UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

Facebook, Inc.
Petitioner

v.

Windy City Innovations, LLC
Patent Owner

U.S. Patent No. 8,407,356

TITLE: REAL TIME COMMUNICATIONS SYSTEM

DECLARATION OF TAL LAVIAN, PH.D.

Table of Contents

Page

Table of Contents
(continued)

Page

Table of Contents
(continued)

Page

I, Tal Lavian, Ph.D., declare as follows:

I. INTRODUCTION AND QUALIFICATIONS

A. Summary of My Opinions

1. U.S. Patent No. 8,407,356 purports to describe a computerized technique for facilitating real-time communication between individuals using computers connected via the Internet. As I will explain below, the challenged claims do not recite any feature that would have been regarded as novel or non-obvious to a person of ordinary skill in the art. By April 1996 (the earliest priority date of the '356 patent), real-time collaboration over computer networks was well-known, including video/audio conferencing, whiteboarding, and messaging. One of these references, U.S. Patent No. 6,608,636 to Robert D. Roseman, was filed more than four years before the earliest priority date for the '356 patent. Roseman discloses a networked "virtual conferencing" system that discloses all of the supposedly inventive features of the '356 patent. As I will explain below, all of the challenged claims would have been obvious based on the prior art.

B. Qualifications and Experience

2. I have more than 25 years of experience in the networking, telecommunications, Internet, and software fields. I received a Ph.D. in Computer Science from the University of California at Berkeley in 2006 and obtained a Master's of Science ("M.Sc.") degree in Electrical Engineering from Tel Aviv

University, Israel, in 1996. In 1987, I obtained a Bachelor of Science ("B.Sc.") in Mathematics and Computer Science, also from Tel Aviv University.

3. I am currently employed by the University of California at Berkeley and was appointed as a lecturer and Industry Fellow in the Center of Entrepreneurship and Technology ("CET") as part of UC Berkeley College of Engineering. I have been with the University of California at Berkeley since 2000 where I served as Berkeley Industry Fellow, Lecturer, Visiting Scientist, Ph.D. Candidate, and Nortel's Scientist Liaison, where some positions and projects were done concurrently, others sequentially.

4. I have more than 25 years of experience as a scientist, educator and technologist, and much of my experience relates to computer networking technologies. For eleven years from 1996 to 2007, I worked for Bay Networks and Nortel Networks. Bay Networks was in the business of making and selling computer network hardware and software. Nortel Networks acquired Bay Networks in 1998, and I continued to work at Nortel after the acquisition. Throughout my tenure at Bay and Nortel, I held positions including Principal Scientist, Principal Architect, Principal Engineer, Senior Software Engineer, and led the development and research involving a number of networking technologies. I led the efforts of Java technologies at Bay Networks and Nortel Networks. In

addition, during 1999-2001, I served as the President of the Silicon Valley Java User Group with over 800 active members from many companies in the Silicon Valley.

5. Prior to that, from 1994 to 1995, I worked as a software engineer and team leader for Aptel Communications, designing and developing mobile wireless devices and network software products. From 1990 to 1993, I worked as a software engineer and team leader at Scitex Ltd., where I developed system and network communications tools (mostly in C and C++).

6. I have extensive experience in communications technologies including routing and switching architectures and protocols, including Multi-Protocol Label Switching Networks, Layer 2 and Layer 3 Virtual Private Networks, and Pseudowire technologies. Much of my work for Nortel Networks (mentioned above) involved the research and development of these technologies. For example, I wrote software for Bay Networks and Nortel Networks switches and routers, developed network technologies for the Accelar 8600 family of switches and routers, the OPTera 3500 SONET switches, the OPTera 5000 DWDM family, and the Alteon L4-7 switching product family. I wrote software for Java based device management including software interface to the device

management and network management for the Accelar routing switch family network management system.

7. I am named as a co-inventor on more than 80 issued patents and I co-authored more than 25 scientific publications, journal articles, and peer-reviewed papers. Furthermore, I am a Senior Member of the Institute of Electrical and Electronics Engineers ("IEEE").

8. I currently serve as a Principal Scientist at my company Telecomm Net Consulting Inc., where I develop network communication technologies and provide research and consulting in advanced technologies, mainly in computer networking and Internet technologies. In addition, I serve as a Co-Founder and Chief Technology Officer (CTO) of VisuMenu, Inc., where I design and develop architecture of visual IVR technologies for smartphones and wireless mobile devices in the area of network communications.

9. Additional details of my background are set forth in my curriculum vitae, attached as **Exhibit A** to this Declaration, which provides a more complete description of my educational background and work experience. I am being compensated for the time I have spent on this matter at the rate of $400 per hour. My compensation does not depend in any way upon the outcome of this

proceeding. I hold no interest in the Petitioner (Facebook, Inc.) or the patent owner (Windy City Innovations, LLC).

C. Materials Considered

10. The analysis that I provide in this Declaration is based on my education and experience in the field of computer systems, as well as the documents I have considered including U.S. Patent No. 8,407,356 ("'356 patent") **[Ex. 1001]**, which states on its face that it issued from an application filed on August 9, 2007, which in turn claims priority to back to an earlier application filed on April 1, 1996. For purposes of this Declaration, I have assumed April 1996 as the relevant priority date.

11. I reviewed various documents dated prior to April 1996 describing the state of the art at the time of the alleged invention of the '356 patent. As explained below, some of these documents are relied upon as actually disclosing the limitations of the '356 patent, while others are being relied upon primarily for background purposes. The prior art documents that I rely upon in this Declaration as actually disclosing the limitations of the claims are:

Exhibit No.	Title of Document
1003	U.S. Patent No. 6,608,636 to Robert D. Roseman
1004	EP 0621532 A1 to Eugene Rissanen, published on April 13, 1994
1005	Ronald J. Vetter, *Videoconferencing on the Internet*, Computer, IEEE Computer Society, Vol. 28, No. 1, at pp.77-79 (Jan. 1995)

Exhibit No.	Title of Document
1006	Excerpts from Mary Ann Pike et al., *Using Mosaic* (1994)
1007	James Gosling, *Java Intermediate Bytecodes*, ACM SIGPLAN Workshop on Intermediate Representations (Jan. 1995)

This Declaration also cites the following additional prior art documents for purposes of describing the relevant technology, including the relevant state of the art at the time of the alleged invention of the ’356 patent:

Exhibit No.	Title of Document
1008	Excerpts from *IEEE Std 100-1996, The IEEE Standard Dictionary of Electrical and Electronics Terms* (6th ed. 1997)
1009	Excerpts from *Microsoft Press Computer Dictionary* (2d ed. 1994)
1010	Declaration of Lee A. Hollaar, Ph.D., dated March 26, 2012 (from the file history of the ’356 patent)
1011	Tim Berners-Lee et al., Request for Comments (RFC) 1738, *Uniform Resource Locators (URL)*, Dec. 1994
1012	James Coates, *A Mailbox in Cyberspace Brings World to Your PC*, Chicago Tribune, Mar. 1995

II. PERSON OF ORDINARY SKILL IN THE ART

12. I understand that an assessment of claims of the ’356 patent should be undertaken from the perspective of a person of ordinary skill in the art as of the earliest claimed priority date, which I understand is April 1996.

13. In my opinion, a person of ordinary skill in the art as of April 1996 would possess at least a bachelor’s degree in electrical engineering or computer science (or equivalent degree or experience) with practical experience or

coursework in the design or development of systems for network-based communication between computer systems. This could have included, for example, experience implementing systems for communicating over Local Area Networks (LANs) and Wide Area Networks (WANs), such as the Internet.

14. Although my qualifications and experience exceed those of the hypothetical person having ordinary skill in the art defined above, my analysis and opinions regarding the ’356 patent have been based on the perspective of a person of ordinary skill in the art as of April 1996.

III. CLAIM CONSTRUCTION

15. I have been informed by counsel that invalidity analysis is a two-step process. In the first step, the scope and meaning of a claim is determined by construing the terms of that claim. In the second step, the claim as interpreted is compared to the prior art. Thus, before I address the application of the prior art to the claims of the ’356 patent in **Part IV** below, I provide constructions for certain terms in those claims.

16. I have been informed by counsel that a claim in an unexpired patent subject to *inter partes* review must be given its “broadest reasonable construction in light of the specification of the patent in which it appears,” which is different

from the manner in which the scope of a claim is determined in litigation. I apply the "broadest reasonable construction" standard in my analysis below.

A. "token"

17. Each independent claim recites a database that provides a "repository of tokens" used to perform user authentication. Claim 1, for example, recites the step of "authenticating a first user identity and a second user identity according to permissions retrieved from [a] repository of tokens of the database." The written description accordingly describes a "token" as a piece of information associated with user identity. As explained in the specification:

> With regard to the arbitrating of the controller computer **3** is directed by the controller computer program **2** to use "identity tokens", which are pieces of information associated with user identity. The pieces of information are stored in memory **11** in a control computer database, along with personal information about the user, such as the user's age.

('356, 8:9-14 (underlining added).) The specification goes on to describe several purposes for tokens, including "to control the ability of a user to gain access to other tokens in a token hierarchy arbitration process" ('356, 8:22-23), "to control a user's group priority and moderation privileges, as well as controlling who joins the group, who leaves the group, and the visibility of members in the group" ('356, 8:31-33), and "to permit a user's control of identity, and in priority contests between 2 users, for example, a challenge as to whether a first user can see a second user." ('356, 8:36-38.)

18. Based on the definitional language in the written description, I have construed "**token**" as a "**piece of information associated with user identity**."

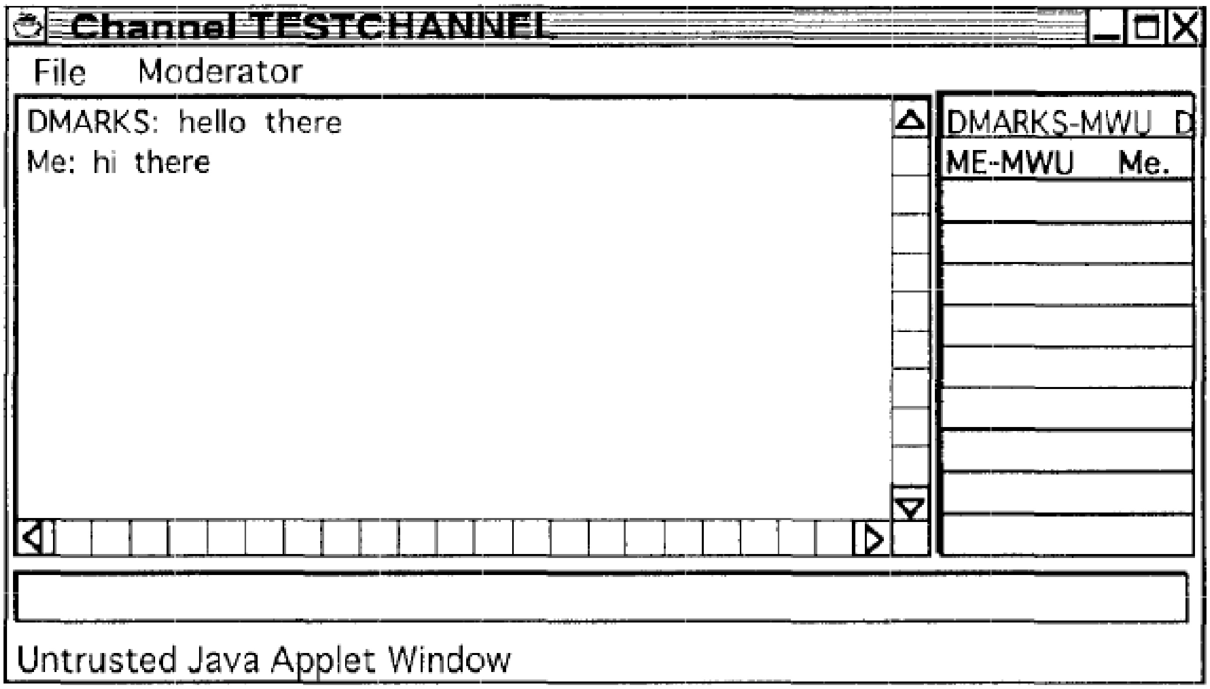
B. "channel"

19. The word "**channel**" in telecommunications generally refers to a transmission medium or connection, but the patent uses the term in a different context. The written description explains that "a group is sometimes known as a channel in multiplexing terminology" ('356, 8:28-29), which is consistent with the use of "channel" throughout the patent. For example, the following passage equates a "channel list" with a "group list":

> Then the <u>channel list</u> area is shown at FIG. 8. The Channel List area is a window which shows a <u>list of all of the groups</u> currently on the server in active communication. Because no one is yet connected in this example, there are no groups currently available on the screen.

('356, 9:24-28 (underlining added).) An exemplary channel is also shown in Figure 14, which shows a screen that many would recognize as functionality similar to a "chat room" from an online service such as America Online:

FIG. 14



('356, Fig. 14; *see also* 3:21-22 ("FIG. 14 is an illustration of a private message window on the new <u>channel screen</u> of the present invention") (underlining added).) The window in Figure 14 has "three regions: the bottom region, where responses are entered; the largest region, where a transcript of the communication is followed; and the rightmost region, which lists the group's current members." ('356, 9:34-37.) In the example shown in Figure 14, DMARKS has typed in a message "hello there," which was sent to the channel members, DMARKS and "Me." ('356, 9:47-51 ("The user DMARKS now types 'hello there' into the response area and presses RETURN (at FIG. **12**). This message is passed to the controller computer **5**, which sends the message to all channel members, i.e., those using participator computers **5**, including DMARKS.") (underlining added).)

20. Based on this description, in my opinion, the term "**channel**" should be construed as a "**group of users that can communicate with each other**."

C. "providing an API on the controller computer, the API multiplexing and demultiplexing API messages by type"

21. This entire phrase warrants an explicit construction because it relies on the terms "API" and "API message" that appear to have special meanings within the '356 patent that differ from their plain and ordinary meaning to a person of ordinary skill in the art. The phrase also uses the terms "multiplexing" and "demultiplexing," which warrant further explanation.

22. To begin with, the term "**multiplexing**" outside the patent ordinarily refers to techniques for combining multiple messages for transmission through a single communications pathway. (*IEEE Std 100-1996, The IEEE Standard Dictionary of Electrical and Electronics Terms* (6th ed. 1997) ("IEEE Dictionary") **[Ex. 1008]**, at p. 673 (defining "multiplex" as "[t]o interleave or simultaneously transmit two or more messages on a signal channel" and defining "multiplexing" as "[t]he combining of two or more signals into a single wave (the multiplex wave) from which the signals can be individually recovered"); *see also Microsoft Press*

Computer Dictionary (2d ed. 1994) **[Ex. 1009]**, at p. 265 (defining "multiplexing" as "[a] technique used in communications and input/output operations for transmitting a number of separate signals simultaneously over a single channel or line.").)

23. The term "**demultiplexing**" to a person of ordinary skill in the art generally refers to the reverse of multiplexing, *i.e.* separating out the individual messages that were incorporated into the combined (multiplexed) signal. (IEEE Dictionary, Ex. 1008, at p. 270 (defining "demultiplexing" as "[t]he separation from a common input into several outputs").) For example, hardware may demultiplex signals from a transmission line based on time or carrier frequency to allow multiple, simultaneous transmissions across a single physical cable.

24. The terms multiplexing and demultiplexing are commonly used to describe the physical transport and processing of digital and analog signals, but the '356 patent uses these terms in a slightly different context. In particular, the Background of the Invention uses the term "multiplexing" to describe messaging technologies such as email, conferencing, and chat messages that may be distributed through a shared network:

> <u>Multiplexing</u> group communications among computers ranges from very simple to very complex communications systems. At a simple level, group communications among computers involves <u>electronic</u>

> mail sent in a one way transmission to all those in a group or subgroup using, say, a local area network. Arbitrating which computers receive electronic mail is a rather well understood undertaking.
>
> On a more complex level, corporations may link remote offices to have a conference by computer. A central computer can control the multiplexing of what appears as an electronic equivalent to a discussion involving many individuals.
>
> Even more complex is linking of computers to communicate in what has become known as a "chat room." Chat room communications can be mere text, such as that offered locally on a file server, or can involve graphics and certain multimedia capability, as exemplified by such Internet service providers as America On Line. Multiplexing in multimedia is more complex for this electronic environment.

('356, 1:34-52 (underlining added).) Each of these examples of "multiplexing" generally refers to the process of transporting an individual message (typically originating from one person) using a shared communications pathway, such as a local area network or the Internet, that also carries other messages. Accordingly, "demultiplexing" would be understood as the reverse of multiplexing, *i.e.* separating an individual message from the combined signal carried by the communications pathway to deliver it to the intended recipient.

25. However, the patent uses the term "**API**" in a manner very different from its ordinary meaning outside the patent. The term "API" outside the patent

generally refers to an "application programming interface" (API), a concept very familiar to persons of ordinary skill in the art. An application programming interface generally provides a way for a set of software services to be accessed by another software system. APIs can exist at many levels of computer software design and usually take the form of a set of defined software routines, procedures, functions or methods that invoke the software services they represent. For example, operating systems such as Microsoft Windows and UNIX provide APIs to allow user applications to interact with the operating system to perform tasks such as creating and opening files, sending and receiving information over a network, receiving user input from a keyboard, mouse or other input device, and performing many other functions.

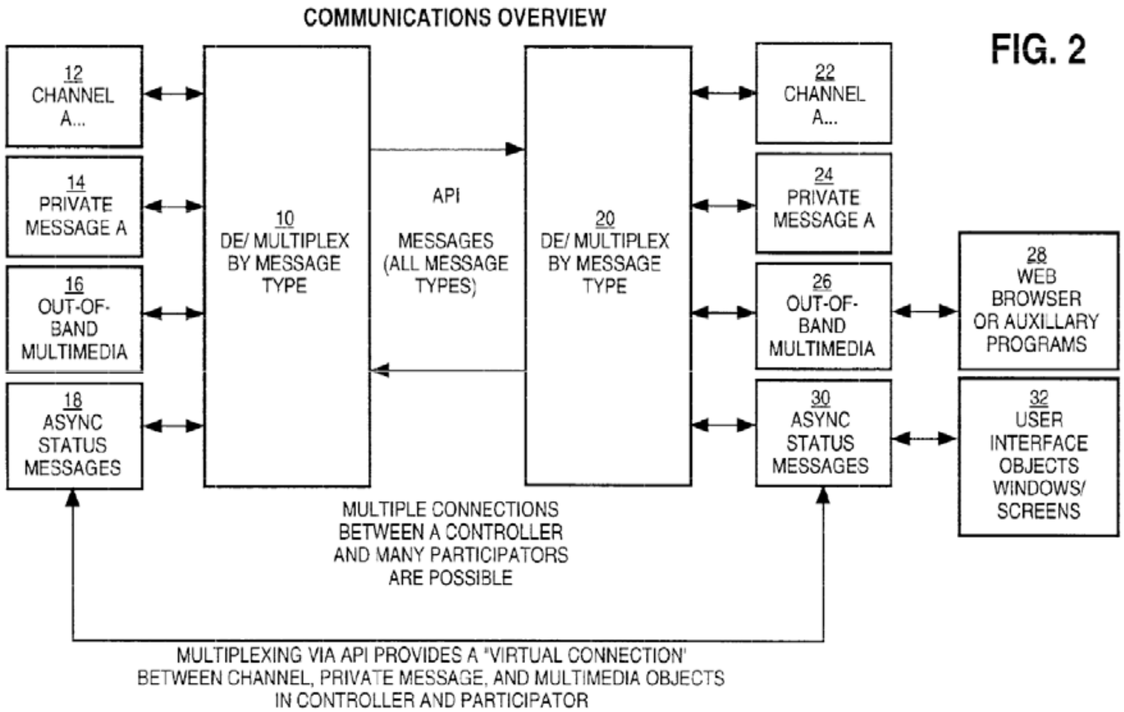
26. During the prosecution of the '356 patent, however, the applicant stated that the claimed "**API**" was **<u>not</u>** an application programming interface. In particular, following a claim rejection based on Brown (U.S. Patent No. 5,941,947), applicant submitted a declaration of Lee A. Hollaar, Ph.D., in an attempt to distinguish Brown that stated:

> With respect to the Marks specification and claims, the Examiner correctly concedes that "Brown does not specifically teach an application programming interface (API) within the reference" at page 7 of the Final Rejection.

> However, the Examiner has misunderstood the Marks specification and claims by assuming that an API means an applications program interface to the operating system (Final Rejection, Page 7 "Horn also teaches that applications access the operating systems functions through an application programming interface (API) ... ").
>
> There is no basis for this misunderstanding because, while the Marks specification does not define API, the Marks specification also never uses the term "application program interface" and only uses "application" once ("JAVA application").
>
> <u>The best illustration of what API means in the context of the Marks specification is in Fig. 2, where API messages are the collection of all message types, between one computer and another, that are multiplexed for transfer and then demultiplexed by message type to rout them to the appropriate place</u>. There is no possible way that Fig. 2 is compatible with the Examiner's interpretation that the API is used to "access the operating systems functions."

(Hollaar Declaration, Ex. 1010, ¶¶ 6-9 (underlining added).)

27. Figure 2, which the applicant claimed is the "best illustration of what API means the context of the Marks specification," shows the label "API" sitting in the middle of the figure between blocks **10** and **20**:



('356, Fig. 2.)

28. The specification indicates that the black boxes in Figure 2 are software processes. Block **10** (to the left of "API") refers to software on the "controller computer" for performing multiplexing and demultiplexing operations on messages. ('356, 5:45-48 ("Beginning <u>with the Controller Computer Software **2**</u>, reference is made to Block **10**, which illustrates demultiplexing and multiplexing operations carried out by message type on API messages of all types.") (underlining added).) The written description uses almost identical language to describe block **20** (to the right of block **10**), except that block **20** refers to software on a "participator computer." ('356, 5:58-61 ("With particular regard to the <u>participator software **4**</u> illustrated in FIG. **2**, Block **20** is illustrative of

demultiplexing and multiplexing operations carried out by message type on API messages of all types.") (underlining added).)

29. The patent does not appear to describe in detail how "demultiplexing and multiplexing operations" are actually carried out. It instead largely repeats the language of the claims by stating: "De/multiplexing via API provides a 'virtual connection' between Channel, Private Message, and Multimedia objects in the controller computer **3** and each participator computer **5**." ('356, 6:3-5.) Based on this description, the term "API" does not describe an application programming interface; it simply describes software functionality that may reside on a controller or participant computer.

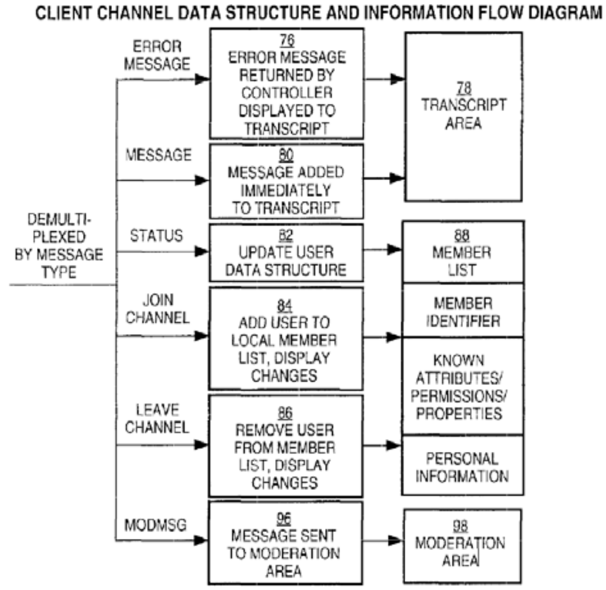
30. With respect to the term "**API messages**," it appears only twice in the written description in stating that blocks **10** and **20** of Figure 2 illustrate "demultiplexing and multiplexing operations carried out by message type on API messages of all types." ('356, 5:57-58 (block 10), 5:60-61 (block 20).) The written description later describes six possible message types, "ERROR MESSAGE, MESSAGE, STATUS, JOINCHANNEL, LEAVECHANNEL, and MODMSG," which are used to mediate communications between the controller and participant computers. ('356, 7:5-8 (capital letters in original); *see also id*., 22:6-8 (Claim 9:

"The method of claim 1, wherein communications among the controller computer and the participator computers are mediated via API messages.").)

31. When an API message is received, it is communicated to the appropriate "Block" for execution based on the message type. A received message of type "**MESSAGE**," for example, represents a message that should be shown on the text display to the user (in the "transcript area" on the screen). ('356, 7:11-12, 7:30-32.) Accordingly, Figure 5 on the right (partial figure) shows messages of type "MESSAGE" being "communicated to Block **78** [sic; **80**] where the message is immediately added to the transcript in transcript area **78**." ('356, 7:11-12.) Figure 5 shows examples of how five other types of messages are routed to the logic blocks 78, 82, 84, 86, and 96. ('356, 7:8-28.)

FIG. 5

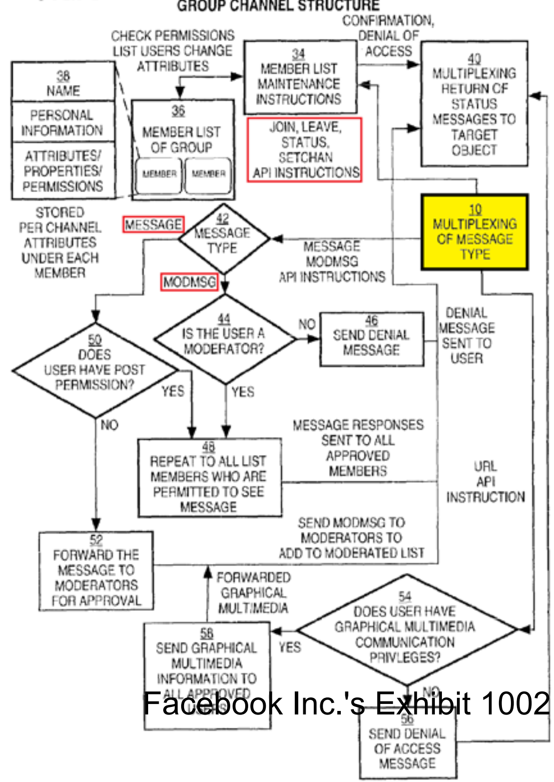
CLIENT CHANNEL DATA STRUCTURE AND INFORMATION FLOW DIAGRAM



32. The patent provides similar disclosures relating to the claimed

FIG. 3

DATA AND COMMUNICATIONS DEPENDENCY DIAGRAM CONTROLLER GROUP CHANNEL STRUCTURE



"**multiplexing**" operation. As shown in Figure 3 (shown at the right), block **10** shows "MULTIPLEXING OF MESSAGE TYPE" (in highlighting), which evaluates the type of message and then routes the message to the appropriate software functionality. For example, for MESSAGE or MODMSG types, the logic moves to block **42**, and branches further to block **50** for MESSAGE and block **44** for MODMSG. ('356, 6:25-40.) For other types of messages, the logic branches to block **34**. ('356, 6:12-15.)

33. Based on my analysis, therefore, I believe the correct broadest reasonable construction of the phrase, "providing an API on the controller computer, the API multiplexing and demultiplexing API messages by type," should be "**providing software functionality on the controller computer for sending and receiving messages of different types and communicating each message to software functionality based on the message type.**"

D. "pointer"

34. The term "pointer" appears in dependent claims 2, 7, 20, and 26. "Pointers" are well-known in computer science and exist at all levels of computer system design – from the lower microprocessor levels to the higher levels where application programs execute. To persons of ordinary skill in the art, a "pointer" is simply a piece of information that "points to," or references, other information.

35. The written description provides only the following mention of pointers, which identifies a Uniform Resource Locator as an example of a pointer:

> The present invention comprehends communicating all electrically communicable multimedia information as Message **8**, by such means as pointers, for example, URLs. URLs can point to pre-stored audio and video communications, which the Controller Computer **3** can fetch and communicate to the Participator Computers **5**.

('356, Ex. 1001, 5:38-43.) Based on this description, the term "pointer" should be construed as a "**piece of information that points to or references other information**."

IV. APPLICATION OF THE PRIOR ART TO THE CLAIMS OF THE '356 PATENT

36. I have reviewed and analyzed the prior art references and materials listed in **Part I.B** above. In my opinion, each and every limitation of claims 1-9, 12, 14-28, 31, and 33-37 is disclosed by the following references (1) U.S. Patent No. 6,608,636 to Robert D. Roseman ("Roseman") **[Ex. 1003]**; (2) EP 0621532 A1 to Eugene Rissanen, published on April 13, 1994 ("Rissanen") **[Ex. 1004]**; (3) Ronald J. Vetter, *Videoconferencing on the Internet*, IEEE Computer, Vol. 28, No. 1, at pp. 77-79 (Jan. 1995) ("Vetter") **[Ex. 1005]**; (4) Mary Ann Pike et al., *Using Mosaic* (1994) ("Pike") **[Ex. 1006]**; and (5) James Gosling, *Java Intermediate*

Bytecodes, ACM SIGPLAN Workshop on Intermediate Representations (Jan. 1995) ("Gosling").

37. As shown below, each limitation of the challenged claims is disclosed by the prior art discussed in this Declaration. In particular: (1) method claims 1-5, 8-9, 12 and 14-16 are obvious over Roseman in view of Rissanen and Vetter; (2) dependent method claims 6-7 and 17 are obvious over Roseman in view of Rissanen and Vetter, in further view of Pike (URLs); and (3) dependent method claim 18 is obvious over Roseman in view of Rissanen and Vetter, in further view of Gosling. Most of my analysis will focus on independent claim 1, to which I turn first below. The remaining claims discussed in this declaration, *i.e.* claims 19-28, 31, and 33-37, are "apparatus" claims that recite substantially the same requirements as the method claims mentioned above, so in the interests of brevity, I will refer back to my analysis of the method claims where appropriate.

38. I am informed that Roseman qualifies as prior art because it issued from an application filed on May 13, 1992, which is several years before the earliest application to which the '356 patent can claim priority (April 1, 1996). I am also informed that Vetter, Rissanen, Pike and Gosling qualify as prior art because they were published more than one year before April 1, 1996.

39. Before explaining how the prior art applies to the claims, I will briefly summarize each piece of art and provide an overview of how I have applied it.

A. Brief Description and Summary of the Prior Art

1. Roseman [Ex. 1003]

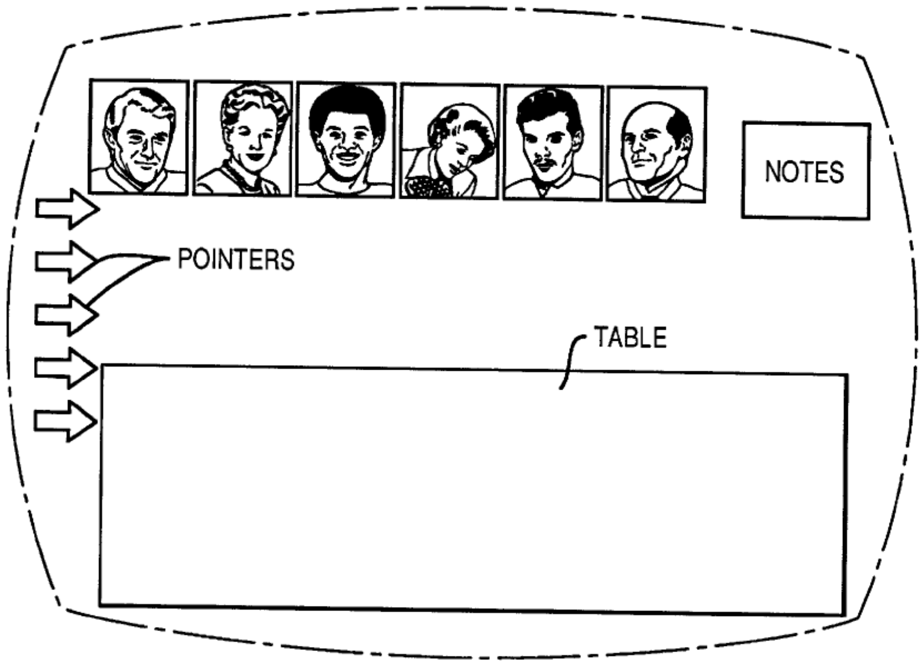
40. Roseman, entitled "Server Based Virtual Conferencing," discloses a system for creating a virtual conference room that allows participants to collaborate in real time over a computer network. My Declaration cites Roseman for the majority of the limitations in the challenged claims, and relies on the other references (Vetter, Rissanen, Pike and Gosling) only for a few limitations to the extent not disclosed in Roseman.

41. The virtual conferencing system in Roseman "allows multiple persons, at different locations, to hold a conference, by providing many of the conveniences which the participants would have if present together in the same physical room." (Roseman, 1:19-23.) Roseman describes "a virtual conferencing system which allows multiple persons to view, and also manipulate, a common video display, which is simultaneously displayed at their different locations." (Roseman, 1:28-31.) Each conference participant has his or her own "local computer." (Roseman, 1:34-35; *see also id.*, 2:64-65.) The local computers "have associated video cameras, speaker-type telephones, and pointing devices (such as

'mouses'). When a conference is established, the local computers become connected to a host computer, via commercially available Local Area Networks (LANs) and Wide Area Networks (WANs)." (Roseman, 1:36-41; *see also id*., 3:14-19.)

42. A user in Roseman creates a virtual conference room by clicking an appropriate icon, identifying the participants of the conference room and providing other information such as the rules that govern the conference. (Roseman, 3:22-56.) Once the parameters of the conference are established, the host computer "creates the conference room. The host does this by creating a common image, such as that shown in FIG. 9. The common image includes a picture of each invitee, a 'table,' and the room decor." (Roseman, 7:30-34.) An example of the Roseman virtual conference room is shown in Figure 9 below:

FIG. 9



(Roseman, Fig. 9.)

43. Roseman explains that when a meeting participant enters a virtual conference room with other participants, "the data connection is made. Audio and video connections are made if supported by the user, the room and the other users. A small picture of each user is displayed in the meeting room to indicate presence." (Roseman, 11:11-14.) Once inside the conference room, "[o]bjects (documents) can be shared in the conference room by placing them on the table. This might be done by dragging an icon . . . onto the table." (Roseman, 11:18-22.) Additionally, the user can click on the picture of another participant to engage in a private voice conversation, or drag a textual note onto the picture of another

participant to send a private text message. (Roseman, 9:16-31.) Other communication features are described in my discussion of the claims below.

44. Roseman also discloses a security mechanism in which users must be invited and have an appropriate "**key**" to enter the conference room. (Roseman, *e.g.*, 9:34-55, 10:61-64 ("To open a door with a key, the user drops the key onto the door lock. If the key is valid and the user has the authority to use the key, the door opens and the user is admitted to the room.").) "The meeting room 'knows' about each key and its invitation level. Persons with improper keys are not admitted to the room." (Roseman, 9:49-51.) These conference room "keys," as I will explain below, correspond to the "tokens" recited in the independent claims.

45. Roseman also discloses a database that stores the keys for the conference room. In particular, Roseman explains that "[t]he meeting room 'knows' about each key and its invitation level." (Roseman, 9:49-50.) The "meeting room," in turn, is stored on the host computer. (Roseman, 9:61-63 ("Meeting Facilitator (or Requestor) creates [sic] meeting room <u>on a host computer</u> which is accessible to all Invitees.") (underlining added); *see also id.*, 12:16-18 ("The conference room itself is actually a combination of stored data and computer programs.").) More details about Roseman are set forth below.

2. Rissanen [Ex. 1004]

46. Each independent claim of the ’356 patent recites “**a database which serves as a repository of tokens for other programs to access**.” As I noted above, the “keys” in Roseman disclose the claimed “tokens,” and those keys are stored on the central host computer. But Roseman does not use the word “database” to describe the storage of keys by the host. In the event it is argued that Roseman fails to disclose a “database” that stores the keys, as recited by the claims, this requirement would have been trivially obvious over Rissanen.

47. Rissanen, entitled “Password Verification System,” discloses a technique for user authentication using passwords stored in a database. My Declaration relies on Rissanen as an alternative basis to teach “**a database which serves as a repository of tokens for other programs to access**,” in the event it is argued that Roseman alone does not disclose this limitation. Rissanen discloses storing user passwords in a database, and subsequently using those stored passwords to verify user identity when users subsequently attempt to log-on. (Rissanen, Ex. 1004, 1:21-28 (“Some business computer systems are arranged to initially record and store passwords assigned to users. In response to a prompt by the system for the user’s password, the user enters the password onto a keyboard and the system compares the keyboard entered password with the stored passwords

and enables the user to access the system when the entered password matches the previously stored password.") (underlining added).) Rissanen also discloses that user login and password information may be stored in a database. (Rissanen, 2:22-29 ("In accordance with an embodiment of the preferred invention, a computer controlled database is linked to a telecommunication network with which users are provided password controlled access. Users are initially entered into a password database stored in the computer system by assigning each user an account code and a password, such as consisting of a number of numerical digits.") (underlining added).) Although Rissanen also describes a technique for using spoken voice passwords, I have cited it for basic teachings relating to database storage of passwords of any form.

48. As I will explain in detail below, the user and password information in the database in Rissanen is analogous to the conference room "key" in Roseman. It would have been obvious to a person of ordinary skill in the art to combine Roseman and Rissanen to produce the virtual conferencing system of Roseman in which the conference room keys are stored in a database serving as a repository of tokens (keys) for other programs to access, as taught in Rissanen.

3. Vetter [Ex. 1005]

49. Certain independent claims of the ’356 patent require that information be transmitted “**via the Internet**.” Roseman discloses using “commercially available” Wide Area Networks (WANs) to communicate with participator computers, but does not specifically disclose that those WANs include the Internet. (Roseman, Ex. 1003, 1:37-41; *see also id.*, 3:14-19.)

50. Vetter, entitled “Videoconferencing on the Internet,” discloses software tools for enabling videoconferencing over the Internet. I have cited Vetter for the proposition that using the Internet to send information to meeting participant computers in Roseman would have been obvious to a person of ordinary skill in the art. Vetter discloses that “[v]ideoconferences are becoming increasingly frequent on the Internet,” and that “[r]eadily available software tools enable real-time audio and video channels as well as shared whiteboards that allow groups to collaborate on distributed group work more easily than ever . . .” (Vetter, Ex. 1005, at p. 77.)

51. As I will explain below, the recitation of the “Internet” does not provide any non-obvious distinction over Roseman. Vetter confirms adding transmission over the Internet to Roseman would have been obvious to a person of

ordinary skill in the art, and a person of ordinary skill in the art would have had ample motivations to combine Roseman with Vetter.

4. Pike [Ex. 1006]

52. Pike, entitled *Using Mosaic*, is a book describing NCSA Mosaic, one of the early browsers for accessing the World Wide Web. (Pike, Ex. 1006, at p. 1-2.) I have cited Pike in connection with dependent claims 6, 7, 17, 26, and 36, which recite that the information communicated between computers can include a "pointer" that allows the content to be produced on demand, or recite that communicated content may be invoked with a Uniform Resource Locator (URL).

53. As explained below, Roseman discloses a pointer in the form of a clickable icon that, when clicked by a meeting participant, presents a document, message or other content to the user. (Roseman, Ex. 1003, *e.g.*, 14:53-57, 14:59-62 (icon representing document placed on table), 9:28-31 (icon representing private message).) Roseman does not disclose the mechanics of how the pointer works, and does not expressly disclose a URL.

54. The specification of the '356 patent does not define the term "pointer" but identifies a URL as an example. ('356, 5:38-43.) For convenience and consistency, I have cited Pike for the dependent claims reciting a "URL" (claims 17 and 36), and in the event it is argued that "pointer" requires something akin to

an Internet URL, for the dependent claims that require a "pointer" (claims 6, 7, and 26). As I will explain below, the "pointer" and "URL" limitations in claims 6, 7, 17, 26, and 36 would have been obvious in view Pike **[Ex. 1006]**.

55. URLs are used today to identify hundreds of millions of resources located on the Internet, and were clearly not an invention of the '356 patent. Pike, which was published in 1994, provides an introductory section describing basic Internet concepts such as URLs. (Pike, at pp. 38-39.) Pike explains that "[a] *URL* is a complete description of an item, including the location of the item that you want to retrieve." (*Id.* at 38 (italics in original).) "The location of the item can range from a file on your local disk to a file on an Internet site halfway around the world." (*Id.*) Pike further explains that a URL can identify any resource on the Internet, and "is not limited to describing the location of WWW [World Wide Web] files." (*Id.*) Pike goes onto describe the familiar URL syntax and how URLs identify documents that can be retrieved from other computers. (*Id.* at pp. 38-39.) As I will demonstrate below, it would have been obvious to a person of ordinary skill in the art to adapt known URL techniques to Roseman.

5. Gosling [Ex. 1007]

56. Gosling, entitled *Java Intermediate Bytecodes*, is a paper describing aspects of the Java programming language. I have cited Gosling solely in

connection with dependent claims 18 and 25, which recite that "the controller software comprises a JAVA™ application." Gosling confirms that Java was a known computer programming language that predated the '356 patent. (Gosling, Ex. 1007, at p. 111 ("Java is a programming language loosely related to C++.").) Gosling also explains that one of the benefits of Java was that compiled programs in Java were "portable" and could execute on any kind of CPU. (*Id.* at p. 115.) As I will demonstrate below, it would have been obvious to a person of ordinary skill in the art to create the software of Roseman in the form of a Java application.

E. Each Limitation of Claims 1-9, 12, 14-28, 31, and 33-37 Is Disclosed by the Prior Art

1. Claim 1

a. "A method of communicating content among users using [sic] of a computer system including a controller computer and a database which serves as a repository of tokens for other programs to access, thereby affording information to each of a plurality of participator computers which are otherwise independent of each other" (Claim 1, Preamble)

57. Roseman discloses each aspect of the preamble of claim 1. Because of the length of the preamble, I will break up the claim language into pieces to ensure that I cover all of the limitations the language potentially imposes.[1]

58. First, Roseman discloses "[**a**] **method of communicating content among users using** [sic] **of a computer system**," as recited in the first part of the preamble. Roseman discloses a virtual conferencing system in which users (*e.g.*, conference participants) share content over a network. For example:

> The parties send the information which they want displayed, such as drawings, to the host computer. The host computer generates a common video screen, which it distributes to the parties: they see the

[1] I am informed by counsel that a claim preamble does not always impose a limitation on the claim. It is unnecessary for me to determine whether the preamble is limiting because the prior art nevertheless discloses it.

> drawings at their own local computers. Each party can move a pointer on the display, and point to features on the drawings. The telephones and video cameras allow the parties to see and speak with each other

(Roseman, 1:42-49.) In addition, "[t]he participants can privately whisper or pass notes to each other, without the knowledge of the others." (Roseman, 2:49-50.) Further details on how Roseman discloses communicating content among users are set forth in the discussion of later claim limitations.

59. The preamble of claim 1 continues by reciting that the computer system "**includ**[es] **a controller computer and a database which serves as a repository of tokens for other programs to access** . . ." The "controller computer" in Roseman takes the form of a networked server computer, which Roseman calls the "host computer" or "host":

> These individual [participant] systems are located at different geographic locations, and, when a virtual conference is to be held, <u>become connected to a central, host, computer (or multiplicity of host computers)</u> via the proper combination of Local Area Networks (LANs) and Wide Area Networks (WANs).

(Roseman, 3:14-19 (underlining added), 1:50-52 ("The host controls many of the events occurring during the conference, as well as those occurring both during initiation of the conference and after termination of the proceedings.").)

60. Roseman also discloses "**a database which serves as a repository of tokens for other programs to access**." The tokens in Roseman take the form of "**keys**," which are stored and distributed by the host computer to potential conference participants. More specifically, Roseman explains that in creating a new virtual conference room, the creator can cause the host to send invitations to participants. Each invitation contains a "key" that relates to the identity of the invitee and provides the permissions allowing access to the conference room:

> Before an invitation list is compiled, the level of invitations must be specified by the invitor. Three levels of invitations are considered.
>
> 1. an invitation is for the Invitee only.
> 2. an invitation is for the Invitee, but can be passed to a delegate, who will attend in place of the Invitee.
> 3. an invitation is an open invitation to anyone wishing to attend.
>
> <u>Invitations contain "keys" which conform to the above invitation level</u>. Level 1 keys may not be passed to any other person and may not be copied. Level 2 keys may be passed to exactly one other person and may not be copied. If the key is returned to the original invitee than it may be passed again. Level 3 keys may be freely distributed and copied. The meeting is considered to be public.
>
> <u>The meeting room "knows" about each key and its invitation level</u>. <u>Persons with improper keys are not admitted to the room</u>. A person without a key may be admitted to the room only by someone already

> in the room or by the person responsible for the room.
> Invitations and keys are distributed electronically. The key is an electronic object attached to the invitation.

(Roseman, 9:34-55 (underlining added).)

61. The passages above show that the "**keys**" in Roseman qualify as "**tokens**" because keys are pieces of information associated with user identity, that control whether a user has permission to enter a conference room. Roseman confirms that a key is a "piece of information" by stating that "the key is, essentially, a block of data, or a code." (Roseman, 6:60-61; *see also id.*, 9:54-55 ("The key is an electronic object attached to the invitation.").)

62. Roseman also confirms that a "key" is associated with user identity. For example, the "Level 1" key described in the passage above is associated with a single invitee, and cannot be passed to or used by any other person. (Roseman, 9:37, 9:43-44.) The key is also used to determine whether or not a user will be allowed access to the conference room. (Roseman, 10:61-64 ("To open a door with a key, the user drops the key onto the door lock. If the key is valid and the user has the authority to use the key, the door opens and the user is admitted to the room.") (underlining added).) The "keys" therefore qualify as "**tokens**."

63. Roseman also discloses that the host computer has a "**database which serves as a repository**" of keys (tokens), because the host computer stores the

keys for a particular conference room. In particular, Roseman discloses that a "meeting room" is stored on the host computer. (Roseman, 9:61-63 ("Meeting Facilitator (or Requestor) creates [sic] meeting room <u>on a host computer</u> which is accessible to all Invitees."), 7:30-31 ("[T]he host creates the conference room."), 12:16-18 ("The conference room itself is actually a combination of stored data and computer programs.").) As noted above, Roseman explains that "<u>[t]he meeting room 'knows' about each key and its invitation level</u>. Persons with improper keys are not admitted to the room." (Roseman, 9:49-51 (underlining added).)

64. A copy of each key is therefore stored on the host computer – otherwise the meeting room could not "know[] about each key and its invitation level" (*id.*), or verify whether the invitee's user's key was valid in response to a request for access. (Roseman, 10:61-64.) Thus, Roseman discloses a host computer with a "database which serves as a repository of tokens" because the host computer stores the keys issued to invitees that control access to the room.

65. As noted previously, although Roseman discloses the claimed database and repository of tokens, it does not expressly use the word "**database**" or describe the storage methodology in detail. In my opinion, this does not provide any distinction between Roseman and the claim. A person of ordinary skill in the art would have understood the claimed "database" under its broadest reasonable

construction to simply refer to a stored collection of tokens. The ’356 patent does not provide any detail about the claimed “database” except stating that the tokens “are stored in memory **11** in a control computer database, along with personal information about the user, such as the user’s age.” (’356, 8:13-15.) The patent does not specify any details regarding the storage of tokens in a database and does require that the database be any particular type, such as relational.

66. In any event, even if one were to argue that Roseman does not sufficiently disclose the claimed “database which serves as a repository of tokens,” the addition of a database to Roseman would have been trivially obvious to a person of ordinary skill in the art. Database technologies predated the ’356 patent by decades, and it was known to use databases to store user identity and authentication information (“tokens”). For example, <u>Rissanen</u>, entitled “Password Verification System,” discloses a technique for user authentication in which user identity information and passwords, which are analogous to and serve the same purpose as the “keys” in Roseman, are stored in a database:

> Some business computer systems are arranged to initially <u>record and store passwords assigned to users</u>. In response to a prompt by the system for the user’s password, the user enters the password onto a keyboard and <u>the system compares the keyboard entered password with the stored passwords</u> and enables the user to access the system

when the entered password matches the previously stored password.

(Rissanen, Ex. 1004, 1:21-28 (underlining added).) Rissanen discloses that this password information, as well as the user's account code (login information), are stored in a database. (Rissanen, Ex. 1004, at 2:26-29 ("Users are initially entered into a password database stored in the computer system by assigning each user an account code and a password, such as consisting of a number of numerical digits."), Fig. 2 (showing password file **101** with passwords for each user).)

67. ***Rationale and Motivation to Combine***: It would have been obvious to a person of ordinary skill in the art to combine Roseman with Rissanen, with no change in their respective functions, predictably resulting in the virtual conference system of Roseman in which the conference room "keys" are stored in a database which serves as a repository of keys for other programs to access. A skilled artisan would understand that the user identity and password information in Rissanen is analogous to the "keys" in Roseman, and would be motivated to make this combination. In fact, a person of ordinary skill in the art reading Roseman would have found it plainly apparent that the host computer would store and maintain a copy of the keys issued to invitees in order to verify the stored key against a key provided by a user seeking access. A person of ordinary skill in the art would have understood that the key verification step in Roseman might not function properly if

the host computer could not store and retrieve previously-issued key information to determine validity when a user presents a key seeking access to a conference room. (Roseman, 9:49-50 ("The meeting room 'knows' about each key and its invitation level."), 10:61-64 ("To open a door with a key, the user drops the key onto the door lock. If the key is valid and the user has the authority to use the key, the door opens and the user is admitted to the room.").) Storing the keys in a database is one of a finite number of predictable, well-known solutions to the problem of verifying whether a previously-issued key matches or otherwise corresponds to a key later presented by a user seeking access to a conference room.

68. In short, a person of ordinary skill in the art would have found nothing inventive or non-obvious about the idea of storing "keys" in database. As noted previously, Rissanen goes on to describe a more advanced technique for storing and recognizing spoken (voice) passwords, but these additional details would not have discouraged my proposed combination. I have relied upon Rissanen for its basic disclosures relating to the ability to store "tokens" in a database, and as such, it does not matter if the passwords are text, audio, or some other media. A person of ordinary skill in the art would have found the basic teachings relating to the storage of user information and passwords applicable to any system that requires user authentication as a prerequisite to access, such as Roseman.

69. Roseman also discloses that the database serves as a repository of tokens "**for other programs to access**." Roseman discloses that the keys on the host computer may be accessed by "other programs," *e.g.*, the various meeting or conference rooms maintained on the host computer. As noted above, Roseman discloses that each conference room "is actually a combination of stored data and computer programs." (Roseman, 12:16-18 (underlining added).) Moreover, in order to access a conference room, the host computer presents a virtual "hallway" containing "doors," each door representing a different conference/meeting room. (Roseman, 9:63-65 ("The meeting room door is accessible from a hallway which has doors to other meeting rooms."), 10:28-29 ("Meeting rooms are child rooms of the hallway.").) Each meeting room therefore contains a number of computer programs, and each meeting room itself can be thought of as a program. These programs access the repository of keys when a user presents a key to obtain access to a conference room.

70. As explained in Roseman: "When a person wants to go to a room, he first enters the hallway. The user's display shows an image of a hallway with various doors to rooms." (Roseman, 10:30-32.) If a user locates the door for the appropriate conference, it can drop the key to attempt to gain access: "To open a door with a key, the user drops the key onto the door lock. If the key is valid and

the user has the authority to use the key, the door opens and the user is admitted to the room." (Roseman, 10:61-64.) The repository of tokens is therefore accessed by the conference rooms and the programs within them, *e.g.* to verify if the user-provided token is valid. Moreover, the repository is also indirectly accessed by programs on participant computers as they must present their key to the host computer, which in turn validates the key against previously-issued keys in the repository to determine whether or not to allow access. Roseman in view of Rissanen therefore discloses multiple embodiments of a repository of tokens "**for other programs to access**," as recited in the preamble.

71. The preamble of claim 1 concludes by reciting, "**thereby affording information to each of a plurality of participator computers**." Roseman explains that, if the key (token) is valid and the user is authorized to use it, "the door opens and the user is admitted to the room. The other users in the room are alerted to a new presence and receive any relevant information." (Roseman, 10:63-65.) The conference room participants are then afforded information:

> When a user enters a room with other occupants, the data connection is made. Audio and video connections are made if supported by the user, the room and the other users. A small picture of each user is displayed in the meeting room to indicate presence. If video links are enabled than [sic] the picture may be replaced with a video signal

> from the user, typically showing the user. The majority of the display shows the room's table, walls, etc.

(Roseman, 11:11-17.) Roseman discloses multiple ways of communicating ("affording") information to meeting participants. For example, participants can place documents on a virtual table of the conference room to share with other users (8:1-4, 11:18-22), write shared notes (8:18-21), engage in private voice conversations with other participants (9:16-25), and send private text messages to other participants (9:26-31). Additional details on "affording information" are provided in my discussion of element 1**[c]**, below.

72. The preamble of claim 1 further concludes by reciting, "**thereby affording information to each of a <u>plurality of participator computers</u> which are otherwise independent of each other**." Each meeting participant in Roseman has a participator computer, which Roseman calls a "local computer." (Roseman, 1:34-37 ("Two (or more) parties each operate their own local computers. The computers have associated video cameras, speaker-type telephones, and pointing devices (such as 'mouses')."); *id.*, 2:64-65 ("Every office is equipped with the following equipment: a computer (termed a 'local computer' herein) . . . .").) The participants' local computers can run conventional operating systems and environments such as Microsoft Windows. (Roseman, 12:1-8.)

73. Finally, each local computer in Roseman is "**otherwise independent of each other**" because the computers are located at different geographic locations and only become part of a virtual conference when connected to the host computer. (Roseman, 3:14-19 ("These individual systems are located at different geographic locations, and, when a virtual conference is to be held, become connected to a central, host, computer (or multiplicity of host computers) . . .").) Roseman confirms, in fact, that the local computers can be separated by considerable distances, *e.g.* in different states or in several cities within a state. (Roseman, 4:47-53, Fig. 4, Fig. 5 (showing company facilities in several cities in Ohio).) In the event it is argued that Roseman's local computers are not "otherwise independent of each other" because they are connected through a network belonging to an enterprise or company, it would have been trivially obvious, as discussed above, to adapt Roseman to the Internet such that the Internet is the only network shared by the individual local computers.

74. The participator computers in Roseman are therefore "otherwise independent of each other," as recited in the preamble. Having addressed the preamble of claim 1, I will now turn to the remaining limitations of claim 1.

b. "authenticating a first user identity and a second user identity according to permissions retrieved from the repository of tokens of the database" (Claim 1[a])

75. Roseman discloses authenticating at least "a first user identity" and "a second user identity" according to permissions retrieved from the repository of tokens of the database, as recited in the claim. As explained in the discussion of the preamble, Roseman discloses that each user must be authenticated by using a "key" to attempt to enter the conference.

> To open a door with a key, the user drops the key onto the door lock. <u>If the key is valid and the user has the authority to use the key, the door opens and the user is admitted to the room</u>. The other users in the room are alerted to a new presence and receive any relevant information.

(Roseman, 10:61-65 (underlining added).)

76. Roseman confirms that this validation and authorization process involves authenticating a "**user identity**." For example, as noted above, the host computer can assign a "Level 1" key to a user, which is "for the Invitee only" and "may not be passed to any other person and may not be copied." (Roseman, 9:37, 9:43-44). Such a key is therefore associated with a single particular user identity, as confirmed by the permissions that prohibit sharing or assignment of the key.

"The meeting room 'knows' about each key and its invitation level. Persons with improper keys are not admitted to the room." (Roseman, 9:49-51.)

77. Finally, the user identity authentication process described above may occur for at least "**a first user identity**" and "**a second user identity**," and more, because a conference may have multiple invitees. (*See* Roseman, Fig. 9 (showing conference having six participants).) Although Roseman describes the validation and authorization process above (e.g. invitee drops a key onto a lock to enter a conference room) by reference to a single invitee, it would have been obvious to a person of ordinary skill that the same procedure would be invoked for multiple invitees. As Roseman explains, "[w]hen time for the conference has arrived, the host computer takes roll of the participants as each arrives." (Roseman, 7:44-45.) "When all participants have arrived, the meeting begins." (Roseman, 7:54.)

78. A person of ordinary skill in the art would have found it obvious that the process of entering a conference room would be repeated for all invitees who received a key. Roseman therefore discloses "authenticating a first user identity and a second user identity according to permissions retrieved from the repository of tokens of the database," as recited in claim 1.

c. "affording some of the information to a first of the participator computers via the Internet network, responsive to an authenticated first user identity" (Claim 1[b])

79. Roseman next discloses that, responsive to the first user identity being authenticated and allowed to enter the conference room, the host computer transmits ("affords") some of the information from the conference room to that participant's computer. As Roseman explains:

> When a user enters a room with other occupants, the data connection is made. Audio and video connections are made if supported by the user, the room and the other users. A small picture of each user is displayed in the meeting room to indicate presence. If video links are enabled than the picture may be replaced with a video signal from the user, typically showing the user. The majority of the display shows the room's table, walls, etc.

(Roseman, 11:10-17.) For example, the "display" that "shows the room's table, walls, etc" and the "small picture[s]" or "video signal[s]" of other users in the conference room are communicated ("afforded") to the participant responsive to the participant being authenticated and allowed to enter the conference room. In addition:

> Objects (documents) can be shared in the conference room by placing them on the table. This might be done by dragging an icon of the object from the outside (users non-"meeting room" windows) onto the

> table. Ownership of the object is still maintained. If the object owner wishes, the object may be copied, borrowed by other users, or given to other users. The object may be altered (changed, annotated) by anyone with permission to do so.

(Roseman, 11:18-26 (under "Inside the Meeting Room").)

80. Roseman discloses several other ways of communicating ("affording") some of the information to the first participant computer. For example:

- A participant can use a "notepad" tool to write on the virtual walls of the conference room. (Roseman, 8:18-37.)
- A participant can enter "Whisper Mode" to engage in a private voice conversation. (Roseman, 9:16-25.) "At this time, the host makes an audio connection between the two whispering parties, and between nobody else. The parties can communicate, until they terminate whisper mode." (Roseman, 9:22-25.)
- A participant can pass a private textual note. (Roseman, 9:26-31.) "When the other party sees the note on his picture, as in Figure 12, he can drag it to a private viewing area, double-click it, and read it. No other people are aware of the passed note." (Roseman, 9:28-31.)

81. The types of information described above are communicated ("afforded") responsive to the participant being authenticated because the participant would otherwise not be able to receive such information.

82. As noted above, this limitation recites "affording **some** of the information to a first of the participator computers." To the extent the word "some" requires that not all of the information in a conference room be made available to the first participant computer, this is readily disclosed by the teachings above. Because meeting participants can engage in private conversions and one-on-one note-passing, Roseman makes clear that less than all of the information for a conference may be made available to a given participant. For example, if the first participant was not party to a private conversation or note between two other participants, the first participant's computer would receive only "some" (but not all) of the information available in the virtual conference/meeting room.

83. Finally, this claim limitation recites "affording some of the information to a first of the participator computers **via the Internet network**." Roseman discloses that the host and participant computers may be connected via a Wide Area Network (WAN). (Roseman, 3:14-19, 1:37-41.) A person of ordinary skill in the art would have understood that the Internet is an example of a Wide Area Network (WAN), but Roseman does not expressly mention the Internet.

84. Nevertheless, adapting the virtual conferencing system of Roseman to communicate over the Internet would have been obvious to a person of ordinary skill. For example, Vetter discloses that, well before April 1996, the Internet was being used to facilitate precisely the same types of computer-based conferencing functions described in Roseman, such as video and audio conferencing and document sharing (via shared whiteboards):

> Videoconferences are becoming increasingly frequent on the Internet and generating much research interest. Readily available software tools enable real-time audio and video channels as well as shared whiteboards that allow groups to collaborate on distributed group work more quickly and easily than ever (see sidebar on available tools).
>
> The Internet infrastructure is beginning to support videoconferencing applications in several ways. First, the emerging multicast backbone (or MBone) can efficiently send traffic from a single source over the network to multiple recipients. At the same time, many workstations attached to the Internet are being equipped with video capture and sound cards to send and receive video and audio data streams. The price/performance of these hardware devices has finally reached a level that makes wide-scale deployment possible, which is perhaps the most important factor in the recent growth of videoconferencing applications.

(Vetter, Ex. 1005, at p. 77 (underlining added).)

85. Vetter describes a number of conferencing tools for performing real-time collaboration over the Internet. (*Id.* at p. 78 (under "Available Conferencing Tools").) One example is "CU-SeeMe," which Vetter describes as "a software platform that supports audio and video conferencing over the Internet." (*Id.*) Vetter explains that CU-SeeMe "is becoming very popular" (*id.* at p. 77), and discloses a server program known as the CU-SeeMe "reflector" that facilitates multiparty conferencing. (*Id.* at p. 78.) Vetter therefore discloses communicating information to participator computers via the Internet network.

86. ***Rationale and Motivation to Combine***: It would have been obvious to a person of ordinary skill in the art to combine Roseman with Vetter, with no change in their respective functions, predictably resulting in the virtual conferencing system of Roseman in which the host (server) and participant computers communicate via the Internet. Vetter provides an express motivation for this combination by confirming that "[v]ideoconferences are becoming <u>increasingly frequent</u> on the Internet" (*id.*, p. 77), and that the "CU-SeeMe videoconferencing tool is also becoming <u>very popular</u>." (*Id.* (underlining added to both).) Moreover, a person of ordinary skill in the art would have recognized the Internet as one of the largest networks for connecting remote computers (if not the

largest), making it the obvious Wide Area Network (WAN) for use with Roseman to connect the host and participant computers.

87. Vetter also discloses that the increasing popularity of videoconferencing was fueled by the fact that, as of January 1995, "[t]he price/performance" of hardware devices had finally reached a level in which widespread deployment was possible. (*Id.*) A person of ordinary skill in the art would have understood that the ratio of price-to-performance would have continued to improve, making videoconferencing even more attractive in April 1996 than it was in January 1995 when Vetter was published.

88. Nothing in Vetter would discourage or teach away from this combination. Vetter has an extended discussion of some of the challenges he encountered in using Internet videoconferencing in a classroom context, but none of those issues would have discouraged my proposed combination. Vetter describes issues such as maintaining software and hardware configurations, coordinating when individuals at a site should speak, audio feedback caused by participants leaving their microphones open, delays in whiteboard performance, and network performance of video streams. (*Id.*, p. 78-79.) None of these issues would have discouraged my proposed combination. Most of the problems identified by Vetter are directly attributable to using Internet videoconferencing in

a very unique classroom context. Vetter even acknowledges that "these tools may not have been designed for such an environment, but my goal is to point out important issues in distance-learning video/audio applications." (*Id.*, p. 78 (top of page).) Vetter nevertheless ends on a decidedly positive note by confirming that "video and audio conferencing are an increasingly important way of carrying out collaborative group work." (*Id.*, p. 79 (right column).)

89. A person of ordinary skill in the art would have understood that the videoconferencing system of Roseman involves a simpler conferencing setup with a smaller number of participants, which could avoid or at least reduce the severity of all of the issues encountered by Vetter. A person of ordinary skill in the art would also have understood that network performance in a real-time conferencing application depends on a multiplicity of different factors including the speed of the connections, the number of participants, the amount and type of information being sent, and many other factors. A person of ordinary skill in the art would have understood that performance considerations are a fact-of-life in any conferencing system (including to this day), and as such, the network performance issues identified by Vetter would not have discouraged a skilled artisan from using the Internet to support the type of conferencing functions disclosed in Roseman. In

my opinion, therefore, Roseman and Vetter disclose and render obvious this claim limitation.

d. "affording some of the information to a second of the participator computers via the Internet network, responsive to an authenticated second user identity" (Claim 1[c])

90. This limitation is substantially identical to the previous limitation except that it pertains to the "<u>second</u> of the participator computers" and the "<u>second</u> user identity." The analysis for the previous limitation applies with full force here. As explained above, Roseman discloses that a conference can contain multiple participants who receive and share information. Because the features described in Roseman are available to multiple conference participants, the same analysis for the "first participator computer" in claim 1**[b]** would apply to the participator computer of any other conference/meeting participant. The system of Roseman could therefore send "some of the information to a second of the participator computers via the Internet network," for the same reasons as above.

e. "running controller software on the controller computer, in accordance with predefined rules, to direct arbitration of which ones of the participator computers interactively connect within a group of the participator computers" (claim 1[d])

91. Roseman discloses that the software running on the controller computer (the "host computer") controls the conference/meeting room. In fact,

Roseman includes an Appendix with "pseudo-code" instructions describing the various programming operations that the host computer can perform to carry out conference functions. (Roseman, 12:66-13:2, 13:7-16:43.)

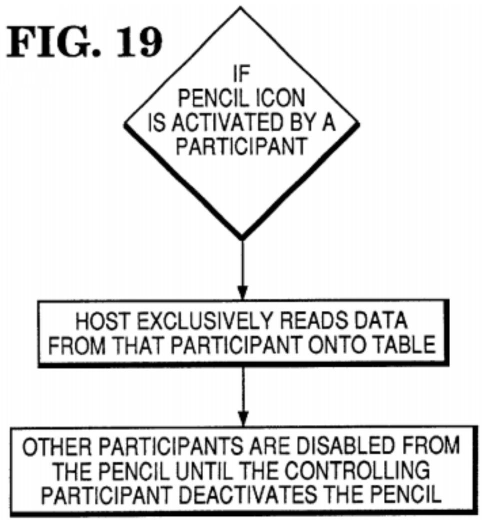
92. Roseman discloses that the controller software on the host computer uses "**predefined rules**" to "**direct arbitration of which ones of the participator computers interactively connect within a group of the participator computers**." For example, a conference requester specifies a number of things upon creating a conference, such as "[w]hat rules govern the conduct of the meeting," "[d]oes the Requester have absolute control of the voice and message interaction among the participants," or "[i]s the meeting a brainstorming free-for-all, where numerous people can speak at once?" (Roseman, 3:52-56.) These predefined rules are further described as follows:

> The room may be used to impose discipline on the meeting procedure. <u>For instance, Robert's Rules of Order may be used to prevent a free for all of communication</u>. The room would require that certain procedural issues be followed before allowing a vote, identified or anonymous, to occur (another built in meeting procedure), or before someone was allowed to speak. Within the room a talking queue might be built so that only one person would speak at a time, followed by the next person and so on.

(Roseman, 11:38-46 (underlining added).)

93. Roseman therefore discloses predefined rules that may be used to "direct arbitration of which ones of the participator computers interactively connect within a group of the participator computers." A person of ordinary skill in the art would have understood that the rules described above are enforced through software functionality on the host computer ("controller software on the controller computer"). (*See* Roseman, 1:50-52 ("The host controls many of the events occurring during the conference, as well as those occurring both during initiation of the conference and after termination of the proceedings.").) Roseman also discloses that these rules qualify as "**predefined rules**" because the user can specify them as part of the conference room creation process, before any participant is invited or has joined. (Roseman, 3:22-28, 3:52-56 (in creating a conference room, the requester may specify "[w]hat rules govern conduct of the meeting").) Roseman discloses many other examples of "predefined rules" enforced by the host computer that separately and independently satisfy this limitation. For example, Roseman teaches a "pencil" tool that allows a participant to "write" in the conference room. As shown in Figure 19 (to the right), when a participant activates the pencil feature, other

FIG. 19



participants may not use the pencil until the controlling participant is finished. This provides another example of "**predefined rules**" that arbitrate which participant may use the pencil, thus "direct[ing] arbitration of which ones of the participator computers interactively connect within a group of the participator computers," as recited in the claim.

94. Roseman describes at least two private messaging techniques that satisfy this claim limitation. First, as described previously, a participant can enter "Whisper Mode" to engage in a private voice conversation with another participant. (Roseman, 9:16-25.) "At this time, the host makes an audio connection between the two whispering parties, and between nobody else. The parties can communicate, until they terminate whisper mode." (Roseman, 9:22-25 (underlining added).)

95. Second, a participant can pass a private textual note to another participant. (Roseman, 9:26-31.) "When the other party sees the note on his picture, as in Figure 12, he can drag it to a private viewing area, double-click it, and read it. No other people are aware of the passed note." (Roseman, 9:28-31 (underlining added).) The privacy of the note is enforced by software functionality on the host. (Roseman, 15:12-15 & Fig. 17C ("HOST TRANSMITS NOTE TO IDENTIFIED PARTICIPANT ONLY.") (capital letters in original).)

96. In both of these examples, Roseman discloses a means of privately sharing information between two participants of the conference that is not shared with other participants. These features provide a further example of rules that determine what information will be provided to which participant computer – in other words, "direct[ing] arbitration of which ones of the participator computers interactively connect within a group of the participator computers," as recited in the claim.

97. All of these functionalities involve software running on the host computer, *i.e.* "**controller software on the controller computer**," as claimed. Roseman confirms as much by including an Appendix with "[p]seudo-code usable for programming the host and the local computers," which is "considered self-explanatory" and also "presented in flow-chart format in FIG. 15, et seq." (Roseman, 12:66-13:2.) That pseudo-code describes how the host computer carries out the Whisper Mode, note-passing and pencil features discussed above. (Roseman, 15:6-9 & Fig. 17B (Whisper Mode), 15:10-13 & Fig. 17C (private note-passing), 15:21-27 & Fig. 19 (pencil).) All of these functionalities further describe "predefined rules" because the access controls are performed by the programming for these conference features (as confirmed by the pseudocode), and as such, before any participant joined the meeting.

f. "providing an API on the controller computer, the API multiplexing and demultiplexing API messages by type, creating a virtual connection and providing the virtual connection between channels, private messages, and multimedia objects in the controller computer and the participator computers" (claim 1[e])

98. Because of the length of this claim limitation, I will address portions of the language to ensure that I cover all of its limitations.

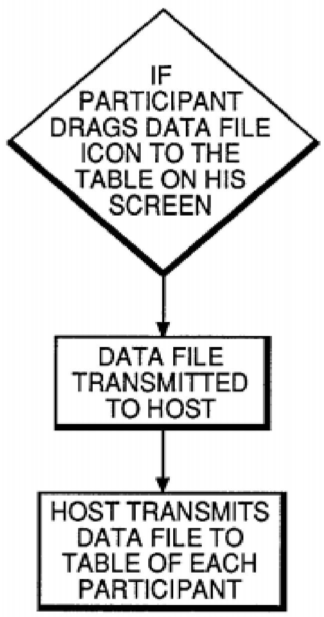
99. As I explained above, the phrase, "**providing an API on the controller computer, the API multiplexing and demultiplexing API messages by type**," should be construed under its broadest reasonable construction as "providing software functionality on the controller computer for sending and receiving messages of different types and communicating each message to software functionality based on the message type."

100. As I explained previously, Roseman discloses pseudo-code programming for the host computer, which is also shown in Figures 15-22. (Roseman, 12:66-13:2.) This software functionality performs functions such as responding to a file being dragged onto the virtual conference/meeting table (Fig. 16A), one participant sending a private note to another (Fig. 17C), one participant using an electronic "pencil" and having his or her actions shown to other participants (Fig. 19), among others. In each of these examples, the host computer

receives a type of message and routes the message to the appropriate software functionality to handle that message type, as shown by my explanation below:

<table>
<tr><th>Pseudocode Functionality from Roseman</th><th>Multiplexing/Demultiplexing Messages by Type</th></tr>
<tr><td>

FIG. 16A



</td><td>

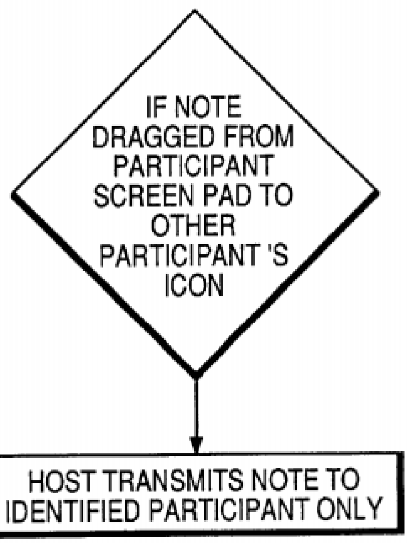
The host computer receives a message of a particular type (e.g. a participant has dragged a file icon onto his or her conference table), to which the host responds by invoking software to receive the file and then send it to each participant.

(*See also* Roseman, 8:1-4 ("Each Invitee can transmit a file (of any suitable kind: data, text, or graphic) to the host, and the host will place the file onto the table, where all participants can see it. To place a document on the table, an Invitee performs a 'drag-and-drop.').)

</td></tr>
</table>

<table>
<tr><th>Pseudocode Functionality from Roseman</th><th>Multiplexing/Demultiplexing Messages by Type</th></tr>
<tr><td>

FIG. 17C



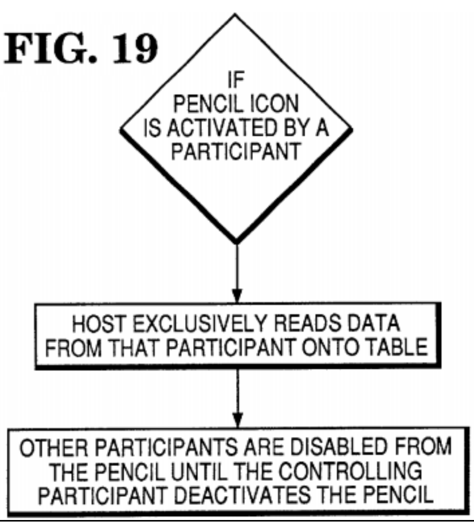
</td><td>

The host computer receives a message of a particular type (request by one participant to send a private note to another), to which the host responds by activating software functionality to receive the note and transmit it only to the other identified participant (and no one else).

(*See also* Roseman, 9:26-31 ("One person can write a note . . . and drag it to the picture of another party. When the other party sees the note on his picture, as in FIG. 12, he can drag it to a private viewing area, double-click it, and read it. No other people are aware of the passed note.").)

</td></tr>
<tr><td>

FIG. 19



</td><td>

The host computer receives a message of a particular type (pencil icon), to which the host responds by activating software functionality to receive the input from the participant and modify the table accordingly, while disabling other participants from using the pencil.

</td></tr>
</table>

101. The fact that the host computer in each example is routing the message to the appropriate software functionality based on the message type is further confirmed by the fact that each of these examples begins with the

conditional "**IF**," which indicates that the host computer analyzes the received message to determine what operations it should perform in response, just like the embodiment in the written description of the '356 patent. (*See supra* **Part III.C**, my analysis of the construction of this claim limitation.)

102. As shown above, Roseman discloses software functionality that transmits and processes particular types of messages, such as placing a document on the table (causing the document to be sent to each participant), using the pencil (causing the participant's actions to be sent to each participant), sending a private message (causing the message to be sent only to the intended recipient), and other messaging functions. Messages corresponding to these commands are multiplexed because the host computer processes each message using the software functionality described above – using the message type to determine the appropriate software.

103. Moreover, the fact that Roseman discloses the claimed "multiplexing" and "demultiplexing" is further confirmed by the Background section of the '356 patent, which identifies conferencing systems like Roseman as examples of systems that perform "multiplexing" as that term is used in the patent:

> On a more complex level, corporations may link remote offices to have a conference by computer. A central computer can control the multiplexing of what appears as an electronic equivalent to a discussion involving many individuals.

('356, Ex. 1002, 1:42-45 (underlining added).) This description mirrors what the "host computer" in Roseman does.

104. Turning to the next part of this limitation, Roseman discloses "**creating a virtual connection and providing the virtual connection between channels, private messages, and multimedia objects in the controller computer and the participator computers**." The specification of the '356 patent does not specifically define the phrase "virtual connection." It mentions it in a single sentence (placing it in quotes) that largely mirrors the claim language. ('356, 6:3-5 ("De/multiplexing via API provides a "virtual connection' between Channel, Private Message, and Multimedia objects in the controller computer **3** and each participator computer **5**.").)

105. The virtual conference room in Roseman provides the claimed "virtual connection" in many ways. First, the software functionality described above (the "API") provides a virtual conference room that connects a group of participants (a "**channel**") via the host computer and allows them to communicate. (Roseman, 7:44-45, 7:54-56 ("When time for the conference has arrived, the host computer takes roll of the participants as each arrives. . . When all participants have arrived, the meeting begins. The table is a common display area which is shown to, and available for work by, each Invitee.").) Each conference room can

also have "child rooms" that conference room participants can join. (Roseman, 10:18-25.) Each "child room" can thus serve as a secondary "**channel**" within the main virtual conference room. (Roseman, 10:22-23 ("A person may not enter the child-room if he cannot enter the parent room.").)

106. Once the meeting begins, the virtual conference room allows participants to send and receive "**private messages**" via the host computer. (Roseman, 2:49-50 ("The participants can privately whisper or pass notes to each other, without the knowledge of the others."); *see also* my analysis of claim 1**[b]** above for further examples of private messages in Roseman.)

107. The virtual conference room also allows the participants to share and collaborate on "**multimedia objects**" through the host computer. The '356 patent does not define the word "multimedia," but in ordinary parlance, it simply refers to content that combines different types of media (such as graphics, text, video, audio, etc.). The patent describes a web page displayed in a browser window with graphics and text (Fig. 26) as an example of a "graphical multimedia message." ('356, 3:53-55; *see also id.*, 11:1-3 ("Inside the Navigator window, the graphical multimedia content, the home page of AIS, is shown.").)

108. Roseman explains that "[w]ith 'multi-media' conferencing, multiple parties are linked by both video and audio media: the parties can see, as well as

hear, each other." (Roseman, Abstract (underlining added).) For example, a user can place a multimedia document, such as a document including text and graphics, on the virtual table and then draw or point to different positions on the document. (Roseman, 2:38-45 ("In the invention, the participants share a common virtual conference table. Each participant can (1) place a document onto the table electronically, (2) write on the document, draw on it, and otherwise manipulate it, and (3) move a pointer to different positions on the document, to point to specific parts of it."), 8:1-4 ("Each Invitee can transmit a file (of any suitable kind: data, text, or graphic) to the host, and the host will place the file onto the table, where all participants can see it.").) At the same time, users can communicate with each other via audio and video. (*See*, *e.g.*, Roseman, 7:65-67 ("With this cursor positioning, each participant can point to items which he or she verbally discusses, using the audio link."), 11:11-13 ("When a user enters a room with other occupants, the data connection is made. Audio and video connections are made if supported by the user, the room and the other users.").) Roseman therefore discloses a virtual connection for multimedia objects.

109. Roseman discloses a virtual conference room that allows groups of conference participants to communicate via the host computer, including through private messages or collaboration on multimedia objects. Roseman therefore

discloses software functionality ("API") on the host computer that "creat[es] a virtual connection and providing the virtual connection between channels, private messages, and multimedia objects in the controller computer and the participator computers," as recited in the claim.

g. "communicating real-time messages within the group of the interactively connected said participator computers" (claim 1[f])

110. This final limitation was already covered in previous claim limitations. Roseman discloses that real-time messages are communicated within the group of the interactively connected participator computers, placing documents on the table and moving from the electronic pencil, among others. Roseman confirms that these messages are communicated in real-time:

> In the invention, the participants share a common virtual conference table. Each participant can
>
> (1) place a document onto the table electronically,
>
> (2) write on the document, draw on it, and otherwise manipulate it, and
>
> (3) move a pointer to different positions on the document, to point to specific parts of it.
>
> <u>All other participants see the [sic] the preceding three events as they occur</u>.

(Roseman, 2:38-47 (underlining added); *see also id.*, 7:54-8:5.)

111. All of these interactions involve "real-time messages" because the messages are communicated to participants as the underlying events occur. In fact, Roseman discloses a feature for recording and archiving the "real-time" events and discussions at a conference. (Roseman, 8:41-46 ("The Requester is given several options of recording the conference. One option is a recording, in real-time, of all events and discussions occurring during the conference."), 12:26-28 ("This persistence allows a person who did not attend the virtual conference in real time to witness it, or parts of it, afterward.") (underlining added to both).) This further confirms that the messages exchanged during a conference are "real-time messages," as recited in the claim.

2. Claims 2-5 (Sound, Video, Graphic & Multimedia Content)

112. Dependent claims 2-5 all recite closely-related subject matter and thus will be treated together. These claims recite:

2. The method of claim 1, wherein the communicating content includes communicating at least one of sound, video, graphic, pointer, and multimedia content.
3. The method of claim 2, wherein said at least one comprises at least two.
4. The method of claim 2, wherein said at least one comprises at least three.
5. The method of claim 2, wherein said at least one comprises at

least four.

('356, 21:57-65.) As shown above, claims 2-5 recite that the content communicated among users must include one or more of five different types of content ("sound, video, graphic, pointer, and multimedia content"), with claim 5 requiring four and thus being the narrowest of these claims. As explained below, Roseman makes clear that at least four types of content (sound, video, graphic, multimedia) may be communicated to conference participants. The fifth type of content ("pointer") will be addressed in a separate section on claims 6 and 7.

113. **<u>Sound and Video</u>**: As explained in Roseman: "When a user enters a room with other occupants, the data connection is made. <u>Audio and video connections are made</u> if supported by the user, the room and the other users. A small picture of each user is displayed in the meeting room to indicate presence. If video links are enabled than [sic] the picture may be replaced with a video signal from the user, typically showing the user." (Roseman, 11:11-16 (underlining added).) As explained at length previously, Roseman discloses that users can talk to each other. (Roseman, 9:16-25, 11:11-13, 11:44-46, 12:34-45.) When the meeting begins, the host can send each participant a picture of each invitee as captured by the local computer's camera. (See Roseman, 7:35-38 ("The pictures

of the invitees can be the actual images seen by the each invitee's close-up camera . . . .").)

114. **Graphic**: Roseman discloses many ways in which graphic content may be communicated. For example: "Each Invitee can transmit a file (of any suitable kind: data, text, or graphic) to the host, and the host will place the file onto the table, where all participants can see it." (Roseman, 8:1-4 (underlining added); *see also id.*, 1:42-46 ("The parties can send the information which they want displayed, such as drawings, to the host computer. The host computer generates a common video screen, which it distributes to the parties: they see the drawings at their own local computers.") (underlining added).) The meeting organizer can specify the décor for the virtual conference room, including "[w]hat paintings are on the walls" (Roseman, 3:40-41.)

115. **Multimedia**: As noted for claim 1, Roseman explains that "[w]ith 'multi-media' conferencing, multiple parties are linked by both video and audio media: the parties can see, as well as hear, each other." (Roseman, Abstract (underlining added).) As explained in detail above and in the preceding paragraphs of this section, Roseman confirms that conference room content can include sound, video, graphics, and text, thus disclosing many examples of "multimedia content."

3. Claims 6-7 ("Pointer" Claims)

116. Claims 6 and 7 both address another type of content known as "pointer" content. Claim 6 requires that sound, video, graphic, multimedia and pointer content be communicated. As explained in the discussion of claims 2-5 above, Roseman clearly discloses at least sound, video, graphic, and multimedia content, so only "**pointer content**" remains to be addressed for claim 6. Claim 7 similarly recites, "[t]he method of claim 1, wherein the communicating content includes communicating **a pointer that allows the content to be produced on demand**." As I explained previously, the term "pointer" refers to a piece of information that points to, or references, other information.

117. Roseman discloses several examples of a "**pointer**." For example, if a user places a document onto the table of the virtual conference room, the host sends an icon to the table of each conference participant. (Roseman, 14:53-57.) This icon serves as a "pointer" because it points to, or references, the underlying document. Clicking on the icon by a participant causes the host computer to present the file to all participants. (Roseman, 14:59-62 ("IF ANY PARTICIPANT ACTIVATES ICON ON TABLE," "DATA FILE PRESENTED ON TABLE BY HOST," "HOST SENDS OPEN FILE TO ALL PARTICIPANTS TABLES") (capital letters in original).) The icon therefore points to the file on the host

computer, and when invoked, the host computer causes the content to appear on the tables of each conference participant.

118. Roseman's note-passing feature provides another example of a "pointer." A user can type a note and drag it onto the picture of another meeting participant. (Roseman, 9:26-28.) A small square icon representing the note appears on the other participant's screen. (Roseman, Fig. 12.) "When the other party sees the note on his picture, as in FIG. 12, he can drag it to a private viewing area, double-click it, and read it. No other people are aware of the passed note." (Roseman, 9:28-31 (underlining added).) The square icon similarly serves as a pointer because it points to, or references, the underlying note content, and retrieves and produces the content on demand from the host computer.

119. The specification of the '356 patent does not define "pointer" but uses a Uniform Resource Locator (URL) as an example of a pointer. ('356, 5:38-43.) In my opinion, the disclosures of Roseman alone disclose the claimed pointer functionality. But in the event it is later argued or determined that "pointer" requires an Internet URL or something functionally similar, then Roseman would render these claims obvious in view of the teachings of Pike **[Ex. 1006.]**

120. Pike is a textbook describing the NCSA Mosaic web browser. (Pike, at 1-2.) Pike provides an introductory section describing several basic and familiar

Internet concepts, such as hypertext links and URLs. (Pike, Ex. 1006, at 36-39.) Pike explains that "[a] *URL* is a complete description of an item, including the location of the item that you want to retrieve." (*Id.* at 38 (italics in original).) "The location of the item can range from a file on your local disk to a file on an Internet site halfway around the world." (*Id.*) Pike explains that a URL can identify any resource on the Internet, and "is not limited to describing the location of WWW [World Wide Web] files." (*Id.*) Pike further explains that a URL can be used to locate and retrieve a document from another computer, and includes "a UNIX-style path for the file that you want to retrieve." (*Id.* at 39.) Pike therefore discloses a "pointer" in the form of a URL.

121. It would have been obvious to a person of ordinary skill in the art to combine Roseman and Vetter with Pike, with no change in their respective functions. This would have predictably resulted in the virtual conferencing system of Roseman in which the clickable icons used to access content (such as documents and notes) included a URL that identified the location of content on the host computer. As explained previously for claim 1**[b]**, Vetter expressly discloses the ability to use the Internet to enable videoconferencing features similar to Roseman. A person of ordinary skill in the art would have understood that, once a system is communicating over the Internet, the URL is a preferred means to identify

resources on the Internet. Pike further confirms that by its publication in 1994, "the Internet has exploded on the computer scene as a topic of national interest." (Pike, Ex. 1006, at 1.)

122. It would have required no leap of inventiveness for a person of ordinary skill in the art to use the ubiquitous Internet URL to identify content stored on the host computer of Roseman which, upon activation, would retrieve the requested content and transmit it to meeting participant computers over the Internet. One of ordinary skill in the art would have recognized that use of the URL method, as taught by Pike, would be particularly advantageous in the context of the Internet and known bandwidth restrictions that existed at the time of the alleged invention. (*See* Pike, Ex. 1006, at p. 43 (top of page).) This is because the file content need not be communicated from the host computer to the participant (this consuming network bandwidth) unless the participant requests to view the content by invoking the URL.

123. Moreover, it was well-known to send messages containing Internet URLs. Pike describes a technique for allowing a user to send URLs for interesting Internet resources in email messages to other people. (Pike, at p. 121.) This capability was well-known because, in part, it was one of the original design goals of the URL. As explained in Request for Comments 1738 by Tim Berners-Lee

(December 1994), the famous standard that defined the syntax of URLs, "there are many occasions when URLs are included in other kinds of text; examples include electronic mail, USENET news messages, or printed on paper." (RFC 1738, Ex. 1011, at p. 22.) RFC 1738 describes techniques for embedding URLs into textual messages so they can be easily used. (*Id.*) By March 1995, URLs were being regularly distributed by businesses, government agencies, academic institutions, and individuals. (Ex. 1012.)

124. By April 1996, therefore, a person of ordinary skill in the art would have found nothing non-obvious about adapting Roseman to send a message containing an Internet URL to meeting participants. Roseman specifically discloses that a document placed on the table of the conference room can include "text" (8:2) and thus, a person of ordinary skill in the art would have understood that such a document could have contained an Internet URL. In my opinion, therefore, the "pointer" limitations in these claims do not provide any meaningful distinction over the prior art.

4. Claims 8-9 and 12 ("API Messages" Claims)

125. Claim 8 recites, "[t]**he method of claim 1, wherein the API includes API messages**," and claim 9 recites, "[t]**he method of claim 1, wherein communications among the controller computer and the participator**

computers are mediated via API messages." Claim 12 recites, "[**t**]**he method of claim 1, wherein the controller software includes multiplexing and de-multiplexing operations carried out as a message type on API messages**."

126. These claim limitations do not appear to impose any limitations that I did not already address in my discussion of claim 1. As I explained in my discussion of the claim construction of the broader "providing an API" phrase from claim 1, the term "API message" simply refers to a message sent, received or processed by the API software functionality. In the case or Roseman, the software functionality on the host computer (the "API") sends and receives messages ("API messages") to communicate information to other participants and to control the features of the conference, such as activating the pencil icon, placing a document on the table, engaging in private audio or voice communications, or sending private textual notes. (*See supra* **Part IV.E.1(f),** my analysis of the "providing an API on the controller computer" limitation of claim 1.) For the reasons stated previously, Roseman discloses these limitations.

5. Claims 14-15 ("Censorship" Claims)

127. Claim 14 recites "[**t**]**he method of claim 1, further including determining censorship of the content**," and claim 15 recites "[**t**]**he method of**

claim 1, wherein the controller computer determines censorship." Roseman describes several means of censorship. For example:

> The room would require that certain procedural issues be followed before allowing a vote, identified or anonymous, to occur (another built in meeting procedure), or before someone was allowed to speak. Within the room a talking queue might be built so that only one person would speak at a time, followed by the next person and so on.

(Roseman, 11:40-46.) In addition, the host computer (the "controller computer") can perform censorship by acting as a conference "moderator" and regulating when and/or how long participants can speak during the conference, or preventing a disruptive participant from continuing to speak:

> Host Can Act as Moderator. The Requestor may wish to hold a conference wherein ideas are freely exchanged among the participants. It is possible that this intent can be defeated by an aggressive person who dominates the conference, and, in effect, maintains a "filibuster."
>
> The host can automatically prevent filibustering, in several ways. One, the host can monitor the speech of each person, and place a limit on the total time allowed to each person. The limit can be overriden by the Requester, or by a vote taken by the host of the other participants.
>
> Two, while one participant is speaking, the host can monitor the audio input of the other participants. The host looks for instances when the speaker refuses to stop talking when the other participants speak.

> When the host finds such instances, the host issues a message to all participants stating that a filibuster appears to be occurring, and requests a vote as to whether to allow the filibuster to continue.

(Roseman, 12:29-45 (underlining added).)

128. These disclosures satisfy the "censorship" limitation of claims 14 and 15. In fact, Roseman's examples of limiting who can communicate mirror the examples of "censorship" in the written description of the '356 patent. ('356, 8:41-46 ("Censorship can control of [sic] access to system **1** by identity of the user, which is associated with the user's tokens. By checking the tokens, a user's access can be controlled per group, as well as in giving group priority, moderation privileges, etc. Censorship can also use the tokens for real time control of data (ascii, text, video, audio) from and to users . . . .").)

6. Claims 16-17 (Internet/URL Claims)

129. Claim 16 recites, "[**t**]**he method of claim 1, wherein the communicating is conducted over the network, including the Internet**." This claim does not appear to impose any additional limitations that were not covered in my discussion of claim 1. As I explained for claim 1**[b]**, Roseman discloses a conferencing system in which communication is conducted over the network, and Vetter discloses that that network may include the Internet. (*See* my analysis in **Part IV.E.1(c)**, for the "via the Internet" limitation of claim 1.)

130. Claim 17 similarly does not recite any features not covered by my discussion of claims 6 and 7. That claim recites, "[**t**]**he method of claim 1, wherein the communicating content includes communicating content invoked with a URL**." As explained in my discussion of claims 6 and 7, Roseman discloses a system in which content may be communicated in response to invoking a pointer, such as an icon on the virtual table. It would have been obvious in view of Pike that the pointer could be a URL. (*See* **Part IV.E.3**, above, my analysis of "pointer" claims 6-7.) For the reasons explained for claims 6 and 7, therefore, it would have been obvious to a person of ordinary skill in the art that the clickable icons used to access content in Roseman could include a URL that transmits content to the participant computer when the user clicks the URL.

7. Claim 18 ("JAVA™ application" Claim)

131. Claim 18 recites, "[**t**]**he method of claim 1, wherein the controller software comprises a JAVA™ application**." In my opinion, this claim recites nothing of patentable significance. Java was a known programming language that could be used to make application software.

132. It would have been obvious to a person of ordinary skill in the art to write the software of Roseman as "a JAVA™ application," as recited in claim 18. Gosling, entitled *Java Intermediate Bytecodes* and published in January 1995,

confirms that Java was a known computer programming language that predated the ’356 patent. (Gosling, Ex. 1007, at p. 111 (“Java is a programming language loosely related to C++.”).) A person of ordinary skill in the art would have been motivated to use Java for a variety of reasons. For example, Gosling explains that “[o]ne of the obvious benefits” to using Java was that compiled programs were “portable,” meaning they could execute on any kind of CPU. (*Id*. at p. 115.) A person of ordinary skill in the art would have recognized this as an “obvious benefit” because the person would no longer have to build separate application programs for different computing platforms. By using Java for the host computer software in Roseman, the developer would be freed from the burden of having to rewrite or change the application in the event of a change in the type of CPU or computer architecture for the server computer.

8. Claim 19 (Apparatus Corresponding to Claim 1)

133. Claim 19 recites an apparatus with substantially similar limitations as claim 1. I have provided a side-by-side comparison below which shows that all of the limitations of claim 19 are recited, using substantially similar language, as claim 1 (underlining showing overlap of language):

Method Claim 1	**Apparatus Claim 19**
1. A method of communicating content among users using [sic] of a computer	19. An apparatus to communicate content among users of a computer

Method Claim 1	Apparatus Claim 19
system including a controller computer and a database which serves as a repository of tokens for other programs to access, thereby affording information to each of a plurality of participator computers which are otherwise independent of each other, the method comprising:	system . . . comprising: a controller computer system, including a controller computer and a database which serves as a repository of tokens for other programs to access, thereby affording information to each of a plurality of participator computers which are otherwise independent of each other in communication with each of the participator computers by
[a] authenticating a first user identity and a second user identity according to permissions retrieved from the repository of tokens of the database;	authenticating a first user identity and a second user identity according to permissions retrieved from the repository of tokens of the database,
[b] affording some of the information to a first of the participator computers via the Internet network, responsive to an authenticated first user identity;	*Limitation not present*
[c] affording some of the information to a second of the participator computers via the Internet network, responsive to an authenticated second user identity;	*Limitation not present*
[d] running controller software on the controller computer, in accordance with predefined rules, to direct arbitration of which ones of the participator computers interactively connect within a group of the participator computers;	wherein the controller computer is running controller software, in accordance with predefined rules, to direct arbitration of which ones of the participator computers interactively connect within a group of the participator computers, to
[e] providing an API on the controller computer, the API multiplexing and demultiplexing API messages by type,	provide an API on the controller computer, whereby the API multiplexes and demultiplexes API messages by

Method Claim 1	**Apparatus Claim 19**
creating a virtual connection and providing the virtual connection between channels, private messages, and multimedia objects in the controller computer and the participator computers; and	type, to create a virtual connection and provide the virtual connection between channels, private messages, and multimedia objects in the controller computer and the participator computers, and to
[f] communicating real-time messages within the group of the interactively connected said participator computers.	allow communication of real-time messages within the group of the interactively connected said participator computers.

134. As shown above, other than the fact that claim 19 does not recite the two "affording" limitations of claims 1**[b]** and **[c]**, the differences between claim 1 and 19 consist largely of insubstantial verb tense differences based on the fact that claim 1 is a method and claim 19 an apparatus claim. As I explained in connection with claim 1 above, this claim is obvious over Roseman and Rissanen. (Vetter is not required here because claim 19 does not recite the Internet.) Roseman discloses an apparatus in the form of a computer system that includes the controller computer ("host computer"), and the participator computers (participants' "local computers"). The host computer carries out all of the functions of claim 19 for the same reasons as claim 1.

9. Claims 20-28, 31, 33-36 (Corresponding Dependent Claims)

135. These claims depend from claim 19 discussed above and are substantially the same as claims depending off independent claim 1 (underlining showing overlap of language):

Apparatus Dependent Claims	Method Dependent Claims
20. The apparatus of claim 19, wherein the content includes at least one of sound, video, graphic, pointer, and multimedia content.	2. The method of claim 1, wherein the communicating content includes communicating at least one of sound, video, graphic, pointer, and multimedia content.
21. The apparatus of claim 20, wherein said at least one comprises at least two.	3. The method of claim 2, wherein said at least one comprises at least two.
22. The apparatus of claim 20, wherein said at least one comprises at least three. 23. The apparatus of claim 20, wherein said at least one comprises at least three. (Duplicates)	4. The method of claim 2, wherein said at least one comprises at least three.
24. The apparatus of claim 20, wherein said at least one comprises at least four.	5. The method of claim 2, wherein said at least one comprises at least four.
25. The apparatus of claim 19, wherein the controller software comprises a JAVA™ application.	18. The method of claim 1, wherein the controller software comprises a JAVA™ application.
26. The apparatus of claim 19, wherein the content includes a pointer which allows the content to be produced on demand.	7. The method of claim 1, wherein the communicating content includes communicating a pointer that allows the content to be produced on demand.
27. The apparatus of claim 19, wherein	8. The method of claim 1, wherein the

Apparatus Dependent Claims	**Method Dependent Claims**
the API includes API messages.	API includes API messages.
28. The apparatus of claim 19, wherein communications among the controller computer and the participator computers are mediated via API messages.	9. The method of claim 1, wherein communications among the controller computer and the participator computers are mediated via API messages.
31. The apparatus of claim 19, wherein the controller software includes multiplexing and de-multiplexing operations carried out as a message type on API messages.	12. The method of claim 1, wherein the controller software includes multiplexing and de-multiplexing operations carried out as a message type on API messages.
33. The apparatus of claim 19, wherein the computer system determines censorship of the content.	14. The method of claim 1, further including determining censorship of the content.
34. The apparatus of claim 19, wherein the controller computer determines censorship.	15. The method of claim 1, wherein the controller computer determines censorship.
35. The apparatus of claim 19, wherein the content is communicated over a network, including the Internet.	16. The method of claim 1, wherein the communicating is conducted over the network, including the Internet.
36. The apparatus of claim 19, wherein the content is communicated by invoking a URL.	17. The method of claim 1, wherein the communicating content includes communicating content invoked with a URL.

136. My analysis of claims 2-5, 7-9, 12, 14-18 applies with full force here.

10. Claim 37 (Similar to Claim 19)

137. For purposes of my analysis, claim 37 is materially the same as claim 19, as shown in the table below. The primary difference is that claim 37 requires

the Internet whereas claim 19 does not. Claim 37 is therefore obvious over Roseman in view of Rissanen and Vetter.

Apparatus Claim 37	**Apparatus Claim 19**
37. An apparatus comprising: a computer system, the computer system including a controller computer and a database which serves as a repository of tokens for other programs to access, thereby affording information to each of a plurality of independent participator computers which are otherwise independent of each other, via the Internet network communicating with the participator computers by	19. An apparatus to communicate content among users of a computer system, the computer system comprising: a controller computer system, including a controller computer and a database which serves as a repository of tokens for other programs to access, thereby affording information to each of a plurality of participator computers which are otherwise independent of each other in communication with each of the participator computers by
authenticating a first user identity and a second user identity according to permissions retrieved from the repository of tokens of the database,	authenticating a first user identity and a second user identity according to permissions retrieved from the repository of tokens of the database,
the controller computer running controller software, in accordance with predefined rules, directing arbitration of which ones of the participator computers interact within a group of the participator computers,	wherein the controller computer is running controller software, in accordance with predefined rules, to direct arbitration of which ones of the participator computers interactively connect within a group of the participator computers, to
providing an API on the controller computer, whereby the API is multiplexing and demultiplexing API messages by type, creating a virtual connection and providing the virtual	provide an API on the controller computer, whereby the API multiplexes and demultiplexes API messages by type, to create a virtual connection and provide the virtual connection between

Apparatus Claim 37	Apparatus Claim 19
connection between channels, private messages, and multimedia objects in the controller computer and the participator computers, and	channels, private messages, and multimedia objects in the controller computer and the participator computers, and to
providing communication of real-time messages within the group of the interactively connected said participator computers.	allow communication of real-time messages within the group of the interactively connected said participator computers.

138. This claim is therefore obvious for the same reasons discussed above.

139. Finally, in my opinion, the disclosures of Roseman, Rissanen, Vetter, Pike and Gosling provide sufficiently detailed disclosures to enable a person of ordinary skill in the art to make the combinations explained above without undue experimentation. The references themselves make clear that the technologies involved were well known to persons of ordinary skill in the art and even commercially available. For example, as noted above, I rely on Roseman for the majority of the limitations in the challenged claims. Roseman specifically identifies "[c]ommercially [a]vailable [e]quipment for use in invention," including screen sharing software, electronic mail software, video conferencing products, computer-controlled telephones, graphics devices and other equipment. (Roseman, 12:46-65.) Although I understand that the disclosures in an issued U.S. patent

(such as Roseman) are presumed enabling, in my opinion, Roseman provides sufficient detail to build the virtual conference room it discloses.

140. I also rely upon Vetter, Rissanen, Pike and Gosling for trivial concepts that were well known to persons of ordinary skill in the art by April 1996 (to the extent not already disclosed in Roseman). Rissanen, which I refer to for its teachings regarding using a database to store tokens, explains that the system can use "a general purpose IBM computer" (Rissanen, 4:32-35) and that the database can be "any one of well known disk, tape, solid state or other type of storage device" (Rissanen, 13:5-14). More generally, by April 1996, commercially-available, off-the-shelf databases such as Oracle, Sybase and Berkeley DB were in wide use, and information regarding their setup and operation widely available, such that persons of ordinary skill in the art would have been capable of using one to store tokens for other programs to access without undue experimentation. Similarly, Vetter explains that "[r]eadily available software tools enable real-time audio and video channels as well as shared whiteboards that allow groups to collaborate on distributed group work" and specifically identifies a number of "available conferencing tools," including Collage, CU-SeeMe, CU-SeeMe Reflector, IVS, MBone, Nevot (Network Voice Terminal), NV (Net Video), SD (Session Director), VAT (Visual Audio Tool) and WB (Whiteboard) (Vetter, Ex.

1005, at p. 77, 78.) More generally, the Internet has been widely used since at least the 1980s. (Pike, Ex. 1006, at 8-10.) Operating systems such as UNIX, which were widely distributed long before April 1996, enabled computers to communicate via the Internet (such as software libraries for creating UDP and TCP socket connections). As to using the Java programming language, Java programming libraries and documentation were freely available before April 1996. Finally, as to the straightforward concept of URLs, the use of URLs in network-based communications was firmly in place by April 1996 and would not have required undue experimentation by a person of ordinary skill in the art in a combination with any of the references upon which I rely in this Declaration.

V. CONCLUSION

141. In signing this Declaration, I recognize that the Declaration will be filed as evidence in a contested case before the Patent Trial and Appeal Board of the United States Patent and Trademark Office. I also recognize that I may be subject to cross-examination in this proceeding. If required, I will appear for cross-examination at the appropriate time. I reserve the right to offer opinions relevant to the invalidity of the ’356 patent claims at issue and/or offer testimony in support of this Declaration.

142. I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true, and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under 18 U.S.C. § 1001.

Dated: June 3, 2016

Respectfully submitted,

Tal Lavian, Ph.D.
Sunnyvale, California

EXHIBIT A

Tal Lavian, Ph.D.

http://telecommnet.com
http://cs.berkeley.edu/~tlavian
tlavian@telecommnet.com

1640 Mariani Dr.
Sunnyvale, CA 94087
(408)-209-9112

Research and Consulting: Telecommunications, Network Communications, and Mobile Wireless technologies

- Scientist, educator, and technologist with over 25 years of experience
- Co-author on over 25 scientific publications, journal articles, and peer-reviewed papers
- Named inventor on over 80 issued and filed patents
- Industry fellow and lecturer at UC Berkeley Engineering – Center for Entrepreneurship and Technology (CET)

EDUCATION

- **Ph.D.**, Computer Science specializing in networking and communications, UC Berkeley
- **M.Sc.**, Electrical Engineering, Tel Aviv University
- **B.Sc.**, Mathematics and Computer Science, Tel Aviv University

EXPERTISE

Network communications, telecommunications, Internet protocols and mobile wireless:

- **Communication networks:** Internet Protocols; TCP/IP suite; TCP; UDP; IP; VoIP; Ethernet; network protocols; network software applications; Data Link, Network, and Transport Layers (L2, L3, L4)
- **Internet Software:** Internet software applications; distributed computing; cloud computing; Web applications; FTP; HTTP; Java; C; C++; client server; file transfer; multicast; streaming media
- **Routing/switching:** LAN; WAN; VPN; routing protocols; RIP; BGP; MPLS;OSPF; IS-IS;DNS; QoS; switching; packet switching; network infrastructure; network communication architectures
- **Mobile Wireless:** Wireless LAN; 802.11; cellular systems; mobile devices; smartphone technologies

LITIGATION SUPPORT SERVICES

- Expert witness in numerous USPTO PTAB – Inter Partes Review (IPR) and CBM cases
- Expert witness in Federal courts and the ITC (over 30 cases)
- Expert reports, depositions, and courtroom testimonies
- Skilled articulation of technical material for both technical and non-technical audiences
- Product and technology analysis, patent portfolios, claim charts, patentability research
- Litigation support and technology education in patent disputes
- Past cases involved Cisco, Juniper, HP, Ericsson, Microsoft, Google, Samsung and Apple

ACCOMPLISHMENTS

- Selected as Principal Investigator for three US Department of Defense (DARPA) projects
- Led research project on networking computation for the US Air Force Research Lab (AFRL)
- Led and developed the first network resource scheduling service for grid computing
- Led wireless research project for an undisclosed US federal agency
- Managed and engineered the first demonstrated transatlantic dynamic allocation of 10Gbs Lambdas as a grid service
- Spearheaded the development of the first demonstrated wire-speed active network on commercial hardware
- Invented over 80 patents; over 50 prosecuted *pro se* in front of the USPTO
- Created and chaired Nortel Networks' EDN Patent Committee
- Current IEEE Senior Member

PROFESSIONAL EXPERIENCE

University of California, Berkeley, Berkeley, CA 2000-Present

Berkeley Industry Fellow, Lecturer, Visiting Scientist, Ph.D. Candidate, Nortel's Scientist Liaison

Some positions and projects were concurrent, others sequential

- Serves as an Industry Fellow and Lecturer at the Center for Entrepreneurship and Technology (CET).
- Studied network services, telecommunication systems and software, communications infrastructure, and data centers
- Developed long-term technology for the enterprise market, integrating communication and computing technologies
- Conducted research projects in data centers (RAD Labs), telecommunication infrastructure (SAHARA), and wireless systems (ICEBERG)
- Acted as scientific liaison between Nortel Research Lab and UC Berkeley, providing tangible value in advanced technologies
- Earned a Ph.D. in Computer Science with a specialization in communications and networking

Telecomm Net Consulting, Inc. (Innovations-IP) Sunnyvale, CA 2006-Present

Principal Scientist

- Consulting in the areas of network communications, telecommunications, Internet protocols, and smartphone mobile wireless devices

- Providing architecture and system consultation for software projects relating to computer networks, mobile wireless devices, Internet web technologies
- Acting as an expert witness in network communications patent infringement lawsuits

VisuMenu, Inc. – Sunnyvale, CA 2010-Present

Co- Founder and Chief Technology Officer (CTO)

- Design and develop architecture of visual IVR technologies for smartphones and wireless mobile devices in the area of network communications
- Design crawler/spider system for IVR / PBX using Asterisk, SIP and VoIP
- Deploy the system as cloud networking and cloud computing utilizing Amazon Web Services (EC2, S3, VPC, DNS, and RDS)

Ixia, Santa Clara, CA 2008-2008

Communications Consultant

- Researched and developed advanced network communications testing technologies:
  • IxNetwork/IxN2X — tests IP routing and switching devices and broadband access equipment. Provides traffic generation and emulation for the full range of protocols: routing, MPLS, layer 2/3 VPNs, Carrier Ethernet, broadband access, and data center bridging.
  • IxLoad — quickly and accurately models high-volume video, data, and voice subscribers and servers to test real-world performance of multiservice delivery and security platforms.
  • IxCatapult — emulates a broad range of wireless access and core protocols to test wireless components and systems. When combined with IxLoad, provides an end-to-end solution for testing wireless service quality.
  • IxVeriWave — employs a client-centric model to test Wi-Fi and wireless LAN networks by generating repeatable large-scale, real-world test scenarios that are virtually impossible to create by any other means.
  • Test Automation — provides simple, comprehensive lab automation to help test engineering teams create, organize, catalog, and schedule execution of tests.

Nortel Networks, Santa Clara, CA 1996 - 2007

Originally employed by Bay Networks, which was acquired by Nortel Networks

Principal Scientist, Principal Architect, Principal Engineer, Senior Software Engineer

- Held scientific and research roles at Nortel Labs, Bay Architecture Labs, and in the office of the CTO

Principal Investigator for US Department of Defense (DARPA) Projects

- Conceived, proposed, and completed three research projects: Active Networks, DWDM-RAM, and a networking computation project for Air Force Research Lab (AFRL)
- Led a wireless research project for an undisclosed US federal agency

Academic and Industrial Researcher

- Analyzed new technologies to reduce risks associated with R&D investment
- Spearheaded research collaboration with leading universities and professors at UC Berkeley, Northwestern University, University of Amsterdam, and University of Technology, Sydney
- Evaluated competitive products relative to Nortel's products and technology
- Proactively identified prospective business ideas, which led to new networking products
- Predicted technological trends through researching the technological horizon and academic sphere
- Developed software for switches, routers and network communications devices
- Developed systems and architectures for switches, routers, and network management
- Researched and developed the following projects:
  - Data-Center Communications: network and server orchestration 2006-2007
  - DRAC: SOA-facilitated L1/L2/L3 network dynamic controller 2003-2007
  - Omega: classified wireless project for undisclosed US Federal Agency 2006
  - Open Platform: project for the US Air Force Research Laboratory (AFRL) 2005
  - Network Resource Orchestration for Web Services Workflows 2004-2005
  - Proxy Study between Web/Grids Services and Network Services 2004
  - Streaming Content Replication: real-time A/V media multicast at edge 2003-2004
  - DWDM-RAM: US DARPA-funded program on agile optical transport 2003-2004
  - Packet Capturing and Forwarding Service on IP and Ethernet traffic 2002-2003
  - CO2: content-aware agile networking 2001-2003
  - Active Networks: US DARPA-funded research program 1999-2002
  - ORE: programmable network service platform 1998-2002
  - JVM Platform: Java on network devices 1998-2001
  - Web-Based Device Management: network device management 1996-1997

Technology Innovator and Patent Leader

- Created and chaired Nortel Networks' EDN Patent Committee
- Facilitated continuous stream of innovative ideas and their conversion into intellectual property rights
- Developed intellectual property assets through invention and analysis of existing technology portfolios

Aptel Communications, Netanya, Israel 1994-1995

Software Engineer, Team Leader

Start-up company focused on mobile wireless CDMA spread spectrum PCN/PCS

- Developed a mobile wireless device using an unlicensed band [Direct Sequence Spread Spectrum (DSSS)]
- Designed and managed a personal communication network (PCN) and personal communication system (PCS), the precursors of short text messages (SMS)
- Designed and developed network communications software products (mainly in C/C++)
- Brought a two-way paging product from concept to development

Scitex Ltd.,Herzeliya, Israel 1990-1993

Software Engineer, Team Leader

Software and hardware company acquired by Hewlett Packard (HP)

- Developed system and network communications (mainly in C/C++)
- Invented Parallel SIMD Architecture
- Participated in the Technology Innovation group

Shalev, Ramat-HaSharon, Israel 1987-1990

Start-up company

Software Engineer

- Developed real-time software and algorithms (mainly in C/C++ and Pascal)

PROFESSIONAL ASSOCIATIONS

- IEEE Senior Member
- IEEE CNSV co-chair Intellectual Property SIG (2013)
- President Next Step Toastmasters (an advanced TM club in the Silicon Valley) (2013)
- Technical Co-Chair, IEEE Hot Interconnects 2005 at Stanford University
- Member, IEEE Communications Society (COMMSOC)
- Member, IEEE Computer Society
- Member, IEEE Systems, Man, and Cybernetics Society
- Member, IEEE-USA Intellectual Property Committee
- Member, ACM, ACM Special Interest Group on Data Communication (SIGCOM)
- Member, ACM Special Interest Group on Hypertext, Hypermedia and Web (SIGWEB)
- Member, IEEE Consultants' Network (CNSV)
- Global Member, Internet Society (ISOC)
- President Java Users Group – Silicon Valley Mountain View, CA,1999-2000
- Toastmasters International

ADVISORY BOARDS

- Quixey (present) – search engine for wireless mobile apps
- Mytopia – mobile social games
- iLeverage – Israeli Innovations

PROFESSIONAL AWARDS

- Top Talent Award – Nortel
- Top Inventors Award – Nortel EDN
- Certified IEEE-WCET - Wireless Communications Engineering Technologies
- Toastmasters International - Competent Communicator (twice)
- Toastmasters International - Advanced Communicator Bronze

Patents and Publications

(Not an exhaustive list)

Patents Issued:

- **US 8,688,796** Rating system for determining whether to accept or reject objection raised by user in social network
- **US 8,572,303** Portable universal communication device
- **US 8,553,859** Device and method for providing enhanced telephony
- **US 8,548,131** Systems and methods for communicating with an interactive voice response system
- **US 8,537,989** Device and method for providing enhanced telephony
- **US 8,341,257** Grid proxy architecture for network resources
- **US8,161,139** Method and apparatus for intelligent management of a network element
- **US 8,146,090** Time-value curves to provide dynamic QoS for time sensitive file transfer
- **US 8,078,708** Grid proxy architecture for network resources
- **US 7,944,827** Content-aware dynamic network resource allocation
- **US7,860,999** Distributed computation in network devices
- **US 7,734,748** Method and apparatus for intelligent management of a network element
- **US 7,710,871** Dynamic assignment of traffic classes to a priority queue in a packet forwarding device
- **US 7,580,349** Content-aware dynamic network resource allocation
- **US 7,433,941** Method and apparatus for accessing network information on a network device
- **US 7,359,993** Method and apparatus for interfacing external resources with a network element
- **US 7,313,608** Method and apparatus for using documents written in a markup language to access and configure network elements
- **US 7,260,621** Object-oriented network management interface

- **US 7,237,012** Method and apparatus for classifying Java remote method invocation transport traffic
- **US 7,127,526** Method and apparatus for dynamically loading and managing software services on a network device
- **US7,047,536** Method and apparatus for classifying remote procedure call transport traffic
- **US7,039,724** Programmable command-line interface API for managing operation of a network device
- **US6,976,054** Method and system for accessing low-level resources in a network device
- **US6,970,943** Routing architecture including a compute plane configured for high-speed processing of packets to provide application layer support
- **US6,950,932** Security association mediator for Java-enabled devices
- **US6,850,989** Method and apparatus for automatically configuring a network switch
- **US6,845,397** Interface method and system for accessing inner layers of a network protocol
- **US6,842,781** Download and processing of a network management application on a network device
- **US6,772,205** Executing applications on a target network device using a proxy network device
- **US6,564,325** Method of and apparatus for providing multi-level security access to system
- **US6,175,868** Method and apparatus for automatically configuring a network switch
- **US6,170,015** Network apparatus with Java co-processor
- **US 8,619,793** Dynamic assignment of traffic classes to a priority queue in a packet forwarding device
- **US 8687,777** Systems and methods for visual presentation and selection of IVR menu
- **US 8,681,951** Systems and methods for visual presentation and selection of IVR menu

- **US 8,625,756** Systems and methods for visual presentation and selection of IVR menu
- **US 8,594,280** Systems and methods for visual presentation and selection of IVR menu
- **US 8,548,135** Systems and methods for visual presentation and selection of IVR menu
- **US 8,406,388** Systems and methods for visual presentation and selection of IVR menu
- **US 8,345,835** Systems and methods for visual presentation and selection of IVR menu
- **US 8,223,931** Systems and methods for visual presentation and selection of IVR menu
- **US 8,160,215** Systems and methods for visual presentation and selection of IVR menu
- **US 8,155,280** Systems and methods for visual presentation and selection of IVR menu
- **US 8,054,952** Systems and methods for visual presentation and selection of IVR menu
- **US 8,000,454** Systems and methods for visual presentation and selection of IVR menu
- **EP 1,905,211** Technique for authenticating network users
- **EP 1,142,213** Dynamic assignment of traffic classes to a priority queue in a packet forwarding device
- **EP 1,671,460** Method and apparatus for scheduling resources on a switched underlay network
- **CA 2,358,525** Dynamic assignment of traffic classes to a priority queue in a packet forwarding device

Patent Applications Published and Pending:
(Not an exhaustive list)

- **US 20140105025** Dynamic Assignment of Traffic Classes to a Priority Queue in a Packet Forwarding Device
- **US 20140105012** Dynamic Assignment of Traffic Classes to a Priority Queue in a Packet Forwarding Device
- **US 20140012991** Grid Proxy Architecture for Network Resources
- **US 20130080898** Systems and Methods for Electronic Communications
- **US 20130022191** Systems and Methods for Visual Presentation and Selection of IVR Menu
- **US 20130022183** Systems and Methods for Visual Presentation and Selection of IVR Menu
- **US 20130022181** Systems and Methods for Visual Presentation and Selection of IVR Menu
- **US 20120180059** Time-Value Curves to Provide Dynamic QOS for Time Sensitive File Transfers
- **US 20120063574** Systems and Methods for Visual Presentation and Selection of IVR Menu
- **US 20110225330** Portable Universal Communication Device
- **US 20100220616** Optimizing Network Connections
- **US 20100217854** Method and Apparatus for Intelligent Management of a Network Element
- **US 20100146492** Translation of Programming Code
- **US 20100146112** Efficient Communication Techniques
- **US 20100146111** Efficient Communication in a Network
- **US 20090313613** Methods and Apparatus for Automatic Translation of a Computer Program Language Code
- **US 20090313004** Platform-Independent Application Development Framework
- **US 20090279562** Content-aware dynamic network resource allocation
- **US 20080040630** Time-Value Curves to Provide Dynamic QoS for Time Sensitive File Transfers
- **US 20070169171** Technique for authenticating network users
- **US 20060123481** Method and apparatus for network immunization
- **US 20060075042** Extensible Resource Messaging Between User Applications and Network Elements in a Communication Network

- **US 20050083960** Method and Apparatus for Transporting Parcels of Data Using Network Elements with Network Element Storage
- **US 20050076339** Method and Apparatus for Automated Negotiation for Resources on a Switched Underlay Network
- **US 20050076336** Method and Apparatus for Scheduling Resources on a Switched Underlay Network
- **US 20050076173** Method And Apparatus for Preconditioning Data to Be Transferred on a Switched Underlay Network
- **US 20050076099** Method and Apparatus for Live Streaming Media Replication in a Communication Network
- **US 20050074529** Method and apparatus for transporting visualization information on a switched underlay network
- **US 20040076161** Dynamic Assignment of Traffic Classes to a Priority Queue in a Packet Forwarding Device
- **US 20020021701** Dynamic Assignment of Traffic Classes to a Priority Queue in a Packet Forwarding Device
- **WO 2007/008976** Technique for Authenticating Network Users
- **WO 2006/063052** Method and apparatus for network immunization
- **WO2000/0054460** Method and apparatus for accessing network information on a network device

Publications

(Not an exhaustive list)

- "R&D Models for Advanced Development & Corporate Research" Understanding Six Models of Advanced R&D - Ikhlaq Sidhu, Tal Lavian, Victoria Howell - University of California, Berkeley. Accepted paper for 2015 ASEE Annual Conference and Exposition- June 2015
- "Communications Architecture in Support of Grid Computing",Tal Lavian,Scholar's Press 2013ISBN 978-3-639-51098-0.
- "Applications Drive Secure Lightpath Creation across Heterogeneous Domains, Feature Topic Optical Control Planes for Grid Networks: Opportunities, Challenges and the Vision." Gommans L.; Van Oudenaarde B.; Dijkstra F.; De Laat C.; Lavian T.; Monga I.; Taal A.; Travostino F.; Wan A.; IEEE *Communications Magazine*, vol. 44, no. 3, March 2006, pp. 100-106.
- *Lambda Data Grid: Communications Architecture in Support of Grid Computing*. Tal I. Lavian, Randy H. Katz; Doctoral Thesis, University of California at Berkeley. January 2006.
- "Information Switching Networks." Hoang D.B.; T. Lavian;*The 4th Workshop on the Internet, Telecommunications and Signal Processing, WITSP2005*, December 19-21, 2005, Sunshine Coast, Australia.
- "Impact of Grid Computing on Network Operators and HW Vendors." Allcock B.; Arnaud B.; Lavian T.; Papadopoulos P.B.; Hasan M.Z.; Kaplow W.; *IEEE Hot Interconnects at Stanford University 2005*, pp.89-90.
- *DWDM-RAM: AData Intensive Grid Service Architecture Enabled by Dynamic Optical Networks*. Lavian T.; Mambretti J.; Cutrell D.; Cohen H.J; Merrill S.; Durairaj R.; Daspit P.; Monga I.; Naiksatam S.; Figueira S.; Gutierrez D.; Hoang D.B., Travostino F.; CCGRID 2004, pp. 762-764.
- *DWDM-RAM: AnArchitecture for Data Intensive Service Enabled by Next Generation Dynamic Optical Networks*. Hoang D.B.; Cohen H.; Cutrell D.; Figueira S.; Lavian T.; Mambretti J.; Monga I.; Naiksatam S.; Travostino F.; Proceedings IEEE Globecom 2004, Workshop on High-Performance Global Grid Networks, Houston, 29 Nov. to 3 Dec. 2004, pp.400-409.
- *Implementation of a Quality of Service Feedback Control Loop on Programmable Routers*. Nguyen C.; Hoang D.B.; Zhao, I.L.; Lavian, T.; Proceedings, 12th IEEE International Conference on Networks 2004. (ICON 2004) Singapore, Volume 2, 16-19 Nov. 2004, pp.578-582.
- *A Platform for Large-Scale Grid Data Service on Dynamic High-Performance Networks*. Lavian T.; Hoang D.B.; Mambretti J.; Figueira S.; Naiksatam S.; Kaushil N.; Monga I.; Durairaj R.; Cutrell D.; Merrill S.; Cohen H.; Daspit P.; Travostino F; GridNets 2004, San Jose, CA., October 2004.
- *DWDM-RAM: Enabling Grid Services with Dynamic Optical Networks*. Figueira S.; Naiksatam S.; Cohen H.; Cutrell D.; Daspit, P.; Gutierrez D.; Hoang D. B.; Lavian T.; Mambretti J.; Merrill S.; Travostino F; Proceedings, 4th IEEE/ACM International Symposium on Cluster Computing and the Grid, Chicago, USA, April 2004, pp. 707-714.
- *DWDM-RAM: Enabling Grid Services with Dynamic Optical Networks*. Figueira S.; Naiksatam S.; Cohen H.; Cutrell D.; Gutierrez D.; Hoang D.B.; Lavian T.; Mambretti J.; Merrill S.; Travostino F.; 4th IEEE/ACM International Symposium on Cluster Computing and the Grid, Chicago, USA, April 2004.

- *An Extensible, Programmable, Commercial-Grade Platform for Internet Service Architecture*. Lavian T.; Hoang D.B.; Travostino F.; Wang P.Y.; Subramanian S.; Monga I.; IEEE Transactions on Systems, Man, and Cybernetics on Technologies Promoting Computational Intelligence, Openness and Programmability in Networks and Internet Services Volume 34, Issue 1, Feb. 2004, pp.58-68.
- *DWDM-RAM: An Architecture for Data Intensive Service Enabled by Next Generation Dynamic Optical Networks*. Lavian T.; Cutrell D.; Mambretti J.; Weinberger J.; Gutierrez D.; Naiksatam S.; Figueira S.; Hoang D. B.; Supercomputing Conference, SC2003 Igniting Innovation, Phoenix, November 2003.
- *Edge Device Multi-Unicasting for Video Streaming*. Lavian T.; Wang P.; Durairaj R.; Hoang D.; Travostino F.; Telecommunications, 2003. ICT 2003. 10th International Conference on Telecommunications, Tahiti, Volume 2, 23 Feb.-1 March, 2003 pp. 1441-1447.
- The SAHARA Model for Service Composition Across Multiple Providers. Raman B.; Agarwal S.; Chen Y.; Caesar M.; Cui W.; Lai K.; Lavian T.; Machiraju S.; Mao Z. M.; Porter G.; Roscoe T.; Subramanian L.; Suzuki T.; Zhuang S.; Joseph A. D.; Katz Y.H.; Stoica I.; Proceedings of the First International Conference on Pervasive Computing. ACM Pervasive 2002, pp. 1-14.
- *Enabling Active Flow Manipulation in Silicon-Based Network Forwarding Engines*. Lavian T.; Wang P.; Travostino F.; Subramanian S.; Duraraj R.; Hoang D.B.; Sethaput V.; Culler D.; Proceeding of the Active Networks Conference and Exposition, 2002.(DANCE) 29-30 May 2002, pp. 65-76.
- *Practical Active Network Services within Content-Aware Gateways*. Subramanian S.; Wang P.; Durairaj R.; Rasimas J.; Travostino F.; Lavian T.; Hoang D.B.; Proceeding of the DARPA Active Networks Conference and Exposition, 2002.(DANCE) 29-30 May 2002, pp. 344-354.
- *Active Networking on a Programmable Network Platform.* Wang P.Y.; Lavian T.; Duncan R.; Jaeger R.; Fourth IEEE Conference on Open Architectures and Network Programming (OPENARCH), Anchorage, April 2002.
- *Intelligent Network Services through Active Flow Manipulation*. Lavian T.; Wang P.; Travostino F.; Subramanian S.; Hoang D.B.; Sethaput V.; IEEE Intelligent Networks 2001 Workshop (IN2001), Boston, May 2001.
- Intelligent Network Services through Active Flow Manipulation. Lavian T.; Wang P.; Travostino F.; Subramanian S.; Hoang D.B.; Sethaput V.; Intelligent Network Workshop, 2001 IEEE 6-9 May 2001, pp.73 - 82.
- *Enabling Active Flow Manipulation in Silicon-based Network Forwarding Engine*. Lavian, T.; Wang, P.; Travostino, F.; Subramanian S.; Hoang D.B.; Sethaput V.; Culler D.; Journal of Communications and Networks, March 2001, pp.78-87.
- *Active Networking on a Programmable Networking Platform*. Lavian T.; Wang P.Y.; IEEE Open Architectures and Network Programming, 2001, pp. 95-103.
- *Enabling Active Networks Services on a Gigabit Routing Switch*. Wang P.; Jaeger R.; Duncan R.; Lavian T.; Travostino F.; 2nd Workshop on Active Middleware Services, 2000.

- *Dynamic Classification in Silicon-Based Forwarding Engine Environments*. Jaeger R.; Duncan R.; Travostino F.; Lavian T.; Hollingsworth J.; Selected Papers. 10th IEEE Workshop on Metropolitan Area and Local Networks, 1999. 21-24 Nov. 1999, pp.103-109.
- *Open Programmable Architecture for Java-Enabled Network Devices*. Lavian, T.; Jaeger, R. F.; Hollingsworth, J. K.; IEEE Hot Interconnects Stanford University, August 1999, pp. 265-277.
- *Open Java SNMP MIB API*. Rob Duncan, Tal Lavian, Roy Lee, Jason Zhou, Bay Architecture Lab Technical Report TR98-038, December 1998.
- *Java-Based Open Service Interface Architecture*. Lavian T.; Lau S.; BAL TR98-010 Bay Architecture Lab Technical Report, March 1998.
- *Parallel SIMD Architecture for Color Image Processing*. Lavian T. Tel – Aviv University, Tel – Aviv, Israel, November 1995.
- *Grid Network Services, Draft-ggf-ghpn-netservices-1.0*.George Clapp, Tiziana Ferrari, Doan B. Hoang, Gigi Karmous-Edwards, Tal Lavian, Mark J. Leese, Paul Mealor, Inder Monga, Volker Sander, Franco Travostino, Global Grid Forum(GGF).
- *Project DRAC: Creating an applications-aware network.*Travostino F.; Keates R.; Lavian T.; Monga I.; Schofield B.; Nortel Technical Journal, February 2005, pp. 23-26.
- *Optical Network Infrastructure for Grid, Draft-ggf-ghpn-opticalnets-1.*Dimitra Simeonidou, Reza Nejabati, Bill St. Arnaud, Micah Beck, Peter Clarke, Doan B. Hoang, David Hutchison, Gigi Karmous-Edwards, Tal Lavian, Jason Leigh, Joe Mambretti, Volker Sander, John Strand, Franco Travostino, Global Grid Forum(GGF) GHPN Standard GFD-I.036 August 2004.
- *Popeye - Using Fine-grained Network Access Control to Support Mobile Users and Protect Intranet Hosts.*Mike Chen, Barbara Hohlt, Tal Lavian, December 2000.

Presentations and Talks

(Not an exhaustive list)

- Lambda Data Grid: An Agile Optical Platform for Grid Computing and Data-intensive Applications.
- Web Services and OGSA
- WINER Workflow Integrated Network Resource Orchestration.
- Technology & Society.
- Abundant Bandwidth and how it affects us?
- Active Content Networking(ACN).
- DWDM-RAM:Enabling Grid Services with Dynamic Optical Networks .
- Application-engaged Dynamic Orchestration of Optical Network Resources .
- A Platform for Data Intensive Services Enabled by Next Generation Dynamic Optical Networks .
- Optical Networks.
- Grid Optical Network Service Architecture for Data Intensive Applications.
- Optical Networking & DWDM.
- OptiCal Inc.
- OptiCal & LUMOS Networks.
- Optical Networking Services.
- Business Models for Dynamically Provisioned Optical Networks.
- Business Model Concepts for Dynamically Provisioned Optical Networks.
- Optical Networks Infrastructure.
- Research Challenges in agile optical networks.
- Services and Applications' infrastructure for agile optical networks.
- Impact on Society.
- TeraGrid Communication and Computation.
- Unified Device Management via Java-enabled Network Devices.
- Active Network Node in Silicon-Based L3 Gigabit Routing Switch.
- Active Nets Technology Transfer through High-Performance Network Devices.
- Programmable Network Node: Applications.
- Open Innovation via Java-enabled Network Devices.
- Practical Considerations for Deploying a Java Active Networking Platform.
- Open Java-Based Intelligent Agent Architecture for Adaptive Networking Devices.
- Java SNMP Oplet.
- Open Distributed Networking Intelligence: A New Java Paradigm.
- Open Programmability.
- Active Networking On A Programmable Networking Platform.
- Open Networking through Programmability.
- Open Programmable Architecture for Java-enabled Network Devices.

- Integrating Active Networking and Commercial-Grade Routing Platforms.
- Programmable Network Devices.
- To be smart or not to be?