

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

VERIZON BUSINESS NETWORK SERVICES INC,

Petitioner,

- vs. -

HUAWEI TECHNOLOGIES CO., LTD.,

Patent Owner

IPR2020-01079
U.S. Patent 7,609,288

**DECLARATION OF SAMRAT BHATTACHARJEE, PH.D.
IN SUPPORT OF PETITION FOR *INTER*
PARTES REVIEW OF U.S. PATENT NO. 7,609,288**

TABLE OF CONTENTS

I.	INTRODUCTION AND QUALIFICATIONS.....	1
II.	UNDERSTANDING OF THE GOVERNING LAW	4
A.	Invalidity by Anticipation	4
B.	Invalidity by Obviousness.....	5
C.	Interpreting Claims Before the Patent Office	8
D.	Materials Relied on in Forming My Opinions.....	9
III.	THE PERSON OF ORDINARY SKILL IN THE ART OF THE '288 PATENT	10
IV.	BACKGROUND OF THE TECHNOLOGY.....	11
A.	Video, Digital Video, and Encoding of Digital Bit Streams	12
B.	Videoconferencing Systems, Architectures, and Standards	21
V.	OVERVIEW OF THE '288 PATENT	25
A.	The Specification of the '288 Patent.....	25
B.	The Prosecution History of the '288 Patent.....	33
C.	The Priority Date of the '288 Patent	37
VI.	STATE OF THE ART PRIOR TO THE '288 PATENT.....	37
A.	Becker.....	37
B.	Kenoyer '423	38
C.	Kenoyer.....	38
D.	Clapp.....	38
E.	Shaw.....	40
F.	<i>Video Compression Techniques</i> textbook.....	42

VII. CLAIM CONSTRUCTIONS	44
VIII. GROUNDS OF INVALIDITY	46
A. Claim 1 is obvious in view of Becker and Kenoyer '423.	46
1. A method for transferring desktop information of a PC to a video communication terminal comprising:.....	46
2. Capturing desktop information of the PC after receiving a triggering command, and converting a PC format of the desktop information into a format of a local video communication channel.....	48
3. Encoding the converted desktop format in a mode ensured by the local video communications terminal.....	51
4. Sending the coded bit stream to the local video communication terminal	51
5. Receiving the coded bit stream by the local video communication terminal and transferring it to a remote video communication terminal through transmission channel after processing.....	52
B. Dependent Claims 2, and 5-7 are obvious in view of Becker and Kenoyer '423.	54
C. Claim 9 is obvious in view of Becker and Kenoyer '423.	57
1. “An apparatus for transferring desktop information of a PC to a video communication terminal, comprises”	58
2. “A background processing device, capturing desktop information, converting the captured desktop information from a PC format to a format of a local video communication terminal”.....	58
3. “[a background device] encoding the converted format to a coded bit stream and outputting”	60

4.	“a terminal processing device, transferring the coded bit stream from the background processing device to a remote video communication terminal.”	62
D.	Dependent Claims 10, 12, and 14 are Obvious in view of Becker and Kenoyer ’423.	64
E.	Claim 1 is Obvious over Kenoyer in view of Clapp and <i>Video Compression Techniques</i> , and it is also obvious over Kenoyer in view of Clapp and Shaw.	74
1.	“A method for transferring desktop information of a PC to a video communication terminal comprising:”	75
2.	“Capturing desktop information of the PC after receiving a triggering command, and converting a PC format of the desktop information into a format of a local video communication channel”.....	76
3.	“Encoding the converted desktop format in a mode ensured by the local video communications terminal”.....	78
4.	“Sending the coded bit stream to the local video communication terminal”.....	88
5.	“Receiving the coded bit stream by the local video communication terminal and transferring it to a remote video communication terminal through transmission channel after processing”	90
F.	Claims 2 and 5-7 are also Obvious in view of Kenoyer and Clapp and further in view of Shaw and <i>Video Compression Techniques</i> . ..	93
G.	Claim 9 is obvious in view of Kenoyer and <i>Video Compression Techniques</i> or Kenoyer and Shaw.	101
1.	“An apparatus for transferring desktop information of a PC to a video communication terminal, comprises”	102
2.	“A background processing device, capturing desktop information, converting the captured desktop information from a PC format to a format of a local video communication terminal”.....	105

3.	“[a background device] encoding the converted format to a coded bit stream and outputting”	108
4.	“a terminal processing device, transferring the coded bit stream from the background processing device to a remote video communication terminal.”	109
H.	Dependent Claims 10, 12, and 14 are Obvious in view of Kenoyer in view of Clapp and <i>Video Compression Techniques</i> , or Kenoyer in View of Clapp and Shaw.....	112

I, Samrat Bhattacharjee, Ph.D., hereby declare as follows:

I. INTRODUCTION AND QUALIFICATIONS

1. I have been retained to provide my technical review, analysis, insights, and opinions concerning the validity of the claims of U.S. Patent No. 7,609,288 (EX1001; “the ’288 Patent”) entitled “Method and Apparatus of Transferring the Desktop of PC to Video Communication Terminal.” The patent is assigned to Huawei Technologies Co., Ltd of China (“Huawei”).

2. I am a tenured professor in the Computer Science Department at the University of Maryland. I am also an Affiliate Professor with the Department of Electrical and Computer Engineering at the University of Maryland. I have been an appointed faculty member at the University of Maryland since 1999.

3. I received a Ph.D. in computer science from the Georgia Institute of technology in 1999. Before that I received a B.S. degree in Mathematics and Computer Science with Highest Distinction (Summa Cum Laude) and was recognized as the Outstanding Department Major at Georgia College and State University, Milledgeville, Georgia in 1994.

4. I have been involved in the research and development of computing systems for over two decades. I perform research and I teach in the areas of computer networks, distributed systems, computer security and operating systems, among others. A major theme of my research has been the development of technology to

improve the performance and security of data transfers on the Internet. My research has examined problems ranging from network and operating system support audio and video streaming, large scale Internet video distribution, the design of novel network architectures for better and secure content distribution, efficiency of content distribution over wireless networks, and so on.

5. For example, starting in the late 1990s, the focus of my research was the development of network and operating system technology to design a new network architecture that allowed for more efficient data transfer. Part of my Ph.D. research was to develop new and more efficient architectures for video delivery on the Internet, and I have published papers on this architecture during my graduate studies. I continued to work on video delivery as a faculty member, and have published various papers on video streaming, content delivery architectures, and on resilient large scale content delivery. During 2007, I was a visiting researcher at AT&T Labs, and one of the projects I focused on was a video content delivery platform. This work resulted in both publications and a granted US patent (US8752100B2).

6. Along with working on research problems related to video distribution, I routinely taught this material in both graduate and undergraduate courses. A number of my papers in this domain are with students at Maryland, many of whom started in this area after taking my classes.

7. I have authored or co-authored over 100 articles in peer-reviewed journals, conference proceedings, and workshops in the aforementioned areas of computer science and others. Over the years I have received several fellowships, prizes, and awards for teaching excellence and technical papers, including an Alfred P. Sloan Jr. Fellowship in 2004 and the National Science Foundation CAREER award in 2001. I have also received an award from the SIGCOMM foundation for a "Test of Time" paper (that is given to an influential paper published ten years ago). My research work has been supported by multiple grants from the US National Science Foundation, and the Department of Defense. I have also started a Joint Ph.D. program with the University of Maryland and the Max Planck Society in Germany, and co-founded the annual Cornell, Maryland, Max Planck Research School that provides research exposure to about 80 students from across the world during a week long school.

8. I have served on numerous paper and proposal review panels and participated on program committees for ACM/IEEE groups, technical journals, and conferences in Networking, Security and Systems in the aforementioned areas of computer science. I have created research software for all of my projects, much of which is available online for others to build on.

9. I am a named inventor on three U.S. Patents. These patents are generally related to computer networking and the delivery of services over networks and video delivery systems.

10. I have served as an expert witness and technical consultant in litigation and *Inter Partes* Review matters concerning videoconferencing, data conferencing and screen sharing, voice over IP (VoIP) telephony, multimedia networking, distributed systems, operating systems, computer networks, and datacenter networking, among others. I have testified in several trials and depositions as an expert witness.

11. I attach my curriculum vitae, which includes a more detailed list of my qualifications. My C.V. also contains a list of all other cases in which, during the previous 4 years, I have been retained as part of the proceeding. My work on this case is being billed at a rate of \$600.00 per hour, with reimbursement for actual expenses. My compensation is not contingent upon the outcome of this *Inter Partes* Review.

II. UNDERSTANDING OF THE GOVERNING LAW

A. Invalidity by Anticipation

12. My understanding is that a patent claim is anticipated if a single patent, publication, or other item of prior art describes all elements of the claim, disclosed and arranged as described in the claim. I understand that such prior art could be

published anywhere in the world and includes patents and published patent applications.

B. Invalidity by Obviousness

13. I understand that obviousness is analyzed from the perspective of a person of ordinary skill in the art (a “POSA”) at the time the alleged invention was made. I also understand that a POSA is presumed to have been aware of all pertinent prior art at the time of the alleged invention.

14. I understand that an obviousness analysis involves comparing a claim to the prior art to determine whether the claimed invention as a whole would have been obvious to a POSA in view of the prior art, and in light of the general knowledge in the art at the time the invention was made. I also understand that the invention may be deemed obvious when a POSA would have reached the claimed invention through routine experimentation.

15. I understand that obviousness can be established by combining or modifying the disclosures of the prior art to achieve the claimed invention. It is also my understanding that where there is a reason to modify or combine the prior art to achieve the claimed invention, there must also be a reasonable expectation of success in so doing to render the claimed invention obvious. I understand that the reason to combine prior art references can come from a variety of sources, not just the prior art itself or the specific problem the patentee was trying to solve. I also understand

that the references themselves need not provide a specific hint or suggestion of the alteration needed to arrive at the claimed invention; the analysis may include recourse to logic, judgment, and common sense available to a POSA.

16. I understand that when there is some recognized reason to solve a problem, and there are a finite number of identified, predictable solutions, a POSA has good reason to pursue the known options within his or her technical grasp. If such an approach leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense. In such a circumstance, when a patent simply arranges old elements with each performing the same function it had been known to perform and yields no more than one would expect from such an arrangement, the combination is obvious.

17. I understand that when considering the obviousness of an invention, one should also consider whether there are any objective indicia that support the non-obviousness of the invention. I further understand that objective indicia of non-obviousness include failure of others, copying, unexpected results, information that “teaches away” from the claimed subject matter, perception in the industry, commercial success, and long-felt but unmet need. I also understand that in order for objective indicia of non-obviousness to be applicable, the indicia must have some sort of nexus to the subject matter in the claim that was not known in the art.

I understand that this nexus includes a factual connection between the patentable subject matter of the claim and the objective indicia alleged.

18. Finally, I understand that Patent Examiners at the U.S. Patent and Trademark Office (“USPTO”) rely upon certain exemplary rationales in reviewing patent applications to understand whether the subject matter of the claims is obvious. I understand that the following is the list of exemplary rationales relied upon by Patent Examiners at the USPTO:

- (A) Combining prior art elements according to known methods to yield predictable results;
- (B) Simple substitution of one known element for another to obtain predictable results;
- (C) Use of a known technique 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) “Obvious to try” – Choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success;
- (F) Known work in one field of endeavor may prompt variations of it 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) 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 teachings to arrive at the claimed invention.

C. Interpreting Claims Before the Patent Office

19. I understand that *Inter Partes* Review is a proceeding before the United States Patent & Trademark Office (“USPTO”) for evaluating the validity of issued patent claims. I understand that, in an *Inter Partes* Review, a claim term is to be interpreted in a manner consistent with the standard used in patent litigation, as set forth in *Phillips v. AWH Corp.*, 415 F.3d 1303 (Fed. Cir. 2005) (en banc). That standard generally construes the claims according to their “ordinary and customary” meaning in view of the claim language, specification, and file history, and where applicable, relevant other evidence.

20. As demonstrated below, the prior art cited herein discloses the claimed method and apparatus, and renders the claims as a whole obvious under the *Phillips* standard.

21. I understand that a patent’s “specification” includes all the figures, discussion, and claims within the patent. I understand that the USPTO will look to

the specification and prosecution history to see if there is a definition for a given claim term, and if not, will apply the ordinary and customary meaning from the perspective of a POSA at the time in which the alleged invention was made. I present a more detailed explanation of certain claim terms in the '288 Patent in Section VII (“CLAIM CONSTRUCTIONS”) below.

D. Materials Relied on in Forming My Opinions

22. In forming my opinions expressed in this declaration, I have relied on my own knowledge, experience, and expertise, as well as the knowledge of a POSA in the relevant timeframe. In addition, I have reviewed and relied upon the following list of materials, and any other cited reference, in this declaration:

- U.S. Patent No. 7,609,288 (“the '288 Patent”) (EX1001);
- Prosecution History of the '288 Patent (EX1002);
- U.S. Publication No. 20020154209A to *Kenoyer* (EX1005);
- U.S Patent No. 6,073,192 to Clapp (EX1006);
- U.S. Patent No. 5,611,038 to Shaw et al (EX1007);
- *Video Compression Techniques* book excerpt, pp. 58-65, published 1998, ISBN 3-920993-13-6 (EX1008);
- U.S. Patent Application Pub. No. 2002/0149617 A1 to Becker (EX1009);
- U.S. Patent No. 9,769,423 to Kenoyer (EX1010);
- *RTP: A Transport Protocol for Real-Time Applications*, RFC 1889, published 1996 (EX1011)

- U.S. Patent No. 6,141,062 to Hall et al (EX1012) and
- IEEE 100 Authoritative Dictionary of IEEE Standards Terms (Dec. 2000) definition of “bit stream.” EX1017.
- IEEE Standard Glossary of Computer Networking Terminology (June 30, 1995) “bit stream” listing. EX1018.

III. THE PERSON OF ORDINARY SKILL IN THE ART OF THE '288 PATENT

23. The claims of the '288 Patent are directed to apparatuses and methods for “transferring desktop information of a PC to a video communication terminal” such as for the purpose of videoconferencing. EX1001, Abstract, 1:24-29. As explained below, I understand the earliest possible effective filing date of the '288 Patent is August 7, 2002.

24. In my opinion, a person of ordinary skill in the art (a “POSA”) as of August 7, 2002, would have had a working knowledge of distributed systems generally, and videoconferencing and screen sharing systems specifically. This would include knowledge of compression and decompression techniques for audio, video, and image data, as well as communication protocols and the use of computer networks for transmitting streams of such data (either compressed or uncompressed). A POSA would have had a bachelor’s degree in computer science, computer engineering, or an equivalent degree, and one-two years of professional experience

with distributed multimedia systems. Lack of work experience could be remedied by additional education, and vice versa.

25. As of August 2002, I had a Ph.D. in computer science and had been working with distributed multimedia systems for several years. I also had taught the concepts behind these systems to computer science students for several years prior to 2002. For these reasons, I am able to assess the disclosures in the '288 Patent and the prior art from the perspective of a person of ordinary skill in the art as of August 2002.

IV. BACKGROUND OF THE TECHNOLOGY

26. The '288 Patent concerns transferring desktop information of a PC to a video communication terminal. The transferring occurs within the context of an on-going videoconference. *See, e.g.*, EX1001, 1:24-29. As of August 2002, which I understand is the earliest possible priority date for the '288 Patent, videoconferencing and desktop/screen sharing systems were well known in the art and the state of the art was quite mature. Commercial products and services providing these features, such as the well-known WebEx product, were readily available and could execute on commodity PCs.

27. To better understand the '288 Patent, the prior art, and my opinions on the validity of the claims of the '288 Patent, in this section I provide some background on videoconferencing and screen sharing as well as digital video and

computer video as a data type, with a particular focus on the concepts of digitization, compression, and media encoding.

28. All of the concepts discussed in this section were well known and widely used well prior to the August 2002. For example, I frequently taught these concepts to computer science students prior to 2002.

A. Video, Digital Video, and Encoding of Digital Bit Streams

29. Computer video has its origins in the standards for analog broadcast television. Originally, television was broadcast in black and white. A single signal that represented the brightness or intensity of each portion of the television screen was transmitted over the air or via a cable. Color television came into being by transmitting two additional signals (at different frequencies) that represented color information along with the original black and white signal. The original black and white signal represented the “luminance” (brightness) component of the video and was augmented with color signals representing two “chrominance” components. The chrominance components specified color information for each portion of the television screen, often referred to as “hue,” and “saturation.”

30. The luminance/chrominance representation of a video signal is commonly referred by the shorthand “YUV,” where Y refers to the luminance component and U and V refer to the two chrominance components.

31. Digital video started out as the process of digitizing the analog television signals. In the case of video, digitizing means that each of the three components of the broadcast television signal, the luminance and two chrominance signals, was sampled in time and quantized (represented as an integer with fixed precision). That is, each analog component was measured on a precisely periodic basis and the value of each measurement was represented as a binary number. How frequently the components of the television signal were sampled determined, in part, how faithfully the digital signal represented the original analog signal.

32. A separate issue was how precisely the value of each analog sample was represented as a binary number. The more binary digits (bits) one used to represent the value of the analog sample, the higher the precision of the representation. For example, if analog samples were represented as 8-bit binary numbers then one could only represent $2^8 = 256$ different values for the sample. Thus, if the values of a measured sample ranged from 0 to K units, with an 8-bit binary encoding of the sample you can only differentiate between samples whose values are at least $K/256$ units apart. The value $K/256$ represents the “step size” or quantization factor of the digitization process. That is, each measured sample of the analog signal will be represented as some number (from 0 to 255) of steps of size $K/256$. Samples whose values are less than a step size apart will (to a first approximation) be coded with the same 8-bit binary number and thus not be

distinguishable from one another after digitization. If 16-bit binary numbers were used to represent the values of analog samples then one could represent $2^{16} = 65,536$ different sample values and thus represent the sampled values with greater precision (i.e., smaller differences between sample values can be represented). Digitizing is also referred to as “quantization” as when you sample and digitize you represent a sample in terms of a discrete integer number of steps. Digitization is also referred to as “quantization,” such as when sampling and digitizing represents a sample in terms of a discrete integer number of steps. Combined, determining a sampling frequency (how many times per unit time a signal is measured) and a quantization level (the number of possible values used to represent the measured value of the signal) determines how accurately the digitized signal matches the analog signal.

33. When digitizing a signal, the analog signal is encoded because it is “coded” (in binary) as a discretized version of the signal — a series of binary numbers where each binary number represents an integer number of steps corresponding to the quantization of a sample of the analog signal at a point in time. In the case of a signal such as the luminance component of a television signal, further encoding will be necessary as part of the digitization process. For example, in addition to encoding values of the signal, one may need to further encode information such as the boundaries of video frames within the sequence of binary numbers produced in the digitization process. That is, one may need to encode

which binary numbers correspond to the first or last luminance sample in a video frame. This may be necessary so that a device that attempts to interpret (e.g., display) the digitized luminance values can determine what part of a video image each luminance value corresponds to.

34. In addition, encoding a digitized continuous signal, such as any of the luminance/chrominance components of analog television, results in a so-called “bit-stream” as the digitizer and/or encoder will be generating a sequence of digital data values (a sequence of bits) at a constant rate for a conceptually never-ending duration.

35. For digitization purposes, in North America, prior to the conversion of broadcast television signals to HDTV, a television display was considered to be comprised of a grid of picture elements (for example, 440 x 480). Thus, when digitizing a frame of video, each of the Y, U, and V components would be sampled, in my example, 440×480 times. Most commonly, each sample component would be represented as an 8-bit number. However, to reduce the data rate, it was common to subsample the UV chrominance components, meaning that UV values would only be generated for every k^{th} picture element, where $k = 4$ was a common value. This was done because humans have relatively poor visual acuity for color changes and using one pair of chrominance values to represent the color of 4 adjacent picture elements resulted in images that appeared indistinguishable to most viewers from

images where every picture element had its own chrominance values. Chrominance subsampling by a factor of 4 resulted in an representing a video image using 12-bits per picture element (8 bits for the luminance component, and 8-bits for each chrominance component for every 4th picture element or an average of 2-bits for each chrominance component for each picture element). This is one-half the number of bits per picture element that would be required if subsampling were not employed and each picture element was represented using 8-bits for luminance and 8-bits for each chrominance component (for a total of 24-bits per picture element).

36. The video generated by computers used both a different color system and a different display resolution from what was used in broadcast television. Computers generate video using the primary colors for projected light: red, green, and blue (“RGB”). Moreover, at the time of the ’288 Patent, computers typically displayed video at a resolution of 640 by 480 picture elements (VGA resolution).¹ Each picture element of the display was represented by three 8-bit values, one each for the R, G, and B components of the video resulting in a 24-bit per picture element representation.

¹ Note that computer video displayed at VGA resolution resulted in higher resolution images than those displayed at television (NTSC) resolution.

37. Applications such as videoconferencing primarily used video inputs from cameras that for much of the history of videoconferencing, generated video according to television standards. As a result, as explained in more detail below, the YUV format for video was the basis for many of the encoding and compression standards developed for videoconferencing. Thus, if one wanted to share images projected on a computer monitor in a conference, the computer-generated video was most commonly converted from the RGB color space to the YUV color space (using a simple mathematical transform). This typically had the advantage of a reduction in the data rate needed to communicate the video images as the YUV encoding typically required 12-bits per picture element whereas the RGB encoding required 24-bits. Thus, a conversion to a YUV color space resulted in a simple form of compression (through techniques such as chrominance subsampling).

38. Videoconferencing and other applications that transmit or store streams of digital audio and video typically further compress the streams. Algorithms for the compression of audio and video are complex but generally work by identifying and eliminating redundancy in the media streams. In the case of video, redundancy can come in the form of spatial redundancy or temporal redundancy. Spatial redundancy exists when adjacent picture elements within the same frame have the same or similar color or luminance. Removing spatial redundancy by, for example, using one set of luminance/chrominance values for multiple picture elements in a

frame, is referred to as intra-frame compression. Temporal redundancy exists when the luminance or color of a picture element is the same or similar in consecutive frames. Removing temporal redundancy by, for example, using luminance/chrominance values of picture elements from a previous frame to represent the values of picture elements in the current frame, is referred to as inter-frame compression. A more sophisticated form of inter-frame compression (temporal redundancy elimination) involves determining if a group of picture elements is vertically and/or horizontally translated between frames (i.e., if the luminance/chrominance values for a group of picture elements in a previous frame match the luminance/chrominance values for a difference group of picture elements in the current frame). This form of inter-frame compression is referred to as “motion prediction” or “motion compensation” and involves the computation of “motion vectors” that describe how groups of picture elements are translated vertically and/or horizontally between frames. With motion prediction, a motion vector is used to represent the luminance/chrominance values of picture elements in the current frame in terms of the values of (different) picture elements in a previous frame.

39. There has long been interest in compressing audio and video signals/streams for transmission over networks. Starting in the 1980s, audio and video compression standards were developed by organizations such as the ITU (the International Telecommunications Union) for use in storing and transmitting

audio/video streams. Some well-known examples of such standardized algorithms include the H.261 and H.263 schemes. These were configurable schemes but could operate to perform both intra-frame and inter-frame compression. H.261, for example, works on YUV data common for video images, and produces a bit stream useful for real-time transmission. Other schemes for compressing audio and video existed including the Motion JPEG scheme.

40. For example, an illustration of the intra-frame encoding/compression process in H.261 is shown below in a figure from *Video Compression Techniques* (EX1008) excerpt at page 60:

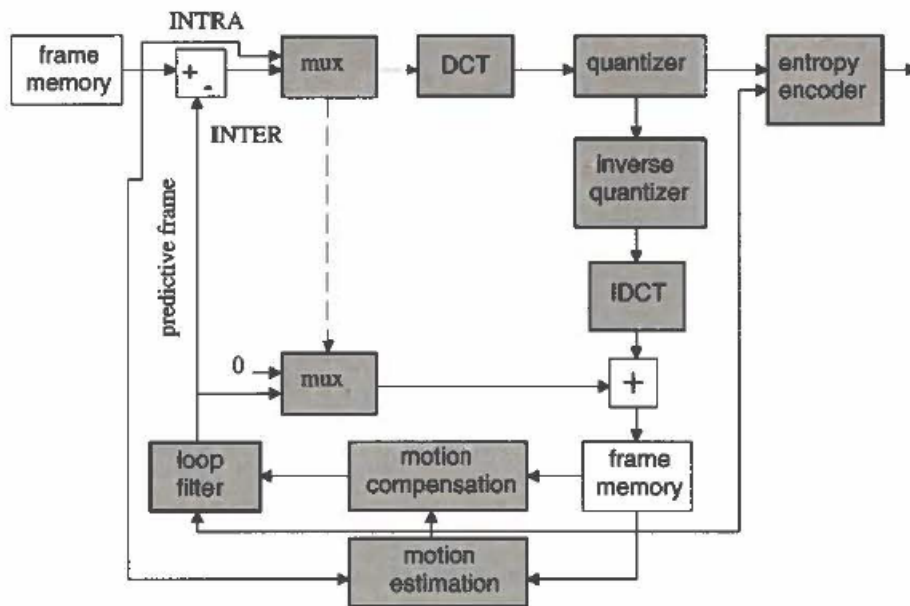


Figure 5-13: H.261 encoder

H.261 Encoder in *Video Compression Techniques* (EX1008)

41. All compression standards required a means of further encoding the compressed data so that an entity receiving or reading the compressed data can interpret the data. As discussed above, video, even in compressed form, is still a multidimensional signal and requires encoding so that it can be, for example, decompressed. The bit-stream has a hierarchical format wherein data for picture elements are combined into data for rectangular groups of picture elements (“macroblocks”) which are then combined into data for groups of block which are then combined into data for video images. The below table from *Video Compression Techniques* depicts the coding layers in the H.261 standard including the characteristics of the macroblocks:

Table 5-5: Coding layers of H.261

Picture	one video frame
Group of Blocks	33 macroblocks (1/12 CIF, 1/3 QCIF)
Macroblock	16 × 16 Y, 8 × 8 Cb/Cr , motion compensation
Block	8 × 8 pixels (coding unit for DCT)

Table 5-5 of Video Compression Techniques (EX1008)

42. Finally, the hardware or software that does some or all of the encoding of a media stream is referred to as a “codec” (“coder/decoder”). In addition to compressing and encoding, codecs also decode and decompress media streams.

43. Having standards for compression and encoding is important as the use of standards enable interoperation between codecs from different vendors and thereby enable differently constructed applications such as videoconferencing applications to interoperate.

B. Videoconferencing Systems, Architectures, and Standards

44. Technological advances in compression, such as the development of the H.261 and H.263 standards, and increased capacity in telecommunications circuits led to higher quality video (e.g., color and higher resolution video), and audio signals being acquired and transmitted.

45. The basic architecture of these systems was a distributed processing pipeline consisting of digitization of audio and video media, compression, encoding, and transmission of compressed, encoded media to a remote site where the data was decoded, decompressed, and displayed.

46. While all videoconferencing systems contained these processing steps, how these steps were realized varied. For example, digitization and display were typically functions implemented in hardware while compression/encoding and decoding/decompression were implemented in some combination of hardware and software (and by the late 1990s it was common for coding/decoding to be implemented in software). In addition, compression and transmission were sometimes performed by a special purpose device that was separate from the device

acquiring the audio and video signals. The corresponding scenario existed on the display side wherein a special purpose device separate from the display device received, decoded and decompressed media.

47. Beyond point-to-point conferencing, there was great interest in multi-point conferences. These were conferences wherein multiple sites transmit their audio and video to all other sites in the conference and thereby conferees see multiple video and audio streams.

48. Organizations such as the ITU promulgated standards for videoconferencing architectures to enable such multi-point conferencing. The H.323 standard, first promulgated in 1996, consists of a suite of protocols for setting up conferences, compressing and transmitting media, as well as an architecture for managing multipoint conferences. The architecture provided for videoconferencing in the H.323 standard, an architecture widely adopted by industry, consists of entities known as terminals, gateways, gatekeepers, and multipoint control units (MCUs). Terminals are end-systems that contain cameras and displays, microphones and speakers, audio and video codecs, and a network interface. Gateways and gatekeepers provide functions related to network management and the ability to support conferences between otherwise incompatible networks such as packet-based networks like the Internet and circuit-switched networks such as the traditional telephone network. MCUs are devices in the network that enable multipoint

conferences by receiving audio/video streams from conference participants and bridging (mixing/compositing) the multiple streams into a single stream which is then forwarded to participants. As a result, although cumulatively, multiple audio/video streams are generated in the conference, each participant only receives a single (composite) stream. In this manner, the network resources required to sustain a conference at each site (other than the site hosting an MCU) are limited to a single audio/video stream being sourced and received and thus is independent of the number of participants in the conference. MCUs require significant processing resources as they typically have to have the ability to receive, decode and decompress streams, mix with other streams, and compress, encode, and transmit the combined stream. RFC 1889 from the Internet Engineering Task Force, which promulgates Internet standards, shows multiplexing was recognized as a standard form of transmitting real-time data such as video data over the Internet at least by 1996. EX1011.

49. In summary, long before the 2002 priority data of the '288 Patent, videoconferencing systems and standards had recognized and satisfactorily addressed the need for end-systems to capture media from various sources and perform analog-to-digital conversion, color-space conversion, and a variety of encodings and compression to generate bit-streams for transmission to remote sites where the inverse operations are performed. Intermediate devices between

conference end-systems also existed and were standardized for a variety of purposes including additional media conversion, encoding/decoding, processing, and compression/decompression.

50. In addition to the transmission of audio and video streams, computer-based videoconferences often involved transmission of all or portions of a participant's computer screen. Termed "screen sharing," "window sharing," "application sharing," or "data conferencing," the ability for conference participants to view and possibly manipulate documents and other artifacts stored on one participant's computer was an adjunct to a traditional videoconference. Such screen sharing was a mature and pervasive technology well before the 2002 priority date of the '288 Patent. Indeed, part of the H.323 standard for videoconferencing included a suite of standards and protocols, called T.120, for data conferencing including screen/application sharing.

51. Virtual network computing (VNC) is an example of a system that created video streams from the video memory that were popular before the '288 Patent filing date and remain popular. *See* <https://ieeexplore.ieee.org/abstract/document/656066>.

52. Screen sharing ranged from simply transferring the contents of a computer's display as an image or as a video, to transferring images or video of individual windows or the windows associated with a specific application, to

transferring the graphics primitives required to render the images generated by an application. In addition, in the case of application sharing, screen sharing systems often provided the ability to share a virtual keyboard and mouse that could be used by remote participants to interact with a shared application (e.g., provide input to the shared application or control the shared application) as if that application were running on the remote participant's computer. Further, such systems also often provided the ability for conference participants to annotate shared images/applications by writing or drawing "on top of" the shared windows/displays using a virtual marker.

53. Thus, as of August 2002 (and even before 2002), the state of the art and practice in screen sharing was quite mature and more sophisticated than the disclosures of simply transferring the desktop information of a PC disclosed in the '288 Patent.

V. OVERVIEW OF THE '288 PATENT

A. The Specification of the '288 Patent

54. The '288 Patent describes a method and apparatus for "transferring the desktop of PC to video communications terminal." EX1001, Title. The '288 Patent background addresses "a video communication field, especially in a videoconferencing system" that needs to "transfer desktop information, such as films or file information, to the remote video communication terminal at the same time

while a local image such as an image taken by a local video camera is transferred” EX1001, Col. 1:23-27. Thus, the ’288 Patent discloses transferring desktop information within the context of an on-going videoconference. As discussed above, transferring desktop information within the context of an on-going videoconference was a mature art with sophisticated commercial products available in the marketplace.

55. The patent acknowledges two existing solutions for sharing desktop information. One solution is to project the desktop information (the information displayed on a computer’s monitor) with a projector onto a surface, capture the projected image with a video camera, and transfer the captured desktop information to a remote video communications terminal after processing. EX1001, Col. 1:30-36. The other is to transform the desktop information to a standard video format signal such as “PAUNTSC” (sic., “PAL/NTSC”) with a video graphic array (VGA) convertor, and input that signal to the videoconference terminal, where the signal is processed and transferred to a remote video communication terminal. *Id.*, 1:36-43. These solutions were not particularly novel or sophisticated at the time the ’288 Patent was filed.

56. The ’288 Patent acknowledges these solutions are “simple and real-time” but have problems. Allegedly, with the first solution, “to keep higher clarity the visual part of the captured image becomes smaller, i.e. the desktop information

cannot be wholly transferred at a time.” *Id.*, 1:44-54. With the second solution, converting the digital VGA signal to an analog, lower resolution television signal results in a loss of image clarity. *Id.*

57. The '288 Patent describes an embodiment of their solution in Figure 4:

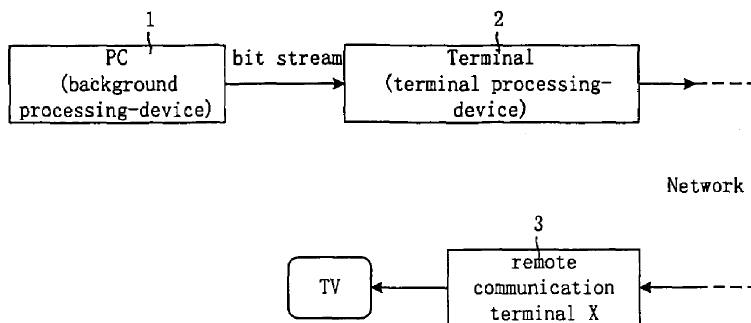


Fig. 4

Figure 4 of the '288 Patent (EX1001)

The images displayed on a PC’s monitor are captured and converted into a format of a local video communication terminal, encoded into a bit stream at a PC, and sent to a local video communication terminal. A local video communication terminal or a terminal processing device then transmits the processed signal to a remote communication terminal where it can be displayed.

58. The '288 Patent includes 17 claims. Two are independent. Claim 1 recites:

1. A method for transferring desktop information of a PC to a video communication terminal, comprising,

a) capturing desktop information of the PC after receiving a triggering command, and converting a PC format of the desktop information into a format of a local video communication terminal;

b) encoding the converted desktop format in a mode ensured by the local video communication terminal;

c) sending the coded bit stream to the local video communication terminal;

d) receiving the coded bit stream by the local video communication terminal and transferring to a remote video communication terminal through transmission channel after processing.

EX1001, Col. 4:58-5:4. Claim 9 is an independent apparatus claim and states as follows:

9. An apparatus for transferring desktop information of a PC to a video communication terminal, comprises,

a background processing-device, capturing desktop information of the PC, converting the captured desktop information from a PC format to a format of a local video communication terminal, encoding the converted format to a coded bit stream and outputting;

a terminal processing-device, transferring the coded bit stream from the background processing-device to a remote video communication terminal.

59. The '288 Patent asserts that the purported invention transfers desktop information to a remote video communication terminal “integrally and clearly.”

EX1001, Col. 1:61-64. The advantages purportedly arise because (*Id.*, Col. 2:36-48):

1. The PC desktop information is transferred directly in a digital coded bit stream mode without converting the digital signal to analog;
2. The desktop information is “pre-processed before transmission” so that it needs less bandwidth.
3. Unchanged desktop information saves system resources; and
4. Moving images of desktop information can be transmitted simultaneously or alternatively with other desktop information.

60. However, a POSA would recognize that all of these purported advantages were present in virtually all videoconferencing systems that supported screen sharing in 2002.

61. The detailed description of the preferred embodiment in the '288 Patent occupies less than two full columns and is accompanied by four figures. Column three is largely directed to the flow chart of Figure 1 which relates to the method of Claim 1:

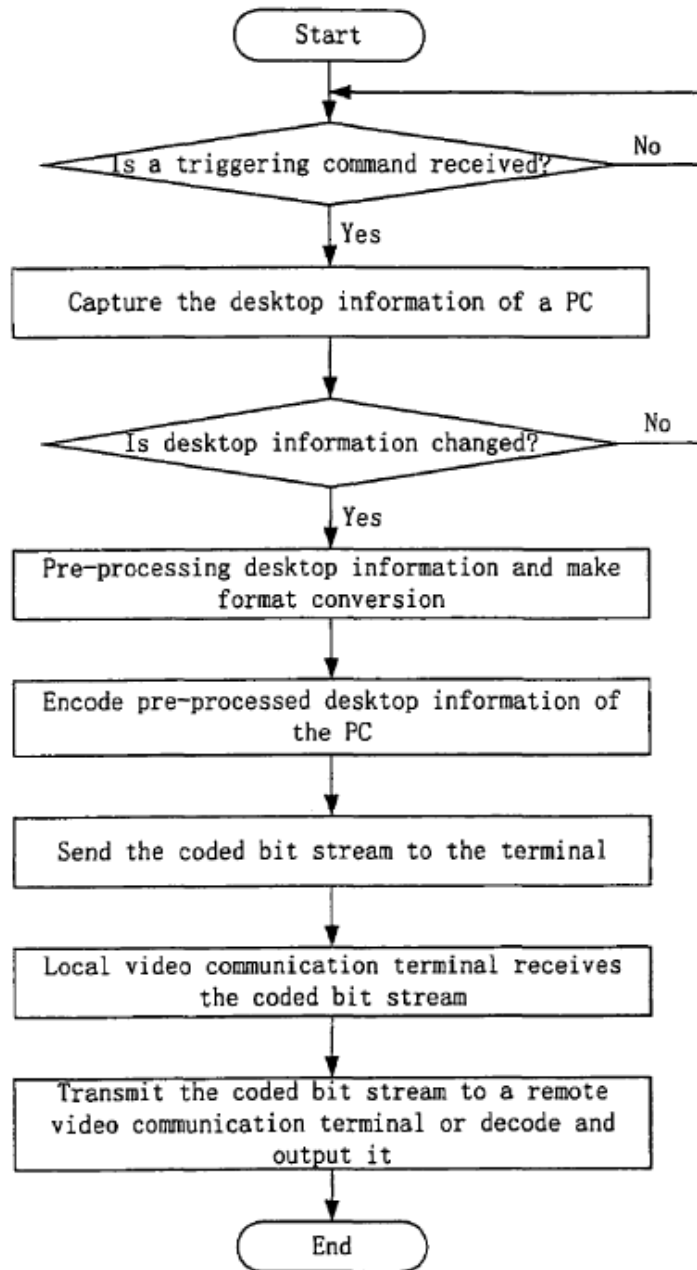


Fig. 1

Figure 1 of the '288 Patent (EX1001)

62. While Figure 1 does not number its steps, the specification numbers them. See EX1001, Col. 3:11-34. Step 5, “Encode pre-processed desktop

information of the PC,” states that the encoding is into a “video bit stream of which type can be a H.261 bit stream, a H.263 bit stream or a JPEG bit stream etc.” EX1001 Col. 3:20-28. The patent thus acknowledges that the H.261 and H.263 standards were known and created a coded video bit stream.

63. Figure 4 shows the apparatus as connected to a PC and remote communication terminal:

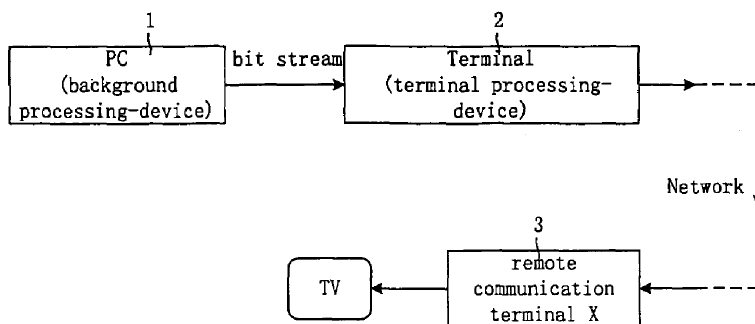


Fig. 4

Figure 4 of the '288 Patent (EX1001)

64. Figure 3 shows a terminal processing device having a terminal encoder 31, terminal bit stream transmitter 33 and terminal decoder 34. EX1001, Col. 4:24-26. The local image is encoded and then outputted to the terminal bit stream transmitter 33. The transmitter can transmit the local image (such as recorded by a video camera), desktop information, or both. *Id.*, Cols. 4:28-34; 1:27-28.

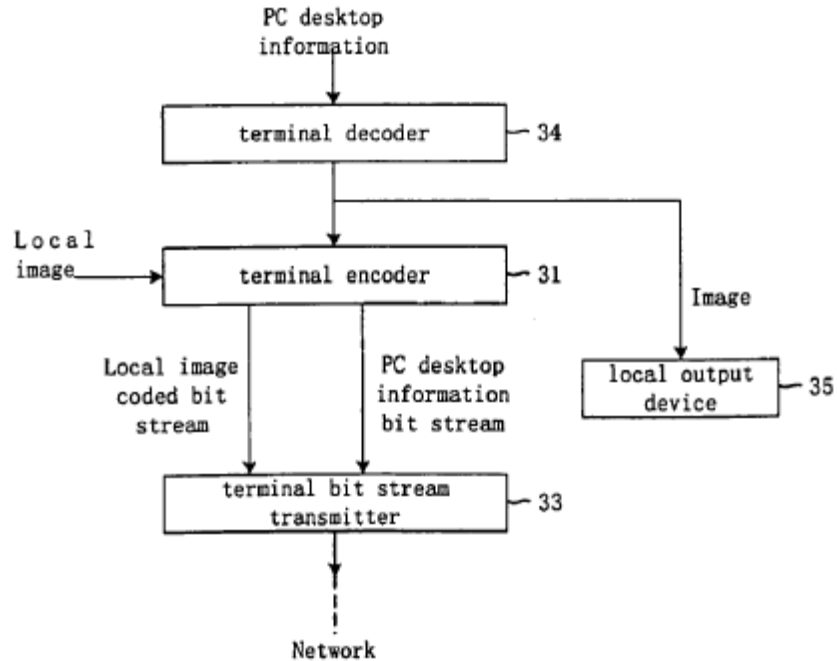


Fig. 3

Figure 3 of the '288 Patent (EX1001)

65. In sum, the '288 Patent is directed to a straightforward and obvious way of transmitting a visual image of a desktop of a PC in a videoconferencing system. The specific way precisely mirrors the way video data in a video conference is processed and transmitted in prior art videoconferencing systems (which is not surprising given that the '288 Patent treats the desktop information of a PC as an image that is acquired in manner similar to how other video is acquired). The '288 Patent describes "capturing" desktop information upon receipt of a "triggering command," converting the information into a format of a local video communication terminal, encoding the converted information using an H.261, H.263, or similar

video encoder, and sending the encoded bit-stream to the local video communication terminal where it is sent to a remote terminal. These are all well-known and well-practiced steps in the videoconferencing art. In addition, the '288 Patent employs a well-known architecture to realize its method steps and apparatus, namely using an intermediary device (the local video communication terminal) to perform processing and transmission of media to a remote terminal. For at least these reasons, as further developed herein, it is my opinion that the claims of the '288 Patent are obvious in view of the prior art.

B. The Prosecution History of the '288 Patent

66. The '288 Patent was filed as U.S. Patent Application No. 11/049,217 (“’217 Application”) on February 2, 2005. The '288 Patent is a continuation of PCT Application No. 1PCT/CM03/00627, filed August 4, 2003 and claims foreign priority to (Chinese) application No. 02128762, filed August 7, 2002. EX1001 at 1.

67. The USPTO issued a non-final rejection of all 17 pending claims on January 27, 2005. EX1002 at 134-136. Claims 2 and 3 were rejected under 35 U.S.C. §112. *Id.* Claims 1-2, 4-8, and 10 were rejected as obvious in view of prior art including Kenoyer et al., (US 2002/0154209) and Clapp (U.S. Patent No. 6,073,192). *Id.* Kenoyer was cited for essentially all elements of Claim 1 other than a triggering command; Clapp disclosed a videoconference system using a triggering command. *Id.* Claims 3 and 11 were also rejected as unpatentable over Kenoyer,

Clapp and Dunlap (U.S. Patent No. 6,560,637). *Id.* Independent Claim 9 and Claims 12-15 and 17 were also rejected as anticipated by Kenoyer. *Id.* Claim 16 was the only claim not rejected on prior art grounds. *Id.*

68. In its April 27, 2009 response, the Applicant amended Claims 1 and 9 as well as dependent claims. EX1002 at 159-160. The Applicant argued that Kenoyer does not teach “encoding the converted desktop format in a mode ensured by the local video communication terminal.” *Id.* Further, The Applicant asserted Kenoyer does not teach sending a coded bit stream but rather teaches sending signals to the video communication terminal in a YUV format (“signals sent to the local video communication terminal in Kenoyer are YUV format data instead of the coded bit stream as recited in claim 1”). *Id.* at 160.

69. The Applicant further argued that Kenoyer’s purported deficiencies were not cured by the other cited art, Clapp and Dunlap. EX1002 at 160. The Applicant relied on its arguments regarding Claim 1 to also contend that Claims 2-8 were distinguished from the cited prior art. *Id.* at 161. The Applicant also separately argued that Claims 6 and 7 were allowable. *Id.* Both of these claims further recited aspects of coded bit stream transmission. *Id.*

70. The Applicant relied on its arguments regarding the obviousness of independent Claim 9 as showing the patentability of Claims 10 and 11 as well. EX1002 at 162. The Applicant similarly relied on its arguments for Claim 9

regarding the anticipation-based rejection to also support the patentability of claims 12-15 and 17. *Id.* Claim 16, which is not at issue in this Petition, was deemed allowable if rewritten in independent form. The Applicant amended Claim 16 accordingly.

71. Kenoyer includes special purpose hardware in the form of an FPGA (Field Programmable Gate Array) for the purpose of converting and encoding the desktop information as YUV data:

To enhance the throughput and performance of the video conversion and compression process, the FPGA 320 works in conjunction with the processor 330 and the memory 340. The FPGA 320 uses the memory 340 as a buffer, storing several images of RGB data into the memory 340 before converting the images to YUV format.

EX1005, ¶[0029].

72. The above-quoted discussion refers to Figure 3, which, as shown below, demonstrates that the RGB signal is digitized (“RGB A/D” 310), converted, compressed, and processed by the FPGA 320 and Processor 330, and sent from the videoconference module 110 to the videoconference system 120 as a bit stream in a YUV format:

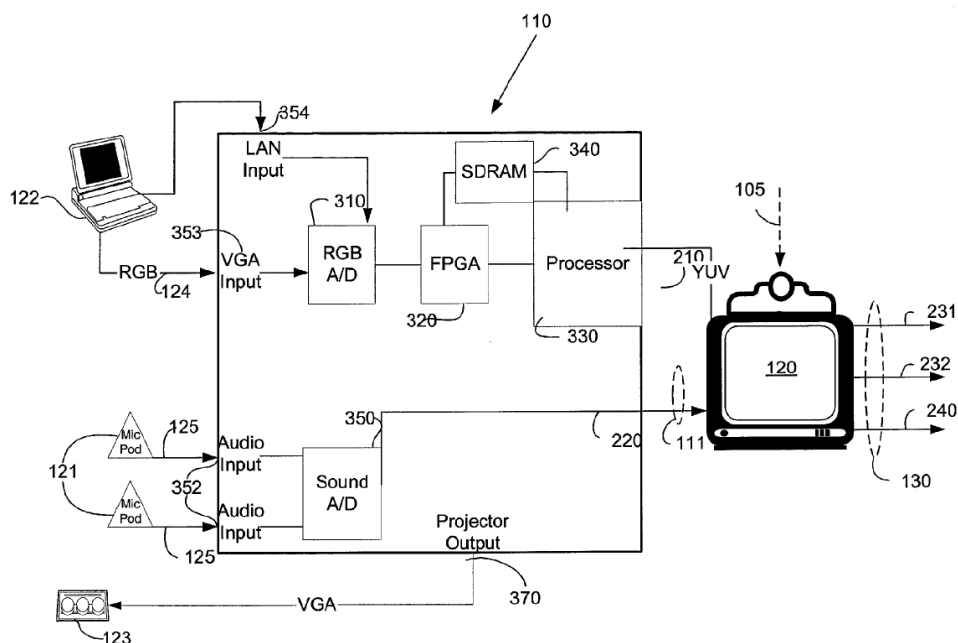


FIG. 3

Figure 3 of Kenoyer (EX1005)

73. The '288 Patent does provide examples of acceptable encodings through its disclosures of well-known encoding and compression standards useful for videoconferencing applications, such as the H.261 and H.263 standards. A POSA would have already understood and known before the '288 Patent was filed that these well-known standards were specifically developed for videoconferencing applications such as the videoconferencing system disclosed in Kenoyer.

74. Following the April 27, 2009 amendment and remarks, the U.S. Patent Office issued a Notice of Allowability for Claims 1-17 on August 7, 2009. EX1002 at 167. No statement of reasons for allowance was provided.

C. The Priority Date of the '288 Patent

75. I understand that Patent Owner filed the '217 Application leading to the '288 Patent on February 2, 2005. I also understand that the '288 Patent claims priority as a continuation application from PCT Application No. CN03/00627, filed on August 4, 2003, and claims foreign application priority to August 7, 2002. EX1001. I have therefore analyzed anticipation and obviousness as of August 7, 2002.

VI. STATE OF THE ART PRIOR TO THE '288 PATENT

A. Becker

76. Becker published as U.S. Patent Application Publication No. 2002/0149617A1 on October 17, 2002, based on an application filed March 28, 2002, and a provisional application filed on March 30, 2001. EX1009. It is my understanding that Becker is prior art to the '288 Patent under 35 U.S.C. §102(e)(1) (pre-AIA). Becker was not cited during examination of the '288 Patent.

77. Becker is entitled "Remote Collaboration Technology Design and Methodology." It teaches converting high-resolution video or analog RGB computer video into signals such as High-Definition Television (HDTV) signals for viewing at a remote location. The signals are converted and encoded. EX1009, Abstract, ¶[0020]. Encoded video signals are sent over ATM 60 to a decoder 152 at a remote location where the signals are decoded and converted for viewing on an HDTV monitor or RGB computer monitor. *Id.*, ¶[0048].

B. Kenoyer '423

78. Kenoyer '423 issued as U.S. Patent No. 9,769,423 on September 19, 2017, based on an application filed January 28, 2016 which claims priority to an application filed June 3, 2002, and a provisional application filed on June 2, 2001. EX1010. It is my understanding that Kenoyer '423 qualifies as prior art to the '288 Patent under 35 U.S.C. § 102(e)(1) (pre-AIA). Kenoyer '423 was not cited during examination of the '288 Patent. Kenoyer '423 is entitled *System and Method for Point to Point Integration of Personal Computers with Videoconferencing Systems*. It teaches capturing desktop information through clicking a Show PC button 210. EX1010, Col. 3:11-13. The signals are converted into Common Intermediate format (CIF) sub images and encoded using the H.261 protocol. *Id.*, Col. 4:31-46.

C. Kenoyer

79. Kenoyer published as U.S. Patent Application Publication No. 2002/0154209A1 on October 24, 2002, based on an application filed March 21, 2002, and a provisional application filed on March 22, 2001. EX1005. It is my understanding that Kenoyer is prior art to the '288 Patent under 35 U.S.C. § 102(e)(1) (pre-AIA). Kenoyer was the primary reference cited during examination of the '288 Patent.

80. Kenoyer is entitled "High Resolution Graphics Side Channel in Video Conference." Kenoyer discloses a method for transferring desktop information of a

PC (122) through a videoconference accessory module 110 to a video communication terminal (120), as shown in Figure 1 below:

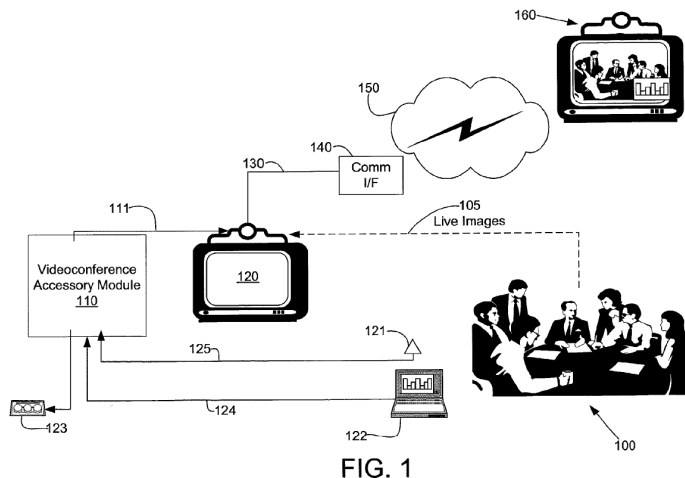


Figure 1 of Kenoyer (EX1005)

D. Clapp

81. Clapp was filed as U.S. Application No. 09/019,114 on February 5, 1988. EX1006. It is based on a continuation application filed September 7, 1994. *Id.* It issued as U.S. Patent No. 6,073,192 on June 6, 2000. *Id.* Accordingly, it is my understanding that Clapp qualifies as prior art at least under 35 USC S. 102(b) (pre-AIA) because it was published more than one year before the August 7, 2002 priority date of the '288 Patent. Clapp was cited by the Examiner during examination of the '288 Patent as a secondary reference for its teachings regarding a triggering command.

82. Clapp is directed to a peripheral videoconferencing system that is compatible with a range of input and display devices. All audio and video processing

boards and components comprising the system are housed in a peripheral housing and coupled with standard connection interfaces to the host computer system (PC). EX1006, Col. 5:4-20. This is in contrast to prior videoconferencing systems which used computer cards installed in a PC to carry out the videoconferencing functionality inside the PC. *Id.*, Col. 3:1-18. Figure 3 of Clapp shows an embodiment of the invention:

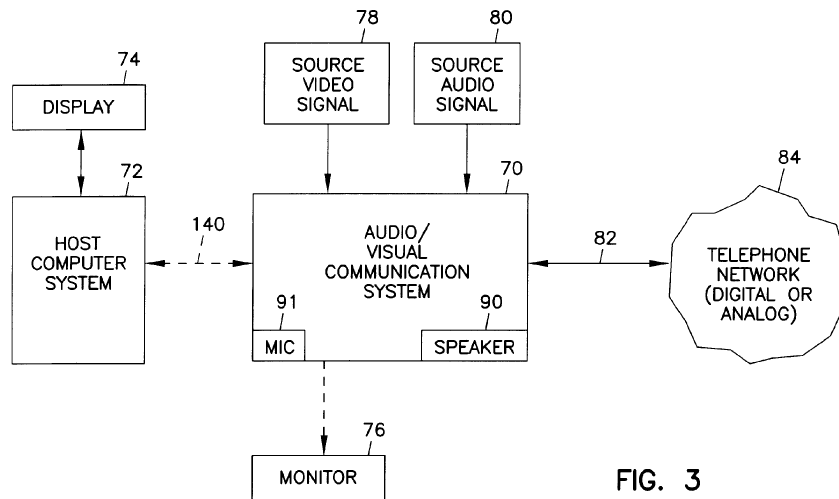


FIG. 3

Figure 3 of Clapp

E. Shaw

83. U.S. Patent No. 5,611,038 to Shaw et al. (“Shaw”) issued March 11, 1997. EX1007. It is my understanding that Shaw is prior art under 35 U.S.C. § 102(b) at least because it was published as a patent more than one year before the August 7, 2002 priority date of the ’288 Patent. Shaw was not cited during prosecution of the ’288 Patent.

84. Shaw is entitled “Audio/Video Transceiver Provided with a Device for Reconfiguration of Incompatibly Received or Transmitted Video and Audio Information.” EX1007. Shaw’s summary indicates that it is an improvement on traditional video conferencing systems in that it provides more options for presenting video while conserving bandwidth. EX1007, Col. 3:30-67.

85. Shaw teaches conversion of a color image from RGB to YUV. EX1007, Col. 9:5-41. The image may come from various displays including those of PC or workstation 106. *Id.*, Col. 22:19-25. Capture processor 316 decodes analog video inputs and converts them to standard CIF YUV format.² *Id.*, Col. 21:39-42. Image video may conform to international standards such as H.261. *Id.*, Col. 12:34-41. The encoded YUV data is transmitted in coded bit stream. *Id.*, Col. 28:41-44.

86. Shaw states that the H.261 standard is “commonly applied” for use with Common Intermediate Format (CIF) and Quarter CIF (QCIF) “to facilitate the transport of video information over the telecommunications network.” EX1007, Col. 11:6-13. Use of a standard such as H.261 provides “ease of interface.” *Id.*, Col.

² CIF (Common Intermediate Format) was and is a known video resolution (a video image size of 352×288 picture elements) for videoconferencing defined in the H.261 standard.

12:34-40. H.261 is also noted for providing motion estimation. *Id.*, Cols. 18:34-48, 29:43-44.

F. *Video Compression Techniques* textbook

87. The cover pages and pages 58-65 of *Video Compression Techniques*, by Effelsberg and Steinmetz, are from a well-known technical publication that has a copyright date of 1998. EX10011. *Video Compression Techniques* (also sometimes referred to herein as *Video Compression*) has a standard ISBN number 3-920993-13-6, and a search at ISBNsearch.org shows this text has a publication date of November 1998 and is readily available from booksellers including amazon.com and ecampusbooks.com. It is my understanding that *Video Compression Techniques* is prior art at least under pre-AIA 35 U.S.C. § 102(b). It was not considered by the Examiner of the '288 Patent.

88. *Video Compression Techniques* describes the H.261 digital video compression/decompression standard. EX1008 at 59-63. The excerpt from that text explains that the standard was motivated by ISDN (Integrated Services Digital Network) communications and designed to be a scalable standard capable of producing a “coded bit-stream” in multiples of 64 Kbits per second. *Id.* at 59. The text includes the following figure illustrating the basic architecture of an H.261 encoder:

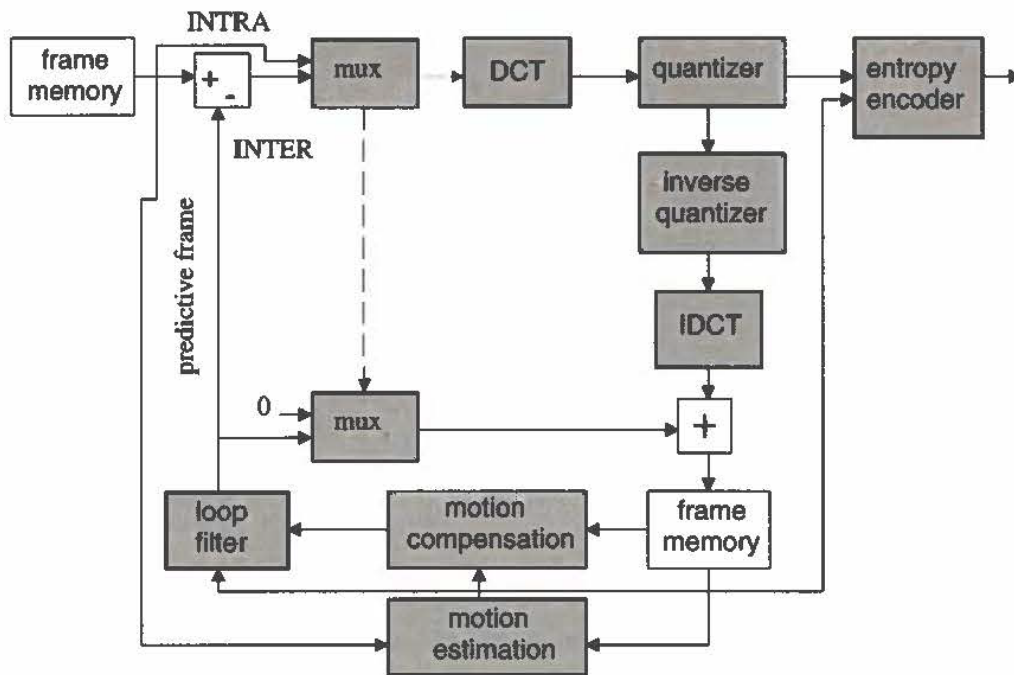


Figure 5-13: H.261 encoder

Figure 5-13 of *Video Compression Techniques (EX1008)*

The excerpt also shows a Table of CIF and QCIF³ formats for processing YUV image data. (The Table uses an alternate syntax/name for the U and V chrominance components where U = Cb (a “blue difference”) and V = Cr (a “red-difference”). Note that the Table also illustrates the (predominant) use of chrominance

³ QCIF (Quarter Common Intermediate Format) was and is also a known video resolution for videoconferencing defined in the H.261 standard. A QCIF image is one-quarter the resolution/size of a CIF image (a video image size of 176×144 picture elements).

subsampling as the chrominance values are sampled at one-half the resolution vertically and horizontally of the luminance component.

Table 5-6: Source image formats of H.261

	CIF	QCIF
Y	352×288	176×144
Cb, Cr	176×144	88×72

Figure 5-6 of *Video Compression Techniques* (EX1008)

Id. at 63.

89. *Video Compression* states that the focus of the H.261 design was “real-time encoding-decoding system with less than 150 ms delay” as well as to allow encoder developers to choose features depending on the desired trade-off between cost and performance. EX1008 at 4 (p. 59). H.261 provides motion estimation at reduced costs. *Id.* at 5 (p. 61). It is designed for applications with “strict timing requirements” including “video conferencing.” *Id.* at 6 (p. 63).

VII. CLAIM CONSTRUCTIONS

90. As discussed above, I analyzed the claims and compared them to the prior art as interpreted in a manner consistent with the standard used in patent litigation, as set forth in *Phillips v. AWH Corp.*, 415 F.3d 1303 (Fed. Cir. 2005) (*en banc*). Specifically, I gave terms their plain and ordinary meaning as understood by a POSA in the context of the '288 Patent and claims.

91. Below I provide a construction for the term “bit stream,” although the construction is also consistent with the plain and ordinary meaning of the term in the context of the claims and the intrinsic record.

92. I construe the term “bit stream” as used in Claim 1 in the phrase “sending the coded bit stream to the local video communications terminal.” EX1001, Col. 4:66-67.

93. Claim 9 uses the phrase, “encoding the converted format to a coded bit string.” EX1001, Col. 5:39-41. I construe the term “bit stream” similarly for Claim 1 and Claim 9 for purposes of my analysis.

94. The term “bit stream” also appears in some dependent claims. I construe the term similarly as used in the relevant dependent claims as well.

95. The term “bit stream” as a term of art used to describe transmission of signals pertaining to video communications.

96. The '288 patent does not define “bit stream” but discusses “encoding the converted desktop format in a mode ensured by the local video communications terminal” in the context of a method for transferring desktop information of a PC to a video communication terminal. EX1001, Col. 2:5-6.

97. The '288 Patent also discusses “encoding the converted format to a coded bit stream and outputting” in the context of an apparatus for transferring desktop information of a PC to a video communication terminal. *Id.*, Col. 2:26-27.

98. The IEEE 100 Authoritative Dictionary of IEEE Standards Terms (Dec. 2000) defines “bit stream” as “a continuous stream of bits transmitted over a channel with no separators between the character groups.” EX1017 at 12. The IEEE Standard Glossary of Computer Networking Terminology (June 30, 1995) provides the same definition. EX1018 at 9. Both of these definitions are consistent with my experience and understanding of the term “bit stream” as used in the context of transmitting video communications data.

99. A POSA thus would understand “bit stream” to mean “a continuous stream of bits transmitted over a channel with no separators between the character groups.” Even if the term is not construed, however, my analysis and conclusions provided herein are unaffected.

VIII. GROUNDS OF INVALIDITY

A. Claim 1 is obvious in view of Becker and Kenoyer ’423.

1. A method for transferring desktop information of a PC to a video communication terminal comprising:

100. US 2002/0149617A1 (Becker), “Remote Collaboration Technology Design and Methodology,” discloses a method for transferring desktop information of a PC to a video communication terminal. EX1009. Figs. 4A-4D are addressed in a section describing routing computer video signals over large distances. *Id.*, ¶¶[0069], [0074].

101. Fig. 1 shows a system including computers 1, 2, 3, 4 such as a desktop PC. EX1009, ¶[0060]. Signals are processed and sent to an ATM switch 60, which sends the signal to an ATM switch 160 at a remote location. *Id.*, ¶[0076].

Figure 1: Generalized representation of the structure of and method for use of the Remote Collaboration System.

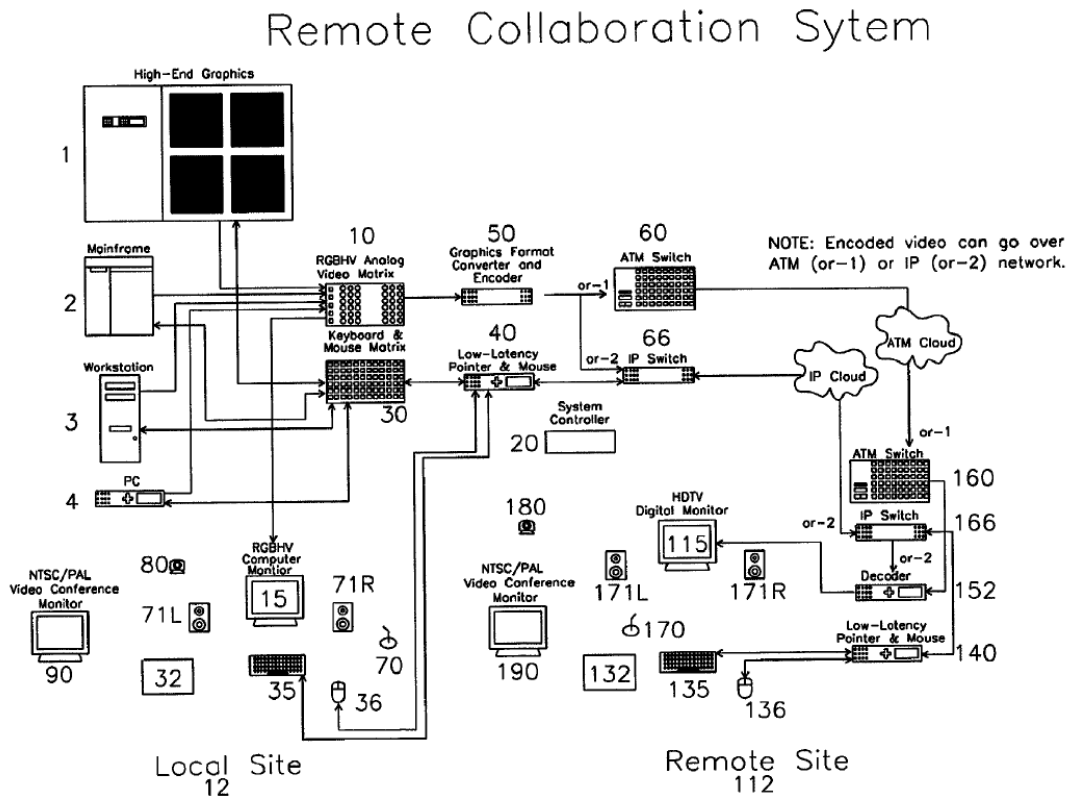


Figure 1 of Becker (EX1009)

102. Becker teaches a method for transferring desktop information of a PC to a video communication terminal. “The present invention captures the graphics signals output from the computer in their raw video format (the format that is sent to the computer monitor for viewing).” EX1009, ¶[0020]. “As shown in FIG. 1, computer RGB information is routed from the computer 1, 2, 3, 4 to both a monitor

15R at the local location and also to a graphics format converter and encoder 50. The encoded signals are sent over ATM 60 or the Internet 64 to a decoder 152 at the remote location 112. From there they are converted back and viewed either on an HDTV-capable monitor 115R, or a normal analog RGB computer monitor.” *Id.*, ¶[0048].

103. Kenoyer ’423 also teaches a method of transferring desktop information of a PC to a video communication terminal. “Clicking on Show PC button 210 will send a desktop presentation such as a PowerPoint® presentation from the videoconferencing system to a remote site.” EX1010, Col. 3:11-22.

2. Capturing desktop information of the PC after receiving a triggering command, and converting a PC format of the desktop information into a format of a local video communication channel

104. Becker teaches capturing desktop information of the PC after receiving a triggering command and converting a PC format of the desktop information into a format of a local video communication channel. “The present invention captures the graphics signals being output from the computer in their raw video format (the format that is sent to the computer monitor for viewing).” EX1009, ¶[0020]. “In the preferred embodiment of the method, the analog RGBHV signals are converted to serial digital high-definition television, SDI-HDTV . . .” *Id.*, ¶[0071].

105. “Referring to FIGS. 4-B-C, the appropriate RGBHV Signals from the matrix Switch (10) are directed into a signal reformatter (50).” EX1009, ¶[0074]. “The reformatter converts the analog 1280 by 1024, RGBHV component video signal into an analog 1920 by 1080i HDTV signal.” *Id.* “From there, the analog HDTV signal is sent into an A/D converter (51).” *Id.* The communications between reformatter 50 and switches 60, 66 use a communications channel as shown for example by the lines between these devices in Figure 1 (see above).

106. Kenoyer '423 also teaches capturing desktop information using a triggering command. “Dual stream communication provides for integration of a coupled personal computer or a laptop computer with an existing videoconferencing system.” EX1010, Col. 2:48-50. “Clicking on Show PC button 210 will send a presentation loaded into a laptop or personal computer from the videoconferencing System to a remote site.” *Id.*, Col. 3:11-13. “[A] method 500 includes a step 510, in which an image being presented is monitored for changes or movement. More particularly, step 510 monitors changes in macroblocks. If a step 520 change or movement is detected in the macroblock, then in a step 530 pixels of the changed macroblock are locally processed, coded, and, in a step 540, transmitted.” *Id.*, Col. 4:65-5:1.

107. A POSA would have found it obvious to modify Becker with Kenoyer '423's teachings of a triggering command for capturing desktop information.

Becker recognizes the benefits of sending digital information and the benefits of minimizing the amount of bandwidth required to send information. EX1009, ¶[0070]. A POSA would be motivated to modify Becker with Kenoyer's teachings, which reduce the required bandwidth by sending image signals only if change or movement is detected.

108. Becker also teaches converting a PC format of the captured desktop information into a format of a local video communication terminal. "[T]he analog RGBHV signals are converted to serial digital high-definition television, SDI-HDTV" EX1009, ¶[0071].

109. Kenoyer '423 also teaches converting a PC format of the captured desktop information into a format of a local video communication terminal. "Annex D provides a sub-sampling of image pixels to generate four Common Intermediate Format (CIF) 352x280 pixel sub-images . . ." EX1010, Col. 4:39-41. This format conversion also converts the desktop image into four separate Common Intermediate Format (CIF) sub-images, consistent with the conversion provided in the '288 Patent. EX1001, Col. 3:27.

110. A POSA would be motivated to implement Kenoyer '423's triggering command as a desirable modification of Becker because it provides a form of control over the display of desktop information. As a POSA would have known, Becker teaches such control is desirable. *See, e.g.*, EX1009, Abstract (disclosing

“[l]ocal and remote mouse and keyboard control of the computer imagery or other high-resolution videosource”), ¶[0021] (“These capabilities are also combined with the appropriate control systems . . . by either manual or automatic control.”) Thus, it would have been obvious to modify Becker with Kenoyer ’423’s teachings to capture desktop information using a triggering command as claimed.

3. Encoding the converted desktop format in a mode ensured by the local video communications terminal

111. Becker teaches the encoding step. “[T]he SD-HDTV signal coming from the A/D converter (51) is then sent to the MPEG compression device (52) for encoding and compression.” EX1009, ¶[0076]. MPEG was a well known video encoding protocol comparable to H.261, which is listed in the ’288 Patent as an example video encoding protocol. *See also id.*, ¶¶[0071], [0072] (teaching encoding the RGBHV signals directly or after conversion to serial digital high-definition television.). The encoding is in a mode capable of being processed by switches 60, 66, which serve as local video communications terminals.

4. Sending the coded bit stream to the local video communication terminal

112. Becker teaches sending the coded video bit stream to a local video communication terminal. “In one embodiment, the encoded HDTV signals from the MPEG compression device (52) are then sent to an ATM computer network switch

(60), FIG. 4-D.” EX1009, ¶[0077]; *see also id.*, ¶[0200] (stating that the local ATM computer network switch (60) may contain “video codecs”).

113. Becker’s encoded information is in video bit stream format. “[T]he analog HDTV signal is sent into an A/D converter (51). “The A/D converter converts the analog HDTV Signal into a Serial digital Stream, SDI, of data.” EX1009, ¶[0074]. Paragraph [0071] of Becker similarly describes that the “signals can be compressed (encoded) using, for example, an MPEG compression algorithm.” *Id.*, ¶[0071]. A POSA would understand that such MPEG conversion would encode the information in a video bit stream format.

114. A POSA would understand that Kenoyer ’423 also discloses sending the encoded signal in bit stream format. Kenoyer ’423 refers to “streaming” the compressed image to a videoconferencing system. EX1010, Col. 3:37-39. It also refers to transmitted image information in the form of bits. *Id.*, Col. 5:10-14. It describes H.261 encoding, which a POSA would understand is used to transmit video signal exclusively in digital video bit stream format. Becker teaches transmitting the encoding information in bit stream format, and a POSA seeking to implement Kenoyer ’423’s teachings would be motivated to use Becker’s teachings to transmit the H.261 encoded information of Kenoyer ’423.

5. Receiving the coded bit stream by the local video communication terminal and transferring it to a remote

video communication terminal through transmission channel after processing

115. Becker teaches receiving the encoded bit stream as claimed. “In one embodiment, the encoded HDTV signals from the MPEG compression device (52) are then sent to an ATM computer network switch (60), FIG. 4-D. From there, the information is transmitted across communication lines to the receiving ATM switch 160 at the remote location (90).” EX1009, ¶[0077]. “Referring to FIG. 4-C, from the ATM switch 160 at the remote location the signals are sent into the MPEG decoder device (152). . . . From there the signals can be directed into a digital HDTV monitor for viewing.” *Id.*, ¶[0078].

116. Becker teaches alternatively transferring the coded bit stream through the ATM devices as described above, or through local and remote IP routers 66, 166. *See* EX1009, Figure 1 (highlighted) below:

118. Kenoyer '423 teaches that a user may select between presentations to be displayed, which will initiate transmission, as well as pause, resume, or end transmission of a presentation. EX1010, Col. 3:8-37. This enables a form of pre-processing by limiting the desktop information captured for transmission. A POSA would have been motivated to modify Becker with this capability to provide the user with greater control during collaborations and to reduce the bandwidth used for transmission, both of which are desirable as taught by Becker. EX1009, ¶¶[0070], [0055].

119. Claim 5 recites “the desktop information in Step a is captured from a video memory of the PC.” EX1001, Col. 5:17-18. Kenoyer '423 teaches capturing video images using a PCMCIA card inserted into, for example, a laptop computer. EX1010, Col. 2:8-19; 2:57-67. A POSA would understand that the PCMCIA device in Kenoyer '423 is used to track changes to video memory and encode the video data for display and transmission by the rest of the system. The video data, obviously in digital form, is stored in macroblocks where it is monitored for changes that will trigger processing and coding. *Id.*, Col. 4:61-5:5. Becker teaches processing captured image data, and Kenoyer '423 provides a mechanism by which the information to be processed may be captured from the memory of a PC in order that the video data may be processed efficiently.

120. Claim 6 recites that step (d) of claim 1 further comprises decoding the received bit stream and then outputting to a local output device. EX1001, Col. 5:19-21. Becker teaches this claim. A POSA would understand that the purpose of sending a coded video bit stream to a remote video communication terminal in Becker is so that the bit stream can be decoded to display the video signal on a remote output device (video communication terminal). “At the receiving location they [the compressed digital signals] are decompressed (decoded).” EX1009, ¶[0071], Figure 4D. “Once decoded the transmitted information can be viewed on an HDTV-capable display 115L, 115R.” *Id.* “Alternatively, the HDTV signal can also be converted back to RGBHV for viewing on an analog computer display device.” *Id.*

121. Claim 7 of the '288 Patent recites that step d of Claim 1 further comprises “simultaneously transferring the coded bit stream of the desktop information and that of a local image in a multiplex encoding mode.” EX1001, Col. 5:22-25. Becker teaches this feature. “In one embodiment, all signals go over the same connection using a virtual private network VPN.” EX1009, ¶[0227]. “However, that does not need to be the case.” *Id.* “The signals can be sent over separate, individual data lines, or can be multiplexed together and sent over the same line.” *Id.* “All signals” broadly refers to transmissions including high-definition signals corresponding to two computer monitors (*id.*, ¶[0225]) or signals from

cameras connected to video codecs located within the local ATM switch 60 (*id.*, ¶¶[0199]-[0200], Figure 4).

122. Kenoyer '423 also teaches this claim. Kenoyer '423 teaches dynamically sharing connection bandwidth between the graphic image and the presenter's video such that "when the computer screen is not changing, all of the bandwidth can be devoted to showing the live view of the presenter." 3:46-38.

123. Multiplexing was well known and in widespread use for transmitting videoconference video signals at the time of the '288 Patent. A POSA would have been well aware of the benefits of multiplexing video signals in a videoconferencing application at least because the RFC 1889 standard would have been known, and would have been motivated to use multiplexing to preserve bandwidth to reduce costs and increase speed of transmission. For all of these reasons, Claim 7 would have been obvious.

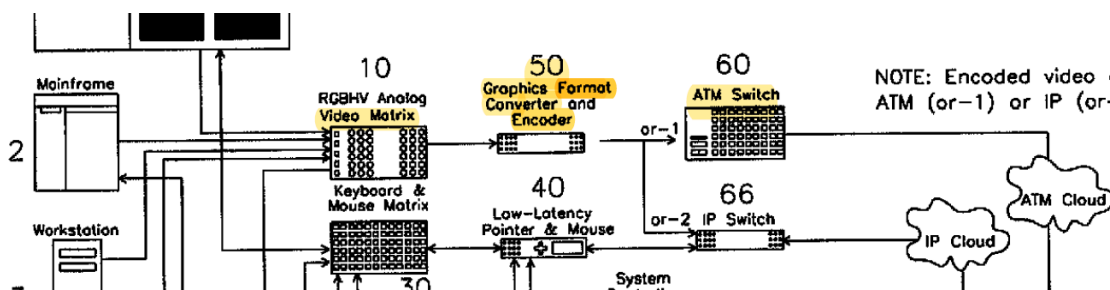
C. Claim 9 is obvious in view of Becker and Kenoyer '423.

124. Claim 9 is an apparatus claim, in contrast to method Claim 1. *Compare* EX1001, Col. 5:33-43 with EX1001, Col. 4:58-5:4. Claim 1 recites a triggering command, but Claim 9 does not. *Id.* Thus Claim 9 is broader than Claim 1 at least in this respect, and the discussion above showing Claim 1 is obvious in view of Kenoyer '423 and Becker also shows that Claim 9 would have been obvious to a

POSA in view of Kenoyer '423 further in view of Becker. An element-by-element analysis is provided below.

1. “An apparatus for transferring desktop information of a PC to a video communication terminal, comprises”

125. Becker teaches an apparatus for transferring desktop information of a PC to a video communication terminal. Figure 1 (excerpt below) shows desktop information routed from the computer to a graphics format converter and encoder 50, sent over ATM 60 or the Internet to a decoder at a remote location where the signals are converted back and viewed on a monitor or computer. EX1009, ¶[0048].



Annotated Excerpt of Figure 1 of Becker (EX1009)

These components are an apparatus for transferring desktop information of a PC to a video communication terminal as further explained above in regard to Claim 1. Converter and decoder 50 serves as at least part of an apparatus for transferring desktop information of a PC to a video communication terminal (ATM Switch 60).

2. “A background processing device, capturing desktop information, converting the captured desktop information

from a PC format to a format of a local video communication terminal”

126. The '288 Patent has few disclosures of the claimed “background processing device” beyond text similar to the language of the claims. Figure 2 shows a block diagram of the background processing device, and Figure 4 shows the PC as the background processing device:

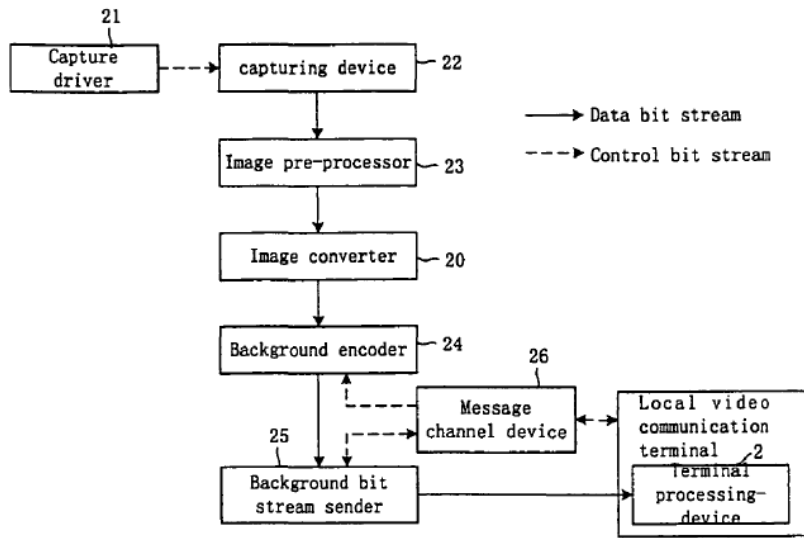


Fig. 2

Figure 2 of the '288 Patent (EX1001)

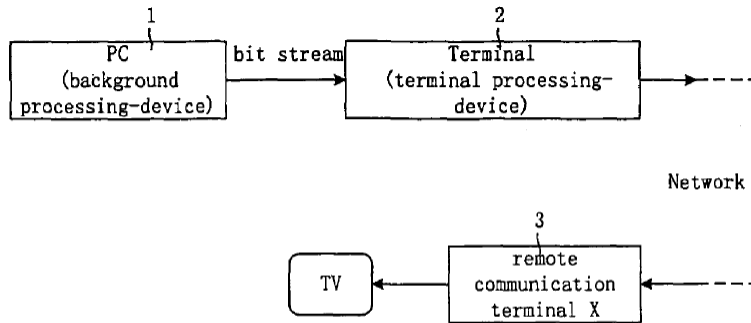
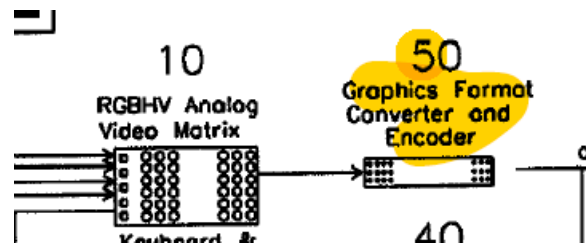


Fig. 4

Figure 4 of the '288 Patent (EX1001)

127. The components for capturing desktop information, processing, converting, and encoding the image are all found in Becker at device 50 in Figure 1 (excerpt below). An excerpt is provided below:



Annotated Excerpt of Figure 1 of Becker (EX1009)

3. “[a background device] encoding the converted format to a coded bit stream and outputting”

128. “The present invention captures the graphics signals being output from the computer in their raw video format (the format that is sent to the computer monitor for viewing).” EX1009, ¶[0020]. At converter 50, “the analog RGBHV signals are converted to serial digital high-definition television, SDI-HDTV . . .” *Id.*, ¶[0071].

129. Also as discussed above, Kenoyer '423 teaches capturing desktop information using a triggering command, and a POSA would have been motivated to implement Kenoyer '423's capturing function as a desirable modification of Becker. "Clicking on Show PC button 210 will send a presentation loaded into a laptop or personal computer from the videoconferencing System to a remote site." EX1010, Col. 3:11-13; Figure 2. This feature of Kenoyer '423 provides a form of control over the display of desktop information, which Becker teaches is desirable. *See, e.g.*, EX1009, Abstract (disclosing "Local and remote mouse and keyboard control of the computer imagery or other high-resolution videource"), ¶[0021] ("These capabilities are also combined with the appropriate control systems . . . by either manual or automatic control.").

130. Becker's converter 50, operating in view of the teachings of Kenoyer '423 as described in Ground 1 for claim 1, operates as the claimed background processing device.

131. This claim element is obvious for the same reasons discussed above for the corresponding elements of Claim 1. In particular, a POSA would understand that the "background device" in Becker is converter and encoder 50. A POSA would have understood that an MPEG encoded digitized video signal in this context would be output as a video bit stream at least based on the additional teachings of Kenoyer '423 as noted above for Claim 1.

4. “a terminal processing device, transferring the coded bit stream from the background processing device to a remote video communication terminal.”

132. The '288 Patent provides few disclosures of the “terminal processing device” other than language similar to the text of the claims. The device is illustrated in Figures 2 and 4 and generally described as the device that receives the encoded bit stream and transmits the stream to the remote site.

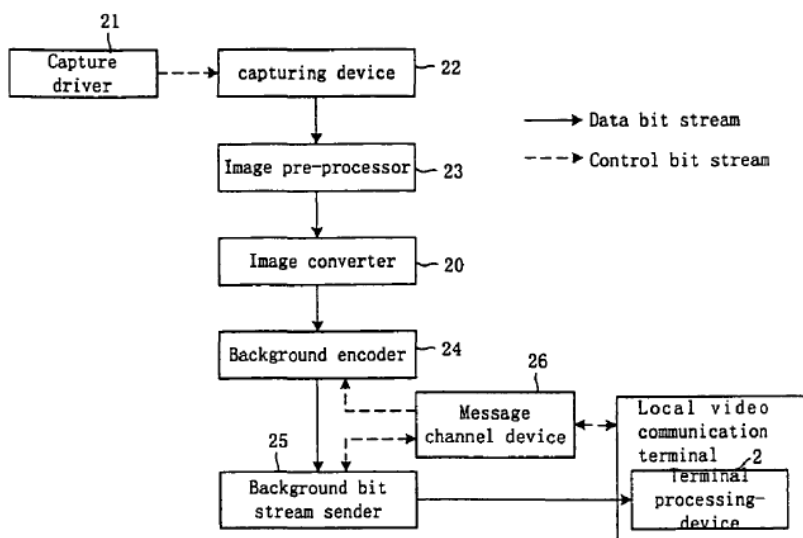


Fig. 2

Figure 2 of the '288 Patent (EX1001)

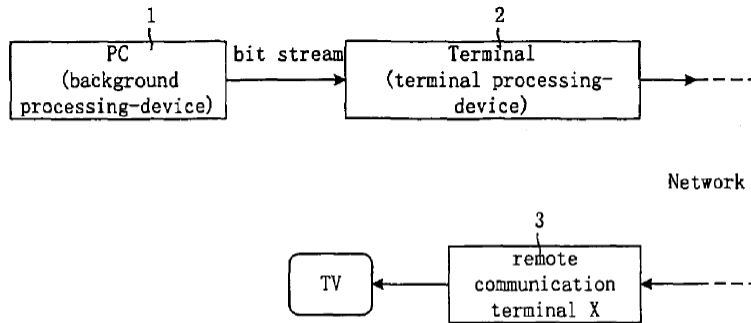
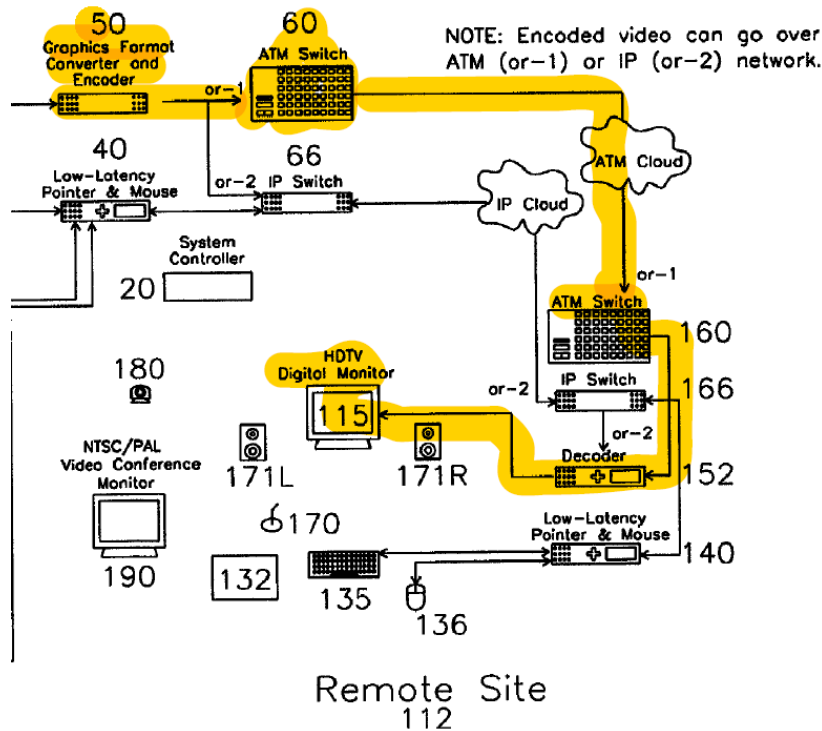


Fig. 4

Figure 4 of the '288 Patent (EX1001)

EX1001, Figures 2 and 4, Col. 4:6-10 (“[a]s shown in FIG. 4, the PC has a background processing device 1 to capture the desktop information of the PC, and then the captured desktop information is encoded and sent, through a network port, to the local video communication terminal that has a terminal processing-device 2, and then the local video communication terminal forwards the received coded bit stream to the remote video communication terminal 3”).

133. Becker teaches that the videoconferencing system transfers the coded video bit stream from converter and encoder 50 through ATM switch 60 to a remote ATM switch 160 which then sends the signal to a decoder so that the decoded signal can be displayed on an HDTV monitor or a computer. *See, e.g.,* EX1009, Figure 1 excerpt below (highlighted):



Annotated Excerpt of Figure 1 of Becker (EX1009)

134. Thus, the combination of Kenoyer '423 and Becker teaches all elements and limitations as arranged in Claim 9, and Claim 9 as a whole is obvious.

D. Dependent Claims 10, 12, and 14 are Obvious in view of Becker and Kenoyer '423.

135. Claim 10 recites as follows:

10. The apparatus according to claim 9, wherein the background processing device comprises:

a capture driver, sending out a triggering command;

a capturing device, receiving the triggering command from the capture driver and capturing the desktop information, then outputting;

an image converter, receiving the captured desktop information from the capturing device and converting into the format of the local video communication terminal, then outputting;

a background encoder, encoding output signal from said image converter into the coded bit stream;

a background bit stream sender, sending the coded bit stream to the terminal processing-device.

EX1001, Col. 5:43-6:8.

136. Figure 2 of the '288 Patent (pictured below) shows a capture driver 21 sending a triggering command to capturing device 22, which captures the desktop information and outputs the information to an image preprocessor. EX1001, Col. 5:52-64.

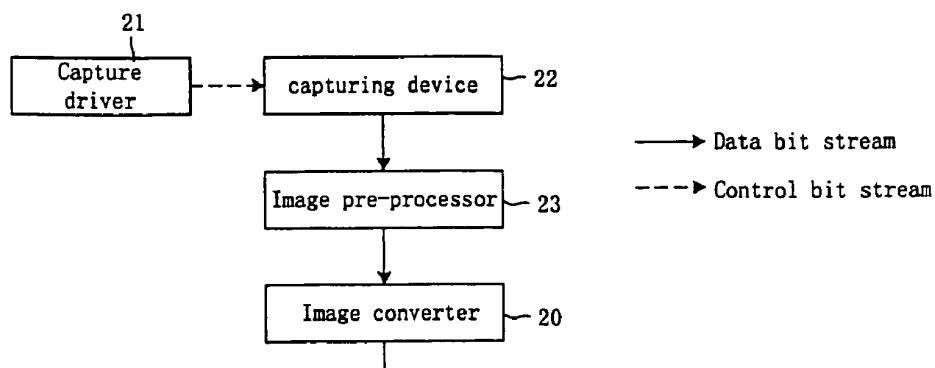


Figure 2 of the '288 Patent (EX1001)

137. Figure 4 of the '288 Patent (pictured below) shows the PC as the “background processing-device,” which sends a bit stream to the terminal/“terminal processing device.” EX1001. A POSA would understand that the PC is at least a possible background device as contemplated by the '288 Patent.

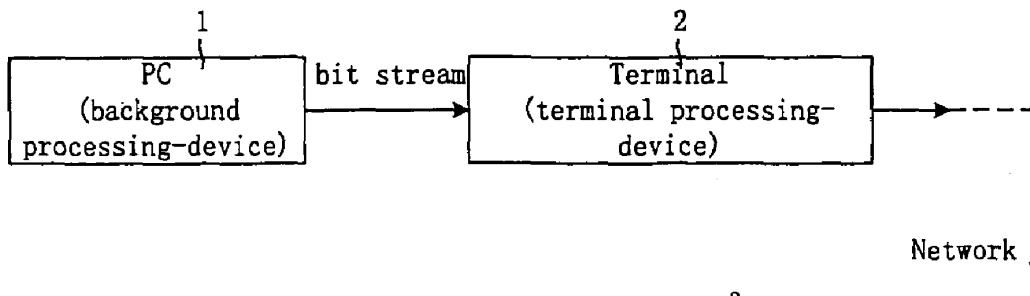
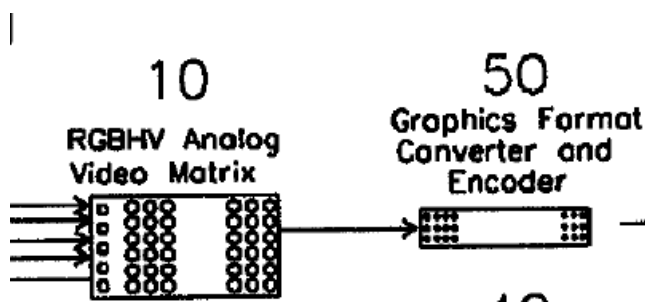


Figure 4 of the '288 Patent (EX1001)

138. Becker teaches capturing desktop information of the PC after receiving a triggering command and converting a PC format of the desktop information into a format of a local video communication channel. “The present invention captures the graphics signals being output from the computer in their raw video format (the format that is sent to the computer monitor for viewing).” EX1009, ¶[0020]. “In the preferred embodiment of the method, the analog RGBHV signals are converted to serial digital high-definition television, SDI-HDTV . . .” *Id.*, ¶[0071]; *see also id.*, Figure 1 (excerpt below):



Excerpt of Figure 1 of Becker (EX1009)

139. Kenoyer '423 also teaches a capture driver capturing desktop information using a triggering command. “Dual stream communication provides for

integration of a coupled personal computer or a laptop computer with an existing videoconferencing system.” EX1010, Col. 2:48-50. Kenoyer ’423 teaches a card 100 installed on a computer which when an application is launched displays an interface including Show PC button 210. “Clicking on Show PC button 210 will send a presentation loaded into a laptop or personal computer from the videoconferencing System to a remote site.” *Id.*, Col. 3:1-13. “[A] method 500 includes a step 510, in which an image being presented is monitored for changes or movement. More particularly, step 510 monitors changes in macroblocks. If a step 520 change or movement is detected in the macroblock, then in a step 530 pixels of the changed macroblock are locally processed, coded, and, in a step 540, transmitted.” *Id.*, Col. 4:65-5:1.

140. A POSA would have found it obvious to modify Becker with Kenoyer ’423’s teachings of a triggering command for capturing desktop information. Becker teaches that it beneficially uses technologies that minimize the amount of bandwidth required to transmit video information. EX1009, ¶[0070]. As discussed in prior grounds regarding the capture/trigger function, a POSA would be motivated to use Kenoyer’s triggering command to reduce the bandwidth used. The triggering feature is readily computer-implemented compatibly with the remaining Becker image processing functions, also implemented in software or firm ware.

141. Becker also teaches an image converter receiving the captured desktop information from the capturing device and converting the format of the captured desktop information into a format of a local video communication terminal. *See* Figure 1 excerpt above (including graphics format converter and encoder 50). “[T]he analog RGBHV signals are converted to serial digital high-definition television, SDI-HDTV. . . .” EX1009, ¶[0071].

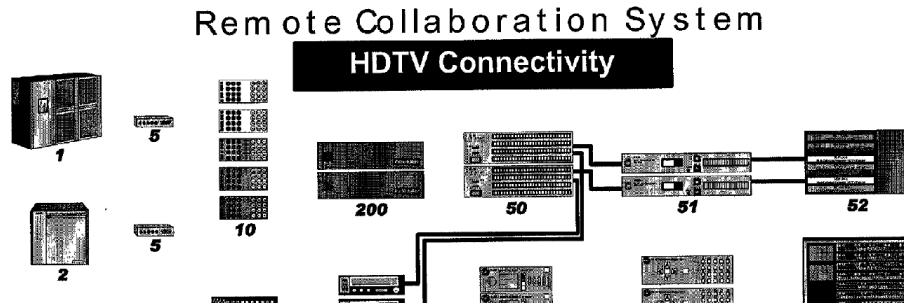
142. Kenoyer ’423 also teaches converting a PC format of the captured desktop information into a format of a local video communication terminal. It teaches capturing the desktop information through a Show PC command. EX1010, Col. 3:26-28. “Annex D provides a sub-sampling of image pixels to generate four Common Intermediate Format (CIF) 352x280 pixel sub-images. . . .” *Id.*, Col. 4:39-41. This format conversion also converts the desktop image into four separate Common Intermediate Format (CIF) sub-images, consistent with the conversion provided in the ’288 Patent.

143. Kenoyer ’423 teaches a background encoder encoding the output signal from the image converter into coded bit stream. “H.261 is a commonly used protocol that specifies encoding and decoding of real-time video data to support video transmission applications. . . . Annex D relates to the protocol for transmitting digital images within the H.261 protocol. . . . Annex D advantageously provides for digital processing of image files and is particularly useful in the processing of static

images.” EX1010, Col. 4:31-46. H.261 encoding, with Annex D, is expressly listed in the ’288 Patent as an example encoding protocol.

144. Kenoyer ’423 refers to “streaming” the compressed image to a videoconferencing system. EX1010, Col. 3:37-39. It also refers to transmitted image information in the form of bits. *Id.*, Col. 5:10-14. It describes H.261 encoding, which a POSA would understand is routinely transmitted in video bit stream format.

145. Becker also teaches a background encoder and bit stream sender for encoding the output signal from the image converter into the coded bit stream and sending it to the terminal processing-device. Becker teaches encoding is performed in conjunction with the graphics format converter and encoder (50) and MPEG compression device (52). MPEG compression device 52 receives the signal from A/D converter 51 and is used “for coding and compression” and operates as part of the claimed background device. EX1009, ¶[0076]; Figure 4-C (excerpt below showing encoder 50, A/D converter 51, and MPEG compression device 52). “[T]he SD-HDTV signal coming from the A/D converter (51) is then sent to the MPEG compression device (52) for encoding and compression.” *Id.*, ¶[0076]; *see, e.g.*, Figure 4-C (excerpt below):



Excerpt of Figure 4-C of Becker (EX1009)

See also *id.*, Figure 1. MPEG was a well-known video encoding protocol comparable to H.261, which is listed in the '288 Patent as an example video encoding protocol. EX1001, Col. 3:20-28; see also EX1009, ¶¶[0071]-[0072] (teaching encoding the RGBHV signals directly or after conversion to serial digital high-definition television.).

146. Becker teaches an encoder and sender for encoding information and sending it in video bit stream format. “[T]he analog HDTV signal is sent into an A/D converter (51). “The A/D converter converts the analog HDTV Signal into a Serial digital Stream, SDI, of data.” EX1009, ¶[0074]. “In one embodiment, the encoded HDTV signals from the MPEG compression device (52) are then sent to an ATM computer network switch (60), Fig. 4-D. From there, the information is transmitted across communication lines to a receiving ATM switch 160 at the remote location (90).” *Id.*, ¶[0077]. Paragraph [0071] of Becker similarly describes that the “signals can be compressed (encoded) using, for example, an MPEG compression

algorithm.” A POSA would understand that such MPEG conversion would encode the information in a video bit stream format.

147. Becker teaches the components of Claim 10 including a background encoder transmitting the encoding information in bit stream format, and a POSA would be motivated to modify Becker’s teachings to incorporate into the background encoder and background bit stream sender the teachings of Kenoyer ’423 to send the H.261 encoded bit stream information generated from the image converter. A POSA would be motivated to use H.261 because it was a standard encoding protocol adapted specifically for use in videoconferencing environments. Becker in view of Kenoyer ’423 thus renders obvious Claim 10.

148. Claim 12 depends from Claim 9 and further recites that the terminal processing device “comprises a terminal bit stream transmitter that transfers the coded bit stream to the remote video communications terminal.” EX1001, Col. 6:14-17. Claim 12 largely corresponds to the “background bit stream sender” element of Claim 10, and that discussion immediately above also shows Claim 12 is obvious.

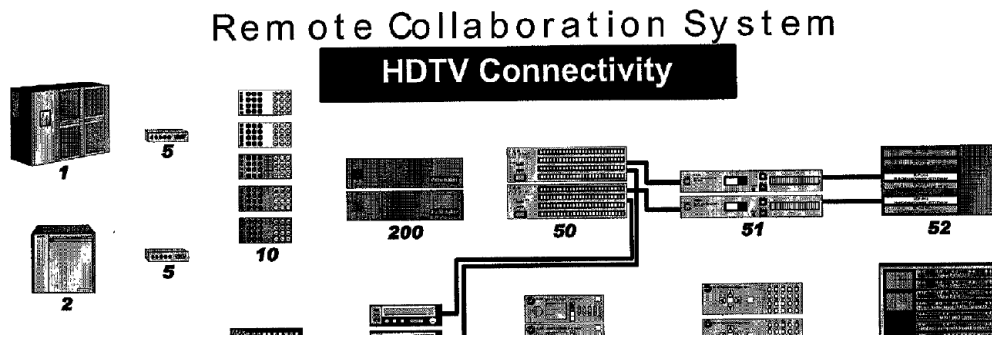
149. Becker further teaches a terminal processing device including a transmitter that transfers the coded video bit stream to a local video communication terminal in the form of ATM switch 160 at the remote location (90). EX1009, ¶[0077]. Becker discusses “Computer video Routed Outside the Local Facility” in a section which discusses Figures 4-B and 4-C. *Id.*, ¶¶[0068]-[0080]. “In one

embodiment, the encoded HDTV signals from the MPEG compression device (52) are then sent to an ATM computer network switch (60), FIG. 4-D.” *Id.*, ¶[0077]; *see also id.*, ¶[0200] (stating that the local ATM computer network switch (60) may contain “video codecs”). Becker in view of Kenoyer ’423 thus renders Claim 12 obvious.

150. Claim 14 depends from Claim 12 and recites that the terminal processing device “further comprises a terminal encoder that encodes a local image and then outputs to the terminal bit stream transmitter.” EX1001, Col. 6:23-26. Becker teaches a terminal processing device (ATM switch 60) that encodes a local image and outputs to the terminal bit stream transmitter. “Cameras (80, 81) at the local site 12 are connected to video codecs, which can be contained within the ATM switch (60). The video codecs are used to compress the NTSC(PAL) video coming from the cameras to use less bandwidth for transmission to the remote site(s). The encoded NTSC(PAL) camera information is sent over the telecommunications network and is received at the remote site via a video codec at the remote site, which can be contained within the ATM switch (160).” EX1009, ¶[0199].

151. MPEG was a well-known video encoding protocol comparable to H.261. Becker taught its use for the camera signals. *See* EX1009, ¶[0073]-[0075] (teaching encoding an analog video signal (e.g. NTSC) using MPEG compression).

152. Becker teaches an encoder and sender at local ATM switch 60 for encoding local image information and sending it in video bit stream format, for example to a receiving ATM switch 160 at the remote location (90). EX1009, ¶¶[0077]. Paragraph [0071] of Becker similarly describes that the “signals can be compressed (encoded) using, for example, an MPEG compression algorithm.” Id., ¶¶[0071], Figure 4-C (excerpt below):



Excerpt of Figure 4-C of Becker (EX1009)

153. Becker teaches an encoder and sender for encoding information and sending it in video bit stream format to a receiving ATM switch 160 at the remote location (90). EX1009, ¶¶[0077]. Paragraph [0071] of Becker similarly describes that the “signals can be compressed (encoded) using, for example, an MPEG compression algorithm.” Id., ¶¶[0071]. For all of these reasons, Claim 14 is obvious over Becker in view of Kenoyer ’423.

E. Claim 1 is Obvious over Kenoyer in view of Clapp and *Video Compression Techniques*, and it is also obvious over Kenoyer in view of Clapp and Shaw.

154. During prosecution of the '288 Patent, the Examiner cited Kenoyer in combination with Clapp as grounds for an obviousness rejection of Claim 1. EX1002 at 134-136. In response, the Applicant distinguished Kenoyer from Claim 1 purportedly because (1) Kenoyer did not disclose “receiving a triggering command,” (2) Kenoyer did not disclose the limitation: “encoding the converted desktop format in a mode ensured by the local video communication terminal,” and (3) Kenoyer therefore did not disclose that what was sent to, and received by, the local video communication terminal was the required coded bit stream. EX1002 at 160.

155. With respect to argument (1) above, the Examiner combined Kenoyer with Clapp to show the triggering command element of Claim 1, and the Applicant did not dispute this showing. EX1002 at 135, 160. As explained below, Kenoyer combined with Clapp and either *Video Compression Techniques* or Shaw provides the all elements of Claim 1 as arranged therein, and it would have been obvious to modify Kenoyer with Clapp’s and *Video Compression Techniques* or Shaw’s teachings in the manner claimed.

1. “A method for transferring desktop information of a PC to a video communication terminal comprising:”

156. The Examiner properly found that (1) the “alternate image source” 122 in Kenoyer (a laptop computer) in Fig. 1 disclosed the desktop information of a PC, (2) the “videoconference system” 120 of Kenoyer disclosed the video communication terminal, and (3) Figures 1-2 illustrated transferring desktop information of a PC to a video communication terminal. EX1002 at 134. Indeed, Figure 1 shows the PC 122 desktop display being transferred to a local videoconference system 120, via a videoconference accessory module 110, for transmission to remote site 160:

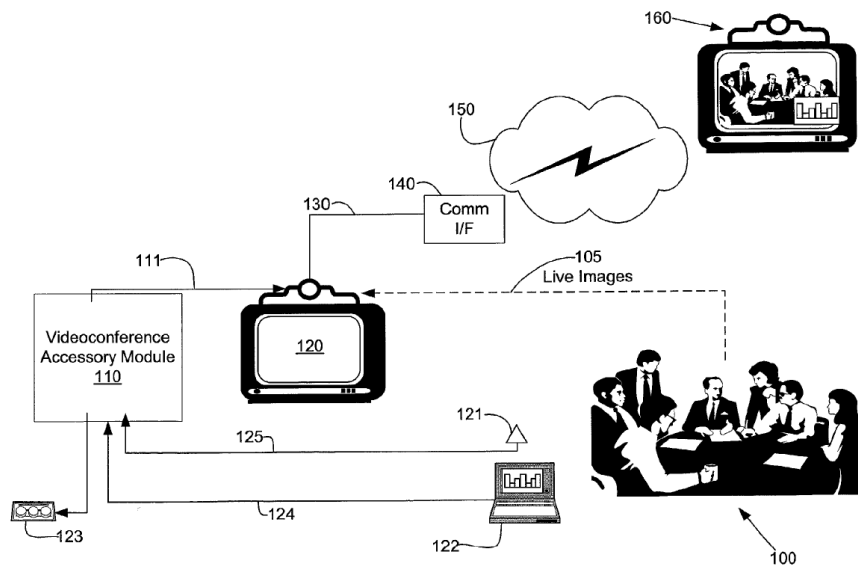


FIG. 1

Figure 1 of Kenoyer (EX1005)

EX1005, ¶[0020], Figures 1-2. I note that the alternate embodiment in Kenoyer also discloses these same elements, with the primary difference being that the video

conference accessory module 110 is realized as a PCMCIA card that is integrated into the PC in the alternate embodiment. EX1005, Figure 4.

2. “Capturing desktop information of the PC after receiving a triggering command, and converting a PC format of the desktop information into a format of a local video communication channel”

157. Clapp teaches a triggering command as claimed at column 11, lines 47-64, where it discusses sensing a “local draw command.” Such commands for capturing desktop images were known and standard in the industry before 2002. Moreover, it would have been obvious to so modify Kenoyer in view of Clapp. Kenoyer teaches that it is desirable to provide users the ability to communicate visibly and audibly with remote users to have meetings in real time. EX1005, ¶[0019]. A POSA would understand Kenoyer thus provides a motive for modifying Kenoyer’s teachings to allow control over when desktop information is captured for transmitting to others, and Clapp provides that teaching.

158. It was obvious to combine with Kenoyer with Clapp as both references taught videoconferencing systems and Clapp’s triggering command for capturing desktop information would facilitate user control in Kenoyer’s system.

159. The Examiner found Clapp’s control unit taught the triggering command. EX1002 at 135 (citing EX1006, Col. 11:47-64). Further, the Examiner concluded it was obvious to combine Kenoyer with Clapp because both taught videoconferencing systems and a POSA would use Clapp’s triggering command for

capturing desktop information to facilitate user control in Kenoyer's system. *Id.* The Patent Owner did not dispute that Clapp taught the triggering command element or that it was obvious to modify Kenoyer using Clapp's teachings. I agree with the Examiner that these passages disclose a triggering command as claimed.

160. Beyond the fact that Kenoyer and Clapp are directed to videoconferencing, an additional motivation to combine comes from the fact that Kenoyer is concerned with providing users with the ability to communicate with remote users to have real-time meetings. EX1005, ¶[0019]. A POSA would be motivated by Kenoyer's teachings to invoke Clapp's teachings of a triggering command which provides the user with control of when desktop information is captured for sharing with other users, which enhances the ability to communicate with other users.

161. With respect to "converting a PC format into a format of a local video communication terminal," Patent Owner did not dispute the finding of this limitation in Kenoyer and in fact acknowledged that "Kenoyer appears to convert images to YUV format from RGB format." EX1002 at 160.

162. Kenoyer states that the FPGA converts the RGB image into YUV format. EX1005, ¶[0028]. The FPGA is a Field Programmable Gate Array which is programmed to convert desktop information into YUV data. *Id.*, ¶[0029].

163. Indeed, both the standalone Video Accessory Module 110 mode and the PCMCIA videoconference interface card 410 mode in Kenoyer disclose that the RGB images from the laptop display (the alternative image source) are captured and converted into YUV format. EX1005, ¶¶[0028], [0035]. Kenoyer further teaches that its local video communication terminal, the videoconference system/unit 120, receives YUV format data. EX1005, ¶[0030] (“After the compression and conversion of the alternate image source signal 124 to YUV format, the videoconference accessory module 110 outputs the processed image to the videoconference unit 120”); *see also id.*, ¶[0035], Figures 1 and 4. Thus, the YUV format into which the desktop information in Kenoyer is converted is a format that is processable by and thus compatible with the local video communication terminal. Accordingly, Kenoyer discloses “converting a PC format into a format of a local video communication terminal” as it discloses converting an image of the information displayed on the monitor of the PC from an RGB format into a format (YUV) that is processable by and compatible with a local video communication terminal.

3. “Encoding the converted desktop format in a mode ensured by the local video communications terminal”

164. Kenoyer discloses that “[t]o achieve simultaneous transmission of the normal conference video stream 231 with the alternate video stream 232, total bandwidth of the video subchannel 230 must be conserved and optimized.” EX1005,

¶[0024]. Indeed, Kenoyer argues that its goal of combining the display of an alternate image source with live videoconferencing data was not previously achievable “due to bandwidth limitations.” *Id.*, ¶[0006].

165. Given these express concerns about bandwidth usage, as well as the general desire in videoconferencing to limit bandwidth, a POSA would have known that data encoding and compression was desirable. Moreover, Kenoyer teaches that its FPGA 320 works in conjunction with processor 330 and memory 340 to “enhance throughput and performance of the video conversion and compression process.” EX1005, ¶[0029]. These statements would motivate a POSA to look for teachings regarding encoding and compressing video signals that would enhance throughput and performance.

166. A POSA also would have been motivated to use videoconferencing media video compression/encoding standards such as H.261 at the video accessory module of Kenoyer at least because the module has signal processing capabilities (FPGA 320 and Processor 330), as shown for example in Figure 3:

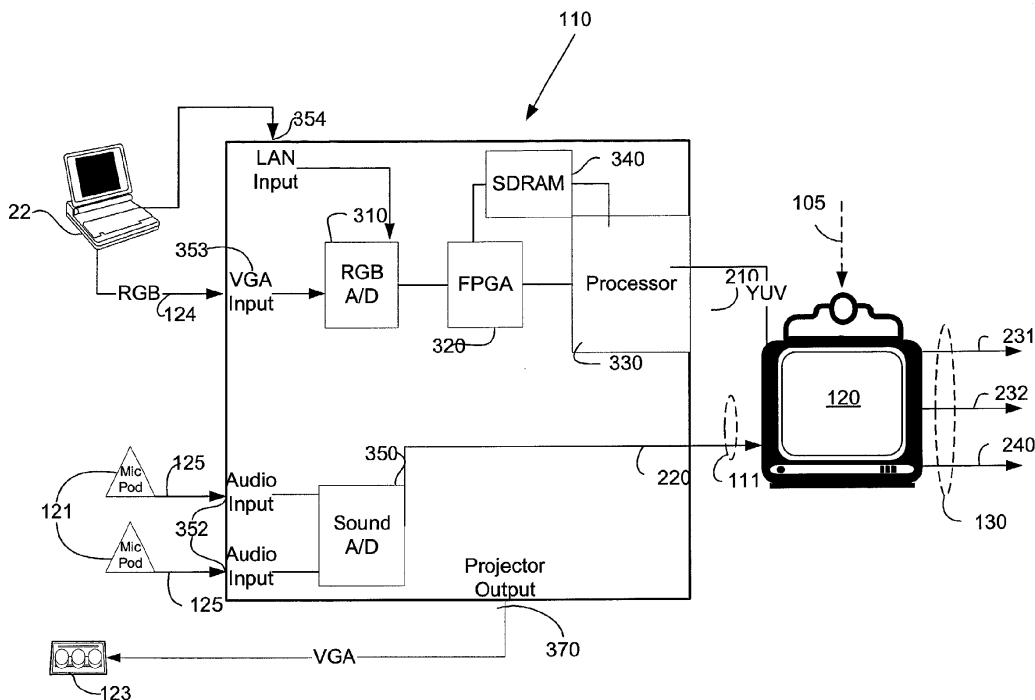


Figure 3 of Kenoyer (EX1005)

167. A POSA would be motivated to use the H.261 protocol to perform these steps, as the H.261 protocol is designed to compress video signals and enhance transmission and throughput. Shaw notes H.261 is “commonly applied” for use with Common Intermediate Format (CIF) and Quarter CIF (QCIF) “to facilitate the transport of video information over the telecommunications network.” EX1007, Col. 11:6-13. It notes that use of a standard such as H.261 provides “ease of interface.” *Id.*, Col. 12:34-40. Such standardized encoding would be readily known to and implemented by a POSA. H.261 is also noted for providing motion estimation. *Id.*, Cols. 18:34-48, 29:43-44.

168. The disclosures of *Video Compression Techniques* would have made it obvious to a POSA to encode and further compress the converted desktop format data of Kenoyer using a video encoding standard such as the H.261 or H.263 standards to enhance throughput and performance. *Video Compression Techniques* describes the H.261 standard. EX1008. A POSA would have been motivated to use the H.261 protocol to perform these steps, as the H.261 protocol was known to be designed to compress such video signals to enhance transmission by reducing the size/rate of bit-stream video transmissions. H.261 was also a known common protocol well-suited for videoconferencing applications and thus likely to improve performance by providing a protocol likely recognized by a variety of other devices.

169. *Video Compression Techniques* explains how H.261 applies a quantization factor to adapt the bit rate to the bandwidth available in the network. EX1008 at 60. H.261 also compresses the signal through entropy coding. *Id.* at 60-61. H.261 is particularly suited to compressing video because it uses motion estimation and motion prediction to efficiently encode the motion in video for effective transmission. *Id.* at 61-62. Indeed, H.261 is mentioned in the '288 Patent as an example known encoding protocol for use in the recited method. EX1001, Col. 3:20-28.

170. A POSA would further be motivated to encode the desktop information signal into a video bit stream before sending the signal to the local video

Table 5-6: Source image formats of H.261

	CIF	QCIF
Y	352×288	176×144
Cb, Cr	176×144	88×72

172. The Table above refers to CIF and QCIF formatting. Shaw relatedly notes H.261 is “commonly applied” for use with Common Intermediate Format (CIF) and Quarter CIF (QCIF) “to facilitate the transport of video information over the telecommunications network.” EX1007, Col. 11:6-13. It notes that use of a standard such as H.261 provides “ease of interface.” *Id.*, Col. 12:34-40. Such standardized encoding would be readily known to and implemented by a POSA. H.261 is also noted for providing motion estimation. *Id.*, Cols. 18:34-48, 29:43-44. All of these factors provide motivations for a POSA to use H.261 encoding with Kenoyer.

173. Kenoyer further teaches the use of hardware in the form of an FPGA 320, which would have provided an obvious and ready basis for implementing H.261 encoding. EX1005, ¶[0029], Figure 3. Kenoyer further teaches that the FPGA 320 and processor 330 work in conjunction to enhance throughput and performance on a video image signal which has been converted into YUV format. *Id.*, ¶[0028]. H.261 encoding was a known encoding technique readily adaptable for

implementation at FPGA 320 and processor 330 as an algorithm with YUV formatted video.

174. Shaw is likewise directed to teleconferencing, including videoconferencing. EX1007, Cols. 1:25-26 (“[t]his invention is dedicated to the specific application of teleconferencing”), 2:40-41 (“[s]ome of the applications examples are distance education (teaching and learning), teleconferencing, messaging, video-phone, video games, cable TV decoders, and HDTV”). Shaw teaches a bandwidth controller that controls the compression ratio of a video codec to generate a coded video bit stream at rate that is compatible with the network. *Id.*, Cols. 8:24-33; 23:43-24:2. Shaw discloses a simplified video encoder for this purpose. *Id.*, Cols. 23:47-24:9. Shaw makes liberal reference to the H.261 coding standard (*see, e.g., id.*, Col. 18:36-39, referencing motion estimation/prediction within the context of the H.261 standard). Nonetheless, in light of Shaw’s reliance on the H.261 standard and the structure disclosed for Shaw’s encoder, a POSA would have understood that Shaw is either relying on an H.261 encoder or a substantially similar video encoder.

175. Shaw’s encoder is illustrated in Figure 28:

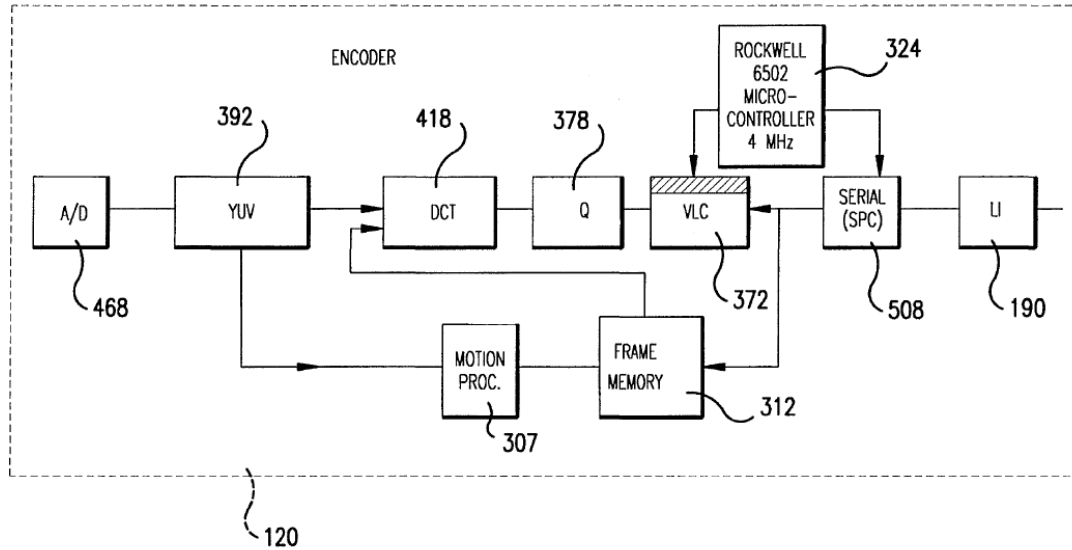


FIG. 28

Figure 28 of Shaw (EX1007)

Figure 28 of Shaw teaches implementing an encoder which receives the converted YUV signal (e.g. as taught by Kenoyer). This is shown by blocks 468 and 392 in Figure 28 of Shaw. EX1007, Col. 23:51-52 (“[t]he digital RGB 389 signals can be further converted into a digital YUV 392 format employing a color space convertor device”). Shaw teaches encoding the converted signal it via a protocol such as H.261. Figure 28 is an example system for performing the encoding. The system performs a Discrete Cosine Transform (DCT 418) on the YUV signal, followed by a quantization step (Q 378) followed by a Variable Length Coder (“entropy”) encoder (VLC 372) as well as feedback of frame memory (312) to provide motion prediction/compensation between frames as is characteristic of H.261 encoding.

176. Motion estimation/prediction is performed (block 307 in Figure 28 of Shaw). EX1007. Further encoding occurs through a DCT (Discrete Cosine

Transform) transformation, quantizing the DCT coefficients, and run-length/variable-length encoding the results. *See id.*, Figure 28 (blocks 418, 378 and 372). Finally, the data is converted into a video bit-stream for transmission. *See id.* (block 508). Shaw contains additional elements in its block diagram, but a POSA would recognize that Shaw was simply providing more implementation detail in its diagram.

177. Excerpts from Shaw shown below (highlighted) confirm that it teaches outputting the video information in bit stream format:

SIMPLIFIED VIDEO ENCODER FUNCTIONAL
MODEL

4

As shown in FIG. 28, we illustrate a simplified block diagram for a general purpose video encoder **120** subsystem.

EX1007, Col. 23:43-47.

Once the coefficients are quantized at different resolution, the final **bit stream** can further be compacted using variable length coding (VLC) **372**. VLC **372** is commonly applied to apply shorter length code for more frequent occurred **bit streams**. The final compacted **bit stream** is first converted

from bit parallel into bit serial form using a parallel-to-serial converter **508**. A line interface **190** can further convert the video from digital into analog format.

Id., Cols. 23:63-24:2 (emphasis added).

178. Accordingly, a POSA would understand Shaw to teach either a H.261 encoder or a substantially similar encoder that was designed to compress and encode video data in a teleconferencing or videoconferencing system. Given the disclosures

of the use of H.261 in Shaw, it would have been obvious to a POSA modify Shaw's encoder to make it conform to the H.261 standard referenced in Shaw. The H.261 encoding process was well known to POSAs and was well known to have been widely used for its intended purpose, namely encoding video data for applications such as videoconferencing.

179. A POSA would have been motivated to modify Kenoyer with Shaw's or *Video Compression's* encoder to optimize the use of bandwidth and standardize the encoded signal so that it was readily recognized and processable by other videoconferencing devices. Kenoyer includes a number of statements indicating the desirability of reducing bandwidth load. *See, e.g.*, EX1005, ¶¶[0011], [0013], [0024]. Further, Kenoyer's teaching of using an FPGA and processor to process YUV image data provides an obvious location and application for the known techniques of Shaw's and *Video Compression's* encoders, and it shows those encoders would be readily implemented in the Kenoyer system with an expectation of favorable results. *Id.*, ¶[0029], Figure 3. All of these prior systems contemplate raw pixel data to be reformatted into YUV format, and use a substantially similar, if not identical mechanism for encoding YUV data into a video bitstream. Accordingly, the combination of Kenoyer and Shaw (and the knowledge of a POSA) renders obvious "encoding the converted desktop format in a mode ensured by the local video communication terminal," as the combination discloses encoding the

converted image into a video bit stream that is recognizable by the local video communication terminal, such as H.261 or a substantially similar video bit stream.

4. “Sending the coded bit stream to the local video communication terminal”

180. Kenoyer in combination with *Video Compression Techniques* or Shaw teaches and renders obvious this element. In responding to the Examiner’s rejection based on Kenoyer, the Applicant acknowledged that the “signals” in Kenoyer were sent to, and received by, the local video communication terminal. Indeed, two modes of Kenoyer disclose sending the converted data to the local video communication terminal (i.e., the videoconferencing system/unit 120). See EX1005, ¶[0030] (“[A]fter the compression and conversion of the alternate image source signal 124 to YUV format, the videoconference accessory module 110 outputs the processed image to the videoconference unit 120”), ¶[0035] (“[i]mages from the alternate image source 122 are digitized, compressed, and converted into YUV format by the videoconference interface card 410 using similar technology as described above with reference to the videoconference accessory module 110”).

181. A POSA would recognize in Kenoyer that the sent signal, that is the output of the FPGA and processor, would necessarily be in digital bit stream format, commonly used before the filing date of the ’288 Patent for digital video data. As discussed above concerning H.261 processing of signals as taught by *Video Compression Techniques* and Shaw, a POSA could have simply used a standard

H.261 codec (or other well-known and widely used standard video codec) to encode the YUV data and transmit it as an encoded bit stream. Such conversion was standard practice in videoconferencing systems given the benefits of using such a codec, confirming that a POSA would have been motivated to use such a codec before the '288 Patent was filed.

182. Moreover, Shaw expressly teaches sending YUV signals in the form of coded video bit stream. Shaw teaches that the encoder “can capture and compress the image or video information for ease of storage and transmission.” EX1007, Col. 12:18-20. As shown above, Shaw’s teachings are readily implemented in the Kenoyer system with predictable results, and a POSA would be motivated to combine these teaches as claimed given the benefits of using the H.261 standard described in Shaw.

183. *Video Compression*’s relevant teachings reinforce Shaw’s regarding the benefits of encoding in H.261 and a POSA would similarly be motivated to adopt *Video Compression*’s teachings to encode in H.261 bit-stream. *Video Compression* explains H.261 is particularly suited to compressing video used for videoconferencing because it invokes motion compensation prediction to efficiently transmit information about motion video in real time. EX1008 at 59, 61-62. *Video Compression* also specifies that H.261 uses “coded bit stream.” *Id.* at 59.

184. It also would have been obvious for a POSA to encode and decode the signal of Kenoyer at the videoconference accessory module 110 using the H.261 standard in view of *Video Compression*. Kenoyer teaches the use of an FPGA 320 and processor 330 which would have provided an obvious and ready means of implementing H.261 compression/encoding for Kenoyer's signals. EX1005, ¶[0029], Figure 3. For reasons similar to those provided above regarding modifying Kenoyer with Shaw, it would have been obvious to a POSA to modify Kenoyer with *Video Compression* to perform the encoding step of Claim 1.

185. Accordingly, the combination of Kenoyer with Shaw or *Video Compression Techniques* would have taught a POSA that it would have been obvious that the coded bit stream resulting from using an H.261 or H.261-like encoder in Kenoyer would result in the required coded video bit stream (e.g., an H.261 coded bit-stream) being sent to the local video communication terminal (videoconferencing system/unit 120).

5. “Receiving the coded bit stream by the local video communication terminal and transferring it to a remote video communication terminal through transmission channel after processing”

186. The combination of Kenoyer and *Video Compression Techniques*, or the combination of Kenoyer and Shaw, teaches this element. The analysis above, showing either Kenoyer and *Video Compression Techniques* or Kenoyer and Shaw teach sending an encoded bit stream to a local video communication terminal, would

also make it obvious to a POSA that the local video communication terminal in Kenoyer (videoconferencing system/unit 120) would receive the coded video bit stream for the purpose of transferring it after processing to a remote video communication terminal. (I note that the '288 Patent acknowledges that transferring videoconferencing data to the remote terminal “after processing” was in the prior art. EX1001, Col. 1:30-36.)

187. In addition, Kenoyer discloses that what is received by the local video communication terminal (videoconferencing system/unit 120) is sent to a communications interface and transmitted to a remote video communication terminal (a remote videoconferencing system) after processing to form an alternate video stream. EX1005, Figure 2, ¶[0030] (“the alternate video component 210 is output as the alternate video stream 232 of the carrier channel 130 by the videoconference unit 120 and transmitted to the communications interface”), Figure 4, ¶[0035] (“[t]he videoconference unit 120 then sends these signals to the communications interface 140 by way of the alternate video stream 232 (FIG. 2) of the carrier channel 130. Subsequently, the communications interface 140 transmits the data on the carrier channel 130 to the remote site 160”), ¶[0032] (“a videoconference system located at remote site 160 will receive the carrier channel 130 sent by the communications interface 140 of the videoconference system 120”); see also *id.*, ¶¶[0032], [0035] (teaching that videoconference unit 120 “sends these

signals to the communications interface 140 by way of the alternate video stream 232 (FIG. 2) of the carrier channel 130. Subsequently, the communications interface 140 transmits the data on the carrier channel 130 to the remote site 160.”) Figure 1 illustrates the overall data-flows, including the transmission from the local device 120 to a remote device 160:

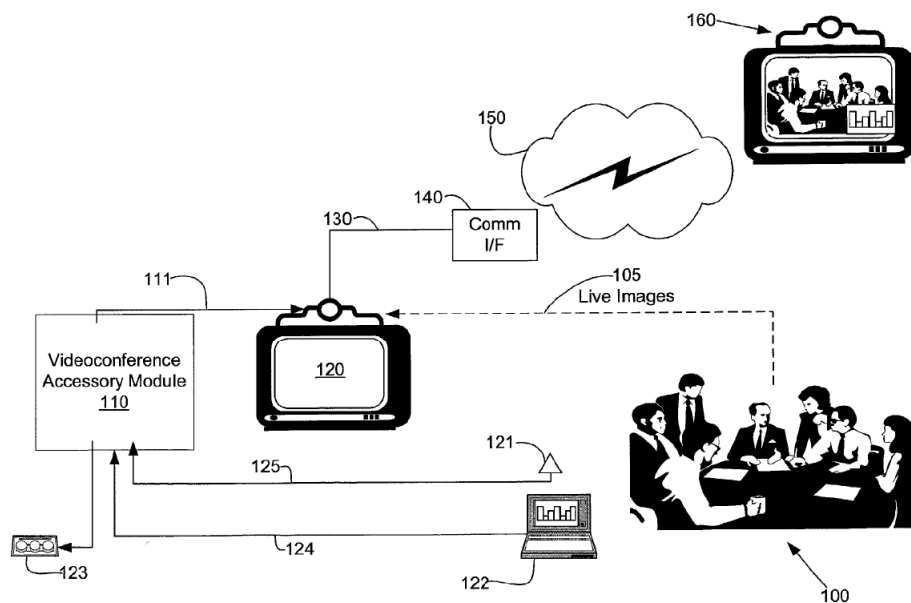


FIG. 1

Figure 1 of Kenoyer (EX1005)

188. Kenoyer further teaches that videoconference unit 120 “sends these signals to the communications interface 140 by way of the alternate video stream 232 (FIG. 2) of the carrier channel 130. Subsequently, the communications interface 140 transmits the data on the carrier channel 130 to the remote site 160.” EX1005, ¶[0035]; *see also id.*, ¶[0032] (“a videoconference system located at remote site 160 will receive the carrier channel 130 sent by the communications interface 140 of the

videoconference system 120”). Kenoyer in view of Shaw or *Video Compression* teaches a local video communication terminal receiving the encoded video stream signal, processing it, and sending it in bit stream format to a remote video communication terminal.

189. In summary, the combination of steps as recited and arranged in method of Claim 1 would have been obvious to a POSA in and before 2002 over Kenoyer in view of Clapp and *Video Compression Techniques*, or over Kenoyer in view of Clapp and Shaw.

F. Claims 2 and 5-7 are also Obvious in view of Kenoyer and Clapp and further in view of Shaw and *Video Compression Techniques*.

190. Dependent Claim 2 of the '288 Patent recites:

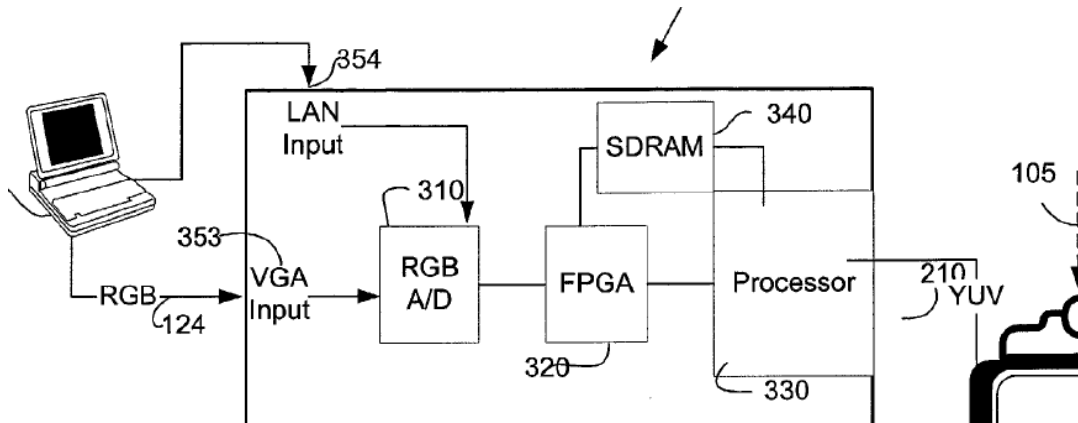
2. The method according to claim 1, wherein Step a [of Claim 1] further comprises: pre-processing the desktop information captured.

EX1001, Col. 5:5-6. The '288 Patent specification contains a single, one-sentence description of “pre-processing”: “Wherein Step a further comprises, pre-processing the captured desktop information. The step of pre-processing comprises, removing the unnecessary desktop information, and then filtering and editing the rest of desktop information.” *Id.*, Col. 2:14-17.

191. In addition, Figure 1 of the '288 Patent discloses “pre-processing” in the context of performing a format conversion of the desktop information. EX1001. This is described in the specification as “making various pre-process for the desktop

information based on requirement.” *Id.*, Col. 3:16-17. From these disclosures it is not clear whether the claimed pre-processing occurs as pre-processing to (or as part of) the “capturing” and/or the “converting” steps of step (a) of Claim 1, or something else.

192. Within the context of videoconferencing, the prior art is replete with references to pre-processing information as part of the image processing pipeline. For example, during prosecution, the Examiner found that Paragraph [0035] of Kenoyer (EX1005) teaches the pre-processing of Claim 2. EX1002 at 135. Paragraph [0035] of Kenoyer teaches that the images from the alternative image source are “digitized, compressed, and converted to YUV format.” EX1005. Paragraph [0035] also indicates that the processing of images uses techniques similar to those shown in the Figure 3 embodiment of Kenoyer, which includes an A/D converter 310, FPGA 320, and processor for YUV conversion 330. *Id.* These components are shown in the excerpt of Figure 3 below, which shows A/D converter 310 and FPGA 320, and processor 330 processing images as part of the process of converting RGB data to a YUV format:



Excerpt of Figure 3 of Kenoyer (EX1005)

193. As indicated above, “converted to YUV format” corresponds to the “converting” of step a of Claim 1. The Examiner thus correctly stated that Kenoyer teaches Claim 2 at least by digitizing and compressing the information in the YUV conversion process. Indeed, it appears that the Applicant did not dispute the Examiner’s conclusion that Kenoyer teaches the elements of Claim 2.

194. Clapp also discloses pre-processing as claimed. Clapp teaches that a local window menu 600 is provided from which a user selects a window for sharing with a remote conferencing site. EX1006, Col. 11:19-45. This teaches removing unnecessary information which the ’288 Patent indicates is at least one form of the claimed pre-processing. It would have been obvious to a POSA to modify Kenoyer to provide this option as Kenoyer teaches it is desirable to provide user control over what is shared and teaches it is desirable to reduce bandwidth usage, which would motivate a POSA to use Clapp’s menu to allow a user to only select a desired window for sharing.

195. Dependent Claim 5 of the '288 Patent recites “the desktop information in Step a [of Claim 1] is captured from a video memory of the PC.” EX1001, Col. 5:17-18. The Examiner found that Paragraph [0036] and Figure 4, interface 410, of Kenoyer taught this feature. EX1002 at 135-136. Paragraph [0036] teaches that PCMCIA interface 410 includes “flash memory” and software to make the card “operable with the alternate image source 122.” EX1005. Figure 4 shows that the alternate image source is at a PC:

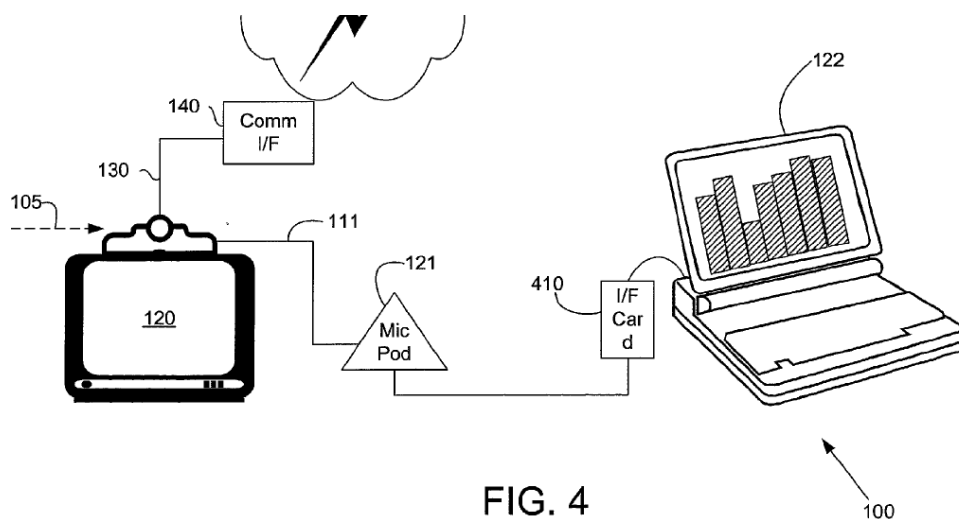


Figure 4 of Kenoyer (EX1005)

196. Kenoyer teaches that the videoconference interface card 410 includes a flash memory “preloaded with application software necessary to make the . . . card 410 operable with the alternative image source 122 and the videoconference system 120.” EX1005, ¶[0036]. Image source 122 may be a computer generating images in RGB format. *Id.*, ¶[0028]. Interface card 410 is a PCMCIA card. *Id.*, ¶[0033]. Card 410 is identified as a storage device by source device 122. *Id.*, ¶[0037]. A POSA

thus would understand that Kenoyer teaches capturing desktop information from the video memory of a PC. Thus, I agree that Kenoyer renders obvious Claim 5. Again, it does not appear that the Applicant disputed the finding of the Examiner that Kenoyer taught the recitation of Claim 5. EX1002 at 161.

197. Kenoyer combined with Clapp also teaches Claim 5. Clapp taught that “All of the pixels comprising the video image within the local active window 602 presented in the foreground of the local display 248 are then copied to the local off-screen buffer at step 636.” EX1006, Col. 11:43-46. Clapp also contemplates data capture using a PCMCIA card. *Id.*, Cols. 13:7-20, 13:66-14:7. Clapp thus teaches capturing desktop information from a video memory of the PC. As indicated above, such capture is obvious in view of Kenoyer alone, and Clapp’s description of capturing information from memory at the local device is readily adaptable to the Kenoyer configuration, which transfers video information from the local device for processing.

198. Claim 6 recites that step (d) of claim 1 further comprises decoding the received bit stream and then outputting to a local output device. EX1001, Col. 5:19-21. As shown above, Kenoyer in view of *Video Compression*, and Kenoyer in view of Shaw teach this claim. A POSA would understand that one common reason to send a coded video bit stream to a remote video communication terminal would be to display the decoded video signal on a local output device. *See, e.g.*, EX1007, Col.

12:14-26; EX1008 at 59 (noting H.261 describes “video coding and decoding methods”).

199. Claim 7 of the '288 Patent recites that step d of Claim 1 further comprises “simultaneously transferring the coded bit stream of the desktop information and that of a local image in a multiplex encoding mode.” EX1001, Col. 5:22-25. “Multiplex encoding mode” is not a term of art and the '288 Patent contains no definition or description of a multiplex encoding mode. The most substantive disclosure of a “multiplex encoding mode” is made in reference to the bit stream transmitter 33 shown in Figure 3, which can “respectively transfer the coded bit stream of desktop information or the local image, or can transfer them simultaneously in a multiplex encoding mode.” *Id.*, Col. 4:24-34. As shown below, that figure shows two streams entering the transmitter 33, a “Local image coded bit stream” and a “PC desktop information bit stream,” and a single stream leaving the transmitter:

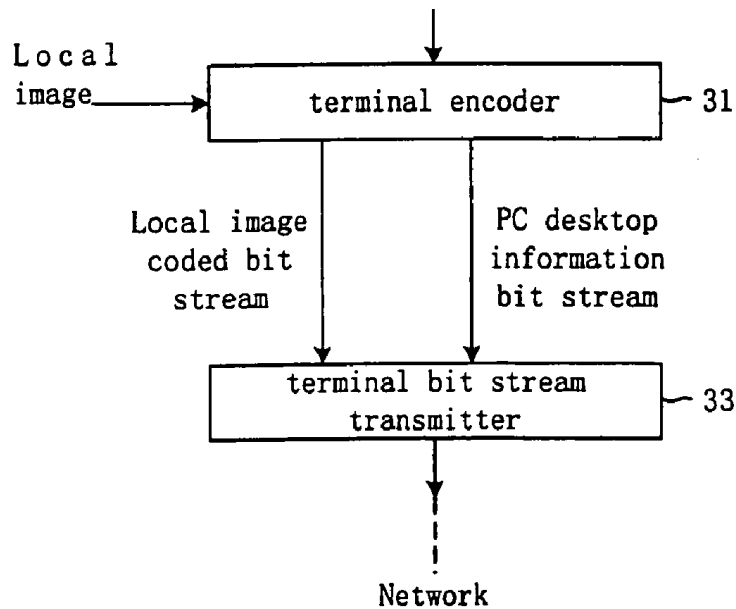


Figure 3 of the '288 Patent (EX1001)

200. Multiplexing signals was well known for several years before the priority date of the '288 Patent as a cost-effective way to transmit video and other information, since only one data stream is needed for multiple signals. RFC 1889 from the Internet Engineering Task Force, which promulgates Internet standards, shows multiplexing was recognized as a standard form of transmitting real-time data such as video data over the Internet at least by 1996. EX1011. Section 2.3 of RFC 1889 is entitled "Mixers and Translators" and describes guidance for multiplexing signals, using as an example a video mixer that scales the images of individual people in separate video streams and "composites them into one video stream to simulate a group scene."

201. Similarly, U.S. Patent No. 6,141,062 to Hall et al, entitled METHOD AND APPRATUS FOR COMBINING VIDEO STREAMS, teaches combining multiple video streams by multiplexing “to reduce interconnection requirements.” EX1012, Col. 2:50-52. A POSA thus would have known of the benefits of multiplexing video streams of data such as the data taught by Kenoyer.

202. The Examiner found that Paragraph [0023] of Kenoyer taught this feature. EX1002 at 136. Paragraph [0023] describes a video subchannel 230 “made up of two video streams: a normal conference video stream 231 and an alternate video stream 232.” EX1005. That paragraph further recites that its teachings “can facilitate *simultaneous* transmission of the normal video conference stream 231 within the same channel as the alternate video stream 232.” *Id.*, ¶[0023] (emphasis added). Kenoyer explains that its “alternate video source” may be a document camera or personal computer that is used with the “normal” conference video. *Id.*, ¶[0012]. The carrier channel 130 consists of video subchannel 230 (consisting of normal conference video 231 and alternate video stream 232) and audio subchannel 240. *Id.*, ¶[0022], Figure 2.

203. The Applicant argued that the recitation of Claim 7 was not taught by Kenoyer because Kenoyer taught “two streams are transmitted in one channel” and “does not teach or suggest transferring the coded bit stream of the desktop information and that of a local image in a multiplex encoding mode.” EX1002 at

161-162. The Applicant did not explain why it believed Kenoyer did not disclose a multiplex encoding mode. However, Figure 2 of Kenoyer depicts a channel 130 as transmitting streams 231, 232, and 240. EX1005. Paragraph [0023] states that video streams 231 and 232 can be transmitted “simultaneous[ly] . . . within the same channel.” *Id.* Thus, while “multiplex encoding mode” is not a term of art, and is not defined in the ’288 Patent, a POSA would understand Kenoyer’s reference to simultaneous transmission of the signals in one channel referred to such a concept.

204. .

205. As discussed above for Claim 1, Kenoyer in combination with Shaw teaches the desktop information being converted into YUV format and transferred as a coded video bit stream. In turn, Paragraph [0023] of Kenoyer teaches that a subchannel carries the desktop video stream as well as a video stream carrying “live video images . . . of the users at the local site 100.” EX1005. Therefore, I agree with the Examiner that Kenoyer teaches Claim 7.

G. Claim 9 is obvious in view of Kenoyer and *Video Compression Techniques* or Kenoyer and Shaw.

206. Claim 9 is an apparatus claim, in contrast to method Claim 1. *Compare* EX1001, Col. 5:33-43 with EX1001, Col. 4:58-5:4. I note that Claim 1 recites a triggering command, but Claim 9 does not. *Id.* During the prosecution of the application that led to the ’288 Patent, the Examiner originally rejected Claim 9 as being anticipated by Kenoyer. EX1002 at 134-136. In response, the Applicant

simply asserted that “[c]laim 9 recited an apparatus having corresponding distinguishing features to claim 1 and, thus, also defines over the art cited by the Examiner.” *Id.* at 162. Presumably, this was a reference to the Applicant’s prior statement that Kenoyer did not teach the “encoding the converted desktop format in a mode ensured by the local video communications terminal” limitation of Claim 1. *Id.* at 159-160. However, as discussed above in Section VIII.A.2., the inclusion of this limitation would have been obvious to a POSA in view of *Video Compression Techniques* or Shaw.

1. “An apparatus for transferring desktop information of a PC to a video communication terminal, comprises”

207. Kenoyer discloses an apparatus for transferring desktop information of a PC to a video communication terminal through its disclosures of videoconference module 110 and the PCMCIA card mode of module 110.

208. Figure 1 of Kenoyer discloses a videoconference accessory module 110 which serves as the claimed apparatus for transferring desktop information of a PC 122 (through its disclosure of an “alternate image source”) to a videoconferencing system/unit 120 in Figures 1-2 which serves as the recited video communication terminal. EX1005. In addition, Figures 1-2 show the transferring of the information to the terminal. *Id.* Figure 1 shows information displayed on the PC 122 (a bar graph) being transferred to a local videoconference system 120 via a videoconference accessory module 110:

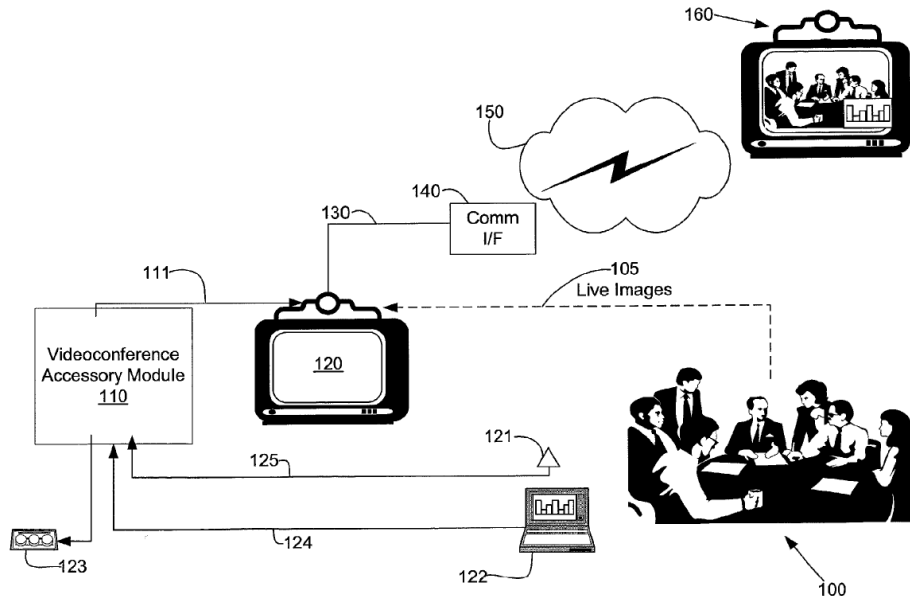


FIG. 1

Figure 1 of Kenoyer (EX1005)

Id., ¶[0020], Figure. 1-2. I note that the alternate mode in Kenoyer also discloses these same elements, the primary difference being that the Videoconference Accessory Module 110 is realized as a PCMCIA card 410 that is integrated into the PC in the alternate embodiment. *See id.*, Figure 4, shown below:

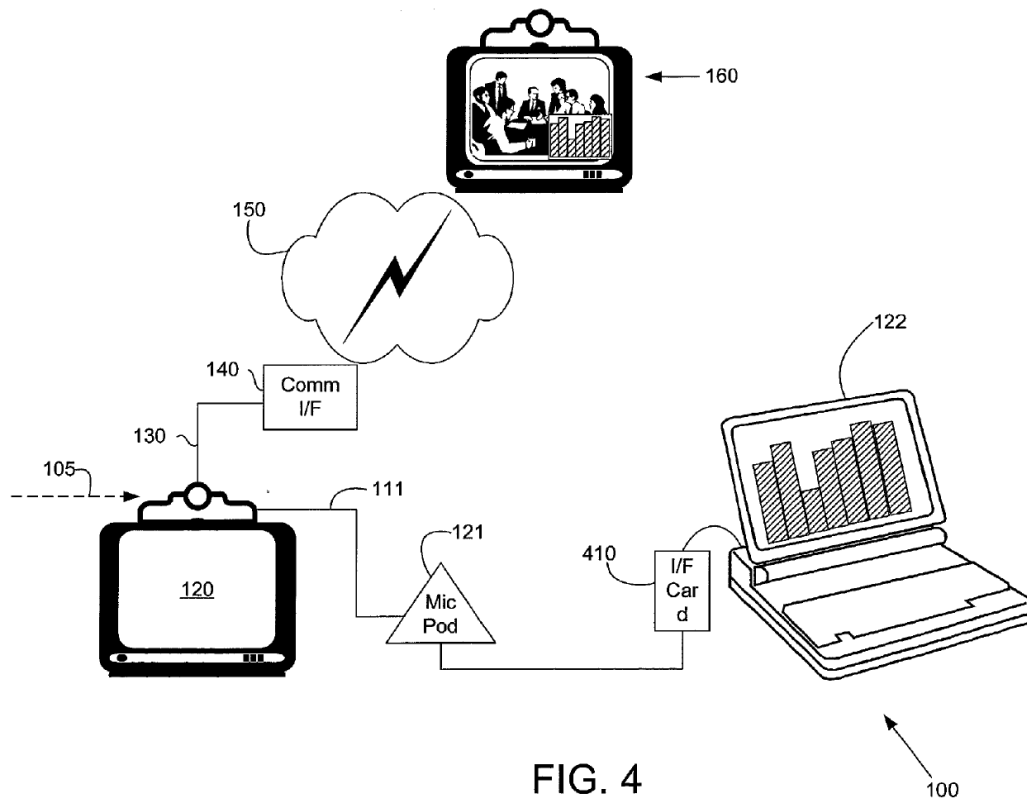


Figure 4 of Kenoyer (EX1005)

209. During the prosecution of the '288 Patent, the Examiner found that Kenoyer taught this element. EX1002 at 134, 138. The Applicant did not dispute the Examiner's finding. I agree that Kenoyer teaches an apparatus for transferring desktop information of a PC to a video communication terminal. In addition to the analysis provided by the Examiner, Kenoyer further discloses an apparatus for transferring desktop information of a PC to a video communication terminal through its disclosures of Videoconference Module 110 and the corresponding mode wherein the module 110 is realized as a PCMCIA card 410. EX1005 Figs. 1, 4. Alone or combined with PC 122, these components are an apparatus for transferring desktop

information of a PC to a video communication terminal for the reasons stated above in regard to Claim 1.

2. “A background processing device, capturing desktop information, converting the captured desktop information from a PC format to a format of a local video communication terminal”

210. During the prosecution of the '288 Patent, the Examiner found that Kenoyer taught this element through its disclosure of the Videoconference Accessory Module 110. EX1002 at 138. The Applicant did not dispute this finding and in particular, in its remarks about Claim 1, did not dispute that Kenoyer taught capturing desktop information and converting the captured desktop information from a PC format to a format of a local video communication terminal. The Applicant was silent on the issue of a background processing device in Kenoyer. I agree with the Examiner that Kenoyer teaches the required device in at least its disclosures of Videoconference Accessory Module 110.

211. The '288 Patent has few disclosures of the claimed “background processing device” beyond text that simply mirrors the language of the claims. Figure 2 shows a block diagram of the background processing device, and Figure 4 shows the PC as the background processing device:

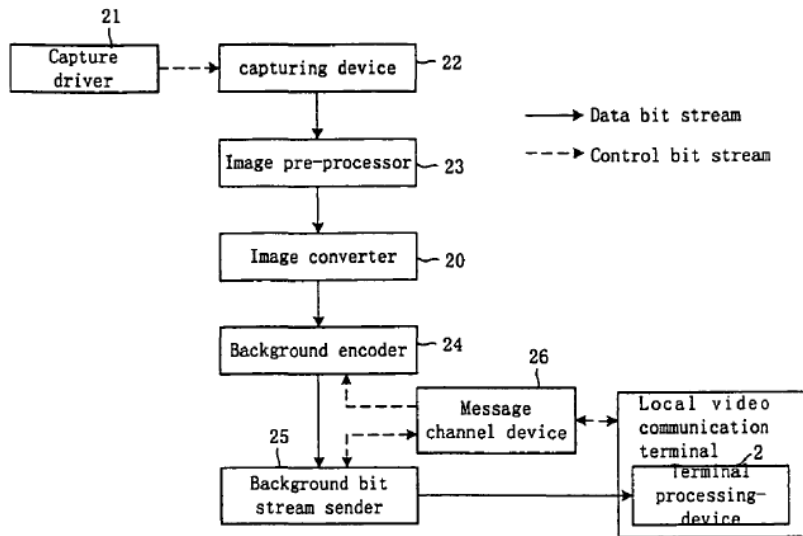


Fig. 2

Figure 2 of the '288 Patent (EX1001)

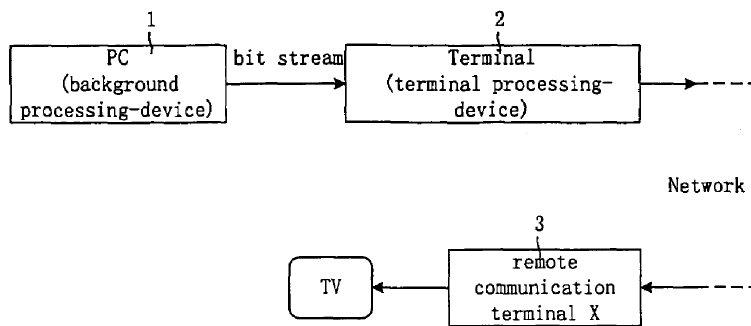


Fig. 4

Figure 4 of the '288 Patent (EX1001)

212. The components shown in Figure 2 for capturing desktop information, processing, converting, and encoding the video are all found in Kenoyer via the PC 122 or the PC 122 plus the Videoconference Accessory Module 110. Alternatively, these elements are taught by Kenoyer in view of *Video Compression Techniques* or Shaw, as explained in the analysis of Claim 1 above. In particular, Kenoyer

describes the claimed device as shown in Figure 3, wherein the Videoconference Accessory Module 110 captures desktop information at VGA input 353 and converts the processed information from a PC RGB format to a YUV format compatible with video communication terminal 120. EX1005, ¶[0028], Figure 3 (below):

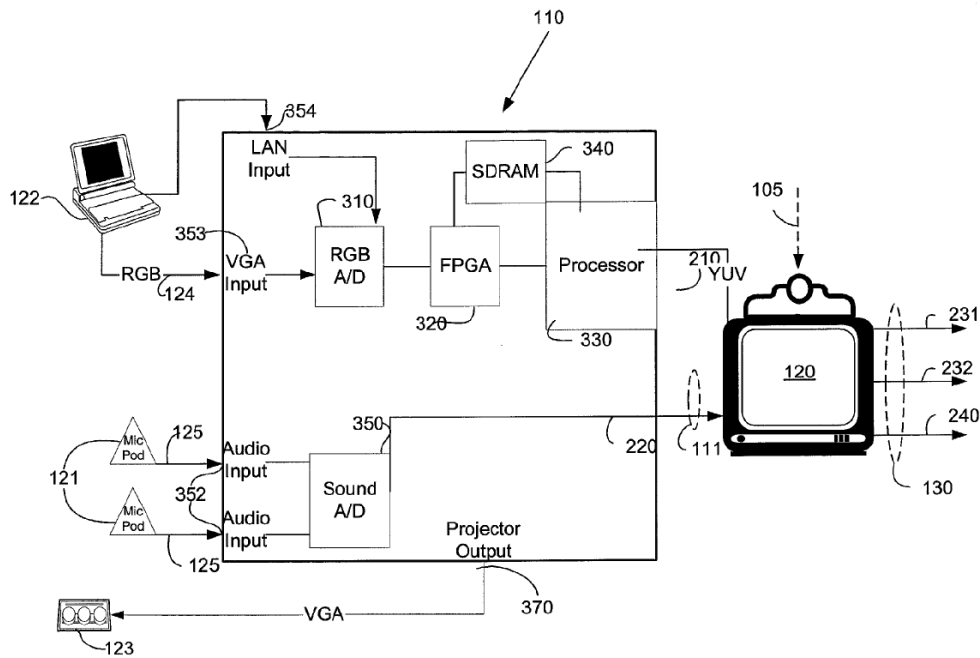


FIG. 3

Figure 3 of Kenoyer (EX1005)

213. The alternate mode of Kenoyer using the PCMCIA card 410 also teaches capturing RGB signals and converting the captured signals into YUV format. EX1005, ¶[0030], [0035].

3. “[a background device] encoding the converted format to a coded bit stream and outputting”

214. This claim element is obvious for the same reasons discussed above for the corresponding elements of Claim 1. In particular, a POSA would understand that the “background device” in Kenoyer is Videoconference Access Module 110, which converts a RGB video signal into YUV format, encodes it in a video mode such as H.261, and outputs it as a video bit stream, based on the teachings of *Video Compression Techniques* or Shaw.

215. As I discussed in detail regarding Claim 1, a POSA would have been motivated to output the signal of Kenoyer in a video bit stream format such as under the H.261 standard based on Kenoyer’s teachings that the FPGA 320 and processor 330 convert and compress to “enhance throughput and performance” the additional explicit teachings either of Shaw or *Video Compression* regarding the advantages of encoding videoconference video signals into a video bit stream using H.261. EX1005, ¶¶0029]; EX1007, Cols. 11:6-13; 12:34-40; 18:34-48, 23:43-47, 23:63-24:229:43-44; EX1008 at 59 (teaching outputting in coded bit stream), 61-62. That analysis also supports my opinion that this element of Claim 9 is obvious.

4. “a terminal processing device, transferring the coded bit stream from the background processing device to a remote video communication terminal.”

216. The '288 Patent provides few disclosures of the “terminal processing device” other than language that simply mirrors the text of the claims. The device is illustrated in Figures 2 and 4 and generally described as the device within the local video communication terminal that receives the video bit stream and transmits the stream to the remote site.

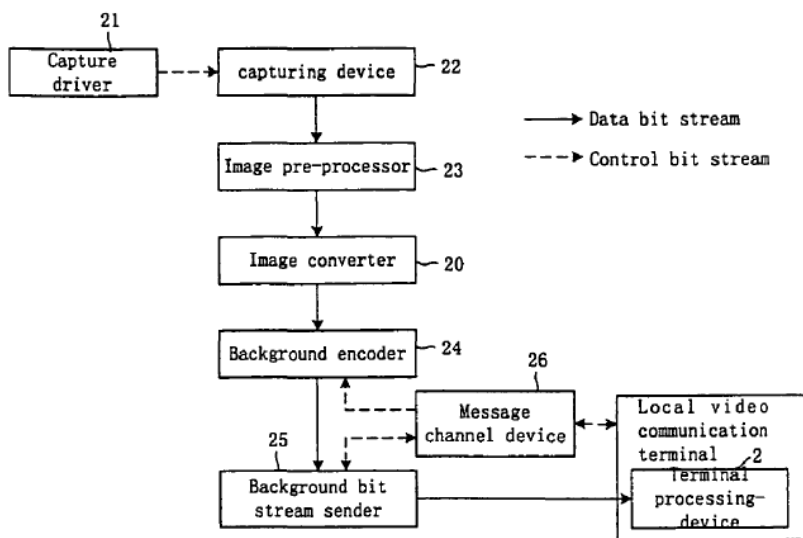


Fig. 2

Figure 2 of the '288 Patent (EX1001)

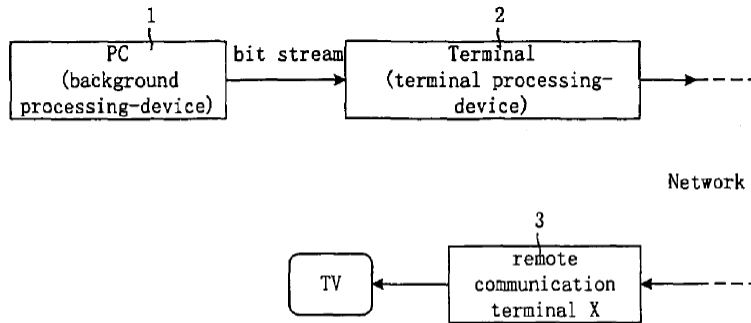


Fig. 4

Figure 4 of the '288 Patent (EX1001)

EX1001, Figures 2 and 4, Col. 4:6-10 (“[a]s shown in FIG. 4, the PC has a background processing device 1 to capture the desktop information of the PC, and then the captured desktop information is encoded and sent, through a network port, to the local video communication terminal that has a terminal processing-device 2, and then the local video communication terminal forwards the received coded bit stream to the remote video communication terminal 3”).

217. Kenoyer’s PCMCIA mode similarly teaches that the videoconferencing system 120 transfers output from the background processing device (PC 122 and PCMCIA card 410 at PC 122) through interface 140 to a remote video communication terminal (remote videoconferencing system/unit 160), as illustrated in Figure 4 below. EX1005, ¶[0035].

218. As explained above in the analysis of Claim 1, Kenoyer in combination with *Video Compression Techniques* and/or Shaw teaches that the

videoconferencing system/unit 120 transfers the coded bit stream from the background processing device (PC 120 and/or Videoconference Accessory Module 110) to a remote video communication terminal (remote videoconferencing system/unit 160) through a communications interface 140, as illustrated below in Figure 1. Unit 120 and communications interface 140 thus serve as the claimed terminal processing device.

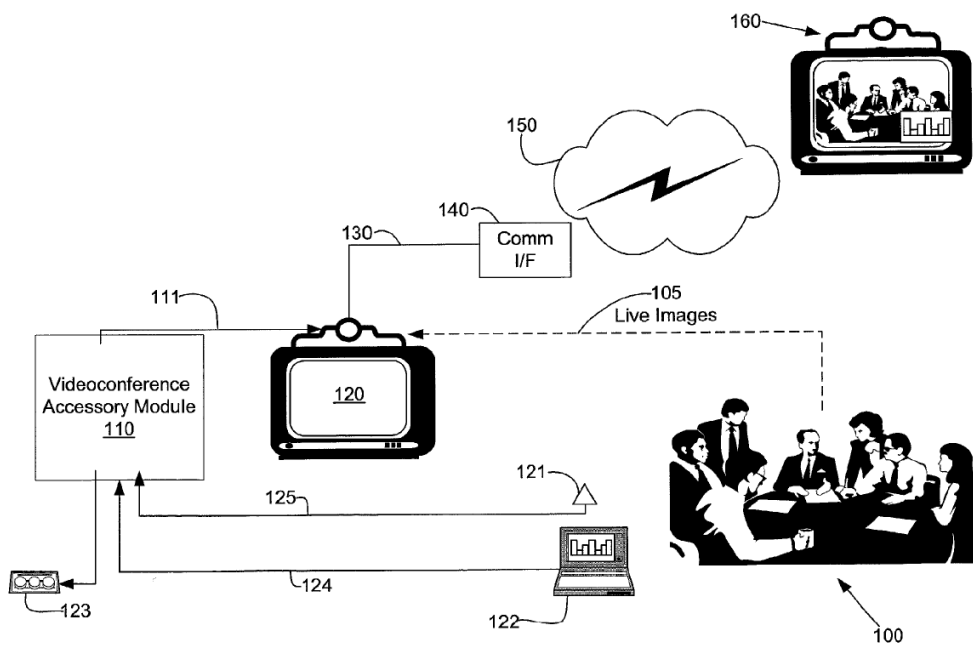


FIG. 1

Figure 1 of Kenoyer (EX1005)

Figure 4 of Kenoyer also teaches the claimed structure. EX1005.

219. Thus, the combination of Kenoyer and *Video Compression Techniques*, or the combination of Kenoyer and Shaw, teaches all elements and limitations as arranged in Claim 9, and Claim 9 as a whole is obvious.

H. Dependent Claims 10, 12, and 14 are Obvious in view of Kenoyer in View of Clapp and *Video Compression Techniques*, or Kenoyer in View of Clapp and Shaw.

220. Claim 10 recites as follows:

10. The apparatus according to claim 9, wherein the background processing device comprises:

a capture driver, sending out a triggering command;

a capturing device, receiving the triggering command from the capture driver and capturing the desktop information, then outputting;

an image converter, receiving the captured desktop information from the capturing device and converting into the format of the local video communication terminal, then outputting;

a background encoder, encoding output signal from said image converter into the coded bit stream;

a background bit stream sender, sending the coded bit stream to the terminal processing-device.

EX1001, Col. 5:43-6:8.

221. Figure 2 of the '288 Patent (pictured below) shows a capture driver 21 sending a triggering command to capturing device 22, which captures the desktop information and outputs the information to an image preprocessor. EX1001, Col. 5:52-64.

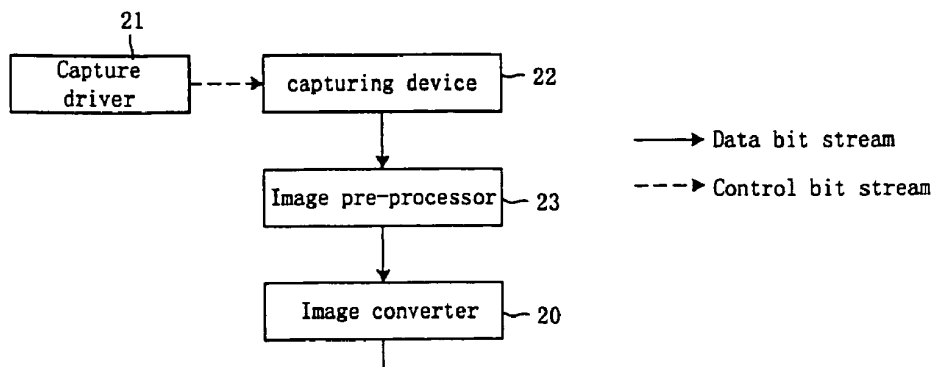


Figure 2 of the '288 Patent (EX1001)

222. Figure 4 of the '288 Patent (pictured below) shows the PC as the “background processing-device,” which sends a bit stream to the terminal/“terminal processing device.” EX1001. A POSA would understand that the PC is at least a possible background device as contemplated by the '288 Patent.

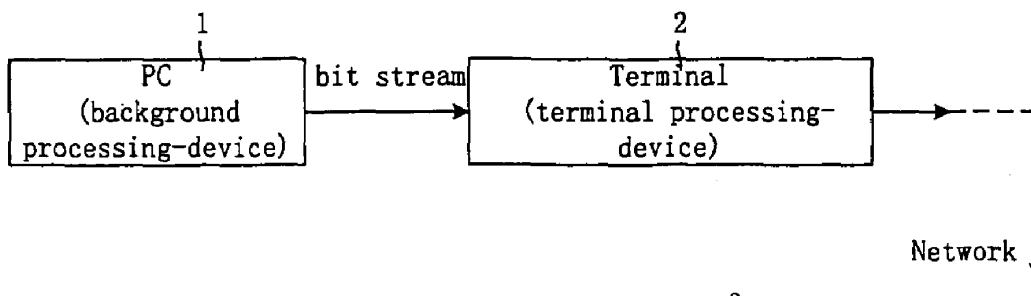


Figure 4 of the '288 Patent (EX1001)

223. The Examiner found that Clapp teaches a capturing driver while Kenoyer teaches a capturing device, a background encoder, and a background bitstream sender, and it would have been obvious to combine them as claimed. EX1002 at 136-137. The Applicant did not dispute these findings, instead relying

on its analysis of Claim 1. *Id.* at 162. I agree with the Examiner for the reasons discussed above for Claim 1 regarding the capturing, encoding, and sending steps.

224. The Examiner relied on Paragraphs [0025] and [0035]-[0037] of Kenoyer as teaching the elements of Claim 10 other than the capturing driver. EX1002 at 136. Kenoyer teaches a Video Interface Card 410 using similar technology to the Videoconference Accessory Module 110 of its Figure 3. EX1005, ¶[0035]. The Videoconference Accessory Module 110, which is comprised of processor 320 and memory 340, captures desktop information and outputs it to an image converter (processor 330). *Id.*, ¶[0026]. Kenoyer also teaches an image converter 330 which receives the information from the capture device, converts it into the format of the local video communication terminal (YUV), and outputs the converted information.

225. The Examiner did not provide reasons for allowing Claim 10. However, the Applicant's arguments for Claim 10 relied on the arguments for Claim 1, which focused on Kenoyer not disclosing encoding signals or transmitting signals in a bit stream. EX1002 at 160. As discussed above regarding Claim 1, this purported distinction from Kenoyer is no more than an obvious modification of Kenoyer in combination with *Video Compression Techniques* or Shaw. A POSA would find it obvious to encode and transmit a converted YUV digital signal in a video bit-stream, at least because the preferred videoconferencing signal

transmission protocols at the time of the invention, such as H.261, encoded and transmitted signals in the claimed video bit-stream format.

226. The Examiner cited Clapp for the capturing driver which triggers the image capture. EX1002 at 135, citing EX1006, Col. 11:47-64. As discussed above for Claim 1, Clapp teaches a device which sends a triggering command, corresponding to the capturing driver of Claim 10. Also, as discussed above in connection with Claim 1, a POSA would have been motivated to modify Kenoyer by implementing Clapp's capturing device to efficiently control when images are captured and processed by the system and avoid needless image capture when, for example, the image has not changed. Therefore, Claim 10 is obvious over Kenoyer in view of Clapp and *Video Compression Techniques*, or obvious over Kenoyer in view of Clapp and Shaw.

227. Claim 12 depends from Claim 9 and further recites that the terminal processing device "comprises a terminal bit stream transmitter that transfers the coded bit stream to the remote video communications terminal." EX1001, Col. 6:14-17.

228. Kenoyer discloses sending the converted data to the local video communication terminal (the videoconferencing system/unit 120). EX1005, ¶[0030] ("[a]fter the compression and conversion of the alternate image source signal 124 to YUV format, the videoconference accessory module 110 outputs the

processed image to the videoconference unit 120”). As shown below in Figure 1 of Kenoyer, Unit 120 and communications interface 140 send the converted data to remote site 160, which serves as the remote video communications terminal. *Id.*, ¶[0019].

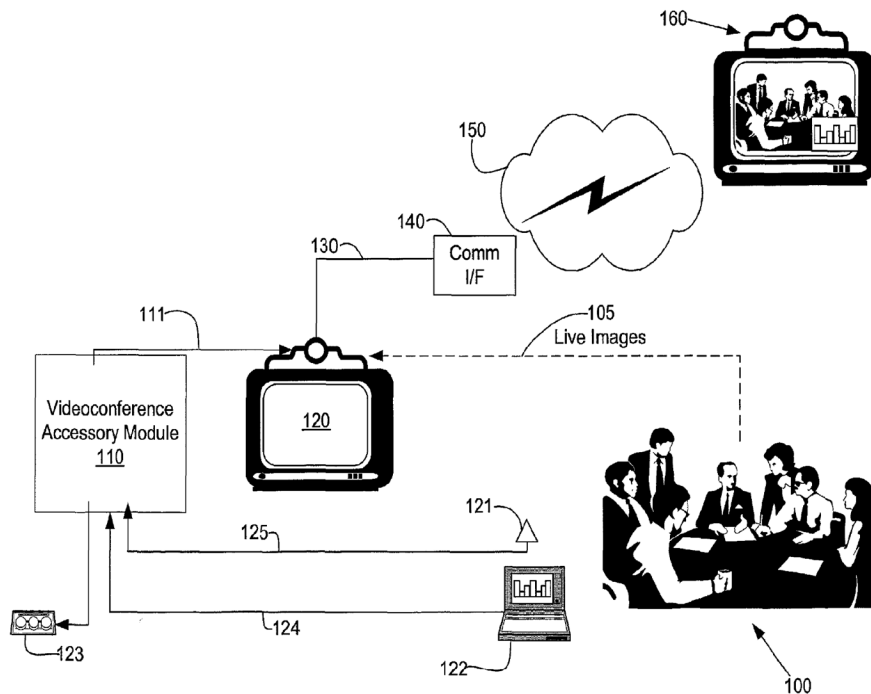


FIG. 1

Figure 1 of Kenoyer (EX1005)

229. As shown in the analysis of Claim 1 above regarding the use of bit stream format, *Video Compression Techniques* and Shaw disclose the features of Claim 12. For example, Shaw teaches a bandwidth controller that controls the compression ratio of a video codec to generate a coded video bit stream at a rate that is compatible with the network. EX1007, Cols. 8:24-33;23:43-24:2. It would have been obvious to a POSA to modify Kenoyer to use the well-known

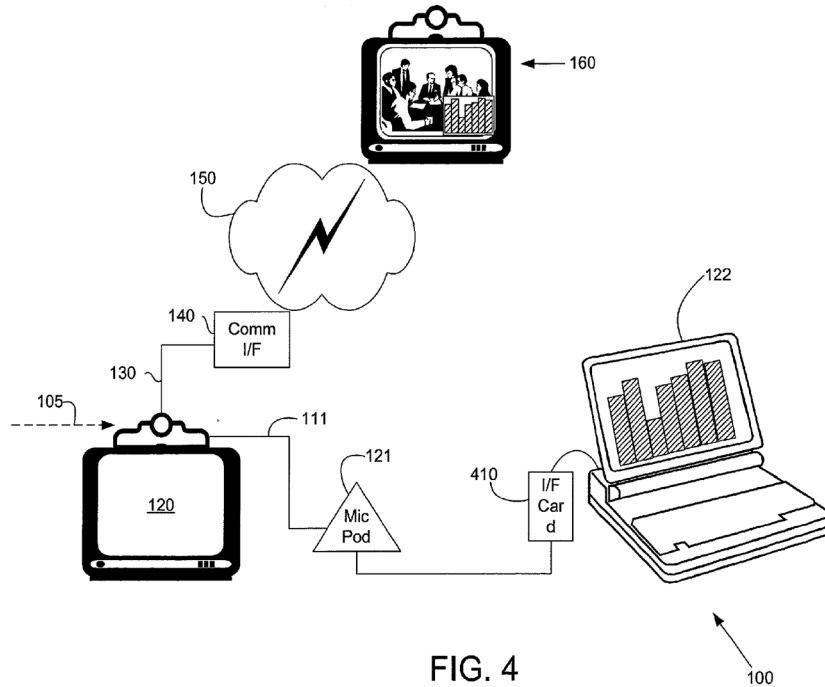
videoconferencing transmission standards as disclosed in these references to encode and transmit the converted video data in a coded video bit stream to a remote device.

230. Thus, Claim 12 is obvious over Kenoyer in view of *Video Compression Techniques* or Shaw.

231. Claim 14 depends from Claim 12 and recites that the terminal processing device “further comprises a terminal encoder that encodes a local image and then outputs to the terminal bit stream transmitter.” EX1001, Col. 6:23-26.

232. Kenoyer teaches that interface card 410 acts as the claimed terminal encoder to encode the local image captured at PC 122 and transmit it to device 120 that corresponds to the claimed terminal bit stream transmitter. EX1005, ¶[0035], Figure 4. Interface 410 has similar functionality to module 110 from Figure 3 of Kenoyer. *Id.*

233. Kenoyer teaches this element at system 120 in their disclosure concerning PC 122 and interface 410, as shown in Figure 4:



Excerpt of Figure 4 of Kenoyer (EX1005)

The interface card 410 of the videoconference system 120 of Kenoyer act as the terminal encoder that encodes a local image and then outputs to the terminal bit stream transmitter. “System 120 captures live video 105 at the local site and transmits the images on the normal conference video stream 231 (FIG. 2) of the carrier channel 130 to the communications interface 140.” EX1005, ¶[0034]. Figure 2 is shown below:

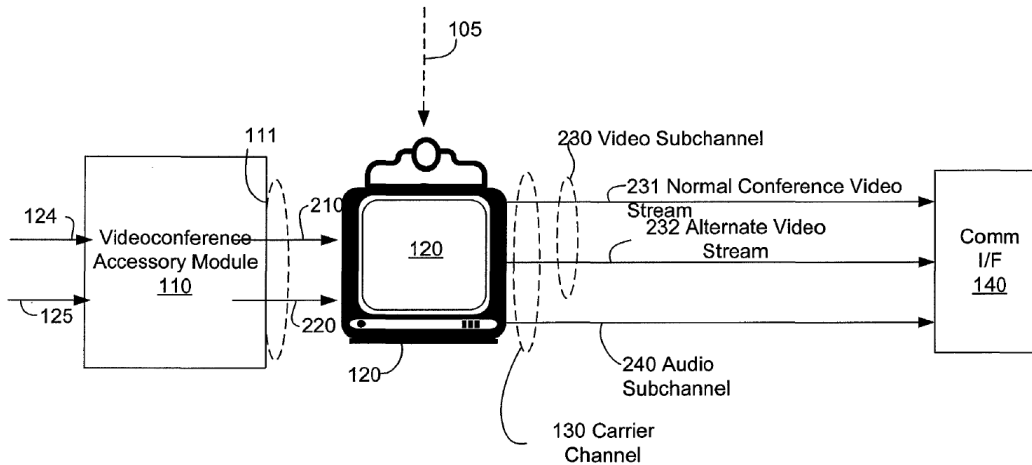


Figure 2 of Kenoyer (EX1005)

The live video and alternate image signals are transmitted as a video stream on carrier channel 130. EX1005, ¶[0022].

234. The Applicant did not dispute the Examiner’s analysis of Claim 14 or otherwise separately argue Claim 14 is patentable over Kenoyer. EX1002 at 162. For all the above reasons, Claim 14 is obvious over Kenoyer in view of *Video Compression Techniques* or Shaw.


235. I reserve the right to supplement my opinions as appropriate if the Patent Owner alleges or offers any purported evidence of any such secondary considerations of non-obviousness.

236. I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful

false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code.

Executed this 22nd day of June, 2020

Respectfully submitted,



Samrat Bhattacharjee

APPENDIX

CURRICULUM VITAE
BOBBY BHATTACHARJEE

DEPARTMENT OF COMPUTER SCIENCE
THE UNIVERSITY OF MARYLAND
COLLEGE PARK

1 Personal Information

Professor,
Computer Science Department and
the Institute for Advanced Computer Studies,
University of Maryland
Appointed Fall, 1999.

Affiliate Professor,
Department of Electrical and Computer Engineering,
University of Maryland.

Alfred P. Sloan Research Fellow (2004–2006).

1.1 Education

- Ph.D. in Computer Science
Georgia Institute of Technology, Atlanta, Georgia, Summer 1999
Dissertation title: Active Networking: Architectures, Composition, and Applications
Advisors: Kenneth L. Calvert and Ellen W. Zegura
- Bachelor of Science in Mathematics and Computer Science
Georgia College and State University, Milledgeville, Georgia, Spring 1994
Graduated *Summa Cum Laude* and Outstanding Department Major

1.2 Employment

Summer 2009 to present	Professor University of Maryland, College Park, Maryland
Summer 2005 to present	Associate Professor University of Maryland, College Park, Maryland
Fall 2006	Visiting Professor Max Planck Institut für Software Systems, Saarbrücken, Germany
Spring, Summer 2007	Visiting Researcher

	AT&T Labs, Florham Park, New Jersey
Fall 1999 to Spring 2005	Assistant Professor University of Maryland, College Park, Maryland
Fall 1995 to Summer 1999	Research Assistant Georgia Institute of Technology, Atlanta, Georgia
Summer 1998	Instructor Georgia Institute of Technology, Atlanta, Georgia
Summer 1997	Member of Technical Staff AT&T Labs, Florham Park, New Jersey
Summer 1995	Member of Technical Staff GTE Labs, Waltham, Massachusetts
Fall 1994 to Spring 1995	Teaching Assistant Georgia Institute of Technology, Atlanta, Georgia

2 Research, Scholarly, and Creative Activities

2.1 Chapters in Books

1. Gisli Hjálmtýsson and Samrat Bhattacharjee. “Control on Demand”, In *Proceedings of the First International Working Conference on Active Networks* volume 1653 of *Lecture Notes in Computer Science (Stefan Covaci, editor)*, pages 315-329, Springer-Verlag, June 1999.
2. Pete Keleher, Samrat Bhattacharjee, and Bujor Silaghi. “Are Virtualized Overlay Networks Too Much of a Good Thing?”, *Peer-to-Peer Systems First International Workshop, Lecture Notes in Computer Science*, Vol. 2429, (Peter Druschel et. al. Editors) pages 225–231, Springer-Verlag, 2002.
3. Bobby Bhattacharjee, Sudarshan S. Chawathe, Vijay Gopalakrishnan, Peter J. Keleher, and Bujor D. Silaghi. “Efficient Peer-To-Peer Searches Using Result-Caching”, *Peer-to-Peer Systems II, Second International Workshop, IPTPS 2003, Lecture Notes in Computer Science*, Vol. 2735, (M. Frans Kaashoek and Ion Stoica, Editors), pages 225–236, Springer-Verlag, 2003.
4. Paolo Massa and Bobby Bhattacharjee. “Using Trust in Recommender Systems: an Experimental Analysis”, In *Second International Conference, iTrust 2004, Lecture Notes in Computer Science, Vol. 2995 Jensen, Christian; Poslad, Stefan; Dimitrakos, Theo (Eds.)*, pages 221-235, Springer-Verlag, 2004.

5. Cristian Lumezanu, Neil Spring, and Bobby Bhattacharjee. “Decentralized Message Ordering for Publish/Subscribe Systems”, *ACM/IFIP/Usenix 7th International Middleware Conference, Lecture Notes in Computer Science*, Vol. 4290 (Maarten van Steen and Michi Henning, Editors), pages 162–179, Springer-Verlag, 2006.
6. Misha Rabinovich and Bobby Bhattacharjee. “Overlay Networks and Resiliency”, in *Guide to Reliable Internet Services and Applications*, Charles R Kalmanek, Sudip Misra, and Y. Richard Yang (Editors). Springer-Verlag, 2010.

2.2 Articles in Refereed Journals

1. Kenneth L. Calvert, Samrat Bhattacharjee, Ellen W. Zegura, and James Sterbenz. “Directions in Active Networks”, *IEEE Communications Magazine*, No. 10, pages 72-78, 1998.
2. Samrat Bhattacharjee, Ellen W. Zegura, and Kenneth L. Calvert. “Active Networking and End-to-End Arguments”, *IEEE Network Magazine*, No. 3, pages 66-71, 1998.
3. Gisli Hjálmtýsson and Samrat Bhattacharjee. “Control on Demand - An Efficient Approach to Router Programmability”, *IEEE Journal on Selected Areas in Communications, JSAC*, Vol. 17, No. 9, pages 1549-1562, September 1999.
4. S. Bhattacharjee, W. C. Cheng, C.-F. Chou, L. Golubchik, and S. Khuller. “Bistro: a Platform for Building Scalable Wide-Area Upload Applications”, *ACM SIGMETRICS Performance Evaluation Review*, Vol. 28, No. 2, pages 29-35, September 2000.
5. Ellen W. Zegura, Mostafa Ammar, Zongming Fei, and Samrat Bhattacharjee. “Application-Layer Anycasting: A Server Selection Architecture and Use in Replicated Web Service”, *Transactions on Networking*, Vol. 8, Issue 4, pages 455-466, August 2000.
6. Suman Banerjee and Samrat Bhattacharjee. “Scalable Secure Group Communications over IP-multicast”, *IEEE Journal of Selected Areas in Communications, JSAC*, Vol. 20, No. 8, pages 1511 - 1527, October 2002.
7. U. Cetintemel, P. J. Keleher, B. Bhattacharjee, and M. J. Franklin. “Deno: A Decentralized, Peer-to-Peer Object-Replication System for Weakly-Connected Environments”, *IEEE Transactions on Computers*, Vol. 52, No. 7, pages 943–959, July 2003.
8. Suman Banerjee, Christopher Kommareddy, and Bobby Bhattacharjee. “Efficient Peer Location on the Internet”, *Computer Networks Journal*, Vol. 5:1, pages 5-17, 2004.
9. Rob Sherwood, Bobby Bhattacharjee, and Aravind Srinivasan. “P5: A Protocol for Scalable Anonymous Communications”, *Journal of Computer Security*, Vol 13:6, pages 839-876, 2005.
10. Suman Banerjee, Christopher Kommareddy, Koushik Kar, Bobby Bhattacharjee, and Samir Khuller. “OMNI: An Efficient Overlay Multicast Infrastructure for Real-time Applications”, *Special Issue of Computer Networks on Overlay Distribution Structures and their Applications*, Vol 50:6, pages 826-842, 2005.

11. Rob Sherwood, Seungjoon Lee, and Bobby Bhattacharjee. “Cooperative Peer Groups in NICE”, *Computer Networks Journal, Special Issue on Management in P2P systems: Trust, Reputation and Security*, Vol 50:4, pages 523-544, 2006.
12. Tuna Guven, Chris Kommareddy, Richard J. La, Mark A. Shayman, and Bobby Bhattacharjee. “Measurement-Based Optimal Routing on Overlay Architectures for Unicast Sessions”, *Computer Networks Journal: Special issue on Network Modeling and Simulation*, Vol. 50, No. 12, pages 1938–1951, August 2006.
13. Suman Banerjee, Seungjoon Lee, Bobby Bhattacharjee, and Aravind Srinivasan. “Resilient Multicast using Overlays”, *IEEE/ACM Transactions of Networking*, Vol. 14, No. 2, pages 237–248, April 2006.
14. Ruggero Morselli, Bobby Bhattacharjee, Michael A. Marsh, and Aravind Srinivasan. “Efficient Lookup on Unstructured Topologies”, *IEEE Journal on Selected Areas in Communications (J-SAC), Special Issue on Peer-to-Peer Communications and Applications*, pages 62-72, 2007.
15. Jik-Soo Kim, Beomseok Nam, Peter Keleher, Michael Marsh, Bobby Bhattacharjee, and Alan Sussman. “Trade-offs in Matching Jobs and Balancing Load for Distributed Desktop Grids”, *Future Generation Computer Systems – International Journal of Grid Computing: Theory, Methods & Applications*, Vol. 25, No. 5, pages 415–424, 2008.
16. Seungjoon Lee, Bobby Bhattacharjee, Aravind Srinivasan, and Samir Khuller. “Efficient and Resilient Backbones for Multihop Wireless Networks”, *IEEE Transactions on Mobile Computing*, Vol 7:11, 2008.
17. T. Guven, R. La, M. Shayman, and B. Bhattacharjee. “A Unified Framework for Multipath Routing for Unicast and Multicast”, *IEEE/ACM Transactions on Networking*, Vol. 6:5, 2008.
18. Seungjoon Lee, Bobby Bhattacharjee, Suman Banerjee, Bo Han. “A Generic Framework for Efficient Geographic Routing in Wireless Networks”, *Elsevier Computer Networks*, Vol 54:5, 2010.
19. Bo Han, Lusheng Ji, Seungjoon Lee, Bobby Bhattacharjee, and Robert R. Miller. “Are All Bits Equal? – Experimental Study of IEEE 802.11 Communication Bit Errors”. *IEEE/ACM Transactions on Networking*, Vol. 20, No. 6, 2012.
20. V. Singh, M. Lentz, B. Bhattacharjee, R. J. La and M. A. Shayman. “Dynamic frequency resource allocation in heterogeneous cellular networks”. *IEEE Trans. on Mobile Computing (TMC)*. 2016.
21. Wagner, Justin and Paulson, Joseph N. and Wang, Xiao and Bhattacharjee, Bobby and Bravo, Hector Corrada, *Privacy-Preserving Microbiome Analysis Using Secure Computation*, *Bioinformatics*, 2016.
22. Suman Banerjee, Bobby Bhattacharjee, and Christopher Kommareddy. “Scalable Application Layer Multicast”, *Transactions on Networking*, Under revision, Submitted in 2002.

2.3 Articles in Refereed Conferences and Workshops

1. Ellen W. Zegura, Kenneth L. Calvert, and Bobby Bhattacharjee. “How to Model an Internetwork”, In *Proceedings of INFOCOM’96*, pages 594–602, 1996.
2. Bobby Bhattacharjee, Mostafa Ammar, Ellen Zegura, Viren Shah, and Zongming Fei. “Application-Layer Anycasting”, In *Proceedings of INFOCOM’97*, pages 1388–1396, Kobe, Japan, 1997.
3. Bobby Bhattacharjee, Kenneth L. Calvert, and Ellen W. Zegura. “Active Networking and the End-to-End Argument”, In *Proceedings of ICNP’97*, pages 220–228, 1997.
4. Bobby Bhattacharjee, Kenneth L. Calvert, and Ellen W. Zegura. “An Architecture for Active Networking”, In *Proceedings of IFIP TC6 Seventh International Conference on High Performance Networking’97*, pages 265–279, 1997.
5. Bobby Bhattacharjee, Kenneth L. Calvert, and Ellen W. Zegura. “Self-Organizing Wide Area Network Caches”, In *Proceedings of INFOCOM’98*, pages 600–608, 1998.
6. Bobby Bhattacharjee, Kenneth L. Calvert, and Ellen W. Zegura. “Reasoning about Active Networks”, In *Proceedings of ICNP’98*, pages 31–41, 1998.
7. Zongming Fei, Bobby Bhattacharjee, Ellen W. Zegura, and Mostafa Ammar. “A Novel Server Selection Technique for Improving the Response Time of a Replicated Service”, In *Proceedings of INFOCOM’98*, pages 783–791, San Francisco, CA, 1998.
8. Samrat Bhattacharjee, Kenneth L. Calvert, and Ellen W. Zegura. “Congestion Control and Caching in CANEs”, In *Proceedings of ICC’98, Workshop on Active Networks and Programmable Networks*, 1998.
9. Samrat Bhattacharjee, Kenneth L. Calvert, and Ellen W. Zegura. “LIANE — Composition for Active Networks”, *Computer Communications Workshop*, 1998.
10. Gisli Hjálmtýsson and Samrat Bhattacharjee. “Control on Demand”, *Proceedings of International Workshop on Active Networking*, Berlin, pages 315–329, 1999.
11. S. Merugu, S. Bhattacharjee, E. Zegura, and K. Calvert. “Bowman: A Node OS for Active Networks”. *Proceedings of IEEE Infocom*, pages 1127–1136, 2000.
12. Y. Chae, S. Merugu, E. Zegura, and S. Bhattacharjee. “Exposing the Network: Support for Topology Sensitive Applications”, *Proceedings of IEEE OpenArch*, pages 65–74, 2000.
13. Samrat Bhattacharjee, William Cheng, Chen-Fu Chou, Leana Golubchik, and Samir Khuller. “Bistro: A Platform for Building Scalable Wide-Area Upload Applications”, *Proceedings of PAWS Workshop*, 2000.
14. R. Jaegar, S. Bhattacharjee, J. K. Hollingsworth, R. Duncan, T. Lavian, and F. Travostino. “Integrating Active Networking and Commercial-Grade Routing Platforms”, *Usenix 2000 Workshop on Intelligence at the Edge*, 2000.

15. Suman Banerjee and Bobby Bhattacharjee. “Scalable Group Communication over IP Multicast”, In *Proceedings of the International Conference on Network Protocols*, pages 261–271, 2001.
16. Narendar Shankar, Christopher Komareddy, and Bobby Bhattacharjee. “Finding Close Friends over the Internet”, In *Proceedings of the International Conference on Network Protocols*, pages 301–311, 2001.
17. Matt Sanders, Ken Calvert, Bobby Bhattacharjee, Stephen Zabele, Mark Keaton, and Ellen Zegura. “Active Reliable Multicast on CANEs: A Case Study”, *IEEE OpenArch*, pages 49–62, 2001.
18. Suman Banerjee and Samrat Bhattacharjee. “Scalable Application-Layer Multicast for Content Distribution”, *Computer Communications Workshop*, 2001.
19. Suman Banerjee, Bobby Bhattacharjee, and Christopher Komareddy. “Scalable Application-Layer Multicast”, *Proceedings of ACM SIGCOMM*, pages 205–217, 2002.
20. Bobby Bhattacharjee, Matt Sanders, Shashidhar Merugu, Ken Calvert, and Ellen Zegura. “CANEs: An Execution Environment for Composable Services”, In *DARPA Active Networks Conference and Exposition (DANCE 2002)*, pages 255–267, 2002.
21. Laura Bright, Samrat Bhattacharjee, and Louiqa Raschid. “Supporting Diverse Mobile Applications with Client Profiles”, In *Proceedings of ACM Workshop on Wireless Mobile Multimedia (WoWMoM)*, pages 88–95, 2002.
22. Suman Banerjee, Christopher Komareddy, and Bobby Bhattacharjee. “Scalable Peer-Finding on the Internet”, *Proceedings of Globecom 2002*, pages 2217–2221, 2002.
23. Rob Sherwood, Bobby Bhattacharjee, and Aravind Srinivasan. “P5: A Protocol for Scalable Anonymous Communications”, In *Proceedings of the IEEE Symposium on Security and Privacy*, pages 58–70, 2002.
24. Pete Keleher, Samrat Bhattacharjee, and Bujor Silaghi. “Are Virtualized Overlay Networks Too Much of a Good Thing?”, *First International Workshop on Peer-to-Peer Systems (IPTPS’02)*, 2002.
25. Bujor Silaghi, Bobby Bhattacharjee, and Pete Keleher. “Routing in the TerraDir Directory Service”, *Proceedings of SPIE/ITCom 2002*, Vol. 4868-30, pages 42–53, 2002.
26. Suman Banerjee, Seungjoon Lee, Bobby Bhattacharjee, and Aravind Srinivasan. “Scalable Resilient Multicast”, *Proceedings of ACM SIGMETRICS*, pages 102–113, 2003.
27. K-T Kuo, S. Phuvoravan, B. Bhattacharjee, R. J. La, M. Shayman, and H. S. Chang. “On the Use of Flow Migration for Handling Short-term Overloads”, *IEEE Globecom*, pages 3108–3112, 2003.
28. Suman Banerjee, Christopher Komareddy, Koushik Kar, Bobby Bhattacharjee, and Samir Khuller. “Construction of an Efficient Overlay Multicast Infrastructure for Real-time Applications”, *Proceedings of IEEE Infocom*, pages 1521–1531, 2003.

29. Seungjoon Lee, Rob Sherwood, and Bobby Bhattacharjee. “Cooperative Peer Groups in NICE”, *Proceedings of IEEE Infocom*, pages 1272–1282, April 2003.
30. K-T Kuo, S. Phuvoravan, T. Guven, L. Sudarsan, H. S. Chang, S. Bhattacharjee, and M. A. Shayman. “Fast Timescale Control for MPLS Traffic Engineering”, *Proceedings of Globecom 2003*, pages 3108–3114, 2003.
31. Bobby Bhattacharjee, Sudarshan Chawathe, Vijay Gopalakrishnan, Pete Keleher, and Bujor Silaghi. “Efficient Peer-To-Peer Searches Using Result-Caching”, *Second International Workshop on Peer-to-Peer Systems (IPTPS’03)*, 2003.
32. Ruggero Morselli, Jonathan Katz, and Bobby Bhattacharjee. “A Game-Theoretic Framework for Analyzing Trust-Inference Protocols”, *In the Proceedings of the Second Workshop on the Economics of Peer-to-Peer Systems*, June 2004.
33. Bujor Silaghi, Pete Keleher, and Bobby Bhattacharjee. “Multi-Dimensional Quorum Sets for Read-Few Write-Many Replica Control Protocols”, *In Proceedings of the IEEE/ACM CCGRID, 4th Fourth International Workshop on Global and Peer-to-Peer Computing (GP2PC)*, April 2004.
34. Suman Banerjee, Seungjoon Lee, Ryan Braud, Bobby Bhattacharjee, and Aravind Srinivasan. “Scalable Resilient Media Streaming”, *In Proceedings of ACM NOSSDAV’04*, pages 4–9, 2004.
35. Seungjoon Lee, Suman Banerjee, and Bobby Bhattacharjee. “The Case for a Multi-hop Wireless Local Area Network”, *In Proceedings of IEEE Infocom*, pages 894–905, 2004.
36. T.Guven, C. Kommareddy, R.J. La, M.A. Shayman, and B. Bhattacharjee. “Measurement Based Optimal Multi-path Routing”, *In Proceedings of IEEE Infocom*, pages 187–196, March 2004.
37. Ruggero Morselli, Bobby Bhattacharjee, Jonathan Katz, and Pete Keleher. “Trust-Preserving Set Operations”, *In Proceedings of IEEE Infocom*, pages 2231–2241, March 2004.
38. Rob Sherwood, Ryan Braud, and Bobby Bhattacharjee. “Slurpie: A Cooperative Bulk Data Transfer Protocol”, *In Proceedings of IEEE Infocom*, pages 941–951, March 2004.
39. V. Gopalakrishnan, B. Silaghi, B. Bhattacharjee, and P. Keleher. “Adaptive Replication in Peer-to-Peer Systems”, *In Proceedings of IEEE International Conference on Distributed Computing Systems (ICDCS)*, pages 360–369, March 2004.
40. David Hovemeyer, Jeff Hollingsworth, and Bobby Bhattacharjee. “Running on the Bare Metal with GeekOS”, *In Proceedings of the Technical Symposium on Computer Science Education (SIGCSE)*, pages 315–319, March 2004.
41. Bujor Silaghi, Vijay Gopalakrishnan, Bobby Bhattacharjee, and Pete Keleher. “Hierarchical Routing with Soft-State Replicas in TerraDir”, *In Proceedings of the 18th IPDPS Conference*, pages 48–57, April 2004.

42. Rob Sherwood, Bobby Bhattacharjee, and Ryan Braud. “Misbehaving TCP Receivers Can Cause Internet-Wide Congestion Collapse”, *Proceedings of Computer and Communications Security (CCS)*, pages 383–392, 2005.
43. Ruggero Morselli, Bobby Bhattacharjee, Michael A. Marsh, and Aravind Srinivasan. “Efficient Lookup on Unstructured Topologies”, *Principles of Distributed Computing*, pages 77–86, 2005.
44. Seungjoon Lee, Bobby Bhattacharjee, and Suman Banerjee. “Efficient Geographic Routing in Multihop Wireless Networks”, *ACM MobiHoc 2005*, pages 230–241, 2005.
45. T. Guven, R. J. La, M. Shayman, and B. Bhattacharjee. “Measurement-based Multipath Multicast,” *In IEEE Global Internet Symposium*, pages 2803–2808, 2005.
46. Vahid Tabatabaee, Bobby Bhattacharjee, Richard La, and Mark Shayman. “Differentiated Traffic Engineering for QoS Provisioning”, *In the Proceedings of INFOCOM’05*, pages 2349–2359, 2005.
47. J. S. Kim, B. Nam, P. Keleher, M. Marsh, B. Bhattacharjee, and A. Sussman. “Resource Discovery Techniques in Distributed Desktop Grid Environments”, *Proceedings of the 7th IEEE/ACM International Conference on Grid Computing - GRID 2006*, pages 9–16, September 2006. *Best paper award.*
48. Cristian Lumezanu, Neil Spring, and Bobby Bhattacharjee. “Decentralized Message Ordering for Publish/Subscribe Systems”, *ACM/IFIP/Usenix 7th International Middleware Conference*, 2006.
49. Abhishek Kashyap, Samrat Bhattacharjee, Richard La, Mark Shayman, and Vahid Tabatabaee. “Single-Path Routing of Time-varying Traffic”, *In the Proceedings of Globecom*, 2006.
50. Vasile Gaburici, Peter Keleher, and Bobby Bhattacharjee. “File System Support for Collaboration in the Wide Area”, *Proceedings of IEEE International Conference on Distributed Computing Systems (ICDCS)*, pages 26–36, 2006.
51. J. H. Li, M. Yu, R. Levy, and B. Bhattacharjee. “A Scalable Key Management and Clustering Scheme for Ad Hoc Networks”, *Proceedings of InfoScale*, pages 28–38, Hong Kong, 2006.
52. Vijay Gopalakrishnan, Bobby Bhattacharjee, and Peter Keleher. “Distributing Google”, *In 2nd IEEE International Workshop on Networking Meets Databases (NetDB’06)*, pages 33–39, April 2006.
53. Dave Levin, Rob Sherwood, and Bobby Bhattacharjee. “Fair File Swarming with FOX”, *Proceedings of the International Workshop on Peer-to-Peer Systems (IPTPS’06)*, 2006.
54. Alan Mislove, Massimiliano Marcon, Krishna Gummadi, Peter Druschel, and Bobby Bhattacharjee. “Measurement and Analysis of Online Social Networks”, *In Proceedings of the ACM/Usenix Internet Measurement Conference (IMC 2007)*, pages 29–42, 2007.

55. Jik-Soo Kim, Peter Keleher, Michael Marsh, Bobby Bhattacharjee, and Alan Sussman. “Using Content-Addressable Networks for Load Balancing in Desktop Grids”, *In Sixteenth IEEE International Symposium on High-Performance Distributed Computing (HPDC)*, pages 189–198, 2007.
56. Vijay Gopalakrishnan, Ruggero Morselli, Bobby Bhattacharjee, Peter J. Keleher, and Aravind Srinivasan. “Distributed Ranked Search”, *14th Annual IEEE International Conference on High Performance Computing (HiPC)*, pages 7–20, 2007. *Best paper award*.
57. Animesh Nandi, Aditya Ganjam, Peter Druschel, T. S. Eugene Ng, Ion Stoica, Hui Zhang, and Bobby Bhattacharjee. “A Shared Control Plane for Overlay Multicast”, *Fourth Usenix Symposium on Networked Systems Design and Implementation (NSDI 2007)*, 2007.
58. Seungjoon Lee, Dave Levin, Vijay Gopalakrishnan, and Bobby Bhattacharjee. “Backbone Construction in Selfish Wireless Networks”, *Proceedings of SIGMETRICS*, pages 121–132, 2007.
59. Jik-Soo Kim, Beomseok Nam, Michael A. Marsh, Peter J. Keleher, Bobby Bhattacharjee, Derek Richardson, Dennis Wellnitz, and Alan Sussman. “Creating a Robust Desktop Grid using Peer-to-Peer Services”, *Proceedings of NSF Next Generation Software Workshop (NSFNGS)* (Appears with proceedings of IPDPS 2007), pages 1–7, 2007.
60. Vahid Tabatabaee, Abhishek Kashyap, Bobby Bhattacharjee, Richard La, and M. Shayman. “Robust Routing with Unknown Traffic Matrices”, *IEEE INFOCOM Minisymposiums*, pages 2336–2440, 2007.
61. Adam Bender, Neil Spring, Dave Levin, and Bobby Bhattacharjee. “Accountability as a Service”, *USENIX Workshop on Steps to Reducing Unwanted Traffic on the Internet (SRUTI)*, 2007.
62. Dave Levin, Adam Bender, Cristian Lumezanu, Neil Spring, and Bobby Bhattacharjee. “Boycotting and Extorting Nodes in an Internetwork”, *Workshop on the Economics of Networked Systems and Incentive-Based Computing*, 2007.
63. B. Bhattacharjee, R. Rodrigues, and P. Kouznetsov. “Secure Lookup without (Constrained) Flooding”, *Workshop on Recent Advances on Intrusion-Tolerant Systems (WRAITS)*, 2007.
64. R. Rodrigues, P. Kouznetsov, and B. Bhattacharjee. “Large-Scale Byzantine Fault Tolerance: Safe but Not Always Live”, *In the Third Workshop on Hot Topics in System Dependability (HotDep’07)*, 2007.
65. Ruggero Morselli, Bobby Bhattacharjee, Jonathan Katz and Michael Marsh. “Exploiting Approximate Transitivity of Trust”, *In Fourth International Conference on Broadband Communications, Networks, and Systems, 2007 (BroadNets 2007)*, pages 515–524, 2007.
66. Jik-Soo Kim, Beomseok Nam, Michael Marsh, Peter Keleher, Bobby Bhattacharjee, and Alan Sussman. “Integrating Categorical Resource Types into a P2P Desktop Grid System”, *In Proceedings of the 9th IEEE/ACM International Conference on Grid Computing (Grid 2008)*, 2008.

67. Dave Levin, Katrina LaCurts, Neil Spring, and Bobby Bhattacharjee. “BitTorrent is an Auction: Analyzing and Improving BitTorrent’s Incentives”, *In Proceedings of Sigcomm*, 2008.
68. Alan Mislove, Hema Swetha Koppula, Krishna P. Gummadi, Peter Druschel, and Bobby Bhattacharjee. “Growth of the Flickr social network”, *In Proceedings of the 1st ACM SIGCOMM Workshop on Social Networks (WOSN’08)*, Seattle, WA, 2008.
69. Dave Levin, Randolph Baden, Cristian Lumezanu, Neil Spring, and Bobby Bhattacharjee. “Motivating Participation in Internet Routing Overlays”, *In NetEcon 2008 (Workshop on the Economics of Networks, Systems, and Computation)*, 2008.
70. Bo Han, Lusheng Ji, Seungjoon Lee, Bobby Bhattacharjee, Robert R. Miller. “All Bits Are Not Equal – A Study of IEEE 802.11 Communication Bit Errors”, INFOCOM 2009.
71. Bo Han, Lusheng Ji, Seungjoon Lee, Robert R. Miller, Bobby Bhattacharjee. “Channel Access Throttling for Overlapping BSS Management”. IEEE ICC, June 2009.
72. Vijay Gopalakrishnan, Bobby Bhattacharjee, K. K. Ramakrishnan, Rittwik Jana, Divesh Srivastava. “CPM: Adaptive Video-on-Demand with Cooperative Peer Assists and Multicast.” IEEE INFOCOM 2009.
73. Bo Han, Lusheng Ji, Seungjoon Lee, Robert Miller, Bobby Bhattacharjee, “Channel Access Throttling for Improving WLAN QoS”, IEEE Secon, Rome, Italy, June 2009
74. Cristian Lumezanu and Randy Baden and Dave Levin and Neil Spring and Bobby Bhattacharjee, “Symbiotic Relationships in Internet Routing Overlays”, Usenix-NSDI, 2009.
75. Cristian Lumezanu and Randy Baden and Neil Spring and Bobby Bhattacharjee, “Triangle Inequality and Routing Policy Violations in the Internet”, Passive and Active Measurement Conference (PAM), 2009.
76. Randy Baden, Adam Bender, Neil Spring, Bobby Bhattacharjee, Daniel Starin, “Persona: An Online Social Network with User-Defined Privacy”, Proceedings of SIGCOMM, August 2009
77. Cristian Lumezanu, Randolph Baden, Neil Spring, Bobby Bhattacharjee “Triangle Inequality Variations in the Internet”, Proceedings of IMC, November 2009
78. Randy Baden, Neil Spring, Bobby Bhattacharjee “Identifying close friends on the Internet”, The Workshop on Hot Topics in Networks (ACM Hotnets), 2009
79. Adam Bender, Rob Sherwood, Derek Monner, Nate Goergen, Neil Spring, Bobby Bhattacharjee, “Fighting Spam with the NeighborhoodWatch DHT”, Proceedings of IEEE INFOCOM, April 2009
80. Cristian Lumezanu, Randolph Baden, Dave Levin, Neil Spring, Bobby Bhattacharjee, “Symbiotic Relationships in Internet Routing Overlays”, Proceedings of USENIX NSDI, April 2009

81. Animesh Nandi, Bobby Bhattacharjee, Peter Druschel, “What a mesh: understanding the design tradeoffs for streaming multicast”, Extended Abstract, ACM SIGMETRICS Performance Evaluation, 2009.
82. Bo Han, Aaron Schulman, Francesco Gringoli, Neil Spring, Bobby Bhattacharjee, Lorenzo Nava, Lusheng Ji, Seungjoon Lee, and Robert Miller, “Maranello: Practical partial packet recovery for 802.11”, Proceedings of USENIX NSDI 2010.
83. Cristian Lumezanu, Dave Levin, Bo Han, Neil Spring, and Bobby Bhattacharjee, “Don’t love thy nearest neighbor”, International Workshop on Peer-to-Peer Systems (IPTPS), April 2010.
84. Cristian Lumezanu, Katherine Guo, Neil Spring, Bobby Bhattacharjee, “The Effect of Packet Loss on Redundancy Elimination in Cellular Wireless Networks”, ACM Sigcomm Internet Measurement Conference (IMC), 2010.
85. Satinder Pal Singh, Randolph Baden, Choon Lee, Bobby Bhattacharjee, Richard J. La, Mark Shayman, “IP Geolocation in Metropolitan Areas”, Extended Abstract, Proceedings of ACM Sigmetrics, 2011.
86. Matthew Lentz, Dave Levin, Jason Castonguay, Neil Spring, Bobby Bhattacharjee, D-mystifying the D-root address change, IMC (International Measurement Conference), 2013.
87. Lentz, Matthew and Erdélyi, Viktor and Aditya, Paarijaat and Shi, Elaine and Druschel, Peter and Bhattacharjee, Bobby, ”SDDR: Light-Weight, Secure Mobile Encounters”, USENIX Security Symposium, 2014.
88. Aditya, Paarijaat and Erdélyi, Viktor and Lentz, Matthew and Shi, Elaine and Bhattacharjee, Bobby and Druschel, Peter, “EnCore: Private, Context-based Communication for Mobile Social Apps”, International Conference on Mobile Systems, Applications, and Services (MobiSys), 2014.
89. Aditya, Paarijaat and Bhattacharjee, Bobby and Druschel, Peter and Erdélyi, Viktor and Lentz, Matthew, Brave New World: Privacy Risks for Mobile Users, Workshop on Security and Privacy Aspects of Mobile Environments (SPME), 2014.
90. Dave Levin, Youndo Lee, Luke Valenta, Zhihao Li, Victoria Lai, Cristian Lumezanu, Neil Spring, Bobby Bhattacharjee “Alibi Routing” Proceedigs of ACM SIGCOMM, 2015.
91. Matthew Lentz, James Litton, Bobby Bhattacharjee “Drowsy Power Management” Proceedings of Symposium on Operating Systems Principles (SOSP), 2015.
92. Raul Herbster, Scott DellaTorre, Peter Druschel, Bobby Bhattacharjee. “Privacy Capsules: Preventing Information Leaks by Mobile Apps” Proceedings of Mobisys, 2016.
93. Paarijaat Aditya, Rijurekha Sen, Seong Joon Oh, Rodrigo Benenson, Bobby Bhattacharjee, Peter Druschel, Tongtong Wu, Mario Fritz, Bernt Schiele. “I-Pic: A Platform for Privacy-Compliant Image Capture” Proceedings of Mobisys, 2016.

94. Vaibhav Singh, Matthew Lentz, Bobby Bhattacharjee, Richard La, Mark Shayman.
95. ‘Dynamic Frequency Resource Allocation in Heterogeneous Cellular Networks’
96. roceedings of IEEE TMC 2016 (IEEE Transactions on Mobile Computing)
97. James Litton, Anjo Vahldiek-Oberwagner, Eslam Elnikety, Deepak Garg, Bobby Bhattacharjee, Peter Druschel “Light-weight Contexts: An OS Abstraction for Safety and Performance” Proceedings of OSDI, 2016.
98. Zhihao Li, Dave Levin, Neil Spring, Bobby Bhattacharjee  Bobby Bhattacharjee. “Internet anycast: performance, problems, and potential” Proceedings of SIGCOMM, 2018.
99. Matthew Lentz, Rijurekha Sen, Peter Druschel, Bobby Bhattacharjee. “SeCloak: ARM TrustZone-based Mobile Peripheral Control” Proceedings of Mobisys 2018 (Conference on Mobile Systems, Applications, and Services)
100. Viktor Erdelyi, Trung-Kien Le, Bobby Bhattacharjee, Peter Druschel, Nobutaka Ono. “Sonoloc: Scalable positioning of commodity mobile devices.” In Proceedings of the Sixteenth International Conference on Mobile Systems, Applications, and Services (MobiSys 2018).
101. Lillian Tsai, Roberta De Viti, Matthew Lentz, Stefan Saroiu, Peter Druschel, Bobby Bhattacharjee. “enClosure: Group Communication via Encounter Closures” Proceedings of Mobisys 2019
102. Composing Abstractions using the null-Kernel James Litton, Deepak Garg, Peter Druschel, Bobby Bhattacharjee HotOS, 2019
103. Sergi Delgado-Segura, Surya Bakshi, Cristina Perez-Sola, James Litton, Andrew Pachulski Andrew Miller, Bobby Bhattacharjee “TxProbe: Discovering Bitcoin’s Network Topology Using Orphan Transactions” Financial Crypto, 2019
104. Venkat Arun, Aniket Kate, Deepak Garg, Peter Druschel, Bobby Bhattacharjee. “Finding Safety in Numbers with Secure Allegation Escrows” NDSS 2020 (to appear)

2.4 Technical Reports and Invited Papers

1. Ellen W. Zegura, Kenneth. L. Calvert, and Samrat Bhattacharjee. “Tera-op networking: Local adaptation to congestion”, In *Gigabit Networking Workshop*, 1996.
2. Samrat Bhattacharjee, Kenneth L. Calvert, and Ellen W. Zegura. “Network Support for Multicast Video Distribution”, Technical Report GIT-CC-98-16, College of Computing, Georgia Institute of Technology, 1996.
3. Samrat Bhattacharjee. “Self-Organizing Wide-Area Network Caches”, Technical Report GIT-CC-97-31, College of Computing, Georgia Institute of Technology, 1996.

4. Zongming Fei, Samrat Bhattacharjee, Ellen W. Zegura, and Mostafa H. Ammar. “A Novel Server Selection Technique for Improving the Response Time of a Replicated Service”, Technical Report GIT-CC-97-24, College of Computing, Georgia Institute of Technology, 1996.
5. Samrat Bhattacharjee, Mostafa Ammar, Ellen Zegura, Viren Shah, and Zongming Fei. “Application-Layer Anycasting”, Technical Report GIT-CC-96-25, College of Computing, Georgia Institute of Technology, 1996.
6. Samrat Bhattacharjee, Kenneth L. Calvert, and Ellen W. Zegura. “An Architecture for Active Networking”, Technical Report GIT-CC-96-20, College of Computing, Georgia Institute of Technology, 1996.
7. Samrat Bhattacharjee, Kenneth L. Calvert, and Ellen W. Zegura. “On Active Networking and Congestion”, Technical Report GIT-CC-96-02, College of Computing, Georgia Institute of Technology, 1996.
8. Samrat Bhattacharjee, Ellen W. Zegura, and Kenneth L. Calvert. “High Performance Web: An Application for Wide Area Caching”, In *Gigabit Networking Workshop*, 1997.
9. Samrat Bhattacharjee and Gisli Hjalmytsson. “Control-on-Demand: A Flow Oriented Approach towards Active Networking”, *AT&T Labs Technical Memorandum*, 1997.
10. Samrat Bhattacharjee, Kenneth L. Calvert, and Ellen W. Zegura. “Improving the Quality of Best Effort Service”, Technical Report GIT-CC-98-31, College of Computing, Georgia Institute of Technology, 1998.
11. S. Bhattacharjee and M. W. McKinnon. “Performance of Application-Specific Buffering Schemes for Active Networks”, Technical Report GIT-CC-98-17, College of Computing, Georgia Institute of Technology, 1998.
12. S. Merugu, S. Bhattacharjee, Y. Chae, M. Sanders, K. Calvert, and E. Zegura. “Bowman and CANEs: Implementation of an Active Network”, *Proceedings of the Thirty-Seventh Annual Allerton Conference on Communication, Control and Computing*, Monticello, Illinois, September 1999 (invited paper).
13. Ugur Cetintemel, Peter J. Keleher, and Bobby Bhattacharjee. “A Security Infrastructure for Mobile Transactional Systems”, University of Maryland, Dept. of Computer Science Technical Report, CS-TR-4171, 2000.
14. Peter J. Keleher, Bobby Bhattacharjee, Kuo Kuo-Tung, and Ugur Cetintemel. “A Security Infrastructure for Mobile Transactional Systems”, University of Maryland, Dept. of Computer Science Technical Report, CS-TR-4077, 2000.
15. Suman Banerjee and Bobby Bhattacharjee. “Scalable Secure Group Communication over IP Multicast”, University of Maryland, Dept. of Computer Science Technical Report, CS-TR-4252, 2001.

16. Suman Banerjee and Samrat Bhattacharjee. “Spatial Clustering for IP Multicast: Algorithms and an Application”, University of Maryland, Dept. of Computer Science Technical Report, CS-TR-4177, 2001.
17. Suman Banerjee and Bobby Bhattacharjee. “Analysis of the NICE Application Layer Multicast Protocol”, University of Maryland, Dept. of Computer Science Technical Report, CS-TR-4380, 2002.
18. K-T Kuo, S. Phuvoravan, T. Guven, L. Sudarsan, S. Bhattacharjee, and M. A. Shayman. “Fast Time Scale Control for MPLS Traffic Engineering”, University of Maryland, Dept. of Computer Science Technical Report, CS-TR-4351, 2002.
19. Suman Banerjee, Bobby Bhattacharjee, and Christopher Kommareddy. “Scalable Application Layer Multicast”, University of Maryland, Dept. of Computer Science Technical Report, CS-TR-4373, 2002.
20. Bobby Bhattacharjee, Pete Keleher, and Bujor Silaghi. “The Design of TerraDir”, University of Maryland, Dept. of Computer Science Technical Report, CS-TR-4299, 2002.
21. Suman Banerjee, Bobby Bhattacharjee, and Srinivasan Parthasarathy. “A Protocol for Scalable Application Layer Multicast”, University of Maryland, Dept. of Computer Science Technical Report, CS-TR-4278, 2002.
22. Vijay Gopalakrishnan, Bujor Silaghi, Bobby Bhattacharjee, and Pete Keleher. “Adaptive Replication in Peer-to-Peer Systems”, University of Maryland, Dept. of Computer Science Technical Report, CS-TR-4515, 2003.
23. Seungjoon Lee, Suman Banerjee, and Bobby Bhattacharjee. “The Case for a Multi-hop Wireless Local Area Network”, University of Maryland, Dept. of Computer Science Technical Report, CS-TR-4504, 2003.
24. Christopher Kommareddy, Tuna Guven, Bobby Bhattacharjee, Richard La, and Mark Shayman. “Intradomain Overlays: Architecture and Applications”, University of Maryland, Dept. of Computer Science Technical Report, CS-TR-4501, 2003.
25. Tuna Guven, Chris Kommareddy, Richard J. La, Mark A. Shayman, and Bobby Bhattacharjee. “Measurement Based Optimal Multi-path Routing”, University of Maryland, Dept. of Computer Science Technical Report, CS-TR-4500, 2003.
26. Ruggero Morselli, Bobby Bhattacharjee, Jonathan Katz, and Pete Keleher. “Trust-Preserving Set Operations”, University of Maryland, Dept. of Computer Science Technical Report, CS-TR-4499, 2003.
27. Suman Banerjee, Ryan Braud, Seungjoon Lee, Bobby Bhattacharjee, and Aravind Srinivasan. “Scalable Resilient Media Streaming”, University of Maryland, Dept. of Computer Science Technical Report, CS-TR-4482, 2003.
28. Bujor Silaghi, Pete Keleher, and Bobby Bhattacharjee. “Multi-dimensional Quorum Sets for Read-Few Write-Many Replica Control Protocols”, University of Maryland, Dept. of Computer Science Technical Report, CS-TR-4440, 2003.

29. Tuna Guven, Richard La, Mark Shayman, and Bobby Bhattacharjee. “Measurement-based Multicast on an Overlay Architecture”, University of Maryland, Dept. of Computer Technical Report, CS-TR-4603, 2004.
30. Vijay Gopalakrishnan, Bobby Bhattacharjee, Sudarshan Chawathe, and Pete Keleher. “Efficient Peer-to-Peer Namespace Searches”, University of Maryland, Dept. of Computer Technical Report, CS-TR-4568, 2004.
31. Rob Sherwood, Bobby Bhattacharjee, and Ryan Braud. “Misbehaving TCP Receivers Can Cause Internet-Wide Congestion Collapse”, University of Maryland, Dept. of Computer Science Technical Report CS-TR-4737, 2005.
32. Jik-Soo Kim, Bobby Bhattacharjee, Peter Keleher, and Alan Sussman. “Matching Jobs to Resources in Distributed Desktop Grid Environments”, University of Maryland, Dept. of Computer Science Technical Report CS-TR-4791, 2006.
33. Ruggero Morselli, Bobby Bhattacharjee, Jonathan Katz, and Michael Marsh. “Key-Chains: A Decentralized Public-Key Infrastructure”, University of Maryland, Dept. of Computer Science Technical Report CS-TR-4788, 2006.
34. Ruggero Morselli, Bobby Bhattacharjee, Michael Marsh, and Aravind Srinivasan. “Efficient Lookup on Unstructured Topologies”, University of Maryland, Dept. of Computer Science Technical Report CS-TR-4772, 2006.
35. Seungjoon Lee, Bobby Bhattacharjee, and Suman Banerjee. “Efficient Geographic Routing in Multihop Wireless Networks”, University of Maryland, Dept. of Computer Science Technical Report CS-TR-4625, 2006.
36. Vijay Gopalakrishnan, Ruggero Morselli, Bobby Bhattacharjee, Peter Keleher, and Aravind Srinivasan. “Ranking Search Results in Peer-to-Peer Systems”, University of Maryland, Dept. of Computer Science Technical Report CS-TR-4779, 2006.
37. Randy Baden, Adam Bender, Dave Levin, Rob Sherwood, Neil Spring, and Bobby Bhattacharjee. “A Secure DHT via the Pigeonhole Principle”, University of Maryland, Dept. of Computer Science Technical Report CS-TR-4884, 2007.
38. Jik-Soo Kim, Peter Keleher, Michael Marsh, Bobby Bhattacharjee, and Alan Sussman. “Using Content-Addressable Networks for Load Balancing in Desktop Grids”, University of Maryland, Dept. of Computer Science Technical Report CS-TR-4863, 2007.

2.5 Tutorials, Talks, Abstracts, and Other Professional Papers Presented

- *Privacy by Design*, NEC Labs, Princeton, November 2012.
- *Systems without Cooperation*, South China University of Technology, October 2008.
- *Systems without Cooperation*, Sichuan University, October 2008.
- *Systems without Cooperation*, University of Maryland, September 2008.

- *Decentralized Applications on the Internet*, University of Lisbon, Portugal, July 2006.
- *Decentralized Applications on the Internet*, Bell Labs, January 2006, Murray Hill, New Jersey.
- *An Overview of Decentralized Applications*, at the *Algorithms in Networking* Workshop, FSCCTS 2005, Hyderabad, India.
- *Security Architectures for Peer-to-Peer Applications*, at the Marconi Foundation Video P2P Conference, Columbia University, 2004, New York City, NY.
- *Replication and Search in Distributed Namespaces*, at IBM Research, 2004, Hawthorne, NY.
- Invited Panelist, *Network Security: How Good Does it Have to Be?* at IEEE INFOCOM, 2003, San Francisco, CA.
- *P5: A Protocol for Scalable Anonymous Communications*, at IEEE S&P, 2002, Oakland, CA.
- *Cooperative Peer Groups in NICE*, at IEEE INFOCOM, April 2003, San Francisco, CA.
- *Overlay and P2P Systems: Protocols, Applications, and Analysis*, Tutorial (with Dan Rubenstein) at Networking Group Communications (NGC '02), October 2002, Boston, MA.
- *Cooperative Peer Groups in NICE*, Invited talk at BBN Technologies, Cambridge, MA, October 2002.
- *Finding Close Friends over the Internet*, at the International Conference on Network Protocols (ICNP), 2001, Riverside, CA.
- *Adaptive Network Processing*, at the Washington University Gigabit Switch Seminar, Washington University at St. Louis, St. Louis, January, 2001.
- *Active Networks: A Possible Future for the Internet?*, Invited Talk to the Washington DC/Northern VA Chapter for the IEEE/Microwave Theory and Techniques Society, April 2000.
- Invited Panelist at Gigabit Networking Workshop, San Francisco, CA, 1998 and International Communications Conference, Atlanta, GA 1998.
- *LIANE - Composition for Active Networks*, at IEEE Computer Communications Workshop, September 1998, Oxford, MS.
- *Self-Organizing Wide Area Network Caches*, at IEEE INFOCOM 1998, San Francisco, CA.
- *Reasoning about Active Networks*, at ICNP 1998, Austin, TX.
- *Finding the Best Server within the Application-Layer Anycasting Architecture*, at IEEE INFOCOM 98, San Francisco, CA.

- *High Speed Web: An Application for Active Caching*. Presented at Gigabit Networking Workshop '97, March 1997, Kobe, Japan.
- *Application-Layer Anycasting*, at IEEE INFOCOM 1997, Kobe, Japan.
- *Active Networking and the End-to-End Argument*, at IEEE ICNP'97, Atlanta, GA.
- *Tera-Op Networking: Local Adaptation to Congestion*. Presented at Gigabit Networking Workshop '96, March 1996, San Francisco, CA.

2.6 Patents

1. *Scalable wide-area upload system and method*, Leana Golubchik, William C. Cheng, Samir Khuller, Samrat Bhattacharjee, and Cheng-Fu Chou. United States Patent # 7,181,623. Granted: February 20, 2007.
2. *Method for encoding frame data*, Lusheng Ji, Samrat Bhattacharjee, Bo Han, Seungjoon Lee, Robert Miller. United States Patent # 7,940,850. Granted: May 10, 2011.
3. *Detection of distributed denial of service attacks in autonomous system domains*, Chris Kommareddy, Samrat Bhattacharjee, Mark Shayman, Richard La. United States Patent # 8,397,284. Granted: March 12, 2013.

2.7 Contracts and Grants

1. "EAGER: Decomposing Operating Systems for Better Control over Policy and Privacy", *National Science Foundation*, PI,
2. "LTS - Securing Critical Networking Infrastructure: DNS Root Servers", *Department of Defense*, PI, (Co-PIs: Neil Spring and David Levin), 2014-2015, \$199,799.00
3. "Interference Management in Heterogeneous Networks", *Air Force Research Laboratory*, Co-PI, (PI: Mark Shayman), 2012-2013, \$358,054.
4. "University Partnership with the Laboratory for Telecommunications Science", *Department of Defense*, Co-PI, 2010-2013, (PI: Joseph Jaja) \$896,814.
5. "University Partnership with the Laboratory for Telecommunications Science", *Department of Defense*, Co-PI, 2010-2013, (PI: Joseph Jaja) \$496,228.
6. "Privacy Preserving Social Systems", *National Science Foundation*, PI, Co-PIs: Neil Spring, Jonathan Katz, 2010 – 2013, \$880,000.
7. "Greed Resistant Protocols", *National Science Foundation*, Co-PI, (PI: Neil Spring), 2009 – 2012, \$499,344.
8. "An Integrated Approach to Computing Capacity and Developing Efficient Cross-Layer Protocols for Wireless Networks", *National Science Foundation*, Co-PI, (Principal Investigator: Aravind Srinivasan), September 2006 – September 2009, \$365,000.

9. “A Postmodern Internetwork Architecture”, *National Science Foundation*, Co-PI, Principal Investigator: Neil T. Spring, September 2006 – September 2009, \$400,000.
10. “Robust Grid Computing using Peer-to-Peer Services”, *NASA*, Co-PI. Principal Investigator: Alan Sussman. Other co-PIs: P. Keleher, D. Richardson, February 2006 – February 2009, \$1,008,251.
11. “A Wide-Area Event Notification System for MENTER”, *Laboratory for Telecommunication Sciences, National Security Agency*, January 2001 – August 2008, \$625,000.
12. “Employing Peer-to-Peer Services for Robust Grid Computing”, Co-PI. Principal Investigator: Alan Sussman. Other co-PIs: P. Keleher, D. Richardson, September 2005 – August 2006, \$60,000.
13. “Resilient Storage and Querying in Decentralized Networks”, *National Science Foundation*, Principal Investigator (Co-PI: Aravind Srinivasan, Sudarshan Chawathe, Jonathan Katz, Michael Marsh), Fall 2004 – Fall 2007, \$720,000.
14. Alfred P. Sloan Jr. Fellowship, September 2004 – September 2007, \$40,000.
15. “Distributed Trust Computations for Decentralized Systems”, *National Science Foundation*, Principal Investigator (Co-PI: Jonathan Katz), Fall 2003 – Fall 2006, \$375,000.
16. “CAREER: Adaptive Network Processing”, *National Science Foundation CAREER Award*, Fall 2001 – Spring 2006, \$500,000.
17. “Decentralized Directories for the Internet”, *National Science Foundation*, Principal Investigator (Co-PI: P. Keleher), Fall 2001 – Spring 2004, \$710,000.
18. “Parametric Design of Embedded Real-Time Systems”, *National Science Foundation*, Principal Investigator, Summer 2002 – Summer 2003. (Original PI: Richard Gerber, Fall 1998 – Summer 2002), \$200,154.
19. *Washington University Gigabit Switch Kit*. NSF, Washington University at St. Louis, Fall 1999 (Equipment only).

2.8 Fellowships, Prizes and Awards

1. Department of Computer Science Faculty Award for Teaching Excellence, 2012.
2. Department of Computer Science Faculty Award for Teaching Excellence, 2008.
3. Best paper award, 14th Annual IEEE International Conference on High Performance Computing (HiPC), 2007; paper co-authored with Vijay Gopalakrishnan, Ruggero Morselli, Peter J. Keleher, and Aravind Srinivasan.
4. Best paper award, 7th IEEE/ACM Conference on Grid Computing, 2006; paper co-authored with Jiksoo Kim, Byomsuk Nam, Peter Keleher, Michael Marsh, and Alan Sussman.

5. Alfred P. Sloan Jr. Fellowship, 2004.
6. Department of Computer Science Faculty Award for Teaching Excellence, 2004.
7. NSF CAREER award, 2001.
8. Recipient of Distinguished Teaching Assistant award from College of Computing, Georgia Tech, Spring 1997.

2.9 Editorial Boards and Reviewing Activities for Learned Publications

Reviewer for

ACM/IEEE Transactions on Networking

IEEE Journal on Selected Areas in Communications

Computer Communications Journal (Special Issue on Network Security)

ACM Transactions on Computer Systems

Performance Evaluation Journal

Computer Communications Review

European Transactions on Telecommunications

IEEE Transactions on Parallel and Distributed Systems

ACM Transactions on Internet Technology

Virtually all conferences in Networking and Systems including SOSP, OSDI, SIGCOMM, Sigmetrics, INFOCOM, SCW, DISC (formerly WDAG), Global Internet Conference, Infocom, IC3N, ICDCS, ICNP, ICPP, ICS, OpenArch, and WWW.

2.10 Research Software

1. *Odyssey: An active networking platform.* This distribution includes complete source and documentation for the Bowman Node OS and the CANEs Execution Environment. Released on the Internet, Summer 1999.
2. *NICE protocol simulator.* This distribution includes source for simulators of the NICE multicast protocols and complete implementation of the NICE protocols for video multicast. Released on the Internet, 2002.
3. *Slurpie.* This distribution includes the entire source code for a file-swarming system. Released on the Internet, 2004.
4. *OptAck Random Segment skip patch.* This software fixes a protocol fault (for Linux kernel versions 2.4 and 2.6). The fault exists in all known versions of TCP. Released on the Internet, 2005.
5. *Local Minima Search (LMS).* LMS is a protocol for unstructured search using virtual namespaces in distributed environments. Released on the Internet, 2006.

6. *KeyChains PKI*. KeyChains is a web-of-trust public key distribution/discovery system; it is built based on LMS local minima search algorithm, and uses CODEX libraries (from Cornell). Released on the Internet, 2006.
7. *Distributed Grid Software* The Distributed Grid software implements a complete distributed job matching system. The software suite is currently being field tested by researchers in Astronomy, and is available upon request, 2007.
8. *Cryptographic library for Chit-based access*. Developed cryptographic library for “chit”-based security. Library is used for different chit-based applications, including a filesystem and a distributed calendar application. Code available upon request, 2007.
9. *CPM on-demand video service*. The CPM software includes a novel video server and associated client software (and other supporting code) for implementing Cooperative Peer-Assisted Multicasting. Co-implemented the full software suite at AT&T Research. Code available upon request, 2007.
10. *IBOBSP*. IBOBSP is an in-network platform targeted towards reducing latency in interactive applications (in particular, games). The software distribution includes the in-network server pieces, and several graphical test applications and games. Co-implemented the full IBOBSP suite at AT&T Research. Code available upon request, 2007.

3 Teaching

3.1 Teaching Awards and Other Special Recognition

1. Teaching Excellence Award for Faculty, Department of Computer Science, Spring 2008.
2. Teaching Excellence Award for Faculty, Department of Computer Science, Spring 2004.
3. Distinguished Teaching Assistant, College of Computing, Georgia Tech, Spring 1997.

3.2 Advising: Research Advisor

3.2.1 Undergraduate

- Sebastian Gomez, Fall 2010 – Spring 2011.
- Chris Heistand, Fall 2010 – Spring 2011.
- Robert Kiefer, Spring 2009 – Summer 2010.
- Anika Cartas, Summer 2008 – Spring 2009.
- Katrina LaCurts, Fall 2007 – Summer 200.
- David Renie, Spring – Fall 2004.
- Ryan Evans Braud, Graduated Spring 2004.
- Mentor for Joseph Barrett, Colin Dixon, Tianzhou Duan, Kevin Genson, Bryant McIver, Ben Roseman, as part of the University of Maryland GEMSTONE program. Project title: *Anonymous Communications*, 2002-2005.

3.2.2 Masters

- Randolf Baden, Spring 2008.
- Chunyuan Liao, Fall 2004.
- Vijay Gopalakrishnan, Spring 2003.
- Kuo-Tung Kuo, Spring 2003.
- Dave Hovemeyer, Fall 2001.
- Vaibhav Kumar (ECE), Spring 2001.
- William Shapiro, Spring 2000.

3.2.3 Doctoral (completed)

- Suman Banerjee, graduated Summer 2003. Current position: Assistant Professor at University of Wisconsin.
- Laura Bright, graduated Spring 2003 (co-advisor). Current position: Research Associate, Oregon Graduate Institute.
- Vijay Gopalakrishnan, graduated Summer 2006. Current position: MTS, AT&T Research.
- Seungjoon Lee, graduated Summer 2006. Current position: MTS, AT&T Research.
- Ruggero Morselli, graduated Summer 2006. Current position: MTS, Google Inc.
- Christopher Kommareddy, graduated Summer 2006. Current position: Researcher, Amazon, Inc.
- Rob Sherwood, graduated Summer 2008. Current position: MTS, Deutsche Telekom Labs.
- Adam Bender, graduated Fall 2010, Current Position: MTS, Google.
- Dave Levin, graduated Summer 2010, Current Position: Visiting Research Professor, University of Maryland.
- Randolph Baden, graduated Summer 2012, Current Position: MTS, LTS-NSA.

3.2.4 Doctoral (current)

- Matthew Lentz
- Yeongsam Park
- Kookjin Lee
- James Litton
- Also advised visiting Ph.D. student Paolo Massa (Univ. of Trento) during Winter 2003-2004

3.3 Advising: Ph.D. Committees

- Nikhil Swami, Expected July 2008.
- Arun Vasani, 2008.
- Stephen Birrer (Northwestern University), 2007.
- Tuna Guven (ECE), 2006.
- Wan, Yung Chun, 2005.

- Andrzej Kochut, 2005.
- Yoo Ah Kim, 2005.
- Mehdi Kalantari (ECE), 2005.
- Arunesh Mishra, 2005.
- Surapich Phuvoravan (ECE), 2003.
- Bujor Silaghi, 2003.
- Suman Banerjee, 2003.
- Laura Bright, 2003.
- Kaushik Kar (ECE), 2002.
- Sungjoon Ahn, 2001.
- Ugur Cetintemel, 2001.
- Gabriel Rivera, 2001.
- Cuneyt Akinlar, 2001.
- Jung-Min Kim, 2001.
- Kritchalach Thitikamol, 2000.
- Saswati Sarkar (ECE), 2000.
- Demet Aksoy, 2000.

4 Service

4.1 Professional

4.1.1 Unpaid Reviewing Activities for Agencies

1. *NSF workshop on Network Testbeds*, attended workshop and co-authored report, 2002. Report basis for new NSF program on research testbeds.
2. NSF Networking Research Panel, Fall 2000, Fall 2001, Spring 2002, Fall 2002, Spring 2003, Spring 2004, Fall 2005, Spring 2008.
3. DoE High Performance Networking Panel, Spring 2001.
4. Evaluator for Intel Science Talent Search, 2001, 2002, 2003, 2004, 2005, 2007.

4.1.2 Other non-University Panels and Positions

- Program Committee Member, W-PIN+Netecon, 2013.
- Program Committee Member, WWW, 2013.
- Program Co-Chair, IEEE ICNP 2012.
- Program Committee Member, IMC, 2011.
- Co-Chair, Internet Research Task Force (IRTF) Peer-to-Peer Research Group, 2003 – 2009.
- Member, Internet Research Steering Group (IRSG), 2003 – current.
- Co Program Committee Chair, NetEcon 2009 Workshop.
- Program Committee Member, INFOCOM, 2009.
- Program Committee Member, Sigmetrics, 2009.
- Program Committee Member, Area TPC Chair, ICNP, 2008.
- Program Committee Member, INFOCOM, 2008.
- Program Committee Member, LANMAN, 2008.
- Program Committee Member, Sigmetrics, 2008.
- Program Committee Member, Workshop on Social Network Systems, 2008.
- Program Committee Member, ACM SIGCOMM, 2007.
- Program Committee Member, NetDB, 2007.
- Program Committee Member, Sigmetrics, 2007.
- Program Committee Member, ICDCS (P2P track), 2007.
- Program Committee Member, ICNP (P2P track), 2007.
- Program Committee Member, Electronic Commerce (EC), 2007.
- Area Chair (Dependable and Trustworthy Computing), ICPADS, 2007.
- Program Committee Member, IEEE Consumer Communications and Networking Conference - Workshop on Peer-to-Peer Multicasting, 2007.
- Program Committee Member, IEEE INFOCOM, 2006.
- Associate Chair (P2P, Grids Track), Fifteenth International Conference on Computer Communications and Networks (IC3N), 2006.

- Associate Chair (P2P Track), The 26th International Conference on Distributed Computing Systems (ICDCS), 2006.
- Program Committee Member, Global Internet (GI), 2006.
- Program Committee Member, IWAN, 2006.
- Program Committee Member, ICNP, 2006.
- Program Committee Member, IEEE INFOCOM, 2006.
- Program Committee Member, The 25th International Conference on Distributed Computing Systems (ICDCS), 2005.
- Program Committee Member, International Workshop on Active Networking (IWAN), 2005.
- Program Committee Member, IEEE INFOCOM 2005.
- Program Committee Member, IPTPS, 2005.
- Program Committee Member, HICSS, 2005.
- Program Committee Member, ACM SIGCOMM, 2004.
- Program Committee Member, 6th International Workshop on Distributed Computing (IWDC), 2004.
- Program Committee Member, 24th International Conference on Distributed Computing Systems (ICDCS), 2004.
- Program Committee Member, IEEE Global Internet Conference, 2004.
- Program Committee Member, International Workshop on Active Networking (IWAN), 2004.
- Program Committee Co-Chair, OpenArch, 2003.
- Program Committee Member, IEEE International Conference on Network Protocols, 2003.
- Program Committee Member, IEEE OpenSig, 2003.
- Program Committee Member, International Workshop on Networked Group Communications (NGC), 2003.
- Program Committee Member, International Workshop on Active Networking (IWAN), 2003.
- Program Committee Member, IEEE Global Internet Conference, 2003.
- Program Committee Member, IEEE International Conference on Network Protocols, 2002.

- Program Committee Member, International Workshop on Networked Group Communications (NGC), 2002.
- Program Committee Member, IEEE Global Internet Conference, 2002.
- Program Committee Member, International Workshop on Active Networking (IWAN), 2002.
- Program Committee Member, IEEE International Conference on Network Protocols, 2001.
- Program Committee Member, Workshop on Performance and Architecture of Web Servers (PAWS), 2001.
- Publications Chair, Member of Organizing and Program Committee, IEEE/ACM Open-Arch, 2001.
- Program Committee member, IEEE Global Internet Conference, 2001.

Bobby Bhattacharjee: List of Prior Engagements in Litigation Matters

- *Net2Phone Inc. vs. EBay Inc. et al.*, Case Number: 2:2006cv02469, New Jersey District Court. Engaged by Williams & Connolly, LLP. Washington DC, Grayson and Kubli, Vienna, VA, Kaye Scholer LLP., Washington DC. 2007
- *Juniper Networks Inc. vs. Abdullah Ali Bahattab*, Case Number: 1:2007cv01771, District of Columbia District Court. Engaged by Kaye Scholer LLP., Washington DC. 2008.
- *01 Communique vs. LogMeIn Inc.*, Case Number: 1-10-cv-01007-CMJ-TRJ, United States District Court, Eastern District of Virginia, Alexandria Division. Engaged by Wilmer Cutler Pickering Hale and Dorr LLP (WilmerHale), Boston. 2010.
- *Enterasys Networks Inc., vs. Foundry Networks, L.L.C. and Exterme Networks, Inc.*, Case Number: 05-11298-DPW (D. Mass). Engaged by Robins, Kaplan, Miller, and Cirese L.L.P., Boston. 2011.
- *Augme Technologies, Inc., vs. Yahoo! Inc.*, Case Number: C 09-5386 (N. D. Cal.). Engaged by Goodwin Procter L.L.P., Menlo Park, California. 2011.
- *Augme Technologies, Inc., vs. Tacoda, Inc. et al*, Case Number: 1:07-cv-0788-CM-GWG (S. D. N.Y.). Engaged by Goodwin Procter L.L.P., Washington DC. 2011.
- *Zions Bancorporation, Zions First National Bank, Amegy Bank, N.A. and Zions Management Service Co., vs. RSA, Inc.* Case Number: 74 117 00683 11 NOLG, American Arbitration Association. Engaged by Akin Gump Strauss Hauer & Feld L.L.P., Washington D.C. 2012.
- *SmartPhone Technologies L.L.C. vs. HTC Corp.*, Case Number 6:10-cv-580 (E.D. Tex.) Engaged by McDermott Will & Emery, Menlo Park, California. 2012.
- *Tranxition, Inc. vs. Appsense Inc.*, Case Number 3:12-cv-01403-BR (D. OR.) Engaged by Cooley L.L.P., Reston, VA. 2013.
- *Nokia Corp. vs. HTC Corp.*, Case Number 1:12-cv-00550-UNA (D. Del.) and 1:12-cv-551-UNA (D. Del). Engaged by McDermott Will & Emery, Menlo Park, California. 2013.
- *Intellectual Ventures II LLC vs. SunTrust Bank.*, Case Number 13-cv-02454 U.S. District Court for the Northern District of Georgia. Engaged by Kecker & Van Nest LLP, California. 2013.

- *Parallel Networks, LLC v. F5 Networks, Inc. C.A. No. 13-2001-LPS.* Engaged by Wilson Sonsini Goodrich and Rosati, Los Angeles, CA. 2014.
- *Cisco Systems, Inc. v. Arista Networks, Inc. (N.D. Cal. and ITC).* Engaged by Kirkland and Ellis LLP, Los Angeles, California. 2014.
- *Cisco Systems, Inc. v. Arista Networks, Inc. (N.D. Cal. and ITC).* Engaged by Quinn Emanuel Urquhart & Sullivan, LLP, San Francisco, California. 2015.
- *CAP Co., Ltd. v. Symantec Corporation U.S.D.C., N.D. Cal., Case No. 3:14-cv-05071-JD.* Engaged by Wilson Sonsini Goodrich & Rosati, San Francisco, California, 2015.
- *Netfuel, Inc. v. F5 Networks, Inc., Case No. 1:13-cv-07895 (N.D. Ill.).* Engaged by Perkins Coie, Seattle, Washington, 2015.
- *Limelight Networks Inc. vs. XO Communications, LLC. et al (3215-cv-720).* Engaged by Chaote Hall & Stewart LLP, Boston, 2016.
- *Arista Networks, Inc. vs. Cisco Systems, Inc., Case No. 5:16-cv-00923-BLF (SVK).* Engaged by Desmarais LLP, New York, 2017.
- *Limelight Networks, Inc. v. Akamai Technologies, Inc., Case No. 3:15-cv-720 (E.D. Va.); Akamai Technologies, Inc. v. Limelight Networks, Inc., Case Nos. 1:16-cv-10253 and 1:16-cv-12624 (D. Mass.); Uniloc USA, Inc., et al. v. Akamai Technologies, Inc., Case Nos. 3:17-cv-2116 and 3:17-cv-2118 (N.D. Tex.)* Engaged by Wilmer Hale and Dorr LLP (WilmerHale), Boston. 2017.
- *Koninklijke Philips N.V. v. Microsoft Corporation* Engaged by Perkins-Coie, LLP. Seattle. 2017.
- *Blackberry Ltd. v. Snap Inc., CACD-2-18-cv-02693.* Engaged by Paul Hastings LLP, New York. New York. 2018.
- *The Ohio State Innovation Foundation v. Akamai Technologies, Inc.* Engaged by Farella Braun + Martel, LLP, San Francisco, CA. 2019
- *Realtime Adaptive Streaming LLC v. Adobe Systems Inc., Case No. 2:18-cv-09344-GW-JC (Central District of California).* Engaged by Fish and Richardson, P.C. Redwood City, CA. 2019
- *Network-1 Technologies, Inc. v. Google , Nos. 1:14-cv-2396 & 1:14-cv-9558 (S.D.N.Y.).* Engaged by Williams & Connolly, LLP. Washington DC. 2019
- *Uniloc 2017 LLC v. Google LLC, No. 2:18-cv-00502-JRG.* Engaged by Cooley, LLP. Washington DC. 2020.

- *Voxer, Inc. and Voxer IP LLC v. Facebook, Inc. and Instagram LLC*, No. 6:20-cv-00011. Engaged by Kecker & Van Nest LLP, California. 2020.
- *Nokia Techs. Oy v. Lenovo (Shanghai) Elects. Tech. Co. Ltd. et al.*, No. 5:19-cv-427 (E.D.N.C.) Engaged by Wilmer Cutler Pickering Hale and Dorr LLP, Washington D.C. 2020.
- *Verizon Business Network Services Inc. v. Huawei Technologies Co., Ltd.*, IPR2020-01079 (PTAB) and *Verizon Business Network Services Inc. v. Huawei Technologies Co., Ltd.*, IPR2020-01080 (PTAB). Engaged by Merchant and Gould P.C., Atlanta, GA. 2020.