

Patent Description of Drawings and Detailed Description with a Focus on the described

"User Specific Operating Systems": Verbatim Excerpted from US 7,155,451B1

"Automated browsing system for publishers and users on networks serving internet and remote devices"

Invented in 1996 by Damon C. Torres, Assigned to RoboCast, Inc.

Focus on the described "User Specific Operating Systems" and definition of a "node" that goes beyond mere content, excerpted here and found below:

"Thus, a show structure may include one or more paths that can be presented sequentially and/or concurrently. It is noted that a path may include portions that is presented sequentially and other portions that are presented concurrently. Furthermore, one node to many nodes paths and many nodes to one node paths can also define a show structure. Every time that a node is detected the corresponding content of that node is presented to the user's terminal regardless of whether a plurality of nodes have been detected at the same time or separately.

It is noted that the process that creates and/or organizes and manages the arrangement of these nodes and ultimately the corresponding resources, **may be implemented as a user specific operating system in accordance with one embodiment of the invention.** The operating system then allows the paths to be selected and arranged in accordance with interactive user specifications or in accordance with predefined criteria. Thus, the operating system, among other things, organizes and arranges the available resources as identified by available nodes; allows end users to select and interact with files and applications; and make certain files and or resources available to be shared by all the other resources. For example, during a presentation of a show, certain sound effects and visual effects may be shared by all resources that may have use for such effects. Such an operating system can be employed for many applications described herein in accordance with various embodiments of the invention. A show structure may be presented on a server that can be accessed by a user who visits a corresponding site on the server. This component on the server is described below with reference to FIG. 3, and is referred to as the program Robosites™. Robosites may perform the actual transfer operations to obtain the pages from other web sites.

In addition to RoboSurf and Robosites, the exemplary system includes RoboGuide™, described below with reference to FIG. 5, a program that allows several users to access a prepared show structure of nodes or links and have it customized to their respective preferences. The system also includes a program referred to as RoboPublisher™, described below with reference to FIG. 6, which functions as a publishing software so as to help a publisher to prepare show structures. The entire suite of programs is referred to below as RoboCast..."

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram which illustrates the Internet and the World Wide Web environment in which the invention operates.

FIG. 2A is a block diagram of a hardware system for utilizing the present software invention.

FIGS. 2B–2F is an illustration of an exemplary node arrangements that define various show structures in accordance with one embodiment of the present invention.

FIG. 3 is a flow-chart diagram which is useful for describing a software application which runs on a server and sequences and displays URLs and predefined structure automatically in accordance with timer settings while monitoring the navigation decisions of the user, if any.

FIG. 4 is a flow-chart diagram of a client-side software application or Apple, which processes URLs and predefined structures automatically while monitoring the timer settings and navigation decisions of the user.

FIG. 5 is a flow-chart diagram which illustrates the process by which initial structures of content are personalized and delivered to the user.

FIG. 6 is a flow-chart diagram which illustrates a method of formatting individual resources and collections of resources for use with the resource scheduling and organizing software shown in FIGS. 3 through 5.

FIG. 6A is a graphical depiction of a process for editing a structure of nodes.

FIG. 7 is a computer screen diagram of an exemplary user interface for the invention that is useful for describing features of the invention.

DETAILED DESCRIPTION

FIG. 1 is a simplified block diagram which is useful for describing the connectivity of users and content providers on the Internet. Briefly, the Internet includes a large collection of content providers **10** which may be accessed by the users **12**. Typically, a user accesses content on a "server" by establishing a data communications connection to the server through the network of computers **14, 16** defined by the Internet. Each server **10** operates a "web site" which includes data files that contain text, images, audio or video content. The web site typically includes a group of linked HTML documents or pages that contain links to the data files. A user **12** may access the content in a file by simply selecting one of these links. The file may be transferred to a cache in the user's computer and presented using the browser or the content may be provided directly by the server **10** to the user **12** as an encoded data stream (e.g. an audio program) which is decoded and played out by the user's computer **12**.

Content accessed by the browser may not be in a file on a remote computer but may be generated by the remote computer and transmitted to the user directly. Much of the data that may be accessed by a browser on a remote computer is classified by Multipurpose Internet Mail Extension (MIME) type. One MIME type is an HTML file, another is a JPEG file, a still picture that is encoded according to the standard developed by the Joint Picture Experts Group. Mime types can also be proprietary data. In this instance, the browser may need to be configured with a "plug-in" program in order to decode and present the proprietary data. An example of such a plug-in is the Macromedia™ flash technology which allows streaming animation to be displayed with accompanying audio. For more information on MIME types, see Netscape's User Handbook at

<http://home.netscape.com/eng/mozilla/1.1/handbook/docs/answers.html>, HTTP (HyperText Transfer Protocol) servers use HTML formatting. Netscape software has the built-in capability to read HTML-formatted pages (as well as the GIF, JPEG, and XBM graphic file formats). Netscape can interpret many other file formats with the help of external helper applications. To accommodate the file formats requiring helper applications, Netscape keeps a mapping of file formats to helper applications. MIME (Multipurpose Internet Mail Extensions) is a standardized method for organizing divergent file formats. The method organizes file formats according to the file's MIME type. When Netscape software retrieves a file from a server, the server provides the MIME type of the file. Netscape uses the MIME type to establish whether the file format can be read by the software's built-in capabilities or, if not, whether a suitable helper application is available to read the file.

For servers that do not provide a MIME type with a file, Netscape interprets the file's extension (a suffix appended to a file name). For example, the .HTML extension in the file name index.html suggests a file in the HTML format. Likewise, a zip extension suggests a compressed file, an .rtf extension suggests a file in the Rich Text Format, and so on.

The Internet content relating to a show structure, in accordance with one embodiment of the invention, may be implemented as a new MIME type. Thus a show structure may be created by a show originator who intends to present a corresponding arrangement of resources to a user at a specifiable rate. The show structure may be transmitted to a plurality of users via the Internet. Thus, in order to ensure the integrity of a show structure a security arrangement in accordance with one embodiment of the invention is provided as described hereinafter.

Thus, the software that creates the show structure allows the show originator, to encode the identity of its creator. Furthermore, a mechanism is employed to automatically verify that the show structure comprising the show has not been tampered with. In one exemplary implementation, this is accomplished by using cryptographic techniques; for example the Digital Signature Initiative (c.f.

<http://w3c.org/Dsig/Overview.html>). The identity of the show originator is thus represented by a digital signature. The security status of the currently selected show structure is then visible at all times to the

user. If a user chooses to edit a sequence of a show, the previously attached signature will be discarded and the user will be made aware of its new unsigned status.

Today it is taken for granted that, to access services, information and entertainment on the Web, a user must make certain choices, position the mouse pointer and click the mouse buttons appropriately to indicate the decisions. For example, in a typical Web access, these selections are made at a rate of approximately once per minute. The result of each mouse click is another page and another set of decisions. Typically, a user is expected to click one of the several dozen links provided on any given page, (see, for example, cnn.com). Alternatively, a user may type in a URL or access a bookmark to visit a favorite search engine. In addition, a user may access a collection of "channels" of "push content" via Netscape Communicator's Netcaster or Microsoft Internet Explorer's Active Desktop and sit back for a few seconds to view an animated presentation of a specific publisher's offerings. The user may even choose a headline of a story presented in Pointcast's screen saver to receive more information about that topic grouping within the application, or to transfer to a relevant website. All of these methods typically require a significant amount of user effort and decision-making to drive the web surfing experience. Although the users and servers are shown as separate machines in FIG. 1, it is well known that a computer being used as a server may host users and may also serve as a conduit for connecting a user to a remote server.

FIG. 2 is a block diagram which illustrates the basic layout of a typical user's computer, be it a personal computer (PC) or a network computer (NC), as typified by webtv™ or a Java Station™. The computer may also be a dedicated workstation coupled to a local area network or a terminal connected to a mainframe computer which is, in turn, coupled to the Internet or other communication networks. The user requires no special hardware to use this invention. No additional cards with TV tuners are required, but they can be used as an enhancement. This diagram illustrates one possible environment for the invention and should not be interpreted as limiting the invention. It is contemplated that the invention may employ user terminals which may have radically different hardware, including but not limited to hand held devices, public kiosks, dumb terminals, set-top boxes, TV sets, and even "credit card-sized" configurations.

The exemplary system includes a microprocessor **32**, a random access memory (RAM) **34**, a memory management circuit, a disk drive **36**, a floppy disk drive **37**, a read only memory (ROM) **33** which includes the basic I/O system (BIOS) for the computer system, a CD ROM drive **42**, a modem **50**, a digital signal processor **43**, a keyboard controller **38**, a mouse controller **39**, a video controller **40**, and an audio controller **41** which are all connected by a system bus **31**. A keyboard **22**, a mouse **23** and a graphic display monitor **24** are connected to the keyboard controller **38**, mouse controller **39**, and video controller **40**. Stereo speakers **25A** and **25B** are connected to the audio controller **41**. In the exemplary system, the connection to the on-line-service provider **16** (shown in FIG. 1) is made through the modem **50** via a telephone line **56**.

As shown in FIG. 2, the memory **34** includes an operating system **60**, for example, Windows 95™; a browser **62** for example, Netscape Navigator™ 4.0; and a cache memory **63**. The hard disk **36** is the main data storage device for the system shown in FIG. 1. It includes the software needed to run the browser and libraries of software used by the operating system to control the operation of all of the components that are connected to the system bus **31**. The hard disk **36** also includes a disk cache (not shown) for the browser. When data on a remote computer is accessed by the browser, it may be transferred by Hypertext Transfer Protocol (HTTP) or File Transfer Protocol (FTP) and stored in this disk cache temporarily. The component of the invention which is used on the user computer **12** may be, for example, a browser plug in. While, in this example, the user component is a browser plug-in, it is contemplated that it may be an Applet which runs on a virtual computer defined by the browser, it may also be a stand-alone program, which includes the functionality that is typically provided by a browser, or it may be a distributed application. The user component is described below with reference to FIG. 4 and is referred to as the program RoboSurf™. The RoboSurf program allows a user to view a show having content that corresponds to a show structure as will be explained in more detail later.

It is noted that in accordance with one embodiment of the invention the nodes that define a show structure or a structure may be arranged as a multidimensional sequence. FIG. 2b illustrates an array **300** of nodes that may be available for a multimedia presentation or show. Each node identifies an available resource and the time the resource may be presented to the user. As illustrated in FIG. 2B nodes **302**, **304**, **306**, **308**, **310** and **312** are available for presentation in accordance with an exemplary arrangement. Each node in the array is accessible from the other nodes in the array. For a typical show or presentations, an arrangement of these nodes or a portion of these nodes is selected for presentation as described in reference with FIGS. 2c through 2f. Thus, a show structure is defined by one or more paths that are spanned through these nodes.

FIG. 2 c illustrates a show structure that simultaneously spans multiple paths from the same starting node such as node **310** to each one of the remaining nodes **312, 302, 304, 306** and **308**. Thus, a show in accordance with the structure presented in FIG. 2 c simultaneously presents the contents identified by nodes **312, 302, 304, 306** and **308**, after the content identified by node **310** is presented.

Alternatively, as illustrated in FIGS. 2 d-1 a show structure can span a path with a length of four beginning at node **308** and traversing to nodes **312, 302** and **306**. The state map of this path is illustrated in FIGS. 2 d-2. Thus, a show in accordance with the structure presented in FIG. 2 d-1 first presents the contents identified by node **308**, followed by contents identified by nodes **312, 302** and **306**.

In a still alternative arrangement, a show structure can be created that also spans multiple paths concurrently, as illustrated in FIG. 2 e. For example a first path spans nodes **308, 302, 306** and **310**. The second path spans nodes **312, 306, 304** and **310**. As illustrated nodes **302, 306** and **310** are spanned simultaneously with nodes **306, 304** and **310**. Thus a show in accordance with the structure presented in FIG. 2 e first presents the contents identified by node **310**, followed by the contents identified by node **312**. Thereafter, the show structure presents contents identified in nodes **302**, and **306** at the same time that it presents the contents identified in nodes **306** and **304**. Finally, the structure presents contents identified in node **310**.

Thus, a show structure may include one or more paths that can be presented sequentially and/or concurrently. It is noted that a path may include portions that is presented sequentially and other portions that are presented concurrently. Furthermore, one node to many nodes paths and many nodes to one node paths can also define a show structure. Every time that a node is detected the corresponding content of that node is presented to the user's terminal regardless of whether a plurality of nodes have been detected at the same time or separately.

It is noted that the process that creates and/or organizes and manages the arrangement of these nodes and ultimately the corresponding resources, may be implemented as a user specific operating system in accordance with one embodiment of the invention. The operating system then allows the paths to be selected and arranged in accordance with interactive user specifications or in accordance with predefined criteria. Thus, the operating system, among other things, organizes and arranges the available resources as identified by available nodes; allows end users to select and interact with files and applications; and make certain files and or resources available to be shared by all the other resources.

For example, during a presentation of a show, certain sound effects and visual effects may be shared by all resources that may have use for such effects. Such an operating system can be employed for many applications described herein in accordance with various embodiments of the invention.

A show structure may be presented on a server that can be accessed by a user who visits a corresponding site on the server. This component on the server is described below with reference to FIG. 3, and is referred to as the program Robosites™. Robosites may perform the actual transfer operations to obtain the pages from other web sites.

In addition to RoboSurf and Robosites, the exemplary system includes RoboGuide™, described below with reference to FIG. 5, a program that allows several users to access a prepared show structure of nodes or links and have it customized to their respective preferences. The system also includes a program referred to as RoboPublisher™, described below with reference to FIG. 6, which functions as a publishing software so as to help a publisher to prepare show structures. The entire suite of programs is referred to below as RoboCast.

Referring again to FIG. 3 a flow-chart diagram which illustrates the operation of the server-side application, Robosites is explained in more detail. The first two steps in FIG. 3 represent activities occurring at a user's computer **12**. At step **70**, the user starts to access the Internet or other network via a Personal Computer (PC) a Network Computer (NC), Web-Enabled TV Set, or other similar device.

At step **72**, the user encounters and selects a URL, for example, within a screen saver, an e-mail message, or a website, which is designated as an entry node in a show structure of resources. The URL may be so designated by its association within a list generated by a publisher, by results that are automatically generated by a search engine or because it is the first URL in a show structure of URL's that was manually collected by the user. The server stores a plurality of show structures each structure including nodes that represent a resource on the Internet. Thus, each structure represents a show that is available for the user's preview.

Accessing the resources referred to by the URL initiates a structure of events in the application Robosites which is running on server **10**, shown in FIG. 1. At step **74**, the server parses the request into task sets, initiating the look-up of the resources indicated by the URL. This step gathers the relevant data and custom packages it for the browser **62**, shown in FIG. 2. As part of its custom packaging application Robosites provides a set of contents that can be displayed on the user's display terminal in the form of a

persistent content window. Therefore, a small window on the user's screen begins to display contents that is provided by the server, such as advertising or user specific messages. Furthermore, as part of its custom packaging application, Robosites may arrange the location on the screen where the user can view a selected show, and also updates show structures that had been previously used by a user and had been cached in the user's computer.

At step **76**, Robosites checks whether the user is intended to receive information in a dynamic arrangement referred to as value added dynamic content. As will be explained in more detail below such value added dynamic content may include additional URL locations or files that are deemed by Robosite system to be of interest to the user, such as advertising material. Such dynamic content is usually based on, among other things, the user's profile. For example, some of the factors relating to a user profile, which are considered include a particular sports team preference; a demographic; or a pre-set profile as specified in the software. This information is used as selection criteria which is sent to a database and, at step **78**, a page, conforming to the user's preferences, is retrieved by the server, either locally or from a remotely located database. Thus, this dynamic content may be presented in the persistent content window of the user's display screen or as information and messaging contents selectively interspersed between a show structure.

At step **80**, if the resource addressed by the URL does not include dynamic content or if the user is unable to receive dynamic content, a resource having pre-defined content is retrieved, either directly from a Robocast-operated resource, or from a third party with prior authorization to handle such content requests. For example, a resource having a default set of advertising content may be presented to the user.

At step **82**, either the dynamic or pre-defined URL is accessed and the content associated with the URL is retrieved, encoded and delivered to the browser at step **84**. The content provided from the location addressed by the URL may be among other formats, include HTML, DHTML or XML documents. The delivered information may contain references for many other data types, including animation, streaming and non-streaming audio and video, Virtual Reality Modeling Language (VRML) Quicktime VR, and front-ends for several flavors of database applications, to name a few.

At step **84** the content retrieved by RoboSites is delivered by the user's browser for a show presentation. This step is inherent in the operation of browser **62** (shown in FIG. 2), accordingly, it is not described in detail. At step **86**, a timer is started in response to a message produced by the browser **62** indicating that the retrieved content has been delivered and is ready to be presented. This timer is set from a value provided in the corresponding node in a show structure which indicates an amount of time that the content accessed from the associated URL is to be displayed. It is noted that RoboSites provides an additional code to the content corresponding to the amount of time the content is displayed. For example, for an HTML file, Robosites includes additional tags corresponding to the duration that the HTML document will be displayed by the user's browser.

At step **88**, Robosites determines if the user has selected another URL before the timer expires. It is noted that the user may select another URL either manually or via a RoboSurf program that resides on the user's computer. In that event the newly selected URL is based on a node within a show structure that is being executed by the user's RoboSurf program. If another URL has been selected, the program will do one of two things: 1) forward that user away from the predetermined course of the show structure that was entered at the beginning of this process and take the user on another course, based on the other URL selected; or 2) skip them forward on the predetermined course that is being played for the user in accordance with the selected structure. This latter scenario is advantageous for a quick reader, or someone who has seen the presented content before, or in the case of content that the viewer chooses to skip (even an advertisement or PSA).

If the user has selected a new URL during the display time allowed by the timer, then, at step **90** Robosites determines whether the next URL relates to a show structure or not. This may be determined simply by the lack of new URL requests. It may also be determined by checking the header information of the new URL, as described below with reference to Table 1. As shown in step **93**, the user's decisions are recorded to a database at this point, providing a list of feedback for advertisers and content providers in some cases. If, for example, an advertisement for cat food is displayed, a user may skip over the advertisement or otherwise indicate that cat food advertisements are not interesting. This information is recorded at step **93** and used to direct only advertisements that are of interest to the user.

At step **94** the system requests the next resource in the show structure and returns to step **74** described above. It is noted that in accordance with another embodiment of the invention, it is possible to change the course of events to be displayed to the viewer by substituting next resource request at step **94** with a previous resource request. For example, if the forward going display provides a first URL content relating

to weather forecast to a second URL content relating to sports news, the reverse going display provides the sports news before the weather forecast.

It is also noted that a user may switch to another set of show structure at any time by selecting a desired structure, for example by a Robosurf program employed by the user's computer. The instructions relating to the newly selected path in a structure may be provided via point A as illustrated in FIG. 3. Furthermore, a newly selected path in a structure may also be diverted back to the prior show structure by using the application's "surf" feature (described below with reference to FIG. 7). The user's selections are tracked by the user's Robosurf program. Thus, for example when a URL relating to a second show structure is selected, by being clicked for the first time, the system remembers where the user left the initial structure and returns the user to that point in the structure upon the second click of that surf button. Alternatively, the history feature, inherent in most browsers, may be used to return the user to pages that were presented moments before or days before.

If, at step **88**, the user does not interact with the system within the allotted time (which is set by the user or publisher), the timer expires at step **92** and that signals the request for another resource, thus enabling the play sequence for automatic surfing.

At step **96**, the program requests the next node in the currently playing show structure. While this sequence has been described as a simple display of preprogrammed URLs, it is contemplated that, in more complex forms, this process may stream URLs to the viewer, while monitoring the user's habits (e.g. which sites are skipped and which sites the user views for the full display time) and injecting targeted content, including news, advertising and PSAs into the data stream.

If the URL selected at step **88**, before the time-out, is not another show structure, the Robosites process displays the selected content and resets the timer. If no additional content is available, a user-specified show structure will be optionally displayed.

FIG. 4 is a flow-chart diagram which illustrates the operation of the exemplary user component of the system, referred to as the RoboSurf program, in accordance with one embodiment of the invention. Briefly, this program allows a user to schedule, personalize and play content obtained from Web addresses. Using RoboSurf, a user may access almost any accessible website as a node in a structure. It is noted that while RoboSurf is described as a browser plug-in, it may be an Applet, stand-alone program or distributed application.

The RoboSurf process illustrated by FIG. 4 begins at step **110** to access a default list of URLs that may be generated by, for example, one of several sources, such as: 1) RoboCast's web server network that will supply a general listing of available show structures; 2) a pre-selected list generated by the server in response to information obtained about the user from a Profiler program (e.g. structures defined for an Intranet described below with reference to FIG. 6); 3) the results of a Search initiated through the RoboCast system or through a non-affiliated search engine; 4) an e-mail message, 5) the contents of a screen saver's play queue; or 6) other delivery methods.

At step **112** the URL is delivered to the browser **62** by one of these methods in a traditional manner. This corresponds to step **84** of FIG. 3, described above. Also at step **112**, the content is received and/or activated either automatically or programmatically for presentation to the user. Once RoboSurf begins to display the content, it starts a timer, at step **114**. This timer has a duration that is determined from the information contained in each node of the current show structure that corresponds to this URL.

At step **116**, while the timer is running, RoboSurf monitors the user's actions to determine if the user wants to manually override the play sequence by clicking a URL before the time-out. The user may act in several ways to override the play sequence. First, the user may select a node or a link on the displayed resource at step **122**. In this instance, if the selected link is itself part of a show structure, the program goes to step **124**, and the newly selected show structure replaces the current structure. If the selected link is not part of a show structure, the program goes to step **120** to get the next resource of the initial structure and the timer is reset and play continues at step **112**.

Second, the user may click on the Surf button, in the control buttons **7**, shown in FIG. 7, and then select a link on the displayed resource. By pressing the Surf button, the user temporarily suspends the display of the original show structure, to review contents of pages accessible by hyperlinks presented on a displayed page. When this review is complete, the show structure may be restarted at the point of departure by simply clicking on the Surf button a second time.

At step **118** if the user has not selected any URLs, the time-out occurs and the play sequence resumes at step **120**. In the exemplary embodiment of the invention, this step highlights the next item in the schedule window **2** (shown in FIG. 7), selects the corresponding URL and branches back to step **112**, causing Robosites to deliver the content at the requested URL to the browser **62**. It is noted that in

accordance with one embodiment of the invention, either RoboSurf or Robosites programs may operate independently without interaction with the other.

FIG. 5 is a flow-chart diagram which illustrates the operation of the RoboGuide component of the RoboCast program suite. RoboGuide is a helper application that provides a viewing guide which offers a personalized look at various show structures of content that have been cached at the server. RoboGuide interacts with information produced by a Profiler program to modify the cached show structures to eliminate content or add alternate content according to a user's preferences.

RoboGuide program starts at step **150** where the user starts access to the Internet or other network via a Personal Computer (PC) a Network Computer (NC), Web-Enabled TV Set, or other similar device. The user encounters and selects a URL which has been designated as the entry node in a show structure of resources by a publisher. This URL may be encountered, for example, within a screen saver, an e-mail message, or a website.

In accordance with one embodiment of the invention, the server terminal that employs the RoboGuide program stores a plurality of various show structures that may be of interest to the users. These show structures may relate to different topics or subjects that visitors may have access via the server. An example of such a server may be a search engine such as those available by YAHOO® or EXCITE®. In accordance with one embodiment of the invention, the RoboGuide program may prepare various show structures based on a user's search specifications. The results of a search may include a plurality of URL locations that contain pertinent information relating to the search topic. In response, the RoboGuide program creates a show structure that includes all or portion of the uncovered URL locations as the result of the search.

At step **152** the RoboGuide program receives a request from a user for a URL location that is provided by the server. At step **154**, the program retrieves from its database a show structure corresponding to the user's request. For example, the server may present a show on travel defined by a show structure prepared by the RoboGuide program.

It is noted that in accordance with one embodiment of the invention, the RoboGuide program may create a customized show structure based on the user's preferences that can be obtained in accordance with various filtering techniques commercially available. The server may already have information on the user, for example, via a profiler program that the user had previously updated via a RoboSites program as described above in reference with FIG. 3. Thus, based on such adaptive filtering technique the RoboGuide program creates a personalized content that can be delivered for each user or groups of users.

At step **155** the program employs a filtering technique to create the customized show structure. An exemplary filtering process may be that which is available from Firefly.net, described below. This filtering is based on rating the resources (e.g. rating a cat food advertisement as inappropriate if the user profile does not indicate that the user has a cat or if the user indicates disinterest in cat food advertisements either explicitly by clicking on an on-screen "button" (not shown) indicating disinterest or implicitly as may be inferred from a pattern of skipping over cat food advertisements when they are presented). The information generated by the Profiler program is fed to the server based on "cookie technology," a common method of maintaining information about a client on the web. The server reads the cookie for preferences and date information then feeds a personalized request to the database which sends a dynamically generated list of URLs to the user.

At step **158**, in response to the received show structures, the RoboSurf program residing at the user's terminal parses the data and displays the nodes in the structure in its Schedule window **2** (shown in FIG. 7). In response to the user's request a show structure is then presented to the user by either a RoboSurf program as described in reference with FIG. 4 or by a RoboSites program as described in reference with FIG. 3.

The following description of an exemplary filtering technique used by Firefly.net was obtained from their Web site at <http://www.firefly.net/products/CollaborativeFiltering.html#Summary>. Collaborative filtering technology is a powerful technology that can effectively address the problem of information and content overload, especially in highly subjective domains. Collaborative filtering systems use human intelligence instead of machine intelligence to address the limitations of content-based filtering systems. By leveraging human opinions, collaborative filtering systems automate the word of mouth process among users. Automated Collaborative Filtering (ACF) systems automate the word of mouth process on a global scale across hundreds of thousands of users and catalog items. Most ACF systems recommend items using the following three step process:

- 1. Capture a set of opinions from the user in the form of numerical ratings for items

- 2. Compute a set of “nearest neighbor users” for the user. These nearest neighbor users are users with opinions most like the given user.
- 3. Use the nearest neighbors ratings to compute a set of recommendations for the given user. While standard ACF systems are effective in subjective domains, they suffer from certain limitations that may cause them to make incorrect conclusions in broad domains. This is mainly due to their tendency to ignore information about the items being recommended, even when such information is readily available.

Feature-Guided Automated Collaborative Filtering (FGACF) is Firefly Network Inc.'s powerful technology that combines the benefits of collaborative filtering with those of content-based filtering to overcome the limitations that affect standard ACF systems. FGACF utilizes simple feature information about items in a catalog to dynamically partition the space of items on a personalized basis, so as to more effectively apply the ACF algorithm in the appropriate sub-space.

Designers and developers of commercial collaborative filtering systems need to address several challenging issues. The foremost issue in building large scale collaborative filtering systems is the scalability of the system. The accuracy of the underlying collaborative filtering technology, and the set of functionality offered, are important factors for designers of such systems.

Firefly Network Inc.'s Catalog Navigator product is scalable to handle loads of hundreds of thousands of users, catalog sizes in the half million range, with number of ratings in the millions, and still provide sub-second response times to queries. It has already been successfully deployed in applications on high-traffic sites. In addition, the Catalog Navigator is the only commercial implementation this powerful FGACF technology and offers one of the most comprehensive sets of catalog and community navigation functionality to a customer. © Firefly Networks Inc. 1997.

It is noted that in accordance with another embodiment of the invention, the information presented to the user based on the show structures may be reformatted in accordance with certain default structure characteristics. For example, such reformatting may include the tasks, such as: making Web resources more reader-friendly when using RoboSurf, RoboSites or RoboGuide programs, by utilizing formatting parameters, such as restraining page size to a default screen size and cuing automatic playing of audio upon displaying a new page; providing “Play,” “Stop,” “Forward,” “Reverse,” and “Pause” indications to facilitate the user's viewing of a show presentation; making pages into parts of a of content; providing information tags relating to the duration that a content is going to be displayed on the user's terminal; and, encoding pages with certain demographic and keyword-based information to aid the filtering operations explained above.

Table 1 shows an exemplary HTML code that may be added in accordance with one embodiment of the invention, to the Web resources or other content that is being reformatted. Note that the HTML of a page can be altered in this way between downloading it from a server and presenting it to the user. This “Tagging” process only requires a few milliseconds and can be done on the fly.

Essentially, this module uses the abilities of a HTML page publishing package like Pagemill from Adobe (www.adobe.com) to add HTML code of the type shown in Table 1.

TABLE 1

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 3.0//EN">
<html>
<head>
<title>RoboGuide 11/3/97</title>
<META name="author" content="© 1997 Damon Torres,
Brian Foy">
<META name="generator" content="RoboSites
1.0">
<META name="keywords" content="news, sports, weather, lifestyles, jane q. public,
nyc weather, san francisco weather, washington d.c. weather, yankees, knicks, non-violent
news,
```

parenting info, content filter 530145x2a, ad filter 398543*kjfd23, psa filter 19284jksd8sfj*659">

<META name="description" content="Jane Q. Public's personalized news, sports and weather. Contains preferences for cities, teams, content filters (i.e. sex, violence, language) and preferences for advertising and public service announcement categories. Wildcard paramaters are

indicated by an asterix (*).">

<META HTTP-EQUIV="Expires" content="Tue, 04 Nov 1997 18:45:23 GMT">

<!-- see http://www.sandia.gov/sci_compute/elements.html for details -->

<LINK rel="previous" href="some url">

<LINK rel="next" href="some url">

<LINK rel="toc" href="url of channel description"><!--
toc=Table of

Contents-->

</head>

<body>

<!-- content goes here -->

</body>

</html>

FIG. 6 is a flow chart diagram that is useful for describing the operation of a RoboPublisher program, which is the publishing tool used to create and modify show structures for viewing via RoboGuide, RoboSurf and/or Robosites. The first step represents activities occurring at the user's computer **12**. At step **200** the user access the program described in this flowchart. At step **210** the user selects between editing an existing show structure or creating a new one. This can be an existing structure that has been previously made, or a structure that another content provider has published. If the user is creating a new structure, the resources, or nodes of the structure, can be gathered from their local hard drive or server or from a remote server.

At step **212** the user is presented with a dialog box. At step **214** the user is presented with the dialog box options for setting general preferences for all of the resources that will be linked in the structure, such as duration for all resources, duration estimate based on word count, and default "page turn" sound to accompany an automatic resource delivery.

At step **216** the user has chosen to load a previously defined structure, and, at step **218** the computer requests and loads the structure from the local or remote drive. Beginning at step **220** the user has many options, presented by an interface that will allow the user to choose to "Add, Edit or Cut" a node for the structure they are building. This can be accomplished graphically, as shown for example, in FIG. 6A or via a list. In step **222** the user may Add a node from a standard dialog box that enables browsing from all available drives and from a special cache area that RoboSites, RoboSurf and/or Robosites produce. This cache area allows users to collect URLs from the Internet by, for example, browsing, implementing searches, or culling their e-mail for additional show structures. At step **224** the user adds the node at the chosen spot in the structure.

At step **226** the user has the option to commit to changes or cancel and start again. At step **228** the Modify Choice selection gives the user options **230** to edit the parameters of the "next" node (i.e. changing the next node destination or previous node in the structure from, for example, a local news link to a local weather link). The duration choice pertains to how long the node will be displayed before a subsequent node is loaded. This can be done on a per resource basis, since some resources require more time than others. The expiration option allows the publisher to choose a "kill date" for the node,

automatically triggering a replacement search, or a skip to next action. This is useful for dated information such as weather.

At step **232** the user has the option to delete a node from the structure. Again, this can be done graphically (as shown in FIG. 6A) or via a list. This step leads to step **234** where the previous structure is linked to the node that succeeds the deleted node. Step **236** allows the user to choose to continue adding, editing and/or cutting by branching back either to step **220** or **238**, to save the work and/or export it for a preview or viewing by others. If the user does not choose to save or export, then, at step **244**, the program reverts to the last saved version or cancels the session. If the user does export then, at step **240** the program writes the changes to the local disk **36** (shown in FIG. 2). The RoboSites program may, in turn, write header information to the files to which the user may access locally or even to remote server files. It should be noted that the writing of header information into the HTML documents is an option, providing a convenience for RoboSurf and Robosites programs. The page playing actions can be achieved without this header information present. The publisher completes its work at step **242** where it gives the option to preview for the publisher using RoboCast browser enhancements, or to publish for other RoboCast product users.

FIG. 7 is a screen diagram of the interface presented by the RoboSurf program. Item **1** on the screen is an interface to the Profiler program. This program allows a user to specify several profiles for RoboSurf. The user may select from prepared profiles, for individuals or groups for example, "Internet Professional," "Lawyer," "Student," "Family.1" and "Girl: Preschool" and modify these or create a new profile. In the Profiler window, the User field allows a name to be attached to the profile. The Schedule field allows a user to associate the profile with a particular structure, also referred to as a Schedule. In this instance, the structure of URLs being displayed in the Schedule window **2** is named "After Dinner."

The Reviews field allows a user to specify one of several sources for website or show structure reviews. Using the Ratings field, the user can specify limitations on the type of website or structure that will be reviewed using the familiar MPAA rating system. The Censor Level field may be used to control the Web content that the user will see. Different Censor Levels will allow different types of content to be displayed. The RoboSurf program may compare the specified Censor Level to voluntary content ratings obtained from the Web resources or it may interface with an existing content monitoring system such as Cyber Patrol to ensure that undesirable content is blocked. The final field in the Profiler window allows a user to specify a Help or Guide function. It is contemplated that both text-oriented and voice synthesized help would be available. For voice-synthesized help, the user may identify a particular voice with a name (e.g. "Chris") as shown in FIG. 7.

Item **2** in FIG. 7 is the Schedule window. As described above, this window shows the current structure of URLs that is being displayed by RoboSurf. The window contains a brief textual description of each URL and a time in hours and minutes and seconds for which content provided from the URL will be displayed. The URL from which content is currently being displayed is highlighted. Each URL in the schedule list also includes a small box which is marked with an "x" after a URL has been viewed and may be marked with a " " to indicate that the user wants to return to this site.

Item **3** in FIG. 7 are controls for volume, channel and speed. The volume control works as would be expected to increase or decrease the level of sound provided. The Channel button allows a viewer to move a pointer up and down in the channel window to select a new channel. As used in this description, a "channel" is a collection of resources, such as web pages, which are linked into a structure by a content provider. The Speed button allows a user to specify the amount of time between the content of resources. The "*" button on each of the volume, channel and speed buttons brings up a separate dialog box in which specific entries may be entered manually.

Item **4** in FIG. 7 is the progress bar. This bar displays progress of the system in whatever it has been requested to do. During play mode, this bar displays progress through the structure as a percentage of time remaining in the display of the structure. In search mode, the bar may display progress in a search and the preparation of a show structure derived from the search. In gather mode, the bar may display the progress of the gathering operation.

Item **5** in FIG. 7 are the programming bars. The Search button allows the user to establish criteria for a search of the Internet for URLs to be included in a show structure. The Pick button lets a user select categories within established information sources in order to make selections within the specified sites. Once Search Criteria and Pick categories have been established, the Gather button retrieves information from the selected sources according to the user's criteria at a specified time and frequency. The Schedule button allows the user to group site content by topic and then set the time when specific group will be played. The Play button begins the presentation of information. In the exemplary embodiment of the invention, the Play button also lets a user set parameters such as size of the content window, length

of time each item is displayed and, if RoboSurf is being used as a screen saver, screen saver activation parameters (e.g. delay time, autostart position of cursor and password).

Channel window **6** displays structure publishers that may be accessed by a "point and click" operation. This window is analogous to a "favorite places" list in a conventional browser.

Control window **7** contains buttons which allow a user to Skip backwards and forwards through the show structure displayed in the Schedule window, Play the information provided from the structure of URLs, Stop playing this information, Print the information that is currently being viewed, and Mark the current resources for later viewing. In the exemplary embodiment of the invention, the Mark button interacts with the Play and Print buttons. While viewing contents relating to a show structure, a user may mark individual resources. After viewing the show structure, the user may specify that the marked resource be displayed again, queued for printing or displayed and printed. This feature may be activated, for example, through the play or print buttons by activating the buttons with a modifier key, for example, the right mouse button.

In addition to these buttons, the control window also includes an Add/Up button which allows a user to add resources to a structure when it is being set up in schedule mode and allows a user to move up the hypertext links that are being displayed on the screen during play mode. The Cut/Down button conversely allows a user to delete a site from the structure during schedule mode and move down the hypertext links during play mode. The function of the Surf button is described above. This button allows a user to temporarily suspend the viewing of a structure with a first button press and resume at the suspended site with a second button press. In accordance with another embodiment of the invention the functions described above in reference with FIG. 7 are embodiment in a separate remote control hardware device. The e-mail and video phone buttons, item **9** on the display shown in FIG. 7, invoke third-party software packages, one for e-mail and one for a video telephone product. Item **10** in the display is the content window. This is where all of the content which is Found, Picked, Gathered, Reviewed, Rated, Censored, Edited and Scheduled is finally and automatically played. The process of playing the content is like television mixed with radio, newspapers and a research library.

While the invention has been described in terms of an Internet browser application which allows a user to view or create a show structure of content from various sites, it is contemplated that it may be used for other purposes. For example, the invention may be used as a jukebox to schedule and present successive audio or video recordings or to schedule and view pay-per-view movies or videos.

As illustrated in FIG. 6A a show structure of resources may be created and played in accordance with various embodiments of the invention. These resources may include, for example, various local application files on a user's computer. Thus, a user may view information relating to various executable programs in a selected arrangement on a desired schedule. A typical example, involves a user system that contains an accounting software, an on-line banking software, an Intranet e-mail program, and a calendar program. The user creates a show structure that executes each of these programs and displays the relevant information at a predetermined rate. This allows the user to automate various tasks that are currently accomplished manually. Thus, for example, the user on a daily basis can check received e-mail, followed by the user's schedule for the day, available funds in the user's bank account. The selected information are presented automatically in an organized arrangement. It is noted that in accordance with another embodiment of the invention, a show structure may include the user's local resources and other remote resources as those explained above in reference with Internet applications.

While only certain features of the invention have been illustrated and described herein, many modifications, substitutions, changes or equivalents will now occur to those skilled in the art. It is therefore, to be understood that the appended claims are intended to cover all such modifications and changes that fall within the true spirit of the invention.