

UNITED STATES PATENT AND TRADEMARK OFFICE

---

BEFORE THE PATENT TRIAL AND APPEAL BOARD

---

ZOHO CORPORATION and  
ZOHO CORPORATION PVT., LTD.  
Petitioner,

v.

MEETRIX IP, LLC,  
Patent Owner.

---

U.S. Patent No. 9,094,525

---

**DECLARATION OF HENRY H. HOUH, PH.D., UNDER 37 C.F.R. § 1.68 IN**  
**SUPPORT OF PETITION FOR *INTER PARTES* REVIEW OF**  
**U.S. PATENT NO. 9,094,525**

## **I. INTRODUCTION**

1. I have been retained by the petitioner, Zoho Corporation, to provide my expert opinions as to certain issues in connection with a petition for Inter Partes Review (IPR) in the United States Patent and Trademark Office challenging certain claims of U.S. Patent Nos. 9,094,525 (“the ’525 patent”), entitled “Audio-Video Multi-Participant Conference Systems Using PSTN and Internet Networks.”

2. This Declaration is made based on my personal knowledge, expertise, and experience and based on my review of the materials cited herein. If required, I would testify competently and truthfully regarding the contents of this Declaration.

3. I have no personal, financial, or other commercial interest in the outcome of this IPR. Aside from this consulting engagement, I have no relationship with the petitioner.

4. I am being compensated for my work in this matter at an hourly rate. I am also being reimbursed for reasonable and customary expenses associated with my work and testimony in this matter. My compensation is not contingent on the outcome of this matter or the specifics of my testimony. I have previously been retained as an expert witness in over 100 cases.

## **II. SUMMARY AND MATERIALS CONSIDERED**

5. I have been requested to provide my expert opinions regarding the multi-participant conferencing technology that is the subject matter of the ’525

patent, including how a person of ordinary skill in the art (POSITA) would have interpreted the '525 patent and the prior art I discuss herein.

6. I have also been asked to provide my opinions regarding the validity of claims 1-20 of the '525 patent ("Challenged Claims"), including what a POSITA as of the earliest possible priority date of the '525 patent would have found obvious or not obvious in light of the state of the art and the cited prior art references. It is my opinion that the Challenged Claims would have been obvious to a POSITA after reviewing the prior art discussed below.

7. In preparing this Declaration, I have considered my own knowledge and experience, including my work experience in the field of networking systems, including audio/video conferencing systems, and my experience in working with others involved in this field.

8. In preparing this Declaration, I have studied the following references:

- the '525 patent (Ex. 1001)
- U.S. Patent No. 7,180,997 to Knappe ("Knappe") (Ex. 1006);
- U.S. Patent No. 6,690,654 to Elliott, *et al.* ("Elliott") (Ex. 1007);
- U.S. Patent No. 7,089,285 to Drell ("Drell") (Ex. 1008);
- U.S. Patent No. 6,215,515 to Voois, *et al.* (Ex. 1038, "Voois"); and
- "MPLS and VPN Architectures, A practical guide to understanding, designing and deploying MPLS and MPLS-enabled VPNs," by Guichard et al., Cisco Press, 2001 ("VPN Textbook"), excerpt (Ex. 1010).

9. In forming the opinions expressed in this Declaration I have also considered the following:

- the prosecution history of the '525 Patent (Ex. 1005);
- U.S. Patent No. 3,060,265 (Ex. 1012);
- U.S. Patent No. 4,139,731 (Ex. 1013);
- U.S. Patent No. 3,601,530 (Ex. 1014);
- U.S. Patent No. 4,796,293 (Ex. 1015);
- RFC 760 (Ex. 1028);
- RFC 791 (Ex. 1029);
- RFC 1112 (Ex. 1030);
- RFC 2401 (Ex. 1031);
- RFC 2406 (Ex. 1032);
- A Primer on the T.120 Series Standard (Ex. 1016);
- T.120 Standard (1996) (Ex. 1017);
- U.S. Patent No. 6,237,025 (Ex. 1018);
- U.S. Patent No. 6,782,413 (Ex. 1019);
- U.S. Patent No. 6,980,526 (Ex. 1020);
- U.S. Patent Pub. No. 2004/0261116 (Ex. 1023);
- "Cisco Packetized Voice & Data Integration," Robert Caputo, McGraw-Hill, 2000 (Ex. 1021);
- "The Internet Multicast from ITS: How it was Done and Implications for the Future," by de Moraes *et al.*, IEEE Communications Magazine, January 1995 (Ex. 1022);

- Computer Desktop Encyclopedia (Ex. 1027);
- U.S. Patent No. 6,850,496 (Ex. 1024);
- U.S. Patent No. 6,553,515 (Ex. 1025);
- U.S. Patent No. 6,633,848 (Ex. 1026);
- H.323 Standard (Ex. 1033);
- U.S. Patent No. 7,266,091 (Ex. 1034);
- U.S. Patent No. 5,606,550 (Ex. 1035); and
- any additional documents discussed in my Declaration below.

10. I reserve the right to respond to further comments or questions regarding this IPR in order to clarify or supplement this Declaration.

### **III. QUALIFICATIONS**

11. My qualifications and professional experience are described in my *curriculum vitae*, a copy of which is submitted as Ex. 1004. The following is a brief summary of my relevant qualifications.

12. I received a Ph.D. in Electrical Engineering and Computer Science from the Massachusetts Institute of Technology (“MIT”) in 1998. I also received a Masters of Science degree in Electrical Engineering and Computer Science in 1991, a Bachelor of Science degree in Electrical Engineering and Computer Science in 1989, and a Bachelor of Science degree in Physics in 1990, all from MIT.

13. I am currently self-employed as an independent technical consultant. From its founding in 2012 until October 2022, I was also President of Einstein's Workshop, which provides supplemental science, technology, engineering, and mathematics ("STEM") education to children of all ages. I am also a member of the board of a spin-off of Einstein's Workshop, BlocksCAD Inc., which provides an education-focused 3D computer-aided design tool with over 275,000 users worldwide.

14. I first worked in the area of telecommunications in 1987 when I worked as a summer intern at AT&T Bell Laboratories as part of a five-year dual degree program at MIT. I continued to work at AT&T Bell Laboratories as part of this MIT program.

15. Later, as part of my doctoral research at MIT from 1991-1998, I was a research assistant in the Telemedia Network Systems ("TNS") group at the Laboratory for Computer Science. The TNS group built a high-speed gigabit network and created applications that ran over the network. Example applications included remote video capture, video processing, and display of video on computer terminals. In addition to working on the design of core network components, designing and building the high-speed links, and designing and writing the device drivers for the interface cards, I also set up the group's web server.

16. The TNS group was the first group to initiate a remote video display over the web. I helped to build the web pages that initiated these video sessions via a web interface. Vice-President Al Gore visited our group in 1996 and received a demonstration of – and remotely drove – a radio-controlled toy car with a wireless video camera mounted on it; the video was encoded by TNS-designed hardware, streamed over the TNS-designed network and displayed using TNS-designed software.

17. We also configured and maintained all the networking components in our research group, including configuring the host computers, occasionally configuring routers, and at one point setting up a multicast router on the Internet's multicast backbone.

18. I submitted and defended my Ph.D. thesis, titled "Designing Networks for Tomorrow's Traffic," in January 1998. As part of my thesis research, I analyzed local-area and wide-area flows to show a more efficient method for routing packets in a network, based on traffic patterns at the time. My thesis also addressed real-time communications of audio and video streams and proposed more efficient methods to route such data within a network. The flow classification methods I analyzed are similar to modern label switching techniques such as Multiprotocol Label Switching (MPLS).

19. From 1997 to 1999, I was a Senior Scientist and Engineer at NBX Corporation, a start-up that made business telephone systems for streaming packetized audio over data networks instead of using traditional telephone lines. NBX was later acquired by 3Com Corporation, and the phone system is still used today by numerous businesses. As part of my work at NBX, I designed the core audio reconstruction algorithms for the telephones, as well as the packet transmission algorithms. I also designed and validated the core packet transport protocol used by the phone system. The protocol was used for all signaling in the phone system, including for the setup of conference calls. The NBX system also featured a computer interface for initiating phone calls and conference calls. Conference calls in the NBX system utilized conference mixing software which ran on every telephone in the system. Participants in the conference call, which could include telephones in the PSTN participating through a gateway, multicast their audio packets onto the packet network. While the NBX system was initially based on the Ethernet, I implemented the prototype IP stack for the system, and demonstrated our IP telephones connecting through a VPN connecting from various trades shows through the Internet to our office in Andover, MA. I configured all the routing and VPN appliances involved in any such public demonstrations. The NBX system also supported TAPI, the Telephony Application Programming Interface, thus allowing other computer programs to

integrate with the system's telephony features. We obtained U.S. Patent No. 6,697,963, titled "Telecommunication method for ensuring on-time delivery of packets containing time-sensitive data," as a result of part of this work.

20. From 1999-2004, I was employed by Empirix or its predecessor company, Teradyne. Empirix was a leader in test tools for telecommunications protocols and systems, providing functional testing tools as well as load testing tools. From 2000-2001, I conceived of and built a test platform for testing Voice-over-IP (VoIP). The first application on this new test platform was a cloud emulator for simulating the effects of transmitting VoIP over a busy network.

21. From 2004 to 2008, I was employed by BBN Technologies Corp., a technology research and development company located in Cambridge, MA. BBN Technologies is a world-renowned company with expertise in acoustics, speech recognition, and communications technology. BBN Technologies staff have pioneered many internetworking technologies and Internet applications, and built some of the world's largest government and commercial data networks.

22. My duties and responsibilities at BBN Technologies generally included commercialization of the technologies developed by the company, which included spinning off companies and growing commercial businesses in-house. More particularly, I was involved in utilizing the award-winning AVOKE STX speech recognition technology to create the public audio/video search engine

EveryZing (formerly known as PodZinger) which was spun out into a stand-alone company now known as RAMP, Inc. PodZinger won the 2006 MITX Technology Award for best Web 2.0 Application and was also named the 2006 Forbes Favorite Video & Audio Search Engine, beating out Google, Yahoo, and other companies. After managing the creation of the initial prototype system, PodZinger built out a full streaming audio and video search solution when I was the Vice President of Operations and Technology there. I was also involved in the Boomerang Mobile Shooter Detection project as the Vice President of Engineering for the program. The Boomerang system was deployed to Iraq and Afghanistan, and was credited with saving many lives. I also worked closely with Steve Kent, who was a Chief Scientist at BBN and one of the authors of IPsec, on several projects.

23. I have been awarded several United States patents, and I have several patent applications pending including the following examples:

- U.S. Patent No. 7,975,296, “Automated security threat testing of web pages”;
- U.S. Patent No. 7,877,736, “Computer language interpretation and optimization for server testing”;
- U.S. Patent No. 7,801,910, “Method and apparatus for timed tagging of media content”;
- U.S. Patent 7,590,542, “Method of Generating Test Scripts Using a Voice-Capable Markup Language”;
- U.S. Patent No. 6,967,963, “Telecommunication method for ensuring on-time delivery of packets containing time-sensitive

data”;

- U.S. Patent Publication No. 20070106685, “Method and apparatus for updating speech recognition databases and reindexing audio and video content using the same”;
- U.S. Patent Publication No. 20070106693, “Methods and apparatus for providing virtual media channels based on media search”;
- U.S. Patent Publication No. 20070106760, “Methods and apparatus for dynamic presentation of advertising, factual, and informational content using enhanced metadata in search-driven media applications”;
- U.S. Patent Publication No. 20070112837, “Method and apparatus for timed tagging of media content”;
- U.S. Patent Publication No. 20070118873, “Methods and apparatus for merging media content”;
- U.S. Patent Publication No. 20090222442, “User-directed navigation of multimedia search results.”

#### **IV. LEVEL OF ORDINARY SKILL IN THE ART**

24. I understand that the level of ordinary skill may be reflected by the prior art of record, and that a person of ordinary skill in the art (“POSITA”) to which the claimed subject matter pertains would have the capability of understanding the scientific and engineering principles applicable to the pertinent art.

25. I understand there are multiple factors relevant to determining the level of ordinary skill in the pertinent art, including (1) the levels of education and experience of persons working in the field at the time of the invention; (2) the

sophistication of the technology; (3) the types of problems encountered in the field; and (4) the prior art solutions to those problems.

26. As discussed above and detailed in my *curriculum vitae*, I am familiar with the conferencing art that is pertinent to the '525 patent. I am also aware of the state of the art as of the earliest claimed priority date. Through my past interaction with work colleagues in industry and in academia, I am also familiar with a POSITA's skill level as of the earliest claimed priority date.

27. I have been informed by counsel that the earliest claimed priority date for the '525 patent is March 10, 2003. Based on the technologies disclosed in the '525 patent, I believe that a POSITA would include someone who had, at the earliest claimed priority date of the '525 patent, a working knowledge of available audio and video conferencing technologies. A POSITA would also have had a bachelor's degree in electrical engineering, computer science, computer engineering, or an equivalent, and three years of professional experience relating to conferencing systems in packet-based networks. However, I recognize that someone with less technical education but more experience, or more technical education but less experience could also have met this standard. I believe that I possessed at least such experience and knowledge as of the earliest claimed priority date of the '525 patent, and that I am qualified to opine on the subject matter in the '525 patent.

28. For purposes of this Declaration, in general, and unless otherwise noted, my statements and opinions, such as those regarding my experience and the understanding of a POSITA generally (and specifically related to the references I consulted herein), reflect the knowledge that existed in the field before the earliest claimed priority date of the '525 patent.

**V. RELEVANT LEGAL STANDARDS**

29. I understand that I am obliged to follow existing law as applicable to the opinions set forth herein. I am not an attorney but have been informed of certain legal principles by attorneys in preparation of this Declaration. I have been asked to apply the legal principles set forth below to my analysis. I understand these legal principles and have applied them in my analysis in this Declaration.

**A. Prior Art**

30. I understand that the applicable patent law provides categories of information that constitute prior art that may be used to anticipate or render obvious patent claims. To qualify as prior art to a particular patent claim under the relevant law, I understand that a reference must have been made, known, used, published, or patented, or be the subject of a patent application by another, before the priority date of the patent, and, in this matter, must satisfy one of the standards of pre-AIA 35 U.S.C. § 102. I also understand that applicable patent law provides

that an admission in a challenged patent also qualifies as prior art that can be used as evidence of obviousness under pre-AIA 35 U.S.C. § 103.

**B. Claim Construction**

31. In making this Declaration, I have been asked to review the terms in the Challenged Claims to determine what a POSITA would have understood those terms to mean at the time of the alleged invention. I understand that my assessment of the Challenged Claims must be undertaken from the perspective of what would have been known or understood by a POSITA reading the '525 patent as of the time of the filing or earliest claimed priority date (i.e., March 10, 2003), and in light of the specification and file history of the patents. In addition, I understand that, in an *inter partes* review proceeding, the claim terms under consideration are to be construed using their ordinary and customary meaning, as would have been understood by a POSITA at the time of the alleged invention, consistent with the specification and the prosecution history of the patent.

**C. Obviousness**

32. I also understand that a patent claim is invalid if the differences between the claimed subject matter and the prior art are such that the subject matter as a whole would have been obvious to a POSITA at the time the invention was made.

33. I understand further that the condition “at the time the invention was made” is imposed to rule out impermissible hindsight. Accordingly, permissible reasoning takes into account only knowledge which was within the level of ordinary skill in the art at the time the claimed invention was made and does not include knowledge gleaned only from the applicant’s disclosure. I also understand that an opining expert is obligated to analyze prior art from the perspective of a POSITA, and may not simply provide his or her own personal conclusions.

34. I also understand that an obviousness determination includes several factual inquiries, including: (1) determining the scope and content of the prior art; (2) ascertaining the differences between the claimed invention and the prior art; (3) resolving the level of ordinary skill in the pertinent art; and (4) taking into consideration any secondary indicia (objective evidence) of non-obviousness.

35. I also understand that such objective evidence of non-obviousness may include: (1) a long felt but unsolved need that was satisfied by the claimed invention; (2) commercial success attributable to the claimed invention; (3) unexpected results achieved by the claimed invention; (4) praise by experts of the claimed invention with factual support; (5) taking of licenses under the patent by others for reasons related to the alleged non-obviousness of the claimed invention; (6) the failure of others; (7) the teaching away of others; (8) skepticism by experts;

and (9) evidence that competitors in the marketplace have copied the invention instead of using the prior art.

36. I also understand that, in order to establish nonobviousness, there must be a demonstrated relationship, or nexus, between any such secondary indicia and the claimed invention, i.e., there must be objective evidence of non-obviousness attributable to the claimed invention.

37. I further understand that near simultaneous invention by two or more equally talented inventors working independently may or may not be an indication of obviousness when considered in light of all the circumstances.

38. I understand that a conclusion of obviousness can be based on a combination of multiple prior art references. I understand that examples of rationales that may support a conclusion of obviousness include:

- a. Combining prior art elements according to known methods to yield predictable results;
- b. Simply substituting one known element for another to obtain predictable results;
- c. Using known techniques to improve similar devices, methods, or products in the same way;
- d. Applying a known technique to a known device, method, or product ready for improvement to yield predictable results;

- e. Using an “obvious to try” solution, that is, choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success
- f. Applying known work in one field of endeavor to yield variations of that work for use in either the same field or a different one based on design incentives or other market forces if the variations are predictable to one of ordinary skill in the art; and
- g. Applying some teaching, suggestion, or motivation in the prior art that would have led one of ordinary skill to modify the prior art reference or to combine prior art reference disclosures to arrive at the claimed invention.

39. I also understand that if a technique has been used to improve one device, and if a POSITA would recognize that the technique would improve similar devices in the same way, then using that technique would be obvious to a POSITA unless actual application of that technique is beyond his or her skill.

40. I also understand that practical and common-sense considerations should guide an analysis of obviousness, because familiar items may have obvious uses beyond their primary purposes. I further understand that applying common sense does not require a “specific hint or suggestion in a particular reference.”

Rather, applying common sense requires a reasoned explanation that avoids false deductions.

41. I also understand that, when addressing a problem, a POSITA will often be able to fit together the disclosures of multiple publications. In this regard, I understand that an analysis of obviousness may consider inferences and creative steps that a POSITA would normally employ.

42. I also understand that a particular combination of prior art items may be proven obvious merely by showing that it was obvious for a POSITA to try that combination. For example, when there is a design need or a market pressure to solve a problem, and/or there exists a finite number of identified, predictable solutions with a reasonable expectation of success, a POSITA has good reason to pursue the known options within his or her technical grasp. I understand that if this leads to anticipated success, that success is likely not the result of innovation but of ordinary skill and common sense.

43. I also understand that a combination according to known methods of familiar elements is presumed to be obvious when that combination does no more than yield predictable results. Work known in one field of endeavor may prompt variations of that work for use in either the same field or a different field, based on design incentives and other market forces. If a technique has been used to improve

some device, and a POSITA can implement a predictable variation of that technique, then that variation is likely obvious.

44. I have been informed and understand that obviousness does not require physical combination of components or bodily incorporation of one device into another, but rather consideration of what the combined teachings would have suggested to a POSITA at the time of the alleged invention.

45. I further understand that to be proper for use in an analysis of obviousness, a reference must teach art that is analogous to the claimed invention. I also understand that under a correct analysis, any need or problem known in the field of endeavor at the time of the invention and addressed by the claimed invention can provide a reason for combining the elements in the manner claimed.

46. I also understand further that, in an analysis of obviousness, two items of prior art need not be like two puzzle pieces that must fit together perfectly. For example, a claimed invention may be found obvious if a POSITA would view rearrangement as an obvious design choice.

47. I also understand that an analysis of obviousness requires a comparison between the properly construed claim language and the prior art on a limitation-by-limitation basis, while observing whether the differences between the claimed subject matter and the prior art are such that the subject matter as a whole would have been obvious to a POSITA at the time the invention was made.

## **VI. OVERVIEW OF THE '525 PATENT**

48. The '525 patent issued as a continuation of a chain of applications that claim priority to Provisional Application No. 60/453,307, which was filed on March 10, 2003. Ex. 1001 at 1:7-13.

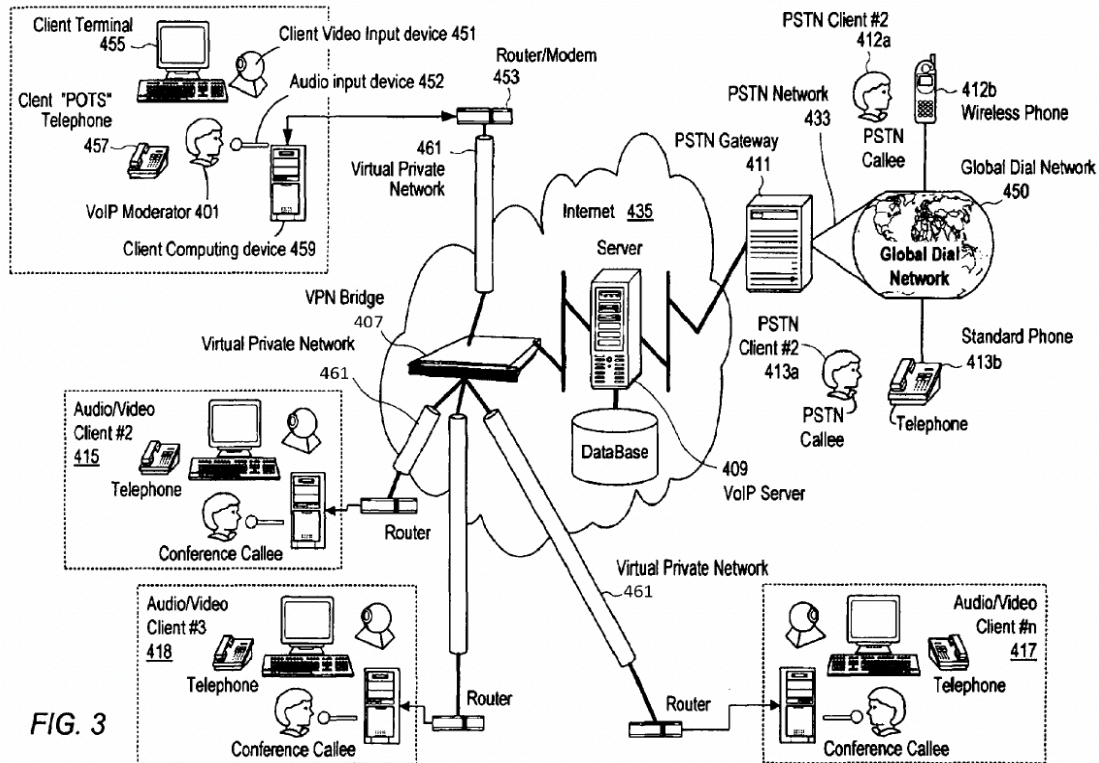
### **A. The '525 Patent Specification**

49. The '525 patent is directed to a “a multi-participant” conferencing system that “includes a PSTN client, at least one remote client and a first participant client.” *Id.* at Abstract. “The PSTN client communicates audio data and the remote client communicates audio-video data.” *Id.* The Abstract explains that the “first participant client includes a voice over IP (VoIP) encoder, a VoIP decoder, a first audio mixer, and a second audio mixer” to compress, decode, mix, and transport data between the PSTN, remote, and first participant clients. *Id.*; *see also id.* at 3:22-4:7.

50. To accomplish this, the specification describes use of a hybrid network connecting a moderator, several remote Internet-based audio video clients, and one or more PSTN clients. *Id.* at FIG. 3, 4:56-5:35. Specifically, the patent describes having a moderator that “dial[s] out from a desktop computer or terminal (using a novel hybrid network structure) connecting an external telephone user’s audio into the audio/video conference.” *Id.* at 4:60-64.

51. The hybrid network structure is depicted in Figure 3 of the '525

patent:



*Id.* at FIG. 3.

52. Specifically, Figure 3 depicts Internet-based clients, including moderator 401, and audio video clients 415, 417, and 418 “connected through routers or modems 453 preferably in a virtual private network configuration 461” with “the VPN bridge 407” located in the center of a “star topology,” with connections to the VPN bridge serving as a “VPN tunnel.” *Id.* at 5:24-49, 6:37-59, FIGS. 4-5. The specification explains that the virtual private network is used to connect the local and remote Internet-based clients together within a secure private network. *Id.* at 5:27-28.

53. The hybrid network also includes “telephony clients using standard wired 413 or wireless telephone 412 systems,” with these PSTN clients connected to the Internet-based clients via PSTN gateway 411. *Id.* at 5:16-18.

54. A connection from the VPN bridge 407 to the a “voice over IP server 409 is used to transfer conference audio from any participant on the IP network to any participant in the PSTN.” *Id.* at 5:27-31. “The voice over IP server 409 is responsible for transcoding audio information from the virtual private network 461 to and from the PSTN gateway 411, thus bridging the PSTN and VPN together. *Id.* at 5:32-35. Specifically, the ’525 patent discloses that “voice over IP server 409 may use standard H.323 or SIP network protocol to establish communications” between PSTN and Internet clients. *Id.* at 7:64-67.

55. The specification also discloses performing encoding and decoding of audio data using codecs of the prior art H.323 standard and the transmission of collaboration data per the prior art T.120 standard. *Id.* at 2:57-63, 3:12-14, FIG. 2. The specification describes application software used to mix and transmit audio in the hybrid network. *See id.* at FIG. 6, 7:38-9:67. Specifically, it describes mixing audio received from the PSTN with audio from the moderator, explaining that audio encoder 520a local to the moderator “combines the PSTN client 412 audio with the local moderator clients 401 audio then encodes the result for compression of the data in preparation across the VPN networks 461” which then “delivers both

the PSTN client's audio and the local moderator client's audio to remote audio video clients 418.” *Id.* at 9:10-20.

56. The patent specification further describes the process of combining audio from the moderator with audio from remote clients, explaining the “remote audio video clients 418 audio is mixed with the local moderator client audio 401 in preparation for compression by the VoIP encoder 522.” *Id.* at 9:34-37. The mixed audio data is then “placed in virtual private network tunnel for transport to the voice over IP server 409 and onto the gateway for audio presentation to the PSTN[.]” *Id.* at 9:37-40.

57. The specification also explains that “local moderator client 401 receives audio from PSTN client 412 . . . and mixes audio previously decoded by the voice over IP decoder 524 with the audio from the remote client's 418 for presentation to the local speaker 454.” *Id.* at 9:21-25.

## **B. The Challenged Claims**

58. Claims 1-20 are the Challenged Claims of the '525 patent. Claims 1, 9, and 17 are independent claims. Independent claim 1 is directed to a system supporting a multi-participant conference that recites:

- a “first mixer” that mixes a PSTN client audio stream with a moderator audio-video data stream into a first mixed data stream;
- a “first transport output” that transmits the first mixed data stream to at least one remote client that receives the first mixed data stream, the

at least one remote client communicatively coupled to the Internet, which generates a remote client audio-video data stream;

- a “second mixer” that mixes the moderator audio-video data stream with the remote client audio-video data stream into a second mixed data stream; and
- a second transport output that transmits a mixed audio data stream, corresponding to the second mixed data stream, to the PSTN client.

*Id.* at 10:2-17.

59. Claim 2 further adds the a “third mixer” that mixes the audio data from the PSTN client with the audio-video data from the remote client into a third mixed data stream and communicates it to the moderator. *Id.* at 10:18-22.

60. Claims 3-8 add further known limitations, such as having a VoIP decoder, audio decompressor, and using a VPN tunnel. *Id.* at 10:23-43.

61. Independent claim 9 is a computer-readable medium claim reciting instructions for a multi-participant conference process executed by a local moderator computer, comprising:

- receiving audio data from a telephone through a PSTN;
- receiving audio and video data from a first remote client through the Internet;
- receiving audio and video data from a second remote client through the Internet;
- receiving audio from a local audio input device communicatively coupled to the local moderator computer;
- receiving video from a local video input device communicatively coupled to the local moderator computer;

- mixing the audio from the telephone with the audio from the local audio input device and the video data from the local video input device into mixed audio-video data;
- transmitting the mixed audio-video data to the first remote computer and the second remote computer;
- mixing the audio data from the first remote computer and the audio data from the second remote computer with audio from the local audio input device into mixed audio data;
- compressing the mixed audio data;
- transmitting the compressed mixed audio data to the telephone;
- transmitting the collaboration data to the first remote computer and the second remote computer.

*Id.* at 10:44-11:6.

62. Claim 10 further requires mixing audio from the telephone with audio from the first and second remote computers into second mixed data. *Id.* at 11:7-10. Claims 11-12 and 15-16 add the further known limitations of receiving IP packets, decoding the IP packets, and encoding the data before transmission. *Id.* at 11:11-17, 11:24-30. Claims 13-14 add the requirement of transmitting data through a VPN. *Id.* at 11:18-23.

63. Independent claim 17 is also a system claim requiring use of a local moderator computer, the system comprising:

- a “voice over IP decoder” configured to decode audio from a telephone through the PSTN;
- a “voice over IP encoder” configured to compress audio transported from the local moderator computer to the telephone over the PSTN;

- a “first mixer” configured to mix decoded audio from the PSTN with audio from a local audio input device and video from a local video input device into an audio-video stream from transmission through the Internet to a first remote computer and a second remote computer;
- a “second mixer” configured to mix audio from the first remote computer and the second remote computer with audio from the local audio input device into an audio stream for transmission to the telephone;
- a “data collaboration unit” configured to transmit collaboration data from the local moderator computer to the first remote computer and the second remote computer.

*Id.* at 11:31-12:23.

64. Claim 18 further requires a third mixer configured to mix decoded audio from the telephone with audio received from the first and second remote computers. *Id.* at 12:24-27. Claims 19-20 further require use of a VPN tunnel. *Id.* at 12:28-35.

## **VII. CLAIM CONSTRUCTION**

65. It is my understanding that in order to properly evaluate the ’525 Patent, the terms of the claims must first be interpreted.

66. It is my understanding that for purposes of *Inter Partes* Review, the claim terms must be construed according to their ordinary and customary meaning as would have been understood by one of ordinary skill in the art. I have also been informed that claim terms need to be construed only to the extent necessary to resolve the obviousness inquiry. Below I provide a construction for the claim term “*mixed data stream*.”

67. For the remaining claim terms, I have analyzed the claims consistent with their ordinary and customary meaning as would have been understood by a POSITA.

68. Claim 1 recites a “*first mixed data stream*” and a “*second mixed data stream*.” The “*mixed data stream*” claimed in the ’525 patent is produced by mixing an input data stream from a different client with a “*moderator audio-video data stream*.”

69. This understanding is confirmed by other language in the ’525 patent claims. For example, claim 1 refers to a “*mixed audio data stream, corresponding to the second mixed data stream*.” This language suggests that the mixed audio data stream is a subset of the mixed data stream. If the “*second mixed data stream*” did not include both audio and video data (e.g., if it included only audio data), then the separately recited “*mixed **audio** data stream*” would have the same meaning as, and thus be redundant of, the “*second mixed data stream*.”

70. Claim 6 further shows that each claimed “*mixed data stream*” includes at least audio and video data. Claim 6 requires “*encod[ing] the second mixed data stream*” and, separately, “*encod[ing] the second mixed voice data*.” Thus, claim 6 further confirms that the “*second mixed data stream*” is distinct from simply mixed voice data (i.e., only audio data).

71. Claim 9 also supports the claim interpretation. Notably, claim 9 includes two separate “*mixing*” steps. In the first mixing step, two “*audio data*” inputs and one “*video data*” input are mixed “*into mixed audio-video data.*” In contrast, the second mixing step includes three “*audio data*” inputs that are mixed “*into mixed audio data.*” Thus, the language of claim 9 confirms that mixing inputs that include both audio and video data produces a mixed data output also including both audio and video data.

72. This interpretation is also consistent with the ’525 patent specification, which refers to a “mix of audio and video data.” Ex\_1001 at 1:51. Furthermore, claim 6 recites, for example, “*transmitting the second mixed data stream to the PSTN [public switched telephone network] client.*” The ’525 patent specification describes providing “audio, **video**, and data **connections** between clients conferencing on the Internet **and clients conferencing on standard telephone systems.**” *Id.* at 4:64-67.

73. Accordingly, as would have been understood by a POSITA in light of the ’525 patent specification, the meaning of “*mixed data stream*” is a “data stream including at least audio data and video data.”

## **VIII. SCOPE AND CONTENT OF THE PRIOR ART**

### **A. Audio, Video, and Collaboration Data Conferencing Technology**

74. The prior art in the field of multi-party conferencing is very extensive. Prior to March 10, 2003 – the earliest claimed priority date for the '525 patent – multiparty conferencing, including mixing and transmitting audio, video, and collaboration data over hybrid networks, was already well known.

75. Multiparty audio telephone conferencing is an age-old feature from the days of circuit-switched analog telephony. For example, U.S. Patent No. 3,060,265 (the “'265 patent”), titled “Conference Call Circuit” was issued on October 23, 1962, and describes an invention that “relates to telephone systems and more particularly to conference call circuits for use in such systems.” '265 patent (Ex. 1012) at 1:11-13. These conference systems would have conference participants’ calls delivered to a “conference bridge” device that would mix all the audio and then deliver the mixed audio back to all the participants.

76. For example, U.S. Patent No. 4,139,731 (the “'731 patent”), titled “Telephone conference system with active analog conference” was issued on February 13, 1979, and discloses that “[a] telephone conference system includes a conference bridge for coupling multiple conference ports associated with corresponding conference stations. The bridge includes means for summing

incoming signals from all the conference ports and supplying the summed signals to all the conference ports.” ’731 patent (Ex. 1013) at Abstract.

77. Video conferencing systems were also well known before the ’525 patent, having existed for decades prior. For example, U.S. Patent No. 3,601,530 (the “’530 patent”), titled “Video conference system using voice- switched cameras,” was issued on August 24, 1971, describes “vi[su]al telephone systems . . . for conference connecting two or more groups of remote conferees in a manner which approaches a true face-to-face conference situation.” ’530 patent (Ex. 1014) at 1:6-10. By using cameras at two locations, “a video conference can be provided for two groups of remotely located conferees” such that the function “approaches a true face-to-face conference situation,” with the benefit of “greatly reducing travel” cost. *Id.* at 1:6-60.

78. The conferencing art that existed prior to the ’525 patent also recognized the benefit of facilitating exchange of collaboration data such as written or graphic material. ’530 patent (Ex. 1014) at Abstract; *see also* U.S. Patent No. 6,237,025 (the “’025 patent”) (Ex. 1018) at 1:28-47 (“In addition to spoken words, demonstrative gestures and behavioral cues, collaboration often involves the sharing of visual information—e.g., printed material such as articles, drawings, photographs, charts and graphs, as well as video tapes and computer-based animations, visualizations and other displays—in such a way that the participants

can collectively and interactively examine, discuss, annotate and revise the information. This combination of spoken words, gestures, visual cues and interactive data sharing significantly enhances the effectiveness of collaboration in a variety of contexts.”); *see also* Elliott (Ex. 1007) at 1:40-44, 5:49-56, 6:53-54.

79. It was also known in the art that participants in a conferencing system can include a leader (which may be referred to as a moderator) that can “a substantial degree of control over the conference...analogous to that appropriated by a leader in a true face-to-face conference situation” so as to more effectively conduct the conference. ’530 patent (Ex. 1014) at 2:34-40, Abstract, 17:20-41 (claims 7-11), 18:30-49 (claims 21-24); *see also* Knappe (Ex. 1006) at 8:52-56.

80. There were also many well-known methods used for initiating a conference call. For example, U.S. Patent No. 4,796,293 (the “’293 patent”), titled “Enhanced dedicated teleconferencing system” was issued on January 3, 1989, and explains that participants may be pre-selected, and “[a]ll conferees can be simultaneously dialed out to by the bridge.” ’293 patent (Ex. 1015) at 5:40-51. In another example, a four-digit code number could be specified, and “[i]n this type of conference, the conferees dial into [] the conference bridge and enter the specified four-digit touch tone code when prompted by a synthesized voice recording which the bridge generates. The touch tone code is used to instruct the bridge as to which conference this heretofore unidentified caller is to be connected.” *Id.* at 5:64-6:4.

81. It was also known that a telephone user may use features built into a PBX system to initiate a conference call. For example, it was known that a telephone user can call someone, place that callee on hold, dial another user, then join the two callees together for a three-way conference. Additional participants may be added in a similar way, up to the limit of the system. I personally, before the earliest claimed priority date of the '525 patent, worked on the NBX 100 system, which had this method of adding a callee to a conference call.

**B. Evolution of the Telephone System from Analog to Digital**

82. One important factor in unleashing the expansion of a myriad of features in the telephone network was the migration from analog circuits to digital circuits and the accompanying digitization of the signaling network.

83. Through the conversion of the analog circuit network to a digital circuit network, the conference bridge remained a complex device that remained centrally located in the network. However, the conversion to digital allowed much of the functionality of a conference bridge to be implemented in software, and allowed the ability to develop more advanced features more rapidly than in an analog world. The telephony network further evolved before the earliest claimed priority date of the '525 patent, as the implementation of IP-based telephony to the endpoint began to be commercialized and as network bandwidth to the residence increased dramatically with the rollout and deployment of cable modems.

### **C. Circuit-Switched Networks**

84. In a circuit-switched network, the entire path through a network is determined and reserved prior to any communications occurring on the circuit, and the path remains fixed for the duration of any communications, until the circuit is closed or “torn down.” A circuit can be thought of as a continuous wire between two parties through which data is transmitted. In fact, in the early days of the telephone system, a series of electrical switches and interconnecting wires did create a continuous wired connection between parties.

85. As the network evolved to a digital network, the concept of a circuit remained, but no longer were entire wires dedicated to a single circuit. Digital representation of information allowed wires to be shared by sending data at regular intervals interleaved with data from other users using the notion of a regular time slot allocated to different “virtual circuits.” For example, 24 voice circuits are bundled into a standard T-1 digital trunk line, where a byte of data from each circuit is bundled together in a frame, and 8,000 frames are transmitted each second. This frame rate and sample size defines the standard unit of carriage in a digital telephone network, a channel with a bit rate of 64 kilobits per second. A specific slot in a frame on a trunk represents a single circuit, for example, the 5th slot in every frame carries data for a single circuit. A route for a virtual circuit still must be determined and reserved ahead of time prior to any communications

occurring, and this involved determining free time slots on every trunk between the endpoints. The time slot allocated to the virtual circuit may also change from trunk to trunk, but overall the allocated time slots on the particular trunks remain fixed until the virtual circuit is torn down.

86. Once the route is determined and the virtual circuit is set up, data sent on the virtual circuit is transmitted through the various reserved time slots on the trunks and arrives at the far end of the virtual circuit. The data need no identifiers as to which virtual circuit they are assigned. The virtual circuit is known by the network based on the appropriate time slot assigned to the virtual circuit for each trunk.

87. The most common and widely deployed example of a circuit-switched network is the public switched telephone network (PSTN).

#### **D. Packet-Switched Networks**

88. Packet-switching was independently proposed by Paul Baran and Donald Davies in the 1960's. Packet-switching was a different concept than traditional circuit-switching. In both analog and digital circuit switching, the circuit defined the communications endpoints and the entire path through the network is set up prior to communications occurring. The circuit's path is fixed throughout the duration of the call.

89. In packet switching, the data to be communicated is broken up into chunks, each chunk being a “packet,” with each packet bearing a packet header that includes the address of the source and the address of the destination. As the packet traverses the network, each node in the network (which is connected to multiple links to other nodes) examines the destination address and makes a local independent decision on the best next hop in the network. This is known as the “routing” step, and the sending of the packet down the selected link to the next hop is known as “forwarding.”<sup>1</sup> These nodes are known as “routers.” The routers receive updates from neighbors about the changing conditions and loads within the network, and update the data structures, known as “routing tables” which the router uses to determine the next hop. This allows packets from a particular source to a particular destination to travel different paths through the network, based on changing conditions in the network. While some people take this to mean that every packet travels a different path, this is not true – network paths *may* change, but that does not mean they change continuously. Network routing path updates are typically not a frequent occurrence compared the interval between packets in a data stream.

---

<sup>1</sup> Often the combined act of making the decision on where to forward the packet and forwarding the packet are collectively referred to as “routing.”

90. The principles set forth by Baran and Davies were adopted in the ARPANET, a project funded by the U.S. government in the late 1960's through the Advanced Projects Research Agency (ARPA). The contract to build the ARPANET was awarded to Bolt, Beranek and Newman (BBN) in 1969. I later joined BBN in 2004 and, in my roughly five years working at BBN, discussed project ideas with several of the engineers that worked on the original ARPANET and later phases of the project.

91. BBN designed and built the Interface Message Processors (IMP) that performed the similar function in the ARPANET that today's routers in a data network perform.

#### 1. **Internet Protocol**

92. A specification of the Internet Protocol titled "DoD Standard Internet Protocol," was prepared for the Defense Advanced Research Projects Agency (DARPA) in 1980, long before the patent at issue. RFC 760 (Ex. 1028) at Cover; *see also* Declaration of Ingrid Hsieh-Yee, Ph.D. ("Yee Decl.") (Ex. 1011) ¶¶ 16, 23-28. RFC 760 specified the length of an address to be 32 bits, or four bytes. Initially, the first byte specified the network number and the remaining three bytes were the local address within the network. This provided for only 256 distinct networks, and this method of addressing networks was later modified several times. RFC 760 also specified a 20-byte header for every packet, which included a

source address, a destination address, and other fields. Thus, every IP packet contains a destination IP address in the header, which is used by the routers to determine the best next hop as the packet travels through the network.

93. RFC 760 was later obsoleted by RFC 791 in 1981, titled “Internet Protocol – DARPA Internet Program – Protocol Specification” and is one of the few documents made a formal Internet Standard as STD 5. Addresses remained 4 bytes (as IPv4 addresses remain today), but RFC 791 changed the concept of network number to be more flexible by introducing the notion of Class A, B, and C network numbers. RFC 791 (Ex. 1029) at 7; *see also* Yee Decl. (Ex. 1011) ¶¶ 16, 23, 29-32. Later updates removed this notion to provide classless address space.

## **2. Unicast and Multicast**

94. The addresses in IP initially identified individual host computers. RFC 791 at 1. These are known as “unicast” addresses as they identify a single host.

95. Several later RFCs introduced and refined the notion of a “multicast” address. RFC 966 was published in December 1985, and was followed by RFC 988 in July 1986, RFC 1054 in May 1988, and RFC 1112 in August 1989. RFC 1112 was made part of STD 5. Part of the IP address space was set aside for exclusive use as multicast address space. RFC 1112 (Ex. 1030) at 3; *see also* Yee

Decl. (Ex. 1011) ¶¶ 16, 23, 33-36. The IP address range from 224.0.0.0 to 239.255.255.255 is reserved for use as multicast address space.

96. Unlike a unicast address which specifies a single host, a multicast address specifies a “host group,” a set of zero or more hosts identified by a single IP address in the multicast address range. RFC 1112 (Ex. 1030) at 1. The multicast-enabled network makes its best effort to deliver any packet sent to a multicast address to all the hosts in the host group. The membership of a host group is dynamic, with hosts able to join or leave at any time, and hosts may be anywhere on the network. The routers in the network enabled to handle IP multicast addresses are known as multicast routers. Multicast routers maintain the notion of the group and where members of the group reside with respect to the various output ports of the router. The Internet Group Management Protocol (IGMP) is used by hosts to signal to their membership status (e.g., joining a group or leaving a group) to the hosts’ neighboring multicast routers. *Id.* at 11. When there are multicast group members that have joined a particular group that can be reached at a particular output port of the multicast router, the router forwards multicast packets addressed to that group to all ports for which it knows there are group members reachable. When all the hosts reachable through a specific output port leave a multicast group, there is no need for a multicast router to forward packets to that output port any more.

97. Routers in the Internet are not typically enabled to route multicast packets. Packets sent into the Internet to an IP multicast address via the same means as unicast packets are sent will not, typically, be further routed through the Internet, as the Internet is not multicast-enabled. However, it was well known to create a multicast network “overlay” on top of the Internet, through the use of “tunnels,” as discussed below.

98. Multicast is not limited to an IP network. Other layers of the networking stack also incorporate this concept. For example, Ethernet, a link-layer protocol, also incorporates the notion of multicast, but typically multicast delivery is limited to Ethernet segments as bounded by IP routers.

### **3. Tunneling**

99. Tunneling refers to a networking technique whereby a packet is encapsulated in another network packet, typically so that it can traverse an underlying network that does not support the capability required by the encapsulated packet. IP in IP tunneling is described in RFC 1853 which was published in the mid-1990s. Tunneling can be performed in both directions. At the input end of the tunnel, packets are encapsulated, or wrapped, in a way where the source address of the encapsulating header is that of the encapsulating device and the destination address of the encapsulating header is the far end of the tunnel. At the far end of the tunnel, the packets are stripped of the encapsulating header, or

unwrapped, and the original packet emerges for further actions such as routing to its destination using the original packet's header.

100. IP in IP tunneling is used in the MBone, a multicast network overlay on top of the Internet, and IPSec, a method for providing secure encrypted communications over the Internet (or any other unsecured or public network). Both these uses are discussed below.

#### 4. **MBone**

101. To multicast on the Internet, host computers connected to the Internet are configured as multicast routers, and tunnels between the hosts acting as multicast routers are created. The tunnels are unicast connections between the multicast routers. *See* de Moraes, Luis F.M., "The Internet Multicast from ITS: How it was Done and Implications for the Future," *IEEE Communications Magazine*, January 1995 (Ex. 1022) at 6; *see also* Yee Decl. (Ex. 1011) ¶¶ 16, 23, 70-78. Thus, the connected hosts operating as multicast routers are performing the multicast routing. This is known as an "overlay" network. Such an overlay network was implemented over the Internet in the mid 1990's, and was known as the "MBone," short for multicast backbone.

102. To participate in multicast using the MBone, a regular host on the Internet sends multicast group join requests to a multicast router, and then the multicast router will forward multicast packets to the host. Similarly, to multicast

to a group of hosts, a host sends an IP multicast packet to a multicast router with the destination address being the address of the group of hosts.

103. When I was at MIT in the 1990's, our research group implemented a multicast router as part of the MBone using a computer in our group. We participated in various events that were multicast on the MBone, including IETF meetings.

## 5. IPsec

104. RFC 2401, titled "Security Architecture for the Internet Protocol" was published in the late 1990s. *See* RFC 2401 (Ex. 1031); *see also* Yee Decl. (Ex. 1011) ¶¶ 16, 23, 37-40. RFC 2401 specifies the architecture of IPsec compliant systems to provide various security services for traffic at the IP layer, in both the IPv4 and IPv6 environments. RFC 2401 describes the components and how they fit together with each other and into the IP environment. It also describes the security services offered by the IPsec protocols, and how these services can be employed in the IP environment. RFC 2401 notes that "[s]ubsequent documents will address additional architectural details of a more advanced nature, e.g., ... more complete support for IP multicast." RFC 2401 (Ex. 1031) at 3. The "subsequent documents" were published contemporaneously with RFC 2401. RFC 2401 discloses two security protocols, Authentication Header (AH) and Encapsulating Security Payload (ESP). AH is further described in RFC 2402,

titled “IP Authentication Header,” and ESP is further described in RFC 2406, titled “IP Encapsulating Security Payload (ESP).” *See* RFC 2406 (Ex. 1032); *see also* Yee Decl. (Ex. 1011) ¶¶ 16, 23, 41-44.

105. ESP may be operated in a “tunnel mode” in which the original packet is encrypted, and the encrypted packet is encapsulated in a new IP header which specifies the endpoints of the tunnel. While host computers may implement ESP in tunnel mode, security gateway appliances must use the tunnel mode. RFC 2406 (Ex. 1032) at 9.

106. A “Security Association” (SA) is a “simplex ‘connection’ that affords security services to the traffic carried by it.” RFC 2401 (Ex. 1031) at 8. Security services are afforded to an SA by the use of AH, or ESP, but not both. If both AH and ESP are used, then two (or more) SAs are created. To secure typical, bi-directional communication, two Security Associations (one in each direction) are required. Thus, when IPsec is used in ESP tunnel mode to secure communications between two devices, such as between a pair of security gateway appliances, a pair of security associations is required (one in each direction). Two simplex (unidirectional) tunnels are established, and together the pair of tunnels form a full-duplex bi-directional encrypted tunnel.

## 6. Virtual Private Network

107. A Virtual Private Network (VPN) is “defined loosely as a network in which customer connectivity amongst multiple sites is deployed on a shared infrastructure with the same access or security policies as a private network.” *See* VPN Textbook (Ex. 1010) at 115; *see also* Yee Decl. (Ex. 1011) ¶¶ 16, 23, 95-111. VPNs were known to be used to securely exchange data between different locations over the Internet. VPN Textbook at 118-119; *see also* U.S. Patent No. 6,980,526 (the “526 patent”) (Ex. 1020) at 2:65-3:3, 5:52-55. It was also known that VPNs may be implemented over the public Internet using IPsec tunnels between sites.

108. It was well known that for companies with multiple sites, the simpler implementation is to deploy a VPN network by using a central router that acts as a hub for other remote site routers. This implementation is commonly referred to as a “hub-and-spoke” or a “star” topology. *See* VPN Textbook (Ex. 1010) at 129; Computer Desktop Encyclopedia (Ex. 1027) at 438; *see also* Yee Decl. (Ex. 1011) ¶¶ 16, 23, 79-94. In contrast, a dense mesh topology of VPN connections between all sites becomes complex to set up and manage, especially as a company expands to add new satellite offices. Using a hub-and-spoke topology for VPN connections makes securely connecting the network of a new satellite office a much simpler process to manage. *See* VPN Textbook (Ex. 1010) at 129.

## 7. Voice over IP (VoIP)

109. Voice over IP (VoIP), refers to voice data (e.g., audio) encapsulated in an IP packet for transmission across an IP network, such as the Internet.

110. When a stream of VoIP packets traverses a network, the interaction with network switches, routers, and other cross-traffic causes the packet latency for each packet in the stream to vary slightly, an effect known as “packet jitter.” Because of packet jitter, the destination receiving node must be coded to deal with the irregular and unpredictable arrival times of the VoIP packets. Typically, packet data are put into the end of a buffer upon arrival, and voice reconstruction occurs with data at the front of a buffer. If the buffer and startup delays are sized appropriately, the buffer is unlikely to be empty when the next byte of voice data is needed to reconstruct the ongoing audio stream. The buffer could be empty because of excessive delays for a particular set of packets in the network (the data didn’t reach the destination by the time it was needed), or a packet simply could have become dropped in the network (often due to congestion in a network). Audio reconstruction algorithms can “conceal” such lost data to an extent, but if the losses become high then users will notice.

111. It was well known to utilize VoIP for conference communications. *See, e.g.,* U.S. Patent No. 6,850,496 (Ex. 1024) at 1:23-28 (“Packet-switched

networks can also carry real time voice data, and therefore, with proper configuration, conference calls. Voice over IP (VoIP) . . . use a packet-switched network for transport of voice data.”); Cisco Packetized Voice & Data Integration, Robert Caputo (Ex. 1021) at 23; *see also* Yee Decl. (Ex. 1011) ¶¶ 16, 23, 53-69. I note that I personally, before the earliest claimed priority date of the ’525 patent, designed a packet voice/audio reconstruction algorithm and coded a prototype of the algorithm for the NBX 100 telephones. Also, before the earliest claimed priority date of the ’525 patent, when I was employed at Empirix, I conceived of and designed a new product for testing the quality of VoIP calls. The product was known as a cloud emulator, which varied the jitter of packets traversing the tool as well as the packet loss, allowing testing of the voice reconstruction algorithms in gateways. We sold the product to manufacturers of large VoIP gateways.

112. Also, it was well known that because many of the standard codecs do not specify security features, VoIP calls (including the calls into a conference bridge) are vulnerable to being reconstructed by anyone that has access to the packets carrying the audio data, such as when the VoIP data stream is sent unprotected over the Internet. It was well known for employees at large companies, with multiple sites, to utilize video conferencing as a way to avoid the expense of holding in-person meetings. However, because private leased data lines between a company’s sites are much more expensive than direct connections to the

Internet at each site, companies frequently utilized VPNs for encrypting site-to-site communications over the Internet, thereby providing the security of a private leased line without the ongoing expense of one. *See, e.g.*, U.S. Patent No. 6,782,413 (Ex. 1019) at 2:66-3:1, 4:23-34, 6:36-54; '526 patent (Ex. 1020) at 2:65- 3:3, 5:52-55. It was also known to utilize VPN tunneling for voice, video, and collaboration data during a conference. *See* U.S. Patent Publication No. 2004/0261116 (Ex. 1023) at ¶¶[1412]-[1414], [1451]-[1453]; Elliott (Ex. 1007) at 2:66-3:1, 4:29-34, 6:53-54.

## **8. Audio and Video Compression**

113. Audio and video compression were well known concepts, where “codecs” specify algorithms for the coding (the “co” of codec) and decoding (the “dec” of codec) of audio and video data. *See, e.g.*, Cisco Packetized Voice & Data Integration, Robert Caputo (Ex. 1021) at 12-13; U.S. Patent No. 6,850,496 (Ex. 1024) at 3:56-67. Various codecs provide different levels of compression and different levels of perceived quality. G.711 is an early standard for an audio codec, and was available in 1972. G.711 is used in H.323. G.711 specifies that audio be sampled at an 8 kHz rate with 8 bit samples, and thus the bitrate of the digitized encoded audio is 64 kilobits/second. As discussed above, the architecture of the digital PSTN is based around this fundamental base rate of digitized speech. Other voice codecs were known to further reduce the bitrate, and lower bitrate codecs were frequently used in networks to conserve bandwidth.

114. The '525 patent incorporates by reference the entire H.261 and H.263 specifications which specify video codecs. *See* '525 patent (Ex. 1001) at 4:32-49. Video codecs specify the algorithms by which video is compressed, and include references to audio encoding standards, since audio is part of the video. Another well-known video codec standard is MPEG-2. Because most video contains moving objects between frames (as opposed to sudden cuts between scenes which occurs less frequently) MPEG uses the concept of motion compensation to increase the compression. During compression, objects in a current frame can be encoded based on a pointer to the same object in an adjacent frame (either before or after the current frame) without having to re-encode the entire visual representation of that same object in the current frame. Small differences in the object also can be encoded. Only at specified intervals is the entire frame encoded based only on the contents of the current frame. Most video codecs utilize this concept.

#### **9. A Gateway between PSTN and IP Networks**

115. As business and consumer VoIP products and services began to be introduced in the late 1990's and early 2000's, it was recognized as important for VoIP devices to be able to communicate with traditional PSTN telephones, not just with other VoIP telephones. For this reason, gateways were widely used to facilitate communications between the VoIP devices and the PSTN telephones. *See, e.g.,* Knappe (Ex. 1006) at FIG. 1. As I mentioned above, the NBX 100

system had a gateway to the PSTN as part of the NBX 100 system itself. Initially, we shipped only a gateway that connected to a conventional analog telephone line, for bridging between our packet voice system and an analog line. In subsequent releases, but prior to the earliest claimed priority date of the '525 patent, we released a line card that interfaced to a digital T1 line.

116. Commercial VoIP providers, such as Vonage, operated larger gateways in the network itself to interconnect its voice customers to the PSTN. These types of carriers were referred to as “over the top” carriers as their VoIP traffic traversed the customers’ existing Internet connection to reach the gateways of the service provider.

117. Large gateways often were partitioned into parts, with one component for handling the audio conversion between networks, often called the “media gateway,” and another component for handling the signaling conversion between networks, often called the “signaling gateway.”

## **10. VoIP for Conferencing**

118. As VoIP’s popularity increased, it was well known to implement VoIP-based conference systems using software-based digital network conference bridges, including replacing the digital circuit interfaces with interfaces to the data network and the software to decode various standard codecs to a standard internal representation

119. It was also known to implement conference bridges towards the edges of a network and even in the endpoints. When conference calls were mixed locally in the endpoint, audio data from the participant was delivered to all the participating endpoints. *See, e.g., Drell (Ex. 1008) at 4:21-31.*

120. From 1997-1999, I worked at a company that designed and shipped a business telephone system private branch exchange (PBX) that worked over a company's existing data network. The product, called the NBX 100, was a full featured business PBX, where every device in the system was connected to the data network via an Ethernet connection, which is almost universally the type of connection a computer uses to connect to the network. A gateway card interconnected the system to the PSTN for inbound and outbound calls. Every telephone and gateway in the system utilized a relatively inexpensive digital signal processor (DSP). Every telephone and gateway in the system was capable of performing a four-party conference call through mixing the audio received from each of the other three callers (which could be a mix of internal callers and external callers via a gateway) using software running on each telephone's DSP. There was no centralized conference bridge in the system. For larger conference calls, that is over four parties, an external conference bridge was used (although a six-party conference could be formed when two different telephones which were each part of a three-party conference joined together in a conference).

## **E. Conferencing Standards**

121. The H.323 standard was first published by the International Telecommunications Union (ITU) in 1996, and focused on video conferencing over a packet network. H.323 encompassed a set of standards for providing audio and video conferences on a packet network, including signaling to set up a conference as well as the compression and transport of media including audio and video. *See, e.g., Cisco Packetized Voice & Data Integration*, Robert Caputo (Ex. 1021) at 23.

122. H.323 is regarded by many as the standard for VoIP communication, and also specifies compression protocols. *See, e.g., U.S. Patent No. 6,553,515* (Ex. 1025) at 4:35-56 (“H.323 is adapted to support many aspects of multimedia...[and] generally is regarded as the standard for VoIP...This standard also specifies lowest-common denominator compression protocols, such as G.711...G.723 and G.729.”). H.323 also provides for initial call setup, including determining support for full-duplex. *See, e.g., Id.* at 4:61-67 (“H.323 requires that Internet telephones also adhere to H.245. . . . During call initiation, the devices use H.245 to determine proper compression protocols and also whether the phones support full-duplex or half-duplex transmission.”).

123. The H.323 standard recommended the use of collaboration data per the T.120 standard, which was established by the ITU in 1996. *See, e.g., A Primer on the T.120 Series Standard* (Ex. 1016); *see also Yee Decl.* (Ex. 1011) ¶¶ 16, 23, 45-

48; T.120 Standard (Ex. 1017) at 8; *see also* Yee Decl. (Ex. 1011) ¶¶ 16, 23, 49-52; '525 patent (Ex. 1001) at 3:12-14. The T.120 standard contains a series of application protocols (e.g., T.122, T.123, T.124, and T.125) for collaboration data such as multipoint file transfer, still image viewing and annotation, shared white board, and facsimile. *See* A Primer on the T.120 Series Standard (Ex. 1016).

## **IX. INVALIDITY ANALYSIS**

124. As noted previously, I have been asked to provide my opinion as to whether the Challenged Claims (claims 1-20) of the '525 patent would have been obvious in view of the prior art. The discussion below provides a detailed analysis of how the prior art references I reviewed teach the limitations of the Challenged Claims of the '525 patent and render the Challenged Claims obvious to a POSITA.

125. As part of my analysis, I have considered the scope and content of the prior art and any potential differences between the claimed subject matter and the prior art. I conducted my analysis as of the earliest claimed priority date of the '525 patent: March 10, 2003. I have also considered the level of ordinary skill in the pertinent art as of that date.

126. I describe in detail below the scope and content of the prior art, as well as any differences between the claimed subject matter and the prior art, on an limitation-by-limitation basis for each Challenged Claim of the '525 patent. This analysis supports my opinion that the differences between the claims of the '525

patent and the prior art discussed herein are such that the subject matter as a whole would have been obvious as of the earliest possible priority date of the '525 patent to a person having ordinary skill in the art to which the subject matter pertains.

127. I will now describe, in the grounds below, on an limitation-by-limitation basis, how the prior art teaches and/or renders obvious all limitations of claims 1-20 of the '525 patent.

**A. Claims 1-4 Are Obvious Over Drell in Combination With Knappe and Voois**

**1. The Prior Art**

**a) Summary of Drell**

128. United States Patent No. 7,089,285 ("Drell") entitled "Videoconferencing Apparatus Having Integrated Multi-Point Conference Capabilities" was filed October 5, 2000, claiming priority to a provisional application filed on October 5, 1999. Drell (Ex. 1008).

129. Like the '525 patent, Drell describes multi-point (MP) conferencing, specifically video conferencing among a plurality of endpoints, as shown in Figure 1:

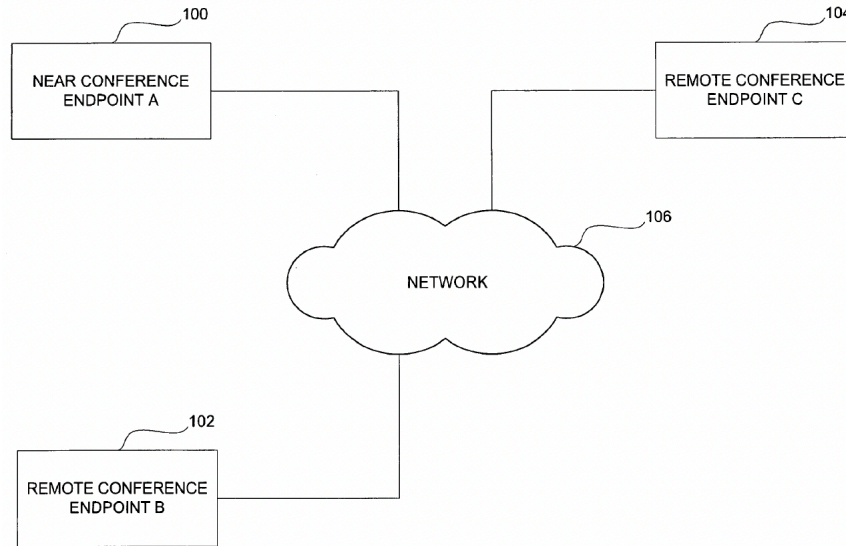
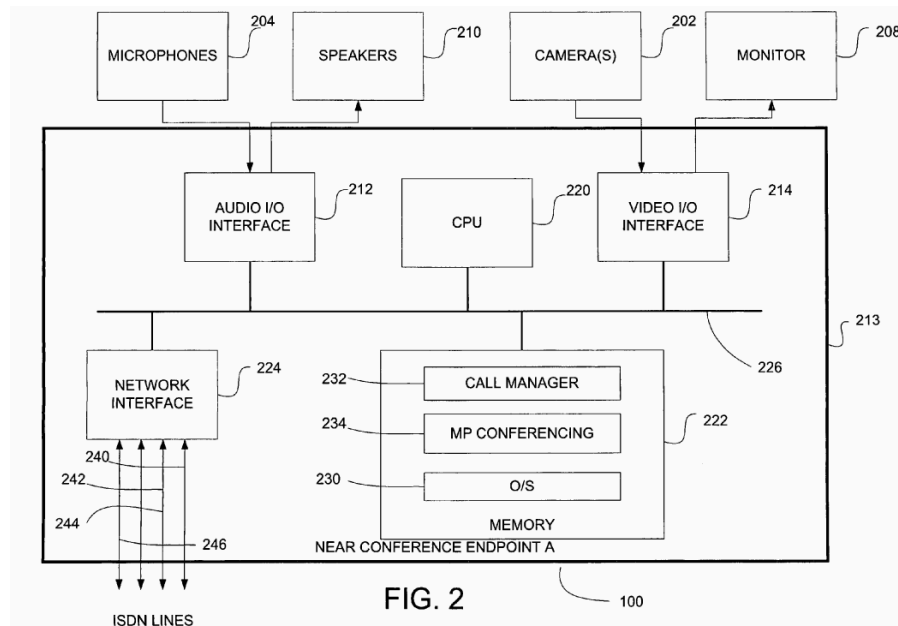


FIG. 1

*Id.* at FIG. 1.

130. Specifically, Drell discloses a videoconferencing apparatus “to combine and distribute audio and video signals received from a plurality of remote conference endpoints[.]” *Id.* at Abstract. Drell explains that “network 106” facilitates the transmission of “audio and video data between and among near conference endpoint 100 and remote conference endpoints 102 and 104.” *Id.* at 3:20-23. Drell’s conferencing system, like the ’525 patent, also can be implemented across a hybrid network that includes the Internet and PSTN. Specifically, Drell notes that “[a]lthough a single network 106 is shown,” Drell “contemplates the use of two or more networks (for example, the PSTN and the Internet) to connect conference endpoints utilizing different communication protocols.” *Id.* at 3:33-38.

131. Drell's system implements a software-based conference manager, explaining that centrally located conference bridges can have significant cost disadvantages. It thus teaches use of "MP conferencing application 234" that is located at a "near conference point 100." *Id.* at 1:5-2:8, 4:6-20. Drell's near conference endpoint 100 is illustrated in Figure 1, below:



*Id.* at FIG. 2.

132. Drell's MP conferencing application 234 is illustrated in Figure 3, reproduced below:

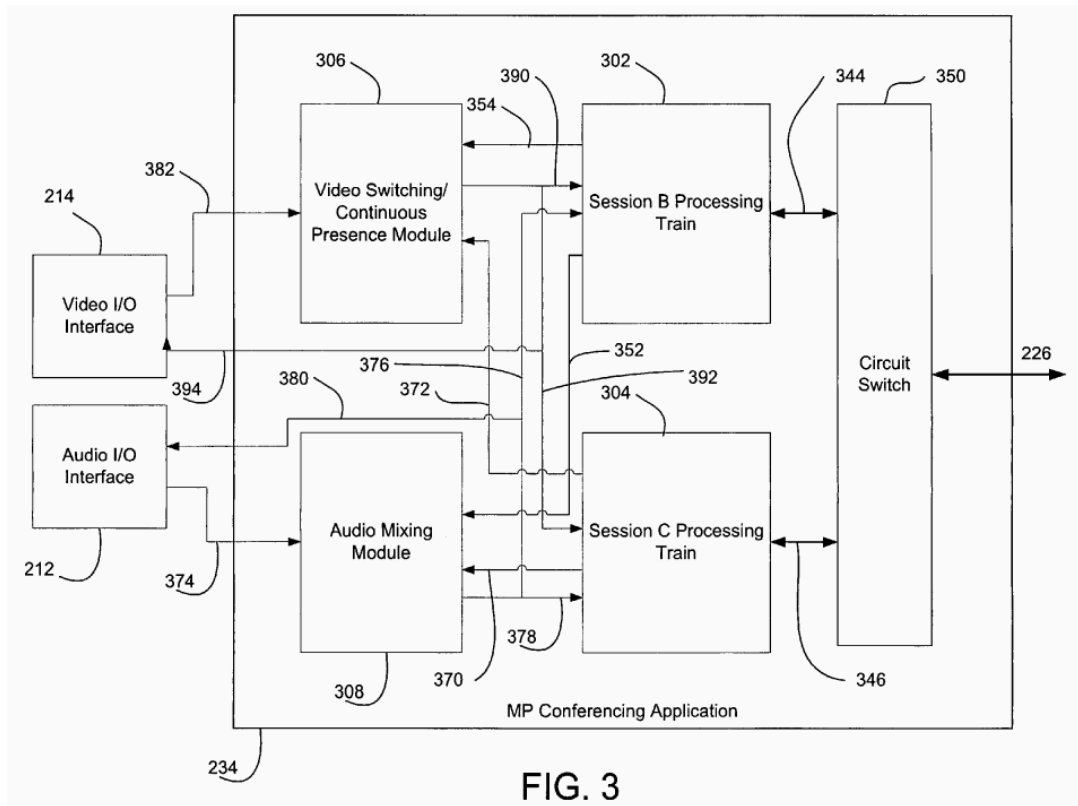


FIG. 3

*Id.* at FIG. 3. Drell explains that “MP conferencing application 234 is configured to instantiate a processing train for each remote conference endpoint” where the “processed audio and video data streams are combined with each other and with locally generated audio and video streams.” *Id.* at 4:21-29. Once combined, the “audio and video streams are thereafter distributed to remote conferencing endpoints 102 and 104.” *Id.* at 4:29-31.

133. Drell further teaches that MP conferencing application 234 uses “a set of codecs” and that the communication process may “comprise an H.320 process (ISDN-based) or H.323 (packet-based) process.” *Id.* at 2:17-33, FIG. 4.

134. To process received audio data, the “[a]udio codec 408 receives the audio data stream ... and applies ... decoding in accordance with a standard ... or proprietary audio compression algorithm.” *Id.* at 6:18-22. The decoded received audio and video data streams “together with the locally generated audio and video signals, are combined at an audio mixer and a video switching/continuous presence module.” *Id.* at 2:17-37. During transmission, “video codec 406 encodes the video data stream output by video switching/continuous presence module 306 (representative, for example, of a “continuous presence” image) using a standard or proprietary video compression algorithm (e.g., H.261) and...[a]udio codec 408 encodes the audio data stream output by audio mixing module 308 (representative, for example, of the blended speech of conference participants located at near conference endpoint 100 and remote conference endpoints 102 and 104) using a standard or proprietary audio compression algorithm.” *Id.* at 6:33-44.

**b) Summary of Knappe**

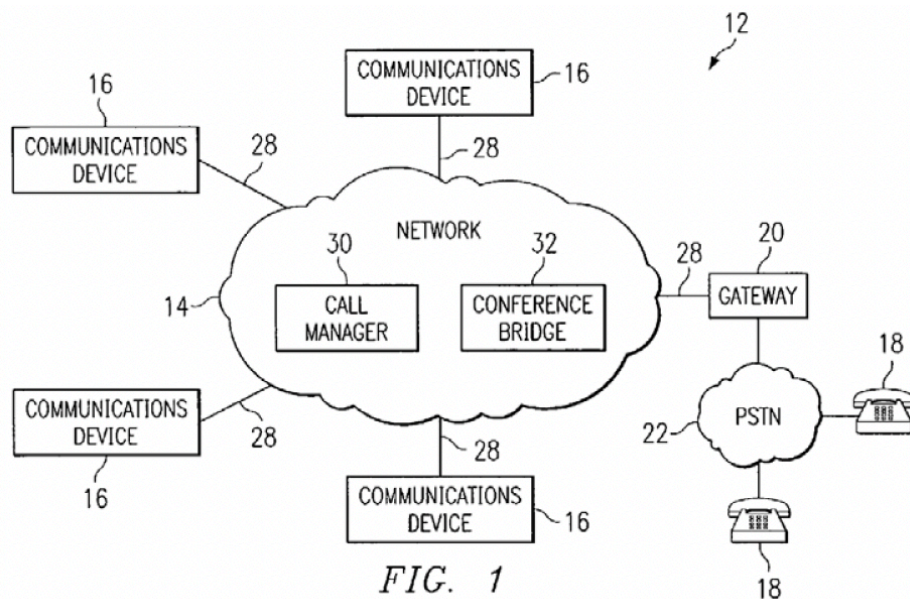
135. United States Patent No. 7,180,997 (“Knappe”) entitled “Method and System for Improving the Intelligibility of a Moderator During a Multiparty Communication Session” was filed September 6, 2002. *See* Ex. 1006.

136. Like the ’525 patent and Drell, Knappe discloses a multiparty conferencing system over a hybrid network that includes a network “connecting a plurality of communication devices 16 to each other and to standard analog

telephones 18 through a gateway 20 and the public switched telephone network.”

*Id.* at 3:41-45; *see also id.* at Abstract, 1:8-11, 8:46-10:50, FIGS. 1-8.

137. Figure 1 of Knappe illustrates the conferencing system, showing multiple conference endpoints, including communication devices 16 and standard analog PSTN telephones 18:



*Id.* at FIG. 1; *see also, id.* at 3:41-4:28. The Knappe system is a “distributed system transmitting audio, video, voice, data, and other suitable types of real-time and/or non real-time traffic between source and destination points.” *Id.* at 3:26-29. As shown in Figure 1, participants in the Knappe conferencing system include “communication devices 16 [that] comprise real-time applications that play traffic as it is received.” *Id.* at 4:5-9.

138. Knappe explains that network 14 can be “the Internet, a wide area network (WAN), a local area network (LAN), or other suitable packet-switched network.” *Id.* at 3:50-52. In the Internet embodiment, Knappe explains that “the network 14 transmits information in Internet Protocol (IP) packets[,]” with telephony voice information “transmitted in the Voice over IP (VoIP) format.” *Id.* at 3:53-56.

139. Knappe further explains:

codec (coder/decoder) converts audio, video or other suitable signals generated by users from analog signals into digital form. The digital encoded data is encapsulated into IP or other suitable packets for transmission over the network 14. IP packets received from the network 14 are converted back into analog signals and played to the user.

*Id.* at 4:5-14. And “at least one participant may be given the priority of moderator for a particular conference call.” *Id.* at 8:52-56.

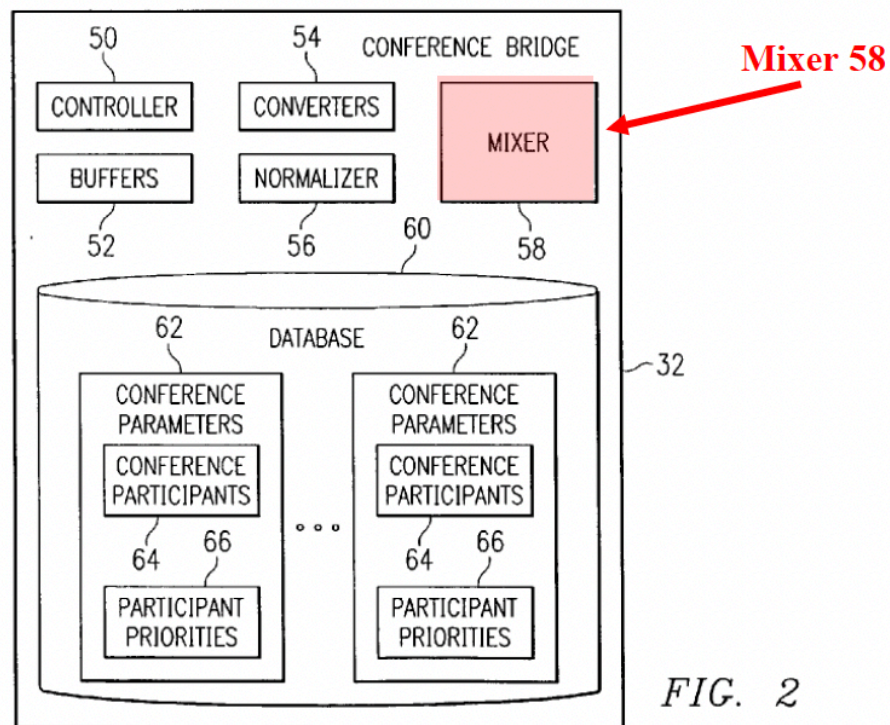
140. The standard analog telephones 18 participating in the conferencing system communicate standard telephony signals through PSTN 22 to the gateway 20.” *Id.* at 4:19-21. Knappe explains that:

[a]t the gateway 20, the signals are converted to IP packets in the VoIP format. Similarly, VoIP packets received from the network 14 are converted into standard telephony signals for delivery to the destination telephone 18 through PSTN 22. The gateway 20 also translates between the network call control system and the Signaling System 7 (SS7) protocol and/or other signaling protocols used in PSTN.

*Id.* at 4:21-28.

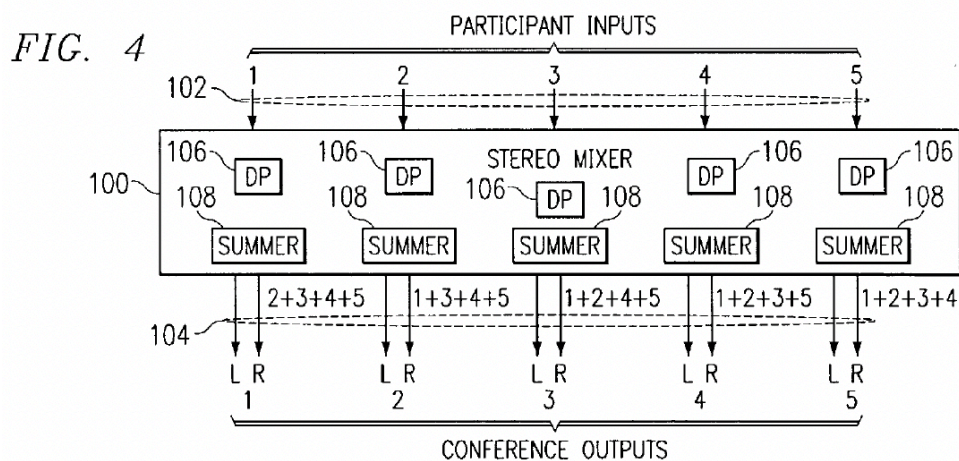
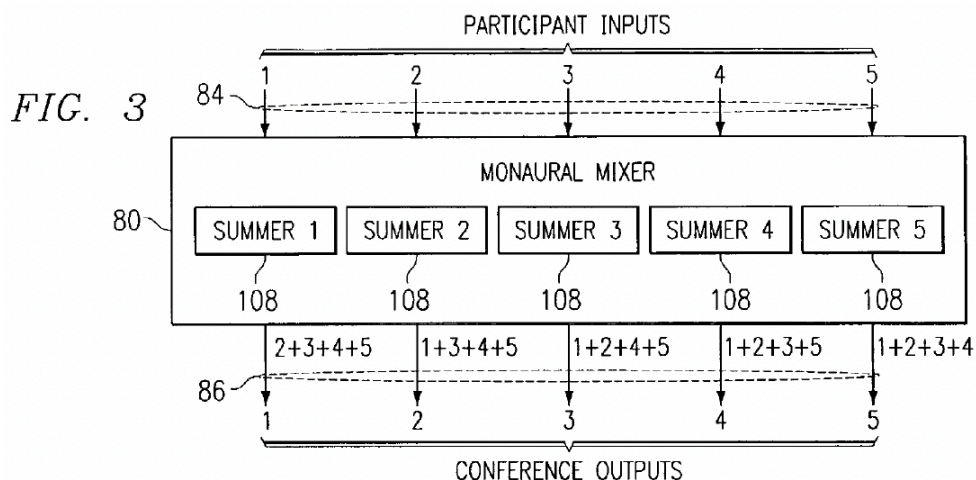
141. The network communication devices 16 and analog telephone 18 “participants are connected and stream media through the conference bridge 32.” *Id.*, 5:32-33. “The conference bridge 32 provides conference call and other suitable audio, video, and/or real-time multi-party communication sessions between communication devices 16.” *Id.* at 4:58-60.

142. As illustrated in Figure 2 below, Knappe’s conference bridge 32 employs mixer 58, which “includes a plurality of summers or other suitable signal processing resources each operable to sum, add or otherwise combine a plurality of input streams into conference output streams for participants to a conference call.” *Id.* at 6:33-37.



*Id.* at FIG. 2 (annotated).

143. Figures 3-4 in Knappe show examples of mixers that “combine[] the streams in summers 82 to generate conference output streams 86 for each participant to a conference call.” *Id.* at 6:60-62 (monaural mixer); *see also id.* at 7:36-39 (stereo mixer).



*Id.* at FIGS. 3-4.

144. In the above figure, “the monaural mixer 80 [of the conference bridge 32] receives participant input streams 84 and combines the streams in summers 82 to generate conference output streams 86 for each participant to a conference call.”

*Id.* at 6:59-62. Knappe further explains that “each participant is assigned a summer 82 that receives audio input streams from each other participant to the conference call” which “combines the audio input streams to generate a conference output stream for delivery to the participant.” *Id.* at 6:62-67.

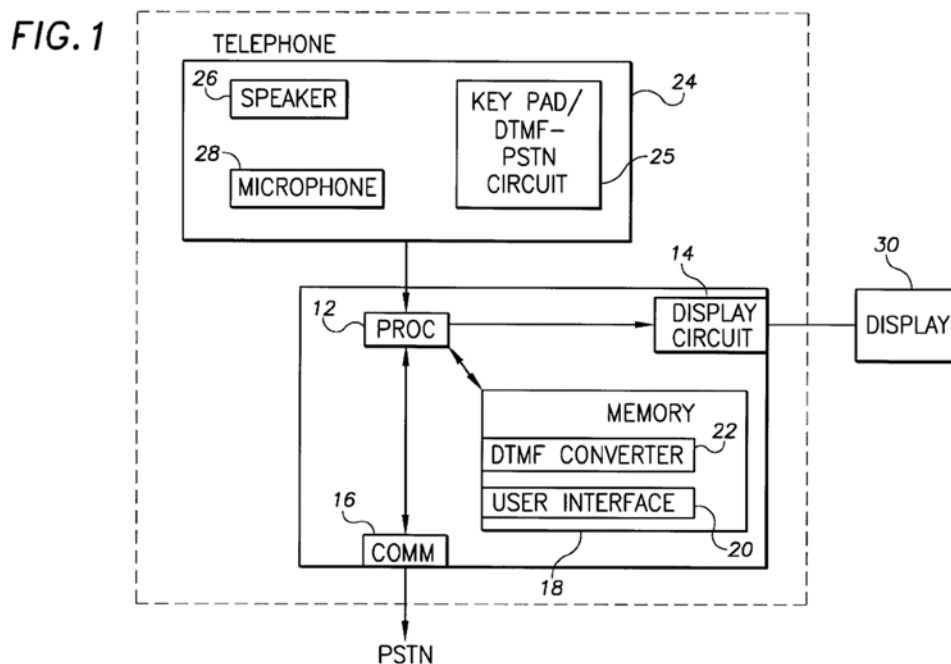
145. In the Knappe system the combined data is then transmitted to each of the other participants, with Knappe explaining that “[d]uring normal operation, each participant receives the audio input of each other participant.” *Id.* at 7:1-2; *see also, id.*, 7:2-15. With respect to the 5-participant example of Figure 3, Knappe explains that “the conference output stream of participant 1 includes the audio inputs of participants 2–5.” *Id.* at 7:2-4. Similarly, the conference output stream of participants 2-5 includes the audio inputs of the other participants. *Id.* at 7:4-15.

### **c) Summary of Voois**

146. U.S. Patent No. 6,215,515 to Voois *et al.* was filed on June 10, 1998, and issued on April 10, 2001. Ex. 1036. Voois is titled “Videocommunicating Device with an On-Screen Telephone Keypad User- Interface Method and Arrangement.” *Id.* at Title. Voois relates to “video communication systems... for videoconferencing over a conventional communications channel.” *Id.* at 1:34-37. Specifically, Voois teaches videophones for communicating “video and audio data

over a plain old telephone service (POTS) line.” *Id.* at Abstract; *see also id.* at 2:39-45.

147. In Figure 1, reproduced below, Voois shows an example of a videophone system “having a telephone for operating a videophone user interface” and a “[c]ommunication component 16 [that] includes hardware along with supporting firmware and/or software... to provide an interface with a communication network, such as a public switched telephone network (PSTN), the internet, local area networks, and/or wide area networks.” *Id.* at 3:62-4:4.



## 2. Reasons to Combine Prior Art

### a) Reasons to Combine Drell and Knappe

148. A POSITA when considering the teachings of Drell would have also considered the teachings of Knappe, as they are analogous prior art both pertaining

to the same field of endeavor of multiparty conferencing, including mixing audio from Internet and PSTN conference endpoints. Drell (Ex. 1008) at Abstract (“A videoconferencing apparatus includes a multi-point (MP) conference application that enables the apparatus to combine and distribute audio and video signals received from a plurality of remote conference endpoints.”); *see also id.* at 3:33-37 (“[T]he invention contemplates the use of two or more networks (for example, the PSTN and the Internet) to connect conference endpoints.”); Knappe (Ex. 1006) at 3:41-51 (“[C]ommunication system 12 includes...network 14 and/or PSTN 22...network 14 and/or PSTN 22 . . . network 14 is the Internet.”); *see also id.* at 6:59-62 (“[M]ixer 80 receives participant input streams 84 and combines the streams...to generate conference output streams 86 for each participant to a conference call.”)

149. A POSITA would have been motivated to combine the teachings of Drell with the teachings of Knappe for multiple reasons, including to produce obvious, beneficial, and predictable results, as discussed below.

(1) Moderator

150. As I discussed above, Drell teaches a conferencing system where audio and video data are transmitted between a “near conference endpoint” and “remote conference endpoints.” Drell (Ex. 1008) at 3:20-23. A POSITA would have been familiar with conferencing systems and would have recognized that it

would be desirable for a person (e.g., a leader or a moderator) to initiate and organize the conference for other participants. Also, persons of skill would have recognized that it is desirable to have a person (e.g., a leader or a moderator) to control the conference. *See, e.g.*, U.S. Patent No. 3,601,530 (Ex. 1014) at 2:34-40 (“[A] conference leader mode of operation is provided. In this mode...the conference leader...[has] a substantial degree of **control over the conference.**”). This would be desirable because frequently in business conferencing, at the time and even now, one person is in charge of the conference.

151. Consistent with this knowledge in the art, Knappe teaches that at least “one participant may be given the priority of moderator for a particular conference call.” Knappe (Ex. 1006) at 8:52-56. According to Knappe, the “designated moderator” is able to “**control** the flow and/or **organization** of the conference.” *Id.* at 1:53-55. Since Knappe teaches that at least one of the participants is a moderator, it would have been obvious to a POSITA for either the user of the remote conference endpoint 104 or near conference endpoint 100 to be a moderator, depending on who initiates the conference. *See, e.g., id.* at claim 1 (“... automatically identifying **a moderator initiating the multi-party communication session.**”). Moreover, designating an endpoint as the moderator would allow for the user (acting as the moderator) to control the flow and organization of the conference. *Id.* at 1:53-55 (“...a designated moderator(s) to **control** the flow

and/or **organization** of the conference.”); *see also id.* at 3:35-37 (“The disclosed embodiments allow the moderator to exercise **control** and **influence** over the telephone conference.”); *see also* U.S. Patent No. 3,601,530 (Ex. 1014) at 2:34-40.

152. Accordingly, it would have been obvious to one of ordinary skill in the art for at least one of Drell’s conference participants, such as the user of the remote conference endpoint or near conference endpoint, to be a moderator per Knappe’s teaching, because doing so would beneficially allow the user (acting as the moderator) to initiate, organize, control, and influence the conference. Furthermore, as there are a finite number of participants in a conference, it would have been obvious to try designating one of Drell’s conference endpoint users as the moderator. The combination of Drell and Knappe is merely the ordinary use of a common technique (identifying the user of the conference endpoint as a moderator per Knappe) in the similar conferencing system of Drell to yield predictable results of initiating, organizing, controlling, and influencing the conference. Each of these reasons separately and together would have motivated a POSITA.

153. The results would have been predictable and there would have been a reasonable expectation of success in the combination given that both Drell and Knappe describe similar conferencing systems. *See, e.g.*, Drell (Ex. 1008) at Abstract; Knappe (Ex. 1006) at Abstract, 3:47-51. Any modification needed to the

teachings of Drell, including implementing software programming or hardware, to accommodate the teachings of Knappe, would have been within the level of ordinary skill in the art. *See, e.g.*, Knappe (Ex. 1006) at 5:49-62, 6:55-58. Indeed, Knappe itself recognizes that its teachings can be applied by POSITA to other network devices. *See, e.g., id.* at 4:30-33 (“The call manager 30 and the conference bridge 32 may be located in a central facility or have their functionality distributed across and/or at the periphery of the network 14.”).

## (2) Mixing

154. As I discussed above, Drell teaches that its conference system “combine[s] and distribute[s] audio and video signals received from a plurality of remote conference endpoints.” Drell (Ex. 1008) at Abstract; *see also id.* at 5:18-20 (“[A]udio mixing module 308 may simply mix the received audio input data streams.”).

155. Notably, Knappe supplements and explains how to mix audio data from multiple conference endpoints. Specifically, in connection with Figure 3, Knappe teaches using a mixer 80 with a plurality of summers (e.g., summers 1 to 5, one for each participant), where each summer mixes audio data from the other participants and also “provide[s] higher prominence to the audio input [] of the conference moderator.” Knappe (Ex. 1006) at 6:59-7:15, FIGS. 1-3.

156. It would have been obvious to a POSITA, when implementing a mixer in Drell, to use a plurality of summers (one for each participant) as Knappe teaches, because this would allow for generating separate audio instances for each conference endpoint. Such an implementation would complement Drell's teaching of generating separate instances for each endpoint. *See* Drell (Ex. 1008) at 2:19-21 ("Generally, the MP conferencing application generates separate processing trains for signal streams from/to each of the remote conference endpoints."); *id.* at 6:66-7:4 ("By dynamically generating a separate instance of a processing train for each remote endpoint session, a videoconferencing system embodying the invention may easily and flexibly accommodate endpoint sessions comprising a range of connection bandwidths and communication protocols.").

157. Further, a POSITA would have applied Knappe's mixing teachings to Drell because using separate summers (one for each participant) would facilitate providing higher prominence to the moderator and "increasing the intelligibility of the moderator relative to the remaining participants." *See, e.g.*, Knappe (Ex. 1006) at 2:30-34, 6:59-7:17; *see also id.* at 2:40-46, 8:10-12. The increased intelligibility of the moderator enhances the moderator's control over the conference. *See, e.g., id.* at 1:59-67.

158. Furthermore, a POSITA would have understood that it is undesirable for a conference participant to receive his or her own audio as part of the mixed

audio. U.S. Patent No. 7,266,091 (Ex. 1034), 6:31-33 (“It is undesirable for a conference participant to receive his or her own audio in the mixed audio stream.”). Including a participant’s own audio in the mixed audio for that participant creates the possibility of audio feedback or echo, which adversely affects the conference experience. It has long been known that “[e]choes are one of the primary factors affecting the perceived quality of voice connections.” *See* U.S. Patent No. 5,606,550 (Ex. 1035) at 1:20-22. Because network latencies are longer in packet-switched networks compared to circuit-switched networks (due in part to the number of store-and-forward and queuing operations for packet data traversing a packet-switched network), “echoes are easily perceptible, annoying and need to be suppressed.” *Id.* at 1:23-26. Furthermore, feedback can occur when the transmitted audio signal is picked up by the far end receiver’s microphone and returned from the receiver back to the sender, such as may occur when the receiver side is on a speakerphone. While the delays in a packet-switched voice network are too long to produce the screeching such as occurs occasionally in a public-address audio system, echoes nevertheless may persist as they continue to be picked up and be amplified by both ends and continue to ping-pong back and forth, such as when both ends are using speakerphone. A POSITA would recognize that in a conference call, sending audio over a network to a mixer that includes the sender’s audio in the audio sent back to the sender may potentially generate an undesirable

echo signal. Accordingly, a POSITA would have recognized that Knappe's teachings of using a separate summer for each participant, would be beneficial to Drell since each participant's mixed audio includes the other participants' audio and excludes its own audio:

During normal operation, **each participant receives the audio input of each other participant.** Thus, for example, the conference output stream of participant 1 includes the audio inputs of participants 2–5. Similarly, the conference output stream of participant 2 includes the audio inputs of participants 1 and 3–5. The conference output stream of participant 3 includes the audio inputs of participants 1–2 and 4–5. The conference output stream of participant 4 includes the audio inputs of participants 1–3 and 5. The conference output stream of participant 5 includes the audio inputs of participants 1–4.

Knappe (Ex. 1006) at 7:1-11. Thus, a POSITA would have been motivated to implement Drell's mixer, per Knappe's teachings to include a separate summer for each participant, because doing so would allow for excluding audio of the participant receiving the mixed audio, thereby avoiding undesirable audio effects in Drell's conferencing system.

159. In summary, a POSITA would have been motivated to apply Knappe's multiple-summer teachings when implementing a mixer in Drell because doing so would beneficially (1) further Drell's goal of generating separate instances for each conference endpoint, (2) increase intelligibility of the moderator (3) enhance the moderator's control over the conference, and (4) exclude audio of the participant receiving the mixed audio thereby avoiding undesirable audio effects, such as

feedback or echo. Each of these reasons separately and together would have motivated a POSITA.

160. The results would have been predictable and there would have been a reasonable expectation of success in the combination given that both Drell and Knappe describe similar conferencing systems. *See, e.g.*, Drell (Ex. 1008) at Abstract; Knappe (Ex. 1006) at Abstract, 3:47-51. The reasonable expectation of success is also confirmed by other artisans, who for decades had been implementing conferencing systems with moderators. *See, e.g.*, U.S. Patent No. 3,601,530 (Ex. 1014) at 2:34-40. Any modification needed to the teachings of Drell, including implementing software programming or hardware, to accommodate the teachings of Knappe, would have been within the level of ordinary skill in the art. *See, e.g.*, Knappe (Ex. 1006) at 5:49-62, 6:55-58. Indeed, Knappe itself recognizes that its teachings can be applied by POSITAs to other network devices. *See, e.g. id.* at 4:29-33 (“The call manager 30 and the conference bridge 32 may be located in a central facility or have their functionality distributed across and/or at the periphery of the network 14.”).

### (3) Encapsulating VoIP Packets in RTP

161. Drell teaches that its conference system uses codecs to encode mixed voice data before it is transmitted over the Internet network. *See, e.g.*, Drell (Ex. 1008) at 6:33- 45 (“In the transmit mode...[a]udio codec 408 encodes the audio

data...using a standard or proprietary audio compression algorithm (e.g., G711) and delivers the encoded audio data to communication process 404.”); *id.* at 3:33-34 (“[N]etwork 106 may comprise a packet switched network, such as the Internet.”); *see also id.* at 5:8-25, FIG. 4. A POSITA would understand Drell is transmitting its audio data using IP packets, because the Internet utilizes IP packets for communications. *See, e.g.,* Knappe (Ex. 1006) at 3:50-55 (“In the Internet embodiment, the network... transmits information in Internet Protocol (IP) packets.”); Cisco Packetized Voice & Data Integration, Robert Caputo (Ex. 1021) at 12-13.

162. Further, Knappe supplements Drell and teaches that “Real-time IP packets such as VoIP packets are encapsulated in real-time transport protocol (RTP) packets for transmission over the network 14.” Knappe (Ex. 1006) at 3:50-58; *see also id.* at 4:9-12. In view of Knappe’s teachings, it would have been obvious to a POSITA to encode Drell’s real-time audio data in VoIP packets and encapsulate them in RTP packets for transmission over the Internet, as Knappe teaches, because RTP was specifically created for applications such as Drell’s real-time conferencing application. *See* Drell (Ex. 1008) at 3:20- 38 (“Network 106 may be of any type suitable for the transmission of audio and video data between and among near conference endpoint 100 and remote conference endpoints... [N]etwork 106 may comprise a packet switched network, such as the Internet...to

connect conference endpoints..”); *see* Caputo (Ex. 1021) at 22 (“The Real- Time Protocol (**RTP**)...**was created to transport real-time traffic**, such as video and audio, over the Internet.”).

163. Furthermore, Drell teaches that its “communication process...may for example comprise an...H.323 (packet-based) process.” Drell (Ex. 1008) at 2:26-28. A POSITA would have known that the VoIP standard community adopted H.323 standards to foster interoperability of equipment from multiple vendors. *See* Caputo (Ex. 1021) at 23 (“H.323 provides standards for voice-encoding...The voice over IP community has adopted H.323 standards in an effort to foster the interoperability of equipment from multiple vendors. H.323...has become critical for developing large-scale VoIP networks.”); U.S. Patent No. 6,553,515 (Ex. 1025) at 4:35-56 (“H.323 is adapted to support many aspects of multimedia...[and] generally is regarded as the standard for VoIP.”). Accordingly, a POSITA would have been motivated to apply Knappe’s VoIP and RTP teachings to Drell because they are well suited for use with H.323 packet processing.

164. The result would have been predictable and there would have been a reasonable expectation of success in applying Knappe’s VoIP and RTP teachings to Drell, since the utilization of both VoIP and RTP were known to be applied in the conferencing context. *See* Caputo (Ex. 1021) at 22-23. Also, the results would have been predictable and there would have been a reasonable expectation of

success given that both references describe similar conferencing systems that utilize real-time audio and communicate over the Internet. *See, e.g.*, Drell (Ex. 1008) at 1:27-28, 3:33-34; Knappe (Ex. 1006) at 3:29- 31, 3:50-58. Furthermore, any modification needed to the teachings of Drell, including implementing software programming or hardware, to accommodate the teachings of Knappe, would have been within the level of ordinary skill in the art. *See, e.g.*, U.S. Patent No. 6,850,496 (Ex. 1024) at 1:23-48; Knappe (Ex. 1006) at 5:49-62, 6:55-58; Caputo (Ex. 1021) at 22-23. Indeed, Knappe itself recognizes that its teachings can be applied by POSITAs to other network devices. *See, e.g.*, Knappe (Ex. 1006) at 4:30-33 (“The call manager 30 and the conference bridge 32 may be located in a central facility or have their functionality distributed across and/or at the periphery of the network 14.”).

#### (4) Gateway

165. Drell teaches that “the invention contemplates the use of two or more networks (for example, the PSTN and the Internet) to connect conference endpoints utilizing different communication protocols.” Drell (Ex. 1008) at 3:20-37. A POSITA would have recognized that since Drell’s Internet network and PSTN utilize different protocols, it would be beneficial to utilize a device that can convert data between the two networks. *See, e.g.*, U.S. Patent No. 6,553,515 (Ex. 1025) at 4:9-14. To that end, Knappe teaches using a “gateway...[that] provides

conversion between analog and/or digital formats.” Knappe (Ex. 1006) at 4:18-20. Specifically, Knappe explains that “standard analog telephones [] communicate standard telephony signals through PSTN [network] to the gateway []. At the gateway [], the signals are converted to IP packets in the VoIP format. Similarly, VoIP packets received from the [Internet] network [] are converted into standard telephony signals for delivery to the destination telephone [] through PSTN.” *Id.* at 4:20-25; *see also id.* at 3:50-58.

166. In view of Knappe’s teachings, it would have been obvious to a POSITA to use in Drell’s system a gateway that receives standard audio data from PSTN conference endpoints and converts the audio data into a format (e.g., VoIP packets encapsulated in RTP) suitable for conference endpoints on the Internet network. Doing so would beneficially allow conference data to be transmitted between Drell’s PSTN conference endpoints and near conference endpoint 100 over networks (PSTN and Internet) that utilize different protocols, thereby furthering Drell’s objective. *See* Drell (Ex. 1008) at 3:20-37.

167. The result would have been predictable and there would have been a reasonable expectation of success in applying Knappe’s gateway teachings to Drell, since the utilization of gateways was conventional and known to be applied in the conferencing context. *See, e.g.,* U.S. Patent No. 6,553,515 (Ex. 1025) at 4:9-14. Also, the results would have been predictable and there would have been a

reasonable expectation of success given that both references describe similar conferencing systems that utilize both Internet networks and PSTN. *See, e.g.*, Drell (Ex. 1008) at 3:33-38; Knappe (Ex. 1006) at 4:18-25. Furthermore, any modification needed to the teachings of Drell, including implementing software programming or hardware, to accommodate the teachings of Knappe, would have been within the level of ordinary skill in the art.

168. For each of the above discussed combinations of Knappe and Drell, I rely on the teachings of the references and do not suggest that physical incorporation of Knappe's elements into Drell's system is required.

**b) Reasons to Combine Drell and Voois**

169. A POSITA when considering the teachings of Drell would have also considered the teachings of Voois, as they are analogous prior art that both pertain to the same field of endeavor of multiparty conferencing between conference endpoints over PSTN. Drell (Ex. 1008) at 3:33-37 (“[T]he invention contemplates the use of two or more networks (for example, the PSTN and the Internet) to connect conference endpoints.”); Voois (Ex. 1036) at 1:33-37 (“The present invention relates to video communication systems, and more particularly, to a user interface for a programmable videocommunicator architecture for video-conferencing over a conventional communications channel.”); *id.* at 3:62-4:4 (“[S]ystem having a telephone for operating a videophone user interface...includes

hardware along with supporting firmware and/or software, and is included to provide an interface with a communication network, such as a public switched telephone network (PSTN), the internet, local area networks, and/or wide area networks.”).

170. As discussed above, Drell in combination with Knappe teaches utilizing a gateway between a PSTN and Internet network. As also discussed above, Drell supports both audio and video communications and Knappe’s gateway can also communicate both audio and video. *See, e.g.* Knappe (Ex. 1006) at 3:65-4:1 (“...**gateway**...capable of communicating real-time **audio, video** and/or other information over the network 14.”). A POSITA looking to implement the teachings of Knappe and Drell, would have recognized that when the gateway is processing both audio and video signals, it would be beneficial to have a PSTN conference endpoint that can receive such signals.

171. Notably, Voois supplements the combination of Drell and Knappe by describing “a videophone” that communicates over “a public switched telephone network (PSTN).” Voois (Ex. 1036) at 3:62-4:4. According to Voois, the videophone is capable of “transmitting (and receiving) audio and video data.” *Id.* at 4:32-39. A POSITA would have recognized that Voois’s PSTN videophone teachings are well suited for Drell’s system (as modified by Knappe to include a gateway capable of communicating both audio and video) because it would allow

for the user of the PSTN conference endpoint to participate in a conference with both audio and video. This would allow more information (e.g., demonstrative gestures and behavioral/visual cues) to be conveyed to and by the user of the PSTN conference endpoint thereby improving collaboration. *See, e.g.*, U.S. Patent No. 6,237,025 (Ex. 1018) at 1:28-47 (“In addition to spoken words, **demonstrative gestures and behavioral cues**, collaboration often involves the sharing of visual information... This combination of spoken words, gestures, visual cues... significantly **enhances the effectiveness of collaboration** in a variety of contexts.”). Thus, it would have been obvious to a POSITA to implement Drell’s PSTN conference endpoint as a videophone, per Voois, because it would beneficially allow for the user of the PSTN conference endpoint to exchange both audio and video data with the other conference participants, thereby improving the experience of the user and enhancing the effectiveness of collaboration.

172. Such a combination would have simply been combining prior art elements (videophone taught by Voois with the conferencing system of Drell as modified by Knappe) according to known methods (disclosed by Voois) to yield predictable results (allow for PSTN conference endpoint participants in Drell’s system to share both audio and video data during a conference for collaboration purposes). Additionally, the combination of Drell and Voois is a simple substitution of one known element (PSTN videophone taught by Voois) for another

(the PSTN conference endpoint of Drell) to obtain predictable results (allow for the conference participant on the PSTN to use a PSTN videophone to communicate both audio and video data).

173. I also note that the results would have been predictable and there would have been a reasonable expectation of success in the combination given that both Drell and Voois describe similar conferencing systems. *See, e.g.*, Drell (Ex. 1008) at Abstract, 3:33-37; Voois (Ex. 1036) at 1:33-37, 3:62-4:4. I rely on the teachings of the references and do not suggest that physical incorporation of Voois's elements into Drell's system is required in the proposed combination, although physical incorporation is permitted.

### **3. Claim 1**

***[1.P] A system for supporting a multi-participant conference call comprising:***

174. The prior art teaches “a system for supporting a multi-participant conference call” as required by the preamble of claim 1.

175. Drell discloses a multiparticipant conferencing system, specifically a “videoconferencing apparatus for use with multi-point conferences.” Drell (Ex. 1008) at 1:16-18.

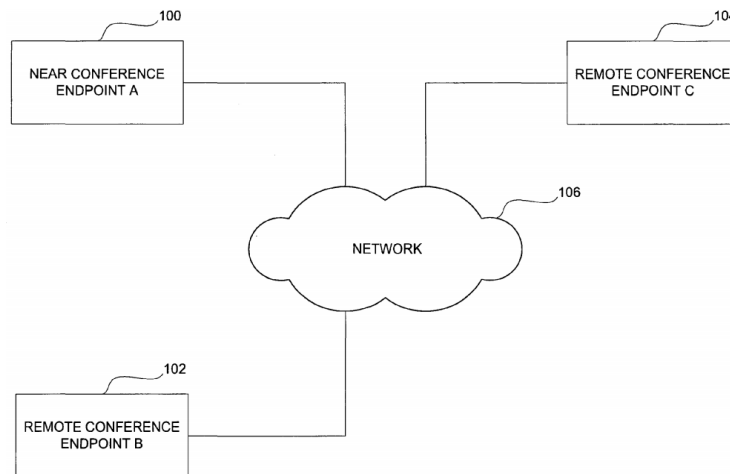


FIG. 1

*Id.* at FIG. 1.

176. Drell explains that its system includes a network 106, which can be a combination of two networks such as the Internet and Public Switched Telephone Network (PSTN), for communicating audio/video data between near conference endpoint 100 and remote conference endpoints 102 and 104:

FIG. 1 depicts an exemplary operating environment of the multi-point (MP) conferencing application of the present invention. **A near conference endpoint 100**, embodying the MP conferencing application, is coupled to **remote conference endpoints 102 and 104 via a network 106**. Remote conference endpoints 102 and 104 may comprise, for example, conventional videoconferencing devices equipped to **transmit and receive both video (image) data and audio (speech) data**.

...

Network 106 may be of any type suitable for the **transmission of audio and video data between and among near conference endpoint 100 and remote conference endpoints 102 and 104**.

...

Alternatively, **network 106 may comprise a packet switched network, such as the Internet**. Although a single network 106 is shown, the invention contemplates the use of **two or more networks**

**(for example, the PSTN and the Internet) to connect conference endpoints** utilizing different communication protocols.

*Id.* at 3:4-47.

177. In view of the above, a POSITA would have understood that Drell's system utilizes network 106, which can comprise the Internet and PSTN, to facilitate video/audio communications during a conference call between near and remote conference endpoints, including IP conference endpoints and PSTN conference endpoints. A POSITA would understand that multiple participants could utilize Drell's multiple conference endpoints. This is because the purpose of a videoconferencing system (such as Drell's) is to "facilitate rich and natural communication between persons or groups of persons located remotely from each other." *Id.* at 1:20-25.

178. Drell further teaches a "call manager application" that "controls the establishment and termination of connections" among the conference endpoints. *Id.* at 4:11-20. The establishment of such connections initiates a "multi-point conference." *Id.* at 3:26-30. Thus, a POSITA would understand that Drell's multi-point conference is a "*conference call*" because it is established by a "call manager" and allows for audio (or audio and video) communication among the participants. *See id.* at 3:8-15, 3:39-48.

179. Thus, Drell discloses a system for supporting a multi-participant conference call, which teaches “[a] *system for supporting a multi-participant conference call*,” as recited in the preamble.

***[1.1] a first mixer that mixes Public Switched Telephone Network (PSTN) client audio data stream with a moderator audio-video data stream into a first mixed data stream;***

180. The prior art renders obvious a first mixer that mixes a Public Switched Telephone Network (PSTN) client audio data stream with a moderator audio-video data stream into a first mixed data stream.

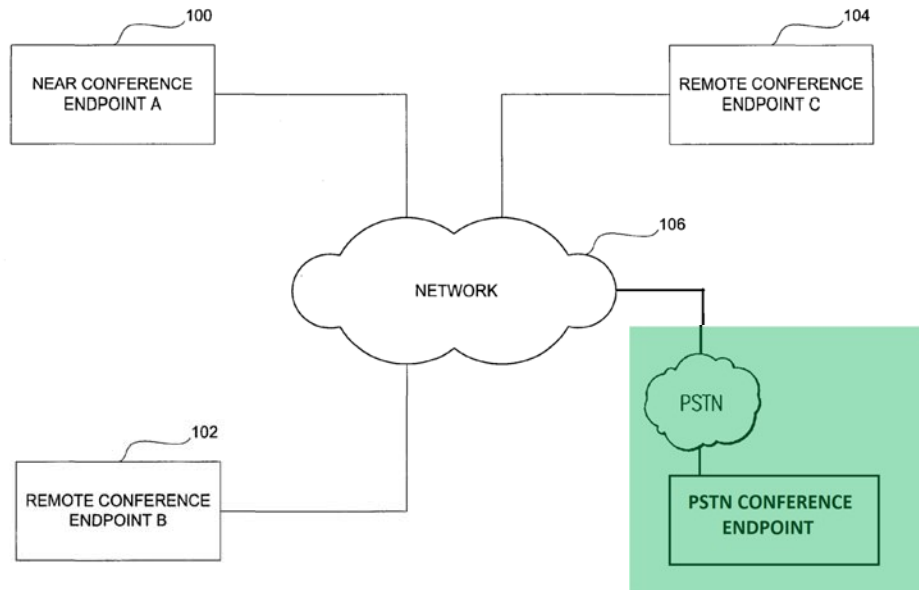
**a) PSTN client**

181. **First**, Drell teaches that a PSTN conference endpoint participates in a conference with a near conference endpoint 100 by communicating audio data streams:

Network 106 may be of any type suitable for the transmission of audio and video data between and among near conference endpoint 100 and remote conference endpoints...the invention contemplates the use of two or more networks (for example, the **PSTN** and the Internet) to connect **conference endpoints** utilizing different communication protocols.

Drell (Ex. 1008) at 3:20-37. The communication between the conference endpoints and the near conference endpoint 100 includes streamed audio. *Id.* at 4:25-26 (“The processing trains process **audio** and video **data streams** received from remote conferencing endpoints 102 and 104.”); *see also id.* at 2:17-33. Drell further explains that “while only two remote conference endpoints are depicted in

FIG. 1 for the purpose of clarity, a greater number of remote conference endpoints may be accommodated by near conference endpoint 100.” *Id.* at 3:15-19.



*Id.* at FIG. 1 (annotated).

182. Thus, a POSITA would have understood that in Drell a remote PSTN conference endpoint can connect to Drell’s videoconferencing system via the PSTN, as I have illustrated above. Thus, Drell’s PSTN conference endpoint discloses a “*PSTN client*.”

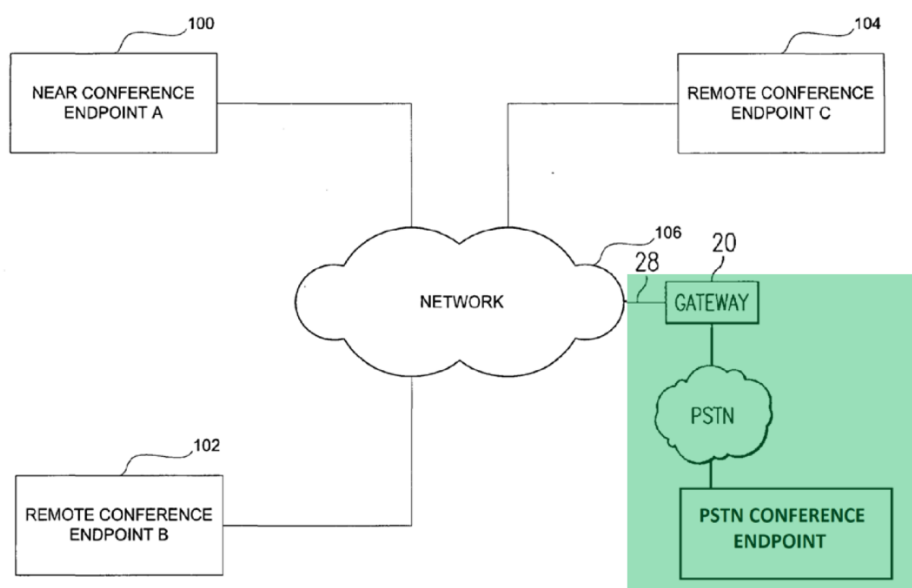
183. Knappe similarly describes “multiparty communications sessions” where conference media streams include “audio, video, voice, data and other suitable types of real-time and/or non real-time traffic.” Knappe (Ex. 1006) at 3:27-28, 5:46- 47; *see also id.* at 3:65-4:1. Knappe further teaches that when a PSTN conference endpoint participates in a conference, over a PSTN network and

an IP network, audio data streams from the PSTN conference endpoint would be communicated through a gateway that provides conversion of data between the two different networks. *See, e.g., id.* at 4:18-25 (“The gateway [] provides conversion between analog and/or digital formats. The standard analog telephones [] communicate standard telephony signals through PSTN [network] to the gateway []. At the gateway [], the signals are converted to IP packets in the VoIP format. Similarly, VoIP packets received from the [Internet] network [] are converted into standard telephony signals for delivery to the destination telephone [] through PSTN [network].”); *id.* at 2:1-6 (“[A] system and method for improving the intelligibility of a moderator during a multi-party communications session includes receiving a plurality of participant **voice streams** from a plurality of respective conference participants.”); *id.* at 3:65-4:1 (“[T]he **telephone 18 and gateway 20 combination [is] capable of communicating real-time audio, video** and/or other information over the network 14.”); *see also id.* at 3:50-58. A POSITA would understand that a PSTN-connected telephone is a “*PSTN client*” as claimed because it receives conferencing services via the PSTN. *See id.* at 4:50-57 (describing “conference call and other multiparty communications and/or other suitable **services**” provided to communications devices).

**b) PSTN client audio data stream**

184. Further, as described above, both Drell and Knappe describe transmitting audio from a conference participant as a “*data stream*.” See Drell (Ex. 1008) at 2:17-33; Knappe (Ex. 1006) at 3:65-4:1, 5:46-47 (describing “multiparty communications sessions including real-time audio streams”). A POSITA would understand that Drell’s signal stream transmitted from a remote conference endpoint is an “*audio data stream*” as claimed.

185. It would have been obvious to a POSITA to use in Drell’s system a gateway (as in Knappe) that converts standard audio data from PSTN conference endpoints into a streaming audio format suitable for transmission over the Internet. As illustrated below, it would have been obvious to a POSITA that Drell’s PSTN conference endpoint would transmit an audio stream through a gateway, like discussed in Knappe:



Drell (Ex. 1008) at FIG. 1 (annotated). Doing so would beneficially allow Drell's PSTN conference endpoints to communicate, during the conference, audio data to the near conference endpoint 100, over a PSTN network and an Internet network 106. *See also* Reasons to Combine Drell and Knappe, *supra* ¶¶ 148-168.

186. Thus, Drell in combination with Knappe that near conference endpoint 100 receives an audio data stream from a PSTN conference endpoint through a PSTN gateway, which renders obvious “*a Public Switched Telephone Network (PSTN) client audio data stream,*” as claimed.

**c) moderator**

187. Further, Knappe teaches that at least one of the conference participants could be designated as a moderator:

**[A]t least one participant may be given the priority of moderator for a particular conference call.** Typically, the moderator will receive the highest priority in a conference call. Therefore, in accordance with one embodiment, the priorities simply include moderator for one.

Knappe (Ex. 1006) at 8:52-56.

188. Since Knappe teaches that at least one participant may be given the priority of a moderator, it would have been obvious to a POSITA to designate one of Drell's conference endpoints as a moderator, such as the user of remote conference endpoint 104 when the user initiates the conference. *See, e.g., id.*, Claim 1 (“... automatically identifying **a moderator initiating the multi-party communication session.**”). Moreover, designating a moderator would allow for

the user (acting as the moderator) to control the flow and organization of the conference. *See id.* at 1:53-55 (“...a designated moderator(s) to **control** the flow and/or **organization** of the conference.”); *see also* Reasons to Combine Drell and Knappe, *supra* ¶¶ 148-168.

189. This is consistent with the ’525 patent, which refers to the moderator as a “voice over IP moderator...(call initiator or caller)” and indicates that the moderator “initiates the call using the application code running on [a separate] voice over IP server” through a first connection established “for call initiation, set-up and control.” ’525 patent (Ex. 1001) at 5:4-5, 6:37-45, 6:59-64. Thus, Drell in combination with Knappe teaches a “*moderator*.”

**d) moderator audio-video data stream**

190. **Second**, Drell teaches that the near conference endpoint 100 has a multi-point (MP) conferencing application 234 with processing train 304 that receives an audio-video data stream from the remote conference endpoint 104 (moderator) via network 106:

FIG. 1 depicts an exemplary operating environment of the multi-point (MP) conferencing application of the present invention. **A near conference endpoint 100**, embodying the MP conferencing application, is coupled to remote conference endpoints 102 and 104 via a network 106. Remote conference endpoints 102 and 104 may comprise, for example, conventional videoconferencing devices equipped to transmit and **receive both video (image) data and audio (speech) data**.

Drell (Ex. 1008) at 3:4-12.

Network 106 may be of any type suitable for the transmission of **audio and video data** between and among near conference endpoint 100 and remote conference endpoints 102 and 104.

*Id.* at 3:20-23.

As will be described in further detail below, MP conferencing application 234 is configured to instantiate a processing train for each remote conference endpoint 102 and 104 to which near conference endpoint 100 is connected. The **processing trains process audio and video data streams received from remote conferencing endpoints 102 and 104**. The processed audio and video data streams are combined with each other and with locally generated audio and video streams, and the combined audio and video streams are thereafter distributed to remote conferencing endpoints 102 and 104.

*Id.* at 4:21-31.

Processing train 304...receives audio and video data transmitted by remote conference endpoint 104.

*Id.* at 5:1-2.

191. Drell further teaches that the audio and video data stream from the remote conference endpoint 104, which is received as a single stream, is separated by the processing train 304 into audio and video signals:

The (MP) conferencing application is operable to process the plural signal streams received through the communication ports. Generally, **the MP conferencing application generates separate processing trains for signal streams from/to each of the remote conference endpoints**. The processing trains each comprise a communication process and a set of codecs. In the receive mode, an IMUX function combines signal streams (representative of a single conference endpoint) distributed over two or more channels into a single, relatively high bandwidth channel. The communication process, which may for example comprise an H.320 process (ISDN-based) or H.323 (packet-based) process, **separates the signal stream into audio and video signals**, and performs certain processing operations (such as delay

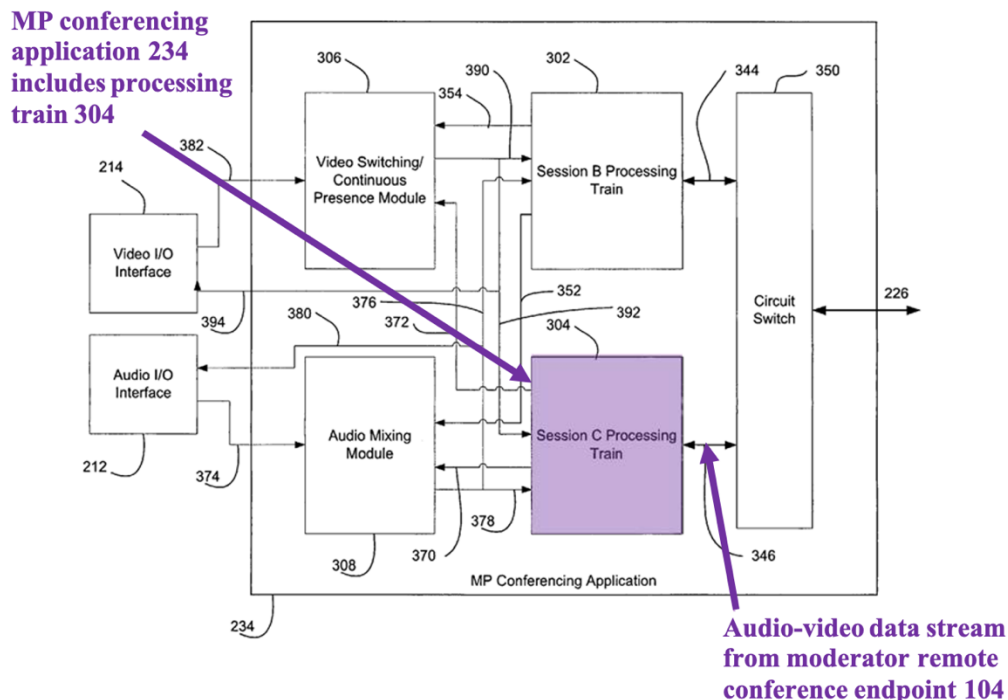
compensation) associated therewith. The audio and video signals are thereafter respectively delivered to audio and video codecs for decoding.

*Id.* at 2:17-33.

In particular, communication process 404 includes a multiplexer/demultiplexer for **(in the receive mode)** extracting separate audio and video 10 signals from **mixed data stream 344**.

*Id.* at 6:7-10

192. As shown below at Drell's Figure 3, the MP conferencing application 234 includes a processing train 304 that receives a mixed audio-video data stream from remote conference endpoint 104:



*Id.* at FIG. 3 (annotated).

193. Drell explains that “[i]n the example depicted in the figures, the circuit switch 350 dynamically allocates two IMUXs and generates two processing trains 302 and 304 respectively corresponding to remote conference endpoints 102 and 104.” *Id.* at 4:44-48. Further, as shown below at Drell’s Figure 4, the processing train receives a single stream, which is separated into audio and video streams by the communication process 404:

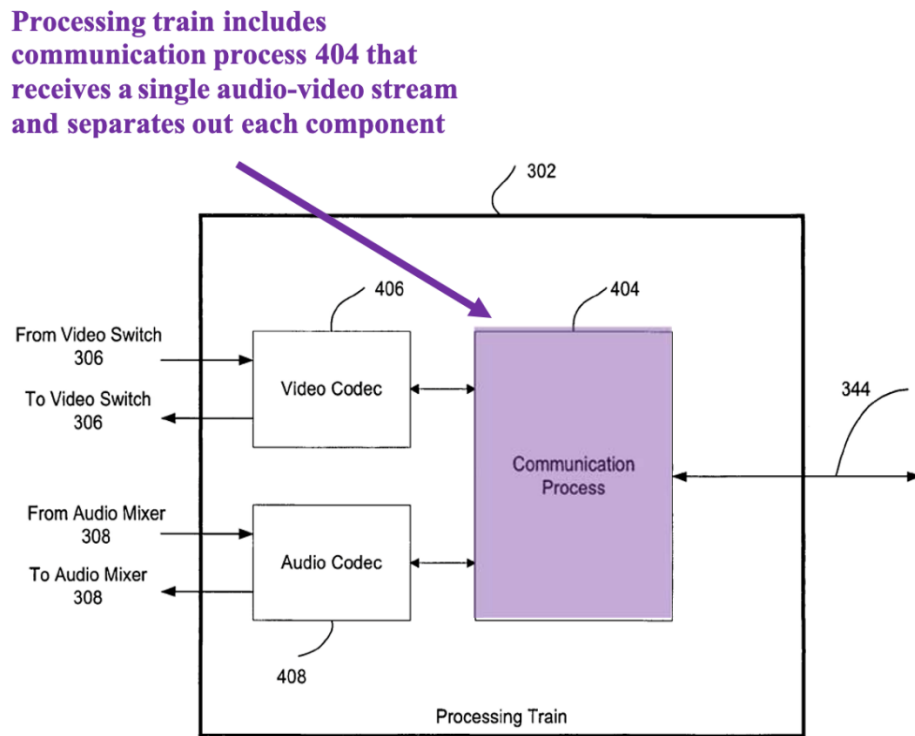


FIG. 4

*Id.* at FIG. 4 (annotated).

194. Since it would have been obvious to a POSITA for remote conference endpoint 104 (which transmits audio and video data) to be used by a moderator (as

suggested by Knappe), the combination of Drell and Knappe render obvious a “*moderator audio-video data stream*,” as claimed.

**e) first mixer mixes . . . into a first mixed data stream**

195. **Third**, the phrase “*first mixed data stream*” is construed as a *first data stream including at least audio data and video data*. See Claim Construction Section, above.

196. Drell teaches that the audio mixing module 308 of the MP conferencing application 234 mixes the audio data stream from the moderator remote conference endpoint 104 with the audio data streams received from other remote conference endpoints:

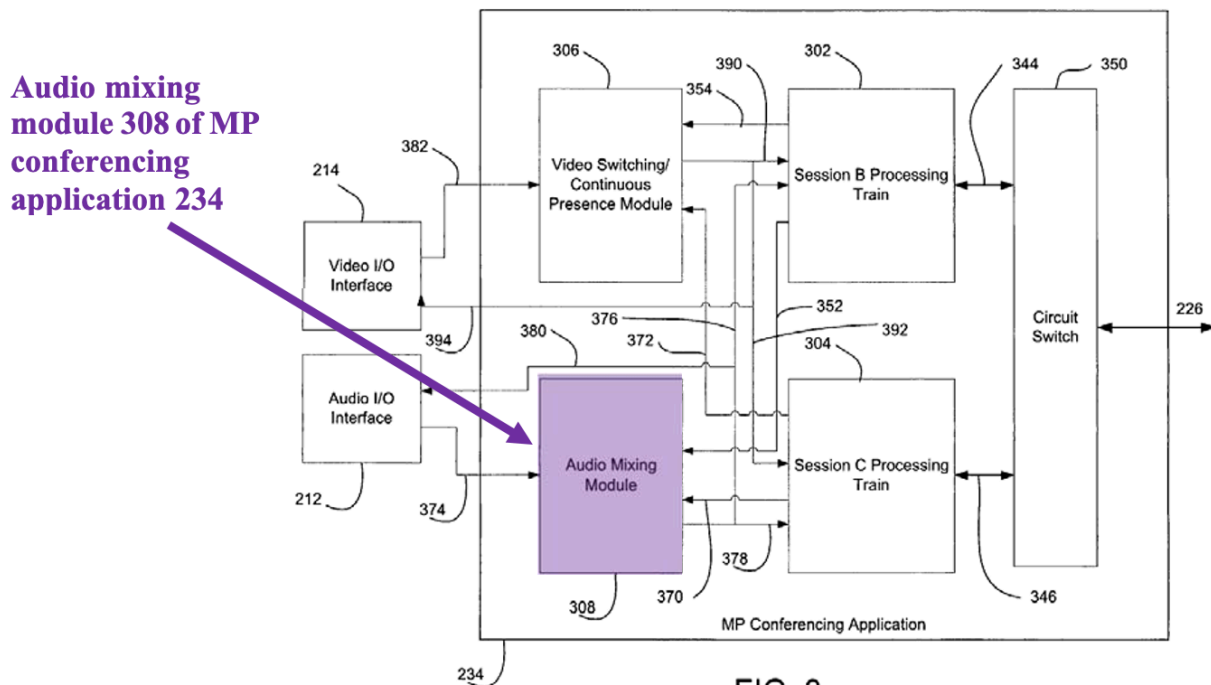
FIG. 3 is a block diagram showing the various components of an embodiment of MP conferencing application 234 and the flow of data between and among the various components. MP conferencing application 234 includes a circuit switch 350, a plurality of processing trains 302 and 304, a video switching/continuous presence module 306, and an **audio mixing module 308**.

Drell (Ex. 1008) at 4:32-38.

**Audio mixing module 308 is configured to combine audio data received from remote conference endpoints 102 and 104** with locally generated audio data (received from audio I/O interface 212 via audio data path 374, and typically being representative of the speech of the near conference participant(s)). The term “combine” is used in its broadest and most general sense and is intended to cover any operation wherein audio mixing module 308 generates an output audio data stream (or plurality of **output audio data streams**) based on information contained in the remotely and locally generated audio data input streams. For example, **audio mixing module 308 may simply mix the received audio input data streams**.

*Id.* at 5:8-23; *see also id.* at Abstract, 2:34-52, 3:45-48, 4:25-31.

197. A POSITA would understand that when Drell's conference system includes a PSTN conference endpoint, *i.e.*, an additional conference endpoint connected via the PSTN (as discussed above), the audio data received from the PSTN conference endpoint would also be combined in the audio mixing module 308. Also, a POSITA would understand that during conferencing, the audio mixing module 308 would mix the audio data stream from the moderator remote conference endpoint 104 with the audio data stream from the remote PSTN conference endpoint.



198. Drell's MP conferencing application 234 also includes a video switching/continuous presence module 306 that mixes the video data stream

received from the remote conference endpoint 102 with the video data stream from moderator conference endpoint 104 and outputs a plurality of a mixed video streams:

Video switching/continuous presence module 306 combines video data received from remote conference endpoints 102 and 104 with locally generated video data (received from video I/O interface 214 via video data path 382, and being typically representative of images of the near conference participants). Again, the term “combine” is used in its broadest and most general sense. In one mode of operation, video switching/continuous presence module 306 may select one of the video data input streams for output based on predetermined criteria (for example, it may **select for output the video data stream** corresponding to the conference endpoint of the currently speaking participants. In a second mode of operation (referred to as the continuous presence mode”), video switching/continuous presence module 306 may construct a composite image wherein images corresponding to conference endpoints are displayed in different sectors of the composite image. **The video data stream output (or plurality of outputs) from video switching continuous presence module 306 is thereafter distributed to 45 processing trains.**

*Id.* at 5:24-47; FIGS. 3-4.

199. Drell further teaches that the processing trains (e.g., 302/304), for each endpoint, include a “communication process 404” that receives mixed audio data from the audio mixing module 308 and receives a mixed video data stream output by the video switching/continuous presence module 306. *Id.* at 6:33-45, FIG. 4. The communication process 404 then “multiplexes the encoded audio and video data streams into **a single audio/video data stream.**” *Id.* at 6:46-48.

200. Drell's audio mixing module 308, and video switching/continuous presence module 306, and communication process 404 collectively operate to mix the audio and video data streams received from multiple remote conference endpoints into a single audio and video data stream, which renders obvious "*first mixed data stream.*"

201. As I noted above, Drell's audio mixing module 308, and video switching/continuous presence module 306, and communication process 404 of the processing train 302 collectively produce the claimed "*first mixed data stream.*" Drell specifically teaches that its videoconference system instantiates a separate "communication process 404" for each remote conference endpoint. *Id.* at 5:65-66; 4:21-23.

202. A POSITA would understand that Drell's audio mixing module 308, and video switching/continuous presence module 306, and the communication process 404 (of the processing train 302) instantiated for the remote conference endpoint 102 collectively teach the claimed "*first mixer.*"

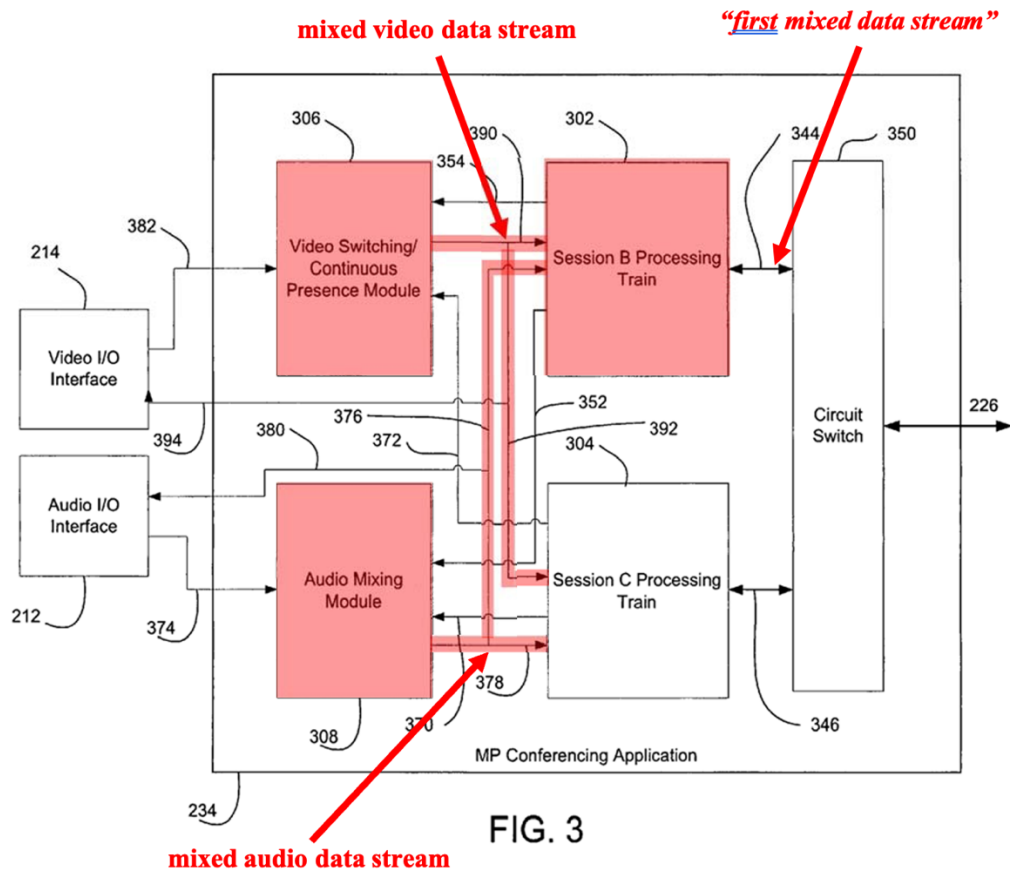


FIG. 3

*Id.* at FIG. 3 (annotated).

203. **Additionally**, for the reasons discussed below, it would have been obvious to a POSITA for Drell’s audio mixing module 308 to include a separate audio mixer for each remote conference endpoint, and for Drell’s video switching/continuous presence module 306 to include a separate video mixer for each remote conference endpoint. Thus, it would have also been obvious to a POSITA that the separate audio mixer, separate video mixer, and the communication process 404 (of the processing train 302) instantiated for the remote

conference endpoint 102 collectively constitute the claimed “*first mixer*,” as discussed further below.

204. Knappe supplements and explains how to mix audio data streams from multiple conference endpoints, including audio from PSTN conference endpoints. *See* Knappe (Ex. 1006) at FIGs. 1-3, 5:49-62, 6:59-7:15. Specifically, in connection with Figure 3, Knappe teaches using a mixer 80 with a plurality of summers (e.g., summers 1 to 5), where each summer mixes audio data from the other participants (e.g., participants 1 to 5) while providing higher prominence to the moderator audio:

**Referring to FIG. 3, the monaural mixer 80 receives participant input streams 84 and combines the streams in summers 82 to generate conference output streams 86 for each participant** to a conference call. In one embodiment, each participant is assigned a summer 82 that receives audio input streams from each other participant to the conference call. The summer 82 combines the audio input streams to generate a conference output stream for delivery to the participant.

**During normal operation, each participant receives the audio input of each other participant.** Thus, for example, the conference output stream of participant 1 includes the audio inputs of participants 2–5. Similarly, the conference output stream of participant 2 includes the audio inputs of participants 1 and 3–5. The conference output stream of participant 3 includes the audio inputs of participants 1–2 and 4–5. The conference output stream of participant 4 includes the audio inputs of participants 1–3 and 5. The conference output stream of participant 5 includes the audio inputs of participants 1-4.

The audio input 84 of the **conference moderator may be amplified** and/or the audio input 84 of the remaining participants attenuated to

focus on or **provide higher prominence** to the audio input 84 of the conference moderator.

*Id.* at 6:59-7:15, FIG. 3.

205. It would have been obvious to a POSITA to use in Drell's mixing module 308 of the MP conferencing application 234 a plurality of summers, as Knappe teaches. Using a plurality of summers (one for each participant) would provide for separately mixing audio data from the PSTN conference endpoint with audio data from the moderator remote conference endpoint 104, while providing higher prominence to the moderator audio and increasing the moderator's intelligibility. *See, e.g., id.* at 7:11-17 ("[T]he conference moderator may be amplified and/or the audio input 84 of the remaining participants attenuated to focus on or provide higher prominence to the audio input 84 of the conference moderator. A higher prominence is provided by increasing the intelligibility of the moderator relative to the remaining participants"); *see also id.* at 2:40-46. By enhancing the intelligibility of the moderator during the conference, Knappe's teachings beneficially enhance the moderator's control over the conference. *See, e.g., id.* at 1:59-67. Additionally, providing a separate summer for each conference participant would provide the benefit of reducing undesirable audio effects since each participant's mixed audio excludes its own audio data. *See* U.S. Patent No. 7,266,091 (Ex. 1034) at 6:31-33 ("It is undesirable for a conference participant to

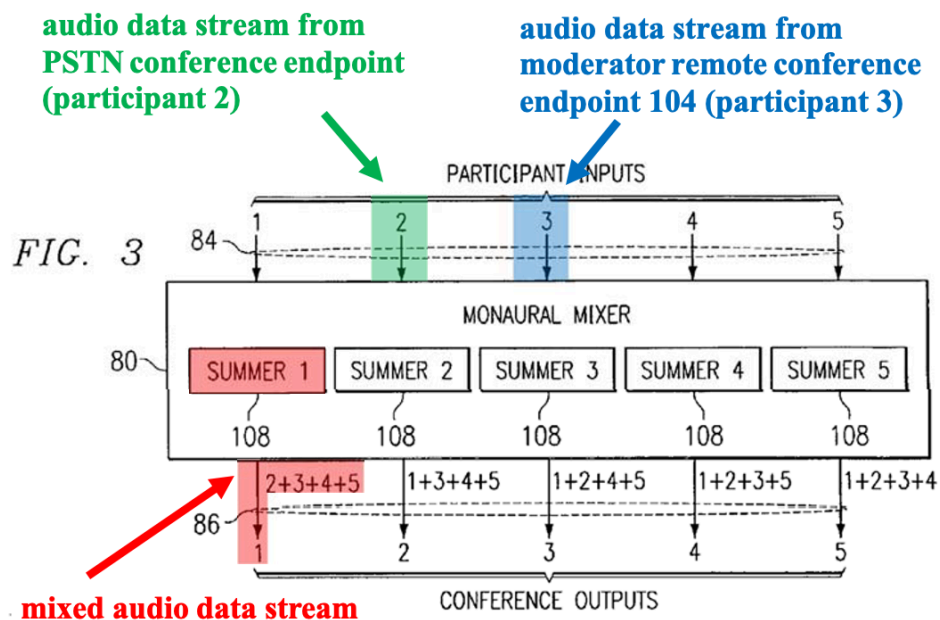
receive his or her own audio in the mixed audio stream.”); *see also* Reasons to Combine Drell and Knappe, *supra* ¶¶ 148-168.

206. In the combination, Drell’s audio mixing module 308 would use a plurality of summers (one summer per participant) to obtain the functionality of generating a separate audio mix for each participant, which is consistent with Drell’s desire to generate a separate instance for each remote endpoint. *See* Drell (Ex. 1008) at 6:66-7:4 (“By dynamically generating a separate instance of a processing train for each remote endpoint session, a videoconferencing system embodying the invention may easily and flexibly accommodate endpoint sessions comprising a range of connection bandwidths and communication protocols.”); *see also id.* at 2:19-21, 4:44-48. The use of multiple summers is also suggested by Drell, which states that audio mixing module 308 may generate a “plurality of output audio data streams.” *Id.* at 5:13-17

207. For explanation purposes, and without limiting Knappe’s disclosure, I have provided below an annotated example of Figure 3 of Knappe identifying which of Drell’s endpoints correspond to which of participants 1-5. In the figure below, participant 2 corresponds to the PSTN conference endpoint that provides audio data to input 2 of mixer 80, and participant 3 corresponds to the moderator remote conference endpoint 104 that provides audio data to input 3 of mixer 80, which

summer 1 mixes into a mixed audio data comprising audio from participants

2+3+4+5:



Knappe (Ex. 1006) at FIG. 3 (annotated).

208. Further, as I already noted above, Drell teaches that the video switching/continuous presence module 306 produces a “video data stream output (or plurality of outputs).” Drell (Ex. 1008) at 5:43-44. A POSITA would understand that the video switching/continuous presence module 306 would include multiple separate video mixers to produce such multiple outputs.

209. For example, Knappe teaches that the “conference output stream for a participant includes the media of all **other** participants.” Knappe (Ex. 1006) at 5:35-37. Knappe’s discussion of “media” includes “video” (*id.* at 3:27), and thus

Knappe suggests generating a mixed video stream output for a participant that includes the video streams “of all other participants.” *Id.* at 5:36-37.

210. Drell states that “video switching/continuous presence module 306 may construct a composite image wherein images corresponding to conference endpoints are displayed in different sectors of the composite image.” Drell (Ex. 1008) at 5:39-42. Based on Knappe’s teaching (and similar to the discussion of audio mixing above), it would have been obvious to a POSITA for Drell’s video switching/continuous presence module 306 to include a separate video mixer so as to generate a separate “composite image” for each conference endpoint that shows only the video data streams from the *other* conference endpoints. By producing an individualized composite image in this way, the video display at each conference endpoint would show the video streams from the other conference endpoints without displaying its own locally produced video stream. Since it is important for participants to see other’s demonstrative gestures and behavioral cues, this approach would maximize the beneficial use of the available space on the videoconference display for displaying such gestures and cues. *See* U.S. Patent No. 6,237,025 (Ex. 1018) at 1:28-47 (“In addition to spoken words, demonstrative gestures and behavioral cues... significantly enhances the effectiveness of collaboration in a variety of contexts.”).

211. In summary, it would have also been obvious to a POSITA that the audio mixer (utilizing a separate summer for each participant), separate video mixer, and the communication process 404 instantiated for the “*at least one remote client*” collectively constitute the claimed “*first mixer.*”

212. Thus, Drell in combination with Knappe teaches that the MP conferencing application 234 includes an audio mixing module 308 (with separate summer for each participant), which mixes audio data streams from the PSTN conference endpoint and the moderator remote conference endpoint 104, video switching/continuous presence module 306 (including a separate video mixer for each participant), and communication process 404 (for remote conference endpoint 102), which further mixes the mixed audio data stream with the moderator’s video data stream into a first mixed data stream, which renders obvious “*a first mixer that mixes a Public Switched Telephone Network (PSTN) client audio data stream with a moderator audio-video data stream into a first mixed data stream,*” as claimed.

***[1.2.1] a first transport output that transmits the first mixed data stream to at least one remote client that receives the first mixed data stream***

213. The prior art renders obvious a first transport output that transmits the first mixed data stream to at least one remote client that receives the first mixed data stream.

214. **First**, as discussed at [1.1], Drell alone and in combination with Knappe teaches that processing train 302 outputs a first mixed data stream.

Specifically, “[c]ommunication process 404 multiplexes the encoded audio and video data streams into a single audio/video data stream 344 of relatively high-bandwidth.” Drell (Ex. 1008) at 6:46-48; *see supra* ¶¶ 199-203.

215. **Second**, Drell teaches that “combined audio and video streams are thereafter **distributed** to remote conferencing endpoints 102 and 104.” Drell (Ex. 1008) at 4:29-31. This includes “**transmitting** the combined audio and video signals to each of the plurality of remote conference endpoints.” *Id.* at 7:30-31; *see also id.* at 4:49-52; 6:33; 6:55-56. As discussed above in [1.1], Drell, alone or together with Knappe, renders obvious producing a separate mixed data stream for each remote conference endpoint. *See supra* ¶¶ 203-210. Thus, it would have been obvious to a POSITA that distributing the combined audio and video streams would include “*transmit[ting]*” the “*first mixed data stream*” to a remote conferencing endpoint, such as remote conferencing endpoint 102. It would have been obvious to a POSITA that the remote conferencing endpoint 102 then receives the “*first mixed data stream*” because receiving transmitted audio and video data is merely conventional videoconferencing device functionality. Drell (Ex. 1008) at 3:8-12 (“**Remote conference endpoints 102** and 104 may comprise, for example, conventional videoconferencing devices equipped to transmit and **receive both video (image) data and audio (speech) data.**”). Thus, the remote conferencing

endpoint 102 is “*at least one remote client that receives the first mixed data stream.*”

216. **Third**, Drell teaches a separate communication process 404 (part of a “processing train”) for each conference endpoint physically couples the near conference endpoint 100 to the network 106 via at least one transmission line for transmitting the mixed audio/video data stream to remote conference endpoint 102:

The combined audio and video data streams are conveyed as input to each **processing train for encoding and transmission to the corresponding conference endpoints**. In the send mode, the audio and video signals are encoded by the audio/video codecs and multiplexed into a **single data stream by the communication process**.

*Id.* at 2:43-48; *see also id.* at Abstract, 4:21-24.

It is noted that, while not depicted in the Figures, processing train 302 may include a data codec for coding and encoding still images and the like...transmitted to remote conference endpoints 102 and 104.

*Id.* at 6:53-56.

**In the transmission mode, processing train 302** is configured to receive the audio and video data streams output by audio mixing module 308 and video switching/continuous presence module 306. The received data streams are then encoded and combined to form a mixed encoded audio/video data stream, and the encoded **audio/video data stream is transmitted...via data path 344**.

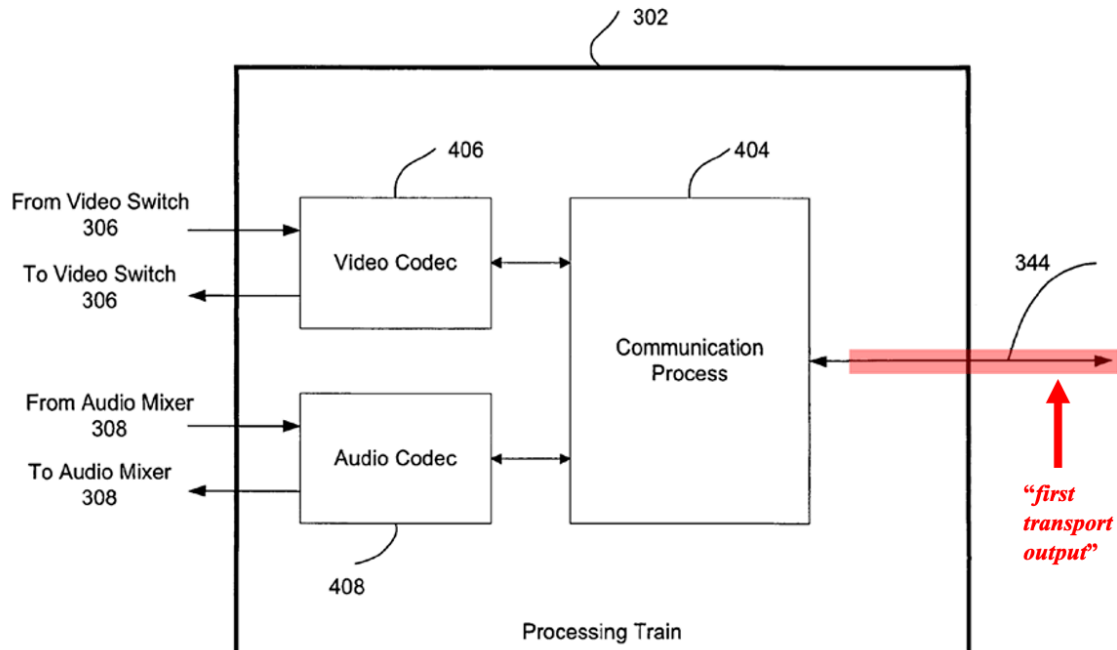
FIG. 4 depicts components of an exemplary processing train 302. Processing train 302 includes a communication process 404 and video and audio codecs 406 and 408. In the receive mode, the combined data stream 344 is directed to communication process 404 which carries out a predetermined set of functions with respect to **data stream 344**.

*Id.* at 5:51-6:2.

FIG. 1 depicts an exemplary operating environment of the multi-point (MP) conferencing application of the present invention. A near conference endpoint 100, embodying the MP conferencing application, is coupled to remote conference endpoints 102 and 104 via a network 106. **Remote conference endpoints 102 and 104 may comprise, for example, conventional videoconferencing devices equipped to transmit and receive both video (image) data and audio (speech) data...** It should be appreciated that while only two remote conference endpoints are depicted in FIG. 1 for the purpose of clarity, a greater number of remote conference endpoints may be accommodated by near conference endpoint 100.

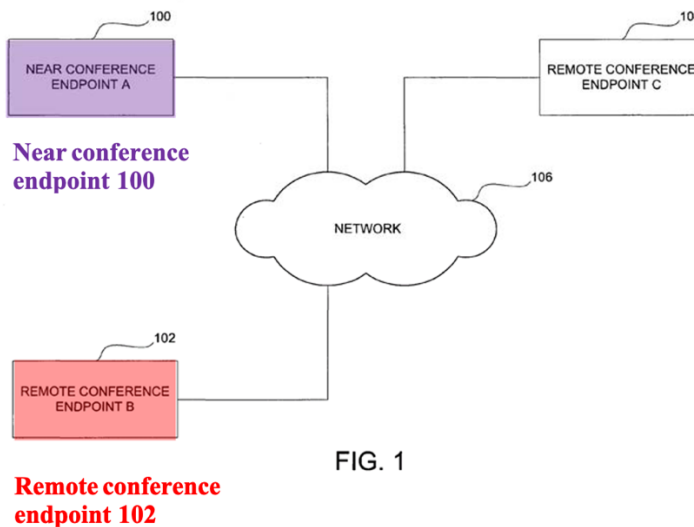
*Id.* at 3:4-19; *see also id.* at FIGS. 1-5.

217. Drell shows in Figure 4 that the communication process 404 provides an output (labeled 344) that conveys the “single audio/video data stream” (e.g., the “*first mixed data stream*”):



*Id.* at FIG. 4 (annotated). Drell’s disclosure is similar to the description of a “transport output” in the ’525 patent. *See* ’525 patent (Ex. 1001) at Fig. 6 (element

548). **Additionally**, Drell illustrates in Figure 1 that remote conference endpoint 102 is coupled to network 106 through an unlabeled connection:

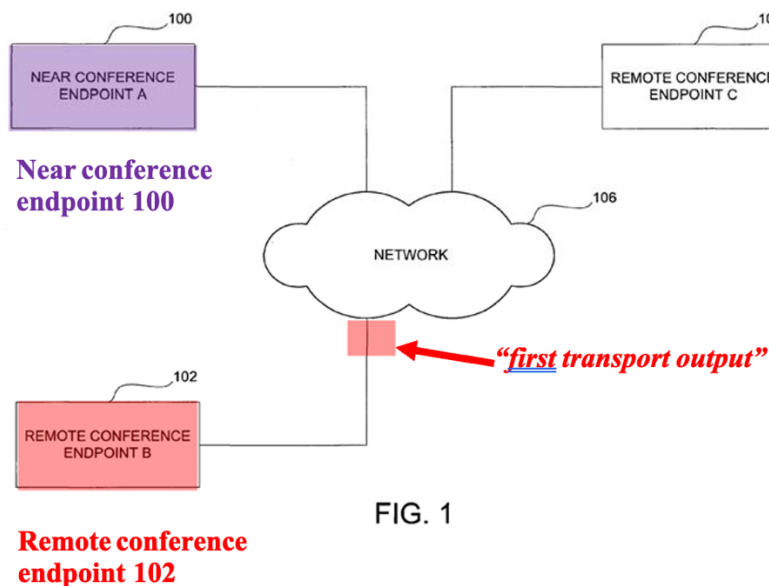


Drell (Ex. 1008) at FIG. 1 (annotated).

218. While Drell illustrates network 106 as a cloud, it would have been obvious to a POSITA that network 106 would include a plurality of network communication devices, such as switches and routers, because that is how networks such as the Internet and PSTN are composed. It would have further been obvious to a POSITA that the line in Figure 1 that connects remote conference endpoint 102 to network 106 represents a connection from remote conference endpoint 102 to a router, switch, or other communication device within network 106. This router, switch, or other communication device would transmit data to remote conference endpoint 102 that network 106 receives and that is destined for the remote conference endpoint 102. The transmitted data would include the “*first mixed data*

*stream*” transmitted by near conference endpoint 100 because Drell teaches that “combined audio and video signals” are transmitted to “each of the plurality of remote conference endpoints.” *Id.* at 7:30-31.

219. Accordingly, it would have been obvious to a POSITA that the output of network 106 leading toward remote conference endpoint 102 is a “*first transport output*” as claimed. In other words, the network side of the connection between network 106 and remote conference endpoint 102 is a “*first transport output*” as claimed. This mapping is illustrated below:



*Id.* at FIG. 1 (annotated).

220. Thus, Drell in combination with Knappe teaches that the near conference endpoint 100 has a first transport output that transmits the first mixed data stream to remote conference endpoint 102 that receives the first mixed data

stream, which renders obvious “*a first transport output that transmits the first mixed data stream to at least one remote client that receives the first mixed data stream,*” as claimed.

**[1.2.2] the at least one remote client communicatively coupled to the Internet, which generates a remote client audio-video data stream;**

221. The prior art renders obvious that the at least one remote client is communicatively coupled to the Internet, which generates a remote client audio-video data stream.

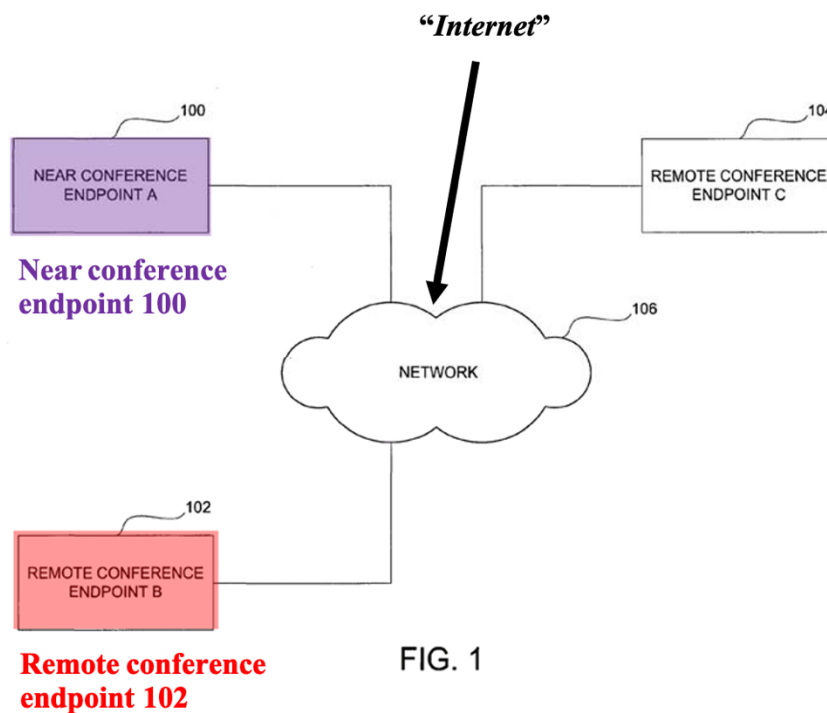
222. **First**, as discussed at [1.2.1], Drell teaches a remote conference endpoint 10 that is “*at least one remote client.*”

223. Drell further teaches that the remote conference endpoint 102 is communicatively coupled to network 106, which comprises the Internet. *See, e.g.,* Drell (Ex. 1008) at 3:33-34 (“**[N]etwork 106** may comprise a packet switched network, such as the **Internet.**”). Further, with reference to Figure 1, Drell explains that remote conference endpoint 102 (which is connected to Internet network 106) receives audio and video data:

FIG. 1 depicts an exemplary operating environment of the multi-point (MP) conferencing application of the present invention. A near conference endpoint 100, embodying the MP conferencing application, is coupled to remote conference endpoints 102 and 104 via a network 106. **Remote conference endpoints 102** and 104 may comprise, for example, conventional videoconferencing devices equipped to transmit and **receive both video (image) data and audio (speech) data.**

*Id.* at 3:4-19.

224. A POSITA would understand that Drell's remote conference endpoint 102 is "*communicatively coupled to the Internet,*" as claimed, since it receives communications from the near conference endpoint over Internet network 106.



*Id.* at FIG. 1 (annotated).

225. **Second**, Drell teaches that the remote conference endpoint 102 includes conventional videoconferencing devices, which transmit audio and video data to the MP conferencing application of the near conference endpoint 100 via the network 106:

FIG. 1 depicts an exemplary operating environment of the multi-point (MP) conferencing application of the present invention. A near conference endpoint 100, embodying the MP conferencing application, is coupled to remote conference endpoints 102 and 104 via a network 106. **Remote conference endpoints 102 and 104 may comprise, for**

**example, conventional videoconferencing devices equipped to transmit...both video (image) data and audio (speech) data.**

*Id.* at 3:4-12.

Reference is now directed to FIG. 2, which depicts in block form various components of near conference endpoint 100. **A conventional video camera 202 and microphone 204 are operative to generate video and audio signals** representative of the images and speech of the near conference participant (the person or persons co-located with near videoconference endpoint 100).

*Id.* at 3:39-45.

226. A POSITA would understand that remote conference endpoint 102 could similarly include conventional equipment, such as a video camera and microphone, to generate video and audio signals of a conference participant because that is typical of videoconference endpoints.

227. Drell further describes that the near conference endpoint 100 receives the audio-video data stream transmitted by the remote conference endpoint 102:

As will be described in further detail below, MP conferencing application 234 is configured to instantiate a processing train for each remote conference endpoint 102 and 104 to which near conference endpoint 100 is connected. The processing trains process **audio and video data streams received from remote conferencing endpoints 102 and 104**. The processed audio and video data streams are combined with each other and with locally generated audio and video streams, and the combined audio and video streams are thereafter distributed to remote conferencing endpoints 102 and 104.

*Id.* at 4:21-31.

In particular, communication process 404 includes a multiplexer/demultiplexer for **(in the receive mode)** extracting separate audio and video 10 signals from **mixed data stream 344**.

*Id.* at 6:7-10. Because the audio and video information are received as a single stream, a POSITA would understand that remote conferencing endpoint 102 “*generates a remote client audio-video data stream,*” as claimed.

228. Thus, Drell teaches that the remote conference endpoint 102 is communicatively coupled to the Internet network 106 and generates an audio-video data stream that is received by the near conference endpoint 100, which renders obvious “*the at least one remote client communicatively coupled to the Internet, which generates a remote client audio-video data stream,*” as claimed.

***[1.3] a second mixer that mixes the moderator audio-video data stream with the remote client audio-video data stream into a second mixed data stream; and***

229. The prior art renders obvious a second mixer that mixes the moderator audio-video data stream with the remote client audio-video data stream into a second mixed data stream.

230. **First**, as discussed at [1.1], Drell in combination with Knappe teaches that the near conference endpoint 100 receives an audio-video data stream from the moderator remote conference endpoint 104. Further, as discussed at [1.2.2], Drell teaches that the near conference endpoint 100 also receives an audio-video data stream from the remote conference endpoint 102.

231. **Second**, the phrase “second mixed data stream” is construed as a *second data stream including at least audio data and video data*. See Claim Construction Section, above.

232. As discussed at [1.1], Drell’s MP conferencing application includes an “audio mixing module 308” that is “configured to combine audio data received from remote conference endpoints 102 and 104.” Drell (Ex. 1008) at 4:32-38; 5:8-10. Drell’s MP conferencing application further includes a “video switching/continuous presence module 306” that “combines video data received from remote conference endpoints 102 and 104.” *Id.* at 5:26-32. Finally, Drell’s MP conferencing application includes a “communication process 404” that “multiplexes the encoded audio and video data streams into a **single audio/video data stream**.” *Id.* at 6:46-48.

233. Thus, Drell’s audio mixing module 308, and video switching/continuous presence module 306, and communication process 404 collectively operate to mix the audio and video data streams received from remote conference endpoints 102 and 104 into a single audio and video data stream, which renders obvious a “*second mixed data stream*.”

234. As discussed immediately above, Drell’s audio mixing module 308, and video switching/continuous presence module 306, and communication process 404 collectively produce the claimed “*second mixed data stream*.” As discussed at

[1.1], Drell teaches instantiating a separate communication process 404 for each remote conference endpoint. *See e.g. supra* ¶¶ 203-211. As discussed below at [1.4], an audio data stream corresponding to the “*second mixed data stream*” is transmitted to the “*PSTN client*.” Because the “*second mixed data stream*” relates to the audio data stream transmitted to the “*PSTN client*” (*see infra*, [1.4]), it would have been obvious to a POSITA for the communication process 404 producing the “*second mixed data stream*” to be the communication process 404 instantiated to support the PSTN conference endpoint (“*PSTN client*,” *see* [1.1]). It would have been obvious to a POSITA, therefore, that Drell’s audio mixing module 308, video switching/continuous presence module 306, and communication process 404 instantiated for the PSTN conference endpoint collectively teach the claimed “*second mixer*.”

235. **Additionally**, for the reasons analogous to those discussed at [1.1], it would have been obvious to a POSITA to implement audio mixing module 308 and video switching/continuous presence module 306 such that they separately mix audio and video for each conference endpoint. *See e.g. supra* ¶¶ 203-211. Thus, it would have also been obvious to a POSITA to employ a separate audio mixer, a separate video mixer, and the communication process 404 instantiated for the PSTN conference endpoint, which collectively constitute the claimed “*second mixer*.”

236. The discussion above explains that it would have been obvious to a POSITA for Drell's MP conferencing application to produce a "*second mixed data stream*" including both audio and video data. As discussed above regarding the "*second mixer*," claim 1 subsequently recites that audio corresponding to the "*second mixed data stream*" is transmitted to the "*PSTN client*." It would have been obvious to a POSITA to produce a "*second mixed data stream*" including both audio and video data for the "*PSTN client*" because it would have been obvious for the "*PSTN client*" to support both audio and video conferencing, as discussed in the following paragraphs.

237. Drell describes that the near conference endpoint 100 can "connect to conference endpoints" over "the PSTN," (Drell (Ex. 1008) at 3:20-38) but does not provide detail regarding the components of the PSTN conference endpoint; that is, Drell does not describe how the PSTN conference endpoint communicates both audio and video. Knappe complements Drell by describing a "telephone 18 and gateway 20 combination" that connects to a videoconference system via the PSTN. Knappe (Ex. 1006) at 3:66-67. The telephone and gateway are "capable of communicating real-time **audio, video** and/or other information." *Id.* at 3:66-4:1. It would have been obvious to a POSITA, therefore, for the PSTN conference endpoint and gateway to support communicating both audio and video data. That the PSTN conference endpoint supports both audio and video is consistent with

Claim 6, which recites “*transmitting the second mixed data stream to the PSTN client.*”

238. Voois supplements Drell and Knappe by describing a PSTN conference endpoint that is a videophone capable of communicating both video and audio:

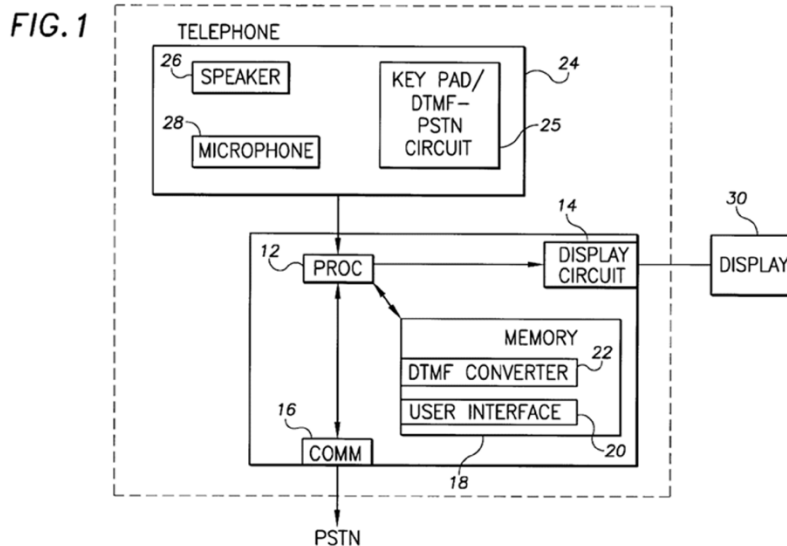
Turning now to the drawings, FIG. 1 is a block diagram illustrating an example system having a telephone for operating a **videophone** user interface. The system includes a processor 12, such as a RISC processor, coupled to a display circuit 14 and a communication component 16. Communication component 16 includes hardware along with supporting firmware and/or software, and is included to provide an interface with a communication network, such as a public switched telephone network (PSTN), the internet, local area networks, and/or wide area networks.

Voois (Ex. 1036) at 3:62-4:4.

The system of FIG. 2 includes a first terminal 110 communicating with a second, similarly-constructed terminal 112. The communication takes place with modem functions provided by processor circuit 120 for **transmitting (and receiving) audio and video data** over a communications channel 114.

*Id.* at 4:32-39.

239. The general organization of Voois’ videophone, including its video display and connection to the PSTN, is shown in Figure 1:



*Id.* at FIG. 1.

240. A POSITA would have recognized that Voois's PSTN videophone teachings are well suited for Drell's system (as modified by Knappe to include a gateway capable of communicating both audio and video). Thus, it would have been obvious to a POSITA for Drell's PSTN conference endpoint to be a videophone, as taught by Voois, because it would beneficially allow for the user of the PSTN conference endpoint (implemented as a videophone) to exchange audio and video with the other conference participants, thereby improving the experience of the user. *See also* Reasons to Combine Drell and Voois, *supra* ¶¶ 169-173.

241. With Drell's video conference system supporting a videoconferencing PSTN conference endpoint (as suggested by Knappe and Voois), it would have been obvious to a POSITA that Drell's MP conferencing application (and specifically, an instance of communication process 404) would produce a mixed

audio-and-video data stream for the videoconferencing PSTN conference endpoint. This approach would have been obvious because it would allow for providing the same rich, multimedia conferencing experience to PSTN-based participants as is provided to participants using other forms of videoconferencing equipment. *See* U.S. Patent No. 6,237,025 (Ex. 1018) at 1:28-47 (noting that a conference providing a “combination of spoken words, gestures, visual cues...significantly enhances the effectiveness of collaboration”).

242. Thus, Drell in combination with Knappe and Voois teaches that the near conference endpoint 100 has an audio mixing module 308 (with separate summers for each participant) that mixes audio data streams from the remote conference endpoint 102 and from the moderator remote conference endpoint 104, a video switching/continuous presence module 306 (with a separate summer for each participant) that mixes video data streams from the remote conference endpoint 102 and from the moderator remote conference endpoint 104, and a communication process 404 of the PSTN conference endpoint, which further mixes the mixed audio data stream (output by audio mixing module 308) and the mixed video data stream (output video switching/continuous presence module 306) into a second mixed data stream for the PSTN conference endpoint, which renders obvious “*a second mixer that mixes the moderator audio-video data stream with*

*the remote client audio- video data stream into a second mixed data stream,” as claimed.*

***[1.4] a second transport output that transmits a mixed audio data stream, corresponding to the second mixed data stream, to the PSTN client.***

243. The prior art renders obvious a second transport output that transmits a mixed audio data stream, corresponding to the second mixed data stream, to the PSTN client.

244. **First**, as discussed at [1.3], Drell, Knappe, and Voois teach that communication process 404 combines mixed audio and video data into a “single audio/video data stream” corresponding to the claimed “*second mixed data stream.*” *See e.g.*, Drell (Ex. 1008) at 6:46-48; *see also supra* ¶¶ 233-234. Further, as discussed at [1.1] and [1.3], Drell in combination with Knappe and Voois teaches that the PSTN conference endpoint (implemented as a videophone) is the “*PSTN client.*” *See e.g. supra* ¶¶ 181-183, 236-242.

245. Drell teaches “**transmitting** the combined audio and video signals to each of the plurality of remote conference endpoints.” Drell (Ex. 1008) at 7:30-31. As discussed above in [1.1], Drell, alone or together with Knappe, renders obvious producing a separate mixed data stream for each remote conference endpoint. *See supra* ¶¶ 203-211. Thus, it would have been obvious to a POSITA to “*transmit[]*” the “*second mixed data stream*” to a remote conferencing endpoint, such as the videophone PSTN conference endpoint (“*PSTN client*”) taught by Drell and Voois.

As discussed in [1.3], the “*second mixed data stream*” includes the mixed audio data stream produced by Drell’s audio mixing module 308. *See supra* ¶¶ 232-242. Because the mixed audio data stream is part of the “*second mixed data stream*,” the mixed audio data stream corresponds to it. In other words, the “*mixed audio data stream*” is the audio portion of the “*second mixed data stream*.” Transmitting the “*second mixed data stream*” therefore includes transmitting a “*mixed audio data stream, corresponding to the second mixed data stream*.”

246. Thus, Drell in combination with Knappe and Voois teaches transmitting a mixed audio-video data stream to the PSTN conference endpoint (implemented as a videophone), which renders obvious to “*transmit[] a mixed audio data stream, corresponding to the second mixed data stream, to the PSTN client*.”

247. **Second**, as discussed at [1.2.1], the output of Drell’s communication process 404 is a “*transport output*.” *See supra* ¶¶ 213-220. Drell further teaches instantiating a separate communication process 404 (part of a “processing train”) for each conference endpoint. Drell (Ex. 1008) at 4:21-24; 5:65-6:2. Thus, the output of communication process 404 instantiated for the PSTN conference endpoint is a “*second transport output*.”

248. **Additionally**, as discussed at [1.1], it would have been obvious to a POSITA to utilize a gateway, as Knappe teaches, in Drell’s system. *See e.g. supra*

¶ 183. Specifically, Knappe teaches utilizing a gateway positioned between an Internet network and a PSTN for converting data into a suitable format for each network:

**The gateway [] provides conversion between analog and/or digital formats.** The standard analog telephones [] communicate standard telephony signals through PSTN [network] to the gateway []. At the gateway [], the signals are converted to IP packets in the VoIP format. Similarly, VoIP packets received from the [Internet] network [] are converted into standard telephony signals for delivery to the destination telephone [] through PSTN [network].

Knappe (Ex. 1006) at 4:18-25, FIG. 1; *see also id.* at 3:50-58. Knappe

further explains that the gateway is capable of communicating both audio and video:

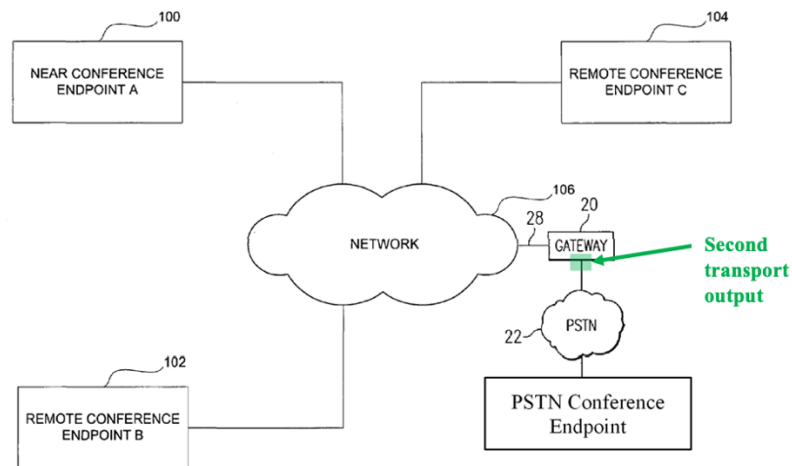
[T]he telephone 18 and **gateway 20** combination [is] capable of communicating real-time **audio, video** and/or other information over the network 14.

*Id.* at 3:65-4:1. This is illustrated in Knappe's Figure 1, where gateway 20 is positioned between Internet network 14 and PSTN 22 to perform conversion between analog and/or digital formats for each network.

249. Since Drell contemplates that its conference endpoints communicate over "PSTN and the Internet" networks (Drell (Ex. 1008) at 3:20-38), it would have been obvious to a POSITA to utilize a gateway positioned between the PSTN and Internet networks, as Knappe teaches, because it would provide Drell the benefit of converting Internet network communications to suitable signals for the

PSTN, and vice-versa (Knappe (Ex. 1006) at 4:18-25). *See also* Reasons to Combine Drell and Knappe, *supra* ¶¶ 148-168.

250. As shown in the annotated figure below (combining portions of Drell’s Figure 1 and Knappe’s Figure 1), the gateway 20 has an output that couples it to a PSTN conference endpoint. It would have been obvious to a POSITA that the gateway would transmit at least the audio portion of the “*second mixed data stream*” to the PSTN conference endpoint (“*PSTN client*”) so that the conference participant using the PSTN conference endpoint could hear the audio portion of the conference (that is, the speech of other conference participants).



251. Moreover, in the combination with Voois, it would have been obvious to a POSITA for the gateway (which is “capable of communicating real-time audio [and] video,” Knappe (Ex. 1006) at 3:65-41) to also transmit the video portion of

the second mixed data stream to the PSTN conference endpoint (implemented as a videophone).

252. Thus, Drell, Knappe, and Voois teach that a PSTN gateway has a second transport output that transmits a mixed audio data stream corresponding to the second mixed data stream, in the form of standard telephony signals for delivery to a PSTN conference endpoint, which renders obvious “*a second transport output that transmits a mixed audio data stream, corresponding to the second mixed data stream, to the PSTN client,*” as claimed.

#### 4. Claim 2

***[2.P] The system of claim 1 further comprising:***

253. See analysis for [1.P]-[1.4].

***[2.1.1] a third mixer that mixes the audio data from the PSTN client with the audio-video data stream from the remote client into a third mixed data stream and***

254. The prior art renders obvious a third mixer that mixes the audio data from the PSTN client with the audio-video data stream from the remote client into a third mixed data stream.

255. **First**, as discussed at [1.1], Drell in combination with Knappe teaches that the near conference endpoint 100 receives audio data stream from the PSTN conference endpoint. See *supra* ¶¶ 184-194. Further, as discussed at [1.2.2], Drell teaches that the near conference endpoint 100 also receives an audio-video data stream from the remote conference endpoint 102. See *supra* ¶¶ 223-228.

256. **Second**, the phrase “*third mixed data stream*” is construed as a *third data stream including at least audio data and video data*. See Claim Construction.

257. As discussed at [1.1], Drell’s MP conferencing application includes an “audio mixing module 308” that is “configured to combine audio data received from remote conference endpoints 102 and 104.” Drell (Ex. 1008) at 4:32-38; *see also id.* at 5:8-10. As discussed in [1.1], a POSITA would understand that the audio data received from the PSTN conference endpoint would also be combined in the audio mixing module 308. *See supra* ¶¶ 196-212. Drell’s MP conferencing application further includes a “video switching/continuous presence module 306” that “combines video data received from remote conference endpoints 102 and 104.” Drell (Ex. 1008) at 5:26-32. Finally, Drell’s MP conferencing application includes a “communication process 404” that “multiplexes the encoded audio and video data streams into **a single audio/video data stream.**” *Id.* at 6:46-48.

258. Thus, Drell’s audio mixing module 308, and video switching/continuous presence module 306, and communication process 404 (of the processing train 304) collectively operate to mix the data streams received from the PSTN conference endpoint and remote conference endpoint 102 into a single audio and video data stream for the moderator remote conference endpoint 104, which renders obvious a “*third mixed data stream.*”

259. **Third**, as discussed immediately above, Drell's audio mixing module 308, and video switching/continuous presence module 306, and communication process 404 collectively produce the claimed "*third mixed data stream*." As discussed at [1.1], Drell teaches instantiating a separate communication process 404 for each remote conference endpoint. Because the "*third mixed data stream*" is communicated to the moderator remote conference endpoint 104 (*see infra*, [2.1.2]), it would have been obvious to a POSITA for the communication process 404 producing the "*third mixed data stream*" to be the communication process 404 instantiated to support the remote conference endpoint 104 (used by the "*moderator*," *see* [1.1]). It would have been obvious to a POSITA, therefore, that Drell's audio mixing module 308, and video switching/continuous presence module 306, and the communication process 404 (of the processing train 304) instantiated for the moderator remote conference endpoint 104 collectively teach the claimed "*third mixer*."

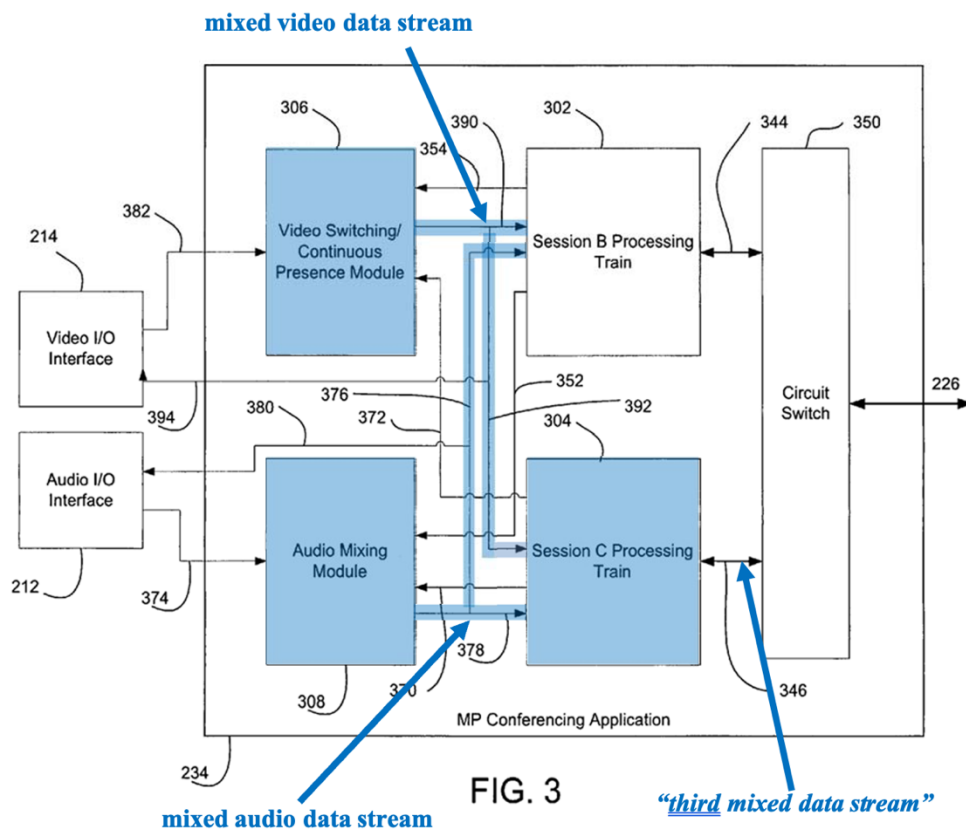


FIG. 3

*Id.* at FIG. 3 (annotated).

260. **Additionally**, for the reasons analogous to those discussed at [1.1], it would have been obvious to a POSITA to implement audio mixing module 308 and video switching/continuous presence module 306 such that they separately mix audio and video for each conference endpoint. Thus, it would have also been obvious to a POSITA that the separate audio mixer, the separate video mixer, and the communication process 404 instantiated for remote conference endpoint 104 collectively constitute the claimed “*third mixer*.”

261. Thus, Drell in combination with Knappe teaches that the near conference endpoint 100 has an audio mixing module 308 (with separate summers for each participant) that mixes audio data streams from the PSTN conference endpoint and from the remote conference endpoint 102, a video switching/continuous presence module 306 (with separate summers for each participant) that mixes video data streams from the remote conference endpoint 102, and a communication process 404 of the moderator remote conference endpoint 104, which further mixes the mixed audio data stream (output by audio mixing module 308) and the mixed video data stream (output video switching/continuous presence module 306) into a third mixed data stream for the moderator remote conference endpoint 104, which renders obvious “*a third mixer that mixes the audio data from the PSTN client with the audio-video data stream from the remote client into a third mixed data stream,*” as claimed.

**[2.1.2] communicates the third mixed data stream to the moderator**

262. The prior art renders obvious a third mixed data stream is communicated to the moderator.

263. **First**, as discussed at [1.1], Drell in combination with Knappe teaches moderator remote conference endpoint 104. *See supra* ¶¶ 187-189. Further, as discussed at [2.1.1], Drell in combination with Knappe teaches generating the third mixed data stream.

264. **Second**, Drell teaches that the near conference endpoint 100 has a processing train 304 that uses a communication process to transmit mixed data as a single stream to the moderator remote conference endpoint 104 via network 106:

[N]etwork 106 may comprise a packet switched network, such as the Internet...Console electronics 213 additionally include a central processing unit (CPU) 220 for executing program instructions, a memory 222 for storing applications, data, and other information, and **a network interface 224 for connecting near conference endpoint 100 to network 106**

Drell (Ex. 1008) at 3:33-58.

Generally, the MP conferencing application generates separate processing trains for **signal streams...to each of the remote conference endpoints.**

*Id.* at 2:19-21.

The combined audio and video data streams are conveyed as input to each processing train for encoding and transmission to the corresponding conference endpoints. In the send mode, the audio and video signals are encoded by the audio/video codecs and multiplexed into a **single data stream by the communication process.**

*Id.* at 2:43-48.

The outputs of the audio mixer and video switching module are sent to each of the plurality of signal **processing trains, which process the combined signals according to a transmit mode for distribution to the remote endpoints over the network**

*Id.* at Abstract.

In the transmission mode, processing train 302 is configured to receive the audio and video data streams.... The received data streams are then encoded and combined to form a mixed encoded audio/video data stream, and the encoded audio/video data stream is transmitted...Similarly, **processing train 304** receives the audio and

video streams..., encodes and combines the audio and video data streams, and **transmits the encoded audio/video data stream.**

*Id.* at 5:48-60

In the example depicted in the figures, the circuit switch 350 dynamically allocates two IMUXs and generates two **processing trains 302 and 304 respectively corresponding to remote conference endpoints 102 and 104**

*Id.* at 4:44-48.

FIG. 1 depicts an exemplary operating environment of the multi-point (MP) conferencing application of the present invention. A near conference endpoint 100, embodying the MP conferencing application, is coupled to remote conference endpoints 102 and 104 via a network **Remote conference endpoints 102 and 104** may comprise, for example, conventional videoconferencing devices equipped to transmit and **receive both video (image) data and audio (speech) data...**It should be appreciated that while only two remote conference endpoints are depicted in FIG. 1 for the purpose of clarity, a greater number of remote conference endpoints may be accommodated by near conference endpoint 100

*Id.* at 3:4-19; *see also id.* at FIGs. 1-5.

265. As shown in Figure 1, the near conference endpoint 100 is communicatively coupled to moderator remote conference endpoint 104 via network 106:

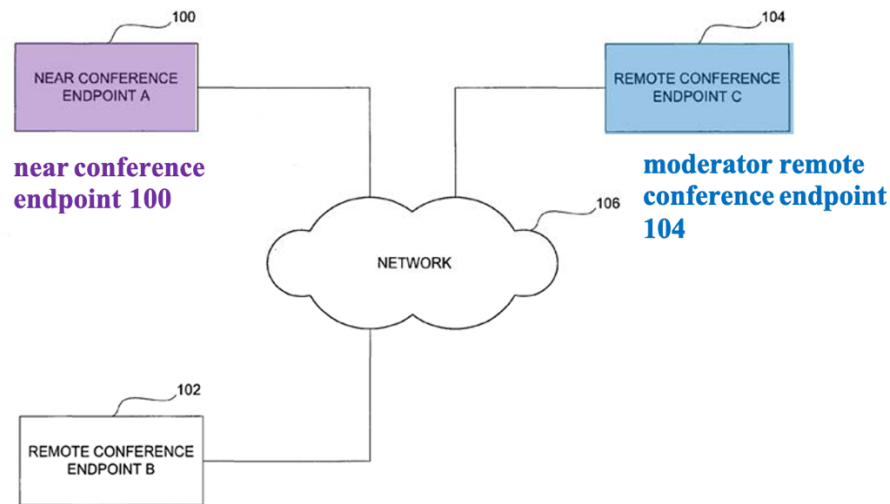


FIG. 1

*Id.* at FIG. 1 (annotated).

266. In view of the above disclosure, it would have been obvious to a POSITA that when the audio and video data are mixed and encoded into a third mixed data stream by the third mixer (as discussed at [2.1.1]), the communication process 404 (of processing train 304) communicates the third mixed data stream (via path 346, *see id.* FIG. 3) to the moderator remote conference endpoint 104 via the network 106.

267. Thus, Drell in combination with Knappe teaches that the third mixer communicates the third mixed data stream to the moderator remote conference endpoint 104, which renders obvious “*communicat[ing] the third mixed data stream to the moderator,*” as claimed.

## 5. Claim 3 Is Obvious in Light of Drell in Combination with Knappe and Voois

***[3.P] The system of claim 1 further comprising:***

268. See analysis for [1.P]-[1.4].

**[3.1] a VoIP decoder that receives IP packets including the audio data stream generated by the PSTN client, the VoIP decoder decodes the IP packets including the audio data from the PSTN client.**

269. The prior art, in multiple ways, renders obvious a VoIP decoder that receives IP packets including the audio data stream generated by the PSTN client and that decodes the IP packets including the audio data from the PSTN client.

270. **First**, as discussed at [1.1], Drell in combination with Knappe teaches that the near conference endpoint 100 receives an audio data stream from the PSTN conference endpoint, via a gateway. See e.g. *supra* ¶¶ 184-186.

271. **Second**, Drell teaches that the audio traverses two or more networks comprising an Internet network and a PSTN network:

[N]etwork 106 may comprise a packet switched network, such as the Internet. Although a single network 106 is shown, the invention contemplates the use **of two or more networks (for example, the PSTN and the Internet)** to connect conference endpoints utilizing different communication protocols.

Drell (Ex. 1008) at 3:33-38.

272. It would have been obvious to a POSITA that near conference endpoint 100 “receives IP packets including the audio data stream generated by the PSTN client” since the conference endpoint 100 is “connect[ed to other] conference endpoints” over the “PSTN and the Internet” because communications over the Internet use IP packets. See, e.g., Knappe (Ex. 1006) at 3:53-54.

273. As already discussed above, it would have been obvious to utilize a gateway as Knappe teaches in Drell's system. Regarding the gateway, Knappe teaches that audio data stream from a PSTN conference endpoint is transmitted over the PSTN network to a gateway that converts the audio into VoIP packets and then encapsulates the packets in RTP packets for transmission over an Internet network:

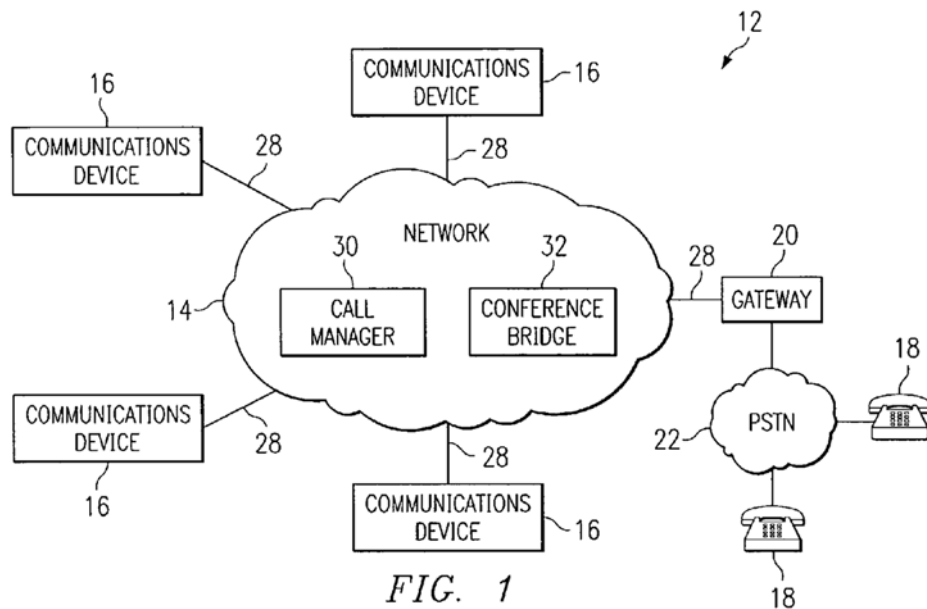
The gateway 20 provides conversion between analog and/or digital formats. The standard analog telephones 18 communicate standard telephony signals through PSTN 22 to the gateway 20. **At the gateway 20, the signals are converted to IP packets in the VoIP format.** Similarly, VoIP packets received from the network 14 are converted into standard telephony signals for delivery to the destination telephone 18 through PSTN 22.

*Id.* at 4:18-25.

In accordance with a particular embodiment, **network 14 is the Internet, a wide area network (WAN), a local area network (LAN) or other suitable packet-switched network.** In the Internet embodiment, the network 14 transmits information in Internet Protocol (IP) packets. Telephony voice information is transmitted in the Voice over IP (VoIP) format. **Real-time IP packets such as VoIP packets are encapsulated in real-time transport protocol (RTP) packets for transmission over the network 14.** It will be understood that the network 14 may comprise any other suitable types of elements and links and that traffic may be otherwise suitably transmitted using other protocols and formats.

*Id.* at 3:50-61.

274. As shown below at Figure 1, Knappe utilizes gateway 20 to convert audio signals received from PSTN conference endpoint (e.g., telephone 18) into VoIP packets for transmission over Internet network 14:



*Id.* at FIG. 1.

275. Thus, it would have been obvious that in the combination with Knappe, the near conference endpoint 100 would “*receive[] IP packets including the audio data stream generated by the PSTN client*” because Knappe describes how “[a]t the gateway 20, the signals [from a PSTN conference endpoint] are converted to **IP packets in the VoIP format**” and that the packets are “**encapsulated in real-time transport protocol (RTP) packets.**” *Id.* at 4:18- 25, 3:50-61.

276. It would have been obvious to a POSITA to utilize a gateway (as Knappe teaches) in Drell’s system, because it would allow for converting audio signals from Drell’s PSTN conference endpoint into VoIP packets that are encapsulated in RTP packets for transmission over the Internet network 106 to the

near conference endpoint 100. That is, by utilizing a gateway between Drell's PSTN and Internet networks, endpoints on these networks can beneficially communicate audio data to each other through the gateway, which converts standard telephone signals to VoIP packets that are encapsulated in RTP packets for transmission over the Internet network and converts received packets to standard telephone signals for transmission over the PSTN network. A POSITA would have recognized that RTP was specifically created for applications such as Drell's real-time conferencing application which communicates audio over the Internet. *See* Drell (Ex. 1008) at 3:20-38 ("Network 106 may be of any type suitable for the transmission of audio and video data between and among near conference endpoint 100 and remote conference endpoints... [N]etwork 106 may comprise a packet switched network, such as the Internet...to connect conference endpoints."); *see also* Caputo (Ex. 1021) at 22 ("The Real-Time Protocol (RTP)...was created to transport real-time traffic, such as video and audio, over the Internet."); *see also* Reasons to Combine Drell and Knappe, *supra* ¶¶ 148-168.

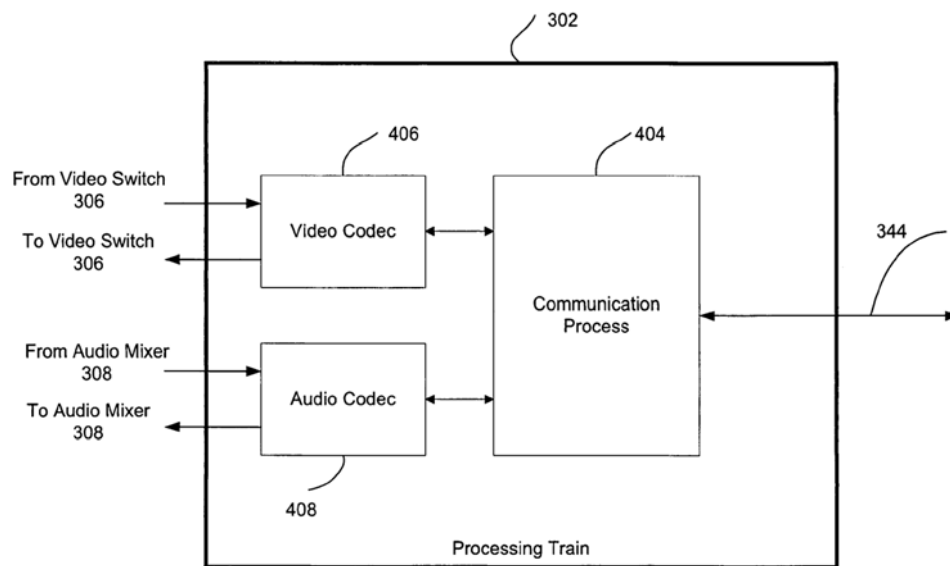
277. **Third**, Drell further teaches that packets from the conference endpoints are received and decoded by communication process 404 and audio codec 408 (using a standard or proprietary compression algorithm):

**Audio codec 408 receives the audio data stream from communication process 404 and...decod[es] in accordance with a**

**standard** (e.g., ITU Recommendation G.711) **or proprietary audio compression algorithm**. The decoded audio data stream is then sent to audio mixing module 308, as described above.

Drell (Ex. 1008) at 6:18-23; *see also id.* at 5:8-23, FIGs. 3, 4.

278. As already explained, Drell’s “network 106 may comprise a **packet** switched network, such as the **Internet**.” *Id.* at 3:33-38. As shown below at Figure 4, Drell utilizes communication process 404 and audio codec 408 to receive IP packets from Internet network 106 and decode the IP packets, whose audio is sent to audio mixer 308:



*Id.* at FIG. 4.

279. A POSITA would understand that the audio codec 408 decodes the IP packets including the audio data from the PSTN client because the audio data must be decoded before it can be mixed. Accordingly, Drell’s disclosure of a communication process and an audio codec configured to receive and decode IP

packets (which in the combination with Knappe are VoIP packets encapsulated in RTP packets) received from the PSTN conference endpoint via a gateway (as discussed above), renders obvious “*a VoIP decoder*,” as claimed.

280. **Additionally**, Drell teaches that its communication process includes a H.323 (packet-based) process:

The (MP) conferencing application is operable to process the plural signal streams received through the communication ports. Generally, the MP conferencing application generates separate processing trains for **signal streams from/to each of the remote conference endpoints**. The processing trains each comprise a communication **process and a set of codecs**...The communication process...may for example comprise an...**H.323 (packet-based) process**...The audio and video signals are...respectively delivered to audio and video codecs for decoding

*Id.* at 2:17-33.

281. It would have been obvious to a POSITA that Drell’s H.323 process includes a “*VoIP decoder*” that “*decodes*...” as claimed because H.323 was a well-known standard that specifies encoding and decoding VoIP communication. *See, e.g.* U.S. Patent No. 6,553,515 (Ex. 1025) at 4:35-56 (“**H.323** is adapted to support many aspects of multimedia...[and] generally is regarded as the **standard for VoIP**...This standard also specifies lowest-common denominator **compression protocols, such as G.711...G.723 and G.729**.”). It also would have been obvious to a POSITA for a packet-based H.323 process to use IP and RTP packets for communication because the standard specifically refers to using it. *See* U.S. Patent

*Id.* at 3:57-65 (“Once digitized, the encoded voice data is then wrapped within **IP packets**, using a **Real-Time Transport Protocol (RTP)**, which runs over UDP...based on the **H.323 Internet telephony standard.**”); H.323 Standard (Ex. 1033) at Fig. 13, 39, 146 (“IP-based H.323 endpoints”), 189 (“H.323 and H.225.0 procedures using RTP/UDP/IP to convey the media streams”); Caputo (Ex. 1021) at 22 (“The Real-Time Protocol (RTP)...was created to transport real-time traffic, such as video and audio, over the Internet.”). Since Drell comprises an H.323 process, it would have been obvious to a POSITA that Drell’s system utilizes a VoIP decoder. In fact, the ’525 patent’s VoIP decoder is simply a well-known prior art H.323 codec. *See, e.g.*, ’525 patent (Ex. 1001) at 2:58-67, 9:37, 9:49-51, FIG. 2.

282. **Additionally**, Knappe teaches that the conference endpoints utilize an audio codec to decode received VoIP packets encapsulated in RTP packets:

[N]etwork 14 is the Internet...[which communicates using] Real-time **IP packets such as VoIP packets...encapsulated in real-time transport protocol (RTP) packets.**

Knappe (Ex. 1006) at 3:50-57.

A **codec (coder/decoder) converts** audio, video...**IP packets** received from the network 14 are converted back into analog signals.

*Id.* at 4:9-14.

283. Consistent with Knappe’s teaching of a PSTN gateway to convert data between an analog format for the PSTN and an RTP VoIP format for the Internet

(as discussed above), it would have been obvious to a POSITA to utilize in Drell's Internet-connected conference endpoints a corresponding audio codec configured to decode received VoIP packets encapsulated in RTP packets as Knappe teaches, because it would beneficially allow for Drell's system to utilize protocols (i.e., RTP and VoIP) specifically created for real-time voice data transmitted over the Internet. *See* Drell (Ex. 1008) at 3:20-38; Caputo (Ex. 1021) at 22 ("The Real-Time Protocol (RTP)...**was created to transport real-time traffic**, such as video and audio, over the Internet."); *see also* Reasons to Combine Drell and Knappe, *supra* ¶¶ 148-168. Knappe's codec also renders obvious a "*VoIP decoder*" as claimed.

284. In the combination, Drell's PSTN conference endpoint sends audio signals over the PSTN network to a gateway which converts the signals into VoIP packets that are encapsulated in RTP packets (per Knappe's teachings). The packets are then transmitted over Drell's Internet network 106 to the near conference endpoint 100 which uses the VoIP decoder to receive and decode the packets containing audio from the PSTN conference endpoint.

285. Thus, Drell in combination with Knappe teaches that the near conference endpoint 100 includes a VoIP decoder that receives and decodes VoIP packets containing audio data from the PSTN conference endpoint, which renders obvious "*a VoIP decoder that receives IP packets including the audio data stream*

*generated by the PSTN client, the VoIP decoder decodes the IP packets including the audio data from the PSTN client,” as claimed.*

**6. Claim 4 Is Obvious in Light of Drell in Combination with Knappe and Voois**

***[4.P] The system of claim 1 further comprising:***

286. See analysis for [1.P]-[1.4].

***[4.1] at least one audio decompressor that decodes IP packets including the audio-video data stream generated by the remote client.***

287. The prior art, in multiple ways, renders obvious at least one audio decompressor that decodes IP packets including the audio-video data stream generated by the remote client.

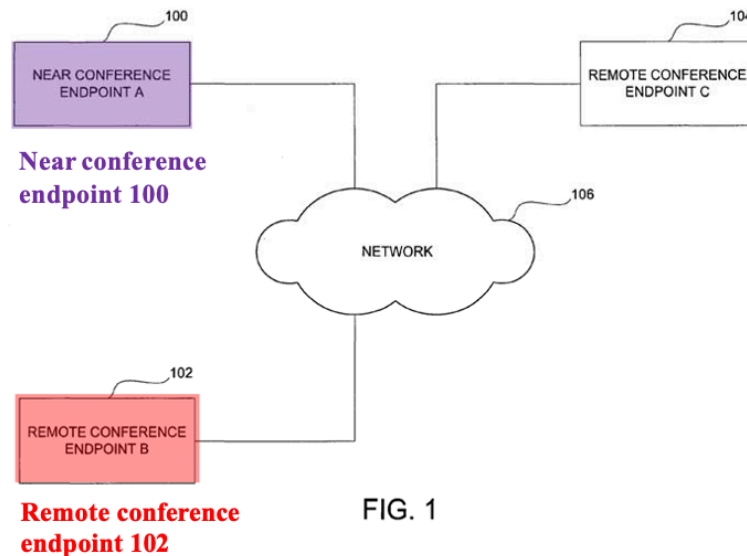
288. **First**, as discussed at [1.2.1], and [1.3], Drell in combination with Knappe teaches that the near conference endpoint 100 receives an audio-video data stream from the remote conference endpoint 102.

289. Also, as noted, Drell teaches that the audio-video data stream is received over a packet switched network, such as the Internet:

[N]etwork 106 may comprise a **packet** switched network, such as the **Internet**. Although a single network 106 is shown, the invention contemplates the use of two or more networks (for example, the PSTN and the Internet) to connect conference endpoints utilizing different communication protocols.

Drell (Ex. 1008) at 3:33-38; *see also id.* at 2:26-28 (“The communication process...may for example comprise an...H.323 (**packet-based**) process.”).

290. As shown below at Figure 1, Drell's near conference endpoint 100 receives an audio-video data stream from the remote conference endpoint 102 over Internet network 106:



*Id.* at FIG. 1.

291. Since the near conference endpoint 100 receives the audio-video data stream from the remote conference endpoint 102 over the Internet network 106, it would have been obvious to a POSITA that it is received as “IP packets.” *See* Knappe (Ex. 1006) at 3:50-55 (“In the Internet embodiment, the network... transmits information in Internet Protocol (IP) packets.”).

292. **Second**, Drell teaches that packets received from the remote conference endpoint 102 are separated by communication process 404 into audio and video signals and that the audio signals are decoded by audio codec 408 (using a standard or proprietary compression algorithm):

In particular, **communication process 404 includes a multiplexer/demultiplexer for (in the receive mode) extracting separate audio and video signals from mixed data stream 344** in accordance with ITU Recommendation H.221. Communication process 404 may further include a delay compensation process for inducing a delay in the audio data path in order to maintain lip synchronization. A system control unit is incorporated into communication process 404 and is configured to establish a common mode of operation with **remote conference endpoint 102** in accordance with ITU Recommendation H.242.

Drell (Ex. 1008) at 6:7-17.

**Audio codec 408 receives the audio data stream from communication process 404 and applies redundancy reduction decoding in accordance with a standard (e.g., ITU Recommendation G.711) or proprietary audio compression algorithm.** The decoded audio data stream is then sent to audio mixing module 308, as described above.

*Id.* at 6:18-23; *see also id.* at FIGs. 3, 4.

Audio mixing module 308 is configured to combine **audio data received from remote conference endpoints 102...**For example, audio mixing module 308 may simply mix the received audio input data streams.

*Id.* at 5:8-23.

293. A POSITA would have understood that decoding audio using an audio compression algorithm is performing decompression. Drell's audio codec that utilizes an audio compression algorithm to decompress IP packets containing audio- video data stream from remote conference endpoint 102 discloses an “*audio decompressor,*” as claimed.

294. **Additionally**, Drell teaches that its communication process includes a H.323 (packet-based) process:

The (MP) conferencing application is operable to process the plural signal streams received through the communication ports. Generally, the MP conferencing application generates separate processing trains for **signal streams from/to each of the remote conference endpoints**. The processing trains each comprise a communication **process and a set of codecs**...The communication process...may for example comprise an...**H.323 (packet-based) process**...**The audio and video signals are...respectively delivered to audio and video codecs for decoding**.

*Id.* at 2:17-33.

295. It would have been obvious to a POSITA that Drell's H.323 process includes an "*audio decompressor*" because H.323 was a well-known standard that specified certain compression protocols. *See, e.g.* U.S. Patent No. 6,553,515 (Ex. 1025) at 4:35-56 ("**H.323** is adapted to support many aspects of multimedia...[and] generally is regarded as the **standard for VoIP**...This standard also specifies lowest-common denominator **compression protocols, such as G.711...G.723 and G.729**."). In fact, the '525 patent's decompressor is simply a well-known prior art H.323 codec. *See, e.g.*, '525 patent (Ex. 1001) at 2:56-62, FIG. 2 ("The H.323 protocol stack consists of an audio codec 214, and a video CoDec 213 connected to the audio/video input and output blocks 452 and 451, respectively. The audio and video CoDecs are responsible for compression and decompression of the audio and video sources.").

296. Thus, Drell in combination with Knappe teaches that the near conference endpoint 100 includes audio decompression algorithm or protocol that

decodes IP packets including the audio-video data stream generated by the remote conference endpoint 102, which renders obvious “*at least one audio decompressor that decodes IP packets including the audio-video data stream generated by the remote client,*” as claimed.

**B. Claims 5-8 Are Obvious Over Drell in Combination with Knappe, Voois, Elliott, and the VPN Textbook**

**1. The Prior Art**

**a) Summary of Elliott**

297. U.S. Patent No. 6,690,654 to Elliott et al. was filed on October 1, 1998, and issued on February 10, 2004. Like Drell, Elliott relates generally to the field of “communicating with a plurality of remote parties...for multi-media collaboration.” Elliott (Ex. 1007) at Abstract.

298. Elliott teaches that it was well known that the data in the transmitted stream includes “collaborative Web browsing, audio conferencing, video conferencing, and application sharing.” *Id.* at 1:40-44. Further, Elliott teaches utilizing a “virtual private network (VPN)...which uses IP or IP-tunneling” for the transmission of “voice, fax, data, video” and “[o]ther types of collaboration...data” during the conference. *Id.* at 2:66-3:1, 4:29-34, 6:53-54.

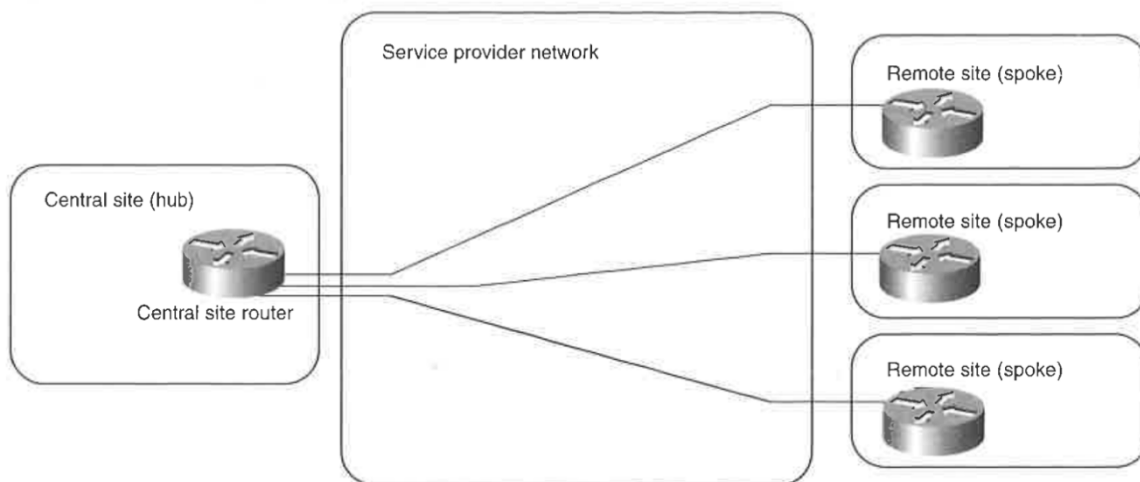
## b) Summary of the VPN Textbook

299. The textbook “MPLS and VPN Architectures, A practical guide to understanding, designing and deploying MPLS and MPLS-enabled VPNs,” (Ex. 1010, “VPN Textbook”) by Guichard et al., was published in 2001 by Cisco Press.

300. The VPN Textbook describes well-known topologies for implementing VPNs. Specifically, under the heading “Typical VPN Network Topologies,” the VPN Textbook describes that “[t]he most commonly encountered topology is a hub- and-spoke topology, where a number of remote offices (spokes) are connected to a central site (hub).” *Id.* at 129. The “hub-and-spoke” topology is also referred to as a “star” topology. *See* Computer Desktop Encyclopedia (Ex. 1027) at 438.

301. The hub-and-spoke (or star) topology is illustrated in Figure 7-10 of the VPN Textbook, below:

**Figure 7-10** *Hub-and-spoke Topology*



*Id.* at 130.

302. The VPN Textbook explains that “hub-and-spoke VPN topology is more common than you might expect.” *Id.* at 129. According to the VPN Textbook, even where an implementation may “benefit from a different topology” “the hub-and-spoke topology [is chosen] for cost or complexity reasons.” *Id.*

## **2. Reasons to Combine Prior Art**

### **a) Reasons to Combine Drell and Elliott**

303. A POSITA would have been motivated to combine the teachings of Drell and Elliott for multiple reasons, including to produce the obvious, beneficial, and predictable results of utilizing VPN tunneling within Drell’s system.

304. I initially note that one of ordinary skill in the art when considering the teachings of Drell would have also considered the teachings of Elliott, as they are analogous prior art both pertaining to the same field of endeavor of multiparty conferencing. Drell (Ex. 1008) at Abstract (“A videoconferencing apparatus includes a multi- point (MP) conference application that enables the apparatus to combine and distribute audio and video signals received from a plurality of remote conference endpoints.”); Elliott (Ex. 1007) at Abstract (“A method for communicating with a plurality of remote parties...for multi-media collaboration.”). Also, both Drell and Elliott support computer and PSTN

telephone conference participants. *See, e.g.*, Drell (Ex. 1008) at 3:33-38; Elliott (Ex. 1007) at 5:35-39, 6:1-11, 8:16-39, FIG. 1.

305. Further, a POSITA would have combined the teachings of Elliott with Drell to obtain numerous predictable and beneficial results.

306. Drell teaches a communications system capable of “transmission of audio and video data between and among near conference endpoint...and remote conference endpoints,” including “still images.” Drell (Ex. 1008) at 3:20-23, 6:53-

56. A POSITA would have recognized that Drell’s disclosure of transmitting “still images” is an example of collaboration data. A POSITA would have recognized that sharing collaboration data in Drell is desirable since it beneficially provides additional ways (in addition to audio and video) to convey information among participants in a multiparty conference.

307. Notably, Elliott evidences this understanding and teaches the sharing of various types of collaboration data, including T.120 data. In one example, Elliott teaches that during a conference, the transmitted stream includes “collaborative Web browsing, audio conferencing, video conferencing, and application sharing” and that “[o]ther types of collaboration...data can be performed,” including transmitting data per the “T.120 standard.” Elliott (Ex. 1007) at 1:40-44, 5:49-56, 6:53-54. To provide context, the T.120 standard was established by the International Telecommunications Union (ITU) and contains a

series of application protocols (i.e., T.122, T.123, T.124, and T.125), including standardizing the exchange of collaboration data. *See, e.g.*, A Primer on the T.120 Series Standard (Ex. 1016) T.120 Standard (Ex. 1017); '525 patent (Ex. 1001) at 3:10-12.

308. It would have been obvious to a POSITA in view of Elliott's teaching, to utilize collaboration data in Drell's system, including "Web browsing...and application sharing" (Elliott at 1:40-44) and "T.120 standard" collaboration data (Elliott at 5:49-56), because this type of collaboration data would allow for conveying a greater variety of information among the participants thereby improving the conferencing experience and enhancing the effectiveness of collaboration in a variety of contexts. *See, e.g.*, U.S. Patent No. 6,237,025 (Ex. 1018) at 1:28-47 ("In addition to spoken words, demonstrative gestures and behavioral cues, collaboration often involves the sharing of visual information—e.g., printed material such as articles, drawings, photographs, charts and graphs, as well as videotapes and computer-based animations, visualizations and other displays—in such a way that the participants can collectively and interactively examine, discuss, annotate and revise the information. This combination of spoken words, gestures, visual cues and interactive data sharing significantly enhances the effectiveness of collaboration in a variety of contexts."). Furthermore, sharing collaborative information such as written or graphic material during a video

conference had been known in the conferencing art for decades. *See, e.g.*, U.S. Patent No. 3,601,530 (Ex. 1014) at Abstract, 17:20-41 (claims 7-11), 18:30-49 (claims 21-24).

309. Accordingly, a POSITA would have recognized that collaboration data (e.g., web browsing data, application data, and data provided by the T.120 standard, as Elliott teaches) would be beneficial to Drell's participants. For example, when the user of one of Drell's conference endpoints connects to the conference via a computer, the user may share web browsing data when doing so helps the other participants better understand the context for the discussion, such as when discussing a web page at a particular URL. Drell (Ex. 1008) at 3:54-64 (CPU, memory, and network interface), FIG. 2; Elliott (Ex. 1007) at 5:58-61 ("During the course of a collaborative session, one party can push a URL to another party. In other words, one party can cause the browser of a remote party to connect to a certain web site."). As another example, when the user of one of Drell's conference endpoints is the presenter, the user may share an application, such as sharing a presentation slide, so that other participants can follow along with the discussion (similar to presenting slides at an in-person meeting); U.S. Patent No. 6,237,025 (Ex. 1018) at 9:23-26 ("'application sharing' (sharing both the control and display of running applications)"). In short, Elliott's collaboration data (e.g., web browsing, application sharing, and other collaboration data) beneficially

provides Drell's participants distinct ways to share information during a conference, thereby enhancing the effectiveness of the collaboration among the participants.

310. Such a combination would have simply been combining prior art elements (sharing collaboration data taught by Elliott in the conferencing system of Drell) according to known methods (disclosed by Elliott) to yield predictable results (allow for participants in Drell's system to share various types of data during a conference for collaboration purposes). Additionally, the combination of Drell and Elliott is merely the ordinary use of common techniques (transmitting collaboration web browsing data, application sharing data, and collaboration data per the T.120 standard as Elliott teaches) to improve the similar conferencing system of Drell in the same way by allowing for enhanced collaboration among participants.

311. Drell also teaches a communications system capable of "transmission of audio and video data between and among near conference endpoint[s]...and remote conference endpoints" across "network 106 [which] may comprise a packet switched network, such as the Internet." Drell (Ex. 1008) at 3:20-38, 6:53-56. It was known, however, that data transmitted over public IP networks is vulnerable to eavesdropping and other problems since the transmission is not secure. This is evidenced, for example, by the VPN Textbook which states that "more and more

business-to-business traffic takes place over the Internet” and that this type of communication is “riddled with security issues.” VPN Textbook (Ex. 1010) at 119. A solution for securing communications over public networks was also known. Specifically, it was known to protect communications using a VPN. *See, e.g., id.* at 115 (“A Virtual Private Network (VPN)...[provides] connectivity amongst multiple sites...with the same access or security policies as a private network.”); *see also id.* at 118, 129-130 (describing that the most common VPN implementation is hub-and-spoke). Thus, it would have been obvious to a POSITA that VPNs would beneficially provide security to the system of Drell, which describes conferencing communications for business over the Internet. Drell (Ex. 1008) at 1:20-25.

312. The obviousness of adding a VPN feature to Drell’s conferencing system is further demonstrated by the work of other artisans making similar feature combinations in the prior art timeframe. For example, U.S. Patent No. 6,980,526 addresses the “need for a videoconferencing system...for delivering secure...services over an IP network” by using a “VPN...over public networks...which is encrypted to make it secure from outside access and interference.” Ex. 1020 at 2:65-3:3, 5:52-55. As another example, U.S. Patent No. 6,782,413 describes a “conference bridge application” that transmits data “over public networks, using technologies such as...[a] Virtual Private Network (VPN)

or any other conventional communication method.” Ex. 1019 at 4:23-27, 6:36-41. That skilled artisans were combining conferencing systems and VPNs confirms that a POSITA would have found it obvious to combine Drell’s conferencing system with the VPN technology described in Elliott and the VPN Textbook (further discussed below).

313. Consistent with well-known approaches, Elliott teaches utilizing a “virtual private network (VPN)...which uses IP or IP-tunneling” to transmit “voice, fax, data, video” and “[o]ther types of collaboration...data.” Elliott (Ex. 1007) at 2:66-3:1, 4:29-34, 6:53-54. In view of Elliott’s VPN teachings, it would have been obvious to a POSITA to utilize VPN tunneling when communicating audio, video, and other data across Drell’s Internet network 106 because VPN tunneling would make the transmitted data secure from outside access and interference. *See, e.g.*, U.S. Patent No. 6,980,526 (Ex. 1020) at 5:52-55 (“VPN refers to a network that is carried over public networks, but which is encrypted to **make it secure** from outside access and interference.”). Additionally, a POSITA would have understood that VPNs are beneficial because they reduce the need to lease private data lines, which reduces or avoids the costs for implementing a private network by communicating securely over the Internet. *See, e.g.*, U.S. Patent No. 6,633,848 (Ex. 1026) at 44:10-12 (“[O]ffices [] create virtual private networks (VPNs) to save money on remote access and interoffice connectivity.”)

314. As such, one of ordinary skill in the art would have been motivated to apply the techniques of using VPN tunneling for conferencing, as Elliott teaches, to Drell's conferencing system because VPN tunneling provides secure communications while also reducing or avoiding costs associated with leased private data lines. Indeed, the combination of Drell and Elliott is merely the ordinary use of a common technique (using VPN tunneling for conferencing data) in the similar conferencing system of Drell to yield predictable results of (1) securing the transmitted data, and (2) reducing or avoiding costs associated with leasing private data lines. Each of these benefits separately and together would have motivated a POSITA.

315. I rely on the teachings of the references and do not suggest that physical incorporation of Elliott's elements into Drell's system is required in the proposed combination.

316. Lastly, I also note that the results would have been predictable and there would have been a reasonable expectation of success in the combination given that both Drell and Elliott describe similar conferencing systems. *See, e.g.,* Drell (Ex. 1008) at Abstract; Elliott (Ex. 1007) at Abstract. This is especially so since VPN tunneling was conventional. U.S. Patent No. 6,782,413 (Ex. 1019) at 4:23-27, 6:36-41; Elliott (Ex. 1007) at 5:49-56. Any modification needed to the

teachings of Drell, to accommodate the teachings of Elliott, would have been within the level of ordinary skill in the art.

317. For example, the modifications would have been accomplished with software programming or employing commercially available hardware, which would have been within the level of skill in the art and predictable. I personally, before the earliest claimed priority date of the '525 patent, configured, deployed and utilized VPN technology for carrying VoIP traffic using the NBX 100 system, which was also capable of performing conference calls.

**b) Reasons to Combine Drell and the VPN Textbook**

318. A POSITA would have been motivated to combine the teachings of Drell and the VPN Textbook for multiple reasons, including to produce the obvious, beneficial, and predictable result of providing a hub-and-spoke VPN topology when implementing VPN tunneling in Drell (per Elliott), and to provide a cost effective and less complex VPN implementation.

319. As discussed above, Drell teaches transmitting conferencing data between various conferencing endpoints over Internet network 106. And as further discussed above, Elliott teaches utilizing VPN tunneling for greater security when transmitting conferencing data over the Internet. A POSITA would have been familiar with conventional VPN tunneling implementations suggested by Elliott, and that such VPN implementations were well documented in readily available

textbooks. Accordingly, a POSITA considering the combination of Drell and Elliott, which employs VPN tunneling, would have found it obvious to consider other references, such as the VPN Textbook, that provide more details and background information on VPN techniques. *See, e.g.*, Drell (Ex. 1008) at 3:20-38, 6:53-56; Elliott (Ex. 1007) at 2:66-3:1, 4:29-34, 6:53-54.

320. Notably, the VPN Textbook provides a useful description of various VPN topologies. A POSITA looking to implement Drell's system with VPN tunneling would have needed to choose a topology, and the VPN Textbook explains that the most common topology is the hub-and-spoke topology. VPN Textbook (Ex. 1010) at 129 ("The most commonly encountered topology is a hub-and-spoke topology, where a number of remote offices (spokes) are connected to a central site (hub).").

321. The hub-and-spoke VPN topology would have been an obvious choice for the POSITA since it is the most commonly used. This would be simply combining prior art elements (the VPN topology teachings of the VPN Textbook and the system of Drell) according to known methods (as taught by the VPN Textbook) to yield predictable results (securing the traffic exchanged in Drell's system during conferencing). In the combination, Drell's system would utilize corresponding hardware and software (e.g., routers configured in a hub-and-spoke topology running VPN software) that would perform the routing and VPN

tunneling functions described in the VPN textbook. Such products were widely commercially available in the prior art time period. I personally researched and purchased VPN hardware routers for a work project during the 1998-1999 time frame.

322. Using a hub-and-spoke VPN topology, as taught by the VPN Textbook in the system of Drell (as modified by Elliott) is simply the use of a known technique (utilizing hub-and-spoke VPN topology) in Drell's system to yield predictable results (provide VPN tunnels to Drell's system for improved security). A POSITA would have been readily able to predict the result of such a combination, namely, that Drell's system would utilize a hub-and-spoke VPN topology to allow the conferencing devices to communicate with each other over secure VPN tunnels. Because hub-and-spoke VPN topology was conventional and the most commonly used topology, making such a combination would have been within the POSITA's skill level.

323. Additionally, a POSITA would have been motivated to combine the teachings of Drell (as modified by Elliott) with the teachings of the VPN Textbook because it would produce predictable and beneficial results. The VPN Textbook explains that the hub-and-spoke VPN topology serves to constrain cost and complexity. VPN Textbook (Ex. 1010) at 129 ("[T]here are many examples where the customer could benefit from a different topology but has nonetheless chosen

the hub-and-spoke topology for **cost or complexity reasons**.”). Accordingly, a POSITA would have been motivated to utilize a conventional hub-and-spoke VPN topology, as the VPN Textbook teaches, when implementing VPN tunneling in Drell’s conferencing system, per Elliott’s teaching, because it would beneficially constrain cost and limit complexity.

324. Moreover, the result would have been predictable and there would have been a reasonable expectation of success since VPN technology was conventional and known to be applied in the conferencing context. Indeed, others had previously combined conferencing and VPN technologies. For example, U.S. Patent No. 6,782,413 describes a “conference bridge application” that transmits data “over public networks, using technologies such as...Virtual Private Network (VPN) or any other conventional communication method.” U.S. Patent No. 6,782,413 (Ex. 1019) at 4:23-27, 6:36-41. Further, any modification needed to the teachings of Drell, such as adding software programming and/or hardware, to accommodate the teachings of the VPN Textbook would have been within the level of ordinary skill in the art, as demonstrated by the examples of other artisans combining conferencing and VPN technologies. *See, e.g.*, U.S. Patent Publication No. 2004/0261116 (Ex. 1023) at ¶ [1412] (“**VPN connectivity...hub-and-spoke** or mixed topologies.”), ¶ [1452] (“Availability time for each service provided to users within the VPN. These services include...**Video-conferencing** etc.”). I

personally, before the earliest claimed priority date of the '525 patent, configured, deployed and utilized VPN technology for carrying VoIP traffic using the NBX 100 system, which was also capable of performing conference calls.

325. Lastly, I rely on the teachings of the references and do not suggest that physical incorporation of elements from the VPN Textbook into Drell's system (as modified by Elliott) is required in the proposed combination.

### 3. Claim 5

***[5.P] The system of claim 1 further comprising:***

326. See analysis for [1.P]-[1.4].

***[5.1] wherein the PSTN client audio data stream and the moderator audio-video data stream are transported across a virtual private network tunnel***

327. The prior art renders obvious that the PSTN client audio data stream and the moderator audio-video data stream are transported across a virtual private network tunnel.

328. **First**, as discussed at [1.1], Drell in combination with Knappe teaches that the near conference endpoint 100 receives an audio data stream from the PSTN conference endpoint through a gateway and an audio-video data stream from the moderator remote conference endpoint 104. See e.g. *supra* ¶¶ 183-186.

329. **Second**, Elliott teaches a conferencing system that transmits audio and video data over an Internet network:

Multi-media collaboration refers to the use of more than one media stream (e.g.: **voice**, fax, data, **video**, etc.) used in collaboration with more than one party.

Elliott (Ex. 1007) at 2:66-3:13.

**As alluded to above, communications between the parties can be made over a variety of networks 18.** A network 18 is at least two (and preferably more) computers interconnected together so that communication between them is possible. In the illustrated embodiment, the public Internet is shown as an exemplary **Internet Protocol-based (IP-based) network 18 for multi-media collaboration.**

*Id.* at 4:6-12.

330. Further, Elliott teaches that the data is transmitted over the Internet using VPN tunneling:

For example, network 18 can be either a public or a carrier supported virtual private network (VPN)...which uses IP or IP-tunneling.

*Id.* at 4:31-34.

331. Based on the above disclosure, a POSITA would have understood that Elliott utilizes VPN tunneling for communicating conferencing voice and video data over a public network 18, such as the Internet. In view of Elliott's teachings, it would have been obvious to a POSITA to utilize VPN tunneling when communicating audio and video data to and from the PSTN conference endpoint's gateway and the moderator remote conference endpoint 104, because it would provide Drell with secure communications as the communications traverse Internet network 106. *See also* Reasons to Combine Drell and Elliott, *supra* ¶¶ 303-317.

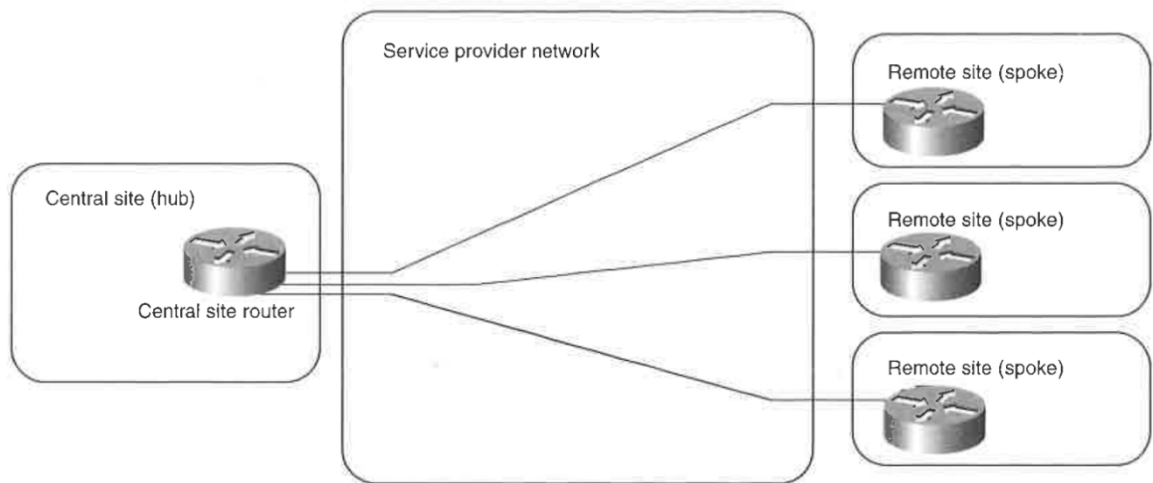
332. **Third**, the VPN Textbook teaches that the most common VPN topology is “hub-and-spoke”:

The **most commonly** encountered topology is a **hub-and-spoke** topology, where a number of remote offices (spokes) are connected to a central site (hub), similar to the setup in Figure 7-10.

VPN Textbook (Ex. 1010) at 129.

333. As shown in Figure 7-10, each remote site router (spoke) is connected to a central router (hub) through a VPN tunnel:

**Figure 7-10** *Hub-and-spoke Topology*

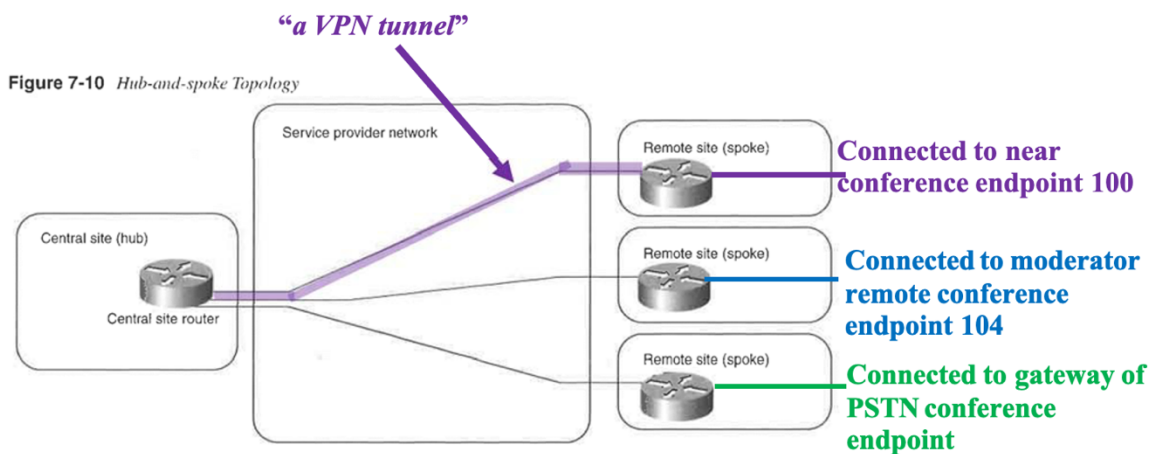


*Id.* at 130. A POSITA would have understood that each remote site router (spoke) would have one or more network devices connected to it.

334. When implementing VPN tunneling in Drell, per Elliott, it would have been obvious to a POSITA to connect each of Drell’s endpoints (which are located at different locations) to a respective remote router (spoke) that is connected to the central router (hub), as the VPN Textbook teaches, because this common topology

provides a cost effective and less complex implementation. *See, e.g., id.* at 129 (“This is based purely on business needs due to higher costs or increased routing complexity associated with other topologies that use these types of technologies. In other words, there are many examples where the customer could benefit from a different topology but has nonetheless chosen the hub-and-spoke topology for **cost or complexity reasons.**”); *see also* Reasons to Combine Drell and the VPN Textbook, *supra* ¶¶ 318-325.

335. Drell’s conference endpoints are located at different locations. Thus, it would have been obvious to a POSITA that when implementing the commonly used hub-and-spoke VPN topology, per the VPN Textbook, Drell’s near conference endpoint 100, the moderator remote conference endpoint 104, and the PSTN conference endpoint’s gateway would be connected to different remote site (spoke) routers, as illustrated below in annotated Figure 7-10:



*Id.* at 130 (annotated).

336. As illustrated above, the remote site routers (spokes) of the moderator remote conference endpoint 104 and the PSTN conference endpoint's gateway are connected to remote site router (spoke) of the near conference endpoint 100, through a VPN tunnel via the central router (hub). When the near conference endpoint 100 transmits data to or receives data from the moderator remote conference endpoint 104 and the PSTN conference endpoint's gateway, the data would be received/sent through the same VPN tunnel, which corresponds to "*a VPN tunnel*," as claimed.

337. Thus, Drell in combination with Knappe, Elliott, and the VPN Textbook teaches that the PSTN conference endpoint audio data stream from the gateway and the moderator remote conference endpoint 104 audio-video data stream are transported across a VPN tunnel, which renders obvious discloses that "*the PSTN client audio data stream and the moderator audio-video data stream are transported across a virtual private network tunnel*," as claimed.

#### **4. Claim 6**

***[6.P] The system of claim 5 further comprising:***

338. See analysis for [5.P]-[5.1].

***[6.1] a VoIP encoder that encodes the second mixed data stream and encodes the second mixed voice data before transmitting the second mixed data stream to the PSTN client***

339. The prior art, in multiple ways, renders obvious a VoIP encoder that encodes the second mixed data stream and encodes the second mixed voice data before transmitting the second mixed data stream to the PSTN client.

340. As a threshold matter, I note that neither of claims 1 and 5, from which claim 6 depends, recites the phrase a “*second mixed voice data*.” For analysis purposes only, to the extent determined that the term “*second mixed voice data*” recited in claim 6 corresponds to the “*mixed audio data stream*” recited in [1.4], I demonstrate below that this claim element would have been obvious.

341. **First**, as discussed at [1.2.1]-[1.3], Drell in combination with Knappe teaches mixing the audio-video data stream from the moderator remote conference endpoint 104 with the audio-video data stream from the remote conference endpoint 102 into a second mixed data stream. *See supra e.g.* ¶¶ 230-242. Further, as discussed at [1.4], Drell in combination with Knappe and Voois teaches encoding and transmitting the second mixed data stream, including the mixed audio data stream, to the PSTN conference endpoint. *See supra e.g.* ¶¶ 243-252.

342. **Second**, Drell teaches that the near conference endpoint 100 uses an audio codec 408 and video codec 406 (including standard or proprietary codecs) to encode the mixed speech/voice and video data stream before it is transmitted over the Internet network 106:

In the transmit mode, video codec 406 encodes the video data stream output by video switching/continuous presence module 306

(representative, for example, of a “continuous presence” image) using a standard or proprietary video compression algorithm (e.g., H.261) and delivers the encoded video data to communication process 404. **Audio codec 408 encodes the audio data stream** output by audio mixing module 308 (representative, for example, of the blended speech of conference participants located at near conference endpoint 100 and remote conference endpoints 102 and 104) **using a standard or proprietary audio compression algorithm** (e.g., G711) and delivers the encoded audio data to communication process 404.

Drell (Ex. 1008) at 6:33-45; *see also id.* at 5:8-25, FIGS. 1-4.

**[N]etwork 106** may comprise a **packet** switched network, such as the **Internet**

*Id.* at 6:33-34.

343. It would have been obvious to a POSITA that data in Drell’s Internet network 106 are transmitted in IP packets, because the Internet is a packet switched network that operates on packets using IP protocol. *See, e.g.*, Knappe (Ex. 1006) at 3:50-55 (“In the Internet embodiment, the network 14 transmits information in Internet Protocol (IP) packets.”).

344. Knappe supplements Drell and teaches that the conference endpoints encapsulate audio and video data into Real-time Transport Protocol (RTP) VoIP packets for transmission over the Internet:

**[N]etwork 14 is the Internet...In the Internet embodiment, the network 14 transmits information in Internet Protocol (IP) packets. Telephony voice information is transmitted in the Voice over IP (VoIP) format. Real-time IP packets such as VoIP packets are encapsulated in real-time transport protocol (RTP) packets for transmission over the network 14.**

*Id.* at 3:50-58.

A codec (coder/decoder) converts **audio, video** or other suitable signals generated by users from analog signals into digital form. **The digital encoded data is encapsulated into IP or other suitable packets** for transmission over the network 14.

*Id.* at 4:9-12.

345. Once Drell's codec encodes the mixed audio data (Drell (Ex. 1008) at 6:33-45) and then generates the second mixed data stream, it would have been obvious to a POSITA to encapsulate the second mixed data stream in RTP packets for transmission over the Internet (as Knappe teaches), because RTP was specifically created for applications such as Drell's real-time conferencing application. *See id.* at 3:20-38 ("Network 106 may be of any type suitable for the transmission of audio and video data between and among near conference endpoint 100 and remote conference endpoints... [N]etwork 106 may comprise a packet switched network, such as the Internet...to connect conference endpoints.."); *see also* Caputo (Ex. 1021) at 22 ("**The Real-Time Protocol (RTP)...was created to transport real-time traffic, such as video and audio**, over the Internet."); *see also* Reasons to Combine Drell and Knappe, *supra* ¶¶ 148-168.

346. Thus, Drell's audio codec and communication process that encodes data for transmission over the Internet using RTP packets (per Knappe) teaches a "*VoIP encoder*" as claimed.

347. **Additionally**, as discussed at [1.2.2]-[1.4], Drell's communication process 404 combines mixed audio and video data into a "single audio/video data

stream” for transmission over the Internet to the PSTN conference endpoint. Drell (Ex. 1008) at 6:46-48, 3:20-37. Drell further explains that the “communication process...compris[es] an **H.323 (packet-based) process.**” *Id.* at 2:26-28. It would have been obvious to a POSITA that Drell’s H.323 process includes a “*VoIP encoder*” for encoding the audio data and the mixed data transmitted to the PSTN conference endpoint (as claimed) because H.323 was a well-known standard for encoding and decoding RTP VoIP communication. *See, e.g.* U.S. Patent No. 6,553,515 (Ex. 1025) at 3:57-65 (“Once digitized, the encoded voice data is then wrapped within IP packets, using a **Real-Time Transport Protocol (RTP)**, which runs over UDP...based on the **H.323.**”), 4:35-56 (“**H.323** is adapted to support many aspects of multimedia...[and] generally is regarded as the **standard for VoIP**...This standard also specifies lowest-common denominator compression protocols, such as G.711...G.723 and G.729.”); Caputo (Ex. 1021) at 22 (“**The Real-Time Protocol (RTP)...**was created to transport real-time traffic, such as **video and audio**, over the Internet.”). Since Drell comprises an H.323 process, it would have been obvious to a POSITA that Drell’s system utilizes a VoIP encoder to encode the mixed audio data and also encode the second mixed data (which includes the mixed audio data) in RTP packets, which renders obvious “a *VoIP encoder*,” as claimed.

348. The prior art teachings provide the same encoder disclosed in the '525 Patent, with the prior art H.323 protocol codec that uses the RTP for transport. *See, e.g., '525 patent (Ex. 1001) at 2:58-67* (“The **H.323** protocol stack consists of...audio and video CoDecs are responsible for compression and decompression of the audio and video sources. The real-time network protocol component 215 is connected to the audio video CoDecs and is also responsible for preparation of the media data for transport according to the **RTP (real-time protocol)** recommendations.”).

349. Thus, Drell in combination with Knappe teaches that the MP conferencing application 234 of the near conference endpoint 100 includes a VoIP encoder that encodes the mixed audio data and encodes the second mixed data stream in VoIP packets that are encapsulated in RTP packets, before transmitting the second mixed data stream to the PSTN conference endpoint, which renders obvious “*a VoIP encoder that encodes the second mixed data stream and encodes the second mixed voice data before transmitting the second mixed data stream to the PSTN client,*” as claimed.

## 5. Claim 7

***[7.P] The system of claim 1 further comprising:***

350. *See analysis for [1.P]-[1.4].*

***[7.1] wherein the moderator audio-video data stream and the remote client audio-video data stream are transported across a virtual private network tunnel***

351. The prior art renders obvious that the moderator audio-video data stream and the remote client audio-video data stream are transported across a virtual private network tunnel.

352. **First**, as discussed at [1.1], Drell in combination with Knappe teaches that the near conference endpoint 100 receives an audio-video data stream from the moderator remote conference endpoint 104 and as discussed at and [1.2.2]-[1.3], the near conference endpoint 100 also receives an audio-video data stream from the remote conference endpoint 102 over an Internet network.

353. **Second**, Elliott teaches a conferencing system that transmits audio and video data over an Internet network:

[Multi-media collaboration refers to the use of more than one media **stream** (e.g.: **voice**, fax, data, **video**, etc.) used in collaboration with more than one party.

Elliott (Ex. 1007) at 2:66-3:1.

**As alluded to above, communications between the parties can be made over a variety of networks 18.** A network 18 is at least two (and preferably more) computers interconnected together so that communication between them is possible. In the illustrated embodiment, the public Internet is shown as an exemplary **Internet Protocol-based (IP-based) network 18 for multi-media collaboration.**

*Id.* at 4:6-12.

354. Further, Elliott teaches that the data is transmitted over Internet 18 using VPN tunneling:

**For example, network 18 can be either a public or a carrier supported virtual private network (VPN)...which uses IP or IP-tunneling.**

*Id.* at 4:31-34.

355. Based on the above disclosure, a POSITA would have understood that Elliott utilizes VPN tunneling for communicating conferencing voice and video data over a public network 18, such as the Internet. In view of Elliott's teachings, it would have been obvious to a POSITA to utilize VPN tunneling when communicating audio and video data to and from the remote conference endpoint 102 and the moderator remote conference endpoint 104, because it would provide Drell with secure communications as the communications traverse Internet network 106. *See also* Reasons to Combine Drell and Elliott, *supra* ¶¶ 303-317.

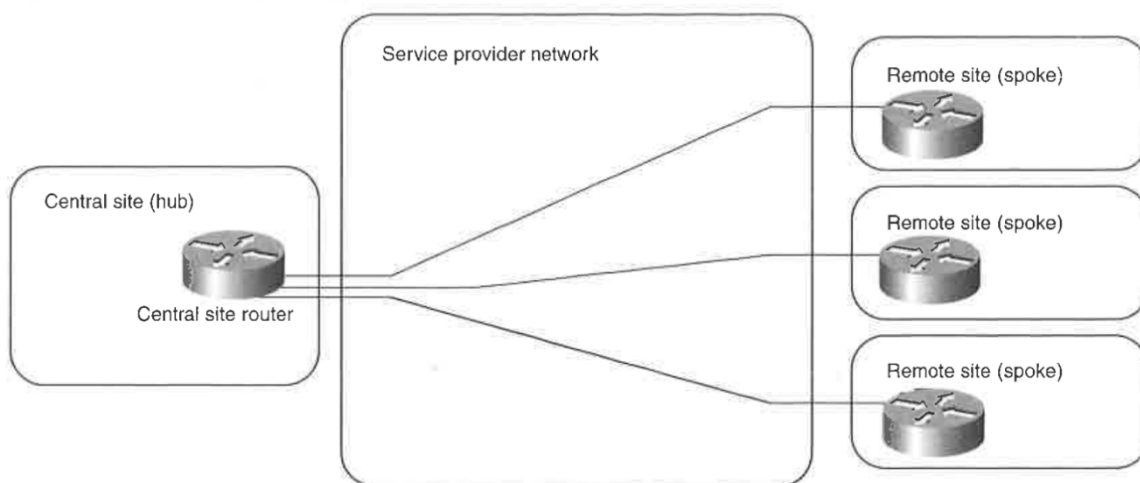
356. **Third**, the VPN Textbook teaches that the most common VPN topology is "hub-and-spoke":

The **most commonly** encountered topology is a **hub-and-spoke** topology, where a number of remote offices (spokes) are connected to a central site (hub), similar to the setup in Figure 7-10.

VPN Textbook (Ex. 1010) at 129.

357. As shown in Figure 7-10, each remote site router (spoke) is connected to a central router (hub) through a VPN tunnel:

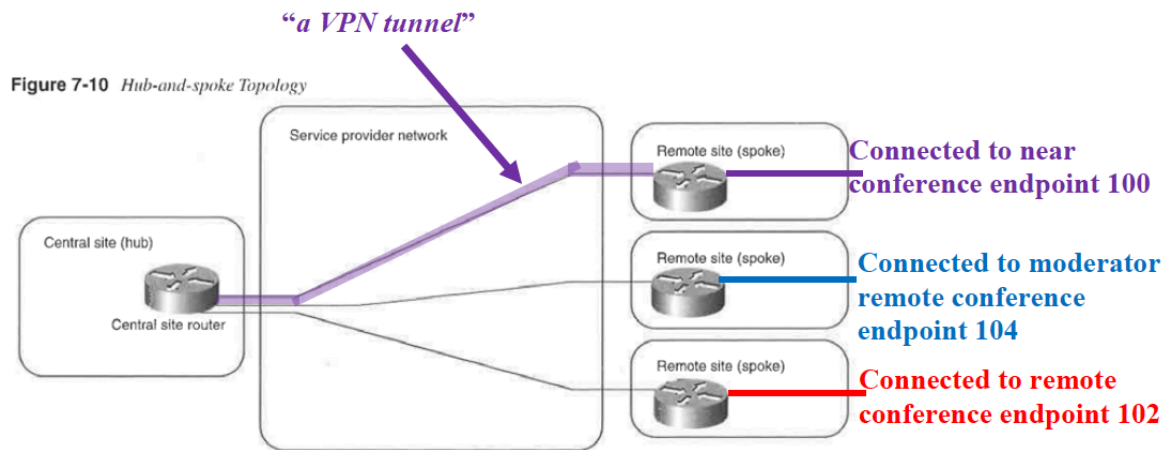
**Figure 7-10** *Hub-and-spoke Topology*



358. A POSITA would have understood that each remote site router (spoke) would have one or more network devices connected to it.

359. When implementing VPN tunneling in Drell, per Elliott, it would have been obvious to a POSITA to connect each of Drell's endpoints (which are located at different locations) to a respective remote router (spoke) that is connected to the central router (hub), as the VPN Textbook teaches, because this common topology provides a cost effective and less complex implementation. *See, e.g.,* VPN Textbook at 129 ("This is based purely on business needs due to higher costs or increased routing complexity associated with other topologies that use these types of technologies. In other words, there are many examples where the customer could benefit from a different topology but has nonetheless chosen the hub-and-spoke topology for **cost or complexity reasons.**"); *see also* Reasons to Combine Drell and the VPN Textbook, *supra* ¶¶ 318-325.

360. Drell's conference endpoints are located at different locations. Thus, it would have been obvious to a POSITA that when implementing the commonly used hub-and-spoke VPN topology, per the VPN Textbook, Drell's near conference endpoint 100, the moderator remote conference endpoint 104, and the PSTN conference endpoint would be connected to different remote site (spoke) routers, as illustrated below in annotated Figure 7-10:



*Id.* at 130 (annotated).

361. As illustrated above, the remote site routers (spokes) of the remote conference endpoint 102 and the moderator remote conference endpoint 104 are connected to remote site router (spoke) of the near conference endpoint 100, through a VPN tunnel via the central router (hub). When the near conference endpoint 100 transmits data to or receives data from the remote conference endpoint 102 and the moderator remote conference endpoint 104, the data would

be received/sent through the same VPN tunnel, which corresponds to “*a VPN tunnel,*” as claimed.

362. Thus, Drell in combination with Knappe, Elliott, and the VPN Textbook teaches that the moderator remote conference endpoint 104 audio-video data stream and the remote conference endpoint 102 audio-video data stream are transported across a VPN tunnel, which renders obvious that “*the moderator audio-video data stream and the remote client audio-video data stream are transported across a virtual private network tunnel,*” as claimed.

## **6. Claim 8**

***[8.P] The system of claim 7 further comprising:***

363. See analysis for [7.P]-[7.1].

***[8.1] an encoder that receives the first mixed data stream and encodes the first mixed voice data before transmitting the first mixed data stream to the remote client.***

364. The prior art renders obvious an encoder that receives the first mixed data stream and encodes the first mixed voice data before transmitting the first mixed data stream to the remote client.

365. As a threshold matter, I note that neither of claims 1 and 7, from which claim 8 depends, recites the phrase a “*first mixed voice data.*” For analysis purposes only, to the extent determined that the term “*first mixed voice data*” recited in claim 8 corresponds to a mixed audio portion of the “*first mixed data stream*” recited in [1.1], I demonstrate below that this claim element would have been obvious.

366. **First**, as discussed at [1.1], Drell in combination with Knappe teaches mixing the audio data stream from the PSTN conference endpoint's gateway with the audio- video data stream from the moderator remote conference endpoint 104 into a first mixed data stream. *See e.g. supra* ¶¶ 181-212. Further, as discussed at [1.2.1], Drell in combination with Knappe teaches transmitting the first mixed data stream, including the mixed audio data stream, to the remote conference endpoint 102. *See e.g. supra* ¶¶ 214-220.

367. **Second**, Drell teaches that the near conference endpoint 100 uses an audio codec 408 and video codec 406 (including standard or proprietary codecs) to encode the mixed speech/voice and video data stream before it is transmitted over the Internet network 106:

In the transmit mode, video codec 406 encodes the video data stream output by video switching/continuous presence module 306 (representative, for example, of a “continuous presence” image) using a standard or proprietary video compression algorithm (e.g., H.261) and delivers the encoded video data to communication process 404. Audio codec 408 encodes the audio data stream output by audio mixing module 308 (representative, for example, of the blended speech of conference participants located at near conference endpoint 100 and remote conference endpoints 102 and 104) **using a standard or proprietary audio compression algorithm** (e.g., G711) and delivers the encoded audio data to communication process 404.

Drell (Ex. 1008) at 6:33-45; *see also id.* at 5:8-25, FIGS. 1-4.

**[N]etwork 106** may comprise a **packet** switched network, such as the **Internet**

*Id.* at 6:33-34.

368. It would have been obvious to a POSITA that data in Drell's Internet network 106 are transmitted in IP packets, because the Internet is a packet switched network that operates on packets using IP protocol. *See, e.g.*, Knappe (Ex. 1006) at 3:50-55 ("In the Internet embodiment, the network 14 transmits information in Internet Protocol (IP) packets.").

369. Knappe supplements Drell and teaches that the conference endpoints encapsulate audio and video data into Real-time Transport Protocol (RTP) VoIP packets for transmission over the Internet:

[N]etwork 14 is the Internet...In the Internet embodiment, the network 14 transmits information in Internet Protocol (IP) packets. Telephony voice information is transmitted in the Voice over IP (VoIP) format. **Real-time IP packets such as VoIP packets are encapsulated in real-time transport protocol (RTP) packets for transmission over the network 14.**

*Id.* at 3:50-58.

A codec (coder/decoder) converts **audio, video** or other suitable signals generated by users from analog signals into digital form. **The digital encoded data is encapsulated into IP or other suitable packets** for transmission over the network 14.

*Id.* at 4:9-12.

370. Once Drell's codec encodes the mixed audio data (Drell (Ex. 1008) at 6:33-45) and then generates the first mixed data stream, it would have been obvious to a POSITA to encapsulate the first mixed data stream in RTP packets for transmission over the Internet (as Knappe teaches), because RTP was specifically

created for applications such as Drell’s real-time conferencing application. *See id.* at 3:20-38 (“Network 106 may be of any type suitable for the transmission of audio and video data between and among near conference endpoint 100 and remote conference endpoints... [N]etwork 106 may comprise a packet switched network, such as the Internet...to connect conference endpoints..”); *see* Caputo (Ex. 1021) at 22 (“**The Real-Time Protocol (RTP)...was created to transport real-time traffic, such as video and audio,** over the Internet.”); *see also* Reasons to Combine Drell and Knappe, *supra* ¶¶ 148-168.

371. Thus, Drell’s audio codec and communication process that encodes data for transmission over the Internet using RTP packets (per Knappe) teaches a an “*encoder*” as claimed.

372. **Additionally**, as discussed at [1.1]-[1.2.1], Drell’s communication process 404 combines mixed audio and video data into a “single audio/video data stream” for transmission over the Internet to the remote conference endpoint 102. Drell (Ex. 1008) at 6:46-48, 3:20-37. Drell further explains that the “communication process...compris[es] an **H.323 (packet-based) process.**” *Id.* at 2:26-28. It would have been obvious to a POSITA that Drell’s H.323 process includes an “*encoder*” for encoding the audio data and the mixed data transmitted to the remote conference endpoint (as claimed) because H.323 was a well-known standard for encoding and decoding RTP VoIP communication. *See, e.g.* U.S.

Patent No. 6,553,515 (Ex. 1025) at 3:57-65 (“Once digitized, the encoded voice data is then wrapped within IP packets, using a **Real-Time Transport Protocol (RTP)**, which runs over UDP...based on the **H.323**.”), 4:35-56 (“**H.323** is adapted to support many aspects of multimedia...[and] generally is regarded as the **standard for VoIP**...This standard also specifies lowest-common denominator compression protocols, such as G.711...G.723 and G.729.”); Caputo (Ex. 1021) at 22 (“**The Real-Time Protocol (RTP)...was created to transport real-time traffic, such as video and audio** over the Internet.”). Thus, Drell and Knappe teach a communication process (“*encoder*”) that receives and encapsulates a mixed audio and video data stream to be transmitted to remote conference endpoint 102, which renders obvious “*an encoder*,” as claimed.

373. The prior art teachings provide the same encoder disclosed in the ’525 patent, with the prior art H.323 protocol codec that uses the RTP for transport. *See, e.g.,* ’525 patent (Ex. 1001) at 2:58-67 (“The **H.323** protocol stack consists of...audio and video CoDecs are responsible for compression and decompression of the audio and video sources. The real-time network protocol component 215 is connected to the audio video CoDecs and is also responsible for preparation of the media data for transport according to the **RTP (real-time protocol)** recommendations.”).

374. Thus, Drell in combination with Knappe teaches that the MP conferencing application 234 of the near conference endpoint 100 includes an encoder that receives the first mixed data stream and encodes the first mixed audio data stream in RTP packets before transmitting the first mixed data stream to the remote conference endpoint 102, which renders obvious “*an encoder that receives the first mixed data stream and encodes the first mixed voice data before transmitting the first mixed data stream to the remote client,*” as claimed.

375. **Additionally**, as discussed in [7.1], the prior art renders obvious transmitting a mixed audio and video data stream to remote conference endpoint 102 via a VPN tunnel. It would have been obvious to a POSITA that the VPN transmission process would include receiving and encrypting this data stream (including the audio (voice) portion thereof) because encryption is generally used to provide security in a VPN. This encryption step would be performed, for example, by the remote site router described in the VPN Textbook. VPN Textbook at 119 (describing “encryption gear”), Fig. 7-3.

376. Thus, the prior art teaches a VPN router (“*encoder*”) that receives and encrypts a mixed audio and video data stream to be transmitted to remote conference endpoint 102, which also renders obvious this element.

**C. Claims 1-4 Are Obvious Over Drell in View of Knappe**

377. To the extent that Patent Owner argues that the claimed “second mixed data stream” does not require both audio and video data, then Drell and Knappe render obvious claims 1-4 for substantially the same reasons discussed above in Section IX.A. Specifically, it would have been obvious to a POSITA that the PSTN conference endpoint suggested by Drell would be, for example, a “standard analog telephone[]” as taught by Knappe. Knappe (Ex. 1006) at 3:41-45. Since a standard telephone supports only audio communication, it would have been obvious to a POSITA for Drell’s communication process 404 instantiated for a PSTN conference endpoint to transmit only the audio portion of the conference. Drell’s audio mixing module 308 and communication process 404 instantiated for the PSTN conference client collectively teach a “second mixer,” and their mixed audio output is a “second mixed data stream.” As discussed above, Knappe’s gateway converts from “VoIP format” to “standard telephony signals.” *Id.* at 4:18-25. Thus, the gateway has an output (“second transport output”) that provides standard telephony signals (“mixed audio data stream, corresponding to the second mixed data stream”) to the PSTN conference endpoint.

**D. Claims 5-8 Are Obvious Over Drell in Combination with Knappe, Elliott, and the VPN Textbook**

378. To the extent that Patent Owner argues that the claimed “*second mixed data stream*” does not require both audio and video data, then Drell,

Knappe, Elliott and the VPN Textbook (without Voois) render obvious claims 5-8 for the same reasons discussed above.

**E. Claims 9-20 Are Obvious Over Drell in Combination With Knappe and Elliott**

**1. Claim 9**

***[9.P] A non-transitory computer-readable medium including instructions for a multi-participant conference process to be executed by a local moderator computer:***

379. The prior art teaches a non-transitory computer-readable medium including instructions for a multi-participant conference process.

380. Specifically, Drell teaches a memory 222 comprising any of various non-volatile memories, such as read-only memory (ROM), programmable ROM (PROM), or non-volatile storage media, for storing applications (and program instructions) within a near conference endpoint 100:

[C]omponents of near conference endpoint 100...include a central processing unit (CPU) 220 for **executing program instructions, a memory 222 for storing applications, data, and other information**, and a network interface 224 for connecting near conference endpoint 100 to network 106. Memory 222 may variously comprise one or a combination of volatile or **non-volatile memories**, such as random access memory (RAM), read-only memory (ROM), programmable ROM (PROM), or non-volatile storage media such as hard disks or CD-ROMs.

Drell (Ex. 1008) at 3:39-63, FIG. 2.

381. In view of the above disclosure, it would have been obvious to a POSITA Drell's non-volatile memory stores applications and instructions executable by a central processing unit (CPU) of Drell's near conference endpoint

100, which renders obvious a “*non-transitory computer-readable medium including instructions*” as claimed

382. Drell further teaches that the executable instructions stored within memory 222 include a call manager application 232 and multi-point (MP) conferencing application 234 configured to process and communicate audio and video data between the near conference endpoint 100 and remote conference endpoints 102 and 104:

Depending on bandwidth requirements, data communicated between near conference endpoint 100 and a remote conference endpoint may be carried on a single ISDN line, or may be distributed (for higher bandwidth connections) among a plurality of ISDN lines.

**Stored within memory 222 are...a call manager application 232, and the MP conferencing application 234.** Operating system 230 controls the allocation and usage of hardware resources, such as CPU 220 and 15 memory 222. **Call manager application 232 controls the establishment and termination of connections between near conferencing endpoint 100 and remote conference endpoints 102 and 104,** and may also furnish information characterizing the nature of individual connections to MP conferencing application 234.

**As will be described in further detail below, MP conferencing application 234 is configured to instantiate a processing train for each remote conference endpoint 102 and 104 to which near conference endpoint 100 is connected. The processing trains process audio and video data streams received from remote conferencing endpoints 102 and 104.** The processed audio and video data streams are combined with each other and with locally generated audio and video streams, and the combined audio and video streams are thereafter distributed to remote conferencing endpoints 102 and 104.

Drell (Ex. 1008) at 4:6-31.

383. In view of the above, a POSITA would have understood that both the call manager application 232 and the multi-point (MP) conferencing application 234 facilitate a multi-participant conferencing process between the participants at the near conference endpoint 100 and remote conference participants 102 and 104 (in other words, a multi-participant conference process). *See also id.* at 1:26-29 (“At times, it may be desirable to conduct multi-point conferences, wherein three or more parties (each party consisting of an individual or group located at a particular conference endpoint) participate in the conference.”). Other participants include PSTN conference endpoint. *See, e.g., id.* at 3:33-37 (“...PSTN...conference endpoints...”).

384. Thus, it would have been obvious to a POSITA that memory 222 (non- transitory computer-readable medium) of the near conference endpoint 100 includes instructions executable by the CPU of the near conference endpoint 100 to implement the multi-participant conference process in the near conference endpoint 100. *See id.* at 6:63-66 (“The present invention advantageously utilizes software-based processing of video and audio data streams to implement a multi-point conferencing capability in a conference endpoint.”). It would have been obvious to a POSITA that the stored applications comprise instructions because computer applications and software generally comprise instructions. Accordingly,

the instructions stored in the non-volatile memory include a call manager application 232 and multi-point (MP) conferencing application 234 which separately, and together, teach “*instructions for a multi-participant conference process*,” as claimed.

385. Thus, Drell teaches that the near conference endpoint 100 includes a non-volatile memory storing a call manager application 232 and a multi-point (MP) conferencing application 234, which renders obvious “[a] *non-transitory computer- readable medium including instructions for a multi-participant conference process*.”

386. The prior art further renders obvious the limitation that the multi-participant conference process is to be executed by a local moderator computer.

387. **First**, as discussed above, Drell teaches that the near conference endpoint 100 includes a non-volatile memory storing a call manager application 232 and a multi-point (MP) conferencing application 234. It would have been obvious to a POSITA that Drell’s applications stored in memory 222 comprise instructions because computer applications generally comprise instructions. Further, Drell teaches that the instructions are executable by central processing unit (“CPU”) of the near conference endpoint 100. Drell (Ex. 1008) at 3:39-58 (“**[C]omponents of near conference endpoint 100...include a central processing unit (CPU) 220 for executing program instructions.**”); *see also id.* at

8:8-18. Accordingly, it would have been obvious to a POSITA that the instructions in Drell's applications would be executed by the central processing unit (CPU). Thus, Drell teaches that the near conference endpoint 100 includes a CPU (a computer) that executes the instructions, which teaches that the instructions are "*to be executed by a local...computer,*" as claimed.

388. Furthermore, Knappe teaches that conference endpoints may be "personal or other suitable computers or computing devices." Knappe (Ex. 1006) at 3:63-64. In view of Knappe's teaching, a POSITA would understand that Drell's near conference endpoint 100 is a "*computer.*"

389. **Second**, Knappe teaches designating at least one of the conference participants as a moderator:

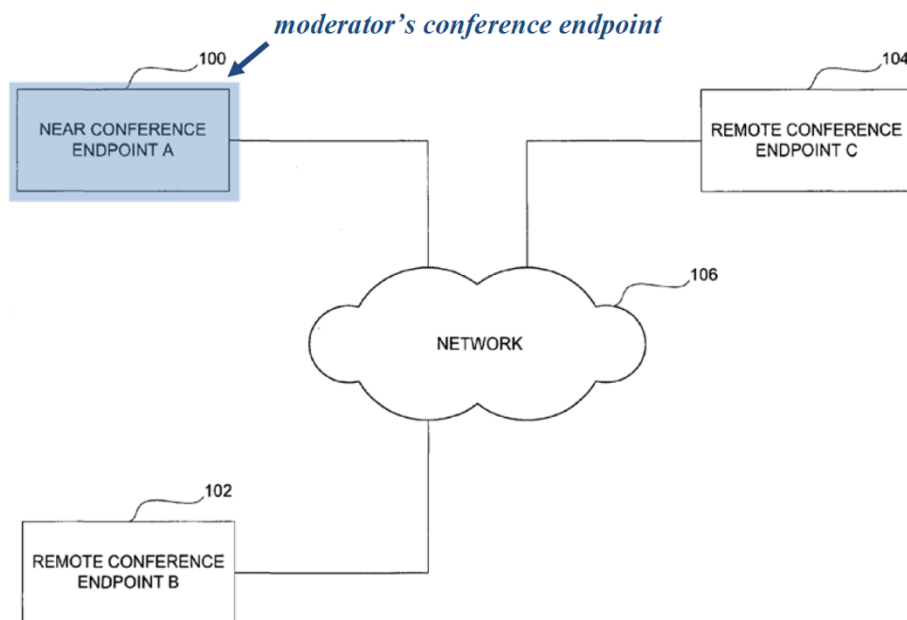
**[A]t least one participant may be given the priority of moderator for a particular conference call.** Typically, the moderator will receive the highest priority in a conference call. Therefore, in accordance with one embodiment, the priorities simply include moderator for one or more parties, and participant for the remaining parties.

*Id.* at 8:52-57.

390. Since Knappe teaches that at least one participant may be given the priority of a moderator, it would have been obvious to a POSITA to designate the user of Drell's near conference endpoint 100 as the moderator (as shown in annotated Figure 1 below), when the near conference endpoint 100 is facilitating the conference call. Doing so would allow for the user of the near endpoint (acting

as the moderator) to control the flow and organization of the conference. *See id.* at 1:53-55 (“...a designated moderator(s) to **control** the flow and/or **organization** the conference.”); *see also* Reasons to Combine Drell and Knappe, *supra* ¶¶ 148-168.

391. Accordingly, Drell in combination with Knappe teaches that the user of the near conference endpoint 100 is a “*moderator*” computer, as claimed:



Drell (Ex. 1008) at FIG. 1 (annotated).

392. Thus, Drell in combination with Knappe teaches that the instructions are to be executed by a CPU of the moderator near conference endpoint 100, which renders obvious that the instructions are “*to be executed by a local moderator computer.*”

***[9.1] receiving audio data from a telephone through a public switched telephone network (PSTN)***

393. The prior art renders obvious receiving audio data from a telephone through a Public Switched Telephone Network (PSTN).

394. **First**, as discussed at [9.P], Drell in combination with Knappe teaches a moderator near conference endpoint 100 that executes instructions for a multi-participant conference process.

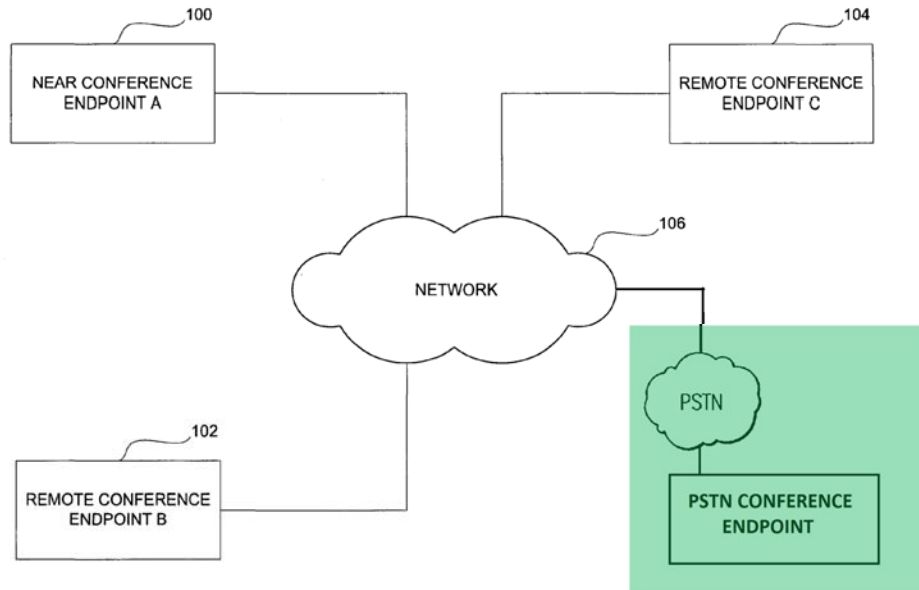
395. **Second**, for the reason discussed in connection with [1.1], Drell in combination with Knappe, teaches receiving audio data from a telephone through a public switched telephone network (PSTN). *See supra* ¶¶ 181-185. Specifically, Drell teaches a PSTN conference endpoint participates in the conference with the moderator near conference endpoint 100:

Network 106 may be of any type suitable for the transmission of audio and video data between and among **near conference endpoint 100** and remote conference endpoints...the invention contemplates the use of two or more networks (for example, **the PSTN** and the Internet) **to connect conference endpoints** utilizing different communication protocols.

Drell (Ex. 1008) at 3:20-27. A POSITA would have been familiar with the PSTN and it would understand that standard telephones communicate over the PSTN.

Drell further explains that “while only two remote conference endpoints are depicted in FIG. 1 for the purpose of clarity, a greater number of remote conference endpoints may be accommodated by near conference endpoint 100.”

Drell at 3:15-19.



*Id.* at FIG. 1 (annotated).

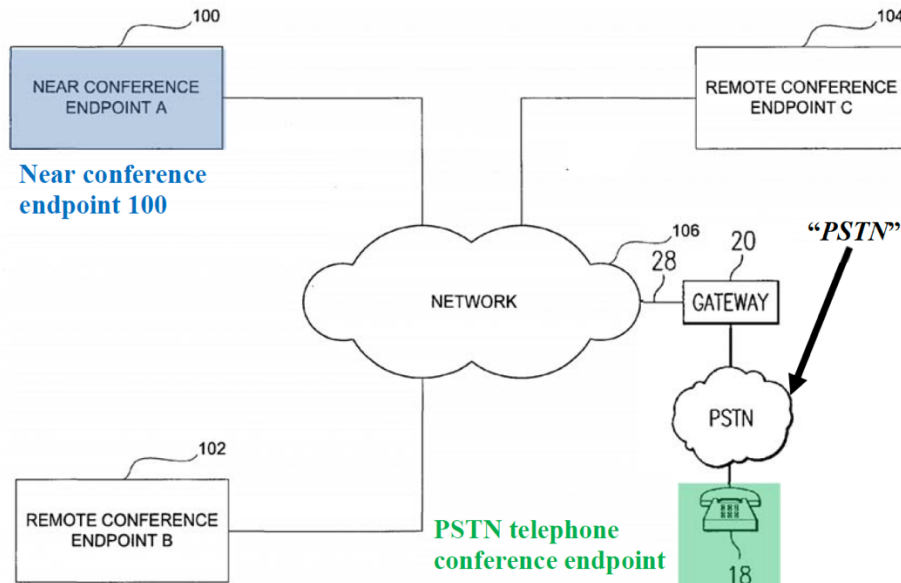
396. **Third**, as discussed in [1.1], Knappe teaches that when a PSTN telephone participates in a conference over a PSTN network and an IP network, audio data from the PSTN telephone endpoint would be communicated through a gateway that provides conversion of data between the two different networks. *See supra* ¶¶ 183-185. Thus, Knappe describes use of a standard analog telephone as a “telephone” that communicates “*through a public switched telephone network (PSTN).*”

397. In view of Knappe’s teachings, it would have been obvious to a POSITA to implement Drell’s PSTN conference endpoint as a PSTN telephone, because it would facilitate the communications of audio data over the PSTN during conferencing. At the time, PSTN telephones were the most common commercially

available endpoints for communicating over the PSTN; therefore, a POSITA would have had a reasonable expectation of success and the result would have been predictable in applying Knappe's PSTN telephone teachings to Drell's PSTN conference endpoint. Indeed, before the earliest claimed priority of the '525 patent, I utilized PSTN telephones in conferencing systems.

398. Further, in view of Knappe's teachings, as discussed in [1.1] it would have been obvious to a POSITA to utilize in Drell's system a gateway that receives standard audio data from PSTN telephone conference endpoints and converts the audio data into a format suitable for conference endpoints on an Internet network. Doing so would beneficially allow Drell's PSTN telephone conference endpoints to communicate, during the conference, audio data to the near conference endpoint 100, over a PSTN network and an Internet network 106. *See supra* ¶¶ 184-185; *see also* Reasons to Combine Drell and Knappe, *supra* ¶¶ 148-168. The combined teachings of Drell and Knappe on this point are illustrated in the annotated Figure 1 below, showing how a POSITA would have found obvious to apply Knappe's teaching of interconnecting the PSTN and the Internet to Drell's system.

399. As illustrated below in annotated Figure 1, it would have been obvious to a POSITA that Drell's PSTN conference endpoint (implemented as a telephone) would transmit audio data through a PSTN and a gateway to the near conference endpoint 100, as Knappe teaches.



400. Thus, Drell in combination with Knappe teaches that the moderator near conference endpoint 100 receives audio data from a PSTN telephone conference endpoint through a PSTN, which renders obvious “*receiving audio data from a telephone through a public switched telephone network (PTSN),*” as claimed.

***[9.2] receiving audio data and video data from a first remote computer through an Internet;***

401. The prior art renders obvious receiving audio data and video data from a first remote computer through an Internet.

402. **First**, as discussed at [9.P], Drell in combination with Knappe teaches a moderator near conference endpoint 100 that executes instructions for a multi-participant conference process.

403. **Second**, for the reasons discussed in [1.2] and [1.3] above, Drell in combination with Knappe, discloses receiving audio and video data from a first remote computer through an Internet. *See also* Drell (Ex. 1008) at 3:33-34 (“[N]etwork 106 may comprise a packet switched network, such as the Internet.”); *see also id.* at 3:4-23 (“**Remote conference endpoints 102 and 104** may comprise, for example, conventional videoconferencing devices equipped to **transmit and receive both video (image) data and audio (speech) data...** **Network 106** may be of any type suitable for the transmission of audio and video data between and among near conference endpoint 100 and remote conference endpoints 102 and 104.”); *id.* at 4:25-31 (“The processing trains process **audio and video data streams received from remote conferencing endpoints 102 and 104.**”).

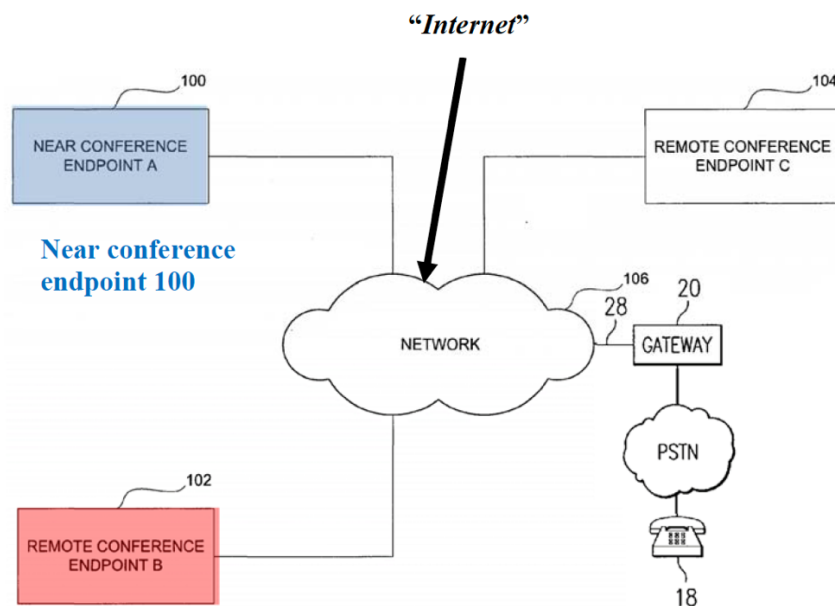
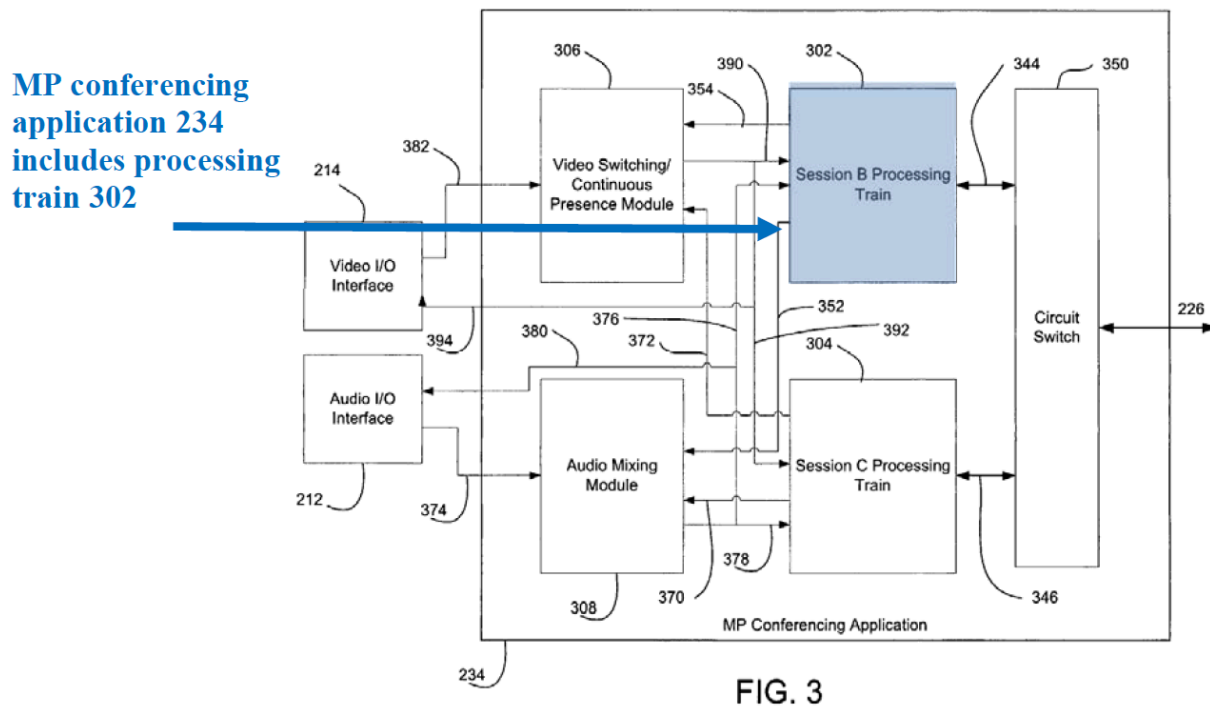


FIG. 1

404. As shown below at Drell’s Figure 3, the MP conferencing application 234 (of the moderator near conference endpoint 100) includes processing train 302 that receives audio and video data from the first remote conference endpoint 102:



405. Accordingly, a POSITA would have understood that the memory in Drell’s near conference endpoint 100 includes instructions for receiving audio and video data from the remote conference endpoint 102 through Internet network 106, which renders obvious “receiving audio data and video data from a first remote [endpoint] through an Internet.”

406. Further, Drell teaches that the “[r]emote conference endpoints 102 and 104...may comprise... conventional videoconferencing devices equipped to transmit and receive both video...and audio,” (*id.* at 3:8-12) and that conference

endpoint 100 is a videoconferencing device that includes software, memory and a CPU (*id.* at 4:39-53, FIG. 2) for communicating both video and audio (*id.* at 3:4-23). Because of the similar videoconferencing functionality of the remote conference endpoints 102 and 104, it would have been obvious to a POSITA for the remote conference endpoints 102 and 104 to similarly employ software, memory and CPU. Accordingly, it would have been obvious to a POSITA for Drell's remote conference endpoints 102 and 104 to each be a "*computer*" as claimed.

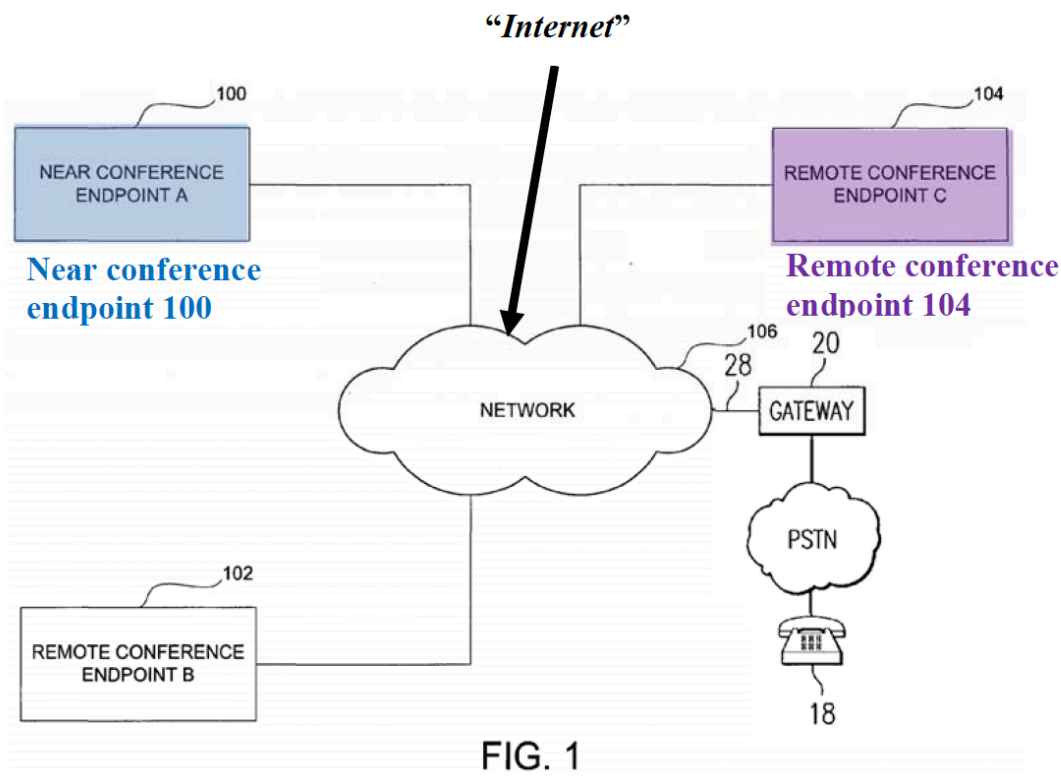
407. Thus, Drell teaches that the moderator near conference endpoint 100 receives audio data and video data from a first remote conference endpoint 102 (a computer) through the Internet network 106, which renders obvious "*receiving audio data and video data from a first remote computer through an Internet,*" as claimed.

***[9.3] receiving audio data and video data from a second remote computer through the Internet.***

408. The prior art renders obvious receiving audio data and video data from a second remote computer through an Internet.

409. **First**, as discussed at [9.P], Drell in combination with Knappe teaches a moderator near conference endpoint 100 that executes instructions for a multi-participant conference process. Further, as discussed at [9.2], Drell teaches that the moderator near conference endpoint 100 receives both audio data and video data

from remote conference endpoints 102 and 104 through the Internet network 106. Further, as discussed at [9.2], each of remote conference endpoints 102 and 104 include a CPU for transmitting and receiving video data and audio data. Thus, Internet network 106 renders obvious “*the Internet*,” and remote conference endpoint 104 renders obvious “*a second remote computer*,” as claimed. This is illustrated in Figure 1, annotated below:



410. As shown below at Drell’s Figure 3, the MP conferencing application 234 (of the moderator near conference endpoint 100) includes processing train 304 that receives audio and video data from the second remote conference endpoint 104:

MP conferencing application 234 includes processing train 304

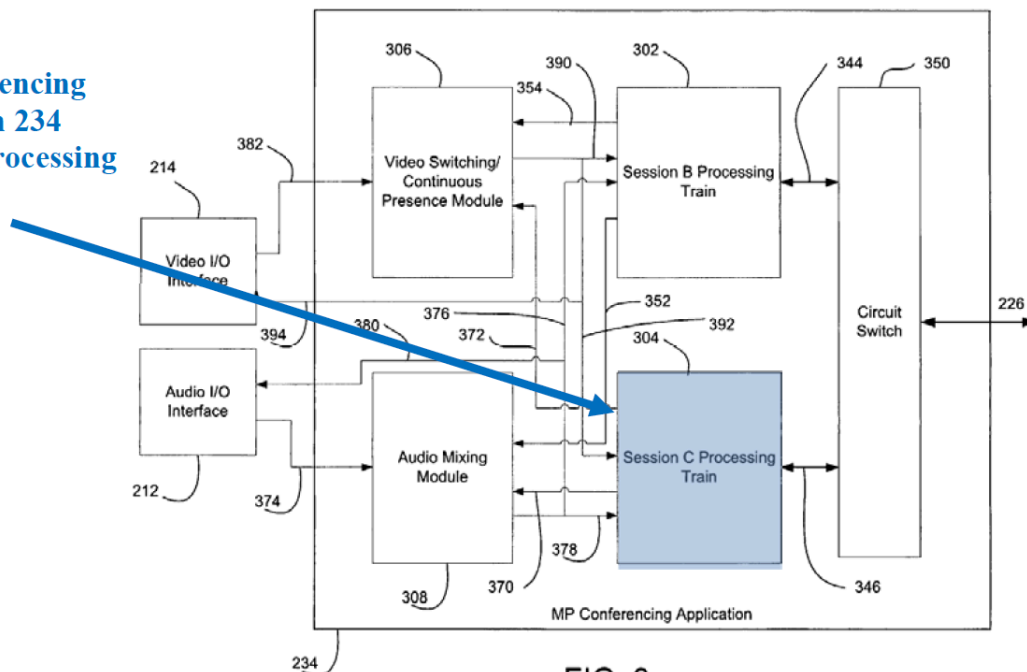


FIG. 3

411. Thus, Drell teaches that its moderator near conference endpoint 100 receives audio data and video data from the second remote conference endpoint 104 (a computer) through the Internet network 106, which renders obvious “receiving audio data and video data from a second remote computer through the Internet,” as claimed.

**[9.4] receiving audio data from a local audio input device communicatively connected to the local moderator computer;**

412. The prior art renders obvious receiving audio data from a local audio input device communicatively connected to the local moderator computer.

413. **First**, as discussed at [9.P], Drell in combination with Knappe teaches that the moderator near conference endpoint 100 includes within its memory a multi-point (MP) conferencing application 234.

414. **Second**, Drell teaches that an audio mixing module 308 of the MP conferencing application 234 receives locally generated audio data from a microphone 204 communicatively connected to the near conference endpoint 100 via an audio I/O interface 212:

Reference is now directed to FIG. 2, which depicts in block form various components of near conference endpoint 100. A conventional video camera 202 and **microphone 204** are operative to generate video and **audio signals representative of the images and speech of the near conference participant (the person or persons co-located with near videoconference endpoint 100)**. A video monitor 208 and loudspeaker 210 present images and speech of the remote conference participants combined with **locally generated** images and **speech**. An audio I/O interface 212, configured to perform A/D and D/A conversion and related processing of audio signals, couples microphone 204 and loudspeaker 210 to CPU 220 and memory 222 through bus 226.

Drell (Ex. 1008) at 3:39-51.

**MP conferencing application 234** includes a circuit switch 350, a plurality of processing trains 30 and 304, a video switching/continuous presence module 306, and an **audio mixing module 308**.

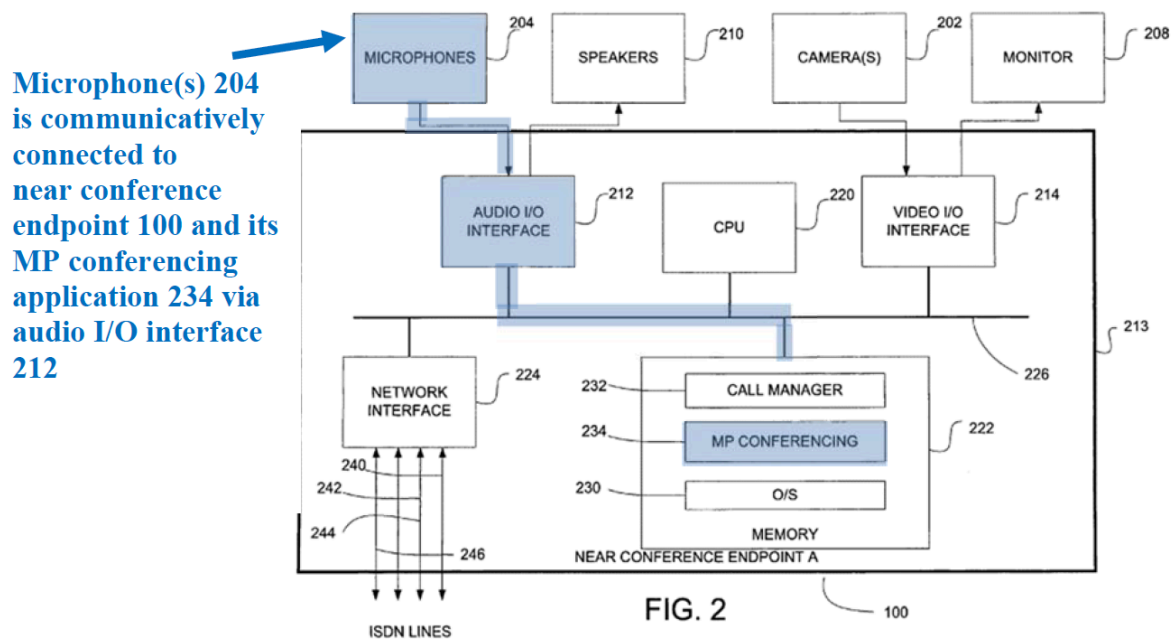
*Id.* at 4:32-38.

**Audio mixing module 308 is configured to...receive[.]...locally generated audio data (received from audio I/O interface 212 via audio data path 374, and typically being representative of the speech of the near conference participant(s)).**

*Id.* at 5:8-13; *see also id.* at Abstract, 2:34-52.

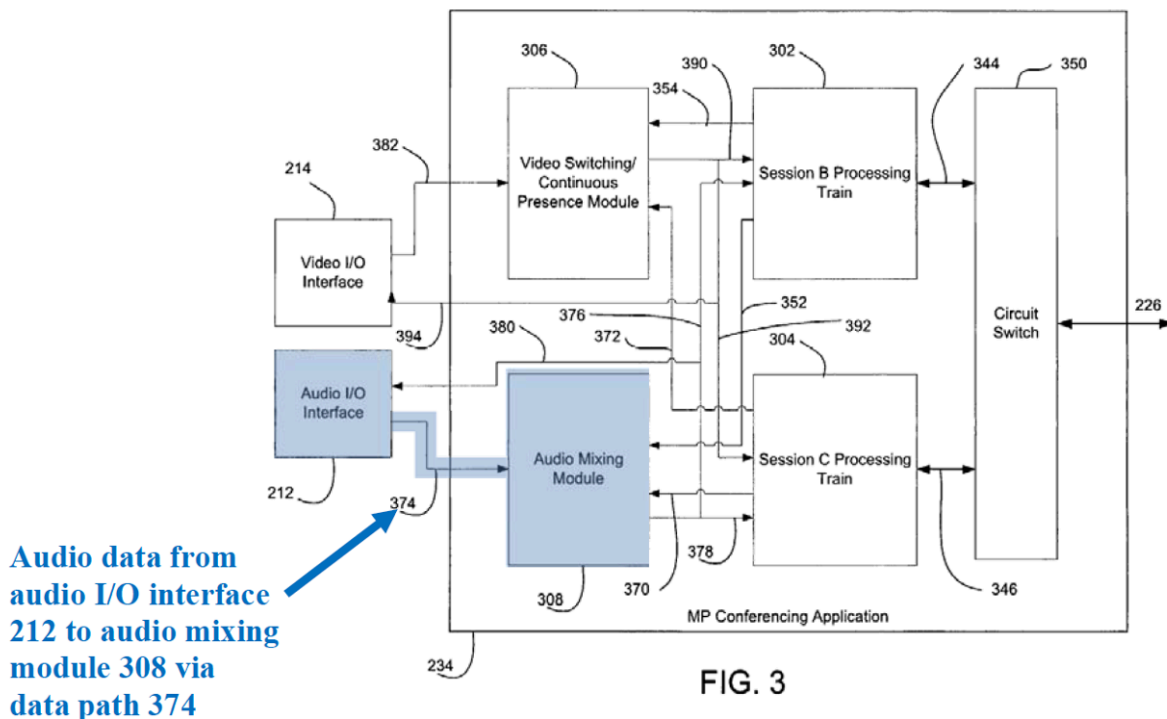
415. A POSITA would have understood that Drell's "audio I/O interface" is an "audio input/output interface," that is, it is an interface that provides for both

audio input and audio output. Drell's microphone 204 is a *"local audio input device communicatively connected to the local moderator computer"* because it provides audio signals representative of the *"locally generated... speech"* of co-located participants to Drell's near conference endpoint via an audio input/output interface. As shown below at Drell's Figure 2, the microphone(s) 204 is communicatively connected via audio I/O interface 212 to the near conference endpoint 100, which includes MP conferencing application 234:



*Id.* at FIG. 2 (annotated).

416. As further shown below at Drell's Figure 3, the locally generated audio data is communicated from the audio I/O interface 212 to audio mixing module 308 of the MP conferencing application 234 via data path 374:



417. Drell’s microphone 204 that is communicatively connected to the moderator near conference endpoint 100 corresponds to “*a local audio input device communicatively connected to the local moderator computer,*” as claimed.

418. Thus, Drell teaches receiving locally generated audio data from a local microphone 204 communicatively connected to the moderator near conference endpoint 100, which renders obvious “*receiving audio data from a local audio input device communicatively connected to the local moderator computer,*” as claimed.

***[9.5] receiving video data from a local video input device communicatively connected to the local moderator computer***

419. The prior art renders obvious receiving video data from a local video input device communicatively connected to the local moderator computer.

420. **First**, as discussed at [9.P], Drell in combination with Knappe teaches that the moderator near conference endpoint 100 includes within its memory a multi-point (MP) conferencing application 234.

421. **Second**, Drell teaches that a video switching/continuous presence module 306 of the MP conferencing application 234 receives locally generated video data from a video camera 202 coupled to the near conference endpoint 100 through a video I/O interface 214:

Reference is now directed to FIG. 2, which depicts in block form various components of near conference endpoint 100. A conventional **video camera 202** and microphone 204 are operative **to generate video and audio signals representative of the images and speech of the near conference participant (the person or persons co-located with near videoconference endpoint 100)**. A video monitor 208 and loudspeaker 210 present images and speech of the remote conference participants combined with **locally generated images** and speech. An audio I/O interface 212, configured to perform A/D and D/A conversion and related processing of audio signals, couples microphone 204 and loudspeaker 210 to CPU 220 and memory 222 through bus 226. Similarly, **video camera 202 and monitor 208 are coupled to console electronics 213 through video I/O interface 214**.

Drell (Ex. 1008) at 3:39-53.

**MP conferencing application 234 includes** a circuit switch 350, a plurality of processing trains 302 and 304, a **video MP conferencing application 234 includes** a circuit switch 350, a plurality of processing trains 302 and 304, a **video**

*Id.* at 4:32-38.

**Video switching/continuous presence module 306...receive[s]... locally generated video data (received from video I/O interface 214 via video data path 382, and being typically representative of images of the near conference participants).**

*Id.* at 5:26-31.

422. A POSITA would have understood that Drell's "video I/O interface" is an "video input/output interface," that is, it is an interface that provides for both video input and video output. Drell's video camera 202 is a "*local video input device communicatively connected to the local moderator computer*" because it provides video signals representative of the "locally generated images" of co-located participants to Drell's near conference endpoint via a video input/output interface.

423. As shown below at Drell's Figure 2, the camera(s) 202 I communicatively connected video I/O interface 214 to the near conference endpoint 100, which includes MP conferencing application 234:

Camera(s) 202 is communicatively connected to near conference endpoint 100 and its MP conferencing application 234 via video I/O interface 214

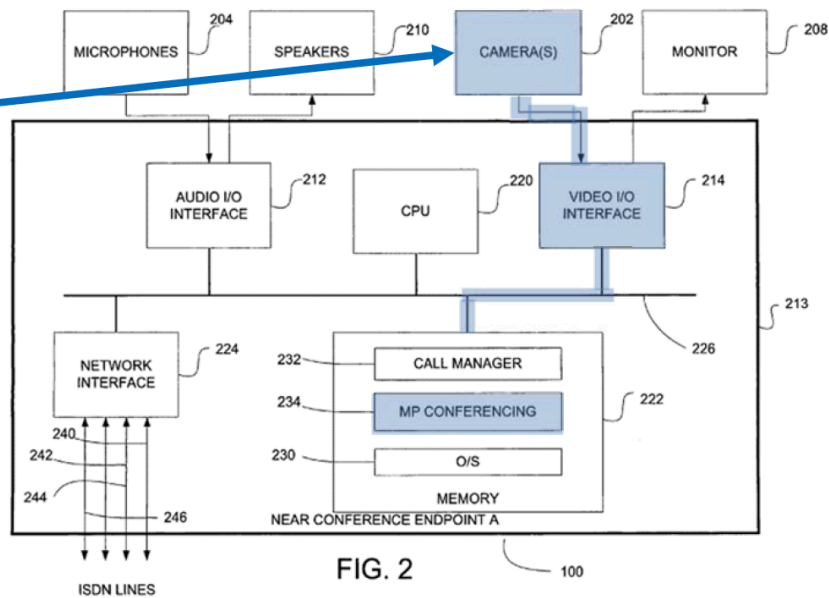


FIG. 2

*Id.* at FIG. 2 (annotated).

424. As shown below at Drell's Figure 3, the locally generated video data is communicated from the video I/O interface 214 to video switching/continuous presence module 306 of the MP conferencing application 234 via data path 382:

Video data from  
video I/O  
interface 214 to  
video switching/  
continuous  
presence module  
306 via data  
path 382

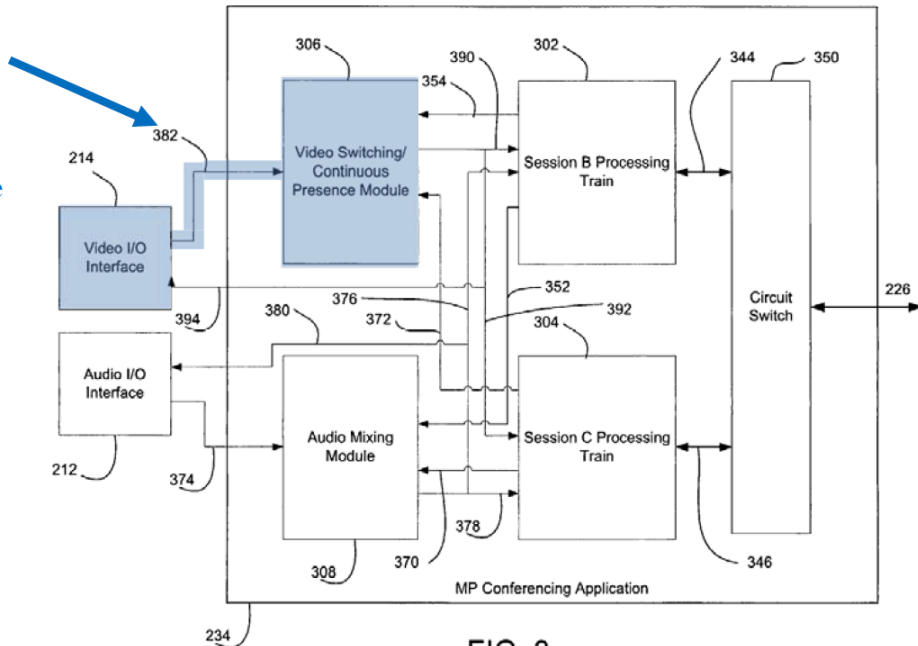


FIG. 3

*Id.* at FIG. 3 (annotated).

425. Drell’s video camera 202 that is communicatively connected to the moderator near conference endpoint 100 corresponds to “a local video input device communicatively connected to the local moderator computer,” as claimed.

426. Thus, Drell teaches receiving locally generated video data from a local video camera 202 communicatively connected to the moderator near conference endpoint 100, which renders obvious “receiving video data from a local video input device communicatively connected to the local moderator computer,” as claimed.

**[9.6] mixing the audio data received from the telephone with the audio data received from the local audio input device and the video data received from the local video input device into mixed audio-video data;**

427. The prior art renders obvious mixing the audio data received from the telephone with the audio data received from the local audio input device and the video data received from the local video input device into mixed audio-video data.

428. **First**, as discussed at [9.1], Drell in combination with Knappe teaches that the near conference endpoint 100 receives audio data from a PSTN telephone conference endpoint. Further, as discussed at [9.4], Drell teaches that an audio mixing module 308 of the MP conferencing application 234 at the near conference endpoint 100 receives locally generated audio data from a microphone 204 (local audio input device) via an audio I/O interface 212. Further still, as discussed at [9.5], a video switching/continuous presence module 306 of the MP conferencing application 234 receives locally generated video data from a video camera 202 (local video input device) via a video I/O interface 214.

429. **Second**, Drell teaches that the audio mixing module 308 of the MP conferencing application 234 mixes the audio data received from remote conference endpoints with the locally generated audio data from microphone 204 via the audio I/O interface 212 a mixed audio data signal:

**Audio mixing module 308 is configured to combine audio data received from remote conference endpoints 102 and 104 with locally generated audio data (received from audio I/O interface 212 via audio data path 374, and typically being representative of the speech of the near conference participant(s)).** The term "combine" is used in its broadest and most general sense and is intended to cover any operation wherein audio mixing module 308 generates an output audio data stream (or plurality of output audio data streams)

based on information contained in the remotely and locally generated audio data input streams. For example, **audio mixing module 308 may simply mix the received audio input data streams.**

*Id.* at 5:8-23; *see also id* at Abstract, 2:34-52, 3:45-48, 4:25-31.

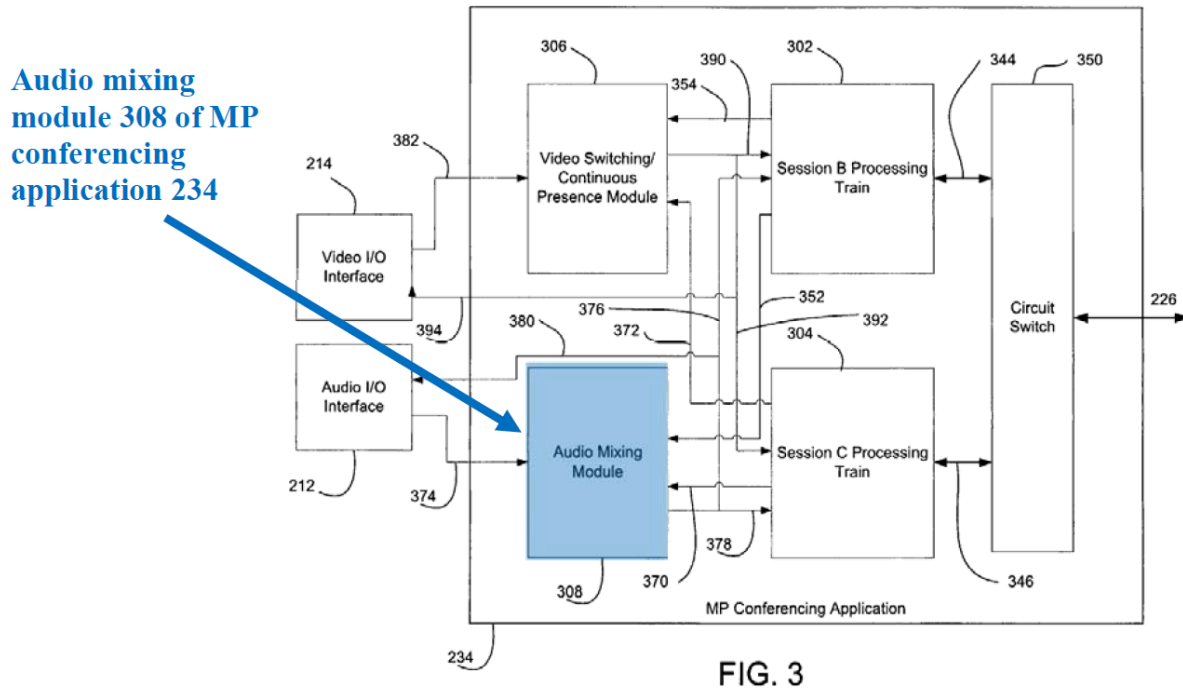


FIG. 3

*Id.* at FIG. 3 (annotated).

430. It would have been obvious to a POSITA that when Drell's conference system includes an additional conference endpoint connected via the PSTN (as discussed above), the audio data received from the PSTN telephone conference endpoint would also be combined in the audio mixing module 308.

431. Also, it would have been obvious to a POSITA that, during conferencing, the audio mixing module 308 would mix the audio data stream from the moderator near conference endpoint 100 with the audio data stream from the

PSTN telephone conference endpoint. This is so that the participant using the PSTN telephone conference endpoint could participate in the conference call, with the participant's speech being included the mixed audio data streams subsequently transmitted to the other participants. *See id.* at 5:48-54, 6:46-52, 7:30-31. Because Drell teaches that audio mixing module 308 combines audio from the remote conference endpoints with the "locally generated audio data" from microphone 204 (*id.* at 5:8-13), it would have been obvious to a POSITA for audio mixing module 308 to combine the audio data from the PSTN telephone conference endpoint with the audio data from microphone 204. Thus, Drell's audio mixing module 308 renders obvious "*mixing the audio data received from the telephone with the audio data received from the local audio input device,*" as claimed.

432. **Third**, Drell's MP conferencing application 234 also includes a video switching/continuous presence module 306 that mixes video data received from remote conference endpoints with the locally generated video data received from video camera 202 via the video I/O interface 214 and outputs a mixed video data:

**Video switching/continuous presence module 306 combines video data received from remote conference endpoints 102 and 104 with locally generated video data (received from video I/O interface 214 via video data path 382, and being typically representative of images of the near conference participants).** Again, the term "combine" is used in its broadest and most general sense. In one mode of operation, video switching/continuous presence module 306 may select one of the video data input streams for output based on predetermined criteria (for example, it may select for output the video data stream corresponding to the conference endpoint of the currently speaking participants. In a

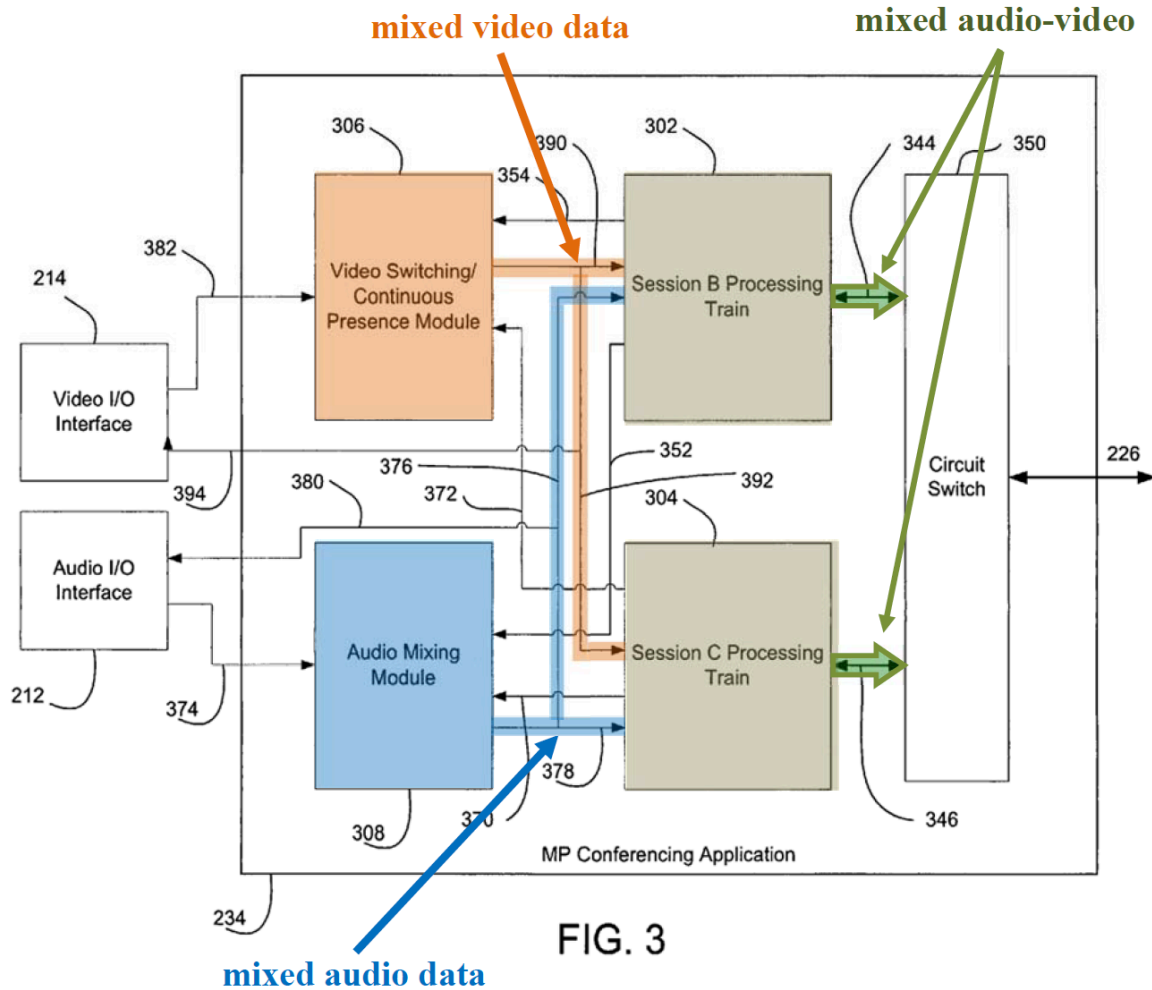
second mode of operation (referred to as the "continuous presence mode"), **video switching/continuous presence module 306 may construct a composite image wherein images corresponding to conference endpoints are displayed in different sectors of the composite image. The video data stream output (or plurality of outputs) from video switching continuous presence module 306 is thereafter distributed to processing trains 302 and 304...via video data paths 390, 392.**

*Id.* at 5:26-47. Because Drell teaches creating a composite image including images from the conference endpoints—including near conference endpoint 100—a POSITA would have understood that the video output by video switching/continuous presence module 306 would include video data from video camera 202 (*“the video data received from the local video input device”*).

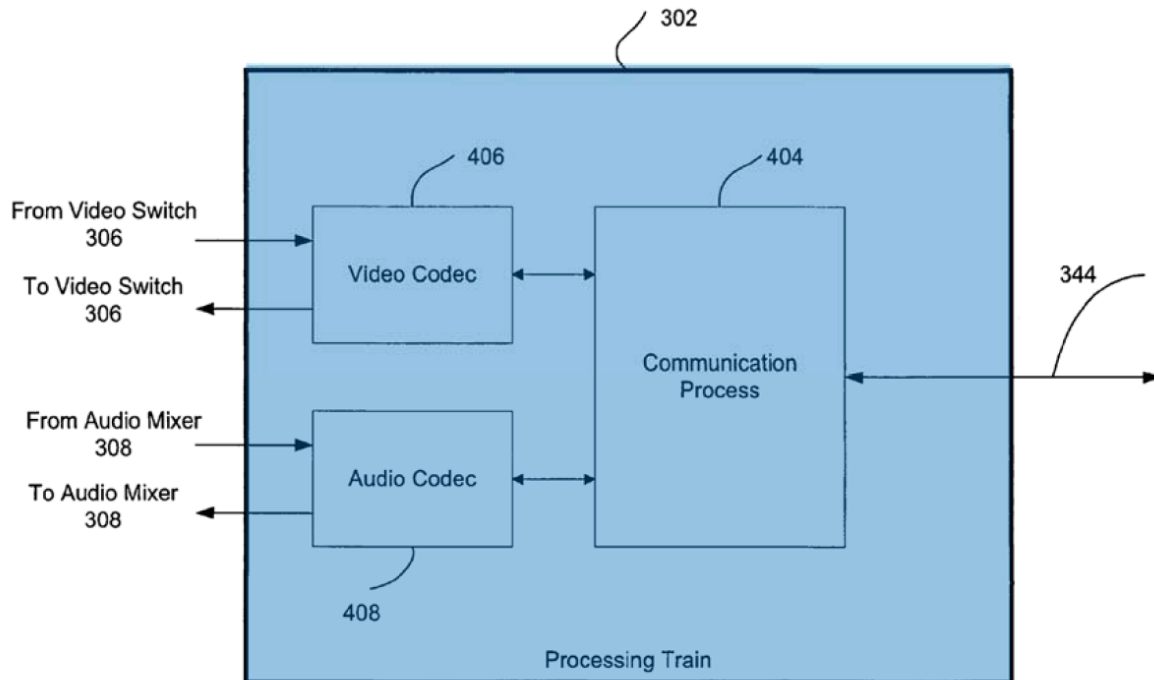
433. Drell further teaches that the processing trains (e.g., 302/304), for each endpoint, include a “communication process 404” that receives mixed audio data from the audio mixing module 308 and receives a mixed video data stream output by the video switching/continuous presence module 306. *Id.* at 4:21-24; 5:65-66, 6:33-45, FIG. 4. The communication process 404 then “multiplexes the encoded audio and video data streams into **a single audio/video data stream.**” *Id.* at 6:46- 48 and 5:51-63 (“For each encoded audio/video data stream, the circuit switch 350 allocates an IMUX which aggregates the data streams into a wideband data stream on the bus 226, preferably utilizing a bonding protocol.”).

434. Drell’s audio mixing module 308, and video switching/continuous presence module 306, and communication process 404 collectively operate to mix

the audio and video data streams received from multiple remote conference endpoints into a single audio and video data stream, which renders obvious “*mixed audio-video data.*”



*Id.* at FIG. 3 (annotated).



*Id.* at FIG. 4 (annotated).

435. Thus, Drell in combination with Knappe teaches that the MP conferencing application 234 includes an audio mixing module 308, which mixes audio data from the PSTN telephone conference endpoint with locally generated audio data from the microphone 204, and processing trains 302 and 304 which further mix the mixed audio data with the locally generated video data from the video camera 204 into mixed audio-video data, which renders obvious “*mixing the audio data received from the telephone with the audio data received from the local audio input device and the video data received from the local video input device into mixed audio-video data,*” as claimed.

436. **Additionally**, I understand that a district court previously construed other “*mixed*” data limitations in other claims of the ’525 patent to “require exclusion of some subject matter—data from the recipient of the mixed data—from the mixed data.” Memorandum Opinion and Order Regarding Claims Construction, *Meetrix IP, LLC v. Citrix Systems, Inc.* (W.D. Tex. Dec. 1, 2017) (Ex. 1038) at 30. The claimed “*mixed audio-video data*” is recited in [9.7], below, as being transmitted to the “*first remote computer*” and “*second remote computer.*” Thus, by analogy, the district court’s claim construction might be argued, by analogy, to require the “*mixed audio-video data*” to not include the “*audio data and video data*” received from the “*first remote computer*” and “*second remote computer.*” Even if such an interpretation were argued for claim 9, however, the prior art teaches this limitation.

437. Specifically, Knappe teaches mixing audio streams so that each participant receives a mixed audio stream that includes only the audio of the other conference participants:

During the multiparty communications sessions, participants are connected and stream media through the conference bridge 32. The media is cross connected and mixed to produce conference output streams for each participant. **The conference output stream for a participant includes the media of all other participants**, a subset of other participants or other suitable mix dictated by the type of multiparty session and/or the participant.

Knappe (Ex. 1006) at 5:35-39.

During normal operation, **each participant receives the audio input of each other participant.** Thus, for example, the conference output stream of participant 1 includes the audio inputs of participants 2–5. Similarly, the conference output stream of participant 2 includes the audio inputs of participants 1 and 3–5. The conference output stream of participant 3 includes the audio inputs of participants 1–2 and 4–5. The conference output stream of participant 4 includes the audio inputs of participants 1–3 and 5. The conference output stream of participant 5 includes the audio inputs of participants 1–4.

*Id.* at 7:1–11.

438. Such an implementation would complement Drell’s teaching of generating “separate processing trains for signal streams from/to each of the remote conference endpoints.” Drell (Ex. 1008) at 2:19–21. A POSITA would have known that receiving one’s own audio creates the possibility of audio feedback or echo, which adversely affects the conference experience.

439. Knappe further describes limiting the number of participants’ audio streams that are mixed together, which avoids unintelligible overlapping speech:

It should be noted that the manner in which participant voice streams are transmitted in accordance with the present invention, may depend upon the priority of each participant. For example, in one embodiment, **only a particular number of speakers will be allowed to transmit simultaneously** (e.g., only four participants will be allowed to speak at once). In this configuration, the system may use the priority associated with each participant to determine which speakers will be allowed to “cut in” to the conversation. If a participant having a high priority attempts to cut in while a participant having a low priority is speaking, the system may automatically stop transmission of the voice stream from the low priority participant, in order to accommodate and transmit the voice stream of the high priority participant, **without rendering the conversation unintelligible.**

Knappe (Ex. 1006) at 9:15-30.

The audio input 84 of the conference moderator may be amplified and/or the audio input 84 of the remaining participants attenuated to focus on or provide higher prominence to the audio input 84 of the conference moderator. A higher prominence is provided by **increasing the intelligibility of the moderator** relative to the remaining participants.

*Id.* at 7:12-17.

440. In view of these teachings in Knappe, it would have been obvious to a POSITA for Drell's audio mixing module 308 to exclude from its mixed audio output the audio data received from the remote conference endpoints 102 and 104 ("*first and second remote computers*" as discussed above in [9.2] and [9.3]). A POSITA would have been motivated to exclude these audio data because doing so would beneficially (1) further Drell's goal of generating separate processing instances for each conference endpoint, (2) increase intelligibility of the moderator (3) enhance the moderator's control over the conference, and (4) avoid undesirable audio effects, such as feedback. *See also* Reasons to Combine Drell and Knappe, *supra* ¶¶ 148-168.

441. Thus, even when the "*mixed audio-video data*" is interpreted not to include the "*audio data and video data*" received from the "*first remote computer*" and "*second remote computer*," Drell and Knappe render obvious "*mixing the audio data received from the telephone with the audio data received from the local*

*audio input device and the video data received from the local video input device into mixed audio-video data.”*

***[9.7] transmitting the mixed audio-video data to the first remote computer and the second remote computer***

442. The prior art renders obvious transmitting the mixed audio-video data to the first remote computer and the second remote computer.

443. **First**, as discussed at [9.6], Drell in combination with Knappe teaches generating mixed audio-video data. As discussed at [9.2] and [9.3], the prior art teaches remote conference endpoints 102 and 104, corresponding to “*the first remote computer and the second remote computer,*” respectively.

444. **Second**, Drell teaches transmitting the mixed audio-video data to each of the plurality of remote conference endpoints. Thus, Drell expressly suggests transmitting combined audio and video signals (“*mixed audio-video data*”) to multiple remote conference endpoints (“*the first remote computer and the second remote computer*”):

processing the received audio and video signals;

combining the processed audio and video signals with the local audio and video signals; and

transmitting the combined audio and video signals to each of the plurality of remote conference endpoints.

Drell (Ex. 1008), Claim 1.

**Network 106 may be of any type suitable for the transmission of audio and video data between and among near conference endpoint 100 and remote conference endpoints 102 and 104....** [N]etwork 106 may comprise a packet switched network, such as the Internet.... Console electronics 213 additionally include a central processing unit (CPU) 220 for executing program instructions, a memory 222 for storing applications, data, and other information, and a network interface 224 for connecting near conference endpoint 100 to network 106.... At least one bus 226 interconnects the components of console electronics 213.

*Id.* at 3:20-64.

**In the transmission mode... [t]he received data streams are then encoded and combined to form a mixed encoded audio/video data stream, and the encoded audio/video data stream is transmitted** to the circuit switch 350 via data path 344. ... For each encoded audio/video data stream, the circuit switch 350 allocates an IMUX which aggregates the data streams into a wideband data stream on the bus 226, preferably utilizing a bonding protocol.

*Id.* at 5:51-63.

445. Thus, the prior art teaches transmitting the mixed audio-video data to first remote conference endpoint 102 and the second remote conference endpoint 104 via network 106, which renders obvious “*transmitting the mixed audio-video data to the first remote computer and the second remote computer,*” as claimed.

**[9.8] *mixing the audio data received from the first remote computer and the audio data received from the second remote computer with the audio received from the local audio input device into mixed audio data***

446. The prior art renders obvious mixing the audio data received from the first remote computer and the audio data received from the second remote

computer with the audio data received from the local audio input device into mixed audio data.

447. **First**, as discussed at [9.2]-[9.3], Drell teaches that the near conference endpoint 100 receives audio data from the first remote conference endpoint 102 and the second remote conference endpoint 104. Further, as discussed at [9.4], Drell teaches that an audio mixing module 308 of the MP conferencing application 234 at the near conference endpoint 100 receives locally generated audio data from a microphone 204 (local audio input device) via an audio I/O interface 212.

448. **Second**, Drell teaches that the audio mixing module 308 of the MP conferencing application 234 mixes the audio data received from remote conference endpoints 102 and 104 with the locally generated audio data to generate a mixed output audio data:

**Audio mixing module 308 is configured to combine audio data received from remote conference endpoints 102 and 104 with locally generated audio data** (received from audio I/O interface 212 via audio data path 374, and typically being representative of the speech of the near conference participant(s)). The term “combine” is used in its broadest and most general sense and is intended to cover any operation **wherein audio mixing module 308 generates an output audio data stream** (or plurality of output audio data streams) **based on information contained in the remotely and locally generated audio data input streams**. For example, **audio mixing module 308 may simply mix the received audio input data streams**.

Drell (Ex. 1008) at 5:8-23; *see also id.* at Abstract, 2:34-52, 3:45-48, 4:25-31.

449. Thus, Drell in combination with Knappe teaches that the audio mixing module 308 mixes the audio data received from the first remote conference endpoint 102 and the second remote conference endpoint 104 with the locally generated audio data received from the microphone 204 into mixed audio data, which renders obvious “*mixing the audio data received from the first remote computer and the audio data received from the second remote computer with the audio data received from the local audio input device into mixed audio data,*” as claimed.

450. **Additionally**, as discussed at [9.6], a district court construed other “*mixed*” data limitations to require the exclusion of data from the recipient of the mixed data. The claimed “*mixed audio data*” is recited in [9.9]-[9.10] as being compressed and transmitted to the “*telephone.*” To the extent that claim 9 is interpreted to require the “*mixed audio data*” not to include the “*audio data*” received from the “*telephone,*” the prior art teaches this limitation.

451. As discussed at [9.6], Knappe teaches mixing audio streams so that each participant receives a mixed audio stream that includes only the audio of the other conference participants. Knappe (Ex. 1006) at 5:35-39; 7:1-11. Knappe also discloses limiting the number of simultaneous speakers. *Id.* at 9:18-21.

452. I note that claim 9 does not recite that the two “mixing” steps in [9.6] and [9.8] occur simultaneously. A POSITA would have understood that during the

course of a multi-participant conference, different speakers will speak at different times. Accordingly, the two “*mixing*” steps may occur at different times, and therefore, the audio streams (or audio and video streams) required to be mixed by the claim language simply reflect different groups of participants who are speaking or otherwise actively participating at different points in time.

453. In view of these teachings in Knappe, it would have been obvious to a POSITA for Drell’s audio mixing module 308 to exclude from its mixed audio output the audio data received from the PSTN telephone conference endpoint (“*telephone*” as discussed above in [9.1]). A POSITA would have been motivated to exclude these audio data because doing so would beneficially (1) further Drell’s goal of generating separate processing instances for each conference endpoint, (2) increase intelligibility of the moderator (3) enhance the moderator’s control over the conference, and (4) avoid undesirable audio effects, such as feedback. *See also* Reasons to Combine Drell and Knappe, *supra* ¶¶ 148-168.

454. Thus, even the “*mixed audio data*” is interpreted as requiring not to include the “*audio data*” received from the “*telephone*,” Drell and Knappe render obvious “*mixing the audio data received from the first remote computer and the audio data received from the second remote computer with the audio data received from the local audio input device into mixed audio data.*”

***[9.9] compressing the mixed audio data;***

455. The prior art renders obvious compressing the mixed audio data.

456. **First**, as discussed at [9.8], Drell in combination with Knappe teaches generating mixed audio data.

457. **Second**, Drell teaches that the MP conferencing application 234 has an audio codec 408 that uses a compression algorithm to compress the mixed audio data:

**In the transmit mode...[a]udio codec 408 encodes the audio data stream output by audio mixing module 308 (representative, for example, of the blended speech of conference participants located at near conference endpoint 100 and remote conference endpoints 102 and 104) using a standard or proprietary audio compression algorithm (e.g., G.711) and delivers the encoded audio data to communication process 404**

Drell (Ex. 1008) at 6:33-45; *see also* Drell at 5:8-25, FIGS. 1-4. It would have been obvious to a POSITA encoding audio data using a compression algorithm is “*compressing*” the audio data.

458. **Alternatively**, Drell teaches that its communication process includes a H.323 (packet-based) process:

The (MP) conferencing application is operable to process the plural signal streams received through the communication ports. Generally, the MP conferencing application generates separate processing trains for **signal streams from/to each of the remote conference endpoints**. The processing trains each comprise a **communication process and a set of codecs**...The communication process...may for example comprise an...**H.323 (packet-based) process**.

*Id.* at 2:17-33.

459. A POSITA would have understood that Drell, which can perform H.323 processing, discloses utilizing codecs and protocols described in H.323 for encoding data into compressed data. *See, e.g.* U.S. Patent No. 6,553,515 (Ex. 1025) at 4:35-56 (“**H.323** is adapted to support many aspects of multimedia...[and] generally is regarded as the standard for VoIP...This standard also specifies lowest- common denominator **compression protocols, such as G.711...G.723 and G.729.**”); U.S. 6,850,496 (Ex. 1024) at 3:56-67 (“[A] codec (compressor/decompressor) algorithm [] will be used to transmit capture channel data from encoder 24 to decoder 30. The codec may use a technique as simple as Pulse-Code Modulation (PCM), or a very complex technique, e.g., one that uses subband coding and/or predictive coding to decrease bandwidth requirements...Many codecs have been standardized and are well known to those skilled in the art, and the particular codec selected is not critical to the operation of the invention.”). And as noted before, the ’525 patent’s encoder is simply a well-known prior art H.323 codec. *See, e.g.*, ’525 patent (Ex. 1001) at 2:58-63, FIG. 2 (“The H.323 protocol stack consists of an audio codec 214, and a video CoDec 213 connected to the audio/video input and output blocks 452 and 451, respectively. The audio and video CoDecs are responsible for compression and decompression of the audio and video sources.”).

460. Thus, Drell in combination with Knappe teaches that the near conference endpoint 100 utilizes an audio compression algorithm or protocol to compress the mixed audio data, which renders obvious “*compressing the mixed audio data*,” as claimed.

***[9.10] transmitting the compressed mixed audio data to the telephone; and***

461. The prior art renders obvious transmitting the compressed mixed audio data to the telephone.

462. **First**, as discussed at [9.9], Drell in combination with Knappe teaches generating compressed mixed audio data. Further, as discussed at [9.1], Drell in combination with Knappe teaches a PSTN telephone conference endpoint.

463. **Second**, Drell teaches that the near conference endpoint 100 is configured to transmit audio data over the Internet to endpoints on PSTN networks:

processing the received audio and video signals;  
combining the processed audio and video signals with the local audio and video signals; and  
transmitting the combined audio and video signals to each of the plurality of remote conference endpoints.

Drell (Ex. 1008), Claim 1.

Network 106 may be of any type suitable for the transmission of audio and video data between and among near conference endpoint 100 and remote conference endpoints 102 and 104....network106 may comprise a packet switched network, such as the Internet. Although

a single network 106 is shown, **the invention contemplates the use of two or more networks (for example, the PSTN and the Internet) to connect conference endpoints** utilizing different communication protocols.

*Id.* at 3:20-38.

The outputs of the audio mixer and video switching module are sent to each of the plurality of signal processing trains, which process the combined signals according to a **transmit mode for distribution to the remote endpoints over the network.**

*Id.* at Abstract.

**In the transmission mode...** processing train 302 is configured to receive the audio...streams output by audio mixing module 308...The received data streams are then encoded and **combined to form a mixed encoded audio/video data stream, and the encoded audio/video data stream is transmitted.**

*Id.* at 5:48-54. As discussed at [9.9], Drell's encoding step includes performing audio compression, and therefore it would have been obvious to a POSITA that Drell's "encoded audio/video data stream" includes compressed mixed audio data.

464. Thus, it would have been obvious to a POSITA to "*transmit[]*" the compressed mixed audio data to the PSTN telephone conference endpoint, as taught by Drell and Knappe.

465. **Alternatively**, as discussed at [9.1], it would have been obvious to a POSITA to utilize a gateway, as Knappe teaches, in Drell's system. Specifically, Knappe teaches utilizing a gateway positioned between an Internet network and a PSTN network for converting data into a suitable format for each network:

**The gateway 20 provides conversion between analog and/or digital formats.** The standard analog telephones 18 communicate standard telephony signals through PSTN 22 to the gateway 20. At the gateway 20, the signals are converted to IP packets in the VoIP format. Similarly, VoIP packets received from the network 14 are converted into standard telephony signals for delivery to the destination telephone 18 through PSTN 22.

Knappe (Ex. 1006) at 4:18-25, FIG. 1.

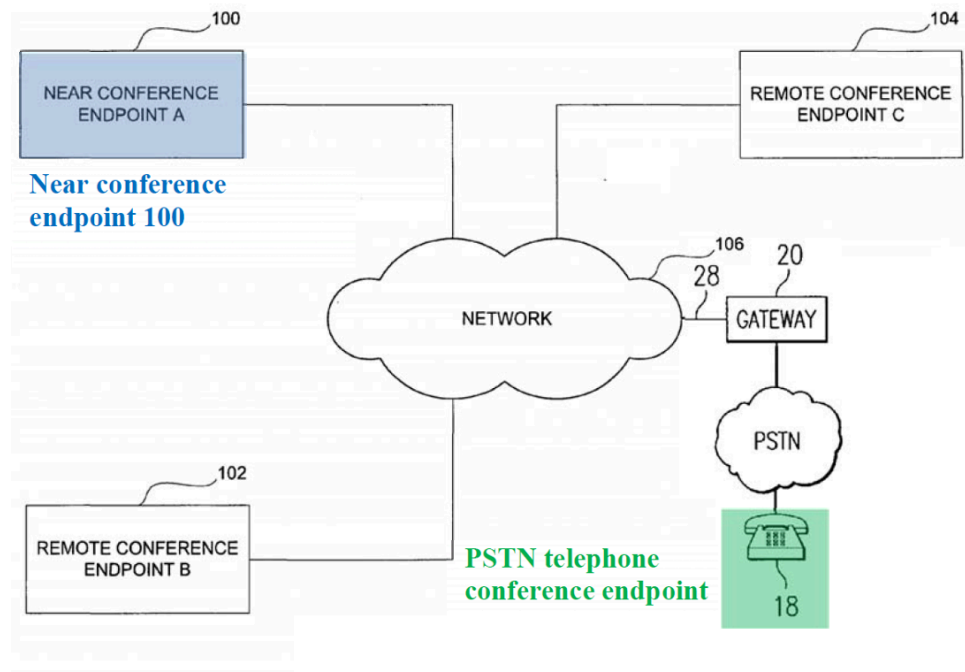
In accordance with a particular embodiment, **network 14 is the Internet**, a wide area network (WAN), a local area network (LAN) or other suitable packet-switched network. In the Internet embodiment, the network 14 transmits information in Internet Protocol (IP) packets. Telephony voice information is transmitted in the Voice over IP (VoIP) format.

*Id.* at 3:50-56. This is illustrated in Knappe's Figure 1, where gateway 20 is positioned between Internet network 14 and PSTN network 22 to perform conversion between analog and/or digital formats for each network.

466. Since Drell contemplates that its conference endpoints communicate over "PSTN and the Internet" networks (Drell (Ex. 1008) at 3:20-38), it would have been obvious to a POSITA to utilize a gateway positioned between the PSTN and Internet networks, as Knappe teaches, because it would provide Drell the benefit of converting Internet network communications to suitable signals for the PSTN, and vice-versa (Knappe (Ex. 1006) at 4:18-25). *See also* Reasons to Combine Drell and Knappe, *supra* ¶¶ 148-168.

467. As shown in the annotated figure below (combining portions of Drell's Figure 1 and Knappe's Figure 1), the gateway 20 has an output that couples

it to a PSTN telephone conference endpoint. It would have been obvious to a POSITA that the gateway would decompress the compressed mixed audio data and transmit a corresponding signal to the PSTN telephone conference endpoint so that the conference participant using the PSTN telephone conference endpoint could hear the audio portion of the conference (that is, the speech of other conference participants).



Drell (Ex. 1008) at FIG. 1 (annotated).

468. Thus, Drell in combination with Knappe teaches that the near conference endpoint 100 is further configured to transmit the compressed mixed audio data to the PSTN telephone conference endpoint, which renders obvious “transmitting the compressed mixed audio data to the telephone,” as claimed.

***[9.11] transmitting collaboration data to the first remote computer and the second remote computer.***

469. The prior art renders obvious transmitting collaboration data to the first remote computer and the second remote computer.

470. **First**, as discussed at [9.2]-[9.3], Drell teaches remote conference endpoint 102, which renders obvious “*the first remote computer*,” and Drell teaches remote conference endpoint 104, which renders obvious “*the second remote computer*.”

471. **Second**, Drell teaches that MP conferencing application 234 has a processing train 302 that utilizes a data codec to encode “still images” for transmission to conference endpoints:

**It is noted that, while not depicted in the Figures, processing train 302 may include a data codec for coding and encoding still images and the like received from and transmitted to remote conference**

Drell (Ex. 1008) at 6:53-56.

The MP conference application is configured to generate, for each remote conference endpoint participating in a conference, discrete instances of a signal processing train by means of dynamically allocable IMUXs, each processing train including a communication process and audio/video/**data codecs**.

*Id.* at Abstract.

472. A POSITA would have understood that Drell’s “still images,” which are encoded with a “data codec,” correspond to the claimed “*collaboration data*.”

*See, e.g.*, T.120 Primer (Ex. 1016) at 10 (“Still Image Exchange and Annotation

(SI – T.126”); T.120 Standard (Ex. 1017) at 2 (“This Recommendation makes provisions to ensure interoperability of commonly required functionality such as ... still image exchange ... through the definition of standardized application protocols.”). Thus, it would have been obvious to a POSITA that Drell’s “still images” transmitted to the remote conference endpoints, which would include the first remote conference endpoint 102 and the second remote conference endpoint 104, renders obvious “*transmitting collaboration data to the first remote computer and the second remote computer.*”

473. **Additionally**, Drell teaches that its “communication process...may for example comprise an...H.323 (packet-based) process.” Drell (Ex. 1008) at 2:26-33. A POSITA would understand that employing an H.323 process also discloses “*collaboration data*,” because H.323 was a well-known standard that recommended T.120 collaboration data. *See, e.g.*, H.323 Standard (Ex. 1033) at 15 (“The standardized data application for real-time audiographics conferencing is ITU-T T.120.”) & 18 (“ITU-T Rec. T.120 is the default basis of data interoperability between an H.323 terminal and other H.323, H.324, H.320, or H.310 terminals.”); T.120 Primer (Ex. 1016) at 13-15; T.120 Standard (Ex. 1017). This is consistent with the ’525 patent, which also relies on the prior art H.323/T.120 standards for transmitting collaboration data as part of the packetizing process. *See, e.g.*, ’525 patent (Ex. 1001) at 3:12-14 (“[T]he **H.323**

recommendation defines a **data collaboration** capability as known and outlined in the **T.120 data** collaboration unit.”), FIG. 2 (Prior Art).

474. Thus, Drell teaches that the near conference endpoint 100 transmits data using the H.323 and T.120 collaboration standards to the first remote conference endpoint 102 and the second remote conference endpoint 104, which renders obvious “*transmitting collaboration data to the first remote computer and the second remote computer,*” as claimed.

475. **Furthermore**, Elliott teaches collaboration data, such as “application sharing” data, “web browsing” data, and also collaboration data per the “T.120 standard”:

Multi-media collaboration services allow two or more parties to exchange information in a conference setting. These services can be readily provided over the Internet and include **collaborative Web browsing**, audio conferencing, video conferencing, and **application sharing**.

Elliott (Ex. 1007) at 1:40-44.

Session Server 12 is preferably able to set up to conduct Video conferences, audio conferences, and data conferences with application sharing. Collaborative Web browsing can also be enabled with URL push technology. During the course of a collaborative session, one party can **push a URL to another party**. In other words, one party can cause the browser of a remote party to connect to a certain web site.

*Id* at 5:54-61.

URLs are but one example of media, and URL push is but one example of a media-sharing feature provided by the present invention.

The URLs which are pushed can be many forms, for example, any or all of **data, graphics, video, and audio can be pushed.**

*Id.* at 3:50-54.

Once a session is established, multi-media collaboration between the agent and client is enabled as illustrated in Step 208. For example, the Java applet running on the client computer 20 will monitor the client computer's IP socket connection for incoming URLs. When a URL is pushed by an agent or another client, the applet will feed that URL to the client's Web browser, which will then go to the Web site addressed by the URL. **Other types of collaboration such as...data conferencing can also be performed.**

*Id.* at 6:46-54.

**The T.120 standard** contains a series of communication and application protocols and services that provide support for real-time, multi-point data communications. Session server 12 is preferably able to set up to conduct video conferences, audio conferences, and **data conferences with application sharing. Collaborative Web browsing** can also be enabled with URL push technology. During the course of a collaborative session, one party can push a URL to another party. In other words, one party can cause the browser of a remote party to connect to a certain web site.

*Id.* at 5:49-61.

The present invention can also be utilized in other applications. For example, during a session URLs can be pushed for collaborative Web browsing. In addition, since a multimedia Session Server 12 is used, other types of data can be shared. First a client computer 20 will be logged onto the Session Server 12 and an IP socket connection established will be established. From that point, the session may proceed with collaborative browsing, audio conferencing, video conferencing, or data conferencing. The URL push described above is simply one example of media sharing.

*Id.* at 9:7-16.

...multi-media collaboration comprises **application sharing.**

*Id.* at 11:12-13.

...multi-media collaboration comprises **collaborative web browsing**.

*Id.* at 11:6-7.

476. Elliott’s teaching of exchanging or pushing collaboration data, such as application sharing, web browsing, and other collaboration data per the T.120 standard additionally discloses “*transmitting collaboration data*,” as claimed.

477. A POSITA would understand that Drell’s processing trains 302 and 304, which include a “data codec” (Drell (Ex. 1008) at 6:53-54), could transmit collaboration data to remote conference endpoints 102 and 104. This is because transmitting collaboration data would allow different ways for the participants to collaborate thereby enhancing the effectiveness of the collaboration among the participants. *See* Reasons to Combine Drell and Elliott, *supra* ¶¶ 303-317.

478. Thus, Drell in combination with Knappe and Elliott teaches processing trains 302 and 304 at near conference endpoint 100 transmit various types of collaboration data to remote conference endpoints 102 and 104, which renders obvious “*transmitting collaboration data to the first remote computer and the second remote computer*,” as claimed.

## **2. Claim 10**

***[10.P] The process of claim 9 further comprising:***

479. See analysis for [9.P]-[9.11]

***[10.1] mixing the audio data received from the telephone with the audio data received from the first and second remote computers into a second mixed audio data.***

480. The prior art renders obvious mixing the audio data received from the telephone with the audio data received from the first and second remote computers into a second mixed audio data.

481. **First**, as discussed at [9.1], Drell in combination with Knappe teaches that the near conference endpoint 100 receives audio data from a PSTN telephone conference endpoint. Also, as discussed at [9.2]-[9.3], Drell teaches that the near conference endpoint 100 receives audio data from the first remote conference endpoint 102 and the second remote conference endpoint 104. Further, as discussed at [9.6] and [9.8], Drell teaches that the audio mixing module 308 of the MP conferencing application 234 at the near conference endpoint 100 mixes the audio data received from each of the conference endpoints

482. **Second**, it would have been obvious to a POSITA to mix together the audio data received from the PSTN telephone endpoint with the audio data from remote conference endpoints 102 and 104 because Drell's audio mixing module 308 mixes the audio data received from each conference endpoint. Drell (Ex. 1008) at 5:8-11. Drell also teaches playing back the speech of all remote conference participants to the participant using near conference endpoint 100:

Reference is now directed to FIG. 2, which depicts in block form various components of near conference endpoint 100...

loudspeaker 210 present... speech of the remote conference participants.

*Id.* at 3:39-48; *see also id.* at 5:23-25 (“**The output audio data stream is directed to... output audio path[]... 380.**”).

483. As discussed above regarding claim 9, the remote conference participants include the users of the PSTN telephone conference endpoint (*see* [9.1]) and remote conference endpoints 102 and 104 (*see* [9.2]-[9.3]). In view of the above disclosure, it would have been obvious to a POSITA that the audio mixing module 308 would combine the audio data from those endpoints and provide the mixed audio data stream to the speaker 201 so that the near conference endpoint 100 can present the “speech of the remote conference participants.” *Id.* at 3:39-48. Thus, the mixed audio data stream provided to speaker 201 is “*second mixed audio data*” as claimed.

484. Thus, Drell teaches that the MP conferencing application 234 includes an audio mixing module 308 that mixes audio data from the PSTN telephone conference endpoint with the audio data received from the first remote conference endpoint 102 and the second remote conference endpoint 104 into a second mixed audio data, which renders obvious “*mixing the audio data received from the telephone with the audio data received from the first and second remote computers into a second mixed audio data,*” as claimed.

### 3. Claim 11

*[11.P] The process of claim 9 further comprising:*

485. See analysis for [9.P]-[9.11]

*[11.1.1] wherein the receiving audio data from the telephone includes receiving internet protocol (IP) packets,*

486. The prior art renders obvious receiving the audio data from the telephone includes receiving Internet Protocol (IP) packets.

487. **First**, as discussed at [9.1], Drell in combination with Knappe teaches that the moderator near conference endpoint 100 receives audio data from the PSTN telephone conference endpoint, via a gateway.

488. **Second**, Drell teaches that the audio traverses two or more networks comprising an Internet network and a PSTN network:

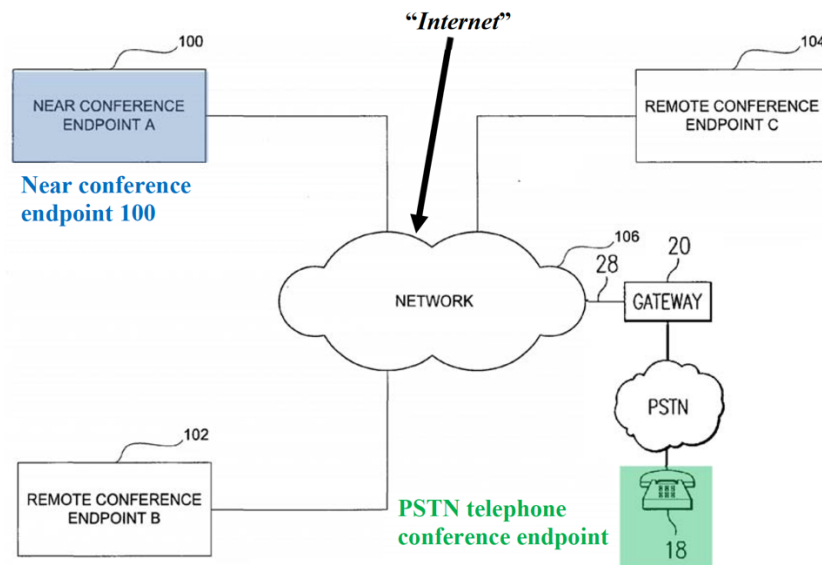
**[N]etwork 106** may comprise a packet switched network, such as the **Internet**. Although a single network 106 is shown, the invention contemplates the use of two or more networks (for example, the **PSTN and the Internet**) to connect conference endpoints utilizing different communication protocols.

*Id.* at 3:33-38.

489. Thus, Drell teaches that the moderator near conference endpoint 100 receives the audio data from the PSTN telephone conferencing endpoint over the Internet network 106. It would have been obvious to a POSITA that the audio data is received as “IP packets.” See Knappe (Ex. 1006) at 3:50-55 (“In the Internet embodiment, the network... transmits information in **Internet Protocol (IP)**”).

packets.”); *see also id.* at 4:18-25. It would also have been obvious to a POSITA to transmit data in IP packets over Drell’s Internet network, because communications over the Internet use IP packets. *See also* Reasons to Combine Drell and Knappe, *supra* ¶¶ 148-168.

490. As already explained at [9.1], it would have been obvious to a POSITA that Drell’s PSTN telephone conference endpoint would transmit audio data via gateway to the near conference endpoint 100:



Drell (Ex. 1008) at FIG. 1 (annotated).

491. It would have been obvious to a POSITA to utilize a gateway (as Knappe teaches) in Drell’s system, because it would allow for converting audio signals from Drell’s PSTN telephone conference endpoint into IP packets for transmission over the Internet network 106 to the near conference endpoint 100. That is, by utilizing a gateway between Drell’s PSTN and Internet networks,

endpoints on each network can beneficially communicate audio data to another endpoint on a different network through the gateway. Such a gateway would convert standard telephony signals to IP packets for transmission over the Internet network and convert IP packets to standard telephony signals for transmission over the PSTN network. *See also* Reasons to Combine Drell and Knappe, *supra* ¶¶ 148-168.

492. In the combination, Drell's PSTN telephone conference endpoint sends audio signals over the PSTN to a gateway which converts the signals into IP packets (per Knappe's teachings). The packets are then transmitted over Drell's Internet network 106 to the near conference endpoint 100, which receives the IP packets containing audio data from the PSTN telephone conference endpoint.

493. **Alternatively**, Drell teaches that its communication process includes a H.323 (packet-based) process:

The (MP) conferencing application is operable to process the plural signal streams received through the communication ports. Generally, the MP conferencing application generates separate processing trains for **signal streams from/to each of the remote conference endpoints**. The processing trains each comprise a communication process and a set of codecs...**The communication process...may for example comprise an...H.323 (packet-based) process...**The audio and video signals are...respectively delivered to audio and video codecs for decoding.

Drell (Ex. 1008) at 2:17-33.

494. It would have been obvious to a POSITA that Drell, which can perform H.323 processing, would use RTP IP packets described in H.323, which renders obvious receiving “*Internet Protocol (IP) packets*,” as claimed. *See, e.g.*, H.323 Standard (Ex. 1033), FIG. 13 at 39, at 146 (“IP-based H.323 endpoints”), at 189 (“H.323 and H.225.0 procedures using RTP/UDP/IP to convey the media streams”); U.S. Patent No. 6,553,515 (Ex. 1025) at 4:35-56; ’525 patent (Ex. 1001) at 2:58-67, FIG. 2.

495. Thus, Drell in combination with Knappe teaches that receiving audio data from the PSTN telephone conference endpoint includes receiving IP packets, which renders obvious “*receiving audio data from the telephone includes receiving internet protocol (IP) packets*,” as claimed.

***[11.1.2] the process further comprising decoding the IP packets.***

496. The prior art renders obvious decoding the IP packets.

497. **First**, as discussed at [11.1.1], Drell in combination with Knappe teaches that the near conference endpoint 100 receives IP packets including the audio data from the PSTN telephone conference endpoint.

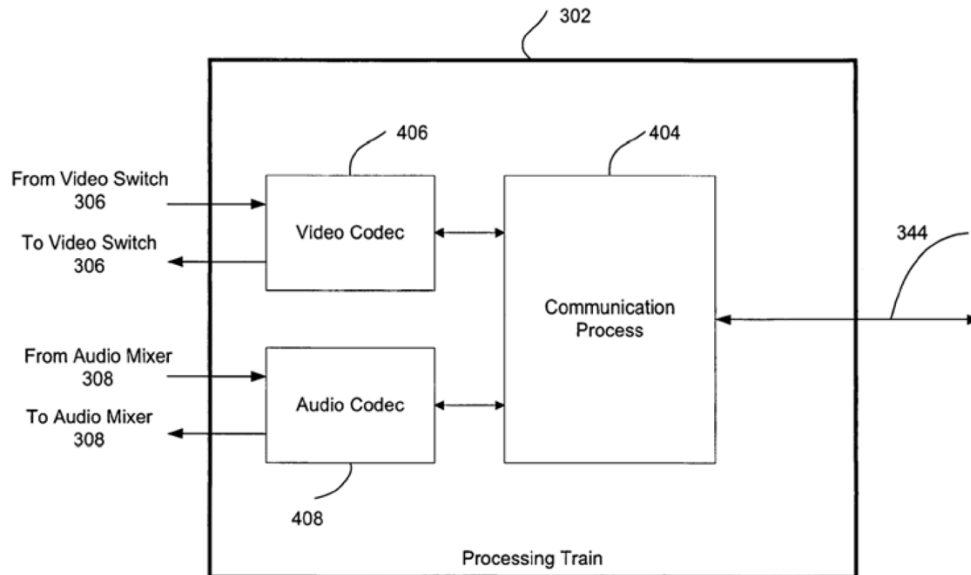
498. **Second**, as discussed at [9.6], Drell teaches instantiating a communication process 404 for each conference endpoint. Drell (Ex. 1008) at 4:21-24; 5:65- 66, FIG. 4. Drell further teaches that each communication process 404 is included in a “processing train... operative to **extract and decode audio**

and video data from signals received from remote conference endpoint.” *Id.* at 4:62-64; 5:2-4. Because the “signals received” from the PSTN telephone conference endpoint (via the gateway, see [9.1]) are in the form of IP packets, it would have been obvious to a POSITA that Drell’s processing train, including communication process 404 and audio codec 408, extracts and decodes audio from the IP packets. Relatedly, Knappe describes how “digital encoded data is encapsulated into IP or other suitable packets.” Knappe (Ex. 1006) at 4:11-13. Thus, it would have been obvious to a POSITA for Drell’s communication process to perform decapsulation of the received IP packets to recover the digital encoded data (e.g., audio data) that was within the IP packets. Drell further teaches that data from the conference endpoints are received and decoded by communication process 404 and audio codec 408 (using a standard or proprietary compression algorithm):

**Audio codec 408 receives the audio data stream from communication process 404 and applies...decoding in accordance with a standard (e.g., ITU Recommendation G.711) or proprietary audio compression algorithm.** The decoded audio data stream is then sent to audio mixing module 308, as described above.

Drell (Ex. 1008) at 6:18-23; *see also id* at 5:8-23, FIGs. 3, 4. As already explained, Drell’s “network 106 may comprise a **packet** switched network, such as the **Internet**,” and therefore audio data is received as IP packets. *Id.* at 3:33-38; *see* Knappe (Ex. 1006) at 3:50-55. As shown below at Figure 4, Drell utilizes

communication process 404 and audio codec 408 to decode received audio data, which is then sent to audio mixer 308:



Drell (Ex. 1008) at FIG. 4.

499. Accordingly, Drell’s disclosure of a communication process and an audio codec configured to receive and decode IP packets (which in the combination with Knappe) are received from the PSTN telephone conference endpoint via a gateway (as discussed above), renders obvious “*decoding the IP packets,*” as claimed.

500. **Alternatively**, Drell teaches that its communication process includes a H.323 (packet-based) process:

The (MP) conferencing application is operable to process the plural signal streams received through the communication ports. Generally, the MP conferencing application generates separate processing trains for **signal streams from/to each of the remote conference endpoints.**

The processing trains each comprise a communication **process and a set of codecs**...The communication process...may for example comprise an...**H.323 (packet-based) process**...The audio and video signals are...respectively delivered to audio and video codecs for decoding.

*Id.* at 2:17-33.

501. It would have been obvious to a POSITA that Drell, which can perform H.323 processing, would use codecs described in H.323 for decoding IP packets, which renders obvious a “*decoding the IP packets*,” as claimed. *See, e.g.* U.S. Patent No. 6,553,515 (Ex. 1025) at 4:35-56 (“**H.323** is adapted to support many aspects of multimedia...[and] generally is regarded as the **standard for VoIP**...This standard also specifies lowest-common denominator **compression protocols, such as G.711...G.723 and G.729**”). As noted previously, the ’525 patent’s decoder is simply a well-known prior art H.323 codec. *See, e.g.*, ’525 Patent (Ex. 1001) at 2:58-67, FIG. 2 (“The H.323 protocol stack consists of an audio codec 214, and a video CoDec 213 connected to the audio/video input and output blocks 452 and 451, respectively. The audio and video CoDecs are responsible for compression and decompression of the audio and video sources. The real-time network protocol component 215 is connected to the audio video CoDecs and is also responsible for preparation of the media data for transport according to the RTP (real-time protocol) recommendations.”).

502. **As a further alternative**, Knappe teaches that the conference endpoints utilize an audio codec to decode received IP packets:

**A codec (coder/decoder) converts audio, video...IP packets received** from the network 14 are converted back into analog signals.

Knappe (Ex. 1006) at 4:9-14.

503. Consistent with Knappe's teaching of a PSTN gateway to convert data between an analog format for the PSTN telephone conference endpoint and an IP format for the Internet (as discussed above), it would have been obvious to a POSITA to utilize in Drell's near conference endpoint 100 a corresponding audio codec configured to decode received IP packets from the PSTN gateway. *See also* Drell and Knappe, *supra* ¶¶ 148-168. Knappe's disclosure of decoding received IP packets also renders obvious that the process further comprises "decoding the IP packets," as claimed.

504. In the combination, Drell's PSTN conference endpoint sends audio signals over the PSTN network to a gateway which converts the signals into IP packets (per Knappe's teachings). The packets are then transmitted over Drell's Internet network 106 to the near conference endpoint 100 which uses a codec (as Knappe teaches) to receive and decode the IP packets containing audio from the PSTN conference endpoint.

505. Thus, Drell in combination with Knappe teaches that the near conference endpoint 100 utilizes an audio codec to decode IP packets received

from the PSTN telephone conference endpoint, which renders obvious that the process further comprises “*decoding the IP packets*,” as claimed.

#### 4. Claim 12

***[12.P] The process of claim 9 further comprising:***

506. See analysis for [9.P]-[9.11]

***[12.1.1] wherein the receiving audio data and video data from the first remote computer includes receiving internet protocol (IP) packets,***

507. The prior art renders obvious receiving audio data and video data from the first remote computer includes receiving Internet Protocol (IP) packets.

508. **First**, as discussed at [9.2], Drell teaches that the near conference endpoint 100 receives audio data and video data from the first remote conference endpoint 102 through Internet network 106, which includes a packet-switched network.

509. **Second**, since the moderator near conference endpoint 100 receives the audio data from the first remote conference endpoint 102 over the Internet network 106, it would have been obvious to a POSITA that it is received as IP packets. See Knappe (Ex. 1006) at 3:50-55 (“In the Internet embodiment, the network...transmits information in **Internet Protocol (IP) packets**.”); see also *id.* at 4:18-25.

510. It would have been obvious to a POSITA to transmit data in IP packets over Drell’s Internet network, because communications over the Internet

use IP packets. *See also* Reasons to Combine Drell and Knappe, *supra* ¶¶ 148-168.

511. **Alternatively**, Drell teaches that its communication process includes a H.323 (packet-based) process:

The (MP) conferencing application is operable to process the plural signal streams received through the communication ports. Generally, the MP conferencing application generates separate processing trains for **signal streams from/to each of the remote conference endpoints**. The processing trains each comprise a communication process and a set of codecs...**The communication process...may for example comprise an...H.323 (packet-based) process...**The audio and video signals are...respectively delivered to audio and video codecs for decoding.

Drell (Ex. 1008) at 2:17-33.

512. It would have been obvious to a POSITA that Drell, which can perform H.323 processing, would use RTP IP packets described in H.323, which renders obvious receiving “*Internet Protocol (IP) packets*,” as claimed. *See, e.g.* H.323 Standard (Ex. 1033), FIG. 13 at 39, at 146 (“IP-based H.323 endpoints”), at 189 (“H.323 and H.225.0 procedures using RTP/UDP/IP to convey the media streams”); U.S. Patent No. 6,553,515 (Ex. 1025) at 4:35-56; ’525 patent (Ex. 1001) at 2:58-67, FIG. 2.

513. Thus, Drell in combination with Knappe teaches that receiving the audio and video data from the first remote conference endpoint 102 includes receiving Internet Protocol (IP) packets over the Internet network 106, which

renders obvious “*receiving audio data and video data from the first remote computer includes receiving internet protocol (IP) packets,*” as claimed.

***[12.1.2] the process further including decoding the IP packets.***

514. As discussed at [12.1.1], the near conference endpoint 100 receives audio data and video data from the first remote conference endpoint 102 in the form of IP packets. Further, for the reasons discussed at [11.1.2], the prior art teaches that the near conference endpoint 100 decodes the IP packets received from each conference endpoint.

**5. Claim 13**

***[13.P] The process of claim 9 further comprising:***

515. See analysis for [9.P]-[9.11]

***[13.1] wherein the transmitting the mixed audio-video data includes transmitting the mixed audio-video data through a Virtual Private Network (VPN).***

516. The prior art renders obvious transmitting the mixed audio-video data through a Virtual Private Network (VPN).

517. **First**, as discussed at [9.7], Drell in combination with Knappe teaches transmitting the mixed audio-video data to the remote conference endpoints 102 and 104 (first and second remote computers) over the Internet network 106.

518. **Second**, Elliott teaches a conferencing system that transmits audio, video, and collaboration data over Internet network:

Multi-media collaboration refers to the use of more than one media stream (e.g.: **voice, fax, data, video, etc.**) **used in collaboration** with more than one party.

Elliott (Ex. 1007) at 2:66-3:1.

Other types of **collaboration...data conferencing** can also be performed.

*Id.* at 6:53-54.

**As alluded to above, communications between the parties can be made over a variety of networks 18.** A network 18 is at least two (and preferably more) computers interconnected together so that communication between them is possible. In the illustrated embodiment, the **public Internet** is shown as an exemplary **Internet Protocol-based (IP-based) network 18 for multi-media collaboration.**

*Id.* at 4:6-12.

519. Further, Elliott teaches transmitting data over Internet 18 using a VPN:

For example, network 18 can be either a public or a carrier supported virtual private network (VPN)...which uses IP or IP-tunneling.

*Id.* at 4:31-34.

520. Based on the above disclosure, a POSITA would have understood that Elliott utilizes a VPN for communicating conferencing voice, video, and collaboration data over a public network 18, such as the Internet. In view of Elliott's teachings, it would have been obvious to a POSITA to utilize a VPN when communicating audio, video, and collaboration data to and from the remote

conference endpoints 102 and 104, because it would provide Drell with secure communications as the communications traverse Internet network 106. *See also* Reasons to Combine Drell and Elliott, *supra* ¶¶ 303-317.

521. Thus, Drell in combination with Knappe and Elliott teaches that transmitting the mixed audio-video data to the remote conference endpoints 102 and 104 includes transmitting the mixed audio-video data through a VPN, which renders obvious “*transmitting the mixed audio-video data includes transmitting the mixed audio-video data through a Virtual Private Network (VPN)*,” as claimed.

## **6. Claim 14**

***[14.P] The process of claim 9 further comprising:***

522. See analysis for [9.P]-[9.11]

***[14.1] wherein the transmitting the mixed audio data includes transmitting the mixed audio data through a Virtual Private Network (VPN).***

523. The prior art renders obvious transmitting the mixed audio data through a Virtual Private Network (VPN).

524. **First**, as discussed at [9.10], Drell in combination with Knappe teaches transmitting the mixed audio data to the PSTN telephone conference endpoint over the Internet network 106 via a gateway.

525. **Second**, as discussed at [13.1], Elliott teaches that the conferencing system transmits audio data over the Internet through a VPN. *See, e.g.*, Elliott (Ex. 1007) at 2:66-3:1, 4:6-34. It would have been obvious to a POSITA to utilize a

VPN when communicating audio data to the PSTN telephone conference endpoint via a gateway (as taught by Knappe), because it would provide Drell with secure communications as the communications traverse Internet network 106. *See also* Reasons to Combine Drell and Elliott, *supra* ¶¶ 303-317.

526. Thus, Drell in combination with Knappe and Elliott teaches that transmitting the mixed audio data to the PSTN telephone conference endpoint includes transmitting the mixed audio data through a VPN, which renders obvious “*transmitting the mixed audio data includes transmitting the mixed audio data through a Virtual Private Network (VPN)*,” as claimed.

## **7. Claim 15**

***[15.P] The process of claim 9 further comprising:***

527. See analysis for [9.P]-[9.11]

***[15.1] encoding the mixed audio-video data before the transmitting of the mixed audio-video data to the first and second remote computers.***

528. The prior art renders obvious encoding the mixed audio-video data before the transmitting of the mixed audio-video data to the first and second remote computers.

529. **First**, as discussed at [9.6] and [9.7], Drell in combination with Knappe teaches producing and transmitting the mixed audio-video data to the first remote conference endpoint 102 and the second remote conference endpoint 104 via Internet network 106.

530. **Second**, Knappe supplements Drell and teaches encapsulating audio and video data into Real-time Transport Protocol (RTP) packets for transmission over the Internet:

[N]etwork 14 is the Internet...In the Internet embodiment, the network 14 transmits information in **Internet Protocol (IP) packets**. Telephony voice information is transmitted in the Voice over IP (VoIP) format. **Real-time IP packets such as VoIP packets are encapsulated in real-time transport protocol (RTP) packets for transmission over the network 14.**

Knappe (Ex. 1006) at 3:50-58.

A codec (coder/decoder) converts **audio, video** or other suitable signals generated by users from analog signals into digital form. **The digital encoded data is encapsulated into IP or other suitable packets** for transmission over the network 14.

*Id.* at 4:9-12.

531. Once Drell generates the mixed audio-video data (*see* [9.6]), it would have been obvious to a POSITA to encapsulate the mixed audio-video data in IP packets for transmission over the Internet because communications over the Internet use IP packets. *See id.* at 3:50-55, 4:9-13. It would also have been obvious to a POSITA to encapsulate the mixed audio-video data in RTP packets for transmission over the Internet (as Knappe teaches), because RTP was specifically created for applications such as Drell's real-time conferencing application. *See* Drell (Ex. 1008) at 3:20-38 ("Network 106 may be of any type suitable for the transmission of audio and video data between and among near

conference endpoint 100 and remote conference endpoints... [N]etwork 106 may comprise a packet switched network, such as the Internet...to connect conference endpoints..”); *see also* Caputo (Ex. 1021) at 22 (“**The Real-Time Protocol (RTP)...was created to transport real-time traffic, such as video and audio,** over the Internet.”); *see also* Reasons to Combine Drell and Knappe, *supra* ¶¶ 148-168.

532. **Alternatively**, as discussed at [9.6]-[9.7], Drell’s communication process 404 generates mixed audio-video data for transmission over the Internet to the first remote conference endpoint 102 and the second remote conference endpoint 104. Drell (Ex. 1008) at 6:46-48, 3:20-37. Drell further explains that the “communication process...compris[es] an **H.323 (packet-based) process.**” *Id.* at 2:26-28. It would have been obvious to a POSITA that Drell’s H.323 process includes encoding the data in RTP packets before transmission to the first remote conference endpoint 102 and the second remote conference endpoint 104 (as claimed) because H.323 was a well-known standard for encoding and decoding RTP communication.

533. The prior art teachings provide the same encoder disclosed in the ’525 patent, with the prior art H.323 protocol codec that uses the RTP for transport. *See, e.g.,* ’525 patent (Ex. 1001) at 2:58-67).

534. Thus, Drell in combination with Knappe teaches encoding the mixed audio-video data in RTP packets before the transmitting of the mixed audio-video data to the first remote conference endpoint 102 and the second remote conference endpoint 104, which renders obvious, “*encoding the mixed audio-video data before the transmitting of the mixed audio-video data to the first and second remote computers,*” as claimed.

535. **Furthermore**, as discussed at [13.1], Elliott teaches that the conferencing system transmits audio data over the Internet through a VPN. *See, e.g.,* Elliott (Ex. 1007) at 2:66-3:1, 4:6-34.

536. A POSITA would have been familiar with VPN communications and would have known that transmitting data via a VPN includes encoding the data to be transmitted with an encryption algorithm. *See* U.S. Patent No. 6,980,526 (Ex. 1020) at 5:41-55 (“VPN refers to a network that is carried over public networks, but which is encrypted to make it secure...”). For example, Elliott describes a VPN employing “IP-tunneling”:

For example, network 18 can be either a public or a carrier supported **virtual private network (VPN)** or local area net work (LAN) which uses IP or **IP-tunneling**.

Elliott (Ex. 1007) at 4:30-33. One of the most popular IP tunneling standards was “IP Security (IPSec) encryption,” which is defined by RFC 2401:

Overlay VPN networks can be implemented with a number of switched WAN Layer 2 technologies, including X.25, Frame Relay, ATM,

or SMDS. In the last years, overlay **VPN networks also have been implemented with IP-over-IP tunneling**, both in private IP backbones and over the public Internet. The two most commonly used IP-over-IP tunneling methods are Generic Route Encapsulation (GRE) tunneling and **IP Security (IPSec) encryption**.

VPN Textbook (Ex. 1010) at 122.

This memo [RFC 2401] specifies the base architecture for IPsec compliant systems. The goal of the architecture is to provide various security services for traffic at the IP layer, in both the IPv4 and IPv6 environments. This document describes the goals of such systems, their components and how they fit together with each other and into the IP environment. It also describes the security services offered by the IPsec protocols, and how these services can be employed in the IP environment.

**IPsec is designed to provide** interoperable, high quality, **cryptographically-based security** for IPv4 and IPv6. The set of security services offered includes access control, connectionless integrity, data origin authentication, protection against replays (a form of partial sequence integrity), confidentiality (**encryption**), and limited traffic flow confidentiality. These services are provided at the IP layer, offering protection for IP and/or upper layer protocols.

RFC 2401 (Ex. 1031) at 3-4. Because VPNs and IP tunnels were designed and intended to provide cryptographically-based security, a POSITA would have read Elliott's discussion of VPN communications and IP-tunneling as referring to encryption, which is a form of data encoding. A POSITA, therefore, would have understood that transmitting the mixed audio and video data to the remote conference endpoints 102 and 104 over a VPN would include encrypting the mixed audio and video data before transmission.

537. Thus, Drell in combination with Knappe and Elliott teaches that transmitting the mixed audio-video data to the remote conference endpoints 102 and 104 includes encrypting the mixed audio-video data through a VPN prior to transmission, which renders obvious “*encoding the mixed audio-video data before the transmitting of the mixed audio-video data to the first and second remote computers,*” as claimed.

## **8. Claim 16**

***[16.P] The process of claim 9 further comprising:***

538. See analysis for [9.P]-[9.11]

***[16.1] encoding the compressed mixed audio data before the transmitting of the mixed audio data to the telephone.***

539. The prior art renders obvious encoding the compressed mixed audio data before the transmitting of the mixed audio data to the telephone.

540. **First**, as discussed at [9.9]-[9.10], Drell in combination with Knappe teaches that the near conference endpoint 100 utilizes an audio codec data with an audio compression algorithm or protocol to compress the mixed audio data before it is transmitted to the PSTN telephone conference endpoint over the Internet and PSTN.

541. **Second**, as discussed at [15.1], Drell and Knappe teach encapsulating mixed audio-video data in IP packets or in RTP packets for transmission over the Internet. For the same reasons, it would have been obvious to a POSITA to

encapsulate mixed audio data in IP packets or in RTP packets. *See also* Reasons to Combine Drell and Knappe, *supra* ¶¶ 148-168.

**542. Alternatively**, as discussed at [9.9]-[9.10], Drell’s communication process 404 generates compressed mixed audio data for transmission over the Internet to the PSTN telephone conference endpoint. Drell (Ex. 1008) at 6:46-48, 3:20-37. Drell further explains that the “communication process...compris[es] an **H.323 (packet-based) process.**” *Id.* at 2:26-28. It would have been obvious to a POSITA that Drell’s H.323 process includes encoding the data in RTP packets before transmission to the PSTN telephone conference endpoint (as claimed) because, as explained previously, H.323 was a well-known standard for encoding and decoding RTP communication and the prior art teachings provide the same encoder disclosed in the ’525 patent, with the prior art H.323 protocol codec that uses the RTP for transport. *See, e.g.*, ’525 patent (Ex. 1001) at 2:58-67.

543. Thus, Drell in combination with Knappe teaches encoding the compressed mixed audio data in RTP packets before the transmitting of the mixed audio data to the PSTN telephone conference endpoint, which renders obvious “*encoding the compressed mixed audio data before the transmitting of the mixed audio data to the telephone,*” as claimed.

**544. Additionally**, as discussed at [13.1] and [14.1], Elliott teaches that the conferencing system transmits audio data over the Internet through a VPN. *See,*

*e.g.*, Elliott (Ex. 1007) at 2:66-3:1, 4:6-34. As discussed at [14.1], it would have been obvious to a POSITA to secure the data transmitted to the PSTN telephone conference endpoint using a VPN. Further, as discussed at [15.1], a POSITA would have known that VPN communications involve encrypting data, which is a form of data encoding, before transmitting the data.

545. It would have been obvious to a POSITA that in a system employing both data compression and data encryption, the compression step is performed first, followed by the encryption step. This is because compressed data can be readily encrypted, but encrypted data is generally not compressible. A POSITA would have found it obvious, therefore, that when combining the teachings of Drell, Knappe, and Elliott, the audio data to be transmitted to the PSTN telephone conference endpoint would be compressed, then encrypted, then transmitted.

546. It would have been obvious to a POSITA that in a system employing both data compression and data encryption, the compression step is performed first, followed by the encryption step. This is because compressed data can be readily encrypted, but encrypted data is generally not compressible. A POSITA would have found it obvious, therefore, that when combining the teachings of Drell, Knappe, and Elliott, the audio data to be transmitted to the PSTN telephone conference endpoint would be compressed, then encrypted, then transmitted.

## 9. Claim 17

***[17.P] A computer-implemented multi-participant conference system utilizing a local moderator computer, the system comprising:***

547. The prior art teaches “a computer-implemented multi-participant conference system utilizing a local moderator computer” as required by the preamble of claim 17.

548. **First**, for the reasons discussed in connection with limitation [1.P], Drell discloses a multiparticipant conferencing system. *See supra* ¶¶ 174-179.

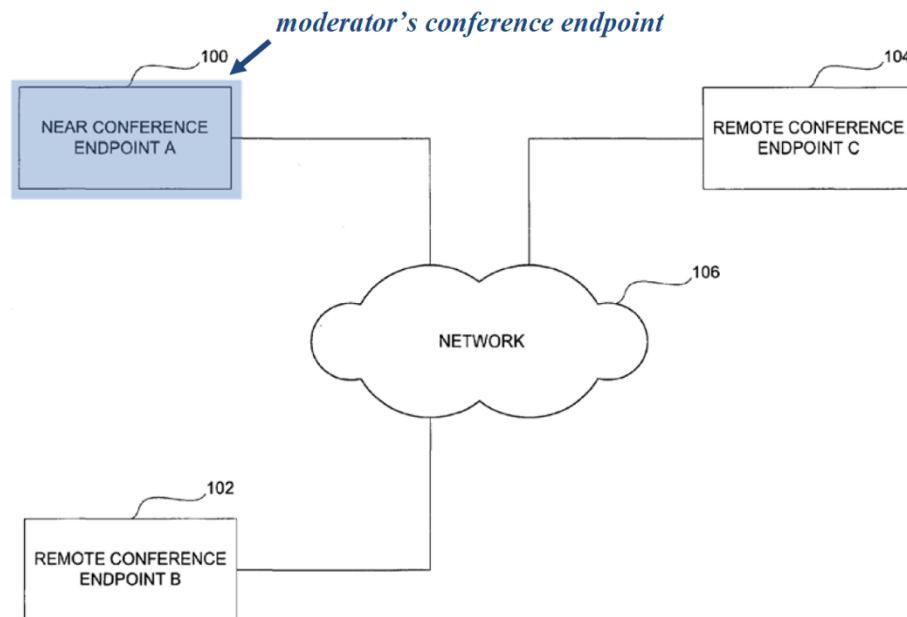
549. **Second**, for the reasons discussed in connection with limitation [9.P], Drell in combination with Knappe further renders obvious a multi-participant conference system that utilizing a local moderator computer. Specifically, as discussed in [9.P], Drell teaches that the near conference endpoint 100 includes a call manager application 232 and a multi-point (MP) conferencing application 234. Thus, Drell teaches that the near conference endpoint 100 is a “local computer.” *See supra* ¶¶ 380-383.

550. **In addition**, as discussed in connection with limitation [9.P], Knappe teaches that conference endpoints may be “personal or other suitable computers or computing devices.” Knappe (Ex. 1006) at 3:63-64. In view of Knappe’s teaching, a POSITA would understand that Drell’s near conference endpoint 100 is a “computer.” Knappe also teaches designating at least one of the conference participants as a moderator:

**[A]t least one participant may be given the priority of moderator for a particular conference call.** Typically, the moderator will receive the highest priority in a conference call. Therefore, in accordance with one embodiment, the priorities simply include moderator for one.

*Id.* at 8:52-56; *see supra* ¶¶ 386-387.

551. As discussed in [9.P], since Knappe teaches that at least one participant may be given the priority of a moderator, it would have been obvious to a POSITA to designate the user of Drell’s near conference endpoint 100 as the moderator. *See supra* ¶¶ 388-390. Accordingly, Drell in combination with Knappe teaches that the user of the near conference endpoint 100 is a “*moderator*” computer, as claimed:



Drell (Ex. 1008) at FIG. 1 (annotated).

552. Thus, Drell in combination with Knappe teaches “*a computer-implemented multi-participant conference system utilizing a local moderator computer*” as claimed.

***[17.1] a voice over IP decoder configured to decode audio data received by the local moderator computer from a telephone through a public switched telephone network (PSTN);***

553. The prior art teaches “a voice over IP decoder configured to decode audio data received by the local moderator computer from a telephone through a public switched telephone network (PSTN).”

554. **First**, for the reasons discussed in connection with [9.1], the combination of Drell and Knappe teaches that a local moderator (the moderator near conference endpoint 100 of Drell) receives audio data from a PSTN telephone conference endpoint through a PSTN. *See supra* ¶¶ 392-398.

555. **Second**, for the reasons discussed in connection with [3.1] and [11.1.2], Drell in combination with Knappe, renders obvious a voice over IP decoder configured to decode audio data received by the local moderator computer from a telephone through the PSTN. *See supra* ¶¶ 271-285, 501-508.

***[17.2] a voice over IP encoder configured to compress audio data transported from the local moderator computer to the telephone through the PSTN***

556. The prior art teaches “a voice over IP encoder configured to compress audio data transported from the local moderator computer to the telephone through the PSTN.”

557. **First**, for the reasons discussed in connection with [9.9], the combination of Drell and Knappe teaches compressing audio data. *See supra* ¶¶ 455-460.

558. **Second**, for the reasons discussed in connection with [9.10], the combination of Drell and Knappe teaches that the compressed audio data is transported from the local moderator computer to a telephone through the PSTN. *See supra* ¶¶ 461-468.

559. **Third**, for the reasons discussed in connection with [6.1], Drell in combination with Knappe teaches a VoIP encoder that uses a standard or proprietary compression algorithm. *See supra* ¶¶ 339-349.

560. Thus, Drell in combination with Knappe renders obvious a “a voice over IP encoder configured to compress audio data transported from the local moderator computer to the telephone through the PSTN.”

***[17.3] a first mixer configured to mix decoded audio data received by the local moderator computer from the telephone through the PSTN with audio received by the local moderator computer from a local audio input device and video data received by the local moderator computer from a local video input device into an audio-video stream for transmission through an Internet to a first remote computer and a second remote computer***

561. This limitation of claim 17 recites the same requirements as limitations [9.6] and [9.7] of claim 9 of the '525 patent. Therefore, the combination of Drell and Knappe renders obvious this limitation for the same reasons discussed in connection with limitations [9.6] and [9.7]. *See supra* ¶¶ 427-445

***[17.4] a second mixer configured to mixed audio data received by the local moderator computer from the first remote computer and the second remote computer through the Internet with audio data received by the local moderator computer form the local audio input device into an audio stream for transmission to the telephone through the PSTN; and***

562. This limitation of claim 17 recites the same requirements as limitations [9.8] and [9.9] of claim 9 of the '525 patent. Therefore, the combination of Drell and Knappe renders obvious this limitation for the same reasons discussed in connection with limitations [9.8] and [9.9]. *See supra* ¶¶ 446-460.

***[17.5] a data collaboration unit configured to transmit collaboration data from the local moderator computer through the Internet to the first remote computer and the second remote computer.***

563. This limitation of claim 17 recites the same requirements as limitation [9.11] of claim 9 of the '525 patent. Therefore, Drell and Knappe, in combination with Elliott renders obvious this limitation for the same reasons discussed in connection with limitations [9.11]. *See supra* ¶¶ 469-478. Specifically, Drell teaches that MP conferencing application 234 has a processing train 302 and 304

that a POSITA would understand acts as the “data collaboration unit” that transmits collaboration data as discussed in [9.1]. *Id.*

#### **10. Claim 18**

***[18.P] The system of claim 17 further comprising:***

564. See analysis for [17.P]-[17.5]

***[18.1] a third mixer configured to mix the decoded audio data received from the telephone with the audio data received from the first and second computers.***

565. This limitation of claim 18 recites the same requirements as [10.1].

Therefore, Drell and Knappe in combination with Elliott renders obvious this limitation for the same reasons discussed in connection with [10.1]. *See supra* ¶¶ 480-484.

#### **11. Claim 19**

***[19.P] The system of claim 17 further comprising:***

566. See analysis for [17.P]-[17.5]

***[19.1] wherein the audio data received by the local moderator computer from the telephone through the PSTN is transported through a virtual private network (VPN).***

567. The prior art renders obvious the local moderator computer’s receipt of audio data from the telephone through the PSTN through a Virtual Private Network (VPN).

568. **First**, as discussed at [9.1], Drell in combination with Knappe teaches the moderator near conference end point receiving audio data from the PSTN telephone over the Internet network 106 via a gateway.

569. **Second**, as discussed at [13.1], Elliott teaches that the conferencing system transmits audio data over the Internet through a VPN. *See, e.g.*, Elliott (Ex. 1007) at,2:66-3:1, 4:6-34. It would have been obvious to a POSITA to utilize a VPN when communicating audio data from the PSTN telephone conference endpoint via a gateway (as taught by Knappe), because it would provide Drell with secure communications as the communications traverse Internet network 106. *See also* Reasons to Combine Drell and Elliott, *supra* ¶¶ 303-317.

570. Thus, Drell in combination with Knappe and Elliott renders obvious a system where “the audio data received by the local moderator computer from the telephone through the PSTN is transported through a virtual private network (VPN)” as claimed.

## **12. Claim 20**

***[20.P] The system of claim 17 further comprising:***

571. See analysis for [17.P]-[17.5]

***[20.1] wherein audio data transported from the local moderator computer to the telephone is transmitted through a virtual private network (VPN).***

572. This limitation of claim 20 recites the same requirements as [14.1].

Therefore, Drell and Knappe in combination with Elliott renders obvious this

limitation for the same reasons discussed in connection with [14.1]. *See supra* ¶¶ 523-526.

## **X. SECONDARY CONSIDERATIONS**

573. Based on my experience and knowledge of this field, I am not aware of any secondary considerations that would provide any evidence of non-obviousness as to any of the claims in the '525 patent.

574. I am not aware of any allegations made by the named inventor of the '525 patent or any assignee of the '525 patent that any secondary considerations tend to rebut the obviousness of any of the claims of the '525 patent.

575. I reserve the right to respond to any evidence or arguments regarding secondary considerations of non-obviousness the patent owner may raise in its response.

## **XI. DECLARATION**

576. I hereby declare that, to the best of my knowledge and understanding, and based on the materials available to me at the time of signing this Declaration, all statements and opinions in this Declaration are true or believed to be true. I further declare that all statements made in this Declaration 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.

Executed this 14th day of December 2022.

By: Henry H. Houh.  
Henry H. Houh, Ph.D.