UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

FACEBOOK, INC. AND INSTAGRAM, LLC
Petitioners

v.

VOXER IP LLC,
Patent Owner

PATENT NO. 10,511,557

DECLARATION OF DR. KEVIN JEFFAY

REGARDING U.S. PATENT NO. 10,511,557

Case No. IPR2021-00112

Aylo Exhibit 1023
Aylo Freesites v. DISH Technologies
IPR2024-00043

TABLE OF CONTENTS

Page

I. BACKGROUND AND QUALIFICATIONS 1

II. MATERIALS CONSIDERED 6

III. LEGAL STANDARDS FOR PATENTABILITY 9

IV. TECHNICAL BACKGROUND 15

A. Video, Digital Video, and Video Coding, Compression, and Transcoding 16

B. Video Adaptation and Real-Time Transmission 19

C. Video Streaming Protocols 25

D. Enabling Simultaneous Rendering of Content from Multiple Sources 29

V. THE ’557 PATENT 32

A. Overview of the ’557 Patent 32

B. Overview of Challenged Claims 35

C. Effective Filing Date of the ’557 Patent 39

D. Level of Ordinary Skill in the Art 41

VI. CONSTRUCTION OF THE TERMS USED IN THE ’557 PATENT 43

VII. OVERVIEW OF PRIOR ART REFERENCES 44

A. Brueck 44

B. Brooks 50

C. Mullig 52

D. Kalaboukis 54

E. Mangal 57

VIII. GROUND 1: CLAIMS 1-3, 8, 9, 14, AND 18 ARE RENDERED OBVIOUS BY BRUECK 58

A. Every Element of the Challenged Claims Is Disclosed or Rendered Obvious by Brueck 58

1. Claim 1 58

a. "A method for operating a video message service infrastructure on a network, the method comprising:" (Preamble) 58

b. "receiving, at the video message service infrastructure, a video message transmitted by a sending device, the video message containing video media, the video media being transmitted as the video media is created;" (1[a]) 60

c. "generating, by the video message service infrastructure, two or more degraded versions of the video media of the video message, where each of the two or more degraded versions is created by degrading the quality of the video media by a different amount; (1[b]) 63

d. "selecting one of the degraded versions of the video media based on an ascertained bandwidth on the network for delivering the video media from the video message service infrastructure to a recipient device on the network; and" (1[c]) 67

e. "transmitting the selected degraded version of the video media from the video message service infrastructure to the recipient device so that the recipient device can render at least a portion of the video media on the recipient device;" (1[d]) 72

f. "the video message service infrastructure operating so that the rendering occurs: (i) while the video media of the video message is being created and transmitted over the network by the sending device; and" (1[e]) 74

g. "(ii) without having to first establish an end-to-end connection over the network between the sending device

and the recipient device before the sending device begins to transmit the video media of the video message." (1[f]) 75

2. Claim 2: "The method of claim 1, wherein at least one of the two or more degraded versions of the video media of the video message is generated by reducing a bit rate used to represent the video media relative to a complete bit rate representation of the video media as received by the video message service infrastructure." 77

3. Claim 3: "The method of claim 1, further comprising transmitting one or more of the two or more degraded versions to two or more additional recipient devices, wherein the transmitted one or more degraded versions are selected at least partially based on being commensurate with ascertained bandwidths on the network along delivery paths for delivering the video message to the two or more additional recipient devices respectively."....79

4. Claim 8: "The method of claim 1, wherein the video message is received at the video message service infrastructure using a first real-time streaming protocol, and the selected degraded version of the video media is transmitted from the video message service infrastructure to the recipient device using a second real-time streaming protocol."....82

5. Claim 9: "The method of claim 8, wherein the first real-time streaming protocol is different from the second real-time streaming protocol."....84

6. Claim 14:....84

a. "ascertaining a first codec that was used by the sending device to encode the video media of the video message when sent;" (14[a])....84

b. "ascertaining a second codec to be used by the recipient device to decode the video media of the video message when received;" (14[b]) 86

c. "trans-coding the video media of the video message from the first codec to the second codec; and" (14[c]).....87

d. "transmitting, to the recipient device, the trans-coded video media." (14[d]).....87

7. Claim 18: "The method of claim 1, further comprising providing an application for execution on the recipient device, the application enabling a recipient using the recipient device to selectively render the video media of the video message: at least partially as the video media of the video message is created and transmitted by the sending device and delivered by the video message service infrastructure; and in a time-shifted mode by rendering the video media of the video message out of storage." 88

IX. GROUND 2: CLAIMS 8, 9, AND 14 ARE RENDERED OBVIOUS BY BRUECK IN COMBINATION WITH BROOKS.....93

A. A Person of Ordinary Skill in the Art Would Have Been Motivated to Combine Brueck with the Teachings of Brooks 93

B. Every Element of Claims 8, 9, and 14 Is Disclosed by Brueck Combined with Brooks.....94

1. Claim 8: "The method of claim 1, wherein the video message is received at the video message service infrastructure using a first real-time streaming protocol, and the selected degraded version of the video media is transmitted from the video message service infrastructure to the recipient device using a second real-time streaming protocol."....94

2. Claim 9 96

3. Claim 14: "The method of claim 1, further comprising …" 96

a. "ascertaining a first codec that was used by the sending device to encode the video media of the video message when sent" (14[a]).....97

b. "ascertaining a second codec to be used by the recipient device to decode the video media of the video message when received" (14[b]) 98

c. "trans-coding the video media of the video message from the first codec to the second codec" (14[c]) 99

d. "transmitting, to the recipient device, the trans-coded video media" (14[d]).....100

X. GROUND 3: CLAIM 16 IS RENDERED OBVIOUS BY BRUECK IN COMBINATION WITH MULLIG 101

A. A Person of Ordinary Skill in the Art Would Have Been Motivated to Combine Brueck with the Teachings of Mullig 104

B. Every Element of Claim 16 Is Disclosed by Brueck Combined with Mullig. 105

1. Claim 16: "The method of claim 1, further comprising:" 105

a. "receiving, at the video message service infrastructure, a second video message transmitted by a second sending device, the second video message including second video media, wherein the second video media being transmitted by the second sending device as the second video media is created;" (16[a]) 105

b. "transmitting the second video media from the video message service infrastructure over the network to the recipient device so that the recipient device can render at least a portion of the second video media;" (16[b]) 106

c. "the video message service infrastructure operating so that the rendering occurs: (i) as the second video media of the second video message is created and transmitted over the network by the second sending device;" (16[c]) 108

d. "(ii) without having to first establish an end-to-end connection over the network between the second sending device and the recipient device before the second sending device begins to transmit the second video media of the

second video message; and" (16[d]) 109

e. "(iii) simultaneously with the rendering of the video media of the video message." (16[e]) 110

XI. GROUND 4: CLAIM 23 IS RENDERED OBVIOUS BY BRUECK IN COMBINATION WITH KALABOUKIS AND MANGAL.....111

A. A Person of Ordinary Skill in the Art Would Have Been Motivated to Combine Brueck with the Teachings of Kalaboukis and Mangal.....113

B. Every Element of Claim 23 Is Disclosed by Brueck Combined with Kalaboukis further in view of Mangal. 116

1. Claim 23: "The method of claim 1, further comprising:" 116

a. "configuring the video message service infrastructure to include a plurality of servers located on the network, the plurality of servers collectively configured for performing:" (23[a]).....116

b. "receiving the video message;" (23[b]) 117

c. "ascertaining locations on the network for multiple recipient devices including the recipient device, the locations derived from multiple identifiers provided by the sending device;" (23[c]).....118

d. "transmitting the video message to the multiple recipient devices so that each of the multiple recipient devices can render at least a portion of the video message as the sending device is sending the video message;" (23[d]).....121

e. "storing the video message as the video media is received by the video message service infrastructure; and" (23[e]) 123

f. "providing the video media of the video message to any of the multiple recipient devices for rendering in a time-shifted mode." (23[f]) 124

XII. CONCLUSION.....126

DECLARATION OF DR. KEVIN JEFFAY

I, Dr. Kevin Jeffay, declare as follows:

1. I make this declaration based upon my own personal knowledge and, if called upon to testify, would testify competently to the matters stated herein.

2. I have been asked by Facebook, Inc. to provide my expert opinion in connection with the *inter partes* review of U.S. Patent No. 10,511,557 (which I will refer to as the "'557 patent") concerning whether the '557 patent is valid over certain prior art. This declaration is a statement of my opinions on issues relating to the patentability of claims 1-3, 8, 9, 14, 16, 18, and 23 (which I will refer to as the "challenged claims") of the '557 patent.

3. My compensation is not based on the content of my opinions or the resolution of this matter.

I. BACKGROUND AND QUALIFICATIONS

4. In formulating my opinions, I have relied on my knowledge, training, and experience in the relevant field, which I will summarize briefly.

5. I am a tenured professor in the Department of Computer Science at the University of North Carolina at Chapel Hill where I currently hold the position of Gillian T. Cell Distinguished Professor of Computer Science. I also currently serve as the Chairman of the Department. I have been a faculty member at UNC since 1989.

6. I received a Ph.D. in computer science from the University of Washington in 1989. Before that I received a M.Sc. degree in computer science from the University of Toronto in 1984, and a B.S. degree with Highest Distinction in mathematics from the University of Illinois at Urbana-Champaign in 1982.

7. I have been involved in the research and development of computing systems for nearly 40 years. I have been a faculty member at the University of North Carolina since 1989 where I perform research and I teach in the areas of multimedia networking, computer networks, distributed systems, real-time systems, and operating systems, among others. A major theme of my research has been the development of technology to improve the performance of data transfers on the Internet. My research has examined problems ranging from network and operating system support for real-time multimedia applications such as audio and video streaming, voice-over-Internet protocol (VoIP) and Internet videoconferencing, to the design of congestion control mechanisms in network routers, to measurements and analysis of network traffic to passively assess the performance of servers on the Internet.

8. For example, starting in the late 1980s, the focus of my research was the development of network and operating system technology to enable the real-time transfer of streams of audio and video data across the Internet. This involved, among other things, the development of media encoders, media players, and

network communication protocols for adaptive transmission of live audio and video data on the Internet. This work culminated in my research group developing some of the first videoconferencing systems for the Internet. Several of the papers authored by myself and members of my research group on this project won awards for their technical contributions.

9. The videoconferencing and adaptive streaming research attracted the attention of industry groups such as IBM, Intel, Digital Equipment Corporation, Cabletron, and AT&T Bell Laboratories. For example, starting in 1991 IBM supported aspects of my research at UNC and these efforts resulted in U.S. Patent No. 5,892,754 on adaptive media streaming being issued to IBM and UNC.

10. During this time, I was also collaborating with researchers at the Intel Architecture Labs in Hillsboro, OR, to modify an Intel product, the ProShare videoconferencing system, to use technology developed in my research lab at UNC for adaptive media transmission. By the late 1990s, the ProShare product included the ability to also transfer desktop information of a PC to a remote PC.

11. My research group also developed other "data conferencing" systems, also known as "shared window systems," that were capable of transferring the desktop information on a PC to a remote PC in real-time. One system that we built, called "XTV," was operational by 1991. The source code for XTV was made freely available and by 1993 had been downloaded by over 600 users and

institutions. The system was functionally and visually equivalent to today's Cisco's WebEx and Zoom's screen sharing products and services. It provided a sophisticated means of transferring the desktop information of a PC to a remote site. In the XTV system, individual windows displayed on the desktop, or the entire desktop itself, could be selected for sharing with remote users. The XTV system also allowed remote users to remotely control and manipulate the desktop whose information was being distributed, as well as remotely control the applications that were generating the desktop information that was being shared.

12. In much of my research I regularly build and use clusters of computers interconnected by network switches, bridges, and routers to form and evaluate experimental and production networks. For example, in the late 1990s and early 2000s my research evolved to consider router-based mechanisms for controlling the performance of network traffic. In much of this research my students and I built and instrumented network routers and performed large scale experiments with this equipment. The instrumentation included, for example, the development of network monitors that would receive copies of packets flowing on a network link and analyze and store packets or the results of analyses on these packets, all in real-time. Based on these experiments, in 2003, my group at UNC won the most prestigious research award for original research in computer networking. UNC applied for, and received, U.S. Patent No. 7,447,209 for aspects

of this research.

13. This project, and others, took place in a networking lab my students and I constructed at UNC over a number of years. The lab consists of several hundred computers and networking devices. Managing this lab required establishing and configuring VLANs, firewalls and other security appliances to isolate the lab from the campus network (and vice versa).

14. More recently, my research group has considered the design and operation of next generation aerial networks. UNC applied for, and received, U.S. Patent No. 9,832,705 for aspects of this research.

15. I have authored or co-authored over 100 articles in peer-reviewed journals, conference proceedings, texts, and monographs in the aforementioned areas of computer science and others. I have served as Editor-in-Chief for the journal *Multimedia Systems* and Associate Editor for the journal *Real-Time Systems*. In addition, I have edited and co-edited numerous published proceedings of technical conferences and have edited a book of readings in multimedia computing and networking (with Hong-Jiang Zhang) published by Morgan Kaufman. I am a co-author (with Long Le and F. Donelson Smith) of a monograph related to computer network protocols, and a co-author (with Jay Aikat and F. Donelson Smith) of a second monograph related to experimental computer networking.

16. I have served on numerous proposal review panels for the National Science Foundation and other international funding agencies in the aforementioned areas of computer science. I have served as a program chair or member of the technical program committee for over 100 professional, international, and technical conferences, workshops, and symposia.

17. In addition to the U.S. patents reference above, I am a named inventor on a fourth patent. Each of these patents is generally related to computer networking and the delivery of services over networks including audio and video transmission.

18. I have served as an expert witness and technical consultant in litigation and *inter partes* review matters concerning videoconferencing, data conferencing and screen sharing, cellular and wireline telephony, voice over IP (VoIP) telephony, multimedia networking, distributed systems, operating systems, computer networks, datacenter networking, embedded systems and embedded software, and real-time systems, among others. I have testified in several trials, arbitrations, and claim construction hearings as an expert witness

19. A copy of my curriculum vitae, which summarizes my relevant education and experience, is attached as Exhibit A.

II. MATERIALS CONSIDERED

20. I have reviewed and considered the following documents in

connection with my analysis of the ’557 patent:

a. **the ’557 patent (Ex-1001);**

b. **U.S. Provisional Patent Application No. 60/937,552, filed Jun. 28, 2007 (Ex-1004);**

c. **U.S. Patent No. 7,818,444 (“Brueck”) (Ex-1005);**

d. **U.S. Patent No. 7,339,993 (“Brooks”) (Ex-1006);**

e. **U.S. Patent No. 7,930,419 (“Mullig”) (Ex-1007);**

f. **U.S. Patent No. 9,762,861 (“Kalaboukis”) (Ex-1008);**

g. **U.S. Patent No. 6,865,398 (“Mangal”) (Ex-1009);**

h. **Delgrossi, et al., “Media Scaling for Audiovisual Communication with Heidelberg Transport System,” IBM European Networking Center (“Delgrossi”) (Ex-1010);**

i. **Franc Kozamernik, “Media Streaming Over the Internet – an Overview of Delivery Technologies,” EBU Technical Review, October 2002 (“Kozamernik”) (Ex-1011);**

j. **David Austerberry, “The Technology of Video and Audio Streaming” (2d Ed.), Focal Press, 2005 (“Austerberry”) (Ex-1012);**

k. **Rowe, et al., “BIBS: A Lecture Webcasting System,” UC Berkeley Research and Occasional Papers Series, Mar. 20, 2001 (“BIBS”) (Ex-1013);**

l. **U.S. Patent Appl. Pub. No. 2002/0131496 (“Vasudevan”) (Ex-1014);**

m. **Request for Comments: 1889, “RTP: A Transport Protocol for Real-Time Applications,” Network Working Group, January 1996 (“RFC 1889”) (Ex-1015);**

n. **Kevin Jeffay, Lecture Slides, COMP 249, Advanced Distributed Systems, Department of Computer Science,**

University of North Carolina at Chapel Hill, October 7, 1997 (Ex-1016);

o. **Kevin Jeffay, Lecture Slides with Notes, COMP 249, Advanced Distributed Systems, Department of Computer Science, University of North Carolina at Chapel Hill, October 7, 1997 (Ex-1017);**

p. **James F. Kurose & Keith W. Ross, "Computer Networking, A Top-Down Approach Featuring the Internet," 2d Ed. 2003 ("Kurose, Ross") (Ex-1018);**

q. **RealNetworks, Helix Server Administration Guide, Helix Server Version 11, September 2, 2015 Rev. (Ex-1019);**

r. **Request for Comments: 2326, "Real-Time Streaming Protocol (RTSP)," Network Working Group, April 1998 (Ex-1020);**

s. **U.S. Patent No. 6,389,473 ("Carmel") (Ex-1021);**

t. **ITU-T, Telecommunication Standardization Sector, "H.323: Packet-based multimedia communications systems," February 1998 ("H.323") (Ex-1022);**

u. **U.S. Patent Appl. Pub. No. 2004/0119814 ("Clisham") (Ex-1023);**

v. **U.S. Patent No. 7,692,683 ("Kenoyer") (Ex-1024);**

w. **Sachin G. Deshpande, "A Real-Time Interactive Virtual Classroom Multimedia Distance Learning System," IEEE Transactions on Multimedia, Vol 3, No. 4, Dec. 2001 ("Deshpande") (Ex-1025);**

x. **Shi, et al., "Project Smart Remote Classroom – Providing Novel Real-Time Interactive Distance Learning Technologies," International Journal of Distance Education Technologies, 1(3), 28-45, July-Sept.**

2003 ("SameView") (Ex-1026);

y. **U.S. Patent No. 6,675, 386 ("Hendricks") (Ex-1027);**

z. **U.S. Patent Appl. Pub. No. 2006/0080407 ("Rengaraju") (Ex-1028).**

21. In addition, my opinions are based on my years of experience in the field of networking and multimedia systems.

III. LEGAL STANDARDS FOR PATENTABILITY

22. My opinions are also based on my understanding of the relevant law. I am not an attorney. For purposes of this declaration, I have been informed about certain aspects of the law as it relates to my opinions.

23. I understand that a patent contains one or more claims that may be written in independent or dependent form. I further understand that an independent claim is a standalone claim that contains all the limitations necessary to define an invention, whereas a dependent claim refers to a claim previously set forth and further limits that claim. I also understand that a claim in dependent form incorporates all the limitations of the claim that it depends from.

24. I understand that for an invention claimed in a patent to be patentable, it must be new and not obvious based on what was known before the invention was made.

25. I understand that the information that is used to evaluate whether an invention was new and not obvious when made is generally referred to as "prior

art." I understand that the prior art includes patents and printed publications that existed before the earliest filing date of the patent (*i.e.*, the "effective filing date"). This includes foreign language material.

26. I understand that in this *inter partes* review proceeding, Petitioner has the burden of proving that the challenged claims of the '557 patent are unpatentable in light of the prior art by a preponderance of the evidence. I understand that a preponderance of the evidence is sufficient to show that a fact is more likely true than not true.

27. I understand that in this *inter partes* review proceeding, the information that may be used to demonstrate unpatentability is limited to patents and printed publications. My analysis compares the challenged claims of the '557 patent to patents and printed publications that I understand are prior art to the challenged claims.

28. I understand that prior art can render a claim "obvious" to a person of ordinary skill in the art and thus unpatentable. My understanding of this legal standard is set out below.

29. I understand that prior art can render a patent claim unpatentable where the differences between the subject matter that falls within the scope of the claim and the prior art are such that the subject matter as a whole would have been obvious to a person of ordinary skill in the art. I understand that the following

standards govern the determination of whether a patent claim is rendered "obvious" in light of the prior art. I have applied these standards in my evaluation of whether the challenged claims of the '557 patent are obvious in light of the prior art.

30. The person of ordinary skill in the art is a hypothetical person who is presumed to have known the relevant art at the time of the invention.

31. Prior art disclosing a method or device that falls within the scope of a claim can render that claim obvious even if other, different methods or devices might also fall within the scope of the claim.

32. I understand that a determination as to whether a claim would have been obvious should be based on four factors (though not necessarily in the following order): (i) the level of ordinary skill in the art at the time the application was filed; (ii) the scope and content of the prior art; (iii) the differences between the claimed invention and the prior art; and (iv) any "objective factors" indicating obviousness or non-obviousness that may exist in a particular case.

33. I understand that an obviousness analysis should not be based on hindsight, but must be done using the perspective of a person of ordinary skill in the relevant art as of the effective filing date of the patent claim.

34. I understand that the objective factors indicating obviousness or nonobviousness may include: commercial success of products covered by the

patent claims; a long-felt but unaddressed need for the invention; failed attempts by others to make the invention; copying of the invention by others in the field; expressions of surprise by experts and those skilled in the art at the making of the invention; and the patentee having proceeded contrary to the accepted wisdom of the prior art. I also understand that any of this evidence must be specifically connected to the claimed invention rather than associated with the prior art or with marketing or other efforts to promote an invention.

35. I understand that the teachings of two or more prior art references may be combined in the manner disclosed in the claim if such a combination would have been obvious to one having ordinary skill in the art. In determining whether a combination would have been obvious, the following exemplary rationales may support a conclusion of obviousness:

- combining prior art elements according to known methods to yield predictable results;
- simple substitution of one known element for another to obtain predictable results;
- use of a known technique to improve similar devices (methods, or products) in the same way;
- applying a known technique to a known device (method, or product) ready for improvement to yield predictable results;

- "obvious to try" – choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success;
- 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;
- 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; and
- common sense.

36. I understand that the obviousness analysis need not seek out precise teachings directed to the specific subject matter of the challenged claim, but instead can take account of the ordinary innovation and experimentation in the relevant field that does no more than yield predictable results.

37. I understand that, in assessing whether there was an apparent reason to modify or combine known elements as claimed, it may be necessary to look to interrelated teachings of multiple patents and printed publications, the effects of commercial demands, and the background knowledge of a person of ordinary skill in the art. I further understand that any motivation that would have applied to a person of ordinary skill in the art, including motivation from common sense or

derived from the problem to be solved, is sufficient to explain why references would have been combined.

38. I understand that modifications and combinations suggested by common sense are important and should be considered. Common sense suggests that familiar items can have obvious uses beyond the particular application being described in a prior art reference, that if something can be done once it would be obvious to do it multiple times, and that in many cases a person of ordinary skill in the art can fit the teachings of multiple patents together in an obvious manner to address a particular problem. Further, the prior art does not need to be directed to solving the same problem that is addressed in the patent.

39. I understand that a person of ordinary skill in the art is also a person of ordinary creativity. In many fields, it may be that there is little discussion of obvious techniques, modifications, and combinations, and it may be the case that market demand, rather than scientific research or literature, will drive a new design. When there is market pressure or design need to solve a particular problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has a good reason to employ the known options. If this leads to the expected success, then it is likely the product of ordinary skill and common sense as opposed to patentable innovation. I understand that if a combination was obvious to try, that may show that it was obvious and therefore unpatentable. That

a particular combination of prior art elements was obvious to try suggests that the combination was obvious even if no one made the combination.

IV. TECHNICAL BACKGROUND

40. The claims of the ’557 patent are directed to methods for operating a video message service infrastructure on a network. The claimed methods recite steps including receiving one or more video messages at the video message infrastructure from a sending device and transmitting video messages from the video message infrastructure to a receiving device as the sending device is sending its video message. Certain methods include additional steps such as ascertaining the bandwidth available in the network between the video message service infrastructure and the receiving device, generating degraded versions of the video message, and selecting a degraded version to match the bandwidth available in the network. Other claimed concepts including using two real-time streaming protocols and transcoding the video message. As of June 2007, which, as discussed below, I understand is the earliest possible priority date for the ’557 patent, video messaging systems as claimed in the ’557 patent were well known in the art. Likewise, infrastructures and protocols for supporting such services were quite mature.

41. To better understand the ’557 patent, the prior art, and my opinions on the validity of the claims of the ’557 patent, in this section I provide some

background on video as a data type and infrastructures that existed to support video messaging. All of the concepts discussed in this section were well known and widely used well prior to June 2007. Indeed, as detailed below, I had frequently taught many of the concepts discussed herein to computer science students at UNC for nearly fifteen years prior to 2007.

A. Video, Digital Video, and Video Coding, Compression, and Transcoding

42. Modern digital video is a multi-dimensional data type. A video image—essentially a still picture—is a two-dimensional structure consisting of an acquired or displayed width and height measured in picture elements ("pixels"). Each picture element in turn is represented by three values, one value for each of the primary colors of projected light: red, green, and blue. In the case of a displayed video image, the three values represent the intensity of red, green, and blue light respectively, at a given picture element. Video is a sequence of video images wherein each image is acquired/displayed for a (typically) fixed duration and then the next image in the sequence is acquired/displayed. Thus, in addition to spatial dimensions, video also has a temporal dimension, typically measured in units of the video image display duration or the number of video images displayed per unit time (a "frame rate").

43. As a multi-dimensional data type, there are a large number of parameters that can be varied when generating or acquiring a video stream.

Varying these parameters affects the size of the video stream and thereby affects the amount of storage required to archive the video or the bandwidth required to transmit the video in real-time (*i.e.*, as the video is generated). These parameters include obvious ones such as video image width, height, color depth (the number of bits used to represent the intensity of red/green/blue light present at each picture element), and frame rate (the inverse of the video image display duration). In addition, virtually all video is compressed to some extent and the use of compression provides numerous other parameters for adjusting the storage or transmission requirements for video. As a data type, there is gross redundancy in video data and video compression schemes seek to reduce the storage or transmission requirements for video by eliminating redundant information in ways that, to the extent possible, have minimal effect on the quality of the resulting compressed video. Each compression scheme typically has a number of parameters that can be used to control the size of the compressed video and the bandwidth required to transmit the compressed video. These parameters are more complex but primarily control how "aggressively" one attempts to remove redundancy from the video data. Generally, compression parameters are tuned to remove or reduce spatial redundancy (redundancy that exists within a single video image) and temporal redundancy (redundancy that exists between video images). Compression, however, comes at the cost of video quality. The higher the

Facebook Exhibit 1002
Page 24

compression rate, the smaller the size of the resulting video and the lower the perceived quality of the video. Like the video data type itself, the notion of video quality is also a multi-dimensional concept but generally is proportional to the resolution ("size") of the video, the color depth, the degree of motion captured (the frame rate), and the degree to which compression artifacts are discernable (which itself is inversely proportional to the size or bit-rate of the compressed video). Simply put, all other things being equal, the quality of a video stream varies proportionally with the bit-rate of the video.

44. The components in a video system that compress and decompress video (or audio) data are called "codecs" ("coders/decoders").[1] As of June 2007, numerous open source and commercial codecs existed that implemented a variety of standardized and proprietary video (and audio) compression schemes. Codecs for internationally standardized video compression schemes such as Motion JPEG, MPEG (specifically, MPEG-1 (for CD-ROM quality video), MPEG-2 (for DVD quality video), and MPEG-4 (for mixed media data encoding and streaming)), H.261, and H.264 were widely available by 2007.

---

[1] Technically, a codec "codes" (encodes or represents) a data source and coding and compression are separate concerns. However, as the term is used in the '557 patent it suffices to focus on the data compression aspects of codecs.

45. Related to codecs is the concept of transcoding, or converting between one video coding/compression scheme and another. A transcoder is simply a multi-function codec that can decode data received in one format and re-encode the data in another format, possibly compressing (or re-compressing) the data in the process. As explained below, transcoding was a commonly available feature of video message infrastructures by 2007.

B. Video Adaptation and Real-Time Transmission

46. For interactive multimedia applications such as voice/video telephony or video conferencing, video data must be acquired and transmitted as the media is generated (*i.e.*, in real-time). The same is true for non-interactive audio/video applications such as broadcasts or general communication of live events so that users can view these events as they are happening. However, today's public data networks, such as the Internet, support only "best effort" data delivery services and do not guarantee real-time data transmission. This means that when attempting to transmit data in real-time, it is quite common for the end-to-end data source to receiver transmission times of media units to vary over time. This causes problems for audio and video applications that must transmit and display data continuously. In such applications, the variations in transmission time can result in a situation wherein a receiver has displayed a video or audio image/sample and is ready to display the next image/sample but that data unit has not yet arrived. This results in

discontinuities in the media playout that can manifest themselves as temporarily frozen or distorted ("pixelated" or "blocky") video or "pops" and "clicks" being heard in the audio.[2]

47. A partial solution is to simply delay the start of the media playout when a communication session begins and build up a buffer of media data units at the receiver. By buffering a number of media data units the receiver can ensure continuous playout of media so long as the variation in data unit transmission times does not exceed the playout duration of the media data units stored in the buffer (*e.g.*, the size of the buffer in media data units multiplied by the data unit playout time). While such buffering can be effective, it increases the end-to-end delay in media acquisition to playout time. In the case of interactive applications, this increase in end-to-end delay can significantly impede user interaction. For this reason, it is highly desirable to always use as small a receiver buffer as possible.

48. However, media buffering will only be effective if real-time data transmission is possible. That is, in order to transmit video in real-time, sufficient bandwidth must be available in the network delivery path between a sender and receiver such that on average, a sender can continuously transmit, and the receiver can continuously receive, video data at a bit-rate equal to the average size of a video image times the frame rate. Said more simply, real-time video transmission

[2] *See* https://www.youtube.com/watch?v=cDxtbSzyRWM for examples.

is only possible if the receiver can receive data at the same rate at which the sender is sending the data. If the receiver is not receiving data at the rate at which the sender is sending the data, data is being "lost" in the network and continuous playout of video at the receiver will not be possible. When real-time video transmission is not possible, it is because all or portions of the network delivery path between the sender and receiver are being used by too many other users at the present time and network links on the path are saturated. Because neither a sender nor a receiver can control the number of other users in the network, in order to sustain real-time communication, a sender and receiver must coordinate to determine a data transmission rate that is sustainable at the present time.

49. Specifically, when conditions in a network are such that real-time transmission of video is not possible, the sender of the video must reduce the bit-rate of the video data it is transmitting. This requires that the sender (1) detect that the data it is transmitting is not being received in real-time (typically by receiving feedback from the receiver), (2) attempt to determine an amount by which to reduce the video bit-rate, and (3) reduce the video bit-rate it is transmitting to be at most the determined amount (by manipulating parameters of either the video data stream, the video codec, or both). In addition, the sender will typically attempt to further monitor network conditions to determine when, if ever, it can increase the rate at which it is transmitting video. This is because as noted, the quality of video

received by a user varies proportionally with the bit-rate of the video. Thus, adaptively increasing the video bit-rate, when possible, increases the quality of the video playout.

50. The processes and principles of transmitting audio and video data in real-time over best-effort networks were well established long before the ’557 patent. Indeed, just as one example, my research group at UNC was publishing on these topics and demonstrating systems for adaptive real-time transmission of audio and video data nearly 15 years before the priority date of the ’557 patent. For example, by 1992 we had constructed a video messaging system that detected whether or not data was being received in real-time by estimating the bandwidth available along the delivery path to a recipient. When the current data rate being transmitted was not sustainable (meaning data was not being received at the rate sent), the system determined an amount by which to reduce the video bit-rate and adaptively modified the current video generation and codec parameters to reduce the video bit-rate. A 1992 video demonstration of the system was published as part of a conference proceedings and is available on YouTube at the URL https://www.youtube.com/watch?v=cDxtbSzyRWM. And my research group was only one of several to work on these problems. For example, a 1992 paper by Delgrossi *et al.* provides a comprehensive framework for “scaling” (adapting) real-time video streams. *See* Ex-1010.

51. Another means of adapting video streams to account for variable bandwidth conditions involved encoding a video stream at various bitrates and selecting an appropriate stream for delivery to the end user based on current bandwidth availability on the network to the receiving device. This was the idea behind RealNetworks' *SureStream* technology, and Microsoft's *Intelligent Streaming* technology—both available commercially years before the alleged invention of the '557 patent. *See* Ex-1011 (Kozamernic) at 12-13; Ex-1012 (Austerberry) at 170-71; Ex-1013 (BIBS) at 8 (2001 publication describing use of *SureStream* technology in UC Berkeley lecture webcasting system; "The Real Networks streaming media system has a multiple-rate streaming feature, called *SureStream*, which allows several versions of the material to be encoded at different bit rates, stored, and delivered to the user. Users connect to the server asking for the highest bit rate their Internet connection can support. The player and server attempt to play the lecture at that rate. The client player will switch rates during playback if the connection cannot handle the requested bit rate.")

52. Other means of adapting video streams to ensure real-time delivery to recipients developed in this time frame included the use of video transcoding. In the case that video media was being sent to multiple recipients simultaneously, systems and protocols were developed wherein video data would be received by an intermediary between a sender and receiver that would receive the video from the

sender, transcode the media to a lower bit-rate form in real-time, and forward the transcoded video to receivers that otherwise could not receive the original video data in real-time (*e.g.*, because those receivers attached to the network via lower capacity links than the sender or other receivers). For instance, in a U.S. patent application published in 2002, Vasudevan describes an "an adaptive analysis and transcoding system" that "dynamically and continuously determines a client's available network bandwidth and transmits a single MPEG stream, altering it to meet the needs of various client devices." Ex-1014 at [0023]; *id*. ("[I]f the system modifies the stream to drop to a lower bit rate, fewer packets are required to send the lower bit rate data during any given time frame").

53. Indeed, the now well-known and widely used Internet standard media transport protocol RTP ("Real-Time Transport Protocol"), first formally published in 1996, included intermediaries called Mixers and Translators that would transcode audio and video to, among other things, ensure that bandwidth constrained recipients behind a Mixer/Translator could now receive the media of the communication session in real-time. Ex-1015 (RFC 1889) at 6-7. Interestingly, RTP also included a companion protocol called RTCP ("Real-Time Control Protocol") that could be used by receivers to communicate performance metrics concerning the reception of the video data being transmitted to a sender for use by the sender to, among other things, adapt the video stream it was

transmitting. And by at least 1997—ten years before the priority date of the ’557 patent—I was teaching concepts of media adaptation and transcoding as performed in protocols such as RTP to students at UNC. Exhibits 1016 and 1017 are copies of some of my lecture notes from 1997 illustrating the operation of RTP Mixers and Translators. By 2001, undergraduate networking textbooks contained comprehensive treatments of the principles and processes of adaptive multimedia data delivery including discussions of real-time streaming protocols. *See* Ex-1018 (Kurose, Ross) at 541-57.

54. Given the state of affairs in adaptive transmission of audio/video data, it is not surprising that the basic concepts claimed in the ’557 patent of ascertaining bandwidth, degrading the quality of video streams in response, and possibly transcoding media data at an intermediary were widespread in the art prior to the priority date of the ’557 patent. Beyond the works described here, the analysis below shows that subject matter of the ’557 patent had been invented and disclosed by other inventors prior to the priority date of the ’557 patent.

C. Video Streaming Protocols

55. The ’557 patent claims the use of real-time streaming protocols for video transmission. But several protocols were in widespread use for video streaming well before 2007.

56. Generally, the phrase “real-time” in “real-time streaming protocols” is

a reference to the fact that the protocol is meant to carry data from a real-time application such as a voice/video telephony application or a video conferencing application. That is, "real-time" is a reference to the requirements of an application and not to any delivery service provided by the protocol. This is because, as mentioned, today's public data networks, such as the Internet, support only "best effort" data delivery services and do not guarantee real-time data transmission. Thus, the focus of a real-time streaming protocol is not on real-time delivery of data but rather on allowing real-time media such as audio and video that is typically continuous, to be packetized and transmitted over best-effort networks. Ex-1018 (Kurose, Ross) at 551.

57. By 2007, the most prominent real-time streaming protocol was the aforementioned RTP protocol. First standardized in 1996, by the early 2000s, it was widely used for Internet voice/video telephony applications as well as video conferencing and broadcasts of live video events. As an Internet standard, RTP included a rich set of "media profiles" — essentially definitions for packetizing a wide variety of media types and encodings—that allowed RTP to carry media data output by a large number of codecs. The publishing of these media profiles contributed greatly to the development of interoperable media sending and receiving applications as they standardized the packet formats for all the popular codecs and compression schemes in use.

58. Other protocols used before 2007 to stream real-time video included proprietary protocols such as the RDT (Real Data Transport) protocol from RealNetworks and MMS (Microsoft Media Server) used by Microsoft. *See* Ex-1019 (Helix Server Admin. Guide) at 25, 31, 259.

59. Higher-layer protocols for streaming such as RTSP (Real-Time Streaming Protocol) were standardized in 1998 to control streaming sessions. Ex-1020 (RFC 2326). RTSP was used to control media servers to, for example, synchronize the media in multiple streams (such as audio and video) that comprise a multimedia presentation and to provide remote control features such as pausing the playout of media, rewinding, fast-forwarding, *etc*. (if supported by the application). RTSP was used to coordinate the transmission of media that would ultimately be carried on lower-layer protocols such as RTP, TCP, or UDP.

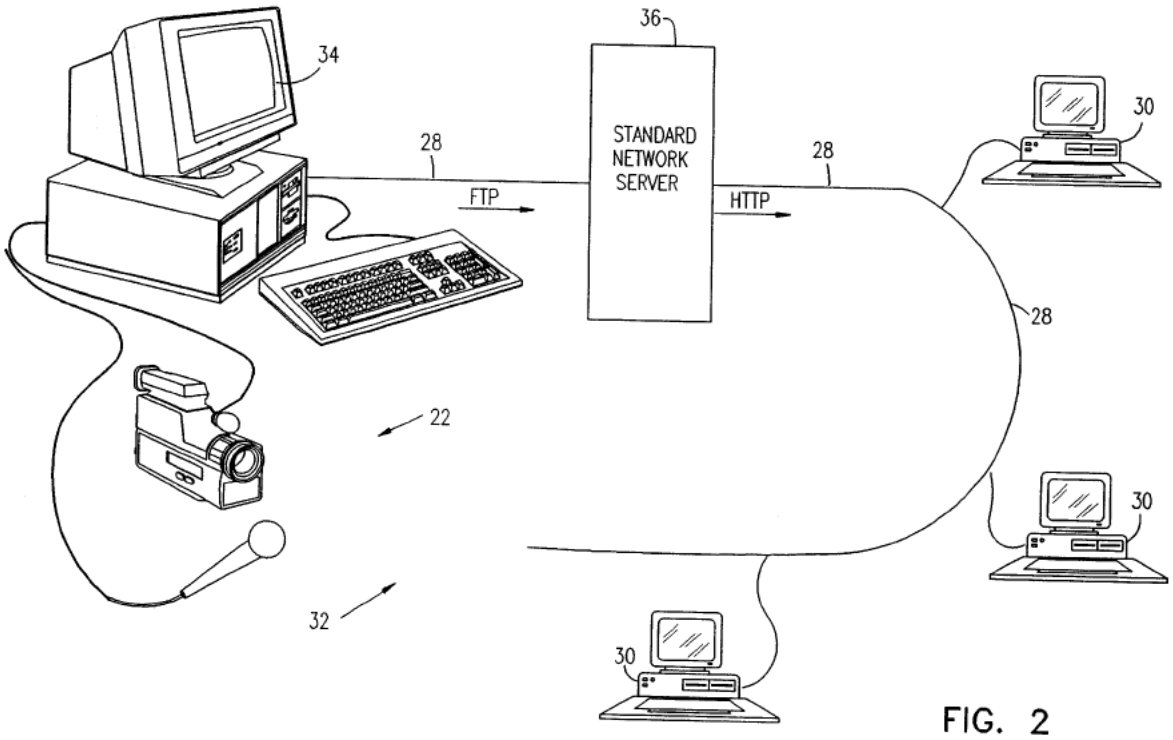
60. Even HTTP has been used as a real-time streaming protocol. For instance, the Carmel patent, filed in March 1999, discloses a method of uploading video to a server that processed the video in real-time such that "[t]he clients download the data stream from the server, preferably using an Internet protocol, as well, most preferably the Hypertext Transfer Protocol (HTTP), or alternatively, using other protocols, such as UDP or RTP, which are similarly known in the art." Ex-1021 at 2:11-15.

61. By 2007, HTTP was in widespread use as a real-time streaming

protocol. Several factors contributed to this. Whereas in the mid- to late-1990s, video was typically displayed via standalone player applications (*e.g.*, the RealNetworks RealPlayer app), by 2007 the web browser was the dominant platform for displaying video. In addition, HTTP was particularly well-suited for video streaming since by the early-2000s, network security was a widespread concern and firewalls were widely deployed to limit the types of network traffic entering a network. But given the dominance of the web, HTTP traffic was almost universally allowed to transit firewalls, hence streaming video over HTTP greatly simplified the problems of video distribution.

62. In certain circumstances, applications and video distributions schemes might use multiple real-time streaming protocols. For example, in video distribution system wherein video producers transmit video to an intermediate server that then distributes the video to users, video producers might use one protocol to communicate with the intermediate server and the server might use a second protocol to distribute the video. For example, the video producers might use RTP to communicate video to the server and the server might use RTSP/RTP to communicate with users to enable users to control the playout of the media they receive (*e.g.*, through VCR-like operations of pause, rewind, fast, forward, etc.). Similarly, a video producer might use RTP while users might use HTTP to retrieved video so it can be viewed in a browser. The Carmel patent, discussed

above, provides an example of the use of multiple streaming protocols as it discloses the use of a file transfer protocol in real-time to communicate video from a video source to a server and then the use of HTTP by users to retrieve the video as illustrated in FIG. 2 below:



Ex-1021, Fig. 2; *see also id.* at 2:7-10.

D. Enabling Simultaneous Rendering of Content from Multiple Sources

63. The ’557 patent also claims methods for operating the claimed video messages service infrastructure so as to receive video content from multiple sources and transmit such content in near real-time to a recipient device such that the recipient device can simultaneously render the media from both sources. But

such functionality was widespread long before the priority date of the ’557 patent. For instance, the International Telecommunication Union (ITU) first published a version of the standards-series H.323 recommendation in November 1996 with an emphasis on enabling videoconferencing capabilities over a local area network (LAN). In 1998, the ITU published a revised version of H.323 that describes a standards-based recommendation for providing multi-party video-conference services over packet-based IP networks, such as the Internet. *See* Ex-1022 (H.323). By the late 1990s, server-based video conferencing systems enabling users to receive and display real-time video feeds from multiple senders were common in the industry. As described in the 1998 H.323 recommendation, to enable video conferencing between three or more participants, such systems often used a central Multipoint Control Unit (MCU) to receive real-time media from the various participants, mix those streams, and deliver a single composite stream to each of the video-conference participants. Ex-1022 at 7, 30-31.

64. It was common for video-conferencing systems to adapt the video media delivered to the end user to match the capabilities of the recipient device and/or to account for bandwidth availability between the server and the recipient. As one example, the Clisham reference describes a video conferencing system that adapts the quality of the media delivered to each end user on-the-fly based on changes to the quality of the communication link. Ex-1023 at [0079-81].

Similarly, the Kenoyer patent describes a video conferencing system that employs a transcoder embedded in a MCU to ensure that the video media received from each conference participant can be delivered and displayed on each participant's screen at the highest possible resolution supported by that screen. Ex-1024 at 1:21-67, 5:4-67, Abstract.

65. Long before 2007, other types of systems also commonly provided a server-level infrastructure to receive, process, and transmit real-time video media from multiple sources such that the video media from multiple sources could be simultaneously displayed on the recipient device. For example, such functionality was commonly provided in distance learning applications, such as the multimedia distance learning system used at the University of Washington. *See* Ex-1025 (Deshpande); *see also* Ex-1026 at 30-32, 38-39; Figs. 1, 8, 9 (describing SameView interactive E-learning system).

66. Likewise, well before 2007, it was well known in the art that live news and entertainment could be provided to end users over the Internet such that content from multiple sources would be simultaneously displayed at the recipient device. For instance, in an application filed in 1997, the Hendricks patent describes a method for communicating multiple live video feeds over the Internet such that recipients may receive and simultaneously render those feeds in real-time. Ex-1027 (Hendricks) at 3:40-4:24, 9:51-1-:15, 16:58-17:3, Abstract; Figs.

7A-B, 14.

67. Years before the priority date of the ’557 patent, it was also well known that such functionality could be provided in group communication applications, such as press-to-view (PTV) systems. For instance, the Rengaraju reference discloses a PTV system wherein each user may display in a split screen mode the video transmitted by other participants in the group call. Ex-1028 at [000029-33, 36].

V. THE ’557 PATENT

A. Overview of the ’557 Patent

68. U.S. Patent No. 10,511,557, titled “Telecommunication and Multimedia Management Method and Apparatus,” was filed on October 16, 2018 and issued on December 17, 2019 to Katis, Panttaja, Panttaja, and Ranney. Ex-1001.

69. The specification of the ’557 patent states that “[t]his invention pertains to telecommunications, and more particularly, to a telecommunication and multimedia management method and apparatus that enables users to review the messages of conversations in either a live mode or a time-shifted mode and to transition the conversation back and forth between the two modes, participate in multiple conversations and to archive the messages of conversations for later review or processing.” Ex-1001 at 1:33-41.

70. To address supposed inadequacies in then-existing communications systems, such as telephones, voicemail systems, and tactical voice communication systems used by emergency responders, the '557 patent purports to provide a method and apparatus "that supports new modes of engaging in conversations and/or managing multiple conversations using a variety of media types." Ex-1001 at 4:8-22. In the disclosed system, "[u]sers can engage in conversations by sending messages to designated recipients." *Id*. at 4:22-23. Recipients may then review the media in real-time or, alternatively, "in a time-shifted mode by reviewing and replying to the recorded message at their convenience." *Id*. at 4:23-29.

71. The '557 patent also purports to enable "users to manage multiple conversations at the same time, where either (a) one conversation is current and all others are paused; (b) multiple conversations are rendered consecutively, such as but not limited to tactical communications; or (c) multiple conversations are active and simultaneously rendered, such as in a stock exchange or trading floor environment." Ex-1001 at 7:10-16.

72. In addition, the '557 patent allegedly enables network bandwidth to be optimized "by making tradeoffs between timeliness and media quality, using the presence and intentions of whether or not recipient(s) intend to review to the message immediately in real time, as well as measures of network latency, network degradation, packet loss or damage, and/or current bandwidth conditions." Ex-

1001 at 5:19-25.

73. To optimize the transmission of media in view of bandwidth conditions, the '557 patent provides a Packet Quality Manager (PQM) running on clients and servers. The PQM "relies on a metric called the Maximum Available Bit Rate (MABR), which is a continually computed approximation of actual transmission capacity or bandwidth (i.e., a measure of the capability of the network at a given point in time) between a sending and receiving node pair." Ex-1001 at 29:57-62. "The PQM further considers an Ideal Bit Rate or 'IBR' for each scheduled Message payload transmission." *Id*. at 30:32-33. "If multiple messages or payloads are to be sent between a sending and receiving pair, then an Aggregate IBR (AIBR) is considered for the transmission." *Id*. at 30:52-55.

74. "The PQM operates by sending data in a series of transmission loops for each sending and receiving pair." Ex-1001 at 30:56-57. To determine the quality of media to transmit, the PQM compares the MABR to the AIBR. *Id*. at 31:57-59. If the MABR is greater than or equal to the AIBR, the data identified as ready for transmission is packetized at the IBR rate and transmitted. *Id*. at 31:63-65. If the MABR is less than the AIBR, then a series of procedures are undertaken to degrade the quality of outgoing media. *Id*. at 31:66-32:4, Fig. 9C.

75. For example, if the MABR is less than the AIBR for a given transmission loop, the system may switch to a lower quality codec. Ex-1001 at

32:19-31, Fig. 9C. If the use of a lower quality codec still does not meet the MABR, the packetization interval may be increased. *Id*. at 32:32-40. If after changing the packetization interval the MABR is still not met, then the bit rate may be progressively lowered. *Id*. at 32:41-43.

76. This sequence is repeated for each transmission loop between the sending and receiving pair. Ex-1001 at 32:48-50. If a sending node is transmitting simultaneously to multiple receiving nodes, the same sequence occurs concurrently for each receiving-sending pair. *Id*. at 32:64-67.

B. Overview of Challenged Claims

77. I understand that claim 1 of the ’557 patent is the only independent claim challenged in the Petition that this declaration is supporting. Claim 1 claims a “method for operating a video message service infrastructure” that involves generating multiple degraded versions of live video media and selecting one such version for delivery to a recipient based on ascertained bandwidth.

78. In particular, claim 1 requires (among other things) receiving live video media from a sending device, generating two or more degraded versions of the video media by degrading the quality by a different amount, selecting one of the degraded versions based on ascertained bandwidth for delivering the video media to a recipient device, and transmitting the selected degraded version to the recipient device such that it may be rendered while the video media is being

created and transmitted by the sending device. ’557 patent, cl. 1.

79. In my opinion, as explained below, none of the elements of claim 1 is remotely novel, individually or in combination. I note that to the extent the ’557 patent’s specification describes the particular method recited in claim 1, it does not describe the claimed method as novel. Indeed, the ’557 patent itself acknowledges that factors such as “network latency, network degradation, packet loss, packet damage, and general bandwidth conditions” were “commonly considered” in prior art systems to “determin[e] how much data to transmit.” Ex-1001 at 3:64-4:1. I agree. As explained above, more than 15 years before the priority date of the ’557 patent, my students at UNC and I developed, demonstrated, and published real-time videoconferencing systems that received live audio and video from a sending device that generated multiple degraded versions of the video and selected one of the degraded versions for transmission based on measurements of the bandwidth available for delivering the video to a recipient device. *See supra*, ¶50; *see also supra*, ¶¶46-49, 41-54.

80. The remaining challenged claims depend from claim 1. The challenged dependent claims add other concepts that also were well known to skilled artisans at the time of the alleged invention. For instance, claim 2 adds the limitation that one of the two degraded versions is generated by reducing the bit rate of the video media received from the sending device. But the concept of

adaptive bitrate streaming was well known to persons of ordinary skill in the art at the time of the alleged invention, as explained below. *See also supra*, ¶¶51-52.

81. Dependent claim 3 requires transmitting at least one of the degraded versions to two or more additional recipient devices and selecting the degraded version(s) based on bandwidth conditions between the "video messages service infrastructure" and each recipient device. But as explained below, this was no more than a simple and straightforward extension of the concept claimed in claim 1, and was itself not novel.

82. Claim 8 requires receiving a video message at the "video message service infrastructure" using a first real-time streaming protocol and sending the selected degrading version to the recipient device using a second real-time streaming protocol. Claim 9 depends from claim 8 and adds the requirement that the first real-time streaming protocol is different than the second real-time streaming protocol.

83. But as explained below, persons of ordinary skill in the art were well aware that any of various real-time streaming protocols, including commonly used protocols such as RTP or RTSP (which stands for "real-time streaming protocol"), could be used to stream video media from a sending device to an intermediate server. Skilled artisans also would have understood that the same real-time streaming protocol, or any of various other real-time streaming protocols, could be

used to stream video from the server to the recipient device. *See supra*, ¶¶55-62. The selection of a particular real-time streaming protocol for receiving and transmitting video media would have been a design choice well within the competence of a person of ordinary skill and would represent nothing more than the use of a known element or technique for its intended purpose.

84. Claim 14 recites transcoding video media encoded using a first codec to a different format to be decoded at the recipient device using a second codec. But, as explained below, there was nothing novel about transcoding streaming media received in one format into a different format suitable for rendering on a recipient device. *See also supra*, ¶¶52-53. Indeed, the '557 patent does not claim otherwise. And while the '557 patent discloses no purpose for its transcoding step, as discussed above, protocols such as RTP included components to transcode media as the media was transmitted to best ensure all recipients can playout the media.

85. Similarly, claim 16 of the '557 patent purports to require streaming video media from two sending devices such that the two streams can be rendered simultaneously by the recipient device. In my opinion, this concept was well known at the time of the alleged invention. *See supra*, ¶¶63-67. As explained below, prior art content delivery systems were able to receive live video streams from any number of content providers and stream any combination of them in real-

time for simultaneous rendering on recipient devices.

86. Claim 23 requires that the "video message service infrastructure" include "a plurality of servers collectively configured for performing" standard streaming functionality, such as receiving, storing, and transmitting video media to multiple recipients. And while claim 23 also requires that "locations" on the network for the recipient devices be ascertained from identifiers provided by the sending device, the specification of the '557 patent does not purport to describe any novel method for accomplishing this function.

87. There was nothing novel about enabling senders to identify the recipients for live video messages, as explained below. Indeed, to the extent the '557 patent describes any method for ascertaining delivery locations based on identifiers from a sending device, the '557 patent relies on long-standing and Internet standard methods such as using the Domain Name System (DNS) to ascertain delivery locations and acknowledges that other such methods were already known in the art. *See* Ex-1001 at 26:9-21 ("the system 10 uses one of the above mentioned ***or other known addressing schemes*** for delivery of Vox packets 95 to the correct location") (emphasis added), 22:45-49.

C. Effective Filing Date of the '557 Patent

88. The '557 patent issued on December 17, 2019, from an application filed on October 16, 2018. '557 patent at cover (Ex-1001). The '557 patent

purports to claim priority to various U.S. patent applications, the earliest of which is application No. 12/028,400, filed on February 8, 2008. The '557 patent also purports to claim priority to several provisional applications, the earliest of which is provisional application No. 60/937,552, filed on June 28, 2007.

89. I have used June 28, 2007, as the effective filing date for my analysis of the asserted claims of the '557 patent because the patent claims priority to the provisional application filed on that date. I note, however, that the provisional application does not discuss all of the elements of the challenged claims.

90. For instance, claim 1 recites the step of "generating, by the video message service infrastructure, two or more degraded versions of the video media of the video message, where each of the two or more degraded versions is created by degrading the quality of the video media by a different amount." Ex-1001 at 44:33-37. The provisional application does not discuss generating multiple versions of video media received from a sending device, nor does it discuss degrading the quality of the video media. *See* Ex-1004 at 1-16 (Ex-1004).

91. Likewise, the provisional application does not discuss the concept recited in claim 1 of "selecting one of the degraded versions of the video media based on an ascertained bandwidth on the network for delivering the video media from the video message service infrastructure to a recipient device on the network." Ex-1001 at 44:38-42. Nor does the provisional application discuss

"transmitting the selected degraded version of the video media from the video message service infrastructure to the recipient device," as recited in claim 1. *Id*. at 44:43-45.

D. Level of Ordinary Skill in the Art

92. I have been informed that, in reaching my opinion as to the appropriate level of one of ordinary skill in the art, I may consider the following factors: (a) the types of problems encountered by those working in the field and prior art solutions thereto; (b) sophistication of the technology in question, and the rapidity with which innovations occur in the field; (c) the educational level of active workers in the field; and (d) the educational level of the inventor.

93. The relevant technical field for the '557 patent is multimedia communication, including technologies for streaming media over a network. Based on this, and the factors above, in my opinion, a person of ordinary skill in the art as of June 2007 would have had a working knowledge of computer networking and distributed systems generally and messaging and multimedia communication systems specifically. Such a person would have had a bachelor's degree in computer science, computer engineering, or an equivalent degree, and 2-3 years of professional experience with distributed multimedia systems. More experience may substitute for education, or vice versa.

94. My opinions regarding the level of ordinary skill in the art are based

on, among other things, my nearly 40 years of experience in the relevant areas and my understanding of the basic qualifications that would be relevant to an individual tasked with investigating the methods and devices in the relevant areas. Given my experience teaching and collaborating with industry, both before and after the priority date of the ’557 patent I had familiarity with the backgrounds of students, researchers, and industry professionals who worked with the aforementioned technologies.

95. Although my qualifications and experience exceed those of the hypothetical person having ordinary skill in the art defined above, my analysis and opinions regarding the ’557 patent have been based on the perspective of a person of ordinary skill in the art as of June 2007.

96. My opinions herein regarding the understanding of a person of ordinary skill in the art and my other opinions set forth herein would remain the same if the person of ordinary skill in the art were determined to have somewhat more or less education and/or experience than I have identified above.

97. Moreover, although I have used June 28, 2007, as the effective filing date for my analysis of the asserted claims of the ’557 patent, my opinions regarding the understanding of a person of ordinary skill in the art and my other opinions set forth herein would remain the same if the effective filing date were determined to be somewhat earlier or later.

VI. CONSTRUCTION OF THE TERMS USED IN THE ’557 PATENT

98. I understand that terms used in patent claims are generally given the meaning that the term would have had to a person of ordinary skill in the art at the time of the invention. I also understand that the person of ordinary skill in the art is deemed to read a claim term not only in the context of the particular claim in which the term appears, but in the context of the entire patent, including the specification.

99. I am informed by counsel that the patent specification, under the legal principles, has been described as the single best guide to the meaning of a claim term, and is thus highly relevant to the interpretation of claim terms. I am further informed by counsel that other claims of the patent in question, both asserted and unasserted, can be valuable sources of information as to the meaning of a claim term. Because the claim terms are normally used consistently throughout the patent, the usage of a term in one claim can often illuminate the meaning of the same term in other claims. Differences among claims can also be a useful guide in understanding the meaning of particular claim terms.

100. I understand that the prosecution history can further inform the meaning of the claim language by demonstrating how the inventors understood the invention and whether the inventors limited the invention in the course of prosecution, making the claim scope narrower than it otherwise would be.

Extrinsic evidence, such as dictionary definitions or expert opinions such as mine, may also be consulted in construing the claim terms.

101. I have been informed by counsel that, in *inter partes* review proceedings, a claim of a patent shall be construed using the same claim construction standard that would be used to construe the claim in a civil action filed in a U.S. district court (which I understand is called the "*Phillips*" claim construction standard), including construing the claim in accordance with the ordinary and customary meaning of such claim as understood by one of ordinary skill in the art and the prosecution history pertaining to the patent.

VII. OVERVIEW OF PRIOR ART REFERENCES

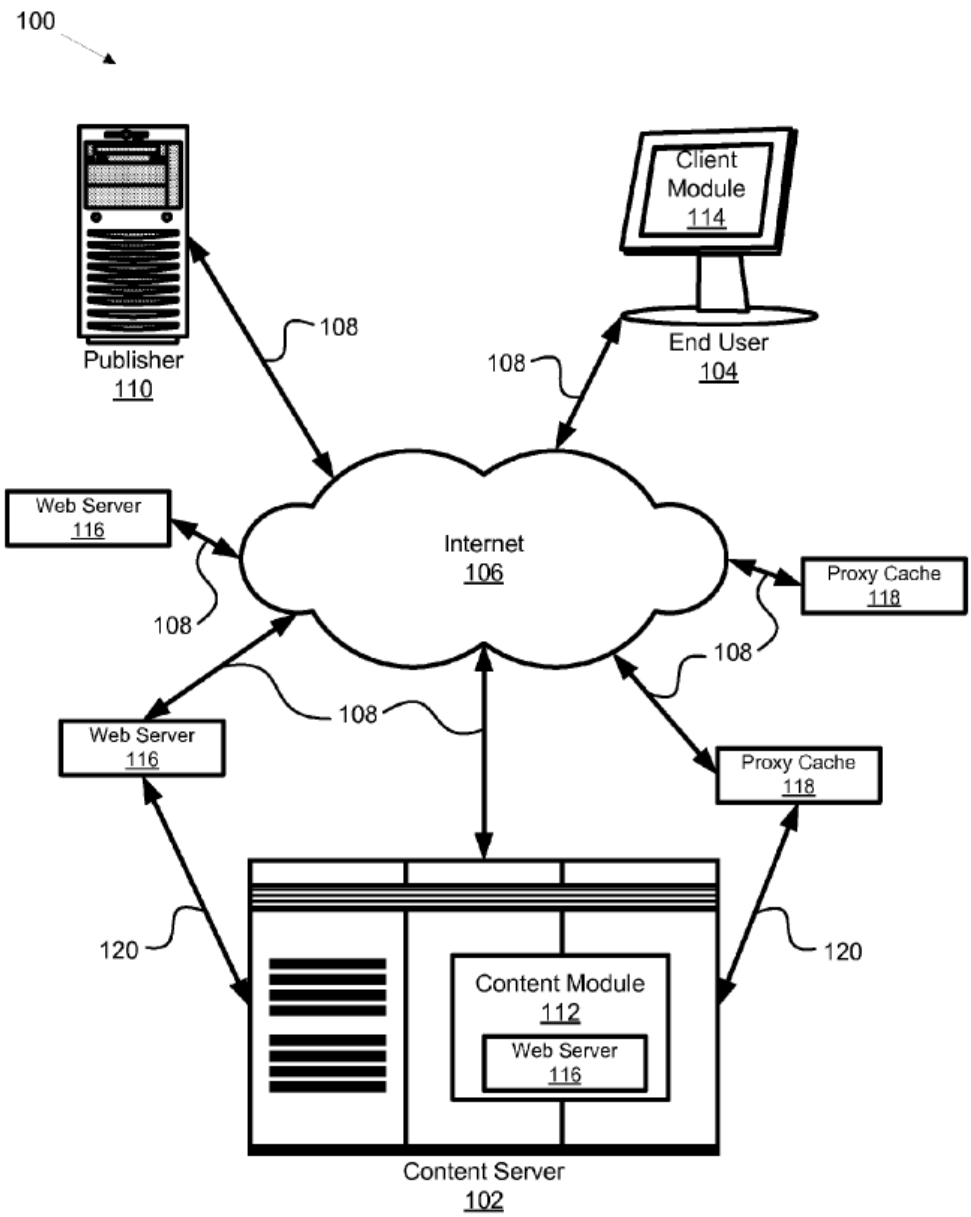
A. Brueck

102. Brueck, a U.S. patent entitled "Apparatus, System, and Method for Multi-Bitrate Content Streaming," describes a multimedia content delivery system that adapts the quality of streaming media to account for network bandwidth conditions.

103. As Brueck explains, streaming media technology provides a number of advantages, including giving the user "the immediate satisfaction of viewing the requested content without waiting for the media file to completely download." Ex-1005 at 1:32-35. However, a particular challenge with real-time streaming is that "the audio/video quality that can be received for real-time presentation is

constrained by the available bandwidth of the user's network connection." *Id*. at 1:29-37. "Streaming is also vulnerable to network failures or congestion" that can adversely impact the end user's experience. *Id.* at 1:58-59, 2:18-19. Thus, streaming "currently sacrifices quality compared with downloading a file of the same content." *Id*. at 1:50-52.

104. Brueck overcomes these challenges by proposing a system that dynamically adapts the quality of media streamed to the user based on current bandwidth conditions. *Id*. at 6:54-8:25, 16:11-17:21. Specifically, Brueck provides a content server, as shown in Figure 1, that receives streaming media, such as live video, from a content publisher, generates multiple versions of the content at varying bitrates and quality levels, and delivers to the end user a version of the video media commensurate with current bandwidth conditions:



Id., Fig. 1; *see also id*. at 1:16-19, 5:64-6:3, 6:15-30, 6:40-47, 6:54-65, 8:6-25, 12:38-44.

105. In Brueck's system, users may review live streaming video in near real-time or in a time-shifted mode by, for instance, pausing, rewinding and replaying live content. *Id*. at 2:40-42, 6:40-47, 8:42-44, 10:44-45, 10:66-67,

12:45-47.

106. Figure 4 provides an example of how content received from a content publisher 110—*i.e.*, a "creator or distributor of content" (*id*. at 6:16-17)—may be adapted to accommodate the bandwidth available to end users:

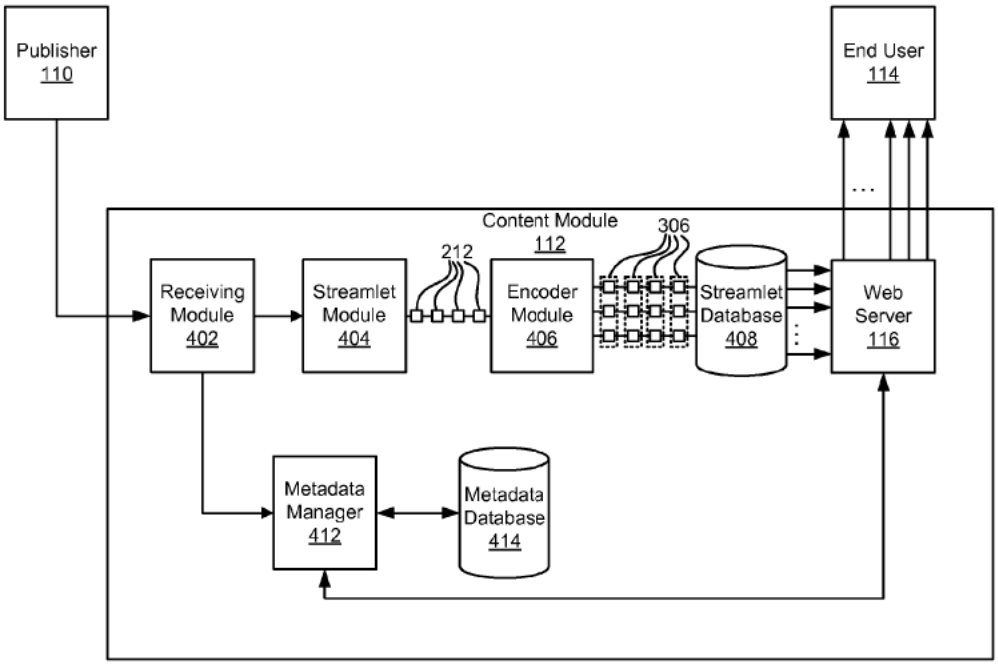


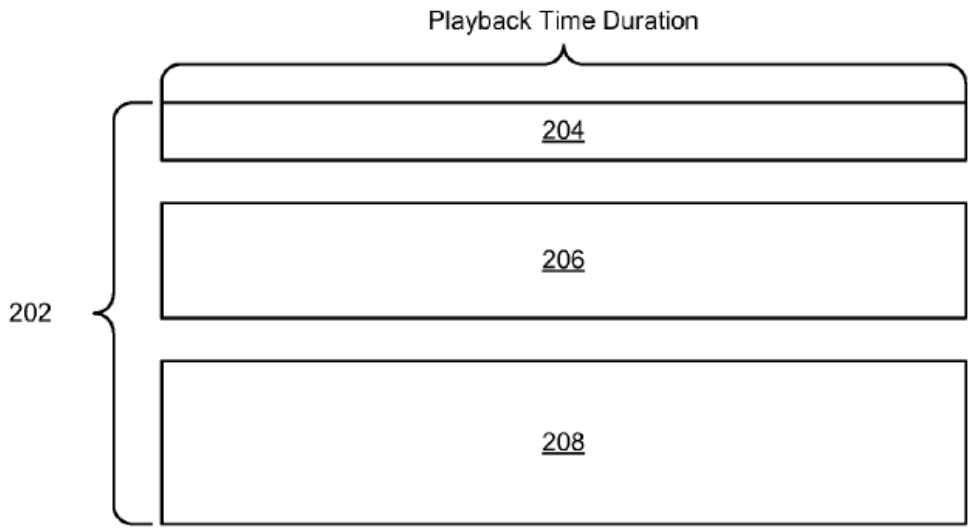
FIG. 4

Brueck, Fig. 4.

107. As shown in Figure 4, a receiving module 402 (also referred to in the specification as the "capture module" (*e.g.*, *id*. at 7:42-45)) receives a content file comprising, among other things, live streaming video, from a publisher 110, then passes it to streamlet module 404, which segments the content file into streamlets 212 having either equal file sizes or representing equal durations. *Id*. at 7:45-8:5.

108. Encoder module 406 then generates, encodes, and stores a plurality of

streams of varying qualities. *Id*. at 8:6-29. In Figure 2B, for instance, Brueck provides a diagram showing "a plurality of streams 202 having varying degrees of quality and bandwidth":



Id., Fig. 2, 6:40-42. "[T]he plurality of streams 202 comprises a low quality stream 204, a medium quality stream 206, and a high quality stream 208." *Id*. at 6:56-58. "For example, the low quality stream 204 may be encoded and compressed to a bit rate of 100 kilobits per second (kbps), the medium quality stream 206 may be encoded and compressed to a bit rate of 200 kbps, and the high quality stream 208 may be encoded and compressed to 600 kbps." *Id*. at 6:60-65.

109. Although in the example of Figure 2B, three streams are encoded at different quality levels, Brueck explains that "the plurality of streams 202 may comprise any number of streams deemed necessary to accommodate end user bandwidth." *Id*. at 8:23-25.

110. An appropriate stream is selected and transmitted to the end user station 114 based on a continuous observation of bandwidth conditions. *Id*. at 12:20-44. As bandwidth conditions change, the system may dynamically switch to a stream of higher or lower quality, as shown in Figure 10:

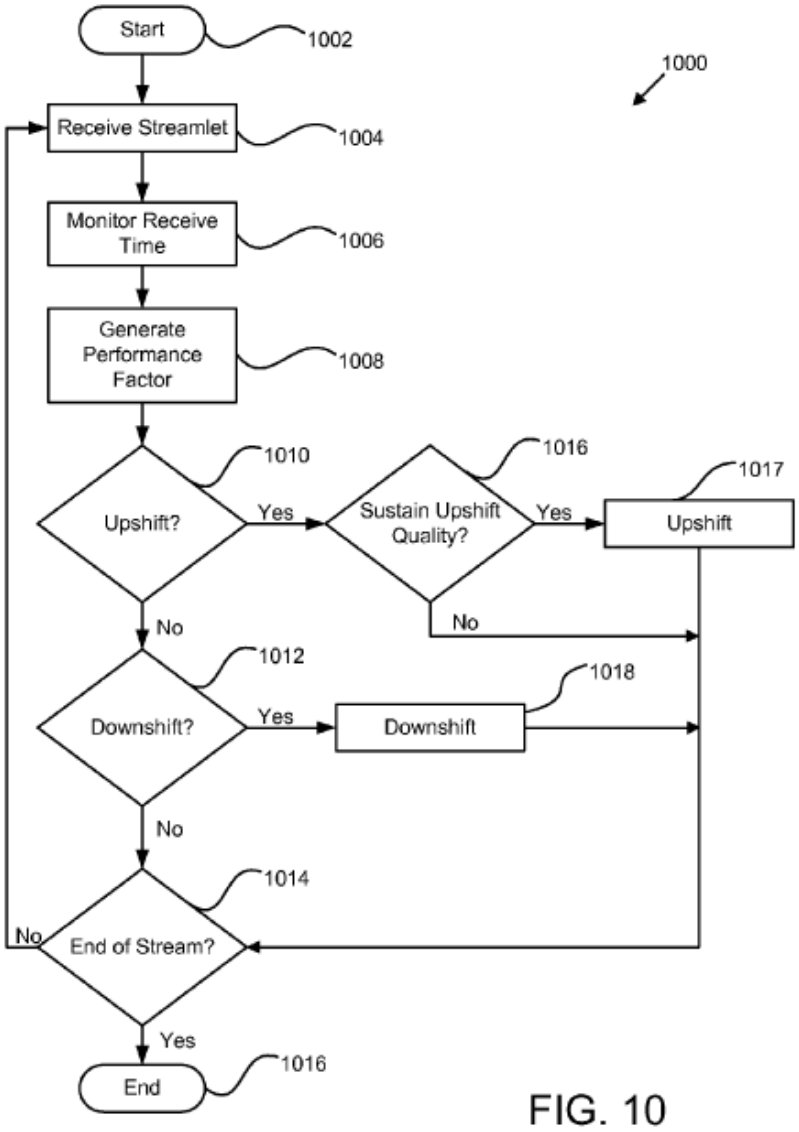


FIG. 10

Id., Fig. 10; *see also id*. at 16:11-17:21.

B. Brooks

111. Brooks, a U.S. patent entitled "Methods for Transforming Streaming Video Data," discloses a multimedia content delivery system that "adapt[s] input streams of video data to meet desired parameters for output streams of video data." Ex-1006 at 1:15-19. In particular, Brooks discloses "methods for transcoding and transforming video streams," such as live video from a video camera, to match the capabilities of recipient devices. *Id*. at 3:8-12, 3:1-4; 6:32-36, 9:14-17.

112. In Brooks' system, live streaming video may be received from content publishers in various encoding formats, resolutions, frame rates, color depths, etc. *Id*. at 9:33-39, 9:53-62, 16:48-58. For instance, the source stream may be "encoded in formats such as, JPEG, JPEG-2000, GIF, WBMP, MPEG-1, MPEG-2, MPEG-4, H.263, *.avi, *.mov, *rm, *.aff, and the like." *Id*. at 9:33-38.

113. As shown in Figure 1, the system then transforms the input streams "to provide video data to each device according to that device's bandwidth limitations, and in the output format desired":

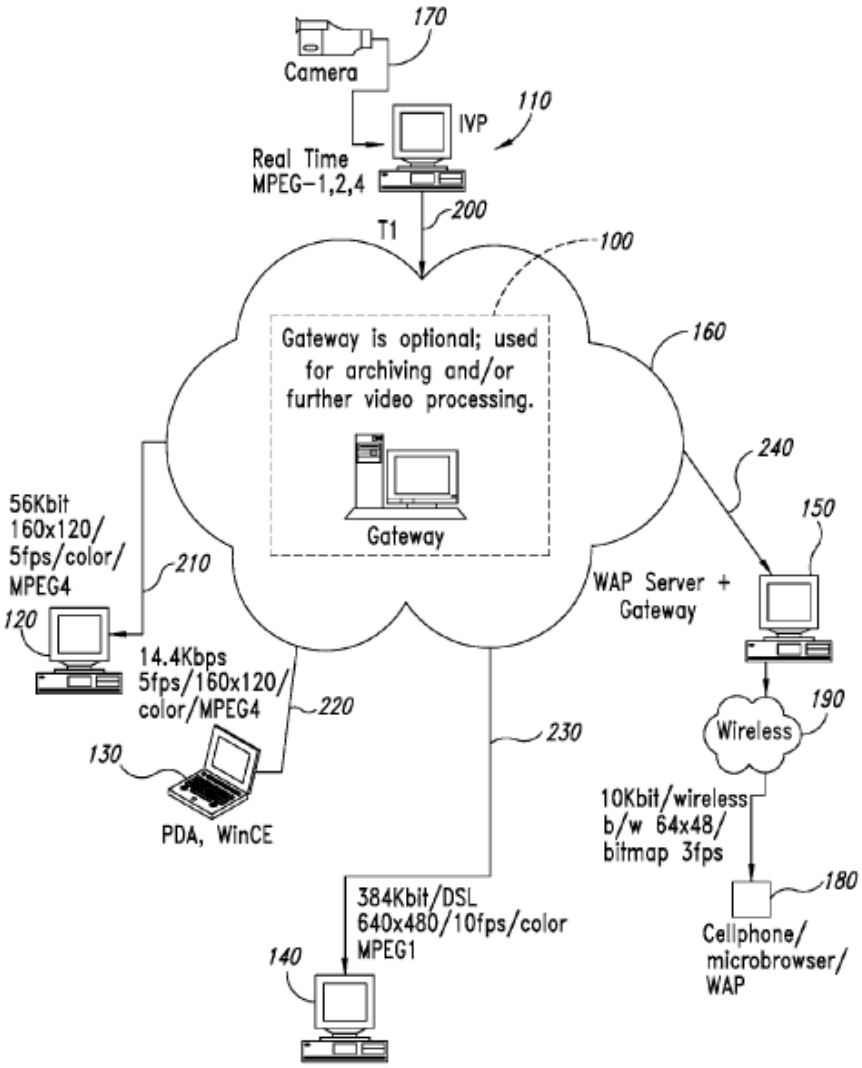


FIG. 1

Ex-1006, Fig. 1, 7:21-25. "On the fly adaptation to desired output parameters may be made with respect to display size, frame rate, bit-depth, bit rate, encoding format, and the like." *Id*. at 3:12-14.

114. Brooks explains that data streams transmitted from sending devices may be received at the content delivery system using "a real-time streaming

protocol known in the industry such as RTP, or the like." Ex-1006 at 9:26-29, 16:46-48 ("the input stream of video data use RTP or other data streaming protocols"). To stream the video media over the network to recipient devices, the system uses any of various steaming protocols, including RTP, or "any other conventional and future-developed streaming protocol." *Id*. at 12:4-12. Brooks notes that such "communication protocols are well known." *Id*. at 6:6-10.

C. Mullig

115. Mullig, a U.S. patent entitled "System and Method for Delivering Video and Audio Content Over a Network," discloses a multimedia content delivery system that "concurrently streams a plurality of live videos" over a network "allowing the user to view two or more live videos at the same time." Ex-1007 at 1:38-42. A content delivery server is configured to receive a plurality of live video feeds from one or more content providers and to stream multiple feeds concurrently to a recipient device. *Id*. at 1:50-52, 9:38-44, 9:67-10:12, 10:57-11:8. For example, multiple live video feeds of a sporting event may be captured by a plurality of cameras and streamed to a recipient device such that the end user may view the feeds simultaneously. *Id*. at 12:3-12.

116. Mullig's system is configured to customize the quality of live video streamed to end users to accommodate user preferences and device capabilities. Mullig discloses, for example, that "the volume and type of data transmitted to and

from the [recipient devices] can be tailored to the requirements of the individual [recipient devices]." *Id*. at 6:3-7. In addition, the system is configured to "save bandwidth" by "stream[ing] at least one of the video feeds in high resolution and at least one of the video feeds in low resolution." *Id*. at 9:44-48. For instance, as shown in Figure 4, the receiving device may display "a high resolution version of one of the live video feeds in a 'primary' video pane 401 and low resolution versions of the additional live video feeds in 'preview' video panes 403 of the main display dialog window 400 substantially concurrently":

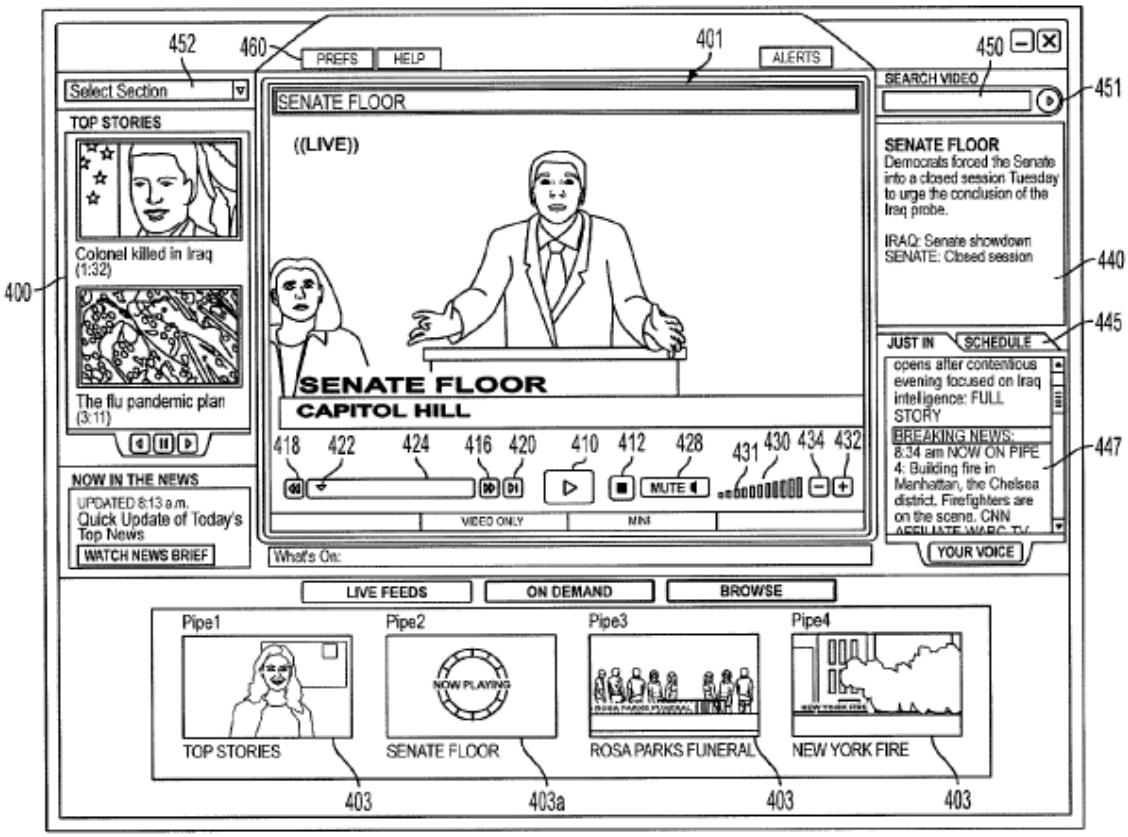


FIG. 4

Id., Fig. 4, 9:49-59. Users can select which feeds to receive in high or low resolution. *Id*. at 11:1-8. For instance, the user may select to receive video feeds belonging to a particular group or genre in low resolution. *Id*. at 11:8-14. The data rate of the high and low-resolution video streams delivered to the end user "may vary depending on the network connectivity between the [recipient device] and the content delivery server 25." *Id*. at 11:30-39.

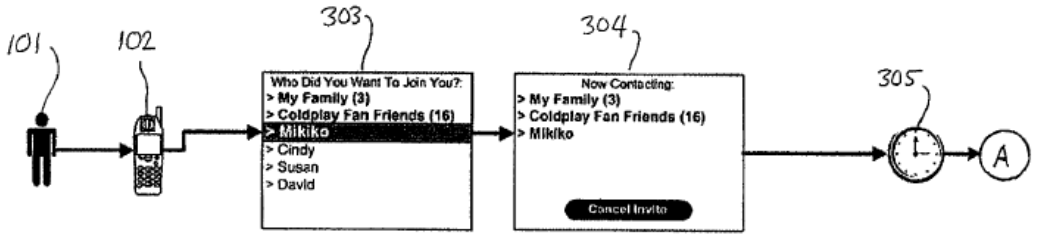
117. Users have the option to review live video feeds in near real-time or in a time-shifted mode. For instance, Mullig's system enables users "to control the play of a live video feed" such as by "pausing, re-starting, stopping, rewinding, fast-forwarding, 'jumping to live', and recording the live video feed." *Id*. at 12:25-29, 12:29-14:48.

D. Kalaboukis

118. Kalaboukis, a U.S. patent entitled "Telepresence Via Wireless Streaming Multicast," discloses a wireless communication system enabling users to transmit live video of an event from a wireless device "thereby allowing geographically dispersed members of a user's social network to experience a user's current locale, so that the user's friends may join him via live video at events (concerts, clubs, or other venues) which they cannot attend in person." Ex-1008 at 1:4-45, 1:34-35, 2:52-3:51.

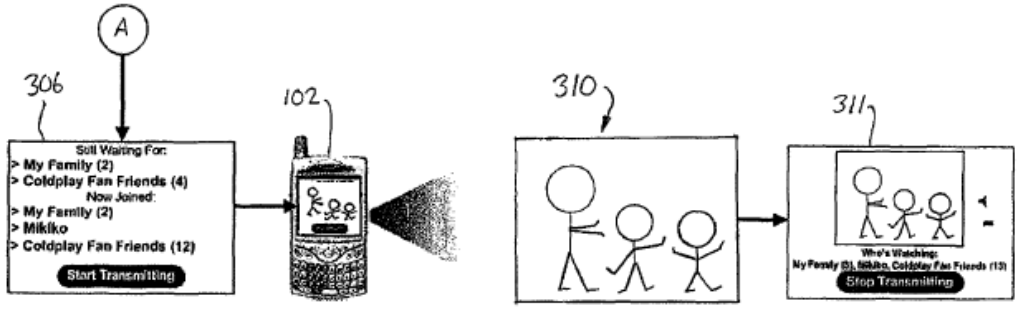
119. Using a "JoinMe" software application running on the user's wireless

device, the user first selects the intended recipients of the live video. *Id*. at 2:52-61. As shown in Figure 3A, invitees may include one or more individually selected users and/or group of users:



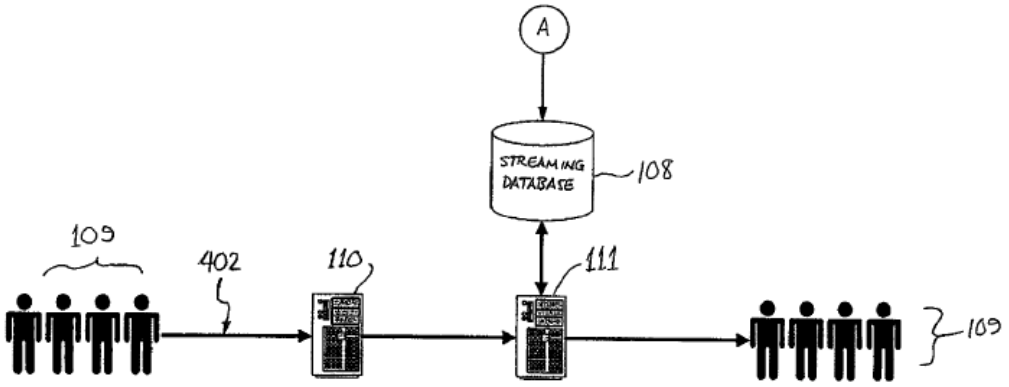
Id., Fig. 3A, 2:61-66.

120. The user's device 102 "is linked to an interaction server 110 which handles transmission of invitations and responses thereto." *Id*. at 2:66-3:1. After the user "has made his selection of whom to invite, the screen display 304 shows the progress of contacting the invitees":



Id., Fig. 3B, 4:27-29. "Some or all of the invitees may accept the invitation, thereby being a group of recipients 109." *Id*. at 3:2-4, Fig. 1C. Invitees may accept an invitation before or during the live broadcast. *Id*. at 3:24-26.

121. Live video is transmitted from the user device to "a central media capture server 107," which "accepts and caches the video stream . . . and causes it to be saved in a live streaming database 108, as shown in FIG. 1C":



Id. at 3:17-23, Fig. 1C.

122. As the video is being transmitted, the device shows the video being transmitted as well as who is watching. *Id*. at 4:38-43. Because the live video stream is stored at the server, an invitee accepting the invitation after the live broadcast has begun may either join the live event in progress or view it from the beginning. *Id*. at 3:34-37.

123. Kalaboukis also discloses customizing the live video transmission "so that each user may receive a streaming video signal in a convenient format." *Id*. at 3:52-55. For example, if a recipient is viewing the video stream on a less powerful wireless device, the streaming server may "reduce resolution or 'dumb down' the signal," while recipients viewing the video on a more powerful device may receive

an enhanced signal. *Id*. at 3:55-61.

E. Mangal

124. Mangal, a U.S. patent entitled "Method and System for Selectively Reducing Call-Set-up Latency Through Management of Paging Frequency and Buffering of User Speech in a Wireless Mobile Station," discloses a communication system enabling mobile device users to stream real-time media, including live video messages, over a network without waiting for a communication link to be established with the intended recipients. Ex-1009 at 1:11-18, 2:36-46, 4:45-5:3, 14:4-13, Fig. 6.

125. In one embodiment, the user may select the target recipients and begin streaming a video message to an intermediate PTT ("push-to-talk") or PTV ("push-to-view") server, which buffers audio and video media while the system locates the intended recipients. *Id*. at 12:65-13:49, 14:4-14, Fig. 6.

126. To identify the target recipients and their location on the network, Mangal uses the Internet standard Session Initiation Protocol ("SIP") so that "an initiating endpoint may send to a SIP proxy server a SIP 'INVITE' request message that designates a terminating SIP address" (the callee). *Id*. at 2:55-63; *see also* 3:33-45, 7:9-20, 8:59-64. "The proxy server may then query a SIP registry to determine an IP address of the terminating endpoint." *Id*. at 2:64-65; *see also* 2:65-3:6. Once a communication session, such as an RTP session between the

PTV server and the recipient device has been established (called a "leg" in Mangal), the server streams the buffered video to the recipient device. *Id*. at 13:19-38, 14:9-13, Fig. 6.

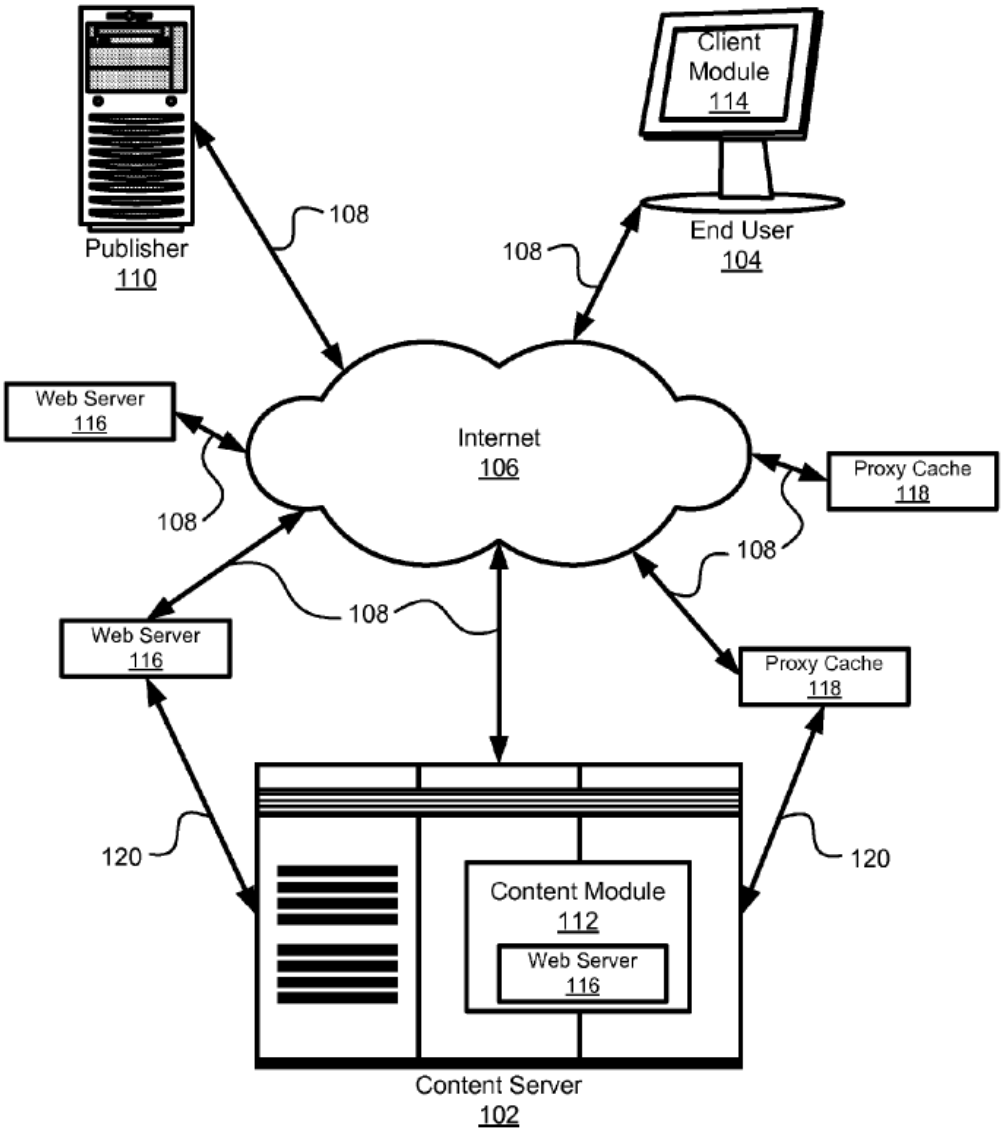
VIII. GROUND 1: CLAIMS 1-3, 8, 9, 14, AND 18 ARE RENDERED OBVIOUS BY BRUECK

A. Every Element of the Challenged Claims Is Disclosed or Rendered Obvious by Brueck

1. Claim 1

a. "A method for operating a video message service infrastructure on a network, the method comprising:" (Preamble)

127. In my opinion, to the extent the preamble is limiting, Brueck discloses it. As shown in Figure 1, Brueck's "video message service infrastructure" comprises a content server 102 with a content module 112 and a web server 116. One or more additional web servers 116 or proxy caches 118 also may be included. Ex-1005 at 6:31-33 ("Content from the content server 102 may be replicated to other web servers 116 or alternatively to proxy cache servers 118.").



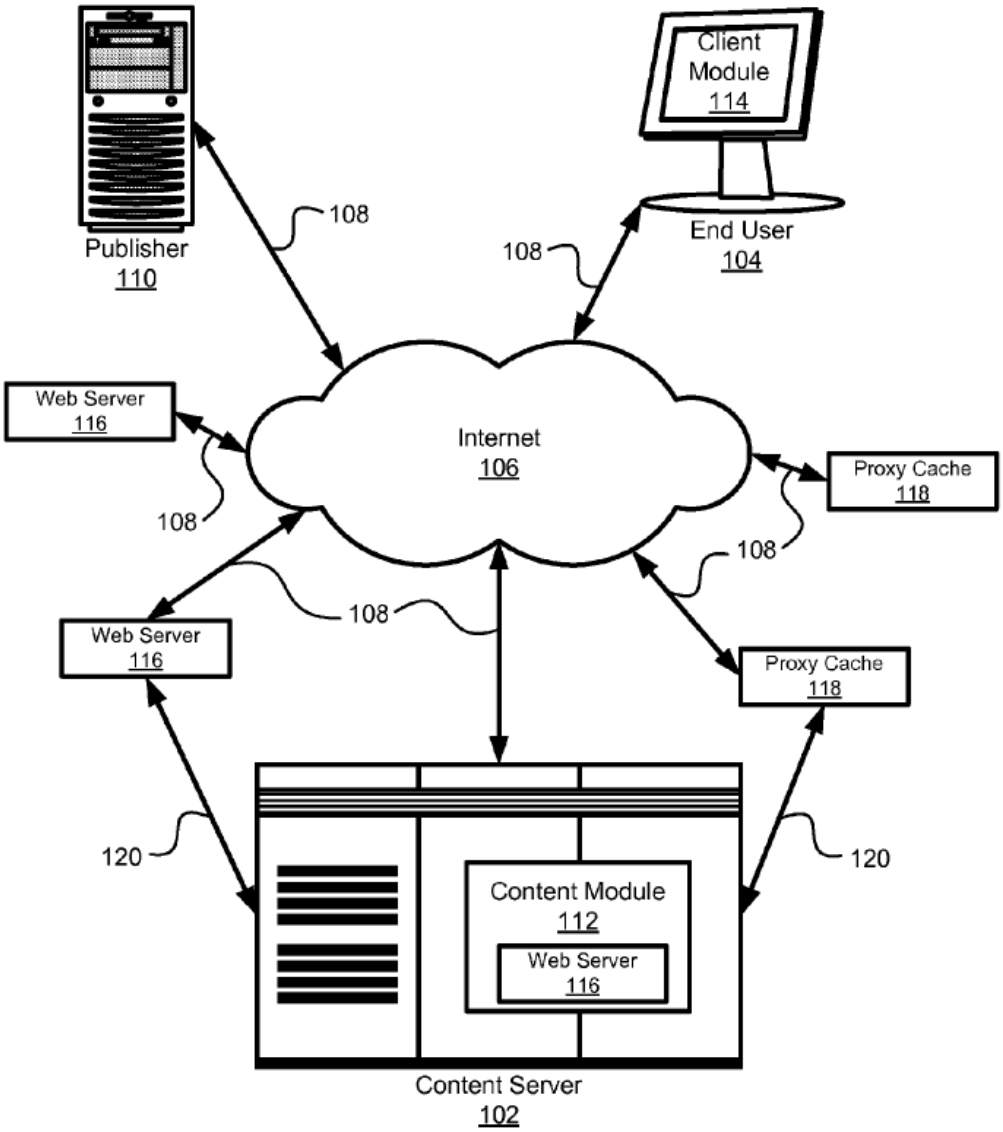
Id., Fig. 1.

128. Brueck discloses that content from a publisher 110 "may be transferred over the Internet 106 to the content server 102, where the content is received by a content module 112." Ex-1005 at 6:20-22. Brueck's content module 112 can receive, process, and store live (or archived) video and/or audio from the publisher 110 and transmit the processed content to an end user via a web server

116. *Id*. at 6:22-30, 7:40-8:55, Fig. 4.

b. "receiving, at the video message service infrastructure, a video message transmitted by a sending device, the video message containing video media, the video media being transmitted as the video media is created;" (1[a])

129. In my opinion, Brueck also discloses this limitation. As shown in Figure 1, Brueck's content server receives content transmitted over the Internet from a publisher 110. Ex-1005, Fig. 1; *see also id*. at Fig. 4, 7:40-45. In my opinion, a person of ordinary skill in the art would have recognized that a sending device is required to be present at a publisher to send data from the publisher, as is illustrated in Brueck's Fig. 1, which shows a computer labelled "Publisher":



Brueck, Fig. 1.

130. Brueck explains that the content sent by the publisher 110 may include video media. Ex-1005 at 6:16-17, 6:40-46. In particular, Brueck discloses that the publisher 110 transmits a content file 200, which may be "live or archived content" comprising "video and audio, or alternatively, video or audio." *Id*. at 6:40-49. Brueck explains that the content may include, among other things, media

from a live sporting event or concert. *Id*. at 6:42-46, 11:22-31.

131. Media transmitted by the publisher 110 is received by a content module 112 of Brueck's content server 102. Ex-1005 at 6:20-22; *see also id*. at 8:8-12. The content server is configured to deliver video media received from the publisher to end user stations, as shown in Figure 1. *Id*. at 5:64-6:8. Brueck explains that "[t]he end user station 104 may comprise a personal computer (PC), an entertainment system configured to communicate over a network, or a portable electronic device configured to present content." *Id*. at 6:8-14.

132. In my opinion, Brueck also discloses that the video media is transmitted as it is created. As noted, Brueck discloses that publisher 110 may be a "***creator*** or distributor of content." Ex-1005 at 6:16-17 (emphasis added). Brueck also discloses that the content transmitted by the publisher 110 may be ***live*** video. *Id*. at 6:45-46 ("The content file 200 may also be live or archived content."), 9:61-10:13, 10:44-60, 11:23-33, 12:1-9. Brueck further teaches that its content server "is capable of streaming the streamlets over the Internet shortly after the content module 112 begins capture of the content file 200." *Id*. at 10:8-10.

133. In my opinion, a person of ordinary skill in the art would further understand Brueck to disclose that when transmitting live video, Brueck's publisher 110 would transmit video messages to Brueck's content server "as the video media is created." Indeed, one of the disclosed benefits of Brueck's

invention "is the ability to perform two-pass encoding on a live content file 200." Ex-1005 at 9:66-67. Whereas prior art systems required a live event to have finished before two-pass encoding could be performed (*id*. at 9:67-10:3, 10:44-52), with Brueck's system, "multi-pass encoding may begin on a live event once the first streamlet is captured." *Id*. at 10:44-56. Performed in this way, "multi-pass encoding is performed on a live event and appears live to a viewer." *Id*. at 10:58-60.

c. "generating, by the video message service infrastructure, two or more degraded versions of the video media of the video message, where each of the two or more degraded versions is created by degrading the quality of the video media by a different amount;" (1[b])

134. In my opinion, Brueck also discloses this limitation. As I explain in my analysis above, Brueck's content server 102 includes a content module 112 configured to receive and process content transmitted from the publisher 110. *See supra*, ¶¶127-131.

135. Brueck teaches that "[t]he original content file 200 from the publisher may be digital in form and may comprise content having a high bit rate such as, for example, 2 mbps." Ex-1005 at 8:8-10. Once received at the content server (*id*. at 7:42-47), the content file is passed to a streamlet module 404, which "segment[s] the content file 200 into streamlets 303 each having an equal duration." *Id*. at 7:62-8:2. An encoder module 406 within Brueck's content module 112 then

generates a plurality of streams comprising sets of streamlets having varying qualities (*id*. at 8:6-8, 8:16-25), as shown in Figure 4 below:

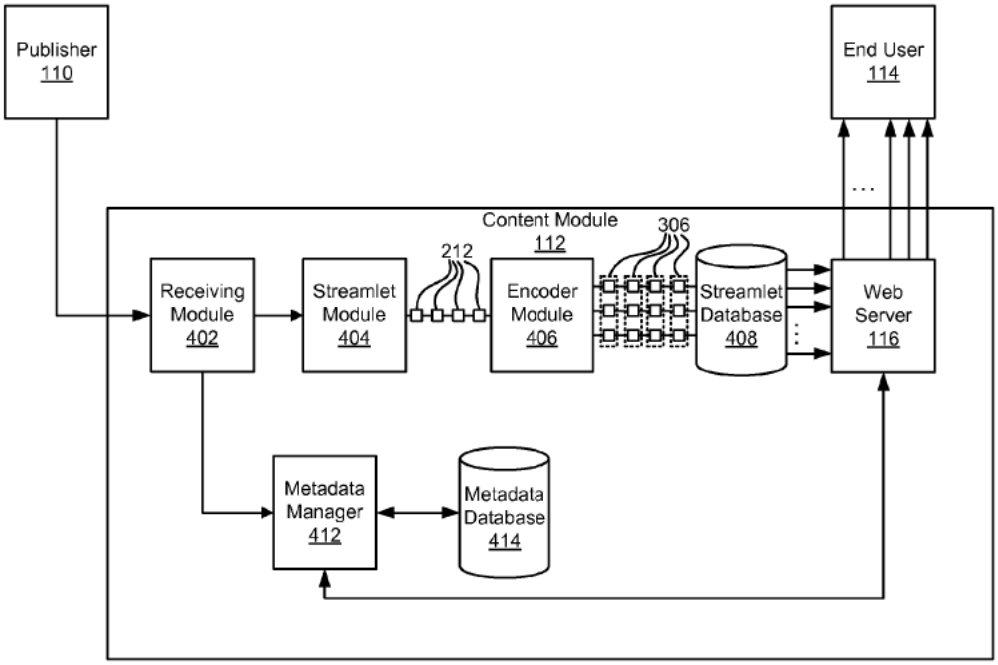


FIG. 4

Id., Fig. 4.

136. The output of the encoder module is also depicted in Figure 2, which shows "a plurality of streams 202 having varying degrees of quality and bandwidth":

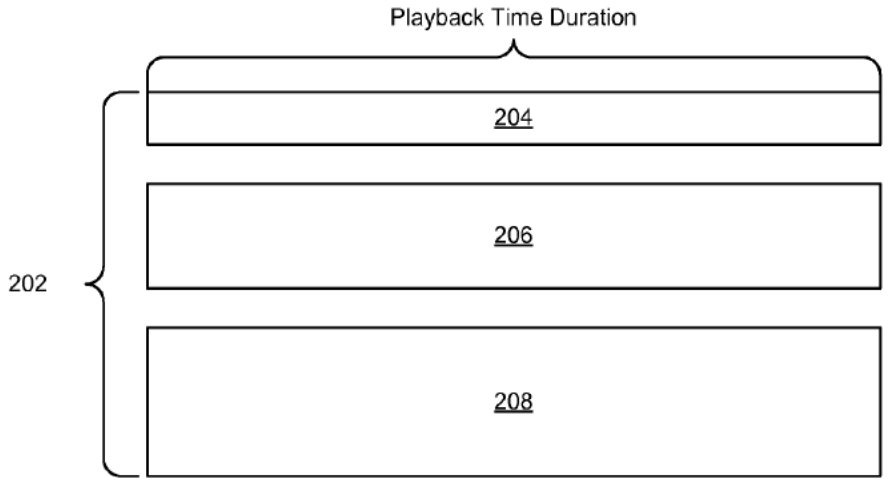


FIG. 2B

Ex-1005 at 4:26-28, Fig. 2B; *see also id*. 6:54-56.

137. Brueck explains that "[e]ach of the streams 204, 206, 208 is a copy of the content file 200 encoded and compressed to varying bit rates." Ex-1005 at 6:59-60. "For example, the low quality stream 204 may be encoded and compressed to a bit rate of 100 kilobits per second (kbps), the medium quality stream 206 may be encoded and compressed to a bit rate of 200 kbps, and the high quality stream 208 may be encoded and compressed to 600 kbps." *Id*. at 6:60-65. Although in the example of Figure 2B only three streams are represented, Brueck explains that "the plurality of streams 202 may comprise any number of streams deemed necessary to accommodate end user bandwidth." *Id*. at 8:23-25.

138. In Figure 3B, Brueck shows sets 306 of encoded streamlets 304 having low, medium, and high 204, 206, 208 bitrates:

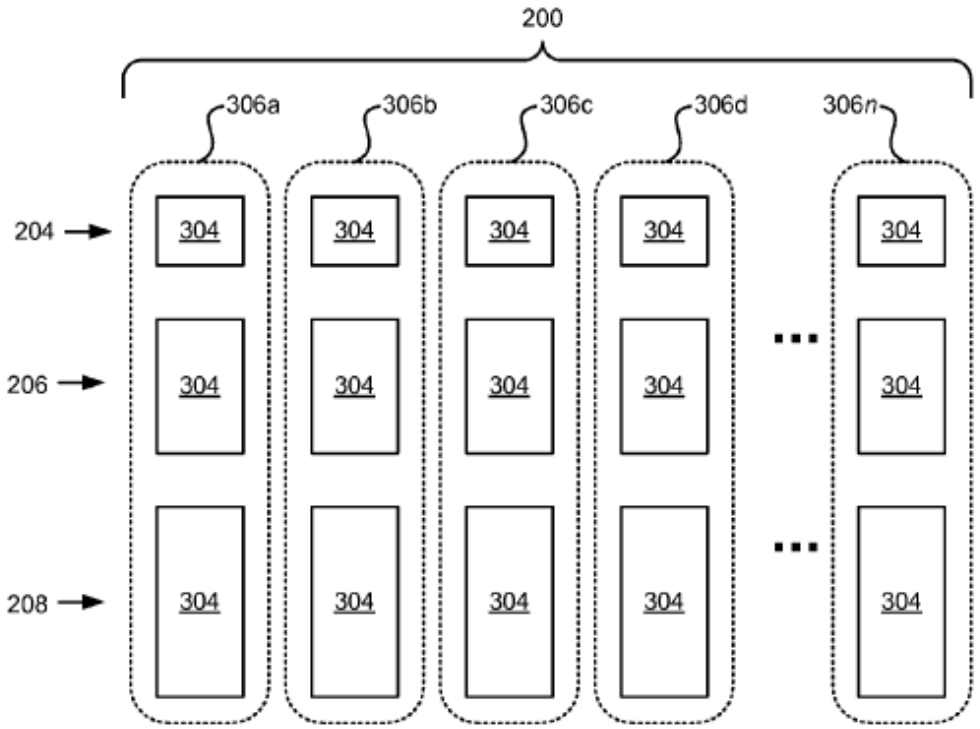


FIG. 3B

Ex-1005, Fig. 3B, 7:16-23. Each "set" represents "a group of streamlets having identical time indices and durations but varying bitrates." *Id*. at 7:18-20. Although in the example of Figure 3B, the set 306 includes encoded streamlets having three different bitrates, Brueck explains that "each set 306 may include more than the depicted three bitrates which are given by way of example only." *Id*. at 7:23-25. Indeed, Brueck correctly teaches that "[o]ne skilled in the art will recognize that any number of streams having different bitrates may be generated from the original content 200." *Id*. at 7:23-27.

139. In my opinion, a person of ordinary skill in the art would have recognized that the streams generated by Brueck's encoder module are degraded

versions of the video media received from the publisher. For instance, whereas the "original content file 200 from the publisher" could "hav[e] a high bit rate such as, for example, 2 mbps" (Ex-1005 at 8:8-10), Brueck discloses that each of the exemplary streams (204, 206, and 208) depicted in Figures 2B and 3B is compressed to a lower bitrate. Indeed, Brueck discloses that even the "high quality stream" (208) may be encoded and compressed to 600 kbps—a bitrate significantly lower than the "high bit rate" (2 mbps) that could be received from the publisher.

140. Given these disclosures, in my opinion, one of ordinary skill in the art would have readily understood that the encoder module of Brueck's content server—part of Brueck's "video message service infrastructure," as explained above—generates multiple degraded versions of the original content file and that each of those degraded versions is created by degrading the quality of the video media by a different amount. Ex-1005 at 6:54-65; 7:16-27, 8:16-25. Indeed, given Brueck's disclosure that "the plurality of streams 202 may comprise any number of streams deemed necessary to accommodate end user bandwidth," a skilled artisan would have recognized that any number of streams could be created by degrading the quality of the media by different amounts, and that creating such degraded versions would advance Brueck's goal of "accommodat[ing] end user bandwidth." *Id*. at 8:23-25.

d. "selecting one of the degraded versions of the video media based on an ascertained bandwidth on the

network for delivering the video media from the video message service infrastructure to a recipient device on the network; and" (1[c])

141. In my opinion, Brueck discloses this limitation as well. In Brueck's system, this limitation is performed by the client module 114 of the end user station 104 in communication with a web server 116. A schematic illustration of a client module 114 is shown in Figure 7:

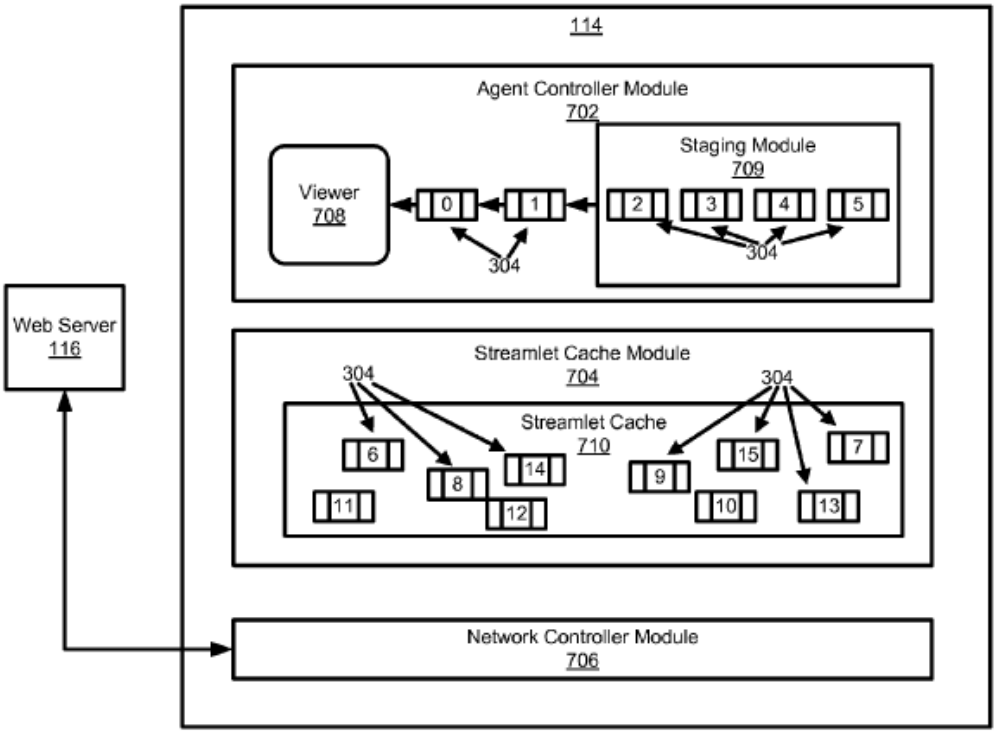


FIG. 7

Ex-1005, Fig. 7.

142. Brueck explains that the content module 114 is configured to access content processed by the client module 114 of Brueck's content server and to play the content on an end user station 104. Ex-1005 at 6:22-26. Brueck explains that

the end user station 104 may comprise, among other things, "a personal computer (PC), an entertainment system configured to communicate over a network, or a portable electronic device configured to present content." *Id*. at 6:8-14. Accordingly, in my opinion, the end user station is a "recipient device on the network."

143. Brueck discloses that the client module 114 may receive streaming media transmitted over the Internet by a web server 116. Ex-1005 at 6:24-30, 8:53-9:1, 12:38-42, 17:15-18; Figs. 1, 4, 7, 9. "[S]ince the client module 114 initiates the request, the web server 116 is only required to retrieve and serve the requested streamlet 304." *Id*. at 8:58-61.

144. Brueck also discloses selecting a degraded version of the video media based on an ascertained bandwidth for delivering the video media from the web server 116 to the end user station. For example, Brueck teaches that the agent controller module 702 of client module 114 is "configured to ***select*** a quality level of streamlets to transmit to the viewer 708." Ex-1005 at 12:38-39 (emphasis added); *see also* 16:11-17:21, Figs. 7, 10.

145. Brueck further explains that "[t]he agent controller module 702 requests lower or higher quality streams ***based upon continuous observation of time intervals between successive receive times of each requested streamlet***." Ex-1005 at 12:39-42 (emphasis added). In my opinion, a person of ordinary skill in

the art would have recognized that the observed time intervals between successive receive times of the requested streamlets represents an indication of whether or not the requested streamlets are being received in real-time (*e.g.,* whether or not streamlets representing *x* seconds of video are being received in more or less time than one every *x* seconds). In particular, a person of ordinary skill in the art would have understood these observations to yield an approximation of the effective bandwidth on the network between the server and the recipient device, and that by continuously observing such intervals, the agent controller module is ascertaining the bandwidth on the network available for delivering the streamlets (the video media) to the recipient device in real-time.

146. Moreover, as I explain in my analysis above, each of the encoded streams available to be requested from the web server is a degraded version of the video media. *See supra*, ¶¶139-40. Thus, in my opinion, a person of ordinary skill in the art would have understood that Brueck's agent controller module ***selects*** one of those degraded versions based on ascertained bandwidth on the network from the web server to the end user station.

147. For instance, if network conditions result in less bandwidth being available between Brueck's web server and the end user station, a lower quality video version would be selected for delivery to the end user. *See* Ex-1005 at 15:57-61 ("Upon retrieving 910 or receiving a streamlet, the agent controller

module 702 makes 911 a determination of whether or not to shift to a higher or lower quality stream 202. This determination will be described below in greater detail with reference to FIG. 10."), 17:8-18; *see also* Figs. 9, 10.

148. With reference to Figure 10, reproduced below, Brueck discloses a particular method for shifting between streams having different quality levels:

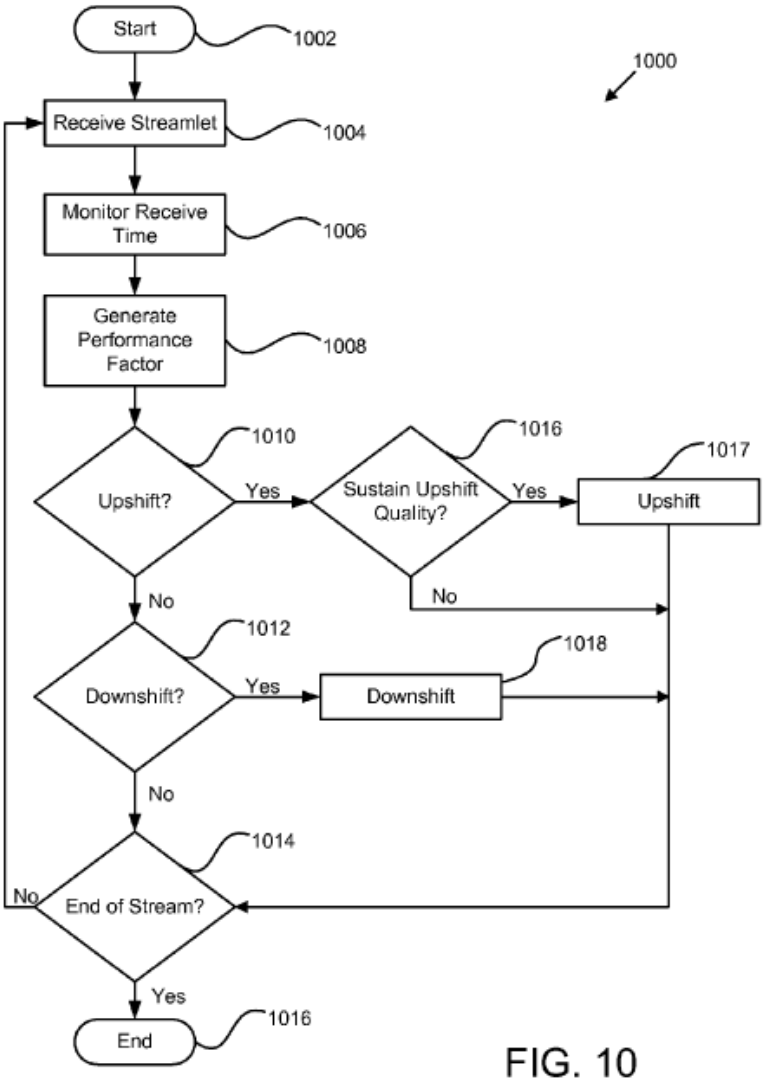


FIG. 10

Ex-1005, Fig. 10.

149. Brueck explains that after requesting an initial streamlet, the agent

controller module monitors the receive time for the requested streamlet, and may monitor the time intervals between successive receive times for each streamlet response. Ex-1005 at 16:16-22. To compensate for fluctuations in the available bandwidth, the agent controller module calculates a performance ratio across a window of samples to determine if media is being received in real-time or not, then applies an algorithm to determine whether to shift to a higher or lower quality stream. Brueck at 16:25-17:21. The performance ratio reflects ascertained effective bandwidth on the network for the media being transmitted over the particular window of samples.

150. For instance, if current bandwidth conditions are insufficient to support the current streaming quality, the agent controller module downshifts the streaming quality. Ex-1005 at 17:8-18; *see also id*. at 15:57-61, 16:25-49, Figs. 9, 10. In my opinion, one of ordinary skill in the art readily would have understood that shifting streams, especially to a lower-quality version, would involve selecting one of the degraded versions based on an ascertained bandwidth on the network for delivering the video media to a recipient device (user station 114).

e. "transmitting the selected degraded version of the video media from the video message service infrastructure to the recipient device so that the recipient device can render at least a portion of the video media on the recipient device;" (1[d])

151. In my opinion, Brueck discloses this limitation. When Brueck's agent

controller module 702 selects an appropriate degraded version of the video media, a request is made to the web server 116. Ex-1005 at 8:53-55 ("Once the streamlets 304 have been received and processed, the client module 114 may request streamlets 304 using HTTP from the web server 116."), 12:39-44 ("The agent controller module 702 is configured to select a quality level of streamlets to transmit to the viewer 708."), 16:50-17:18; Figs. 4, 7.

152. In response, the web server 116 transmits the selected degraded version to the end user station. Ex-1005 at 17:15-18 ("the agent controller module 702 begins to request and receive 1004 lower quality streamlets"). "[S]ince the client module 114 initiates the request, the web server 116 is only required to retrieve and serve the requested streamlet 304." *Id*. at 8:58-61, 15:39-43 ("serving 814 the streamlets 304 comprises receiving streamlet requests from the client module 114, retrieving the requested streamlet 304 from the streamlet database 408, and subsequently transmitting the streamlet 304 to the client module 114"); *see also id*. at 15:45-16:10, 16:11-22, 16:50-17:21, Figs. 4, 7.

153. As the end user station receives the media stream, a viewer 708 interfacing with the agent controller module may render it. Brueck explains, for instance, that "the agent controller module 702 is configured to interface with a viewer 708, and transmit streamlets 304 to the viewer 708." Ex-1005 at 12:20-27, 12:31-36. Brueck further teaches that "[t]he viewer 708 may be a media player

(not shown [in Fig. 7]) operating on a PC or handheld electronic device." *Id*. at 12:36-37. A person of ordinary skill in the art would have understood that media players are designed to render or "play" media (hence the word "player" in their name). Indeed, Brueck confirms that "delivering 914 streamlets 304 to the end user comprises ***playing video and or audio streamlets on the viewer 708***." *Id*. at 15:65-67 (emphasis added), 6:23-25 (explaining that "a client module 114 [is] configured to play the content on the end user station 104"), 12:45-47 ("[A]gent controller module 702 may be configured to receive user commands from the viewer 708. Such commands may include play, fast forward, rewind, pause, and stop.").

154. Accordingly, in my opinion, Brueck discloses transmitting the selected degraded version of the video media from the video message service infrastructure to the recipient device so that the recipient device can render at least a portion of the video media on the recipient device.

f. "the video message service infrastructure operating so that the rendering occurs: (i) while the video media of the video message is being created and transmitted over the network by the sending device; and" (1[e])

155. In my opinion, Brueck discloses this limitation. Brueck's content server, which as discussed above is part of a video message service infrastructure, operates such that rendering on the end user station may occur in near real-time as the video message is being created and transmitted by the publisher 110.

156. As I explain in my analysis above, Brueck teaches that the publisher 110 may be a creator of content (Ex-1005, 6:15-26) that transmits live video to Brueck's content server. *See supra*, ¶¶129-32; *see also* Ex-1005 at 6:40-53, 9:66-10:3, 10:11-13, 10:44-60, 11:22-33.

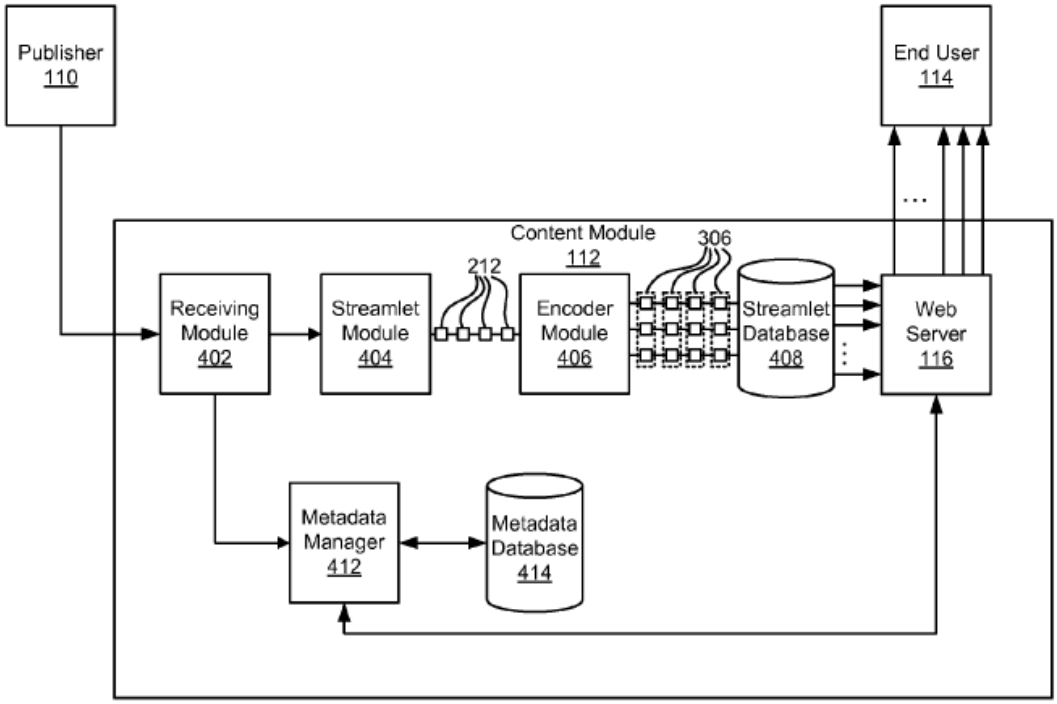
157. Brueck further teaches that its content server "is capable of streaming the streamlets over the Internet shortly after the content module 112 begins capture of the content file 200." Brueck at 10:8-10. In this way, "the content file 200 is encoded for streaming over the Internet and appears live." Ex-1005 at 10:35-37; *see also* 10:40-43 ("The delay may be shortened to an almost unperceivable level by the addition of high CPU powered systems, or alternatively multiple low powered systems."), 10:56-60 ("Shortly after multi-pass encoding of the first streamlet 303-*a* is finished, multi-pass encoding of the second streamlet 303-*b* finishes, and as such multi-pass encoding is performed on a live event and appears live to a viewer."), 9:61-10:60, 11:13-33.

158. Thus, in my opinion, Brueck discloses that its video message service infrastructure operates so that the rendering occurs while the video media of the video message is being created and transmitted over the network by the sending device.

g. "(ii) without having to first establish an end-to-end connection over the network between the sending device and the recipient device before the sending device begins to transmit the video media of the video

message." (1[f])

159. In my opinion, Brueck also discloses this limitation. Brueck discloses that a publisher 110 transmits video media to the content server (part of Brueck's video message service infrastructure), which then receives, processes and stores the video media, as shown in Figure 4:



Ex-1005, Fig. 4; *see also* 6:20-24, 6:54-65, 7:40-53, 7:62-8:32, 15:15-21, Fig. 1.

160. Brueck explains that video media is transmitted from the content server to the end user in response to a request from the client module of the end user station. Ex-1005 at 6:24-26 ("processed content is accessed by a client module 114 configured to play the content on the end user station"), 8:53-61 ("Once the streamlets 304 have been received and processed, the client module 114

may request streamlets 304 using HTTP from the web server 116. . . . [S]ince the client module 114 initiates the request, the web server 116 is only required to retrieve and serve the requested streamlet."), 12:20-44, *see also* Figs. 4, 7.

161. Because in Brueck's system, an intermediate content server receives, processes, and stores the video media received from the publisher, and separately responds to end user requests for media, no "end-to-end connection" (or any other connection) need ever be established between the publisher and the end user station before the sending device begins to transmit the video media of the video message. Indeed, Brueck contains no disclosure or suggestion of any such connection existing or being required. Accordingly, in my opinion, no "end-to-end connection" need be established in Brueck between the publisher and the end user station before the publisher begins to transmit the video media.

2. Claim 2: "The method of claim 1, wherein at least one of the two or more degraded versions of the video media of the video message is generated by reducing a bit rate used to represent the video media relative to a complete bit rate representation of the video media as received by the video message service infrastructure."

162. In my opinion, this limitation is satisfied for the reasons explained above in my analysis of claim element 1[b]. *See supra*, ¶¶129-33.

163. The phrase "complete bit rate representation" is not used outside of claim 2 and is not a term of art. However, a person of ordinary skill in the art would understand the term to refer to the original version of the video media, as

received from the sending device, before any subsequent processing to reduce the bit rate. As discussed above, the capture module 402 of Brueck's content module 112 (part of Brueck's content server) receives a "content file 200" from the publisher 110. Ex-1005 at 7:40-48, Fig. 4. Brueck teaches that this "original content file 200 . . . may comprise content having a high bit rate such as, for example, 2 mbps." *Id*. at 8:8-10. A person of ordinary skill in the art would have understood that the original content file received by the capture module, before any processing by the content module, represents a complete bit rate representation of the video media as received by Brueck's content server.

164. As I explain above, Brueck's encoder module 406 generates a plurality of streams each reflecting the original content received from the publisher compressed to bit rates of varying quality. *See supra*, ¶¶135-39. Brueck explains that "[e]ach of the streams 204, 206, 208 [shown in Figure 2B] is a copy of the content file 200 encoded and compressed to varying bit rates." Ex-1005 at 6:59-60. For example, a "low quality stream 204 may be encoded and compressed to a bit rate of 100 kilobits per second (kbps), [a] medium quality stream 206 may be encoded and compressed to a bit rate of 200 kbps, and [a] high quality stream 208 may be encoded and compressed to 600 kbps." *Id*. at 6:60-65, 8:23-25 ("the plurality of streams 202 may comprise any number of streams deemed necessary to accommodate end user bandwidth"); *see also* 7:16-39, Figs. 2B, 3B.

165. As explained above, a person of ordinary skill in the art would have understood that Brueck's encoder module 406 generates degraded versions of the original content file received from the publisher by reducing a bit rate used to represent the video media relative to the original content file. *See supra*, ¶¶135-40. For instance, as I explain above, a person of ordinary skill in the art would have understood that each of the three streams (204, 206, and 208) depicted in Figures 2B and 3B is a degraded version of the original content file 202. *See supra*, ¶139. Indeed, Brueck explains that any number of degraded streams may be created "to accommodate end user bandwidth." Ex-1005 at 8:20-25.

166. Accordingly, it is my opinion that Brueck discloses that at least one of the two or more degraded versions of the video media of the video message is generated by reducing a bit rate used to represent the video media relative to a complete bit rate representation of the video media as received by the video message service infrastructure.

3. Claim 3: "The method of claim 1, further comprising transmitting one or more of the two or more degraded versions to two or more additional recipient devices, wherein the transmitted one or more degraded versions are selected at least partially based on being commensurate with ascertained bandwidths on the network along delivery paths for delivering the video message to the two or more additional recipient devices respectively."

167. Claim 3 adds two limitations to the method of claim 1. First, claim 3 requires transmitting at least one of the degraded versions to at least two additional

recipient devices. Second, claim 3 requires selecting the degraded version(s) for transmission based on bandwidth availability along the delivery path to each of the additional recipient devices. In my opinion, each limitation is disclosed and/or rendered obvious by Brueck.

168. Regarding the first limitation, Brueck discloses a system that may be used, for instance, to transmit television programming, live sporting events, and concerts (among other things) to end users. Ex-1005 at 6:43-46; 6:16-20 ("the publisher 110 may be a television or cable network channel such as NBC®, or MTV®"). In my opinion, a person of ordinary skill in the art would have understood that Brueck's system is intended to stream such broadcast media to multiple recipient devices. Indeed, Brueck's claims consistently recite "end user station**s**" (plural). *E.g.*, *Id.*, cl. 1, 4, 12, 14, 24, 31. For instance, claim 1 requires, among other things, a web server "coupled over the Internet to the end user station**s** to receive requests from the end user station**s** for different ones of the separate content files from different ones of the sets and to transmit the requested ones of the separate content files to the requesting one of the end user station**s** . . . ." *Id.*, cl. 1 (emphasis added).

169. To the extent Brueck does not specify the number of end user stations that may be served, in my opinion, one of ordinary skill in the art would have understood that one or more degraded version of the video media could be

transmitted to "at least two or more additional recipient devices."

170. Regarding the second limitation, as I explain in my analysis above, in Brueck's agent controller module 702 selects lower or higher quality streams based on its observation of bandwidth available between the web server and an end user station. *See supra*, ¶¶141-50**;** Ex-1005 at 12:38-44, 16:11-17:21. As I discuss above, if the available bandwidth is not sufficient to sustain real-time transmission of video, the agent controller module would shift to a lower-quality stream. *See supra*, ¶¶148-50**;** Brueck at 17:8-21.

171. A person of ordinary skill in the art would have recognized that, in the case of multiple recipient devices, an agent controller module running on each end user station would likewise select a stream based on being commensurate with available bandwidth between the web server and that particular end user station. Ex-1005 at 2:37-46 ("Beneficially, such an apparatus, system, and method would utilize multiple connections between a source and destination, requesting varying bitrate streams depending upon network conditions."), 8:16-25, 12:38-44, 16:11-17:21; Figs. 7, 9, 10.

172. Thus, in my opinion, Brueck discloses or renders obvious the limitation of transmitting one or more of the two or more degraded versions to two or more additional recipient devices, wherein the transmitted one or more degraded versions are selected at least partially based on being commensurate with

ascertained bandwidths on the network along delivery paths for delivering the video message to the two or more additional recipient devices respectively.

4. Claim 8: "The method of claim 1, wherein the video message is received at the video message service infrastructure using a first real-time streaming protocol, and the selected degraded version of the video media is transmitted from the video message service infrastructure to the recipient device using a second real-time streaming protocol."

173. In my opinion, Brueck renders this claim limitation obvious in view of the knowledge of a person of ordinary skill in the art.

174. The '557 patent contains no disclosures of any protocol that is termed a real-time streaming protocol. The patent discloses the use of the standard Internet transport protocols TCP and UDP but as a person of ordinary skill in the art would understand, these are generic protocols with no special features for real-time communications or streaming. The '557 patent does contain a description of an application-layer protocol termed "Vox"; however, this protocol is disclosed as using (being layered on top of) generic transport protocols such as UDP and TCP. As such, a person of ordinary skill in the art would understand that the Vox protocol has no special abilities for real-time communication other than that which TCP or UDP (or whatever lower layer protocol is used) can provide.

175. Based on these disclosures, as well as the disclosures of the use of TCP/IP and the Internet, as explained above, a person of ordinary skill in the art

would understand the ’557 patent’s use of the term “real-time streaming protocol” to simply refer to an application layer protocol for carrying real-time media such as live video. *See supra,* ¶¶128-29, 132, 143, 157; *see also* Ex-1005 at 13:26-29, 13:48-50, 14:10-12.

176. Brueck discloses using the HTTP protocol to stream the selected degraded version of the video media in real-time from the content server to the end user station. Ex-1005 at 8:53-63, 13:23-37. As noted, Brueck’s content server comprises part of its video message service infrastructure.

177. Although Brueck does not expressly disclose the particular real-time streaming protocol it uses for streaming the video message from the publisher to the content server, one of ordinary skill in the art would have understood that some real-time streaming protocol would be used because Brueck is intimately concerned with latency and its strategy for dynamically adapting the bitrate of video media streamed to the end user relies on tracking the time that the video media (streamlets) are received. *See* Ex-1005 at 2:28-30, 2:38-39, 10:44-60, 12:38-42, 16:11-38, cl. 13. A person of ordinary skill in the art also would have recognized that commonly used real-time streaming protocols such as RTP and RTSP were available for ready use and could be used to deliver media in Brueck. Such use would require no experimentation and would simply be the use of a well-known and standards-based technology for its intended purpose.

178. Accordingly, it is my opinion that Brueck renders obvious the requirement in claim 8 of receiving the video message at the video message service infrastructure using a first real-time streaming protocol, and transmitting the selected degraded version of the video media to the recipient device using a second real-time streaming protocol.

5. Claim 9: "The method of claim 8, wherein the first real-time streaming protocol is different from the second real-time streaming protocol."

179. Claim 9 depends from claim 8, adding the requirement that the first real-time streaming protocol be different from the second real-time streaming protocol. For the reasons explained above with respect to claim 8, Brueck renders this claim obvious in view of the knowledge of a person of ordinary skill in the art. *See supra*, ¶¶173-78. As explained above, Brueck discloses using the HTTP protocol to stream live video media from the content server to end user stations. It would have been obvious to a person of ordinary skill in the art to use a different real-time streaming protocol, such as RTP or RTSP, to stream the video message from the content publisher to Brueck's content server. The use of different protocols would simply be a design choice from among a limited set of possible choices (the set of real-time streaming protocols).

6. Claim 14:

a. "ascertaining a first codec that was used by the sending device to encode the video media of the video message

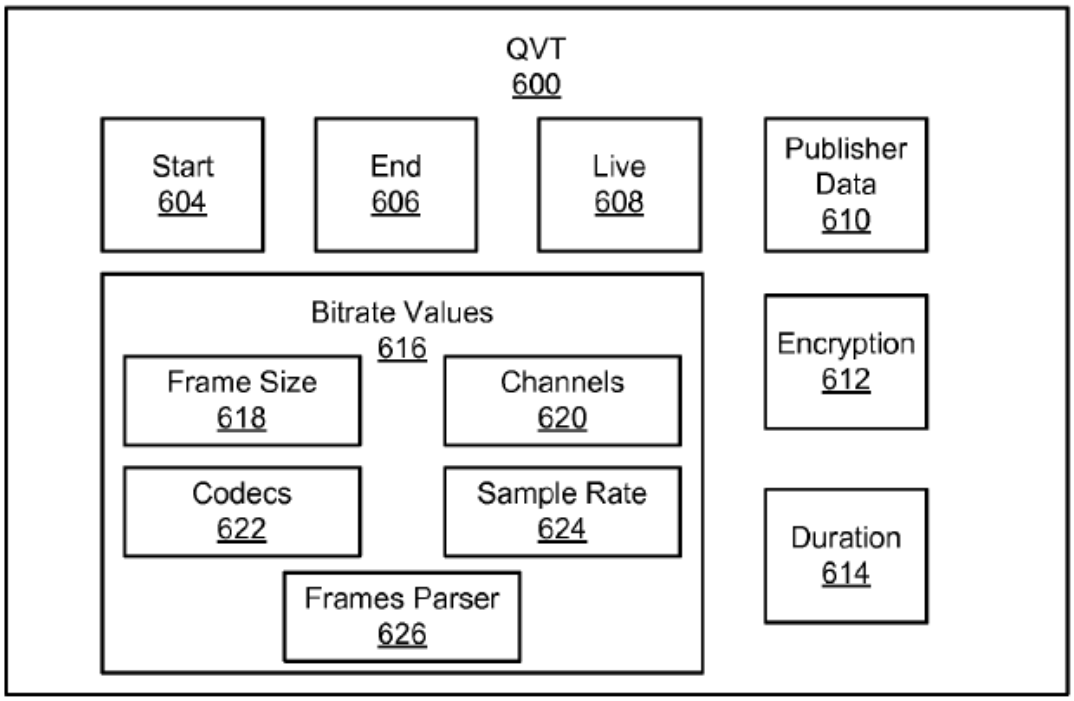
when sent;" (14[a])

180. In my opinion, Brueck discloses this limitation. Brueck explains that the content file 200 received from the publisher "may be compressed using standard or proprietary encoding schemes." Ex-1005 at 6:47-49. "Examples of encoding schemes capable of use with the present invention include, but are not limited to, DivX®, Windows Media Video®, Quicktime Sorenson 3®, On2, OGG Vorbis, MP3, or Quicktime 6.5/MPEG-4® encoded content." *Id*. at 6:49-53. A person of ordinary skill in the art would have understood that a first codec would be used by the sending device to compress the media for transmission to Brueck's content server.

181. Brueck further explains that the capture module 402 of the content server "may be configured to 'decompress' the content file 200." Ex-1005 at 7:45-48. "For example, if the content file 200 arrives having been encoded with one of the above described encoding schemes, the capture module 402 may convert the content file 200 into raw audio and/or video." *Id*. at 7:48-51.

182. In my opinion, a person of ordinary skill in the art would have understood that the process of receiving and decompressing a content file, as disclosed in Brueck, would typically involve ascertaining the codec that was used by the sending device to encode the video media. Indeed, Brueck describes recording the ascertained codec in a "quantum media extension" (QMX) file. Ex-

1005 at 11:52-54, 12:1-11. With reference to Figure 6C, for example, Brueck explains that the QMX may contain a multitude of information describing the content file 200, including (among other things) the "***codecs 622 used***." *Id*. at 12:3-11 (emphasis added); Fig. 6C:



Id., Fig. 6C, 11:49-60, 12:1-11. Thus, in my opinion, Brueck discloses ascertaining a first codec that was used by the sending device to encode the video media of the video message when sent.

b. "ascertaining a second codec to be used by the recipient device to decode the video media of the video message when received;" (14[b])

183. In my opinion, Brueck also discloses this limitation. Brueck discloses, for instance, that the encoding module 406 of Brueck's content server

may be configured to "produce streamlets that are specific to a particular playback platform." Ex-1005 at 11:39-42. For example, the encoding module "may produce streamlets for different quality levels for personal computer playback" or "for playback on cell phones with a different, proprietary codec." *Id*. at 11:42-48; *see also id*. at 11:35-36 (the encoding scheme may comprise "a proprietary codec such as WMV9®").

184. In order to transmit video media for playback on "a particular playback platform," as disclosed in Brueck, the system would need to understand which codec would be used by the recipient device to decode the video. Accordingly, it is my opinion that Brueck discloses ascertaining a second codec to be used by the recipient device to decode the video media of the video message when received.

c. "trans-coding the video media of the video message from the first codec to the second codec; and" (14[c])

185. In my opinion, Brueck discloses this limitation when it describes decompressing video media encoded in a first format and re-encoding the video media into a different format suitable for decoding at the recipient device. *See supra*, ¶¶181-84; Ex-1005 at 6:47-53, 7:45-48, 11:35-48.

d. "transmitting, to the recipient device, the trans-coded video media." (14[d])

186. In my opinion, Brueck discloses transmitting the trans-coded video

media to the recipient device.

187. Brueck discloses transmitting video media to the recipient device. Ex-1005 at 6:22-26, 8:53-55. As discussed above, Brueck also discloses transcoding video media received from the publisher so that it may be played back on a particular playback platform. *Id*. at 11:39-48, 6:49-53, 7:45-51.

188. In view of these disclosures, a person of ordinary skill in the art would have understood that the trans-coded video media would be transmitted to a recipient device having the relevant playback platform. Indeed, a person of ordinary skill in the art would have understood that the reason to transcode the video media, as described in Brueck, would have been to transmit the trans-coded media to a recipient device.

7. Claim 18: "The method of claim 1, further comprising providing an application for execution on the recipient device, the application enabling a recipient using the recipient device to selectively render the video media of the video message: at least partially as the video media of the video message is created and transmitted by the sending device and delivered by the video message service infrastructure; and in a time-shifted mode by rendering the video media of the video message out of storage."

189. In my opinion, Brueck discloses or renders obvious this limitation in view of the knowledge of a person of ordinary skill in the art.

190. First, it is my opinion that Brueck discloses an "application for execution on the recipient device." In particular, Brueck discloses that content

processed by the content server is "accessed by a ***client module 114***" which is "configured to play the content on the end user station 104." Ex-1005 at 6:24-26 (emphasis added), 6:26-30; *see also* Fig. 1 (showing client module 114 on end user station 104), 6:24-26. As shown in Figure 7, the client module 114 comprises a software application having various submodules:

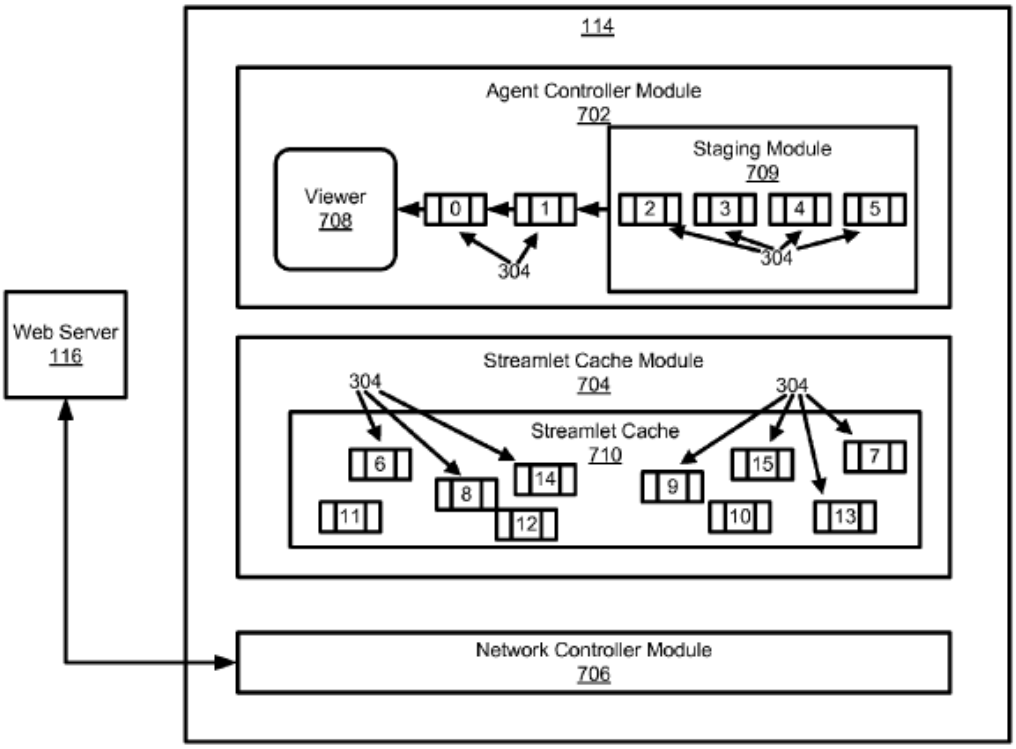


FIG. 7

Id., Fig. 7; *see also* 5:13-14 ("Modules may . . . be implemented in software for execution by various types of processors."), 5:14-33, 12:20-13:5. Accordingly, a person of ordinary skill in the art would have recognized that Brueck's client module 114 is an application configured to be executed on the recipient device (end user station 104).

191. To the extent Brueck does not expressly disclose ***providing*** that application for execution on the recipient device, a person of ordinary skill in the art would have understood that the disclosed client module could be provided to a device user, and that providing such a software application would advantageously allow the end user to enjoy the benefits of Brueck's invention. For instance, as explained above, the client module is responsible for monitoring available bandwidth and shifting between higher and lower quality versions of the video media. *See supra*, ¶¶141-50. It would have been obvious to provide an application providing this functionality because it is central to Brueck's adaptive rate-shifting technology.

192. Moreover, as I explain in my analysis of claim elements 1[d] and [e], the client module enables the end user using the end user station to render the video media in near real-time as it is created and transmitted by the publisher and delivered by the content server. *See supra*, ¶¶151-58**.** For example, Brueck explains that "[t]he agent controller module 702 requests lower or higher quality streams based upon continuous observation of time intervals between successive receive times of each requested streamlet." Ex-1005 at 12:39-42; *see also* 8:53-63 ("Once the streamlets 304 have been received and processed, the client module 114 may request streamlets 304 using HTTP from the web server 116.").

193. In response to such requests, the web server 116 transmits the selected

stream for rendering on the end user station. Ex-1005 at 8:58-63, 15:35-44, 16:11-22, 16:50-17:21; Figs. 4, 7, 9, 10. Brueck explains, for instance, that "the agent controller module 702 [of the client module 114] is configured to interface with a viewer 708" (*id*. at 12:25-27), which may then render the video in near real-time as it is being created by the publisher and received from the content server. Brueck at 6:24-26 ("processed content is accessed by a client module 114 configured to play the content on the end user station 104"), 10:30-43 ("the content file 200 is encoded for streaming over the Internet and appears live"), 10:56-60 ("multi-pass encoding is performed on a live event and appears live to a viewer"); *see also id.* 10:11-13, 10:61-11:21, 12:20-30, 12:31-37, 12:45-47; Figs. 7, 9, 10. Thus, it is my opinion that the client module enables the recipient to render the video media as it is created.

194. Brueck's client module 114 also enables a user using the end user station (recipient device) to render video media in a time-shifted mode, including by using the provided DVR capability. Brueck explains, for instance, that "[t]he agent controller module 702 [of the client module] may be configured to receive user commands from the viewer 708" and that "[s]uch commands may include play, fast forward, rewind, pause, and stop." Ex-1005 at 12:45-47; *see also* 8:42-50 (explaining that "publisher 110 may define functions such as fast forward, rewind, pause, and play that may be used with the content file 200"). "[T]o enable

fast forward and rewind capabilities, the streamlet cache module 704 [of the client module 114] is configured to store the plurality of streamlets 304 in the streamlet cache 710 for a specified time period after the streamlet 304 has been viewed." *Id*. at 13:16-20.

195. To render in a time-shifted mode, the client module retrieves the requested video media from the streamlet cache 710 shown above in Figure 7. Ex-1005, Fig. 7. Brueck explains, for instance, that "the streamlet cache module 704 is configured to receive streamlet 304 requests from the agent controller module 702. Upon receiving a request, the streamlet cache module 704 first checks a streamlet cache 710 to verify if the streamlet 304 is present." *Id*. at 13:6-10. Brueck further explains that, if "the streamlets 304 have been deleted [from the streamlet cache], they [still] may be requested again from the web server 116." *Id*. at 13:20-22; *see also* 13:23-26 (disclosing that the network controller module 706 of the client module may "receive streamlet requests from the streamlet cache module 704 and open a connection to the web server 116 or other remote streamlet 304 database").

196. In my opinion, by giving users the option to pause, stop, rewind and replay live video, Brueck's client module enables users to selectively render the video media in near real-time as the video media of the video message is created and transmitted by the sending device, and in a time-shifted mode by retrieving the

video media out of Brueck's streamlet cache module where such media is stored.

IX. GROUND 2: CLAIMS 8, 9, AND 14 ARE RENDERED OBVIOUS BY BRUECK IN COMBINATION WITH BROOKS.

197. In my analysis above, I explained why each of the limitations recited in these dependent claims is disclosed or rendered obvious by Brueck. *See supra*, ¶¶173-88. However, to the extent a combination is required, it is my opinion that such limitations are rendered obvious by the combination of Brueck and Brooks.

A. A Person of Ordinary Skill in the Art Would Have Been Motivated to Combine Brueck with the Teachings of Brooks

198. In my opinion, a person of ordinary skill in the art would have been motivated to combine Brueck with the teachings of Brooks, for a number of reasons. Brooks and Brueck are in the same field of art, namely delivering streaming media, such as audio and video, across a computer network, and both references are specifically directed to the problem of streaming video data. Like Brueck, Brooks discloses receiving live or stored video streams (Ex-1006 at 9:14-17) and processing the streams to generate differently encoded streams at different bitrates or sizes. *Id*. at 3:8-14; 14:4-11, Fig. 6B; Ex-1005, 6:54-65. In addition, both references disclose using an intermediate infrastructure to receive, process, and deliver live media to end users. Ex-1002 ¶198; Ex-1005, 6:15-30, 7:40-8:63, 10:44-60, Figs. 1, 4; Ex-1006, 4:48-56, 6:21-36, 9:14-17, 15:34-42; 17:23-25, Fig. 1. Moreover, Brueck and Brooks each disclose methods for transforming an input

media stream to match the capabilities of the recipient device. Ex-1006, 1:15-19, 3:1-4, Figs. 1, 5B; Ex-1005, 6:54-65, 8:16-25.

199. Further, both references address the challenges of transmitting live streaming video over variable bandwidth networks, such as the Internet. Ex-1002 ¶198. Brueck explains that a challenge with live streaming technology is that "the audio/video quality that can be received for real-time presentation is constrained by the available bandwidth of the user's network connection." Ex-1005, 1:35-37. Similarly, Brooks explains that, "in practice, many network connection speeds may vary greatly with network traffic, time of day, and the like." Ex-1006, 5:21-23. Thus, to the extent a person of ordinary skill desired additional detail regarding the implementation details recited in dependent claims 8, 9, and 14, such a person would have been motivated to consider the teachings of Brooks. One of ordinary skill would have understood that combining Brueck with Brooks, as described below, would have yielded predictable results without requiring undue experimentation.

B. Every Element of Claims 8, 9, and 14 Is Disclosed by Brueck Combined with Brooks.

1. Claim 8: "The method of claim 1, wherein the video message is received at the video message service infrastructure using a first real-time streaming protocol, and the selected degraded version of the video media is transmitted from the video message service infrastructure to the recipient device using a second real-time streaming

protocol."

200. In my opinion, Brueck in view of Brooks and discloses and renders obvious this limitation.

201. Brooks discloses a content delivery system wherein live video media may be streamed over the Internet from a sending device to an intermediate server using "a real-time streaming protocol known in the industry such as RTP, or the like." Ex-1006 at 9:26-33, 16:43-47 ("the input stream of video data use RTP or other data streaming protocols"). Brooks further explains that "[o]nce the data has been bandwidth reduced and format encoded, the stream of data is encoded for transmission [to the recipient device] via RTP ***or other protocol***." *Id*. at 17:17-19 (emphasis added). Brooks also notes that the RTP and RTSP protocols were "well known." *Id*. at 6:6-10.

202. In the combination of Brueck and Brooks, video media would be received at Brueck's content server via RTP, as disclosed in Brooks, and transmitted to the recipient device using HTTP, as disclosed in Brueck. HTTP would have been the "other protocol" contemplated by Brooks.

203. In my opinion, it would have been obvious for one of ordinary skill to combine Brueck and Brooks in this manner, and such a person would have readily understood that doing so would have yielded predictable results, namely the delivery of media in real-time from the sending device to the video message

infrastructure, and the delivery of the same media (possibly suitably degraded depending on ascertained bandwidth conditions) from the video message infrastructure to the recipient device. A person of ordinary skill would have recognized that using RTP or RTSP to transmit media from the content publisher, as described in Brooks, would have advantageously allowed the publisher to stream live video over an IP network, as described in Brooks and Brueck. A person of ordinary skill in the art also would have understood that commonly-used and Internet standard protocols, such as RTP and RTSP, would ensure a high degree of compatibility between the independently developed publisher system and Brueck's video message service infrastructure.

2. Claim 9

204. Claim 9 depends from claim 8, adding the requirement that "the first real-time streaming protocol is different from the second real-time streaming protocol." For the reasons explained in my analysis of claim 8, above, this limitation is disclosed and rendered obvious by the combination of Brueck and Brooks. *See supra*, ¶¶201-03.

3. Claim 14: "The method of claim 1, further comprising …"

205. As discussed in my analysis above, Brueck discloses or renders obvious this claim even without a combination. *See supra*, ¶¶180-88. However, if it is determined that Brueck lacks sufficient specificity regarding the particular

implementation details recited in claim 14, a person of ordinary skill in the art would have been motivated to consider the teachings of Brooks.

206. In my opinion, it would have been obvious to adapt Brueck's system so as to transcode between codecs as described in Brooks. Indeed, a person of ordinary skill in the art would have understood that implementing transcoding, as disclosed in Brueck, in the manner described in Brooks would have involved nothing more than the application of a known technique to a known method. Moreover, such a person would have expected such a combination to yield predictable results, particularly because transcoding video media was well understood and commonly used in the networking arts long before the alleged invention of the '557 patent.

a. "ascertaining a first codec that was used by the sending device to encode the video media of the video message when sent" (14[a])

207. In my opinion, Brueck in view of Brooks discloses this limitation. Brooks discloses that live streaming video transmitted from a sending device may be "encoded in formats such as, JPEG, JPEG-2000, GIF, WBMP, MPEG-1, MPEG-2, MPEG-4, H.263, *.avi, *.mov, *.rm, *.aff, and the like." Ex-1006 at 9:34-44, 9:14-17, 5:48-6:36.

208. Brooks also discloses "ascertaining" the codec used to encode the video media. For instance, Brooks explains that a control block 450 "***receives***

information associated with the stream of input video data," including (among other things) "***the video format, i.e. how the input stream of data is encoded***, such as MPEG format, Windows Media format, H.263, QuickTime format, Real Video format, or the like." Ex-1006 at 9:52-62 (emphasis added); *see also* 9:45-50; Fig. 3. "The data associated with the input video data may be derived from the input video data itself." *Id*. at 9:62-67.

b. "ascertaining a second codec to be used by the recipient device to decode the video media of the video message when received" (14[b])

209. In my opinion, Brueck in view of Brooks also discloses this limitation. Brooks explains, for example, that "[c]ontrol block 450 also receives information associated with desired properties of streams of output video data," including (among other things) "information regarding what format the output stream of data should be encoded in, such as M-JPEG, GIF, MPEG format, H.263 format, Windows Media format, Quicktime format, Real Video format, or the like." Ex-1006 at 10:1-13, 20:4-11, *see also* 19:63-65 ("The next step in this example is to determine the data encoding format, step 930. . . ."), 20:3-11; Figs. 3, 6B.

210. Brooks explains that "[t]he data associated with the output video data is typically derived from the requesting device." Ex-1006 at 10:14-15. For instance, when requesting a video stream, the recipient device may "inform [the

system] which output video format should be used to encode the data" for decoding on the recipient device. *Id*. at 10:17-28. In the alternative, "the requesting device (computer, PDA, cell phone, etc.) merely identifies itself to gateway system 100." *Id*. at 10:31-34; Fig. 1. "Gateway system 100 then receives the identifier, uses the identifier to locate an entry in a database file or other external source that specifies the bandwidth requirements, ***format requirements***, and the like, for that device." *Id*. at 10:29-42 (emphasis added); *see also id*. at 10:61-64 ("In response to the unique identifier, gateway system 100 will determine the bandwidth requirements, ***desired output format***, and the like of the device. This data is then passed to control block 450 for processing.") (emphasis added).

c. "trans-coding the video media of the video message from the first codec to the second codec" (14[c])

211. Brueck in view of Brooks discloses this limitation. Brooks discloses, for instance, that streaming video data received by the frame buffer 410 is then passed to the transcoder block 420 such that a customized output stream may be generated. In particular, Brooks teaches that the transcoder block 420 in Figure 3 "retrieves incoming data from frame buffer 410, fully decompresses or partially decompresses the data, reduces the bandwidth of the data, ***and forms a stream of output data in a desired format***." Ex-1006, 11:45-49 (emphasis added); Fig. 3.

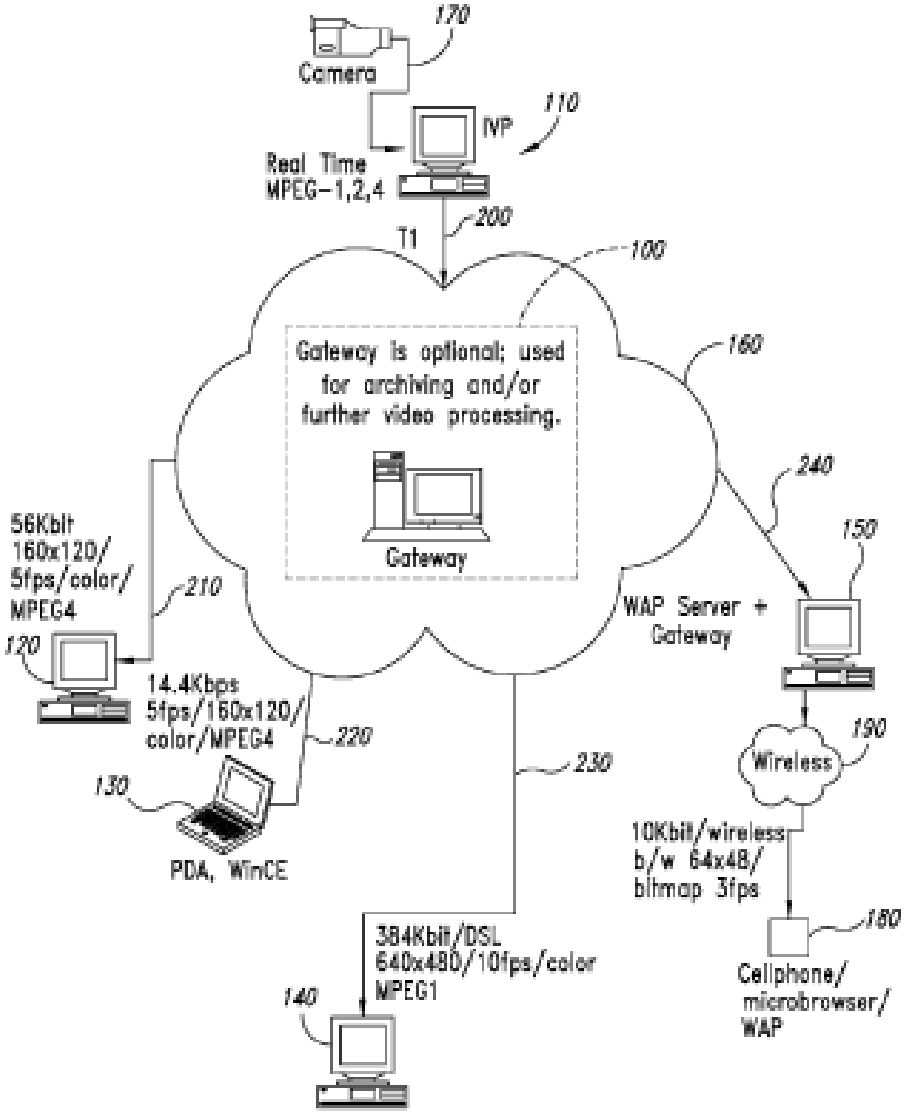
212. "Transcoder block 420 receives the bandwidth requirements and the desired output format from control block 450." Ex-1006 at 11:49-52; *see also* Fig.

4, 12:13-16 ("FIG. 4 illustrates functional blocks available in a transcoder 500"), 14:55-15:12 (describing encoding block 560, which "receive[s] a request for an encoding data format, specified for by control block 450," then encodes the appropriate output format), Fig. 5A, 17:13-15 ("the transcoder retrieves the frame of data and formats the data according to the bandwidth and format requirements for the output video stream, step 680").

d. "transmitting, to the recipient device, the trans-coded video media" (14[d])

213. Brueck in view of Brooks discloses this limitation. For instance, Brooks discloses that stream caster block 430 receives a stream of output video data from transcoder block 420 and formats the data for transmission to network interface 440, which transmits the transcoded video media to a recipient device via a network, such as the Internet. Ex-1006 at 11:53-12:3, 17:17-35. "The data is then transmitted to a requesting device . . . ." *Id*. at 17:21-22, Fig. 5B (showing at step 680 "transcod[ing] input video data to the desired output video" and at step 700 "send[ing] TCP/IP packets to the requester").

214. In Figure 1, for instance, Brooks provides an example of a system in which transcoded video media is transmitted to recipient devices "in the output format desired":



Ex-1006, Fig. 1; 7:21:25; *see also id*. at 6:21-7:31, 20:33-44.

X. <u>GROUND 3</u>: CLAIM 16 IS RENDERED OBVIOUS BY BRUECK IN COMBINATION WITH MULLIG

215. Claim 16 adds limitations requiring that the "video message service infrastructure" receive a second video message from a second sending device and transmit the video media from that message to a recipient device such that the recipient device can render the media of the first and second messages

simultaneously. Mullig discloses this basic concept. *See supra*, ¶¶115-17. As described above, Mullig's content server is configured to receive live video streams from multiple content sources and to deliver them concurrently to a recipient device, "allowing the user to view two or more live videos at the same time." Ex-1007 at 1:38-42. For example, as shown in Figure 4, the receiving device may display "a high resolution version of one of the live video feeds in a 'primary' video pane 401 and low resolution versions of the additional live video feeds in 'preview' video panes 403 of the main display dialog window 400 substantially concurrently":

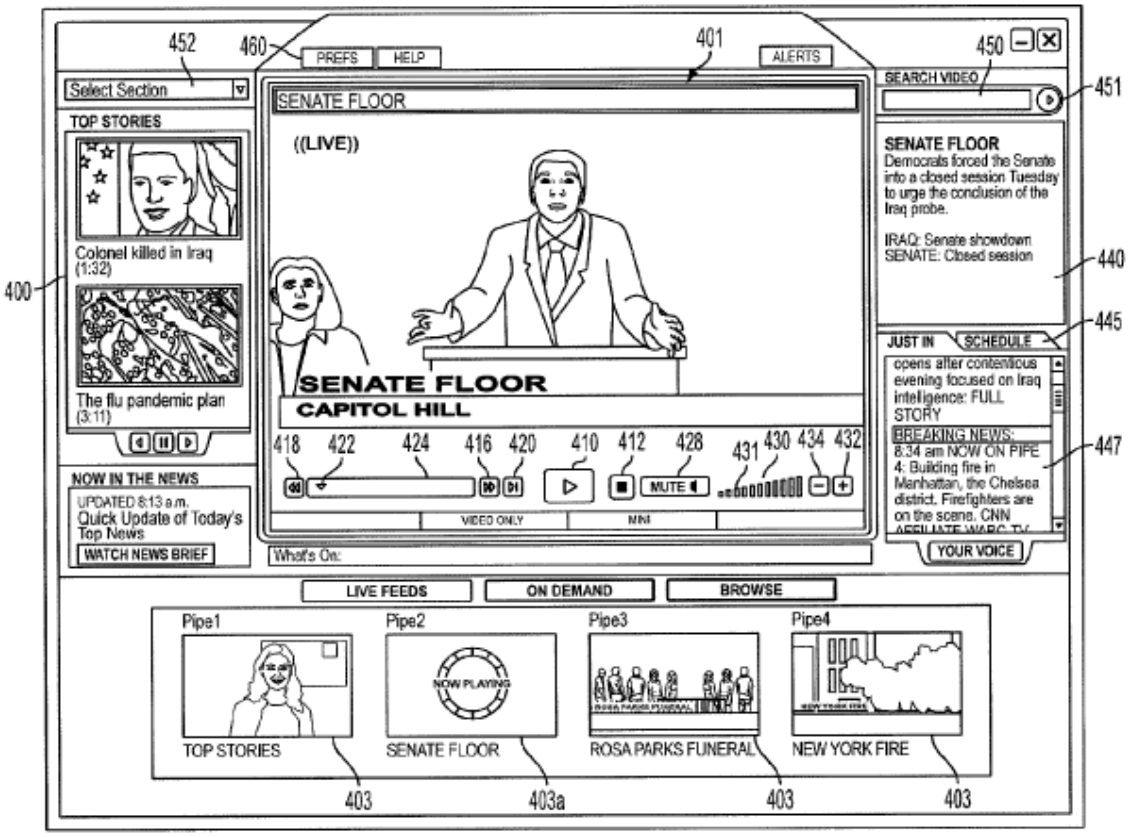


FIG. 4

Id., Fig. 4, 9:49-59.

216. In my opinion, it would have been obvious to a person of ordinary skill in the art to adapt Brueck's content server for use with Mullig's system, among other reasons, because Brueck's content server is capable of concurrently receiving and processing video media from multiple publishers. It was well known, for instance, that computer-based methods for providing a network service, especially those realized on server computers, were typically implemented such that multiple clients could be supported simultaneously. Indeed, Brueck explains that its content server could receive streaming video (comprising, for instance, a live sports event or concert) from a "television or cable network channel such as NBC®, or MTV®." Ex-1005 at 6:17-20. A person of ordinary skill in the art would have understood that, in such implementations, multiple sending devices commonly would be used to send video media streams to Brueck's content server. One of ordinary skill would have recognized, for instance, that when broadcasting a live concert or sporting event, more than one sending device is commonly used to capture the live event from different angles.

217. Likewise, a person of ordinary skill in the art would have understood that Brueck's content server could be adapted for use with Mullig's system to transmit more than one live video feed for simultaneous rendering on a recipient device. Content delivery systems providing this functionality were well known in

the art at the time of the alleged invention.

218. Thus, in my opinion, combining Brueck with Mullig in the manner described below would have required nothing more than the application of a known technique to a known method ready for improvement. In my opinion, such an implementation would have been well within the competence of a skilled artisan and would have yielded predictable results (the ability to process and deliver multiple video streams concurrently) without requiring undue experimentation.

A. A Person of Ordinary Skill in the Art Would Have Been Motivated to Combine Brueck with the Teachings of Mullig

219. Brueck and Mullig are in the same field of art, namely delivering streaming media, such as live or on-demand video, to end users. Moreover, both references disclose receiving live video at an intermediate server and delivering it to end users for rendering in near real-time. Ex-1007 at 1:38-42, 9:39-48, 10:57-11:8, Fig. 1; Ex-1005 at 6:15-30, 7:40-8:63, 10:44-60, Figs. 1, 4.

220. Both references likewise disclose that the video content may comprise streaming video of live events, such as sports events, and that such content may be provided by producers such as television or cable networks. Ex-1007 at 9:12-37, 10:64-67, 12:3-12, Fig. 4; Ex-1005 at 6:15-22, 6:43-45, 11:22-33. Both references disclose providing DVR-like controls allowing users to pause, rewind, or fast forward live content. Ex-1007 at 2:3-15, 12:24-49; Ex-1005 at 12:45-47. Finally, both references disclose methods for optimizing bandwidth. Ex-1007 at 9:44-48,

11:30-39, 25:3-16; Ex-1005 at 6:54-7:39, 12:38-44; Figs. 4, 10.

221. Thus, a person of ordinary skill in the art would have had ample reason to combine Brueck with Mullig's teachings regarding the transmission and simultaneous rendering of video streams received from multiple sending devices. A person of ordinary skill in the art would have understood that the combined system would beneficially allow recipients to view multiple video streams simultaneously, as disclosed in Mullig, while allowing the quality of the video streams to be adapted dynamically in view of bandwidth conditions, as disclosed in Brueck.

B. Every Element of Claim 16 Is Disclosed by Brueck Combined with Mullig.

1. Claim 16: "The method of claim 1, further comprising:"

a. "receiving, at the video message service infrastructure, a second video message transmitted by a second sending device, the second video message including second video media, wherein the second video media being transmitted by the second sending device as the second video media is created;" (16[a])

222. In my opinion, Brueck in view of Mullig discloses this limitation. As I explain in my analysis above, Brueck discloses a content server configured to receive live video messages from a sending device. *See supra*, ¶¶129-33. Although Brueck does not expressly disclose concurrently receiving video streams from multiple sending devices, a person of ordinary skill in the art would have understood that Brueck's system is capable of doing so, as discussed above. *See*

supra, ¶¶216-18. For additional detail regarding the receipt of multiple video messages, a person of ordinary skill in the art would have been motivated to combine Brueck with the teachings of Mullig. *Id*.; *see also id*., ¶¶219-21.

223. Mullig discloses a content delivery server 25 configured to receive live video streams from multiple sources and to deliver them concurrently to a recipient device, "allowing the user to view two or more live videos at the same time." Ex-1007 at 1:38-42, 9:39-48, Abstract. "The live video feeds may be streamed from independent sources to the content delivery server 25 (e.g., by producers), or the live video feeds may be streamed from a common source." *Id*. at 9:67-10:3. For example, the "content delivery server 25 may maintain links with sources that provide video feeds of various camera angles showing the same event, such as for a sports event." *Id*. at 10:64-67; *see also id.* at 10:58-67, 11:1-29, 12:3-12; Fig. 6.

224. In the combined system, Brueck's content server would receive a plurality of live video streams from multiple publishers, as disclosed in Mullig, and would process those streams for delivery to a recipient device in a bandwidth-optimized format, as described in Brueck.

b. "transmitting the second video media from the video message service infrastructure over the network to the recipient device so that the recipient device can render at least a portion of the second video media;" (16[b])

225. In my opinion, Brueck in view of Mullig also discloses this limitation.

As discussed above, Brueck's content server is configured to transmit the video media of a video message so that it can be rendered on a recipient device. *See supra*, ¶¶151-58. To the extent Brueck does not expressly disclose sending multiple video messages to a recipient device, in my opinion, a person of ordinary skill in the art would have been motivated to consider the teachings of Mullig, as explained above. *See supra*, ¶¶216-21.

226. Mullig discloses "concurrently stream[ing] a plurality of live videos to a client computing device over a network, allowing the user to view two or more live videos at the same time." Ex-1007 at 1:38-42, 9:38-44. For instance, Mullig discloses that recipient devices (client computing devices 115) "execute a client application 15 that is configured to communicate with the content delivery server 25, request content from the content delivery server 25, and receive, manage, and display content distributed from the content delivery server 25." *Id*. at 5:58-63; Figs. 1, 4.

227. The content delivery server 25 is "configured to stream a plurality of live video feeds to the client application 15 over a network substantially concurrently, and the client application 15 is configured to display one or more of the plurality of live video feeds to the user substantially concurrently." Ex-1007 at 9:38-44, 1:38-42; Fig. 4.

228. In the combination of Brueck and Mullig, Brueck's content server

would be configured to receive live video streams from a plurality of sending devices, process the video media as disclosed in Brueck, and deliver multiple streams for simultaneous rendering on a requesting recipient device, as taught by Mullig.

c. "the video message service infrastructure operating so that the rendering occurs: (i) as the second video media of the second video message is created and transmitted over the network by the second sending device;" (16[c])

229. It is my opinion that Brueck in view of Mullig discloses this limitation. As discussed above, Brueck's content server may transmit live video media to the recipient device such that rendering occurs as the video media is being created and transmitted by the sending device. *See supra*, ¶¶151-58. A person of ordinary skill in the art would have been motivated to combine Brueck with Mullig in order to transmit multiple video streams to a recipient device, as discussed above. *See supra*, ¶¶216-21.

230. Mullig discloses that its content delivery server 25 is configured to stream a plurality of live video feeds to a recipient device such that the recipient device may render the two or more live videos at the same time. Ex-1007 at 1:38-54, 9:39-44; Fig. 4. For instance, in one embodiment, "live video feeds of a sporting event are captured by a plurality of cameras, and the live video feeds are streamed to the client application 15 and displayed substantially concurrently by the client application 15 to the user." *Id*. at 12:3-7, 12:7-12, 12:24-58.

d. "(ii) without having to first establish an end-to-end connection over the network between the second sending device and the recipient device before the second sending device begins to transmit the second video media of the second video message; and" (16[d])

231. Brueck in view of Mullig also discloses this limitation. As I explain in my analysis above, in Brueck's system, no end-to-end connection between the sending device and the recipient device need be established before the sending device begins to transmit a live video message. *See supra*, ¶¶159-61. Indeed, Brueck contains no disclosure or suggestion of any such connection existing or being required.

232. Mullig discloses a similar architecture wherein content from a plurality of sources may be received at a central server, processed and stored, then delivered to the recipient device at the request of an end user. Ex-1007 at 5:49-63, 6:26-35, 10:7-13; Figs. 2, 6. Accordingly, in my opinion, a person of ordinary skill in the art would have understood that Mullig does not require an end-to-end connection to be established between sending and recipient devices before the sending device begins transmission. Indeed, Mullig does not require the establishment of any connection between the sending and recipient devices before the sending device begins to transmit video media, and contains no disclosure or suggestion of any such connection being required.

233. Likewise, the combined system would employ an intermediate server

that receives, processes, stores, and delivers content to end users. A person of ordinary skill in the art would have understood that in such a system no end-to-end connection (and, indeed, no connection at all) would need to be established between the first or second sending device, on the one hand, and the recipient device, on the other, before the sending device begins to transmit the video message.

e. "(iii) simultaneously with the rendering of the video media of the video message." (16[e])

234. In my opinion, Brueck in view of Mullig discloses this limitation. As discussed above, Mullig's content delivery server 25 is configured to stream multiple live video streams to a recipient device such that recipient device may render those streams simultaneously. Ex-1007 at 1:38-54, 9:38-56, Fig. 4. For instance, Mullig explains that the content delivery server 25 may "stream a plurality of live video feeds to the client application 15 over a network substantially concurrently" and that "the client application 15 [may] display one or more of the plurality of live video feeds to the user substantially concurrently." *Id*. at 9:38-44. In one embodiment, the system "display[s] a high resolution version of a live video feed in a 'primary' pane of a main display dialog window and one or more additional live video feeds in low resolution in 'preview' panes of the main display dialog window substantially concurrently, wherein the additional live video feeds are different from the first video feed and each other." *Id*. at 1:38-54; *see*

also 1:55-64, 2:3-5.

235. In the combined system, Brueck's content server would be configured to process live video received from publishers, as disclosed in Brueck, and to transmit multiple live video streams for simultaneous rendering on a recipient device, as taught by Mullig.

XI. GROUND 4: CLAIM 23 IS RENDERED OBVIOUS BY BRUECK IN COMBINATION WITH KALABOUKIS AND MANGAL

236. Claim 23 claims the combination of various concepts—such as using "a plurality of servers," and enabling recipient devices to render video media in a time-shifted mode—that are disclosed by Brueck. *See supra*, ¶¶127, 194-96. However, claim 23 also requires the element of "ascertaining locations on the network for multiple recipient devices including the recipient device, the locations derived from multiple identifiers provided by the sending device."

237. Given Brueck's focus on adaptive streaming technology, Brueck does not describe in detail the publishers of the streaming content. Nevertheless, a person of ordinary skill in the art would have been motivated to implement Brueck's invention in a streaming platform that ascertains the locations of the recipient devices using identifiers provided by the sending device. It would have been obvious to create a combination to advantageously enable content providers to specify the intended recipients of a video message (*e.g.*, subscribers), while beneficially allowing end users to receive the video message in a bandwidth-

optimized format. This is disclosed by Brueck combined with Kalaboukis and Mangal.

238. Kalaboukis discloses a video streaming system "allowing geographically dispersed members of a user's social network to experience a user's current locale, so that the user's friends may join him via live video at events (concerts, clubs, or other venues) which they cannot attend in person." Ex-1008, Abstract.

239. In Kalaboukis's system, a user may select one or more recipients for the live video message and any of the selected recipients may render the video in near real-time or in a time-shifted mode. Ex-1008 at 2:59-3:4, 3:24-37. An interaction server receives input from the sender identifying the intended recipients, transmits invitations and responses thereto, and signals a media server to stream the video message to invitees who elect to view the live broadcast. *Id*. at 2:52-3:7, 3:38-46.

240. Although Kalaboukis does not disclose the particular method by which its interaction server ascertains the location of recipient devices, techniques and protocols for translating identifiers into network locations were well known in the art. For instance, Mangal teaches that users may identify the recipient of a video message using the recipient's SIP address and that a "[SIP] proxy server may then query a SIP registry to determine an IP address of the terminating endpoint."

Ex-1009 at 2:55-65.

A. A Person of Ordinary Skill in the Art Would Have Been Motivated to Combine Brueck with the Teachings of Kalaboukis and Mangal.

241. Brueck and Kalaboukis are in the same field of art, namely delivering streaming media, including live audio and video, from content publishers to end users. Brueck and Kalaboukis each describe systems that include a central infrastructure for receiving, processing, and routing the live video and audio media between users of the system. Ex-1005, Figs. 1, 4; Ex-1008 at 3:17-61, Figs. 4. Moreover, Brueck and Kalaboukis each disclose storing the live media, thereby enabling recipients to render live video messages in near real-time or time-shifted modes. Ex-1005 at 13:16-20 ("to enable fast forward and rewind capabilities, the streamlet cache module 704 is configured to store the plurality of streamlets 304 in the streamlet cache 710"), 10:53-60, 12:45-47; Ex-1008 at 1:51-53 ("[T]he processing of the live video signal may also include caching the video signal, so that the event may be viewed other than in real time.").

242. In addition, both references disclose adapting the delivery of video media to account for the capabilities of the recipient device. Ex-1005 at 8:23-25, 12:38-42; Ex-1008 at 3:52-61. But while Kalaboukis teaches that "each user may receive a streaming video signal in a convenient format," and that the system may (if appropriate) "dumb down" the signal, Kalaboukis does not describe in detail

how the system would adapt the quality of the video media for delivery to end users. Ex-1008 at 3:52-58. Thus, a person of ordinary skill in the art reviewing Kalaboukis would have been motivated to consider the teachings of Brueck, which describes in detail how the delivery of streaming media may be adapted dynamically to account for the bandwidth available to recipient devices on the network.

243. Similarly, Brueck discloses a system capable of receiving live video media from various sorts of content publishers, including creators of content, but does not describe in detail how live video content is created and transmitted from the sending device. Ex-1005 at 6:15-20. Thus, a person of ordinary skill in the art reviewing Brueck would have been motivated to consider the teachings of Kalaboukis, which describes an application executing on a sending device, such as a cell phone, enabling publishers of live video content to transmit video media to a central server, as disclosed in Brueck, for delivery to a recipient device. Ex-1008 at 2:52-3:46, 2:40-47.

244. In my opinion, a person of ordinary skill in the art would have recognized the benefit of implementing Brueck's adaptive rate-shifting technology in a system such as described by Kalaboukis. For instance, one of ordinary skill would have understood that Kalaboukis's media server could incorporate Brueck's adaptive rate-shifting technology and that such an implementation would

beneficially allow "each user [to] receive a streaming video signal in a convenient format," as described in Kalaboukis. Ex-1008 at 3:52-65.

245. A person of ordinary skill in the art reviewing Brueck and Kalaboukis would have understood that various methods well-known in the art could be used to ascertain network locations for the recipient devices from identifiers provided by a sending device. But if additional detail were desired, a person of ordinary skill in the art would have been motivated to consider the teachings of Mangal.

246. Mangal, Brueck, and Kalaboukis all disclose receiving and storing live streaming video at an intermediate server and transmitting the video media to one or more recipient devices for rendering in near real-time. Moreover, like Kalaboukis, Mangal discloses streaming live video messages in a mobile environment to recipients identified by the sender. Ex-1009 at 1:11-13, 2:35-3:45, 11:64-12:13, 14:4-13.

247. Likewise, Kalaboukis and Mangal both disclose an intermediate server that processes invitations and responses thereto. Ex-1009 at 2:55-3:45, 7:9-22; Ex-1008 at 2:66-3:16. It would have been obvious to a person of ordinary skill in the art that Kalaboukis's interaction server could be configured to translate identifiers into recipient IP addresses, as disclosed in Mangal, and that doing so would advantageously enable invitations and streaming video to be delivered to recipient devices, as disclosed in Kalaboukis.

B. Every Element of Claim 23 Is Disclosed by Brueck Combined with Kalaboukis further in view of Mangal.

1. Claim 23: "The method of claim 1, further comprising:"

a. "configuring the video message service infrastructure to include a plurality of servers located on the network, the plurality of servers collectively configured for performing:" (23[a])

248. Both Brueck and Kalaboukis disclose a video message service infrastructure that includes a plurality of servers. As discussed, Brueck discloses a content server with a content module and a web server. *See supra*, ¶127. As shown in Figure 1, Brueck's system may also include one or more additional web servers and/or proxy cache servers. Ex-1005 at Fig. 1; 6:20-33.

249. Kalaboukis discloses a media capture server 107, an interaction server 110, and a media server 111 that collectively permit an individual to stream live video messages over the Internet to selected remote users. Ex-1008 at 1:62-2:2; 3:17-23, 3:38-46, 5:1-8; Figs. 1A, 1C, 4. The media capture server 107 is configured to receive a live video message from a sending device and cause it to be saved in a live streaming database 108 accessible to the media server 111. *Id*. at 3:16-23; Figs. 1B, 1C. The interaction server 110 receives input from the user of a sending device identifying the intended recipients of a video message and, in response, generates invitations. *Id*. at 2:52-3:7. If a targeted recipient chooses to view the video, the interaction server signals the media server 111 to retrieve the stored video media and initiate a streaming session with the recipient device. *Id*. at

3:24-51.

250. In my opinion, a person of ordinary skill in the art would have understood that Brueck could be combined with Kalaboukis in various ways to achieve the benefits of a combined system. Though various configurations are possible, as one example, a combined system could consist of a content server (with web server), as described in Brueck, operating in cooperation with an interaction server, as described in Kalaboukis. Ex-1008 at 5:1-3 ("Although the servers 107, 110, 111 have been described as different servers to point out their different functions, it will be appreciated that they need not be separate devices.").

b. "receiving the video message;" (23[b])

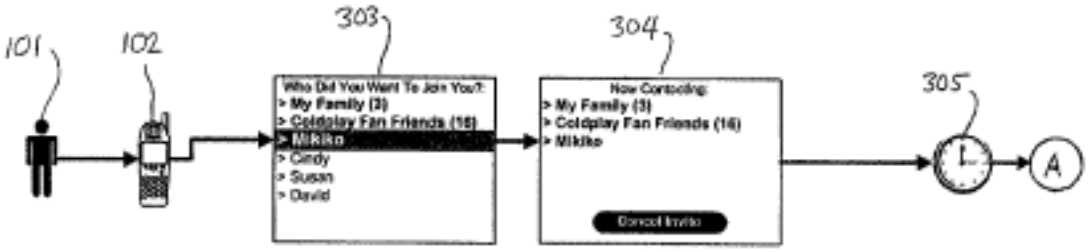
251. As I explain in my analysis above, Brueck discloses that live video messages from the publisher 110 are received by the content module 112 of the content server 102. *See supra*, ¶¶129-31; Ex-1005 at 6:20-22. Kalaboukis discloses that the video message is received by media capture server 107. Ex-1008 at 3:16-23; Figs. 1B.

252. In the combination of Brueck and Kalaboukis, a video message also would be received. For example, the combined system would include a content server, as described in Brueck, that includes a content module for receiving the video message. The decision whether or not to provide a separate media capture server, as disclosed by Kalaboukis, would have been a design choice from among a

limited set of possibilities.

c. "ascertaining locations on the network for multiple recipient devices including the recipient device, the locations derived from multiple identifiers provided by the sending device;" (23[c])

253. Kalaboukis discloses that "user 101 starts the JoinMe application on his telephone camcorder device 102, as shown in FIG. 1A," then "decides which people to invite to join him; that is, who he wishes to be recipients of the live video (step 202)." Ex-1008 at 2:52-61. "When user 101 starts the application, the screen displays a list 303 of possible invitees (that is, members of group 103), as shown in FIG. 3A":



Ex-1008, Fig. 3A; 4:25-27.

254. Invitees may include "individually selected persons" and/or "members of a predetermined group with a single group identifier." Ex-1008, 2:61-64. As shown in Figure 3A, "[a]fter user 101 has made his selection of whom to invite, the screen display 304 shows the progress of contacting the invitees." *Id*. at 2:27-29.

255. It would have been obvious to a person of ordinary skill in the art that,

when a user selects more than one individual or group for receipt of the video message (as in Figure 3A), the sending device typically would provide multiple identifiers. Although Kalaboukis does not disclose the transmission of any particular identifier, the '557 patent likewise provides no such details. *See* Ex-1001 at 22:45-47 (disclosing only that "[w]hen a Client 12 transmits a message, the message contains an identifier associated with the target recipient").

256. Kalaboukis further discloses that, once the user has selected the target recipients, an interaction server 101 communicates with the selected individuals to generate invitations and process responses. Ex-1008 at 2:66-3:7; Fig. 1C. As invitations are accepted, the interaction server signals the media server 111 to transmit the video message to the recipient device(s). *Id*. at 3:8-51. Accordingly, a person of ordinary skill in the art would have understood that Kalaboukis's system ascertains a delivery location on the network for each of the recipient devices. Ex-1001 at 26:9-11 ("The address of a target Client 12 needs to be known so that the underlying network delivers the Vox packet 95 to the correct location.").

257. Although Kalaboukis does not explain in detail how the locations for the recipient devices are ascertained, techniques for translating identifiers into network locations were well known in the art, as the '557 patent itself acknowledges. *See* Ex-1001 at 26:15-21 ("IP networks use the Domain Name System (DNS) to resolve human-readable names into IP addresses, and the

Address Resolution Protocol (ARP) to resolve IP addresses into physical addresses. Regardless of the underlying networking technology, the system 10 uses one of the above-mentioned ***or other known addressing schemes*** for delivery of Vox packets 95 to the correct location.") (emphasis added), 22:45-49.

258. It would have been obvious to a person of ordinary skill in the art to combine Brueck and Kalaboukis such that the system would include an interaction server, or the like, that is configured to receive multiple identifiers from a sending device and to ascertain locations on the network for each of the target recipients using those identifiers. This would have been involved nothing more than the application of a known technique to a known method and would not have required undue experimentation.

259. However, if additional detail were required regarding how the location of recipient devices could be ascertained from identifiers provided by a sending device, a person of ordinary skill in the art would have been motivated to consider the teachings of Mangal, as discussed above. *See supra*, ¶¶240, 245-47.

260. Mangal discloses a real-time communication system allowing users of mobile devices to send live video messages to designated recipients. Mangal explains that a user may "initiate a [real-time] session with a designated group of one or more other users" by pressing a PTT button, resulting in the transmission of a SIP "INVITE" message to a SIP proxy server. Ex-1009 at 3:32-40, 2:61-63.

Mangal discloses using a Session Description Protocol ("SDP") block in the INVITE message to identify the recipients: "[a]n SDP in the INVITE request may identify or list a 'communication group' for the user, i.e., a group of other users with whom the initiating user would like to communicate." *Id*. at 3:41-44. "The proxy server may then query a SIP registry to determine an IP address of the terminating endpoint." *Id*. at 2:64-65. Mangal further discloses that this process can be applied to any real-time media such as video. *Id*. at 14:4-13. Indeed, a person of ordinary skill in the art would know that the Internet standard SIP protocol and architecture was content agnostic and was used to establish media communication sessions for a wide variety of media including audio, video, and audio and video.

261. As explained above, a person of ordinary skill in the art would have been motivated to apply Mangal's teachings, among other reasons, so that the combined system would be capable of sending invitations and streaming video to the recipients identified by the sender. *See supra*, ¶¶245-47.

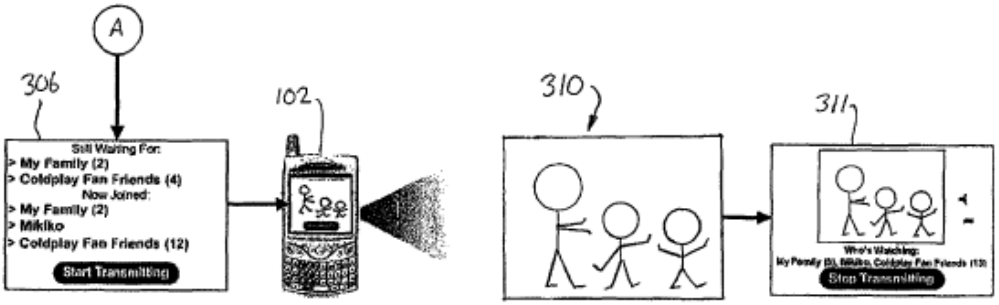
d. "transmitting the video message to the multiple recipient devices so that each of the multiple recipient devices can render at least a portion of the video message as the sending device is sending the video message;" (23[d])

262. As I explain in my analysis above, Brueck discloses transmitting live video messages to multiple recipient devices for rendering in near real-time as the

sending device is sending the video message. *See supra*, ¶¶155-58, 168-69.

263. Likewise, Kalaboukis discloses transmitting a video message to multiple recipient devices such that rendering may occur as the sending device is sending the video message. Kalaboukis explains, for instance, that live video received from the sending device is delivered to a group of recipients 109 via the media server 111. Ex-1008 at 3:38-51, 2:31-34 ("FIGS. 1A-1C schematically illustrate an embodiment of the disclosure in which a user 101 is attending an event 100, and wishes to share live images of the event with a number of other persons who are located elsewhere."), 2:37-51 ("A 'live' video image of an event is understood to refer to an image available to a person operating a remote device at or nearly at the time the event itself transpires (taking into account, by way of non-limiting example, brief time periods required for transmitting, buffering, caching and/or storing a signal representing the event).").

264. As the video message is being transmitted in near real-time to the group of recipients 109, the user "device 102 displays screen 311, showing the image being transmitted and identifying who is watching," as shown in Figure 3B:



Ex-1008, Fig. 3B; 4:41-43. The "display changes as viewers join or leave the group." *Id*. at 4:43-44.

265. In the combination of Brueck and Kalaboukis, the system likewise would be configured to transmit the video message to multiple recipient devices for rendering in near real-time. In my opinion, a person of ordinary skill in the art would have understood that this could be accomplished, for instance, by providing a content server with a web server, as described in Brueck, that cooperates with an interaction server, as described in Kalaboukis, to deliver live video in near real-time to the recipient devices.

e. "storing the video message as the video media is received by the video message service infrastructure; and" (23[e])

266. As explained above, the content module 112 of Brueck's content server 102 is configured to receive, process, and store the video message as the video media is received at the content server. *See supra*, ¶¶128-31, 134-39, 159; *see also* Ex-1005 at 6:22-30, 5:64-6:20, 6:40-47, 7:40-8:32, 8:53-9:10, 9:66-67; Fig. 4. For instance, the encoder module 406 is configured "to encode each source

streamlet 303 into the plurality of streams 202 and streamlet sets 306 and store the streamlets in the streamlet database 408." Ex-1005 at 8:26-29, 6:31-33 ("Content from the content server 102 may also be replicated to other web servers 116 or alternatively to proxy cache servers 118."); *see also id*. at 9:38-42, 15:36-44; Fig. 5.

267. Kalaboukis also discloses this limitation. For instance, Kalaboukis discloses that "processing of the live video signal may . . . include caching the video signal, so that the event may be viewed other than in real time." Ex-1008 at 1:50-53, 3:20-23, 3:33-37; Figs. 1C, 2, 4.

268. In the combined system, the video message also would be stored as it is received at the video message service infrastructure. The combined system would include a content server, as described in Brueck, with a streamlet database for storing multiple versions of the live video at different bitrates and different levels of quality. A person of ordinary skill in the art would have understood that such storage would have beneficially allowed the combined system to transmit video media to recipient devices for rendering in near real-time or in a time-shifted mode, as disclosed by Brueck and Kalaboukis.

f. "providing the video media of the video message to any of the multiple recipient devices for rendering in a time-shifted mode." (23[f])

269. As I explain above, Brueck discloses providing the video media of the

video message to recipient devices for rendering in a time-shifted mode. *See supra*, ¶¶194-96. For instance, Brueck's system allows recipients to pause, rewind, and fast forward live video messages. Ex-1005 at 12:45-47. If a requested streamlet is not stored locally, the video media "may be requested again from the web server 116." *Id*. at 13:20-22, 13:23-26.

270. Kalaboukis discloses that live video messages are stored in a server-level database, thereby allowing any of the multiple recipients to review video messages in a time-shifted mode. Ex-1008 at 1:50-53 ("The processing of the live video signal may . . . include caching the video signal, so that the event may be viewed other than in real time."); *see also id*. at 2:52-3:4, 3:18-27, 3:38-46, Fig. 3B. For example, Kalaboukis explains that "[s]ince the stream is cached, an invitee accepting the invitation after the live image capture has begun may be given the option of joining the live event in progress, or viewing the event from the beginning." Ex-1008 at 3:33-37.

271. The combined system also would allow any of the multiple recipients to review the video message in a time-shifted mode. As discussed, for example, a person of ordinary skill in the art would have understood that the combined system would include Brueck's streamlet database. This would allow any of the multiple recipients to retrieve a version of the video message for rendering in a time-shifted mode, as described by Brueck and Kalaboukis.

XII. CONCLUSION

272. For all of the foregoing reasons, my opinion is that the challenged claims of the ’557 patent are unpatentable as obvious.

I declare under penalty of perjury under the laws of the United States of America that the foregoing is true and correct.

Dated: October 22, 2020

By: _____
Dr. Kevin Jeffay
Declarant

Exhibit A

CURRICULUM VITAE

Kevin Jeffay
University of North Carolina at Chapel Hill
Department of Computer Science
Chapel Hill, NC 27599-3175
(919) 590-6238
jeffay@cs.unc.edu
http://www.cs.unc.edu/~jeffay

Education

Ph.D. Computer Science, University of Washington.
Honors: IBM Graduate Fellowship.

M.Sc. Computer Science, University of Toronto.
Honors: University of Toronto Open Fellowship.

B.S. Mathematics with Highest Distinction, University of Illinois at Urbana-Champaign.
Honors: Phi Beta Kappa, James Scholar.

Academic Experience

Chairman, Department of Computer Science, University of North Carolina at Chapel Hill, Chapel Hill, NC, 2014–present.

Gillian T. Cell Distinguished Professor in Computer Science, University of North Carolina at Chapel Hill, Department of Computer Science, Chapel Hill, NC, 2008–present.

S. S. Jones Distinguished Professor in Computer Science, University of North Carolina at Chapel Hill, Department of Computer Science, Chapel Hill, NC, 2001–2008.

S. S. Jones Distinguished Term Associate Professor of Computer Science, University of North Carolina at Chapel Hill, Department of Computer Science, Chapel Hill, NC, 2000.

Associate Professor, University of North Carolina at Chapel Hill, Department of Computer Science, Chapel Hill, NC, 1996–2000.

Visiting Professor, Carnegie Mellon University, School of Computer Science, Pittsburgh, PA, 1994.

Assistant Professor, University of North Carolina at Chapel Hill, Department of Computer Science, Chapel Hill, NC, 1989–1995.

Honors and Awards

Alumni Achievement Award, University of Washington, Department of Computer Science and Engineering, June 2015.

Excellence in Teaching Award, Computer Science Student Association, University of North Carolina at Chapel Hill, Department of Computer Science, May 2011.

Favorite Faculty Award, an award given by the Computer Science majors of the 2010 graduating class, May 2010.

Carolina Women's Center Women's Advocacy Award, University of North Carolina at Chapel Hill, Chapel Hill, NC, March 2010.

Gillian T. Cell Distinguished Professor in Computer Science, University of North Carolina at Chapel Hill, College of Arts and Sciences, Chapel Hill, NC, June 2008.

Outstanding Teaching Award, an award given by the Computer Science majors of the 2008 graduating class, May 2008.

Favorite Faculty Award, an award given by the Computer Science majors of the 2004 graduating class, May 2004.

Edward Kidder Graham Outstanding Faculty Award, an award given by the Class of 2004 and the General Alumni Association, April 2004.

Edward Kidder Graham Advisor of the Year Award, an award given by the Class of 2004 and the General Alumni Association, April 2004.

IEEE Service Award, Service to the IEEE Technical Committee on Real-Time Systems, December 2001.

Outstanding Teaching Award, Computer Science Student Association, University of North Carolina at Chapel Hill, Department of Computer Science, May 2000.

S. Shepard Jones Distinguished Term Professorship, University of North Carolina at Chapel Hill, College of Arts and Sciences, Chapel Hill, NC, March 1999.

Award Papers:

ACM SIGCOMM 2000, 2003.

IEEE International Symposium on High Assurance Systems Engineering 1999, 2000.

IEEE Real-Time Technology and Applications Symposium, 1995.

ACM International Conference on Multimedia, 1994.

Computer Networks and ISDN Systems, 1994.

International Workshop on Network and Operating System Support for Digital Audio and Video, 1991, 1992, 1993, 1997.

Testifying Experience (Within Last Four Years)

In re: Inter Partes review of U.S. Patent Nos. 7,710,978, 7,773,588 & 8,560,828.

Provided deposition testimony on behalf of Patent Owner directPacket Research, Inc.

In the Matter of: Certain Audio Players and Controllers, Components Thereof, and Products Containing the Same, United States International Trade Commission, Investigation No. 337-TA-1191.

Provided deposition testimony on behalf of respondent Alphabet Inc. and Google LLC.

Centripetal Networks, Inc., v. Cisco Systems, Inc., United States District Court for the Eastern District of Virginia, Norfolk Division, Case No. 2:18-cv-00094-HCM-LRL.

Provided deposition and trial testimony on behalf of defendant Cisco Systems, Inc.

In re: Inter Partes review of U.S. Patent Nos. 8,064,935 & 9,602,655.

Provided deposition testimony on behalf of Petitioner Resideo Technologies, Inc.

In the Matter of: Certain Wireless Communication Devices and Related Components Thereof, United States International Trade Commission, Investigation No. 337-TA-1180.

Provided deposition testimony on behalf of respondent HTC Corporation and HTC America, Inc.

Gummarus LLC, v. Samsung Electronics Co., Ltd. and Samsung Electronics America, Inc., United States District Court for the Eastern District of Texas, Sherman Division, Case No. 4:19-cv-00251.

Provided deposition testimony on behalf of defendant Samsung Electronics Co., Ltd. and Samsung Electronics America, Inc.

Uniloc 2017, LLC, v. Netflix, Inc., United States District Court for the Central District of California, Los Angeles Division, Case No. 8:18-cv-02055-JVS-DFM.

Provided a tutorial at a Markman Hearing on behalf of defendant Netflix, Inc.

directPacket Research, Inc. v. Polycom, Inc., United States District Court for the Northern District of California, San Jose Division, Case No. 5:19-cv-03918-LHK.

Provided deposition testimony on behalf of plaintiff directPacket Research, Inc.

Symantec Corp. and Symantec Limited v. Zscalar, Inc., United States District Court for the Northern District of California, San Francisco Division, Case No. 3:17-cv-04426-JST.

Provided deposition testimony on behalf of defendant Zscalar, Inc.

In re: Inter Partes review of U.S. Patent Nos. 8,631,120 & 9,148,493.

Provided deposition testimony on behalf of Petitioner Avi Networks, Inc.

Finite State Machine Labs Inc., v. Spectracom Corporation, Orolia USA, Inc., and Orolia S.A., United States District Court for the Western District of Texas, Austin Division, Case Nos. 1:17-cv-01040-SS.

Provided deposition testimony on behalf of defendants Spectracom Corporation, Orolia USA, Inc. and Orolia S.A.

Implicit LLC, v. NetScout Systems Inc. and Sandvine Corporation, United States District Court for the Eastern District of Texas, Marshal Division, Case Nos. 2:18-cv-0053-JRG and 2:18-cv-0054-JRG.

Provided deposition testimony on behalf of defendants NetScout Systems Inc. and Sandvine Corporation, and trial testimony on behalf of defendant NetScout Systems Inc.

In re: Inter Partes review of U.S. Patent No. 7,844,037.

Provided deposition testimony on behalf of Patent Owner Qualcomm Inc.

In re: Inter Partes review of U.S. Patent No. 9,203,940.

Provided deposition testimony on behalf of Patent Owner Qualcomm Inc.

Koninklijke Philips N.V. and U.S. Philips Corp., v. HTC Corp. and HTC America, Inc., United States District Court for the Northern District of California Oakland Division, Case No. 4:18-01885-HSG.

Provided deposition testimony on behalf of defendant HTC Corp. and HTC America, Inc.

In re: Inter Partes review of U.S. Patent No. 8,508,607.

Provided deposition testimony on behalf of Petitioner Axis Communications AB.

In re: Inter Partes review of U.S. Patent No. 9,560,077.

Provided deposition testimony on behalf of Petitioner Cisco Systems, Inc.

In re: Inter Partes review of U.S. Patent No. 9,137,205.

Provided deposition testimony on behalf of Petitioner Cisco Systems, Inc.

In re: Inter Partes review of U.S. Patent No. 9,565,213.

Provided deposition testimony on behalf of Petitioner Cisco Systems, Inc.

Egenera, Inc., v. Cisco Systems, Inc., United States District Court for the District of Massachusetts, Case No. 1-16-cv-11613-RGS.

Provided deposition and summary judgement hearing testimony on behalf of defendant Cisco Systems, Inc.

Symantec Corp. and Symantec Limited v. Zscalar, Inc., United States District Court for the Northern District of California, San Francisco Division, Case No. 3:17-cv-04414-JST.

Provided deposition testimony on behalf of defendant Zscalar, Inc.

SEVEN Networks, LLC v. Google LLC, United States District Court for the Eastern District of Texas, Marshall Division, Case No. 2:17-CV-00442-JRG.

Provided deposition testimony on behalf of defendant Google LLC.

Qualcomm Inc. v. Apple Inc., United States District Court for the Southern District of California, Case No. 3:17-cv-02403-CAB-MDD.

Provided deposition testimony on behalf of plaintiff Qualcomm, Inc.

S.I.SV.EL. Societa Italiana Per Lo Sviluppo Dell'Elettronica S.P.A., v. Spotify USA, Inc. et al., United States District Court for the District of Delaware, C.A. No. 18-68-VAC-CJB, 18-69-VAC-CJB, 18-70-VAC-CJB.

Provided deposition testimony on behalf of defendants Spotify USA, Inc., Rakuten Kobo, Inc.,

Rhapsody International, Inc.

Meetrix IP, LLC, v. Citrix Systems, Inc., GetGo, Inc., & LogMeIn, Inc., United States District Court for the Western District of Texas, Austin Division, Case No. 1:16-CV-1033-LY.

Provided a tutorial at a Markman Hearing on behalf of defendants LogMeIn, *et al*.

Intellectual Ventures II, LLC, v. AT&T Corp., et al., United States District Court for the Western District of Texas, Austin Division, Case No. 1:13-CV-00116-LY.

Provided deposition testimony on behalf of defendant AT&T Corp.

Industry Experience

Consultant, Mälardalen University, Vásteras, Sweden, 2001.

Member, Technical Advisory Board, Ganymede Software Inc., Research Triangle Park, NC, 1996–1999.

Consultant and Author, INTEROP Graduate Institute, SOFTBANK Inc., Foster City, CA, 1996–1997.

Consultant, National Science Foundation, Arlington, VA, 1995–present.

Consultant, Monterey Technologies Inc., Cary, NC, 1994–1996.

Consultant, Hewlett-Packard Inc., Ink Jet Components Division, Corvallis, OR, 1993–1994.

Visiting Researcher, IBM T.J. Watson Research Center, Computer Systems Principles Group, Yorktown Heights, NY, 1986.

Consultant, Boeing Aerospace, Kent, WA, 1985–1986.

Software Engineer, R.R. Donnelley & Sons Company Inc., Technical Center, Chicago, IL, 1984.

Computer Programmer, US Army Corps of Engineers Construction Engineering Research Lab, Champaign, IL, 1981–1982.

Community Service

Consultant (unpaid), Children's Museum About the World (now called *Exploris*), Raleigh, NC, 1995–1996.

Professional Activities

Editor

Associate Editor, *Real-Time Systems*, Kluwer Academic Publishers, The Netherlands, 2003-2012.

Editor in Chief, *Multimedia Systems*, ACM/Springer-Verlag, Heidelberg, Germany, 2000–2001.

Editorial Board

Journal of Multimedia Tools and Applications, Kluwer Academic Publishers, 1994–1999.

Guest Editor

Computer Communications, special issue on system support for multimedia computing, Volume 18, Number 10, October 1995.

Executive Committees

ACM/SIGCOMM Internet Measurement Conference Steering Committee, 2005-2009.

College of Reviewers, Canada Research Chairs Program, 2004-present.

Statistical and Applied Mathematical Sciences Institute (SAMSI) program on Network Modeling for the Internet, 2002-2004.

IEEE Technical Committee on Real-Time Systems, 2000-present.

ACM Special Interest Group on Multimedia, 2000-2002.

Other Committees

IEEE Distinguished Visitors Program, 2004–2007.

Conference Program Chair

11th International Workshop on Quality-of-Service (Co-Chair), Monterey, CA, June 2003.

Sixth IEEE Symposium on Computers and Communications, Hammamet, Tunisia, July 2001.

21st IEEE Real-Time Systems Symposium, Orlando, FL, November 2000.

Tenth International Workshop on Network and Operating System Support for Digital Audio and Video, Chapel Hill, NC, June 2000.

Seventh ACM International Conference on Multimedia (Co-Chair), Orlando, FL, November 1999.

SPIE/ACM Multimedia Computing and Networking 1999 (Co-Chair), San Jose, CA, January 1999.

Sixth ACM International Conference on Multimedia (Associate Chair), Bristol, UK, September 1998.

IEEE International Conference on Multimedia Computing Systems (Associate Chair), Austin, TX, June 1998.

SPIE/ACM Multimedia Computing and Networking 1998 (Co-Chair), San Jose, CA, January 1998.

SPIE/ACM Multimedia Computing and Networking 1997 (Associate Chair), San Jose, CA, February 1997.

IEEE Workshop on Resource Allocation Problems in Multimedia Systems, Washington D.C., December 1996.

Second IEEE Real-Time Technology and Applications Symposium, Boston, MA, June 1996.

Conference General Chair

ACM SIGCOMM Internet Measurement Conference, San Diego, CA, October 2007.

22nd IEEE Real-Time Systems Symposium, London, UK, December 2001.

Tenth International Workshop on Network and Operating System Support for Digital Audio and Video, Chapel Hill, NC, June 2000.

Third IEEE Real-Time Technology and Applications Symposium, Montreal, Canada, June 1997.

IEEE Workshop on Resource Allocation Problems in Multimedia Systems, Washington D.C., December 1996.

Member of Conference Program Committee

21st IEEE International Conference on Network Protocols, Goettingen, Germany, October 2013.

18th IEEE International Conference on Network Protocols, Kyoto, Japan, October 2010.

ACM SIGCOMM 2010, New Delhi, India, August 2010.

17th IEEE International Conference on Network Protocols, Princeton, NJ, October 2009.

9th Passive and Active Measurement Conference 2008, Cleveland, OH, April 2008.

ACM SIGMETRICS 2008, Annapolis, MA, June 2008.

IEEE INFOCOM 2008, Phoenix, AZ, April 2008.

6th ACM SIGCOMM Workshop on Hot Topics in Networks, Atlanta, GA, November 2007.

15th IEEE International Conference on Network Protocols, Beijing, China, October 2007.

5th IEEE Workshop on Embedded Systems for Real-Time Multimedia, Salzburg, Austria, October 2007.

Workshop on Experimental Computer Science, ACM FCRC, San Diego, CA, June 2007.

15th International Workshop on Quality-of-Service, Chicago, IL, June 2007.

17th International Workshop on Network and Operating System Support for Digital Audio and Video, Urbana-Champaign, IL, June 2007.

10th IEEE Global Internet Symposium 2007, Anchorage, AK, May 2007.

ACM SIGMETRICS 2007, San Diego, CA, June 2007.

14th IEEE International Conference on Network Protocols, Santa Barbara, CA, October 2006.

ACM Internet Measurement Conference 2006, Rio de Janeiro, Brazil, October 2006.

Second ACM SIGCOMM Conference on Future Networking Technologies, Lisbon, Portugal, December 2006.

16th International Workshop on Network and Operating System Support for Digital Audio and Video, Newport, RI, June 2006.

IEEE Workshop on Research Directions for Security and Networking in Critical Real-Time and Embedded Systems, San Jose, CA, April 2006.

IEEE INFOCOM 2006, Barcelona, Spain, April 2006.

13th IEEE International Conference on Network Protocols, Boston, MA, November 2005.

15th International Workshop on Network and Operating System Support for Digital Audio and Video, Skamania, WA, June 2005.

IEEE INFOCOM 2005, Miami, FL, March 2005.

SPIE/ACM Multimedia Computing and Networking 2005, San Jose, CA, January 2005.

ACM Multimedia 2004, New York, NY, October 2004.

ACM Internet Measurement Conference 2004, Taormina, Italy, October 2004.

16th EUROMICRO Conference on Real-Time Systems, Catania, Italy, June-July 2004.

14th International Workshop on Network and Operating System Support for Digital Audio and Video, Cork, Ireland, June 2004.

12th International Workshop on Quality-of-Service, Montreal, Canada, June 2004.

10th IEEE Real-time and Embedded Technology and Applications Symposium, Toronto, Canada, May 2004.

IEEE INFOCOM 2004, Hong Kong, March 2004.

24nd IEEE Real-Time Systems Symposium, Cancun, Mexico, December 2003.

19th ACM Symposium on Operating Systems Principles, Lake George, New York, October 2003.

15th EUROMICRO Conference on Real-Time Systems, Porto, Portugal, July 2003.

Ninth IEEE Real-Time and Embedded Technology and Applications Symposium, Toronto, CA, May 2003.

SPIE/ACM Multimedia Computing and Networking 2003, San Jose, CA, January 2003.

23nd IEEE Real-Time Systems Symposium, Austin, TX, November 2002.

10th International Conference on Network Protocols, Paris, France, November 2002.

Second Workshop on Embedded Software, Grenoble, France, October 2002.

22nd IFIP International Symposium on Computer Performance Modeling, Measurement and Evaluation (Performance 2002), Rome, Italy, September 2002.

Eighth IEEE Real-Time and Embedded Technology and Applications Symposium, San Jose, CA, September 2002.

10th International Workshop on Quality-of-Service, Miami, FL, June 2002.

Seventh Global Internet Symposium (held in conjunction with Globecom 2002), Taipei, Taiwan, November 2002.

14th EUROMICRO Conference on Real-Time Systems, Vienna, Austria, June 2002.

ACM SIGMETRICS 2002, Marina del Rey, CA, June 2002.

SPIE/ACM Multimedia Computing and Networking 2002, San Jose, CA, January 2002.

22nd IEEE Real-Time Systems Symposium, London, UK, December 2001.

Sixth Global Internet Symposium (held in conjunction with Globecom 2001), San Antonio, TX, November 2001.

27th EUROMICRO Conference, Warsaw, Poland, September 2001.

Sixth IEEE Symposium on Computers and Communications, Hammamet, Tunisia, July 2001.

Seventh IEEE Real-Time Technology and Applications Symposium, Taipei, Taiwan, June 2001.

Ninth IFIP International Workshop on Quality of Service, Karlsruhe, Germany, June 2001.

SPIE/ACM Multimedia Computing and Networking 2001, San Jose, CA, January 2001.

Fifth Global Internet Mini-Conference (held in conjunction with Globecom 2000), San Francisco, CA, November 2000.

21st IEEE Real-Time Systems Symposium, Orlando, FL, December 2000.

Tenth International Workshop on Network and Operating System Support for Digital Audio and Video, Chapel Hill, NC, June 2000.

Sixth IEEE Real-Time Technology and Applications Symposium, Washington, D.C., June 2000.

SPIE/ACM Multimedia Computing and Networking 2000, San Jose, CA, January 2000.

19th IEEE Real-Time Systems Symposium, Phoenix, AZ, December 1999.

Fourth Global Internet Mini-Conference (held in conjunction with Globecom '99), Rio de Janero, Brazil, November 1999.

Fifth International Workshop on Multimedia Information Systems, Palm Springs, CA, October 1999.

Ninth International Workshop on Network and Operating System Support for Digital Audio and Video, Basking Ridge, NJ, June 1999.

IEEE Workshop on QoS Support for Real-Time Internet Applications, Vancouver, Canada, June, 1999.

Second ACM Workshop on Internet Server Performance (held in conjunction with SIGMETRICS '99), Atlanta, GA, May 1999.

Seventh IEEE International Workshop on Parallel and Distributed Real-Time Systems, San Juan, Puerto Rico, April 1999.

SPIE/ACM Multimedia Computing and Networking 1998, San Jose, CA, January 1999.

19th IEEE Real-Time Systems Symposium, Madrid, Spain, December 1998.

Third Global Internet Mini-Conference (held in conjunction with Globecom '98), Sydney, Australia November 1998.

Sixth ACM International Conference on Multimedia, Bristol, UK, September 1998.

Third IEEE/ACM International Workshop on Multimedia Database Management Systems, Dayton, OH, August 1998.

Eighth International Workshop on Network and Operating System Support for Digital Audio and Video, Cambridge, UK, July 1998.

SPIE Interactive Multimedia Services and Equipment, Zurich, Switzerland, May 1998.

IEEE Workshop on Dependable and Real-Time E-Commerce Systems, Denver, CO, June 1998.

Fourth IEEE Real-Time Technology and Applications Symposium, Denver, CO, June 1998.

IEEE International Conference on Multimedia Computing Systems, Austin, TX, June 1998.

SPIE/ACM Multimedia Computing and Networking 1998, San Jose, CA, January 1998.

16th IEEE Symposium on Reliable Distributed Systems, Durham, NC, October 1997.

Third IEEE Real-Time Technology and Applications Symposium, Montreal, Canada, June 1997.

IEEE Real-Time Education Workshop, Montreal, Canada, June 1997.

Seventh International Workshop on Network and Operating System Support for Digital Audio and Video, St. Louis, MO, May 1997.

17th International Conference on Distributed Computing Systems, Distributed Real-Time Systems Track, Performance of Distributed Systems Track, Distributed Multimedia Track, Baltimore, MD, May 1997.

Fifth IFIP International Workshop on Quality of Service, New York, NY, May 1997.

Third International Conference on Computer Science & Informatics, Raleigh-Durham, NC, March 1997.

SPIE/ACM Multimedia Computing and Networking 1997, San Jose, CA, February 1997.

Fourth ACM International Conference on Multimedia, Boston, MA, November 1996.

Second USENIX Symposium on Operating Systems Design and Implementation, Seattle, WA, October/November 1996.

ACM SIGSOFT '96: Fourth Symposium on the Foundations of Software Engineering, San Francisco, CA, October 1996.

Second IEEE International Workshop on Multimedia Database Management Systems, Blue Mountain Lake, NY, August 1996.

Second IEEE Real-Time Technology and Applications Symposium, Boston, MA, June 1996.

Sixth International Workshop on Network and Operating System Support for Digital Audio and Video, Zushi, Japan, April 1996.

First International Workshop on Real-Time Databases: Issues and Applications, Newport Beach, CA, March 1996.

SPIE Multimedia Computing and Networking 1996, San Jose, CA, February 1996.

16th IEEE Real-Time Systems Symposium, Pisa, Italy, December 1995.

Third ACM International Conference on Multimedia, Technical Paper Program, San Francisco, CA, November 1995.

Third ACM International Conference on Multimedia, Video Program, San Francisco, CA, November 1995.

1995 ACM Conference on Organizational Computing Systems, Technical Track, Milpitas, CA, August 1995.

Fifth IEEE Workshop on Hot Topics in Operating Systems, Orcas Island, WA, May 1995.

15th International Conference on Distributed Computing Systems, Distributed Real-Time Systems Track and CSCW track, Vancouver, British Columbia, Canada, May 1995.

Fifth International Workshop on Network and Operating System Support for Digital Audio and Video, Durham, NH, April 1995.

IEEE Workshop on the Role of Real-Time in Multimedia, Research Triangle Park, NC, December 1993.

14th IEEE Real-Time Systems Symposium, Research Triangle Park, NC, December 1993.

13th International Conference on Distributed Computing Systems, Real-Time Issues Track, Pittsburgh, PA, May 1993.

12th IEEE Real-Time Systems Symposium, San Antonio, TX, December 1991.

IEEE Conference on Communication Software: Communications for Distributed Applications and Systems, Chapel Hill, NC, April 1991.

Conference Organizing Committees

Finance Chair, IEEE Real-Time Systems Symposium, Miami, FL, December 2005.

Finance Chair, IEEE Real-Time Systems Symposium, Lisbon, Portugal, December 2004.

Finance Chair, IEEE Real-Time Systems Symposium, Cancun, Mexico, December 2003.

Poster Session Chair, ACM Symposium on Operating System Principles, Bolton Landing, NY, October 2003.

Demonstrations Chair, ACM Conference on Computer-Supported Cooperative Work, Research Triangle Park, NC, November 1994.

Wine Steward, ACM Symposium on Operating System Principles, Asheville, NC, December 1993.

Local Arrangements Chair, 14th IEEE Symposium on Real-Time Systems, Durham, NC, December 1993.

Refereed Publications

Technical Papers

Explicit Topology Management for Continental-Scale Airborne Networks, B. Newton, J. Aikat, K. Jeffay, IEEE INFOCOM Workshop on Wireless Communications and Networking in Extreme Environments, Atlanta, GA, May 2017, pages 72-77.

Scientific Training in the Era of Big Data: A New Pedagogy for Graduate Education, J. Aikat, T. Carsey, K. Fecho, K. Jeffay, A. Krishnamurthy, P.J. Mucha, A. Rajasekar, S. Ahalt, Big Data, Volume 5, Number 1, 2017, pages 12-18.

Geographic Routing in Extreme-Scale Highly-Dynamic Mobile Ad hoc Networks, B. Newton, J. Aikat, K. Jeffay, Proceedings of the 24th IEEE/ACM International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), London, UK, September 2016, pages 205-210.

Simulating Large-Scale Airborne Networks with ns-3, B. Newton, J. Aikat, K. Jeffay, Proceedings of the 2015 ACM Workshop on ns-3, Barcelona, Spain, May 2015, pages 32-39.

Explicit Topology Control for Airborne Networks, B. Newton, K. Jeffay, J. Aikat, Proceedings of the 22nd IEEE International Conference on Network Protocols, Research Triangle, NC, October 2014, pages 368-373.

The Continued Evolution of Web Traffic, B. Newton, K. Jeffay, J. Aikat, Proceedings of the 21st IEEE/ACM International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), San Francisco, CA, August 2013, pages 80-89.

Gini in a Bottle: A Case Study of Pareto's Principle in the Wild, A. Blate, K. Jeffay, International Journal of Computer Networks and Communications Security, Vol. 1, No. 1, June 2013, pages 30-39.

Experiment Replication Using ProtoGENI Nodes, D. O'Neill, J. Aikat, K. Jeffay, Proceedings of the Second GENI Research and Educational Experiment Workshop, Salt Lake City, UT, March 2013, pages 9-15.

Towards Traffic Benchmarks for Empirical Networking Research: The Role of Connection Structure in Traffic Workload Modeling, J. Aikat, S. Hasan, K. Jeffay, F.D. Smith, Proceedings of the 20th IEEE/ACM International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Arlington, VA, August 2012, pages 78-86.

Generating Realistic Synthetic TCP Application Workloads, J. Aikat, K. Jeffay, F.D. Smith, Proceedings, GENI Infrastructure Workshop, Princeton, NJ, June 2010, 8 pages.

Correlations of Size, Rate, and Duration in TCP Connections: The Case Against, C. Park, F. Hernández-

Campos, J.S. Marron, K. Jeffay, F.D. Smith, Annals of Applied Statistics, Volume 4, Number 1, May 2010, pages 26-52.

Passive, Streaming Inference of TCP Connection Structure for Network Server Management, J. Terrell, K. Jeffay, F.D. Smith, J. Gogan, J. Keller, IFIP International Workshop on Traffic Monitoring and Analysis, Aachen, Germany, May 2009, in, Lecture Notes in Computer Science, Volume 5537, Springer, Berlin, Germany, pages 42-53.

Exposing Server Performance to Network Managers Through Passive Network Measurements, J. Terrell, K. Jeffay, F.D. Smith, J. Gogan, J. Keller, IEEE Internet Network Management Workshop 2008, Orlando, FL, October 2008, pages 1-6.

Multi-Resolution Anomaly Detection for the Internet, L. Zhang, Z. Zhu, K. Jeffay, J.S. Marron, F.D. Smith, IEEE Workshop on Network Management, Phoenix, AZ, April 2008, 6 pages.

The Effects of Active Queue Management and Explicit Congestion Notification on Web Performance, L. Le, J. Aikat, K. Jeffay, F.D. Smith, IEEE/ACM Transactions on Networking, Volume 15, Number 6, December 2007, pages 1217-1230.

Co-Scheduling Variable Execution Time Requirement Real-time Tasks and Non-Real-Time Tasks, A. Singh, K. Jeffay, Proceedings of the 19th Euromicro Conference on Real-Time Systems, Pisa, Italy, July 2007, pages 191-200.

Quantifying the Effects of Recent Protocol Improvements to Standards-Track TCP: Impact on Web Performance, M.C. Weigle, K. Jeffay, F.D. Smith, Computer Communications, Volume 29, Number 15, September 2006, pages 2853-2866.

Tmix: A Tool for Generating Realistic TCP Application Workloads in ns-2, M.C. Weigle, P. Adurthi, F. Hernández-Campos, K. Jeffay, F.D. Smith, ACM Computer Communications Review, Volume 36, Number 3, July 2006, pages 67-76.

A Loss and Queuing-Delay Controller for Router Buffer Management, L. Le, K. Jeffay, F.D. Smith, Proceedings of the 26th IEEE International Conference on Distributed Computing Systems, Lisbon, Portugal, July 2006, 10 pages.

Understanding Patterns of TCP Connection Usage with Statistical Clustering, F. Hernández-Campos, A.B. Nobel, F.D. Smith, K. Jeffay, 13th IEEE/ACM International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Atlanta, GA, September 2005, pages 35-44.

Delay-Based Early Congestion Detection and Adaptation: Impact on Web Performance, M.C. Weigle, K. Jeffay, F.D. Smith, Computer Communications, Volume 8, Number 8, May 2005, pages 837-850.

Extremal Dependence: Internet Traffic Applications, F. Hernández-Campos, K. Jeffay, C. Park, J.S. Marron, S.I. Resnick, Stochastic Models, Volume 21, Number 1, 2005, pages 1-35.

Generating Realistic TCP Workloads, F. Hernández-Campos, F.D. Smith, K. Jeffay, Proceedings of the Computer Measurement Group's 2004 International Conference, Las Vegas, NV, December 2004, pages 273-284.

Differential Congestion Notification: Taming the Elephants, L. Le, J. Aikat, K. Jeffay, F.D. Smith, Proceedings of the 12th IEEE International Conference on Network Protocols, Berlin, Germany, October 2004, pages 118-128.

Stochastic Models for Generating Synthetic HTTP Source Traffic, J. Cao, W.S. Cleveland, Y. Gao, K. Jeffay, F.D. Smith M.C. Weigle, Proceedings of IEEE INFOCOM 2004, Hong Kong, March 2004,

Volume 3, pages 1546-1557.

Variability in TCP Roundtrip Times, J. Aikat, J. Kaur, D. Smith, K. Jeffay, Proceedings of the 2003 ACM SIGCOMM Internet Measurement Conference, Miami Beach, FL, October 2003, pages 279-284.

Tracking the Evolution of Web Traffic: 1995-2003, F. Hernández-Campos, K. Jeffay, F.D. Smith, Proceedings of the 11th IEEE/ACM International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Orlando, FL, October 2003, pages 16-25.

The Effects of Active Queue Management on Web Performance, L. Le, J. Aikat, K. Jeffay, F.D. Smith, Proceedings of ACM SIGCOMM 2003, Karlsruhe, Germany, August 2003, pages 265-276.

Managing Latency and Buffer Requirements in Processing Graph Chains, S.M. Goddard, K. Jeffay, The Computer Journal, Volume 44, Number 6, 2001, (special issue on high-assurance systems), pages 486-503.

Rate-Based Resource Allocation Methods for Embedded Systems, K. Jeffay, S.M. Goddard, in, *Embedded Software*, Proceedings of the First International Workshop on Embedded Software (*EMSOFT 2001*), Tahoe City, CA, October 2001, Lecture Notes in Computer Science, Volume 2211, T. Henzinger, C. Kirsch, editors, Springer-Verlag, Berlin, Germany, 2001, pages 204-222.

Beyond Audio and Video: Multimedia Networking Support for Distributed, Immersive Virtual Environments, K. Jeffay, T. Hudson, M. Parris, Proceedings of the 27th EUROMICRO Conference, Warsaw, Poland, September 2001, pages 300-307.

Tuning RED for Web Traffic, M. Christiansen, K. Jeffay, D. Ott, F.D. Smith, IEEE/ACM Transactions on Networking, Volume 9, Number 3, (June 2001), pages 249-264.

What TCP/IP Protocol Headers Can Tell Us About the Web, F.D. Smith, F. Hernández Campos, K. Jeffay, D. Ott, Proceedings of ACM SIGMETRICS 2001/Performance 2001, Cambridge, MA, June 2001, pages 245-256.

Experiments in Best-Effort Multimedia Networking for a Distributed Virtual Environment, T. Hudson, M.C. Weigle, K. Jeffay, R.M. Taylor II, in *Multimedia Computing and Networking 2001*, Proceedings, SPIE Proceedings Series, Volume 4312, San Jose, CA, January 2001, pages 88-98.

Analyzing the Real-Time Properties of a U.S. Navy Signal Processing System, S.M. Goddard, K. Jeffay, International Journal of Reliability, Quality and Safety Engineering, Volume 8, Number 4, December 2001, (special issue of HASE '99 best papers) pages 301-322.

A Comparative Study of the Realization of Rate-Based Computing Services in General Purpose Operating Systems, K. Jeffay, G. Lamastra, Proceedings of the Seventh IEEE International Conference on Real-Time Computing Systems and Applications, Cheju Island, South Korea, December 2000, pages 81-90.

The Synthesis of Real–Time Systems from Processing Graphs, S.M. Goddard, K. Jeffay, Proceedings of the Fifth IEEE International Symposium on High Assurance Systems Engineering, Albuquerque, NM, November 2000, pages 177-186.

Tuning RED for Web Traffic, M. Christiansen, K. Jeffay, D. Ott, F.D. Smith, Proceedings of ACM SIGCOMM 2000, Stockholm, Sweden, August-September 2000, pages 139-150.

Towards a Better-Than-Best-Effort Forwarding Service for Multimedia Flows, K. Jeffay, IEEE Multimedia, Volume 6, Number 4, October-December 1999, pages 84-88.

A Theory of Rate-Based Execution, K. Jeffay, S.M. Goddard, Proceedings of the 20th IEEE Real-Time

Systems Symposium, Phoenix, AZ, December 1999, pages 304-314.

Parallel Switching in Connection-Oriented Networks, S. Baruah, K. Jeffay, J. Anderson, Proceedings of the 20th IEEE Real-Time Systems Symposium, Phoenix, AZ, December 1999, pages 200-209.

Analyzing the Real-Time Properties of a U.S. Navy Signal Processing System, S.M. Goddard, K. Jeffay, Proceedings of the Fourth IEEE International Symposium on High Assurance Systems Engineering, Washington, DC, November 1999, pages 141-150.

Application-Level Measurements of Performance on the vBNS, M. Clark, K. Jeffay, Proceedings of the IEEE International Conference on Multimedia Computing and Systems, Volume 2, Florence, Italy, June 1999, pages 362-366.

Lightweight Active Router-Queue Management for Multimedia Networking, M. Parris, K. Jeffay, F.D. Smith, in *Multimedia Computing and Networking 1999*, Proceedings, SPIE Proceedings Series, Volume 3654, San Jose, CA, January 1999, pages 162-174.

Proportional Share Scheduling of Operating System Services for Real-Time Applications, K. Jeffay, F.D. Smith, A. Moorthy, J.H. Anderson, Proceedings of the 19th IEEE Real-Time Systems Symposium, Madrid, Spain, December 1998, pages 480-491.

Efficient Object Sharing in Quantum-Based Real-Time Systems, J.H. Anderson, R. Jain, K. Jeffay, Proceedings of the 19th IEEE Real-Time Systems Symposium, Madrid, Spain, December 1998, pages 346-355.

A Better-Than-Best-Effort Service for Continuous Media UDP Flows, M. Parris, K. Jeffay, F.D. Smith, J. Borgersen, Proceedings of the Eighth International Workshop on Network and Operating System Support for Digital Audio and Video, Cambridge, UK, July 1998, pages 193-197.

Managing Memory Requirements in the Synthesis of Real-Time Systems from Processing Graphs, S. Goddard, K. Jeffay, Proceedings of the Fourth IEEE Real-Time Technology and Applications Symposium, Denver, CO, June 1998, pages 59-70.

Fair On-Line Scheduling of a Dynamic Set of Tasks on a Single Resource, S.K. Baruah, J.E. Gehrke, C.G. Plaxton, I. Stoica, H. Abdel-Wahab, K. Jeffay, *Information Processing Letters*, Volume 64, Number 1, October 1997, pages 43-51.

A Two-Dimensional Audio Scaling Enhancement to an Internet Videoconferencing System, P. Nee, K. Jeffay, M. Clark, G. Danneels, Proceedings of the International Workshop on Audio-Visual Services over Packet Networks, Aberdeen, Scotland, UK, September 1997, pages 201-206.

Feasibility Concerns in PGM Graphs With Bounded Buffers, S. Baruah, S. Goddard, K. Jeffay, Proceedings of the Third IEEE International Conference on Engineering of Complex Computer Systems, Como, Italy, September 1997, pages 130-139.

Analyzing the Real-Time Properties of a Dataflow Execution Paradigm Using a Synthetic Aperture Radar Application, S. Goddard, K. Jeffay, Proceedings of the Third IEEE Real-Time Technology and Applications Symposium, Montreal, Canada, June 1997, pages 60-71.

Real-Time Computing with Lock-Free Shared Objects, J.H. Anderson, S. Ramamurthy, K. Jeffay, ACM Transactions on Computing Systems, Volume 15, Number 2, May 1997, pages 134-165.

The Performance of Two-Dimensional Media Scaling for Internet Videoconferencing, P. Nee, K. Jeffay, G. Danneels, Proceedings of the Seventh International Workshop on Network and Operating System Support for Digital Audio and Video, St. Louis, MO, May 1997, pages 237-248. Republished in "Readings in Multimedia Computing," K. Jeffay, H.J. Zhang, editors, Morgan Kaufmann, 2002.

On the Duality between Resource Reservation and Proportional Share Resource Allocation, I. Stoica, H. Abdel-Wahab, K. Jeffay, in *Multimedia Computing and Networking 1997*, Proceedings, SPIE Proceedings Series, Volume 3020, San Jose, CA, February 1997, pages 207-214.

Strategic Directions in Real-Time and Embedded Systems, J.A. Stankovic, A. Burns, K. Jeffay, M. Jones, G. Koob, I. Lee, J. Lehoczky, J. Liu, A. Mok, K. Ramamritham, J. Ready, L. Sha, A. van Tilborg, *ACM Computing Surveys*, Volume 28, Number 4, December 1996, pages 751-763.

A Proportional Share Resource Allocation Algorithm For Real-Time, Time-Shared Systems, I. Stoica, H. Abdel-Wahab, K. Jeffay, S.K. Baruah, J.E. Gehrke, C.G. Plaxton, Proceedings of the 17th IEEE Real-Time Systems Symposium, Washington, DC, December 1996, pages 288-299.

A General Framework For Continuous Media Transmission Control, T.M. Talley, K. Jeffay, Proceedings of the 21st IEEE Conference on Local Computer Networks, Minneapolis, MN, October 1996, pages 374-383.

A Router-Based Congestion Control Scheme For Real-Time Continuous Media, K. Jeffay, M. Parris, T. Talley, F.D. Smith, Proceedings of the Sixth International Workshop on Network and Operating System Support for Digital Audio and Video, Zushi, Japan, April 1996, pages 79-86.

Lock-Free Transactions for Real-time Systems, J.H. Anderson, S. Ramamurthy, M. Moir, K. Jeffay, Proceedings of the First Workshop on Real-Time Databases: Issues and Applications, Newport Beach, CA, March 1996, pages 107-114.

Real-Time Computing with Lock-Free Shared Objects, J.H. Anderson, S. Ramamurthy, K. Jeffay, Proceedings of the 16th IEEE Real-Time Systems Symposium, Pisa, Italy, December 1995, pages 28-37.

Early Experience with the Repository for Patterned Injury Data, D. Stotts, J.B. Smith, K. Jeffay, P. Dewan, D.K. Smith, W. Oliver, Proceedings of the SPIE International Symposium on Investigative and Trial Image Processing, San Diego, CA, July 1995, SPIE Volume 2567, 1995, pages 249-260.

Early Prototypes of the Repository for Patterned Injury Data, P. Dewan, K. Jeffay, J. Smith, D. Stotts, W. Oliver, Proceedings of Digital Libraries '95, The Second Annual Conference on the Theory and Practice of Digital Libraries, Austin, TX, June 1995, pages 123-130.

Support For Real-Time Computing Within General Purpose Operating Systems: Supporting co-resident operating systems, G. Bollella, K. Jeffay, Proceedings of the IEEE Real-Time Technology and Applications Symposium, Chicago, IL, May 1995, pages 4-14.

A Rate-Based Execution Abstraction For Multimedia Computing, K. Jeffay, D. Bennett, Proceedings of the Fifth International Workshop on Network and Operating System Support for Digital Audio and Video, Durham, NH, April 1995, published in *Lecture Notes in Computer Science*, T.D.C. Little, R. Gusella, editors, Volume 1018, pages 64-75, Springer-Verlag, Heidelberg, Germany, 1995.

An Empirical Study of Delay Jitter Management Policies, D.L. Stone, K. Jeffay, *ACM Multimedia Systems*, Volume 2, Number 6, January 1995, pages 267-279. Republished in "Readings in Multimedia Computing," K. Jeffay, H.J. Zhang, editors, Morgan Kaufmann, 2002.

On the Partitioning of Function in Distributed Synchronous Collaboration Systems, J. Menges, K. Jeffay, ACM CSCW '94, Proceedings of the Workshop on Distributed Systems, Multimedia, and Infrastructure Support in CSCW, Research Triangle Park, NC, October 1994, SIGOIS Bulletin, Volume 15, Number 2, December 1994, pages 34-37.

Two-Dimensional Scaling Techniques For Adaptive, Rate-Based Transmission Control of Live Audio and Video Streams, T.M. Talley, K. Jeffay, Proceedings of the Second ACM International Conference on Multimedia, San Francisco, CA, October 1994, pages 247-254.

Transport and Display Mechanisms For Multimedia Conferencing Across Packet-Switched Networks, K. Jeffay, D.L. Stone, F.D. Smith, *Computer Networks and ISDN Systems*, Volume 26, Number 10, July 1994, pages 1281-1304. Republished in "A Guided Tour of Multimedia Systems and Applications," B. Furht, M. Milenkovic, editors, IEEE Computer Society Press, 1995.

A Patterned Injury Digital Library for Collaborative Forensic Medicine, D. Stotts, J.B. Smith, P. Dewan, K. Jeffay, F.D. Smith, D. Smith, S. Weiss, J. Coggins, W. Oliver, Proceedings of Digital Libraries '94, The First Annual Conference on the Theory and Practice of Digital Libraries, College Station, TX, June 1994, pages 25-33.

The Artifact-Based Collaboration System: An infrastructure for supporting and studying collaboration, K. Jeffay, J.B. Smith, F.D. Smith, D.E. Shackelford, J. Menges, Proceedings of the 15th Interdisciplinary Workshop on Informatics and Psychology, Schärding, Austria, May 1994, 27 pages.

On Latency Management in Time-Shared Operating Systems, K. Jeffay, Proceedings of the 11th IEEE Workshop on Real-Time Operating Systems and Software, Seattle, WA, May 1994, pages 86-90.

Inverting X: An Architecture for a Shared Distributed Window System, J. Menges, K. Jeffay, Proceedings of the Third Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, Morgantown, WV, April 1994, IEEE Computer Society Press, pages 53-64.

Issues, Problems, and Solutions in Sharing X Clients on Multiple Displays, H. Abdel-Wahab, K. Jeffay, *Internetworking — Research and Practice*, Volume 5, Number 1, March 1994, pages 1-15.

Dynamic Participation in a Computer-based Conferencing System, G. Chung, K. Jeffay, H. Abdel-Wahab, *Computer Communications*, Volume 17, Number 1, January 1994, pages 7-16.

Accounting for Interrupt Handling Costs in Dynamic Priority Task Systems, K. Jeffay, D.L. Stone, Proceedings of the 14th IEEE Real-Time Systems Symposium, Raleigh-Durham, NC, December 1993, pages 212-221.

Queue Monitoring: A Delay Jitter Management Policy, D.L. Stone, K. Jeffay, Proceedings of the Fourth International Workshop on Network and Operating System Support for Digital Audio and Video, Lancaster, UK, November 1993, published in *Lecture Notes in Computer Science*, D. Shepherd, G. Blair, G. Coulson, N. Davies, F. Garcia, editors, Volume 846, pages 149-160, Springer-Verlag, Heidelberg, Germany, 1994.

The Real-Time Producer/Consumer Paradigm: A paradigm for the construction of efficient, predictable real-time systems, K. Jeffay, Proceedings of the 1993 ACM/SIGAPP Symposium on Applied Computing, Indianapolis, IN, ACM Press, February 1993, pages 796-804.

Accommodating Late-Comers in Shared Window Systems, G. Chung, K. Jeffay, H. Abdel-Wahab, *IEEE Computer*, Volume 26, Number 1, January 1993, pages 72-74.

Scheduling Sporadic Tasks with Shared Resources in Hard-Real-Time Systems, K. Jeffay, Proceedings of the 13th IEEE Real-Time Systems Symposium, Phoenix, AZ, December 1992, pages 89-99.

Adaptive, Best-Effort, Delivery of Audio and Video Data Across Packet-Switched Networks, K. Jeffay, D.L. Stone, T. Talley, F.D. Smith, Proceedings of the Third International Workshop on Network and Operating System Support for Digital Audio and Video, La Jolla, CA, November 1992, published in *Lecture Notes in Computer Science*, V. Rangan, editor, Volume 712, pages 3-14, Springer-Verlag, Heidelberg, Germany, 1993.

Architecture of the Artifact-Based Collaboration System Matrix, K. Jeffay, J.K. Lin, J. Menges, F.D. Smith, J.B. Smith, ACM CSCW '92, Proceedings of the Conference on Computer-Supported Cooperative Work, Toronto, Canada, ACM Press, November 1992, pages 195-202.

Client/Server Protocol Filtering: An infrastructure for supporting group collaborations, K. Jeffay, J. Menges, J.-K. Lin, ACM CSCW '92, Proceedings of the Workshop on Tools and Technologies, Toronto, Canada, November 1992, pages 96-99.

Kernel Support for Live Digital Audio and Video, K. Jeffay, D.L. Stone, F.D. Smith, *Computer Communications*, Volume 15, Number 6, July/August 1992, pages 388-395.

On Kernel Support for Real-Time Multimedia Applications, K. Jeffay, Proceedings of the Third IEEE Workshop on Workstation Operating Systems, Key Biscayne, FL, April 1992, pages 39-46.

On Non-Preemptive Scheduling of Periodic and Sporadic Tasks, K. Jeffay, D.F. Stanat, C.U. Martel, Proceedings of the Twelfth IEEE Real-Time Systems Symposium, San Antonio, TX, December 1991, pages 129-139.

Kernel Support for Live Digital Audio and Video, K. Jeffay, D.L. Stone, F.D. Smith, Proceedings of the Second International Workshop on Network and Operating System Support for Digital Audio and Video, Heidelberg, Germany, November 1991, published in *Lecture Notes in Computer Science*, R.G. Herrtwich, editor, Volume 614, pages 10-21, Springer-Verlag, Heidelberg, Germany, 1992.

UNC Collaboratory Project Overview, J.B. Smith, F.D. Smith, P. Calingaert, J.R. Hayes, D. Holland, K. Jeffay, M. Lansman, Proceedings of the 1991 Symposium on Command and Control Research, National Defense University, Washington, D.C., June 1991, pages 341-391.

YARTOS: Kernel support for efficient, predictable real-time systems, K. Jeffay, D. Stone, D. Poirier, Proceedings of the Joint Eighth IEEE Workshop on Real-Time Operating Systems and Software and IFAC/IFIP Workshop on Real-Time Programming, Atlanta, GA, May 1991, in Real-Time Systems Newsletter, Volume 7, Number 4, Fall 1991, pages 8-13. Republished in "Real-Time Programming," W. Halang, K. Ramamritham, editors, 1992.

System Design for Workstation-Based Conferencing With Digital Audio and Video, K. Jeffay, F.D. Smith, Proceedings of the IEEE Conference on Communication Software: Communications for Distributed Applications and Systems, Chapel Hill, NC, April 1991, pages 169-180.

Designing a Workstation-Based Conferencing System Using the Real-Time Producer/Consumer Paradigm, K. Jeffay, F.D. Smith, Proceedings of the First International Workshop on Network and Operating System Support for Digital Audio and Video, International Computer Science Institute, Berkeley, CA, November 1990, pages 40-55.

Analysis of a Synchronization and Scheduling Discipline for Real-Time Tasks with Preemption Constraints, K. Jeffay, Proceedings of the Tenth IEEE Real-Time Systems Symposium, Santa Monica, CA, December 1989, pages 295-305.

Corset & Lace: Adapting Ada Runtime Support to Real-Time Systems, T.P. Baker, K. Jeffay, Proceedings of the Eighth IEEE Real-Time Systems Symposium, San Jose, CA, December 1987, pages 158-167.

Research in Real-Time Systems, A.C. Shaw, C. Binding, W.L. Hu, K. Jeffay, Proceedings of the Third IEEE Workshop on Real-Time Operating Systems, Boston, MA, February 1986, pages 121-131.

Book Chapters

Rate-Based Resource Allocation Methods, K. Jeffay, in, "Handbook of Real-Time and Embedded Systems," I. Lee, J. Y-T. Leung, S.H. Son, editors, Chapman & Hall/CRC Press, Boca Raton, FL, 2008, pages 4-1 – 4-15.

Visualization and Natural Control Systems for Microscopy, R.M. Taylor II, D. Borland, F.P. Brooks Jr., M. Falvo, M. Guthold, T. Hudson, K. Jeffay, G. Jones, D. Marshburn, S.J. Papadakis, L.-C. Qin, A. Seeger, F.D. Smith, D.H. Sonnenwald, R. Superfine, S. Washburn, C. Weigle, M.C. Whitton, P. Williams, L. Vicci, W. Robinett, in "Visualization Handbook," C. Johnson, C. Hansen, editors, Harcourt

Academic Press, 2005, pages 875-900.

An Empirical Study of Delay Jitter Management Policies, D.L. Stone, K. Jeffay, in "Readings in Multimedia Computing and Networking," K. Jeffay, H.J. Zhang, editors, Morgan Kaufmann, San Francisco, CA, 2002, pages 525-537.

The Performance of Two-Dimensional Media Scaling for Internet Videoconferencing, P. Nee, K. Jeffay, G. Danneels, in "Readings in Multimedia Computing and Networking," K. Jeffay, H.J. Zhang, editors, Morgan Kaufmann, San Francisco, CA, 2002, pages 581-592.

Lock-Free Transactions for Real-time Systems, J.H. Anderson, S. Ramamurthy, M. Moir, K. Jeffay, in "Real-Time Database Systems: Issues and Applications," A. Bestavros, K.J. Lin, S.H. Son, editors, Kluwer Academic Publishers, Norwell, MA, 1997, pages 215-234.

Transport and Display Mechanisms For Multimedia Conferencing Across Packet-Switched Networks, K. Jeffay, D.L. Stone, F.D. Smith, in "A Guided Tour of Multimedia Systems and Applications," B. Furht, M. Milenkovic, editors, IEEE Computer Society Press, Los Alamitos, CA, 1995, pages 439-461.

Contributor to: *R&D for the NII: Technical Challenges*, M.K. Vernon, E.D. Lazowska, S.D. Personick, editors, Interuniveristy Communications Council (EDUCOM), 1994.

YARTOS: Kernel support for efficient, predictable real-time systems, K. Jeffay, D. Stone, D. Poirier, in "Real-Time Programming," W. Halang, K. Ramamritham, editors, Pergamon Press, Oxford, UK, 1992, pages 7-12.

Videotapes

Adaptive, Best-Effort Delivery of Live Audio and Video Across Packet-Switched Networks, K. Jeffay, D.L. Stone, ACM Multimedia '94 Video Proceedings, San Francisco, CA, October 1994, 6 minutes. Excerpts also appear on the CD-ROM version of the conference proceedings.

Abstracts & Short Papers

Quantifying the Effects of Recent Protocol Improvements to Standards-Track TCP, M.C. Weigle, K. Jeffay, and F.D. Smith, Proceedings of the 11th IEEE/ACM International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Orlando, FL, October 2003, pages 226-229.

Engineering of Rate Based Services for Real-Time Computing, G. Lamastra, K. Jeffay, 20th IEEE Real-Time Systems Symposium Work in Progress Proceedings, Phoenix, AZ, December 1999, pages 51-55.

Lightweight Active Router-Queue Management for Multimedia Networking, M. Parris, K. Jeffay, F.D. Smith, IS&T/SPIE International Symposium on Electronic Imaging: Science and Technology, San Jose, CA, January 1999, pages 280-281.

Support For Real-Time Computing in Windows NT, K. Jeffay, Usage Abstracts, USENIX Windows NT Workshop, Seattle, WA, August 1997, page 84.

On the Duality between Resource Reservation and Proportional Share Resource Allocation, I. Stoica, H. Abdel-Wahab, K. Jeffay, IS&T/SPIE Ninth Symposium on Electronic Imaging: Science and Technology 1997, San Jose, CA, February 1997, pages 135-136.

Technical and Educational Challenges For Real-Time Computing, K. Jeffay, *ACM Computing Surveys*, Volume 28A, Number 4(es), December 1996, URL *http://www.acm.org/pubs/contents/journals/surveys/1996-28/#4es*, 3 pages.

Distributed Real-Time Dataflow: An Execution Paradigm for Image Processing and Anti-Submarine Warfare Applications, S. Goddard, K. Jeffay, 19th IEEE Real-Time Systems Symposium Work in Progress Proceedings, Washington, DC, December 1996, pages 55-58.

Why Using the Request Abstraction in Proportional Share Allocation Systems is Useful, I. Stoica, H.

Zhang, K. Jeffay, Proceedings of the IEEE Real-Time Systems Symposium Workshop on Resource Allocation Problems in Multimedia Systems, Washington, DC, December 1996, 4 pages.

Future distributed embedded and real-time applications will be adaptive: Meanings, challenges, and research paradigms, A.K. Mok, C.L. Heitmeyer, K. Jeffay, M.B. Jones, C.D. Locke, R. Rajkumar, 15th Proceedings of the 15th IEEE International Conference on Distributed Computing Systems, Vancouver, British Columbia, Canada, May 1995, pages 182-184.

Adaptive, Best-Effort Delivery of Live Audio and Video Across Packet-Switched Networks, K. Jeffay, D.L. Stone, Proceedings of the Second ACM International Conference on Multimedia, San Francisco, CA, October 1994, pages 487-488.

Adaptive Rate-Based Flow and Latency Management of Audio and Video Streams, T.-M. Chen, T. Talley, K. Jeffay, Abstracts of the Fourth Workshop on Network and Operating System Support for Digital Audio and Video, Lancaster, UK, November 1993, pages 17-20.

Client/Server Protocol Filtering: An infrastructure for supporting group collaborations, K. Jeffay, J. Menges, J.-K. Lin, ACM CSCW '92, Proceedings of the Workshop on Tools and Technologies, Toronto, Canada, November 1992, pages 96-99.

Network and Operating Systems Support for Digital Audio and Video, K. Jeffay, in 13th ACM Symposium on Operating System Principles "Work in Progress" Abstracts, E.D. Lazowska, editor, Operating Systems Review, Volume 26, Number 2, April 1992, pages 7-31.

Posters

Education Modules for Networking, Cloud Computing, and Security in Systems Courses, J. Aikat, M.K. Reiter, K. Jeffay, ACM Special Interest Group on Computer Science Education Technical Symposium, Memphis, TN, March 2016.

Efficient Management of a High-Capacity Airborne Network of Commercial Aircraft, B. Newton, J. Aikat, K. Jeffay, NSF Cyber-physical Systems Young Professional Workshop, Washington, DC, March 2014.

Variability of TCP Round-trip Times (RTTs) Over the Years, W. Yau, J. Aikat, K. Jeffay, Grace Hopper Celebration of Women in Computing (GHC), Baltimore, MD, October 2012.

Characterizing Modern Web Traffic using TCP Headers, S. Zolayvar, J. Aikat, K. Jeffay, Grace Hopper Celebration of Women in Computing (GHC), Baltimore, MD, October 2012.

Multivariate SVD Analyses For Network Anomaly Detection, J. Terrell, L. Zhang, K. Jeffay, A. Nobel, H. Shen, F.D. Smith, Z, Zhu, ACM SIGCOMM 2005 Poster Session, Philadelphia, PA, August 2005.

How Real Can Synthetic Traffic Be?, F. Hernández-Campos, K. Jeffay, F.D. Smith, ACM SIGCOMM 2004 Poster Session, Portland, OR, August 2004.

A Non-Parametric Approach to Generation and Validation of Synthetic Network Traffic, F. Hernández-Campos, A. Nobel, F.D. Smith, K. Jeffay, IMA Workshop on Measurement, Modeling, and Analysis of the Internet, Minneapolis, MN, January 2004.

Sync-TCP: Using GPS Synchronized Clocks for Early Congestion Detection in TCP, M.C. Weigle, K. Jeffay, F.D. Smith, ACM SIGCOMM 2000 Poster Session, Stockholm, Sweden, August 2000, page 6.

Reviews

Advance Reservation Systems, K. Jeffay, Proceedings of the Fifth International Workshop on Network and Operating System Support for Digital Audio and Video, Durham, NH, April 1995, published in *Lecture Notes in Computer Science*, T.D.C. Little, R. Gusella, editors, Volume 1018, pages 1-2, Springer-Verlag, Heidelberg, Germany, 1995.

Books and Proceedings

The Effects of Traffic Structure on Application and Network Performance, J. Aikat, K. Jeffay, and F. D. Smith, Springer Publishing, New York, NY, 2012, 289 pages.

The Effects of Active Queue Management on TCP Application Performance, An experimental performance evaluation, L. Le, K. Jeffay, F.D. Smith, Lambert Academic Publishing, Berlin, Saarbrücken, Germany, 2009, 222 pages.

Quality-of-Service — IWQoS 2003, K. Jeffay, I. Stoica, K. Wehrle, editors, Lecture Notes in Computer Science, Springer-Verlag, Berlin, Heidelberg, Germany, Volume 2707, 2003, ISBN 3-540-40281-0, 517 pages.

Readings in Multimedia Computing and Networking, K. Jeffay, H.J. Zhang, editors, Morgan Kaufman, San Francisco, CA, 2002, ISBN 1-55860-651-3, 863 pages.

Proceedings, The 22nd IEEE Real-Time Systems Symposium, K. Jeffay, G. Buttazzo, editors, IEEE Computer Society Press, Los Alamitos, CA, 2001, ISBN 0-7695-1420-0, 321 pages.

Proceedings, The Sixth IEEE International Symposium on Computers and Communications, K. Jeffay, R. Steinmetz, editors, IEEE Computer Society Press, Los Alamitos, CA, 2001, ISBN 0-7695-1177-5, 754 pages.

Proceedings, The 21st IEEE Real-Time Systems Symposium, K. Jeffay, W. Zhao, editors, IEEE Computer Society Press, Los Alamitos, CA, 2000, ISBN 0-7695-0900-2, 311 pages.

Proceedings, The 10th International Workshop on Network and Operating System Support for Digital Audio and Video, K. Jeffay and H. Vin, editors, Technical Report, Department of Computer Science, University of North Carolina at Chapel Hill, 2000, 320 pages. Proceedings also published on-line at *http://www.cs.unc.edu/nossdav2000*.

ACM Multimedia '99, D. Bulterman, K. Jeffay, H.J. Zhang, editors, ACM Press, Los Angeles, CA, 1999, ISBN 0-8194-3125-7, 500 pages.

Multimedia Computing and Networking 1999, D.D. Kandlur, K. Jeffay, T. Roscoe, editors, Proceedings of SPIE, Volume 3654, SPIE, Bellingham, WA, 1998, ISBN 0-8194-3125-7, 328 pages.

Multimedia Computing and Networking 1998, K. Jeffay, D.D. Kandlur, T. Roscoe, editors, Proceedings of SPIE, Volume 3310, SPIE, Bellingham, WA, 1997, ISBN 0-8194-250-0, 266 pages.

Proceedings, 1997 IEEE Real-Time Technology and Applications Symposium, R. Rajkumar, K. Jeffay, editors, IEEE Computer Society Press, Los Alamitos, CA, 1997, ISBN 0-8186-8016-4, 269 pages.

Proceedings, IEEE Real-Time Systems Symposium Workshop on Resource Allocation Problems in Multimedia Systems, K. Jeffay, editor, *http://www.cs.unc.edu/~jeffay/meetings/mm-wrkshp96/prog. html,* 1996, 250 pages.

Proceedings, 1996 IEEE Real-Time Technology and Applications Symposium, K. Jeffay, W. Zhao, editors, IEEE Computer Society Press, Los Alamitos, CA, 1996, ISBN 0-8186-7448-2, 264 pages.

Unrefereed Publications

Invited Papers

Modeling and Generation of TCP Application Workloads, F. Hernández-Campos, K. Jeffay, F.D. Smith, Proceedings of the 4th IEEE International Conference on Broadband Communications, Networks, and Systems, Raleigh, NC, September 2007, 10 pages.

Statistical Clustering of Internet Communication Patterns, F. Hernández-Campos, A.B. Nobel, F.D. Smith, K. Jeffay, Proceedings of the 35th Symposium on the Interface of Computing Science and

Statistics, Salt Lake City, UT, July 2003, Computing Science and Statistics, Volume 35, 2004.

Rate-Based Resource Allocation Methods for Embedded Systems, K. Jeffay, S.M. Goddard, in, *Embedded Software*, Proceedings of the First International Workshop on Embedded Software (*EMSOFT 2001*), Tahoe City, CA, October 2001, Lecture Notes in Computer Science, Volume 2211, T. Henzinger, C. Kirsch, editors, Springer Verlag, Berlin, Germany, 2001, pages 204-222.

Beyond Audio and Video: Multimedia Networking Support for Distributed, Immersive Virtual Environments, K. Jeffay, T. Hudson, M. Parris, Proceedings of the 27th EUROMICRO Conference, Workshop on Multimedia and Telecommunication, Warsaw, Poland, September 2001, pages 300-307.

Experiments in Best-Effort Multimedia Networking for a Distributed Virtual Environment, T. Hudson, M.C. Weigle, K. Jeffay, R.M. Taylor II, in *Multimedia Computing and Networking 2001*, Proceedings, SPIE Proceedings Series, Volume 4312, San Jose, CA, January 2001, pages 88-98.

Towards a Better-Than-Best-Effort Forwarding Service for Multimedia Flows, K. Jeffay, IEEE Multimedia, Volume 6, Number 4, October-December 1999, pages 84-88.

Network Support For Distributed, Immersive, Virtual Laboratories K. Jeffay, Proceedings of the NSF Workshop on Automated Control of Distributed Instrumentation, Beckman Institute for Advanced Science and Technology, University of Illinois at Urbana-Champaign, Urbana, IL, April 1999, pages 73-78.

Storage Requirements For Distributed Virtual Laboratories, K. Jeffay, Proceedings of the Internet2 Distributed Storage Infrastructure Application Workshop, University of North Carolina at Chapel Hill, Chapel Hill, NC, March 1999, pages 57-59.

Efficient Kernel Support for Continuous Time Media Systems, K. Jeffay, Proceedings of the ITC Workshop on Continuous Time Media, Carnegie Mellon University, Pittsburgh, PA, June 1991, 4 pages.

White Papers

Contributor to: *NSF Report on Network Research Testbeds*, B. Braden, M. Gerla, J. Kurose, J. Lepreau, R. Rao, J. Turner, editors, December 2002. A white paper commission by NSF that led to the creation of a $10 million funding program for network research testbeds.

Contributor to: *R&D for the National Information Infrastructure: Technical Challenges*, M.K. Vernon, E.D. Lazowska, S.D. Personick, editors, Interuniveristy Communications Council (EDUCOM), 1994.

Technical Reports

Beyond Window Sharing Hacks: Support for First-Class Window Sharing, J. Menges, K. Jeffay, Report TR97-021, University of North Carolina at Chapel Hill, Department of Computer Science, Chapel Hill, NC, April 1997.

A Proportional Share Resource Allocation Algorithm For Real-Time, Time-Shared Systems, I. Stoica, H. Abdel-Wahab, K. Jeffay, S.K. Baruah, J.E. Gehrke, C.G. Plaxton, Report TR96-038, University of North Carolina at Chapel Hill, Department of Computer Science, Chapel Hill, NC, September 1996.

Predicting Worst Case Execution Times on a Pipelined RISC Processor, S.J. Bharrat, K. Jeffay, Report TR94-072, University of North Carolina at Chapel Hill, Department of Computer Science, Chapel Hill, NC, April 1994.

The Design, Implementation, and Use of a Sporadic Tasking Model, K. Jeffay, D. Becker, D. Bennett, S. Bharrat, T. Gramling, M. Housel, Report TR94-073, University of North Carolina at Chapel Hill, Department of Computer Science, Chapel Hill, NC, April, 1994.

UNC Collaboratory Project Overview, J.B. Smith, F.D. Smith, P. Calingaert, J.R. Hayes, D. Holland, K. Jeffay, M. Lansman, Report 90-042, University of North Carolina at Chapel Hill, Department of Computer Science, Chapel Hill, NC, November 1990.

An Implementation and Application of the Real-Time Producer/Consumer Paradigm, D. Poirier, K. Jeffay, Report 90-038, University of North Carolina at Chapel Hill, Department of Computer Science, Chapel Hill, NC, October 1990.

The Real-Time Producer/Consumer Paradigm: Towards Verifiable Real-Time Computations, K. Jeffay, Report 89-09-15, University of Washington, Department of Computer Science, Seattle, WA, September 1989, (Ph.D. Dissertation).

On Optimal, Non-Preemptive Scheduling of Periodic and Sporadic Tasks, K. Jeffay, R. Anderson, Report 88-11-06, University of Washington, Department of Computer Science, Seattle, WA, November 1988.

On Optimal, Non-Preemptive Scheduling of Periodic Tasks, K. Jeffay, Report 88-10-03, University of Washington Department of Computer Science, Seattle, WA, October 1988.

Software Engineering of Real-Time Operating Systems, A.C. Shaw, K. Jeffay, Report 88-01-01, University of Washington Department of Computer Science, Seattle, WA, January 1988.

Concurrent Programming with Time, A thesis proposal, K. Jeffay, Report 87-10-03, University of Washington Department of Computer Science, Seattle, WA, October 1987.

A Lace for Ada's Corset, T.P. Baker, K. Jeffay, Report 86-09-06, University of Washington Department of Computer Science, Seattle, WA, September 1986.

Software Distributions

GENI tmix — A synthetic TCP workload generator for GENI, J. Aikat, D. O'Neil, B. Newton, K. Jeffay, June 2012.

Distributed to the GENI networking research community.

tmix — A synthetic TCP workload generator, F. Hernández Campos, K. Jeffay, F.D. Smith, June 2005.

Distributed to other academic and industry networking research groups via the Web.

UNC VNC — A version of the public domain workspace sharing system VNC to support secure sharing of workspace regions and windows of individual applications. J. Branscomb, L. Fowler, K. Jeffay, August 2004.

Distributed to other research groups and the public via the Web and SourceForge.

thttp 2003 — An HTTP/v1.0 and v1.1 traffic generation program used to simulate the traffic generated by a collection of geographically distributed web browsers and servers. L. Le, A. Van Osdol, F.D. Smith, K. Jeffay, January 2004.

Distributed to other academic and industry networking research groups via the Web.

thttp 2001 — An HTTP/v1.0 and v1.1 traffic generation program used to simulate the traffic generated by a collection of geographically distributed web browsers and servers. F. Hernández Campos, F.D. Smith, K. Jeffay, August 2002.

Distributed to other academic and industry networking research groups via the Web.

Packmime Implementation in ns — An implementation of the Lucent Bell Labs (Bill Cleaveland) HTTP traffic model in ns. M.C. Weigle, K. Jeffay, F.D. Smith, September 2001.

Distributed as part of the Network Simulator (**ns-2**) distribution.

UNC Campus Network Trace Data and Empirical Distributions — A collection of network traces and empirical distributions illustrating the nature of web browsing and the structure of web pages. F. Hernández Campos, K. Jeffay, F.D. Smith, June 2001.

Distributed to other academic and industry networking research groups via the Web.

HTTP Analysis Tools— A set of tools for analyzing http connections. F. Hernández Campos, K. Jeffay,

F.D. Smith, June 2001.

Distributed as part of the Network Simulator (**ns-2**) distribution.

thttp — An HTTP traffic generation program used to simulate the traffic generated by a collection of geographically distributed web browsers and servers. M. Christiansen, F.D. Smith, K. Jeffay, January 2000.

Distributed to other academic and industry networking research groups via the Web.

FlowGen — A general purpose, programmable network flow generator. M. Clark, K. Jeffay, F.D. Smith, December 1999.

Distributed to Advanced Networking Systems Inc. for use in performance evaluation studies of Internet 2 and Abilene. Also available via anonymous *ftp* from *http://www.cs.unc.edu/Research/Dirt*.

TruTime Device Drivers for FreeBSD — A set of FreeBSD device drivers for the TruTime GPS boards, A. Moorthy, K. Jeffay, and F.D. Smith, December 1998.

Distributed to Advanced Networking Systems Inc. for use in the IETF Internet Performance Measurement Initiative (IPMI) Surveyor network.

YARTOS (*Yet Another Real-Time Operating System*) — A real-time operating system kernel for Intel x86 platforms that uses a novel task implementation based on a single, shared run-time stack, an earliest-deadline-first processor scheduling algorithm developed at UNC, and integrates processor scheduling and inter-process communication. K. Jeffay and D. Stone, January 1996.

Distributed to several universities for use in operating system research and teaching. Also adopted by researchers at IBM's Networking Software Division, RTP, NC, for use in constructing a network protocol evaluation testbed.

XPEL (*The X Protocol Engine Library*) — A C^{++} library for constructing modular X Window System pseudo-servers, J. Menges and K. Jeffay, November 1993.

Distributed on the Internet via anonymous *ftp* from *ftp.cs.unc.edu*.

XTV (*X Terminal View*) — An X Window System pseudo-server that supports window sharing and conferencing across the Internet, H. Abdel-Wahab (Old Dominion University) and K. Jeffay, August 1991.

Distributed by MIT as part of the contributed software portion of the X Window System distribution (release 11, version 5). Last revised January 1993.

Also adopted by the NEC Corporation of Japan as the basis for their *Xprotodist* (X protocol distributor) product.

Patents

Methods, Systems, and Computer Readable Media For Topology Management and Geographic Routing in Mobile Ad-Hoc Networks, U.S. Patent Application No. 15/256,192, B. Newton, K. Jeffay, J. Aikat, September 2016.

High-Performance Virtualized Applications Platform, U.S. Patent Application No. 421-343. A. Blate, K. Jeffay, May 2014.

Methods, Systems, and Computer Program Products For Network Server Performance Anomaly Detection, U.S. Patent No. 8,938,532, J.S. Terrell. K. Jeffay, F.D. Smith, R. Broadhurst, January 2015.

Method for Understanding the Use of TCP/IP Networks by Users, and Non-Parametric Generation of Synthetic Internet Traffic, U.S. Patent Number 7,447,209, F. Hernández-Campos, K. Jeffay, F.D. Smith, A. Nobel, November 2008.

User Controlled Adaptive Flow Control for Packet Networks, U.S. Patent Number 5,892,754, V. Kompella, F.D. Smith, J.P. Gray, and K. Jeffay, April 1999.

Public Demonstrations

Education Modules for Networking, Cloud Computing and Security in Systems Courses, J. Aikat, K. Jeffay, Demo session at the ACM SIGCSE, Special Interest Group for Computer Science Education, Memphis, TN, March 2016.

Exploring the Internet's Future with GENI (Global Environment for Network Innovations), J. Aikat, K. Jeffay, NSF Higher Education Technology Conference 2013, Morgantown, WV, October 2013.

Your first experiment on GENI: a Hands-on Tutorial, J. Aikat, K. Jeffay, NSF Higher Education Technology Conference 2013, Morgantown, WV, October 2013.

NSF Workshop on GENI in Education (one day), J. Aikat, K. Jeffay, New York University - Polytechnic, NY, October 2013.

Using the GENI Networking Testbed, J. Aikat, K. Jeffay, ACM SIGCSE, Special Interest Group for Computer Science Education, Denver, Colorado, March 2013.

Run your Systems/Networking Experiments using (free) GENI Resources, Lunchtime Table Topics, S. Zolayvar, W. Yau, J. Aikat, K. Jeffay, Grace Hopper Celebration of Women in Computing, Baltimore, Maryland, October 2012.

NSF Workshop on Designing Tools and Curricula for Undergraduate Courses in Distributed Systems, J. Aikat, K. Jeffay, Boston, MA, July 2012.

Introduction to Network Experiments using the GENI CyberInfrastructure, J. Aikat, K. Jeffay, ACM SIGMETRICS/Performance Joint International Conference on Measurement and Modeling of Computer Systems, London, UK, June 2012.

The nanoManipulator "reverse field trip" to Orange County High-School, G. Jones, R. Superfine, R.M. Taylor, M. Whitton, T. Lovelace, R. Parameswaran, K. Jeffay, F.D. Smith, Orange County High School, Hillsboro, NC, November 1999.

The Distributed NanoManipulator Project, R.M. Taylor, K. Jeffay, T. Hudson, M. Clark, an invited demonstration for the Internet 2 Spring 1999 Members Meeting, Washington, DC, April 1999.

A Demonstration of Internet Access to UNC-CH Advanced Microscopy Facilities for Science Education Outreach, K. Jeffay, F.D. Smith, R. Superfine, G. Jones, R.M. Taylor, T. Hudson, M. Clark, Orange County High School, Hillsboro, NC, June 1998.

The Artifact-Based Collaboration (ABC) System, J.B. Smith, K. Jeffay, D. Shackelford, J. Menges, J. Hilgedic, B. Ladd, M. Parris, E. Kupstas, T. Ellis, and T. Hudson, CSCW '94, ACM Conference on Computer-Supported Cooperative Work, RTP, NC, October 1994.

CONCUR: A Window System Supporting Window Sharing, J. Menges and K. Jeffay, CSCW '94, ACM Conference on Computer-Supported Cooperative Work, RTP, NC, October 1994.

Adaptive, Best-Effort Delivery of Live Audio and Video Across Packet-Switched Networks, K. Jeffay, Second MCNC/NSF Packet Video Workshop, Research Triangle Park, NC, December 1992.

Grants and Awards

GENI in the classroom: Course Modules for Teaching Networking Concepts, J. Aikat, K. Jeffay, National

Science Foundation (GENI Project: Raytheon Corporation), October 2013, 3 years, $178,978.

II-New: Seeing the Future: Ubiquitous Computing in EyeGlasses, J.-M. Frahm, *et al*., National Science Foundation, November 2013, 3 years, $613,623.

GENI Tutorial at ACM SIGCSE, J. Aikat, K, Jeffay, National Science Foundation, grant number CNS 13-32340, March 2013, 1 year, $13,388.

Development of Education and Training Resources for GENI Experimenters, K, Jeffay, J. Aikat, Global Environment for Network Innovations/National Science Foundation, December 2011, 3 years, $388,590.

Exploring Privacy Breaches in Encrypted VoIP Communications, F. Monrose, K, Jeffay, National Science Foundation, grant number CNS 10-17318, August 2010, 3 years, $496,492.

Synthetic Traffic Generation Tools and Resource: A Community Resource for Experimental Networking Research, K, Jeffay, F.D. Smith, UNC, M. Weigle, Old Dominion University, A. Vahdat, University of California San Diego, P. Barford, University of Wisconsin, National Science Foundation, grant number CRI 07-09081, August 2007, 3 years, $799,745.

Modeling and Testing of Application Workloads on Corporate Enterprise Networks, K, Jeffay, F.D. Smith, IBM Global Services Faculty Award, September 2006, 2 years, $70,000 (exempt from indirect costs).

Tera-pixels: Using High-resolution Pervasive Displays to Transform Collaboration and Teaching, K, Jeffay, A. Lastra, K. Mayer-Patel, L. McMillan, F.D. Smith, National Science Foundation (CISE RI Program), grant number EIA 03-03590, August 2003, 5 years, $962,902.

Extracting and Using Semantic Information in Network Workloads, K, Jeffay, F.D. Smith, IBM Global Services Faculty Award, September 2006, 1 year, $40,000 (exempt from indirect costs).

Modeling and Testing of Application Workloads on Corporate Enterprise Networks, K, Jeffay, F.D. Smith, IBM Global Services Faculty Award, September 2006, 1 year, $40,000 (exempt from indirect costs).

Generation and Validation of Synthetic Internet Traffic, K, Jeffay, F.D. Smith, A.B. Noble, National Science Foundation, grant number ANI 03-23648, September 2003, 3 years, $470,000.

Rate-Based Resource Allocation Methods for Real-Time Embedded Systems, K, Jeffay, F.D. Smith, National Science Foundation, grant number CCR 02-08924, August 2002, 3 years, $179,990.

Empirical Workload Characterizations for Advanced Networks, K. Jeffay, F.D. Smith, Cisco Systems, November 2000, 3 years, $212,500.

Support for Active Queue Management, Distributed XP Programming, and a Teaching Laboratory for Web Services, F.D. Smith, K. Jeffay, J.B. Smith, P.D. Stotts, IBM Shared University Research Grant, August 2002, 1 year, $83,000.

Internet Traffic Measurement and Analysis and A Teaching Laboratory for Enterprise Java Computing, F.D. Smith, K. Jeffay, J.B. Smith, P.D. Stotts, IBM Shared University Research Grant, April 2001, 1 year, $83,000.

An NS Implementation of the HTTP Connection Model, F.D. Smith, K. Jeffay, M. Clark Weigle, Lucent Technologies, April 2001, $10,000.

Multimedia Networking Research in Support of Virtual Field Trips, K. Jeffay, F.D. Smith, R.M. Taylor II, Dell Computer Corporation Strategic Technology and Research (STAR) program, December 2000, 1

year, $25,000.

Rate-based Scheduling Technology for Latency-Sensitive Graphics Applications, J. Anderson, S. Baruah, K. Jeffay, R. Taylor, National Science Foundation, grant number ITR 00-82866, September 2000, 3.5 years, $350,000.

Active Router Queue Management for Congestion Control and Quality-of-Service, K. Jeffay, F.D. Smith, National Science Foundation, grant number ITR 00-82870, September 2000, 3 years, $451,903.

Internet Measurements and Analysis to Support the nanoManipulator and Tele-Immersion, F.D. Smith, K. Jeffay, P. Jones, IBM Shared University Research Grant, September 2000, 1 year, $250,000.

The Performance of Differentiated Services Implementations for Supporting Distributed Virtual Environment Applications, K. Jeffay, North Carolina Network Initiative, September 2000, 1 year, $25,000.

Empirical Application-Workload Characterizations for Advanced Networks, K. Jeffay, F.D. Smith, Sun Microsystems, September 2000, $79,000.

Empirical Application-Workload Characterizations for Advanced Networks, F.D. Smith, K. Jeffay, MCNC, September 2000, 1 year, $50,000.

A Quality-of-Service Network for the nanoManipulator, K. Jeffay, Cabletron Inc. and the North Carolina Network Initiative, September 1999, $25,000.

Interactive Graphics for Molecular Studies and Microscopy — Supplement for Collaborative Science, F.P. Brooks, D. Erie (Chemistry), K. Jeffay, J. Samulski (Gene Therapy), F.D. Smith, D. Sonnenwald (Information and Library Science), R. Superfine (Physics and Astronomy), R. Taylor, National Institute of Health, October 1998, 4 years, $1,902,544.

Computing Power for Collaborative Science, S.R. Aylward, G. Bishop, D.W. Brenner, E. Bullitt, E.L. Chaney, V.L. Chi, B.J. Dempsey, N. England, A.G. Gash, D. Fritsch, H. Fuchs, B. Hemminger, J. Hermans, K. Jeffay, K. Keller, A. Lastra, M. Lin, D. Manocha, L.S. Nyland, S.M. Pizer, J.W. Poulton, J.F. Prins, J. Rosenman, F.D. Smith, R. Superfine, R.M. Taylor, S. Tell, G. Tracton, A. Tropsha, S. Washburn, G. Welch, Intel Corporation, August 1998, 3 years, $2,858,747. PI on *Multimedia Networking* and *A Distributed Teaching Laboratory for Networking and Internet Technologies* sections (F.D. Smith, B.J. Dempsey, Co-PIs), $557,669.

Empirical Application-Workload Characterizations for Advanced Networks, F.D. Smith, K. Jeffay, North Carolina Network Initiative, September 1998, $40,000.

Congestion control for high-speed networks, F.D. Smith, K. Jeffay, IBM Corporation, 1998, $20,000.

A Communications Middleware For Immersive Distributed Virtual Environments, K. Jeffay, North Carolina Network Initiative, 1998, $15,000.

Technology for Real-Time Services in Protocol Stack Implementations, K. Jeffay, IBM Corporation, 1997, $80,000.

Internet Access to UNC-CH Advanced Microscopy Facilities for Science Education Outreach, K. Jeffay, R. Superfine (Physics and Astronomy), G. Jones (Education), University of North Carolina at Chapel Hill Instructional Technology Award, 1997, $29,750.

Support for the Real-Time Technology and Applications Symposium, K. Jeffay, Office of Naval Research, 1997, $7,000.

Research Experience For Undergraduates supplement to *Object Sharing Technology For Real-Time*

Systems, J. Anderson, K. Jeffay, National Science Foundation, 1996, $10,000.

Infrastructure for Research in Collaborative Systems, S.F. Weiss, J.B. Smith, K. Jeffay, P. Dewan, P. D. Stotts, D.K. Smith, F.D. Smith, W. Oliver (U.S. Armed Forces Institute of Pathology), National Science Foundation, grant number CDA-9624662, August 1996, 5 years, $1,260,830.

Collaboration Bus: An Infrastructure for Supporting Interoperating Collaborative Systems, P. Dewan, K. Jeffay, H. Abdel-Wahab (Old Dominion University), P. D. Stotts, L. Nyland, J.B. Smith, J. Mchugh (Portland State University), J. Menges (Hewlett Packard), Advanced Research Projects Agency, grant number 96-06580 (High Performance Distributed Services Technology), 1996, $973,334.

Object Sharing Technology For Real-Time Systems, J. Anderson, K. Jeffay, National Science Foundation, grant number CCR 95-10156, 1996, $209,442.

Flexible Shared Windows, P. Dewan, K. Jeffay, National Science Foundation, grant number IRIS 95-08514, 1995, $343,811.

An ATM Testbed For Multimedia Networking and Computer-Supported Cooperative Work, K. Jeffay, IBM Corporation, 1995, $400,000.

An Examination of Flow and Congestion Control Mechanisms for Media Transmission in Collaborative Systems, K. Jeffay, IBM Corporation, 1995, $125,242.

Software Infrastructure for the Rapid Development of Interactive and Collaborative Educational Simulations, J.F. Prins, K. Jeffay, P.D. Stotts, L.S. Nyland, Advanced Research Projects Agency, grant number 95-36871 (Computer Aided Education and Training Initiative), 1995, $200,000.

A Proposal For Network Routers, K. Jeffay, IBM Corporation, Research Triangle Park, NC, 1994, $40,000.

System Support For Video Teleconferencing Across Local Area Networks, K. Jeffay, Intel Corporation, 1993, $253,385.

The Integration and Use of Digital Audio and Video in Desktop Computing Environments, K. Jeffay, IBM Corporation, 1993, $100,000.

Construction of a UNC-Tektronix MBONE Link, K. Jeffay, Tektronix Corporation, 1993, $10,000.

An Empirical Determination of the Limits of Human Perception of Properties of Digital Audio and Video Streams, K. Jeffay, University of North Carolina Research Council, 1993, $2,000.

Processor and Resource Allocation Problems in Hard-Real-Time Systems: Theory and Practice, K. Jeffay, National Science Foundation, RIA Award, 1991, $59,400.

Accommodating Continuous Media in a Local Area Network: An exercise in distributed real-time computing, K. Jeffay, University of North Carolina Junior Faculty Development Award, 1990, $3,000.

Building and Using a Collaboratory: A Foundation for Supporting and Studying Group Collaborations, J.B. Smith, F.D. Smith, P. Calingaert, K. Jeffay, J.R. Hayes, D. Holland, M. Lansman, National Science Foundation, grant number ICI-9015443, 1990, $946,000.

Enhanced program of research and teaching in communications software and distributed systems, K. Jeffay, IBM Corporation, 1989, $429,000.

The Design and Construction of Predictable Real-Time Systems, K. Jeffay, Digital Faculty Program Award, Digital Equipment Corporation, 1989, $180,000.

Invited Presentations

Keynote Addresses

Network Neutrality Considered Harmful,

PearlHacks, University of North Carolina at Chapel Hill, Chapel Hill, NC, March 2014.

Carolina Science Café, University of North Carolina at Chapel Hill, Chapel Hill, NC, May 2015.

Network Neutrality Considered Harmful, Revisiting the Quality-of-Service Morass

ACM International Workshop on Network and Operating System Support for Digital Audio and Video, Newport, RI, May 2006.

Rate-Based Resource Allocation Methods for Multimedia Computing

SPIE Multimedia Computing and Networking 2003, Santa Clara, CA, January 2003.

Network Support For Distributed Virtual Environments: The Tele-nanoManipulator

NCNI Advanced Networking Symposium, Research Triangle Park, NC, May 1999.

The Internet 2 Spring Members Meeting (presented jointly with R.M. Taylor), Washington, DC, April 1999.

Network Support For Distributed, Immersive, Virtual Environments

Workshop on Automated Control of Distributed Instrumentation, Beckman Institute for Advanced Science and Technology, University of Illinois at Urbana-Champaign, Urbana, IL, April 1999.

The Future of Networking: The Networking Revolution Has Yet to Begin

North Carolina Governor's Board of Science and Technology Retreat, Banner Elk, NC, September 1998.

Distinguished Lectures

The Synthetic Traffic Generation Problem: The least sexy problem in computer networking

University of Minnesota, Minneapolis, MN, January 2008.

The Effect of Active Queue Management on Web Performance: The Good, the Bad, and the Ugly

University of Nebraska, Lincoln, NE, November 2003.

University of Pennsylvania, April 2004.

The Evolution of Quality-of-Service on the Internet

Mälardalen University, Sweden, February 2001.

Colloquia

Efficient Topology Management and Geographic Routing in High-Capacity Continental-Scale Airborne Networks

University of Virginia, November 2017.

The Evolution of Quality-of-Service on the Internet

IEEE Computer Society Seattle Chapter, Seattle, WA, September 2005.

Modeling and Generating TCP Application Workloads

Georgia Institute of Technology, Atlanta, GA, September 2005.

Microsoft Research, Redmond, WA, September 2005.

Worcester Polytechnic Institute, Worcester, MA, July 2005.

Cisco Systems, RTP, NC, May 2005 (presently jointly with F. Hernandez-Campos and F.D. Smith).

A Rate-Based Execution Abstraction For Embedded Real-Time Systems

University of Pennsylvania, April 2004.

How "Real" Can Synthetic Network Traffic Be?

University of Virginia, March 2004.

Non-Parametric Approach to Generation and Validation of Synthetic Network Traffic

Columbia University, New York, NY, January 2004.

Is Explicit Congestion Notification (ECN) Worthwhile?

Cisco Systems, San Jose, CA, October 2003.
Intel Architecture Labs, Hillsboro, OR, October 2003.

Tuning RED for Web Traffic: RED Considered Harmful?

Sprint Advanced Technology Research Laboratories, San Francisco, CA, March 2000.
University of Illinois at Urbana-Champaign, Urbana, IL, March 2000.

Internet Traffic: Measurement & Generation

Ganymede Software, Research Triangle Park, NC, June 1999. (Presented jointly with F.D. Smith.)

Lightweight Active Router-Queue Management for Multimedia Networking

University of Toronto, October 2000.
HP Laboratories, Palo Alto, CA, January 1999.
Sprint Advanced Technology Research Laboratories, San Francisco, CA, January 1999.

A Better-Than-Best-Effort Service For UDP: Lightweight Active Router-Queue Management for Multimedia Networking

University of Virginia, January 1999.

Congestion Control Mechanisms for Real-Time Communications on the Internet

IBM Networking Systems, Research Triangle Park, NC, November 1998.

A Better Than Best-Effort Forwarding Service For UDP

Microsoft Research, Redmond, WA, March 1998.
IBM Networking Systems, Research Triangle Park, NC, March 1998.

Multimedia Networking Research at UNC Chapel Hill

IBM Networking Systems, Research Triangle Park, NC, February 1998.
Lucent Technologies, Bell Laboratories, Holmdel, NJ, June 1997.

Design Principles for Distributed, Interactive, Virtual Laboratories

Mitsubishi Electric Research Laboratory, Cambridge, MA, January 1998.
Lucent Technologies, Bell Laboratories, Holmdel, NJ, June 1997.

The Performance of Two-Dimensional Media Scaling for Internet Videoconferencing

Mitsubishi Electric Research Laboratory, Cambridge, MA, January 1998.
Carnegie Mellon University, Pittsburgh, PA, March 1997.

Some Approaches to Enabling Real-Time Computation on Desktop Operating Systems

Honeywell Technology Center, Minneapolis, MN, October 1996.

Two-Dimensional Scaling Techniques for Adaptive, Rate-Based Transmission Control of Live Audio and Video Streams

Oregon State University, Corvallis, OR, May 1996.

A Router-Based Congestion Control Scheme For Real-Time Continuous Media

Intel Architecture Development Laboratory, Hillsboro, OR, May 1996.
NetEdge Inc., Research Triangle Park, NC, March 1996.
North Carolina State University, Raleigh, NC, March 1996.

A Hybrid Reservation-Based/Best-Effort Transmission Scheme For Real-Time Transmission of Multimedia Data

Intel Research Council, Hillsboro, OR, November 1995.

Operating System Support For Multimedia Computing

IBM Networking Systems, Research Triangle Park, NC, July 1995.

Network Support For Multimedia Computing

IBM Networking Systems, Research Triangle Park, NC, July 1995.

A Rate-Based Execution Abstraction For Multimedia Computing

Tektronix Computer Research Laboratory, Beaverton, OR, May 1996.
INRIA, Rocquencourt, France, December 1995.
AT&T Bell Laboratories, Murray Hill, NJ, July 1995.

A Theory of Rate-Based Scheduling

Carnegie Mellon University, Pittsburgh, PA, August 1994.

Demonstration of Videoconferencing Over Packet-Switched Internetworks

Global Lecture Hall of the U.S.-Russian Electronic Distance Education System/TELE-TEACHING '93, Trondheim, Norway (via satellite from Chapel Hill), August 1993.

Transport and Display Mechanisms For Multimedia Conferencing Across Packet-Switched Networks

York University, York, England, November 1993.
Purdue University, West Lafayette, IN, February 1993.
Intel Architecture Development Laboratory, Hillsboro, OR, February 1993.
Oregon Graduate Institute, Beaverton, OR, February 1993.
Second MCNC/NSF Packet Video Workshop, Research Triangle Park, NC, December 1992.
Carnegie Mellon University, Pittsburgh, PA, November 1992.

Multimedia and Conferencing Research at UNC-CH

IBM Multimedia Networking, Research Triangle Park, NC, September 1992.

System Support for Synchronous Collaboration

North Carolina Artificial Intelligence and Advanced Computing Symposium, Raleigh, NC, March 1992.

Software Architectures for Predictable Real-Time Computer Systems

Carnegie Mellon University, Pittsburgh, PA, March 1992.

Operating System Requirements for Digital Audio and Video

MCNC/NSF Packet Video Videoconferencing Workshop, Research Triangle Park, NC, December 1991.

Some Experiments With Live Digital Audio and Video on a Local-Area Network

IBM European Networking Center, Heidelberg, Germany, November 1991.

Network and Operating System Support for Digital Audio and Video

Duke University, Durham, NC, November 1994.
Carnegie Mellon University, Pittsburgh, PA, April 1994.
Intel Corporation, Hillsboro, OR, December 1993.
IBM Networking Systems, Research Triangle Park, NC, June 1992.
IBM TJ Watson Research Center, Yorktown Heights, NY, March 1992.
University of Washington, Seattle, WA, February 1992.
Old Dominion University, Norfolk, VA, October 1991.

Systems Research in the UNC Collaboration Project

Intel Architecture Development Laboratory, Hillsboro, OR, May 1992.
University of Colorado, Boulder, CO, March 1991.

Some Deterministic Resource Allocation Problems in Real-Time Computer Systems

Research Triangle Institute, Research Triangle Park, NC, June 1990.
University of North Carolina at Chapel Hill, Department of Operations Research, Chapel Hill, NC, October 1989.

The Real-Time Producer/Consumer Paradigm: Towards Verifiable Real-Time Computations

Carnegie-Mellon University, Pittsburgh, March 1991.
Duke University, Durham, NC, June 1990.
Technical University of Denmark, Lyngby, Denmark, September 1989.
University of California at Davis, Davis, CA, April 1989.
University of Minnesota, Minneapolis, MN, April 1989.
Purdue University, West Lafayette, IN, April 1989.
University of North Carolina at Chapel Hill, Chapel Hill, NC, April 1989.
University of British Columbia, Vancouver, BC, Canada, April 1989.
Rice University, Houston, TX, March 1989.
University of Arizona, Tucson, AZ, March 1989.
Oregon Graduate Center, Beaverton, OR, March 1989.

Tektronix Computer Research Laboratory, Beaverton, OR, March 1989.
University of Colorado, Boulder, CO, March 1989.
IBM TJ Watson Research Center, Yorktown Heights, NY, March 1989.
IBM Systems Integration Division, Owego, NY, March 1989.
New York University, New York, NY, March 1989.
Bell Communications Research, Morristown, NJ, March 1989.

Concurrent Programming With Time

Olivetti Research Center, Menlo Park, CA, December 1987.
IBM TJ Watson Research Center, Yorktown Heights, NY, September 1987.
Columbia University, New York, NY, September 1987.

Short Courses

Trends in Congestion Control and Quality-of-Service: Active queue management on the Internet of the future

Mälardalen University, Sweden, February 2001.

Rate-Based Execution Models For Real-Time Multimedia Computing

Scuola Superiore Santa Anna, Pisa, Italy, September 1997.

Tutorials

Issues in Multimedia Delivery Over Today's Internet

IEEE International Conference on Multimedia Computing Systems, Austin, TX, June 1998.

Systems Issues in the Design and Realization of Desktop Videoconferencing Systems,

Second ACM International Conference on Multimedia, San Francisco, CA, October 1994.