

INFORMATION RETRIEVAL SYSTEM

This invention relates to an information retrieval system. It is of particular application to teleconferencing facilities provided over the "Internet" but is not limited thereto. In this specification the term "teleconferencing" is used to describe a facility for allowing the sharing of information between two or more individuals at locations remote from each other, using telecommunications links for the purpose. Teleconferencing facilities include the capability to support voice communication between the parties, with additional capabilities such as the transfer, between the parties to the call, of information in the form of data, text, video images (possibly, but not necessarily, of the parties to the call) etc, such that all parties to the call can study the same data. The facility may be used for multi-way calls between three or more parties, but calls involving only two parties are not excluded.

The transfer of data in the form of text or images from one user to another according to standard Internet procedures can be quite lengthy, if the document to be transferred is substantial. In many cases, a document to be studied may itself be available from an information network and could be called up by each user independently. However, it is inconvenient for each user to have to individually generate the necessary address or search request (known as a url: "Uniform Resource Locator"), as such requests require a complex string of alphanumeric and punctuation characters to be entered, for example <http://www.bt.co.uk/about/site/how/index.htm>. Unless all users enter the same address (which in turn requires that they all know, or are told, the correct character string) they cannot access the same information.

A system is described in the present applicant's International Patent Specification WO98/13995, published on 2nd April 1998, in which users connected together in a teleconference can share a URL with each other. This may be useful for instance when a person wants others with whom he is talking to see the same WWW page at which he is looking. To share a URL, a user either manually types, or copies and pastes, the relevant text string into an HTML (hypertext mark-up language)-generated text box in a shared field to which all the users have access. The shared field contents are made available to other users through a dynamically-generated hyperlink. All other users can then load the document referred to by the

URL by clicking on the link. However, this merely gives the users the necessary character string, and does not control when the individual users access the network using this URL, so different users may be looking at different versions of the same "page".

5 According to the present invention there is provided an information retrieval system comprising at least two user interfaces, each user interface having means for accessing an information database system by means of retrieval commands, and wherein at least one, controlling, user interface has means for transmitting, to one or more of the other user interfaces, a signal containing a
10 retrieval command, the other user interfaces being operable under control of the retrieval command signal to transmit the retrieval command to the information database to retrieve a specified item of data.

 According to a second aspect there is provided a method of controlling an information retrieval system comprising at least two user interfaces, each user
15 interface having means for accessing an information database system by means of retrieval commands, wherein one, controlling, user interface transmits, to one or more of the other user interfaces, a signal containing a retrieval command, the other user interfaces operating under control of the retrieval command signal to transmit the retrieval command to the information database to retrieve a specified
20 item of data.

 In this way a user can minimise the amount of data to be transferred between himself and other users, allowing one user to access the data from the correct address on behalf of all the others, without the need to individually generate the retrieval command.

25 This allows one user to take control of other users' user interfaces (for example WorldWideWeb browsers), thereby ensuring that all browsers stay in lockstep throughout an online meeting, seminar, lecture etc. When used in conjunction with a textual "chat" application, or a teleconferencing system, this capability dramatically enhances the utility of the WorldWideWeb for group use, in
30 both synchronous and asynchronous contexts. Many types of application can be envisaged e.g. online business meetings, entertainment, and education.

 Preferably, the user interfaces are linked to each other by a server application. The instructions sent from one user to the others by way of the server

This Sheet Has Been Replaced



THIS SHEET HAS BEEN

may be a search instruction, but in general will be the address of the item of information which is to be retrieved.

To avoid the possibility of two users attempting to perform this operation simultaneously, a process may be applied wherein the receiving users confirm that
5 they accept control of their user interfaces for this purpose by another user. Alternatively, the system may be set up to have one of the user interfaces designated to have control of the interfaces of other users in a specified group. In this arrangement the other users may themselves transmit a proposed information address to the controlling user, for him to disseminate to the other users.

10 The present invention is particularly suited for use with Internet-based teleconferencing applications, which have recently started to become available. They allow groups of people anywhere in the world to talk to each other using packet switched protocols such as RCP. These systems allow teleconferences to be set up and controlled relatively easily via a graphical user interface (GUI). This
15 communicates with a server-based application which controls which people can talk to each other. Users can see text labels or images which represent both themselves and other users and can take advantage of the relatively intuitive and powerful control and feedback facets of a graphical user interface. Many of these systems follow a 'rooms' based metaphor in which each teleconference takes
20 place in a different "virtual" room. Users can wander from room to room taking part in conversations as they go.

An embodiment of the invention will be described with reference to an improved teleconferencing system of this type, which is the subject of the International Patent Specification WO98/13995 referred to above. This, in turn,
25 makes use of the Acculab "Millenium CT" (Registered Trade Mark) platform, a commercially-available PC-based device described in International patent specification WO98/13993, also published on 2nd April 1998.

An information retrieval system according to an embodiment of the present invention will now be described, by way of example only, with reference
30 to the accompanying Figures in which:

FIG. 1

Figure 1 shows a diagram of the system platform and its context;

Figures 2, 3, 4, 5, 6, and 7 relate to the prior system described in WO98/13995, and will be discussed later

Figure 8 is a schematic representation of the teleconferencing environment within which the system operates

Figure 9 is a screenshot when running the system of the invention, with a Java "applet" running in the left-hand frame of a WorldWideWeb browser;

5 Figure 10 illustrates a main window for the applet;

Figure 11 illustrates a browser window showing the requested URL in the right hand frame;

Figure 12 is a flow diagram illustrating the processes by which one user controls the browsers of others;

10 Figures 13, 14 and 15 illustrate in more detail the information flows between the client and server applications in the server.

The operative elements of the system will first be described with reference to Figure 1.

Each user has a personal computer 100 and a telephone 115, connected, 15 respectively through the "Internet" 105 and the normal telephone network (PSTN or ISDN) 110, with a server 120 under the control of an Acculab "Millenium CT" PC-based system 135, which provides the teleconferencing facilities to allow co-ordination between the various users, and between the telephone and internet connections, as described in detail in International Patent Specification 20 WO98/13995. This teleconferencing system provides a management and control unit for a network-based teleconferencing system, the unit comprising:

- i) an interface for outputting control signals to a platform for establishing audio connections across a network between users;
- ii) an interface for receiving control signals from at least one platform 25 for providing a graphical user interface to a user for use in controlling the network-based teleconferencing system; and
- iii) access to a database for maintaining, including updating, management data relating to one or more existing teleconferences

such that the management and control unit can receive control signals 30 input by a user at the graphical user interface in respect of a teleconference, output control signals to the platform for establishing audio connections, thereby establishing a teleconference connection between the user and at least two other users over the network, and output management data to the graphical user

interface during an existing teleconference for use by the user in managing the teleconference.

Preferably the network is a telecommunications network while the interface for receiving control signals is an interface to a data network, such as the Internet. Preferred embodiments can then enable users to enjoy high quality audio-conferencing which they can manage using a World Wide Web screen-based interface. Such embodiments can allow users to work on worldwide web based material yet not require that they

- i) set up calls via an operator,
- 10 ii) remember DTMF control codes,
- iii) invest in new telephony hardware, or
- iv) install specialist software.

This system avoids the problem, particularly associated with carrying voice signals over packet-switched protocols, that voice quality is contingent upon both the bandwidth of the connection to each user being sufficiently large and the overall performance of the network being above a given threshold. This makes it difficult to guarantee an acceptable level of sound quality for all users at all times. There is also a problem caused by delays in digitally encoding speech from each user; this means that users hear their own voices repeated after a delay at the remote end(s) unless all users wear headphones. Finally some service providers are seeking to ban, limit or charge extra for services which use demanding protocols such as Remote Procedure Calls (RPC).

The management and control unit can be supported by Internet-connected servers, such as a Web server for supplying documents, and a Java server for coordinating remote interfaces, whilst the graphical user interface may be provided at a client, also connected to the Internet. The management and control unit can then provide a powerful and very versatile tool in providing audio-conferencing.

Conferencing systems according to preferred embodiments of the present invention can combine the ease of use of a GUI-based system whilst also leveraging the reliable voice quality associated with the phone network. Such systems can impose minimum technical or cognitive requirements on each user and preferably use established protocols wherever possible.

The system can allow anybody who has simultaneous access to an Internet (or similar) connection, to provide the graphical user interface, and a

separate directly diallable phone line to setup, control, record and clear down high quality teleconferences.

As described in the earlier cases, the database can be used to maintain updatable information specific to each user. This information can include for
5 instance images of the users involved in a teleconference so that, whilst they are using the system, they can see annotated pictures of anybody else who is connected.

Not only is there no need for users to remember control codes but they do not need to know the phone number of other participants. Teleconferences can be
10 made private and users can change their outgoing telephone number when they move from one location to another.

Clearly, this system could be used with other information networks, for example a less extensive network than the Internet, such as a company "Intranet".

Tight integration between software running on the client, the associated
15 WWW server, a teleconferencing platform and a database is possible.

A World Wide Web (WWW)-based graphical user interface (GUI) can control a teleconference using this system. No additional software is necessary at the client and the system can be available to any user with a Transmission Control Protocol/Internet Protocol (TCP/IP) connection to the Internet and a phone line.

20 The system can be made capable of keeping track of several parallel teleconferences each of which may involve several users. Each user can be shown the appropriate information detailing where they are on the system as well as where others are; this information can be updated whenever changes relating to a given user occur.

25 Each user can preferably control aspects of the system which they have the privileges to change. This should preferably be done without creating a conflict.

Finally the entire system is preferably designed so that users can be billed appropriately.

30 The system of the present invention is an application 170 which may run on any computer forming part of the user group, for example the same computer as that supporting the web server 120. the Java server is built on an object-oriented client-server framework written in Java as outlined by David Flanagan [*Java in a Nutshell*, O'Reilly & Associates Inc.]. Java client applets communicate

with the Java server 170 using "sockets" objects. The job of the server 170 in the system of the invention is to synchronise communication between the applets, and to manage other client requests such as indicating which user is currently in control of the set of group Web browsers.

- 5 The server is basically responsible for managing state: users are members of groups, may move between groups, and may make use of group resources.

Previous work by Smythe *et al* on RISE (Real-time Interactive Social Environment), to be found at:

<http://rise1.labs.bt.com:8000/rise.bundle/public/index.html>

- 10 has led to the development of a sophisticated database-driven Web real-time teleconferencing platform offering such group services. RISE is therefore an ideal environment in which the system of the invention may be exploited.

The system of the invention offers services which in several respects mirror those of PointCast, Marimba and other 'push' providers. The crucial
15 difference is that content organisation and management is delegated to each client, so that the users can freely control which Web resources are to be broadcast to the group.

Figure 8 shows a group of such users' computers 100, 101, 102 linked through the internet 105 to a java server 170, which has the individual users
20 identified as a user group by an attribute 160 stored in an address file 155 stored in the server. One user (100) is defined as the controlling user, again by means of a server attribute 165. (Note that throughout this specification each user is identified by the same reference numeral as his or her respective user interface). Each user 100, 101, 102 has a respective browser program 150, 151, 152, with
25 which URLs of other websites 145 (only one shown) can be generated for accessing data from the website, again through the internet 105. They are also linked to a Java server

The system of the invention is operated in a manner which will now be described.

- 30 The sequence of screenshots shown in Figures 9, 10 and 11, and the information flows shown in Figures 13, 14 and 15, illustrate how the system of the invention is used. In Figure 9, the user currently having control of the system has loaded a pair of frames into Internet Explorer. The left hand frame contains the program - a Java applet. The right hand frame in this example is the target for

all URL requests made by the applet. The 'url-push' button in the left-hand frame is used to reveal the main window, shown in Figure 10.

Figure 12 is a flow diagram showing the main information flows taking place between the users and the server 170. The three users 100, 101, 102 are all
5 connected as a user group, identified as such in the attribute list 155 of the server 170, with a first user 100 having control of the browsers 151, 152 of the other users 101, 102. This control facility is stored as another attribute 165 in the server 170. It will be seen in Figure 16 that the attribute 165 may be transferred within the server from one user 100 to another 101. The user currently having the
10 attribute 165 is the controlling user.

The main window contains the user interface. Three text fields are used to organise URLs. The controlling user 100 types or pastes URLs into the top text field which is editable (step 902). Pressing the 'Add URL' button results in the new URL being added to the list of URLs shown in the middle text field (step 903).
15 The program conducts an internal check that the URL is of valid construction, and warns the user if this not so (step 904). The middle text field contains the list of URLs that the user wishes to show to other users in the group. Clicking on any URL in the middle field, or clicking on the up/down 'arrow' buttons transfers the URL to the bottom text field (step 905).

20 The controlling user 100 presses the "PushURL" button to drive the URL to the remote users' browsers 151, 152 (step 906). A status message stating "Pushing URL <url>" appears. The URL is rapidly distributed to all users (step 907). A status message stating "Being pushed to URL <url>" appears in all users' windows (step 908). The applet then forces each user's browser to load
25 the requested URL (step 909), as shown in Figure 11. All the users can then simultaneously view the results of loading the URL.

Figures 13, 14 and 15 illustrate in more detail the processes which take place in the server and clients in order to load a new URL into each browser belonging to the group. Each diagram shows a sequence of messages passed
30 between various objects. Fig. 13 illustrates the message flow in the sender client applet up to the point where the URL is passed down the network to the server. Fig. 14 shows what happens in the Java group-coordination server 170 in order to redirect the URL to all clients. Fig. 15 shows the response of a receiving applet to the URL pushed at it from the server. The client-server architecture used can be

replaced by an architecture in which the applets and application communicate via distributed objects through remote-method-invocation (RMI), CORBA or similar process.

The Java group-coordination server 170 co-ordinates the client user
5 interfaces by allocating connection "objects" 712, 715 (see Figure 14) to each client. Service requests from the clients 100, (101, 102) are passed through these connection objects. URLs are broadcast to the group of which the user is currently a member (defined by a connection vector object 714), and the client processors (101, 102) then change the page in the user's browser (also defined as an "object"
10 723). Other objects in the transmitting user's set-up 100 define the URL "push" button on the display of the user 100 (object 701), the user interface (702), the text field (703), the client (705) and the output (706). the server 170 has objects defining the server itself (711), the sending user 100 (object 712), and the data input (713), and the receiving user has an object defining the input (object 721).
15 Other objects 704, 722 are created during the process. Other objects, not involved in the essential processes to be described, and not shown in Figures 13, 14 and 15, communicate with the database, to store URL requests for example, or to identify who is entitled to 'controller' rights.

Figure 13 illustrates the transmission of a URL from a user 100 to the
20 server 170, for onward transmission to other users 101, 102. When the user "pushes" this button (step 801) (in practice, by "clicking" on it with his "mouse") a "push URL button" object 701 transmits this event to a browser control interface frame 702. This frame 702 is configured to monitor for such an event, and on detecting it (step 802) generates an instruction to send the URL to the server (step
25 803). This causes the interface frame 702 to obtain the URL character string from the text field (object 703) and generate a new object "url" (704) (steps 804, 805). This string is then written to the client object 705 (step 806) and thus to an output object 706 (step 807).

Figure 14 shows how the server 170 reads URL requests received from
30 one user 100 and routes them out to other client objects 101, 102. The server object 711 and sender's connection socket object 712 are set to run (steps 811, 812). On receiving a character string the connection sends a message (step 813) to an input object 713 which in turn instructs the server object 711 to control the

This Sheet Has Been Replaced



THIS SHEET HAS BEEN
REPLACED

connection set-up objects 714 (step 815) and then transmit the URL instruction to each connection 715 (step 816).

Figure 15 shows how the receiving "client" 101, 102 responds to such an instruction. Whilst the listener object 721 is set to run (step 821) it obtains any
5 URLs transmitted to it from the server 170 (step 822). Any such URL is used to create a new "url" object 722 (step 823) which is transmitted to an Applet context object 723, namely the web browser 120, causing it to load the URL and access the data required.

Figure 16 shows the process employed when one of the remote users
10 (101) wishes to take over control from the current controlling user 100. The user 101 transmits a control request (step 910) to the user 100, this request being stored as an attribute of the requesting user in the attribute list 155 of the server 170. On receiving such a control request (step 911) the controlling user's computer 100 will display the request. The controlling user 100 may have the
15 authority to refuse such a request (step 912) in which case a message is transmitted to the server 170 and thus to the remote user 101 communicating the refusal. However, assuming that the user 100 accepts the request (step 912) the server 170 modifies the existing application objects in the attribute list 165 relating to the current controlling user 100 and the remote user 101 (step 913), to record
20 the requesting user 101 as the current controlling user and to remove that attribute from the user 100. The user 101 has now become the controlling user and can proceed as described above with reference to Figure 12 (step 914).

The attribute list may allow some users to refuse a control request 910, and others to be required to relinquish control on request from any, or a specified
25 subgroup, of the other users. For example, in Example 2, discussed below, individual participants 101, 102 may be allowed by the initially controlling user 100 to take control, but cannot refuse to return control to the controlling user 100 when he requests it.

In an alternative arrangement shown in Figure 17 the controlling user 100
30 retains control throughout. However a remote user 101 may prepare a URL, and transmit that URL, by way of the server 170 (step 901) to the controlling user 100. The controlling user may choose to select or refuse a proposed URL (step 902) and, if selecting the URL, will then add it to the list (step 903) as described above with reference to Figure 12. The process then continues as described with

reference to Figure 12. In this embodiment the controlling user 100 retains control, "pushing" URLs generated by other users 101, 102.

The controlling user interface 100 may use its internal timer to transmit a sequence of "push" commands to control the remote browsers 151, 152 such that
5 a sequence of pages may be downloaded to the browsers at a fixed rate.

The controlling user 100 may also store URLs or sequences of URLs for each user group in a database 125, for playback. This can be augmented by audio recordings of group conversations or other teleconference elements, as described in the earlier application.

10 The system of the invention can be exploited in any situation where groups of people distributed over a Wide Area Network need to share visibility of Web pages. Four example scenarios will now be described, with reference to Figure 8. For simplicity, the same reference numerals will be used for analagous performers in these scenarios, but it will be appreciated that different websites and
15 users will be fulfilling these functions in each example scenario.

EXAMPLE 1 BUSINESS - Sales

The customer (101) wishes to obtain a mortgage. She checks the WorldWideWeb,
20 and finds that there are a large range of options advertised, some of which offer online financial analysis of her situation. Due to flaws in the design of Web service, or because of safeguards required by the financial services industry to prevent customers making decisions purely on the basis of on-line information, without professional advice, she does not have the URLs needed to "drill-down" from the
25 high-level websites to some key URLs which could prove advantageous to her.

On some of the Web pages she consults, there is a link to a respective online sales representative 100. The customer clicks on such a link, which causes her telephone to ring that of the sales representative 100 (see Figure 1: link 110).
30 Whilst the customer 101 is talking to the sales representative 100, the sales representative asks for permission to drive the customer's Web browser 151, and the customer agrees by clicking an acceptance. This creates an association in the server 170, forming a two-member user group in which the customer's browser 151 is controlled by the sales representative's browser 150. Thus when the server 170 receives a "push" request from the sales representative's browser 151 (step

906), it transmits it (step 907) to any browser identified in the server's attribute list as being linked to the browser 150, namely the customer's browser 151. On receiving the URL from the sales representative, (step 908) the browser 151, like the browser 150, can access the data from the website 145 (step 909). The sales
5 representative can now lead the customer through a series of web pages not otherwise accessible to the customer, the necessary access passwords being included in the URLs transmitted to the customer's browser 151 for the purpose, but not being stored by, or known to, the customer 101 herself.

When the customer has located a mortgage option which appears to suit
10 her needs, the sales representative can offer step-by-step advice to the customer on how to perform the online analysis of her financial situation. When the customer is satisfied with the results she can sign up for the mortgage.

15 EXAMPLE 2 ENTERTAINMENT

A football club, having a website 145, wishes to broaden the scope of its match-day media offerings to its supporters. It signs up a radio station (user 100) to deliver multiple telephony/Internet talk-shows linked to live broadcast match
20 commentary, using the telephone/Internet interface shown in Figure 1. The radio station (100) offers the services of talk-show hosts who are able to select music and sound clips to be played over the telephone in traditional disc jockey fashion, but who are also able to select photographs and other data relating to the match from the website 145 (steps 902-905), and push them to the supporters 101, 102
25 (steps 906-908) using the system of the invention, so that the photographs etc can form the context for the supporters' conversations. The supporters 101, 102 can also access data and photographs (steps 900, 901) for submitting to the host for transmission to the other supporters (steps 902, 903: Figure 17), so that the supporters can control the context for their discussions under the overall
30 supervision of the host 100.

Alternatively, the host 100 may allow one supporter 101 to take control and transmit "push" commands to the host 100 and other supporters 102 (steps 910-914) himself.

EXAMPLE 3 EDUCATION

A tutor (100) can push individual pages (145) from the vast educational resources of the Web to her students (101, 102), located remotely, as they discuss a
5 research project together.

EXAMPLE 4 ARCHITECTURE & DESIGN

The development of a large engineering project, such as a new airport terminal,
10 involves many interested parties, such as the client, the architects, the intended users (airlines, hotels, consumer outlets), government organisations and other stakeholders. These parties can spend years shuttling back and forth between meetings concerning the major issues. There will be no generally accessible single repository of design and planning documents, or records of previous meetings. A
15 Web-based 'virtual organisation' can be formed, which allows the management of teleconferenced meetings, the storage of records of such meetings, and ready access of the latest version of the design documents. At a teleconference meeting involving some or all of the participants (100, 101, 102), data from each participant's website (100, 101, 102, 145 etc) can be made available to all the
20 others, under the control of a chairman (100). The chairmanship role can be transferred to another participant 101 during the teleconference (steps 910-914: Figure 16), for example if that other participant 101 is to make a presentation to the teleconference.

25 There follows a verbatim description of the preferred embodiment of the aforementioned International patent specification WO98/13995, which describes the Millenium CT-based teleconferencing facility on which the preferred embodiment of the present invention is based. It should be noted that the United Kingdom applications 9619958.3 and 9707712.7 referred to therein have since
30 been used to establish priority for the other International Patent Application, published on 2nd April 1998 as WO98/13993, to which reference has already been made above.

Figure 1 has already been briefly described.

Figure 2 shows a schematic diagram of the server side architecture for the system shown in Figure 1;

Figure 3 shows a graphical user interface for use at a client browser for
5 the system of Figure 1; and

Figures 4, 5, 6 and 7 show functional breakdowns of objects for use in an application residing on a server in a system as shown in Figure 1.

Referring to Figure 1, each user has a personal computer 100 with an Internet IP
10 connection 105, and a web browser application which supports HTML3.2 (or later), together with Frames and either Javascript or Jscript. They must also have a separate telephony connection 110 with a telephone 115 which can be directly dialled. Possible network configurations to support this include the use of

- 15 a) 2 Public Switched Telecommunications Network (PSTN) lines,
- b) an Integrated Services Digital Network (ISDN) 2 connection where one B channel is used for voice and the other for data, or
- 20 c) the combination of a Local Area Network (LAN) and Private Branch Exchange (PBX) connection.

At the server end of the system, there is a Worldwide Web server 120 which can dynamically produce HTML pages by Simple Query Language (SQL) based
25 reference to a database 125. It can also write information to the database 125.

The WWW server 120 is also connected via a RFC1006 socket level connection 220 to an Acculab Millenium CT (RTM) platform 135. This is a PC-based device, described in copending British patent applications 9619958.3 filed on 25th
30 September 1996 and 9707712.7 filed on 16th April 1997 by the same applicant, the contents of which are incorporated herein by reference, which has a number of capabilities including the ability to set up, control and record audio conferences. Prior to its launch as a product it was known as Minor Applications Platform (MAP). The Millenium CT is connected between a telecommunications network,

- such as the Public Switched Telecommunications Network (PSTN), and a data network, such as the Internet. It accepts incoming service requests over the PSTN via an ISDN30 connection 140. It can also accept incoming service requests in other ways, for instance via a RFC1006 socket level connection as mentioned
- 5 above. In this case, the protocol will be implemented on both sides of the socket level link. It also provides processing capacity and it can respond to an incoming call or message by identifying and launching an appropriate computing application which calls on and manages resources to run the service requested.
- 10 The Millenium CT 135 is equipped with means for providing audio bridges between conference participants, in the form of digital line interface cards and controls therefor. Additionally, the Millenium CT provides speech related resources, such as recording and delivery. It therefore can provide facilities which are important in communicating with a user during conference set-up and for recording.
- 15
- For the purposes of the audioconferencing system however, any device which can accept and send commands over a network connection and use these commands to generate and record audio conferences could be used as an alternative to the Millenium CT. It should be noted there are also a number of equivalent link
- 20 protocols which could be used in substitution for ISDN30.

Server Software Architecture

- The Server 120 may be any workstation with a WWW server and an object-oriented development environment providing integrated database access. The
- 25 system shown in Figure 2, provides a "WebRex" (TM) WWW server 120, an "Oracle" (TM) database 125, and a "NextStep" (TM) operating system, and four objects.

- Referring to Figure 2, the server 120 supports an application comprising four
- 30 objects. These objects 200, 205, 210, 215, are further described below. Functions within the objects are called from other objects by name; the NextStep operating system automatically generates the messages interchanged between the objects in a manner similar to Remote Procedure Calls (RPC).

The application object 200

This is linked into the WWW server 120 to provide a set of multiply re-entrant, memory resident functions. These functions are called in an HTML request from the user and allow conferences to be set-up and managed using functions provided
5 by the Millenium CT interface object 205 (or MAP object).

The MAP object 205

This receives requests from the application object 200 and controls the Millenium CT 135 by sending it commands and polling it for commands and for responses to
10 previously sent messages. The RFC1006 protocol (running over TCP/IP) is implemented in the MAP object 205 and on the Millenium CT 135 to provide a reliable peer-to-peer protocol for message passing.

As it may take some time (a few seconds) to process a request from the
15 application object, the MAP object queues a request in a CONFERENCE-REQUEST database table (further discussed below under "The Database"). An initial (HTML) response is sent back to the client indicating that a request is being processed.

The MAP object 205 is implemented as an event driven state machine since the
20 requests usually require a number of messages to be interchanged with the Millenium CT. The state of the request is stored in the CONFERENCE REQUEST database table entry (further discussed below under "The Database"). The MAP object 205 polls the CONFERENCE-REQUEST table for user requests frequently (for instance once every 5 seconds). Once a request is found, the command sequence
25 is initiated by sending the first command to the Millenium CT 135. The MAP object 205 polls the Millenium CT 135 for a response at regular intervals (for instance once every second); once a response is received the MAP object 205 identifies which conference the response refers to (from a conference number field in the response). It then determines the current state from the request stored in the
30 CONFERENCE REQUEST table and issues the next command to the Millenium CT 135.

The messages sent to the Millenium CT 135 are:

- Register Platform (establishes a TCP/IP connection)
- Conference Registration (register or de-register a conference with the Millenium CT)
- Call Dial (call a specified telephone number)
- 5 • Mix all Participants (mix the calls into an audio conference)
- Call Clear (remove caller from the conference)
- Record Start (start recording the conference)
- Record Stop (stop recording the conference)
- Record Delete (delete the recording)
- 10 • Record Save (save the recording to a file)
- Playback Start (playback the recording from the beginning)
- Playback Stop (stop the playback process)

and the commands returned by the Millenium CT 135 are:

15

- Call clear (call cleared by remote party)
- Record stop (stop recording the conference (run out of resources))

The Database Interface object 210

- 20 This may more simply be called the Database object 210. It provides functions for use by the application and MAP objects 200, 205. The functions store and retrieve information about the people who are using the system and the audio-conferences in an Oracle database 125. The functions use embedded SQL.

25 The Heartbeat object 215

This logs users out if it believes the user has closed down their meeting place window and left the system. This is further described under "The Heartbeat Process" below.

30 Millenium CT Application

In order to implement the setting up and management of a conference from the server, the Millenium CT 135 is provided with an application which can receive commands from the MAP object 205 and respond by supplying the functions required. These are for instance establishing audio bridges between the

- conference participants, delivering speech messages, such as "Please wait", and recording and storing sound for later access. The form of such an application of course is adapted to receive and respond to the messages set out above under "Millenium CT Object 205", and this determines the functionality of the Millenium
- 5 CT application. The application will be compatible with the operating system of the Millenium CT 135 which may for instance be UNIX but could be another operating system.

The database

- 10 The database 125 is used to store information about people using the system and about audio conferences. Any relational or object-oriented database is sufficient though the present system uses an Oracle database accessed using Simple Query Language. This is called from functions in the application and Database objects 200, 210. Oracle sequences are used to generate unique conference numbers.
- 15 The tables used in the Oracle database are as follows:

TABLE NAME	FUNCTION
CONFERENCE REQUEST	A queue of requests from users to take part in an audio conference
CONFERENCE	A list of all the current conferences together with information on whether they are being recorded
PERSON CONFERENCE	A list of the people who are in each conference
PERSON	Information about each person such as their login name, password, on-line status (1 = logged in, 0 = logged out), personal URL (see under " <u>Sharing URLs</u> " below), image and conference number (if in a conference).
ADDRESS	Contains each person's telephone number

Logging onto the system

The database 125 holds information which includes the name, phone number and image of each user who is registered with the system. A registered user logs onto

the system by starting up their browser 100 and then submitting a request for the system's access URL to the server 120.

Referring to Figure 3, the user's browser 100 uses the javascript "OpenWindow"
5 command to open up a main window as well as a smaller secondary window 300 which represents the 'meeting-place'. Additionally the on-line status field in the PERSON table in the database 125 is set to a value of 1 to indicate that they are logged in and their 'heartbeat' is initiated (see under "The Heartbeat process" below) by setting the heartbeat field in the PERSON table for this person to the
10 current time to initiate the heartbeat process.

The meeting-place window 300 consists of 4 frames. A column 305 on the left shows a scrollable list of people who are logged on to the system. Beneath this, there is a very small frame 310 which is used to control the update process. In a
15 right hand column, there is a main frame 315 which provides details about either a person or a conference. Below this, there is a smaller frame 320 which contains controls for recording the conference, setting privacy or sharing URLs.

As soon as a user logs onto the system, they see the names of all other users who
20 are logged on at that time as well as a list of the current conferences and their participants in the form of a scrollable text list. The application object 200 achieves this by retrieving a list of those who are in the CONFERENCE and PERSON CONFERENCE tables in the database 125. It also gets a list of those who are logged on but not in a conference by looking at who has an on-line status
25 of 1 in the PERSON table.

Others will also see the new user's name as soon as their respective meeting-place windows 300 are updated. When the system starts up, users are shown their own image together with the telephone number at which the system is currently
30 programmed to dial them. All audio conferences taking place on the system are given a two or three digit identifier (room number) when they are initiated to assist users in telling each other where to rendezvous (the numbers are taken from a cycling range from 01 to 999). The arrival of a new user online is indicated by the temporary display of an indicator such as a coloured dot 335 adjacent the name (or

names) of anybody who is newly arrived (see the "Butler" process below) and by the updating of the meeting-place screen of each user.

(Figure 3 shows "Andrew" twice. These are different people.)

5 Meeting-place User Interface

The Meeting-place User Interface appears as a window 300 generated by a user's web browser on their personal computer 100. Figure 3 shows a point where a user (Philip) has just clicked on the 'ROOM 17' link in the logged-on frame 305. This causes the system to show images of Andrew and Debra who are currently talking in this room by accessing the tables in the database 125. An icon 325 in the room frame 315 also shows that the conference is being recorded. User 'Andrew' has just logged onto the system so the Butler process is showing an indicator 335 adjacent to his name. Selecting the 'Enter Room 17' button causes the system to call up Philip's number and add him in to the conference.

15

Starting a Conference with Somebody who is Logged Onto the System

To find out about another user who is currently connected, the user clicks on the person's name. They are then shown a picture of the person together with a note of the dialback number of the nearest phone (see "Telephone Number Assignment and Mobility" below). There is also a button which gives the user (henceforth called the "originator") the option to set-up a conference. Once pressed, the request to set-up a conference is sent to the server 120.

The server 120 queues the request in the CONFERENCE-REQUEST table. The MAP object 205 subsequently retrieves the request from the table, creates entries in the CONFERENCE and PERSON-CONFERENCE tables, and instructs the Millenium CT 135 to set-up the conference using the following commands:

- Conference Registration (register a conference with the Millenium CT)
- 30 • Call Dial (to the originator)
- Call Dial (to the other person)
- Mix all Participants (mix the calls into an audio conference)

When the originator answers the call they are played a message telling them that an audio conference is being set up and asking them to wait. When the called person answers they too are given a message telling them that a conference is being set up.

5

Once both parties have answered, the calls generated by the system are connected together into a telephony based audio-conference and both entries in the PERSON table are updated with the conference number. The MAP object 205 updates the status field (to indicate success) in the CONFERENCE-REQUEST table
10 in the database 125.

Whilst the conference is being established, the originator is shown a screen of cycling dots (generated by an animated GIF graphic) asking them to wait. Meanwhile a small secondary frame is reloaded using the client-pull HTML
15 construct (for instance every 5 seconds). When the server receives a reload request, it inspects the status in the CONFERENCE-REQUEST table. If the status indicates the conference has been set-up, it returns HTML to the updated frame which causes the entire layout of the window to be reloaded. This mechanism thus updates the meeting-place windows 300 of both people to show they are
20 connected in a conference and deletes the request from the CONFERENCE_REQUEST table. All other people who are connected to the system also have their meeting-place windows updated to show that the conference is in progress. This is achieved by updating the 'dummy' person entry in the PERSON table with the current time to indicate that the displays should be updated (see
25 "The Update Process" below for details).

Telephone Number Assignment and Mobility

In environments such as schools, users of the system may wish to log on from any one of a number of terminals. To prevent the user from having to manually assign
30 themselves the telephone number of the nearest phone each time they log on, this number is stored on the hard disk of each client machine. This can be achieved for instance by using a system such as that known as the "Netscape Client-side Cookie". This specifies the telephone number for the machine as well as a text string describing where it is located, for example "Andrew's office". When a user

initially logs on to the system, a Javascript function located in the rooms frame of the meeting place will attempt to read any cookie which has previously been set by the system. If no cookie is found, or if a previously set cookie has expired, the rooms frame in the meeting place will inform the user that no telephone has been
5 associated with the machine that they are using. (Note: cookies are set to have a fixed life, for example one year, after which they expire.) The rooms frame will then invite the user to choose a telephone by selecting from a pop-up menu of numbers which are allowable for that particular user. (This page is generated dynamically from information in the ADDRESS table in the database.)

10

Once a number has been selected from the menu, a cookie will be written with the information and the frame will reload to show the user's personal information with their new dialback number.

15 During initial logon, if the system detects that a cookie has been previously set, it will automatically assign the number from this cookie to the user for the duration of the session. Since the cookie setting activity need only be done rarely, it is possible for a system administrator to tour all machines which will be used and set
20 cookies for each one. In this way, the end user need never have to go through the above process.

From time to time, it may be necessary to move a computer from one location to another or to use a different telephone line with a particular computer. In these circumstances, either a user or a system administrator can assess a "Phone
25 number setting form" which contains Javascript code broadly similar to that mentioned above. The page has to be manually invoked and occupies a main window of the browser in this case. When the page is loaded, the appropriate cookie is read, if there is one, and the current number and location shown. The page also contains a popup menu of legal numbers which is assembled from the
30 database. If a new number is selected, then the existing cookie is overwritten with the updated information. The page is also reloaded which causes the changed number and location to be shown.

Joining an Existing Conference

The user joins an existing conference by clicking on a text link denoting the conference's number. They are then shown the list of participants in the conference. There is also a button which gives the person the option to join the
5 conference.

The server 120 queues the request in the CONFERENCE-REQUEST table. The MAP object 205 subsequently retrieves the request from the table, creates a new entry in the PERSON-CONFERENCE table, and instructs the Millenium CT 135 to
10 set-up the conference using the following commands:

- Call Dial
- Mix all Participants (mix the calls into an audio conference)

15 The system starts by making an outgoing call to the person. When the person answers the call they are played a message telling them that they are being added to the conference. Whilst the conference is being established the person is shown a screen of cycling dots asking him or her to wait. Meanwhile a small secondary frame is reloaded using the client pull HTML construct (for instance every 5
20 seconds). When the server receives a reload request it inspects the status in the CONFERENCE-REQUEST table. If the status indicates that the conference has been established, it returns HTML to the updated frame which causes the entire layout of the window to be reloaded. This mechanism thus updates the meeting-place windows of both people to show they are connected in a conference and deletes
25 the request from the CONFERENCE-REQUEST table. After a short delay they are added in to the conference and the request is deleted from the CONFERENCE-REQUEST table.

The arrival of a new user in a conference is indicated with a short auditory tone,
30 by the temporary display of an indicator such as a coloured dot adjacent to the names of any users who are newly arrived on-line (see the "Butler process" described below) and by the updating of the meeting-place screen 300 of all users. Those users who are in the room in question are shown an image of the person who has just joined. If they move their mouse cursor over this image they are

shown the person's name in the status bar of the window. Users may continue to join a conference until it has grown to the maximum size allowable by the system (30 people in the embodiment being described here). Whenever more than 8 people are present in the same conference the display is changed to show only
5 the names of users rather than their pictures.

Inviting Into a Conference

Users who are already in an established conference can choose to invite other people who are logged on to the system to take part as well. They do this by
10 clicking on the name of such a person in the logged-on frame (only the names of people who are not logged in to a conference are shown as active HTML links at this time). When the person's name is clicked, the user is shown a confirm message in the control frame. If they confirm that they do indeed want to invite the person into the conference, the system will initiate a call to that person and try
15 to add them to the conference.

20

The commands to the Millenium CT under these circumstances are the same as when someone joins a conference (see "Joining an Existing Conference", above).
* Once the person has answered the phone, they are added into the conference and
20 * the meeting-place window 300 is updated for all users.

Removing a Person from a Conference

From time to time, those already in a conference may invite another user to take part but in practice only reach the person's answering machine or voicemail. To
25 recover from this situation, any of the other conference participants may clear the call to the answering machine. They do this by clicking on the name of the "person" that they want to remove in the logged-on frame. This results in a confirmation dialogue which is shown in the control frame. If the action is confirmed, the system will clear the call to the answering machine and remove this
30 "person" from the conference. The conference can then continue as normal with the remaining participants.

The commands to the Millenium CT 135 under these circumstances are as when somebody leaves a conference. See "Ending a Conference" below.

Ending a Conference

Once created, a conference continues to exist as long as more than one person is in it. Users can leave a conference at any time by pressing a button on the meeting-place window 300. When this happens, the Millenium CT 135 is sent the "Call Clear" command to clear the call, causing the Millenium CT 135 to drop the phone connection. The meeting-place 300 is then updated to show the person is no longer in the conference. This is achieved by updating the entry in the CONFERENCE table to indicate that the displays should be updated (see "The Update Process" below).

The Millenium CT 135 disconnects the last person if there is only a single person left in a conference after someone has left. Once the last person has been disconnected, the MAP object sends a 'conference registration' command to the Millenium CT 135 to de-register the conference. The meeting-place windows 300 of all users are then updated.

An alternative way of leaving a conference is for users to simply put down their phone. In the current implementation of ISDN in the UK the Millenium CT 135 is not notified by the telephone exchange that a person has cleared their call for 2 minutes. (After this time, the exchange sends a "call clear" DASS2 message to the Millenium CT.) A forthcoming modification of the BT network to European ISDN (ETSI) standards should mean that the Millenium CT 135 will be able to detect cleared-down calls in the near future.

25

The Update Process

It is necessary to update the meeting-place window 300 of each user who is connected to the system whenever a significant event occurs. These events are when:

- A user comes on-line or is logged out (see next section)
- A user enters or leaves an audio conference
- A user shares a URL (see below)
- A user starts or stops recording an audio-conference (see below)
- When the conference privacy status is changed

When such an event happens, the time and date at which the event occurred is stored in the GROUPS table (for individual users) and in the CONFERENCE table (for conference related events).

5

The update process works by using the client-pull HTML construct from within a small frame 310 in the meeting-place window 300. Every few seconds (for instance every 15 seconds) this window is set to "update". When the server 120 receives an update request from the update frame 310, it compares the time and
10 date when it last received an update request from the update frame 310 for this user (this is called the heartbeat - see "The Heartbeat Process" below) with the time the meeting-place was last changed. If the information to be displayed has changed since the last time, then a new version of the update page is sent back. This includes a javascript function which stipulates that the other frames of the
15 meeting-place 300 should be re-loaded as soon as the update frame is itself loaded (the OnLoad event handler is used). The advantage of this approach is that it does not increase traffic to the server, load on the client or visual distraction for the user by reloading all the frames even when nothing applicable to that user has
changed. On the other hand it does not require multiple channels to be kept open
20 for each user in the way that a protocol such as "server push" does.

The Heartbeat Process

The update requests are also used as a 'heartbeat' function which lets the server know the user is still logged in ('alive'). The Heartbeat object 215 running on the
25 server 120 polls the database 125 frequently (for instance every 30 seconds). If the heartbeat value in the PERSON table for this user is over a minute old, the heartbeat object 215 deems that the user has closed down their meeting-place
300 and left the system. In such cases the user is logged out by updating their on-line status field in the PERSON table. That user will no longer appear as being on-
30 line in other users' meeting-places.

Sharing URLs

When users are connected together in a conference they can share a URL with each other. This may be useful for instance when a person wants others who they

are talking with to see the same WWW page that they are looking at. To share a URL, a user either manually types or copies and pastes the relevant text string into an HTML generated text box in the control frame 320 of the meeting-place 300. The user then either presses the 'return' button or activates a 'share URL' button.

5 This causes the URL to be associated with the user for the rest of the conference by storing this URL in the PERSON table for that user. This is indicated visually in the meeting place window 300 of all conference participants by showing a small graphic with the word 'link' beneath the image 330 of the sending user. This is achieved by updating the entry in the CONFERENCE table to indicate that the

10 displays should be updated (see "The Update Process" above).

If the person stipulates a subsequent link during the course of the same conference then the colour of the link graphic changes to show that a new link has been specified for that person (this is also stored in the PERSON table). Others in the

15 conference can click on a link graphic to see the associated URL opened up in a new window on their browser.

Recording a Conference

Users can choose to record a conference so that they can play it back later. The

20 originator's meeting-place 300 has an additional icon 325 which allows them to record the conference. If the originator drops out of the conference leaving others talking, then the recording control is passed to whoever has been in the conference for the next longest time. The idea behind this approach is to avoid conflicting requests for starting and stopping a recording that may be generated if

25 all users have their own recording control.

Alternatively, all participants in a conference may have an icon which allows them to record the conference. When a person presses the 'start recording' button they are prompted to give the recording a name and to type this into a text entry box.

30 (The recording has a separate, system-generated unique name so it is not mandatory that the user give the recording a unique name). The user then submits the recording name using a button. A recording request is then queued with the server 120 requesting that the conference be recorded. The MAP object 205 then sends the Millenium CT 135 the 'Record Start' message to start recording the

conference. Whilst the recording is being initiated, the person is shown the screen of cycling dots asking them to wait. When the recording is started the request is deleted from the CONFERENCE-REQUEST table. The start of the recording is indicated to the users with an auditory tone and a short voice announcement made to the conference. The recording button in the person's meeting-place is replaced by an animated 'recording-in-progress' icon (this is pressed again to stop recording). Other users are shown an animated icon indicating that a recording is taking place. This is achieved by updating the entry in the CONFERENCE table to indicate that the displays should be updated (see "The Update Process" above).

10

When the originator presses the 'stop recording' button, a confirmation dialogue appears in the control frame. If the user confirms that they want to stop the recording, the system sends 'Record Stop' and 'Record Save' messages to the Millenium CT 135. Once the recording has been saved, the animated recording icon from the person's meeting-place 300 is removed and replaced by the 'start recording' button - other users are also now shown an icon indicating that no recording is taking place. This is achieved by updating the entry in the CONFERENCE table to indicate that the displays should be updated (see "The Update process" above). Subsequent parts of the conference may then be recorded if desired.

20

The MAP object 205 will automatically stop a recording if the last person leaves a conference and the originator has not specifically stopped the recording. The recording will still be saved in these circumstances.

25

The Millenium CT 135 stores the recording as a 64Kbps PCM encoded speech file. The file is subsequently converted and placed on the WWW server where it can be listened to by the conference participants. Preferably, the file is converted into "RealAudio 3" format thus allowing long sound files to be heard without having to first download the entire file to their computers.

30

Pages on the system can contain links to the RealAudio files representing previously recorded conversations. Since the system uses a database to build HTML pages dynamically it is possible to provide the link to the sound file in the

context of a page which indicates when the recording was made, who originated it, and who took part in it. This is achieved by storing this information in the database when the 'Record Save' message is sent to the Millenium CT 135.

5 Making a Personal Recording

In addition to recording a conference it is possible to use the system to record one's own voice only. This feature can be used for recording voice messages or for pronunciation exercises for example. Whenever a user is on line but not in a conference they have the option of clicking on the 'Record' button in the control
10 frame of the meeting place. When they do this they are prompted to give the recording a name and to type this into a text entry box (the recording has a separate system generated unique name so it is not mandatory that the recording be named with a unique identifier). Once a name has been submitted to the system an outgoing call is initiated to the user's number and they are placed in a private
15 (or "single user") conference in which they are the only participant. As soon as the conference is set up the recording request is automatically queued to the server. The MAP object 205 then sends the Millenium CT 135 the 'Record Start' message to start recording the conference. The user hears a voice announcement shortly after the start of the conference to tell them that recording has begun. They also
20 are shown an animated 'recording' icon.

When the user has finished making their recording they can press the animated recording button to stop the recording. This is achieved by sending the Millenium CT 135 a 'Record Stop' message. At this stage they are shown a series of icons
25 which allow them to play back, delete, or save the recording. If they choose the 'Play' icon, the 'Playback start' message is sent to the Millenium CT 135 and the recording is played back to the user over the phone connection from the beginning. Whilst this is happening the play icon is replaced with an animated version. When the recording reaches the end of playback it loops to start from the beginning
30 again. If the user selects the animated play icon then playback stops. If the user selects 'Save' then the recording is saved to RealAudio format. This is achieved by sending the Millenium CT 135 a 'Save Recording' message. If the user does not want to keep the recording they can select the 'Delete' icon which removes the recording by sending the Millenium CT 135 a 'Delete Recording' message. If the

user leaves the private conference without saving their personal recording then the recording will be saved automatically. After choosing to save or delete a recording the user may go on to make more recordings whilst in the same conference

5 Privacy

The system aims to give users the greatest opportunity to see who else is logged on to the system whilst also protecting people from unwanted intrusion. This is done by allowing individuals to set their status as 'Do not disturb' and by letting the originator of a conference set up its status as 'Private'. The 'Do not disturb' status flag is stored in the PERSON table for that user whilst the 'Conference Privacy' status is stored in the CONFERENCE table for that conference. If a person registers themselves as 'Do not disturb' then others will have this status explained to them when they click on the person's picture; they will not be able to enter into a conference with that person. If a conference is private then other people will not be able to enter it and will be shown that it is private when they ask for detail on it. Both "Do Not Disturb" and "Privacy" functions are implemented with toggling on/off controls.

Another privacy advantage of embodiments of the present invention is that subscribers to a service need not disclose their phone number to other users - it is held on the database and used to dial calls to the user but need not be visible directly to others.

The Sleep Function

One of the problems to be overcome with any virtual meeting-place is that users may log on to the system and then leave their computers to go somewhere else. This can lead to other users attempting to set-up conferences with them only to be confronted with answering machines. This problem can be solved as described above under "Removing a Person from a Conference" and this is the preferred method. However, another solution to this problem is to have the database keep a record of all page load requests which are made by a particular user (other than regular meeting-place update requests). Each time such a request is received, a counter is reset. If the counter reaches a designated value (equivalent to, say, 10 minutes of elapsed 'silence' from the user) the user will be designated as

asleep. At this stage if another user clicks on the "sleeping" person's name they will be shown a page in the main frame 315 of the meeting-place 300 telling them that the other person is asleep. In such cases, users can manually dial the person or send them email.

5

The meeting-place window 300 of the person who is designated as asleep changes to tell them this. There is a button labelled 'wake me up' which can be pressed to restore the user's status to 'awake' on the system's database. They will then be able to initiate or be invited into audio conferences once more.

10

Handling Errors and Unusual States

Error messages are displayed to the user whenever an audio-conference request (eg set-up a conference or leave a conference) fails. The request usually fails because the Millenium CT 135 has been unable to make a call since the called party is either engaged, not answering, or their phone number is unavailable (NU). In such cases the Millenium CT 135 indicates the reason for failure in the 'Make Call' response message returned to the MAP object 205. The MAP object 205 then sets the status in the request in the CONFERENCE-REQUEST table to indicate the reason for failure. When the server 120 receives a reload request from the client (that is currently displaying the cycling dots) it inspects the status in the request in CONFERENCE-REQUEST table. Upon finding that the request has failed it returns an error page to the client 100 describing the reason for failure. The user may then read this and return to the meeting-place 300 by clicking on a "continue" button. If they do not do this the error page will automatically re-load the meeting-place 300 after 30 seconds.

Billing

The system provides the opportunity to bill users on the basis of subscription, usage or some combination of the two. Since all audio conferencing telephone calls are outgoing from the Millenium CT platform 135 it is possible to offer users a special tariff for exclusive use with calls to other registered users. The charging structure for other general calls which the user makes need not be adjusted.

The database stores a range of allowable dialback numbers for each user. This allows people to use the service from more than one location. Since users are prevented from entering their own dialback numbers, which have to be authorised by a system administrator, the potential for using the system fraudulently to obtain discounted telephone calls to any destination is removed. The database records the name of each user who initiates an audioconference, the duration of that audioconference and the names of people who took part in the conference, together with the lengths of time for which they took part. This data can be used to charge the conference originator, the participants or both parties on a time basis. All such information is stored in the database 125 together with time and date stamps. The database 125 records the time and date at which each user account is created or closed down. This information can be used to facilitate subscription based charging.

15 The "Butler" Process

The butler process is intended to alert users whenever a new person comes on line, for example by playing a sound such as a bell. At this point an indicator 335 (e.g. a coloured dot) is also shown adjacent to the names of any newly arrived users in the frame 305 and the meeting place window is brought to the front of the browser. The indicator persists until a new update takes place or for 2 minutes - whichever period is shorter.

For each person's name that appears in the "logged on" frame of the meeting place a 'registerUser' javascript function is called in the layout code for the meeting place window. This function adds the name of the user in question to an array. It also checks a 'previous' array which contains a list of all those users who were present the last time the logged on frame was loaded. If a particular name features in the current list but not in the previous list then it is judged to be new and the function returns a 'true' value to the code in the logged on frame. This is what causes the indicator to be displayed by the name of each new user. Once all names in the logged_on frame have been 'registered' in this way the 'onLoad' event for this frame is used to call a second 'butler' function in the layout document. This simply checks whether any items in the 'current' array are different from those in the 'previous' array. If this is the case then a java applet is

called in the 'tools' frame of the meeting place using the LiveConnect protocol embedded in the browser. The function of this applet is simply to play a sound of a designated name. In the case of the butler a 'bell' sound is played by the client. The registerUser function increments a variable value each time it is called (starting
 5 from 0) . Whenever this value is less than 1 the function still adds the names to the current array but it does not display an indicator near a 'new' name, and the butler function does not generate its bell sound. This is to avoid names appearing as 'new' when the system is starting up.

10 Functional Breakdown of the Objects Supporting the System

Referring to "Server Software Architecture" above, four objects installed at the server 120 support the functionality described above. These are the application, Millenium CT, database and Heartbeat objects 200, 205, 210, 215. Functionally, these deal as described with different aspects of the system.

15

Referring to Figure 4, the application object 200 deals with the conference controls offered to the user by means of the meeting place screen 300:

- 1 Login person
- 2 Display meeting place
- 20 2.1 Do updater
- 2.2 Display logged-on
- 2.3 Display room
- 2.3.1 Change telephone number
- 2.3.2 Start conference
- 25 2.3.3 Invite conference
- 2.3.4 Join conference
- 2.3.5 Leave conference
- 2.3.6 Display error message
- 3 Display tools
- 30 3.1 Share URL
- 3.2 Start recording
- 3.3 Stop recording

Referring to Figure 5, the database object 210 deals with the interface to the database 125:

- 1 Create conference
- 2 Get conference-request
- 5 3 Create person-conference
- 4 Update conference
- 5 Create conference-request
- 6 Get person-conference
- 7 Update person-conference
- 10 8 Get conference
- 9 Update conference request
- 10 Delete person conference
- 11 Delete conference-request
- 12 Delete conference
- 15 13 Update person

Referring to Figure 6, the MAP object 205 deals with the interface to the Millenium CT 135:

- 1 Poll conference request table
- 20 2 Start conference
- 3 Invite conference
- 4 Poll Millenium CT
- 4.1 Record stop response received
- 4.2 Conference registration response received
- 25 4.3 Record start response received
- 4.4 Call clear command received
- 4.5 Mix all participants response received
- 4.6 Call clear response received
- 4.7 Call dial response received
- 30 4.8 Register platform response received
- 5 Leave conference
- 6 Join conference
- 7 Start recording
- 8 Stop recording

Referring to Figure 7, the Heartbeat object 215 simply deals with monitoring an area of the database 125 and logging out users appropriately:

- 1 Poll database
- 5 2 Log-out person

Potential Uses for the System

- The system allows groups of people to talk to each other and achieve a shared view of information without requiring that participants either learn a complex set of
- 10 controls or invest in specialist equipment or software at the client end. The equipment at the server end is not particularly complex either and can be readily set up as part of a small business operation, company department or educational service for example. The system is about providing a basis for conversation and information exchange - the exact nature of the conversation and information traded
- 15 is not constrained so there might be a wide variety of potential uses such as:
- application as an educational tool - in particular for teaching languages
 - use by sales teams to encourage greater communication with clients
 - use by fan clubs to talk about ongoing or recent events such as football matches
- 20 • application as part of a customer support system for a product or service
- use as a social meeting device
 - use within business as a method for helping geographically separated groups to collaborate
- 25 The system can be deployed either as a small scale set-up involving PC based audio conferencing platforms such as the Millenium CT 135, or to provide larger scale services on platforms such as the "Integrated Speech Applications Platform" (iSAP) which has been developed by British Telecommunications plc. In the case of deployment on a PC based audio conferencing platform such as the Millenium CT
- 30 135, the overall group size on any platform is generally limited to 60 users by the system's capacity although a service can be spread over a bank of such platforms running in parallel. On the iSAP, the maximum size of an individual group would again tend to be 60 (limited by the capacity of each shelf on the iSAP) although a very much larger overall user group could be catered for.

Whatever the scale of the platform on which the audio conferencing takes place it is possible to implement the service in such a way that more than one separate group can share the same platform (or bank of platforms). Users in group A will
5 see only the names of other users in group A - but not the names of users in group B or C for example. This separation can be achieved entirely through partitioning into groups in the database 125.

Whilst the system described above is based on standard PCs and telephones, using
10 separate connections for voice and data, it is envisaged that the approach could be adapted to work equally effectively on systems where both datatypes were sent down the same line. This would include Internet based systems. The approach is also adaptable to systems which use mobile telephony, which involve phones with built in web browsers or which use a headset connected via the soundcard of a PC
15 rather than a separate telephone.

**CLAIMS**

1. An information retrieval system comprising at least two user interfaces, each user interface having means for accessing an information database system by means of retrieval commands, and wherein at least one, controlling, user interface has means for transmitting, to one or more of the other user interfaces, a signal containing a retrieval command, the other user interfaces being operable under control of the retrieval command signal to transmit the retrieval command to the information database to retrieve a specified item of data.
- 10 2. An information retrieval system according to claim 1, wherein the user interfaces are linked to each other by a server application through which the retrieval command signals are transmitted.
- 15 3. An information retrieval system according to claim 2 comprising a management and control unit for a network based teleconferencing system, comprising an interface for establishing connections across a network between users, through which the signals carrying the retrieval commands may be transmitted.
- 20 4. An information retrieval system according to claim 1 2 or 3, comprising means for preventing more than one controlling user interface simultaneously transmitting a retrieval command signal to the same group of user interfaces.
- 25 5. An information retrieval system according to claim 4, comprising means for permitting different user interfaces to operate as the controlling user interface at different times.
- 30 6. An information retrieval system according to claim 4, wherein only one of the user interfaces has the capability to transmit retrieval command signals to the other users.
7. An information retrieval system according to claim 6, wherein at least one other user interface has means for transmitting signals comprising retrieval



proposals to the controlling user interface, and the controlling user interface has means for converting said retrieval proposals into retrieval command signals for transmission to the other users.

- 5 8. An information retrieval system according to any preceding claim, having means for allowing one user to transmit a timed sequence of retrieval command signals to the other users.
9. An information retrieval system according to any preceding claim, having
10 storage means for storing retrieval commands, and means for extraction of retrieval commands from said storage means for transmission as retrieval command signals.
10. A method of controlling an information retrieval system comprising at least
15 two user interfaces, each user interface having means for accessing an information database system by means of retrieval commands, wherein one, controlling, user interface transmits, to one or more of the other user interfaces, a signal containing a retrieval command, the other user interfaces operating under control of the retrieval command signal to transmit the retrieval command to the information
20 database to retrieve a specified item of data.
11. A method according to claim 10, wherein the user interfaces are linked to each other by a server application through which the retrieval command signals are transmitted.
- 25 12. A method according to claim 10 or 11, arranged to prevent more than one controlling user interface simultaneously transmitting a retrieval command signal to the same group of user interfaces.
- 30 13. A method according to claim 12, wherein different user interfaces of the same user group operate as the controlling user interface at different times.
14. A method according to claim 12, wherein only one user interface has the capability to transmit retrieval command signals to the other users.



15. A method according to claim 14, wherein one of the other user interfaces transmits signals comprising retrieval proposals to the controlling user interface, and the controlling user interface converts said retrieval proposals into retrieval
5 command signals for transmission to the other users.

16. A method according to any of claims 10 to 15, wherein the controlling user interface transmits a timed sequence of retrieval command signals to the other
10 users.

17. A method according to any of claims 10 to 16, wherein retrieval commands are extracted from a storage means for transmission as retrieval command signals.

18. An information retrieval system substantially as described with reference to
15 the drawings.

19. A method of controlling an information retrieval system substantially as described with reference to the drawings.