

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization  
International Bureau



(43) International Publication Date  
26 June 2003 (26.06.2003)

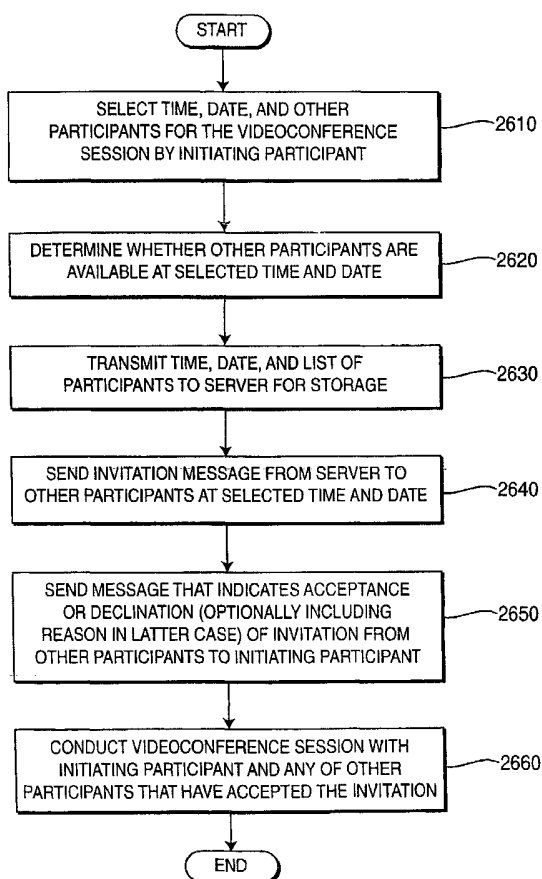
PCT

(10) International Publication Number  
**WO 03/052570 A1**

- (51) International Patent Classification<sup>7</sup>: **G06F 3/00**
- (21) International Application Number: PCT/US02/36773
- (22) International Filing Date:  
15 November 2002 (15.11.2002)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:  
60/341,819 15 December 2001 (15.12.2001) US  
60/341,720 15 December 2001 (15.12.2001) US  
60/341,671 15 December 2001 (15.12.2001) US  
60/341,797 15 December 2001 (15.12.2001) US  
60/341,800 15 December 2001 (15.12.2001) US  
60/341,799 15 December 2001 (15.12.2001) US  
60/341,801 15 December 2001 (15.12.2001) US
- (71) Applicant (for all designated States except US): **THOMSON LICENSING S.A.** [FR/FR]; 46 Quai A. Le Gallo, F-92648 Boulogne (FR).
- (72) Inventors; and  
(75) Inventors/Applicants (for US only): **RICHARDSON, John, William** [US/US]; 30 Matthew Drive, Hamilton, NJ 08690 (US). **CAHNBLEY, Jens** [DE/US]; 239 South Lane, Princeton Junction, NJ 08550 (US). **RA-MASWAMY, Kumar** [IN/US]; 7701 Tamarron Drive, Plainsboro, NJ 08536 (US).
- (74) Agents: **TRIPOLI, Joseph, S.** et al.; Thomson Multimedia Licensing, Inc., 2 Independence Way, Suite 2, Princeton, NJ 08543 (US).
- (81) Designated States (national): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC,

[Continued on next page]

(54) Title: SERVER INVOKED TIME SCHEDULED VIDEOCONFERENCE



(57) Abstract: There is provided a method for automatically initiating a videoconference session over a network. The method includes the steps of receiving a pre-selection that specifies a time and clients for the videoconference session (2610), and sending an invitation message regarding the videoconference session to at least one of the clients at the specified time (2640).

WO 03/052570 A1



LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SC, SD, SE, SG, SI, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

**Published:**

— with international search report

**(84) Designated States (regional):** ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, SK,

*For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.*

## **SERVER INVOKED TIME SCHEDULED VIDEOCONFERENCE**

### **BACKGROUND OF THE INVENTION**

5

#### **CROSS-REFERENCE TO RELATED APPLICATIONS**

This is a non-provisional application claiming the benefit under 35 U.S.C. § 119 of provisional application Serial No. 60/341,819, entitled "SERVER INVOKED TIME SCHEDULED VIDEO CONFERENCE", filed on 15 DECEMBER 2001, Attorney Docket  
10 No.: PU010309, which is commonly assigned and incorporated by reference herein. This application is also related to commonly assigned to provisional application Serial No. 60/366,331, entitled "VIDEOCONFERENCE SYSTEM ARCHITECTURE ", filed on 20 MARCH 2002, which is incorporated by reference herein. This application is also related to commonly assigned provisional application Serial No. 60/341,720, entitled "VIDEO  
15 CONFERENCING BANDWIDTH SELECTION MECHANISM", filed on 15 DECEMBER 2001, which is incorporated by reference herein, and commonly assigned provisional application Serial No. 60/341,671, entitled "QUALITY OF SERVICE SETUP ON A TIME RESERVATION BASIS", filed on 15 DECEMBER 2001, which is incorporated by reference herein, and commonly assigned provisional application Serial No. 60/341,797, entitled  
20 "VIDEO CONFERENCING CALL SET UP METHOD ", filed on 15 DECEMBER 2001, which is incorporated by reference herein, and commonly assigned provisional application Serial No. 60/341,800, entitled " VIDEOCONFERENCE SESSION SWITCHING FROM UNICAST TO MULTICAST", filed on 15 DECEMBER 2001, which is incorporated by reference herein, and commonly assigned provisional application Serial No. 60/341,799,  
25 entitled " METHOD AND SYSTEM FOR PROVIDING A PRIVATE CONVERSATION CHANNEL IN A VIDEOCONFERENCING SYSTEM ", filed on 15 DECEMBER 2001,

which is incorporated by reference herein, and commonly assigned provisional application Serial No. 60/341,801, entitled "VIDEOCONFERENCE APPLICATION USER INTERFACE", filed on 15 DECEMBER 2001, which is incorporated by reference herein.

## 5 **FIELD OF THE INVENTION**

The present invention generally relates to videoconferencing and, more particularly, to methods for automatically initiating a time scheduled videoconference session by a server in a network.

## 10 **BACKGROUND OF THE INVENTION**

In a videoconference session between different videoconferencing systems, common audio and video formats are determined at session set up. Accordingly, a time delay is imposed at the session set up so that the preceding determinations can be made.

15 Accordingly, it would be desirable and highly advantageous to have a method for automatically initiating a time scheduled videoconference session by a server in a network that overcomes the deficiencies of the prior art.

## **SUMMARY OF THE INVENTION**

20 The problems stated above, as well as other related problems of the prior art, are solved by the present invention, which is directed to methods for automatically initiating a time scheduled videoconference session by a server in a network.

Advantageously, the present invention allows for the automatic initiation by a server of a time scheduled videoconference session that can be unicast or multicast.

Moreover, the present invention allows for the automatic initiation by a server of a time scheduled distribution of content (audio and/or video) across a network using multicast.

According to an aspect of the present invention, there is provided a method for  
5 automatically initiating a videoconference session over a network. The method includes the steps of receiving a pre-selection that specifies a time and clients for the videoconference session, and sending an invitation message regarding the videoconference session to at least one of the clients at the specified time.

According to another aspect of the present invention, there is provided a  
10 method for automatically initiating a multicast session over a network. The method includes the steps of assigning a common multicast Internet Protocol (IP) address for the multicast session, and sending an invitation message regarding the multicast session to at least one client at a time that is pre-selected. The invitation message specifies the common multicast IP address.

15 According to yet another aspect of the present invention, there is provided a system for automatically initiating a videoconference session over a network. The system includes means for receiving a pre-selection that specifies a time and clients for the videoconference session, and means for sending an invitation message regarding the videoconference session to at least one of the clients at the specified  
20 time.

According to still another aspect of the present invention, there is provided a system for automatically initiating a multicast session over a network. The system includes means for assigning a common multicast Internet Protocol (IP) address for the multicast session, and means for sending an invitation message regarding the  
25 multicast session to at least one client at a time that is pre-selected. The invitation

message specifies the common multicast IP address.

According to a further aspect of the present invention, there is provided a method for joining a videoconference session over a network. The method includes the steps of providing an ability to receive an invitation message regarding the videoconference session at a time that is pre-selected for the videoconference session to take place, providing an ability to send an accept message in response to the invitation message, and providing an ability to automatically receive content corresponding to the videoconference session, in response to sending the accept message.

According to a still further aspect of the present invention, there is provided a method for joining a multicast session over a network. The method includes the steps of providing an ability to receive an invitation message regarding the multicast session at a time that is pre-selected, providing an ability to send an accept message in response to the invitation message, and providing an ability to automatically receive content corresponding to the multicast session in response to sending the accept message. The invitation message specifies a common multicast IP address for the multicast session.

These and other aspects, features and advantages of the present invention will become apparent from the following detailed description of preferred embodiments, which is to be read in connection with the accompanying drawings.

## BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1A is a block diagram illustrating a computer system 100 to which the present invention may be applied, according to an illustrative embodiment of the present invention;

FIG. 1B is a block diagram illustrating a unicast videoconference session, according to an illustrative embodiment of the present invention;

FIG. 1C is a block diagram illustrating a multicast videoconference session, according to an illustrative embodiment of the present invention;

5        FIG. 2 is a block diagram illustrating a network 200 to which the present invention may be applied, according to an illustrative embodiment of the present invention;

FIG. 3 is a block diagram illustrating the videoconference server 205 of FIG. 2, according to an illustrative embodiment of the present invention;

10        FIG. 4 is a diagram illustrating a member database entry 400 for the member database 314 included in the database entity of FIG. 3, according to an illustrative embodiment of the present invention;

FIG. 5 is a block diagram illustrating an active session entry 500 for the active session database 312 included in the database entity 302 of FIG. 3, according to an illustrative embodiment of the present invention;

FIG. 6 is a block diagram illustrating a Simple Network Management Protocol (SNMP) client-server architecture 600, according to an illustrative embodiment of the present invention;

FIG. 7 is a diagram illustrating a method for registering for a videoconference session using Session Initiation Protocol (SIP), according to an illustrative embodiment of the present invention;

FIG. 8A is a diagram illustrating a method for setting up a unicast videoconference session using Session Initiation Protocol (SIP), according to an illustrative embodiment of the present invention;

25        FIG. 8B is a diagram illustrating the steps taken by the videoconference server

205 of FIG. 2 when an INVITE request is received from the client #1 802 (step 810 of FIG. 8A), according to an illustrative embodiment of the present invention;

FIG. 9 is a diagram further illustrating the method of FIG. 8A, according to an illustrative embodiment of the present invention.

5        FIG. 10 is a diagram illustrating a method for setting up a multicast videoconference session using Session Initiation Protocol (SIP), according to another illustrative embodiment of the present invention;

FIG. 11 is a diagram illustrating a method for canceling a videoconference session using Session Initiation Protocol (SIP), according to an illustrative  
10        embodiment of the present invention;

FIG. 12 is a diagram illustrating a method for terminating a videoconference session between two clients using Session Initiation Protocol (SIP), according to an illustrative embodiment of the present invention;

FIG. 13 is a diagram illustrating a method for terminating a videoconference  
15        session between three clients using Session Initiation Protocol (SIP), according to an illustrative embodiment of the present invention;

FIG. 14 is a diagram illustrating a method for terminating a videoconference session between three clients using Session Initiation Protocol (SIP), according to another illustrative embodiment of the present invention;

20        FIG. 15 is a diagram illustrating a signaling method for resolution and frame rate adjustment, according to an illustrative embodiment of the present invention;

FIG. 16 is a diagram illustrating signaling before resolution and frame rate adjustment (clients 2 and 3), according to an illustrative embodiment of the present invention;

25        FIG. 17 is a diagram illustrating signaling after resolution and frame rate



adjustment (clients 2 and 3), according to an illustrative embodiment of the present invention;

FIG. 18A is a block diagram of a videoconference client application 1800, according to an illustrative embodiment of the present invention;

5        FIG. 18B is a block diagram further illustrating the audio mixer 1899 included in the multimedia interface layer 1802 of FIG. 18A, according to an illustrative embodiment of the present invention;

FIG. 18C is a block diagram further illustrating the echo cancellation module 1898 included in the multimedia interface layer 1802 of FIG. 18A, according to an  
10 illustrative embodiment of the present invention;

FIG. 19 is a diagram illustrating a method employed by a decoder 1890 included in either of the audio codecs 1804a and/or the video codecs 1804b, according to an illustrative embodiment of the present invention;

FIG. 20 is a diagram illustrating a user plane protocol stack 2000, according to  
15 an illustrative embodiment of the present invention;

FIG. 21 is a diagram illustrating a control plane protocol stack 2100, according to an illustrative embodiment of the present invention;

FIG. 22 is a block diagram illustrating a screen shot 2200 corresponding to the user interface 1808 of FIG. 18A, according to an illustrative embodiment of the  
20 present invention;

FIG. 23 is a diagram illustrating a login interface 2300, according to an illustrative embodiment of the present invention;

FIG. 24 is a block diagram illustrating a user selection interface 2400 for session initiation, according to an illustrative embodiment of the present invention;

25        FIG. 25 is a block diagram illustrating an invitation interface 2500 for accepting

or rejecting an incoming call, according to an illustrative embodiment of the present invention;

FIG. 26 is a flow diagram illustrating a method for automatically initiating a videoconference session with multiple participants, according to an illustrative  
5 embodiment of the present invention; and

FIG. 27 is a flow diagram illustrating a method for automatically initiating a multicast session between multiple participants, according to an illustrative embodiment of the present invention.

## 10 DETAILED DESCRIPTION OF THE INVENTION

The present invention is directed to methods for automatically initiating a time scheduled videoconference session by a server in a network. As used herein, the phrase "initiating participant" refers to the person initiating a videoconference session, and the phrase "other participants" refers to the other persons with whom the  
15 initiating participant desires to include in the videoconference session.

It is to be understood that the present invention may be implemented in various forms of hardware, software, firmware, special purpose processors, or a combination thereof. Preferably, the present invention is implemented as a combination of hardware and software. Moreover, the software is preferably  
20 implemented as an application program tangibly embodied on a program storage device. The application program may be uploaded to, and executed by, a machine comprising any suitable architecture. Preferably, the machine is implemented on a computer platform having hardware such as one or more central processing units (CPU), a random access memory (RAM), and input/output (I/O) interface(s). The  
25 computer platform also includes an operating system and microinstruction code. The

various processes and functions described herein may either be part of the microinstruction code or part of the application program (or a combination thereof) which is executed via the operating system. In addition, various other peripheral devices may be connected to the computer platform such as an additional data  
5 storage device and a printing device.

It is to be further understood that, because some of the constituent system components and method steps depicted in the accompanying Figures are preferably implemented in software, the actual connections between the system components (or the process steps) may differ depending upon the manner in which the present  
10 invention is programmed. Given the teachings herein, one of ordinary skill in the related art will be able to contemplate these and similar implementations or configurations of the present invention.

FIG. 1A is a block diagram illustrating a computer system 100 to which the present invention may be applied, according to an illustrative embodiment of the  
15 present invention. The computer processing system 100 includes at least one processor (CPU) 102 operatively coupled to other components via a system bus 104. A read only memory (ROM) 106, a random access memory (RAM) 108, a display adapter 110, an I/O adapter 112, a user interface adapter 114, a sound adapter 199, and a network adapter 198, are operatively coupled to the system bus 104.

20 A display device 116 is operatively coupled to system bus 104 by display adapter 110. A disk storage device (e.g., a magnetic or optical disk storage device) 118 is operatively coupled to system bus 104 by I/O adapter 112.

A mouse 120 and keyboard 122 are operatively coupled to system bus 104 by user interface adapter 114. The mouse 120 and keyboard 122 are used to input and  
25 output information to and from system 100.

At least one speaker (herein after "speaker") 197 is operatively coupled to system bus 104 by sound adapter 199.

A (digital and/or analog) modem 196 is operatively coupled to system bus 104 by network adapter 198.

5        A description will now be given of policy based network management (PBNM), according to an illustrative embodiment of the present invention. PBNM is a technology that provides the ability to define and distribute policies to manage networks (an example network to which the present invention may be applied is described below with respect to FIG. 2). These policies allow the coordinated control  
10 of critical network resources such as bandwidth and security. PBNM enables applications, such as IP based videoconferencing, that require differentiated treatment on the network. PBMN provides the basis for allowing different types of applications to co-exist on a single network and provide the required resources to each of these applications.

15        In further detail, PBNM defines policies for applications and users that consume network resources. For example, business critical applications can be given the highest priority and a percentage of the bandwidth on the network, videoconferencing and voice over IP can be given the next highest priority, and finally web traffic and file transfers that do not have strict bandwidth or time critical  
20 constraints can be given the remaining amount of resources on the network. This differentiation of users and applications can be accomplished using PBNM.

The videoconference system ties into a PBNM system by querying a network policy server for the policy that corresponds to the videoconference application. The videoconference server obtains the policy from the network policy server and  
25 determines the resources available in the network for videoconferencing based on

the received parameters. The policy will typically correspond to, for example, the bandwidth available to this application during certain times of the day or only to certain users. The configuration is readily modified by, for example, adding, deleting, replacing, modifying, etc., policies and/or portions thereof. As a result, the videoconference server will use the information provided in the policy to manage conferencing sessions on the network.

FIG. 2 is a block diagram illustrating a network 200 to which the present invention may be applied, according to an illustrative embodiment of the present invention. The network 200 includes: a videoconference server 205; a policy and QoS manager 210; a MADCAP server 215; a first plurality of computer 220a-f; a first local area network 225; a first router 240; a second plurality of computers 230a-e; a second local area network 235; a second router 245; and a wide area network 250.

A description will now be given of a server architecture, according to an illustrative embodiment of the present invention. FIG. 3 is a block diagram illustrating the videoconference server 205 of FIG. 2, according to an illustrative embodiment of the present invention. The videoconference server 205 can be considered to include the following three basic entities: the database entity 302; the network communications entity 304; and the session management entity 306.

The session management entity 306 is responsible for managing videoconference session setup and teardown. The session management entity 306 also provides most of the main control for the videoconference server 205. The session management entity 306 includes a session manager 320 for implementing functions of the session management entity 306.

The network communications entity 304 is responsible for encapsulating the many different protocols used for the videoconference system. The protocols may

include Simple Network Management Protocol (SNMP) for remote administration and management, Common Open Policy Services (COPS) or another protocol such as Lightweight Directory Access Protocol (LDAP) for policy management, Multicast Address Dynamic Client Allocation Protocol (MADCAP) for multicast address allocation, Session Initiation Protocol (SIP) for videoconference session management, and Server to Server messaging for distributed videoconferencing server management. Accordingly, the network communications entity 304 includes: an SNMP module 304a; an LDAP client module 304b; a MADCAP client module 304c; a SIP module 304d; and a server-to-server management module 304e. Moreover, the preceding elements 304a-e respectively communicate with the following elements: a remote administration terminal 382; a network policy server (bandwidth broker) 384; a MADCAP server 215; desktop conferencing clients 388; and other videoconferencing servers 390. Such communications may be implemented also using Transmission Control Protocol (TCP), User Datagram Protocol (UDP), Internet Protocol (IP), collectively represented by protocol module 330. It is to be appreciated that the preceding list of protocols and corresponding elements are merely illustrative and, thus, other protocols and corresponding elements may be readily employed while maintaining the spirit and scope of the present invention.

It is to be further appreciated that the architecture of the videoconference server 205 is also suitable for a user on a portable device to connect into the corporate infrastructure through a Virtual Private Network (VPN) in order to send and receive content from a videoconference session.

The database entity 302 includes the following four databases: a scheduling database 310, an active session database 312, a member database 314, and a network architecture database 316.

The videoconference system server 205 further includes or, at the least,  
5 interfaces with, a company LDAP server (user information) 340 and an optional external database 342. The optional external database 342 includes an LDAP client 304b.

A description will now be given of the member database 314 included in the database entity 302 of FIG. 3, according to an illustrative embodiment of the present  
10 invention. The member database 314 includes information on each user that has logged into the videoconference system. As an example, the following information may be kept in the member database 314 for each user: username; password (if applicable); supported video codecs and capture resolutions; supported audio  
codecs; current IP address; current call number (if currently a member of an active  
15 call); availability (available or unavailable); video camera type and model; location on the network (each location is connected by a limited bandwidth wide area network link); and CPU type and processing power. It is to be appreciated that the preceding items are merely illustrative and, thus, other items in addition to or in place of some or  
all of the preceding items may also be kept in the member database 314 for each  
20 user, while maintaining the spirit and scope of the present invention.

FIG. 4 is a diagram illustrating a member database entry 400 for the member database 314 included in the database entity 302 of FIG. 3, according to an illustrative embodiment of the present invention. In the illustrative embodiment of FIG. 4, the member database 314 is implemented using a simple linked list.  
25 However, it is to be appreciated that in other embodiments of the present invention,

different implementations of the member database 314 may be employed while maintaining the spirit and scope of the present invention. As one example, an LDAP type of database may be used to store the member information.

A description will now be given of the active session database 312 included in the database entity 302 of FIG. 3, according to an illustrative embodiment of the present invention. The active session database 312 includes information on each videoconference session currently taking place. As an example, the following information may be kept for each call in the active session database 312: call ID; description; multicast (yes/no); if multicast, then multicast IP address; for each participant, network location, current transmitting resolution, current transmitting bit rate, video and audio codec; public/private call (can others join?); scheduled time of session; start time of session; and any additional options. It is to be appreciated that the preceding items are merely illustrative and, thus, other items in addition to or in place of some or all of the preceding items may also be kept in the active session database 312, while maintaining the spirit and scope of the present invention.

FIG. 5 is a block diagram illustrating an active session entry 500 in the active session database 312 included in the database entity 302 of FIG. 3, according to an illustrative embodiment of the present invention. In the illustrative embodiment of FIG. 5, the active session database 312 is implemented using a simple linked list. However, it is to be appreciated that in other embodiments of the present invention, different implementations of the active session database 312 may be employed while maintaining the spirit and scope of the present invention.

Referring again to FIG. 3, a description will now be given of the network architecture database 316 included in the database entity 302 of FIG. 3, according to an illustrative embodiment of the present invention. The network architecture



database 316 includes a full mapping of the entire network. The network architecture database 316 includes information on each active network element (i.e., IP Routers, Ethernet switches, etc.) and information on links that connect the routers and switches together. To effectively manage the bandwidth and quality of service in the network, the videoconference server 205 needs to know this information.

Policy information concerning the number of videoconference sessions that are allowed to take place simultaneously, the videoconference session bit rates, and bandwidth limits can also be defined in the network architecture database 316. The network architecture could be represented as a weighted graph within the network architecture database 316. It is to be appreciated that the network architecture database 316 is an optional database in the videoconference server 205. The network architecture database 316 may be used to cache the policies that are requested from the policy server 210.

A description will now be given of the scheduling database 310 included in the database entity 302 of FIG. 3, according to an illustrative embodiment of the present invention. The scheduling database 310 contains a schedule for users to reserve times to use the videoconference system. This is dependent on the policies that, for example, an Information Systems department has in place concerning the number of videoconference sessions that can take place simultaneously on certain links over the wide area network 250.

A description will now be given of the network communications entity 304 of FIG. 3. The network communications entity 304 includes: a Simple Network Management Protocol (SNMP) module 304a; a Lightweight Directory Access Protocol (LDAP) client module 304b; a Multicast Address Dynamic Client Allocation Protocol

(MADCAP) client module 304c; a Session Initiation Protocol (SIP) module 304d; and a server-to-server management module 304e.

A description will now be given of the Simple Network Management Protocol (SNMP) module 304a included in the network communication entity 304 of FIG. 3,

5 according to an illustrative embodiment of the present invention. FIG. 6 is a block diagram illustrating a Simple Network Management Protocol (SNMP) client-server architecture 600, according to an illustrative embodiment of the present invention.

The architecture 600 represents one implementation of the SNMP module 304a;

however, it is to be appreciated that the present invention is not limited to the

10 architecture shown in FIG. 6 and, thus, other SNMP architectures may also be employed while maintaining the spirit and scope of the present invention. SNMP will be used for remote administration and monitoring of the videoconferencing server.

The Simple Network Management Protocol (SNMP) client-server architecture 600 includes an SNMP management station 610 and an SNMP managed entity 620.

15 The SNMP management station 610 includes a management application 610a and an SNMP manager 610b. The SNMP managed entity 620 includes managed resources 620a, SNMP managed objects 620b, and an SNMP agent 620c.

Moreover, each of the SNMP management station 610 and an SNMP managed entity 620 further include a UDP layer 630, an IP layer 640, a Medium Access Control

20 (MAC) layer 650, and a physical layer 660.

The SNMP agent 620c allows monitoring and administration from the SNMP management station 610. The SNMP agent 620c is the client in the SNMP architecture 600. The SNMP agent 620c basically takes the role of responding to requests for information and actions from the SNMP management station 610. The

25 SNMP management station 610 is the server in the SNMP architecture 600. The

SNMP management station 610 is the central entity that manages the agents in a network. The SNMP management station 610 serves the function of allowing an administrator to gather statistics from the SNMP agent 620c and change configuration parameters of the SNMP agent 620c.

5           Using the SNMP model, the resources in the videoconference server 205 can be managed by representing these resources as objects. Each object is a data variable that represents one aspect of the managed agent. This collection of objects is commonly referred to as a Management Information Base (MIB). The MIB functions as a collection of access points at the SNMP agent 620c for the SNMP  
10 management station 610. The SNMP management station 610 is able to perform monitoring by retrieving the value of MIB objects in the SNMP agent 620c. The SNMP management station 610 is also able to cause an action to take place at the SNMP agent 620c or can change the configuration settings at the SNMP agent 620c.

          SNMP operates over the IP layer 640 and uses the UDP layer 630 for its  
15 transport protocol.

          The basic messages used in the SNMP management protocol are as follows: GET; SET; and TRAP. The GET message enables the SNMP management station 610 to retrieve the value of objects at the SNMP agent 620c. The SET message enables the SNMP management station 610 to set the value of objects at the SNMP  
20 agent 620c. The TRAP message enables the SNMP agent 620c to notify the SNMP management station 610 of a significant event.

          A description will now be given of the SNMP managed resources 620a included in the SNMP managed entity 620, according to an illustrative embodiment of the present invention. The remote administration could monitor and/or control the  
25 following resources within the videoconference server 205: active sessions and

associated statistics; session log; network policy for videoconferencing; Session Initiation Protocol (SIP) parameters and statistics; and MADCAP parameters and statistics.

From the SNMP management station 610, the following three types of SNMP  
5 messages are issued on behalf of a management application: GetRequest; GetNextRequest; and SetRequest. The first two are variations of the GET function. All three messages are acknowledged by the SNMP agent 620c in the form of a GetResponse message, which is passed up to the management application 610a. The SNMP agent 620c may also issue a trap message in response to an event that  
10 has occurred in a managed resource.

Referring again to FIG. 3, a description will now be given of the Lightweight Directory Access Protocol (LDAP) client module 304b included in the network communications entity 304 of FIG. 3, according to an illustrative embodiment of the present invention. The LDAP module 304b utilizes LDAP, which is a standard IP  
15 based protocol for accessing common directory information. LDAP defines operations for accessing and modifying directory entries such as: searching for entries meeting user-specific criteria; adding an entry; deleting an entry; modifying an entry; and comparing an entry.

A description will now be given of the Multicast Address Dynamic Client  
20 Allocation Protocol (MADCAP) client module 304c included in the network communications entity of FIG. 3, according to an illustrative embodiment of the present invention. The MADCAP module 304c utilizes MADCAP, which is a protocol that allows hosts to request multicast address allocation services from multicast address allocation servers. When a videoconferencing session is setup to use  
25 multicasting services, the videoconference server 205 needs to obtain a multicast

address to allocate to the clients in the session. The videoconference server 205 can dynamically obtain a multicast address from a multicast address allocation server using the MADCAP protocol.

A description will now be given of the Session Initiation Protocol (SIP) module 304d included in the network communications entity 304 of FIG. 3, according to an illustrative embodiment of the present invention. The SIP module 304d utilizes SIP, which is an application layer control protocol for creating, modifying and terminating multimedia sessions with one or more participants on IP based networks. SIP is a text message based protocol.

In a SIP based videoconference system, each client and server is identified by a SIP URL. The SIP URL takes the form of user@host, which is in the same format as an email address, and in most cases the SIP URL is the user's email address.

A description will now be given of the server-to-server management module 304e included in the network communications entity 304 of FIG. 3, according to an illustrative embodiment of the present invention. The server-to-server management module 304e utilizes messages for exchanging information between videoconference servers. The server-to-server management module 304e is preferably utilized in a typical deployment wherein a unique videoconference server (e.g., videoconference server 205) is set up locally to the network (e.g., LAN 225) that it is supporting, therefore several videoconference servers may exist in a company wide network (e.g., network 200). Some of the primary purposes of the messages for exchanging information include synchronizing databases and checking the availability of network resources.

The following messages are defined: QUERY – query an entry in a remote server; ADD – add an entry to a remote server; DELETE – delete an entry from a remote server; and UPDATE – update an entry on a remote server.

The server-to-server messaging can use a TCP based connection between  
5 each server. When the status of one server changes, the remaining servers are updated with the same information.

A description will now be given of operational scenarios of the  
videoconference server 205, according to an illustrative embodiment of the present  
invention. Initially, a description of operational scenarios corresponding to the setting  
10 up of a videoconference session is provided, followed by a description of operation  
scenarios corresponding to resolution and frame rate adjustment during the  
videoconference session. Session operational scenarios include SIP server  
discovery, member registration, session setup, session cancel, and session  
terminate.

15 A description will now be given of a session operational scenario  
corresponding to SIP server discovery, according to an illustrative embodiment of the  
present invention. A user (videoconference client application) can register with a  
preconfigured videoconference server (manually provisioned) or on startup by  
sending a REGISTER request to the well-known “all SIP servers” multicast address  
20 “sip.mcast.net” (224.0.1.75). The second mechanism (REGISTER request) is  
preferable because it would not require each user to manually configure the address  
of the local SIP server in their videoconference client application. In this case, the  
multicast addresses would need to be scoped correctly in the network to ensure that  
the user is registering to the correct SIP server for the videoconference. In addition  
25 to the previous methods, in another method to make the provisioning process

simpler, the SIP specification recommends that administrators name their SIP servers using the sip.domainname convention (for example, sip.princeton.tce.com).

A description will now be given of a session operational scenario corresponding to member registration, according to an illustrative embodiment of the present invention. FIG. 7 is a diagram illustrating a method for registering for a videoconference session using Session Initiation Protocol (SIP), according to an illustrative embodiment of the present invention. The example of FIG. 7 includes a videoconference client application (client) 702 and a videoconference server (server) 205. It is to be appreciated that the phrases "client application" and "client" are used interchangeably herein.

In the member registration function, the client 702 sends a SIP REGISTER request to the server 205 (step 710). The server 205 receives this message and stores the IP address and the SIP URL of the client 702 in the member database 314.

The REGISTER request may contain a message body, although its use is not defined in the standard. The message body can contain additional information relating to configuration options of the client 702 that is registering with the server 205.

The server 205 acknowledges the registration by sending a 200 OK message back to the client 702 (step 720).

Descriptions will now be given of unicast and multicast videoconference sessions, according to illustrative embodiments of the present invention. FIGs. 1B and 1C are block diagrams respectively illustrating a unicast videoconference session and a multicast videoconference session, according to two illustrative embodiments of the present invention. The examples of FIGs. 1B and 1C includes a client 1 130, a

client 2 132, a client 3 134, an Ethernet switch 136, an IP router 138, and an IP router 140, and a WAN 142.

In the unicast example, a unique stream is sent from each client to each other client. Such an approach can consume a large amount of bandwidth as more  
5 participants join the network. In contrast, in the multicast approach, only one stream is sent from each client. Thus, the multicast approach consumes less of the network resources such as bandwidth in comparison to the unicast approach.

A description will now be given of a session operational scenario corresponding to a unicast videoconference session set up, according to an  
10 illustrative embodiment of the present invention. FIG. 8A is a diagram illustrating a method for setting up a unicast videoconference session using Session Initiation Protocol (SIP), according to an illustrative embodiment of the present invention. The example of FIG. 8A includes a videoconference client application #1 (client #1) 802, a videoconference server (server) 205, and a videoconference client application #2  
15 (client #2) 806.

An INVITE request is sent from the client #1 802 to the server 205 (step 810). The INVITE request is forwarded from the server 205 to the client #2 806 (step 815).

A 180 ringing message is sent from the client #2 706 to the server 205 (step 820). The 180 ringing message is forwarded from the server 205 to the client #1 702  
20 (step 825).

A 200 OK message is sent from the client #2 706 to the server 205 (step 830). The 200 OK message is forwarded from the server 205 to the client #1 702 (step 835).



An acknowledge message ACK is sent from the client #1 702 to the client #2 706 (step 840). The videoconference session (media session) takes place between the two nodes (clients #1 802 and #2 806) (step 845).

FIG. 8B is a diagram illustrating the steps taken by the videoconference server 205 when an INVITE request is received from the videoconference client application #1 802 (step 810 of FIG. 8A), according to an illustrative embodiment of the present invention.

The server 205 initially checks to see if the requesting user (client #1 802) is registered with the server 205 and it also checks to see if the user that is being called (client #2 806) is registered with the server 205 (step 850).

The server 205 determines the location of each user on the network (step 855) and determines if there is a low bandwidth WAN link (e.g., WAN 250) connecting their two locations (if different) (step 860).

If there is not a low bandwidth link WAN connecting the two locations together, the server 205 proceeds with the call (step 865). However, if there is a low bandwidth link between the two users, then the method proceeds to step 870.

At step 870, the server 205 checks the policy on videoconference sessions on the WAN 250; this basically translates into "X sessions can take place at a maximum bit rate of Y". The server 205 checks for availability based on this policy (step 875). If there is no availability, then the server 205 rejects the INVITE request by sending any of the following messages, "600 – Busy Everywhere", "486 – Busy Here", "503 – Service Unavailable", or "603 – Decline" (step 880), and the method is terminated (without continuation to step 815 of the method of FIG. 8A). However, if there is availability, then the server 205 proceeds with the call (step 865). It is to be appreciated that step 865 is followed by step 815 of the method of FIG. 8A.

FIG. 9 is a diagram further illustrating the method of FIG. 8A, according to an illustrative embodiment of the present invention. The example of FIG. 9 includes a client application 1 998, a client application 2 997, videoconference server 205, and other videoconference servers 986. Elements of the videoconference server 205 that are also shown in FIG. 9 include member database 314, active session database 312, a policy database 999 that is included in network architecture database 316, session manager 320, SIP module 304d, and server to server management module 304e.

FIG. 9 is provided to depict the internal interaction within the videoconference server 205, and thus is only shown at a basic level to provide an example of the signaling flow between the entities of the videoconference server 205.

An INVITE request is sent from client application 1 998 to SIP module 304d within the videoconference server 205 (step 903). The SIP module 304d decodes the message and forwards the INVITE requires to the session manager 320 (step 906). The session manager 320 checks the active session database 312, the member database 314, and the policy database 999 within the network architecture database 316 to ensure that the session can be correctly set up (steps 909, 912, and 915, respectively). If the session can be correctly set up, then the active session database 312, the member database 314, and the policy database 999 transmit an OK message to the session manager 320 (steps 918, 921, and 924). Once this verification process is completed, the videoconference server 205 will notify other videoconferencing servers of the change in system status (step 927 and 930).

The session manager 320 will forward an INVITE message to the SIP module 304d (step 933) which will then forward the INVITE message to client application 2 997 (step 936). Upon receiving the INVITE message, client application 2 997 will

respond to the SIP module 304d with a 180 Ringing message that indicates that the SIP module 304d has received the INVITE message (step 939). The 180 Ringing message is received by the SIP module 304d, decoded and then forwarded to the session manager 320 (step 942). The status of the client is updated (steps 945, 948, 5 951, 954, 957, and 958) in each of the databases shown in FIG. 9 within the videoconference server 205.

The 180 Ringing message is forwarded from the session manager 320 to client application 1 998 (step 960 and 963). A 200 OK message is then sent from client application 2 997 to the SIP module 304d (step 966) and forwarded from the 10 SIP module 304d to the session manager 320 (step 969). The 200 OK message indicates that client application 2 997 is accepting the invitation for the videoconference session.

The status of the client is updated (steps 972, 975, 978, 981, 984, and 985) in each of the databases shown in FIG. 9 within the videoconference server 205. An 15 OK message is sent from session manager 320 to SIP module 304d and is forwarded from SIP module 304d to client application 1 998 (steps 988 and 991). An ACK message is sent from client application 1 998 to client application 2 987 completing the session set up (step 994).

A description will now be given of a session operational scenario 20 corresponding to a multicast videoconference session set up, according to an illustrative embodiment of the present invention. To provide multicast session set up, the Session Description Protocol (SDP) is used. The SDP protocol is able to convey the multicast address and port numbers.

The multicast session setup is similar to the unicast session setup except that a multicast address is required. The multicast address is allocated by the MADCAP server 215 in the network.

FIG. 10 is a diagram illustrating a method for setting up a multicast  
5 videoconference session using Session Initiation Protocol (SIP), according to another illustrative embodiment of the present invention. The example of FIG. 10 includes a videoconference client application #1 (client #1) 1002, a videoconference server (server) 205, a videoconference client application #2 (client #2) 1006, and a  
MADCAP server 215.

10 An INVITE request is sent from the client #1 1002 to the server 205 (step 1010). A MADCAP request is sent from the server 205 to the MADCAP server 215 (step 1015). An acknowledge message ACK is sent from the MADCAP server 215 to the server 205 (step 1020). The INVITE request is forwarded from the server 205 to the client #2 1006 (step 1025).

15 A 180 ringing message is sent from the client #2 1006 to the server 205 (step 1030). The 180 ringing message is forwarded from the server 205 to the client #1 1002 (step 1035).

A 200 OK message is sent from the client #2 1006 to the server 205 (step 1040). The 200 OK message is forwarded from the server 205 to the client #1 1002  
20 (step 1045).

An acknowledge message ACK is sent from the client #1 1002 to the client #2 1006 (step 1050). The videoconference session (media session) takes place between the two nodes (clients #1 1002 and #2 1006) (step 1055).

A description will now be given of a session operational scenario  
25 corresponding to the cancellation of a videoconference session, according to an

illustrative embodiment of the present invention. The CANCEL message is used to terminate pending session set up attempts. A client can use this message to cancel a pending videoconference session set up attempt the client had earlier initiated. The server forwards the CANCEL message to the same locations with pending requests that the INVITE was sent to. The client should not respond to the CANCEL message with a "200 OK" message. If the CANCEL message is unsuccessful, then the session terminate sequence (i.e., BYE message) can be used.

FIG. 11 is a diagram illustrating a method for canceling a videoconference session using Session Initiation Protocol (SIP), according to an illustrative embodiment of the present invention. The example of FIG. 11 includes a videoconference client application #1 (client #1) 1102, a videoconference server (server) 205, and a videoconference client application #2 (client #2) 1106.

An INVITE request is sent from the client #1 1102 to the server 205 (step 1110). The INVITE request is forwarded from the server 205 to the client #2 1106 (step 1115).

A 180 ringing message is sent from the client #2 1106 to the server 205 (step 1120). The 180 ringing message is forwarded from the server 205 to the client #1 1102 (step 1125).

A CANCEL message is sent from the client #1 1102 to the server 205 (step 1130). The CANCEL message is forwarded from the server 205 to the client #2 1106 (step 1135).

A description will now be given of a session operational scenario corresponding to the termination of a videoconference session, according to an illustrative embodiment of the present invention. FIG. 12 is a diagram illustrating a method for terminating a videoconference session between two clients using Session

Initiation Protocol (SIP), according to an illustrative embodiment of the present invention. The example of FIG. 12 includes a first client (videoconference client application #1) 1202, a videoconference server (server) 205, and a second client (videoconference client application #2) 1206.

5       The client #1 1202 decides to discontinue a call with the client #2 1206. Thus, the client #1 1202 sends a BYE message to the server 205 (step 1210). The server 205 forwards the BYE message to client #2 1206 (step 1220).

          The client #2 1206 sends a 200 OK message back to the server 205 indicating it (client #2 1206) has disconnected (step 1230). The server 205 forwards the 200  
10   OK message to client #1 1202 indicating a successful disconnect (step 1240).

          FIG. 13 is a diagram illustrating a method for terminating a videoconference session between three clients using Session Initiation Protocol (SIP), according to an illustrative embodiment of the present invention. The example of FIG. 13 includes a first client (videoconference client application #1) 1302, a videoconferencing server  
15   (server) 205, a second client (videoconference client application #2) 1306, and a third client (videoconference client application #3) 1308.

          The client #1 1302 decides to discontinue a call with the client #2 1306 and the client #3 1308; this does not tear down the session between the client #2 1306 and the client #3 1308.

20       The client #1 1302 sends a BYE message to the server 205 (step 1310). The server 205 interprets the BYE message and understands that the client #2 1306 and the client #3 1308 are involved in the videoconference session with the client #1 1302 and forwards the BYE message to both client #2 1306 and client #3 1308 (steps 1320 and 1330).

The client #2 1306 sends a 200 OK message back to the server 205 (step 1340). The server 205 forwards the 200 OK message back to client #1 1302 (step 1350). The client #3 1308 sends a 200 OK message back to the server 205 (step 1360). The server 205 forwards the 200 OK message back to client #1 1302 (step 1370).

FIG. 14 is a diagram illustrating a method for terminating a videoconference session between three clients using Session Initiation Protocol (SIP), according to another illustrative embodiment of the present invention. The example of FIG. 14 includes a first client (videoconference client application #1) 1402, a videoconference server (server) 205, a second client (videoconference client application #2) 1406, and a third client (videoconference client application #3) 1406.

The client #1 1402 decides to discontinue the call with the client #2 1406 and the client #3 1406; this does not tear down the session between the client #2 1406 and the client #3 1406.

The client #1 1402 sends a BYE message to the server 205 intended for the client #2 1406 (step 1410). The server 205 forwards the BYE message to the client #2 1406 (1420). The client #1 1402 sends a BYE message to the server 205 intended for client #3 1406 (1430). The server 205 forwards the BYE message to the client #3 1406 (step 1440).

The client #2 1406 sends a 200 OK message back to the server 205 (step 1450). The server 205 forwards the 200 OK message back to the client #1 1402 (step 1460). The client #3 1408 sends a 200 OK message back to the server 205 (step 1470). The server 205 forwards the 200 OK message back to the client #1 1402 (step 1480).

In addition to the previous examples described with respect to FIGs. 12 through 14, a termination can be invoked by transmitting the BYE message to the multicast group address to which belong the videoconference subscribers. Using this method, the server and the other client applications will receive the message. It is a more universal and efficient mechanism for terminating the session due to the lower amount of overhead associated with it.

A description will now be given of operation scenarios corresponding to resolution and frame rate adjustment, according to an illustrative embodiment of the present invention. Videoconferencing involves transmitting live, two-way interactive video between several users at different locations on a computer network. Real-time interactive video requires transmission of large amounts of information with constrained delay. This requires that the computer network that the videoconference system is tied to must be able to provide an adequate amount of bandwidth and quality of service for each user involved in the session. Bandwidth can be a limited resource at times and quality of service cannot always be guaranteed in all networks, therefore some limitations will exist. In a private corporate network, it is possible to guarantee quality of service, but it is not always possible to guarantee large amounts of bandwidth.

The basic corporate computer network infrastructure includes several high speed local area networks (LANs) connected together through low speed links (see, e.g., FIG. 2). Each of the high speed LANs usually represent the network infrastructure at a single geographical location and the low speed links are the long haul links that connect the multiple geographic locations together. The reason low speed links are used is because the cost of the long haul links are relatively high and



also most of the network traffic is usually localized within a local area network, therefore large amounts of data are not usually exchanged over these long haul links.

Recent advances in quality of service over IP based networks are now providing a means for allowing other types of information to be transmitted across these networks. This opens the door for transmitting real-time information (i.e., audio and video) across the infrastructure in addition to the non-real-time data traffic. Video conferencing services that take advantage of network quality of service are well suited to overlay onto this infrastructure. It is now possible that two users at two different geographic locations can take place in a real-time videoconference session. One disadvantage of a videoconference session is that the transmission of real-time video can consume an extremely large amount of bandwidth and easily deplete available network resources. The bit rates of real-time video transmitted across a network mainly depend on the video resolutions and compression algorithms used. Typically, one videoconference session between two, three, or four users at different geographic locations can be properly supported on a network with a reasonable amount of bandwidth. However, it has been the case that, in general, additional users beyond four in a videoconference session could not be supported nor could a second videoconference session be supported due to bandwidth constraints. The limiting factors of the videoconference system are the low speed long haul links between the geographic locations.

One possible solution is to increase the bandwidth of the long haul links between the two geographic locations in order to support more users in the system. The drawback to this approach is that the bandwidth is very expensive. A second solution is to have a system where only a limited amount of users (i.e., the active users) in the videoconference session are allowed to transmit at a high resolution and

high bit-rate, and the remaining users (i.e., the passive users) in the session can only transmit at a limited bit-rate and limited resolution. The videoconference session organizer will have control of which users will transmit in high resolution and which users will transmit in low resolution. If a user is not actively talking or interacting in the session, then there is no need to send their video in high resolution. Such an approach can provide a tremendous amount of savings in bandwidth.

Referring ahead to the videoconference client application 1800 of FIG. 18A, this approach involves having a user interface 1808 in the videoconference client application 1800 that supports various window sizes (i.e., different sized display windows to represent the high-resolution and low-resolution decoded video streams) and a messaging system 1842 (included in the network entity 1806 that, in turn, is included in the videoconference client application 1800 of FIG. 18A) that specifies communication between the server 205 and the other client's applications. The messaging system 1842 will include messages that control the encoding resolution and transmitting bit-rate of each of the client's applications.

A description will now be given of messages corresponding to resolution and frame rate adjustment, according to an illustrative embodiment of the present invention. In particular, an MSG\_WINDOW\_SWITCH message and a MSG\_ADJUST\_CODEC message will be described.

The MSG\_WINDOW\_SWITCH message is sent from the client to the server indicating a switch between an active user and a passive user; that is, the active user becomes passive, and the passive user becomes active. The videoconference server will acknowledge this request with the client.

The MSG\_ADJUST\_CODEC message is sent from the server to each client. The MSG\_ADJUST\_CODEC message will indicate to the client what resolution (i.e.,

CIF or QCIF) and frame rate the client should be sending. The MSG\_ADJUST\_CODEEC message is acknowledged by each client.

FIG. 15 is a diagram illustrating a signaling method for resolution and frame rate adjustment, according to an illustrative embodiment of the present invention.

5 The example of FIG. 15 includes a videoconference server (server) 205, a client 1 1504, a client 2 1506, a client 3 1508, and a client 4 1510.

A MSG\_WINDOW\_SWITCH message is sent from the client 1 1504 to the server 205 (step 1520). An acknowledge message ACK is sent from the server 205 to the client 1 1504 (step 1525).

10 A MSG\_ADJUST\_CODEEC (low) message is sent from the server 205 to client 1 1504 (step 1530). An acknowledge message ACK is sent from client 1 1504 to the server 205 (step 1535).

A MSG\_ADJUST\_CODEEC (high) message is sent from the server 205 to the client 2 1506 (step 1540). An acknowledge message ACK is sent from the client 2 1506 to the server 205 (step 1545).

A MSG\_ADJUST\_CODEEC (low) message is sent from the server 205 to the client 3 1508 (step 1550). An acknowledge message ACK is sent from the client 3 1508 to the server 205 (step 1555).

20 A MSG\_ADJUST\_CODEEC (low) message is sent from the server 205 to the client 4 1510 (step 1560). An acknowledge message ACK is sent from the client 4 1510 to the server 205 (step 1565).

FIG. 16 is a diagram illustrating signaling before resolution and frame rate adjustment (clients 2 and 3), according to an illustrative embodiment of the present invention. FIG. 17 is a diagram illustrating signaling after resolution and frame rate adjustment (clients 2 and 3), according to an illustrative embodiment of the present

25

invention. The examples of FIGs. 16 and 17 include a client 1 1602, a client 2 1604, a network router 1606, a client 3 1608, and a client 4 1610.

A "send at low bit-rate/resolution" message is sent from the client 1 1602 to network router 1606 (step 1620). A "send at high bit-rate/resolution" message is sent  
5 from the client 3 1608 to network router 1606 (step 1625). A "send at low bit-rate/resolution" message is sent from the client 2 1604 to network router 1606 (step 1630). A "send at high bit-rate/resolution" message is sent from the client 4 1610 to network router 1606 (step 1635).

Data is sent from the network router 1606 to the client 2 1604, the client 3  
10 1608, the client 1 1602, and the client 4 1610, using the multicast address (steps 1640, 1645, 1650, and 1655, respectively).

Proceeding to FIG. 17, a "send at low bit-rate/resolution" message is sent from the client 1 1602 to network router 1606 (step 1720). A "send at high bit-rate/resolution" message is sent from the client 3 1608 to network router 1606 (step  
15 1725). A "send at high bit-rate/resolution" message is sent from the client 2 1604 to network router 1606 (step 1630). A "send at low bit-rate/resolution" message is sent from the client 4 1610 to network router 1606 (step 1635).

Data is sent from the network router 1606 to the client 2 1604, the client 3  
20 1608, the client 1 1602, and the client 4 1610, using the multicast address (steps 1740, 1745, 1750, and 1755, respectively).

A description will now be given of a client application architecture, according to an illustrative embodiment of the present invention. The client application is responsible for interacting with a user, exchanging of multimedia content with other client applications and for managing calls with the server application. Moreover, it is  
25 to be appreciated that the client application is also capable of including server

functionality within itself. FIG. 18A is a block diagram of a videoconference client application 1800, according to an illustrative embodiment of the present invention. It is to be appreciated that the videoconference client application 1800 may be found on a computer such as any of computers 220a-f and/or any of computers 230a-c.

5           The videoconference client application 1800 includes the following four basic functional entities: a multimedia interface layer 1802; codes 1804 (audio codecs 1804a & video codecs 1804b); a network entity 1806; and a user interface 1808.

          The multimedia interface layer 1802 is the main controlling instance of the videoconference client application 1800. All intra-system communication is routed  
10   through and controlled by the multimedia interface layer 1802. One of the key underlying features of the multimedia interface layer 180 is the ability to easily interchange different audio and video codecs 1804. In addition to this, the multimedia interface layer 1802 provides an interface to the Operating System (OS) dependent user input/output entity and network sub-systems. The multimedia interface layer  
15   1802 includes a member database 1820, a main control module 1822, an audio mixer 1899, and an echo cancellation module 1898.

          The user interface 1808 provides the point of interaction for an end user with the videoconference client application 1800. The user interface 1808 is preferably but not necessarily implemented as an OS dependent module. Many graphical user  
20   interfaces are dependent on the particular OS that they are using. The four major functions of the user interface 1808 are video capture, video display, audio capture, and audio reproduction. The user interface 1808 includes an audio/video capture interface 1830, an audio/video playback module 1832, a member view module 1834, a chat module 1836, and user selection/menus 1838. The audio/video capture  
25   interface 1830 includes a camera interface 1830a, a microphone interface 1830b,

and a file interface 1830c. The audio/video playback module 1834 includes a video display 1832a, an audio playback module 1832b, and a file interface 1832c.

The network entity 1806 represents the communication sub-system of the videoconference client application 1800. The functions of the network entity 1806 are client to server messaging that is based on Session Initiation Protocol (SIP) and the transmission and reception of audio and video streams. The network entity 1806 also includes basic security functions for authentication and cryptographic communication of the media streams between clients. The network entity 1806 includes a security module 1840, a messaging system 1842, a video stream module 1844, an audio stream module 1846, and IP sockets 1848a-c.

The audio codecs 1804a and the video codecs 1804b are the sub-systems that handle the compression and decompression of the digital media. The interfaces to the codecs should be simple and generic in order to make interchanging them easy. A simple relationship between the multimedia interface layer 1802 and the codecs 1804 is defined herein after as an illustrative template or guide for implementation. The audio codecs 1804a and video codecs 1804b each include an encoder 1880 and a decoder 1890. The encoder 1880 and decoder 1890 each include a queue 1895.

The videoconference client application 1800 interfaces with, at the least, the videoconference server 205 and other clients 1870.

A description will now be given of the member database 1820 included in the multimedia interface layer 1802 of FIG. 18A, according to an illustrative embodiment of the present invention. The member database 1820 stores information about each participating user on a per session basis. The member database 1820 includes information pertaining to the sending/receiving IP address, client capabilities,

information about particular codecs, and details about the status of the different users. It is to be appreciated that the preceding items are merely illustrative and, thus, other items in addition to or in place of some or all of the preceding items may also be kept in the member database 1820, while maintaining the spirit and scope of the present invention. The information included in the member database 1820 is used for controlling incoming information destined for the audio and video decoders 1890. The media information incoming from the network needs to be routed to the correct audio and video decoders 1890. Equally important, the media information coming from the audio and video encoders 1890 needs to be routed to the correct unicast or multicast address for distribution. Basic information included in the member database 1820 is also routed to the user interface 1808 in order for the end user to be aware of the participants in the session and their capabilities. A user is added to the member database 1820 as soon as an *INVITE* request is received from the videoconference server 205 and a user is removed as soon as a *BYE* request is received from the videoconference server 205. The member database 1820 is flushed when a session is terminated.

A description will now be given of the main control module 1822 included in the multimedia interface layer 1802 of FIG. 18A, according to an illustrative embodiment of the present invention.

The main control module 1822 is a very important part of the multimedia interface layer 1802. The main control module 1822 functions as the central management sub-system and provides the following key functions: synchronization mechanism for audio and video decoders and playback; connects destination of a decoder to screen or to file for recording purposes; and application layer Quality of Service.

The synchronization of audio and video playback is crucial for an optimal videoconferencing user experience. In order to accurately synchronize the two media streams, timestamps will need to be used and transmitted with the media content. Real Time Protocol (RTP) provides a generic header for including timestamps and sequence numbers for this purpose. The timestamps provided are NOT intended to  
5 synchronize the two network node clocks, but are intended to synchronize the audio and video streams for consistent playback. These timestamps will need to be derived from a common clock on the same node at the time of capture. For example, when a video frame is captured, the time when the video frame was captured must be  
10 recorded. The same applies to audio. Additional details and guidelines for using RTP are described elsewhere herein.

The function of the main control module 1822 in synchronizing the audio and video is to make the connection between the network entity 1806 and the codecs 1804 in order for proper delivery of the metadata (including timestamps and  
15 sequence numbers) and multimedia data. If packets are late, then they can be dropped before or after decoding depending on the current conditions of the system. The RTP timestamps are subsequently used to create the presentation and playback timestamps.

The main control module 1822 is also responsible for directing the output of  
20 the audio and video decoders 1890 to the screen for playback, to file for recording, or to both. Each decoder 1890 is treated independently, therefore this allows in an example situation for the output of one decoder to be displayed on the screen, the output of a second decoder to be recorded in a file, and the output from a third decoder to go both to a file and to the screen simultaneously.



In addition to the above-mentioned responsibilities, the main control module 1822 is also involved in application layer quality of service. The main control module 1822 gathers information regarding packet drops, bytes received and sent, and acts accordingly based on this information. This could involve sending a message to  
5 another client or to the videoconference server 205 to help remedy a situation that is occurring in the network. Real Time Control Protocol (RTCP) can be used for reporting statistics and packet losses, and can also be used for application specific signaling.

FIG. 18B is a block diagram further illustrating the audio mixer 1899 included  
10 in the multimedia interface layer 1802 of FIG. 18A, according to an illustrative embodiment of the present invention. The audio mixer 1899, also referred to herein as a "gain control module"), is operatively coupled to a plurality of audio decoders 1890. The multiple audio decoders 1880 receive compressed audio streams and output uncompressed audio streams. The uncompressed audio streams are input to  
15 the audio mixer 1899 and output as a combined audio stream.

FIG. 18C is a block diagram further illustrating the echo cancellation module 1898 included in the multimedia interface layer 1802 of FIG. 18A, according to an illustrative embodiment of the present invention. The echo cancellation module (also referred to herein as "echo canceller") 1898 is operatively coupled to a speaker 1897  
20 (e.g., audio playback module 1832b) and a microphone 1896 (e.g., microphone interface 1830b). When sound from the speaker 1897 is produced in a full duplex or two-way communication system, it is intended to be heard only from the local listener. However, the produced sound is also heard by the local microphone 1896, which then allows the signal to transmit back to the distant end and is heard as echo. For  
25 this reason, the videoconference client application 1800 requires the echo

cancellation module 1898 to mitigate this effect, thereby creating a better user experience.

A description will now be given of interfaces available to the sub-systems of the videoconference client application 1800, according to an illustrative embodiment of the present invention. The interfaces include the points of interaction with the user interface 1808, the network entity 1806, and the codecs 1804. The user interface 1808 provides functions for receiving captured audio and video along with their corresponding timestamps. In addition to this, functions must be provided for sending audio and video to the user interface 1808 for display and reproduction. The network entity 1806 interface provides functions for signaling incoming and outgoing messages for session control and security. The audio and video codecs 1804a,b provide a basic interface for configuration control as well as to send and receive packets for compression or decompression.

A description will now be given of the audio and video codecs 1804a,b, according to an illustrative embodiment of the present invention.

There are several audio and video codecs available for use in videoconferencing. Preferably but not necessarily, the codecs employed in accordance with the present invention are software based. According to one illustrative embodiment of the present invention, H.263 is used for video compression and decompression due to the processing power constraints of typical desktop computers. As desktop computers become more powerful in the future, the ability to use a more advanced codec such as H.26L can be realized and taken advantage of. Of course, the present invention is not limited to the preceding types of codecs and, thus, other types of codecs may be used while maintaining the spirit and scope of the present invention.

A description will now be provided of the interface to the codecs 1804a,b, according to an illustrative embodiment of the present invention. The description will encompass a DataIn function, callback functions, and codec options. The interface to the codecs 1804a,b should be flexible enough and defined in a general sense to  
5 allow interchangeability of codecs as well as to allow the addition of new codecs in the future. The proposed interface for implementing this flexible and general interface is a very simple interface with a limited number of functions provided to the user.

The DataIn function is simply used to store a frame or a packet of the encoder  
10 or decoder class.

In order to provide a simple connection between the multimedia interface layer 1802 and the multimedia codecs 1804, the data output function should be implemented as a callback. The multimedia interface layer 1802 sets this callback function to the input function of the receiving entity. For example, when the codec  
15 has completed encoding or decoding a frame, this function will be called by the codec in order to deliver the intended information from the encode or decode process. Due to the constraints that the codec is not able to do anything while in this callback, this function should return as quickly as possible to prevent waiting and unnecessary delays in the system. The only additional wait that should be performed in this  
20 function should be a mutex lock when accessing a shared resource.

The range of options available to different types of codecs will vary. In order to satisfy the requirements for managing these options, a simple interface should be used. A text-based interface is preferred (but not mandated) because of the flexibility that it offers. There should be a common set of commands such as START and  
25 STOP, and then codec specific commands. This method offers a simple interface,

but adds additional complexity to the codec because a simple interpreter is required. As an example, an Options function can be generic enough to read and write options.

Example: Result = Options("start"); Result = Options("resolution=CIF"); etc.

5

For example, some of the common options between codecs should be standardized as follows: start; stop; pause; quality index (0 – 100); and resolution.

The quality index is a factor that describes the overall quality of the codec as a value between 0% and 100%. It follows the basic assumption that the higher the value the better the video quality.

10

FIG. 19 is a diagram illustrating a method employed by a decoder 1890 included in either of the audio codecs 1804a and/or the video codecs 1804b, according to an illustrative embodiment of the present invention. The method is described with respect to a decoder context 1901 and a caller context 1902. The method operates using at least the following inputs and outputs: "data in" 1999; "signal in" 1998; "signal out callback" 1997; "set callback function" 1996; and "data out callback" 1995. The input "data in" 1999 is used to store data into an input queue (step 1905).

15

An initialization step (Init) is performed to initialize the decoder 1890 (step 1910). A main loop is executed, that waits for a start or exit command (step 1920). If an exit command is received, then the method is exited (step 1922) and a return is made to, e.g., another operation (1924).

20

Data is read out of an input queue 1895 or a wait condition is imposed if the input queue 1895 is empty (step 1930). The data, if read out at step 1930, is decoded (step 1940). The "data out callback" 1995 is provided to step 1920.

25

A description will now be given of the communications employed by the network 200, according to an illustrative embodiment of the present invention. The description supplements that provided above with respect to network communications.

5           The messaging system 1842 (included in the network entity 1806 of FIG. 18A) provides the interface between the videoconference client application 1800 and the videoconference server 205. It is intended to be used for session management (i.e., session setup and teardown). All signaling messages are communicated through the videoconference server 205 and not directly from client to client. Data such as  
10 multimedia content and private chat messages comprise the only information sent directly between clients. The messaging system will use the standards based Session Initiation Protocol (SIP).

There are several different protocols that govern the functionality of the videoconference client application 1800. For example, Session Initiation Protocol  
15 (SIP), Real Time Protocol (RTP), Real Time Control Protocol (RTCP), and Session Description Protocol (SDP) may be employed.

The purpose of Session Initiation Protocol (SIP) is session management. SIP is a text based application layer control protocol for creating, modifying and terminating multimedia sessions with one or more participants on IP based networks.  
20 SIP is used between the client and the server to accomplish this. SIP is described further above with respect to the videoconference server 205.

Real Time Protocol (RTP) is used for the transmission of real-time multimedia (i.e., audio and video). RTP is an application layer protocol for providing additional details pertaining to the type of multimedia information it is carrying. RTP resides  
25 above the transport layer and is usually carried on top of the User Datagram Protocol

(UDP). The primary function of RTP in the client application will be for transporting timestamps (for audio and video synchronization), sequence numbers, as well as identify the type of payload it is encapsulating (e.g., MPEG4, H.263, G.723, etc.).

FIG. 20 is a diagram illustrating a user plane protocol stack 2000, according to an illustrative embodiment of the present invention. The stack 2000 includes video 2010 and voice 2020 on one layer, RTP 2030 for both video 2010 and voice 2020 on another layer, UDP Port #X 2040 and UDP Port #Y 2050 on yet another layer, an IP layer 2060, a link layer 2070, and a physical layer 2080. Codec specific RTP headers are used in addition to a generic RTP header.

Real Time Control Protocol (RTCP) is part of the RTP standard. RTCP is used as a statistics reporting tool between senders and receivers. Each videoconference client application 1800 will gather their statistics and send them to one another as well as to the server 205. The videoconference server 205 will record information about problems that may have occurred in the session based on this data.

FIG. 21 is a diagram illustrating a control plane protocol stack 2100, according to an illustrative embodiment of the present invention. The stack 2100 includes SIP 2110, UI codec change messaging 2120, and RTCP 2130 on one layer, a TCP layer 2140, an IP layer 2150, a link layer 2160, and a physical layer 2170.

The main purpose of SDP is to convey information about media streams of a session. SDP includes, but is not limited to, the following items: session name and purpose; time the session is active; the media comprising the session; information to receive the media (i.e., addresses, ports, formats, etc.); type of media; transport protocol (RTP/UDP/IP); the format of the media (H.263, etc.); multicast; multicast

address for the media; transport port for the media; unicast; and remote address for the media.

The SDP information is the message body for a SIP message. They are transmitted together.

5           A further description will now be given of the user interface 1808 of FIG. 18A, according to an illustrative embodiment of the present invention. The user interface 1808 is a very important element of the videoconference client application 1800. The user interface 1808 includes several views (display/buttons/menus/...) and can handle all the input data (audio/video capture, buttons, keystrokes).

10           FIG. 22 is a block diagram illustrating a screen shot 2200 corresponding to the user interface 1808 of FIG. 18A, according to an illustrative embodiment of the present invention. The screen shot 2200 includes "big views" 2210, "small views" 2220, a chat view portion 2230, a member view portion 2240, and a chat edit portion 2250.

15           Referring again to FIG. 18A, the video capture interface 1830 can include any of the following: web cam (not shown); capture card and high quality camera (not shown); camera interface 1830a; microphone interface 1830b; file interface 1830c; and so forth.

          The web cam should be supported through either the USB or Firewire  
20   (IEEE1394) interface using the Video For Windows (VFW) Application Programming Interface (API) provided by the Windows operating system or through an alternative capture driver used under a different operating system such as Linux. Of course, the present invention is not limited to the preceding interfaces, operating systems, or drivers and, thus, other interfaces, operating systems, and drivers may also be used,  
25   while maintaining the spirit and scope of the present invention.

The member view module 1834 is used to show the members participating in the ongoing call. The initiator (i.e., Master) of the call can either drop unwanted members or select active members. Every member can select one or more members for a private chat message exchange. In addition, the status of a member is signaled  
5 in the member view module 1834. A member can then set their own status to, e.g., "Unavailable", to signal the other they are currently not available but will be back soon.

In addition to the video stream, every member has the opportunity to send chat messages to either all or only some other members using the chat module 1836. The  
10 messages are displayed in the chat view and edited in the chat edit view. A scrollbar allows viewing of older messages.

A description will now be given of operational scenarios for the client application 1800, according to an illustrative embodiment of the present invention. The following description is simply a basic guideline of some of the features of the  
15 client application 1800 and is not intended to represent a complete list of features. The description will encompass login, initiation of a call, acceptance of a call, and logoff.

The login is done when the client application 1800 is initially started. The login can be done automatically based on the login name provided to the operating system  
20 at startup, or a different interface can be used that is independent of the login. It depends on the preferred method of authentication for the network that is currently used and how policies are administrated. The simplest method would be to use the same login name as that used in the windows operating system to keep naming consistent and also to have the ability to reuse existing user databases (if applicable).



FIG. 23 is a diagram illustrating a login interface 2300, according to an illustrative embodiment of the present invention. The sign up feature 2330 is used if a user does not currently have an account on the server. Email addresses can be provided in any e-mail address input box 2340 for easy access.

5 To initiate a call, the client application 1800 will query the server 205 for a list of available candidates. The client can select the users he or she wishes to engage in a videoconference session. A session will be setup as unicast when two participants are involved; otherwise, when more than two participants are involved the session is set up as a multicast session.

10 FIG. 24 is a block diagram illustrating a user selection interface 2400 for session initiation, according to an illustrative embodiment of the present invention.

Once the user is invited to a call, a message showing the name of the initiator is displayed on their screen. The user can then either accept or reject the call. If the user accepts the call, then the client application 1800 sends an accept (or  
15 acknowledgement) message to the server 205. The server 205 then informs every member currently participating in the call about the new member. If the user declines the call by sending the cancellation message to the server 205, then all other members are also informed about that event. FIG. 25 is a block diagram illustrating an invitation interface 2500 for accepting or rejecting an incoming call, according to  
20 an illustrative embodiment of the present invention.

The logoff will remove the user from the member database 314 included in the database entity 302 of the videoconference server 205. A BYE message is sent to each participating client of the session. This can be done either through multicast or unicast. Multicast is the preferred method for sending this message.

FIG. 26 is a flow diagram illustrating a method for automatically initiating a videoconference session with multiple participants, according to an illustrative embodiment of the present invention. The multiple participants include an initiating participant and other participants. The videoconference session may be a unicast  
5 videoconference session or a multicast videoconference session.

A time and date for a videoconference session, as well as the other participants to the videoconference session, are selected by an initiating participant (2610). The other participants may be selected, e.g., using a participant selection mechanism.

10 It is then determined whether the other participants are available for a videoconference at the selected time and date (step 2620). Step 2620 may be performed by, for example, verifying the selected time and date against the schedules of the other participants to ensure that the other participants are available at the selected time and date.

15 The selected time, date, and a list of participants (i.e., the initiating participant and the other participants) are transmitted from the initiating participant to the server and stored by the server (step 2630). It is to be appreciated that in another embodiment of the present invention, the list of participants may omit the initiating participant; however, in such a case, the initiating participant may be ascertained by  
20 the server (from, e.g., IP address of initiating participant) and automatically included in the list at the server end or, at the least, be associated with the list at the server end. Of course, other arrangements are possible while maintaining the spirit and scope of the present invention.

An invitation is sent from the server to the other participants at the selected time and date that indicates that the initiating participant has requested a videoconference with them (step 2640).

The other participants can accept or decline the invitation message. A message that indicates an acceptance or a declination of the invitation is sent from each of the other participants to the initiating participant (step 2650). In the case that the message declines the invitation, such a message may include a reason for declining. The videoconference session is conducted with the initiating participant and any of the other participants that have accepted the invitation (step 2660).

FIG. 27 is a flow diagram illustrating a method for automatically initiating a multicast session between multiple participants, according to an illustrative embodiment of the present invention. The multiple participants include an initiating participant and other participants. The initiating participant desires to setup a scheduled multicast session to distribute content over a network to the other participants. The multicast session can be initiated from a live or pre-recorded source.

A time and date for the multicast session, as well as the other participants to the multicast session, are selected by the initiating participant (2710). The time, date, a list of participants (i.e., the initiating participant and the other participants), and a request for a multicast address to distribute the content on are transmitted from the initiating participant to the server (step 2720). It is to be appreciated that in another embodiment of the present invention, the list of participants may omit the initiating participant; however, in such a case, the initiating participant may be ascertained by the server (from, e.g., IP address of initiating participant) and automatically included in the list at the server end or, at the least, be associated with the list at the server

end. Of course, other arrangements are possible while maintaining the spirit and scope of the present invention.

The time, date, and list of participants are stored by the server (step 2730), and a multicast address is allocated by the server and sent to a multicast source  
5 (step 2740).

Each person in the list of participants (i.e., the initiating participant and the other participants) are notified (e.g., in their calendar, their e-mail program, and so forth) that a multicast session will be taking place at the (selected) time and date and provided with an option to record the multicast session on their local client device  
10 (step 2750). The latter is preferable when an intended participant is unable to participate in the multicast session or if an actual participant desires to have a record of the multicast session despite having participated therein.

At the (selected) time and date, an invitation is sent from the server to each person in the list of participants; the invitation includes (or may be separately  
15 associated with) the multicast address of the multicast session (step 2760).

The participants can accept or decline the invitation. If the invitation is declined, then the declining participants can choose to join at a later time. However, if the invitation is accepted, then the videoconference client application of each of the accepting participants transmits the Internet Group Multicast Protocol (i.e., IGMP) to  
20 the network (step 2770). The multicast session is then conducted between the initiating participant and accepting participants, and the content is sent to the multicast address from the multicast source (step 2780). In particular, the content is sent out to the network from the multicast source on the multicast address and is replicated at, e.g., the network routers.

Although the illustrative embodiments have been described herein with reference to the accompanying drawings, it is to be understood that the present invention is not limited to those precise embodiments, and that various other changes and modifications may be affected therein by one skilled in the art without departing  
5 from the scope or spirit of the invention. All such changes and modifications are intended to be included within the scope of the invention as defined by the appended claims.

WHAT IS CLAIMED IS:

1. A method for automatically initiating a videoconference session over a network, comprising the steps of:

5 receiving a pre-selection that specifies a time and clients for the videoconference session; and

sending an invitation message regarding the videoconference session to at least one of the clients at the specified time.

10 2. The method of claim 1, further comprising the step of receiving information that represents the specified time and the clients.

3. The method of claim 2, further comprising the step of storing the information.

15

4. The method of claim 1, further comprising the step of verifying that the clients are available for the videoconference session at the specified time.

5. The method of claim 4, wherein said verifying step comprises the step  
20 of checking the specified time against schedules of the clients.

6. The method of claim 1, further comprising the step of receiving a decline message from declining ones of the clients.

7. The method of claim 6, wherein the decline message includes a reason for declining the videoconference session.

8. The method of claim 1, further comprising the step of receiving an  
5 accept message in response to the invitation message.

9. A method for automatically initiating a multicast session over a network, said method comprising the steps of:

assigning a common multicast Internet Protocol (IP) address for the multicast  
10 session; and

sending an invitation message regarding the multicast session to at least one client at a time that is pre-selected, the invitation message specifying the common multicast IP address.

15 10. The method of claim 9, further comprising the step of transmitting the common multicast IP address to a source of content for the multicast session.

11. The method of claim 9, further comprising the step of providing an ability to record the content on a corresponding client device of the at least one client.  
20

12. The method of claim 9, further comprising the step of notifying the at least one client that the multicast session is to take place at the pre-selected time, prior to an actual arrival of the pre-selected time.

13. The method of claim 9, further comprising the step of receiving an Internet Group Multicast Protocol (IGMP) from accepting ones of the at least one client in response to the invitation message.

5 14. The method of claim 9, further comprising the step of receiving information that represents the pre-selected time and the clients.

15. The method of claim 14, further comprising the step of storing the information.

10

16. A system for automatically initiating a videoconference session over a network, comprising:

means for receiving a pre-selection that specifies a time and clients for the videoconference session; and

15 means for sending an invitation message regarding the videoconference session to at least one of the clients at the specified time.

17. The system of claim 16, further comprising means for receiving information that represents the specified time and the clients.

20

18. The system of claim 17, further comprising means for storing the information.

19. The system of claim 16, further comprising means for verifying that the  
25 clients are available for the videoconference session at the specified time.



20. The system of claim 19, wherein said means for verifying comprises means for checking the specified time against schedules of the clients.

5 21. The system of claim 16, further comprising means for receiving a decline message from declining ones of the clients.

22. The system of claim 21, wherein the decline message includes a reason for declining the videoconference session.

10

23. The system of claim 16, further comprising means for receiving an accept message in response to the invitation message.

24. A system for automatically initiating a multicast session over a network,  
15 comprising:

means for assigning a common multicast Internet Protocol (IP) address for the multicast session; and

means for sending an invitation message regarding the multicast session to at least one client at a time that is pre-selected, the invitation message specifying the  
20 common multicast IP address.

25. The system of claim 24, further comprising means for transmitting the common multicast IP address to a source of content for the multicast session.

25 26. The system of claim 24, further comprising means for record the

content on a corresponding client device of the at least one client.

27. The system of claim 24, further comprising means for notifying the at least one client that the multicast session is to take place at the pre-selected time, prior to an actual arrival of the pre-selected time.

5

28 The system of claim 24, further comprising means for receiving an Internet Group Multicast Protocol (IGMP) from accepting ones of the at least one client in response to the invitation message.

10

29. The system of claim 24, further comprising means for receiving information that represents the pre-selected time and the clients.

15

30. The system of claim 29, further comprising means for storing the information.

31. A method for joining a videoconference session over a network, comprising the steps of:

20

providing an ability to receive an invitation message regarding the videoconference session at a time that is pre-selected for the videoconference session to take place;

providing an ability to send an accept message in response to the invitation message; and

25

providing an ability to automatically receive content corresponding to the videoconference session, in response to sending the accept message.

32. The method of claim 31, further comprising the step of providing an ability to send a decline message so that the method is terminated prior to a receipt of the content.

5

33. The method of claim 32, wherein the decline message includes a reason for declining the videoconference session.

34. A method for joining a multicast session over a network, said method  
10 comprising the steps of:

providing an ability to receive an invitation message regarding the multicast session at a time that is pre-selected, the invitation message specifying a common multicast IP address for the multicast session;

15 providing an ability to send an accept message in response to the invitation message; and

providing an ability to automatically receive content corresponding to the multicast session, in response to sending the accept message.

35. The method of claim 34, further comprising the step of providing an  
20 ability to record the content.

36. The method of claim 34, further comprising the step of providing an ability to notify a corresponding client that the multicast session is to take place at the pre-selected time, prior to an actual arrival of the pre-selected time.

25

37. The method of claim 34, wherein said step of providing the ability to send the accept message comprises the step of providing an ability to send an Internet Group Multicast Protocol (IGMP) message to the network.

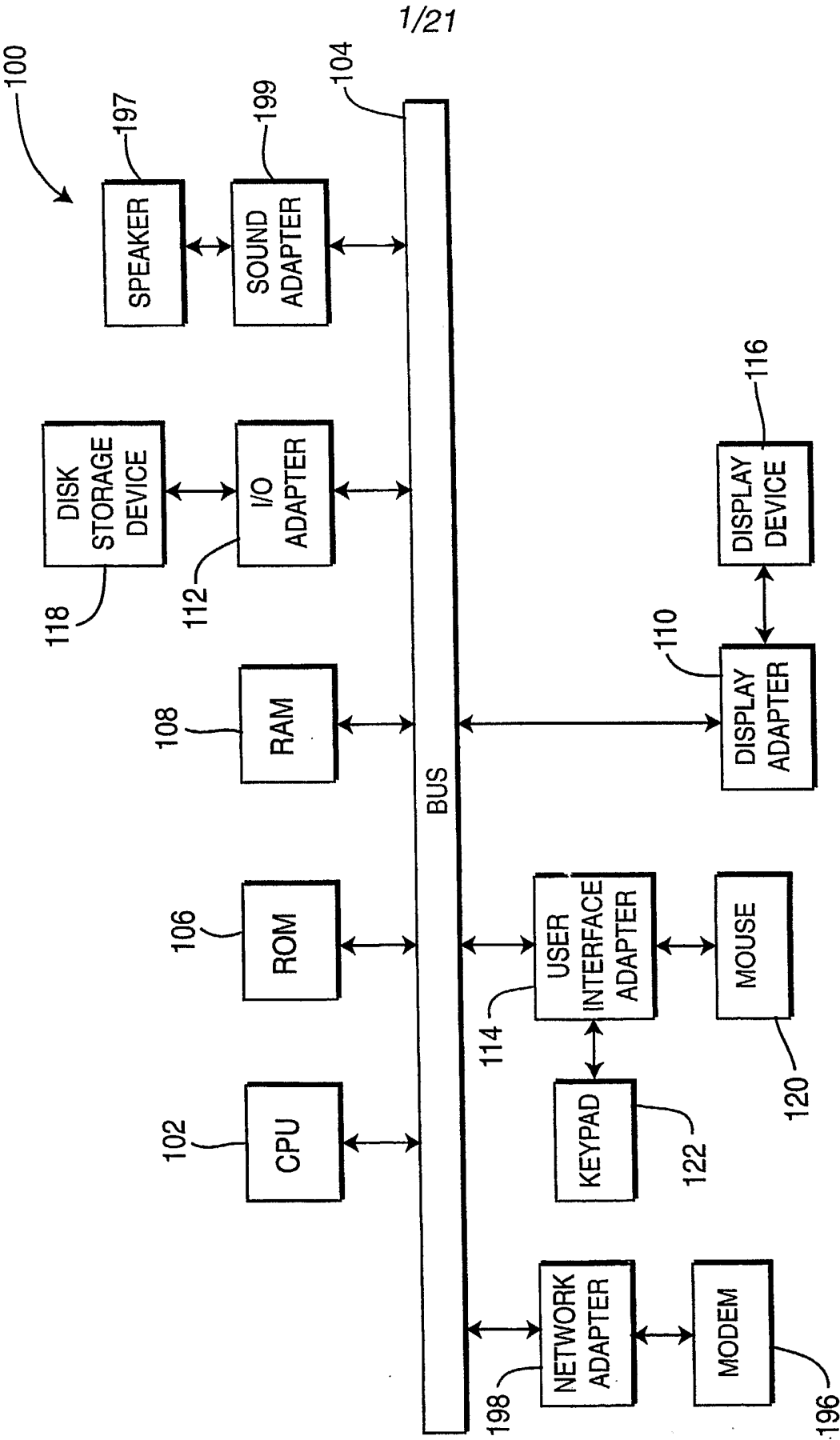
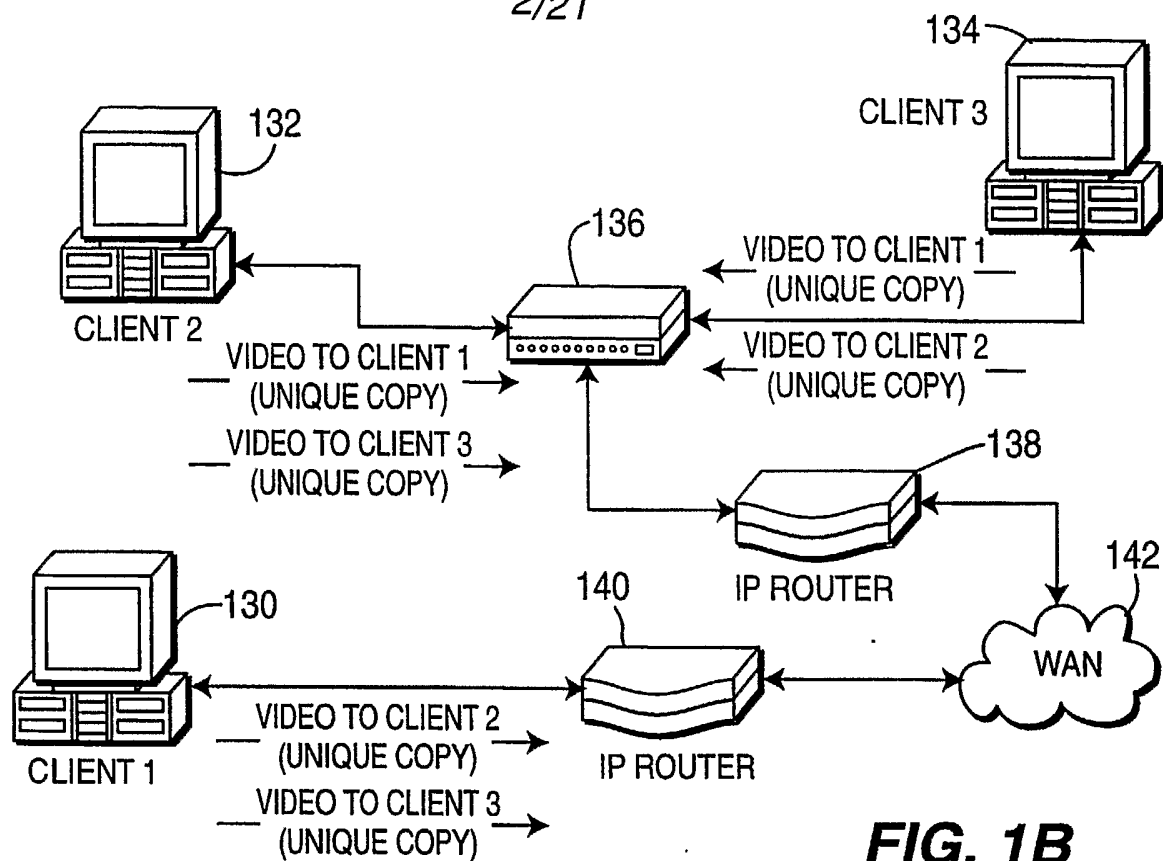
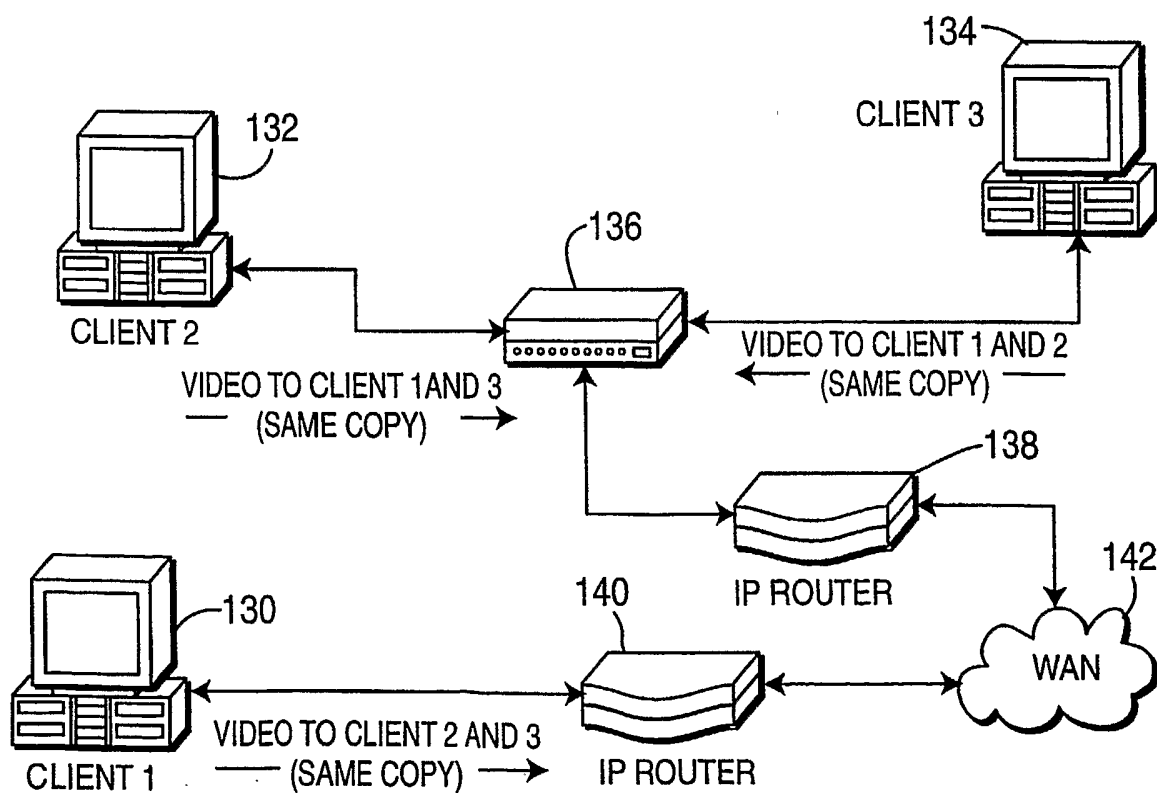


FIG. 1A

2/21

**FIG. 1B****FIG. 1C**

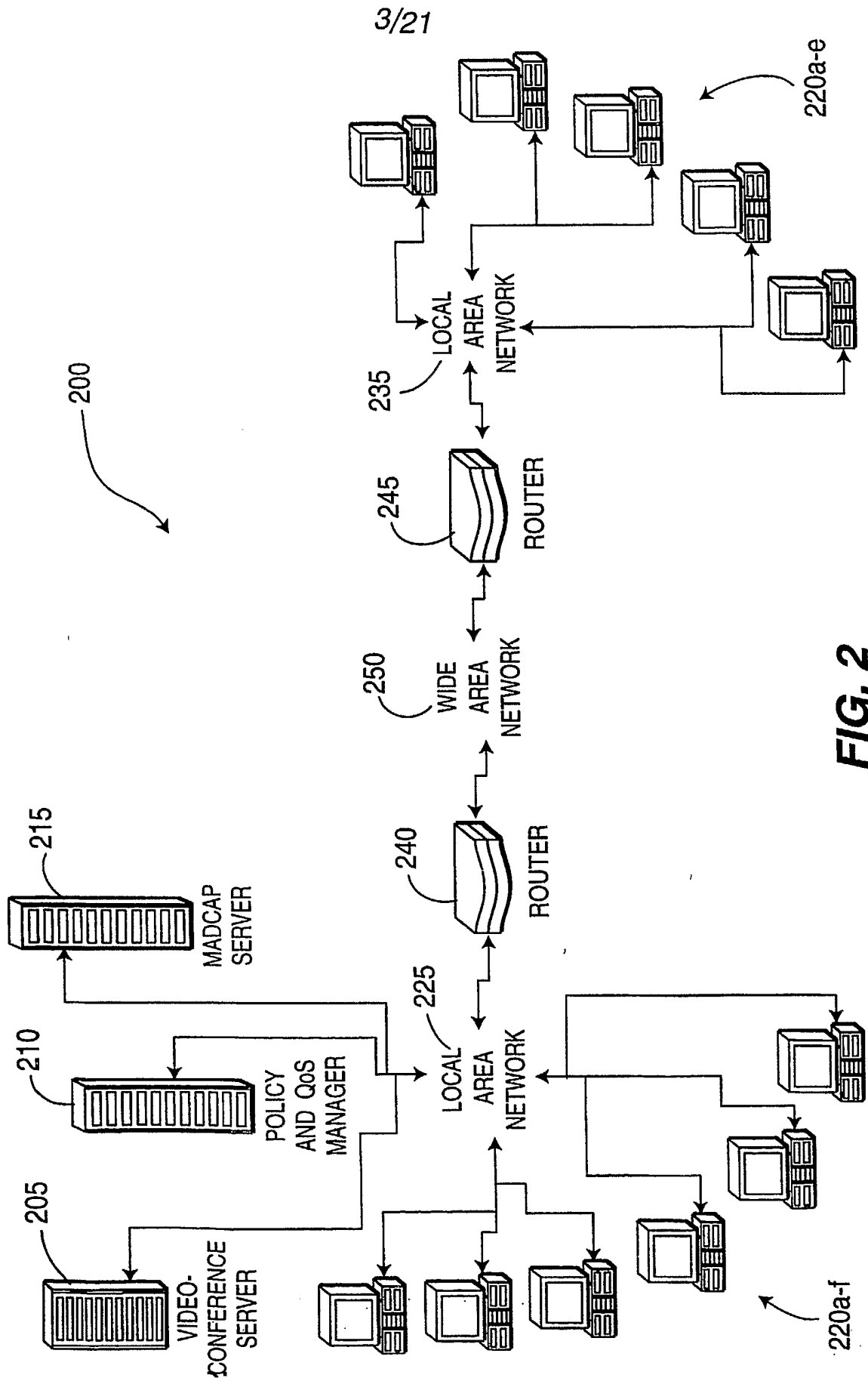
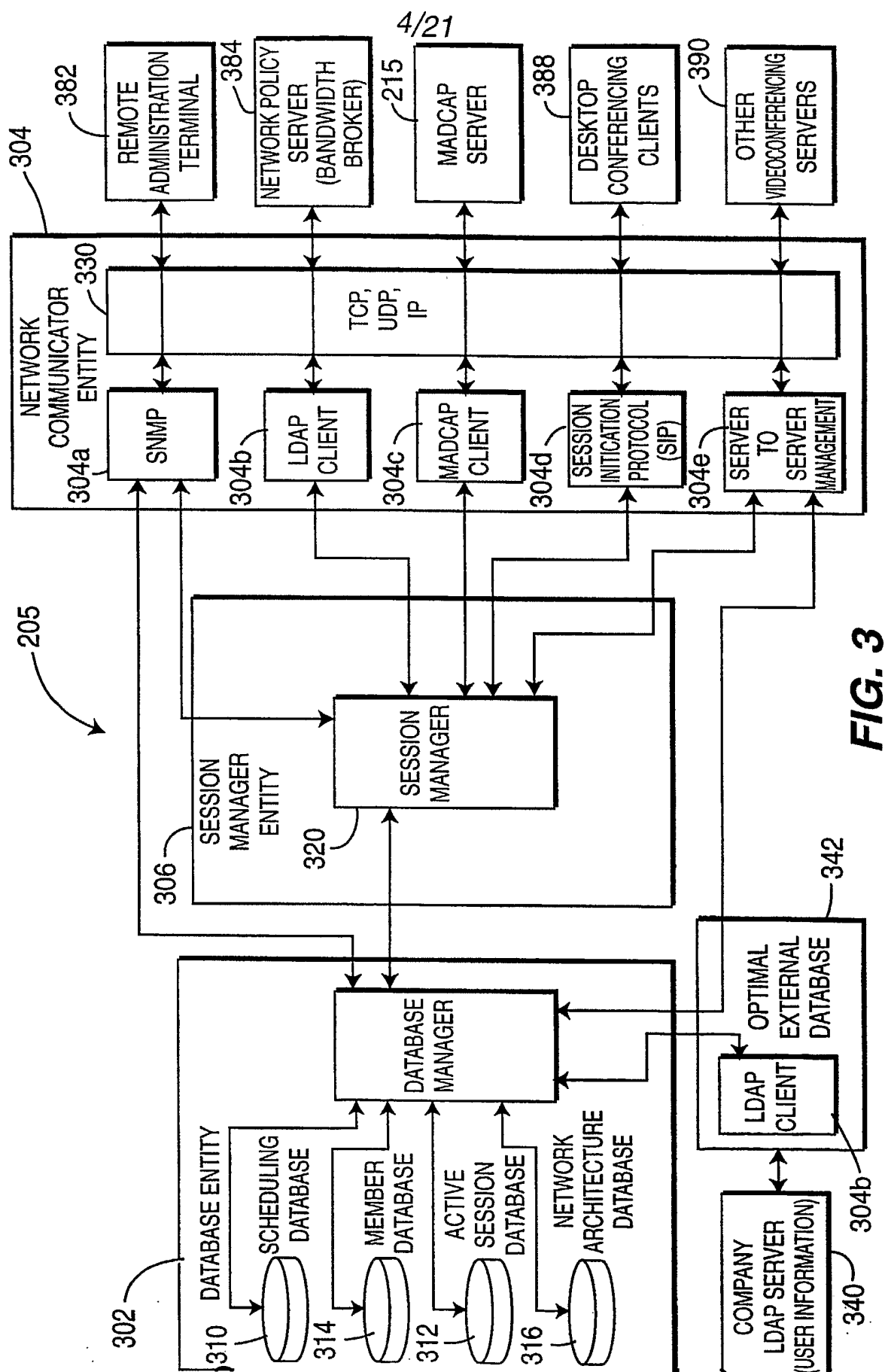


FIG. 2

**FIG. 3**



5/21

400

INFORMATION FIELD	INFORMATION
USERNAME	JOHN RICHARDSON
PASSWORD	THOMSON
SUPPORTED VIDEO CODECS	H.263
SUPPORTED AUDIO CODECS	G.723      G.71      1
SUPPORTED RESOLUTIONS	CIF      QCIF
IP ADDRESS	157.254.122.23
CURRENT CALL NUMBER	0 (NONE)
AVAILABILITY	AVAILABLE
LOCATION	1

**FIG. 4**

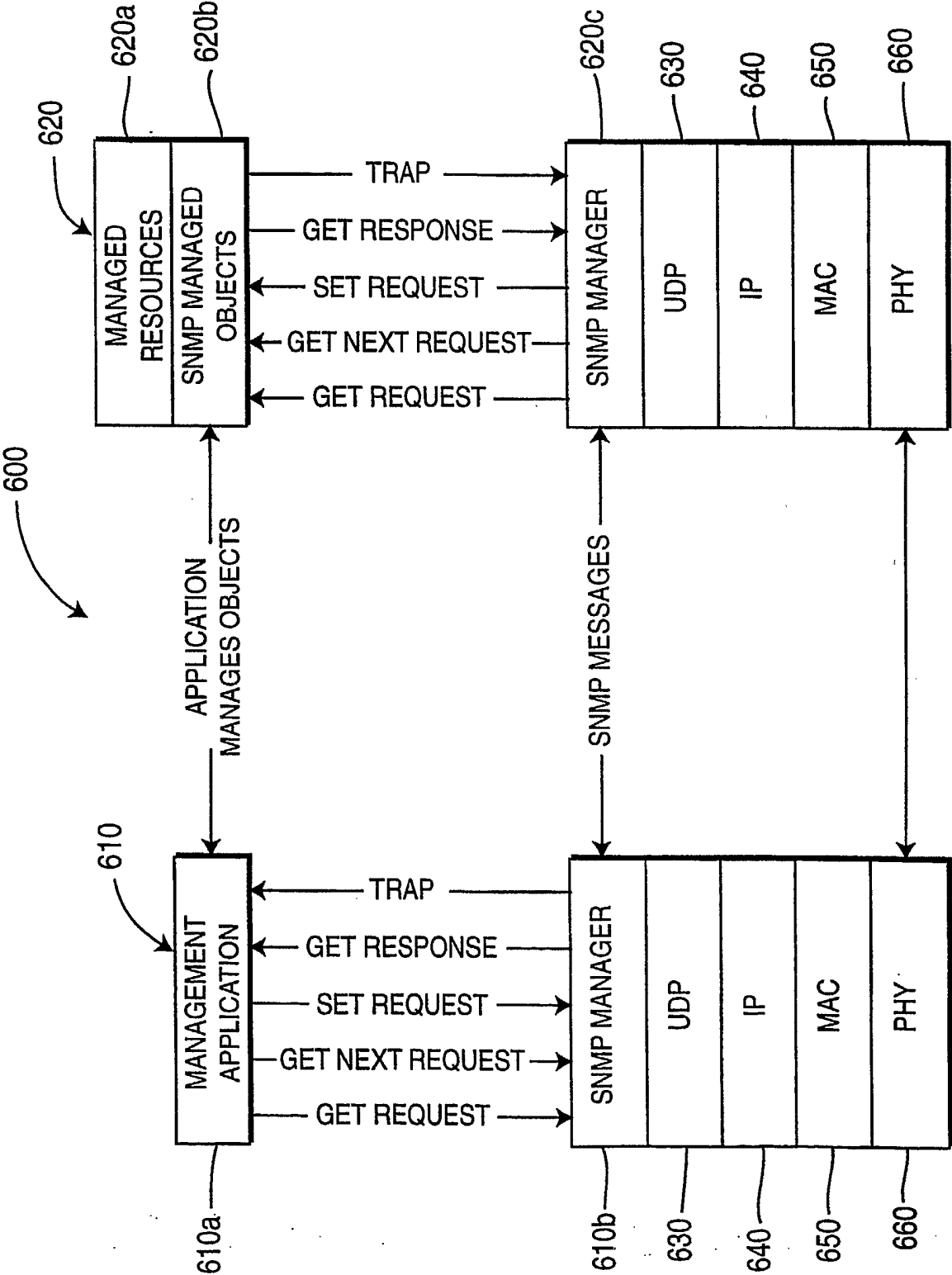
500

INFORMATION FIELD	INFORMATION
CALL ID	1
DESCRIPTION	MEETING ABOUT VIDEOCONFERENCE PROJECT
MULTICAST SESSION	YES--MULTICAST ADDRESS (226.3.4.5)
PARTICIPANT 1 IP ADDRESS	157.254.122.23--LOCATION (1)
TRANSMITTING RESOLUTION	CIF--(ACTIVE USER)
TRANSMITTING BITRATE	256kbps--(ACTIVE USER)
VIDEO CODEC	H.263
AUDIO CODEC	G.723
PARTICIPANT 2 IP ADDRESS	157.254.122.151--LOCATION (1)
TRANSMITTING RESOLUTION	QCIF--(PASSIVE USER)
TRANSMITTING BITRATE	30kbps--(PASSIVE USER)
VIDEO CODEC	H.263
AUDIO CODEC	G.723
PARTICIPANT 2 IP ADDRESS	157.254.96.99--LOCATION (2)
TRANSMITTING RESOLUTION	CIF--(ACTIVE USER)
TRANSMITTING BITRATE	256kbps--(ACTIVE USER)
VIDEO CODEC	H.263
AUDIO CODEC	G.723

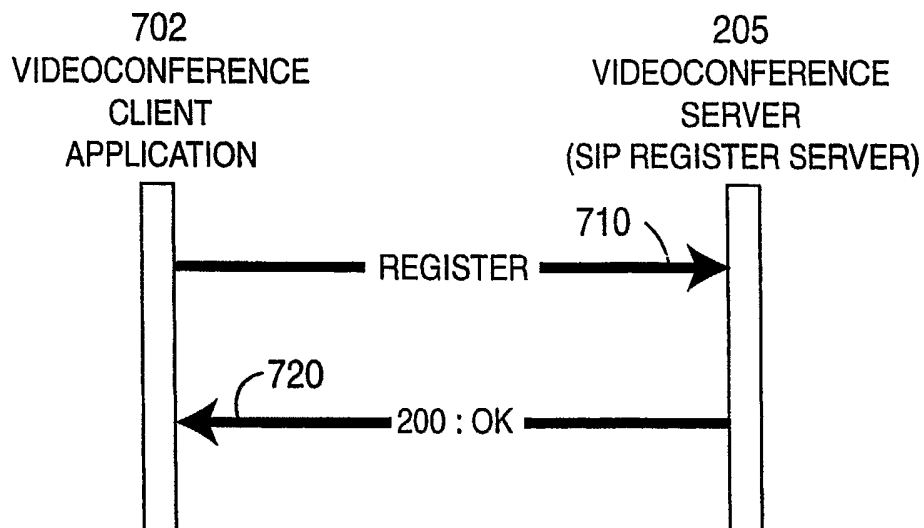
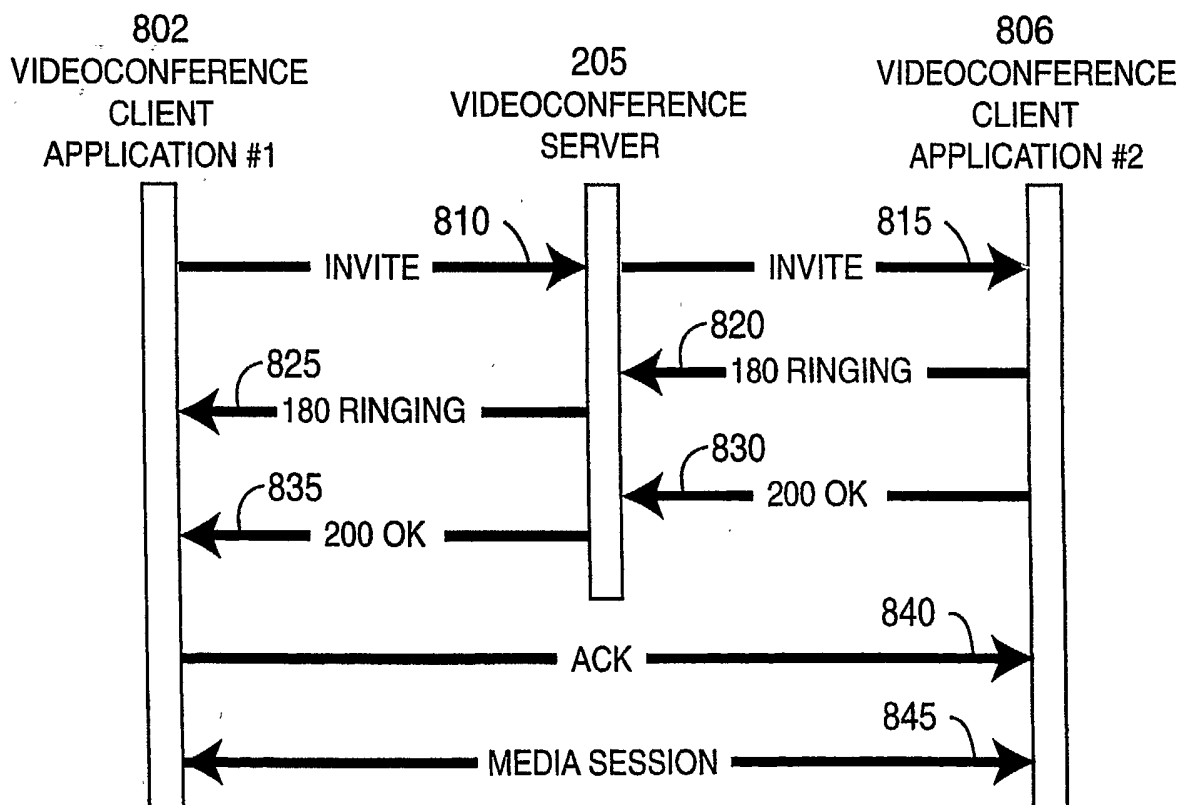
**FIG. 5**

6/21

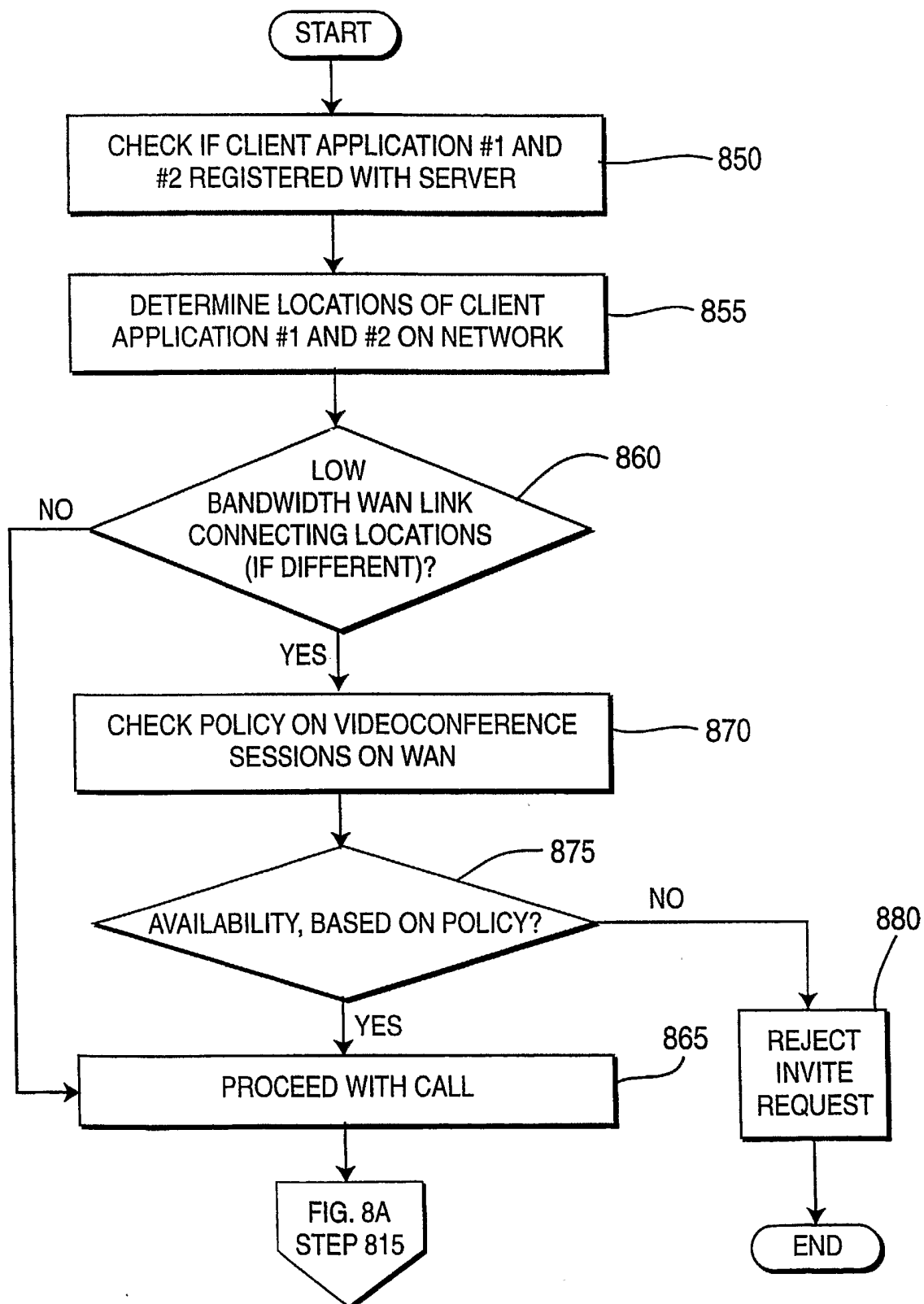
FIG. 6



7/21

**FIG. 7****FIG. 8A**

8/21

**FIG. 8B**

9/21

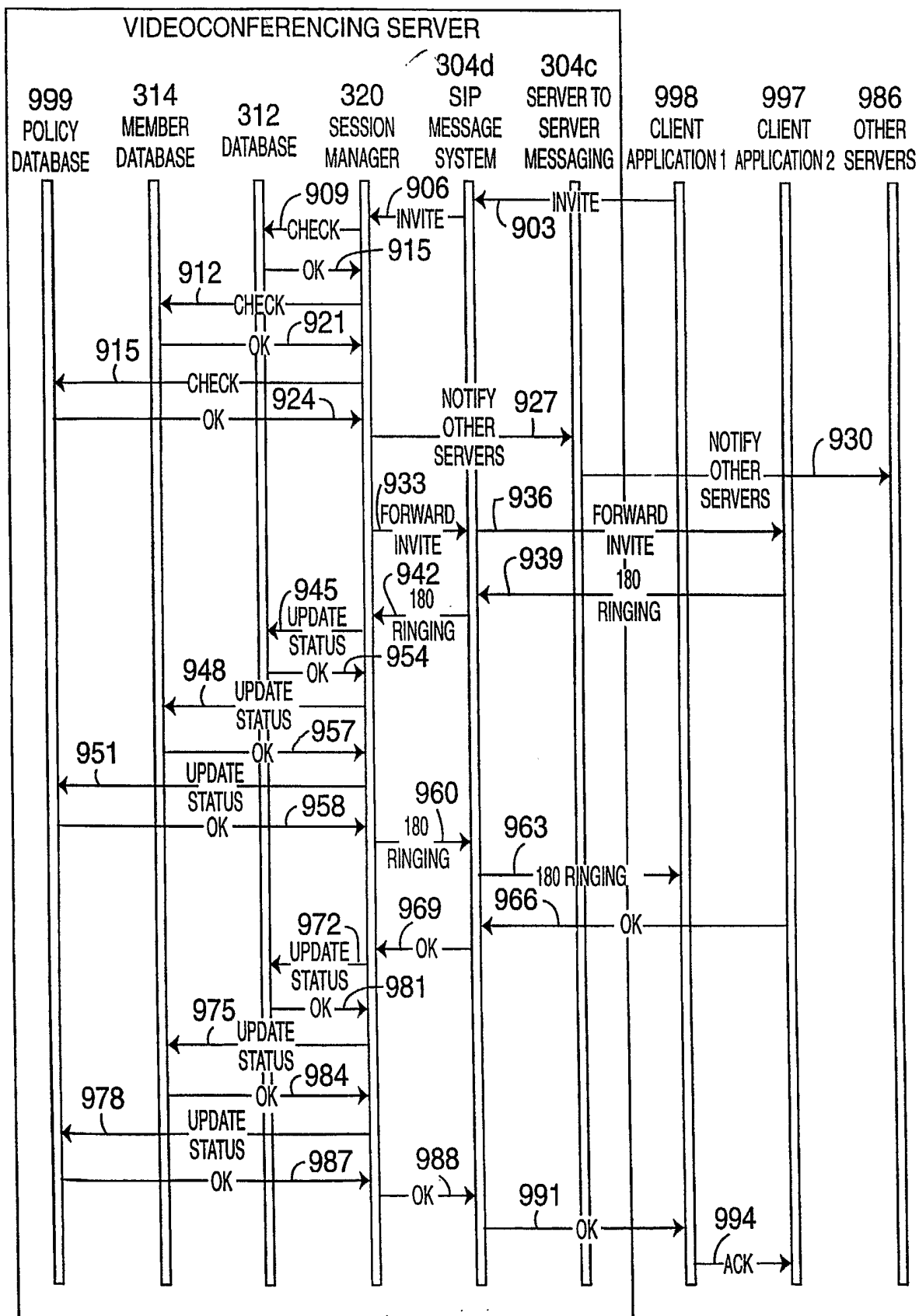
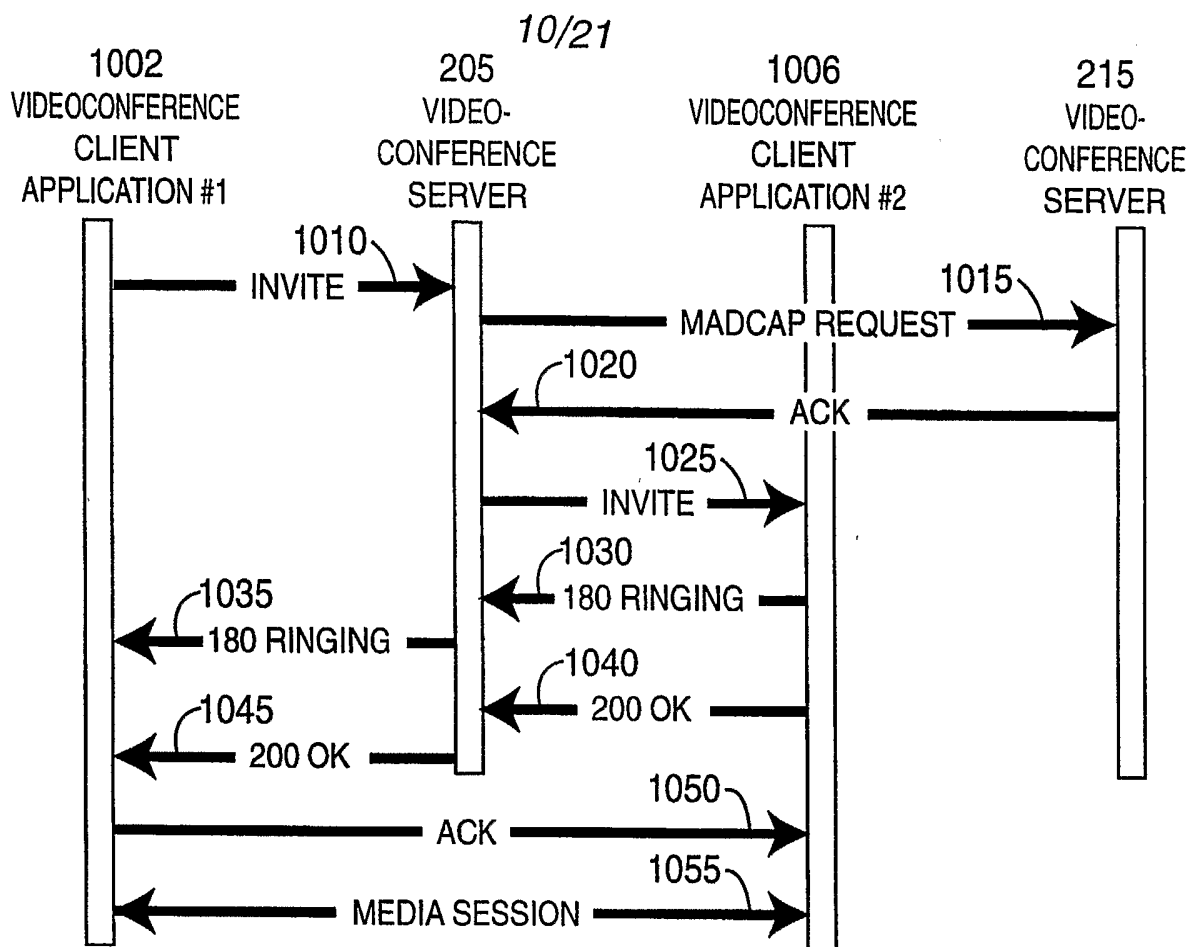
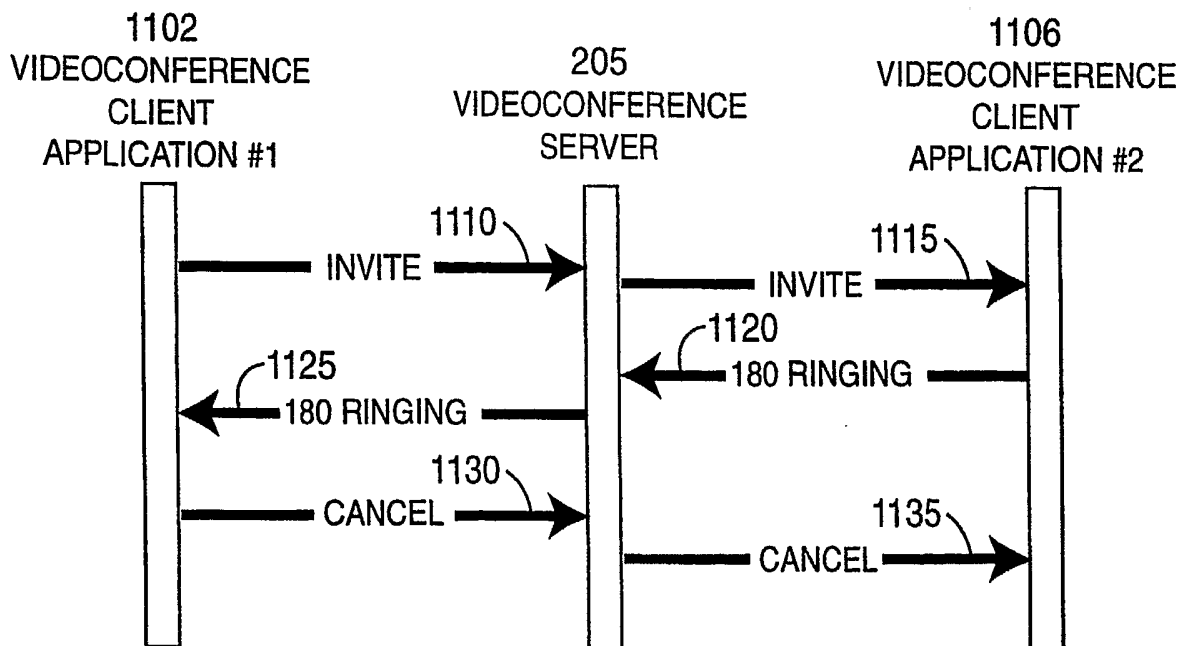
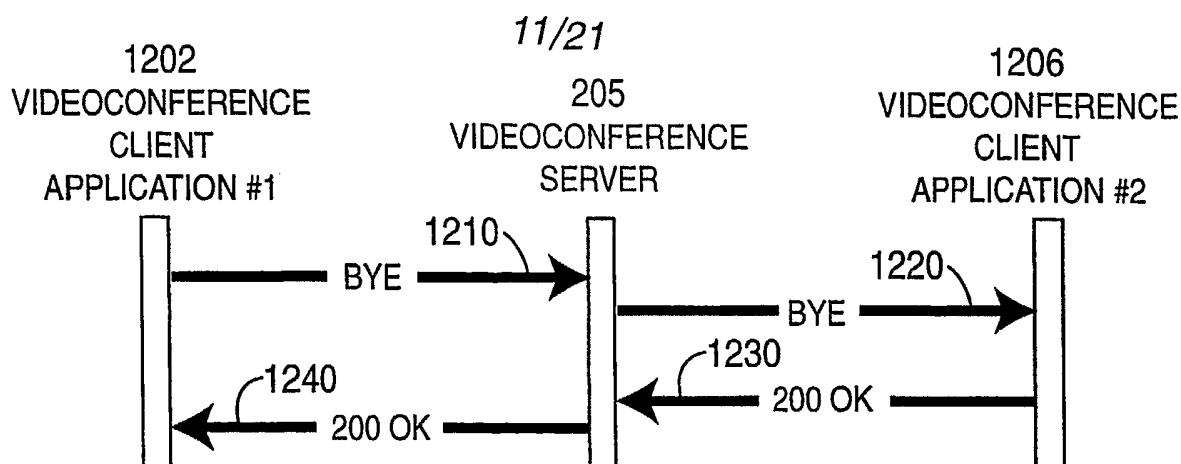
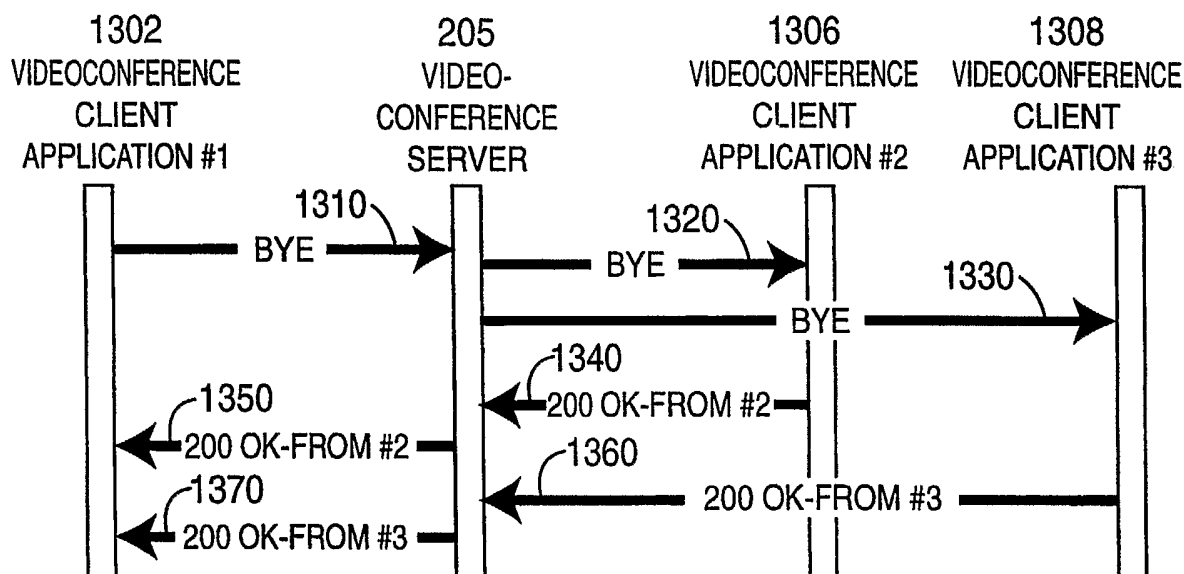
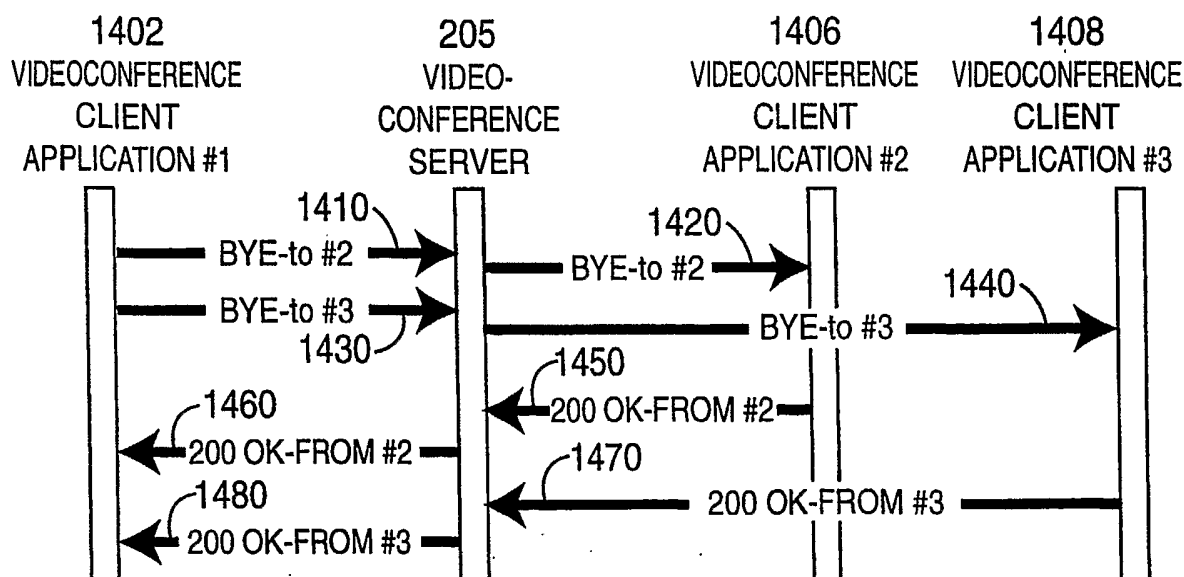
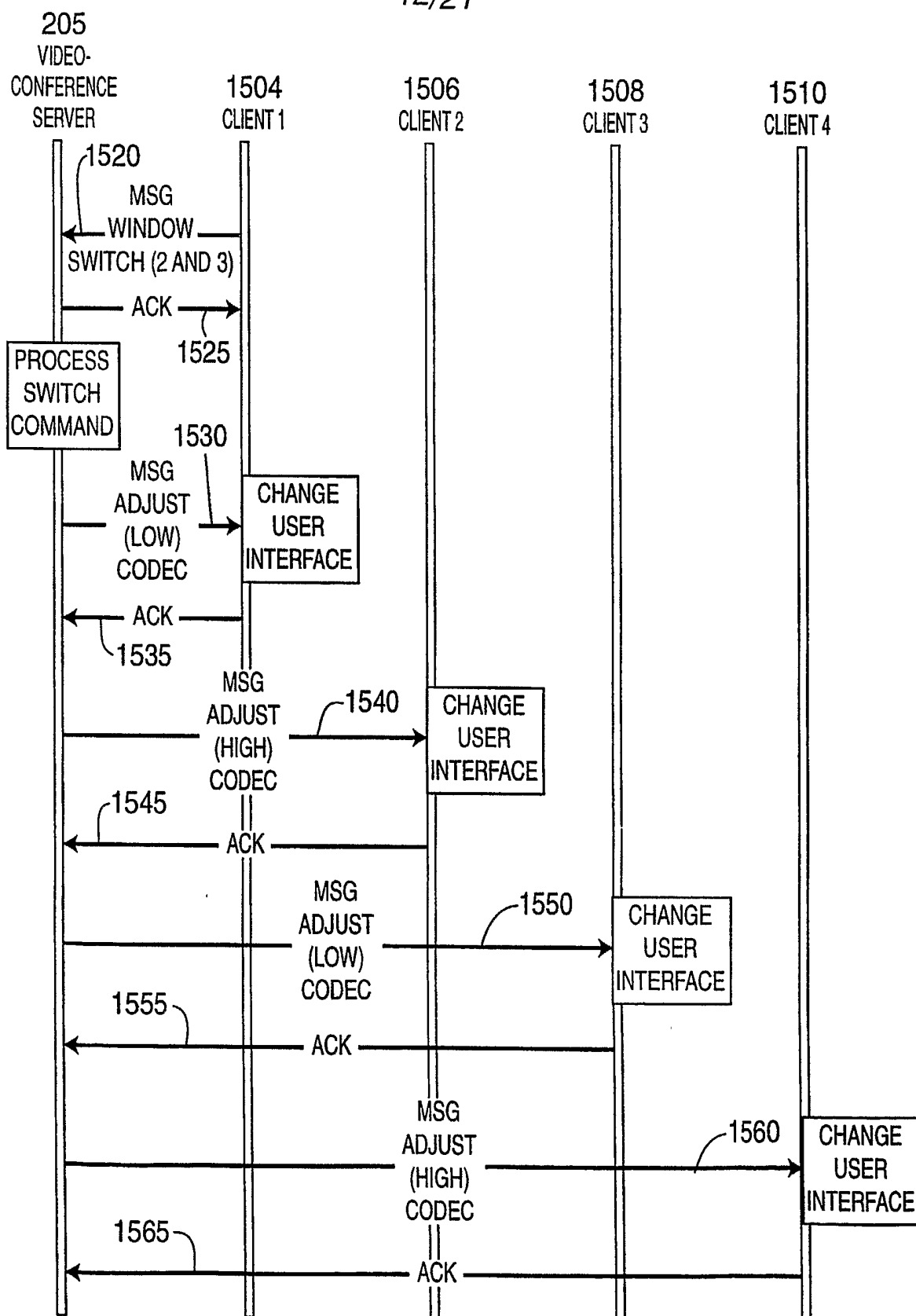


FIG. 9

**FIG. 10****FIG. 11**

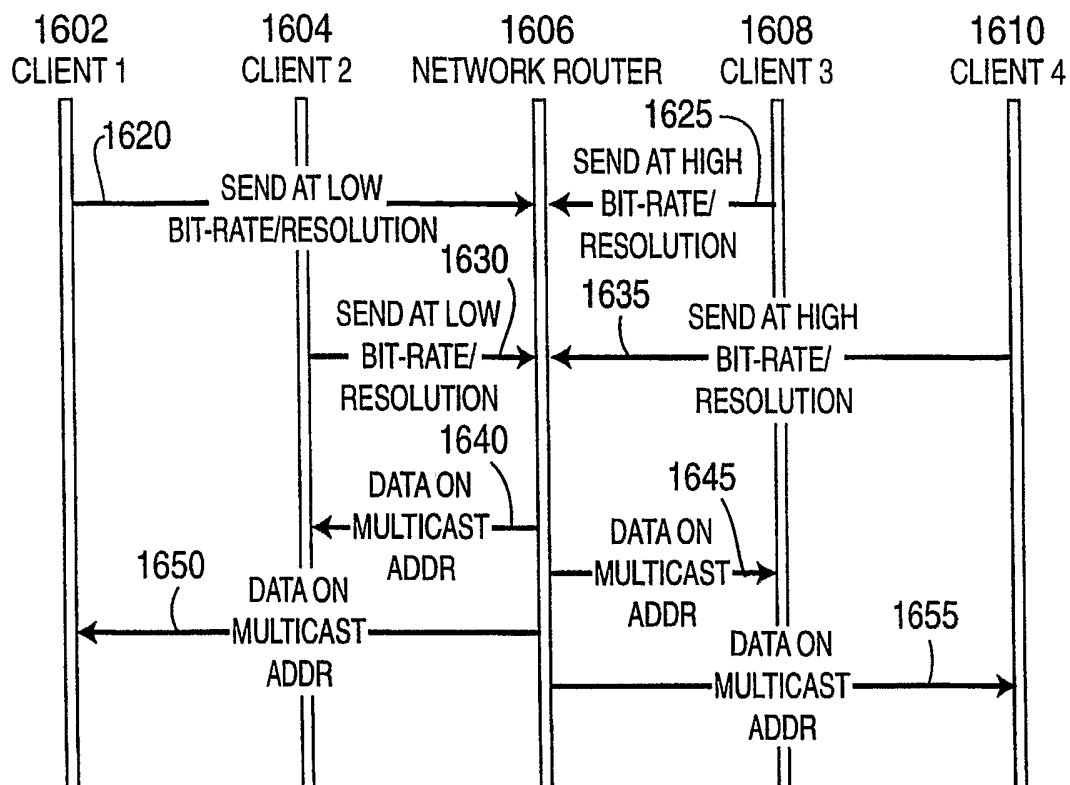
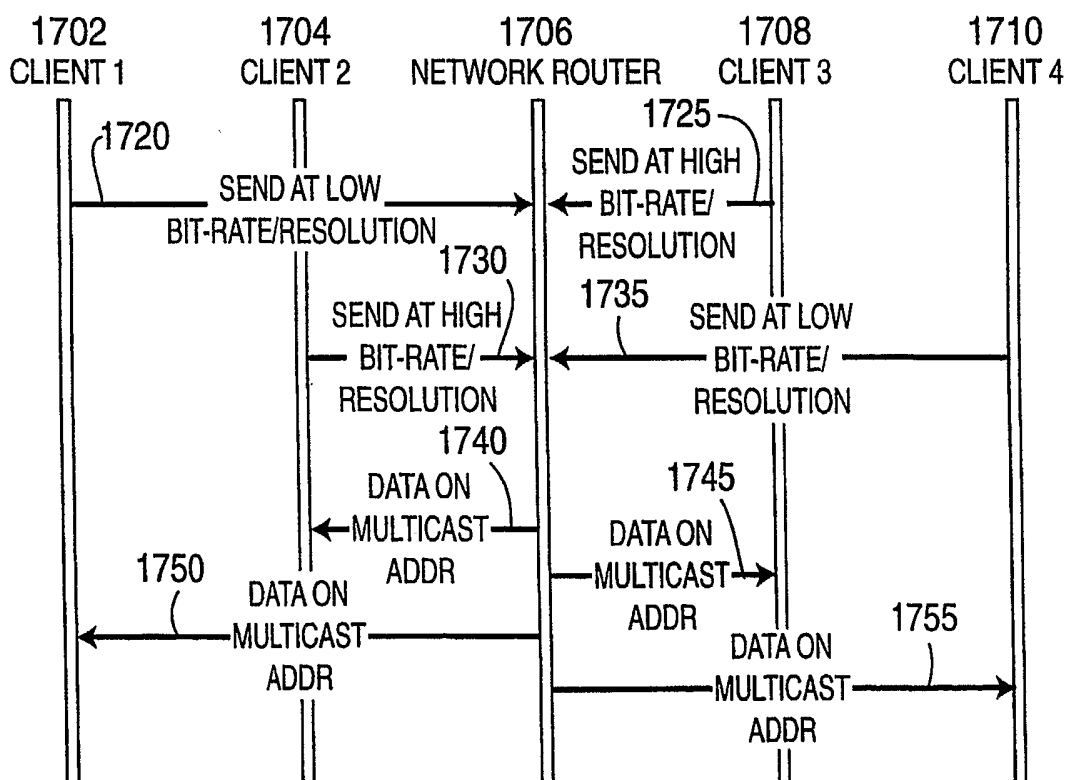
**FIG. 12****FIG. 13****FIG. 14**

12/21

**FIG. 15**



13/21

**FIG. 16****FIG. 17**

14/21

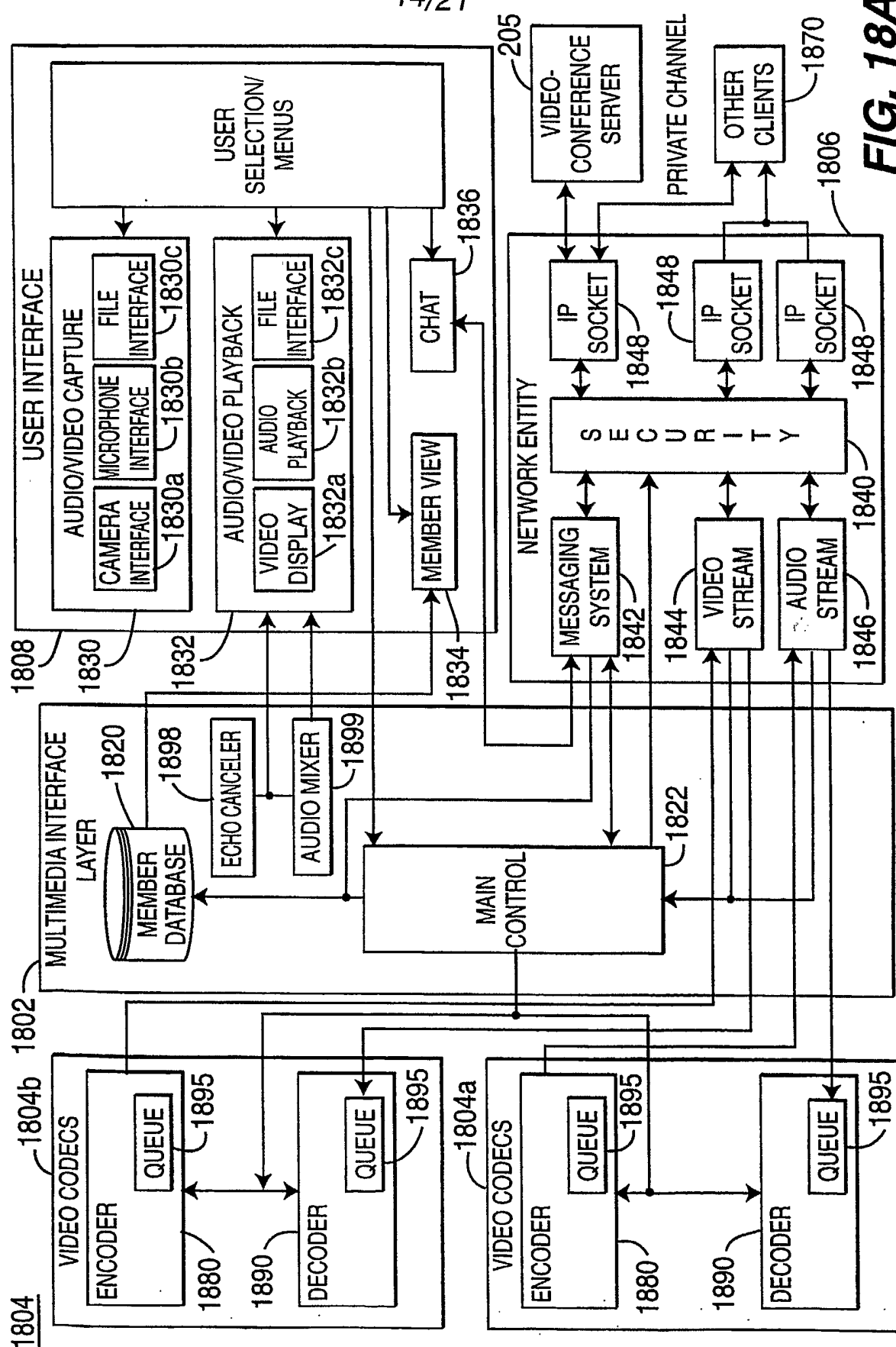
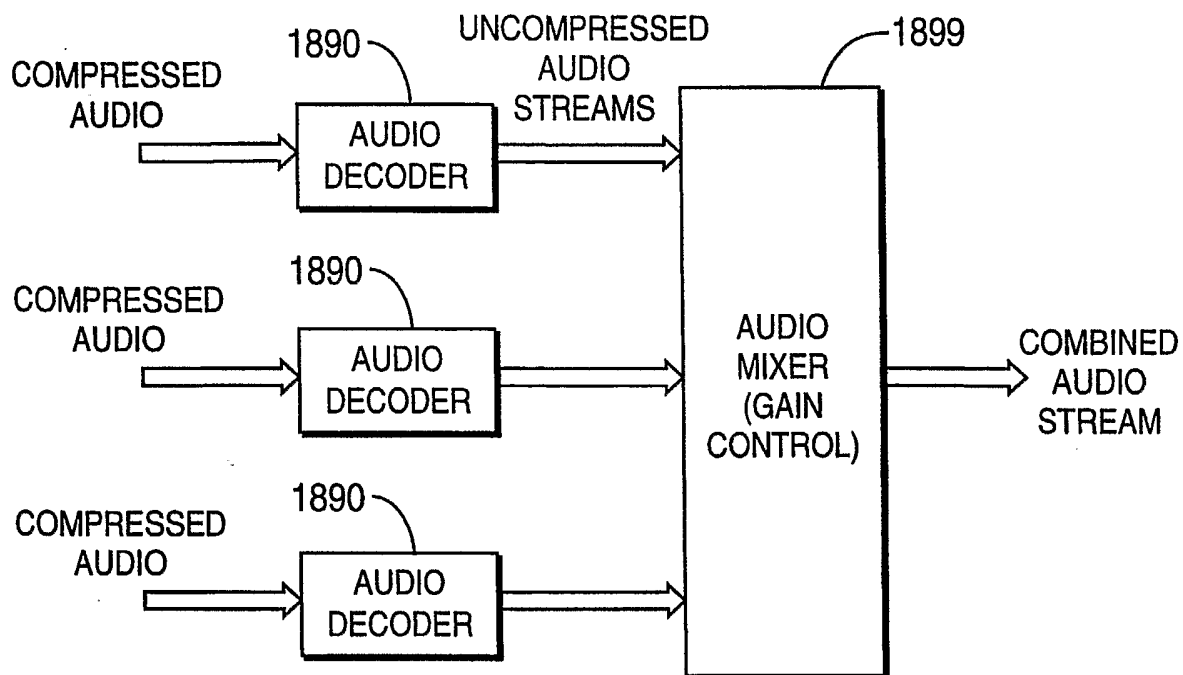
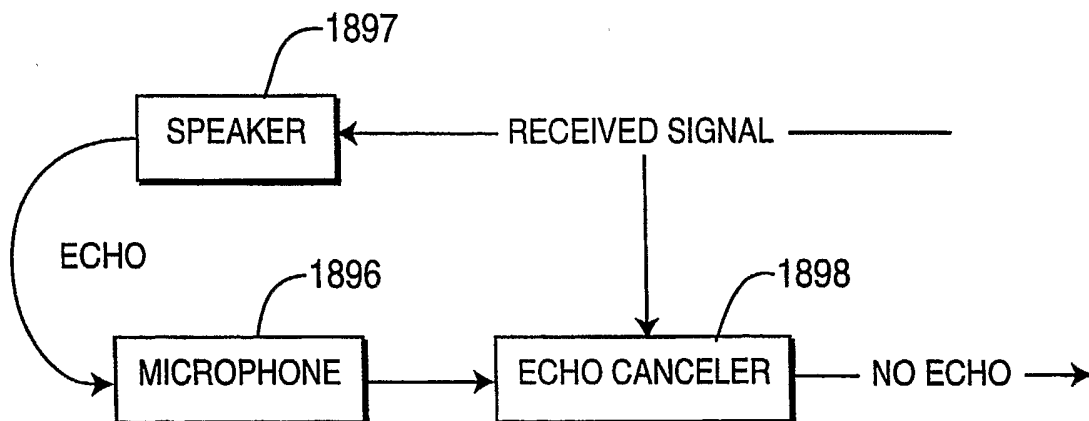


FIG. 18A

15/21

**FIG. 18B****FIG. 18C**

16/21

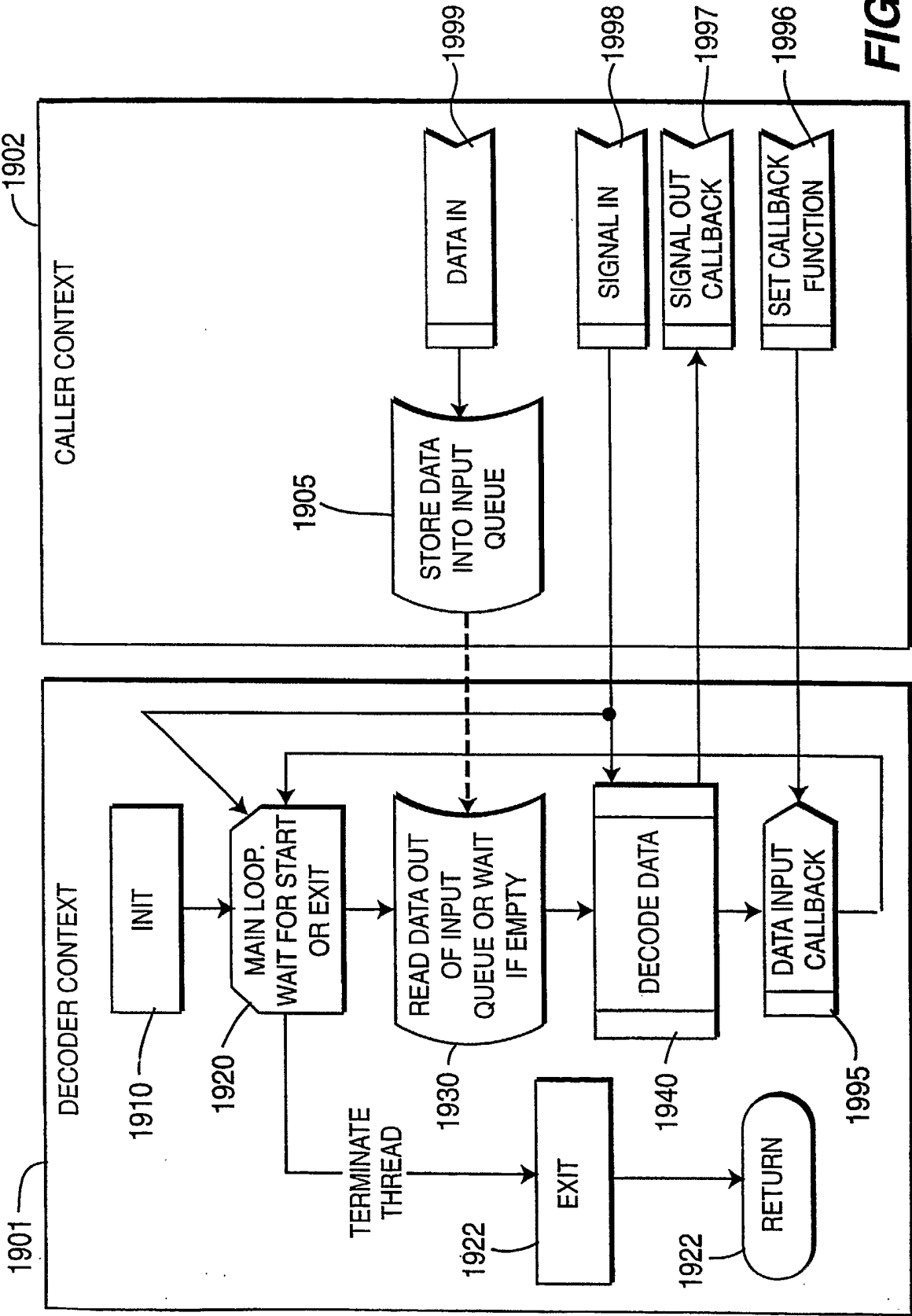


FIG. 19

17/21

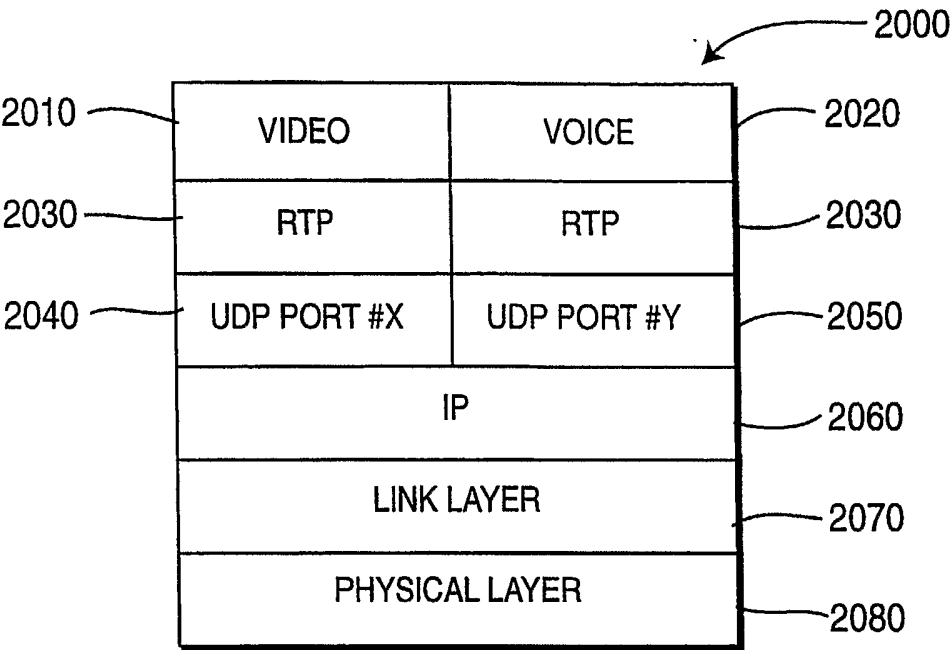


FIG. 20

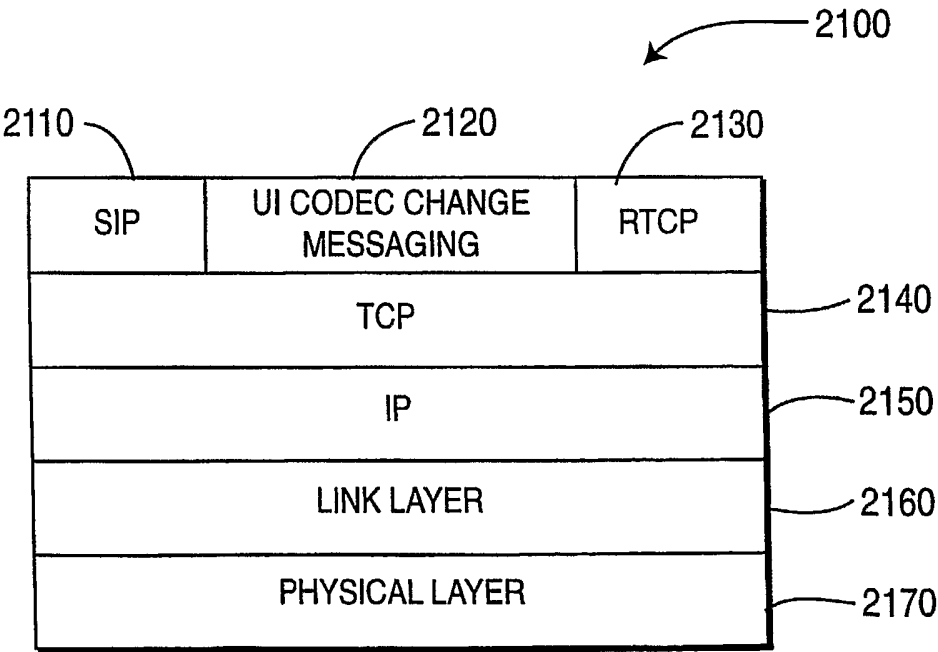
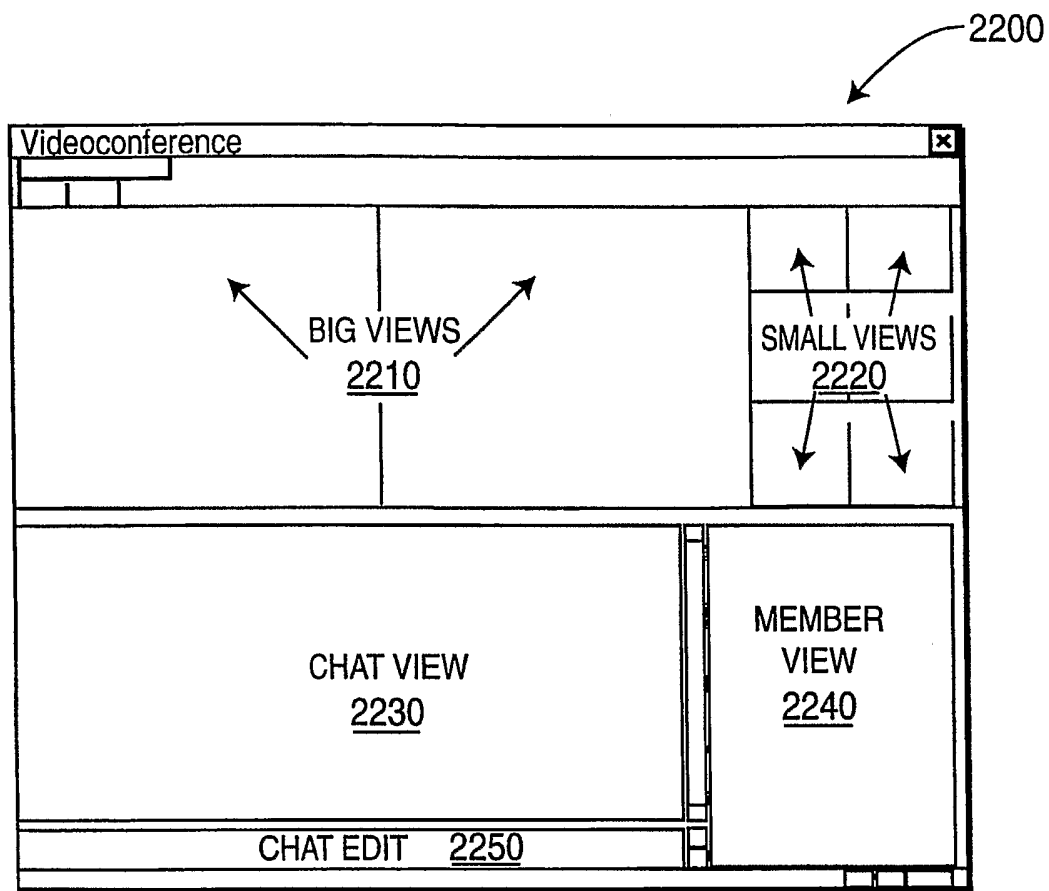
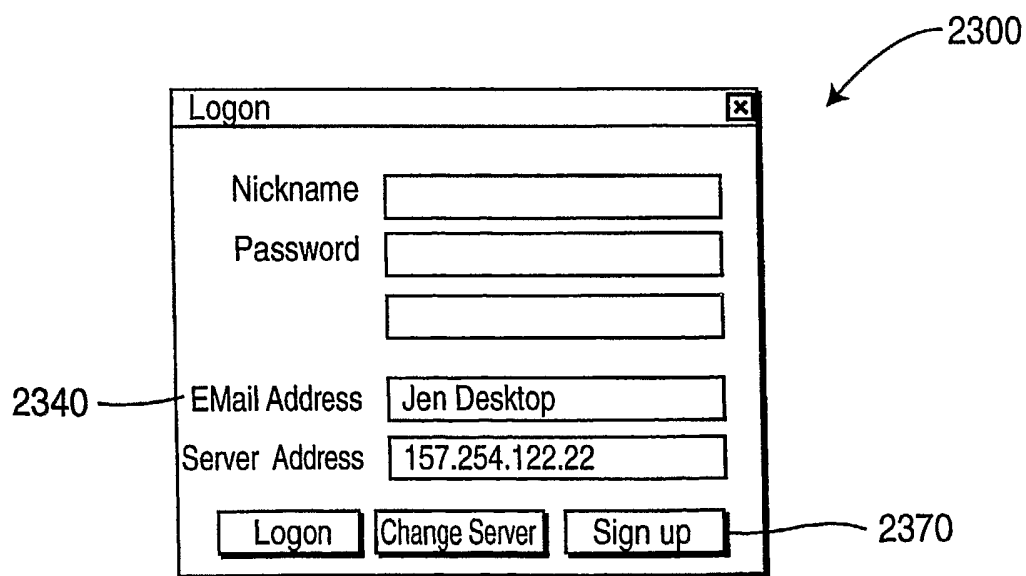
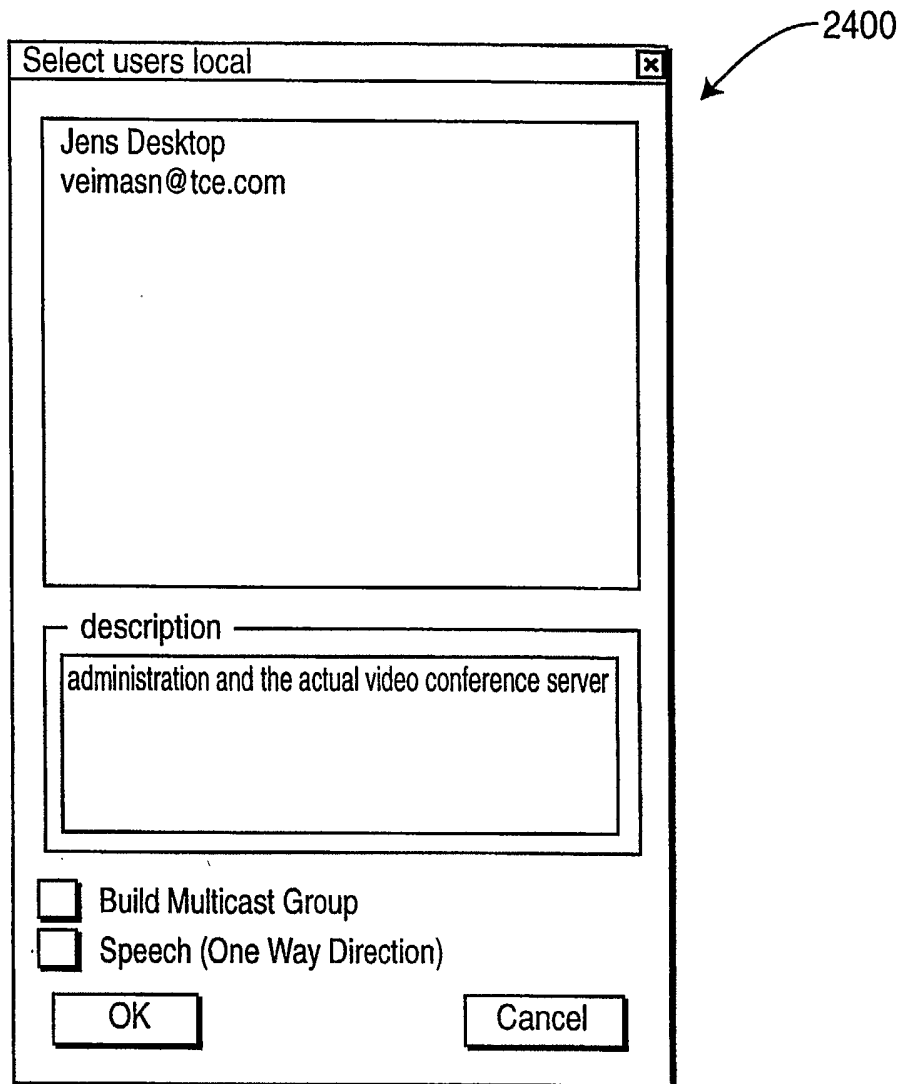
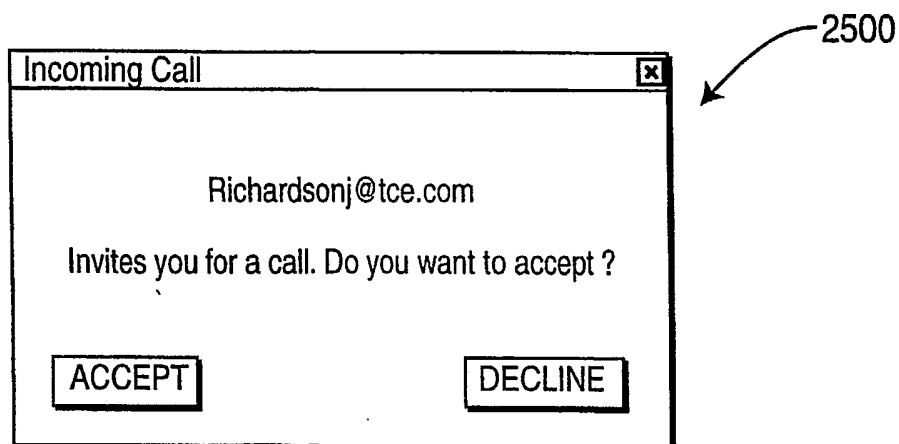


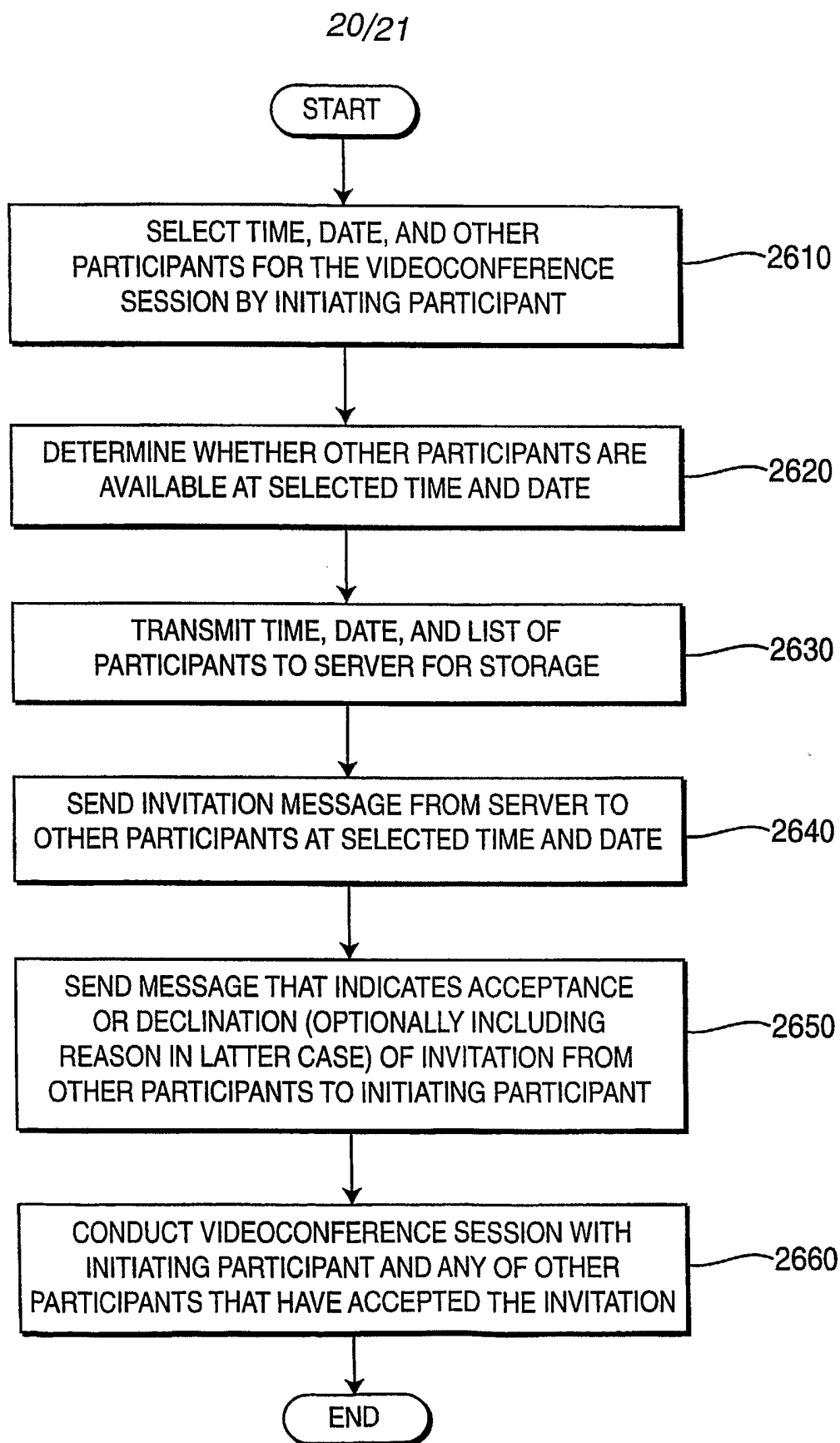
FIG. 21

18/21

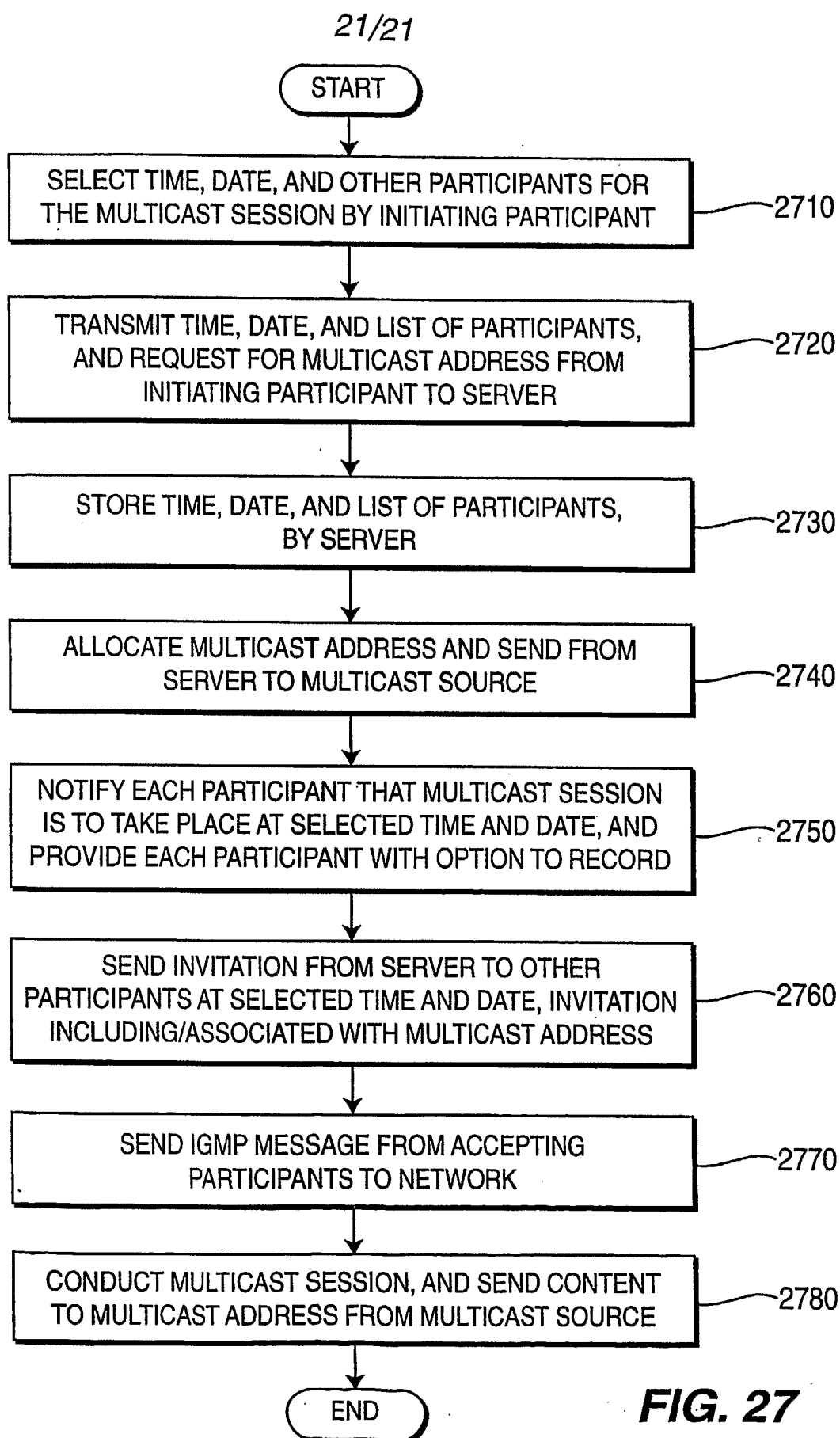
**FIG. 22****FIG. 23**

19/21

**FIG. 24****FIG. 25**

**FIG. 26**





# INTERNATIONAL SEARCH REPORT

International application No.

PCT/US02/36773

## A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) : G 06 F 3/00

US CL : 345/753, 719, 733, 751, 755, 758, 759; 709/201, 204, 205

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 345/753, 719, 733, 751, 755, 758, 759; 709/201, 204, 205

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)  
videoconference, time

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5,907,324 (LARSON et al) 25 May 1999 (25.05.1999), column 1, lines 65-67, column 6, lines 42-45, column 7, lines 4-23, column 7, lines 58-64, column 9, lines 53-57, column 10, lines 27-31.	1-3, 8-12, 14-18, 24-27, 29-31, 34-36
Y		4-8, 13, 19-23, 28, 32-33, 37
A	US 6,219,044 B1 (ANSBERRY et al) 17 April 2001 (17.04.2001)	1-37
A	US 5,949,414 (NAMIKATA et al) 07 September 1999 (07.09.1999)	1-37
Y	US 6,072,463 (GLASER) 06 June 2000 (06.06.2000), column 6, lines 39-51, column 8, lines 49-65.	4-8, 13, 19-23, 28, 32-33, 37

☐ Further documents are listed in the continuation of Box C.

☐ See patent family annex.

* Special categories of cited documents:	
"A" document defining the general state of the art which is not considered to be of particular relevance	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"E" earlier application or patent published on or after the international filing date	"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"O" document referring to an oral disclosure, use, exhibition or other means	"&" document member of the same patent family
"P" document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search

10 February 2003 (10.02.2003)

Date of mailing of the international search report

24 MAR 2003

Name and mailing address of the ISA/US

Commissioner of Patents and Trademarks  
Box PCT  
Washington, D.C. 20231

Facsimile No. (703)305-3230

Authorized officer

John Cabeca / KV

Telephone No. 703-308- 3116

*Robert Harrod*