a schematic block diagram depicting a cable network infrastructure in which one embodiment of the present invention functions.

Fig. 1B is a schematic block diagram depicting a satellite network infrastructure in which another embodiment of the present invention functions.

Fig. 1C is a schematic block diagram depicting a Digital Subscriber Line (DSL) network infrastructure in which still another embodiment of the present invention functions.

Fig. 1D is a schematic block diagram depicting a wireless network infrastructure in which another embodiment of the present invention functions.

Fig. 2A is a block diagram depicting the interaction of the components of the server system and the embedded client system according to the invention.

Fig. 2B is a process diagram illustrating the interaction between a queue manager and the message router according to the invention.

Fig. 2C is a process diagram illustrating a process for delivering a message via the message router according to the invention.

Fig. 3 is a state line diagram illustrating a process for automatically registering a device and generating user profiles for targeting content according to the invention.

Fig. 4A is a state line diagram illustrating a process for targeting consumers with content (e.g., graphical promotion) and deploying the content to a targeted device according to the invention.

Fig. 4B is a process diagram illustrating a process for scheduling activation of content with a predetermined date and time according to the invention.

Fig. 4C is a process diagram illustrating a process for scheduling activation of content by server activation according to the invention.

Fig. 4D is a process diagram illustrating a process for event driven activation of content according to the invention.

Fig. 4E is a block diagram depicting event driven activation involving trigger signals in the video stream according to the invention.

Fig. 5A is an example of the Document Type Definition used to define the document structure of the user activity and event logs written in XML (Extensible Markup Language) according to the invention.

Fig. 5B is an example of the representation of events stored in a user activity and event log written in XML according to the invention.

Fig. 6 is a state line diagram illustrating a process for automatically for updating user profiles through uploading and parsing user activity and event logs according to the invention.

Fig. 7 is a process diagram illustrating a process for dynamically installing a driver on a network device according to the invention.

Fig. 8 is a flow diagram showing the uninstallation of the driver on the network device according to the invention.

DETAILED DESCRIPTION OF THE INVENTION

The present invention is a communications and management system executing over a data network for targeting content, including promotions, to users of network devices whose attributes match the attributes of a group profile along with maintaining those network devices.

The system and methods of the present invention can be implemented over a variety of data network infrastructure including cable, satellite, Digital Subscriber Lines (DSL), and wireless networks.

Fig. 1A shows one embodiment of the present invention in which the communications and management system is implemented over a cable network. In this environment, audio and video broadcasts are typically frequency multiplexed with data transmissions on the coaxial cables extending from the head end 50 to the exemplary network devices 80A, 80B, 80C, and 80D (collectively referred to as 80).

Video content providers 20 as well as Internet content providers 10 (*i.e.* host web servers) deliver their audio/video/data signals to a cable service provider/internet service provider (CSP/ISP) data center 40. The Internet content providers 10 deliver their data to the data center 40 via the Internet 30. Typically, video content providers 20 transmit their video signals to the data center 40 via some broadcast medium, such as conventional radio-frequency television broadcasting techniques, or via a digital satellite downlink.

The CSP/ISP data center 40 transmits the audio/video/data signals to multiple head ends 50 (only one being shown for simplicity of illustration). The connection between the data center 40 and the head end 50 is typically a hybrid CATV/data connection, which is supported by an optical fiber infrastructure. Part of this infrastructure carries the audio/video signals, which are directed from the data center 40 to the network devices 80. Part of this network also carries the bi-directional data communications associated with network control and internet service provisioning.

The head end 50 distributes the audio/video/data signals over a cable network of hubs 60 and local nodes 70 to a variety of network devices 80, such as set-top boxes, web phones, and cable modems. Some network devices 80D, such as a web phone, have a built-in video display 310 and speaker system 315. Other network devices 80C are peripherally attached to a video display device and speaker system such as a television 300.

In one embodiment of the present invention, the server system (100A and 100B, collectively referred to as 100), is located at the CSP/ISP data center 40 and the head end 50 of the cable network. Installation of the server system at the data center 40 and the head end 40 allows for scalability. The server system 100A at the data center 40 typically provides centralized management for configuring group profiles and content deployment options, while the server system 100B at the head

end 50 preferably handles the registration, user profile updates, content deployment, and other services among the network devices 80.

There are alternative schemes for deploying the server system 100 within the cable network infrastructure depending on the capacity of the server system 100 and number of network devices 80. For example, the server system 100 is deployed at the hub 60 level when the population of devices is sufficiently dense to necessitate such distribution of the communication load.

Fig. 1B shows an embodiment of the present invention in which the communications and management system is implemented on a satellite network. In this environment, the server system 100 is located at data center 45.

The server system 100 transmits data to the network devices 80 via a satellite uplink device 90 to a satellite 93, which, in turn, transmits the data to a residential satellite downlink dish 95. The data are received by the network devices 80 connected to the downlink feed 97.

For the return, upload, path, the network devices 80 transmit data to the server system 100 through a built-in modem, other dial-up device, or a land line system such as ISDN or DSL. The modem connects to a central office or point-of-presence (POP) 55, which, in turn, transmits the return-path data over the Internet 30 to the data center 45.

Fig. 1C shows an embodiment of the present invention in which the communications and management system is implemented on a Digital Subscriber Line (DSL) network. In this environment, the server system 100 is located at data center 45.

The server system 100 communicates bi-directionally with the network devices 80 via the Internet 30 or closed network connection, such as frame-relay, to

a central office or point-of-presence (POP) 55. In one embodiment, the Internet connection between the server system 100 and the central office 55 is over a Virtual Private Network (VPN) providing a private, secure, encrypted connection tunnel.

The network devices 80 are connected to the Internet 30 by the central office 55 via Digital Subscriber Lines (DSL).

Fig. 1D shows an embodiment of the present invention in which the communications and management system is implemented on a wireless network. This environment is similar to the DSL network with the exception that the network devices 80 are connected to the Internet 30 by the central office 55 via wireless, typically CDMA, connections.

Fig. 2A shows the organization of the server system 100 and the embedded client system 200 interacting to implement the communications and management system.

In brief overview, the server system 100 includes a management console 110, a system manager 120, a data store 130, a queue manager 140, a message router 150, a bulk data transfer manager 160, and an XML file processor 170. The embedded client system 200 executing in the network devices include a web browser 210, a system agent 220, a promotion notification agent 230, a queue manager 240, a logging agent, 250, and a bulk data transfer agent 260.

In more detail, the management console 100 is preferably implemented as a web server. In one embodiment of the present invention, the management console 100 is a Microsoft® Internet Information Server (IIS) implementing Active Server Pages (ASP).

The management console 100 provides, upon request by a system administrator, a web page interface for specifying the content to deploy, the attributes of a group profile that target a market segment of potential consumers, installation information, and criteria for activating the content or displaying the promotions at the network devices. Upon submitting the web page, the management console 100 communicates with the system manager 120, via an Application Programming Interface (API) to store the targeted group profile, activation criteria, and the content and promotions to the data store 130. In one embodiment the API is a Microsoft® COM interface.

Content includes, but is not limited to, applications, device drivers, data files, registry sub-hives, and promotions. Promotions are a special type of content that advertise goods and services. Promotions overlay the video display of a network device with a graphic, image, or animated icon that launches a web browser to a host web server in response to the user clicking or selecting it. Promotions also include audio and video clips or data streams. One or more promotions can be displayed on the video display at one time.

The management console 110 also provides a web page interface to users upon request during the initial registration of their network devices. Users will register through the web page interface, providing data about themselves. Upon submitting the web page, the management console 110 communicates with the system manager 120, via the COM interface, to store the user data as attributes of a user profile. The attributes of the user profile are associated with the attributes of the group profile in order to target potential consumers who would be interested in the content or promotions.

The system manager 120 is an application-level process that manages the reading and writing of data to the data store 130. The system manager 120, through its COM interface, allows the management console 110 to store user profiles,

content including promotions along with associated group profiles, installation information, and activation criteria. In addition, the system manager 120 updates the user and group profiles whenever new attributes are received.

The system manager 120 also interacts with the system agent 220 of the targeted network devices by sending and receiving messages through a messaging protocol. The interaction of the system manager 120 and the system agent 220 implement the scheduling of content deployment as well as installation and activation of the content. In one embodiment, the system manager is implemented as a Microsoft® COM object.

The queue manager 140 is an application-level process that communicates with the message router 150 on behalf of other processes, such as the system manager 120, in order to send and receive messages among the embedded client systems 200. In one embodiment, the queue manager 140 is implemented as a C++ object. The queue manager 140 also manages incoming and outgoing queues on behalf of the other processes in the system server 100.

The queue manager 140 handles two types of queues, persistent queues and volatile queues. Messages, whose message type indicates persistent storage, are stored such that the message will not be lost during power outages and lost network connections. A persistent queue is stored in persistent flash memory or in a location on the hard disk of the network device. Other messages, not intended for persistent storage, are stored to volatile queues and might be lost during power outage and lost network connections.

The data store 130 is a database that stores the attributes of the user profiles, group profiles, content and promotions along with the activation criteria. In addition, the data store 130 stores messages intended for network devices that are unavailable during the initial delivery attempt. The data store provides persistence to the data

stored such that the content, profiles, and messages will not be lost during a power outage. In one embodiment, the data store 130 is a Microsoft® SQL version 7 database. Since the data store stores content, it is also known as a content store.

The bulk data transfer manager 160 is an application-level process that is responsible for the transfer of bulk data to targeted network devices. Bulk data include large stream-oriented data, such as a promotions, files, or registry sub-key hives. The bulk data transfer manager 160 does not transmit data over the messaging protocol. Instead, it transmits serialized data over a network transport protocol, such as TCP/IP.

The bulk data transfer manager 160 has access to the data store 130 for transmitting content and promotions.

The XML File Processor 170 is an application-level process that is responsible for parsing out the user attributes from the raw user activity and event logs and updating the appropriate user profiles. In one embodiment, the logs are stored as XML files in the data store 130.

In more detail of the embedded client system 200, the system agent 220 is an application-level process that communicates with the system manager 120 handling various request messages and registration. In handling the various request messages, the system agent communicates with the other embedded client system components in order to effect a proper response or behavior. In one embodiment, the system agent 220 is implemented as a C++ object.

As in the system server, the queue manager 240 is an application-level process that communicates with the message router 150 on behalf of other processes, such as the system agent 220, in order to send and receive messages to the system server 100 and other network devices. In one embodiment, the queue manager 240

is implemented as a C++ object. The queue manager 240 also manages incoming and outgoing queues on behalf of the other processes in the embedded client system 200.

The queue manager 240 handles two types of queues, persistent queues and volatile queues. Messages, whose message type indicates persistent storage, are stored such that the message will not be lost during power outages and lost network connections. A persistent queue is stored in persistent flash memory or in a location on the hard disk of the network device. Other messages, not intended for persistent storage, are stored to volatile queues and might be lost during power outage and lost network connections.

The bulk data transfer agent 260 is an application-level process that handles requests from the system agent 220 to either download content and promotions or upload user activity and event logs. The bulk data transfer agent 260 communicates with the bulk data transfer manager 160 of the system server 100 over a network transport protocol, such as TCP/IP. The bulk data transfer agent 260 notifies the system agent 220 upon completion or failure of the data transfer. In one embodiment, the system agent 220 is implemented as a C++ object.

The promotion notification agent 230 is an application-level process that triggers and handles the display of promotions. The promotion notification agent 230 overlays the promotion or promotions onto the video signal that gets displayed on a monitor connected to a set-top box or to a web phone display. The promotion notification agent 230 coordinates the activation of promotions. A promotion notification agent 230 will display the promotion in response to an event, invocation by the system manager 120, or scheduling information provided with the promotion itself.

The web browser 210 is an application-level process that displays web pages from web host servers such as the management console 110 of the system server 100 enabling registration of user attributes.

The logging agent 250 is an application-level component that monitors and logs a variety of user activities and events. In one embodiment, the logging agent 250 stores the log files in XML format. User activities and events that are tracked by the logging agent 250 are channel events, promotion events, power events, peripheral events, and application events.

Channel events occur whenever the network device stays tuned to a channel for a configurable amount of time. Promotion events occur in response to consumer actions taken with respect to promotions displayed on the video display. For example, a promotion event is recorded when the consumer clicks or selects the promotion icon to navigate to the web server hosting the promotion.

The interaction of the server system 100 and the embedded client system 200 provides a system for targeting and scheduling deployment of promotional content to consumers of a targeted market segment, for managing the activation of the promotional content, and for tracking consumer response to the promotion.

Application-level processes, such as the system manager 120 of the server system and the system agent 220 of the network device, communicate over the data network through messages.

Messages transfer requests for action, responses to requests, and small data transfers. Messages are transported in the payload of a network transport protocol, such as TCP/IP. Messages are sent to destinations using a globally unique identifier, GUID, in order to identify the destination network device or application. This messaging protocol allows application-level processes to transmit data without

knowing about the network transport interface, the device's network address, or whether the device is active on the data network.

The interaction of the message router 150 with the queue managers of the source and destination processes implements the messaging protocol. Any queue manager whether it is executing on the system server 100 or the embedded client system 200 communicates with the message router 150 in the same manner.

Fig. 2B illustrates the interaction between a queue manager and the message router 150 according to the invention. For example, when the system manager 120 needs to transmit a message to a system agent 220, the system manager 120, in step 1000, sends the message to the queue manager 140 indicating the message type, a globally unique identifier (*i.e.*, GUID) of the destination device, and the message data. The details on how a network device obtains a GUID is described later with reference to Fig. 3.

In step 1002, the queue manager 140 stores the message in a queue for the system manager 120 and then attempts to establish a connection with the message router 150.

In brief overview, there are three steps in order for the queue manager 140 to establish a connection to the message router 150. In step 1004, the queue manager 140 determines the IP address of the message router 150. In step 1006, the queue manager 140 creates and opens a socket pair connection to the message router 150 for transmitting serialized data. In step 1008, the queue manager 140 sends a message to the message router 150 indicating that the queue manager 140 is alive and connected and ready to transmit serialized data.

In more detail of step 1004, the queue manager 140 has one of its properties being the location or name of the message router 150. Using DNS, or IP host name

services, the queue manager 140 determines the IP address of the message router 150. If the queue manager 140 cannot resolve the IP address of the message router 150, the queue manager 140 resorts to a broadcasting scheme. The queue manager 140 broadcasts a locator message on its subnet attempting to locate the message router 150. If there is a message router 150 on that subnet, the message router 150 responds back with its IP address. This address is cached by the queue manager 140 for future connections.

In step 1006, the queue manager 140, knowing the IP address of the message router 150, creates and opens a socket pair on predetermined, known ports for transmitting serialized data.

In step 1008, once the socket pair is opened, the queue manager 140 sends a message notifying the message router 150 that the queue manager 140 is alive and connected and has socket pairs on which to read or write serialized data.

In step 1010, the queue manager 140 writes the message for delivery to the message router 150 through the established TCP/IP socket connection

Fig. 2C illustrates the process of delivering the message once received by the message router 150.

Upon completion of the writing of the message, the message router 150 extracts the message type and the destination GUID from the message in step 1012.

In step 1014, the message router 150 resolves the destination GUID by looking-up the IP address associated with the GUID in the data store 130.

In step 1016, the message is encapsulated in an IP packet, with the appropriate destination IP address. In one embodiment, the IP address of the

network device becomes known to the system server 100 during initial registration of the network device and is stored as an attribute of the user profile.

In step 1018, the message router 150 determines the type of the message. The message type indicates the quality of service that the message router 150 provides for delivery of the message.

If the message type is a standard datagram, the message router 150 simply transmits the message in step 1022. The message router 150 will not keep track of whether the message was actually received.

If the message type indicates guaranteed delivery, the message router 150 will transmit the message and wait for an acknowledgment from the destination device in step 1024.

If no acknowledgment is received after several attempts, the destination is deemed unavailable and the message is stored in the data store 130 for later retransmission when the destination is active on the data network in step 1028. Specifically, in step 1029, the message router waits for the network device 80 to become active in order to deliver the message. In one embodiment, the message router 150 is notified that the network device is active by receiving a message from the network device 80 indicating its active status. In an alternative embodiment, the message router is notified of the active status of a previously unavailable network device by the system manager 120 which monitors the status of the network devices 80. When the network device becomes active, the message router proceeds back to step 1018 to begin the process of delivery again.

If the acknowledgment is received, then, in step 1030, the delivery is complete and the message is removed from the data store 130 if the network device was previously unavailable.

The interaction of the message router 150 and the queue manager 140 for delivering messages occurs whenever a message is sent or received using the messaging protocol.

In order for the server system 100 to target content and promotions to a particular market segment, the server system 100 references its stored user profiles, each user profile being a collection of user and device attributes associated with a network device. All network devices whose user profiles match the attributes of the group profile, targeted by a system administrator, are scheduled for content deployment.

However, when a network device is connected to the network infrastructure for the first time, the system server does not have a user profile for the network device. The present invention provides an automated system and method for initially registering and generating a user profile for an network device.

Fig. 3 is a state line diagram showing the interaction of the server system 100 and the embedded client system 200 for generating an initial user profile for a network device.

In step 1, the system agent 220 of the network device generates and transmits a registration request message containing a number of device attributes to the system manager 120.

The device attributes describes the network device and is configured during the manufacturing process of the device itself. For example, the network device may be configured with a model number attribute for a particular group of network devices, such as intelligent set-top boxes, version 1.0.

In step 2, the system manager 120 receives the registration request message and, in response, retrieves a globally unique identifier, GUID, from an available pool of GUIDs stored in the data store 130.

In step 3, the system manager 120 generates and transmits a registration response message containing the assigned GUID to the system agent 220 of the registering device.

The assigned GUID is used by the network device to identify itself in messages transmitted to the system server 100 and to other network devices. The assigned GUID is also used by the system server 100 to associate the network device with a user profile within the data store 130.

In step 4, the system agent 220 launches a web browser 210. The web browser 210 transmits an HTTP request to the URL (Uniform Resource Locator) of the management console 110 for a registration web page. The assigned GUID is included in the URL string in order to identify the registering network device.

In step 5, the management console 110 receives the HTTP request. In response, the management console 110 makes a call via the COM interface of the system manager 120 to retrieve the device and user attributes, if any, associated with the GUID of the registering network device. The management console 110 generates the registration web page customized for the registering network device. The web page is transmitted via HTTP to the web browser 210.

In step 6, the web browser 210 displays the registration web page wherein the user submits information which will be used to generate a user profile of user attributes associated with the network device. Such information includes, but is not limited to, name and address information, channels frequently watched, requests for installation of optional value-add services and applications, and various

demographic and personal information.. Upon submitting the registration data, the web browser 210 transmits the user attributes, represented as HTML data via HTTP, to the management console 110.

In step 7, the management console 110 interprets the HTML data stream and makes calls via the COM interface of the system manager 120 to update the user profiles in the data store 130 with the provided user attributes.

In step 8, the system manager 120 updates the user profile of the registering network device with the user attributes on the data store 130. After updating the user profile, the system manager 120 associates the user profile with group profiles whose attributes match user attributes of the user profile.

For example, the user profile will be added to the group profile for network devices with the same model number attribute. The user profile is added to any number of group profiles that target particular attributes of the registered user or network device. These group profiles are used by the system manager 120 for targeting consumers of particular market segments for various e-commerce promotions or application services.

Once the network device is registered and is associated with a user profile, the device is capable of being targeted for deployment of content or promotions.

Fig. 4A is a state line diagram showing the interaction of the server system and the embedded client system for deployment of content and promotions to a network device. Deployment includes, but is not limited to, downloading and installing.

Before content can be deployed to a targeted device, the content must be stored in the data store 130 along with a group profile and an activation schedule.

The group profile indicates the attributes of network devices to target. The activation schedule indicates when to activate the content or promotions. Activation can be event driven, scheduled from the system server 100, or initiated by the system manager 120.

In step 1, a system administrator with access to the management console 110 populates a server-based web page indicating the content to deploy as well as the criteria with which to define the group profile. Additionally, the system administrator indicates when to activate the content. Upon submitting the data, the management console 110 makes a call to the COM interface of the system manager 120 to generate a group profile in the data store 130 with user profiles whose attributes match the criteria defined by the system administrator.

In step 2, the system manager 120 updates the data store 130 creating the group profile and populating the group profile with user profiles with matching attributes.

In step 3, the management console 110 make a call through the COM interface of the system manager 120 to download the content or promotion to the data store 130.

In step 4, the system manager 120 writes the content to the data store 130.

The system manager 120 is configured to schedule deployment of content during off-peak hours when bandwidth utilization is typically at a minimum. For example, during the hours of 3:00AM and 5:00AM, more bandwidth is available for efficient deployment of content and promotions. Alternatively, the system manager 120 monitors network utilization and is configured to schedule deployment of content when the detected bandwidth utilization falls below a predetermined level.

In step 5, the system manager 120 sends a download and install request message to each of the system agents 220 of the network devices whose user attributes match the attributes of the group profile. The download and install message informs the system agent 220 to download install the content or promotion referenced by a GUID.

Alternatively, the system manager 120 sends a download, install, and start request message which indicates, in addition, when or under what event conditions the content should be activated (i.e. promotion displayed or an application launched).

In step 6, the system agent 220 makes a C++ object method call to the bulk data transfer agent 260 to download the content having the provided GUID.

In step 7, the bulk data transfer agent 260 sets up a TCP/IP socket connection to the bulk data transfer manager 160 of the server system to initiate the delivery of the application.

In step 8, the bulk data transfer manager 160 delivers the requested content to the bulk data transfer agent 260 through the TCP/IP socket connection. In cases where the connection is broken, the bulk data transfer agent 260 and the bulk data transfer manager 160 can detect that a connection was broken and will continue the download the content from the point in the transfer where the break occurred.

In step 9, the bulk data transfer agent 260 notifies the system agent 220 the result of the data transfer via an C++ object method call.

In step 10, the system agent 220 sends a message to the system manager 120 indicating the result of the data transfer.

After the content is installed on the targeted network device, the present invention provides a system and method for activating that content. Activation allows the user to interact with the installed content, such as playing a game or initiating an e-commerce transaction.

There are two types of activation that the present invention implements - scheduled activation and event driven activation. Scheduled activation allows the system administrator to specify when to activate the content, whereas event driven activation allows the system administrator to specify an event which triggers the activation of the content.

Scheduled activation is implemented in two ways, predetermined scheduling and activation by the system server.

Fig. 4B illustrates the steps associated with content activation via predetermined scheduling. Predetermined scheduling based upon date and time provides the most autonomy to the network device. After the content is deployed, the network device simply waits for the specified date and time to arrive, at which time it displays the content.

In step 1110, when the group profiles are configured and the content is downloaded to the data store 130, the system administrator also specifies the date and time to activate the content. Where the content is a promotion, a duration period is specified along with the activation date and time.

In step 1112, the system manager 120 sends the download, install, and start request message to the system agent 220 of a targeted network device. In addition to requesting the system agent 220 to install content, the message indicates the date and time to activate the installed content.

If the content is a not a promotion, the system agent 220 waits for the specified activation date and time in step 1116.

In step 1118, the content is activated by the system agent 220 at the specified date and time.

If the content is a promotion, the system agent 220 transfers the predetermined date, time, and duration to the promotion notification agent 230 in step 1120.

In step 1122, the promotion notification agent 230 waits for the specified activation date and time.

In step 1124, the promotion is activated by the promotion notification agent 230 at the specified activation date and time.

If the promotion is a icon or graphic linked to a URL of a host web server, the promotion notification agent 230 overlays the promotion on a portion of the video display built-in or attached to the network device. If the promotion is audio clip or data stream, the audio is played through a speaker built-in or attached to the network device. If the promotion is a video clip or data stream, the video overlays a portion of the video display built-in or attached to the network device.

Fig. 4C illustrates the steps associated with server activation of content. Server activation of content allows control to reside at the server system 100, therefore, maximizing the control by the institution operating the server system 100.

As described previously, the content is installed on the network device in step 1210 with no activation information.

In step 1212, the system agent 220 waits for a start message from the system server 100.

In step 1214, upon request of the system administrator, the system manager 120 sends a start message to the system agent 220 specifying the installed content to activate.

If the installed content is not a promotion, the system agent activates the content in response to receiving the message in step 1218. This may include, but not limited to, launching an application installed within the network device.

If the installed content is a promotion, the system agent 220 notifies the promotion notification agent 230 via a C++ object method call to activate the specified promotion in step 1220.

In step 1222, the promotion is activated by the promotion notification agent 230.

Event driven activation is particularly suited for coordinating the activation of content with a particular event or a particular moment in a corresponding analog and/or digital video stream.

Event activation has advantages associated with high scalability. The content can be loaded in the days or weeks preceding the general time period when it is to be displayed.

In one embodiment, events that trigger content activation are channel events, power events, and peripheral events. A power event is an event relating to the power supply, such as the network device being powered on or off. A peripheral event is

an event relating to peripheral devices being connected or disconnected from the network device, such as a joy stick or other gaming console. A channel event is an event relating to the channel being watched by the user.

Fig. 4D illustrates the steps associated with event driven activation according to the invention.

In step 1310, when the group profiles are configured and the content is downloaded to the data store 130, the system administrator also specifies an event map. An event map associates events to content indicating when to activate the content. Where the content is a promotion, a duration period is specified as well.

In step 1312, the system manager 120 sends the event map in a message, such as a download, install, and start message, to the system agent 220 of a targeted network device.

If the content is not a promotion, the system agent 220 waits for the specified event or events that trigger the activation of the content in step 1316.

In step 1318, the system agent 220 activates the content when the specified event or events occur.

Conversely, if the content is a promotion, the system agent 220 transfers the event map to the promotion notification agent 230 in step 1320.

In step 1322, the promotion notification agent 230 waits for the specified event or events to occur.

In step 1324, the promotion notification agent 230 activates the promotions associated with the event or events that occurred.

An example of event driven activation is where the event map provides for the activation of promotions involving sporting goods when the potential consumer has been watching a particular sports channel for a period of time. The watching of the sports channel for a period of time triggers a channel event. The channel event triggers the activation of the promotion or promotions.

The present invention provides an additional implementation of event driven activation involving technology from ATVEF (Advanced Television Enhancement Forum). ATVEF provides a standard for embedding HTML tags within a video signal.

The promotion content agent 230 monitors the video signal for the embedded triggers, such as the HTML tag. The capture of this embedded trigger causes the activation of one or more promotions in real-time coinciding with the video signal.

Such a system has advantages in that very little video signal editing is required. Only a small trigger has to be embedded in the video signal, requiring little analog video editing capabilities at the data center.

The content, is simultaneously activated on all of the network devices allowing high levels of synchronization to the video signal. For example, the promotion can be synchronized to occur during a television commercial.

The provider of the commercial simply embeds an initialization or start HTML tag within its video signal. In response to the promotion notification agent 230 capturing the HTML tag, the promotion notification agent 230 activates the appropriate promotion or promotions specified in the event map for that HTML tag.

Fig. 4E shows one embodiment of this system on a conventional set-top box 80 for an analog or an analog/digital video display device such as a television 300.

Specifically, the data and audio/video stream is received by the set-top box 80. This data audio/video stream is received from the head end 50 via the hub 60.

The promotion content agent 230 of the embedded client system 200 monitors the video stream for the embedded trigger signal. When the trigger signal is detected, the promotion content agent 230 inserts the associated graphical promotion content indicated in the event map into the analog or digital video stream to the display 310 of the television 300. As a result, the promotion 320 appears on the display screen 310, overlaying the video.

User selection of this promotion through a selecting device, such as a remote control device, sends a URL to the web browser 210 bringing the browser window to the forefront of the display 310 of the television 300. In this way, user selection of the promotion allows the user to receive and view data from the URL enabling e-commerce transactions.

Therefore, the event-driven activation, as well as the other scheduling options, presents promotions in an appropriate context to further increase the likelihood of consumer e-commerce transactions.

In addition to the initial registration process, the present invention includes a system and method for updating user profiles through uploading distributed user activity and event logs to the system server 100, parsing out the user and device attributes from the logs, and updating the user profiles in the data store 130.

The logs provide useful information, because the logs track a variety of information which the system can use in order to more accurately target users for content and promotional deployment. In one embodiment, the logs track channel events and e-commerce transactions initiated through the display of promotions. In another embodiment, the logs track peripheral events, such as the addition of a

joystick and console for gaming purposes, power events, application events, and promotion events. Continuous updating of user profiles through this system and method improves the targeting of consumers for content deployment, and in particular, for promotional content deployment.

In brief overview, the network device includes a logging component 250 that monitors and logs user activity and events in an generic file format. For example, the logging component 250 monitors user activity at a user interface device such as the television remote channel control. In one embodiment, the generic file format is Extensible Markup Language (XML). The format of the log files, using XML, correspond to the structure of the user profiles in the data store 130. This allows for processing of the logs in an automated fashion.

In addition, the user activity and event logs include a description of the structure of the document itself. Using XML, the description of the structure of the document is the Document Type Definition (DTD). Providing a description of the document structure within the logs themselves, allows for the server system 100 to process logs with different document structures. This avoids the necessity of having to update the server system 100 every time a new document structure is used within the logs.

Fig. 5A is an example of a DTD for an event log where each event that occurs is recorded with the GUID of the device, the time of the event, the event type, and a description of the event.

Fig. 5B is an example of an event log written in XML using the DTD of Fig. 5A. The event log contains two events, stored in the structure described by the DTD. Fig. 5B demonstrates how a channel event and an application event are described within this log.

The uploading and parsing of these user activity and event logs will provide additional user attributes for targeting consumers including information regarding responses to prior promotions. If the logs indicate that a user is interested in a particular type of promotion, the system server modifies the user attributes of his user profile such that they match the attributes of a group profile associated with that type of promotion.

Fig. 6 is a state line diagram showing the interaction of the server system 100 and the embedded client system 200 for updating user profiles through the upload and parsing of user activity and event logs.

In step 1, the system manager 120 sends a message to the system agent 220 to upload its activity logs to the server system.

In step 2, the system agent 220 makes a C++ object method call to the bulk data transfer agent 260 to upload the user activity and event logs.

In step 3, the bulk data transfer agent 260 sets up a TCP/IP socket connection to the bulk data transfer manager 160 of the server system to initiate the delivery of the logs.

In step 4, the bulk data transfer agent 260 delivers the logs to the bulk data transfer manager 160 through the TCP/IP socket connection where they are stored in the data store 130.

In cases where the connection is broken, the bulk data transfer agent 260 and the bulk data transfer manager 260 can detect that a connection was broken and will continue the download the content from the point in the transfer where the break occurred.

In step 5, the bulk data transfer agent 260 notifies the system agent 220 the result of the data transfer via a C++ object method call.

In step 6, the system agent 220 sends a message to the system manager 120 indicating the result of the data transfer.

In step 7, the system manager 120 makes a call to the XML file processor 170 to update the user profiles from the user activity and event logs.

In step 8, an XML file processor 170 at the server system parses the logs stored on the database and updates the user attributes of the user profile of the network device. This system and method for scheduling remote uploads of the user activity and event logs provides improves the efficiency for targeting consumers for content and promotion.

There are situations where a user will change the hardware configuration of a network device in order to expand its capabilities. In that situation, the present invention provides a system and method by which the network device notifies the server system 100 of a change to its hardware configuration and, in return, receives the appropriate device drivers to support the new hardware configuration.

Fig. 7 shows the dynamic installation of device drivers on the network device according to the present invention.

Specifically, the dynamic driver installation process is triggered when the user installs a peripheral device on a network device for which the network device requires a driver. In the typical example, the process occurs when the user plugs in a peripheral device such as a joy stick into a port such as a serial port or USB (universal serial bus) port.

In step 1002, the system agent 220 intercepts the plug and play string from the peripheral device when it is attached to the USB port.

In step 1004 the system agent 220 then sends this plug and play string to the system manager 120 via the message router 150 and the queue managers along the path between the system agent 220 and the system manager 120.

In step 1006, the system manager 220 then searches for a matching driver in its data store 130. Specifically, it compares the plug and play string received from the network device to plug and play strings of supported operating systems and supporting peripheral devices for which drivers are available.

In step 1008, assuming the valid device driver has been located, the system manager 120 sends a message to the system agent 220 to download the driver providing its location in the data store 130.

In step 1010, the system agent requests the bulk data transfer agent 260 on the network device to download the driver. The bulk data transfer agent 260 then contacts the bulk data transfer manager 160 and downloads and stores the device driver on the network device. In parallel, the system manager 120 instructs the system agent 220 on how to install the device driver on the network device.

In step 1012, in the typical implementation, the device driver is dynamically loaded onto the network device.

In step 1014, when the driver has been successfully installed, the system agent 220 notifies the system manager 120. The system manager, in turn, updates the status of the network device in the system manager's data store 130.

Fig. 8 illustrates the mirror process in which the peripheral device driver is uninstalled.

In step 1016, the system agent 220 is notified when the peripheral is disconnected by the user from the network device.

In step 1018, the system agent 220 then requests an uninstall program from the system manager 120.

In step 1020, typically, the bulk data transfer agent 260 obtains the uninstalled program from the bulk data transfer manager 160.

In step 1022, the driver is then uninstalled.

In step 1024, upon the successful uninstall, the system agent 220 notifies the system manager 120 that the driver has been installed and the system manager 120 updates the network device's status.

While this invention has been particularly shown and described with references to preferred embodiments thereof, it will be understood by those skilled in the art that various changes in form and details may be made therein without departing from the scope of the invention encompassed by the appended claims.

CLAIMS

What is claimed is:

1. A message router system for a server system that communicates with embedded devices over a data network, the router system comprising:

a router that transfers messages to the embedded devices on the data network when the embedded devices are accepting messages; and

a message store that temporarily stores messages addressed to embedded devices until the embedded devices can accept messages.

2. A message router system as recited in Claim 1, further comprising a system manager that tracks states of embedded devices on the data network and whether the embedded devices are able to receive messages.

3. A message router system as recited in Claim 2, further comprising a queue manager for facilitating the transfer of messages between the message router and a process.

4. A message router system as recited in Claim 1, wherein the router retrieves messages from the message store when a system manager indicates that an embedded device to which the messages are addressed is able to accept the messages.

5. A message router system as recited in Claim 1, further comprising a bulk data transfer manager for transferring larger data files between the server system and the embedded devices.

6. A message router system as recited in Claim 5, wherein larger data files are transferred to the embedded devices by the router sending the embedded devices a message to download a file and a location of the file, the embedded devices contacting the bulk data transfer manager to obtain the file.

7. A message router system as recited in Claim 6, wherein the embedded devices directly contact the bulk data transfer manager to obtain the file without sending a message via the router.

8. A method for routing messages from a server system to embedded devices over a data network, the method comprising:

transferring messages to the embedded devices over the data network when the embedded devices are accepting messages; and

storing messages addressed to embedded devices until the embedded devices can accept the messages.

9. A method as recited in Claim 8, further comprising tracking states of embedded devices on the data network and whether the embedded devices are able to receive messages.

10. A method as recited in Claim 9, further comprising queuing messages that are received from a server system prior to being transferred to the embedded devices.

11. A method as recited in Claim 8, further comprising:

detecting whether a previously unavailable embedded device is available to receive messages; and

retrieving stored messages for the embedded device and transferring the messages to the embedded device.

12. A method as recited in Claim 8, further comprising transferring larger data files from the server system to the embedded devices.

13. A method as recited in Claim 12, wherein the step of transferring the larger data file comprises:

sending the embedded devices a message to download a file and a location of the file; and

the embedded devices contacting a bulk data transfer manager to obtain the file.

14. A method as recited in Claim 13, further comprising the embedded devices directly contacting the bulk data transfer manager to obtain the file.

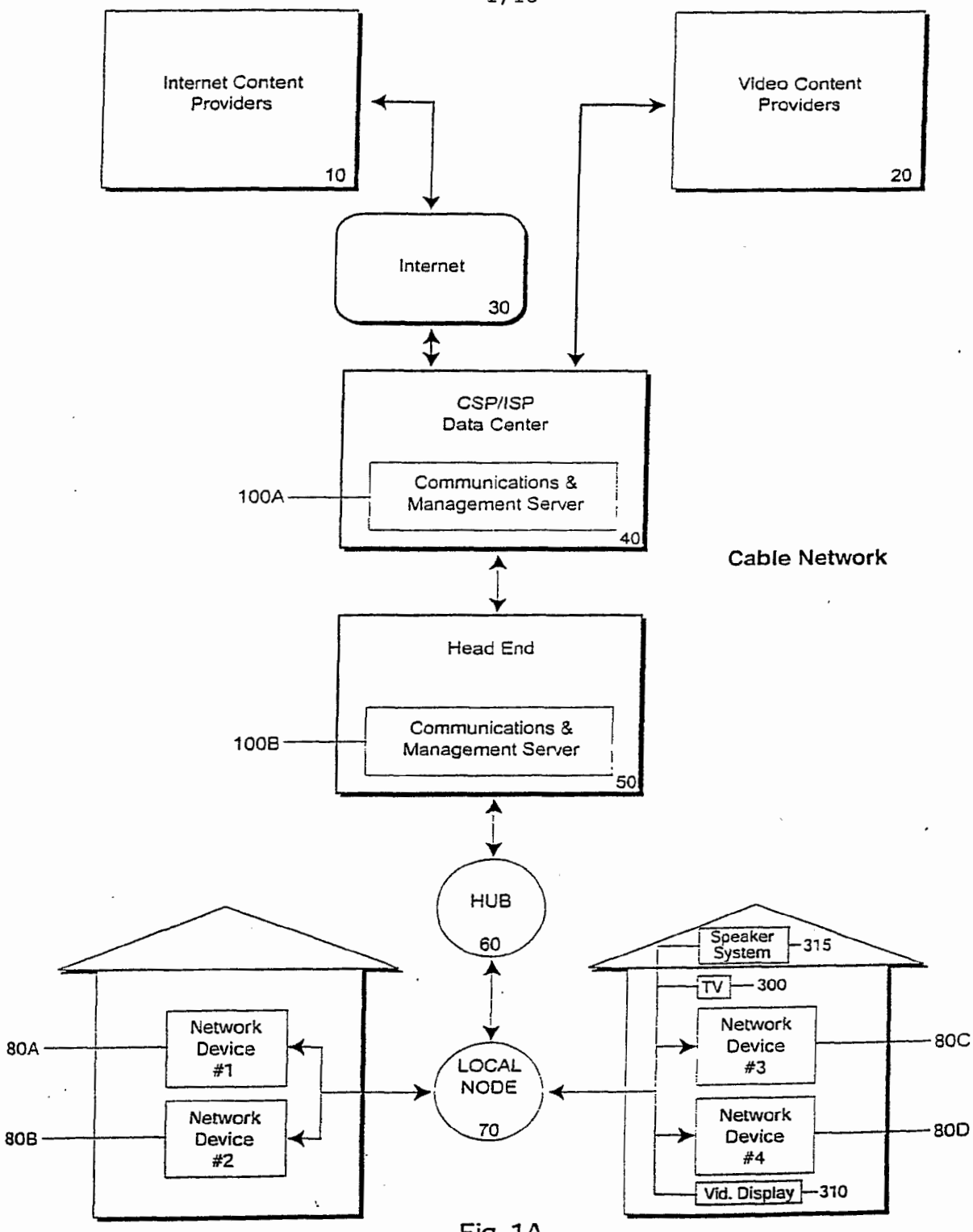

Internet Content Providers
10
Video Content Providers
20
Internet
30
CSP/ISP Data Center
Communications & Management Server
40
100A
Cable Network
Head End
Communications & Management Server
50
100B
HUB
60
LOCAL NODE
70
Network Device #1
80A
Network Device #2
80B
Speaker System
315
TV
300
Network Device #3
80C
Network Device #4
80D
Vid. Display
310

Fig. 1A

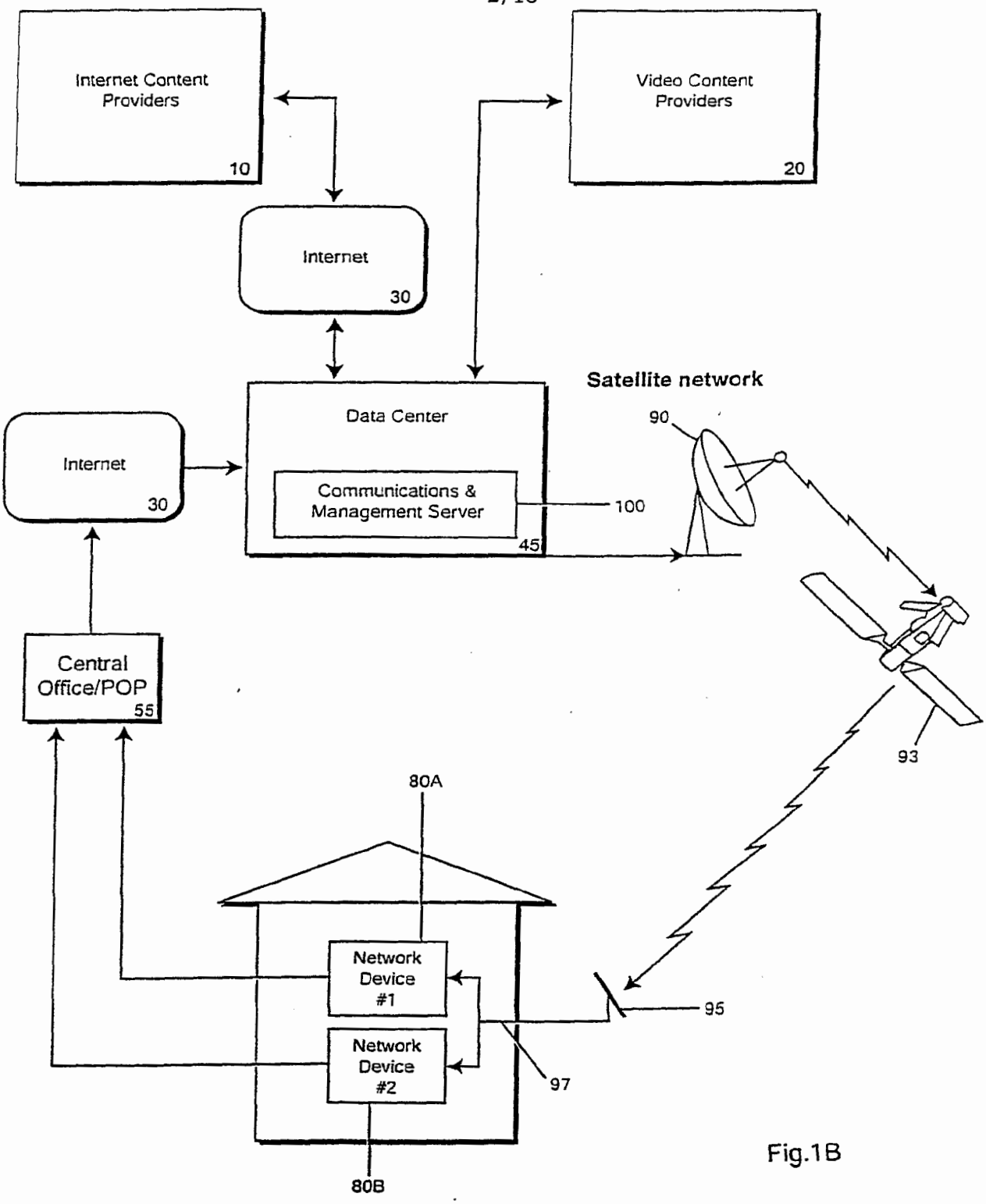

Internet Content Providers
10
Video Content Providers
20
Internet
30
Satellite network
Data Center
Internet
30
Communications & Management Server
45
100
90
Central Office/POP
55
93
80A
Network Device #1
Network Device #2
95
97
80B

Fig.1B

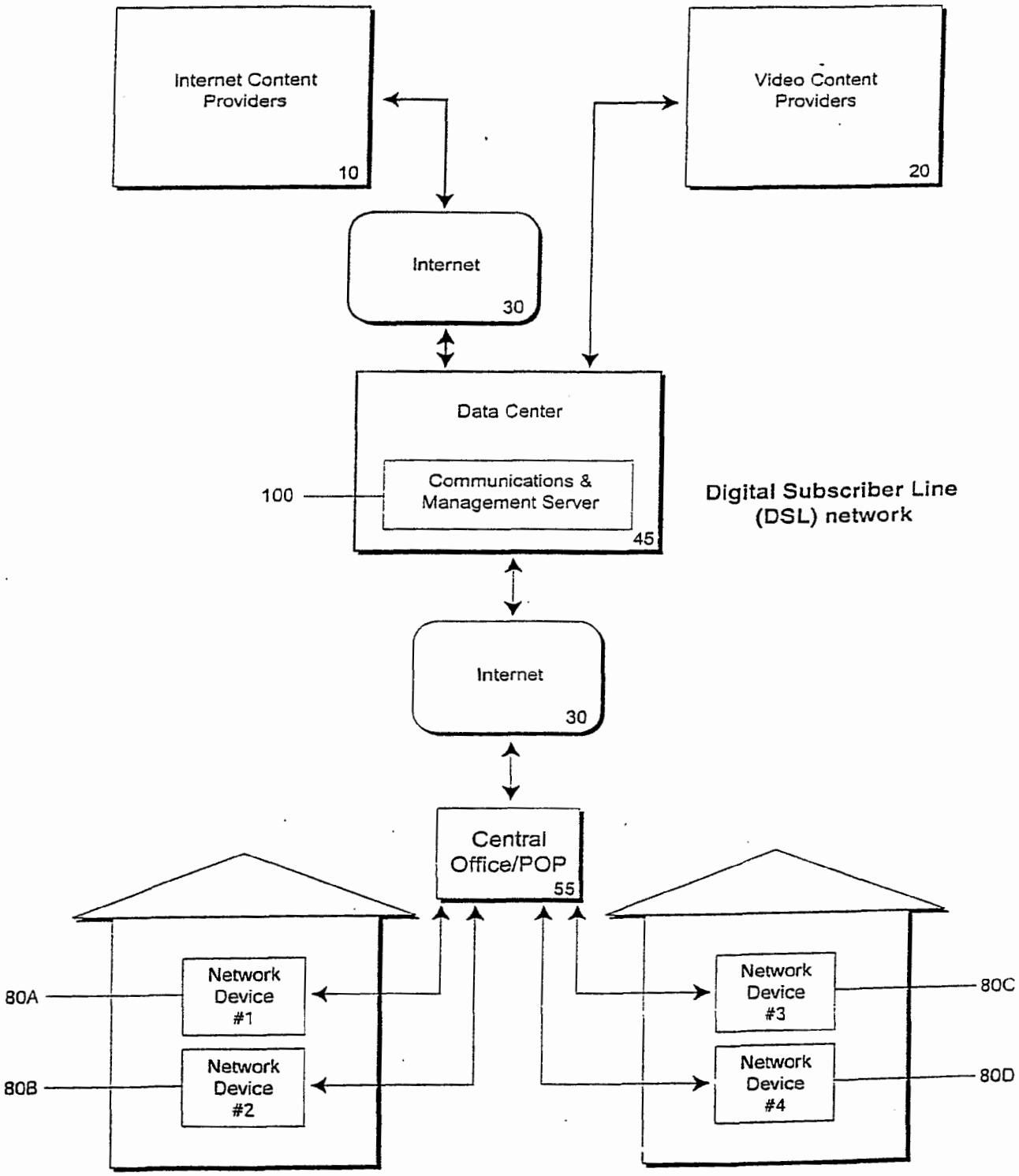

Internet Content Providers
10
Video Content Providers
20
Internet
30
Data Center
100
Communications & Management Server
45
Digital Subscriber Line (DSL) network
Internet
30
Central Office/POP
55
80A
Network Device #1
80B
Network Device #2
Network Device #3
80C
Network Device #4
80D

Fig. 1C

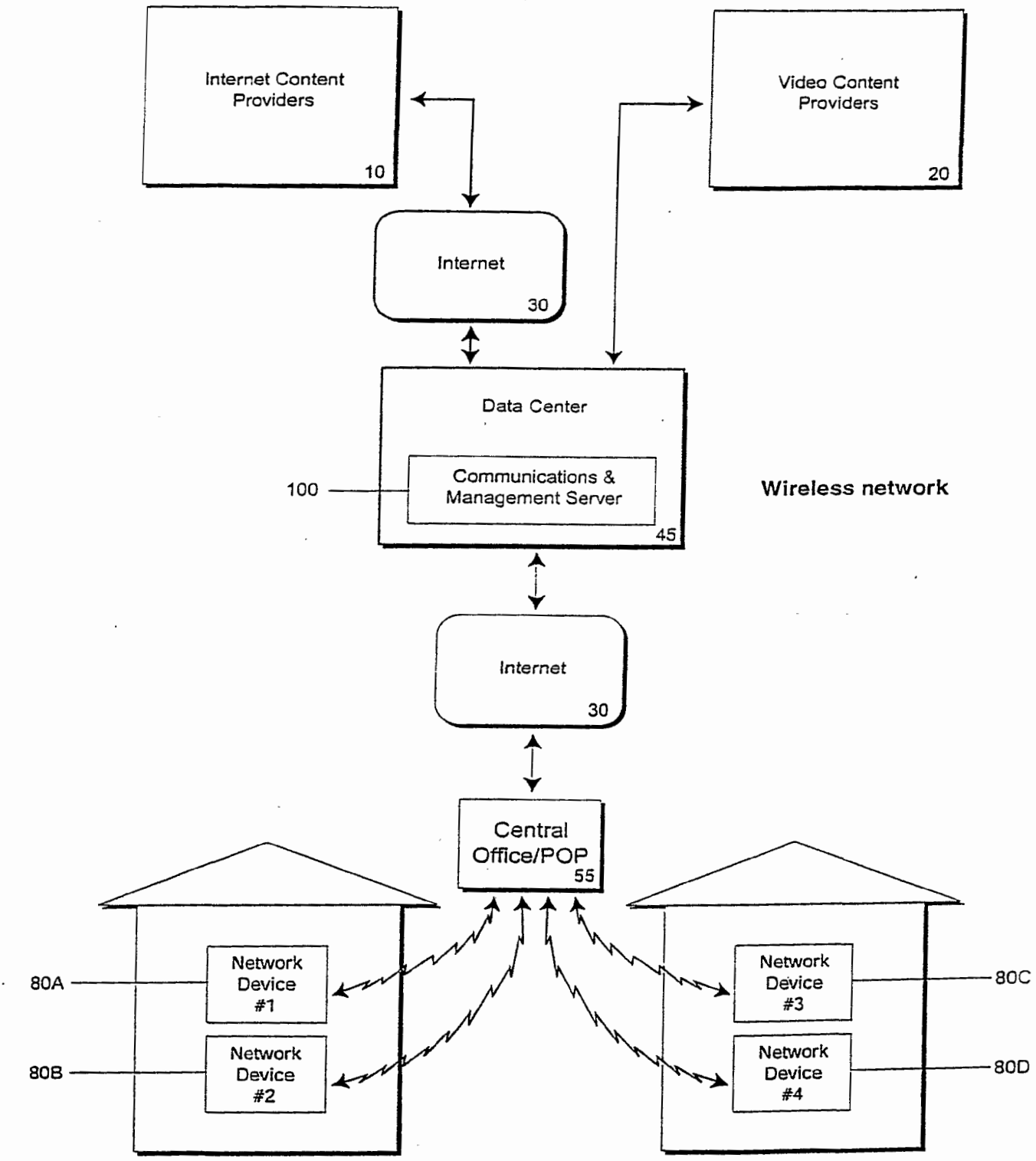

Internet Content Providers
10
Video Content Providers
20
Internet
30
Data Center
100
Communications & Management Server
45
Wireless network
Internet
30
Central Office/POP
55
80A
Network Device #1
80B
Network Device #2
Network Device #3
80C
Network Device #4
80D

Fig. 1D

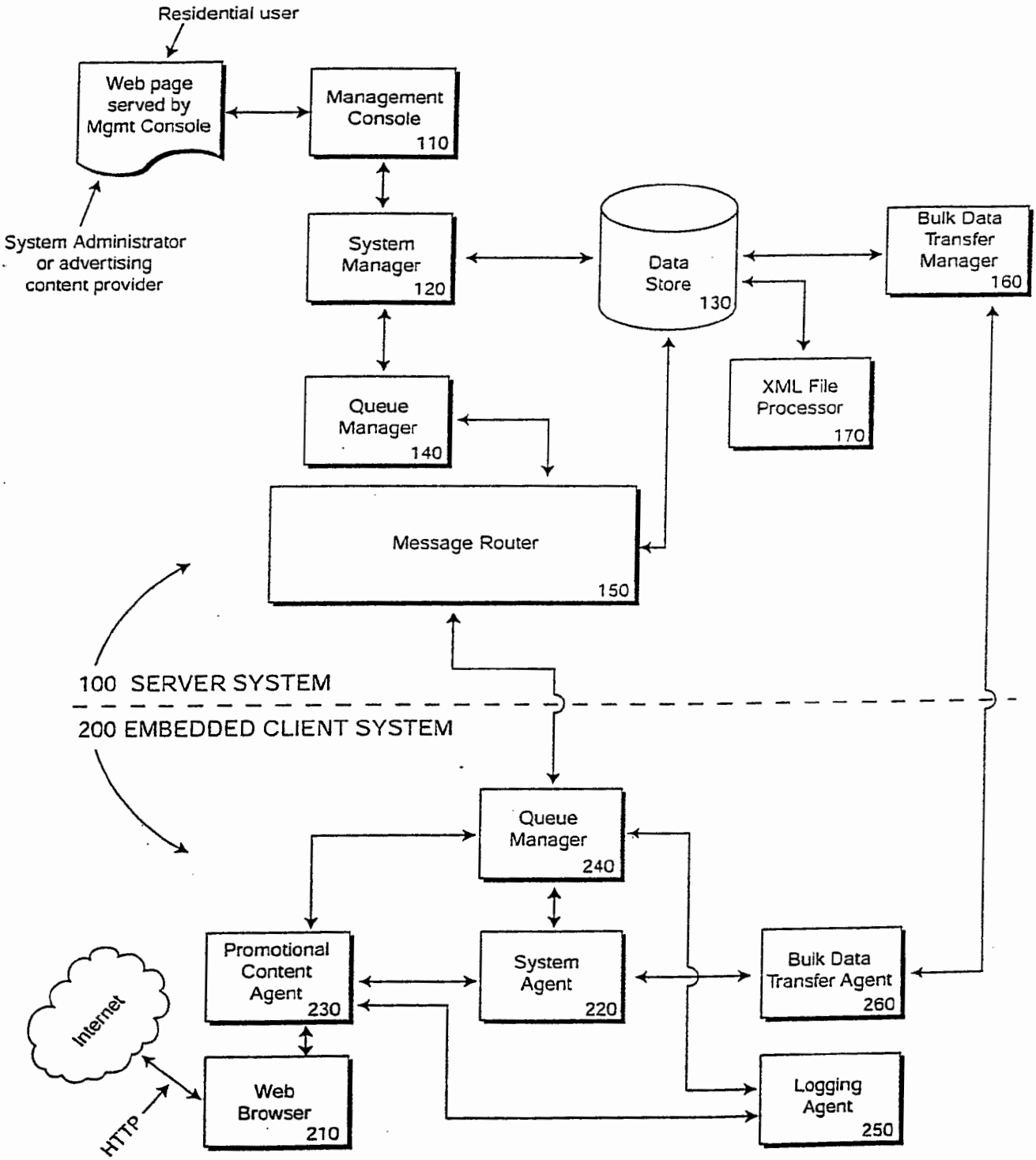

Residential user
Web page served by Mgmt Console
Management Console 110
System Administrator or advertising content provider
System Manager 120
Data Store 130
Bulk Data Transfer Manager 160
XML File Processor 170
Queue Manager 140
Message Router 150
100 SERVER SYSTEM
200 EMBEDDED CLIENT SYSTEM
Queue Manager 240
Promotional Content Agent 230
System Agent 220
Bulk Data Transfer Agent 260
Internet
Web Browser 210
HTTP
Logging Agent 250

Fig. 2A

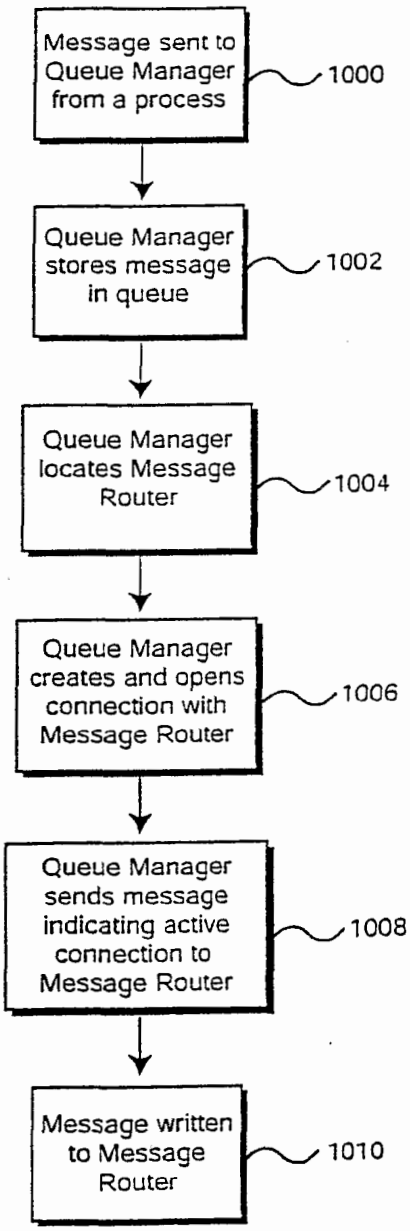

Message sent to Queue Manager from a process
1000
Queue Manager stores message in queue
1002
Queue Manager locates Message Router
1004
Queue Manager creates and opens connection with Message Router
1006
Queue Manager sends message indicating active connection to Message Router
1008
Message written to Message Router
1010

Fig.2B

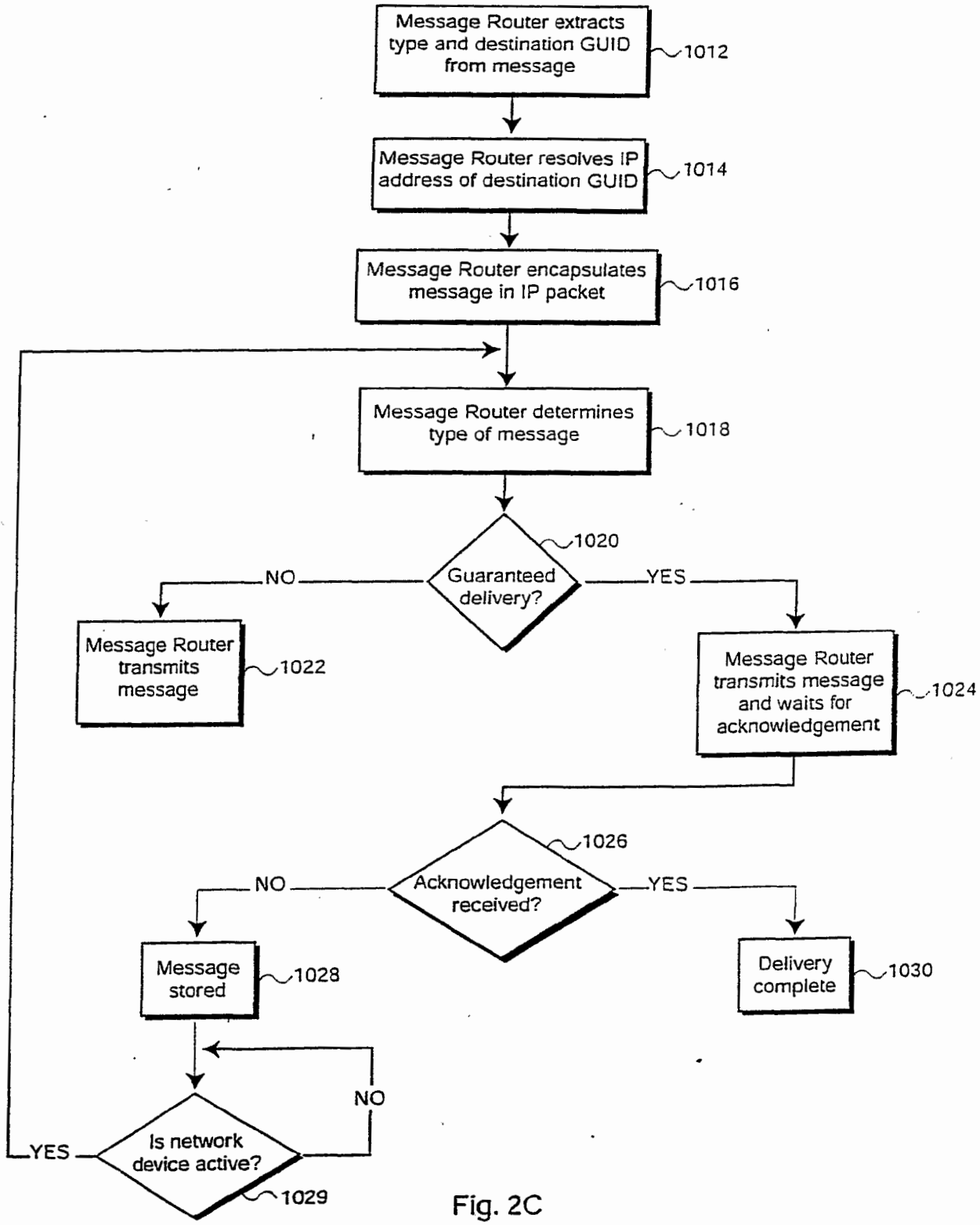

Message Router extracts type and destination GUID from message
1012
Message Router resolves IP address of destination GUID
1014
Message Router encapsulates message in IP packet
1016
Message Router determines type of message
1018
Guaranteed delivery?
1020
NO
YES
Message Router transmits message
1022
Message Router transmits message and waits for acknowledgement
1024
Acknowledgement received?
1026
NO
YES
Message stored
1028
Delivery complete
1030
Is network device active?
1029
NO
YES

Fig. 2C

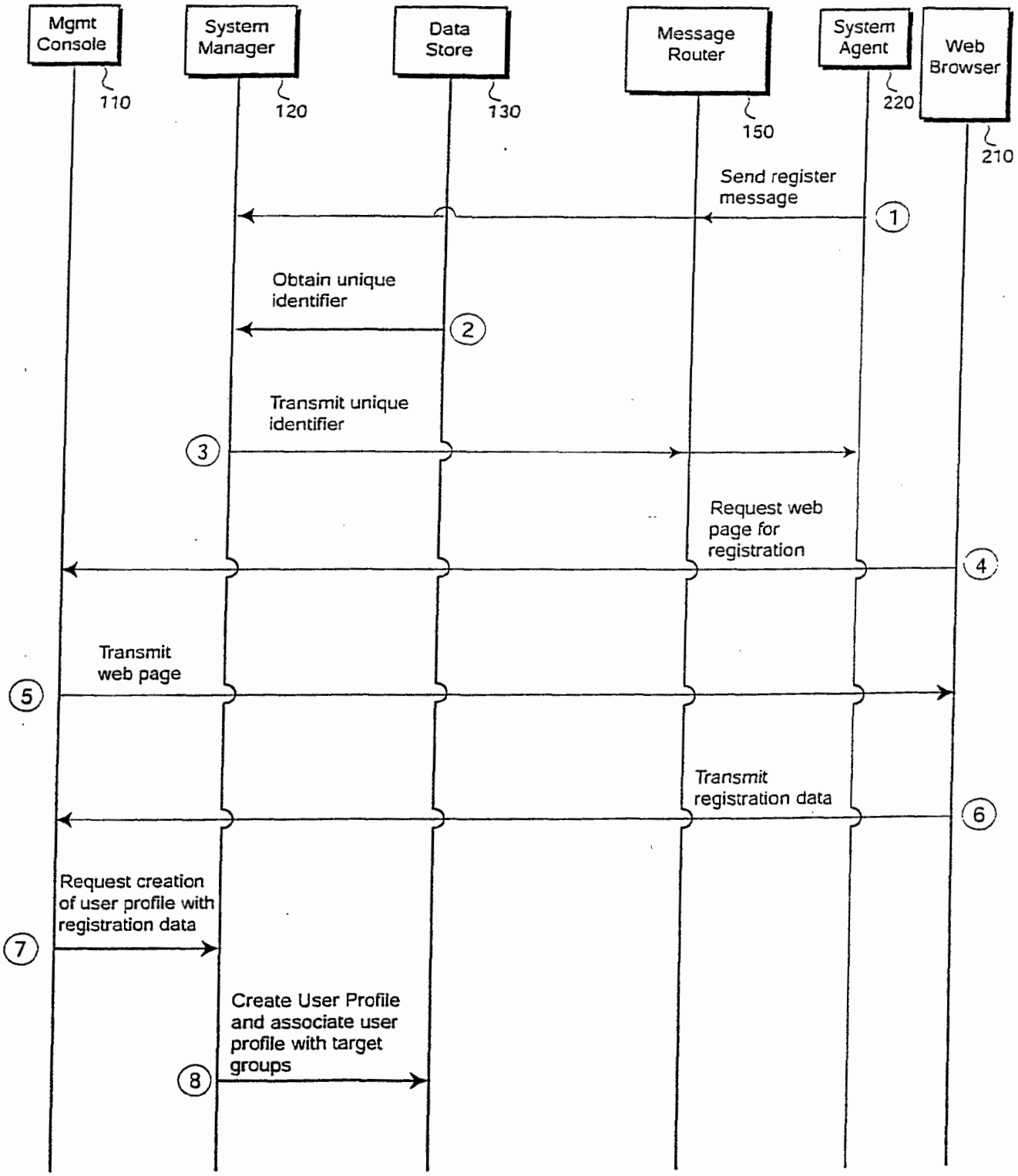

Mgmt Console
110
System Manager
120
Data Store
130
Message Router
150
System Agent
220
Web Browser
210
Send register message
1
Obtain unique identifier
2
Transmit unique identifier
3
Request web page for registration
4
Transmit web page
5
Transmit registration data
6
Request creation of user profile with registration data
7
Create User Profile and associate user profile with target groups
8

Fig. 3

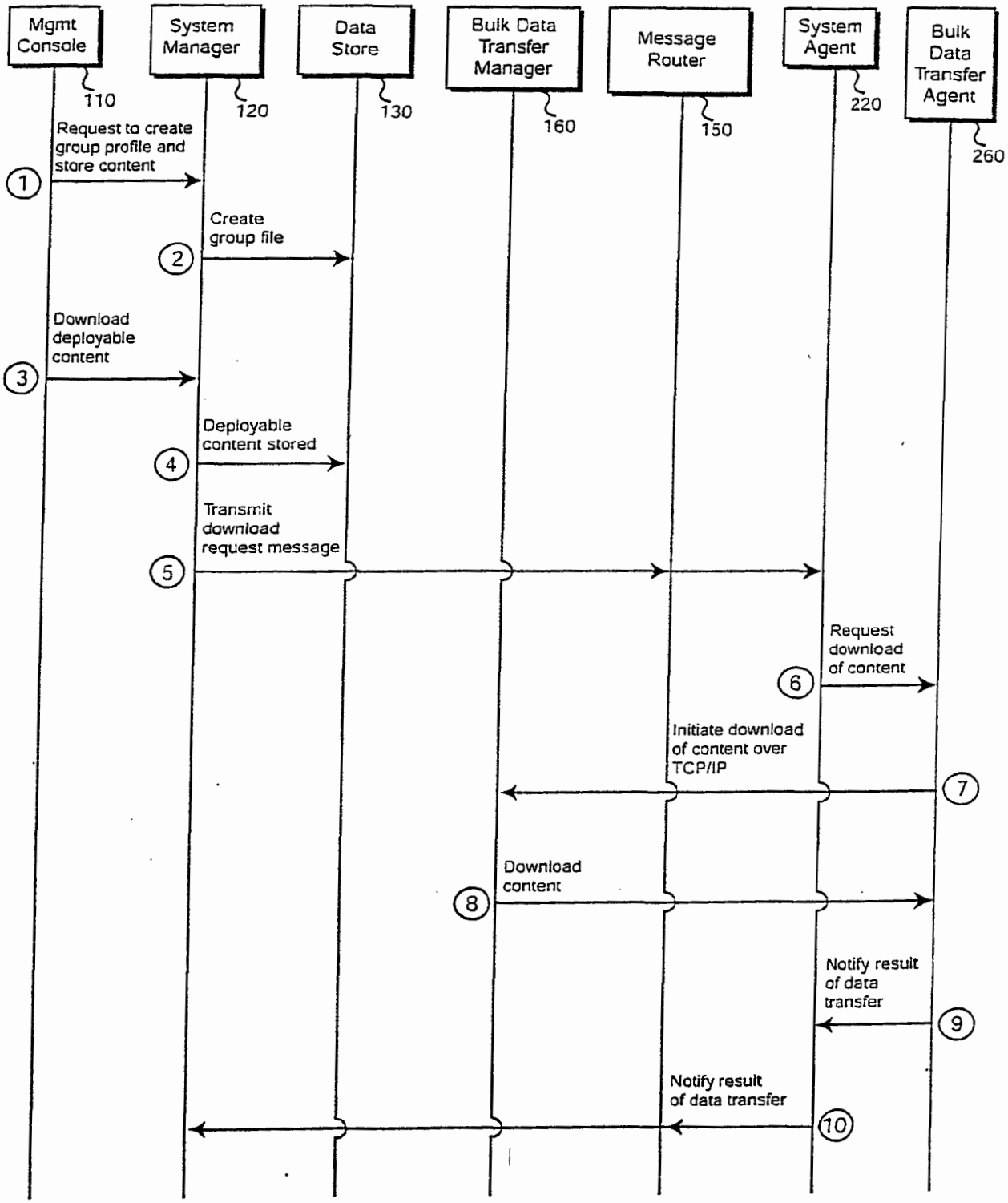

Mgmt Console
110
System Manager
120
Data Store
130
Bulk Data Transfer Manager
160
Message Router
150
System Agent
220
Bulk Data Transfer Agent
260
Request to create group profile and store content
1
Create group file
2
Download deployable content
3
Deployable content stored
4
Transmit download request message
5
Request download of content
6
Initiate download of content over TCP/IP
7
Download content
8
Notify result of data transfer
9
Notify result of data transfer
10

Fig. 4A

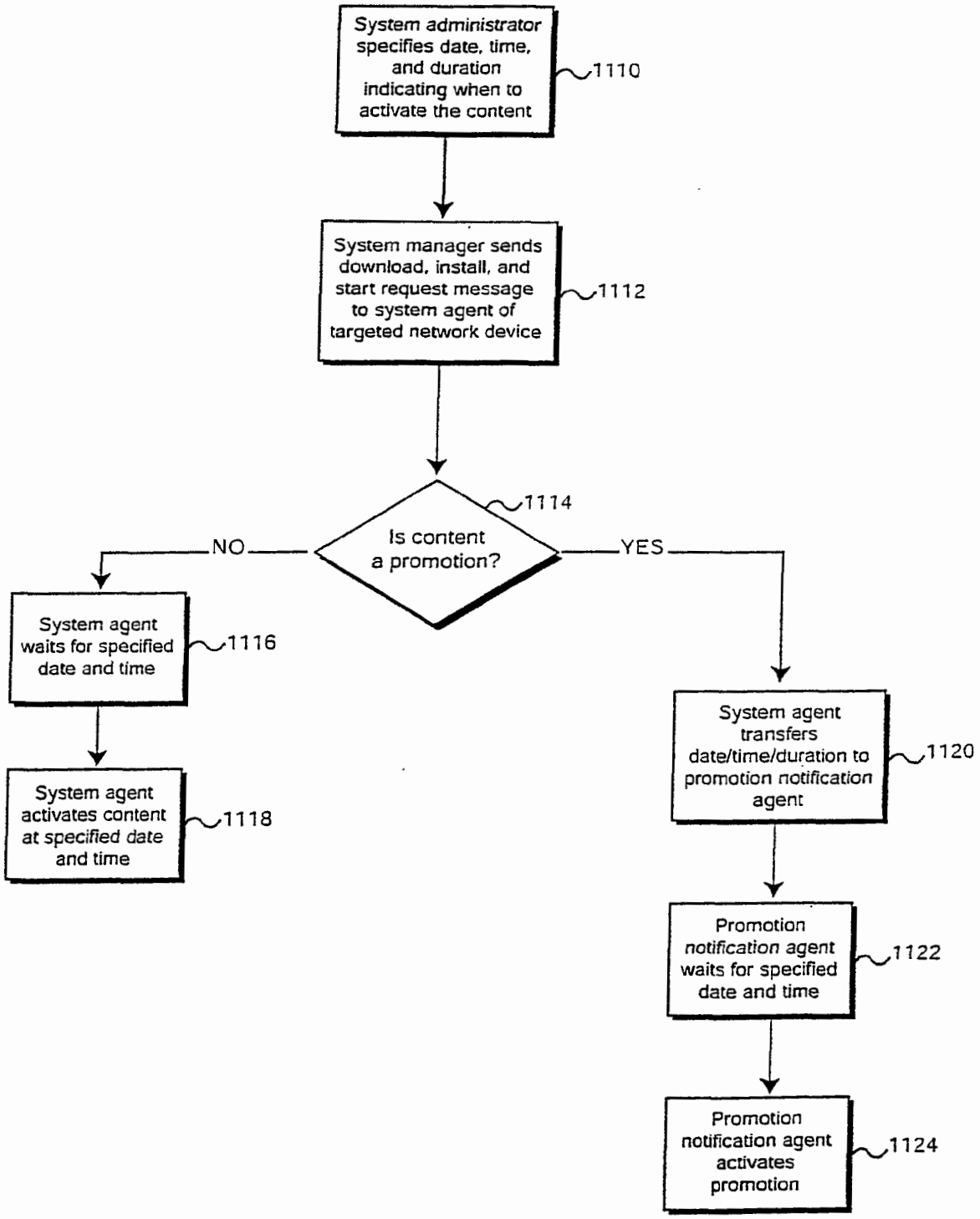

System administrator specifies date, time, and duration indicating when to activate the content
1110
System manager sends download, install, and start request message to system agent of targeted network device
1112
1114
Is content a promotion?
NO
YES
System agent waits for specified date and time
1116
System agent activates content at specified date and time
1118
System agent transfers date/time/duration to promotion notification agent
1120
Promotion notification agent waits for specified date and time
1122
Promotion notification agent activates promotion
1124

Fig. 4B

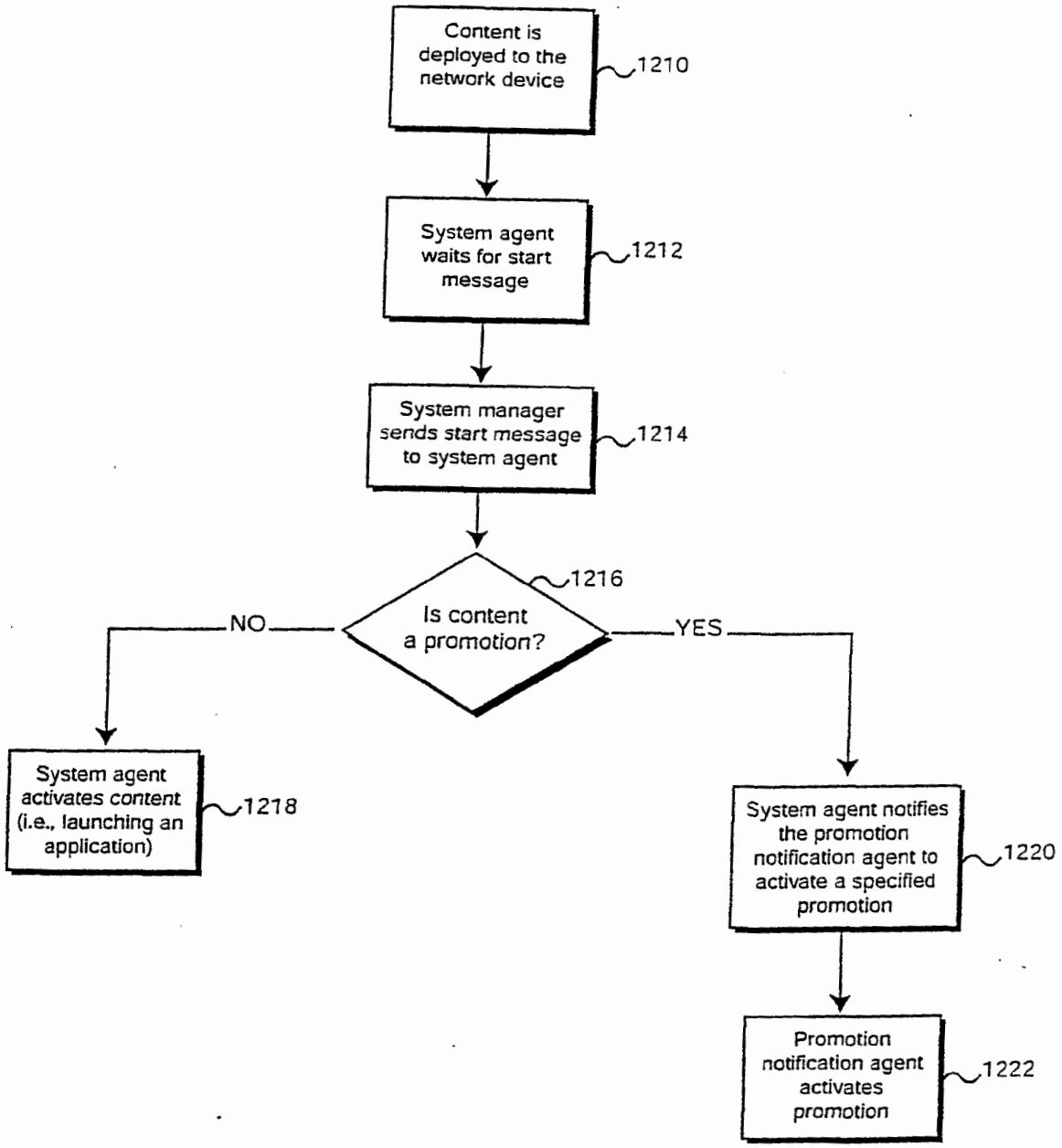

Content is deployed to the network device
1210
System agent waits for start message
1212
System manager sends start message to system agent
1214
Is content a promotion?
1216
NO
YES
System agent activates content (i.e., launching an application)
1218
System agent notifies the promotion notification agent to activate a specified promotion
1220
Promotion notification agent activates promotion
1222

Fig. 4C

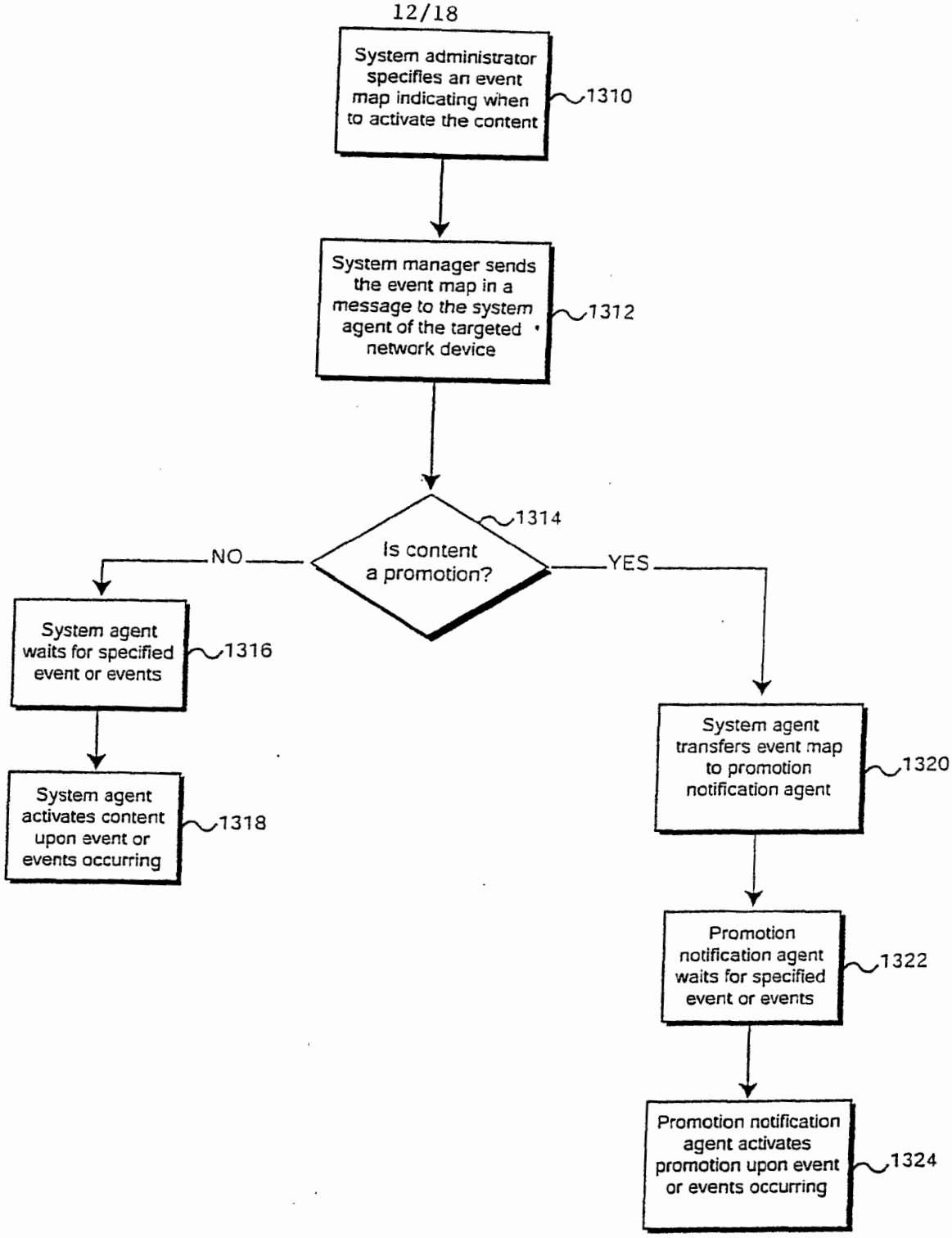

System administrator specifies an event map indicating when to activate the content
1310
System manager sends the event map in a message to the system agent of the targeted network device
1312
Is content a promotion?
1314
NO
YES
System agent waits for specified event or events
1316
System agent activates content upon event or events occurring
1318
System agent transfers event map to promotion notification agent
1320
Promotion notification agent waits for specified event or events
1322
Promotion notification agent activates promotion upon event or events occurring
1324

Fig. 4D

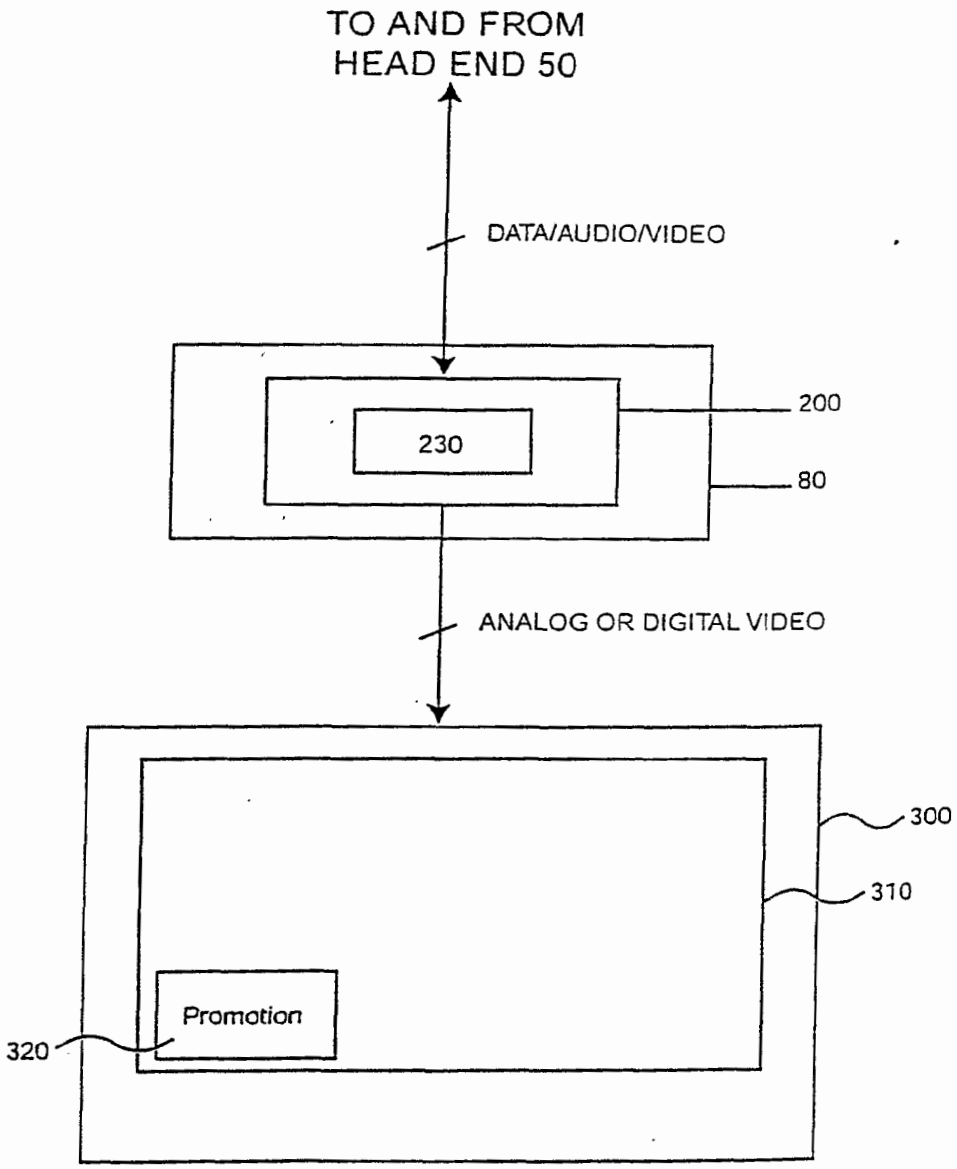

TO AND FROM
HEAD END 50
DATA/AUDIO/VIDEO
230
200
80
ANALOG OR DIGITAL VIDEO
300
310
Promotion
320

Fig. 4E

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT EventLog(Event)+>
<!ELEMENT Event(gDevice, tTime, wsEventType, wsEventData)+>
<!ELEMENT gDevice(#PCDATA)>
<!ELEMENT tTime(#PCDATA)>
<!ELEMENT wsEventType(#PCDATA)>
<!ELEMENT wsEventData(#PCDATA)>
```

Fig. 5A

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE LoggerEvents SYSTEM "LoggerEvents.dtd">
<EventLog>
        <Event>
                <gDevice>E2CDBE0-6F85-11D3-BFFB-000039ABB70C</gDevice>
                <tTime>2000/02/04 18:31:02</tTime>
                <wsEventType>Channel</wsEventType>
                <wsEventData>53</wsEventData>
        </Event>

        <Event>
                <gDevice>0000000-0000-0000-0000-000039ABB70C</gDevice>
                <tTime>2000/02/03 06:31:02</tTime>
                <wsEventType>Application</wsEventType>
                <wsEventData>Launched</wsEventData>
        </Event>
<EventLog>
```

Fig. 5B

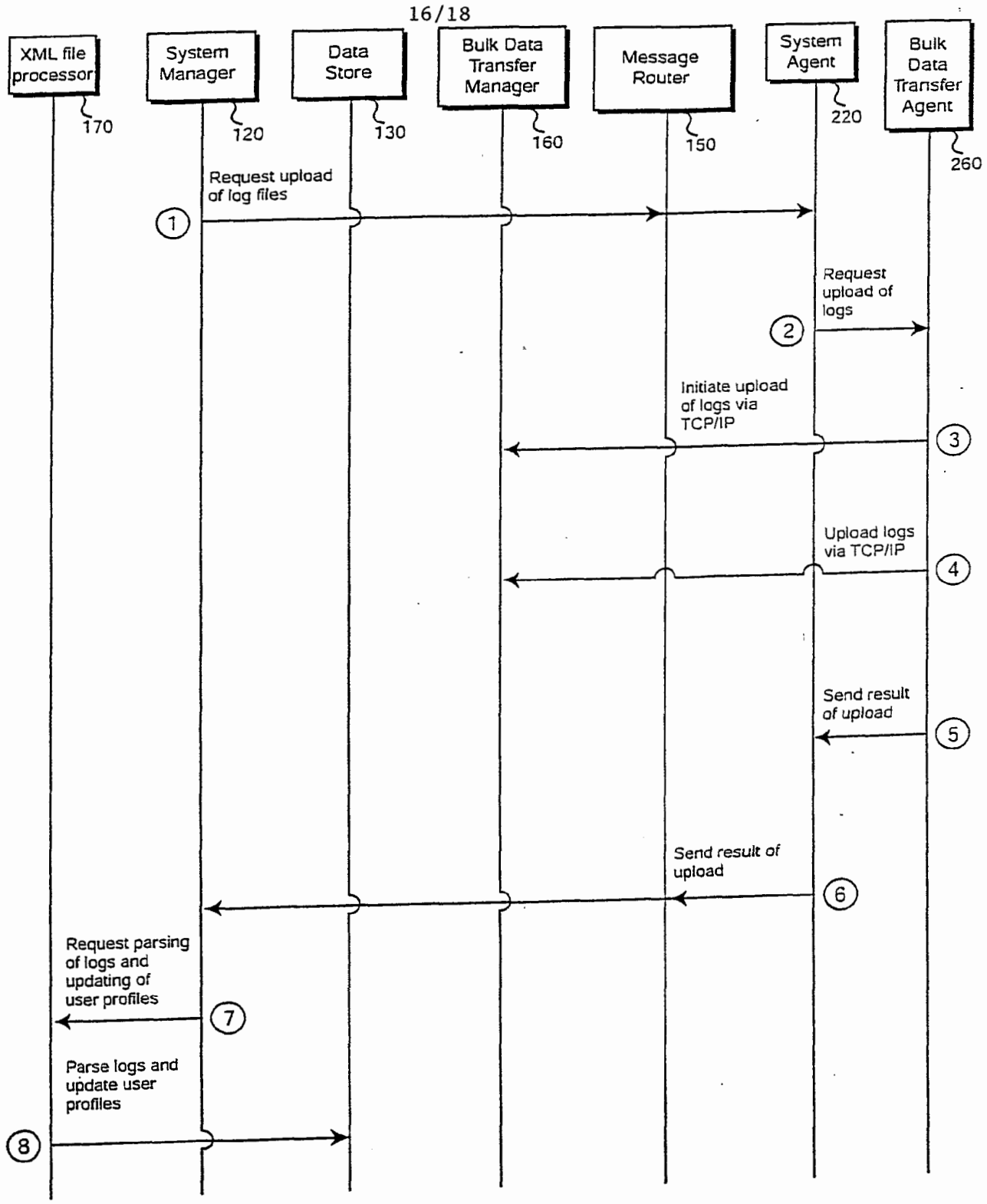

XML file processor
170
System Manager
120
Data Store
130
Bulk Data Transfer Manager
160
Message Router
150
System Agent
220
Bulk Data Transfer Agent
260
Request upload of log files
1
Request upload of logs
2
Initiate upload of logs via TCP/IP
3
Upload logs via TCP/IP
4
Send result of upload
5
Send result of upload
6
Request parsing of logs and updating of user profiles
7
Parse logs and update user profiles
8

Fig. 6

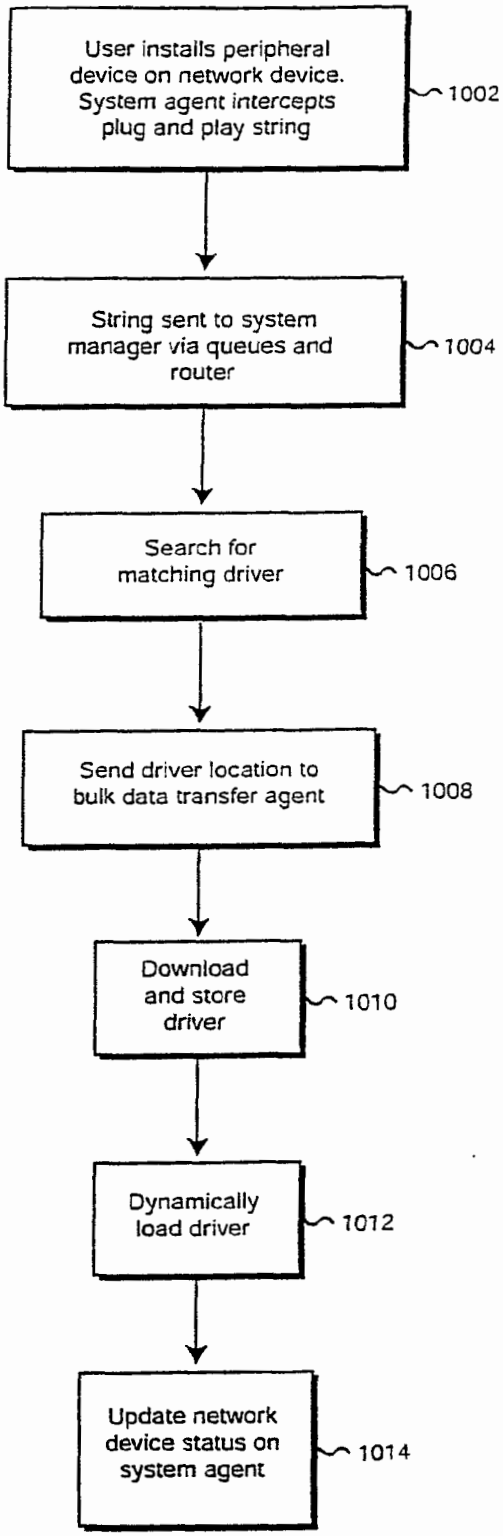

User installs peripheral device on network device. System agent intercepts plug and play string
1002
String sent to system manager via queues and router
1004
Search for matching driver
1006
Send driver location to bulk data transfer agent
1008
Download and store driver
1010
Dynamically load driver
1012
Update network device status on system agent
1014

Fig. 7

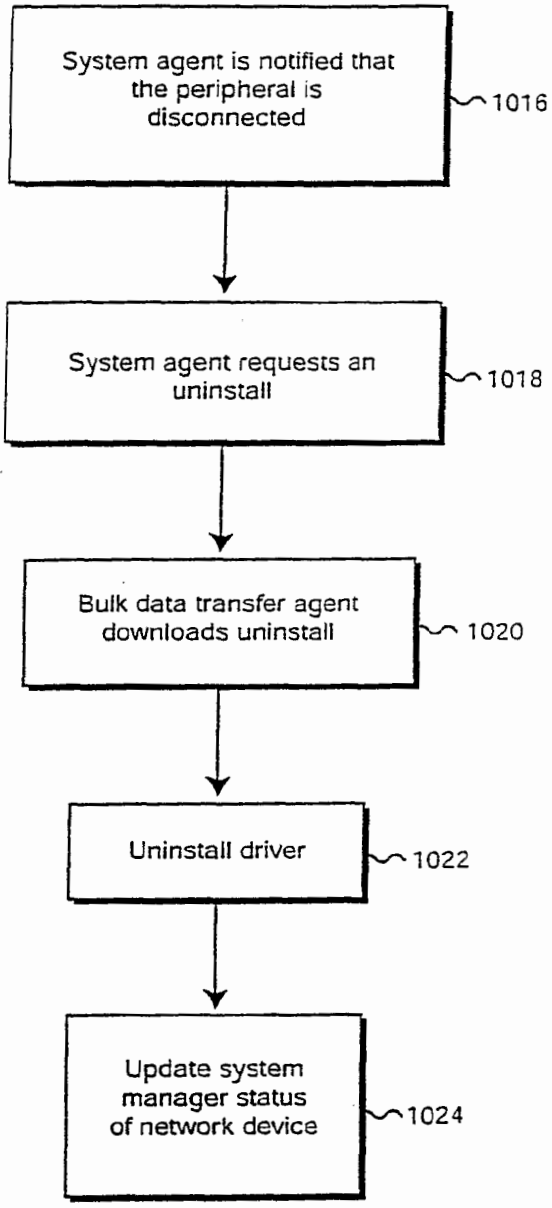

System agent is notified that the peripheral is disconnected
1016
System agent requests an uninstall
1018
Bulk data transfer agent downloads uninstall
1020
Uninstall driver
1022
Update system manager status of network device
1024

Fig. 8

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
19 May 2005 (19.05.2005)

PCT

(10) International Publication Number
WO 2005/046184 A1

(51) International Patent Classification[7]: H04L 29/12, G06F 17/30

(21) International Application Number: PCT/GB2004/004639

(22) International Filing Date: 4 November 2004 (04.11.2004)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
0325691.4 4 November 2003 (04.11.2003) GB

(71) Applicant *(for all designated States except US)*: **DOTWORLDS LTD** [GB/GB]; 4th Floor, Elvin House, Stadium Way, Wembley, Middlesex HA9 0DW (GB).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **RETKIN, Brian, Anthony** [GB/GB]; 14 Grendon Gardens, Wembley Park, Middlesex HA9 9NE (GB). **FOAKES, Simon, E.** [GB/—]; * (**).

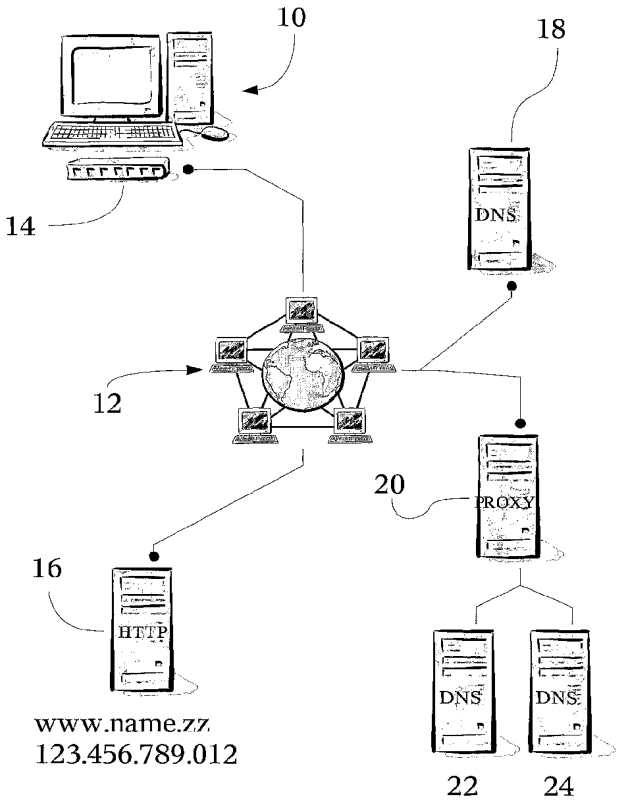
(74) Agent: **MURGITROYD & COMPANY**; 165-169 Scotland Street, Glasgow G5 8PL (GB).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM,

[Continued on next page]

(54) Title: RESOLUTION OF DOMAIN NAMES



(57) Abstract: A plugin for an Internet web browser is disclosed. The plugin operates upon detection of a DNS look-up failure to attempt a DNS look-up on an alternative name server. By providing this functionality in a browser plugin, installation and operation are transparent to the user. Moreover, the plugin operates only in the event of a look-up failure so resolution of names using a previously configured server is unaffected. The alternative name server can contain references to names within an arbitrary namespace that can be determined by the operator of the server.

WO 2005/046184 A1

ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LU, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report*

— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

Resolution of network names

This invention relates to resolution of network names.

Most users of a computer network, and in particular, the Internet, rely on symbolic names to identify hosts and domains on the network. The symbolic names must be translated into machine addresses (IP addresses, in the case of the Internet) to enable data to be routed to and from a host. The relation between symbolic names and addresses was originally maintained in a host file on each machine on the network. However, as networks grew in size, maintenance of host databases became an ever-increasing burden, so the task was entrusted to name resolution (DNS) severs that maintain a centralised database of names and addresses.

Owners of private networks have virtually unlimited control over the names that can be used in their networks. However, in the context of the Internet, domain names are normally resolved with reference to DNS servers that are maintained by an Internet service provider or a broadband provider. Such servers operate within a hierarchy of servers, ultimately controlled by The Internet Corporation for Assigned Names and Numbers (ICANN). The policies of ICANN, and national domain registers recognised by ICANN restrict the choice of names that can be obtained by users.

In principle, anyone could set up a publicly-accessible DNS server upon which they can record arbitrary names, and make the address of the server known. An Internet user could then add the server's address to the list of DNS servers accessed by their computer when a name is to be resolved. This would be quite acceptable to experienced users.

However, many users access the Internet through a dial-up connection that configures their computer's settings, including DNS server addresses, automatically using the dynamic host configuration protocol (DHCP). Such users may not be competent to or authorised to configure their computer's settings manually.

An aim of this invention is to provide a system whereby an Internet user can access a domain namespace that contains arbitrary names without the need to reconfigure their computer manually.

From a first aspect, this invention provides a plugin for a browser, which operates upon detection of a DNS look-up failure to attempt a DNS look-up on an alternative name server.

By providing this functionality in a browser plugin, installation and operation are transparent to the user. Moreover, the plugin operates only in the event of a look-up failure so resolution of names using a previously configured server is unaffected.

The plugin may operate by detecting when the browser is about to display an error to inform a user that a referenced name cannot be found. It may then perform a further name lookup before the error is displayed. This can be achieved readily by providing a script to execute in the context of the browser.

The plugin may be suitable for installation in a web browser by being downloaded from a remote web site. Preferably, the plugin is packaged with an installer that installs the plugin with a minimum of user intervention.

To ensure that the plugin always directs its look-ups to a valid alternative DNS server, it may be operative to contact a remote server to obtain data relating to the alternative name server. For example, the data may be obtained each time the operating system is started or each time the plugin is initiated.

For example, the plugin may operate by configuring proxy server settings of a web browser in which it is installed. A plugin embodying the invention may communicate with the alternative DNS sever through a proxy server.

From a second aspect, this invention provides a web browser that has a plugin that operates upon detection of a DNS look-up failure to attempt a DNS look-up on an alternative name server. The browser may be configured by the plugin to refer DNS look-ups to an alternative server through configuration of its proxy settings. This is advantageous because the proxy settings of many browsers can be configured by configuration of the contents of a settings file.

From a third aspect, this invention provides a method of resolving a network name, the method comprising:

using a browser to make a request for a resource with reference to a host by name;

attempting to resolve the name by reference to a first DNS system; and

in the event that the browser receives a negative response from the DNS system, it executes a plugin that attempts to resolve the name from an alternative server.

An embodiment of the invention will now be described in detail, by way of example, and with reference to the accompanying drawings, in which:

Figure 1 is a schematic diagram of a computer operating a browser embodying the invention to communicate with Internet locations.

To place the embodiment in context, a brief overview of the operation of a computer running an Internet browser will now be described.

A browser is normally an unprivileged user program executing on a networked client computer 10. In this example, the client computer 10 is a stand-alone computer that is connected to the Internet 12 through a dial-up device 14 such as ISDN or DSL terminal adapter or a modem. The browser can receive a request for a resource (e.g., a web page or a file on an ftp site), download that resource from a remote server 16, and do something with it on the client computer 10, such as render it, if it is a web page, or save it if it is a binary file.

A resource is identified by a string called a uniform resource location (URL) that identifies the host upon which the resource is to be found (in this example, the fictitious www.name.zz) and the location of the resource on the host. The host may be identified by its IP address (shown, again fictitiously, as 123.456.789.012) in dotted-decimal form, but it is more usually identified by a symbolic name. In the latter case, the browser must translate the symbolic name into an IP address, because this address is required to address the remote host computer on which the resource is located.

To resolve an IP address, the browser makes a request to a resolver service of the operating system under which it is executing. Typically, the resolver service will first look in a local database of host names, and if the name cannot be found, it passes the request to a remote DNS server 18. The address of the remote DNS server is configured within the operating system either statically by a system administrator or dynamically using DHCP, the latter being common on computers that access the Internet by way of a dial-up connection. (In fact, most operating systems allow several DNS servers to be specified that will be referred to in turn.) There can be two outcomes of the DNS look-up. First, the look-up can be successful, in which case, the resolver will return an IP address to the browser. This address can then replace the symbolic name within the URL. Alternatively, the name look-up may fail, in which case, the resolver will return a failure code to the browser.

In the event that the DNS look-up fails, a conventional web browser displays an error message or error page to inform the user that a request for a URL cannot be met. The behaviour and configuration of a browser embodying the invention will now be

described. This embodiment will be described in the context of a computer running the Microsoft Windows operating system executing the Microsoft Internet Explorer web browser. However, it will be understood that the same principles can be applied to any other browser that can support plugins of a sufficient degree of complexity.

First, in order to prepare the browser for operation in accordance with the invention, a plugin must first be installed. This is achieved by first visiting a web site provided by the operator of a domain registration and domain name resolution service. Note that the installation need be performed only once by each user.

The user clicks on a link to download the plugin. The plugin is downloaded onto the user's computer, where the following actions occur:

a) Five files are installed that enable the domain name resolution system to operate. These include two dynamically linked libraries in DLL files (**dotworlds.dll** and **dotdetect.dll**), two executables in EXE files (**execdll.exe** and **kill.exe**) and a proxy configuration script in a PAC file (**dotworlds.pac**).

b) The dynamically linked libraries, the executables and the proxy configuration script are installed in the standard directory for such files in the Microsoft Windows file system.

c) The following items are written to the Microsoft Windows registry: Run key entry to run **dotdetect.exe**, which starts **execdll.exe**, plugin version entries and uninstall configurations.

The registry key for running the plugin at start-up is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersio
n\Run
```

The registry key for adding the plugin to add/remove programs feature of Microsoft windows:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersio
n\Uninstall\Dotworlds
```

d) An uninstall system for the plugin package is also installed (unsdot.exe in Program Files stabdar folder, dotworlds sub folder).

The file **dotworlds.dll** is a function library that exports functions required by the plugin. The file **dotdetect.dll** detects the function library and runs the **execdll.exe** file. The executable **execdll.exe** serves to detect an Internet connection, configure the connection and update the proxy configuration file. The executable **kill.exe** is used during the uninstall process. The purpose of the proxy configuration file is to offer proxy configurations for the browser.

In the context of Microsoft Internet Explorer web browser, a user can configure separate proxy settings for use with each Internet dial-up account that is configured in the operating system. After installation, the plugin configures the user's current connection to use the proxy settings specified in the proxy configuration file. This can be achieved by activating the automatic configuration script option in Internet Explorer. Specifically, this is done by functions in **dotdetect.dll**, running **execdll.exe** during system start-up

To ensure use of the automatic configuration script on any Internet connection, and to ensure that the information in the proxy configuration file is up to date, the library **dotdetect.dll** is run whenever Microsoft Windows is started, this activating the **execdll.exe** executable. The executable **execdll.exe** starts a process that runs in the background and waits until an Internet connection is detected. Once this occurs, the process configures the current connection (from within the browser) to use the automatic configuration script (written to the proxy configuration file). The process file also checks the IP/domain configurations required by the proxy configuration file by visiting a server provided by the operator of the domain registry, from where it downloads a remote configuration file. The information in the remote configuration file is then re-

written to the proxy configuration file, so that the configuration script always uses up-to-date proxy IP information. This allows changes in the operator's systems to be communicated transparently to the user's computer. Having run once, the process then shuts down.

Once the above installation procedures have been performed, operation of the browser can continue.

Now consider what occurs if the remote DNS server 18 is unable to find the name requested (as would be the case with domains registered by the operator of the domain registry). The configuration script is used for every name lookup, but it tells the browser to use its normal configurations first, and only the alternative proxy configurations in case of error. Therefore, before the browser informs the user with of an error condition (having been configured to use the automatic configuration script) the browser uses the information contained in the proxy configuration script (found in the proxy configuration file) to try to resolve the URL using an alternative IP address, in this case, pointing to a proxy server 20 run by the operator of the domain registry. This proxy server is in turn linked in to two proprietary DNS servers 22, 24, which have been configured to resolve the domains offered by the operator of the registry, optionally along with all other conventional domain names. The browser makes the URL request to the proxy server 20, which in turn resolves the URL (if it can) using the proprietary DNS servers 22, 24, returning the correct IP address for the domain to the browser. This means that the operator's domain names are true domain names and not sub-domains of other ICANN domains. In this example, **.zz** is not a valid extension within the ICANN naming system, but can, should a user wish, be registered within the proprietary DNS server.

Thus, a domain registrar can install a DNS server, with an optional redundant second server. These may be made directly available over the Internet or may be protected by a proxy server.

Claims

1. A plugin for a browser, which plugin operates upon detection of a DNS look-up failure to attempt a DNS look-up on an alternative name server.

2. A plugin according to claim 1 in that operates by detecting when the browser is about to display an error to inform a user that a referenced name cannot be found.

3. A plugin according to claim 2 in which the plugins perform a further name lookup before the error is displayed.

4. A plugin according to claim 2 or claim 3 in which This can be achieved readily by providing a script to execute in the context of the browser.

5. A plugin according to any preceding claim that is suitable for installation in a web browser by being downloaded from a remote web site.

6. A plugin according to claim 5 that is packaged with an installer that installs the plugin with a minimum of user intervention.

7. A plugin according to any preceding claim that is operative to contact a remote server to obtain data relating to the alternative name server.

8. A plugin according to claim 7 in which the data may be obtained each time the operating system is started or each time the plugin is initiated.

9. A plugin according to any preceding claim that operates by configuring proxy server settings of a web browser in which it is installed.

10. A plugin according to any preceding claim that communicates with the alternative DNS sever through a proxy server.

11. A plugin for a web browser substantially as herein described with reference to the accompanying drawing.

12. A web browser that has a plugin that operates upon detection of a DNS look-up failure to attempt a DNS look-up on an alternative name server.

13. A web browser according to claim 12 that is configured by the plugin to refer DNS look-ups to an alternative server through configuration of its proxy settings.

14. A method of resolving a network name, the method comprising: using a browser to make a request for a resource with reference to a host by name; attempting to resolve the name by reference to a first DNS system; and in the event that the browser receives a negative response from the DNS system, it executes a configuration file that attempts to resolve the name from an alternative server.

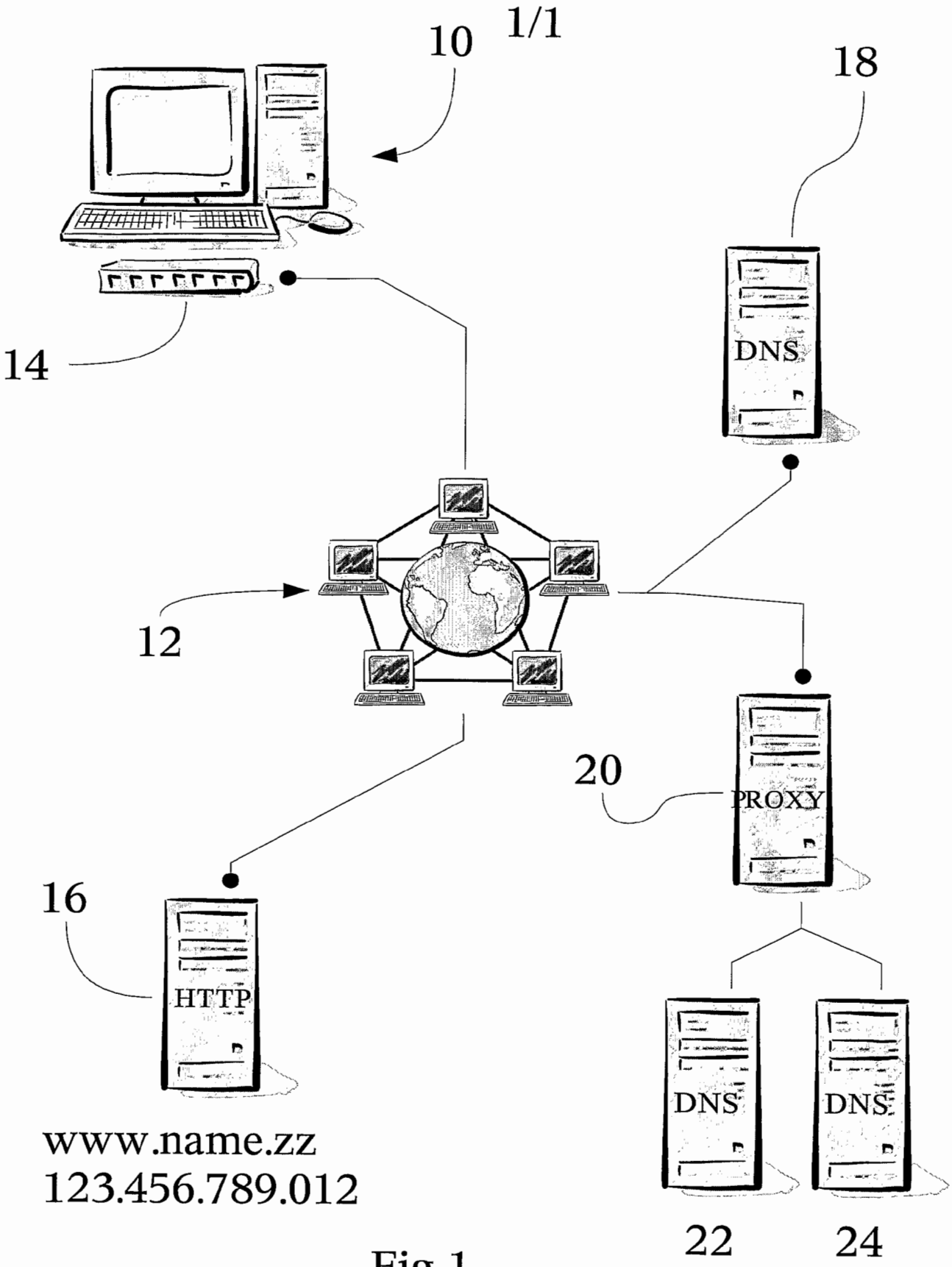

1/1
10
18
DNS
14
12
20
PROXY
16
HTTP
www.name.zz
123.456.789.012
DNS
DNS
22
24

Fig 1

INTERNATIONAL SEARCH REPORT

Int Application No
PCT/GB2004/004639

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 H04L29/12 G06F17/30

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 H04L G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, INSPEC, IBM-TDB, PAJ, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	MOCKAPETRIS P: "RFC 1034 Domain Names - Concepts and Facilities" NETWORK WORKING GROUP REQUEST FOR COMMENTS, XX, XX, November 1987 (1987-11), pages 1-55, XP002278155	1,3-14
A	page 6 - page 7, paragraph 3 page 15, last paragraph - page 16 page 22 - page 25	2
A	US 2002/059396 A1 (HOLZER DIPL.-ING. RENE ET AL) 16 May 2002 (2002-05-16) abstract page 1, paragraph 13 page 2, paragraph 18 - paragraph 21	2,3

-/--

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
14 March 2005	18/03/2005

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Riegler, J

Form PCT/ISA/210 (second sheet) (January 2004)

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	WO 02/13057 A (SHARINGA NETWORKS INC; LEISHMAN, CHRISTOPHER; LOW, SYDNEY, GORDON; WAL) 14 February 2002 (2002-02-14) abstract page 4, line 11 - page 5, line 15	1,12,14
A	ANONYMOUS URL - HTTP://DEVELOPER.NETSCAPE.COM/DOCS/MANUALS /COMMUNICATOR/PLUGINDEX.HTM DWWW- 1997-01-15: "Plug-in Guide, Chapter 1: Plug-in Basics" NETSCAPE COMMUNICATOR MANUAL, 15 January 1997 (1997-01-15), XP002252262 the whole document	1-14

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No
PCT/GB2004/004639

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2002059396 A1	16-05-2002	DE 10048112 A1	18-04-2002
		EP 1193941 A2	03-04-2002
WO 0213057 A	14-02-2002	WO 0213057 A1	14-02-2002
		AU 7740001 A	18-02-2002

Form PCT/ISA/210 (patent family annex) (January 2004)

Electronic Acknowledgement Receipt

EFS ID:	2356409
Application Number:	11396251
International Application Number:	
Confirmation Number:	1129
Title of Invention:	Asynchronous messaging using a node specialization architecture in the dynamic routing network
First Named Inventor/Applicant Name:	Timothy Tuttle
Customer Number:	26111
Filer:	Glenn J. Perry
Filer Authorized By:	
Attorney Docket Number:	2222.775000E
Receipt Date:	23-OCT-2007
Filing Date:	30-MAR-2006
Time Stamp:	16:33:31
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes) /Message Digest	Multi Part /.zip	Pages (if appl.)
1	Information Disclosure Statement Letter	2222775000E_TL.pdf	54718 3a147d9ebf999168c8ff9116e68eb98de0fdcb05	no	2
Warnings:					