ENHANCED VIDEO PROGRAMMING SYSTEM AND METHOD UTILIZING USER-PROFILE INFORMATION

The present invention relates to a method, device and apparatus for compiling and maintaining, or accessing, information for routing and transmitting content to a machine via a network.

Computers have the capability to provide massive amounts of educational and entertainment information by way of the Internet. Currently, on-line systems offer a variety of different services to users, including news feeds, electronic databases (either searchable by user directly on the on-line system, or downloadable to the user's own computer), private message services, electronic newsletters, real time games for play by several users at the same time, and job placement services, to name a few. However, currently most on-line communications occur merely through text. This is in contrast to the audio/visual presentation of the alternative electronic medium, television. However, it is expected that as multi-media's incessant growth continues, audio/visual programs will proliferate and text will become less and less dominant in the on-line environment.

Even though these programs will be introduced, the Internet will remain essentially user unfriendly due to its very massiveness, organization, and randomness. Simply stated, there is no order or direction in the Internet. Specific pieces of information can be hard to find, and it is even harder to put that piece of information into a meaningful context.

Television, on the other hand, has been criticized for being a passive medium. Whilst interactive television systems have increased the level of user interaction, and thus, provided greater learning and entertainment opportunities, vast information resources such as databases are inaccessible from such a medium.

The present invention seeks to close the gap between video programming and the vast information resources of the Internet.

According to the first aspect of the present invention, there is provided a device for compiling and maintaining information for use in routing and transmitting content to a machine via a network, comprising:

a computer-readable medium including information for use in transmitting content to a machine;

said medium comprising fields for specifying an identification of the machine, an address of the machine, and user-profile information, for use in determining a type of content to transmit to the machine, the user-profile information being specified in an hierarchical attribute value pair data structure.

A device of an embodiment of the invention can be used to change the nature of advertising. It enables pertinent information to be given to consumers automatically.

In a preferred embodiment of a device of the invention, which may be an article of manufacture, the computer-readable medium further specifies attributes of a user associated with the user-profile information and the hierarchical structure identifies hierarchical relationships among the attributes.

For example, the medium may specify information identifying the user, identifying preferences of the user, and/or identifying a room assigned to a user for a chat service. In this latter case, the information specified may identify members of the room for a chat service.

Additionally and/or alternatively, the medium may further specify an indication of a directory for use in obtaining instructions for routing the content.

Preferably, the medium includes user-profile information for use in selecting at least one of the following to transmit to the machine: information available via a Uniform Resource Identifier, video content, audio content, multimedia content, a particular video stream, or an executable object.

The medium preferably includes user-profile information for use in selectively transmitting survey questions to the user.

In an embodiment, the medium specifies the address of one or more of a personal computer, a television, a cable box, a satellite box, a video game console, and/or of a personal digital assistant.

According to a further aspect of the present invention there is provided a method for compiling and maintaining information for use in routing and transmitting content to a machine via a network by specifying particular fields within a computer-readable medium, the method comprising the steps of:

receiving information for use in generating a user profile;

specifying in the medium, using the information, an identification of a machine, an address of the machine, and user-profile information for use in determining a type of content to transmit to the machine; and

storing the user-profile information in an hierarchical attribute value pair data structure.

Preferably, the method further comprises the steps of:

specifying in the medium attributes of a user associated with the user-profile information; and

specifying in the hierarchical structure hierarchical relationships among the attributes.

In embodiments, the method further comprises specifying in the medium information identifying the user, and/or specifying in the medium information identifying preferences of the user, and/or specifying in the medium information identifying a room assigned to a user for a chat service. In the latter case, the specifying step may further comprise specifying in the medium information identifying members of the room for the chat service.

The invention also extends to a method of accessing information for use in routing and transmitting content to a machine via a network, the method comprising the steps of:

establishing a network connection from a machine;

accessing via the network connection an hierarchical attribute value pair data structure stored in a computer-readable medium; and

transmitting information via the network connection for specifying in the data structure an identification of the machine, an address of the machine, and

user-profile information for use in determining a type of content to transmit to the machine.

In preferred embodiments, the data structure is stored in storage means associated with the machine and/or associated with a server.

Preferably, the method further comprises dynamically updating the user-profile information. For example, the dynamically updating step may comprise monitoring activity of a user associated with the user-profile information; and updating the user-profile information based upon the monitored activity.

A method of an embodiment of the invention may include selecting, based upon the user-profile information, at least one of the following for transmission to the machine: information available via a Uniform Resource Identifier, video content, audio content, multimedia content, a particular video stream, or an executable object.

The present invention also extends to apparatus for accessing information for use in routing and transmitting content to a machine via a network, comprising:

network means establishing a network connection from a machine;

access means for accessing via the network connection an hierarchical attribute value pair data structure stored in a computer-readable medium; and

means for transmitting information via the network connection for specifying in the data structure an identification of the machine, an address of the machine, and user-profile information for use in determining a type of content to transmit to the machine.

The apparatus may have storage means associated with the machine for storing the data structure, and/or storage means associated with a server having the network connection with the machine, the storage means being arranged to store the data structure.

Embodiments of the present invention will hereinafter be described, by way of example, with reference to the accompanying drawings, in which;

Figure 1 is a diagram showing the receipt and decoding of video signals at a subscriber location using a method of the invention;

Figure 2 is a diagram showing an alternative embodiment to achieve the integration of Internet information with video content;

Figure 3 is a flow diagram of the basic software of the invention;

Figure 4 is a diagram showing an embodiment in which URLs are directly transmitted to a user;

Figure 5 shows an embodiment of a system comprising a digital cable box;

Figure 6 shows an embodiment of a system including a digital T.V.;

Figure 7 shows an example of a user interface;

Figure 8 shows an example of a display providing a user interface;

Figure 9 is a diagram showing an embodiment of a system having distributed communication servers;

Figure 10 illustrates a logical structure of a donut for implementing user-profile information; and

Figures 11A and 11B show a flow chart of a process for generating and implementing donuts specifying user-profile information.

Figure 1 illustrates an embodiment of a computer based system for receiving a video program along with embedded uniform resource locators (URLs) which direct a user's computer 16 to address locations, or web sites, on the Internet 20 to retrieve related web pages. The web pages correspond to the video presentation. The particular video programming can be delivered in analog, digital or digitally compressed formats (e.g. MPEG2) via any transmission means, including satellite, cable, wire, television broadcast or sent via the web.

The video programming is preferably created at a centralized location, for example, as content creation 4 indicated in Figure 1, for distribution to subscribers. Program creation may be accomplished by any appropriate means. After a video program is created, uniform resource locators (URLs) are embedded. In one embodiment, the URLs are embedded into the vertical blanking interval of the video programming by a URL encoder 8, as shown in Figure 1. In this embodiment, the URLs are encoded onto eight fields of line 21 of the VBI. Line 21 is the line associated with close captioning, among other

things. However, the URLs may additionally and/or alternatively be embedded in other fields of the VBI, in the horizontal portion of the video, as part of the audio channel, in any subcarrier to the video, or if the video is digital, in one of the data fields.

Although Figure 1 shows the video with the URLs broadcast over the same transmission line, the URLs may be sent down independently of the video program on a data channel. In this embodiment, the URLs may be forwarded to the remote sites either prior to initiation or during the program. Preferably, the URLs have associated time stamps which indicate to the subscriber stations when, during the video program, to display the particular web pages addressed by the URLs. Alternatively, the user can select when to call the particular web pages for display with the video program.

The particular information in line 21 is not part of the visual part of the program, and thus, is not perceptible to the human eye, thereby making it ideal to send data information to the users. Whilst the bandwidth capacity of line 21 is limited, as a system as described transmits only the URLs, and not full web pages, there is more than enough capacity. Furthermore, no additional hardware is necessary at the computer 16 to receive the video and retrieve the web pages.

Once the video program is created, it may be transmitted to user sites over any transmission means, including broadcast, cable, satellite, or Internet, and may reside on video servers. Furthermore, the video program, with or without embedded URLs, may be encoded onto storage means such as a video tape, for example of VHS or Beta format, or an optical disc such as CD or DVD, or any other medium.

Preferably, each receiver station comprises any Intel x86 machine (preferably a 486 processor, pentium processor, etc), an Apple Computer, UNIX or any other type of standard computer workstation. The local computer 16 is preferably connected to either a cable and/or broadcast television or to a local VCR or other video source. At each subscriber site, the local personal computer 16 preferably receives the cable transmission by cable connection on the back of the personal computer 16. The video/audio program may be processed for

display on the computer screen using a PC card capable of displaying video signals on a computer monitor in an appropriate TV format such as PAL or NTSC. One example of a PC card is a WinTV card. In addition to the cable connection, there is the Internet 20 connection created concurrently with the cable connection.

The Internet 20 connection may be via high-speed line, RF, conventional modem or by way of two-way cable carrying the video programming. The local PC 16 has Internet access via, for example, an ASCII software mechanism. In an embodiment, at each subscriber site, an associated local URL decoder 12 extracts the URLs, preferably embedded in the vertical blanking interval, with the use of a suitable VBI decoder device. The URL decoder 12 may be either a stand-alone unit or a card which is implemented into the personal computer 16.

In the embodiment shown in Figure 2, the uniform resource locators (URLs) are encoded into the video as described above. Again, the URLs are preferably encoded onto eight fields of line 21 of the VBI, but may also be sent independently of the video. In this embodiment, a URL decoder 24 is located at the server site rather than at the subscriber location. When the decoder 24 receives the video program signal, it strips out the URL codes on line 21 of the VBI and delivers these codes independently to an Internet server 28. The URL code is then subsequently delivered over the Internet 20 to the user PC 16. Simultaneously, the video is broadcast over conventional broadcast or cable transmission means 36 to the user's personal computer 16.

The alternative shown in Figure 4, does not use the VBI. In this embodiment, the system runs an online service over the Internet 20. This service is in the form of an Internet web site 62 which provides a user-interface to a database 78 and to one or more associated data servers 90. The service provides member accounts to TV broadcasters 66 who sign up to use the illustrated system in conjunction with their broadcasts. Each member broadcaster will enter the service at their computer 70 through web browser software 74 using their member account by entering various identification and password information. Once within their account, the member will be provided with a graphical user interface for pre-scheduling URLs for transmission to users 118 over a direct Internet connection 94 at particular times of day. The same

user interface, or a variation of it, can be used by broadcasters for live transmission 82 of URLs to users at the same time as a broadcast 86.

One example of this interface might be a scheduling calendar (daily, weekly, monthly, yearly) in which the broadcaster 66 may allocate time periods which coincide with their broadcasts 86, and during which they will send out URLs to their users to link to web pages. For each time period (for example, a particular hour long period during the day) determined by the broadcaster 66 to be a broadcast period (a period during which they want to transmit URLs that correspond to a television show being broadcast from their TV broadcast facility 110 to the external TV 114 of the user 118 at that time), the broadcaster 66 may then enter a series of URLs into an associated file ("Link File") for transmission over the Internet 20 at that time. This Link File may have a user interface such as a spreadsheet, table, or list, or it may be simply a tab-delimited or paragraph-delimited text-file. As an example, each of the records in the Link File consists of a data structure which may contain information such as:

(<timecode>,<URL>,<label or title>, <additional information>,<additional information>,...)

The above data structure is just one example. The records in the Link File preferably specify the time, Internet address (i.e. URL), label (such as an associated name), and some optional additional information, for each web page the broadcaster 66 desires to launch during a show.

When a broadcaster 66 modifies their calendar and/or the Link File associated with any given time period(s) in their calendar, this information is saved into the database 78 which is attached to the site 62. Each broadcaster 66 may maintain multiple calendars in the database 78 if they broadcast in different time zones, for example.

The database 78 provides the Link File records for upcoming time periods to a server 90, which may be one server or a distributed network of server programs on multiple computers across the network, to be utilized for scaling to large national or global audiences. The server 90 provides the Link File records, including the URLs, to the user's personal computer 16, which is connected via

a network. Examples of possible networks include the public Internet 94, a direct private network, or even a wireless network.

One feature of the embodiment illustrated in Figure 4 is that one or more broadcasters 66 may utilize the same schedule in the database 78 for their own broadcasts 86 or during the same broadcast. For example, a network broadcaster may develop a master schedule and various affiliate broadcasters may subscribe to that schedule or copy it (in the database) and add or delete specific URLs in the schedule for their local audiences or unique programming. This scheme enables affiliates to insert URLs for local advertisers or local subjects into a sequence of more general URLs provided by their network broadcaster 66. In other words, the affiliate can add links that ride on the network feed and then redistribute it to their local audiences.

The system of Figure 4 also enables personalization in the form of unique series of URLs specific to each user's unique profile, which are directly sent over the Internet 20 to each user's specific client software 106. This can be achieved from the broadcaster 66 to each individual user 118, or to particular collections of users. To accomplish personalization, the service may send a different stream of URLs to each user's client software program 106. The stream of URLs sent depends upon a user profile stored in the database 78 or the client software program 106, a user profile which is built on demand or over time for each user 118 based on criteria such as the location of the user, choices the user makes while using a client software program 106, choices the broadcaster 66 makes during a broadcast 86, or automatic choices made by an algorithm (such as a filter) residing on the service 62. Personalization enables each user to receive URLs which are uniquely relevant to their interests, demographics, history, or behaviour in the system.

Once the URLs have reached the personal computer 16, the operation of all of the systems shown in Figures 1, 2 and 4 is similar.

In one embodiment, a JAVA enabled browser 98 as well as specialized software 106 are installed on the computer 16. The JAVA enabled browser 98 allows the computer 16 to retrieve the web pages 102 and is presently the preferred software, as it is platform independent, and thus, enables efficient and

flexible transfer of programs, images, etc., over the Internet 20. The specialized interface software 106 (hereinafter, "client software") acts as an interface between the video programming and the Internet functions. The client software 106 retrieves URLs from the video program (embodiment of Figure 1) or directly from the Internet connection (embodiments of Figures 2 and 4), interprets these URLs and directs the JAVA enabled browser 98 to retrieve the particular relevant web pages 102. The client software 106 also synchronizes web pages to the video content for display on the user's computer 16, as shown in Figures 3 and 4 and explained in more detail below.

As explained above, the URLs may be encoded and embedded into the video signal by inserting them into the vertical blanking interval (VBI).

Alternatively, the URLs may be entered by member TV broadcasters 66 along with specified times for transmitting the URLs to the user. At the appropriate times, the URLs are sent directly over the Internet to the user's PC 16 via the client software 106 over a direct point-to-point or multicasting connection.

The system may have the capability to detect identical URLs sent directly after one another and to cause the browser not to fetch URLs in these particular cases. As shown in Figure 3, once the URL code is received at the computer, the client software 106 first interprets the URL and determines in step 42 whether the particular URL has been received previously. If it has already been received, the next received URL is interpreted for determination of prior receipt. If the particular URL has not been detected before, the software checks for misspelling in step 46 and any other errors, and if errors exist, corrects these particular errors. Once again, it is determined whether the URL has been previously detected. If it has, the next URL is accessed in step 38. If the URL has not been detected, the specific URL is added to the URL list in step 54. The specific URL is then sent to the web browser, preferably a JAVA enabled browser 98. Upon receipt of the URL, the browser 98, in step 58, will access the web site address 122 (Figure 4) indicated by the URL and retrieve the cited web page(s) 102 via the Internet.

Viewers can view the integrated presentation in the following manner. As mentioned above, the video signal is processed and displayed on a video window on the PC screen using a WinTV card, for example. The corresponding audio is forwarded to the audio card and sent to the PC speakers.

The retrieved web pages 102, referenced by the URL, are optionally time stamped to be displayed on the computer screen when predetermined related video content is displayed in the video window, thus enhancing the video presentation by providing in-depth information related to the video content thereto. Another section on the screen is also preferably used to represent an operational control panel. This control panel provides a list of the URLs which have been broadcast and correspondingly received by the computer 16. This control panel is updated to add a URL code each time a new URL code is received by the PC 16. This list gives the subscriber the flexibility to go back and retrieve particularly informative or interesting web pages that have already been displayed earlier in the program, or alternatively, to print them out for future reference. Furthermore, the list may include URLs referring to web pages not displayed with the broadcast program, but which provide further information on a certain topic of interest to the viewer.

In an example, a viewer may begin watching a musical video featuring a band. As the video is received by the PC 16, URLs are either being received with the video signal or are being received directly via the Internet 20 or another data channel, and are interpreted by the client software 106. Upon direction and command, the JAVA enabled browser 98 retrieves particular web pages 102 from Internet 20 web sites identified in the URLs. These web pages 102 are then displayed on the video screen at particular times. So, for example, whilst the viewer is watching the music video, biographical information on the band may also be displayed adjacent to the video window. Web pages 102 may also include an upcoming concert schedule, and/or audio clips of the band's music may be downloaded from the Internet 20.

As another example, a user may be watching a program relating to financial news. Whilst the narrator is shown discussing high tech stocks, web pages corresponding to detailed financial performance information on high tech stocks, environment and characteristics may be displayed with the video on the

computer screen. If the personalization features are included, web pages associated with a particular user's stock may be fetched and displayed on the computer screen with the video program. When the program narrator switches to a discussion on the weekly performance of the Dow Jones, web pages presenting related financial performance information may be simultaneously displayed.

A user may view the interactive program using a television set 114 or other display monitor in conjunction with the display screen of the personal computer 16. In this case, the relevant web pages are shown on the personal computer 16 whilst the video program is displayed on the television monitor 114. In this alternative, a cable set top box receives the television program from the multi-channel cable. The personal computer 16 also receives the video program from the multi-channel cable and extracts the URLs, embedded in the vertical blanking interval of the video signal or directly transmitted 94 over the Internet 20. The client software 106 extracts the URLs and retrieves the particular web pages as described above. The web pages are then synchronized with the particular video frames and presented to the user. It is understood that a hyperlink may exist on the web site that will allow the user to automatically load the client software and call up the specific television channel referenced in the web site. For example, someone browsing the Internet 20 may come upon a major television network's web site. It is possible then to scroll to an interesting story and then to click on an hyperlink to turn on the software which tunes the TV window to the network.

Instead of receiving the video program from a transmission means, the video program may be addressed directly from the user site if the video program, with or without embedded URLs, has been stored on appropriate means. The storage means may be a videotape in any format, such as VHS or Beta, or an optical disc in any format, such as DVD or CD-ROM. In this case, the user PC 16 and/or television 114 are connected to a video tape player, a disc drive, or other appropriate device.

Figures 5 and 6 show two alternative examples of systems which may be employed. As shown in Figure 5, a user may view an interactive program using a television set 18 or other display monitor in conjunction with a digital cable box

140. In this case, the digital cable box 140 performs the functions of the personal computer 16 shown in Figures 1, 2 and 4, and the client software is stored in memory in the digital cable box 140. In one embodiment, the digital cable box 140 includes two tuners, thus allowing both the web page and the video program to be simultaneously viewed on the same screen. If video and web stream, however, are carried on one channel, then only one tuner is necessary.

The client software retrieves URLs from the received video program, directly from the Internet connection 20 or via a separate data channel, interprets these URLs and directs the web enabled browser to retrieve the particular relevant web pages, and synchronizes the retrieved web pages to the video content for display on the television 18. The relevant web pages are preferably shown in one frame of the television 18 while the video program is displayed in another frame. Alternatively, the web page can replace the video program on the display.

In this embodiment, the digital cable set top box 140 receives the television program from the multi-channel cable. The URLs can be encoded into the digital program channel using MPEG1, MPEG2, MPEG4, MPEG7 or any other compression video scheme. Alternatively, the URLs can be transmitted to the digital cable boxes 140 from an Internet server 148. The digital cable box 140 decodes the URLs from the digital video signal or directly transmitted over the Internet 20. The client software decodes the URLs and retrieves the particular web pages as described above. Preferably, the web pages are synchronized with the particular video frames and presented to the user.

As with all the embodiments described above, instead of receiving the video program from a transmission means, the video program may be addressed directly from a local video source 144 if the video program, with or without embedded URLs, is stored on a storage means such as a video tape or optical disc. In this embodiment, the digital cable box 140 is connected to a VCR, disc drive or other appropriate device.

Figure 6 illustrates an embodiment where a digital TV 152 is the remote reception unit and performs the functions of the personal computer, shown in

Figures 1, 2 and 4, and the digital cable box 140 shown in Figure 5. A processor means and memory are incorporated in the digital TV 152, and the client software and web browser software are implemented in memory in the digital TV 152. All of the functions described above with reference to the other embodiments are performed in a similar manner by the digital TV 152 embodiment.

Although the digital cable box/TV 140, 18 and digital TV 152, shown in Figures 5 and 6, are incorporated into the embodiment of Figure 1, in substitution for the PC 16, they may also be substituted for the PC 16 shown in Figures 2 and 4.

A user may view the video and web content on one screen (in two windows), or with the video on one display screen and the web content on a separate display monitor. Alternatively, a user may access the video or web content separately. Thus, a user may branch from video to web content and vice versa.

The systems described herein are well-suited to the education environment. Thus, students and teachers may access one or more web servers. Software components including instructor and student user software, authoring software and database assessment software are provided. An instructor may, for example, use content creation software on a personal computer to easily integrate into the curriculum current information published on the web through an interface 156 shown in Figure 7. The instructor creates a playlist (i.e. linkfile) 160, the playlist 160 comprising a list of web pages, text notes and questions. The web sites and questions are set out in a predetermined order and can be assigned times. Preferably, the URLs identifying the web site and time stamps are sent automatically to the desktop of each student in the virtual community, either during a playback of a pre-recorded program or during a live event.

At each of the student workstations, the program is directed by the playlist 160. In other words, the playlist 160 provides the structure for the program. At predetermined times as indicated by the playlist 160, the browser will fetch and display a web page in a frame on the computer screen. Because program

events can be set up in this manner at predetermined times, the entire program and playlist can be prerecorded and stored in a web database for later access by students.

It will be appreciated that the students and the instructor may be located anywhere, as long as they are all connected to the web. Because a server controls the program, the instructor output comes from the server and the student workstations are automatically updated by the web server.

This educational embodiment integrates web content and other media with collaborative groupware functionality to create an interactive environment for students and teachers. The student may receive a traditional video lesson through a frame in his or her web browser, or from a television. Separate frames may be simultaneously provided as shown in Figure 8, which shows the browser displaying: web pages 176 automatically delivered to each student's desktop with information or exercises that complement the video presentation; a chat dialogue frame 168 for conversing with the instructor and/or other students online; and an interactive playlist 164 of web pages and questions comprising the lesson.

In the student interface of Figure 8, each student may perform a virtual experiment, for example, during a physics lesson to learn about gravity. In addition, the students may converse with one another and with the instructor using the chat dialogue frame 168. They may also send web pages to one another and provide answers to questions from the teacher via the chat dialogue frame 168 of the student interface 176. With the chat feature, students may break into subgroups for collaborative learning. Whenever a student in the group sends a message, the message is sent to the Internet server 20 and every other student in the subgroup receives and views the message in their chat dialogue frame 168.

The instructor, however, may retain control over the chat feature. For example, the instructor may terminate the chat feature or web push to terminate unruly on-line conversations or the sending of web pages by students.

The systems described herein are more powerful than conventional distance learning systems as they allow the instructor to freely and conveniently exercise almost any type of testing strategy. The instructor may test students using a combination of the chat dialogue feature and web pages. For example, multiple choice questions and short answer questions can appear in the chat window 168. Essay questions, requiring longer answers, become web pages. As mentioned above, students can perform virtual experiments on-line. Once the instructor's personal computer receives student answers, student scoring may be presented to the instructor in any format including tables, charts, diagrams, bar graphs, etc. The instructor, thus, may analyze the results and has the capability of providing real-time feedback to the students.

Students may also receive individualized feedback via branched interactive audio, video and/or graphics responses. For example, the workstation may branch to a particular audio response, preferably prerecorded in the instructor's own voice, based on the student response to a multiple-choice question. A plurality of potential audio responses may be made available at the student's workstation, for example, by a method as described in US patent No. 5,537,141. Additionally and/or alternatively, personalized video, audio and graphics segments may be delivered and displayed to the student based on a student answer or personal profile, for example, in a manner as described in US patent No. 5,724,091.

Responses to student answers may be more substantive using a memory feature comprising an algorithm which selects an interactive response to the user based not only on the student's current answer selection, but also on the student's previous responses. The algorithm, preferably stored in memory at each student's workstation and under processor control, selects an output interactive response based on student responses. In an example, a student who gets three or more answers in sequence right receives a more difficult question. However, a student who fails to correctly answer one or more of the three questions receives an easier question.

The system illustrated in Figure 9 is capable of servicing large numbers of users, for example, several schools. As shown, communications servers 180 distribute and route message across a LAN, WAN and the Internet. At the heart

of the system is a group database server 184, and this is surrounded by several communication servers 180 which each serve an area 192. Each communication server 180 is surrounded by squares representing user stations 188. The communication servers 180 are organized in node relationships with one another.

Each node is responsible for serving an area 192. An area 192 is defined as a virtual location serviced by a single communication server 180 (or "com server"). An area 192 may be a single school, an office, or may consist of several actual physical locations. The defining characteristic of an area 192 is that messages sent from one member of an area 192 to another need not be routed outside of the servicing com server 180.

An area member is analogous to the frequently used term "user". For example, a "user" may be a student in an educational environment.

The distributed communication system shown in Figure 9 permits the dynamic addition of communication servers 180 within a group with little or no administrative tasks as well as the addition of groups within an overall communications network. A communication server group consists of several defined virtual areas 192 (preferably, consisting of no more than 250 members each), each area 192 serviced by a single com server 180. This system allows members of one area 192, or group, to easily communicate with members of another area 192 or group without any configuration changes.

In the past, service of very large numbers of users has required large expensive servers and networks. Furthermore, as the user base increased, performance suffered and the hardware had to be upgraded to service the demand.

The distributed communication system allows the same, relatively inexpensive, machines to serve an ever-increasing user base. This is accomplished by routing messages from one server to another when necessary following substantially the same core pattern as IP routing and DNS lookups. If a message is for a member not belonging to the current area 192 or group, the message is routed through the distributed communication system until its

destination, or someone who knows the destination and can deliver the message, is found. The destination may be cached so subsequent messages for that member or group may be more efficiently delivered.

Referring to Figure 9, if a message is posted by member "A" and is intended only for the members of group 1, the message never leaves the area 1 com server. However, if the message is intended for members of area 1 and for members of area 2, the area 1 com server forwards the message to the group database server 184. The message is broadcast to the members of area 1 and tagged in the group database server 184 as belonging to area 2. The message is then routed to area 2 and broadcast to area 2 members. With this technique, any member may potentially send a message to any other member. If the area com server 180 does not recognize the destination, the message is forwarded up the line. Each com server 180 does not need to know about any other server 180. Messages are routed until they delivered. If undeliverable, the original sender is notified.

New areas 192 can be added on the fly. When a new com server 180 is added to the network, it registers itself with the database application. Henceforth, any message destined for the new area 192 may be routed properly without altering the other area servers 180.

This method and system works for global messages or for user to user messages. Furthermore, new groups may also be dynamically added. Once added, each new group database server 184 registers itself with the existing database servers 184. This distribution of load permits nearly unlimited expansion with existing software and hardware. Each server manages a finite number of members, cumulatively serving a growing community.

Users need not be informed as to the particular com server 180 they should connect to. Members are directed to a single URL. The selection of the server for user connection is determined by load balancing software. In this manner, the network may appear to be a global network of servers or simply a local classroom.

The architecture described, which uses database servers as routing gateways, enables the system to serve with minimum administration and configuration and with lower end, cost-effective hardware.

In accordance with an embodiment of the invention, a "donut" of dynamic, hierarchical, shared user-profile information may be used. A donut may specify, for example, user characteristics, viewing preferences, hobbies, and spending habits as a user profile. The donut contains a user profile or acts as a key to a data repository containing a user profile, and the donut may be stored in an appropriate store in a suitable manner. For example, a donut may be stored in a file-type structure in a computer-readable memory. The donut may be accessed by browser programs, associated web server programs, and/or other applications for use in routing content to a user's machine associated with the donut. The user machine may include a wide variety of devices such as, for example, a personal computer, a television, a cable box, a satellite box, video game console, and/or a personal digital assistant.

Browser programs typically include a file created by a web server to locally store data and track web sites, identified by URLs, accessed by the user through the browser program. These files are referred to as cookie files, and contain a range of URLs for which they are valid. When the browser encounters the URLs again, it sends the corresponding cookie files to the web server identified by the URLs.

In an embodiment of the invention, browsers may access a donut file, and/or a database structure storing donuts, and web servers may include files or other database structures for storing copies of the donut. The donut file for a particular user is typically stored only on the server, but it may be stored locally on the user's machine, or on both the server and the user's machine. The donut implements a dynamic store of shared profile data which may be exchanged between client and server, and may have the flexibility to collect and process that data in three ways: client-side evaluation, http-based server-side evaluation, and network-based server-side evaluation. The donut has an advantage over a cookie file in that the donut file is database driven and typically stored on a server, preventing a user from deleting or otherwise tampering with the donut

file. Cookie files, in comparison, are stored locally on a user's machine and still access the user's donut file and receive content based upon the donut file.

A donut, more specifically, is an hierarchical attribute value pair data structure including a collection of crumbs. A crumb is the smallest unit of data corresponding to a meaning value pair associated with a particular donut. For example, a user's age would be a crumb associated with the user's primary donut. The donut data structure includes names plus associated crumbs. At the top level of the hierarchy, a top donut is associated with a user, a chat room, a network service, or other appropriate business entity or service. Each donut contains a set of crumbs and a set of sub-donuts.

A sub donut is a donut associated with another donut rather than an external entity such as, for example, a user. An example of a sub-donut is a user's address. The user's address references the user's primary donut and is stored in a sub-donut, and the sub-donut contains crumbs for each piece of information in the address. Donuts are stored with hierarchical relationships, meaning that a donut can have associated sub-donuts and the sub-donuts can also each have associated sub-donuts. The donut may thus have many levels of sub-donuts within its hierarchical structure.

Some donuts are owned by only one parent donut. Other donuts are shared among several or many parent donuts. These shared donuts profile a business entity, such as an individual program, which is common to all. An example includes members of the same chat room. This provides a way to identify a group of users for a chat service. Also, by sharing the donut a user need not enter a new profile for using different services; the services share and use the one donut.

Figure 10 is a diagram of logical structure 200 of a donut for implementing user-profile information. This structure may be stored on a computer-readable medium, such as a memory, for access by an associated machine. The term "donut" is used only as a label and refers to information residing on a server and accessible by a client for use in pushing or assigning particular content to the client.

Structure 200 uses database tables for storing and maintaining the user-profile information, which includes any type of information identifying a user or corresponding client machine. Structure 200 includes a user table 202, identifying a particular network user, and each user would typically be identified by a separate table. Table 202 may also include information identifying each user's name and an associated password and identification (ID). A directory table 204 maintains a directory listing of the network users.

A separate donut table 206 maintains user-profile information for a particular user. Donut table 206 is also associated with the corresponding table 202 for that user and with the directory table 204. Donut table 206 is also associated with a donut hierarchy table 208, which identifies and maintains hierarchical relationships for table 206. Each donut table 206 may include an associated crumb table 212 for use in identifying and maintaining particular attributes for the user-profile information. An associated attribute table 210 stores and maintains the information for those attributes.

The user profile may contain a wide variety of information concerning user characteristics for use in determining content to push to a user. As further explained below, the content may include any type of information such as video, audio, graphics, text, and multimedia content. Examples of content to be selectively pushed to the user based upon the user-profile information include, but are not limited to, the following: advertisements; player profiles for sporting events; music or other audio information; icons representing particular services; surveys; and program suggestions, Also, when a video program provides different video streams for different camera angles, such as a sporting event, the particular camera angle may be chosen based upon the user profile. In addition, particular drama presentations provide different video streams for various plot changes, and a video stream for a particular plot to be displayed to a user may be chosen based upon the user profile.

The surveys may involve selectively transmitting questions to a user based upon the user's donut. The user's answers to the questions may be used to further update the donut. As an example of survey content, consider an automobile manufacturer as an advertiser that has determined that the answers to ten questions, asked in a specific order, are vital to determining how the

manufacturer is going to market to a particular user. In this example, the server or network uses the donut to maintain what questions have been answered and which have not been answered. The donut can be utilized to determine which of the ten possible questions should be pushed to the user when the network determines, according to a particular schedule, that it is time for the manufacturer's survey to be pushed to the user.

This example further illustrates how individual advertising may be selected for particular users. The answers to the surveys may be used to provide a second level of information within an advertisement pushed to a particular user. The network may use demographic data in the user's donut, for example, to determine which advertisement and survey to push to the user. The user's answers to the questions in the survey may be used to push additional advertisements to the user or additional content for the advertisement already pushed. Also, the network may tailor a survey to a particular user by selecting additional questions for the survey, and an order of presentation of the questions, based upon the user's answers. Accordingly, the network may dynamically modify and update a user's donut to further fine-tune the process of selecting particular content to push to the user based upon the user's donut.

An execution environment for donuts may be implemented with an easily programmed JAVA module, an example of which is provided in Table 1 illustrating template-based code, generated by a graphical user interface (GUI). This module may be implemented as an application program interface (API) on a user's machine for accessing the user's donut file on a server. If the user's machine does not contain such a module, the user's machine may download it as a JAVA Archive (JAR) file for local execution.

Table 1

```
import actv.nub.*;
public class Pilot Yankees implements Processor {
        public void process (Nub,nub,Donut client, Donut action)
                throws NubException
        {
```

```
                if (client.check("OWNS_PILOT")&&
                        action,getValue("FAN_OF").equals(Yankees")) {
                    nub.change("content",YankeePilotChallenge);
                }
        }
}
```

The module in Table 1 can obtain crumb values from a donut service, branch on those values and other conditions, set existing crumb values, and create new crumbs as desired. Additionally, the module can interact directly with systems such as a distributed community network for additional functionality, including dynamic assignment to content and advertising push channels. An example of a distributed community network is described in US patent application No. 09/396693 filed 15th September 1999. The donut may be used with other types of networks as well.

A module processing a donut can execute in three modes: as an http servlet connected directly to a database for donut persistence; as a service responding to requests coming through a distributed community network and unconnected to a database for donut persistence; and on the client machine, which has direct access to user input, but uses the donut persistence by proxy through a distributed community network.

Figures 11A and 11B show a flow chart of a process 220 for generating and implementing donuts specifying user-profile information. Process 220 may be implemented in software modules on a machine such as a web server in the exemplary network described above. In the process 220, a user attempts to log onto the network (step 222). In response, the server determines if a donut exists in the database for this user (step 224). If a donut does not exist (step 226), the server queries the user, receives a response to the query, and generates a donut for the user based upon the response, using exemplary database structure 220 (step 232). An example of a query is provided below. Otherwise, if a donut does exist for the user (step 226), the server queries the user in order to update the donut (step 228) and implements the updates to the donut, using exemplary database structure 200 (step 230). The queries may

involve the server transmitting a series of questions to the user and receiving from the user replies to the questions. The queries may also involve survey questions as described above. The server may require a response in order for the user to receive content from the network.

The server may optionally receive user-profile content for the user from another source. For example, a user profile may already exist for the user from another network-based system, and the server may receive that profile. Upon optionally receiving the other user-profile content, such as a pre-existing user profile, the server selectively incorporates the content into the donut (step 234). The server may base the incorporation of the pre-existing other user-profile content on particular criteria such as the types of content required for the user profile and when the pre-existing user profile was updated. In addition, the server may incorporate all of the content or only selected portions of it.

The server assigns the user to a team based upon the user's donut and saves an indication of the assignment in the user's donut (step 236). A team specifies a chat room for this user for a chat service; the donut information may be used with other network services as well. If the user already had a donut in the system, the server may use a pre-existing room assignment as a default assignment, or assign the user to a new team, particularly if the user's donut has been substantially updated. The server stores the user's donut in the database and optionally transmits a copy of the donut or particular portions of it for storage on the user's machine (step 238). The user's machine, if it receives the donut, locally stores the donut or the updates to it.

After creating or updating a donut for the user, the server selects content for transmission to the user based upon the user-profile information stored in the user's donut (step 240). The selected content may be based upon particular criteria involving the user-profile information in the user's donut, and it may include a wide variety of types of information. The content may include networked content, meaning any type of information available via a Uniform Resource Identifier (URI) (step 254). A URI is a compact string of characters for identifying an abstract or physical resource. More specifically, URIs provide a simple and extensible means for identifying a resource, and a URI can be further classified as a locator, a name, or both. The specification of URI syntax and

semantics is derived from concepts introduced by the World Wide Web global information initiative. URIs include, for example, URLs and Uniform Resource Names (URNs). A URL is a subset of a URI which identifies resources via a representation of their primary access mechanism, such as their network "location", rather than identifying the resource by name or other attribute of that resource. The term URN refers to a subset of URI that is required to remain globally unique and persistent even when the resource ceases to exist or becomes unavailable.

The selected content may also include video content (step 255); audio content (step 256); or any type of multimedia content (step 257). The multimedia content may include, for example, particular types of animations or slide shows selected for transmission to the user's machine based upon the user's preferences or characteristics as identified in the user's donut. The video and audio content may include, for example, particular video and audio advertisements potentially of interest to the user and selected for transmission to the user's machine based upon the user's identified preferences. The video, audio, or multimedia content may include content related to a chat room discussion among the user and other members of the user's chat room or team. It may also include content related to a program being viewed by the user. For example, if the user views a sports program, the content may include statistics or videos of past sports programs between the same teams or players, depending upon the user's identified preferences or interests as saved in the user's donut.

The selected content may also include a particular video stream, as identified above (step 258). In particular, several video streams captured from different camera angles may be available for a particular program. The server may select a video stream for display to the user based upon the user's stated preferences. For example, a user may have stated a preference for watching the quarterback during a football game and, therefore, a video stream following the quarterback is selected for transmission to the user's machine. Another user may have expressed an interest in watching wide receivers, for example, and a video stream of the receivers is selected for transmission to that user's machine. One method for providing multiple video streams is described, for example, in

US patent No. 5861,881 and this method, or any other suitable method may be used to provide the multiple video streams.

The selected content may also include an executable object or application (step 259). For example, the server may transmit to the user a JAVA game or other types of electronic games based upon the user's preferences. The executable objects may also be used to dynamically push customized code to the user's machine while another application executes. In addition, selected executable objects may be used to facilitate electronic commerce transactions. In particular, identification of particular products available for purchase may be transmitted to the user based upon the user's preferences, and an executable object may be used to provide a common electronic "shopping cart" where the user is able to drag and drop identification of products to purchase among multiple vendors. The electronic shopping cart saves an identification of the products, and potentially other information, for use in transmitting and executing a purchase request for the products.

The selected content may include other types of content as well (step 260). Based upon the determination, the server pushes the particular content to the user's machine (242).

The server also monitors the user's activity in order to dynamically update the user's donut (step 244). The user's activity may involve any type of information relating to the user's interaction with the network or program content provided to the user. For example, the server may detect one or more of the following: the rate at which the user selects or "clicks on" URLs to request particular content; which URLs the user selects; the amount of elapsed time the user has remained logged onto the network; the extent to which the user participates in chat room discussions; and/or any other information which is detectible.

The server also determines whether to update the user's donut based upon the monitored user activity (step 246). This determination may be based upon particular criteria related to the user's activity. For example, the server may store particular types of activity or thresholds for activity and compare them to the user's monitored activity, providing for an update when the user's activity

matches the particular types of activity or exceeds the thresholds. It may also be updated based upon survey questions. If the server has determined, based on the criteria, that the user's donut is to be updated (step 248), it dynamically updates the donut based on the user's activity, saves the updates, and optionally sends the updates to the user's machine (step 250). Otherwise, if the criteria have not been met, the server does not update the donut.

The server also detects whether the user has logged off the network (step 252). If the user remains logged onto the network, the server continues to select and push content to the user based upon the user's donut (steps 240, 242, 254-260), monitor the user's activity (step 244), and dynamically update the user's donut (steps 246, 248, and 250).

The following provides an example of the use of a donut. During a program, the server sends a user, Bob Smith, a question which states, "Do you own a Personal Digital Assistant (PDA)?" Bob returns a "Yes" answer. The corresponding crumb includes the meaning value pair PDA=True, which is then sent via the Internet to a distributed community network, as identified above. This crumb is captured from the distributed community network and stored in the database as a crumb in the sub-donut "Technology" under Bob Smith's donut. A copy of the crumb is stored locally in the client browser on Bob Smith's machine.

Later in the program, the host of the television program is reviewing the latest PDAs. The producer of the program wants to send web content specific to all the users on-line about PDAs. The producer pushes to all the users on-line a playlist item, as identified above, parameterized on whether a user has a PDA. Bob Smith's browser receives the playlist item, which references his donut. The browser recalls the value pair PDA=True, and from the logic in the playlist item, determines that Bob should see in his browser a web page which offers a discount on a PDA upgrade, rather than a web page which offers the user the opportunity to buy a new PDA for the first time.

In addition to this route of the crumb and the decision making occurring on the client side, the same process may occur solely on the server side.

Table 2 provides an Extensible Markup Language (XML) code listing for an exemplary donut. Table 3 provides an XML code listing for an exemplary collection of donuts. In Tables 2 and 3 the indentation represents the hierarchical structure of the donuts.

Table 2

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE htvuser SYSTEM "donut.dtd" []>

<htvuser name="Bart Simpson" uid="bart" pw="dude!">
 <donut desc="bart" id="bart">

   <crumb desc="email" value="barf@fox.net" />
   <crumb desc="ccn" value="amex 5592 3800 0165 1872 exp 01" />

   <donut desc="thebox">
     <crumb desc="style" value="rock" />
     <crumb desc="region" value="NE" />
     <crumb desc="fanof" value="Offspring, Limp Bizkit" />
     <crumb desc="sex" value="yes" />
   </donut>

 </donut>
</htvuser>

<!--
Client code can refer to:
   bart.ccn
   bart.thebox.region
   bart.thebox.fanof
   etc.
-->
```

Table 3

```
<?ml version="1.0" encoding="UTF-8"?>
<!DOCTYPE teamlist SYSTEM "donut.dtd" []>

<teamlist>

    <donut        desc="team"
                  id="hot-sax-mets"
                  shared="true"
                  members="lisa,sally,lukas">
        <crumb desc="name"
               value="Hot Saxophonists Lusting after the Met's Shortstop" />
        <crumb desc="team-homepage"
               value="http://home.springfield.net/~lisa/mitt-lusters.html" />
        <crumb desc="trivia-tries" value="22" />
        <crumb desc-"trivia-correct" value="5" />
        <crumb desc="trivia-prizes" value="05" />
    </donut>

    <htvuser name="Sally Brown" uid="sally" pw="beethoven"">
        <donut desc="sally" id="sally">
            <crumb desc="email" value="sally@shulz.net" />
            <crumb desc-"cnn" value="mc 5592 3800 0165 1872 exp 012" />
            <donut desc="espn" subs="hot-sax-mets">
                <crumb desc="sports" value="skiing" />
                <crumb desc="agegroup" value="2-5" />
            </donut>
        </donut>
    </htvuser>
    <htvuser name="Lisa Simpson" uid="lisa" pw = "trane">
        <donut desc="lisa" id="lisa">
            <crumb desc="email" value="lisa@fox.net" />
            <crumb desc="ccn" value="visa 5592 3800 0165 1872 exp 02" />
            <donut desc="espn" subs="hot-sax-mets">
                <crumb desc="sports" value="curling.wwf" />
```

```
                <crumb desc="agegroup" value="5-10" />
                <crumb desc="education" value="Springfield" />
            </donut>
        </donut>
    </htvuser>
    <htvuser name="Lukas Doright" uid="duanne" pw="tomgirl">
        <donut desc="lukas" id="lukas">
            <crumb desc="email" value="lukas@aol.com" />
            <crumb desc="ccn" value="amex 5592 3800 0165 1872 exp 00" />
            <donut desc="espn" subs="hot-sax-mets">
                <crumb desc="sports" value="baseball" />
                <crumb desc-="agegroup" value="18-24" />
            </donut>
        </donut>
    </htvuser>

</teamlist>
```

It will be appreciated that modifications and variations may be made to the embodiments of the invention described and illustrated within the scope of the accompanying claims.

CLAIMS

1. A device for compiling and maintaining information for use in routing and transmitting content to a machine via a network, comprising:

a computer-readable medium including information for use in transmitting content to a machine;

said medium comprising fields for specifying an identification of the machine, an address of the machine, and user-profile information, for use in determining a type of content to transmit to the machine, the user-profile information being specified in an hierarchical attribute value pair data structure.

2. A device as claimed in Claim 1, wherein the computer-readable medium further specifies attributes of a user associated with the user-profile information and the hierarchical structure identifies hierarchical relationships among the attributes.

3. A device as claimed in Claim 1 or Claim 2, wherein the medium specifies information identifying the user.

4. A device as claimed in any of Claims 1 to 3, wherein the medium specifies information identifying preferences of a user.

5. A device as claimed in any preceding claim, wherein the medium specifies information identifying a room assigned to a user for a chat service.

6. A device as claimed in Claim 5, wherein the medium specifies information identifying members of the room for the chat service.

7. A device as claimed in any preceding claim, wherein the computer-readable medium further specifies an indication of a directory for use in obtaining instructions for routing the content.

8. A device as claimed in any preceding claim, wherein the medium includes user-profile information for use in selecting at least one of the following to transmit to the machine: information available via a Uniform Resource Identifier,

video content, audio content, multimedia content, a particular video stream, or an executable object.

9. A device as claimed in any preceding claim, wherein the medium includes user-profile information for use in selectively transmitting survey questions to the user.

10. A device as claimed in any preceding claim, wherein the medium specifies the address of one or more of a personal computer, a television, a cable box, a satellite box, a video game console, and/or of a personal digital assistant.

11. A method for compiling and maintaining information for use in routing and transmitting content to a machine via a network by specifying particular fields within a computer-readable medium, the method comprising the steps of:

receiving information for use in generating a user profile;

specifying in the medium, using the information, an identification of a machine, an address of the machine, and user-profile information for use in determining a type of content to transmit to the machine; and

storing the user-profile information in an hierarchical attribute value pair data structure.

12. A method as claimed in Claim 11, further comprising the steps of:

specifying in the medium attributes of a user associated with the user-profile information; and

specifying in the hierarchical structure hierarchical relationships among the attributes.

13. A method as claimed in Claim 11 or Claim 12, further comprising specifying in the medium information identifying the user.

14. A method as claimed in any of Claims 11 to 13, further comprising specifying in the medium information identifying preferences of the user.

* C L M S *

15. A method as claimed in any of Claims 11 to 14, further comprising specifying in the medium information identifying a room assigned to a user for a chat service.

16. A method as claimed in Claim 15, further comprising specifying in the medium information identifying members of the room for the chat service.

17. A method as claimed in any of Claims 11 to 16, further comprising specifying in the medium an indication of a directory for use in obtaining instructions for routing the content.

18. A method as claimed in any of Claims 11 to 17, further comprising dynamically changing the user-profile information in the hierarchical structure based upon updated information.

19. A method as claimed in any of Claims 11 to 18, further comprising querying the user in order to obtain user-profile information.

20. A method as claimed in any of Claims 11 to 19, further comprising transmitting content to the machine for a particular service based upon user-profile information.

21. A method as claimed in any of Claims 11 to 20, further comprising dynamically updating the user-profile information.

22. A method as claimed in Claim 21, wherein the dynamically updating step comprises:

monitoring activity of a user associated with the user-profile information; and

updating the user-profile information based upon the monitored activity.

23. A method as claimed in any of Claims 11 to 22, further comprising specifying the user-profile information for use in selecting at least one of the following to transmit to the machine: information available via a Uniform Resource Identifier, video content, audio content, multimedia content, a particular video stream, or an executable object.

24. A method as claimed in any of Claims 11 to 23, further comprising specifying the user-profile information for use in selectively transmitting survey questions to a user.

25. A method as claimed in any of Claims 11 to 24, further comprising specifying the address of one or more of a personal computer, a television, a cable box, a satellite box, video game console, and/or of a personal digital assistant.

26. A method of accessing information for use in routing and transmitting content to a machine via a network, the method comprising the steps of:

establishing a network connection from a machine;

accessing via the network connection an hierarchical attribute value pair data structure stored in a computer-readable medium; and

transmitting information via the network connection for specifying in the data structure an identification of the machine, an address of the machine, and user-profile information for use in determining a type of content to transmit to the machine.

27. A method as claimed in Claim 26, further comprising storing the data structure in a memory associated with the machine.

28. A method as claimed in Claim 26 or Claim 27, further comprising storing the data structure in a memory associated with a server having the network connection with the machine.

29. A method as claimed in any of Claims 26 to 28, further comprising dynamically updating the user-profile information.

30. A method as claimed in Claim 29, wherein the dynamically updating step comprises:

monitoring activity of a user associated with the user-profile information; and

updating the user-profile information based upon the monitored activity.

31. A method as claimed in any of Claims 26 to 30, further comprising selecting, based upon the user-profile information, at least one of the following for transmission to the machine: information available via a Uniform Resource Identifier, video content, audio content, multimedia content, a particular video stream, or an executable object.

32. Apparatus for accessing information for use in routing and transmitting content to a machine via a network, comprising:

network means establishing a network connection from a machine;

access means for accessing via the network connection an hierarchical attribute value pair data structure stored in a computer-readable medium; and

means for transmitting information via the network connection for specifying in the data structure an identification of the machine, an address of the machine, and user-profile information for use in determining a type of content to transmit to the machine.

33. Apparatus as claimed in Claim 32, further comprising storage means associated with the machine for storing the data structure.

34. Apparatus as claimed in Claim 32 or Claim 33, further comprising storage means associated with a server having the network connection with the machine, the storage means being arranged to store the data structure.

35. Apparatus as claimed in any of Claims 32 to 34, further comprising means for dynamically updating the user-profile information.

36. Apparatus as claimed in Claim 35, wherein said means for updating comprises:

means for monitoring activity of a user associated with the user-profile information; and

means for updating the user-profile information based upon the monitored activity.

37. Apparatus as claimed in any of Claims 32 to 36, further comprising means for selecting, based upon the user-profile information, at least one of the following for transmission to the machine: information available via a Uniform

* C L M S *

Resource Identifier, video content, audio content, multimedia content, a particular video stream, or an executable object.

