

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

ORACLE CORPORATION

Petitioner,

v.

COMPETITIVE ACCESS SYSTEMS, INC.

Patent Owner.

Case No. IPR2023-00880

U.S. Patent 9,350,649

Declaration of Bill Lin, Ph.D.

TABLE OF CONTENTS

	Page
I. OVERVIEW OF THE TECHNOLOGY.....	5
II. THE '649 PATENT	8
A. District Court Allegations	10
B. Prosecution History	10
C. Priority Date	12
D. Challenged Claims	12
E. Secondary Considerations	12
III. SUMMARY OF THE PRIOR ART	14
A. Invalidity Grounds.....	14
B. US Patent Application Publication No. 2003/0223450 ("Bender") (Ex-1004).....	15
C. US Patent No. 6,160,808 ("Maurya") (Ex-1005).....	16
D. "Multiple Paths Protocol for a Cluster Type Network" ("Mineno") (Ex-1006).....	18
E. RFC793 (Ex-1007).....	19
F. RFC1990 & RFC1661 (Ex-1009 & Ex-1008).....	19
IV. LEVEL OF ORDINARY SKILL IN THE ART AND THE FIELD OF THE ART.....	20
V. CLAIM CONSTRUCTION	22
A. "A method for establishing and sending data during a multipath connection" (claims 1 and 7).....	22
B. "Multipath connection"	23

C.	“the data-source device receiving <i>second</i> multipath information about a <i>second</i> network-edge device that is capable of joining the multipath connection” (claim 3).....	23
D.	“the particular multilink connection” (claim 10)	24
VI.	INVALIDITY	24
A.	Ground 1	24
1.	Motivation to Combine	25
2.	Independent Claim 1	37
3.	Claim 2	68
4.	Claim 3	71
5.	Claim 4	73
6.	Claim 5	77
7.	Claim 6	79
8.	Independent Claim 7	81
9.	Claim 8	95
10.	Claim 9	96
11.	Claim 10	98
12.	Claim 11	99
13.	Claim 12	100
14.	Independent Claim 21	100
15.	Claim 22	110
B.	Grounds 2-3	112
1.	Motivation to Combine	112
2.	Independent Claim 1	124

3.	Claim 2	164
4.	Claim 3	170
5.	Claim 4	173
6.	Claim 5	179
7.	Claim 6	180
8.	Independent Claim 7	181
9.	Claim 8	197
10.	Claim 9	198
11.	Claim 10	199
12.	Claim 11	201
13.	Claim 12	201
14.	Independent Claim 21	202
15.	Claim 22	211
VII.	BACKGROUND AND QUALIFICATIONS	212
A.	Compensation	215
B.	Materials and Other Information Considered	215
VIII.	UNDERSTANDING OF THE LAW	215
IX.	RESERVATION OF RIGHTS	215

I, Bill Lin, declare as follows:

1. My name is Bill Lin. I am a Professor of Electrical and Computer Engineering and an Adjunct Professor of Computer Science and Engineering at the University of California, San Diego. I have prepared this report as an expert witness retained by Oracle. In this report I give my opinions as to whether certain claims of U.S. Patent No. 9,350,649 (“the '649 patent”) are invalid. I provide technical bases for these opinions as appropriate.

2. This report contains statements of my opinions formed to date and the bases and reasons for those opinions. I may offer additional opinions based on further review of materials in this case, including opinions and/or testimony of other expert witnesses. I make this declaration based upon my own personal knowledge and, if called upon to testify, would testify competently to the matters contained herein.

I. OVERVIEW OF THE TECHNOLOGY

3. By the 2001-2002 timeframe, the Internet had exploded in popularity. It was widely used across America and most computers were capable of accessing the Internet. End-user computers were called the “client” in the client-server model that was on the Internet. In this model, the client—typically a desktop PC or a mobile host such as a laptop—requested data from or transmitted data to servers.

For example, client computers requested webpages from servers, which responded by providing the requested content.

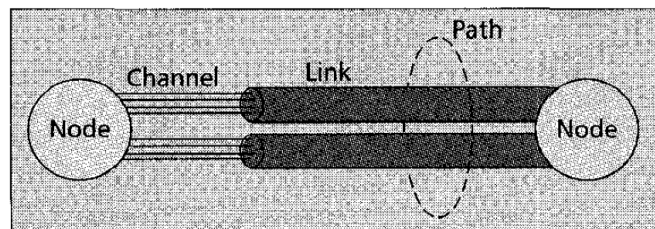
4. Internet servers provided an assortment of content, including web pages and ftp access. One popular use of the Internet was Web browsing, in which client computers requested web pages—along with corresponding images, audio, video, etc.—from servers. By 2001, web browsing had become a mainstream activity for millions of people. As another example, the File Transfer Protocol (“FTP”) was widely-used for transferring files between computers over the Internet, where clients typically connected to FTP servers to request files (e.g., using the “GET” command) or upload files (e.g., using the “PUT” command).

5. The Internet was driven by packet networks that divided data into packets that were then routed to their destination. The Internet Protocol (“IP”) was used for this purpose. Internet-connected computers were assigned an IP address that other devices could use to identify the computer and send data to it.

6. IP was commonly used in conjunction with the TCP protocol—this was referred to as TCP/IP. *E.g.*, Ex-1007, 000008-9, 000013, 000015. TCP was the predominant transport-layer protocol on the Internet and was used by client devices when requesting and receiving web pages from Internet servers. TCP was also used for FTP service. *Id.*, 000015. TCP uses a unique ID to identify each connection

between two endpoints, comprising the IP addresses of the endpoints and the port numbers used. *Id.*, 000011, 000016, 000033, 000085, 000090.

7. Internet connections tended to be slow in the 1990s, which led to various approaches were used in the art to increase bandwidth by aggregating multiple connection paths together, for example using multiple dial-up modem lines or multiple ISDN lines. This approach was known in the art as traffic dispersion, where data traffic was dispersed over multiple parallel paths. Ex-1006, ¶0006-9; Ex-1024, 000001, Fig.3 (explaining how traffic dispersion works):



■ Figure 3. Multiple channels and links between nodes in a network.

8. One approach for traffic dispersion was called multilink PPP. The Multilink PPP (Point-to-Point Protocol) was a protocol used to combine multiple physical links between two endpoints into one logical link, allowing for greater bandwidth and improved resiliency. *See, e.g.*, Ex-1006, ¶0008; Ex-1009.

9. Other approaches used multiple TCP paths. *See, e.g.*, Ex-1006, ¶0017. Indeed, it was common to use concurrent TCP connections. For example, the ubiquitous Netscape Navigator Web browser used multiple concurrent TCP connections. *See* Ex-1020.

II. THE '649 PATENT

10. The '649 patent is a continuation that cites an original application from October 14, 2003, and a provisional application dated October 15, 2002. The patent describes a “Residential Communications Gateway (RCG)” which “is a broadband communications device that combines all voice, data and video communications to and from a typical residence or small business for transmission over a single, or a plurality of Plain Old Telephone Service (POTS) lines ... with, a wireless broadband backbone.” Ex-1001, Abstract. The specification summarizes the alleged invention as the RCG, which is purportedly a ““small stand-alone external modem like device” that “combines the capabilities and features of IP routers, Class 5 circuit switches and wireless LANs.” Ex-1001, 2:32-34. The summary of the invention lists various features of the RCG, including, e.g., a large display, POTS telephone line, Ethernet, Wi-Fi, and support for up to three telephones or fax machines. *Id.*, 2:36-47. “No computer connection is required” but “an Ethernet port, USB port or firewire (IEEE 1394) port is used to connect to a PC or LAN.” Ex-1001, 2:54-59.

11. The '649 patent does not purport to invent the various networking standards that it references. It admits the IEEE 802.11 wireless protocol was “ubiquitous.” Ex-1001, 4:24. It refers to Ethernet and USB. *Id.*, 2:43-45, 2:57-59, 2:67-3:1, 3:32, 9:19-21, 9:58-60, 10:14-15. And it refers to several IETF standards,

including RFC 2543 for message handling (Ex-1001, 7:27-28), Real Time Protocol (RTP) (*id.*, 7:43-47), RFC1990 for multilink PPP (*id.*, 11:4-7), Link Control Protocol (LCP) for PPP session management (*id.*, 11:48-51). These standards were all known in the art in the relevant timeframe.

12. Much of the specification describes the RCG's use of phone lines and support for telephones and fax machines. *See* Ex-1001, cols. 2-10. Column 11 describes connecting RCGs in neighboring homes using Wi-Fi and aggregating their dial-up (POTS) Internet connections using the Multilink PPP standard, as described in RFC1990. *Id.*, 11:3-10, Fig. 7. Accordingly, the specification mostly uses terminology from RFC1990, referring to multilink bundles, etc. *Compare* Ex-1001, 11:59-60, 12:60-61, 12:66, 13:1, 13:5-7, 13:13, 13:36, 13:44, 15:56, 15:66 *with* Ex-1009, 000007 ("Multilink is based on an LCP option negotiation that permits a system to indicate to its peer that it is capable of combining multiple physical links into a 'bundle'.").

13. In January 2013, the IETF published the Multipath TCP standard, RFC6824. Subsequently, in 2014, the applicant filed the application that led to the '649 patent, introducing new claims using new terminology from multipath TCP, for example referring to a "multipath connection" and "connection ID." Neither term appears in the '649 specification.

14. Unlike the Applicant's original 2003 application, which had claims directed at the RCG described in the specification, the claims of the '649 patent attempt to capture broader notions of multipath communications. However, Multilink PPP communications had long been known and taught by Multilink PPP references, such as Bender and Maurya, and using multiple TCP communication paths was already taught by various references including Mineno. *Supra* §I.

A. District Court Allegations

15. In district court, Patent Owner accused a variety of networking devices based on the alleged use of multiple connection paths, including purported support for the Multipath TCP standard. *See, e.g.*, Ex-1010, ¶11, ¶45; Ex-1015, 000003-4, 000011-25, 000043-59, 000076-86; Ex-1011; Ex-1016; Ex-1018.

B. Prosecution History

16. The prosecution history of the '649 patent shows that the Examiner was focused on "multicast" prior art, for broadcasting media over the Internet,¹ rather than prior art teaching RFC1990 Multilink PPP (cited in the specification) or multipath TCP (which PO now accuses). The Examiner's search strategy included "multicast" in 22 out of 28 total searches, not counting searches for the inventor's

¹ Cain states that multicast is used to broadcast data from one device to many, for example for Internet television. (Ex-1027, 1:38-40) (multicasting is "one technique for transporting information from an information provider to a group of information consumers over [a] communication network."), (*id.*, 1:64).

name. Ex-1002, 000451 (8 out of 10), 000528 (10 out of 12), 000555 (4 out of 6 total searches). The Examiner relied on four multicast references in the only two Office Actions: Perkins ((Ex-1002, 000442 (stating that Perkins discloses “multicasts communication stream based on group membership)), Pung ((Ex-1002, 000443 (stating that “Pung discloses... removing the branch from the multicast tree”)), Li ((Ex-1002, 000516 (stating that “Li discloses using multiple links for providing multicast communication”)), and Cain ((Ex-1002, 000516 (stating that “Cain discloses distributing multicast service to different multicast groups”)). The Examiner stated the motivation to combine Perkins with Pung was for “removing an entry in multicast tree table” (Ex-1002, 000443, 000518) and the motivation to combine Perkins, Li, and Cain was “for the benefit of providing reliable multicasting service.” (Ex-1002, 000516).

17. In response, the Applicant distinguished the cited art by arguing that “[t]he whole purpose of Perkins is multicasting[,]” (Ex-1002, 000468); that Li teaches “multicast hosts” (Ex-1002, 000538-9); and Cain teaches “different data is being sent to different multicast groups.” (Ex-1002, 000539). The Examiner did not provide substantive analysis of other prior art, such as Mineno, that teaches the use of parallel TCP connections over multiple communication paths.

C. Priority Date

18. The '649 patent was filed on October 11, 2014. The '649 patent cites a continuation application of U.S. Patent No. 7,606,156, which was filed on October 14, 2003 and a U.S. Provisional Application No. 60/418,521, which was filed on October 15, 2002. For purposes of this Declaration, I have analyzed obviousness from the time period of 2001-2003, and my opinions remain consistent throughout this period that the challenged claims would have been obvious. I do not offer an opinion as to whether the '649 patent claims are entitled to either priority date.

D. Challenged Claims

19. I understand that Petitioner is challenging the validity of claims 1-12 and 21-22 of the '649 patent in the Petition for Inter Partes Review to which this declaration will be attached. Those claims are reproduced in Appendix 3. While this Petition and declaration are directed to the challenged claims, I have considered all claims 1-23 of the '649 patent, as well as portions of the '649 patent prosecution history in forming my opinions.

E. Secondary Considerations

20. It is my understanding that PO may allege secondary considerations of non-obviousness, such as a purported long-felt need, failure of others, copying, praise, and commercial success, based on the adoption of the multipath TCP

standard in 2013.² However, I have seen no evidence of the named inventor being involved with the multipath TCP standard in any way.

21. As explained above, the '649 patent does not discuss multipath TCP, and the claims that use multipath TCP terminology were not introduced until 2014—a year after the adoption of the multipath TCP standard. *Supra* §II. Therefore, the multipath TCP standard did not copy the '649 patent, and the adoption of the multipath TCP standard support any secondary considerations of non-obviousness.

22. Moreover, even if multipath TCP is used in commercial products, I have seen no evidence or justification of any nexus between the success of any commercial products and multipath TCP. Nor have I seen any evidence that multipath TCP is coextensive with a commercial product to the extent that the success of the product could be attributed to multipath TCP. Indeed, multipath TCP is a communication protocol, and therefore it would typically be just one feature of many in a typical commercial product, such as a networking product or operating system. And I have seen evidence that multipath TCP is not an important feature—for example, it is being disabled in Oracle Linux. *See* Ex-1023.

² I note that prior art such as Mineno teaches the use of multiple TCP paths, and therefore the '649 patent cannot be credited with the invention of multipath TCP. *Infra* §VI.B.

23. In short, I have not seen any secondary considerations of non-obviousness that would alter my opinions in this Declaration.

III. SUMMARY OF THE PRIOR ART

24. There are a number of patents and publications that constitute prior art to the '649 patent. I have reviewed and considered the prior art discussed in this section, along with the materials listed in Appendix 2.

A. Invalidity Grounds

25. Based on my review and analysis of the materials cited herein, my opinions regarding the understanding of a person of ordinary skill in the art (“POSITA”) in the relevant timeframe (*supra* §II.C), and my training and experience, it is my opinion that the challenged claims of the '649 patent are invalid based on the following grounds:

Grounds	Claims	Statutory Basis	Prior Art
1	Claims 1-12, 21-22	§ 103	Bender in view of Maurya
2	Claims 1-12, 21-22	§ 103	Mineno in view of RFC793
3	Claims 1-12, 21-22	§ 103	Mineno in view of RFC793, RFC1990, and RFC1661

26. I understand that Bender, Maurya, and RFC1661 were not cited or considered during prosecution of the '649 patent. None of these references or any

related patents are listed on the face of the '649 patent. I reviewed the file history for the '649 patent and am not aware of any of these references being discussed in any office action or in the prosecution history generally. While Mineno, RFC793, and RFC1990 were cited in an IDS (Ex-1002, 000084, 000472), the file history does not include any substantive analysis of these references and instead focuses on other prior art directed to multicast for streaming media. *Supra* §II.B.

B. US Patent Application Publication No. 2003/0223450 (“Bender”) (Ex-1004)

27. I have reviewed the Bender reference. Bender teaches techniques for “aggregating multiple air interfaces to achieve higher data rates using standard link interfaces.” Ex-1004, ¶0008. Bender teaches that “data throughput or data rate can be increased by establishing more than one link between the access terminal and the access network, and using the multiple links to transmit data.” Ex-1004, ¶0005. These techniques for aggregating multiple links utilize the Multilink PPP Protocol, as described in RFC1990. Ex-1004, ¶0005, ¶0026. Bender illustrates these techniques for aggregating multiple links in an example system involving a mobile station (e.g., user equipment, subscriber unit, subscriber station, access terminal, remote terminal, or other device that supports the CDMA standard) and a Packet Data Service Node. Ex-1004, ¶0004, ¶0018, Fig. 1, Fig. 5:

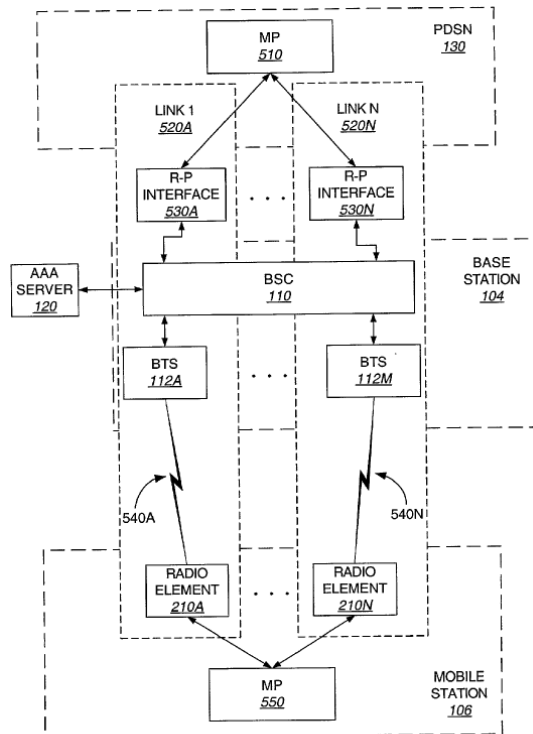


FIG. 5

28. Bender is in the same field of endeavor as the '649 patent (telecommunications and networking). *See infra* §IV. Bender is directed to “communications” and describes various telecommunications and networking standards. Ex-1004, ¶0001-5. Bender relies on the same established standard (e.g., RFC1990) as the '649 patent for aggregating multiple interfaces. *Compare* Ex-1004, ¶0005, ¶0026 with Ex-1001, 11:4-7.

C. US Patent No. 6,160,808 (“Maurya”) (Ex-1005)

29. I have reviewed the Maurya reference. Maurya teaches techniques for “properly handling incoming multi-link PPP packets... appearing on any one of several incoming physical links... thereby providing significantly increased transmission bandwidth.” Ex-1005, 4:56-65. Maurya’s techniques rely on the

“multi-link protocol... referred to as simply multi-link PPP.” Ex-1005, 3:1-8.

Maurya teaches that these techniques are “readily utilized in conjunction with any equipment, such as illustratively a computer or terminal adapter, that can generate multi-link point-to-point (PPP) packets.” Ex-1005, 6:61-66; *see also, id.*, Fig. 6A:

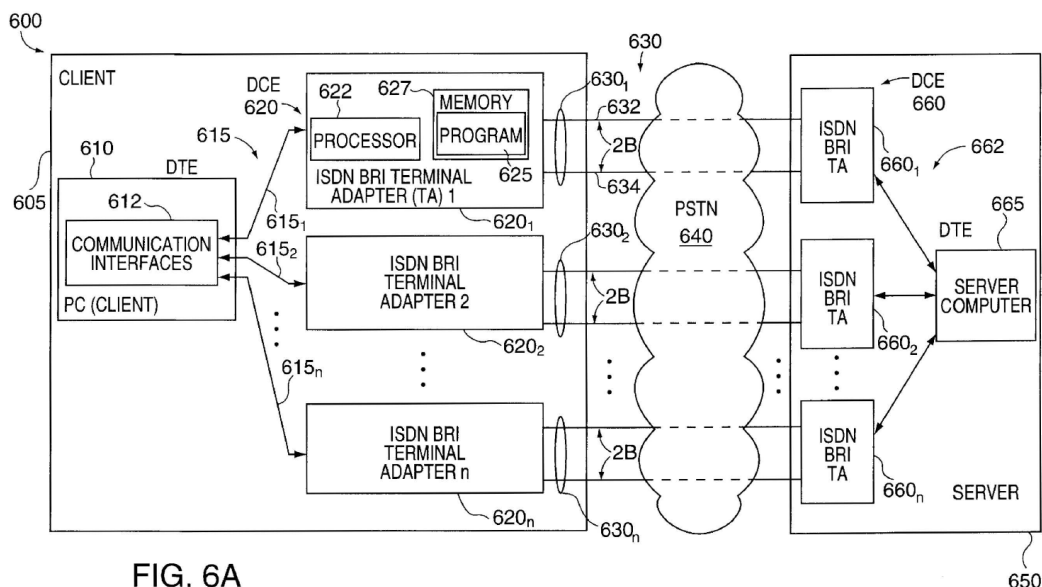


FIG. 6A

30. Maurya is in the same field of endeavor as the '649 patent (telecommunications and networking). *See infra* §IV. Maurya is directed to a multilink “communications technique.” Ex-1005, 1:10-17, 4:56-65. Maurya relies on the same established standard (e.g., RFC1990) as the '649 patent for handling data received through multiple links. *Compare* Ex-1005, 1:10-16, 8:60-9:13, 10:5-11:39, Fig. 5 *with* Ex-1001, 11:4-7.

D. “Multiple Paths Protocol for a Cluster Type Network” (“Mineno”) (Ex-1006)

31. I have reviewed the Mineno reference, which is a research paper published in February 1999 by the IEEE.³ Mineno teaches a SHAKE protocol for “sharing multiple paths in cluster type network.” Ex-1006, ¶0003. SHAKE “provides the functions for composing cluster type network, and dispersing traffic efficiently by measuring transfer rate and round-trip time.” Ex-1006, ¶0001. SHAKE allows an Internet server, referred to as a fixed host, to send data packets to a laptop using multiple TCP paths between them. *E.g.*, Ex-1006, ¶0017, Figs. 1-3:

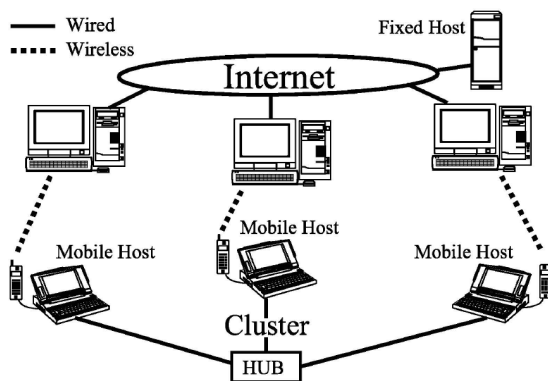


Figure 1: An Example of the System.

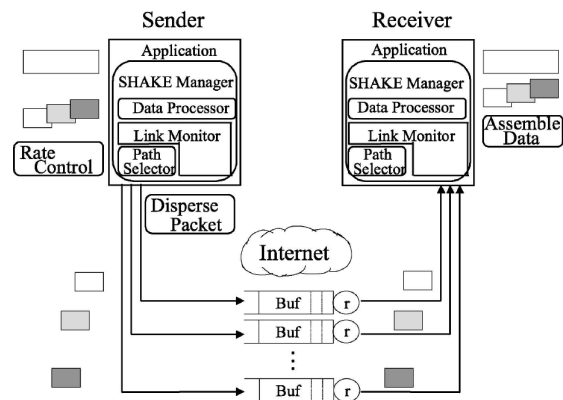


Figure 3: Illustration of SHAKE Modules.

³ I have cited to two PDFs for the Mineno reference. One reference was downloaded from the IEEE website, with paragraph numbers added to facilitate citations, and the other was provided by the IEEE as an attachment to a declaration. I have reviewed both PDFs and concluded that they are substantively the same. *See* Ex-1006; Ex-1025.

32. Mineno also teaches that “[e]ach packet is transmitted to transport layer after it is added a special header of SHAKE[,]” and the SHAKE header includes information such as a packet identifier, link number, sending time stamp, etc. *E.g.*, Ex-1006, ¶¶0026, ¶0035.

33. Mineno is in the same field of endeavor as the ’649 patent (telecommunications and networking). Mineno is directed to improving bandwidth over telecommunications links, and it uses networking standards such as TCP/IP for Internet access. Ex-1006, ¶¶0001-0009, Figs. 1-3.

E. RFC793 (Ex-1007)

34. I have reviewed RFC793, which teaches the TCP standard. RFC793 is in the same field of endeavor as the ’649 patent (telecommunications and networking). *See infra* §IV. It is a networking protocol used on the Internet and a variety of telecommunications networks. It was cited by the applicant in an IDS as relevant to the ’649 patent. Ex-1002, 000084.

F. RFC1990 & RFC1661 (Ex-1009 & Ex-1008)

35. I have reviewed RFC1661 and RFC1990, which teach the PPP and Multilink PPP standards, respectively. RFC1990 refers to and builds on the RFC1661. Together, these references teach the aggregation of multiple links into a single logical bundle. *E.g.*, Ex-1009, ¶000005, 000007-8, 000019.

36. RFC1990 and RFC1661 are in the same field of endeavor as the '649 patent (telecommunications and networking). *See infra* §IV. Both are standards for telecommunications and networking, and RFC1990 is discussed in the '649 specification and cited by the applicant in an IDS as relevant to the '649 patent. Ex-1002, 000472.

IV. LEVEL OF ORDINARY SKILL IN THE ART AND THE FIELD OF THE ART

37. I have analyzed the '649 and determined that the field of the patent is telecommunications and networking. The specification describes the background of the invention primarily in terms of telecommunications and networking, including telephone service (provided by CLECs and LECs), Voice over IP, broadband Internet, DSL, and cable modems. Ex-1001, 1:17-2:28. The summary of the invention discusses telecommunications and networking capabilities of the RCG. Ex-1001, 2:32-3:62. The claims are also directed to telecommunications and networking.

38. In determining the characteristics of a hypothetical POSITA of the '649 patent at the time of the claimed invention, I considered several things, including various prior art techniques relating to the field, the type of problems that such techniques gave rise to, and the rapidity with which innovations were made. I also considered the sophistication of the technologies involved, and the educational background and experience of those actively working in the field at the time. I also

considered the level of education that would be necessary to understand the '649 patent. Finally, I placed myself back in the relevant period of time and considered the engineers and programmers that I have worked with and managed in the field.

39. I came to the conclusion that a person of ordinary skill in the field of art of the '649 patent would have been a person with a bachelor's degree in electrical engineering with at least three years of work experience in the field, or a person with a master's degree in electrical engineering with a specialization in the field. A person with less education but more relevant practical experience may also meet this standard.

40. In the relevant timeframe (*supra* §II.C), a POSITA would have had the knowledge and skills necessary to program and build systems that meet all limitations of the challenged claims, including the following:

- Using computers, including desktops and laptops, to access webpages from servers over the Internet;
- Recognizing and applying the client-server architecture;
- Using and applying concepts from known telecommunication and networking standards, including TCP, PPP, Multilink PPP, IEEE 802.11, IEEE 802.3, cellular systems including CDMA and other known standards; and

- Designing systems for routing packets through packet-switched networks.

V. CLAIM CONSTRUCTION

41. For purposes of this inter partes review, I have considered the claim language, specification, and portions of the prosecution history, to determine the meaning of the claim language as it would have been understood by a person of ordinary skill in the art at the time of the invention. It is my understanding that the “plain and ordinary meaning” has traditionally been applied in district court litigation, where a claim term is given its plain and ordinary meaning in view of the specification from the view point of a person of ordinary skill in the art. I understand that this is referred to as the *Phillips* standard. I have applied the plain and ordinary meaning, or *Phillips* standard, in my analysis.

42. For purpose of this declaration, I have applied the claim constructions discussed below. For all other claim language, I applied the plain and ordinary meaning as would have understood to a POSITA.

A. **“A method for establishing and sending data during a multipath connection” (claims 1 and 7)**

43. This limitation, read literally, would require “[a] method for establishing ... data” and “[a] method for ... sending data” during a multipath connection.” It does not make sense to “establish[] data,” and there is no specification support for this notion. Therefore, a POSITA would have understood

the preamble to mean “a method for establishing a multipath connection and sending data during the multipath connection.” *See, e.g.*, Ex-1001, 5:2-8, 5:31-37.

B. “Multipath connection”

44. PO has stated that a “multipath connection” is a connection that uses multiple network connections to transfer data. Ex-1021, 000006 (“[R]ather than transferring data over a single network connection, data is transferred over multiple connections, i.e. a ‘multipath connection[.]’”), 000005 (“The claimed methods of the ’649 patent enable a ‘multipath connection,’ i.e. a simultaneous connection across multiple network connections ... where data can be routed across the multiple connections.”), 000013, 000017-18. It is my understanding that PO contends that a “multipath connection,” as recited in the claims of the ’649 patent, can be satisfied by a multilink PPP connection (as a connection with multiple PPP paths) or by a multipath TCP connection (a connection that has multiple TCP paths). *See id.*; *see also* Ex-1011; Ex-1015, 000003-4, 000011-25, 000043-59, 000070, 000076-86. I have applied PO’s interpretation to my analysis in this Declaration.

C. “the data-source device receiving *second* multipath information about a *second* network-edge device that is capable of joining the multipath connection” (claim 3)

45. Claim 3, which depends on claim 1, recites “second” multipath information and a “second” network-edge device, even though claim 1 does not

recite a “first” instance of either. A POSITA would have interpreted claim 3 to be satisfied by a data-source device receiving additional (“second”) multipath information about an additional (“second”) network-edge device that is capable of joining the multipath connection.

D. “the particular multilink connection” (claim 10)

46. Claim 10 recites “the particular multilink connection,” which lacks antecedent basis because neither claim 10 nor claim 7 (on which it depends) recites a “particular *multilink* connection.” Claim 7 does recite “the particular *multipath* connection.” For purposes of this declaration, I have assumed that claim 10 contains a typo and was meant to refer to what was recited in claim 7, from which it depends; therefore, I have assumed this term refers to “the particular multipath connection” recited in claim 7.

VI. INVALIDITY

47. Based on my review and analysis of the materials cited herein, my opinions regarding the understanding of a POSTA in the relevant timeframe, and my training and experience, it is my opinion that the challenged claims are rendered obvious by the Grounds explained below.

A. Ground 1

48. It is my opinion that a POSITA would have found Claims 1-12 and 21-22 obvious based on the teachings of Bender and Maurya.

1. Motivation to Combine

49. Bender includes various teachings for multilink communications between a mobile station and a Packet Data Service Node (“PDSN”). *See, e.g.*, Ex-1004, ¶¶0033-43. This includes various aspects of how mobile stations connect with the cellular network, how authentication is handled, and how they establish multiple links with a PDSN, e.g., through the use of the Multilink PPP Protocol (RFC1990). *See, e.g.*, Ex-1004, ¶¶0035-39, ¶0005 (“One multi-link protocol, for use with the Point-to-Point Protocol (PPP), is described in “The PPP Multilink Protocol (MP)”, IETF RFC 1990.”), ¶0025 (“The radio elements 210A-210N are connected to processor 230, which comprises a multi-link protocol, such as Multi-link PPP[.]”), ¶0026 (“The multi-link protocol running on processor 230 may use an end-point discriminator to associate each link with an aggregate bundle, as specified in “The PPP MultiLink Protocol (MP)”... identified as IETF RFC 1990.”), ¶0033 (“In the exemplary embodiment, the multi-link protocol is multi-link PPP[.]”). Maurya provides additional details on the use and operation of Multilink PPP, e.g., for establishing a multilink bundle, adding and terminating links for a multilink bundle, identifying PPP peers, and transferring data. *See, e.g.*, Ex-1005, 9:17-20, 13:61-14:21, 14:61-15:21, 16:6-17:19.⁴ Therefore, a POSITA

⁴ Maurya teaches Multilink PPP, which was known in the art to refer to RFC1990; indeed, Maurya references the teachings in RFC1990. *See, e.g.*,

would have been motivated to combine these complementary teachings by applying Maurya's teachings to Bender's with additional operational details for Multilink PPP, for reasons explained below.

50. Bender and Maurya both teach the use of the Multilink PPP Protocol to increase bandwidth. Ex-1004, ¶0005, ¶0026, ¶0039, ¶0045; Ex-1005, 8:60-9:13, 10:5-11:39, Figs. 1-5. And both references refer to the Multilink PPP Protocol for their operation. Bender teaches handling the multilink bundle in accordance with the Multilink PPP Protocol (RFC1990) and the Point-to-Point Protocol ("PPP") (RFC1661). Ex-1004, ¶0039 ("Reference IETF RFC1990 for details. Also see "The Point-to-Point Protocol (PPP)", IETF RFC1661."), ¶0005, ¶0026, ¶0045. Maurya teaches various details of the Multilink PPP Protocol and the PPP protocol referenced by Bender. Ex-1005, 7:19-8:20, 8:25-53, 8:54-9:13, 9:14-34, 9:35-10:5, 10:6-11:28, 11:29-33, Figs. 1-5.

51. Since Bender leaves some aspects of Multilink PPP operation to the knowledge of a POSITA, it would have been natural and obvious for a POSITA to apply Maurya's teachings on those topics to complement Bender. Multilink PPP

Ex-1005, 8:60-9:13 (referencing Multilink Maximum Receive Reconstructed Unit(MRRU)), 10:5-11:39 (referencing MRRU, long sequence number format, short sequence number format), Fig. 5 (illustrating multilink segment formats); Ex-1009, §§5.1.1 (Multilink MRRU LCP option), 3 (Packet Formats), Fig. 2 (Long Sequence Number Fragment Format), Fig. 3 (Short Sequence Number Fragment Format).

had been known and used in the art for years, and Maurya provides details on how Multilink PPP had been used in the art. *See, e.g.*, Ex-1005, 3:1-13, 13:61-14:21, 14:61-15:21, 16:6-17:19. Therefore, a POSITA looking for details on use of Multilink PPP to add links to a multilink bundle, terminate links in a multilink bundle, identify PPP peers in a multilink bundle, and transfer data through a multilink bundle would have been motivated to apply Maurya's teachings on the topic, particularly to other teachings regarding the Multilink PPP Protocol, as is the case for Bender.

52. For example, Bender teaches merging links in a multilink bundle by setting the "Endpoint Discriminator Option" of those links to specify the same value, e.g., the same IP address.⁵ Ex-1004, ¶0039 ("sending a Link Control Protocol (LCP) option—the Endpoint Discriminator Option—specifying the same <class, address> pair as the <class, address> pair of the links already merged with MP 510."), ¶0045. Maurya teaches the same approach as Bender, with additional details. Maurya teaches, "for each multi-link PPP device connected to one end of a pipe," setting the "endpoint discriminator value to the same value." Ex-1005,

⁵ In accordance with RFC 1990, the Endpoint Discriminator Option "advises a system that the peer on this link could be the same as the peer on another existing link." Ex-1009, §5.1.3 (Endpoint Discriminator Option). The Endpoint Discriminator Option "contains a Class which selects an identifier address space and an Address which selects a unique identifier within the class address space." Ex-1009, §5.1.3. For example, IP address corresponds with class 2. Ex-1009, §5.1.3.

13:47-60. This allows links with the same endpoint discriminator value to be added to the same multilink bundle, and Maurya teaches details on the steps used to establish and use the multilink bundle. Ex-1005, 7:19-8:19, 9:14-10:5, 12:10-65, 13:47-60, 16:8-55. Regardless of whether this approach is conventional or not, the fact that both Maurya and Bender use the same approach provides additional motivation to combine their teachings. *See* Ex-1005, 13:47-60. Therefore, a POSITA would have recognized that Maurya's teachings regarding endpoint discriminators and multilink bundles would have been readily applied to Bender and been motivated to do so. RFC1990 provides additional motivation for this combination. *See, e.g.*, Ex-1009, 000019-23.

53. Furthermore, Maurya provides express motivation to apply its teachings to other Multilink PPP prior art. Maurya teaches that "those skilled in the art will clearly realize that the teachings of the present invention can be readily utilized in conjunction with any equipment ... that can generate multi-link point-to-point (PPP) packets." Ex-1005, 6:61-66. Bender teaches equipment, e.g., mobile stations, base stations, and *Packet* Data Service Nodes, that generate multilink PPP packets. *See, e.g.*, Ex-1004, ¶0003-5, ¶0025-26, ¶0033, ¶0036, ¶0039, ¶0044-45, Fig. 5, Fig. 7. As its name suggests, the Packet Data Service Node sends and receives packets. *Id.* Bender teaches the use of packet-based CDMA interfaces for establishing multiple links between a mobile station and *Packet* Data Service

Node, where the links are combined into a multi-link PPP bundle, and therefore the generation of multi-link PPP packets. *See id.* The data flow between the interfaces of the mobile station and the PDSN are delivered in packets generated by the interfaces and PDSN because the PDSN passes packets received from the radio elements of the mobile station to an external network, such as the Internet.

Ex-1004, ¶0004 (“Data communication can be supported wirelessly between a user and a network, such as the Internet.”), ¶0032 (“PDSN... sinks data to and sources data from... an external network such as the Internet[.]”). Internet traffic uses packets, e.g., Internet Protocol (“IP”) and TCP packets, which would have been well-known to a POSITA given the ubiquitous use of TCP/IP over the Internet in the relevant timeframe. Therefore, a POSITA would have recognized, based on Bender’s teachings of packet-based networks such as the Internet and packet-based cellular standards such as CDMA, that Bender’s mobile station, radio elements, base stations, and PDSN all generate packets in accordance with RFC1990/1661. Ex-1004, ¶0005, ¶0026. Therefore, the references themselves provide express motivation to combine their teachings.

54. Moreover, Maurya’s teachings are not limited to its exemplary embodiment. Ex-1005, 17:58-61 (“... [T]hose skilled in the art can readily devise many other embodiments that still utilize these teachings.”). Instead, Maurya provides broadly-applicable teachings for the Multilink PPP protocol, which can

operate over a variety of physical links, and that is why Maurya provides motivation to apply its teachings to Multilink PPP generally. *See id.*, 6:61-66. Therefore, a POSITA would have been motivated to do so regardless of any purported differences between its embodiment and other Multilink PPP teachings, and a POSITA would not have constrained Maurya's teachings to only the physical interfaces of its embodiment because Maurya expressly teaches that it is not so limited and instead apply to Multilink PPP generally. *Id.*, 6:61-66, 17:58-61.

55. Bender uses the Multilink PPP protocol, and therefore Maurya's teachings would naturally have complemented and benefited Bender's. Indeed, Maurya teaches several advantages to its teachings. For example, Maurya teaches that "setting identical endpoint discriminators is readily accomplished" and allows all links to be viewed "as part of a common bundle. Ex-1005, 5:29-58. One advantage to Maurya's teachings is improved efficiency when transmitting packets over a multilink bundle because "advantageously no changes need to be made to the contents of any multi-link PPP packet to transmit that packet[.]" Ex-1005, 5:59-65. Maurya's teachings further improve bandwidth when transmitting packets because "multi-link PPP packets can be simultaneously transmitted through several high speed communication devices... thus providing significantly higher transmission bandwidths than heretofore possible[.]" Ex-1005, 6:6-12, 4:60-65. Indeed, Maurya teaches that "[o]ne can readily appreciate that, by simultaneously

utilizing available transmission bandwidth provided by multiple links to carry packet traffic... multi-link PPP can significantly increase overall transmission bandwidth for that user.” Ex-1005, 3:36-40. Therefore, a POSITA would have been motivated by the advantages of Maurya’s teachings to combine Maurya’s teachings with Bender’s teachings.

56. Bender utilizes the Multilink PPP Protocol, and Maurya teaches various details for using the Multilink PPP Protocol. For example, Maurya teaches details of how devices in a multilink bundle set identical values for their endpoint discriminator options and details of how data is transmitted through a multilink bundle for improved bandwidth. Ex-1005, 5:7-58 (“[S]etting identical endpoint discriminators is readily accomplished during LCP negotiations wherein a multi-link PPP originator... assigns itself a unique endpoint discriminator value and informs all its immediate PPP peers[.]”), 13:47-60, 3:14-36 (“[A]t a transmitting peer, multi-link PPP... assigns these multi-link PPP packets on an approximately equal basis[.]”), 9:56-10:5 (“[T]he multi-link protocol... assigns these multi-link PPP packets on an approximately equal basis, generally speaking and where practical, between all the links in the pipe.”). Bender uses the multilink PPP protocol to increase bandwidth (Ex-1004, ¶0040), and Maurya’s teachings further Bender’s stated goal by aggregating multiple links into a bundle and then assigning packets on an approximately equal basis across the links. This utilizes the

bandwidth available across the links, spreads traffic over the radio elements, and increases the total available bandwidth for sending packets, effectively aggregating the data rates of the links, as Bender teaches. Ex-1004, ¶0040 (“[D]ata can flow over the virtual link... at the aggregate data rate corresponding to the sum of the data rates of the underlying links[.]”); Ex-1005, 3:14-36, 9:56-10:5. Therefore, a POSITA would have been motivated to apply Maurya’s teachings.

57. A POSITA would also have been motivated to apply Maurya’s teachings on Multilink PPP for terminating links in a multilink bundle. First, as discussed above, Bender and Maurya teach the addition of links into a multilink bundle, and therefore it would have been obvious and natural for a POSITA to look to the inverse operation: removing links, so that the multilink bundle can be adjusted as needed. Second, the termination of links was a known aspect of Multilink PPP operation. Ex-1005, 3:1-14 (“In essence, multi-link operations involves real-time allocation (and deallocation) of any number of these links in the bundle[.]”), 8:4-19, Fig. 1. For example, a link in a multilink bundle is terminated when a failure in the link is detected or when data traffic on the multilink bundle has concluded. Ex-1005, 8:4-19. By terminating a link in a multilink bundle when the link has failed, the multilink bundle operates more efficiently because data traffic is not sent over the failed link, which allows the multilink bundle to operate more efficiently.

58. Additionally, like Maurya, Bender also provides express motivation to modify its teachings and to apply techniques that were known in the art. Ex-1004, ¶0050 (“Various modifications to these embodiments will be readily apparent to those skilled in the art.... Thus, the present invention... is not intended to be limited to the embodiments shown herein but is to be accorded the *widest scope* consistent with the principles and novel features disclosed herein.”).

59. A POSITA would have been motivated to make the combination because the application of Maurya’s teachings would have been complementary to Bender and would not have changed the principle of operation for either reference. As explained above, Maurya expressly applies its teachings to Multilink PPP generally, and Bender uses Multilink PPP. Ex-1004, ¶0005, ¶¶0025-26, ¶0033, ¶¶0035-39; Ex-1005, 8:60-9:13, 10:5-11:39, 13:61-14:21, 14:61-15:21, 16:6-17:19. Moreover, Maurya and Bender both teach the same approach to Multilink PPP: using the same value for the endpoint discriminator for each link in a multilink bundle. Ex-1004, ¶0039, ¶0045; Ex-1005, 7:19-8:19, 9:14-10:5, 12:10-65, 13:47-60, 16:8-55. Both teach using a multilink bundle to improve bandwidth. Ex-1004, ¶0040; Ex-1005, 3:14-36, 9:56-10:5. And Maurya explains that its other teachings, as discussed above and explained for this combination, are conventional for Multilink PPP, as explained below.

60. A POSITA would have had a reasonable expectation of success in combining Bender and Maurya because their teachings are complementary and used for their stated purposes in a conventional manner, as taught by the respective references. Maurya's Multilink PPP teachings would still be used for Multilink PPP in the combination, as explained earlier in this section. *See also*, Ex-1004, ¶¶0005, ¶0026, ¶0039, ¶0045; Ex-1005, 7:19-8:20, 8:25-53, 8:54-9:13, 9:14-34, 9:35-10:5, 10:6-11:28, 11:29-33, Figs. 1-5. The combination would not have changed the principle of operation for either reference, as explained earlier in this section. And Maurya explains that its teachings are not limited to its embodiment, but rather apply to Multilink PPP generally. Ex-1005, 6:61-66, 17:58-61. Both Bender and Maurya teach Multilink PPP. Ex-1004, ¶¶0005, ¶¶0025-26, ¶0033, ¶¶0035-39; Ex-1005, 8:60-9:13, 10:5-11:39, 13:61-14:21, 14:61-15:21, 16:6-17:19. Indeed, the combination follows the express motivation to combine provided by Bender and Maurya, as explained earlier in this section. Therefore, a POSITA would have had a reasonable expectation of success in combining Bender and Maurya because the references themselves instruct a POSITA to do so.⁶

61. A POSITA would have had a reasonable expectation of success in applying Maurya's teachings on aspects of Multilink PPP operations—such as

⁶ It is my understanding that Bender and Maurya, as a prior art patent and patent publication, are presumed to be enabled. Ex-1004; Ex-1005.

adding links to a multilink bundle, terminating links in a multilink bundle, identifying end systems in a multilink bundle, and transferring data through a multilink bundle—to Bender because Maurya teaches these aspects are conventional aspects of Multilink PPP. Ex-1005, 7:19-8:20 (“FIG. 1 diagrammatically depicts conventional states of a link that carries PPP traffic and associated state transitions therefor.”), Fig. 1. For example, adding links to a multilink bundle and identifying end systems in a multilink bundle is “readily accomplished during LCP negotiations” such as during the Establish state. Ex-1005, 5:29-37, 7:19-39. After the multilink bundle is established, then transferring data through the multilink bundle “can then commence” following the Network state. Ex-1005, 7:40-65. When data traffic has concluded, then the links in the multilink bundle can be terminated in the Terminate state. Ex-1005, 8:4-19. Because these states and Maurya’s related teachings are “conventional states” of a link in a multilink bundle, a POSITA would have had a reasonable expectation of success in applying Maurya’s teachings to Bender, which would merely be a combination of known aspects of Multilink PPP operation, with predictable results, and without changing the principle of operation for either Maurya or Bender. A POSITA would have been aware of the Multilink PPP Protocol and understood that references, such as Bender and Maurya, that use the same standard are highly likely to be compatible with each other. *Supra* §IV.

62. The combination of Bender and Maurya would have been a natural and routine application of their teachings—well within the level of ordinary skill for a POSITA given the years of education and work experience explained above. *Supra* §IV. Indeed, the Multilink PPP Protocol was standardized and widely known and used in the art. Ex-1009; Ex-1004, ¶¶0005, 0026; Ex-1005, 8:60-9:13, 10:5-11:39, Fig. 5. Bender and Maurya both use Multilink PPP to increase bandwidth over known telecommunications interfaces. Ex-1004, ¶0040; Ex-1005, 3:14-36, 9:56-10:5. Moreover, a POSITA would have been motivated to make the combination, with a reasonable expectation of success, because it merely combines prior art elements (e.g., regarding Multilink PPP operation) according to known methods (e.g., as taught by Maurya, which applies its teachings to Multilink PPP generally) to yield predictable results (e.g., as taught by Bender and Maurya). It is the use of known techniques (e.g., the teachings of Maurya described above) to improve similar devices (e.g., all using Multilink PPP) in the same way (e.g., to aggregate links in a multilink bundle). And as explained above, the prior art references themselves provide teaching, suggestion, or motivation that would have led one a POSITA to combine their teachings to arrive at the claimed invention. *Infra* Appendix 4.

2. Independent Claim 1

[1pre] A method for establishing and sending data during a multipath connection comprising:

63. I understand a preamble generally does not state a claim limitation. However, to the extent the preamble limits claim 1, it is my opinion that Ground 1 teaches a method for establishing and sending data during a multipath connection (e.g., multilink bundle), which as explained in the sections below, comprises the elements recited in claim 1.

64. Bender and Maurya teach **a method for establishing a multipath connection** (e.g., multilink bundle). *Supra* §V.B (discussing claim construction of “multipath”). For example, Bender teaches establishing a multilink bundle comprising multiple wireless links. Ex-1004, ¶0004. Once established, “data can flow between applications connected with or residing on the access terminal and applications connected to the PDSN.” Ex-1004, ¶0004.

65. Bender teaches methods for establishing a multipath connection that includes an aggregation of multiple air interfaces, or links, as further explained for limitation [1a] below. *Infra* §VI.A.2[1a]; *see also* Ex-1004, Abstract, ¶0001, ¶¶0007-8, ¶0016, ¶0022, ¶0027, ¶0043, Fig. 5. Therefore, Bender establishes a multilink bundle having multiple links, where each link is a distinct path between the mobile station processor and PDSN. *See, e.g.*, Ex-1004, Fig. 5:

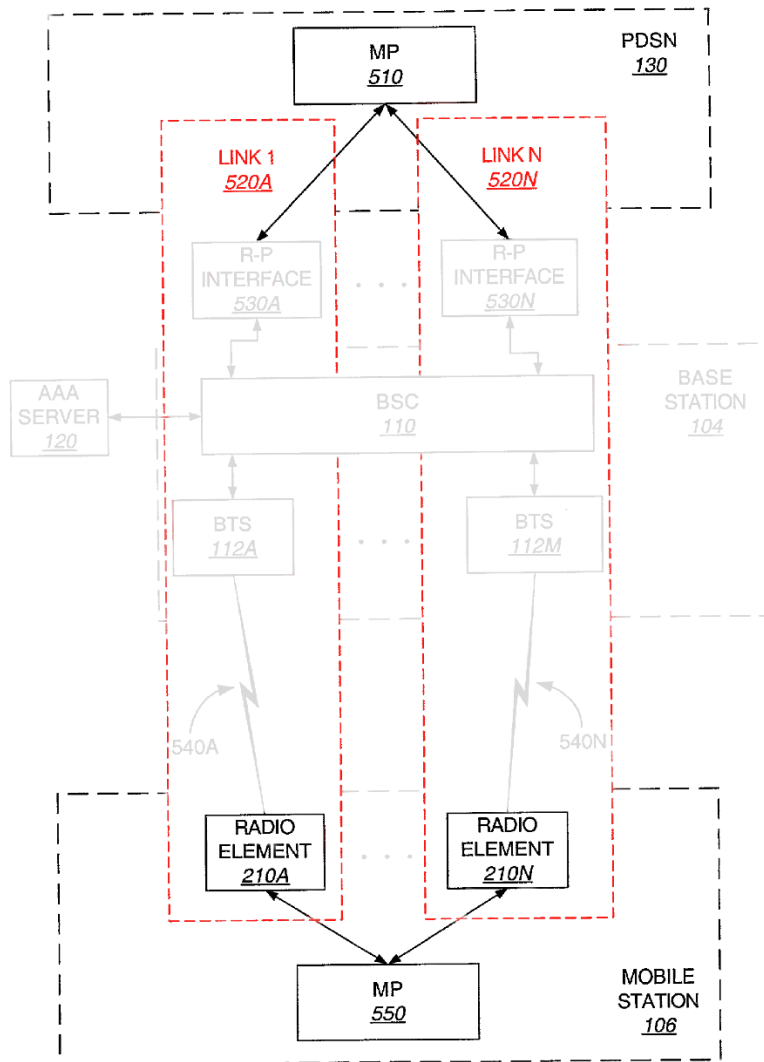


FIG. 5
(Emphasis Added)

66. Bender teaches processes for establishing a multilink bundle in accordance with RFC1990/1661. For example, Bender teaches that “multiple data links can be *set up for aggregation* with a Multi-link Protocol (MP).” Ex-1004, ¶¶0022, ¶¶0001, ¶¶0007-8, ¶¶0016, ¶¶0043. The Multilink PPP Protocol, standardized in RFC1990, and referenced by Bender here, is a known standard for establishing a

multilink bundle. Ex-1004, ¶0005, ¶0026, ¶0045. Thus, Bender teaches a method for establishing a multilink bundle, which satisfies limitations [1pre].

67. Maurya offers complementary teachings, “which permit[] incoming multi-link point-to-point (PPP) packets of a common PPP frame to be simultaneously transmitted across more than one outgoing link in a multi-link bundle.” Ex-1005, Abstract. For example, Maurya teaches that the Point-to-Point Protocol (PPP), which the Multilink Protocol is based on, “is itself formed of three basic constituents” one of which is “a link control protocol (LCP) that is capable of establishing, configuring and testing a link which is to carry PPP packet traffic between two PPP peers[.]” Ex-1005, 1:28-38; Ex-1004, ¶0039; *see also* Ex-1005, 1:32-55, 7:22-39 (“[A] link control protocol (LCP)... exercises responsibility for establishing and configuring the link”), Fig. 1:

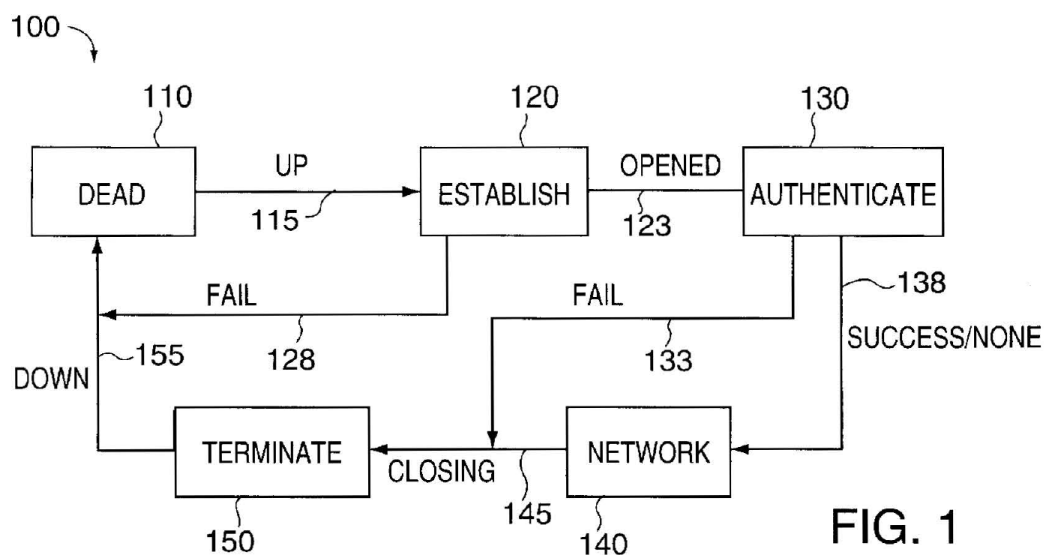


FIG. 1
PRIOR ART

68. In combination with Bender, Maurya teaches the details by which a multilink bundle is established. For example, Maurya teaches that in the Establish state (illustrated in Fig. 1 above), “the LCP, executing on each end of the link and through exchanging and acknowledging LCP Configure packets through the link, forms a PPP connection through the link.” Ex-1005, 1:31-39. A POSITA would have understood Maurya’s teachings for establishing a PPP connection to be compatible with Bender and would have been motivated to combine Maurya’s teachings with Bender because both Bender and Maurya rely on Multilink PPP and PPP. *Supra* §VI.A.1 (explaining how and why the teachings of Bender and Maurya would have been combined, e.g., for the operation of Multilink PPP bundles; analysis incorporated here).

69. Bender and Maurya teach **a method for sending data during a multipath connection**. For example, Bender teaches that “[o]nce these connections making up the link are established, data can flow between applications connected with or residing on the access terminal and applications connected to the PDSN.” Ex-1004, ¶0004, Fig. 7:

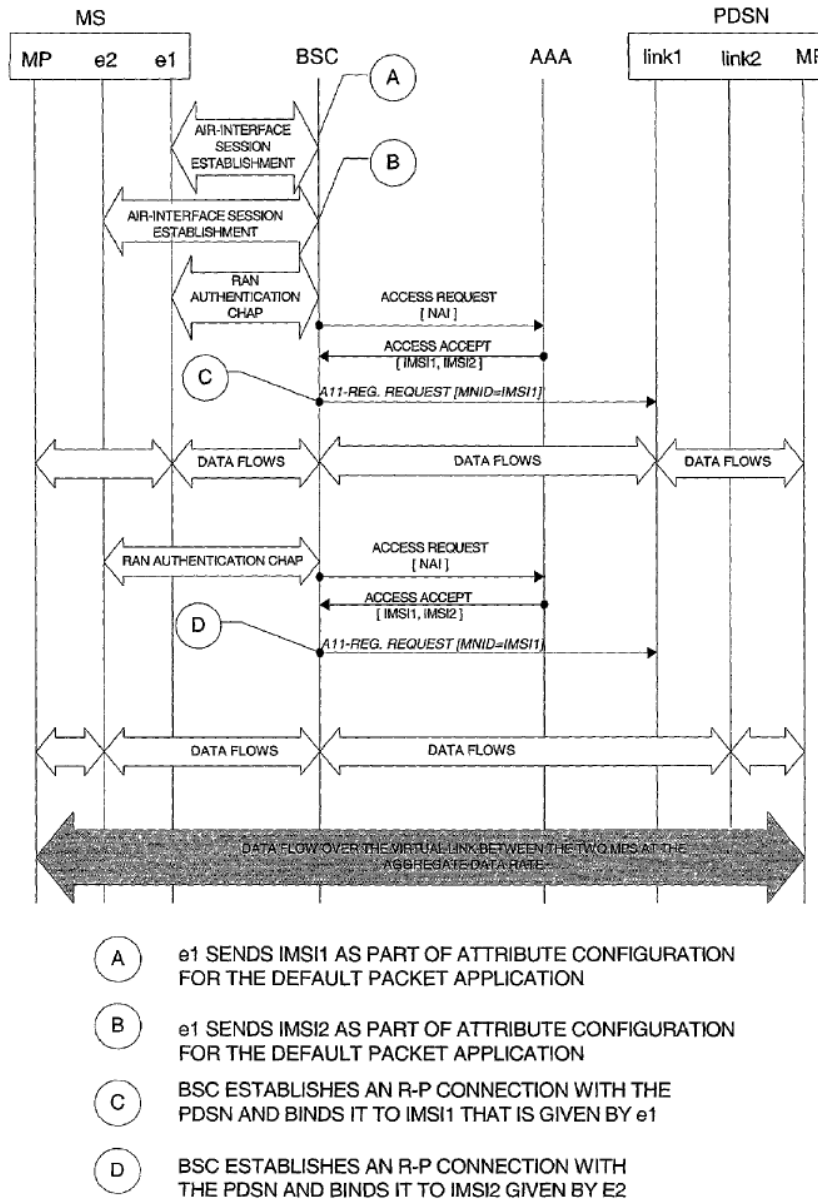


FIG. 7

70. A POSITA would have understood, and indeed Bender teaches, that data flow involves the transmission of data between two ends. Ex-1004, ¶0005 (“The data throughput or data rate can be increased by establishing more than one link between the access terminal and the access network, and *using the multiple links to transmit data between applications.*”), ¶0020. The data is transmitted in

accordance with Multilink PPP. Ex-1004, ¶0027. Therefore, Bender teaches sending data over a multilink bundle once the connections that make up the multilink bundle are established.

71. Maurya teaches complementary techniques for sending data once a multilink bundle is established. For example, Maurya teaches that once the multilink bundle is established, “then PPP packet traffic... can commence over that link.” Ex-1005, 1:59-64. The PPP traffic is sent across all the connections in the multilink bundle. Ex-1005, 9:56-10:5.

72. The PPP traffic is sent across all connections in a multilink bundle until “PPP traffic has concluded” or “either PPP peer desires to terminate the link[.]” Ex-1005, 8:4-19. Therefore, Maurya teaches that, once the multilink bundle is established, PPP traffic is sent across all the connections in the multilink bundle until the PPP traffic concludes or the links are terminated. Thus, Maurya teaches sending data during the multilink bundle.

73. In combination with Bender, Maurya teaches the details by which data is sent through a multilink bundle. In Bender, data flows between two ends once a multilink bundle is established between the two ends. Ex-1004, ¶¶0004-5, ¶0022, Fig. 7. Maurya teaches that data flow over a multilink bundle between two ends is achieved by sending traffic across all the connections in the multilink bundle “on an approximately equal basis[.]” Ex-1005, 3:14-23, 9:56-10. Therefore, the

combination of Bender and Maurya teaches that once a multilink bundle is established, data flows are spread approximately equally over all the connections in the multilink bundle. Ex-1004, ¶¶0004-5, ¶0022, Fig. 7; Ex-1005, 3:14-23, 9:56-10. A POSITA would have been motivated to combine Maurya's teachings with Bender's teachings because Bender's teachings relies on the Multilink PPP Protocol and Maurya teaches details of the Multilink PPP Protocol. *Supra* §VI.A.1 (explaining how and why the teachings of Bender and Maurya would have been combined, e.g., for the operation of Multilink PPP bundles; analysis incorporated here).

74. The steps of the method also include the teachings described below for [1a]-[1d]. Thus, Ground 1 teaches [1pre].

[1a] a data-source device receiving multipath information about each of a plurality of network-edge devices that is capable of joining a multipath connection with a data-requesting device,
--

75. Ground 1 teaches **a data-source device** (e.g., PDSN) **receiving multipath information** (e.g., Endpoint Discriminator Option) **about each of a plurality of network-edge devices** (e.g., radio elements) **that is capable of joining a multipath connection** (e.g., multilink bundle) **with a data-requesting device** (e.g., processor).

76. Starting from the end of this limitation, Bender teaches a **data-requesting device**, e.g., a processor in a mobile station that sends and receives

requested data using a multi-link protocol. Ex-1004, ¶0012, ¶0025 (explaining how a mobile station includes a “processor 230, which comprises a multi-link protocol, such as Multi-link PPP, for aggregating the data from radio elements 210A-210N.”), ¶0027 (“The multi-link protocol running on processor 230 communicates with one or more data applications, delivering data for transmission across radio elements 210A-210N and aggregating data received therefrom[.]”), Fig. 2:

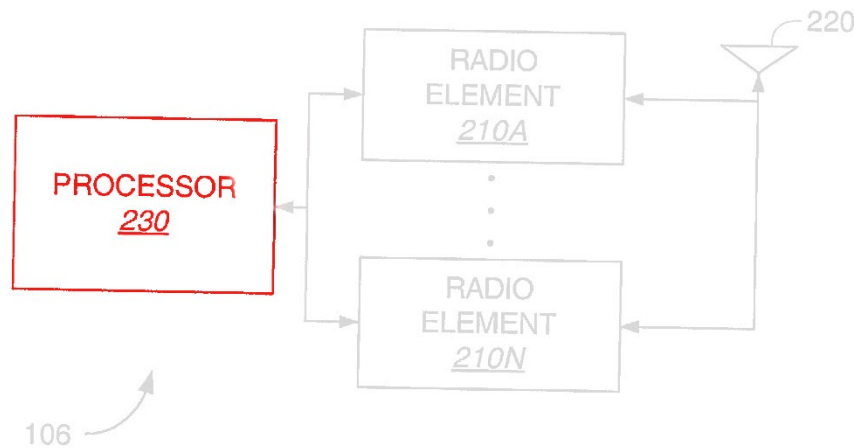


FIG. 2
(Emphasis Added)

Bender teaches the mobile station as, e.g., “a laptop with wireless modem.”

Ex-1004, ¶0043. Laptops were known in the art for browsing the Web, where the processor requests and receives webpage/image data from a server over the Internet, which Bender teaches or at least suggests with the PDSN passing traffic between the mobile station and the Internet. Ex-1004, ¶0032, ¶0004.

77. Bender’s processor “may be a general purpose microprocessor, a digital signal processor (DSP), or a special purpose processor” and “may have

embedded memory, or be connected to a memory (not shown) for storing instructions[.]” Ex-1004, ¶0027. Bender’s processor is a computer that includes memory for storing instructions and a processor for executing them; a POSITA would have recognized this because these basic computer components are taught in high school and undergraduate courses. Therefore, Bender teaches a data-requesting device.⁷

78. Bender’s data-requesting device is connected to **a plurality of network-edge devices**, e.g., two or more radio elements. Ex-1004, ¶0025, ¶0027, ¶0032, ¶¶0033-0042, Fig. 5, Fig. 2:

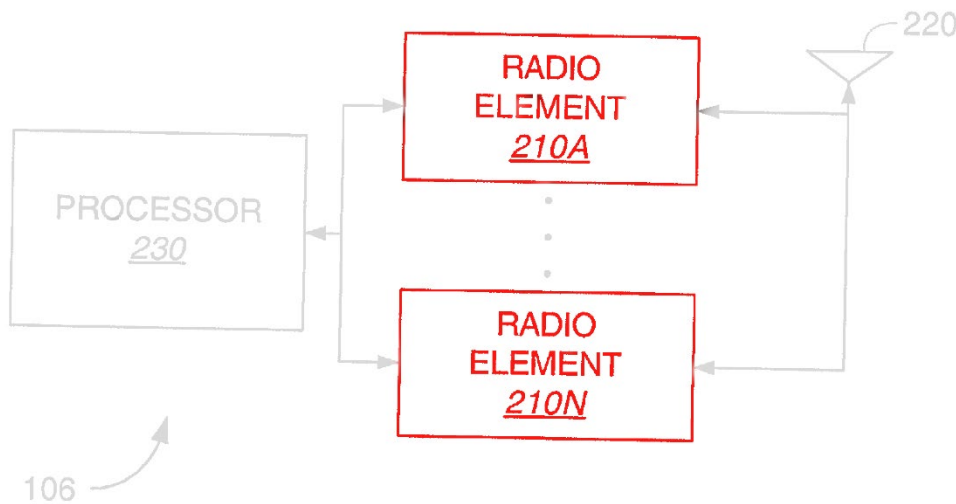


FIG. 2
(Emphasis Added)

⁷ The '649 patent confirms that a handheld phone or computer falls within the scope of a data-requesting device because, for example, claim 19 recites that “the data-requesting device is one of a cell phone, ... a computer, a Wi-Fi device, ... a mobile communications device...” Ex-1001, 18:44-48.

79. Bender teaches that “radio elements are known in the art” and “[e]ach radio element can establish a wireless connection[.]” Ex-1004, ¶0025. As explained below, Bender’s radio elements were used by the processor to access the network and in that sense were at the edge of the network from the processor’s and the network’s perspective. Therefore, Bender teaches the use of known wireless radio elements, which would have included wireless modems, wireless adapters, cellphones, Wi-Fi, hot spot devices, Wi-MAX devices, LTE devices, GSM devices, or other known wireless devices for facilitating communication between mobile stations and base stations. *Id.* A POSITA with basic knowledge of telecommunications and networking would have recognized these to be known radio elements because they were widely used in the art for radio frequency communications (for example for cell phones and Wi-Fi), were commonly used to connect mobile stations to base stations (for example for cellular systems), and were taught in basic undergraduate courses. Therefore, Bender teaches a plurality of network-edge devices.⁸

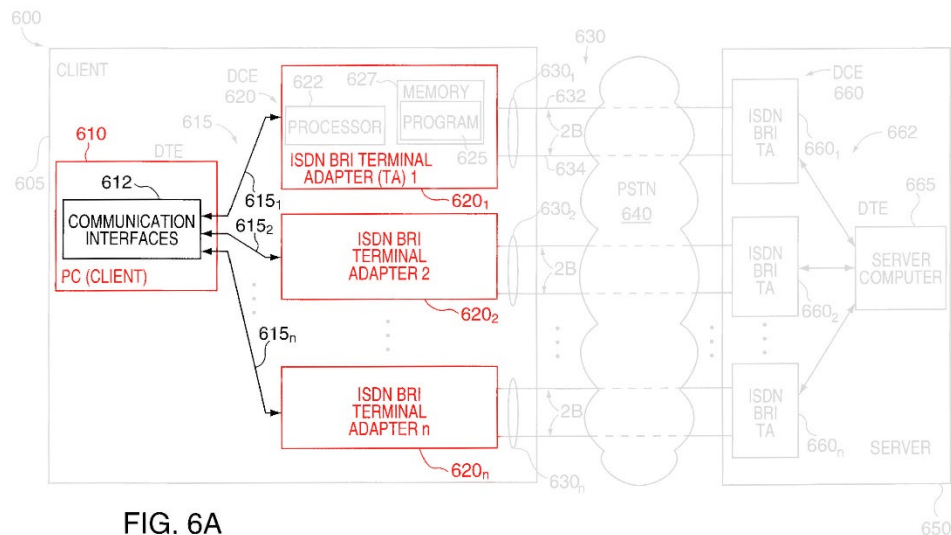
⁸ The ’649 patent confirms that these types of radio elements fall within the scope of the recited network edge device. For example, various dependent claims recite that the network edge devices may be “routers, modems, cellphones, tablets, computers, cellular towers, wireless towers, base stations, wireless base stations, repeaters, wireless repeaters, Wi-Fi devices, networking devices, hot spot devices, Wi-MAX devices, LTE devices, GSM devices, consumer electronic devices, an Internet-ready device, a network switch, a network router, a multipath server and a

80. Additionally, I note that Bender teaches its processor and radio elements as separate devices that may be inside the same mobile station or each in a separate mobile station. Ex-1004, ¶0042. Bender explains that, while “the radio elements 210 were discussed in context of residing on a single mobile station[,] ... [t]he principles discussed herein translate readily to radio elements residing on multiple mobile stations 106.” *Id.* Therefore, Bender expressly applies its teachings to scenarios where the radio elements in the multilink bundle reside on separate mobile stations. *Id.* For example, Bender’s teachings regarding Figure 5 depicts the processor and radio elements residing on a single mobile station, Bender applies those teachings to scenarios where the processor and radio elements are housed separately on multiple respective mobile stations. *Id.*

81. Indeed, Bender teaches that “[a]s long as the multi-link protocol 550 can connect with the various links 520 created, the physical paths of those links 520 are not restricted.” *Id.* Therefore, the processor and radio elements are not limited to a particular arrangement; rather, Bender teaches that the processor, which comprises the multi-link protocol (Ex-1004, ¶0032), need only be able to connect to the radio elements for the processor and the radio elements to be joined into a multilink bundle. Ex-1004, ¶0042.

device capable of providing multipath services.” Ex-1001 18:14-23 (claim 16), 18:48-55 (claim 20).

82. Maurya further reinforces Bender’s teachings and confirms that it was known and obvious in the art for modem-type devices to either be within the same physical arrangement with a processor or separate from the processor and each other. For example, Maurya teaches a client that “is formed of client PC 610 and “multiple... terminal adapters 620” that are connected to the client PC 610 “through internal communication interfaces 612, typically separate serial ports[.]” Ex-1005, 11:40-45, 7:6-13 (explaining the client PC includes, for example, “a personal computer (PC)”), FIG. 6A:



83. The terminal adapters are used to communicate over a link with a server, and each terminal adapter “contains, to the extent relevant, processor 622, typically a conventional microprocessor, and memory 627, the latter having program store 625 which stores computer executable instructions[.]” Ex-1005, 11:49-56. A POSITA would have understood Maurya’s client-server apparatus

teachings to illustrate the conventional nomenclature by which separate devices on a client side are collectively referenced as one entity (e.g., client). This confirms that Bender's processor and Bender's radio elements are separate devices even though Bender occasionally references them collectively as one entity (e.g., mobile station). Thus, as confirmed by Maurya's teachings, Bender's processor and Bender's radio elements are separate devices.

84. Bender teaches that **each of a plurality of network-edge devices** (e.g., radio elements) **is capable of joining a multipath connection** (e.g., multilink bundle) **with a data-requesting device** (e.g., processor). For example, Bender teaches a mobile station with a processor and two or more radio elements, each radio element capable of joining a multipath connection with the processor by forming a multilink bundle. Ex-1004, ¶0026, ¶0032, Fig .5:

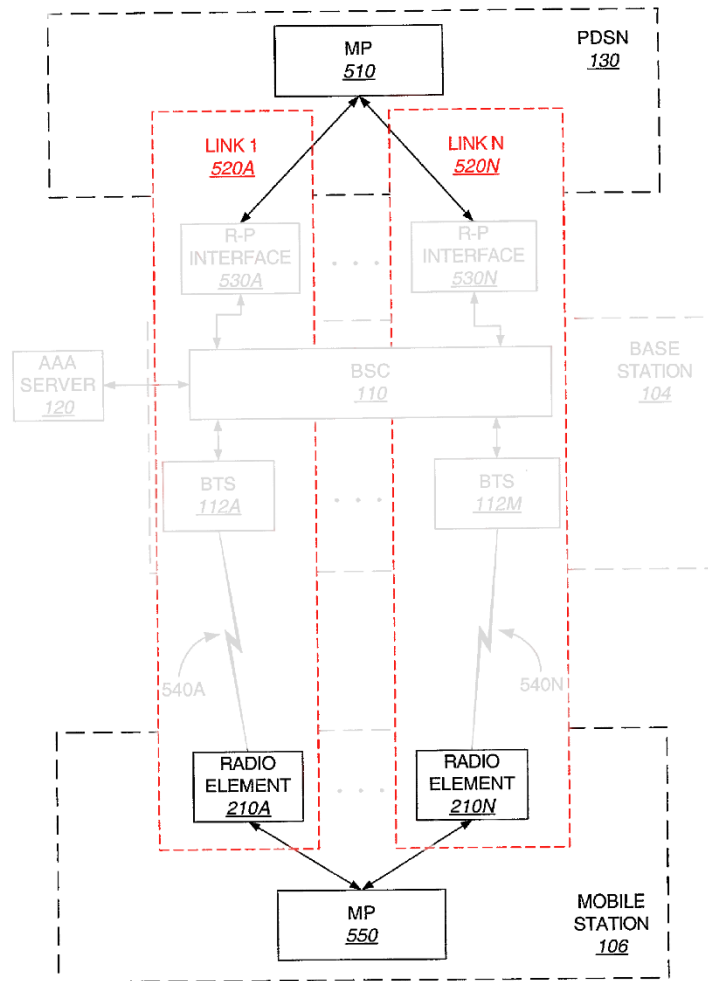


FIG. 5
(Emphasis Added)

85. The processor is connected to radio elements, which, in Fig. 5, are each joined into a multilink bundle, comprising multiple links, Link 1-N, between the processor of a mobile station and a Packet Data Service Node (“PDSN”).

Ex-1004, ¶0032 (“Multi-link Protocol (MP) 510 within PDSN 130 communicates with MP 550 within mobile station 106 through multiple links[.]”). Figure 5 shows a Multi-link Protocol (“MP”) 550 in the mobile station; Bender explains that the

mobile station 106 includes a processor 230 for carrying out the Multilink PPP Protocol. Ex-1004, ¶0032.

86. The processor 230 “may have embedded memory, or be connected to a memory... storing instructions... for aggregating multiple air interfaces[.]” Ex-1004, ¶0027. The links are aggregated together by a multilink protocol 550 on the mobile station side and a multilink protocol 510 on the PDSN side. Ex-1004, ¶0032. Data flows through these links between a processor running a data application in the mobile station and the PDSN; data can pass through the PDSN to and from the Internet 130. Ex-1004, ¶0032 (“MP 550... sinks data to and sources data from a data application (not shown)... MP 510 in PDSN 130... sinks data to and sources data from... an external network such as the Internet[.]”).

87. Bender and Maurya teach a multilink bundle as a connection with multiple connection pathways aggregated into a single “communication channel”; each link is a distinct connection pathway for simultaneous data transfer between a radio element and the PDSN. *Supra* §V.B (explaining claim construction for “multipath”); e.g., Ex-1004, Figure 5. Multiple links are bundled together to act as an aggregated communication channel. Ex-1005, 2:66-3:14 (“[T]wo PPP peers can be connected through multiple physical links in a common bundle... In essence, multi-link operation involves real-time allocation (and deallocation) of any number of these links in the bundle to a *communication channel*[.]”), 3:14-36, 9:56-10:5;

Ex-1004, ¶0040 (“Once multiple links 520 are formed, and merged, then the multi-link session should be complete between MP 550 and MP 510. At this point, data can flow over the *virtual link* between the two MPs, 510 and 550[.]”), ¶0005. Data is spread over the links that make up the multilink bundle on an approximately equal basis. Ex-1005, 3:14-36, 9:56-10:5; *infra* §VI.A.2[1d] (explaining simultaneous transfer of data over the multiple links of a bundle; that analysis is incorporated here). Therefore, each radio element is capable of joining a multilink bundle with the processor of the mobile station.

88. Bender teaches **a data-source device** (e.g., PDSN) **receiving multipath information about each of a plurality of network-edge devices** (e.g., the radio elements explained above). Regarding the data-source device, Bender teaches known Packet Data Service Node, e.g., those specified in the cdma2000 Wireless IP Network Standard. Ex-1004, ¶0004. The PDSN is a source of data for the mobile station processor (when the data application running on the mobile station processor receives data); it is also a source of data to the external network (when the mobile station transmits data). Ex-1004, ¶0032 (“PDSN... sinks data to

and sources data from a data application, or an external network such as the Internet.”).⁹

89. The PDSN receives multipath information about each radio element in the mobile station, which as explained above are capable of joining a multipath connection with a data-requesting device. For example, Bender teaches a process for establishing a multilink communication session. Ex-1004, ¶¶0033-0039, Figs. 6, 7:

⁹ The '649 patent confirms that Bender's PDSN falls within the scope of the recited data-source device. *See infra* §VI.A.7 (claim 6) (analysis incorporated here).

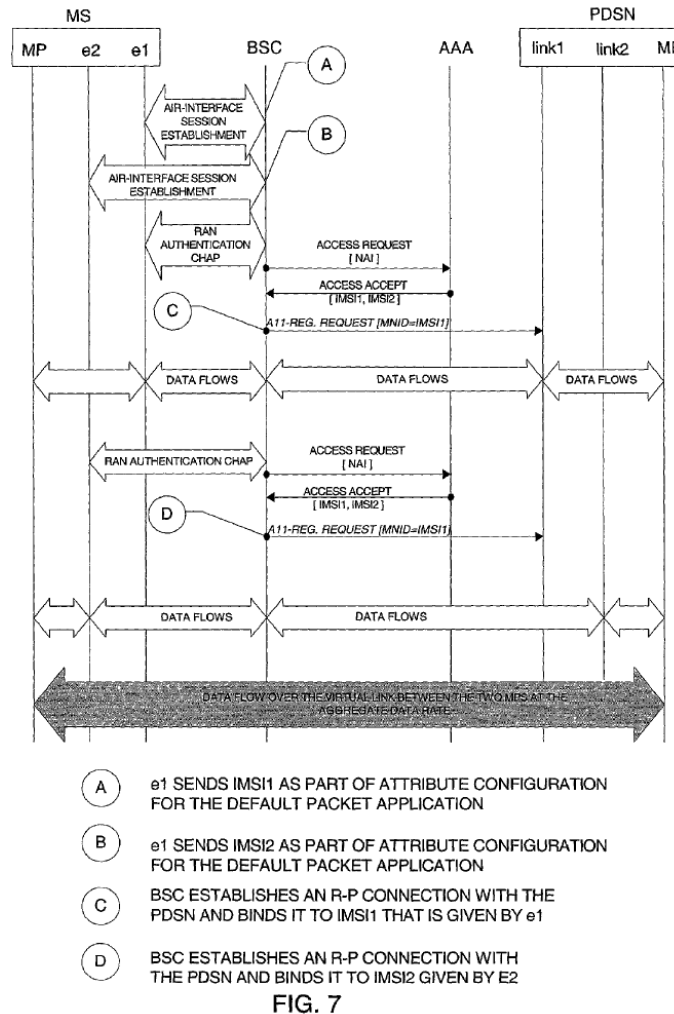


FIG. 7

90. This process includes each radio element establishing a connection with an intermediary base station (Ex-1004, ¶0034), authenticating with the base station (Ex-1004, ¶¶0035-0037), and establishing a connection with the PDSN (Ex-1004, ¶¶0038-0039). When each radio element establishes a link with the PDSN, an Endpoint Discriminator Option for that radio element is send to, and received by, the PDSN. Ex-1004, ¶0039 (“the radio element 210 send[s] a Link Control Protocol (LCP) option—the Endpoint Discriminator Option—specifying the same <class, address> pair as the <class, address> pair of the links already

merged[.]”). The Endpoint Discriminator Option includes, among other things, a class address pair for the radio element that sent it. Ex-1004, ¶¶0039, ¶0045; *see also* Ex-1005, 9:14-55 (describing contents of a “conventional LCP Multi-link Endpoint Discriminator Option packet”). Bender teaches that, by specifying the same class address pair, the radio elements and corresponding links (between the radio elements and the PDSN) are merged together to form a multilink bundle, which as Maurya teaches, is a common communication channel made up of multiple physical links. Ex-1004, ¶¶0039-0040; Ex-1005, 2:66-3:14; *supra* §VI.A.1 (explaining the aggregation of links using the Endpoint Discriminator Option; analysis incorporated here); *infra* §VI.A.3 (claim 2); §VI.A.4 (claim 3). Therefore, the PDSN receives multipath information, e.g., an Endpoint Discriminator Option, about each radio element in the mobile station, and Bender teaches each of these radio elements (e.g., 210A-210N) joining a multilink bundle with a data-requesting device (as discussed above) and the PDSN.

91. The Endpoint Discriminator Option taught by Bender and Maurya is multipath information, and the PDSN receives this multipath information about each of a plurality of network-edge devices. As explained above, the Endpoint Discriminator Option includes a class-address pair, sent from a radio element to the PDSN. This information is used by Bender and Maurya to join the radio element to an existing multilink PPP bundle. Ex-1004, ¶¶0039, ¶0042, ¶0045, Fig. 7. For

example, the PDSN uses this information to merge links into a multilink bundle.

Id.

92. Maurya's complementary teachings confirm this. Maurya teaches that, when "multi-link PPP devices at one end of a pipe... have set their endpoint discriminators to the same value[,]" then "all the links in the pipe will be viewed... as part of a common bundle." Ex-1005, Abstract, 5:29-58, 13:54-60, 14:14-21, 14:50-60. Furthermore, Maurya teaches the address for the <class, address> pair in the Endpoint Discriminator Option is "an Internet Protocol (IP) address" when the class is set to two. Ex-1005, 9:38-50.

93. In combination with Bender, a radio element sends the Endpoint Discriminator Option to a PDSN to provide information about the particular path for that radio element—for example, informing the PDSN that the path to the radio element goes to a particular device having a particular IP address. Ex-1004, ¶0039, ¶0042, ¶0044, Fig. 5; Ex-1005, 9:38-50, 13:54-60, 14:50-60. The PDSN receives an Endpoint Discriminator Option from multiple devices—every radio element for the mobile station that will join the multilink bundle. *Id.* The PDSN uses the Endpoint Discriminator Option to aggregate multiple links into a multilink bundle. Therefore, the PDSN receives multipath information, in the form of an Endpoint Discriminator Option, about each of a plurality of radio elements that are capable of joining the multilink bundle with the PDSN and processor. *Id.*

94. Thus, Ground 1 teaches limitation [1a].

[1b] each of the network-edge devices being connected to one or more networks,
--

95. Ground 1 teaches **each of the network-edge devices** (e.g., radio elements) **being connected to one or more networks.**

96. For example, Bender teaches that each of the radio elements is connected to one or more networks, e.g., internal cellular networks and external networks such as the Internet. Ex-1004, ¶0004, ¶0032, Fig. 5:

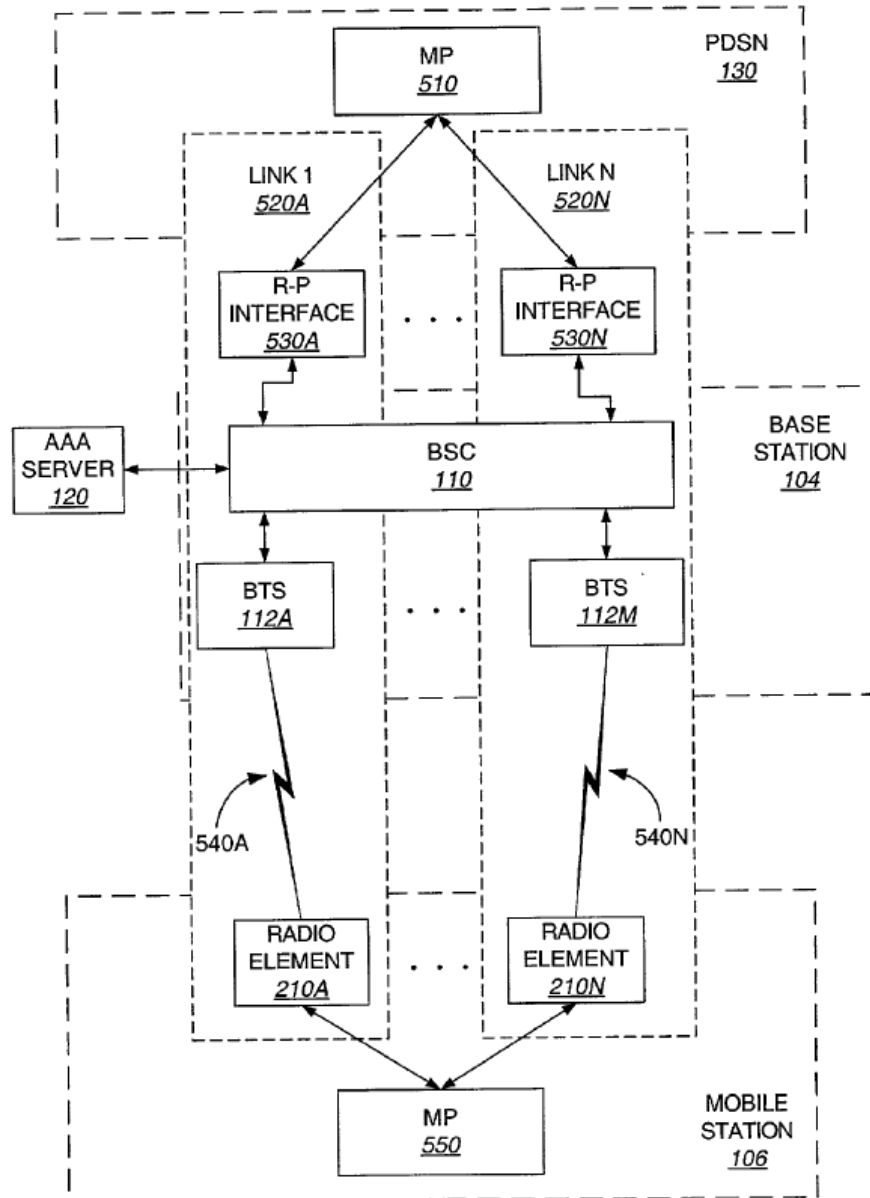


FIG. 5

97. First, each of the radio elements is connected to a cellular network. As Bender explains, each radio element is connected to a cellular network through a base station. Ex-1004, ¶¶0002-4, ¶0021, ¶0025, ¶0032, Figs. 1, 5. The base station is part of a wireless communication network, and Bender teaches the use of known cellular networks including CDMA and TDMA networks. *Id.*, ¶¶0002-3. Base

stations, including BSC and BTS components, were common nomenclature for cellular networks, including as described in the 3G standards cited by Bender. *Id.*

98. Each radio element has a wireless link that connects through the base station to a PDSN, which connects each radio element to a network. Ex-1004, Fig. 5, ¶0032 (“PDSN 130 communicates with MP 550 within mobile station 106 through multiple links.... Each link comprises an air interface, or wireless communication link... established between radio elements... and one or more BTSs[.]”), ¶0004 (“A data link may be comprised of an air interface connection between a wireless access terminal, such as a mobile station, and a base station, as well as an interface connection between the base station and a Packet Data Service Node (PDSN).”). Therefore, Bender teaches that each radio element is connected to a PDSN, which is a node in a network, e.g., a “cdma2000 Wireless IP Network.” *Id.* The PDSN is a known component of a mobile network as per the CDMA2000 mobile technology standards. Ex-1026. Therefore, Bender teaches, or at least suggests, that each radio element is connected to one or more networks because the radio elements are connected to the PDSN, which is a component of a mobile network.

99. Second, the external network provides an independent basis for satisfying limitation [1b] because each of the radio elements is connected to an external network such as the Internet through the PDSN. Ex-1004, Fig. 5, ¶0004

(“Data communication can be supported wirelessly between a user and a network, such as the Internet.”), ¶0032.

100. Links 520A-520N include wireless communication links 540A-540N which connect radio elements 210A-210N to BTSs 112A-112M, continue to BSC 110, and continue further to the PDSN through R-P interfaces 530A-530N.

Ex-1004, ¶0032. Therefore, the links connect the radio elements to the PDSN, which provides access to an external network such as the Internet.

101. Thus, Ground 1 teaches limitation [1b].

[1c] and each of the network-edge devices being connected to the data-requesting device over a different connection; and
--

102. Ground 1 teaches **each of the network-edge devices** (e.g., radio elements) **being connected to the data-requesting device** (e.g., processor) **over a different connection**.

103. Bender teaches that the processor in the mobile station runs a “multi-link protocol, such as Multi-link PPP” (Ex-1004, ¶0025), which is connected to an MP within the PDSN over different links, with a separate link for each radio element. Ex-1004, ¶0025, ¶0027, ¶0032 (“Multi-link Protocol (MP) 510 within PDSN 130 communicates with MP 550 within mobile station 106 *through multiple links, Link1-Link N, 520A-520N, respectively. Each link comprises an air interface,*

or wireless communication link, 540A-540N, established between radio elements 210A-210N”), Fig. 2, Fig. 5:

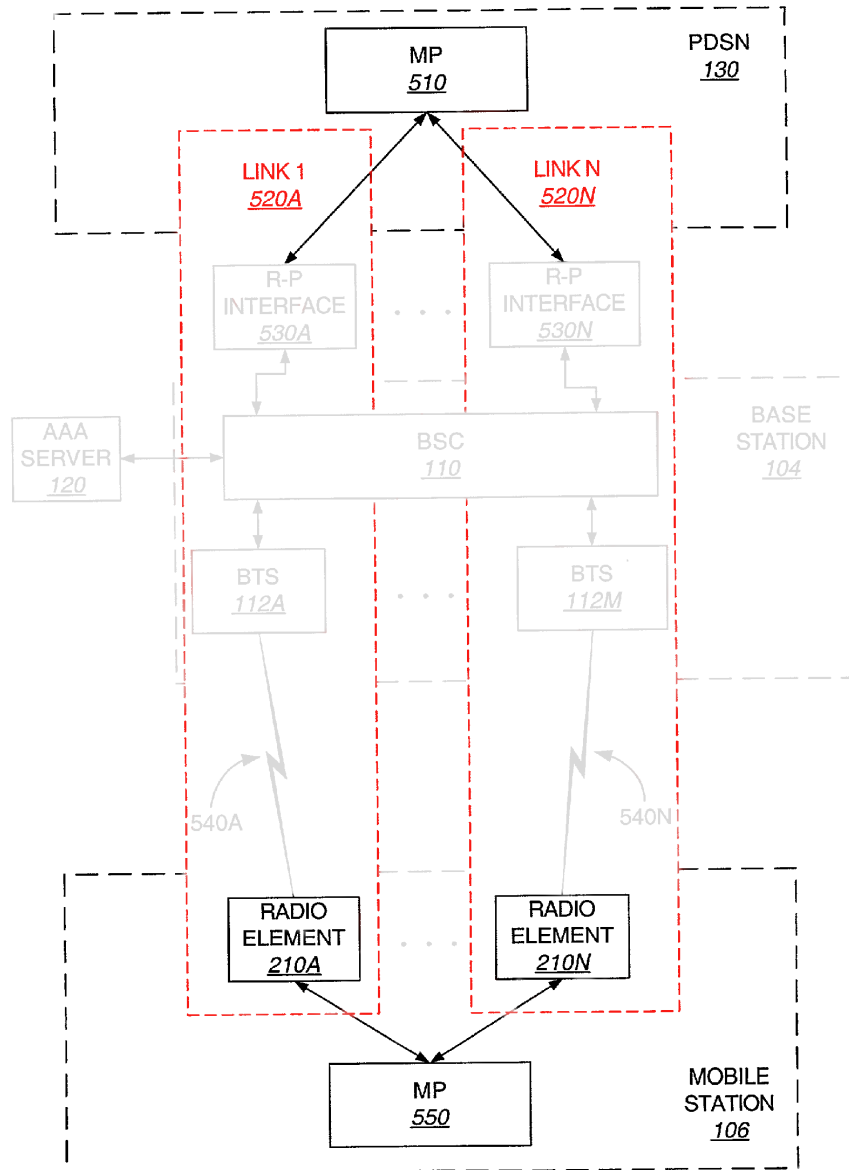


FIG. 5
(Emphasis Added)

104. As illustrated in Figure 5 and explained previously, the radio elements 210A-210N are connected to one or more networks, each radio element being part

of a separate link (links 520A-520N). *Supra* §VI.A.2.[1c]. The radio elements 210A-210N are connected to MP 550, which refers to the Multilink PPP Protocol carried out by the processor, over different connections from each other. Ex-1004, ¶0032, Fig. 5. Therefore, each radio element is connected to the processor over a different connection, including different physical and/or logical connections. *Id.*

105. To the extent Patent Owner argues that limitation [1c] requires each of the network-edge devices to be connected to the data-requesting device over a different connection from those described in limitation [1b] (the connection(s) between the network edge devices and one or more networks), Bender also teaches this. For example, each of the radio elements is connected to the processor on one side (highlighted orange below) which is different from the connections between the radio elements and one or more networks, explained above for limitation [1b] (highlighted blue below). *See* Ex-1004, Fig. 5:

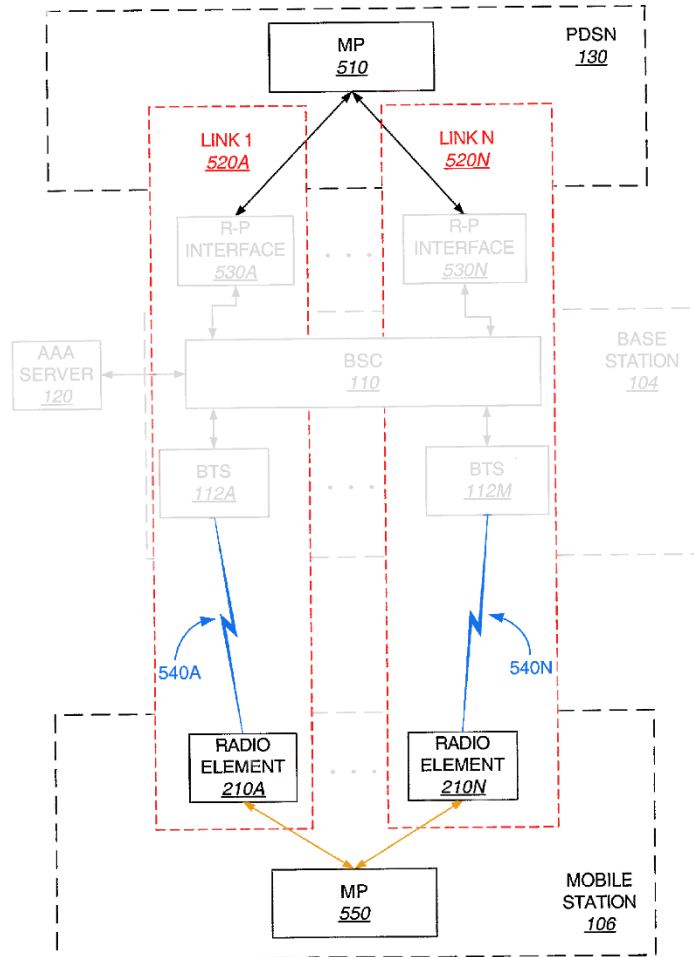


FIG. 5
(Annotations/Emphasis Added)

106. Additionally, Bender teaches, for Figure 2, that “[t]he radio elements 210A-210N are connected to processor 230” and separately “connected to antenna 220” for wireless communication. Ex-1004, ¶0025, Fig. 2.

107. Thus, Ground1 teaches limitation [1c].

[1d] the data-source device performing at least one of (i) sending different groups of data packets to different network-edge devices that are participating in the multipath connection, (ii) sending different streams of data to one or more of the network-edge devices that are participating in the multipath connection, and (iii) sending combinations of different streams and different groups of data packets to one or more of the network-edge devices that are participating in the multipath connection.

108. Ground 1 teaches this limitation. As an initial matter, limitation [1d] recites sub-parts (i), (ii), and (iii). I understand that limitation [1d] is satisfied when any one of these sub-parts is satisfied. Ground 1 teaches sub-part (i).

109. Ground 1 teaches **the data-source device** (e.g., PDSN) **performing (i) sending different groups of data packets to different network-edge devices** (e.g., radio elements) **that are participating in the multipath connection** (e.g., multilink bundle). For example, Bender teaches the PDSN sending packets to each of the mobile station's radio elements. Ex-1004, ¶¶0022, ¶¶0032, ¶¶0040. The links between the PDSN and radio elements are aggregated into a multilink bundle, and the flow of data over the multilink bundle has an “aggregate data rate corresponding to the sum of the data rates of the underlying links[.]” Ex-1004, ¶¶0040, ¶¶0025-26.

110. Therefore, Bender teaches sending data over each link in the multilink bundle, with packets split between the links, so that their data rates are aggregated together. *Id.* For example, given two links having a data rate of 100 kbps each,

Bender's teachings allow the links to be aggregated into a multilink bundle with an aggregate data rate of 200 kbps by sending half of the data packets down one link and the other half down the other link. This is, at least, suggested and obvious because Bender teaches aggregating (combining) links to aggregate their individual data rates. In contrast, if the same packets were sent over each link (duplicating the data), then the data rates of the individual links would not be aggregated because the data rate for the combined bundle would be the same as an individual link. Therefore, Bender allows the aggregate data rate of the links in the multilink bundle to be utilized by teaching the PDSN sending different groups of data packets to each of the radio elements in the multilink bundle, e.g., group A being the data packets sent over Link 1 520A to Radio Element 210A, group B being the data packets sent over Link 2 520B to Radio Element 210B, and continuing up to group N being the data packets sent over Link N 520N to Radio Element 210N. *See, e.g.,* Ex-1004, ¶0025, ¶0027, ¶0032, ¶0040, Fig. 5.

111. The combination of Bender and Maurya further teaches [1d]. *Supra* §VI.A.1 (explaining how and why the teachings of Bender and Maurya would have been combined, e.g., for Multilink PPP operations; analysis incorporated here). Maurya teaches complementary techniques for sending data through a multilink bundle, for example by sending packets from server to client, and by assigning data

packets “on an approximately equal basis” between all the links in the multilink bundle. Ex-1005, 3:14-36, 3:46-53, 9:27-30, 9:56-64, 16:48-55.

112. The packets are received and reassembled by the receiving peer. Ex-1005, 9:67-10:5.

113. Therefore, Bender and Maurya operate in a similar fashion, by aggregating links in a multilink PPP bundle and sending different groups of packets over each link to aggregate their data rates. *See, e.g.*, Ex-1004, ¶¶0025-0026, ¶0032, ¶0040; Ex-1005, 3:14-32, 3:46-53, 9:56-10:5. Maurya’s teachings complement Bender’s teachings with additional details on the assignment of packets to each link, which although taught by Bender are more explicitly described by Maurya. The combination teaches the PDSN sending different groups of data to different radio elements by dividing the packets, on an approximately equal basis, between the links in the bundle. Ex-1005, 3:14-32, 3:46-53, 9:56-10:5. This aggregates the data rates of the individual links, as Bender teaches (Ex-1004, ¶¶0025-0026, ¶0040) and results in the PDSN sending different groups of data packets to each of the radio elements in the multilink bundle, e.g., group A being data packets sent over link 1 to Radio Element 210A, group B being data packets sent over link 2 to Radio Element 210B, and so on. *See, e.g.*, Ex-1004. ¶0040,

Fig. 5.¹⁰ Each radio element receives a different group of data packets, compared to the other radio elements.

114. At the very least, the combined teachings of Bender and Maurya suggest, and it would have been obvious, sending different groups of packets to different radio elements in the multilink bundle because doing so increases the bandwidth available to the multilink bundle, a goal shared by both Bender and Maurya. Ex-1004, ¶0040; Ex-1005, 3:14-36, 9:56-10:5. To do otherwise, by sending the same packets to each radio element, would create duplicate traffic that negates any potential bandwidth increase, frustrates the goals of Bender and Maurya. Thus, the combination of Bender and Maurya teaches sending different groups of data packets to different network-edge devices that are participating in the multipath connection.

115. Thus, Ground 1 teaches limitation [1d].

¹⁰ The '649 patent confirms that the teachings from Bender and Maurya, described above, fall within the scope of the claims. The written description for this limitation is a description of packets being sent over the different links of a multilink PPP bundle. Ex-1001, 12:44-48.

3. Claim 2

[2] **The method of claim 1, wherein the multipath information includes a connection ID that is generated by one of the data-source device, one of the plurality of network-edge devices, the data-requesting device and a network device.**

116. Ground 1 teaches **the method of claim 1 wherein the multipath information** (e.g., Endpoint Discriminator Option) **includes a connection ID** (e.g., <class, address> pair) **that is generated by one of the data-source device** (e.g., PDSN), **one of the plurality of network-edge devices** (e.g., radio elements), **the data-requesting device** (e.g., processor) **and a network device.**

117. As an initial matter, claim 2 recites a list of devices that may generate the connection ID. I understand that claim 2 is satisfied when any one of these devices performs the recited step because claim 2 only recites “one of” these devices. Therefore, Bender and Maurya teach claim 2 because Bender and Maurya teach the processor generating the connection ID in the form of a <class, address> pair in the Endpoint Discriminator Option.

118. The Endpoint Discriminator Option includes a <class, address> pair. *Supra* §§VI.A.2[1a]-[1b]. Bender teaches adding links into a multilink bundle by “specifying the same <class, address> pair as the <class, address> pair of the links already merged[.]” Ex-1004, ¶0026, ¶0039, ¶0045. As links are merged into a multilink bundle, the links are set to have the same <class, address> pair specified in their Endpoint Discriminator Option. In short, Bender teaches that all links in a

multilink bundle (e.g., 1 to N) have the same <class, address> pair. The <class, address> pair therefore identifies the multilink connection, along with all the links within the multilink connection, and it is the connection ID for the multilink bundle.

119. The combination of Bender and Maurya further renders claim 2 obvious with additional details related to generating the same <class, address> pair for all connections in a multilink bundle. *Supra* §VI.A.1 (explaining how and why the teachings of Bender and Maurya would have been combined, e.g., regarding the Endpoint Discriminator Option; analysis incorporated here). Maurya teaches “a server or a client application executing at a PC, assigns itself a unique endpoint discriminator value and informs all its immediate PPP peers” so that “all the links... will be viewed... as part of a common bundle.” Ex-1005, 5:7-58, 13:47-14:60, 16:8-55, Figs. 6B, 7. Each of these peers, “assumes the value that has been provided to it” and “in turn, sends an LCP Endpoint Discriminator packet” containing its discriminator value... to its immediate PPP peers, and so forth.” *Id.* In short, Maurya teaches that an application executing on a server or client computer—the two sides of Maurya’s multilink connection—generates a unique endpoint discriminator value (e.g., <class, address> pair) for the Endpoint Discriminator Option, which uniquely identifies the multilink bundle and the links within it, allowing all links with that same endpoint discriminator value to operate

as part of a common bundle. *Id.* Thus, Maurya teaches an application, executing on one of the sides of the multilink bundle, generating a unique value (class-address pair) for the endpoint discriminator value option, that which identifies the multilink bundle.

120. Bender and Maurya use the same Endpoint Discriminator Option for the same purpose. *Supra* §VI.A.1. Both teach the Endpoint Discriminator Option includes a class field and an address field, which Maurya teaches is, for example, “an Internet Protocol (IP) address[.]” Ex-1005, 9:38-49, 16:11-17, 16:25-30, Fig. 4; Ex-1004, ¶0026, ¶0039, ¶0045. This Endpoint Discriminator Option was known in the art and described in RFC1990, cited by Bender and referenced by Maurya. Ex-1004, ¶0005, ¶0026, ¶0039, ¶0045; Ex-1005, 8:60-9:13, 9:38-49, 10:5-11:39, Figs. 1-5.

121. Applying Maurya’s teachings to Bender, the two sides of Bender’s multilink PPP bundle are the PDSN and mobile station (Ex-1004, Fig. 5) which correspond to Maurya’s teachings of a server and client, respectively. The mobile station includes a processor that runs a multilink PPP “application” (Ex-1004, ¶0027, ¶0039, ¶0045), which corresponds with Maurya’s teaching of a client running an application (Ex-1005, 5:29-42, 13:47-14:20, 16:8-17, Fig. 6B). In this context, Maurya teaches the processor (on the mobile station) generating a unique endpoint discriminator value containing the <class, address> pair for identifying a

multilink bundle. Ex-1005, 5:7-58, 9:38-49, 13:47-14:60, 16:8-55, Figs. 4, 6B, 7.

The endpoint discriminator value containing the <class, address> pair is provided to the radio elements via the Endpoint Discriminator Option. Ex-1004, ¶0039, ¶0045; Ex-1005, 5:29-42, 13:47-14:20, 16:8-17. The radio elements assign themselves the <class, address> pair generated by the processor, allowing them to specify the same <class, address> pair when establishing their respective links with the PDSN and be merged into a multilink bundle. Thus, the combination of Bender and Maurya teaches the <class, address> pair is generated by the processor, which satisfies the data-requesting device.

122. Even if the PDSN or another device on the network were to generate the connection ID, claim 2 also allows this by reciting “the data-source device” and “a network device” as options.

123. Thus, Ground 1 teaches limitation [2].

4. Claim 3

[3]	The method of claim 1, further comprising: the data-source device receiving second multipath information about a second network-edge device that is capable of joining the multipath connection; and the data-source device joining the second network-edge device to the multipath connection.
------------	--

124. Ground 1 teaches the method of claim 1 and further teach the additional limitations of claim 3. *Supra* §V.C (explaining claim construction).

125. The combination of Bender and Maurya teaches **the data-source device** (e.g., PDSN) **receiving second multipath information** (e.g., Endpoint Discriminator Option) **about a second network-edge device** (e.g., radio element) **that is capable of joining the multipath connection** (e.g., multilink bundle) **and the data-source device** (e.g., PDSN) **joining the second network edge device** (e.g., radio element) **to the multipath connection** (e.g., multilink bundle). *Supra* §VI.A.1 (explaining how and why the teachings of Bender and Maurya would have been combined, e.g., for adding links into a multilink bundle; analysis incorporated here).

126. For example, as explained for limitation [1a], Bender and Maurya teach the PDSN receiving an Endpoint Discriminator Option about each of a plurality of radio elements that is capable of joining a multilink bundle and using the Endpoint Discriminator Option to add each radio element and corresponding link into the multilink bundle. *Supra* §VI.A.2[1a]; Ex-1004, ¶¶0026, ¶¶0039, ¶¶0045; Ex-1005, Abstract, 5:29-58, 13:54-60, 14:14-21, 14:50-60. Bender explains that this process is repeatable to add additional links to the multilink bundle, for example up to the N radio elements and N links shown in Figure 5. Ex-1004, ¶¶0034, ¶¶0035, Fig. 5. Therefore, while Bender provides one example of a multilink bundle having two links, in the context of Figure 7, Bender teaches that the process is repeated to add additional links, for example up to the N links shown in

Figure 5. Therefore, the PDSN receives second multipath information about a second network-edge device that is capable of joining the multilink bundle because the process for adding additional links to the bundle includes the steps described for initially establishing the multilink bundle, including the step of providing an additional Endpoint Discriminator Option (with the same class-address pair) from the additional radio element to the PDSN. *Supra* §VI.A.2[1a].

127. Thus, Ground 1 teaches claim [3].

5. Claim 4

[4] **The method of claim 1, further comprising:**

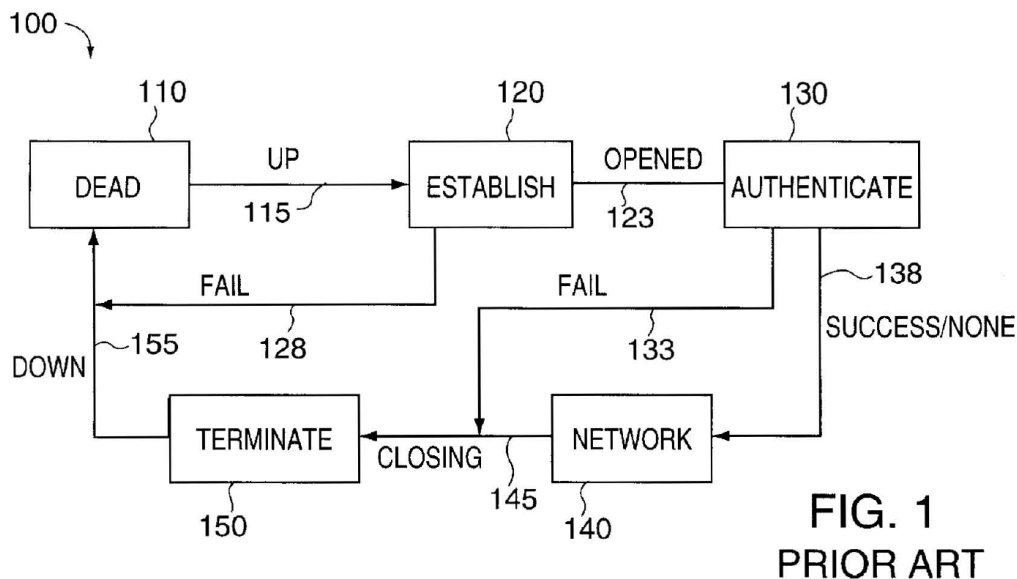
 the data-source device receiving multipath information about which network-edge devices are to be removed from the multipath connection; and

 the data-source device removing the network-edge devices from being associated with the multipath connection.

128. Ground 1 teaches the method of claim 1 and the limitations of claim 4. *Supra* §VI.A.1 (explaining how and why the teachings of Bender and Maurya would have been combined, e.g., for link termination; analysis incorporated here).

129. Bender and Maurya teach **the data-source device** (e.g., PDSN) **receiving multipath information** (e.g., PPP terminate packets) **about which network-edge devices** (e.g., radio elements) **are to be removed from the multipath connection** (e.g., multilink bundle). For example, Bender and Maurya both teach the operation of a multilink bundle in accordance with RFC1990/1661.

Ex-1004, ¶0005, ¶0026, ¶0039, ¶0045; Ex-1005, 7:19-8:20, 8:25-53, 8:54-9:13, 9:14-34, 9:35-10:5, 10:6-11:28, 11:29-33, Figs. 1-5. Maurya teaches that PPP peers “exchange appropriate PPP terminate packets” and “acknowledgments thereof” before terminating a link, bringing the link to a “Terminate state[.]” Ex-1005, 7:29-8:19, Fig. 1:

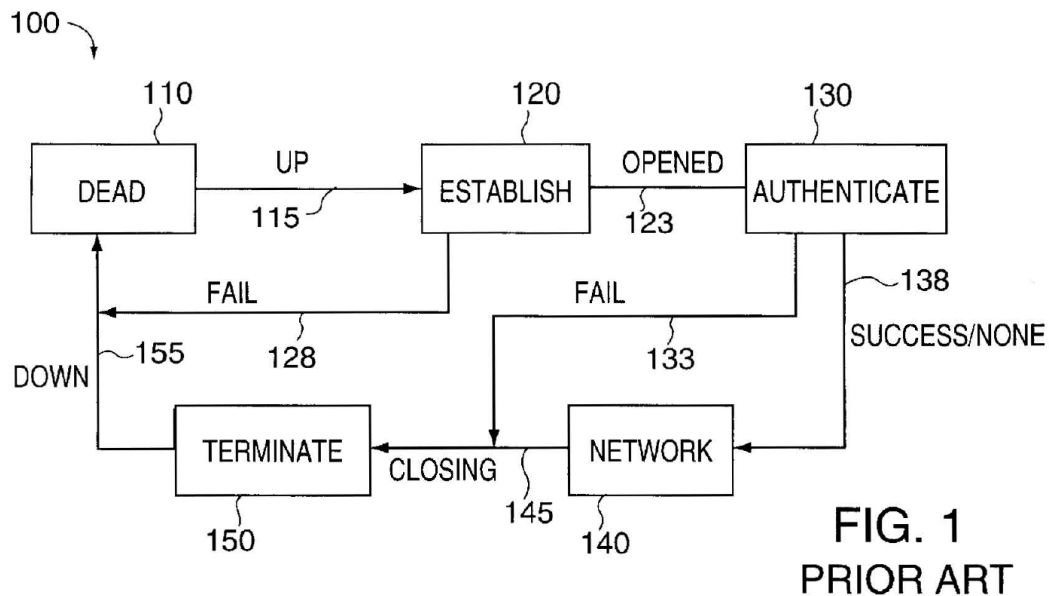


130. For example, Maurya teaches entering the Terminate state when there is no active communication on a link and “either PPP peer desires to terminate the link[.]” Ex-1005, 8:4-19. Therefore, in the combination of Bender and Maurya, the PDSN receives a PPP terminate packet when the mobile station processor decides to terminate a link. Conversely, the PDSN sends a terminate packet and receives a PPP acknowledge packet in return when the PDSN decides to terminate a link. Either way, the PDSN receives a PPP terminate or acknowledge packet. *Id.*

131. The PPP terminate/acknowledge packets are **multipath information** received by the PDSN about which radio elements are to be removed from a multilink bundle. They identify a link (over which the PPP terminate packets were exchanged) and a corresponding device (the device at the end of the link to be terminated) that is to be removed from a multilink bundle because the PPP terminate packets and acknowledgements thereof assure a graceful termination of the link, thereby removing the link, and the corresponding device, from the multilink bundle. Ex-1005, 8:4-19, 16:47-55. Thus, the PPP terminate packets are information received by the PDSN about which radio element is to be removed. Ex-1004, ¶0032; Ex-1005, 8:4-19, 16:47-55.

132. Furthermore, Bender and Maurya **teach the data-source device** (e.g., PDSN) **removing the network-edge devices** (e.g., radio elements) **from being associated with the multipath connection** (e.g., multilink bundle). In accordance with RFC1990/1661, when the connection between the PDSN and the radio element is terminated, the radio element is removed from the multilink bundle because the radio element no longer has an active link to the PDSN. Ex-1004, ¶0032; Ex-1005, 8:4-19. Furthermore, once the link has been terminated, it transitions to a “Dead state” where it waits to “be re-initiated at some future time[.]” Ex-1005, 8:4-19, Fig. 1. Therefore, the radio element is no longer

participating in the multilink bundle. *Id.*¹¹ The link remains in the Dead state “with no activity” until the connection is “appropriately signaled” to transition to an “Establish state[.]” Ex-1005, 7:22-39, Fig. 1:



133. Therefore, the combination of Bender and Maurya **teach the data-source device removing the network-edge devices from being associated with the multipath connection.**

134. Thus, Ground 1 teaches claim 4.

¹¹ The '649 specification confirms that these teachings fall within the scope of the claims, describing an RCG that is removed from “participating” in the bundle. Ex-1001, 12:66-13:14.

6. Claim 5

[5] The method of claim 1, further comprising at least one of the data-source device, the network-edge devices and the data-requesting device determining which of the network-edge devices are to be removed from the multipath connection.

135. Ground 1 teaches the method of claim 1 and the limitations of claim 5. *Supra* §VI.A.1 (explaining the motivation to combine Bender and Maurya, e.g., for link termination; analysis incorporated here).

136. I note that claim 5 is satisfied by any one of the data-source device, one of the network-edge devices, or the data-requesting device performing the step of claim 5. This is because claim 5 requires “at least one of” these devices to perform the recited step.

137. Bender and Maurya teach **at least one of the data-source device** (e.g., PDSN), **the network-edge devices** (e.g., radio elements) **and the data-requesting device** (e.g., processor in mobile station) **determining which of the network-edge devices are to be removed from the multipath connection** (e.g., multilink bundle). For example, Maurya teaches that either the of the PPP peers for a link—which would be the PDSN or radio element in this combination—decide whether to terminate the particular link, thereby removing the corresponding radio element from the multilink bundle, as explained above. *Supra* §VI.A.5 (claim 4; analysis incorporated here); Ex-1005, 8:4-19 (transitioning connection to

Terminate state when “PPP traffic has concluded^[12] and *either PPP peer desires to terminate[.]*”), Fig. 1. Therefore, the combination teaches at least one of the PDSN and radio elements determining which of the network-edge devices are to be removed from the multilink bundle.

138. The PDSN and radio elements also determine which radio elements are to be removed from a multilink bundle when they exchange and receive PPP terminate/acknowledge packets on particular links. Bender and Maurya teach a PDSN and radio elements transitioning their connections to a Terminate state based on an “exchange [of] appropriate PPP terminate packets and, where possible, acknowledgements thereof[.]” Ex-1004, ¶0005, ¶0026, ¶0039, ¶0045; Ex-1005, 8:4-19. When PPP terminate packets are exchanged and acknowledged over a particular link, the PDSN, processor, and corresponding radio element determine that the link should be terminated, causing the particular radio element for that link to be removed from the multilink bundle.¹³ Ex-1005, 8:4-19. For example, when a radio element sends PPP terminate packets over a link, the PDSN determines that the radio element is to be removed from the multilink bundle; the PDSN sends

¹² Maurya teaches mechanisms for determining when traffic has concluded on a particular link, e.g., through ending flags. Ex-1005, 10:30-53, 11:12-28.

¹³ Maurya also teaches determining the link should be terminated “[i]n the event of a link failure” by “simply terminat[ing] after a predefined timeout interval occurs.” Ex-1005, 8:11-14.

appropriate acknowledgements in response, and the radio element determines that it is to be removed from the multilink bundle. *Id.*

139. Thus, Ground 1 teaches claim 5.

7. Claim 6

[6] **The method of claim 1, wherein the data-source device is one of a multipath link aggregator, a network relay, a multipath link server, a server, a network switch, a network router and a device capable of providing multipath services.**

140. Ground 1 teaches **the method of claim 1 wherein the data-source device (e.g., PDSN) is one of a multipath link aggregator, a network relay, a multipath link server, a server, a network switch, a network router and a device capable of providing multipath services.** *Supra* §VI.A.1 (explaining how and why the teachings of Bender and Maurya would have been combined; analysis incorporated here), §VI.A.2[1a] (explaining how and why Maurya’s client-server teachings would have been applied to Bender’s mobile station and PDSN, respectively; analysis incorporated here).

141. As an initial matter, claim 6 recites a list of devices that satisfy the data-source device. I understand claim 6 is satisfied by any of these devices because claim 6 only requires “one of” these devices.

142. Bender teaches the PDSN is **a multipath link aggregator**. The PDSN aggregates multiple links by merging them into multilink bundle. Ex-1004, ¶0008, ¶0022 (“[M]ultiple data links can be set up for aggregation with a Multi-link

Protocol (MP) in PDSN 130.”), ¶0039, Figs. 5, 7. Each link is a distinct path to the PDSN, and therefore the PDSN aggregates multiple links over multiple paths—it is a multipath link aggregator. *Id.*

143. Bender also teaches that the PDSN is **a network relay, a network switch, or a network router**. The PDSN is a “source[]” of data from “an external network such as the Internet.” Ex-1004, ¶0032; *supra* §VI.A.2[1a] (explaining how the PDSN performs a relay, switch, or router function; analysis incorporated here). And Bender’s teachings for the PDSN encompass the PDSN specified in the cdma2000 Wireless IP Network Standard. Ex-1004, ¶0004. The PDSN was known in the art to perform the role of a network relay, a network switch, or a network router, by allowing mobile stations on a CDMA network to access external networks, which included relaying packets between internal and external networks, switching packets that pass through the PDSN, and routing packets to their destination. *See id.*, ¶0032; Ex-1026, Fig. 5 and accompanying text.

144. Bender’s PDSN is also **a multipath link server or a server**. The PDSN sources data from a data application and sends data through a multilink bundle to a mobile station. Ex-1004, ¶0032; Ex-1005, 3:14-23, 3:46-53, 9:56-64. Therefore, PDSN is a multipath link server and a server. Ex-1004, Fig. 5; Ex-1005, 5:29-37, 11:34-39, 14:50-60, Fig. 6A.

145. The PDSN is also a **device capable of providing multipath services**.

The PDSN utilizes a multilink PPP bundle with multiple radio elements connected to a mobile station processor to provide packet data services to mobile stations.

Supra §VI.A.2[1a] (analysis incorporated here); Ex-1004, ¶¶0022, ¶¶0025, ¶¶0027, ¶¶0032-42, Figs. 2, 5. The PDSN provides network access services, delivering data from the Internet, over a multilink bundle to a client. Ex-1004, ¶¶0032.

146. Thus, Ground 1 teaches claim 6.

8. Independent Claim 7

[7pre] A method for establishing and sending data during a multipath connection, comprising:

147. [7pre] is identical to [1pre] except [7pre] includes a comma following “multipath connection”. This comma does not represent a material difference between [7pre] and [1pre]. Therefore, the reasoning I provided explaining how Ground 1 teaches [1pre] applies here to [7pre]. *Supra* §VI.A.2[1pre].

148. The steps of the method also include the teachings described below for [7a]-[7e]. Thus, Ground 1 teaches [7pre].

[7a] the data-source device receiving multipath connection requests from a plurality of data-requesting devices, each of the multipath connection requests involving a particular multipath connection to a data-requesting device;
--

149. Ground 1 teaches **the data-source device (e.g., PDSN) receiving multipath connection requests from a plurality of data-requesting devices**

(e.g., processors), **each of the multipath connection requests** (e.g., requests for multilink PPP connections, such as via LCP Configure Requests) **involving a particular multipath connection** (e.g., multilink bundle) **to a data-requesting device** (e.g., processor).

150. As I explained for limitation [1a], Bender and Maurya teach a **data-source device** and a **data-requesting device**. *Supra* §VI.A.2[1a] (analysis incorporated here).

151. Furthermore, Bender teaches a **plurality of data-requesting devices**, e.g., processors in mobile stations. Ex-1004, ¶0018 (“[S]ystem 100 is shown to include... *three mobile stations* 106.”), ¶0022, ¶0042, Fig. 1:

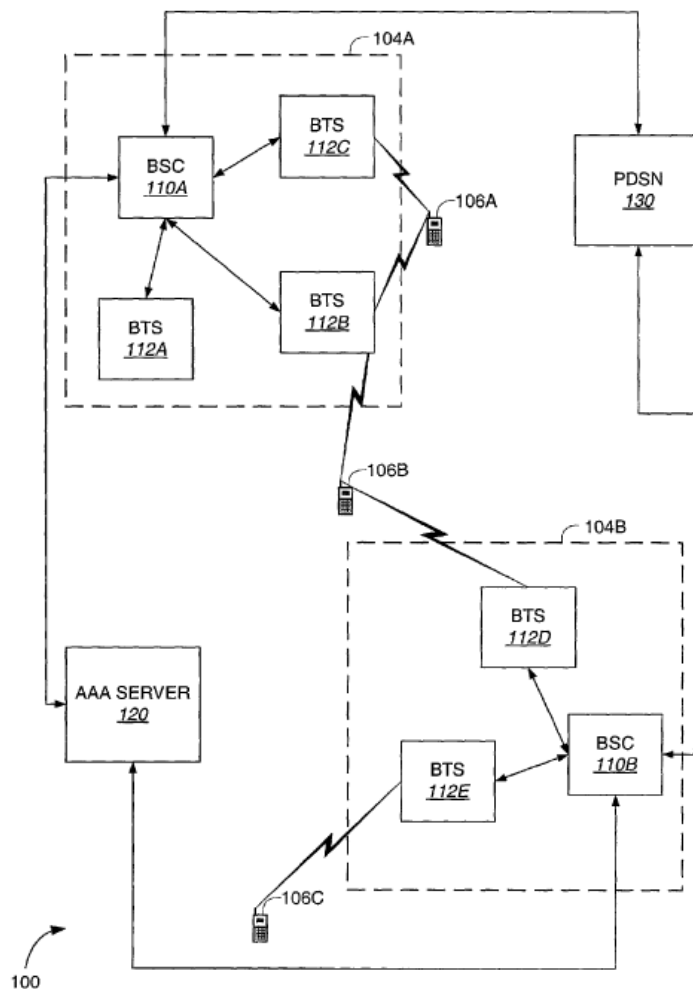


FIG. 1

152. Bender's mobile stations each include a processor. Ex-1004, ¶0025, ¶0027, Fig. 2. Each processor in each mobile station is a computer with memory and a processor for executing instructions, such as for Multilink PPP operations. *Id.* Therefore, Bender teaches a plurality of data-requesting devices.¹⁴

¹⁴ The '649 patent confirms Bender's teachings fall within the scope of a data-requesting device. *Supra* n. 7.

153. Bender and Maurya teach **the data-source device receiving multipath connection requests from a plurality of data-requesting devices**, for example the PDSN receiving requests for multilink PPP connections, e.g., via LCP Configure Requests, from processors in mobile stations. Ex-1004, ¶0026, ¶0027, ¶¶0032-40 (describing “steps to establish a multi-link communication session” using “multi-link PPP” including “sending a Link Control Protocol (LCP) option[.]”), ¶0042 (“The principles discussed herein translate readily to radio elements residing on multiple mobile stations 106.”), ¶0045, Figs. 6, 7; Ex-1005, 7:19-8:53 (describing “conventional states of a link that carries PPP traffic” including an “Establish state” in which “the LCP, executing on each end of the link and through exchanging and acknowledging LCP Configure packets through the link, forms a PPP connection through the link.”), 8:25-43, 8:54-59, Figs. 1-3, 4. *Supra* §VI.A.1 (explaining how and why the teachings of Bender and Maurya would have been combined, e.g., for adding links; analysis incorporated here). The LCP Configure Request is a request for opening a multilink PPP connection. Ex-1005, 1:51-55, 7:34-37, 8:54-59, 8:33-36, 9:14-21, 9:35-49 (“[A]n LCP Configure *Request* packet specifies a multi-link endpoint discriminator option whenever the type field of the LCP Configure Request packet is set to value nineteen[.]”); *see also* Ex-1008, 000016 (“In order to establish communications over a point-to-point

link, each end of the PPP link MUST first send LCP packets to configure and test the data link.”), 000017, 000024-26, 000028, 000032, 000038.

154. The combination of Bender and Maurya teaches that the request comes from the processor in the mobile station. The processor initiates the request for a multilink PPP connection through the radio elements, and it provides the Endpoint Discriminator Option that goes into the LCP Configure Request.

Ex-1004, ¶0026, ¶¶0032-40, Figs. 6, 7. Therefore, the processor in the mobile station originates the request for a multilink PPP connection, which is sent through the radio elements, and the PDSN receives the request via LCP Configure Requests that contain the Endpoint Discriminator Option from the processor. *Id.*; Ex-1005, 7:19-8:53, 8:25-43, 8:54-59, Figs. 1-3, 4; *see also* Ex-1004, ¶0027, ¶0042, ¶0045. The PDSN receives the request and uses the class-address pair in the Endpoint Discriminator Option to identify the links that are part of the multilink bundle and to form the multilink bundle. Ex-1004, ¶0026; Ex-1005, 2:66-3:14, 5:29-58, 13:54-60, 14:14-21, 14:50-60; *supra* §VI.A.1 (explaining the aggregation of links using the Endpoint Discriminator Option; analysis incorporated here).

155. The combination further teaches the processor (executing a software application) initiates the request for a multilink PPP connection through the network edge devices. Maurya teaches, in the context of an Endpoint Discriminator Option, the “client application executing at a PC, assigns itself a unique endpoint

discriminator value and informs all its immediate PPP peers.” Ex-1005, 13:60-14:5, Abstract, 5:29-37. The client application (running on a processor in the client device) determines the endpoint discriminator value that is to be assigned to the multilink bundle and provides it to its terminal adapters, so that all links in the bundle share the same endpoint discriminator value. Thus, the client processor is the ultimate originator of LCP Configure Request. Ex-1005, 9:14-21, 9:35-49. Maurya’s teachings therefore reinforce Bender’s teaching that the mobile station processor originates the request for a Multilink PPP connection. Ex-1004, ¶0027, ¶0048; Ex-1005, 13:60-14:5, Abstract.

156. At the very least, Bender and Maurya suggest, and it would have been obvious, for the LCP Configure Request to come from the mobile station processor. Even if the actual packet is sent by network edge devices, such as terminal adapters or radio elements, Bender teaches that “Processor 230 may perform some or all of the functions of radio elements[.]” Ex-1004, ¶0027. It would have been natural for the processor to originate the LCP Configure Requests because Bender teaches that the processor controls the radio elements in a mobile station. Ex-1004, ¶0027. Furthermore, forming messages, forming packets, configuring links, and other general purpose functions were commonly implemented through software, running on a processor, and since LCP Configure Requests do not happen often—only when starting up a link or changing a

configuration—the ease and flexibility of software is preferable to implementing this functionality in hardware. *See id.*, ¶¶0027, ¶¶0048. Thus, Bender and Maurya teach the processor sends LCP Configure Request packets to the PDSN.

157. The PDSN receives multipath connection requests, e.g., LCP Configure Requests, from a plurality of processors in a plurality of mobile stations, such as in the example system with “three mobile stations 106” and one “PDSN 130” with “[d]ata connections for mobile stations 106... set up... to PDSN 130.” Ex-1004, ¶¶0018, ¶¶0022, ¶¶0032, Figs. 1, 5.¹⁵ At the very least, Bender and Maurya suggest this, and it would have been obvious, because it would have been impractical for every mobile station to have a dedicated PDSN. Rather, it was common for wireless infrastructure to support multiple client devices. Therefore, Bender and Maurya teach the data-source device receiving multipath connection requests from a plurality of data-requesting devices.

158. Bender and Maurya teaches **each of the multipath connection requests involving a particular multipath connection to a data-requesting device**. For example, each request for a multilink PPP connection received by the PDSN involves a particular multipath connection to a particular mobile station

¹⁵ I note that the PDSN 130 referenced here in Bender’s Figure 5 is the same PDSN 130 in Bender’s Figure 1. Ex-1004, ¶¶0018, ¶¶0022, ¶¶0032, Fig. 1, 5. Likewise, the mobile station 106 referenced in Bender’s Figure 5 is one of the mobile stations 106 referenced in Bender’s Figure 1. *Id.*

processor because the request includes an Endpoint Discriminator Option that identifies a particular multilink bundle between the PDSN and a mobile station processor. Ex-1004, ¶¶0039, ¶0045; Ex-1005, 5:7-58, 13:47-14:60, 16:8-55, Figs. 6B, 7. The LCP Configure Requests are used to establish a particular multilink bundle, corresponding to the <class, address> pair specified in the Endpoint Discriminator Option. Ex-1004, ¶¶0026, ¶0032, ¶0039, ¶0045, Fig. 5; Ex-1005, 5:7-58, 13:47-14:60, 16:8-55, Figs. 6B, 7. The bundle is a connection with multiple paths between from the PDSN to a particular mobile station processor, as described above. *Supra* §VI.A.2[1a] (analysis incorporated here).

159. Thus, Ground 1 teaches limitation [7a].

[7b] the data-source device associating a connection ID to each multipath connection request;
--

160. Ground 1 teaches **the data-source device** (e.g., PDSN) **associating a connection ID** (e.g., class, address pair) **to each multipath connection request** (e.g., requests for multilink PPP connections, such as via LCP Configure Requests). *Supra* §VI.A.1 (explaining how and why the teachings of Bender and Maurya would have been combined, e.g., regarding the Endpoint Discriminator Option). The connection ID and multipath connection requests were explained above. *Supra* §VI.A.3 (claim 2) (explaining connection ID); §VI.A.8[7a] (explaining multipath connection requests). That analysis is incorporated here.

161. Bender teaches the PDSN associates a connection ID to each multipath connection request—as explained above, each request includes an Endpoint Discriminator Option, which in turn includes a <class, address> pair. *Supra* §VI.A.3 (analysis incorporated here), §VI.A.8[7a] (analysis incorporated here); Ex-1004, ¶¶0026, ¶¶0032, ¶¶0039, ¶¶0045; Ex-1005, 5:7-58, 13:47-14:60, 16:8-55, Figs. 6B, 7.

162. The PDSN extracts the <class, address> pair and associates it with each request and the link that the request was received on; the PDSN uses the <class, address> pair to merge that link into a multilink bundle with other links sharing the same <class, address> pair. Ex-1004, ¶¶0039 (“The link can also be merged with any existing links 520 already *merged using multi-link protocol 510 in the PDSN*, or access node.”), ¶¶0045; *supra* §VI.A.3. A multilink bundle is uniquely identified by the <class, address> pair in the Endpoint Discriminator Option. Ex-1004, ¶¶0026, ¶¶0039, ¶¶0045; Ex-1005, 5:7-58, 13:47-14:60, 16:8-55, Figs. 6B, 7.

163. Thus, Ground 1 teaches limitation [7b].

[7c] **the data-source device using multipath information about each of a plurality of network-edge devices that is capable of joining in a particular multipath connection to a particular data-requesting device, the particular multipath connection having a particular connection ID;**

164. Ground 1 teaches **the data-source device** (e.g., PDSN) **using multipath information** (e.g., Endpoint Discriminator Option) **about each of a plurality of network-edge devices** (e.g., radio elements) **that is capable of joining in a particular multipath connection** (e.g., multilink bundle) **to a particular data-requesting device** (e.g., processor), **the particular multipath connection having a particular connection ID** (e.g., <class, address> pair).

Supra §VI.A.1 (explaining how and why the teachings of Bender and Maurya would have been combined, e.g., regarding the Endpoint Discriminator Option; analysis incorporated here).

165. The first portion of [7c] (shown right) tracks the language of [1a] (shown left):

1 a data-source device receiving multipath information about each of a plurality of network-edge devices that is capable of joining a multipath connection with a data-requesting device,

1 the data-source device using multipath information about each of a plurality of network-edge devices that is capable of joining in a particular multipath connection to a particular data-requesting device,

Therefore, Ground 1 teaches the overlapping aspects for reasons explained previously. *Supra* §VI.A.2[1a] (analysis incorporated here).

166. Ground 1 further teaches the additional language of [7c], e.g., **using** the multipath information about each radio element. For example, as explained

above, the PDSN uses the Endpoint Discriminator Option to join a radio element to a multilink bundle, e.g., by using the <class, address> pair specified by the Endpoint Discriminator Option to merge the radio element to the multilink bundle identified by the <class, address> pair. *Supra* §VI.A.2[1a] (analysis incorporated here), §VI.A.3 (explaining how the Endpoint Discriminator Option is used to form a multilink bundle; analysis incorporated here). The PDSN repeats this until “each radio element has been processed.” Ex-1004, ¶0035, ¶0034.

167. Bender teaches that each of the plurality of radio elements is capable of joining in a **particular** multipath connection to a **particular** mobile station processor. The PDSN communicates with each mobile station processor using a particular multilink bundle identified by a unique class-address pair. Ex-1004, ¶0032, ¶0040, Figs. 5, 7; *supra* §VI.A.2[1a] (explaining the multilink bundle between the PDSN and a mobile station processor; analysis incorporated here), §VI.A.3 (explaining how the class-address pair is a connection ID; analysis incorporated here), §VI.A.8[7a] (explaining how multiple mobile stations communicate with the PDSN using their own multilink bundles; analysis incorporated here); Ex-1004, ¶0018, ¶0022, Fig. 1. The radio elements for a particular mobile station are capable of joining the particular multipath connection to the processor for that particular mobile station. E.g., Ex-1004, Figs. 2, 5, 7.

168. Ground 1 teaches **the particular multipath connection has a particular connection ID**, e.g., a unique <class, address> pair that identifies the multilink bundle. *Supra* §VI.A.2[1a], §VI.A.3, §VI.A.8[7a] (analysis incorporated here); Ex-1004, ¶¶0026, ¶¶0039, ¶¶0040, ¶¶0045. The <class, address> pair for the multilink bundle is unique. Ex-1005, 5:7-58, 9:38-49, 16:11-17, 16:8-30, 13:47-14:60, 16:8-55, Figs. 6B, 7 Because the unique <class, address> is used to identify a particular multilink bundle and all the links in the bundle have the same <class, address> pair, each particular multipath connection has a particular connection ID.

169. Thus, Ground 1 teaches limitation [7c].

[7d] the data-source device notifying each of the network-edge devices that is participating in the particular multipath connection associated with the particular connection ID; and

170. Ground 1 teaches **the data-source device** (e.g., PDSN) **notifying each of the network-edge devices** (e.g., radio elements) **that is participating in the particular multipath connection** (e.g., multilink bundle) **associated with the particular connection ID** (e.g., <class, address> pair). For example, Bender and Maurya teach the PDSN responding to LCP Configure Requests with an LCP Configure Positive Acknowledgement. Ex-1004, ¶¶0026, ¶¶0039, ¶¶0045; Ex-1005, 8:25-43 (“[A]n LCP Configure Positive Acknowledgment (ACK)... specifies that options... in the Configuration Request were... all accepted[.]”), 8:34-59, 9:14-21; *supra* §VI.A.1 (explaining how and why the teachings of Bender and Maurya

would have been combined, e.g., for the operation of Multilink PPP bundles; analysis incorporated here).

171. The PDSN responds to the LCP Configure Request with either an LCP Configure Positive Acknowledgement (ACK) or an LCP Configure Negative Acknowledgement (NACK). Ex-1005, 8:60-9:20. As explained above, each LCP Configure Request includes an Endpoint Discriminator option, which in turn includes a class-address pair that uniquely identifies a particular multilink bundle. *Supra* §VI.A.2[1a] (explaining how the class-address pair is used for a multilink bundle; analysis incorporated here), §VI.A.3 (explaining how the class-address pair is a connection ID; analysis incorporated here), §VI.A.8[7a] (explaining how multiple mobile stations communicate with the PDSN using their own multilink bundles; analysis incorporated here). The PDSN sends an ACK to a radio element to notify the radio element that the LCP Configure Request is accepted. Ex-1005, 8:25-43; *see also* Ex-1008, §5.2 (“[T]he implementation MUST transmit a Configure-Ack.”). Likewise, the NACK is also a notification because it notifies the radio element the LCP Configure Request was rejected. This is repeated for each radio element to be merged in the particular multilink bundle. Ex-1005, 8:60-9:20. Thus, the PDSN notifies each of the radio elements that is participating in the particular multilink bundle associated with the particular connection ID.

172. Thus, Ground 1 teaches limitation [7d].

[7e] the data-source device sending data associated with the particular connection ID to the network-edge devices participating in the particular multipath connection associated with the particular connection ID.
--

173. As explained above, Ground 1 teaches that **the particular multipath connection is associated with a particular connection ID**. *Supra* §VI.A.8[7a]-[7b] (analysis incorporated here); §VI.A.3 (explaining class-address pair; analysis incorporated here).

174. Bender and Maurya teach **the data-source device** (e.g., PDSN) **sending data associated with the particular connection ID** (e.g., <class, address> pair) **to the network-edge devices** (e.g., radio elements) **participating in the particular multipath connection associated with the particular connection ID** (e.g., multilink bundle). For example, Bender and Maurya teach the PDSN sending data to radio elements participating in a particular multilink bundle (*supra* §VI.A.2[1a], [1d] (analysis incorporated here)); each bundle is associated with a particular connection ID (*supra* §VI.A.8[7c]-[7d] (analysis incorporated here)). The data sent to the radio elements are aggregated by the processor in the mobile station, which had assigned the <class, address> pair to the multilink bundle. Ex-1004, ¶¶0025-0027. Therefore, the PDSN uses the <class, address> pair shared by the radio elements and the multilink bundle to send data to the processor, and the data sent by the PDSN is associated with the <class, address> pair. The data

sent over the multilink bundle is associated with the bundle and the particular <class, address> pair for that bundle.

175. Thus, Ground 1 teaches limitation [7e].

9. Claim 8

[8] The method of claim 7, further comprising: the data-source device sending different groups of data packets to different network-edge devices that are participating in the particular multipath connection associated with the particular connection ID.

176. Ground 1 teaches claim 8. Claim 8 includes some identical language from limitation [1d], as shown below. Ground 1 teaches these shared limitations, as explained above. *Supra* §VI.A.2[1d] (analysis incorporated here).

1 [1d] the data-source device performing at least one of (i) sending different groups of data packets to different network-edge devices that are participating in the multipath connection

1 8. The method of claim 7, further comprising: the data-source device sending different groups of data packets to different network-edge devices that are participating in the particular multipath connection associated with the particular connection ID.

177. Ground 1 further teaches the different language in claim 8. Ground 1 teaches the data-source device sending different groups of data packets to different network-edge devices that are participating in the multipath connection. *Supra* §VI.A.2[1d]. Ground 1 further teaches the data-source device sending data to the network-edge devices participating in the particular multipath connection associated with the particular connection ID. *Supra* §VI.A.8[7e] (analysis incorporated here). Each particular multilink bundle is associated with a particular connection ID. *Supra* §VI.A.8[7c]-[7d], §VI.A.3 (claim 2), §VI.A.2[1a] (analysis incorporated here).

178. Thus, Ground 1 teaches claim 8.

10. Claim 9

[9] **The method of claim 7, further comprising:**

 the data-source device receiving identification of the network-edge devices to be removed from the particular multipath connection; and

 the data-source device removing the identified network-edge devices from the particular multipath connection.

179. Claim 9 is nearly identical to claim 4, with the only differences highlighted below. Ground 1 teaches these shared limitations, for the reasons explained previously. *Supra* §VI.A.5 (claim 4) (analysis incorporated here).

1 4. The method of claim 1, further comprising:
2 the data-source device receiving multipath information about
 which network-edge devices are to be removed from the multipath connection; and
3 the data-source device removing the network-edge devices from being associated with the multipath connection.

1 9. The method of claim 7, further comprising:
2 the data-source device receiving identification of the network-edge devices to be removed from the particular multipath connection; and
3 the data-source device removing the identified network-edge devices from the particular multipath connection.

180. Ground 1 teaches the different language in claim 9. Bender and Maurya teach receiving **identification of the** radio elements to be removed from the **particular** multilink bundle. For example, Maurya teaches that PPP peers exchange PPP terminate packets and acknowledgments over a link before terminating the link and bringing the link to a Terminate state. Ex-1005, 7:29-8:19, Fig. 1:

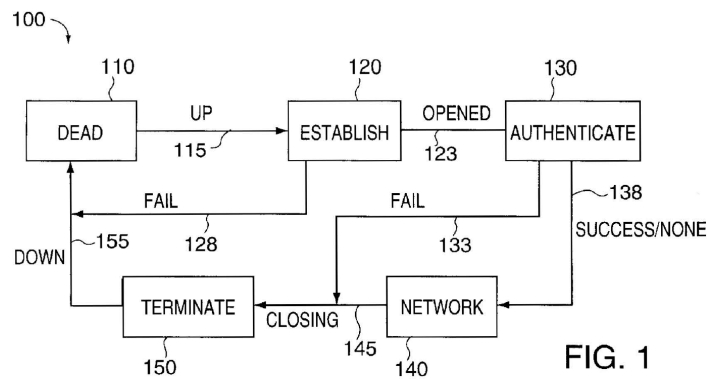


FIG. 1
PRIOR ART

181. The PPP terminate packets exchanged between the PPP peers identify a link (e.g., the link over which the PPP terminate packets were exchanged) and a corresponding device (e.g., the device at the end of the link to be terminated) that are to be removed from a multilink bundle because the PPP terminate packets and acknowledgements thereof are exchanged with the corresponding device over the identified link. Ex-1005, 8:4-19, 16:47-55. Therefore, the PPP terminate packets are identification of radio elements that are to be removed from the particular multilink bundle, in accordance with RFC1990/1661, as taught by Bender and Maurya. Ex-1004, ¶0005, ¶0026, ¶0039, ¶0045; Ex-1005, 7:19-8:20, 8:25-53, 8:54-9:13, 9:14-34, 9:35-10:5, 10:6-11:28, 11:29-33, Figs. 1-5; *supra* §VI.A.1 (explaining how and why the teachings of Bender and Maurya would have been combined, including for link termination; that analysis is incorporated here). The PDSN removes the **identified** radio elements from the **particular** multilink bundle. *Supra* §VI.A.5 (claim 4; analysis incorporated here).

182. Thus, Ground 1 teaches claim 9.

11. Claim 10

[10] The method of claim 7, further comprising the data-source device notifying the data-requesting device when all data has been sent over the particular multilink connection.

183. Ground 1 teaches **the method of claim 7, further comprising the data-source device (e.g., PDSN) notifying the data-requesting device (e.g., processor) when all data has been sent over the particular [multipath] connection (e.g., multilink bundle).** *Supra* §V.D (claim construction).

184. Maurya teaches beginning and ending flags in the headers of segments in a data stream that identify the first segment in the data stream and the last segment in the data stream. Ex-1005, 10:37-53 (“Flag field 530 contains 8-bits, the first two of which... indicate, by a high setting, whether the segment data contained in this packet is a beginning (B) or ending I segment in a packetized data stream, or neither.”), 11:12-28 (“[T]he first two bits of flag field 570... contain values for the beginning and ending flags (B and E)[.]”), 15:7-11, Fig. 5. In combination with Bender, Maurya’s teachings provide details on the structure of data packets sent from the PDSN to the processor, e.g., an ending flag identifying the last segment of a data stream, per RFC1990/1661. Ex-1004, ¶0005, ¶0026, ¶0039, ¶0045; Ex-1005, 7:19-8:20, 8:25-53, 8:54-9:13, 9:14-34, 9:35-10:5, 10:6-11:28, 11:29-33, Figs. 1-5; *supra* §VI.A.1 (explaining how and why the teachings of Bender and Maurya would have been combined, including for multilink PPP

operation; analysis incorporated here). When the PDSN sends a data stream to the processor, the last segment in the data stream includes an ending flag, notifying the processor the last segment in the data stream was received. Ex-1005, 10:37-53, 11:12-28, 15:7-11, Fig. 5; Ex-1004, ¶0025, ¶0027, ¶0040.

185. Additionally, Bender and Maurya teach the PDSN notifying the mobile station processor when all data has been sent over the particular multilink bundle by sending termination packets. Maurya, for example, teaches that “[i]f PPP traffic has concluded and either PPP peer desires to terminate the link, then the link transitions, as symbolized by CLOSING line 145, to Terminate state 150.” Ex-1005, 8:4-19. The termination packets and procedure were explained previously. *Supra* §VI.A.5-6 (claims 4-5; analysis incorporated here).

12. Claim 11

[11] The method of claim 7, wherein the data-source device is one of a multipath link aggregator, a network relay, a multipath link server, a server, a network switch, a network router and a device capable of providing multipath services.
--

186. Dependent claim 11 is substantially the same as dependent claim 6, with the exception that claim 11 is dependent on claim 7 and claim 6 is dependent on claim 1. Therefore, the reasoning I provided explaining how Ground 1 teaches claim 6 applies here to claim 11. *Supra* §VI.A.7 (claim 6).

187. Thus, Ground 1 teaches claim 11.

13. Claim 12

[12] The method of claim 7, wherein the connection ID is generated by one of the data-source device, one of the plurality of network-edge devices, the data-requesting device, and a network device.
--

188. Ground 1 teaches the method of claim 7, with the additional limitation of claim 12. The language recited in claim 12 is also recited in claim 2 and is satisfied by Ground 1 for the same reasons explained previously. *Supra* §VI.A.3 (claim 2).

14. Independent Claim 21

[21pre] A method for creating and using a multipath connection, comprising the steps of:
--

189. Ground 1 teaches **a method for creating and using a multipath connection**. Limitation [21pre] includes some identical language from limitation [1pre], and Ground 1 teaches these shared limitations, as explained above. *Supra* §VI.A.2[1pre] (analysis incorporated here).

190. Bender and Maurya teach **creating a multipath connection**, e.g., by establishing a multilink bundle as explained previously. *Supra* §§V.A.2[1pre]-[1a], §VI.A.3 (claim 2), §§VI.A.8[7a]-[7c] (analysis incorporated here). Bender creates a multilink bundle having multiple links. *See, e.g.*, Ex-1004, Fig. 5:

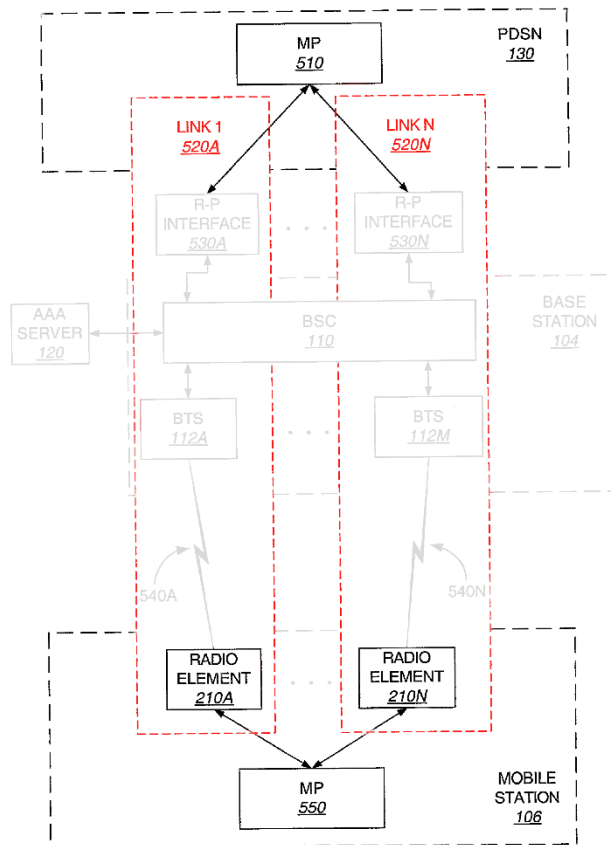


FIG. 5
(Emphasis Added)

191. Bender and Maurya teach **using a multipath connection**, e.g., by sending data over a multilink bundle. *Supra* §VI.A.2[1pre]-[1a], [1d] (analysis incorporated here).

192. The steps of the method also include the teachings described below for [21a]-[21d]. Thus, Ground 1 teaches limitation [21pre].

[21a] a data-requesting device sending first multipath control information to or through a first network-edge device, the first multipath control information indicates that the first network-edge device will join the multipath connection, the data-requesting device being connected over a first connection to the first network-edge device;

193. Ground 1 teaches limitation [21a]. As an initial matter, as I explained for limitation [1a], Bender and Maurya teach a **data-requesting device** and a first **network-edge device**. *Supra* § VI.A.2[1a] (analysis incorporated here).

194. Ground 1 teaches a **data-requesting device** (e.g., processor) **sending first multipath control information** (e.g., Endpoint Discriminator Option) **to or through a first network-edge device** (e.g., radio element). Bender teaches the processor uses Endpoint Discriminator Options “to associate each link with an aggregate bundle, as specified in “The PPP MultiLink Protocol (MP)”[.]” Ex-1004, ¶0026. The Endpoint Discriminator Option specifies a <class, address> pair for a link and indicates the link is to be merged with a multilink bundle that shares the same <class, address> pair. Ex-1004, ¶0026, ¶0039, ¶0045. Bender teaches the processor sends Endpoint Discriminator Options to the radio elements because the processor, through its multilink protocol, uses the Endpoint Discriminator Options to associate the radio elements, and their corresponding links, with a multilink bundle. Ex-1004, ¶0025, ¶0026. At the very least, Bender suggests, and it would have been obvious, that the processor sends Endpoint Discriminator Options to the radio elements because the processor controls the multilink protocol for the mobile

station, and all radio elements for a mobile station must use the same class-address pair in the Endpoint Discriminator Option in order to be joined into the same multilink bundle. Ex-1004, ¶¶0025, ¶¶0026, ¶¶0039, ¶¶0045. Therefore, the processor sends Endpoint Discriminator Options specifying the same class-address pair to the radio elements so that their respective links are merged into a multilink bundle that specifies the same class-address pair.

195. Maurya’s complementary teachings confirm this. Maurya teaches a client “will establish its own endpoint discriminator value” and “supply, via LCP Endpoint Discriminator packets... to each of terminal adapters[.]” Ex-1005, 14:22-37, 16:9-17. Therefore, the combination teaches the processor sending Endpoint Discriminator Options specifying the same class-address pair to the radio elements. Ex-1004, ¶¶0025, ¶¶0026, ¶¶0039, ¶¶0045; Ex-1005, 14:22-37, 16:9-17; *supra* §VI.A.1 (explaining how and why the teachings of Bender and Maurya would have been combined, e.g., regarding the Endpoint Discriminator Option; analysis incorporated here).

196. Ground 1 teaches **the first multipath control information** (e.g., Endpoint Discriminator Option) **indicates that the first network-edge device** (e.g., radio element) **will join the multipath connection** (e.g., multilink bundle). As explained above, the Endpoint Discriminator Option includes a <class, address> pair, and the Endpoint Discriminator Option is sent from a radio element

to the PDSN. This class-address pair is used by Bender and Maurya to join the radio element to the multilink bundle identified by the pair. *Supra* §VI.A.2[1a], §VI.A.3 (claim 2) (analysis incorporated here); Ex-1004, ¶0026, ¶0039 , ¶0045, Fig. 7. Therefore, when the processor sends the Endpoint Discriminator Option to a radio element, the Endpoint Discriminator Option indicates that the radio element will join the multilink bundle.

197. Ground 1 teaches **the data-requesting device** (e.g., processor) **being connected over a first connection to the first network-edge device** (e.g., radio element). As explained above, Ground 1 teaches a plurality of network-edge devices, each being connected to the data-requesting device over a different connection; the connection between the processor and one of the radio elements is a first connection. *Supra* §§VI.A.2[1a], [1c] (analysis incorporated here); Ex-1004, ¶0025, ¶0032 , Fig. 2, Fig. 5:

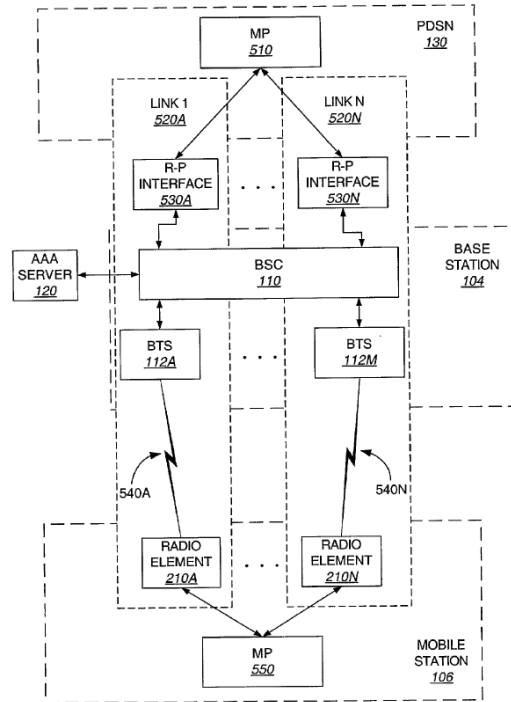


FIG. 5

198. As illustrated in Figure 5, the multilink protocol 550 is connected to radio elements 210A-210N, and each radio element is part of a separate link, links 520A-520N. *Id.* Each radio element has a separate connection to the multilink protocol, which is included in the processor. Ex-1004, ¶0025.

199. Thus, Ground 1 teaches limitation [21a].

[21b] the data-requesting device sending a second multipath control information to or through a second network-edge device, the second multipath control information indicates that the second network-edge device will join the multipath connection, the data-requesting device being connected over a second connection to the second network-edge device, the data-requesting device capable of receiving different data over the first connection and the second connection at the same time; and

200. Ground 1 teaches **the data-requesting device sending a second multipath control information to or through a second network-edge device, the second multipath control information indicates that the second network-edge device will join the multipath connection, the data-requesting device being connected over a second connection to the second network-edge device.**

This portion of [21b] is identical to [21a], but replacing “first” with a “second” network-edge device and “second” connection. As explained above, Ground 1 teaches a plurality of network-edge devices, each being connected to the data-requesting device over a different connection. *Supra* §VI.A.14[21a], §§VI.A.2[1a]-[1c] (analysis incorporated here). Any one of the plurality of network-edge devices and corresponding connection satisfies the “first” network-edge device and connection; any other network-edge device and corresponding connection satisfies the “second” network-edge device and “second” connection. *Id.* Indeed, Bender explains that its teachings are repeated for each radio element in a multilink bundle. Ex-1004, ¶¶0034-35.

201. Ground 1 teaches **the data-requesting device capable of receiving different data over the first connection and the second connection at the same time**. As explained above, data sent over the multilink bundle is divided between the links and sent in parallel, at the same time, to utilize the aggregate bandwidth of the links. *Supra* §VI.A.2[1c]-[1d] (analysis incorporated here), §VI.A.1(explaining how and why the teachings of Bender and Maurya would have been combined, e.g., aggregating bandwidth over different links for increased bandwidth; analysis incorporated here). Therefore, the processor is capable of receiving different data over the first connection and the second connection at the same time—otherwise, link bandwidth would not be aggregated for the bundle. *See id.*; Ex-1004, ¶¶0025-27, ¶0040; Ex-1005, 3:14-53, 6:66-7:6, 9:56-10:5.

202. Thus, Ground 1 teaches limitation [21b].

[21c] after the first network-edge device joins the multipath connection, the data-requesting device receiving first data over the first connection from the first network-edge device; and

203. Ground 1 teaches **after the first network-edge device** (e.g., radio element) **joins the multipath connection** (e.g., multilink bundle), **the data-requesting device** (e.g., processor) **receiving first data over the first connection from the first network-edge device**.

204. For example, Bender teaches that “Once multiple links 520 are formed, and merged, then... [a]t this point, data can flow over the virtual link[.]”

Ex-1004, ¶¶0040, ¶0004. Here, once the multilink bundle between the processor and the PDSN is established, including when the links that form the multilink bundle have joined the multilink bundle, then data flows over the multilink bundle—this includes the PDSN sending data to the processor over the multilink bundle.

Ex-1004, ¶¶0004, ¶0025, ¶0027, ¶0040; *supra* §VI.A.2[1d] (explaining how data is sent from PDSN to the processor over links; analysis incorporated here).

Therefore, the PDSN sends data to the processor over the multilink bundle, and the processor receives the data over the multilink bundle after all the radio elements, including the first radio element, have joined the multilink bundle. *Id.* The data received over the connection from the first radio element is the recited first data. *Id.*

205. The processor receives the data through the multilink bundle from the PDSN with different data sent over each radio element and their respective connections to the processor, on an approximately equal basis. *Supra* §VI.A.2[1c]-[1d] (explaining how different data is sent over each link to each radio element; analysis incorporated here), §VI.A.1 (explaining how and why the teachings of Bender and Maurya would have been combined, e.g., regarding Multilink PPP operations; analysis incorporated here); Ex-1004, ¶¶0025, ¶0027, ¶0040; Ex-1005, 3:14-35, 3:46-53, 9:56-10:5. Therefore, after all the radio elements, including the first radio element, that form the multilink bundle have joined the multilink bundle,

the processor receives data over all the connections from the radio elements, including the first connection from the first radio element.

206. Thus, Ground 1 teaches limitation [21c].

[21d] after the second network-edge device joins the multipath connection, receiving second data at the data-requesting device over the second path from the second network-edge device.

207. Ground 1 teaches limitation [21d], which replaces references to the “first” network-edge device and connection with references to the “second.” As explained above, Ground 1 teaches a plurality of network-edge devices, each being connected to the data-requesting device over a different connection path. *Supra* §VI.A.14[21a]-[21b], §§VI.A.2[1a]-[1c] (analysis incorporated here). Any one of the plurality of network-edge devices and corresponding connection satisfies the “first” network-edge device and connection; any other network-edge device and corresponding connection satisfies the “second” network-edge device and “second” connection, for reasons explained previously. *Id.*

208. Therefore, the above analysis from [21c] also applies to the “second” network-edge device and connection here because Bender’s and Maurya’s teachings apply to both. *Supra* §VI.A.14[21c] (analysis incorporated here). Therefore, Bender and Maurya teach **after the second network-edge device** (e.g., radio element) **joins the multipath connection** (e.g., multilink bundle), **the data-requesting device** (e.g., processor) **receiving second data over the second**

path from the second network-edge device. As I noted above, Bender and Maurya teach that after all the radio elements that form the multilink bundle have joined the multilink bundle, the processor receives data over all the connections from the radio elements. Ex-1004, ¶¶0025, ¶¶0027, ¶¶0040; Ex-1005, 3:14-35, 3:46-53, 9:56-10:5.

209. The data received over the second connection from the second radio element is different from the data received over the first connection from the first radio element. *Supra* §VI.A.14[21b]-[21c] (explaining how the processor is capable of receiving different data over the first connection and the second connection at the same time; analysis incorporated here), §VI.A.2[1c]-[1d] (explaining how different data is sent over each link to each radio element; analysis incorporated here). Thus, Bender and Maurya teach after the second radio element joins the multilink bundle, the processor receives data over the second connection from the second radio element. Thus, Ground 1 teaches limitation [21d].

15. Claim 22

[22] The method as in claim 21, wherein the first network-edge device is connected to a network, and wherein the second network-edge device is connected to the network or a different network.

210. Ground 1 teaches the method as in claim 21, **wherein the first network-edge device (e.g., radio element) is connected to a network, and**

wherein the second network-edge device (e.g., radio element) is connected to the network or a different network.

211. For example, as explained above, Bender teaches that each of the radio elements is connected to one or more networks, e.g., internal cellular networks and external networks such as the Internet. *Supra* §VI.A.2[1b]. Each of the radio elements is connected to a cellular network. Ex-1004, ¶¶0021, ¶¶0025, ¶¶0032, Figs. 1, 5. Furthermore, each radio element is connected to an external network, such as the Internet, through the PDSN. Ex-1004, ¶¶0004, ¶¶0032, Fig. 5.

212. Additionally, Bender teaches each radio element being connected to a different base station (“BTS”), for example as shown in Figure 5:

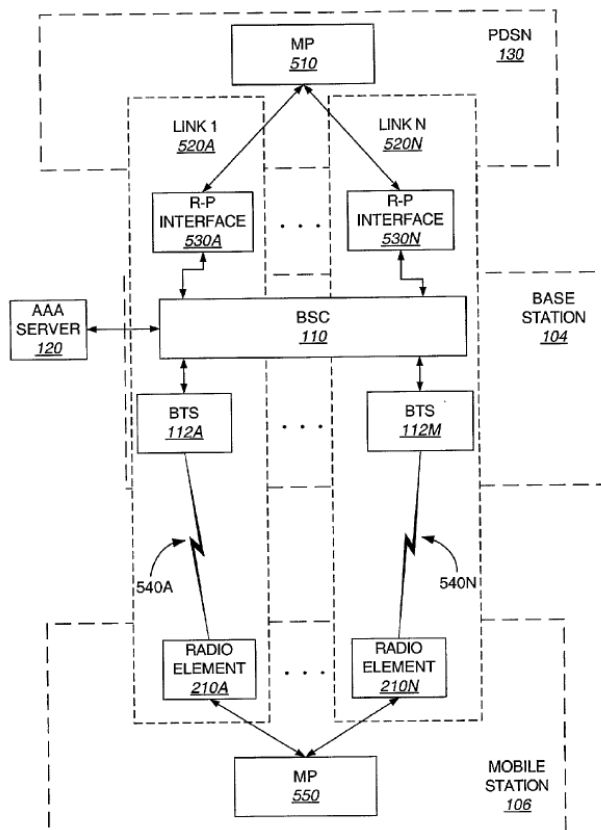


FIG. 5

213. Therefore, Bender teaches or suggests the first radio element is connected to a network, and the second radio element is connected to either the same network or a different network, through different base stations. Thus, Ground 1 teaches claim 22.

B. Grounds 2-3

214. It is my opinion that a POSITA would have found Claims 1-12 and 21-22 obvious based on Grounds 2-3.

1. Motivation to Combine

215. **Mineno and TCP.** Mineno teaches a SHAKE (“SHAed [sic] multiple paths protocol for a cluster type network Environment”) protocol that runs over

TCP. Mineno teaches “SHAKE uses TCP to transmit data.”. Ex-1006, ¶0017.

Mineno Fig. 2 shows the SHAKE protocol running between the application layer and the TCP layer of the network stack. Therefore, Mineno provides express teaching, suggestion, and motivation to combine its teachings with the TCP protocol, and a POSITA would have been motivated to do so.

216. The TCP protocol includes, for example, various teachings for packet communication across networks, including a three way handshake to establish a TCP connection between two endpoints. Ex-1007, 000033-34, 000036-38. In a typical three way handshake, the initiating endpoint A sends a SYN packet to receiving endpoint B, which then responds with a SYN-ACK packet, indicating that it has received the SYN packet and is willing to establish a connection; A then sends an ACK packet to B, indicating that it has received the SYN-ACK packet and is ready for data transmission. Ex-1007, 000037. Data can then be transmitted over the TCP connection, in either direction. Ex-1007, 000016. Since Mineno teaches using TCP to transfer data between hosts (Ex-1006, ¶0017), a POSITA would have been motivated to apply these teachings from the TCP standard to establish and close TCP connections between hosts and to use them to transfer data.

217. This would not have changed the principle of operation of either reference, and a POSITA would have had a reasonable expectation of success in

making the combination, because Mineno teaches the ordinary use of TCP, for sending and receiving data. Indeed, the TCP protocol had been widely used on the Internet since before the 1990s, and by the relevant timeframe, TCP was ubiquitous as a dominant Internet protocol. *Supra* §I. TCP was commonly used with the Internet Protocol, so much so that it was commonly referred to as TCP/IP. Mineno uses TCP/IP. *E.g.*, Ex-1006, Fig. 2. A POSITA would have been more than capable of using TCP/IP in the manner taught by Mineno, which is consistent with the way TCP/IP was used across the Internet. *Supra* §IV.

218. TCP was standardized in RFC793. Therefore, a POSITA would have had a reasonable expectation of success and been motivated to combine Mineno with the teachings of RFC793, to use the TCP protocol to accomplish Mineno's SHAKE teachings. A POSITA with would have been more than capable of using TCP because it would have been a basic skill in computer networking or telecommunications, and a requirement for working with the Internet. *Supra* §IV. This would have simply combined prior art elements (*e.g.*, Mineno's SHAKE and TCP) according to known methods (*e.g.*, as taught by Mineno) to yield predictable results (*e.g.*, the SHAKE protocol taught by Mineno). It would also have applied a known technique (*e.g.*, TCP) to a known method (*e.g.*, SHAKE).

219. **Mineno in view of TCP (RFC793) and MLPPP teachings (RFC1990/1661).** The motivation to combine Mineno and RFC793 was explained

above. A POSITA would also have been motivated to apply teachings from RFC1990/1661, which together describe Multilink PPP (“MLPPP”).¹⁶ In the combination of Ground 3, Mineno’s teachings, along with RFC793, provide the base reference, while teachings from RFC1990/1661 provide additional details and enhancements, as explained below.

220. Mineno provides express teaching, suggestion, and motivation to combine its teachings with Multilink PPP teachings, including as taught in RFC1990/1661, which Mineno refers to “Related Works.” Ex-1006, ¶0006-0008. Mineno explains that it was generally known that resource aggregation “can improve the performance and efficiency in shared system or communication network” where “[o]ne of the resource aggregation methods is that the traffic is transmitted in parallel over multiple paths.” *Id.*, ¶0006. Mineno points to MLPPP as an example of network resource aggregation: “IETF standardizes MP(Multilink Point-to-point Protocol) with the technique which consists of a number of different switched WAN services such as ISDN and ATM to form a virtual path[.]” *Id.*, ¶0008 (citing RFC1990). MLPPP, as taught in RFC1990, “can strengthen the

¹⁶ RFC1661 introduced the Point-to-Point Protocol, known as “PPP.” RFC1990 refers back to RFC1661. Ex-1009, 000009, 000013, 000029. RFC1990 builds on RFC1661, adding multilink features to PPP, hence the name Multilink PPP. Therefore the teachings of RFC1990 are intertwined with RFC1661, and their teachings were meant to be used together, which a POSITA would have recognized based on the shared concepts and references from RFC1990 to RFC1661.

functions of PPP(Point-to-Point Protocol)”—which was standardized in RFC1661. *Id.* Specifically, MLPPP strengthens “data communications between two points, and a number of virtual connections can be set up between equipments [sic] by taking advantages of characters of switched WAN service,” including by dividing user traffic, “and a number of physical lines and virtual lines can be used as a single logical line.” *Id.* ¶0008. After discussing MLPPP, Mineno concludes, “[t]he technology of aggregating multiple communication paths and dispersing traffic into each communication path are being developed to improve the performance and efficiency of transmission.” *Id.*, ¶0009. In summary, Mineno discusses IETF MLPPP standards as “Related Works” with teachings for “aggregating multiple communication paths and dispersing traffic into each[.]” *Id.*

221. Like RFC1990/1661, Mineno also seeks to aggregate multiple communication paths and disperse traffic into each path, in order to increase bandwidth. Ex-1006, ¶0001-4, ¶¶0006-8, ¶¶0011-13, ¶¶0037-47. Both references aggregate multiple physical links together, using them as a single logical link. Ex-1009, 000007 (“The system offering the option is capable of combining multiple physical links into one logical link[.]”); Ex-1006, ¶0012 (“And the latter communication protocol in which other hosts’ communication capacity can be shared is called SHAKE, which we are developing. By using this, the communication paths of other mobile hosts which compose the cluster are shared

and they are considered logically one path.”), *id.*, ¶0029 (“It realizes the shared multilink procedure by using multimedia through multiple paths to form a logical path.”), Fig. 3:

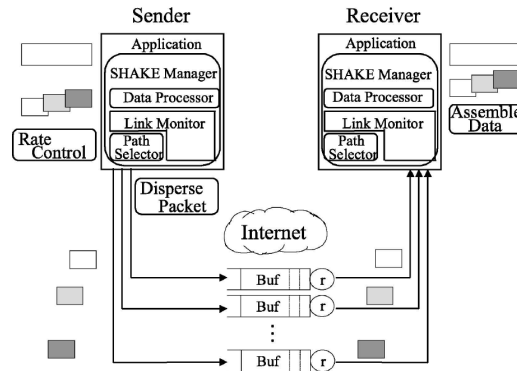


Figure 3: Illustration of SHAKE Modules.

Indeed, Mineno uses “PPP” as part of its network stack (*id.*, Fig. 2), and therefore a POSITA would naturally have looked to the teachings and concepts of RFC1990/1661, which teach the PPP and MLPPP protocols. Therefore, in view of their common goals and use of traffic dispersion, the teachings of RFC1990/1661 would have complimented Mineno’s SHAKE teachings, including for (a) establishing, (b) using, and (c) terminating links and bundles. *See, e.g., id.*, ¶¶0006-9.

222. (a) *Establishing.* RFC1990/1661 teach ways to establish multiple links and then to aggregate them together into a single multilink “bundle.” *E.g.*, Ex-1009, 000007-8. RFC1661 includes teachings for initializing and configuring communication links, e.g., by sending a Configure Request message from one

endpoint to the other. Ex-1008, 000013, 000021, 000028, 000034. The Configure Request includes configuration information, such as an endpoint discriminator that, as its name suggests, allows the recipient to identify the multiple links between the two endpoints and discriminate between those links and links to other endpoints. Ex-1009, 000019-23. For example, RFC1990 teaches that endpoint discriminators can be IP addresses or magic number. *Id.*, 000019-23. This information is then used to bundle multiple links together into a single logical connection between the endpoints. *Id.*, 000007, 000019-20.

223. Applying these teachings from RFC1990/1661 to Mineno, it would have been obvious to likewise use a configure request to initiate a SHAKE connection between two hosts, such as a mobile host and a fixed/server host, with an identifier for the logical connection between the hosts—indeed, Mineno teaches multiple communication paths in the logical connection between Sender and Receiver host. *See* Ex-1006, ¶0012, ¶0016, ¶0029, Figure 3. This would have been advantageous because it provides a simple way for Mineno’s hosts to distinguish the communication paths in their logical connection from other paths to other devices. *See* Ex-1009, 000007, 000019-23 (teaching endpoint discriminator option to aggregate links into a bundle). Indeed, the use of ids was common in computer networking, and even in human society, because of its simplicity for this purpose. For example, TCP used IP addresses as an ID for the connection between two

endpoints, which makes sense because the purpose of IP addresses was to uniquely identify devices on a network. *See* Ex-1007, 000011, 000016-18, 000090. RFC1990/1661 teach the use of magic numbers and IP addresses in the endpoint discriminator option, which would have been a natural fit with Mineno, which likewise used link numbers and IP addresses in its networks stack for communication between hosts. Ex-1006, ¶¶0016, ¶¶0031-32, Fig. 2; Ex-1009, 000019-23. Indeed, Mineno already teaches this because it adds a “Link Number” to the SHAKE header that is used to forward packets. *Infra* §VI.B.3 (claim 2). The combination with RFC1990/1661 renders this further obvious. For example, Mineno teaches SHAKE connections between mobile hosts and Internet hosts, and a POSITA would have been motivated for the data requesting host, typically a PC/laptop requesting a webpage from a server, to send a configure request to the Internet host, in the manner explained above, to initiate a SHAKE transfer of a webpage and related resources, using an IP address as an endpoint discriminator to identify each of the connections as well as their aggregate. *See, e.g.*, Ex-1006, ¶¶0001-2, ¶¶0013-17, ¶¶0031-32, ¶¶0037-47, Figs. 1-3, 6-8, 10. Therefore, the combination was a known and obvious way to address the known issue of distinguishing between connections, including when initiation communications between two endpoints.

224. (b) *Using*. RFC1990/1661 also teach the ways to use the aggregated bundle, e.g., distributing data packets between the multiple links in the bundle. *Infra* §VI.B.2[1d]; e.g., Ex-1009, 000007 (“[O]ne can split packets among parallel virtual circuits between systems[.]”), 000008-9, 000011-19, 000023-24. It would have been obvious to apply these teachings to Mineno by splitting up the stream of data packets onto the different communication paths from Sender to Receiver; indeed, Mineno teaches this. *E.g.*, Ex-1006, ¶0001, ¶¶0003-4, ¶0006, ¶0009, ¶¶0011-15, ¶0029, ¶0048, Fig. 3. This would have been advantageous because it takes advantage of the multiple communication paths and increases bandwidth between the Sender and Receiver. *Id.*, ¶0003, ¶¶0011-12, ¶¶0037-47, Figs. 6-10.

225. (c) *Terminating*. RFC1990/1661 teach ways to terminate links in a bundle, e.g., through the use of Link Termination packets. *E.g.*, Ex-1008, 000032 (“There are three classes of LCP packets: ... Link Termination packets used to terminate a link (Terminate-Request and Terminate-Ack).”), 000012, 000015, 000021, 000025-000026, 000029 (“A Terminate-Request packet is transmitted. This indicates the desire to close a connection.”), 000039-40. RFC1661 teaches the termination of a particular link in a multilink bundle, by transitioning it to a “Link Dead phase” in which a particular link is no longer ready to carry packets. Ex-1008, 000012, 000015, 000022-23.

226. Applying these teachings from RFC1990/1661 to Mineno, it would have been obvious to likewise use a termination procedure to terminate a connection between two hosts using the SHAKE protocol, for example by terminating a connection path that is no longer active so that packets are not sent over a disconnected path. *See, e.g.*, Ex-1006, ¶¶0019-0024, ¶¶0033-35. Mineno provides teaching, suggestion, and motivation to monitor the Internet connections of relay hosts for disconnections and take appropriate action. *Id.*, ¶¶0019-0024, ¶¶0033-35. RFC1990/1661 teach a natural and obvious response: remove the disconnected host from the bundle, by transitioning it to a dead phase so that packets are not sent through that disconnected path. *See, e.g.*, Ex-1008, 000012, 000015, 000022-23. Mineno already teaches this approach, which would have been advantageous because it prevents packet loss from packets being sent to a relay host down a disconnected path to a relay host that has lost its Internet connection. Therefore the combination was a known and obvious way to address the known issue of packet loss and disconnected Internet service.

227. *In summary*, a POSITA would have been motivated to combine teachings from RFC1990/1661 to Mineno, including as outlined in (a), (b), and (c) above, given the complimentary aspects of their teachings and Mineno's express teaching, suggestion, and motivation to combine its teachings with related teachings from RFC1990/1661.

228. Doing so would not have changed the principle of operation for any of the references because the teachings of each would have been applied in a conventional manner, consistent with the teachings of the references. First, RFC1990/1661's teachings for establishing individual connections and aggregating them together would likewise be used in combination with Mineno to establish individual connections between hosts and then aggregating them together. *See* Ex-1006, Fig. 3. The combination uses connection requests and IDs in a manner that was common in networking and on the Internet, and consistent with the teachings and suggestions that Mineno already includes, as explained above. Second, RFC1990/1661's teachings for using aggregated connections would have been used in combination with Mineno to split packets between the multiple connections between two endpoints. Mineno already teaches this with its path selector (*id.* ¶0014, ¶0016-17, ¶0024, ¶0026-28, ¶0033-35.), consistent with the teachings of RFC1990/1661 (Ex-1009, 000007-8, 000011-12). Third, RFC1990/1661's teachings for terminating particular connections would be used in combination with Mineno for the same purpose: to terminate particular connections. Mineno monitors relay hosts for disconnections to their Internet service, and RFC1990/1661 provide a natural response to a disconnection by transitioning to a dead phase.

229. I note that the combination applies *teachings* from RFC1990/1661 with teachings from Mineno. As Mineno explains, RFC1990/1661 are “Related Works” because they teach the concept of “Traffic Dispersion” over multiple aggregated connections. Ex-1006, ¶¶0006-9. Mineno analyzes similarities with MLPPP in terms of the concepts and approaches taught in the references, and therefore a POSITA would have been motivated to apply those teachings in the manner explained above. Ground 3 does not attempt to literally plug MLPPP into an embodiment of Mineno; instead it applies teachings from RFC1990/1661, in the manner explained. As such, purported differences in implementation or other aspects of MLPPP would not be relevant to the combination and would not affect the motivation to make the combination or reasonable expectation of success in doing so.

230. A POSITA would have had a reasonable expectation of success in combining the teachings of Mineno and RFC1990/1661 because, as explained above, their teachings are complementary and they are used in a conventional manner as taught by the references, without changing their principle of operation. Indeed, the combination follows the express teaching, suggestion, and motivation in Mineno itself, which explains how RFC1990/1661 are “Related Works.” Ex-1006, ¶¶0006-9. A POSITA would have been more than capable of using known networking standards such as RFC1990/1661, and been more than capable

of applying teachings from those standards. *Supra* §IV. Ground 3 combines teachings from RFC1990/1661 in natural and obvious ways with Mineno, as explained above, consistent with basic concepts in networking such as using requests, IDs, IP addresses, and terminating downed connections. This would have been well within the skill of a POSITA—based on my personal experience, even students in the field would have been more than capable of doing so.

231. Furthermore, a POSITA would have been motivated and had a reasonable expectation of success in making the combination of Ground 3 because it combines prior art elements (e.g., from RFC1990/1661, as explained above) according to their known methods to yield predictable results, consistent with basic networking concepts. Ground 3 is also the use of known techniques (e.g., from RFC1990/1661, as explained above) to improve similar methods (e.g., as taught in Mineno) in the same way, and it applies known techniques to a known method, which was ready for improvement to yield predictable results. This was evident from Mineno's teaching of RFC1990/1661 as related works in traffic dispersion over multiple connections. Ex-1006, ¶0006-0009.

2. Independent Claim 1

[1pre] A method for establishing and sending data during a multipath connection comprising:
--

232. Grounds 2-3 teach **a method for establishing and sending data during a multipath connection**. For example, Mineno teaches a method for

establishing a multipath connection, e.g., between a Sender host and a Receiver/destination host (e.g., between a mobile host and fixed host), and sending data using multiple paths between them. Mineno establishes a multipath connection over TCP, with multiple paths for data transmission between a sender and receiver, e.g., between a mobile host and an Internet host server. *E.g.*, Ex-1006, ¶¶0001, ¶¶0003 (“In this paper, we propose a protocol SHAKE (SHA[r]ed *multiple paths protocol* for a cluster type network Environment), *for sharing multiple paths in a cluster type network*. Cluster type network is a kind of LAN in which some mobile hosts temporarily connect mutually....”), ¶¶0017-18 (“As a protocol of Transport Layer, SHAKE uses TCP to transmit data . TCP connection is formed between mobile hosts in the cluster and fixed hosts in the Internet.”), Figs. 2-3:

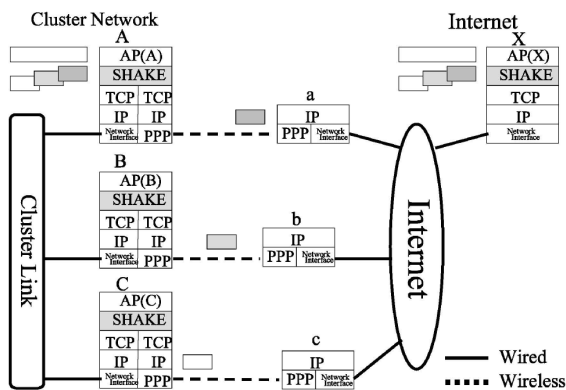


Figure 2: SHAKE Protocol Stack.

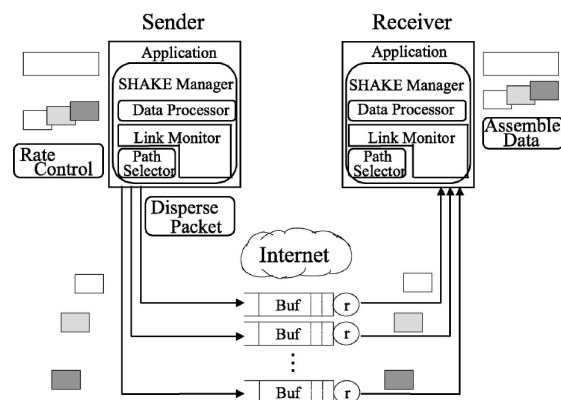


Figure 3: Illustration of SHAKE Modules.

233. Mineno teaches **a multipath connection**. Mineno describes a scenario where mobile hosts are locally networked into a cluster, for example using ad-hoc

Wi-Fi, and they access the Internet using relatively slow cell phone connections.

E.g., Ex-1006, ¶0001-4:

We can access Internet by carrying a portable computer and using the wireless communication. ... In the present status, the wireless network with PHS (Personal Handy phone System), portable cellular telephone or in-house wireless LAN has only rates of tens of Kbps to a few Mbps. ... We can select several kinds of paths to communicate with outside. So the communication media such as PHS or portable cellular telephone can be freely selected by each mobile host.

Id., ¶0016, ¶0042, ¶0047, Fig. 1, Fig. 10. Mineno allows the mobile hosts, connected on a local network such as an ad-hoc Wi-Fi network, to achieve higher bandwidth by aggregating their cellular connections, where each laptop can use the cellular connections of the other laptops to send and receive data. *Id.*, ¶0001, ¶0003, ¶¶0010-18, ¶¶0024-29, ¶¶0031-42, ¶¶0048-49. To facilitate this multipath connection, a “SHAKE” protocol is implemented on mobile hosts and fixed/Internet hosts. *Id.*, ¶0001, ¶0003, ¶¶0013-18, ¶¶0031-32, ¶¶0037-42, ¶¶0048 (“We proposed a protocol, SHAKE, for sharing multiple paths in a cluster type network in which a number of mobile hosts temporarily connect mutually. Each host in the cluster can share other hosts’ communication links.”).

234. The SHAKE protocol distributes data packets between the available links, where packets destined for one mobile host can be sent to the other hosts, which then further route the packets to the destination host. For example, as shown in Fig. 2 of Mineno, when Server X sends data packets to mobile host A using the SHAKE protocol, packets are sent to mobile hosts A, B, and C over their respective Internet (e.g., cellular) connections; the SHAKE protocol then forwards packets from hosts B and C to the destination, mobile host A. *Id.*, ¶¶0001-4, ¶¶0010-26, ¶¶0027-28 (“When each host receives a SHAKE packet ... it judges where to relay the packet. Each host has a table which relates host with link number. So if the packet is destined the host itself, the packet is given to the application and the others are relayed to the next host. Here we explain how to put in order the received data packets with irregular sequence since they were transmitted from multiple paths ...”), ¶0029, ¶¶0031-33, ¶0034 (“The path selecting module searches if the destination host is in the information table when the application requests for sending data. And then it checks the path, and simply sends through these paths in order.”), ¶¶0035-48, Figs. 2-3, 6-8, 10. The same principle applies with data flowing in the inverse direction; when mobile host sends data packets to fixed host X, packets are also routed through mobile hosts A and C, in the same manner explained above. *Id.*, ¶¶0001-4, ¶¶0010-29, ¶¶0031-42, ¶0048, Figs. 2-3, 6-8, 10.

235. Therefore, Mineno teaches a multipath connection between a sending host (e.g., an Internet server) and a receiving host (e.g., a mobile host), with multiple communication paths where the other mobile hosts in the cluster participate in the data transmission. Ex-1006, Figs. 13. The multiple paths are highlighted below, where the blue arrow and two green arrows highlight the multiple communication paths in the multipath connection between the sending Internet host X and the receiving mobile host A:

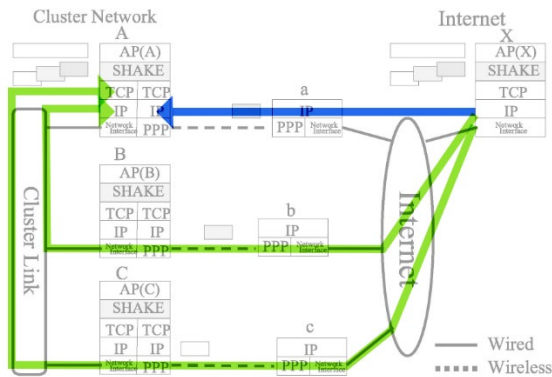


Figure 2: SHAKE Protocol Stack.
(Annotations/Emphasis Added)

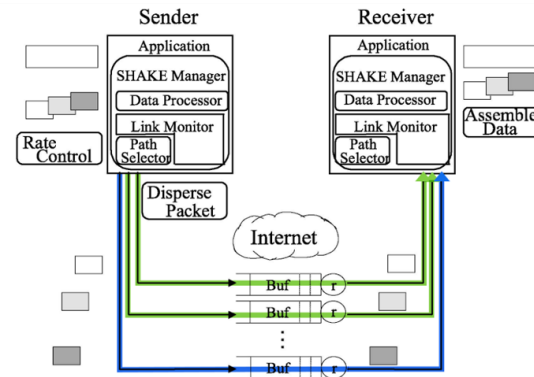


Figure 3: Illustration of SHAKE Modules.
(Annotations/Emphasis Added)

236. Mineno teaches steps for establishing and sending data during a multipath connection, for example as shown for the multipath connection between Sender and Receiver illustrated in Figures 2 and 3. E.g., Ex-1006, ¶¶0001-4, ¶¶0010-16, ¶¶0017-18, ¶¶0024-29, ¶¶0033-36, ¶¶0048, Figs. 1-3. Additionally, Mineno establishes TCP connections between the mobile hosts and an Internet host, e.g., Sender and Receiver hosts, and uses them to send data over multiple

paths between a Sender and Receiver. Ex-1006, ¶¶0001-4, ¶¶0010-16, ¶0017-18, ¶¶0024-29, ¶¶0033-36, ¶0048, Fig. 2 (highlighting added for the example where fixed host X sends data using SHAKE to mobile host A):

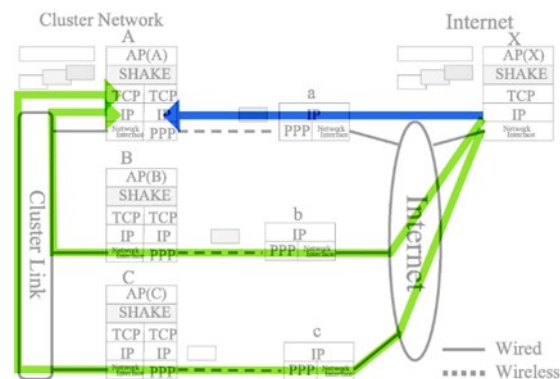


Figure 2: SHAKE Protocol Stack.
(Annotations/Emphasis Added)

Id., Figs. 2-3, 8-10; *see also infra* §VI.B.2[1a] (explaining how Mineno teaches a mobile host, as a data-requesting device, requests data from a server; that analysis is incorporated here).

237. TCP teaches additional details on establishing connections between hosts. *Supra* §VI.B.1 (explaining how and why TCP’s teachings would have been combined with Mineno; that analysis is incorporated here). *E.g.*, Ex-1007, 00005 (“When two processes wish to communicate, their TCP’s must first establish a connection (initialize the status information on each side).”), §2.7 (“The procedures to establish connections utilize the synchronize (SYN) control flag and involves an exchange of three messages. This exchange has been termed a three-

way hand shake”), §3.4 (“The “three-way handshake” is the procedure used to establish a connection.”), Figs. 7-8.

238. TCP also teaches details for sending data using the TCP protocol, including bidirectionally. *E.g.*, Ex-1007, §1.5 (“Basic Data Transfer ...”), §2.7, §2.8, §3.7, 000013 (“Hosts are computers attached to a network, and from the communication network’s point of view, are the sources and destinations of packets”), 000016 (“A connection can be used to carry data in both directions, that is, it is ‘full duplex’.”).

239. Ground 3 provides additional teachings of a multipath connection. *Supra* §VI.B.1 (explaining how and why RFC1990/1661’s teachings would have been combined with Mineno and TCP; that analysis is incorporated here). For example, RFC1990/1661 teach additional details for establishing a multipath connection. *E.g.*, RFC1990 teaches the concept of using multiple physical links as one logical link. Ex-1009, 000005-7 (“Multilink is based on an LCP option negotiation that permits a system to indicate to its peer that it is capable of combining multiple physical links into a “bundle”.... The system offering the option is capable of combining multiple physical links into one logical link[.]”), 000008-9 (“The bundled links can be different physical links, as in multiple async lines, but may also be instances of multiplexed links, such as ISDN, X.25 or Frame Relay. The links may also be of different kinds, such as pairing dialup async links

with leased synchronous links.”), 000009-12 (“[L]arge packets are broken up into multiple segments sized appropriately for the multiple physical links.”), 000015-17, 000019-20, 000023-26; Ex-1008, 000005 (“The Point-to-Point Protocol (PPP) provides a standard method for transporting multi-protocol datagrams over point-to-point links.”), 000007-8.

240. RFC1661 further teaches how links are established. Ex-1008, 000013 (“The Link Control Protocol (LCP) is used to establish the connection through an exchange of Configure packets”), 000021 (“In the Request-Sent state an attempt is made to configure the connection. A Configure-Request has been sent ...”), 000028 (“Send-Configure-Request (scr) A Configure-Request packet is transmitted. This indicates the desire to open a connection with a specified set of Configuration Options.”), 000030. RFC1990 teaches how multiple physical links are aggregated into a logical bundle. *E.g.*, Ex-1009, 000005-7 (“Multilink is based on an LCP option negotiation that permits a system to indicate to its peer that it is capable of combining multiple physical links into a “bundle”.... The system offering the option is capable of combining multiple physical links into one logical link[.]”), 000008-9 (“The bundled links can be different physical links, as in multiple async lines, but may also be instances of multiplexed links, such as ISDN, X.25 or Frame Relay. The links may also be of different kinds, such as pairing dialup async links with leased synchronous links.”), 000018-23.

241. RFC1990/1661 teach sending data over the multiple links in a logical bundle. *E.g.*, Ex-1009, 000005, 000007 (“[O]ne can split packets among parallel virtual circuits between systems[.] ... The method discussed here is similar to the multilink protocol described in ISO 7776 [4], but offers the additional ability to split and recombine packets ...”), 000008-9 (“We suggest that multilink operation can be modeled as a virtual PPP link-layer entity wherein packets received over different physical link-layer entities are identified as belonging to a separate PPP network protocol (the Multilink Protocol, or MP) and recombined and sequenced according to information present in a multilink fragmentation header.”), 000011-12 (“Network Protocol packets are first encapsulated (but not framed) according to normal PPP procedures, and large packets are broken up into multiple segments sized appropriately for the multiple physical links.”), 000015-16 (“The sequence number is not reset upon each new PPP packet”), 000017 (“[T]he transmitter would first queue up as many consecutively numbered packets on one link as it could[.]”), 000018-19 (“[T]he system will construe all packets received on this link as being able to be processed by a common protocol machine with any other packets received from the same peer on any other link”), 000023-24 (“[A] system MUST transmit a Multilink-encapsulated packet on X before transmitting any Multilink- encapsulated packets on Y.”) ; Ex-1008, 000005 (“The Point-to-

Point Protocol (PPP) provides a standard method for transporting multi-protocol datagrams over point-to-point links.”), 000007-10.

242. Therefore, Grounds 2-3 teach [1pre]. Ground 2’s and Ground 3’s method for establishing and sending data during a multipath connection further comprises the steps described below. *Infra* §VI.B.2[1a]-[1d].

[1a] a data-source device receiving multipath information about each of a plurality of network-edge devices that is capable of joining a multipath connection with a data-requesting device,
--

243. Grounds 2-3 teach **a data-source device** (e.g., a Sender host being a fixed host, or alternatively, a mobile host) **receiving multipath information about each of a plurality of network-edge devices** (e.g., other mobile hosts) **that is capable of joining a multipath connection with a data-requesting device** (e.g., a Receiver/destination host, such as a mobile host requesting data from the fixed host or alternatively a fixed host requesting data from a mobile host). Note that the combination of Mineno and TCP teaches a symmetrical, bidirectional protocol, meaning a device running the SHAKE protocol can either send or receive data using the protocol. Ex-1006, ¶¶0001-4, ¶¶0010-29, ¶¶0031-42, ¶0048, Figs. 2-3, 6-8, 10; Ex-1007, 000016 (“A connection can be used to carry data in both directions, that is, it is ‘full duplex’.”).

244. Mineno teaches SHAKE being implemented on both the mobile hosts and the fixed host. *Id.*, ¶0015 (“SHAKE is implemented at all mobile hosts which

form a cluster (cluster means cluster type network) and at fixed hosts which are connected to Internet and communicate with one of mobile hosts in a cluster.”), ¶0017, ¶0032. The SHAKE implemented mobile hosts and fixed host use TCP to transfer data through a multipath connection. *Id.*, ¶0001, ¶¶0003-4, ¶¶0010-18, ¶¶0026-29, ¶¶0032-36, ¶0048, Figs. 1-3. For example, in Fig. 3, Mineno teaches multipath data transfer using SHAKE between a “Sender” and a “Receiver,” which can be any of the hosts so long as SHAKE is installed on both. *Id.*, ¶0015, ¶¶0017-18, ¶0032, ¶¶0037-42, Figs. 2-3. Taken together, Figures 2 and 3 teach an example of the Internet server X as the Sender host and a mobile host as the receiver, with three packets (colored white, light grey, and grey) transmitted from the Sender to the Receiver. Figure 3 shows that the white packet has travelled the furthest from the Sender X, and Figure 2 likewise shows the white packet having travelled furthest, meaning the Sender is the Internet host X. Ex-1006, ¶¶0001-4, ¶¶0014-17, ¶¶0031-32, Figs. 2-3:

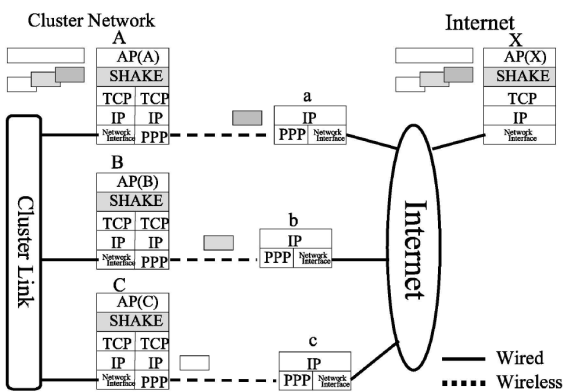


Figure 2: SHAKE Protocol Stack.

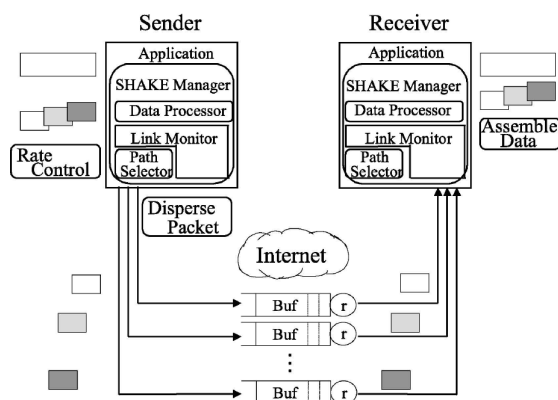


Figure 3: Illustration of SHAKE Modules.

Of course, Mineno teaches that SHAKE is bidirectional and includes further examples of tests that were run to measure bandwidth, where those exemplary test setups included data going from the mobile host, as the Sender, to the fixed host over the Internet. *Id.*, ¶¶0037-47, Figs. 6-10.

245. Therefore, while Mineno satisfies limitation [1a] with data being sent from a fixed host to a mobile host, as explained above, Mineno also satisfies limitation [1a] when data is sent in the inverse direction, with a mobile host sending data to a fixed host using SHAKE. In that situation, Ground 2 teaches **a data-source device** (e.g., the mobile host acting as the Sending host) **receiving multipath information about each of a plurality of network-edge devices** (e.g., other mobile hosts) **that is capable of joining a multipath connection with a data-requesting device** (e.g., a Receiver/destination host).

246. Mineno teaches **a data-source device**. For example, Mineno teaches a “fixed host,” also referred to as a “server host” and “Internet” host. Ex-1006, ¶¶0015, ¶¶0017, ¶¶0037-42, Figs. 1-3, 8-10. Mineno teaches the fixed host as a server that is connected to the mobile hosts over the Internet, and as explained earlier in this section, Mineno Figures 2 and 3 teach the Internet host sending data to a Receiver mobile host. *Id.*, ¶¶0001-2, ¶¶0013-18, ¶¶0031-32. ¶¶0037-42, Figs. 1-2:

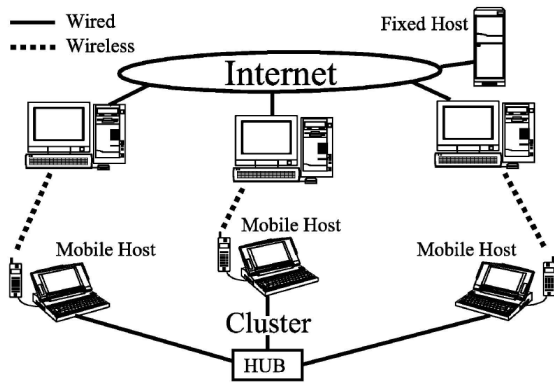


Figure 1: An Example of the System.

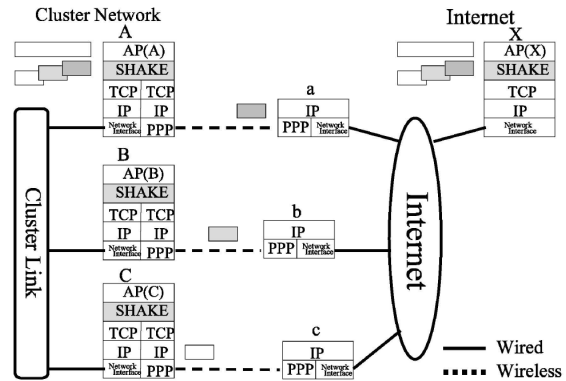


Figure 2: SHAKE Protocol Stack.

247. As Mineno explains, portable computers, such as laptops, were used to access the Internet, including to transfer files and multimedia data. Ex-1006, ¶¶0001-2, ¶¶0010-11, ¶0014. Mineno’s stated goal was to increase capacity for data transfers between a mobile host and Internet hosts. *Id.*, ¶¶0001-4, ¶0006, ¶¶0010-16, ¶0029, ¶0044, ¶0048. By the relevant timeframe, computer networks and the Internet had become ubiquitous, and it was common for client computers, e.g., laptops, to request and receive data from Internet servers like Mineno’s fixed host. *E.g., id.*, ¶¶0001-2; *supra* §I. In short, Mineno teaches that the fixed host sends data to the mobile hosts, and therefore, the fixed host is a data-source device.

248. Alternatively, Mineno also teaches data transmission in the inverse direction, where the mobile host is a data-source device sending data to the fixed host. For example, Mineno teaches the mobile hosts sending data packets to a receiver host. *E.g.*, Ex-1006, ¶¶0017-25, ¶0031, ¶¶0037-40, ¶0041 (“The used file

is about 4Mbytes (4139kbytes) AVI file. It was sent from the client host in the cluster to server host in fixed network.”), ¶¶0042-48, Figs. 3, 6-8, 10:

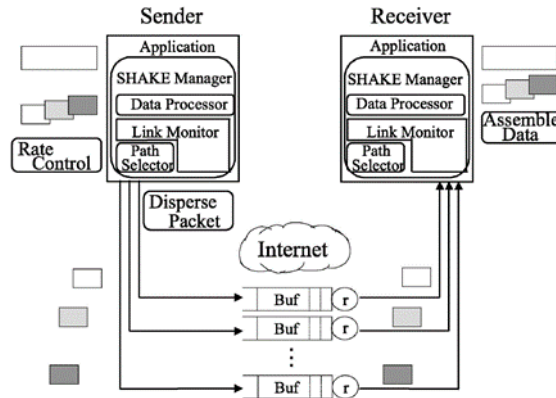


Figure 3: Illustration of SHAKE Modules.

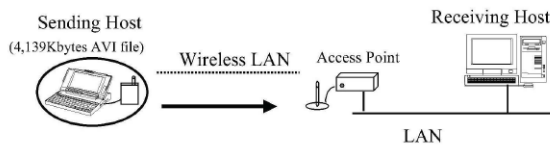


Figure 6: Experiment 1.

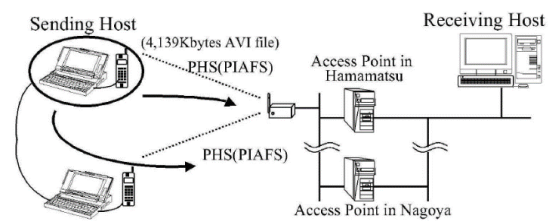


Figure 10: Experiment with PHS terminals.

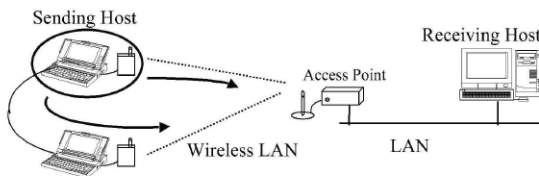


Figure 7: Experiment 2.

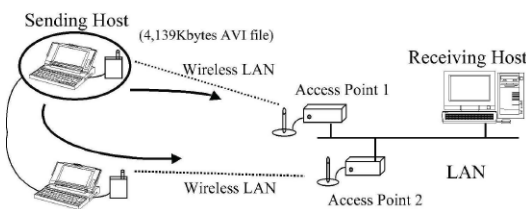


Figure 8: Experiment 3.

249. Additionally, Mineno teaches that the fixed host is a multipath link aggregator, a multipath link server, a server, and a device capable of providing

multipath services, as explained below. *Infra* §VI.B.7 (claim 6). The fixed host is therefore a **data-source device**.¹⁷

250. TCP provides additional teachings regarding different devices such as host computers. *E.g.*, Ex-1007, 000007 (“The Transmission Control Protocol (TCP) is intended for use as a highly reliable host-to-host protocol between hosts in packet-switched computer communication networks, and in interconnected systems of such networks.”), 000013 (“Hosts are computers attached to a network, and from the communication network’s point of view, are the sources and destinations of packets.”), 000086 (“host[:] A computer. In particular a source or destination of messages from the point of view of the communication network.”), 000014-15 , 000008 (“Much of this document is written in the context of TCP implementations which are co-resident with higher level protocols in the host computer. Some computer systems will be connected to networks via front-end computers which house the TCP and internet protocol layers, as well as network specific software.”).

251. Mineno teaches **the data-source device receiving multipath information** (e.g., from report and control packets, and stored in the path

¹⁷ The ’649 patent confirms that a Sender host, such as a fixed host, falls within the scope of a data-source device because claims 6 and 11 recite that the data-requesting device is “one of a multipath link aggregator, a network relay, a multipath link server, a server, a network switch, a network router and a device capable of providing multipath services.” Ex-1001, claims 6, 11.

information table). As an initial matter, the mobile hosts and fixed hosts all implement the SHAKE protocol and modules, including a link monitoring module. Ex-1006, ¶0015 (“SHAKE is implemented at all mobile hosts ... and at fixed hosts which are connected to Internet ...”), *id.*, ¶0017-18 (“The protocol SHAKE is realized from the following 4 modules. Link monitoring module ...”), *id.*, ¶¶0019-24, ¶¶0026-28, ¶0032-36, ¶0048, Fig. 2. The link monitoring module in the mobile hosts and fixed hosts receives report and control packets and uses them to form a “path information table” about the hosts and the multiple paths between the mobile hosts and fixed hosts. *Id.*, ¶¶0017-24, ¶¶0026-28 (“a report packet which includes: packet identifier (Report), link number, packet length, ST1, RT1, RT2, and receiving rate of the link. ... [T]here is a control packet which is only used to form a cluster and make a link table.”), *id.*, ¶¶0034-36, ¶0048, ¶0033:

The present link monitoring module forms a path information table of all end-to-end path information when the cluster is formed, and the information is updated by receiving the report packets. When some new hosts are added to the cluster, new sections are added to this table after new connecting. It performs a role as a routing table indicated among link number and relay host and destination host, and it contains various information like theoretical bandwidth, round-trip delay, and receiving rate. These values are estimated by receiving report packet from the receiving host at regular

intervals, and each link's suitable sending rate is also estimated by this report.

252. The path information table contains “end-to-end path information” and performs a role as a routing table” for sending data packets simultaneously over multiple available connection paths from sender to receiver, through the other mobile hosts. Ex-1006, ¶¶0033-36, ¶0024 (“A sending host gets information about communication paths with outside from the link monitoring module, and then distributes the data packets which come from the application layer.”), Figs. 1-3. It further includes information, received from the report and control packets, such as link number, relay host and destination host, theoretical bandwidth, round-trip delay, and receiving rate. *Id.*, ¶¶0026-28, ¶0033, Figs. 4-5; *supra* §V.B (explaining claim construction for “multipath”). Therefore, Mineno teaches the data-source device—e.g., either a mobile or fixed host—receiving multipath information from report and control packets, which is stored in the path information table; this information includes “end-to-end path information” for each path, e.g., routing information for the full path from sender to receiver. *Id.*, ¶0033-36, ¶0024.

253. Report packets contain various path information about the mobile hosts, including for example link number, ST1, RT1, RT2, and Receiving Rate. Ex-1006, ¶¶0026-27, ¶¶0033-36, Fig. 5 (*see also* Ex-1025):

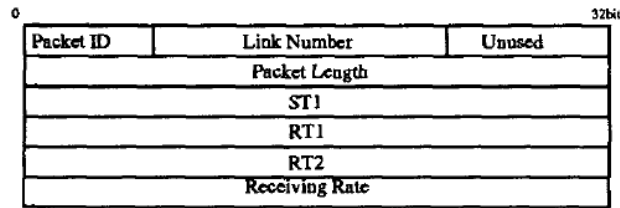


Figure 5: Report Packet Format.

254. The link number field contains information about multiple paths; for example, “[e]ach host has a table which relates host with link number” and when a host receives a packet, it “judges where to relay the packet” “according to the link number in the header[.]” Ex-1006, ¶0027. Therefore, the link number field contains multipath information about each host that is capable of joining a multipath connection with the data-requesting device.

255. The ST1, RT1, and RT2 fields also contain information about the multiple paths. For each of the dispersed packets in a multipath connection, SHAKE “add[s] a sending time stamp ST1 in SHAKE header and sent to the receiving host”; the receiving host then “sends back a receiving report” “which contains the newest received packet’s ST1, RT1 which is the time it received, RT2 which is the time this report was sent[.]” Ex-1006, ¶0021. ST1, RT1, and RT2 are used to estimate the Round Trip Time (RTT)—a value that relates to the status of a communication path in a multipath connection corresponding to a relay host. *Id.*, ¶0019 (“Here the status of communication paths means the theoretical bandwidth of used medium, throughput which changes actively, transit delay, cut of

communication paths, etc.”), *id.*, ¶¶0020-23. Mineno further teaches the link monitoring module of SHAKE uses such information to monitor the status of the multiple communication paths and “change the sending rate to use each path’s bandwidth effectively.” *Id.*, ¶¶0019-20 (“This module monitors the status of communication paths with outside networks and estimate the sending rate. ... [E]ach communication path has a buffer and all buffers are managed by Link Monitoring module.”). Therefore, ST1, RT1 and RT2 is multipath information because it is used to monitor the status of the paths and hosts in a multipath connection.

256. Additionally, the report packet includes multipath information because it is “used to report communication status” of a multipath cluster. Ex- 1006, ¶¶0017, ¶¶0026. Mineno teaches that the link monitoring module of SHAKE forms “path information table of all end-to-end path information when the cluster is formed, and the information is up dated by receiving the report packets.” *Id.*, ¶¶0033. Mineno further teaches that new section will be added to the path information table when new hosts are added to the cluster, and the path information table “performs a role as a routing table indicated among link number and relay host and destination host, and it contains various information like theoretical bandwidth, round-trip delay, and receiving rate.” *Id.*, ¶¶0033. Values in the path information table are “estimated by receiving report packet from the

receiving host at regular intervals, and each link's suitable sending rate is also estimated by this report." *Id.*, ¶0033. Therefore, the report packet contains multipath information regarding mobile hosts in the multipath cluster, is used to generate and update a routing table that contains "end-to-end path information" of a multipath cluster, and includes various information such as bandwidth and delay. The report packet includes multipath information.

257. RFC1990/1661 include additional teachings for a multipath connection, as explained for limitation [1pre], and teachings for providing multipath information to a server. *E.g.*, Ex-1009, 000008 ("The goal of multilink operation is to coordinate multiple independent links between a fixed pair of systems, providing a virtual link with greater bandwidth than any of the constituent members. The aggregate link, or bundle, *is named by the pair of identifiers for two systems connected by the multiple links*. A system identifier may include information provided by PPP Authentication[] and information provided by LCP negotiation."). For example, RFC1990/1661 teach an endpoint discriminator option ("EDO") as multipath information; the EDO includes a class-address pair, such as IP address, magic-number, etc. *E.g., id.*, 000019-20 ("The option contains a Class which selects an identifier address space and an Address which selects a unique identifier within the class address space."), 000021 ("The Class field is one octet and indicates the identifier address space. The most up-to-date

values of the LCP Endpoint Discriminator Class field are specified in the most recent “Assigned Numbers” RFC [7]. Current values are assigned as follows: 0 Null Class 1 Locally Assigned Address 2 *Internet Protocol (IP) Address ... PPP Magic-Number Block...*”), 000022 (“Class 2 - Internet Protocol (IP) Address ... An address in this class contains an IP host address”), 000023 (“Class 4 - PPP Magic-Number Block ... This is not an address but a block of 1 to 5 concatenated 32 bit PPP Magic-Numbers as defined in [2].”).

258. RFC1990/1661 teach one endpoint, typically the client, sending the EDO to another endpoint, typically a server that uses the EDO to discriminate between links to a particular client, which are to be aggregated together, from links to other clients. *E.g.*, Ex-1009, 000023 (“The Endpoint Discriminator Option ... advises a system that the peer on this link could be the same as the peer on another existing link. If the option distinguishes this peer from all others”), 000024-27 (“The option contains a Class which selects an identifier address space and an Address which selects a unique identifier within the class address space.”), 000024 (“As used here, magic-numbers are used to determine if two links are in fact from the same peer endpoint or from two distinct endpoints.”). Therefore, the combination teaches multipath information about the hosts and the paths between them, including the use of an ID. *Infra* §VI.B.3 (claim 2).

259. Mineno further teaches that the recipient of the report packet being either a mobile or fixed host of the cluster. Mineno teaches that each SHAKE host, e.g., the mobile and fixed hosts, has a link monitoring module with a path information table that tracks “all end-to-end path information when the cluster is formed,” and the information in the path information table is updated by each host when it receives the report packets. Ex-1006, ¶¶0015, ¶¶0017-18, ¶¶0026-28, ¶¶0033-36.

260. Mineno teaches the data-source device receiving multipath information (explained above) **about each of a plurality of network-edge devices** (e.g., other mobile hosts). For example, the information from the report and control packets is stored in the path information table, which includes multipath information, as explained above, about each of the plurality of mobile hosts in the cluster network. Each SHAKE host, e.g., the mobile and fixed hosts, has a link monitoring module with a path information table that tracks paths between each of the hosts—Mineno explains how the table includes “all *end-to-end* path information when the cluster is formed” and is updated with new sections when “new hosts are added to the cluster.” Ex-1006, ¶¶0033, ¶¶0027 (“Each host has a table which relates host with link number.”). Therefore, each host receives routing information about the full path to the other hosts (the end-to-end path information). *Id.*, ¶¶0033 (“It performs a role as a routing table indicated among link number and

relay host and destination host ...”), Figs. 2-3 (arrows added or colored to emphasize the multiple paths depicted):

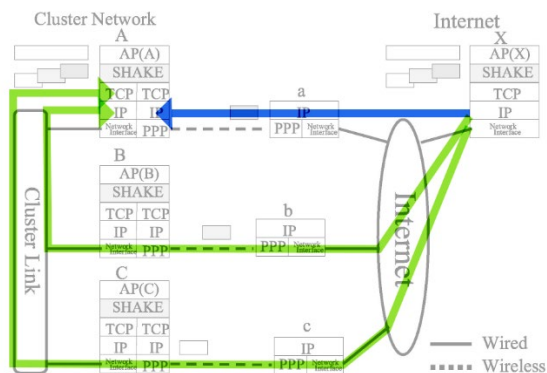


Figure 2: SHAKE Protocol Stack.
(Annotations/Emphasis Added)

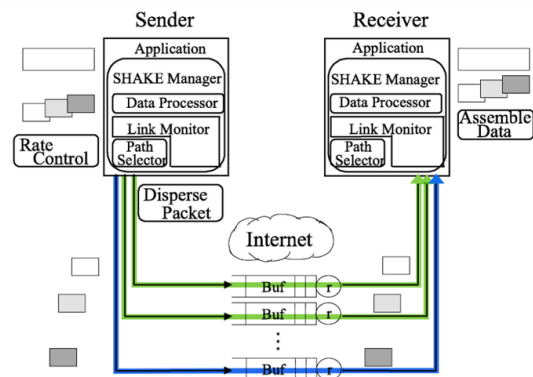


Figure 3: Illustration of SHAKE Modules.
(Annotations/Emphasis Added)

261. Each host thus receives multipath information, including routing information, about each of the other hosts. For example, in the context of Figs. 1-2, a SHAKE-enabled Internet host receives routing information about each of the mobile hosts in the cluster, and it uses that “end-to-end path information” to send packets (a) directly to a destination mobile host A and also (b) through the other mobile hosts B and C. Ex-1006, ¶¶0001-4, ¶¶0013-14 (“If one of mobile hosts in cluster has not a path to communicate with outside, it can transfer data by relaying other’s path. ... For those hosts which have no connection with outside, the communication can be also done through the relay hosts.”), ¶¶0015-18, ¶¶0026-28, ¶¶0033-34, , ¶¶0044-45, ¶0048 (“We proposed a protocol, SHAKE, *for sharing multiple paths* in a cluster type network in which a number of mobile hosts

temporarily connect mutually. *Each host in the cluster can share other hosts' communication links.*”), Figs. 1-2. Here, the Internet host knows it can route packets to the destination host A through the other mobile hosts (B and C) because it has “end-to-end path information” for routing packets, as Mineno teaches, from one end (the Internet host) to the other end (the destination mobile host). Ex-1006, ¶¶0026-28, ¶¶0033-34, Figs. 1-2. The Internet host therefore receives multipath information about each of a plurality of network-edge devices (e.g., the other mobile hosts A and C for this example). *See id.*

262. Additionally, the multipath information includes “information like theoretical bandwidth, round-trip delay, and receiving rate,” which is used to make routing decisions. *Id.*, ¶¶0033-34. For example, Mineno teaches distributing packets between paths based on the “sending rate.” *Id.*, ¶0034. In the context of Figs. 1-2, a SHAKE-enabled Internet host receives various bandwidth, delay, and receiving rate information about the paths through each of the mobile hosts, and it uses that information to distribute the number of packets routed to mobile host A, B, and C. *Id.*, ¶¶0001-4, ¶¶0013-23, ¶¶0026-28, ¶¶0033-36, Figs. 1-3. Therefore, the Internet host receives multipath information about each of the mobile hosts, including with respect to the bandwidth, delay, and sending/receiving rate for the paths for each host. *Id.*, ¶¶0019-23, ¶¶0033-36.

263. Mineno teaches a **plurality of network-edge devices** (e.g., a plurality of mobile hosts). Ex-1006, ¶¶0001-4, ¶¶0010-18, ¶¶0024-25, ¶0029, ¶¶0031-36, ¶0048, Figs. 1-3. Each of the other mobile hosts—the ones other than the destination mobile host—is a **network-edge device**. For example, Mineno teaches a cluster of mobile hosts that are connected using a local network, for example a local Ethernet or Wi-Fi network. Ex-1006, ¶¶0001-4, ¶¶0010-12, ¶0016 (“Cluster is formed by using Ethernet, serial cable, wireless LAN, IrDA and so on ...”), ¶0031, ¶¶0037-47. The cluster comprises a plurality of mobile hosts, each one being a device on the network. Mineno teaches an exemplary cluster three mobile hosts, though of course a local Ethernet or Wi-Fi network can include more laptops and desktops, and any of those additional computers can act as a relay host if it has its own Internet connection. *Id.*, ¶¶0001-4, ¶¶0010-18, ¶¶0024-25, ¶0029, ¶¶0031-36, ¶0048, Figs. 1-3. When a mobile host sends or receives data from a fixed host, the SHAKE protocol facilitates sending packets through other mobile hosts in the cluster. *Id.*, ¶¶0001-4, ¶¶0010-29, ¶¶0031-48, Figs. 2-3, 6-8, 10. Each of these other mobile hosts is a network-edge device, sitting on the edge of the cluster network and acting as a relay to send packets from the cluster to outside networks, e.g., the Internet. *Id.*, ¶¶0001-4, ¶¶0010-13, ¶¶0014-0016 (“In the cluster, we call a mobile host that has a communication path with outside network (like Internet) as a relay host.), ¶¶0017-18, ¶¶0024-28, ¶¶0033-36, Figs. 1-2 (blue emphasis added to

network-edge devices in an example with mobile host A as the Receiver/destination host, and green highlighting added to the cluster):

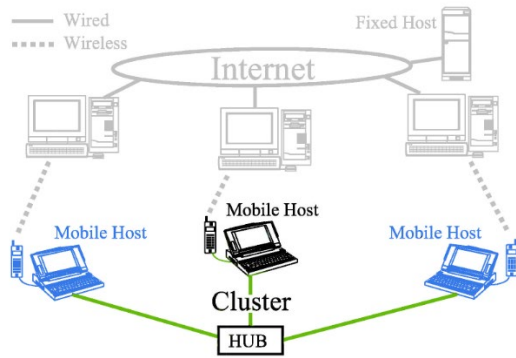


Figure 1: An Example of the System.
(Emphasis Added)

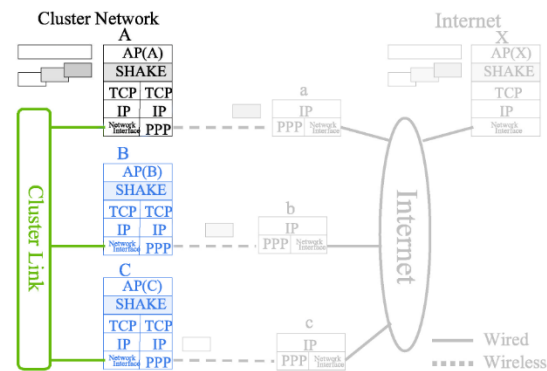


Figure 2: SHAKE Protocol Stack.
(Emphasis Added)

264. Mineno's teachings regarding the other mobile hosts fall within the scope of network-edge devices recited by the '649 patent.¹⁸ As explained above, the other mobile hosts route packets from the data source to the destination mobile host during a multipath connection (e.g., Ex-1006, ¶¶0001-4, ¶¶0010-18, ¶¶0024-28, ¶¶0033-36, Figs. 1-3); they therefore act as *routers* or *network switches* and are *devices capable of providing multipath services*. Mineno teaches each mobile host

¹⁸ This is confirmed by the '649 patent, where claim 16 recites that the network-edge device is "one or more *routers, modems, cell phones, tablets, computers, cellular towers, wireless towers, base stations, wireless base stations, repeaters, wireless repeaters, Wi-Fi devices, networking devices, hot spot devices, Wi-MAX devices, LTE devices, GSM devices, consumer electronic devices, an Internet-ready device, a network switch, a network router, a multipath server and a device capable of providing multipath services.*" Ex-1001, 18:14-22 (claim 16).

having a *cell phone* and/or *modem* (cell phones obviously included cellular modems inside) to connect with external networks. *Id.*, ¶¶0001-4, ¶0042, , ¶0047, Fig. 1, Fig. 10. Mineno teaches the mobile hosts can be *computers, tablets, or consumer electronic devices*. *Id.*, ¶¶0001-2, , ¶0031, Fig. 1, Figs. 6-8, Fig. 10. Mineno teaches the mobile hosts as *networking devices*, e.g., Ethernet and wireless devices, which obviously would have included at that time *Wi-Fi, LTE, and GSM devices*. *Id.*, ¶¶0001-2, ¶0016, ¶¶0037-47, Figs.1-3, Figs. 6-8. The mobile hosts are *Internet-ready devices*. *Id.*, ¶¶0001-4, ¶0010-12, ¶¶0014-16, ¶0017, ¶0024, ¶¶0031-32, ¶0037-42, Figs. 1-3, Figs. 6-8, Fig. 10.

265. Mineno teaches that each of the **network-edge devices** (e.g., the other mobile hosts, as explained above) **is capable of joining a multipath connection with a data-requesting device** (e.g., a Receiver/destination host). As explained above, the other hosts in the cluster are capable of joining a multipath connection with the data-requesting device, which allows their Internet connections—each part of a distinct connection pathway—to be aggregated for simultaneous data transfer. Ex-1006, ¶0001-4, ¶¶0010-18, ¶¶0024-29, ¶¶0031-42, ¶¶0048, Figs 2-3; *supra* §V.B (explaining claim construction for “multipath”), §VI.B.2[1pre] (explaining how Mineno teaches a multipath connection; that analysis is incorporated here); *infra* §VI.A.2[1d] (explaining simultaneous data transfer across multiple connection paths; that analysis is incorporated here).

266. Mineno teaches a **data-requesting device** (e.g., a Receiver/destination host) that requests data packets through the cluster. Mineno teaches the hosts requesting and receiving data packets through a multipath connection over TCP. *E.g.*, Ex-1006, ¶¶0001-3, ¶¶0017-18 (“As a protocol of Transport Layer, SHAKE uses TCP to transmit data. TCP connection is formed between mobile hosts in the cluster and fixed hosts in the Internet.”). For example, this includes a mobile host, such as a laptop, requesting data, such as a web page or FTP file, from a server over the Internet. *Id.*, ¶¶0001-2, ¶0004, ¶¶0015-17, ¶¶0031-32, Figs. 1-3. By the late 1990s and even more so by the relevant timeframe, the Internet and the World Wide Web were widely used, and mobile hosts, such as laptops, were routinely used to browse the Web, i.e., by requesting data from websites. *Supra* §I. Therefore, Mineno teaches a mobile host as a device that requests data from an Internet server. *Id.*, ¶¶0001-2, ¶0004, ¶¶0015-17, ¶¶0031-32, Figs. 1-3. At the very least Mineno suggests this, and it would have been obvious, because Web browsing was ubiquitous by the relevant timeframe, and it was common to use mobile hosts such as laptops to request data from Internet servers to browse the Web. This would have been common even to non-technical laptop users. A POSITA would have found this even more obvious. Therefore, Mineno teaches a mobile host as a data-requesting device.

267. Alternatively, Mineno also teaches data transfer in the other direction, with the fixed server as the data-requesting device. As explained above, the combination of Mineno and TCP teaches a symmetrical, bidirectional protocol, meaning a device running the SHAKE protocol can either send or receive data using the protocol. *Id.*, ¶¶0001-4, ¶¶0010-29, ¶¶0031-42, ¶0048, Figs. 2-3; Ex-1007, 000013, 000016. Mineno teaches SHAKE being implemented on the mobile hosts and fixed hosts. Ex-1006, ¶0015 (“SHAKE is implemented at all mobile hosts which form a cluster (cluster means cluster type network) and at fixed hosts which are connected to Internet and communicate with one of mobile hosts in a cluster.”), ¶0017, ¶0032. The SHAKE implemented mobile hosts and fixed host use TCP to transfer data through a multipath connection. *Id.*, ¶0001, ¶¶0003-4, ¶¶0010-18, ¶¶0026-29, ¶¶0032-36, ¶0048, Figs. 1-3. For example, in Fig. 3, Mineno teaches multipath data transfer using SHAKE between a “Sender” and a “Receiver,” which can be any of the hosts so long as SHAKE is installed on both. *Id.*, ¶0015, ¶¶0017-18, ¶0032, ¶¶0037-42, Figs. 2-3. Mineno further teaches examples of tests that were run to measure bandwidth, where those exemplary test setups included data going from the mobile host, as the Sender, to the fixed host over the Internet. *Id.*, ¶¶0037-47, Figs. 6-10. Therefore, Mineno also teaches usage models with the fixed host as a data-requesting device.

268. Mineno's teachings regarding the mobile hosts fall within the scope of data-requesting devices recited by the '649 patent.¹⁹ Mineno teaches each mobile host having a *cell phone* to connect with external networks. Ex-1006, ¶¶0001-4, ¶0042, ¶0047, Fig. 1, Fig. 10. Mineno teaches mobile hosts, which would include *computers* and *tablets*. *Id.*, ¶¶0001-2, ¶0031, Fig. 1, Figs. 6-8, Fig. 10. The hosts are *networking devices* and *mobile communication devices*, including with Ethernet and wireless LAN, e.g., *Wi-Fi* devices. *Id.*, ¶¶0001-2, ¶0016, ¶¶0037-47, Figs.1-3, Figs. 6-8.

269. For the reasons set forth above, Grounds 2-3 teach limitation [1a].

[1b] each of the network-edge devices being connected to one or more networks,

270. Grounds 2-3 teach **each of the network-edge devices being connected to one or more networks.**

271. Mineno teaches **each of the network-edge devices** (e.g., the other mobile hosts) **being connected to one or more networks.** For example, Mineno teaches each of the mobile hosts, including the relay hosts, being connected to an outside network such as the Internet. *E.g.*, Ex-1006, ¶¶0001-5, ¶0011, ¶¶0013-16

¹⁹ This is confirmed by the '649 patent, where claim 19 recites that the data-requesting device is "one of a *cell phone*, a *tablet*, a switch, a *network device*, a router, a *computer*, a *Wi-Fi device*, a *Wi-Max device*, a *mobile communications device*, a near field communications device, and a Bluetooth device." Ex-1001, 18:44-48 (claim 19).

(“SHAKE is implemented at all mobile hosts which form a cluster (cluster means cluster type network) and at fixed hosts which are connected to Internet and communicate with one of mobile hosts in a cluster ... In the cluster, *we call a mobile host that has a communication path with outside network (like Internet) as a relay host*. Each relay host in the cluster has two network interfaces: one is used to communicate with outside network, and the other is assigned the local private IP address before forming a cluster.”), *id.*, ¶¶0017 (“TCP connection is formed between mobile hosts in the cluster and fixed hosts in the Internet.”), ¶¶0019-25, ¶¶0029, ¶¶0031-42, ¶0048, Figs. 1-3:

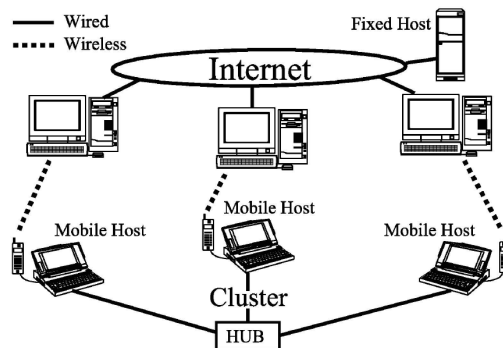


Figure 1: An Example of the System.

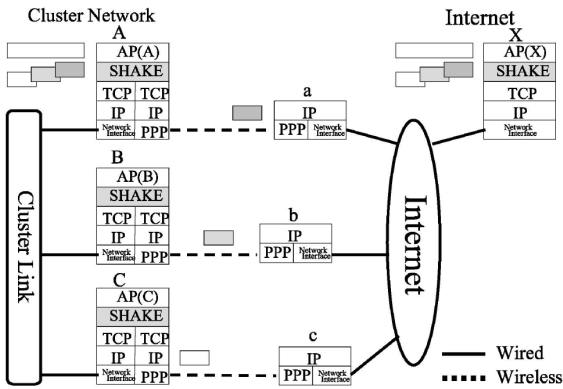


Figure 2: SHAKE Protocol Stack.

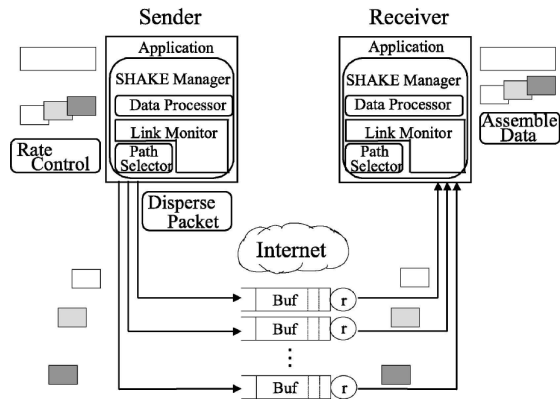


Figure 3: Illustration of SHAKE Modules.

272. Mineno teaches that the communication path with outside network, such as the Internet, includes a cellular data connection, and depicts laptop mobile hosts accessing a server, over the Internet, through the connection over a cellular phone. Ex-1006, ¶¶0001-4, ¶0042, ¶¶0047-48, Figs. 1, 10.

273. For the reasons set forth above, Grounds 2-3 teach limitation [1a].

[1c] and each of the network-edge devices being connected to the data-requesting device over a different connection; and

274. Grounds 2-3 teach **each of the network-edge devices being connected to the data-requesting device over a different connection.**

275. Mineno teaches **each of the network-edge devices** (e.g., the other mobile hosts) **being connected to a data-requesting device** (e.g., a Receiver/destination host) **over a different connection.** For example, Mineno teaches that each mobile host having a different connection to the cluster network,

including via Ethernet or wireless LAN. *E.g.*, Ex-1006, ¶0001 (“This paper proposes a protocol, SHAKE, for sharing multiple paths in cluster type network that is a kind of LAN in which *some mobile hosts temporarily connect mutually.*”), ¶¶0002-4, ¶¶0010-13, ¶0014 (“Any mobile host in a cluster can freely select any kinds of medium for communication between the cluster and the outside network. If one of mobile hosts in cluster has not a path to communicate with outside, it can transfer data by relaying other’s path.”), ¶¶0015-18, ¶0024, ¶¶0026-29, ¶¶0031-36, ¶¶0037-42, ¶0048, ¶0016 (“Each relay host in the cluster has two network interfaces: one is used to communicate with outside network, and the other is assigned the local private IP address before forming a cluster ”), Figs. 1-2 (emphasis added in red to connection between mobile hosts A and B, and in green to a different connection between A and C):

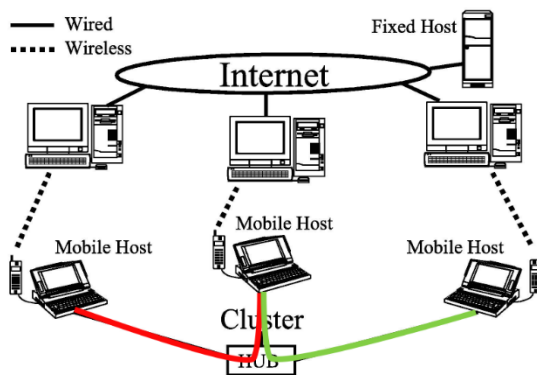


Figure 1: An Example of the System.
(Annotations/Emphasis Added)

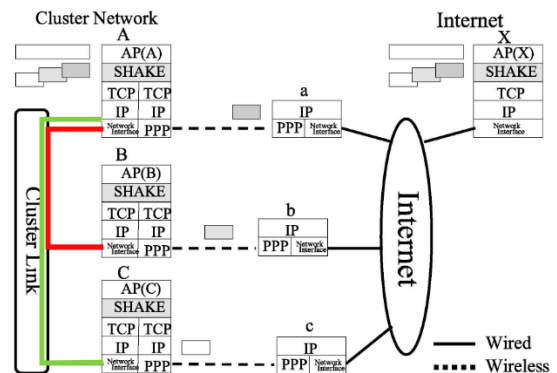


Figure 2: SHAKE Protocol Stack.
(Annotations/Emphasis Added)

276. For example, when mobile host A requests data from an Internet server, it is the data-requesting device, and the other mobile hosts are the network-edge devices. *Supra* §VI.B.2[1a] (explaining why the other mobile hosts are network-edge devices; that analysis is incorporated here). Each of the other mobile hosts (B and C) is connected with the data-requesting device over a different network connection. Mobile host B is connected to A over one connection, highlighted here in red, while mobile host C is connected to A over a different connection, highlighted here in green.

277. To the extent Patent Owner argues that limitation [1c] requires each of the network-edge devices to be connected to the data-requesting device over a different connection from those described in limitation [1b] (the connection(s) between the network-edge devices and one or more networks), Mineno also teaches this. For example, each of the other mobile hosts is connected to the data-requesting device over a connection to the cluster network (as explained above), which is different from the connection to the Internet/outside network (*supra* §VI.B.2[1b] (explaining connection to one or more networks; that analysis is incorporated here)). As Mineno explains, “[*e*]ach relay host in the cluster has two network interfaces: one is used to communicate with outside network, and the other is assigned the local private IP address before forming a cluster.” Ex-1006, ¶¶0003-4, ¶0011, ¶¶0013-16, 0024, ¶¶0031-32, ¶0037, Figs. 1-2.

278. For the reasons set forth above, Grounds 2-3 teach limitation [1c].

[1d] the data-source device performing at least one of (i) sending different groups of data packets to different network-edge devices that are participating in the multipath connection, (ii) sending different streams of data to one or more of the network-edge devices that are participating in the multipath connection, and (iii) sending combinations of different streams and different groups of data packets to one or more of the network-edge devices that are participating in the multipath connection.

279. Grounds 2-3 teach this limitation. As an initial matter, limitation [1d] recites sub-parts (i), (ii), and (iii). I understand that limitation [1d] is satisfied when any one of these sub-parts is satisfied. Ground 2 teaches all three.

280. For example, Mineno teaches **the data-source device** (e.g., a Sender host being a fixed host, or alternatively, a mobile host) **performing (i) sending different groups of data packets to different network-edge devices** (e.g., the other mobile hosts) **that are participating in the multipath connection**. As explained above, Mineno teaches the sender host sending data packets to the other (non-destination) mobile hosts in the cluster that are participating in the multipath connection, using different paths for each. *Supra* §§VI.B.2[1pre], [1a] (that analysis is incorporated here); Ex-1006, ¶0001 (“This paper proposes a protocol, SHAKE, for sharing multiple paths in cluster type network that is a kind of LAN in which some mobile hosts temporarily connect mutually.”), ¶¶0002-3, ¶0004 (“[U]sing different paths to communicate ... it is important to consider how to

distribute the burst traffic and how to reconstruct the packets.”) ¶¶0010-18, ¶¶0026-29, ¶¶0031-32, ¶¶0033-36, ¶0048 (“We proposed a protocol, SHAKE, for sharing multiple paths in a cluster type network in which a number of mobile hosts temporarily connect mutually.”), Figs. 1-2. For example, in Fig. 2, the fixed host X sends data packets to different mobile hosts A, B, and C. *Id.*, ¶¶0014-16, Figs. 1-3:

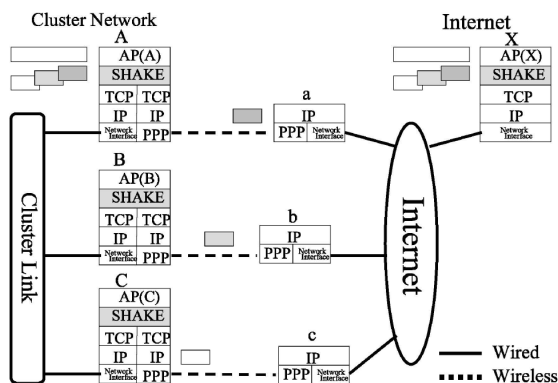


Figure 2: SHAKE Protocol Stack.

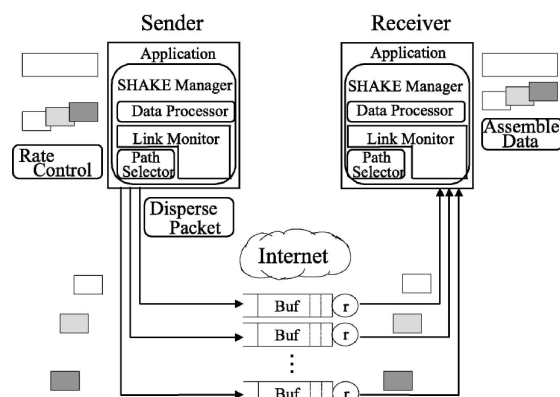


Figure 3: Illustration of SHAKE Modules.

281. Mineno teaches the Sender host sending **different groups of data packets** to different mobile hosts that are participating in the multipath connection. For example, Mineno teaches the Sender (e.g., fixed host in Figure 2) sending groups of different data packets through the different paths of the cluster—each path corresponds to one of the mobile hosts. Ex-1006, ¶¶0002-5 (“But using different paths to communicate, the quality of each paths such as transfer rate, delay time, and the rate of packet loss are different, and for this reason, it is important to consider how to distribute the burst traffic and how to reconstruct the

packets.”), ¶¶0019-23 (“The SHAKE evaluates how many packets have been sent to the object and how many packets still overstock in each network buffer. ... In sending host, each of the dispersed packets is added a sending time stamp ST1 in SHAKE header and sent to the receiving host.”), ¶¶0024-25 (“A sending host ... distributes the data packets which come from the application layer.”), ¶¶0026-28 (“The received data from application layer are divided into a number of packets.”), ¶¶0031-32 (“Then it can divide an application data into a number of determined size SHAKE packets and distributes to the destination.”), ¶¶0033-36 (“The data processing module ... divides application data into a number of SHAKE packets, and adds a special SHAKE header to each packet.”), ¶¶0037-42, ¶¶0043-47, Figs. 2-3 (using greyscale to show Internet server X sending different groups of data packets to different mobile hosts in the multipath SHAKE connection; colors added for emphasis):

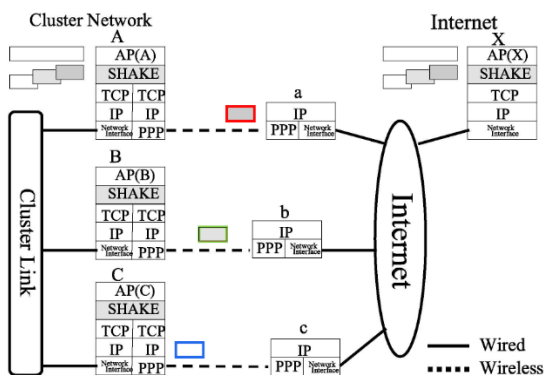


Figure 2: SHAKE Protocol Stack.
(Emphasis Added)

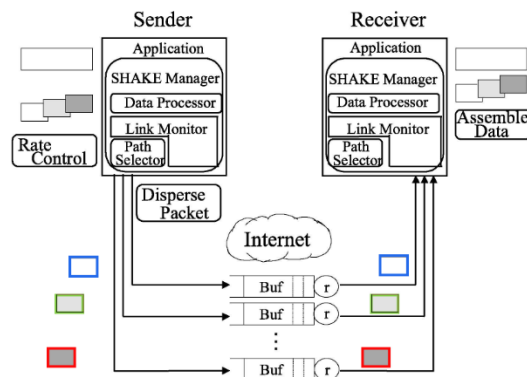


Figure 3: Illustration of SHAKE Modules.
(Emphasis Added)

282. As explained above, Mineno teaches multipath data transfer using SHAKE between a “Sender” and a “Receiver,” which can be any of the hosts so long as SHAKE is installed on both. *Id.*, ¶¶0015, ¶¶0017-18, ¶0032, ¶¶0037-42, Figs. 1-3. SHAKE sends different groups of data packets, indicated by a different “Link Number,” through the different paths in its multipath connection. *E.g., id.*, Figs. 4-5. For example, in Fig. 3 of Mineno, a Sender (such as a fixed host) sends different groups of data packets (each group highlighted in a different color) to a Receiver (such as a mobile host) through multiple paths, each going through a different mobile host; once received, these different groups of data packets will be assembled at the Receiver host. *E.g.*, Ex-1006, ¶¶0031-32 (“Then it can divide an application data into a number of determined size SHAKE packets and distributes to the destination. And in destination, the received packets is reconstructed and given to the application in order.”), ¶¶0017-18, ¶¶0026-28.

283. Mineno teaches the fixed host sending different groups of data packets to each mobile host in the cluster, e.g., group A being the data packets sent over link 1 to mobile host A, group B being the data packets sent over link 2 to mobile host B, and continuing up to group N being the data packets sent over link n to mobile host N. By splitting up the data packets sent between the fixed host and the destination host, with different groups sent over each link, Mineno allows the aggregate data rate of the links in the cluster to be utilized. *E.g., id.*, ¶0001 (“As a

mobile host has only low transfer capacity in individual to communicate with outside, if whole capacities of other hosts which compose cluster type network are shared, we can get larger transfer capacity and satisfy the required communication services.”), ¶¶0002-3, ¶¶0011-16, ¶¶0026-29, ¶¶0033-35, ¶0048, Fig. 3. These concepts also apply when data is sent in the inverse direction, from a mobile host to a fixed host.

284. TCP provides additional teachings for sending data packets to hosts. *E.g.*, Ex-1007, 00007 (“The Transmission Control Protocol (TCP) is intended for use as a highly reliable host-to-host protocol between hosts in *packet-switched computer communication networks*, and in interconnected systems of such networks. ... In principle, the TCP should be able to operate above a wide spectrum of communication systems ranging from hard-wired connections to packet-switched or circuit-switched networks.”), 000013 (“Processes transmit data by calling on the TCP and passing buffers of data as arguments. *The TCP packages the data from these buffers into segments* and calls on the internet module to transmit each segment to the destination TCP.”), 000013-14 (“The packet switches may perform further packaging, fragmentation, or other operations to achieve the delivery of the local packet to the destination internet module”), 000035 (“... emit packets with sequence numbers identical to or overlapping with packets still in the network which were emitted on an earlier incarnation of the same connection.”),

000035-36 (“... be treated as new packets by the receiver of the new incarnation of the connection ... Hosts which prefer to avoid waiting are willing to risk possible confusion of *old and new packets at a given destination* may choose not to wait for the ‘quite time’.”).

285. For Ground 3, RFC1990 provides additional teachings where one of the endpoints (it can be either one) sends different groups of data packets to different network-edge devices in a bundle. *E.g.*, Ex-1009, 000007 (“[O]ne can split packets among parallel virtual circuits between systems[.] ... The method discussed here is similar to the multilink protocol described in ISO 7776 [4], but offers the additional ability to split and recombine packets ...”), 000008-9 (“We suggest that multilink operation can be modeled as a virtual PPP link-layer entity wherein packets received over different physical link-layer entities are identified as belonging to a separate PPP network protocol (the Multilink Protocol, or MP) and recombined and sequenced according to information present in a multilink fragmentation header.”), 000011-12 (“Network Protocol packets are first encapsulated (but not framed) according to normal PPP procedures, and large packets are broken up into multiple segments sized appropriately for the multiple physical links.”), 000015-16 (“The sequence number is not reset upon each new PPP packet”), 000017 (“[T]he transmitter would first queue up as many consecutively numbered packets on one link as it could[.]”), 000018-19 (“[T]he

system will construe all packets received on this link as being able to be processed by a common protocol machine with any other packets received from the same peer on any other link”), 000023-24 (“[A] system MUST transmit a Multilink-encapsulated packet on X before transmitting any Multilink- encapsulated packets on Y.”); Ex-1008, 000005 (“The Point-to-Point Protocol (PPP) provides a standard method for transporting multi-protocol datagrams over point-to-point links.”), 000007-10 (“A datagram may be encapsulated in one or more packets passed to the data link layer.”).

286. Applying RFC1990’s teachings to Mineno, the combination teaches sending different data packets to the different mobile hosts participating in the multipath SHAKE connection. Ex-1009, 000007-12, 000015-19, 000023-24. Mineno explains that this teaching from RFC1990 as a known way to use traffic dispersion to increase bandwidth—a stated goal of Mineno. Ex-1006, ¶¶0001-3, ¶¶0006-9, ¶¶0011-13.

287. For the reasons set forth above, Grounds 2-3 teach limitation [1d].

3. Claim 2

[2] The method of claim 1, wherein the multipath information includes a connection ID that is generated by one of the data-source device, one of the plurality of network-edge devices, the data-requesting device and a network device.

288. Grounds 2-3 teach **the method of claim 1** (*supra* §VI.B.2), wherein **the multipath information includes a connection ID that is generated by one of**

the data-source device, one of the plurality of network-edge devices, the data-requesting device and a network device. As an initial matter, I understand that claim 2 is satisfied when any one of the data-source device, a network-edge device, the data-requesting device, or a network device generates the connection ID included in the multipath information. This is because claim 2 only requires that the connection ID be generated by “one of” these devices.

289. The combination of Mineno and TCP teaches that **the multipath information** (explained for claim 1) **includes a connection ID that is generated by one of the data-source device** (e.g., a Sender host being a fixed host, or alternatively, a mobile host), **one of the plurality of network-edge devices** (e.g., one of the other mobile hosts), **the data-requesting device** (e.g., a Receiver/destination host), or **a network device**. For example, Mineno teaches the method of claim 1 wherein the multipath information includes a “link number.” *E.g.*, Ex-1006, ¶¶0026-28, ¶¶0033-36, Figs. 4-5. When packets are transmitted by a host using the SHAKE protocol, a SHAKE header, including a link number, is added. *E.g.*, Ex-1006, ¶¶0017-25, ¶¶0026-28 (“Each packet is transmitted to transport layer after it is added a special header of SHAKE. This packet is a data packet, and the header includes: packet identifier (Data), link number ... And there is a report packet which includes: packet identifier (Report), link number ...”). The link number identifies a connection. *Id.*, ¶¶0016-20, ¶¶0026-28, ¶¶0033-36. As

Mineno teaches, SHAKE hosts use the link number in a packet header to determine where to relay the packet. Ex-1006, ¶0027. In particular, “[e]ach host has a table which relates host with link number.” *Id.* Relay hosts, such as the other mobile hosts in the cluster, forward packets to the destination host by looking up the packet’s link number in its table to identify the endpoint hosts associated with that link number, and then routing the packet to the destination endpoint. *Id.*, ¶0027 (“So if the packet is destined the host itself, the packet is given to the application and the others are relayed to the next host.”). Therefore, the link number identifies the connection between the sender and destination hosts, and SHAKE uses the link number to forward packets to the destination host. *Id.*, ¶0027, ¶0033-36.

290. Mineno teaches that the link number is added to headers by either the fixed hosts or mobile hosts; therefore, the link number is generated by either the fixed hosts or the mobile hosts—either satisfies the claim. Mineno teaches three types of hosts in a SHAKE connection: Sender, Receiver, and other mobile hosts—and any of these satisfy claim 2. Mineno at least suggests, and it would have been obvious, that the link number was generated by one of the devices taught by Mineno. Even if the link number were somehow generated by some other device on the network, this would still satisfy the claim because it would be a “network device,” as recited by claim 2. Therefore, Mineno teaches claim 2.

291. The combination of Mineno and RFC1990 further teaches claim 2 with additional details related to how **the multipath information includes a connection ID**. *Supra* §VI.B.1 (explaining how and why RFC1990’s connection ID teachings would have been combined with Mineno; that analysis is incorporated here). Mineno teaches a protocol that uses multiple paths between two endpoints, and RFC1990 teaches using an ID to identify the multiple links in the bundle. *See id.* Therefore, when RFC1990’s teachings are applied to Mineno, the combination teaches using an ID for Mineno’s multipath connection, and to include that ID in the multipath information that is exchanged between Mineno’s hosts, so that hosts can populate their host and routing tables, and so that the SHAKE protocol can route packets to their intended destination hosts. *See id.* Mineno itself teaches the combination, citing to the “[r]elated” teachings of IETF “Multilink Point-to-Point Protocol.” Ex-1006, ¶0008.

292. The combination with RFC1990/1661 teaches using a class-address pair that includes, e.g., IP address or magic-number, etc., as a connection ID. RFC1990/1661 teach including an endpoint discriminator option, which further includes a class-address pair, such as IP address, magic-number, etc.—that distinguishes the links in one bundle from others. *Supra* §VI.B.2[1a] (explaining how and why Ground 3 teaches multipath information; that analysis is incorporated here); Ex-1009, 000019 (“The Endpoint Discriminator Option ... advises a system

that the peer on this link could be the same as the peer on another existing link. If the option distinguishes this peer from all others”), 000020 (“The option contains a Class which selects an identifier address space and an Address which selects a unique identifier within the class address space.”), 000021 (“The Class field is one octet and indicates the identifier address space. ... Current values are assigned as follows: ... 2 Internet Protocol (IP) Address ... 4 PPP Magic-Number Block”), 000022 (“Class 2 - Internet Protocol (IP) Address ... An address in this class contains an IP host address”), 000023 (“Class 4 - PPP Magic-Number Block”), 000023 (“As used here, magic-numbers are used to determine if two links are in fact from the same peer endpoint or from two distinct endpoints.”). As RFC1990/1661 explain, “[t]he aggregate link, or bundle, *is named by the pair of identifiers for two systems connected by the multiple links*. A system identifier may include information provided by PPP Authentication[] and information provided by LCP negotiation.” Ex-1009, 000008.

293. Therefore, the combined teachings of Mineno and RFC1990/1661 at the very least suggest, and it would have been obvious, to assign an ID, e.g., a class-address pair such as IP address or magic-number, to the multipath connection between the Sending and Receiving hosts. Ex-1009, 000019-23. The combination further teaches including the connection ID in multipath information exchanged between hosts. For example, Mineno teaches adding new hosts to a cluster (Ex-

1006, ¶0033), and the combined teachings of Mineno and RFC1990/1661 suggest, and it would have been obvious, for the new host to present an identifier to allow other hosts to distinguish between connections to the host from other connections. *See, e.g.*, Ex-1009, 000020 (“To securely join an existing bundle, a PPP authentication protocol[] must be used to obtain authenticated information from the peer to prevent a hostile peer from joining an existing bundle by presenting a falsified discriminator option.”), 000023.

294. Ground 3 further teaches that the connection ID is generated by **one of the data-source device** (e.g., Sender host being a fixed host, or alternatively, a mobile host), **one of the plurality of network-edge devices** (e.g., other mobile hosts), **the data-requesting device** (e.g., a Receiver/destination host) **and a network device**. The combination of Ground 3 teaches this limitation because there are only a finite number of possibilities as to which device generates the connection ID: the sender/source endpoint, receiver/destination endpoint, or another mobile host. These are the devices taught by Mineno (Ex-1006, Figs. 1-3), and claim 2 is satisfied if the connection ID is generated by any of them. Even if another device on the network were to somehow generate the connection ID, the claim also allows this by reciting “a network device.”

295. Therefore, Grounds 2-3 teach claim 2.

4. Claim 3

[3] The method of claim 1, further comprising:

the data-source device receiving second multipath information about a second network-edge device that is capable of joining the multipath connection; and

the data-source device joining the second network-edge device to the multipath connection.

296. Grounds 2-3 teach **the method of claim 1** (*supra* §VI.B.2) and the additional limitations of claim 3. *Supra* §V.C (explaining claim construction).

297. Mineno teaches a **data-source device** (e.g., a Sender host being a fixed host, or alternatively, a mobile host) **receiving second multipath information about a second network-edge device** (e.g., an additional mobile host) **that is capable of joining the multipath connection**. As explained above, Grounds 2-3 teach a Sender host receiving multipath information about each of a plurality of network-edge devices that is capable of joining a multipath connection, e.g., the two other mobile hosts taught in the examples of Figures 1-2. *Supra* §VI.B.2[1a]. Mineno further teaches that additional mobile hosts—beyond the two other hosts shown in Figs. 1-2—can provide additional pathways between the Sender and Receiver. Ex-1006, ¶0001, ¶¶0003-4, ¶¶0010-12, ¶¶0014-16, ¶0024, ¶¶0026-28, ¶¶0033-36, ¶0048. Fig. 3 shows three pathways with ellipses to indicate the presence of additional pathways. Ex-1006, Fig. 3. Mineno teaches one path directly from the Sender to Receiver; each of the other paths flows through

one of the other mobile relay hosts in the cluster. *Id.*, ¶¶0001, ¶¶0003-4, ¶¶0010-12, ¶¶0014-16, ¶0019, ¶¶0026-28, ¶¶0033-36, Figs. 1-3. Therefore, Fig. 3 teaches adding additional mobile hosts, beyond the three shown in Figs. 1-2, to the cluster, and using them in the multipath connection between the Sender and Receiver. *Id.*

298. Mineno teaches that the fixed hosts and mobile hosts receive multipath information when “new hosts are added to the cluster.” Ex-1006, ¶0033. When new, additional hosts are added to the cluster, report and control packets provide path information to the other hosts, e.g., the fixed hosts and other mobile hosts in the cluster, so that their routing tables can be updated, and the new hosts can participate in the SHAKE protocol. Ex-1006, ¶0033 (“new sections are added to this table after new connecting”), ¶0029, ¶0036. Therefore, Mineno teaches the fixed and mobile hosts receiving multipath information, e.g., from report and control packets, regarding new mobile hosts added to the cluster, in the same manner as explained previously. *Supra* §VI.B.2[1a] (explaining how Mineno teaches the data-source device receiving multipath information about each of a plurality of network-edge devices that are capable of joining the multipath connection; that analysis incorporated here). The new mobile hosts are capable of joining the multipath connection via the SHAKE protocol; Mineno teaches that routing tables are updated to accommodate this, and Fig. 3 teaches the inclusion of

additional paths (through additional mobile hosts). *Id.*; Ex-1006, 0010-18, ¶¶0024, ¶¶0026-29, ¶¶0031-40, ¶¶0048, Fig. 3.

299. Mineno teaches that the fixed hosts and mobile hosts receive multipath information through the SHAKE management module that is installed on each of them. Ex-1006, ¶¶0017-18. ¶¶0048, Fig. 3. The SHAKE management module “control[s] the other modules, *form[s] the cluster*, ensur[es] communication paths,” “*check[s] among hosts* and manage[s] any other modules.” Ex-1006, ¶¶0029, ¶¶0036. Information received by the hosts through the SHAKE management module is multipath information that is used to “realize[] the shared multilink procedure by using multimedia through multiple paths to form a logical path”; moreover, the SHAKE management module keeps “an additional window of SHAKE” “opened for the request of *forming a cluster*,” which SHAKE uses to form its multipath connection. Ex-1006, ¶¶0029, ¶¶0036.

300. Mineno teaches **the data-source device** (e.g., a Sender host being a fixed host, or alternatively, a mobile host) **joining the second network-edge device** (e.g., an additional mobile host) **to the multipath connection**. For example, additional mobile hosts are added to routing tables as they join the cluster. Ex-1006, ¶¶0033, ¶¶0014-18, ¶¶0026. The Sending host—e.g., a fixed host sending data to a mobile host, or vice versa—uses the routing tables to select paths to use for the multipath connection between the Sender and Receiver. *E.g.*,

Ex-1006, ¶¶0033-34, ¶0024, ¶0026-28, ¶¶0014-18, Fig. 3. The Sending host joins paths, and the corresponding mobile host for each path, to the multipath connection. *Id.*, ¶0001, ¶¶0003-4, ¶¶0010-12, ¶¶0014-17 (“A mobile host can freely select any communication path and any kind of medium to communicate with outside.”), ¶0024, ¶0026-29, ¶¶0033-36, ¶0048, Figs. 1-3, 6-8, 10. Indeed, Fig. 3 teaches the Sender using additional mobile hosts to send data through additional paths to the Receiver host. *Id.* Therefore, the fixed host joins the additional mobile host to the multipath connection; when data flows in the opposite direction, with the fixed Internet host receiving data, a mobile host is the Sender that joins the additional mobile host to the multipath connection to the fixed host. Ex-1006, ¶¶0001-4, ¶¶0010-29, ¶¶0031-42, ¶0048, Figs. 2-3, 6-8, 10; *supra* §VI.B.2[1a] (explaining why and how the data flow is bidirectional; that analysis is incorporated here).

301. For the reasons set forth above, Grounds 2-3 teach claim 3.

5. Claim 4

[4] The method of claim 1, further comprising:

the data-source device receiving multipath information about which network-edge devices are to be removed from the multipath connection; and

the data-source device removing the network-edge devices from being associated with the multipath connection.

302. Grounds 2-3 teach claim 4.

303. For example, Mineno teaches that the Sending host, e.g., the fixed host, receives multipath information about which of the mobile hosts have been disconnected from the Internet, indicating that they should be removed from the multipath connection. “A sending host gets information about communication paths with outside from the link monitoring module, and then distributes the data packets which come from the application layer.” Ex-1006, ¶0024. The sending host chooses communication paths—which go through the respective mobile hosts—based on network conditions, which “is done by analyzing the information about the sending rate R_{snd} , and *disconnection report* from link monitoring module.” Ex-1006, ¶¶0019-24. The link monitoring module “monitors the status of communication paths” through the relay hosts, including the transmission rate and whether the path has been disconnected (“cut of communication paths”). Ex-1006, ¶0019-0024, ¶¶0033-35. “The present link monitoring module forms a path information table of all end-to-end path information” which is “updated by receiving the report packets.” Ex-1006, ¶0033. Report packets are received by the Sending host at regular intervals. Ex-1006, ¶0033. And the SHAKE management module in each host further “ensur[es] communication paths.” Ex-1006, ¶0029.

304. Therefore, the Sending host receives information, for example, from control packets, report packets, and/or disconnection reports, about relay hosts that have lost their Internet connection. Ex-1006, ¶0024, ¶0033. This is multipath

information about which paths, in Mineno's multipath connection, are no longer capable of transmitting data and therefore should be removed from the multipath connection. Ex-1006, ¶¶0011-12, ¶¶0014-16, ¶0019, ¶0024, ¶¶0026-29, ¶0032-36. Removal is further explained below.

305. Mineno further teaches **the data-source device** (e.g., a Sender host being a fixed host, or alternatively, a mobile host) **removing the network-edge devices** (e.g., disconnected mobile hosts) **from being associated with the multipath connection**. When a Sender host, e.g., a fixed host sending data to a mobile host over the Internet, receives information that a mobile host has been disconnected from the Internet, it removes that host from the multipath connection. Ex-1006, ¶0014-17, ¶0019, ¶0024, ¶¶0026-29, ¶0033-36. For example, the Sender host chooses the communication paths to use when sending data to the Receiver host (*id.*, ¶0014-17, ¶0019, ¶0024, ¶¶0026-29, ¶0033-36), and this consideration includes whether a path has been disconnected/cut (*id.*, ¶0019, ¶0024). When that happens, the "path information table" in the Sending host is updated, e.g., by receiving report packets, which affects the routing of packets through the multiple connection paths. *E.g.*, *Id.*, ¶¶0033-34. Each communication path corresponds to a path through a mobile host with a connection to an outside network such as the Internet. Ex-1006, ¶¶0001-4, ¶¶0010-12, ¶¶0014-17, ¶¶0031-32, Figs. 1-3. Therefore, removing a path from the multipath connection includes removing the

corresponding host, and Mineno teaches detecting and removing disconnected mobile hosts from the multipath connection between the Sender and Receiver. *See* Ex-1006, ¶¶0014-17, ¶0019, ¶0024, ¶¶0026-29, ¶0033-36, Fig. 3.

306. Additionally, Mineno suggests, and it would have been obvious, to remove a disconnected mobile host from the multipath connection because that host can no longer relay data to/from the outside network. It would make no sense to monitor the cluster for disconnections and then, after detecting a disconnection, to continue to send data down a disconnected path to a disconnected mobile host. Doing so would result in data loss. It would undermine Mineno's stated goal of improving data transfers, requiring retransmission of data. That is why Mineno monitors the links for disconnections: to prevent a communication path from being used if they are "cut." Ex-1006, ¶0019. It has long been known to avoid using disconnected paths. This is common sense. For example, when a broken pipe is discovered, it is common sense to disconnect that pipe to prevent water leaking. The same concept applies in networking.

307. And TCP provides additional teachings for closing connection paths. *See, e.g.,* Ex-1007, §3.5, 000011 ("When their communication is complete, the connection is terminated or closed to free the resources for other uses."), 000027-28, 000055 ("The user may CLOSE the connection at any time on his own initiative, or in response to various prompts from the TCP (e.g., remote close

executed, transmission timeout exceeded, destination inaccessible).”), 000009 (“For example, there are calls to open and close connections and to send and receive data on established connections.”), 000043-45 (“Closing a Connection ...”), 000066-67, Figs. 13-14 and accompanying text, 000018.

308. The combination with RFC1990/1661 further teaches **receiving multipath information about which network-edge devices are to be removed from the multipath connection**. For example, RFC1661 teaches endpoints receiving multipath information such as the Link Termination packets, which are used to remove a link from a multilink bundle. *E.g.*, Ex-1008, 000032 (“There are three classes of LCP packets: ... Link Termination packets used to terminate a link (Terminate-Request and Terminate-Ack).”), 000012 (“The link will remain configured for communications until explicit *LCP or NCP packets close the link down....*”), 000012 (Phase diagram and accompanying text), 000011 (“*PPP can terminate the link at any time ... LCP is used to close the link through an exchange of Terminate packets. ... After the exchange of Terminate packets, the implementation SHOULD signal the physical-layer to disconnect in order to enforce the termination of the link*”), 000021, 000025-000026 (“The Terminate-Request packet indicates the desire of the peer to close the connection.”), 000029 (“A Terminate-Request packet is transmitted. This indicates the desire to close a connection.”), 000039-40 (“LCP includes Terminate-Request

and Terminate-Ack Codes in order to provide a mechanism for closing a connection.”).

309. RFC1661 further teaches circumstances for removing a link, including for example, “loss of carrier” or link failure—which is analogous to Mineno’s teaching of monitoring for relay connections that are “cut.” *Compare* Ex-1006, ¶¶0019, ¶0024 *with* Ex-1008, 000015 (“loss of carrier, authentication failure, link quality failure, the expiration of an idle-period timer, or the administrative closing of the link.”), 000008, 000020, 000026.

310. Therefore, the combination of Mineno with RFC1990/1661 teaches a host—including the Sending host (e.g., fixed Internet host)—receiving multipath information in the form of termination packets about which of the other mobile hosts in the cluster (e.g., which relay hosts) are to be removed from the multipath connection, and the host removing those other mobile hosts. *E.g.*, Ex-1006, ¶¶0019, ¶0024; Ex-1008, 000012, 000021, 000025-000026, 000029, 000032, 000039-40. When a host is removed, RFC1990/1661 teach that it is no longer associated with the bundle as the corresponding link proceeds to “Link Dead phase” and data sent over the multilink bundle is no longer be assigned to that path. Ex-1008, 000015, 000012 (Phase diagram and accompanying text), 000009 (“Link Dead (physical-layer not ready) ... Typically, a link will return to this phase automatically after the disconnection of a modem.”), 000022-23 (“Down[:] This event occurs when a

lower layer indicates that it is no longer ready to carry packets. ... to signal LCP that the link is entering Link Dead phase.”). The link can later be re-initiated, but that is consistent with the scope of claim 4. *See supra* n. 10; *see, e.g.*, Ex-1008, 000021, 000024.

311. Therefore, Grounds 2-3 teach claim 4.

6. Claim 5

[5] The method of claim 1, further comprising at least one of the data-source device, the network-edge devices and the data-requesting device determining which of the network-edge devices are to be removed from the multipath connection.

312. Grounds 2-3 teach **the method of claim 1** (*supra* §VI.B.2) further comprising **at least one of the data-source device, the network-edge devices and the data-requesting device determining which of the network-edge devices are to be removed from the multipath connection**. As an initial matter, I note that claim 5 is satisfied by any one of the data-source device, one of the network-edge devices, or the data-requesting device performing the step of claim 5. This is because claim 5 requires “at least one of” these devices to perform the recited step.

313. As explained above, Mineno teaches that the data-source device (e.g., a Sender host being a fixed host, or alternatively, a mobile host) receiving multipath information about which network-edge devices are to be removed from the multipath connection. *Supra* §VI.B.5 (claim 4) (analysis incorporated here). The Sender host, e.g. a fixed host, receives information about which mobile hosts

have lost their Internet connection and therefore should be removed from the multipath connection. Ex-1006, ¶¶0014-16, ¶0019, ¶0024, ¶¶0026-29, ¶¶0033-36. Mineno teaches the Sender host removing those disconnected mobile hosts from being associated with the multipath connection. *Supra* §VI.B.5. Therefore, Mineno teaches, or at least suggests, the data-source device determining which of the network-edge devices are to be removed from the multipath connection—namely, the mobile hosts whose Internet connections have been disconnected/cut. Ex-1006, ¶¶0013-17, ¶¶0019-20, ¶0024, ¶¶0026-29, ¶¶0033-36.

314. Even if another mobile host or the destination made the determination, it still satisfies claim 5. Therefore, Grounds 2-3 teach claim 5.

7. Claim 6

[6] The method of claim 1, wherein the data-source device is one of a multipath link aggregator, a network relay, a multipath link server, a server, a network switch, a network router and a device capable of providing multipath services.

315. Grounds 2-3 teach **the method of claim wherein the data-source device** (e.g., a Sender host being a fixed host, or alternatively, a mobile host) **is one of a multipath link aggregator, a network relay, a multipath link server, a server, a network switch, a network router and a device capable of providing multipath services.**

316. Mineno teaches that the Sender host—either a fixed host or a mobile host—is **a device capable of providing multipath services.** As explained above,

the Sender host is capable of providing multipath services, for example by sending data using multiple communication paths to the Receiver host. *Supra* §VI.B.2[1pre], §VI.B.2[1a], §VI.B.2[1d] (analysis incorporated here); *e.g.*, Ex-1006, ¶¶0013-28, ¶¶0031-42, Figs. 1-3, 6-8, 10.

317. Mineno further teaches that the fixed host is **server**, connected to the mobile hosts over the Internet. Ex-1006, ¶¶0001-2, ¶¶0013-18, ¶¶0031-32. ¶¶0037-42, Figs. 1-2:

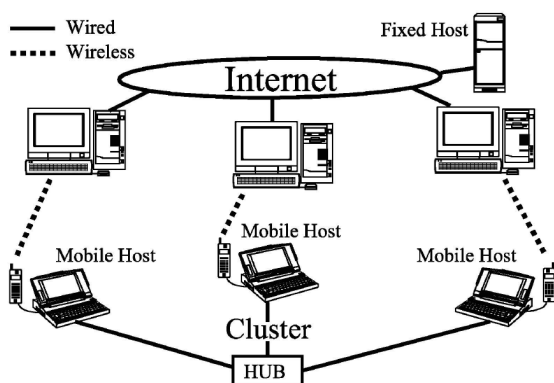


Figure 1: An Example of the System.

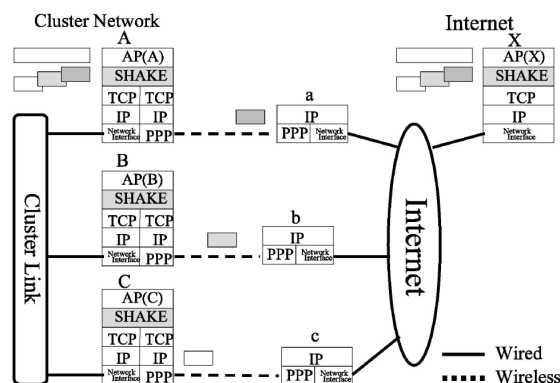


Figure 2: SHAKE Protocol Stack.

318. For the reasons set forth above, Grounds 2-3 teach claim 6.

8. Independent Claim 7

[7pre] A method for establishing and sending data during a multipath connection, comprising;

319. Grounds 2-3 teach limitation [7pre]. This limitation is identical to [1pre] and satisfied for the same reasons explained previously. *Supra* §VI.B.2[1pre].

320. Ground 2's and Ground 3's method for establishing and sending data during a multipath connection further comprises the steps described below. *Infra* §VI.A.2[7a]-[7e].

[7a] the data-source device receiving multipath connection requests from a plurality of data-requesting devices, each of the multipath connection requests involving a particular multipath connection to a data-requesting device;

321. Grounds 2-3 teach limitation [7a]. *Supra* §VI.B.1 (explaining how and why the references would have been combined; that analysis is incorporated here).

322. As an initial matter, as explained above, Grounds 2-3 teach **a data-source device** (e.g., a Sender host being a fixed host, or alternatively, a mobile host), **a data-requesting device** (e.g., a Receiver/destination host), and **a multipath connection** between them. *Supra* §VI.B.2 (explaining claim 1; that analysis is incorporated here).

323. The combination of Mineno and TCP further teaches **the data-source device receiving multipath connection requests from a plurality of data-requesting devices**. Mineno teaches the Sender an Internet server. *Supra* §VI.B.2[1a] (analysis incorporated here); Ex-1006, ¶¶0015-17, ¶¶0031-32, ¶¶0038-42, Figs. 1-2, 6-8. By the relevant timeframe, Internet servers were well known in the art. An Internet server serviced requests from many client devices. Otherwise, if an Internet server only existed to serve a single client, then each

website would have needed millions of servers to service millions of clients. This was not how servers operated. As was widely known, Internet servers served multiple clients at the same time. By teaching the fixed server as an Internet server, Mineno teaches that the fixed server will receive data requests from many client devices, either in the same or different clusters. Ex-1006, ¶¶0015-17, ¶¶0031-32, ¶¶0038-42, Figs. 1-2, 6-8, Figs. 1-2, 6-8.

324. Furthermore, this is reflected and taught in TCP (the dominant transport layer protocol on the Internet), which for example teaches utilizing multiple concurrent connections. Ex-1007, 000016 (“To provide for unique addresses within each TCP, we concatenate an internet address identifying the TCP with a port identifier to create a socket which will be unique throughout all networks connected together. ... TCPs are free to associate ports with processes however they choose.”), 000033, 000011 (emphasis added):

Multiplexing:

To allow for many processes within a single Host to use TCP communication facilities simultaneously, the TCP provides a set of addresses or ports within each host. Concatenated with the network and host addresses from the internet communication layer, this forms a socket. A pair of sockets uniquely identifies each connection. That is, a socket may be simultaneously used in multiple connections.

The binding of ports to processes is handled independently by each Host. ... Establishing and learning the port addresses of other processes may involve more dynamic mechanisms.

325. Therefore, TCP teaches hosts utilizing multiple concurrent TCP connections. Mineno teaches an Internet host as a data-source device and mobile hosts as data-requesting devices. *Supra* §VI.B.2[1a]. The combination with TCP therefore teaches the Internet host receiving multipath connection requests from a plurality of data-requesting devices, such as mobile hosts. At the very least, the combination suggests this, and it would have been obvious because, as explained above, Internet servers were expected to receive numerous data requests from numerous client devices (including mobile laptop hosts) concurrently. This was the norm for a server on the Internet, for example to receive numerous concurrent requests for data using HTTP over TCP (or FTP over TCP), and it is reflected in the teachings of TCP, as explained above.

326. Mineno teaches a SHAKE protocol that uses multipath connections between mobile hosts and Internet hosts. *Supra* §VI.B.2 (claim 1). Therefore, in the context of Mineno's teachings, each request for data from a mobile host to an Internet server is a multipath connection request because it requests data to be transferred utilizing a multipath SHAKE connection using TCP. *Supra* §VI.B.2[1pre]-[1a]. Additionally, when a SHAKE-enabled mobile host requests

data from a SHAKE-enabled server, it initiates and therefore is a request for a multipath connection between the hosts to transfer the requested data. *See id.*

327. The combination with RFC1990/1661 further teaches hosts (endpoints) sending multipath connection requests. RFC1990 teaches a negotiation process, described in RFC1661, with an exchange of Configure packets. Ex-1008, 000013 (“The Link Control Protocol (LCP) is used to establish the connection through an exchange of Configure packets.”), 000021 (“In the Request-Sent state an attempt is made to configure the connection. A Configure-Request has been sent . . .”), 000028-29 (“Send-Configure-Request (scr)[:] A Configure-Request packet is transmitted. This indicates the desire to open a connection with a specified set of Configuration Options.”), 000032-35 (“An implementation wishing to open a connection MUST transmit a Configure-Request.”). This packet exchange is used to configure the multilink bundle. Ex-1009, 000005 (“This document proposes a method for splitting, recombining and sequencing datagrams across multiple logical data links. . . . This is accomplished by means of new PPP [2] options and protocols.”) (citing RFC1661), 000007 (“Multilink is based on an LCP option negotiation that permits a system to indicate to its peer that it is capable of combining multiple physical links into a ‘bundle’.”), 000008-10 (“In order to establish communications over a point-to-point link, each end of the PPP link must first send LCP packets to configure the data link during Link Establishment

phase.”), 000019-24. Applying these teachings to Mineno, the combination teaches the mobile host sending configuration packets to the Internet server, to request a SHAKE transfer and establish a multipath connection between them. *See id.*

328. The combination of Mineno and TCP further teaches **each of the multipath connection requests involving a particular multipath connection to a data-requesting device**. Each request, as discussed above, is from a mobile host to the Internet server; when a SHAKE-enabled mobile host requests data (thereby becoming a Destination host), SHAKE utilizes a particular multipath connection between that host and the Internet server. Ex-1006, ¶¶0001-4, ¶¶0010-17, ¶0024, ¶¶0026-29, ¶¶0031-36, Figs. 1-3. For example, in Figs. 2-3, each host (e.g., the Receiver host, the mobile host A) can be the destination host for a multipath SHAKE connection to the Internet server X, whereby data from X is routed through the other mobile hosts. Ex-1006, ¶0024, ¶¶0026-28, ¶¶0033-35. I have annotated the below figures to reflect Mineno’s teachings, showing (a) the multipath connection between Internet server X and mobile host A on the left, and (b) the multipath connection between Internet server X and mobile host B on the right. Ex-1006, Fig. 2:

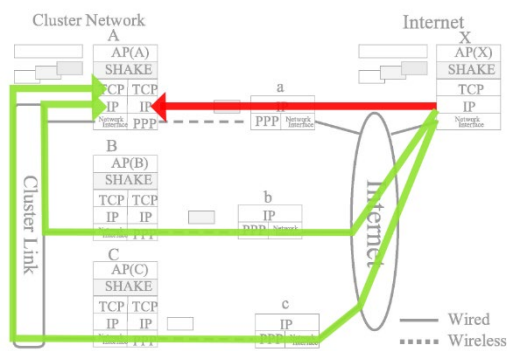


Figure 2: SHAKE Protocol Stack.
(Annotations/Emphasis Added)

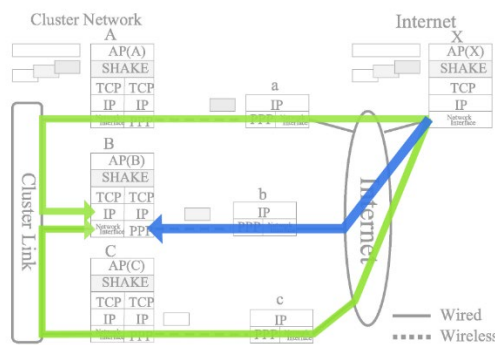


Figure 2: SHAKE Protocol Stack.
(Annotations/Emphasis Added)

329. The combination with RFC1990/1661 further teaches this limitation.

For example, RFC1990/1661 teaches a server using the EDO to, as its name implies, distinguish between links to various other endpoints, allowing an endpoint, such as a server, to have several distinct multilink bundles to several different endpoints. *E.g.*, Ex-1009, 000019-23 (“The Endpoint Discriminator Option represents identification of the system transmitting the packet. This option advises a system that the peer on this link could be the same as the peer on another existing link. If the option distinguishes this peer from all others, a new bundle MUST be established from the link being negotiated. If this option matches the class and address of some other peer of an existing link, the new link MUST be joined to the bundle containing the link to the matching peer or MUST establish a new bundle”). Therefore, the combination teaches each of the multipath connection requests involving a particular multipath connection to a particular mobile host.

330. Therefore, Grounds 2-3 teach limitation [7a].

[7b] the data-source device associating a connection ID to each multipath connection request;

331. Grounds 2-3 teach this limitation. As explained above, Grounds 2-3 teach **a connection ID** and **multipath connection requests**. *Supra* §VI.B.3 (claim 2) (explaining how Grounds 2-3 teach a connection ID generated by one of the host devices; that analysis is incorporated here); §VI.B.8[7a] (explaining the multipath connection request; that analysis is incorporated here); §VI.B.1 (explaining how and why the references would have been combined; that analysis is incorporated here).

332. Grounds 2-3 further teach **the data-source device** (e.g., a Sender host being a fixed host, or alternatively, a mobile host) **associating a connection ID to each multipath connection request**. For example, Mineno teaches the Sender host associating a connection ID (e.g., link number) to each multipath connection request—indeed, each multipath SHAKE connection has an associated link number that is used by hosts to direct packets to the requesting (destination) host. *Supra* §VI.B.3 (claim 2); Ex-1006, ¶¶0017-18, ¶¶0026-0028, ¶¶0033-36, Figs. 2-5. When sending packets to the destination host, the Sender host appends a SHAKE header to each packet that includes the link number; therefore, when sending data to a mobile host, the fixed Internet host associates a connection ID with each

multipath connection request for a particular mobile host. *Id.*; *see also* Ex-1006, ¶¶0017, ¶¶0026-28, ¶¶0033-35.

333. The combination with RFC1990/1661 further teaches this. RFC1990/1661 teaches associating a class-address pair (from the EDO) as a unique identifier for each multilink bundle. Ex-1009, 000019-23; *supra* §VI.B.3 (claim 2). The combination of Mineno and RFC1990/1661 therefore teaches associating a connection ID to each multipath connection request, so that hosts can identify that particular multipath connection, e.g., the Internet server that uses the particular multipath connection to send data to the particular destination host that requested it. *Id.*; Ex-1006, ¶¶0027-28, ¶¶0033-34, Figs. 2-3. Therefore, the combination teaches Sender host, e.g., the Internet host, associating a connection ID to each multipath connection request.

334. Therefore, Grounds 2-3 teach limitation [7b].

[7c]	the data-source device using multipath information about each of a plurality of network-edge devices that is capable of joining in a particular multipath connection to a particular data-requesting device, the particular multipath connection having a particular connection ID;
-------------	--

335. Grounds 2-3 teach limitation [7c]. The first portion of [7c] (shown right) tracks the language of [1a] (shown left):

1 the data-source device using multipath information about each of a plurality of network-edge devices that is capable of joining in a particular multipath connection to a particular data-requesting device,

1 a data-source device receiving multipath information about each of a plurality of network-edge devices that is capable of joining a multipath connection with a data-requesting device,

Therefore, Grounds 2-3 teach the overlapping aspects for reasons explained previously. *Supra* §VI.B.2[1a] (analysis is incorporated here).

336. Grounds 2-3 further teach the additional language of [7c], e.g., **using** the multipath information. Grounds 2-3 teach receiving the recited multipath information (*supra* §VI.B.2[1a]), and the whole purpose of the data-source device receiving the multipath information is to use it. For example, Mineno teaches the data-source device (e.g., a Sender host being a fixed host, or alternatively, a mobile host) using multipath information, which includes a link number, by appending it in the SHAKE header to packets that it sends through the SHAKE protocol to the destination host. Ex-1006, ¶0026, ¶0017, ¶0035, Fig. 4 (*see also* Ex-1025). The Sender host also uses the multipath information to generate and update a path information table, to keep track of changes in the cluster (e.g., new hosts added to the cluster), and to “search[] if [a] destination host is in the information table when the application requests for sending data.” Ex-1006, ¶¶0033-34.

337. Grounds 2-3 further teach that the multipath information includes a connection ID, which is used to route packets through the multipath connection and to distinguish a particular connection between a Sender and Receiver host from other connections to other hosts. *Supra* §VI.B.8[7c] (analysis incorporated here).

338. Grounds 2-3 teach that each of a plurality of network-edge devices is capable of joining in a **particular** multipath connection to a **particular** data-requesting device. As explained above, each multipath SHAKE connection uses a plurality of network edge devices (e.g., the other mobile hosts) that join in a particular multipath connection to a particular data-requesting device (e.g., a Receiver/destination host). *Supra* §VI.B.2[1a] (analysis incorporated here), §VI.B.8[7a] (further explaining multipath connection request; that analysis is incorporated here); *see also*, e.g., Ex-1006, Figs. 2-3:

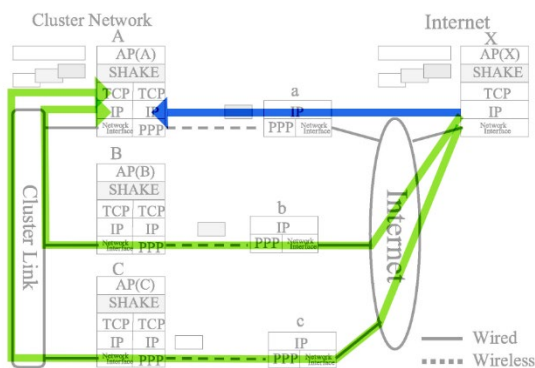


Figure 2: SHAKE Protocol Stack.
(Annotations/Emphasis Added)

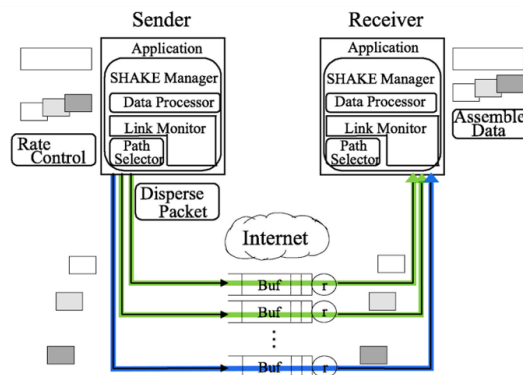


Figure 3: Illustration of SHAKE Modules.
(Annotations/Emphasis Added)

339. **The particular multipath connection has a particular connection ID.** Grounds 2-3 teach, as explained above, that each multipath SHAKE connection has a connection ID. *Supra* §VI.B.8[7a]-[7b] (explaining how a connection ID is associated with each multipath connection request; that analysis is incorporated here), §VI.B.3 (claim 2; analysis incorporated here). Therefore, the

multipath SHAKE connection has a particular connection ID, e.g., the link number (Ground 2) and the class-address pair (Ground 3). *Id.*

340. For reasons set forth above, Grounds 2-3 teach limitation [7c].

[7d] the data-source device notifying each of the network-edge devices that is participating in the particular multipath connection associated with the particular connection ID; and

341. Grounds 2-3 teach **the data-source device** (e.g., a Sender host being a fixed host, or alternatively, a mobile host) **notifying each of the network-edge devices** (e.g., another mobile host that is not the Receiver host) **that is participating in the particular multipath connection associated with the particular connection ID**. As explained above, Grounds 2-3 teach that the particular multipath connection between a Sender and Receiver host is associated with a particular connection ID. *Supra* §VI.B.8[7a]-[7b] (analysis incorporated here).

342. Before the Sender host transmits data using the multipath SHAKE protocol, it notifies each of the other mobile hosts that is participating in the particular multipath connection, e.g., when a TCP connection is established between them. Mineno teaches that SHAKE uses TCP to send data between hosts, e.g., between the Internet server X and mobile hosts A, B, and C. Ex-1006, ¶¶0013-17, ¶0024, ¶¶0026-28, ¶¶0033-36, Figs. 2-3:

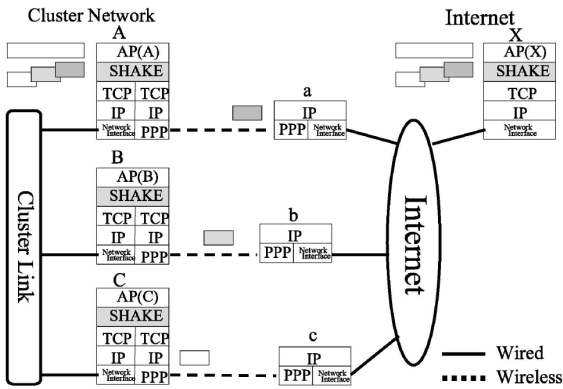


Figure 2: SHAKE Protocol Stack.

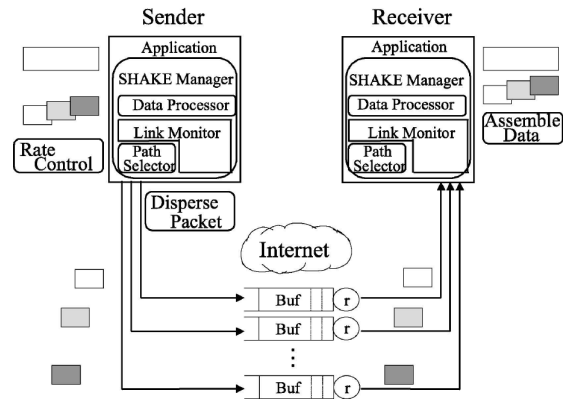


Figure 3: Illustration of SHAKE Modules.

343. Mineno teaches the Sender host sending packets through multiple paths to the receiver (e.g., as shown in Figure 3), with each path corresponding to a different mobile host (e.g., as shown in Figure 2). *Supra* §VI.B.2[1a] (explaining Mineno's multipath connection; that analysis is incorporated here). This includes an example where the Sender is the Internet server X. *See id.* Since Mineno uses TCP, the Sender uses TCP connections to send packets in each path, e.g., to the Receiver host and the other mobile hosts in the cluster that relay packets to the Receiver host. Ex-1006, ¶¶0013-17, ¶¶0026-28, ¶¶0033-36, Figs. 2-3. Therefore, Mineno teaches, or at least suggests, the Sender host forming TCP connections with the mobile hosts because those TCP connections are used to send data as taught in Figures 2 and 3. *Id.*

344. When the server establishes a TCP connection with the mobile hosts—including the other mobile hosts in the cluster that relay packets to the

Receiver host—the server notifies the mobile hosts during the three-way handshake that establishes the TCP connection. For example, the server notifies a mobile host by initiating a TCP connection, which starts by sending a SYN segment to the mobile host. Ex-1007, 000018 (“The procedures to establish connections utilize the synchronize (SYN) control flag and involves an exchange of three messages. This exchange has been termed a three-way hand shake”), 000022, 000027-29, 000033 (“For a connection to be established or initialized ... This is done in an exchange of connection establishing segments carrying a control bit called ‘SYN’ (for synchronize)”), 000036-38, Fig. 8 and accompanying text, Fig. 7 (depicting “[t]he simplest three-way handshake”) and accompanying text:

TCP A		TCP B
1. CLOSED		LISTEN
2. SYN-SENT	--> <SEQ=100><CTL=SYN>	--> SYN-RECEIVED
3. ESTABLISHED	<-- <SEQ=300><ACK=101><CTL=SYN,ACK>	<-- SYN-RECEIVED
4. ESTABLISHED	--> <SEQ=101><ACK=301><CTL=ACK>	--> ESTABLISHED
5. ESTABLISHED	--> <SEQ=101><ACK=301><CTL=ACK><DATA> --> ESTABLISHED	

Basic 3-Way Handshake for Connection Synchronization

Figure 7.

345. Mineno teaches that the Sending host decides which paths to use (Ex-1006, ¶¶0013-17, ¶0024, ¶¶0026-28, ¶¶0033-36) and therefore teaches, or at least suggests, that the Sending host notifies the other mobile hosts by initiating

TCP connections to them. At the very least, Mineno suggests this, and it would have been obvious as one of a finite number of possibilities. And even if the other mobile hosts were to initiate the TCP connection, the Sending host would still notify each of the mobile hosts that is participating in the particular multipath connection during the three-way handshake that initiates a TCP connection, e.g., by sending SYN-ACK to each of the other mobile hosts. Ex-1007, 000036-38.

346. Therefore, Grounds 2-3 teach limitation [7a].

[7e] the data-source device sending data associated with the particular connection ID to the network-edge devices participating in the particular multipath connection associated with the particular connection ID.

347. As explained above, Grounds 2-3 teach that **the particular multipath connection** between a Sender and Receiver host is **associated with a particular connection ID**. *Supra* §VI.B.8[7a]-[7b] (analysis incorporated here), §VI.B.3. Grounds 2-3 further teach **the data-source device** (e.g., a Sender host being a fixed host, or alternatively, a mobile host) **sending data associated with the particular connection ID to the network-edge devices** (e.g., other mobile hosts) **participating in the particular multipath connection associated with the particular connection ID**.

348. Ground 2 teaches the Link Number as the connection ID. *Supra* §VI.B.3 (claim 2). Data sent over the particular multipath connection between

Sender and Receiver is associated with the connection ID—the Sender host adds a SHAKE header to each packet that includes the Link Number. Ex-1006, ¶¶0017, ¶¶0026-28, ¶¶0033-35, Fig. 4. Additionally, the data sent over a multipath connection between the Sender host and Receiver host is associated with the multipath connection and the particular connection ID for that connection.

349. Ground 3 further teaches using an ID, such as a class-address pair, to identify the multiple paths in the connection. Data sent over the particular multipath connection between Sender and Receiver is associated with the connection ID. RFC1990/1661 teach, for example, using IP addresses for the class-address pair (*supra* §VI.B.3 (analysis incorporated here)), and data packets sent over a SHAKE multipath connection would be associated with the IP addresses of the Sender and Receiver because SHAKE runs on top of IP, which includes IP addresses in the header. *E.g.*, Ex-1006, Fig. 2. Additionally, the data sent over a multipath connection between the Sender host and Receiver host is associated with the multipath connection and the particular connection ID for that connection.

350. For Grounds 2-3, Mineno teaches that the Sending host sends data for a particular multipath connection to the other mobile hosts participating in the particular multipath connection, and those other mobile hosts route packets to the destination/Receiver host. *Supra* §VI.B.2[1a], [1d] (explaining how the SHAKE

protocol is used to send data and route packets through other mobile hosts; that analysis is incorporated here).

351. Therefore, Grounds 2-3 teach limitation [7e].

9. Claim 8

[8] The method of claim 7, further comprising: the data-source device sending different groups of data packets to different network-edge devices that are participating in the particular multipath connection associated with the particular connection ID.

352. Grounds 2-3 teach claim 8. Claim 8 includes some identical language from limitation [1d], as shown below. Grounds 2-3 teach these shared limitations, as explained above. *Supra* §VI.B.2[1d] (analysis incorporated here).

¹ [1d] the data-source device performing at least one of (i) sending different groups of data packets to different network-edge devices that are participating in the multipath connection

¹ 8. The method of claim 7, further comprising: the data-source device sending different groups of data packets to different network-edge devices that are participating in the particular multipath connection associated with the particular connection ID.

353. Grounds 2-3 further teach the different language in claim 8. Grounds 2-3 teach the data-source device sending different groups of data packets to different network-edge devices that are participating in the multipath connection. *Supra* §VI.B.2[1d]. Grounds 2-3 further teach the data-source device sending data to the network-edge devices participating in the particular multipath connection associated with the particular connection ID. *Supra* §VI.B.8[7e], §VI.B.3. Therefore, Grounds 2-3 teach the data-source device sending different groups of data packets to different network-edge devices that are participating in the particular multipath connection associated with the particular connection ID.

10. Claim 9

[9] The method of claim 7, further comprising:

the data-source device receiving identification of the network-edge devices to be removed from the particular multipath connection; and

the data-source device removing the identified network-edge devices from the particular multipath connection.

354. Grounds 2-3 teach **the method of claim 7**, further comprising **the data-source device** (e.g., a Sender host being a fixed host, or alternatively, a mobile host) **receiving identification of the network-edge devices** (e.g., a mobile host) **to be removed from the particular multipath connection; and the data-source device removing the identified network-edge devices from the particular multipath connection.**

355. As an initial matter, claim 9 is nearly identical to claim 4, with the only differences highlighted below. Ground 2 teaches these shared limitations, as explained above. *Supra* §VI.B.5 (claim 4) (explaining claim 4; that analysis is incorporated here). Ground 2 and 3 further teaches the different language in claim 9.

1 4. The method of claim 1, further comprising:
2 the data-source device receiving multipath information about
which network-edge devices are to be removed from the multipath connection; and
3 the data-source device removing the network-edge devices from being associated with the multipath connection.

1 9. The method of claim 7, further comprising:
2 the data-source device receiving identification of the network-edge devices to be removed from the particular multipath connection; and
3 the data-source device removing the identified network-edge devices from the particular multipath connection.

356. Grounds 2-3 teach the data-source device (e.g., a Sender host being a fixed host, or alternatively, a mobile host) receiving **identification of the**

network-edge devices (e.g., disconnected mobile hosts) to be removed from the **particular** multipath connection. As explained above, Grounds 2-3 teach a Sender host receiving multipath information about which network-edge devices have been disconnected and are to be removed from the multipath connection between the Sender and Receiver. *Supra* §VI.B.5 (claim 4). Therefore, this information identifies the network-edge devices that are to be removed from the particular multipath connection between the Sender and the Receiver host, e.g., by identifying the mobile hosts, and their respective connections, that have been disconnected/cut. *Id.*; Ex-1006, ¶¶0017, ¶¶0019, ¶¶0024, ¶¶0026-29, ¶¶0033-35, Fig. 5. The Sender then removes the **identified** mobile hosts from the **particular** multipath connection between the Sender and Receiver. *Supra* §VI.B.5 (claim 4).

357. Therefore, Grounds 2-3 teach claim 9.

11. Claim 10

[10] The method of claim 7, further comprising the data-source device notifying the data-requesting device when all data has been sent over the particular multilink connection.
--

358. Grounds 2-3 teach **the method of claim 7, further comprising the data-source device** (e.g., a Sender host being a fixed host, or alternatively, a mobile host) **notifying the data-requesting device** (e.g., a Receiver/destination host) **when all data has been sent over the particular [multipath] connection.** *Supra* §V.D (claim construction).

359. Mineno, for example, uses TCP to send data from the Sender to Receiver host. Ex-1006, ¶0017. TCP, in turn, teaches the sender sending a control bit FIN which notifies the receiving end that all data has been sent and there is “[n]o more data from Sender.” Ex-1007, 000021-22 (“Control Bits: 6 bits (from left to right): ... *FIN: No more data from sender[.]*”), Figure 3 and accompanying text, 000028-29 (including Figure 6 and accompanying text), 000032 (“For sequence number purposes, the SYN is considered to occur before the first actual data octet of the segment in which it occurs, while *the FIN is considered to occur after the last actual data octet in a segment in which it occurs.*”). TCP further teaches the sender using a “CLOSE” operation, “meaning ‘I have no more data to send.’” Ex-1007, 000043. The sender “who expects no data in return need only wait to hear the connection was CLOSED successfully to know that all his data was received at the destination TCP. Users must keep reading connections they close for sending until the TCP says no more data.” *Id.*; *see also id.*, 000044-45, 000018, 000085 (“FIN[:] A control bit (finis) occupying one sequence number, which indicates that the sender will send no more data or control occupying sequence space.”). Therefore, when the Sender host closes its TCP connection with the Receiver host, as taught by TCP, the Sender is notifying the Receiver that “I have no more data to send” and there is “[n]o more data from Sender.” Ex-1007, 000021-22, 000043. In the combination with Mineno, this is a

notification to the Receiver that all data has been sent by the Sender over the multipath connection.

360. Therefore, Grounds 2-3 teach claim 10.

12. Claim 11

[11] The method of claim 7, wherein the data-source device is one of a multipath link aggregator, a network relay, a multipath link server, a server, a network switch, a network router and a device capable of providing multipath services.
--

361. Grounds 2-3 teach the method of claim 7, with the additional limitation recited in claim 11, which is identical to claim 6 and satisfied for the same reasons explained previously. *Supra* §VI.B.7 (claim 6).

13. Claim 12

[12] The method of claim 7, wherein the connection ID is generated by one of the data-source device, one of the plurality of network-edge devices, the data-requesting device, and a network device.
--

362. Grounds 2-3 teach the method of claim 7, with the additional limitation of claim 12. The language recited in claim 12 is also recited in claim 2 and is satisfied by Grounds 2-3 for the same reasons explained previously. *Supra* §VI.B.3 (claim 2).

14. Independent Claim 21

[21pre] A method for creating and using a multipath connection, comprising the steps of:

363. Grounds 2-3 teach **a method for creating and using a multipath connection**. Limitation [21pre] includes some identical language from limitation [1pre], and Grounds 2-3 teach these shared limitations, as explained above. *Supra* §VI.B.2[1pre] (analysis incorporated here).

364. Grounds 2-3 further teach **creating a multipath connection**, for example by establishing a multipath connection, and explained above. *Supra* §VI.B.2[1pre]-[1a] (analysis incorporated here).

365. Grounds 2-3 teach **using a multipath connection**. For example, Grounds 2-3 teach using a multipath connection to send data from Sender to Receiver, as explained above. *Supra* §§VI.B.2[1pre]-[1a], [1d] (analysis incorporated here).

366. Additionally, Grounds 2-3 teach a method for creating and using a multipath connection further comprises the steps described below. *Infra* §VI.B.14[21a]-[21d].

[21a] a data-requesting device sending first multipath control information to or through a first network-edge device, the first multipath control information indicates that the first network-edge device will join the multipath connection, the data-requesting device being connected over a first connection to the first network-edge device;

367. Grounds 2-3 teach limitation [21a].

368. Mineno teaches **a data-requesting device** (e.g., a Receiver/destination host) **sending first multipath control information to or through a first network-edge device** (e.g., one of the other mobile hosts), **the first multipath control information indicates that the first network-edge device will join the multipath connection**. As an initial matter, Grounds 2-3 teach a destination/receiving host as a mobile host requesting data from a server, and a plurality of network-edge devices, each being capable of joining a multipath connection with a data-requesting device, as explained above. *Supra* §VI.B.2[1a] (analysis incorporated here). Any one of the plurality of network-edge devices satisfies the “first” network-edge device of [21a].

369. Mineno teaches a data-requesting device (e.g., a Receiver/destination host) **sending multipath control information** to or through network-edge devices (e.g., other mobile hosts), e.g., sending a “control packet” that is used “to form a cluster” of mobile hosts and to “make a link table” that is used to route packets through the multiple paths of a SHAKE connection. Ex-1006, ¶¶0017-18, ¶¶0026-28; *supra* §VI.B.2[1pre]-[1a] (analysis incorporated here). This indicates to

the other mobile hosts that they will join the multipath connection. *Id.* For example, when a new, SHAKE-enabled mobile host joins the cluster to initiate a multipath SHAKE transfer from an Internet host, the new host sends control information to the other mobile hosts indicating that they will join the multipath connection. Ex-1006, ¶¶0017-18, ¶¶0026-29, ¶¶0033-36, ¶¶0017-18.

370. The information from Mineno's control packets is stored in the SHAKE management module of each host, which is responsible for "controlling the other modules, forming the cluster, ensuring communication paths and so on." Ex-1006, ¶0029, ¶¶0033-36, ¶¶0017-18, Fig. 3. Each host has a table that relates hosts with a link number, which is used to route SHAKE packets to their destination host. *Id.*, ¶0027. Therefore, Mineno teaches, or at least suggests, the control packets including multipath control information regarding the link number and paths for a multipath SHAKE connection. *Id.*, ¶¶0027-28, ¶0033-35. When a mobile host sends this link information to the other mobile hosts, it indicates that the other mobile hosts will join the multipath connection. *Id.*

371. RFC1990/1661 reinforce Mineno's link number teachings. For example, RFC1990/1661 teaches the inclusion of a unique connection ID to distinguish one multipath connection from others. *Supra* §VI.B.3 (explaining EDO teachings; that analysis is incorporated here). Therefore, the combination teaches a destination/receiver mobile host providing a connection ID to other mobile hosts to

indicate that they will join a multipath SHAKE connection and facilitate the connection, as Mineno already teaches or suggests. *See id.*

372. Furthermore, mobile hosts use TCP to send SHAKE packets to each other. *E.g.*, Ex-1006, ¶¶0017, Figs. 2-3. Therefore, Mineno teaches the Receiver mobile host establishing TCP connections with other mobile hosts to initiate a multipath SHAKE connection, and TCP teaches a three-way handshake (*e.g.*, SYN and SYN-ACK exchange), which comprises multipath control information because Mineno uses it to establish the TCP communication paths in a multipath SHAKE connection. Ex-1006, ¶¶0012-18, ¶0001, ¶0003, ¶¶0026-29, ¶¶0033-36, ¶0048, Figs. 1-3; Ex-1007, 000018, 000032-33, 000090, 000021-22 (“Control Bits: 6 bits ...”), 000013, Figs. 3, 6-8 and accompanying text. The Receiver mobile host either sends SYN or SYN-ACK to the other mobile hosts, either of which is sufficient to satisfy multipath control information in this context. *See id.* When a destination/receiver mobile host sends this multipath control information to another mobile host, it indicates that the other mobile host will join the multipath connection—that is, after all, the purpose of establishing TCP connections between the mobile hosts: to facilitate a multipath SHAKE connection between Sender and Receiver hosts. Ex-1006, ¶¶0017-18, ¶¶0026-29, ¶¶0033-36.

373. Grounds 2-3 further teach **the data-requesting device being connected over a first connection to the first network-edge device**. Grounds 2-3

teach a plurality of network-edge devices, each being connected to the data-requesting device over a different connection; the connection between the destination mobile host and another mobile host is a first connection. *Supra* §§VI.B.2[1a], [1c] (analysis incorporated here).

374. Therefore, Grounds 2-3 teach limitation [21a].

[21b] the data-requesting device sending a second multipath control information to or through a second network-edge device, the second multipath control information indicates that the second network-edge device will join the multipath connection, the data-requesting device being connected over a second connection to the second network-edge device, the data-requesting device capable of receiving different data over the first connection and the second connection at the same time; and
--

375. Grounds 2-3 teach limitation [21b].

376. Grounds 2 or 3 teach **the data-requesting device sending a second multipath control information to or through a second network-edge device, the second multipath control information indicates that the second network-edge device will join the multipath connection, the data-requesting device being connected over a second connection to the second network-edge device.**

This portion of [21b] is identical to [21a], but replacing “first” with a “second” network-edge device and “second” connection. As explained above, Grounds 2-3 teach a plurality of network-edge devices, each being connected to the data-requesting device over a different connection. *Supra* §VI.B.14[21a], §§VI.B.2[1a]-

[1c] (analysis incorporated here). Any one of the plurality of network-edge devices and corresponding connection satisfies the “first” network-edge device and connection; any other network-edge device and corresponding connection satisfies the “second” network-edge device and “second” connection, for reasons explained previously. *Id.* For example, when mobile host A is the data requesting device, mobile host B is a first network-edge device and mobile host C is a second network-edge device. *Id.*

377. Grounds 2-3 further teach **the data-requesting device capable of receiving different data over the first connection and the second connection at the same time**. As explained above, different groups of data packets are sent to the different network-edge devices, which correspond to the different connection paths in the multipath SHAKE connection. *Supra* §§VI.B.2[1c]-[1d] (analysis incorporated here). Mineno shows data being sent in parallel over these paths, each of which traverses a separate Internet connection, and therefore the data-requesting device is capable of receiving different data over the first and second connections at the same time. *E.g.*, Ex-1006, ¶¶0001-4, ¶¶0012-16, ¶¶0026-28, ¶¶0037-47, Figs. 1-3.

378. Therefore, Grounds 2-3 teach limitation [21b].

[21c] after the first network-edge device joins the multipath connection, the data-requesting device receiving first data over the first connection from the first network-edge device; and
--

379. Grounds 2-3 teach limitation [21c]. For example, Mineno teaches that, in Fig. 2, **after the first network-edge device** (e.g., one of the other mobile hosts) **joins the multipath connection, the data-requesting device** (e.g., Receiver/a destination host) **receiving first data over the first connection from the first network-edge device**. As explained above, different groups of data packets are sent to the different network-edge devices, which correspond to the different connection paths in the multipath SHAKE connection. *Supra* §§VI.B.2[1c]-[1d], §§VI.B.14[21a]-[21b] (analysis incorporated here). Mineno shows data being sent in parallel over these paths, each of which traverses a separate Internet connection. *E.g.*, Ex-1006, ¶¶0003-4, ¶¶0011-16, ¶¶0026-28, ¶¶0033-35, Figs. 1-3. Any one of the plurality of network-edge devices and corresponding connection satisfies the “first” network-edge device and “first” connection. *Supra* §VI.B.14[21a].

380. Therefore, Mineno teaches that, after the first network-edge device joins the multipath connection, for example as illustrated in Figures 2-3, the data-requesting device receives data over the first connection that is different from the data sent through other connection paths. *Supra* §VI.B.2[1d] (analysis incorporated here). The data received over the connection from the first network-edge device would be the recited first data. *E.g.*, Ex-1006, Figs. 2-3 (annotations added to

illustrate mobile host A as the destination host and mobile host B as the first network-edge device):

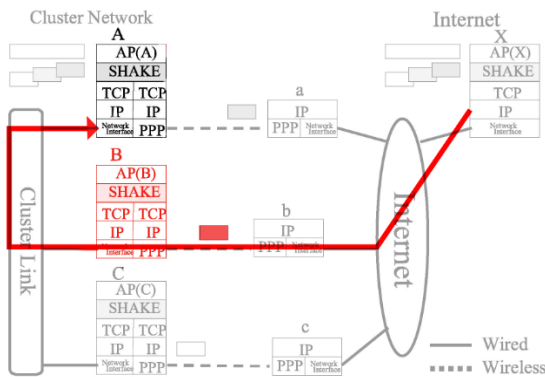


Figure 2: SHAKE Protocol Stack.
(Annotations/Emphasis Added)

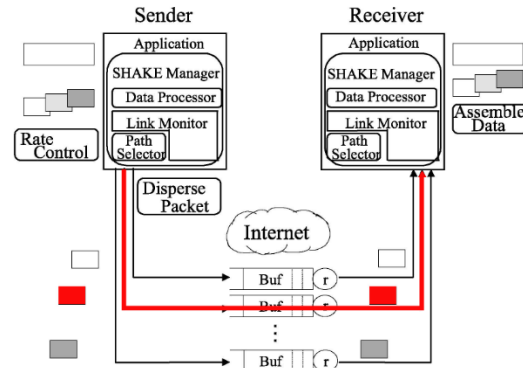


Figure 3: Illustration of SHAKE Modules.
(Emphasis Added)

381. Therefore, Grounds 2-3 teach limitation [21c].

[21d] after the second network-edge device joins the multipath connection, receiving second data at the data-requesting device over the second path from the second network-edge device.

382. Grounds 2-3 teach limitation [21d], which replaces references to the “first” network-edge device and connection with references to the “second.” As explained above, Grounds 2-3 teach a plurality of network-edge devices, each being connected to the data-requesting device over a different connection path. *Supra* §§VI.B.14[21a]-[21b], §§VI.B.2[1a]-[1c] (analysis incorporated here). Any one of the plurality of network-edge devices and corresponding connection satisfies the “first” network-edge device and connection; any other network-edge

device and corresponding connection satisfies the “second” network-edge device and “second” connection, for reasons explained previously. *Id.*

383. Therefore, the above analysis from [21c] also applies to the “second” network-edge device and connection here because Mineno’s teachings apply to both. *Supra* §VI.B.14[21c] (analysis incorporated here). Mineno teaches that, after the second network-edge device joins the multipath connection, for example as illustrated in Figures 2-3, receiving second data at the destination/receiving host over the second path from the second network-edge device. *Supra* §VI.B.2[1d] (analysis incorporated here). The data received over the connection path from the second network-edge device would be the recited second data. *E.g.*, Ex-1006, Figs. 2-3 (annotations added to illustrate mobile host A as the destination host and mobile host C as the second network-edge device):

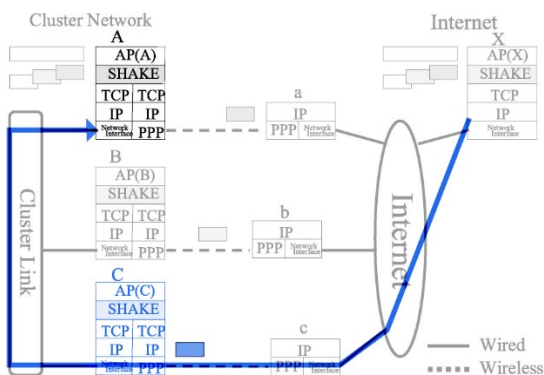


Figure 2: SHAKE Protocol Stack.
(Annotations/Emphasis Added)

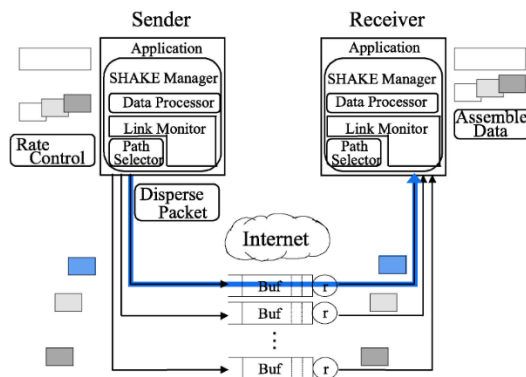


Figure 3: Illustration of SHAKE Modules.
(Emphasis Added)

384. Therefore Grounds 2-3 teach limitation [21d].

15. Claim 22

[22] The method as in claim 21, wherein the first network-edge device is connected to a network, and wherein the second network-edge device is connected to the network or a different network.

385. Grounds 2-3 teach claim 22. As explained above, Mineno teaches each of the network-edge devices (e.g., each of the other mobile hosts) being connected to one or more networks. *Supra* §VI.B.2[1c] (analysis is incorporated here). Mineno teaches each of the mobile hosts being connected to a network that is different from the other mobile hosts. For example, in Figs. 1-2, each mobile host has its own connection to a different network (e.g., b and c), and the different networks have separate connections to the Internet. *E.g.*, Ex-1006, Figs. 1-2:

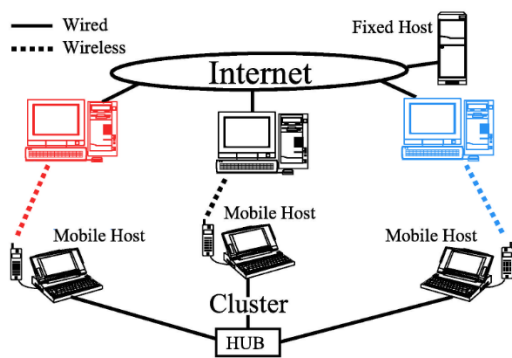


Figure 1: An Example of the System.
(Emphasis Added)

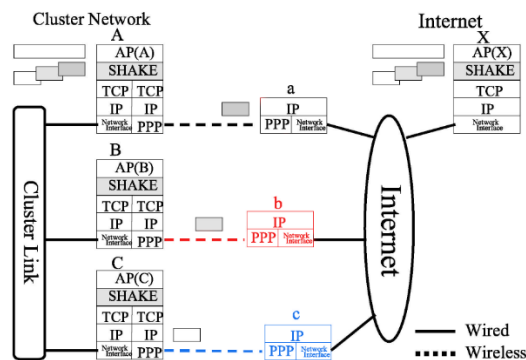


Figure 2: SHAKE Protocol Stack.
(Emphasis Added)

386. Therefore, Grounds 2-3 teach claim 22.

VII. BACKGROUND AND QUALIFICATIONS

387. This section contains a summary of my educational background, career history, publications, and other relevant qualifications. My full curriculum vitae is attached as Appendix 1 to this declaration.

388. I am an expert in the field of telecommunications and networking and have been an expert in that field since before the relevant timeframe, and before the challenged patent was filed. A copy of my *curriculum vitae* and prior testimony is provided as Appendix 1 to this report and provides a comprehensive description of my relevant experience, including my academic and employment history, publications, professional activities, and speaking engagements.

389. I received a B.S. in Electrical Engineering and Computer Sciences from the University of California, Berkeley, in 1985. I received an M.S. in Electrical Engineering and Computer Sciences in 1988 and a Ph.D. in Electrical Engineering and Computer Sciences in 1991, both from the University of California, Berkeley.

390. I am currently a Full Professor and the Department Chair of Electrical and Computer Engineering at the University of California, San Diego (“UCSD”), where I have been a Professor since 1997.

391. At UCSD, my teaching and research has focused on computer architecture and computer networking problems, including the design of high-

performance switches and routers, high-speed packet processors, wireless networks, data networks, and various hardware accelerators for networking applications.

392. At UCSD, I am a Principal Investigator in the UCSD Center for Wireless Communications (CWC). CWC brings together researchers to work on a range of challenges in the design of future wireless communications systems. I am also a Principal Investigator in the UCSD Center for Networked Systems (CNS). CNS brings together researchers to work on a range of challenges in the design of computer architecture solutions for packet processing and computer networking.

393. Before joining UCSD, I led the System Control and Communications Group at IMEC, Europe's largest independent microelectronics and information technology research center. At IMEC, I led a research team working on a range of computer design problems, including specialized processors for wireless communications and computer networking. I also previously held positions at Hewlett Packard, Hughes Electronics, and Western Digital.

394. During my career, I have received or worked on research efforts that received millions of dollars in research funding from both government agencies and industry, including funding for research in packet processing, computer networks, network measurements, wireless communications, processor design, and hardware/software co-design.

395. I have authored numerous publications, including books, chapters in books, and journal papers, and I have presented numerous invited lectures and keynote presentations around the world. In total, I have authored or co-authored over 200 journal articles and conference papers. A number of my publications have received best paper awards or distinguished paper citations, including best paper awards in IEEE Transactions and ACM/IEEE sponsored conferences. I have also given numerous invited and keynote talks around the world.

396. I have also been awarded five patents.

397. I have served as an Associate or Guest Editor for several prestigious journals published by the Association for Computer Machinery (“ACM”) and the Institute of Electrical and Electronics Engineers (“IEEE”). I have also served on the Editorial Board of the International Journal of Embedded Systems. Further, I have served as General Chair of several ACM/IEEE conferences, and on the Organizing or Steering Committees of many ACM/IEEE conferences, including the ACM/IEEE Symposium on Architectures for Networking and Communication Systems (ANCS) and the ACM/IEEE Symposium on Networks-on-Chips (NOCS). I have served on the Technical Program Committees of over 45 ACM/IEEE conferences.

A. Compensation

398. For my efforts in connection with the preparation of this declaration I have been compensated at my standard rate for this type of consulting activity. My compensation is in no way contingent on the results of these or any other proceedings relating to the above-captioned patent.

B. Materials and Other Information Considered

399. I have considered information from various sources in forming my opinions. I have reviewed and considered each of the exhibits listed in the attached Appendix 2 (Materials Considered in the Preparation of This Declaration) in forming my opinions.

VIII. UNDERSTANDING OF THE LAW

400. I am not an attorney. In forming my opinions in this Declaration, I applied the relevant legal principles provided to me by counsel, which are summarized in Appendix 4.

IX. RESERVATION OF RIGHTS

401. My opinions are based upon the information that I have considered to date. I am unaware of any evidence of secondary considerations with respect to the '649 Patent that would render any of the challenged claims non-obvious. I reserve the right, however, to supplement my opinions in the future to respond to any arguments raised by the owner of the '649 Patent and to take into account new information that becomes available to me.

402. I declare that all statements made herein of my knowledge are true, and that all statements made on information and belief are believed to be true, and that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code.

Executed on April 28, 2023.

By:

A handwritten signature in black ink, appearing to be 'Bill Lin', written over a horizontal line.

BILL LIN

APPENDIX 1: CURRICULUM VITAE OF BILL LIN

BILL LIN
Curriculum Vitae
Tel: 858.531.2441
Email: billmbox@gmail.com

EDUCATION

- **Ph.D**, Electrical Engineering and Computer Sciences
University of California, Berkeley, May 1991
- **Masters of Science**, Electrical Engineering and Computer Sciences
University of California, Berkeley, May 1988
- **Bachelor of Science**, Electrical Engineering and Computer Sciences
University of California, Berkeley, May 1985

PROFESSIONAL EXPERIENCE

- | | |
|------------------|---|
| 1/1997 – present | University of California, San Diego
Department Chair, Electrical and Computer Engineering
Professor, Electrical and Computer Engineering |
| 2/1992 – 12/1996 | IMEC, Leuven, Belgium
Group Head, Systems Control and Communications Group |

AREAS OF EXPERTISE

- All aspects of computer architecture and computer network problems, including the design of heterogeneous multicore processors, systems-on-chips, data networks, wireless communications and mobile computing systems, and multimedia and graphics systems.

PROFESSIONAL ACTIVITIES

- Served or serving as Editor on 3 ACM or IEEE journals, as General Chair on 4 ACM or IEEE conferences, on the Organizing or Steering Committees for 6 ACM or IEEE conferences, and on the Technical Program Committees of 45 ACM or IEEE conferences.

SELECTED PROFESSIONAL ACTIVITIES

- Associate Editor, ACM Transactions on Design Automation and Electronics Systems (TODAES), 2010-2015
- Editorial Board, International Journal of Embedded Systems, 2003-present
- General Chair, ACM/IEEE Symposium on Networks-on-Chips (NOCS), 2009
- Steering Committee, ACM/IEEE Symposium on Networks-on-Chips (NOCS), 2009-present
- General Chair, ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS), 2010
- Chair of Steering Committee, ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS), 2012-present
- Guest Editor, IEEE Transactions on Network and Service Management (TNSM), 2012-2013
- General Co-Chair, ACM SIGMETRICS, 2015
- General Chair, ACM/IEEE International Symposium on Quality-of-Service (IWQoS), 2011
- Steering Committee, ACM/IEEE International Symposium on Quality-of-Service (IWQoS), 2011-2015

PUBLICATIONS

- Published over 200 journal articles and conference papers in top-tier venues and publications.

PATENTS

- Mitigating Low-Rate Denial-of-Service Attacks in Packet-Switched Networks. United States Patent 8,443,444. Issued in May 14, 2013.
- Systems and Methods for Proactive Surge Protection. United States Patent 7,860,004. Issued in December 28, 2010.
- Method and Apparatus for Scan Block Caching. United States Patent 7,672,005. Issued in March 2, 2010.
- Design Environment and a Design Method for Hardware/Software Co-Design. United States Patent 5,870,588. Issued in February 9, 1999.
- System and Method for Generating a Hazard-Free Asynchronous Circuit. United States Patent 5,748,487. Issued in May 5, 1998.

EXPERT WITNESS HISTORY (BILL LIN)

I have served as an expert witness in 30 engagements (most engagements involved multiple patents in suit and multiple matters), of which I wrote expert reports or declarations in 21 engagements, analyzed source code in 14 engagements, was deposed 20 times, and testified at trial 4 times. The specific engagements and services provided are as follows:

1. Vizio, Inc. v. LG Electronics, Inc. and LG Electronics USA Inc., Civil Action No. 09-cv-1481-BEL
 - Expert witness on behalf of Vizio, Inc.
 - Law firm: Jones Day.
 - Services provided: Provided expert report, analyzed source code.
 - Technologies: Digital TV.
2. Vizio Inc. v. LG Electronics, Inc. and LG Electronics USA Inc., In the Matter of CERTAIN FLAT PANEL DIGITAL TELEVISIONS AND COMPONENTS THEREOF, U.S. International Trade Commission Inv. No. 337-TA-733
 - Expert witness on behalf of Vizio, Inc.
 - Law firm: Jones Day.
 - Services provided: Provided expert report, analyzed source code.
 - Technologies: Digital TV.
3. Vizio Inc. v. Coby Electronics Corp., Curtis International Ltd., E&S International Enterprises, Inc., MStar Semiconductor, Inc., ON Corp US, Inc., Renesas Electronics Corporation, Japan, Renesas Electronics America, Inc., Sceptre, Inc., and Westinghouse Digital, LLC, In the Matter of CERTAIN DIGITAL TELEVISIONS AND COMPONENTS THEREOF, U.S. International Trade Commission Inv. No. 337-TA-789
 - Expert witness on behalf of Vizio, Inc.
 - Law firm: Jones Day.
 - Services provided: Provided expert reports, was deposed, analyzed source code.
 - Technologies: Digital TV.
4. Linex Technologies Inc. v. Hewlett-Packard Company, Apple Inc., Aruba Networks, Inc., Meru Networks, and Ruckus Wireless, In the Matter of CERTAIN WIRELESS COMMUNICATION DEVICES AND SYSTEMS, COMPONENTS THEREOF, AND PRODUCTS CONTAINING SAME, U.S. International Trade Commission Inv. No. 337-TA-775
 - Expert witness on behalf of Hewlett-Packard Company, Apple Inc., Aruba Networks, Inc., Meru Networks, and Ruckus Wireless
 - Law firms: Covington & Burling, K&L Gates, Morrison & Foerster, Lewis Roca & Rothgerber.
 - Services provided: Provided expert reports, was deposed, analyzed source code.
 - Technologies: WiFi networks.
5. MOSAID Technologies, Inc. v. Dell, Inc., Qualcomm Atheros, Inc., et al., Case No. 2:11-cv-00179-TJW
 - Expert witness on behalf of Qualcomm Atheros, Inc.
 - Law Firm: McDermott Will & Emory.
 - Services provided: Provided expert analysis.
 - Technologies: WiFi networks.
6. Hitachi Consumer Electronics Co. Ltd., et al. v. Top Victory Electronics (Taiwan) Co. Ltd., et al., Civil Action No. 2:10-cv-260-JRG.
 - Expert witness on behalf of Vizio, Inc.
 - Law Firm: Akin Gump Strauss Hauer & Feld
 - Services provided: Provided expert reports, analyzed source code.
 - Technologies: Digital TV.

7. United States International Trade Commission (USITC) Investigation Number: 337-TA-853 and Technology Properties Limited, LLC v. Kyocera Corporation and Kyocera Communications, Inc. in the Northern District of California, Case No. 3:12-cv-03860-JSC
 - Expert witness on behalf of Kyocera Corporation and Kyocera Communications.
 - Law firm: Morrison & Foerster.
 - Services provided: Provided expert analysis.
 - Technologies: Cell phones, microprocessors.
8. Netgear, Inc. v. Ruckus Wireless, Inc., Case No. 1:10-cv-00999-SLR / D. Del.
 - Expert witness on behalf of Ruckus Wireless, Inc.
 - Law firms: Lewis Roca & Rothgerber, Orrick.
 - Services provided: Provided expert report, analyzed source code, was deposed, and testified at trial.
 - Technologies: WiFi networks.
9. Sonics, Inc. v. Arteris, Inc., Case No. 11 CV-05311 SBA, and Arteris S.A.S. v. Sonics, Inc., Case No. C 12-00434 (WHA)
 - Expert witness on behalf of Arteris, Inc. and Arteris S.A.S.
 - Law firm: DLA Piper.
 - Services provided: Retained as an expert.
 - Technologies: Systems-on-Chips (SoCs).
10. Linx Technologies Inc. v. Hewlett-Packard Company, Apple Inc., Aruba Networks, Inc., Meru Networks, and Ruckus Wireless, U.S. District Court, Northern District of California, C 13-00159 CW
 - Expert witness on behalf of Hewlett-Packard Company, Apple Inc., Aruba Networks, Inc., Meru Networks, and Ruckus Wireless
 - Law firms: Covington & Burling, Milbank, K&L Gates, Morrison & Foerster, Lewis Roca & Rothgerber.
 - Services provided: Provided expert reports, analyzed source code.
 - Technologies: WiFi networks.
11. Frontier Communications Corp. v. Google, C.A. No. 10-545 (D. Del)
 - Expert witness on behalf of Frontier Communications Corp.
 - Law firm: Steese, Evans & Frankel
 - Services provided: Provided source code analysis.
 - Technologies: Voice over IP.
12. U.S. Ethernet Innovations LLC v. Acer, Inc. et al., Case No. 10-cv-3724 (N.D. Cal.)
 - Expert witness on behalf of Qualcomm Atheros, Inc., Sigma Designs, Inc., and AT&T Services, Inc.
 - Law firms: Reed Smith LLP, Pillsbury Winthrop Shaw Pittman LLP, and Vinson & Elkins LLP.
 - Services provided: Provided expert reports, was deposed, analyzed source code.
 - Technologies: Intelligent network interfaces, data center networks.
13. Inter Partes Review of U.S. Patents.
 - Expert witness on behalf of Netgear, Inc. and Belkin International, Inc.
 - Law firm: Reed Smith LLP.
 - Services provided: Provided declaration for IPR petition.
 - Technologies: Wireless networks.
14. Inter Partes Review of U.S. Patents.
 - Expert witness on behalf of Arista Networks.
 - Law firm: Fish & Richardson.
 - Services provided: Provided declarations for IPR petitions, deposed 3 times.
 - Technologies: Network routers, data center networks.

15. Invalidity Expert Witness.
 - Expert witness on behalf of Intel Corp.
 - Law firm: Wilmer Cutler Pickering Hale and Dorr LLP.
 - Services provided: Provided expert analysis.
 - Technologies: WiFi networks, bluetooth networks.
16. Parthenon Unified Memory Architecture LLC v. HTC Corporation et al., Case No. 2:14-CV-690-JRG-RSP (E. D. Texas).
 - Expert witness on behalf of HTC Corporation and HTC America, Inc.
 - Law firm: Sidley Austin LLP.
 - Services provided: Retained as an expert.
 - Technologies: Memory architectures.
17. System Architecture Information Technology dba SAI Technology, Inc. v. Qualcomm Inc.
 - Expert witness on behalf of Qualcomm Inc.
 - Law firm: McDermott Will & Emory.
 - Services provided: Retained as an expert, provided reports, analyzed source code, was deposed, and testified at trial.
 - Technologies: WiFi networks.
18. Inter Partes Review of U.S. Patents.
 - Expert witness on behalf of Sandvine Corp.
 - Law firm: Erise IP.
 - Services provided: Provided declarations.
 - Technologies: Network monitoring and measurements.
19. Inter Partes Review of U.S. Patents.
 - Expert witness on behalf of Intel Corp., Cavium Inc., and Dell Inc.
 - Law firms: Weil, Gotshal & Manges LLP, Duane Morris LLP, Alston & Bird LLP.
 - Services provided: Provided declarations for IPR petitions, deposed 3 times.
 - Technologies: Intelligent network interfaces, data center networks, memory management.
20. Expert Witness.
 - Law firm: Covington & Burling.
 - Services provided: Retained as an expert.
 - Technologies: WiFi networks.
21. Qualcomm Inc. v. Apple Inc., Inv. No. 337-TA-1065 (ITC)
 - Expert witness on behalf of Apple Inc. and Intel Corp.
 - Law firm: Wilmer Cutler Pickering Hale and Dorr LLP.
 - Services provided: Provided expert reports/declarations, analyzed source code, was deposed.
 - Technologies: Mobile systems, application and modem processors, inter-processor communications, embedded software, processor and bus power management, memory management.
22. Qualcomm Inc. v. Apple Inc., Civ. Action No. 3:17-cv-01375-JAH-MDD (S.D. Cal.)
 - Expert witness on behalf of Apple Inc. and Intel Corp.
 - Law firm: Wilmer Cutler Pickering Hale and Dorr LLP.
 - Services provided: Provided expert reports/declarations, analyzed source code, deposed 2 times, and testified at trial.
 - Technologies: Mobile systems, application and modem processors, inter-processor communications, embedded software, processor and bus power management, memory management.

23. Inter Partes Review of U.S. Patents.
 - Expert witness on behalf of Panasonic Avionics Corp.
 - Law firm: Haynes and Boone LLP.
 - Services provided: Provided declarations for IPR petitions.
 - Technologies: Network security.
24. Inter Partes Review of U.S. Patents.
 - Expert witness on behalf of Intel Corp.
 - Law firm: Wilmer Cutler Pickering Hale and Dorr LLP.
 - Services provided: Provided declarations for IPR petitions, deposed 4 times.
 - Technologies: Mobile systems, application and modem processors, inter-processor communications, embedded software, power management, memory management.
25. Expert Witness.
 - Law firm: Wilmer Cutler Pickering Hale and Dorr LLP.
 - Services provided: Retained as an expert.
 - Technologies: Inter-processor communications and security.
26. Bell Northern Research LLC v. Coolpad Technologies, Inc. et al., Civil Action No. 3:18-cv-01783-CAB-BLM (S.D. Cal.)
 - Expert witness on behalf of Coolpad Technologies, Inc. and Yulong Computer Communications
 - Law firm: Perkins Coie LLP.
 - Services provided: Provided expert reports, analyzed source code, deposed.
 - Technologies: Processors, wireless communications, power management.
27. Inter Partes Review of U.S. Patents.
 - Expert witness on behalf of Google LLC.
 - Law firm: Paul Hastings LLP.
 - Services provided: Provided declarations for IPR petitions.
 - Technologies: Computer networks, wireless communications.
28. In the Matter of CERTAIN ROUTERS, ACCESS POINTS, CONTROLLERS, NETWORK MANAGEMENT DEVICES, OTHER NETWORKING PRODUCTS, AND HARDWARE AND SOFTWARE COMPONENTS THEREOF, U.S. International Trade Commission Inv. No. 337-TA-1227.
 - Expert witness on behalf of Hewlett Packard Enterprise, Netgear, and CommScope
 - Law firm: Morgan, Lewis & Bockius LLP.
 - Services provided: Provided expert reports, analyzed source code, deposed, and testified at trial.
 - Technologies: WiFi networks.
29. Inter Partes Review of U.S. Patents.
 - Expert witness on behalf of Intel Corp.
 - Law firm: Desmarais LLP.
 - Services provided: Provided declarations for IPR petitions.
 - Technologies: Computer systems.
30. PACT XPP Schweiz AG v. Intel Corp. 1:19-cv-1006-RGA (D. Del.).
 - Expert witness on behalf of Intel Corporation
 - Law firm: Kirkland & Ellis.
 - Services provided: Provided expert report.
 - Technologies: Multiprocessors, reconfigurable processors, power management.

APPENDIX 2: MATERIALS CONSIDERED IN THE PREPARATION OF THIS DECLARATION

Exhibit No.	Description
1001	U.S. Patent No. 9,350,649 to Delangis (“the ’649 Patent”)
1002	Prosecution History for the ’649 Patent
1004	U.S. Patent Application No. 2003/0223450 to Bender (“Bender”)
1005	U.S. Patent No. 6,160,808 to Maurya (“Maurya”)
1006	Mineno (Annotated)
1007	RFC793
1008	RFC1661
1009	RFC1990
1010	EDTX 2:22-cv-287 Competitive Access Systems, Inc. v Oracle Corporation and Oracle America, Inc., Complaint
1011	EDTX 2:22-cv-287 Competitive Access Systems, Inc. v Oracle Corporation and Oracle America, Inc., Exhibit F-649 MPTCP
1012	EDTX-2:22-cv-287 Competitive Access Systems, Inc. v Oracle Corporation and Oracle America, Inc., Exhibit G-649 SD-WAN
1013	Declaration of Gordon MacPherson for Mineno
1014	Declaration of Sandy Ginoza for RFC793
1015	EDTX 2:22-cv-287 Competitive Access Systems, Inc. v Oracle Corporation and Oracle America, Inc. Competitive Access Systems, Inc.’s Disclosure of Asserted Claims and Infringement Contentions
1016	EDTX 2:22-cv-287 Competitive Access Systems, Inc. v Oracle Corporation and Oracle America, Inc., Exhibit D-801 MPTCP
1017	EDTX 2:22-cv-287 Competitive Access Systems, Inc. v Oracle Corporation and Oracle America, Inc., Exhibit E-801 SD-WAN
1018	EDTX 2:22-cv-287 Competitive Access Systems, Inc. v Oracle Corporation and Oracle America, Inc., Exhibit H-908 MPTCP
1019	EDTX 2:22-cv-287 Competitive Access Systems, Inc. v Oracle Corporation and Oracle America, Inc., Exhibit I-908 SD-WAN
1020	Cohen, Managing TCP Connections under persistent HTTP
1021	EDTX 2:22-cv-287 Competitive Access Systems, Inc. v Oracle Corporation and Oracle America, Inc., CAS’s Opposition to Oracle’s Motion to Dismiss

Exhibit No.	Description
1022	Chen, New Approaches to Routing for Large-Scale Data Networks
1023	GitHub page
1024	Gustafsson, A Literature Survey on Traffic Dispersion
1025	IEEE Xplore Article – Multiple Paths Protocol for a Cluster Type Network (Annotated)
1026	Wireless IP Network Standard
1027	U.S. Patent No. 8,370,507 to Cain (“Cain”)

APPENDIX 3: CHALLENGED CLAIMS

[1pre]. A method for establishing and sending data during a multipath connection comprising:

[1a] a data-Source device receiving multipath information about each of a plurality of network-edge devices that is capable of joining a multipath connection with a data-requesting device,

[1b] each of the network-edge devices being connected to one or more networks,

[1c] and each of the network-edge devices being connected to the data-requesting device over a different connection: and

[1d] the data-source device performing at least one of the (i) sending different groups of data packets to different network-edge devices that are participating in the multipath connection, more of the network-edge devices that are participating in the multipath connection, and (iii) sending combinations of different streams and different groups of data packets to one or more of the network-edge devices that are participating in the multipath connection.

2. The method of claim 1, wherein the multipath information includes a connection ID that is generated by one of the data-source device, one of the

plurality of network-edge devices, the data-requesting device and a network device.

3. The method of claim 1, further comprising:

the data-Source device receiving second multipath information about a second network-edge device that is capable of joining the multipath connection; and the data-source device joining the second network-edge device to the multipath connection.

4. The method of claim 1, further comprising:

the data-source device receiving multipath information about which network-edge devices are to be removed from the multipath connection; and the data-source device removing the network-edge devices from being associated with the multipath connection.

5. The method of claim 1, further comprising at least one of the data-source device, the network-edge devices and the data-requesting device determining which of the network-edge devices are to be removed from the multipath connection.

6. The method of claim 1, wherein the data-source device is one of a multipath link aggregator, a network relay, a multipath link server, a server, a

network Switch, a network router and a device capable of providing multipath services.

[7pre]. A method for establishing and sending data during a multipath connection, comprising:

[7a] the data-source device receiving multipath connection requests from a plurality of data-requesting devices, each of the multipath connection requests involving a particular multipath connection to a data-requesting device;

[7b] the data-source device associating a connection ID to each multipath connection request;

[7c] the data-source device using multipath information about each of a plurality of network-edge devices that is capable of joining in a particular multipath connection to a particular data-requesting device, the particular multipath connection having a particular connection ID;

[7d] the data-Source device notifying each of the network-edge devices that is participating in the particular multipath connection associated with the particular connection ID; and

[7e] the data-source device sending data associated with the particular connection ID to the network-edge devices participating in the particular multipath connection associated with the particular connection ID.

8. The method of claim 7, further comprising: the data-source device sending different groups of data packets to different network-edge devices that are participating in the particular multipath connection associated with the particular connection ID.

9. The method of claim 7, further comprising:

the data-source device receiving identification of the network-edge devices to be removed from the particular multipath connection; and

the data-source device removing the identified network-edge devices from the particular multipath connection.

10. The method of claim 7, further comprising the data-source device notifying the data-requesting device when all data has been sent over the particular multilink connection.

11. The method of claim 7, wherein the data-source device is one of a multipath link aggregator, a network relay, a multipath link server, a server, a network switch, a network router and a device capable of providing multipath services.

12. The method of claim 7, wherein the connection ID is generated by one of the data-source device, one of the plurality of network-edges, the data-requesting device, and a network device.

[21pre]. A method for creating and using a multipath connection, comprising the steps of:

[21a] a data-requesting device sending first multipath control information to or through a first network-edge device, the first multipath control information indicates that the first network-edge device will join the multipath connection, the data-requesting device being connected over a first connection to the first network-edge device;

[21b] the data-requesting device sending a second multipath control information to or through a second network-edge device, the second multipath control information indicates that the second network-edge device will join the multipath connection, the data-requesting device being connected over a second connection to the second network-edge device, the data-requesting device capable of receiving different data over the first connection and the second connection at the same time; and

[21c] after the first network-edge device joins the multipath connection, the data-requesting device receiving first data over the first connection from the first network-edge device; and

[21d] after the second network-edge device joins the multipath connection, receiving second data at the data-requesting device over the second path from the second network-edge device.

22. The method as in claim 21, wherein the first network-edge device is connected to a network, and wherein the second network-edge device is connected to the network or a different network.

APPENDIX 4: UNDERSTANDING OF THE LAW

I have applied the following legal principles provided to me by counsel in arriving at the opinions set forth in this report.

Legal Standard for Prior Art

I am not an attorney. I have been informed by attorneys of the relevant legal principles and have applied them to arrive at the opinions set forth in this declaration.

I understand that the petitioner for inter partes review may request the cancelation of one or more claims of a patent based on grounds available under 35 U.S.C. § 102 and 35 U.S.C. § 103 using prior art that consists of patents and printed publications.

Anticipation and Prior Art

I understand that § 102 specifies when a challenged claim is invalid for lacking novelty over the prior art, and that this concept is also known as “anticipation.” I understand that a prior art reference anticipates a challenged claim, and thus renders it invalid by anticipation, if all elements of the challenged claim are disclosed in the prior art reference. I understand the disclosure in the prior art reference can be either explicit or inherent, meaning it is necessarily present or implied. I understand that the prior art reference does not have to use the same words as the challenged claim, but all of the requirements of the claim must

be disclosed so that a person of ordinary skill in the art could make and use the claimed subject-matter.

In addition, I understand that § 102 also defines what is available for use as a prior art reference to a challenged claim.

Under § 102(a), a challenged claim is anticipated if it was patented or described in a printed publication in the United States or a foreign country before the challenged claim's date of invention.

Under § 102(b), a challenged claim is anticipated if it was patented or described in a printed publication in the United States or a foreign country more than one year prior to the challenged patent's filing date.

Under § 102(e), a challenged claim is anticipated if it was described in a published patent application that was filed by another in the United States before the challenged claim's date of invention, or was described in a patent granted to another that was filed in the United States before the challenged claim's date of invention.

I understand that a challenged claim's date of invention is presumed to be the challenged patent's filing date. I also understand that the patent owner may establish an earlier invention date and "swear behind" prior art defined by § 102(a) or § 102(e) by proving (with corroborated evidence) the actual date on which the

named inventors conceived of the subject matter of the challenged claim and proving that the inventors were diligent in reducing the subject matter to practice.

I understand that the filing date of a patent is generally the filing date of the application filed in the United States that issued as the patent. However, I understand that a patent may be granted an earlier effective filing date if the patent owner properly claimed priority to an earlier patent application.

I understand that when a challenged claim covers several structures, either generically or as alternatives, the claim is deemed anticipated if any of the structures within the scope of the claim is found in the prior art reference.

I understand that when a challenged claim requires selection of an element from a list of alternatives, the prior art teaches the element if one of the alternatives is taught by the prior art.

Legal Standard for Obviousness

I understand that even if a challenged claim is not anticipated, it is still invalid if the differences between the claimed subject matter and the prior art are such that the claimed subject matter would have been obvious to a person of ordinary skill in the pertinent art at the time the alleged invention.

I understand that obviousness must be determined with respect to the challenged claim as a whole.

I understand that one cannot rely on hindsight in deciding whether a claim is obvious.

I also understand that an obviousness analysis includes the consideration of factors such as (1) the scope and content of the prior art, (2) the differences between the prior art and the challenged claim, (3) the level of ordinary skill in the pertinent art, and (4) “secondary” or “objective” evidence of non-obviousness.

Secondary or objective evidence of non-obviousness includes evidence of: (1) a long felt but unmet need in the prior art that was satisfied by the claimed invention; (2) commercial success or the lack of commercial success of the claimed invention; (3) unexpected results achieved by the claimed invention; (4) praise of the claimed invention by others skilled in the art; (5) taking of licenses under the patent by others; (6) deliberate copying of the claimed invention; and (7) contemporaneous and independent invention by others. However, I understand that there must be a relationship between any secondary evidence of non-obviousness and the claimed invention.

I understand that a challenged claim can be invalid for obviousness over a combination of prior art references if a reason existed (at the time of the alleged invention) that would have prompted a person of ordinary skill in the art to combine elements of the prior art in the manner required by the challenged claim. I understand that this requirement is also referred to as a “motivation to combine,”

“suggestion to combine,” or “reason to combine,” and that there are several rationales that meet this requirement.

I understand that the prior art references themselves may provide a motivation to combine, but other times simple common sense can link two or more prior art references. I further understand that obviousness analysis recognizes that market demand, rather than scientific literature, often drives innovation, and that a motivation to combine references may come from market forces.

I understand obviousness to include, for instance, scenarios where known techniques are simply applied to other devices, systems, or processes to improve them in an expected or known way. I also understand that practical and common-sense considerations should be applied in a proper obviousness analysis. For instance, familiar items may have obvious uses beyond their primary purposes.

I understand that the combination of familiar elements according to known methods is obvious when it yields predictable results. For instance, obviousness bars patentability of a predictable variation of a technique even if the technique originated in another field of endeavor. This is because design incentives and other market forces can prompt variations of it, and predictable variations are not the product of innovation, but rather ordinary skill and common sense.

I understand that a particular combination may be obvious if it was obvious to try the combination. For example, when there is a design need or market

pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. This would result in something obvious because the result is the product not of innovation but of ordinary skill and common sense. However, I understand that it may not be obvious to try a combination when it involves unpredictable technologies.

It is further my understanding that a proper obviousness analysis focuses on what was known or obvious to a person of ordinary skill in the art, not just the patentee. Accordingly, I understand that any need or problem known in the field of endeavor at the time of invention and addressed by the patent can provide a reason for combining the elements in the manner claimed.

It is my understanding that the Manual of Patent Examining Procedure §2143 sets forth the following as exemplary rationales that 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 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;
- 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.

A person of ordinary skill in the art looking to overcome a problem will often use the teachings of multiple publications together like pieces of a puzzle, even though the prior art does not necessarily fit perfectly together. Therefore, I understand that references for obviousness need not fit perfectly together like puzzle pieces. Instead, I understand that obviousness analysis takes into account inferences, creative steps, common sense, and practical logic and applications that a person of ordinary skill in the art would employ under the circumstances.

I understand that a claim can be obvious in light of a single reference, if the elements of the challenged claim that are not explicitly or inherently disclosed in the reference can be supplied by the common sense of one of skill in the art.

I understand that obviousness also bars the patentability of applying known or obvious design choices to the prior art. One cannot patent merely substituting one prior art element for another if the substitution can be made with predictable results. Likewise, combining prior art techniques that are interoperable with respect to one another is generally obvious and not patentable.

In order for a claim to be found invalid based upon a modification or combination of the prior art, there must be reasonable expectation that a person of ordinary skill would have successfully modified or combined the prior art to arrive at the claimed arrangement. This does not mean that it must be certain that a person of ordinary skill would have been successful – the law only requires that the person of ordinary skill in the art would have perceived a reasonable expectation of success in modifying or combining the prior art to arrive at the claimed invention.

In sum, my understanding is that obviousness invalidates claims that merely recite combinations of, or obvious variations of, prior art teachings using understanding and knowledge of one of skill in the art at the time and motivated by the general problem facing the inventor at the time. Under this analysis, the prior art references themselves, or any need or problem known in the field of endeavor

at the time of the invention, can provide a reason for combining the elements of or attempting obvious variations on prior art references in the claimed manner.

Legal Standard for Claim Construction

I understand that before any invalidity analysis can be properly performed, the scope and meaning of the challenged claims must be determined by claim construction.

I understand that a patent may include two types of claims, independent claims and dependent claims. I understand that an independent claim stands alone and includes only the limitations it recites. I understand that a dependent claim depends from an independent claim or another dependent claim. I understand that a dependent claim includes all the limitations that it recites in addition to the limitations recited in the claim (or claims) from which it depends.

In comparing the challenged claims to the prior art, I have carefully considered the patent and its file history in light of the understanding of a person of skill at the time of the alleged invention.

I understand that to determine how a person of ordinary skill would have understood a claim term, one should look to sources available at the time of the alleged invention that show what a person of skill in the art would have understood disputed claim language to mean. It is my understanding that this may include what is called “intrinsic” evidence as well as “extrinsic” evidence.

I understand that, in construing a claim term, one should primarily rely on intrinsic patent evidence, which includes the words of the claims themselves, the remainder of the patent specification, and the prosecution history. I understand that extrinsic evidence, which is evidence external to the patent and the prosecution history, may also be useful in interpreting patent claims when the intrinsic evidence itself is insufficient. I understand that extrinsic evidence may include principles, concepts, terms, and other resources available to those of skill in the art at the time of the invention.

I understand that words or terms should be given their ordinary and accepted meaning unless it appears that the inventors were using them to mean something else or something more specific. I understand that to determine whether a term has special meaning, the claims, the patent specification, and the prosecution history are particularly important, and may show that the inventor gave a term a particular definition or intentionally disclaimed, disavowed, or surrendered claim scope.

I understand that the claims of a patent define the scope of the rights conferred by the patent. I understand that because the claims point out and distinctly claim the subject matter which the inventors regard as their invention, claim construction analysis must begin with and is focused on the claim language itself. I understand that the context of the term within the claim as well as other claims of the patent can inform the meaning of a claim term. For example, because

claim terms are normally used consistently throughout the patent, how a term is used in one claim can often inform the meaning of the same term in other claims. Differences among claims or claim terms can also be a useful guide in understanding the meaning of particular claim terms.

I understand that a claim term should be construed not only in the context of the particular claim in which the disputed term appears, but in the context of the entire patent, including the entire specification. I understand that because the specification is a primary basis for construing the claims, a correct construction must align with the specification.

I understand that the prosecution history of the patent as well as art incorporated by reference or otherwise cited during the prosecution history are also highly relevant in construing claim terms. For instance, art cited by or incorporated by reference may indicate how the inventor and others of skill in the art at the time of the invention understood certain terms and concepts. Additionally, the prosecution history may show that the inventors disclaimed or disavowed claim scope, or further explained the meaning of a claim term.

With regard to extrinsic evidence, I understand that all evidence external to the patent and prosecution history, including expert and inventor testimony, dictionaries, and learned treatises, can also be considered. For example, technical dictionaries may indicate how one of skill in the art used or understood the claim

terms. However, I understand that extrinsic evidence is considered to be less reliable than intrinsic evidence, and for that reason is generally given less weight than intrinsic evidence.

I understand that in general, a term or phrase found in the introductory words or preamble of the claim, should be construed as a limitation if it recites essential structure or steps, or is necessary to give meaning to the claim. For instance, I understand preamble language may limit claim scope: (i) if dependence on a preamble phrase for antecedent basis indicates a reliance on both the preamble and claim body to define the claimed invention; (ii) if reference to the preamble is necessary to understand limitations or terms in the claim body; or (iii) if the preamble recites additional structure or steps that the specification identifies as important.

On the other hand, I understand that a preamble term or phrase is not limiting where a challenged claim defines a structurally complete invention in the claim body and uses the preamble only to state a purpose or intended use for the invention. I understand that to make this determination, one should review the entire patent to gain an understanding of what the inventors claim they invented and intended to encompass in the claims.

I understand that 35 U.S.C. § 112 ¶ 6 created an exception to the general rule of claim construction called a “means plus function” limitation. These types of

terms and limitations should be interpreted to cover only the corresponding structure described in the specification, and equivalents thereof. I also understand that a limitation is presumed to be a means plus function limitation if (a) the claim limitation uses the phrase “means for”; (b) the “means for” is modified by functional language; and (c) the phrase “means for” is not modified by sufficient structure for achieving the specified function.

I understand that a structure is considered structurally equivalent to the corresponding structure identified in the specification only if the differences between them are insubstantial. For instance, if the structure performs the same function in substantially the same way to achieve substantially the same result. I further understand that a structural equivalent must have been available at the time of the issuance of the claim.