

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

CISCO SYSTEMS, INC.
Petitioner,

v.

OPTIMNET LLC,
Patent Owner,

U.S. Patent No. 10,020,961
Issue Date: July 10, 2018
Title: METHOD AND APPARATUS FOR NETWORK
VIRTUALIZATION

Inter Partes Review No.: IPR2026-00322

**DECLARATION OF DR. A.L. NARASIMHA REDDY IN SUPPORT OF
PETITION FOR INTER PARTES REVIEW OF
U.S. PATENT NO. 10,020,961**

Mail Stop “PATENT BOARD”

Patent Trial and Appeal Board
U.S. Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

TABLE OF CONTENTS

I.	INTRODUCTION	3
II.	MY BACKGROUND AND QUALIFICATIONS	5
III.	RELEVANT LEGAL STANDARDS	9
IV.	OVERVIEW	12
V.	BACKGROUND	15
A.	Summary of the '961 Patent.....	15
B.	Claim Construction	24
C.	Priority Date of Claims	25
D.	Person of Ordinary Skill in the Art (POSITA)	25
VI.	State of the Art.....	26
1.	Overlay-Based Network Virtualization Using Tunnels Was Well Known	27
2.	Tunnel Managers That Collect Tunnel End Point Information and Establish Tunnel Connections Were Well Known	29
3.	Tunnel Packet Processing—including Encapsulation and Decapsulation—Was Well Known.....	31
4.	Virtual Network Identifiers (VNIDs) for Identifying and Isolating Virtual Networks Were Well Known	33
5.	Virtual Switching Instances (VSIs) Corresponding to Virtual Network Identifiers That Perform L2 Switching Were Well Known.....	34
6.	Modified Flooding Rules Based on Input Port Type—including Split-Horizon Forwarding—Were Well Known.....	36
VII.	CHALLENGES	38
A.	Identification of Challenges	38

B.	Prior Art.....	39
C.	Ground 1: Claims 1-13	39
1.	The Ye Reference.....	39
2.	Detailed Application of Ye to the Challenged Claims	45
	Claim 1	45
	Claim 2	56
	Claim 3	60
	Claim 4	63
	Claim 5	65
	Claim 6	68
	Claim 7	70
	Claim 8	72
	Claim 9	73
	Claim 10	74
	Claim 11	75
	Claim 12	75
	Claim 13	76
	CONCLUSION AND DECLARATION.....	1
	APPENDIX A – MATERIALS RELIED UPON.....	A
	APPENDIX B – CURRICULUM VITAE	B

I, Dr. A.L. Narasimha Reddy, declare as follows:

I. INTRODUCTION

1. I am over the age of eighteen (18) and otherwise competent to make this declaration.

2. I have been retained by Cisco Systems, Inc. (Petitioner) as an independent expert consultant in this proceeding before the United States Patent and Trademark Office. Although I am being compensated at my rate of \$650 per hour for the time I spend on this matter, no part of my compensation depends on the outcome of this proceeding, and I have no other interest in this proceeding. To the best of my knowledge, I have no financial interest in Client.

3. This Petition for *inter partes* review involves U.S. Patent No. 10,020,961 (the “’961 Patent”) (EX-1001), titled “METHOD AND APPARATUS FOR NETWORK VIRTUALIZATION” and lists Sung Hyuk Byun as the inventor.

4. For the purposes of this *inter partes* review as discussed later, I have been instructed to assume that the effective filing date of the Claims of the ’961 Patent challenged by the Petitioner in this *inter partes* review is no earlier than December 27, 2013, the filing date of Korean Patent Application 10-2013-0165299.

5. I understand that according to USPTO records, the '961 Patent is currently assigned to Electronics and Telecommunications Research Institute ("ETRI"). I also understand that OptimNet LLC ("OptimNet" or "Patent Owner") claims to be an assignee of the '961 Patent but has not registered this assignment at the USPTO.

6. The '961 Patent is directed to connecting multiple virtual networks together to form a single, larger virtual network by adding intermediate tunnel endpoints at the threshold of each of the smaller virtual networks. The '961 Patent describes that overlay-based virtual networks require tunnels to connect all endpoints in the network, and that as the number of endpoints grows, the number of required tunnels becomes difficult to manage. EX-1001, 1:33-56. The '961 Patent's described solution divides the network into smaller domains and connects them through a relay device called a "domain tunnel end point." I am familiar with the technology described in the '961 Patent as of the earliest possible priority date of December 27, 2013.

7. In preparing this Declaration, I have reviewed the '961 Patent (EX-1001), the file history of the '961 Patent (EX-1002), and each of the documents cited herein, and I have considered these documents in light of the general knowledge in the art as of December 27, 2013. In formulating my opinions, I have relied upon my experience in the relevant art. I have also considered the viewpoint

of a person of ordinary skill in the art (“POSITA”) in the relevant field, as of January 28, 2015, who I describe below (§IV.D). For convenience, the materials I considered in forming my opinion are listed in Appendix A.

8. The opinions and comments formulated during this assessment are based on observations and information available at the time of the investigation. The findings presented herein are made to a reasonable degree of scientific certainty. I have made every effort to accurately and completely investigate all areas of concern identified during our investigation.

9. I have been asked to provide my technical expertise, analysis, insights, and opinions regarding the ’961 Patent and relevant references that form the basis of the arguments set forth in the accompanying Petition for *inter partes* review of the ’961 Patent. As described in detail below, I offer the following opinions in this Declaration:

- A POSITA would have found Claims 1-13 of the ’961 Patent obvious over Ye.

II. MY BACKGROUND AND QUALIFICATIONS

10. I am currently the Truchard Foundation Chair Professor of Electrical and Computer Engineering at Texas A&M University in College Station, Texas. I have over 25 years of experience in a wide variety of technologies and industries

relating to data communications, storage systems, and distributed systems, including packet-switched network communications.

11. My academic credentials include a Bachelor of Technology degree in Electronics and Electrical Communications Engineering from the Indian Institute of Technology, Kharagpur, India, in August 1985. I then received a Master of Science and a Ph.D. in Computer Engineering from the University of Illinois at Urbana-Champaign in May 1987 and August 1990, respectively.

12. I have worked for over 25 years in the field of Electrical Engineering. My primary focus and research interest has been on computer networks, storage systems, multimedia systems, and computer architecture. I have authored and co-authored over a hundred technical papers and several book chapters related to several of these interests, including on such topics as multipath routing, route control, high-speed networks, network congestion, packet management, quality of service regulation, network security, network modeling, differentiated services, storage system enhancements, caching strategies, and multimedia system enhancements to name a few examples. I am listed as an inventor on six U.S. patents in the field of multi-node communication networks and computer systems.

13. My employment history following my graduation from the University of Illinois at Urbana-Champaign began at the IBM Almaden Research Center in

San Jose, California in 1990. At IBM, I worked on projects related to disk arrays, multiprocessor communication, hierarchical storage systems and video servers.

14. In 1995, I joined the faculty of the department of Electrical Engineering at Texas A&M University initially as an Associate Professor and was later promoted to full, tenured Professor. At Texas A&M, during 2016-21, I was the Associate Agency Director for Strategic Initiatives and Centers for the Texas A&M Engineering Experiment Station (TEES), which engages in engineering and technology-oriented research and educational collaborations. Further, I also served as Associate Dean for Research in the College of Engineering during 2016-21.

15. At Texas A&M, I have taught dozens of courses related to computer networking and communications, as well as computer architecture, multimedia systems and networks, topics in networking security, multimedia storage and delivery, as well as networking for multimedia applications. I have also served on various committees for the benefit of the scientific community and the Texas A&M University community.

16. I am a member of a number of professional societies, including the Institute of Electrical and Electronic Engineers (IEEE), where I have been elevated to an IEEE Fellow, and the Association for Computing Machinery (ACM). I have been responsible for chairing or co-chairing numerous conferences and programs, as well as presenting research at major IEEE and ACM conferences. For example,

I served as program co-chair for the 2008 5th International Conference on Broadband Communications, Networks and Systems, panels co-chair for the 2008 3rd International Conference on Communication Systems Software & Middleware, and panel chair of the IEEE Conference of High Performance Computer Architecture.

17. My presentations include a Keynote speech at International Conference on Information Technology-New Generations in 2013, a Keynote speech at IEEE International Symposium on Computers and Communications 2010, and several invited talks including at Georgia Tech (2013), COMSNETS Conference (2013), Int. Conf. on Networking and Communications (2012), Samsung (2011), Korea University (2011), Aijou University (2011), Catedra Series at University of Carlos III, Madrid (2009), Thomson Research, Paris (2009), Telefonica Research, Barcelona (2009), and a Distinguished Seminar at IBM Austin Research Lab (2008).

18. I have received multiple awards in the field of networks and computer architecture. I received the NSF Career Award from 1996-2000. I received an outstanding professor award by the IEEE student branch at Texas A&M during 1997-1998, an outstanding faculty award by the department of Electrical and Computer Engineering during 2003-2004, a Distinguished Achievement award for teaching from the former students' association of Texas A&M University, and a

citation “for one of the most influential papers” from the 1st ACM Multimedia conference.

19. My curriculum vitae, which includes a more detailed summary of my background and experience, is attached as Appendix B.

III. RELEVANT LEGAL STANDARDS

20. I am not a lawyer and do not provide any legal opinions in this Declaration. Although I am not a lawyer, I have been informed and understand that certain legal standards are to be applied by technical experts in forming opinions regarding the meaning and validity of patent claims. I have been asked to provide my opinions regarding whether the claims of the '961 Patent are anticipated or would have been obvious to a person having ordinary skill in the art at the time of the alleged invention, in light of the prior art.

21. I have been informed and understand that a patent claim is not patentable under 35 U.S.C. § 103 if the differences between the patent claim and the prior art are such that the claimed subject matter as a whole would have been obvious at the time the claimed invention was made to a person having ordinary skill in the relevant art. Obviousness, as I have been informed and understand, is based on the scope and content of the prior art, the differences between the prior art and the claim, the level of ordinary skill in the art, and, to the extent that they exist, certain objective indicia of non-obviousness.

22. I understand that objective indicia can be important evidence regarding whether a patent is obvious or nonobvious, if it has an appropriate nexus to the claimed invention, i.e., is a result of the merits of a claimed invention (rather than the result of design needs or market-pressure advertising or similar activities). Such indicia include: commercial success of products covered by the patent claims; a long-felt need for the invention; failed attempts by others to make the invention; copying of the invention by others in the field; unexpected results achieved by the invention as compared to the closest prior art; praise of the invention by the infringer or others in the field; the taking of licenses under the patent by others; expressions of surprise by experts and those skilled in the art at the making of the invention; and the patentee proceeded contrary to the accepted wisdom of the prior art.

23. I have been informed that whether there are any relevant differences between the prior art and the claimed invention is to be analyzed from the view of a person of ordinary skill in the relevant art at the time of the invention. As such, my opinions below as to a person of ordinary skill in the art are as of the time of the invention, even if not expressly stated as such; for example, even if stated in the present tense.

24. In analyzing the relevance of the differences between the claimed invention and the prior art, I have been informed that I must consider the impact, if

any, of such differences on the obviousness or non-obviousness of the invention as a whole, not merely some portion of it. The person of ordinary skill faced with a problem is able to apply his or her experience and ability to solve the problem and also look to any available prior art to help solve the problem.

25. I have been informed that a precise teaching in the prior art directed to the subject matter of the claimed invention is not needed. I have been informed that one may take into account the inferences and creative steps that a person of ordinary skill in the art would have employed in reviewing the prior art at the time of the invention. For example, if the claimed invention combined elements known in the prior art and the combination yielded results that were predictable to a person of ordinary skill in the art at the time of the invention, then this evidence would make it more likely that the claim was obvious. On the other hand, if the combination of known elements yielded unexpected or unpredictable results, or if the prior art teaches away from combining the known elements, then this evidence would make it more likely that the claim that successfully combined those elements was not obvious.

26. I have been informed and understand that there are recognized, exemplary, rationales for combining or modifying references to show obviousness of claimed subject matter. Some of the rationales include the following:
combining prior art elements according to known methods to yield predictable

results; simple substitution of one known element for another to yield predictable results; use of a known technique to improve a similar device (method or product) 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; and 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 teachings to arrive at the claimed invention.

IV. OVERVIEW

27. Claims 1-13 of the '961 Patent (the “Challenged Claims”) are unpatentable as obvious over the prior art. The claims are directed to prior art systems and combinations of conventional components that perform conventional functions regarding network routing and network services, all of which were well-known in the art of network communications.

28. Specifically, the Challenged Claims are directed to an apparatus (Claims 1-6 and 13) and method (Claims 7-12) for connecting multiple virtual networks together to form a single, larger virtual network by adding intermediate

tunnel endpoints at the threshold of each of the smaller virtual networks that communicate with a “domain tunnel end point.” The ’961 Patent addresses a Layer 2 (L2) network realized on top of a Layer 3 (L3) network. Tunneling is performed by attaching a L3 IP header to a L2 Ethernet frame being transmitted between virtual machines. During prosecution, the Applicant argued that it was the feature of a domain tunnel endpoint configured to add a L3 tunnel header to the packet for which the L2 switching has been performed to generate a new tunnel packet that distinguished over the prior art.

29. However, the same type of architecture was known in the art for connecting multiple L2 networks located at different physical sites so that they operate collectively as if they were a single L2 network. U.S. Patent Application Publication No. 2015/0146727 (“Ye”) (EX-1004) is directed to methods and devices for forwarding packets in a Virtual Private LAN Service (VPLS) network—a technology for connecting geographically dispersed Layer 2 (L2) networks into a single, unified virtual LAN over a service provider’s backbone network. EX-1004, [0001], [0026]. Like the ’961 Patent, Ye discloses a domain tunnel endpoint configured to add a L3 tunnel header to the packet for which the L2 switching has been performed to generate a new tunnel packet.

30. Thus, Ye demonstrates that the purported invention recited in the Challenged Claims was well-known at the time of the '961 Patent and is unpatentable as set forth below in Ground 1.

31. As detailed below, I explain how each limitation recited in the Challenged Claims are found in the prior art. The Challenged Claims are directed to prior art systems and combinations of conventional components that perform conventional functions regarding network routing and network services, all of which were well-known in the art of network communications.

32. Specifically, Claim 1 (representative of independent Claims 7 and 13) of the '961 Patent recites:

1. A network virtualization apparatus comprising:

a tunnel manager configured to collect tunnel end point information

within a network, and to connect each tunnel end point with a tunnel

based on the tunnel end point information collected;

a tunnel packet end point configured to receive a tunnel packet and

process the packet such that an L2 switching may be performed, and to

transmit the packet to a domain VSI (virtual switching instance);

the domain VSI configured to perform the L2 switching on the processed

packet, and to transmit the packet to a tunnel packet generator; and

the tunnel packet generator configured to add a tunnel header to the packet for which the L2 switching has been performed to generate a new tunnel packet, and to transmit the new tunnel packet.

33. However, a comparison of Ye to the Challenged Claims demonstrates that the system and method recited in the Challenged Claims were well-known prior to the purported invention of the '961 Patent.

34. Ye discloses a Provider Edge (PE) device within a Virtual Private LAN Service (VPLS) overlay network—a technology for connecting geographically dispersed Layer 2 (L2) networks into a single, unified virtual LAN over a service provider's backbone network. EX-1004, [0001], [0026]. A POSITA would find that it would be obvious to modify the PE device disclosed in Ye, including its components and behavior, to meet all the all of the elements of the '961 Patent.

V. BACKGROUND

A. Summary of the '961 Patent

35. The '961 Patent is directed to connecting multiple virtual networks together to form a single, larger virtual network by adding intermediate tunnel endpoints at the threshold of each of the smaller virtual networks. The '961 Patent describes that overlay-based virtual networks require tunnels to connect all endpoints in the network, and that as the number of endpoints grows, the number

of required tunnels becomes difficult to manage. EX-1001, 1:33-56. The '961 Patent's described solution divides the network into smaller domains and connects them through a relay device called a "domain tunnel end point."

36. The '961 Patent explains that overlay-based virtual networks—such as VXLAN (Virtual Extensible Local Area Network) and NVGRE (Network Virtualization using Generic Routing Encapsulation)—are Layer 2 (L2) networks realized on top of a Layer 3 (L3) network. EX-1001, 1:27-44. The layer numbers refer to the Open Systems Interconnection (OSI) model seven layer model of the internet. In these networks, tunneling is performed by attaching an L3 IP header to an L2 Ethernet frame being transmitted between virtual machines. EX-1001, 1:27-32. To create a virtual L2 network, a full-mesh tunnel must be set between all Tunnel End Points (TEPs) in the network. EX-1001, 1:36-40.

37. To set a full-mesh tunnel between N number of TEPs, $N \times (N-1)$ tunnels are needed. EX-1001, 1:40-41. The '961 Patent states that this "restricts expandability, thereby causing a problem of restricting the size of the network (number of network nodes) where a VXLAN or NVGRE may be applied." EX-1001, 1:42-44. As a practical matter, this limitation means that a single network virtualization domain is generally restricted to one data center. EX-1001, 1:50-52. The '961 Patent recognizes a need for virtual networks that span multiple, geographically dispersed data centers, but notes that "it is not easy to provide these

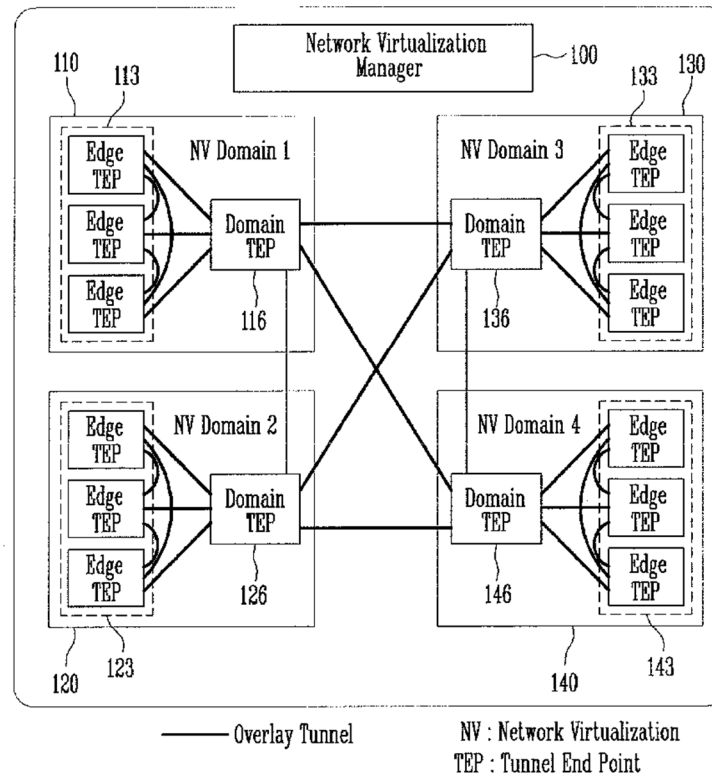
requirements with conventional network virtualization methods since they lack expandability.” EX-1001, 1:52-56.

38. Rather than attempting to scale a single domain, the ’961 Patent proposes dividing the network into multiple smaller network virtualization domains of a manageable size, each connected internally by a full-mesh of tunnels in the conventional way. EX-1001, 2:31-42. To link these domains together, the ’961 Patent introduces a “domain tunnel end point” in each domain. *Id.* The domain tunnel end points of each domain are then connected to one another by L3 tunnels in full-mesh topology. *Id.* In this way, “it is possible to create one large scale network virtualization domain where a plurality of domains are integrated,” while “the number of tunnels necessary in a network virtualization domain may be significantly reduced.” *Id.*

39. As illustrated in FIG. 1, four network virtualization domains (110, 120, 130, and 140) are each populated with edge tunnel end points (113, 123, 133, and 143) and at least one domain tunnel end point (116, 126, 136, and 146). EX-1001, 3:55-65. A network virtualization manager (100) sets and manages the edge and domain tunnel end points for each domain. EX-1001, 3:65-4:1. Within each domain, the edge tunnel end points are connected to one another and to their domain tunnel end point by unicast tunnels in full-mesh topology. EX-1001, 4:19-

28. Between domains, the domain tunnel end points are connected to one another by L3 tunnels, also in full-mesh topology. *Id.*

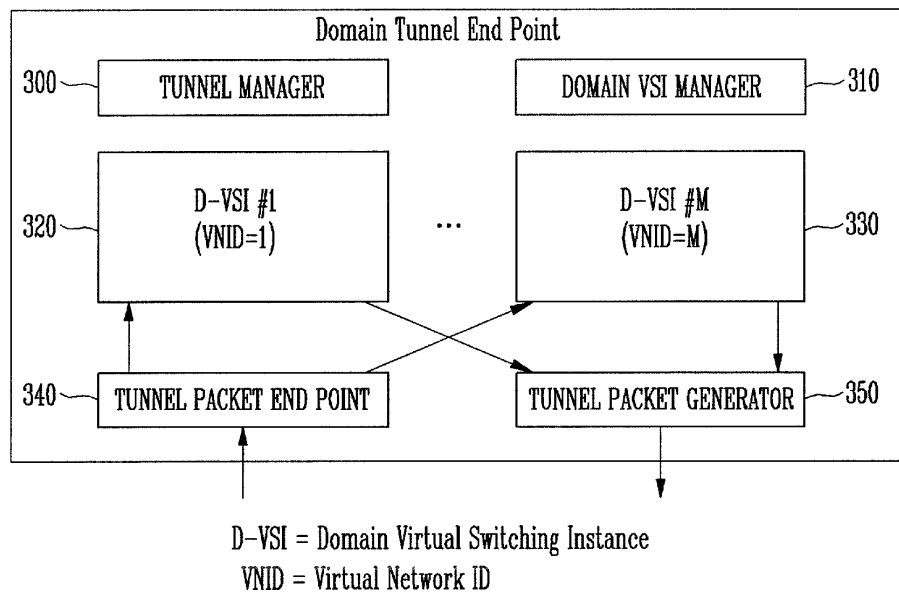
FIG. 1



40. In this arrangement, an edge tunnel end point connects only to its local domain tunnel end point, which in turn relays the packet to the appropriate domain tunnel end point in the destination domain. The '961 Patent describes that a domain tunnel end point is “configured to perform like a tunnel switch which can relay the single-hop tunnel packet from each tunnel end point to any other tunnel end points not reachable by single-hop tunnel.” EX-1001, 5:17-30.

41. The claims of the '961 Patent are directed to the domain tunnel end point and its methods of operation. As shown in FIG. 3, the domain tunnel end point comprises five components: (1) a tunnel manager (300); (2) a domain VSI manager (310); (3) one or more domain VSIs (320, 330); (4) a tunnel packet end point (340); and (5) a tunnel packet generator (350). EX-1001, 5:31-37.

FIG. 3



42. The tunnel manager (300) connects tunnels between all edge tunnel end points and the domain tunnel end point within the same domain, and between the domain tunnel end point and other domain tunnel end points in other domains. EX-1001, 5:39-43. The tunnel manager receives information on all the edge tunnel end points within the domain and information on other domain tunnel end points, which may be transmitted by an operator's setting or a protocol. EX-1001, 5:39-46.

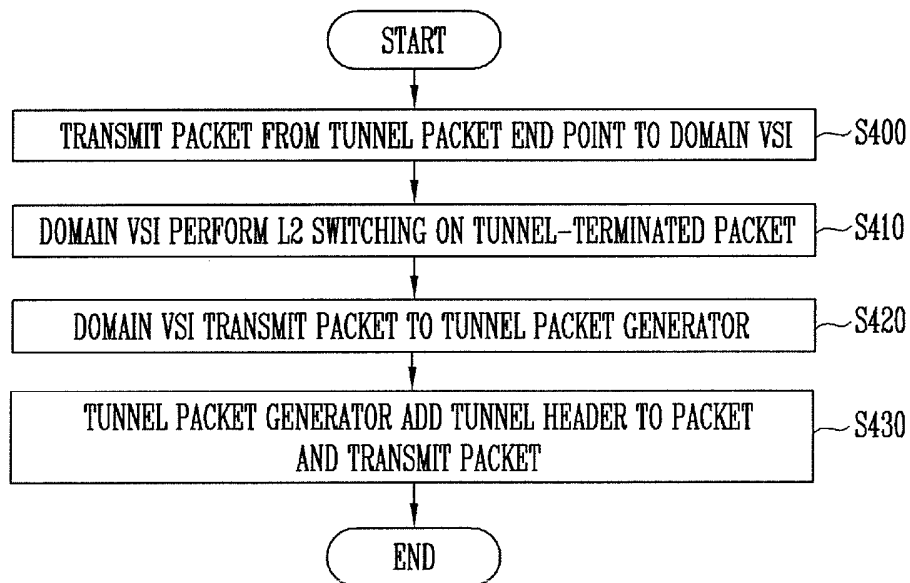
Based on this information, the tunnel manager connects individual tunnels between each edge tunnel end point within the domain and also connects tunnels in full-mesh topology between domain tunnel end points. EX-1001, 5:47-52.

43. The domain VSI manager (310) receives a virtual network identifier (VNID) from “outside” of the domain tunnel endpoint and creates a domain virtual switching instance (VSI) corresponding to that VNID. EX-1001, 6:4-14. A domain VSI (320, 330) exists for each VNID, and therefore each virtual network (VLAN). EX-1001, 6:15. The domain VSI creates edge virtual ports associated with connected edge tunnel end points, and domain virtual ports associated with other domain tunnel end points. EX-1001, 5:59-6:3. The domain VSI performs L2 switching—which may be modified depending on the type of packet it receives—and sends packets out through the virtual port associated with the destination tunnel end point indicated by the received packet. EX-1001, 6:15-20.

44. FIG. 4 illustrates the general packet processing process in the domain tunnel end point. First, when a domain tunnel end point receives an incoming tunnel packet, it removes the tunnel header, obtains the sending tunnel end point’s L3 address and information that identifies the VNID and virtual network associated with the packet from the tunnel header, and transmits the processed packet through the appropriate virtual port to the domain VSI associated with that VNID and virtual network (S400). EX-1001, 6:51-62. Next, the domain VSI performs L2

switching on the packet based on its L2 header (S410). EX-1001, 6:63-7:4. The domain VSI then transmits the switched packet to the tunnel packet generator through an output virtual port (S420). EX-1001, 7:5-11. Finally, the tunnel packet generator (350) adds a new tunnel header—including a VN header based on the associated VNID and an L3 tunnel header with the appropriate source and destination addresses based on the tunnel end points—and transmits the new tunnel packet to the next destination (S430). EX-1001, 7:12-25.

FIG. 4

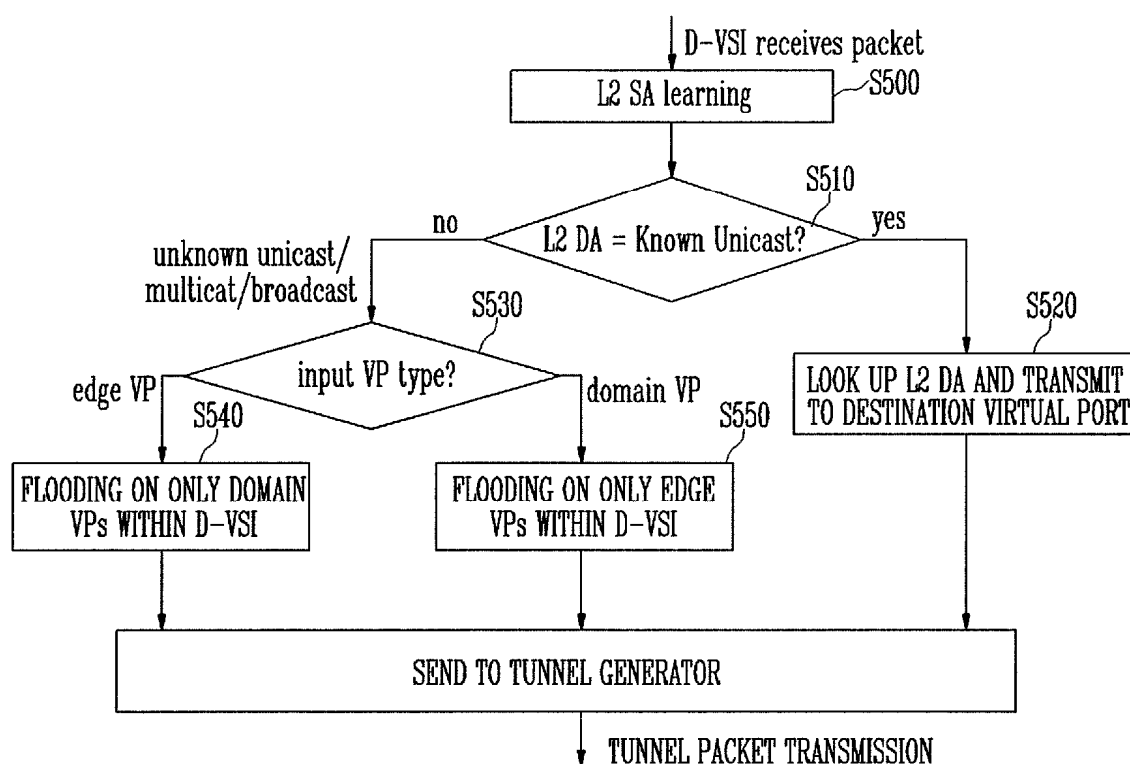


45. In short, the domain tunnel end point operates as a relay. It decapsulates the incoming tunnel packet, performs L2 switching to determine the correct outbound path, and re-encapsulates the packet with a new tunnel header for

transmission to its next destination—whether that destination is an edge tunnel end point within the same domain or a domain tunnel end point in another domain.

46. As noted above, the L2 switching that the domain VSI performs may be modified depending on the type of packet it receives. FIG. 5 illustrates this process in greater detail. If the destination L2 address is a known unicast address, the domain VSI forwards the packet to the corresponding destination virtual port. EX-1001, 7:33-36. If the destination L2 address is unknown, or is a multicast or broadcast address where there are multiple destinations, the domain VSI performs different flooding methods. EX-1001, 7:37-41. Specifically, if the input virtual port is an edge virtual port, the packet is flooded only to domain virtual ports; if the input virtual port is a domain virtual port, the packet is flooded only to edge virtual ports. EX-1001, 7:42-47.

FIG. 5



47. The '961 Patent describes three types of tunnel packets, illustrated in FIGS. 2a, 2b, and 2c. All tunnel packets share a common structure: an original packet (containing an L2 header, an L3 header, and payload) and a tunnel header. EX-1001, 4:35-49. The tunnel header includes a VN header containing a virtual network identifier—identifying the virtual network where the packet belongs to—and an L3 tunnel header with L3 source address (SA) and L3 destination address (DA) fields. EX-1001, 4:15-55. As a packet traverses a domain tunnel end point, its tunnel header is stripped and replaced: for example, a packet arriving from an

edge tunnel end point (FIG. 2b) may be re-encapsulated with a new tunnel header for transmission between domain tunnel end points (FIG. 2C).

48. Independent claim 1 is directed to a network virtualization apparatus that reads on the domain tunnel end point in FIG. 3 and comprises four components: a tunnel manager, a tunnel packet end point, a domain VSI, and a tunnel packet generator. EX-1001, 8:19-34. Independent claim 7 recites an operating method of the same apparatus, covering the steps of collecting tunnel end point information, receiving and processing tunnel packets, performing L2 switching, and generating new tunnel packets. EX-1001, 9:23-38. Independent claim 13 recites a similar apparatus claim but specifies the generation of a “new unicast tunnel packet.” EX-1001, 10:44-59.

49. Dependent claims 2-6 and 8-12 add limitations that deal with the domain VSI’s L2 switching behavior (including the flooding rules explained in the specification), the tunnel packet end point use of the VNID and L3 address, the tunnel packet generator’s construction of the VN header and L3 tunnel header, and the creation of domain VSIs based on received VNIDs. EX-1001, 8:35-9:22; 9:39-10:43.

B. Claim Construction

50. I have been informed and understand that in order to properly evaluate the ’961 Patent, the terms of the claims must first be interpreted.

51. I understand that Petitioner proposes that each claim term in the Challenged Claims be given its plain and ordinary meaning in this proceeding, and that no specific construction of any claim term is required because the prior art relied on in the Petition meets each of the claim terms under any reasonable construction. The plain meaning is the ordinary and customary meaning to those skilled in the art.

C. Priority Date of Claims

52. For the purpose of this *inter partes* review, I have been instructed to assume that the effective filing date of each of the Challenged Claims is no earlier than the filing date of Korean Patent Application 10-2013-0165299. Solely for the purpose of this IPR, I have been instructed to assume the priority date is December 27, 2013.

D. Person of Ordinary Skill in the Art (POSITA)

53. I am familiar with the knowledge and capabilities of one of ordinary skill in the art. Unless otherwise stated, my testimony below refers to the knowledge of one of ordinary skill in the art as of December 27, 2013, the earliest possible effective filing date of the '961 Patent.

54. I have been informed and understand that a Person of Ordinary Skill in the Art ("POSITA") is a hypothetical person who is presumed to be aware of all

pertinent prior art, thinks along conventional wisdom in the art, and is a person of ordinary creativity.

55. With respect to the '961 Patent, a POSITA in December 2013 would have been someone knowledgeable and familiar with communications networks including the use of virtual networks. A POSITA would have gained knowledge of these concepts through a mixture of training and work experience, such as by having at least a bachelor's degree in electrical engineering or computer science, or related field, and at least two to three years of training or additional work experience in the area of communications networks, or a related field. Additional hands-on and design experience could compensate for less formal education, and vice versa.

56. A POSITA may work as part of a multi-disciplinary team and draw upon not only his or her own skills, but also take advantage of certain specialized skills of others on the team to solve a given problem..

VI. State of the Art

57. The following section describes the state of the art for network communications as of the priority date of the '961 Patent. The prior art references, and the discussions of what was known to a POSITA, provide the factual support for the general description of the state of the art at the time of the invention, and provide additional motivation to modify or combine the references.

1. Overlay-Based Network Virtualization Using Tunnels Was Well Known

58. By December 2013, the practice of overlaying virtual networks onto network infrastructure using tunnels was well known. Such overlay networks were known under various names, including VXLAN (Virtual eXtensible Local Area Network), VPLS (Virtual Private LAN Service), NVGRE (Network Virtualization using Generic Routing Encapsulation), and Overlay Transport Virtualization (OTV). The '961 Patent itself acknowledges the state of the art in its Description of Related Art:

Recent trends of IT (information technology) include server virtualization, VM (virtual machine)-based computing, concentration of IT resources on data center, and network virtualization. Network virtualization is a technology where a plurality of virtual networks may be created on a physical network when necessary and can be deleted. This technology may be utilized to configure a network between virtual machines. A generally used overlay-based virtual network is a L2 network that may be realized on top of a L3 network in an overlay method. In the overlay-based virtual network, tunneling may be performed that is attaching a L3 IP (internet protocol) header to a L2 Ethernet frame being transmitted between virtual machines.

Most representative overlay-based network virtualization includes a VXLAN (virtual extensible local area network) and NVGRE (network virtualization using generic routing encapsulation). In a VXLAN or NVGRE, a virtual L2 network may be created by connecting a tunnel

between servers. For this purpose, a full-mesh tunnel must be set between TEPs (Tunnel End Points) being realized in a server or gateway.

EX-1001, 1:20-40.

59. A tunnel is a virtual point-to-point connection between two network devices, created by encapsulating a packet of one protocol type inside a packet of another protocol type—allowing, for example, L2 Ethernet frames to be transported across a L3 IP network. Tunnels in overlay networks have end points at either side of a tunnel. Such tunnel end points were variously referred to as VTEPs (VXLAN Tunnel End Points), NVEs (Network Virtualization Edges), PE devices (Provider Edge devices), and edge devices. For example, Kamath explains that overlay networks such as VXLAN “connect geographically separated Layer-2 (L2) networks using tunnels” that are “L2 over Layer-3 (L3) tunnels,” and that “L2 packets originated by a virtual machine (VM) in a VXLAN and destined to another VM or group of VMs in same VXLAN in another physical location are carried over L3 tunnels.” EX-1008, [0003]. Yong similarly describes that:

[a]n overlay virtual network may provide Layer 2 (L2) or Layer 3 (L3) services to the connected TSs via network virtualization edges (NVEs), where NVEs may be implemented as part of a virtual switch within a hypervisor, and/or physical switch or router. An NVE encapsulates ingress tenant traffic and sends the encapsulated traffic over a tunnel across an underlying network toward an egress NVE. ...

There are a number of encapsulation protocols available in the industry today, such as virtual eXtensible Local Area Network (VXLAN) encapsulation, Microsoft's Network Virtualization over Generic Routing Encapsulation (NVGRE), and Internet Protocol (IP) Generic Routing Encapsulation (GRE).

EX-1009 [0005].

60. Farinacci describes Overlay Transport Virtualization (OTV), a system that “provide[s] an ‘overlay’ method of doing virtualization” where “state is maintained at the network edges” and “edge devices perform L2 learning and forwarding functions (similar to a traditional L2 switch) on their site-facing interfaces (internal interfaces) and perform IP based virtualization functions on their core-facing interfaces, for which an overlay network is created.” EX-1010, 3:1-13. The VPLS architecture described in Ye similarly provides “a multipoint-to-multipoint virtual private network that connects geographically isolated Ethernet-based LANs over a service provider’s MPLS backbone network so the LANs operate collectively as if they were a single LAN.” EX-1004, [0026].

2. Tunnel Managers That Collect Tunnel End Point Information and Establish Tunnel Connections Were Well Known

61. By December 2013, components that managed tunnels in an overlay network— such as by collecting information about tunnel end points within a network and using that information to establish tunnel connections between

them—were well known. These components gather information about tunnel end points within a network, including their IP addresses and associated virtual networks, and use that information to establish tunnel connections. For example, Koganti discloses a “tunnel management module” that “operates the switch as a tunnel gateway capable of terminating an overlay tunnel” and identifies “in a data packet a virtual Internet Protocol (IP) address associated with a virtual tunnel gateway.” EX-1011, Abstract, [0008], [0100].

62. It was additionally well known for overlay networks to permit automatic discovery of tunnel end points and to exchange information used to configure tunnel connections, including forwarding addresses and virtual network information. Ye discloses that “[a] PE device of a VPLS instance may automatically discover other PE devices in the same VPLS instance by using an automatic discovery protocol, such as the border gateway protocol (BGP)” and that “[a] PE device uses a signaling protocol, such as label distribution protocol (LDP) or BGP, to establish a PW with another PE device in the same VPLS instance.” EX-1004, [0033]-[0034]. Kamath similarly discloses a system that receives “an information exchange packet from each end point station in a virtual network having a specified virtual network identifier (VNID)” and processes each packet “to retrieve information about connections at each end point station in the virtual network having the specified VNID,” including assembling a Tunnel Information

Base (TIB) containing tunnel IDs, VNIDs, MAC addresses, and tunnel end point IP addresses. EX-1008, Abstract, [0008], [0070]. Yong similarly discloses that NVEs and overlay virtual gateways (OVGs) employ a control plane protocol, such as BGP, to perform control plane functions including “establishing an OVN [overlay virtual network], advertising encapsulation types and tunnel routes, tracking peers’ routes and corresponding encapsulation types, and maintaining lookup tables for routing and forwarding.” EX-1009, [0047]. FIG. 7 in Yong illustrates a method where an NVE or OVG advertises its supported encapsulation types and routes, receives advertisements from peers, updates a forwarding table with the peer’s encapsulation types and routes, and establishes an overlay tunnel with the peer when one has not been established. EX-1009, [0048]-[0049].

63. Similarly, it was well known for an external network manager to provide networking information to virtual switches within a virtual network. Kancherla discloses that a “virtual switches 114 and 134 are logically coupled to network manager 164, which provides virtual switches 114 and 134 with networking information required to communicate with each other.” EX-1017, [0049], [0053].

3. Tunnel Packet Processing—Including Encapsulation and Decapsulation—Was Well Known

64. By December 2013, tunnel packet processing functions—including receiving and decapsulating tunnel packets for L2 switching, and encapsulating

packets with tunnel headers for transmission through an overlay network—were well known. For example, Kamath describes that a:

tunnel is actually implemented by encapsulating packets transmitted by a source end station into an overlay header that transports the packet from the source switch to a target switch in user datagram protocol (UDP) transport via an interne[t] protocol (IP)-based network. The overlay header includes an identifier (ID) that uniquely identifies the virtual network. The target switch strips off the overlay header encapsulation, UDP transport header, and IP header, and delivers the original packet to the destination end station.

EX-1008 [0002].

65. It was additionally well known for such tunnel headers to include both a virtual network header containing a virtual network identifier and an L3 tunnel header with source and destination address fields. Mahalingam, for example, defines a MAC-in-UDP encapsulation scheme where the original Layer 2 frame has a VXLAN header added—including a 24-bit VXLAN Network Identifier (VNI)—and is then placed in a UDP-IP packet with source and destination VTEP IP addresses. EX-1007, pp. 7-16 (Sections 4.1 & 5). Yong similarly describes that a packet header “may comprise an inner address field 330, a VXLAN encapsulation header 320, and an outer address field 310” where “the outer address field 310 may comprise a source IP address and a destination IP address” set to the respective NVE IP addresses. EX-1009, [0043]. Grover likewise describes an

encapsulation format that includes an outer IP header with source and destination OTV Edge Device addresses, an OTV header with an Instance ID, and the original L2 frame. EX-1012, pp. 14-18. Yong also describes the same encapsulation and decapsulation process at an overlay virtual gateway (OVG), which receives an overlay data packet, decapsulates it, retrieves the destination address from an inner address field, retrieves the egress NVE's address from an address mapping table, re-encapsulates the packet with a new encapsulation header, updates the tunnel header by "setting the destination IP address to the egress NVE's IP address," and forwards the re-encapsulated packet. EX-1009, [0055]-[0056], FIG. 9. In Yong, the packet header comprises "an inner address field, an encapsulation header, and an outer address field," where the encapsulation header comprises "a VNID and/or other encapsulation type specific information" and the outer address field comprises "IP addresses of the source and egress tunnel endpoints." EX-1009, [0041].

4. Virtual Network Identifiers (VNIDs) for Identifying and Isolating Virtual Networks Were Well Known

66. By December 2013, the use of virtual network identifiers to identify and isolate virtual networks within overlay architectures was well known. These identifiers were variously referred to as VNIDs, VNIs (VXLAN Network Identifiers), VPLS IDs, Instance IDs, and I-SIDs (Instance Service Identifiers). As Kamath explains, "[c]omponents of an overlay network 300 typically identify

where to route packets based on a virtual network identifier, referred to as a VNI or VNID. This is typically a 24-bit code or number, which excludes 0x0 and 0xFFFFFFFF.” EX-1008, [0047]. Further, Ye discloses that “[e]ach PE device of a VPLS instance establishes a VSI [virtual switching instance] for that VPLS instance” with a corresponding VPLS ID, and that the PE device uses signaling protocols to exchange VSI labels that identify the particular VSI associated with a given VPLS instance. EX-1004, [0034]-[0035]. Yong discloses that the encapsulation header of a tunnel packet “may comprise a VNID and/or other encapsulation type specific information.” EX-1009, [0041].

5. Virtual Switching Instances (VSIs) Corresponding to Virtual Network Identifiers That Perform L2 Switching Were Well Known

67. By December 2013, virtual switching instances, which may be tied to a specific overlay virtual network, that perform Layer 2 switching functions—including obtaining source address information and forwarding packets based on destination address information—were well known.

68. A virtual switching instance (VSI) is a software-based component within a network device that emulates the functions of a physical Ethernet switch within a Layer 2 virtual network, such as learning the MAC addresses of connected devices and forwarding packets to the appropriate destination based on those addresses. For example, Ye explains that a “VSI is an Ethernet bridge function

entity of a VPLS instance on a PE device[, that] forwards packets based on MAC address or both MAC destination and VLAN tag,” and “maintains a MAC address table and supports MAC address learning and aging, and packet forwarding and flooding.” EX-1004, [0030], [0035]. Figueira explains that a “VSI is an IETF term for a bridge used to construct a VPLS service over an MPLS network. As defined in IETF RFC 4762 and IETF RFC 4761, a VSI provides standard MAC address learning, filtering, and forwarding actions, as defined in IEEE 802.1D and IEEE 802.1Q standards, which define how learning bridges work.” EX-1006, [0018].

69. It was well known for a VSI to forward packets based on destination address information contained in forwarding tables within the VSI—including by receiving decapsulated packets, switching according to Layer 2 information, and forwarding the packet for encapsulation and sending through a tunnel. For example, Ye discloses that when a PE device receives a unicast packet, it “maps the packet to a VSI of a particular VPLS instance and its MAC address table” and, when the MAC destination address is known, “the PE device forwards the packet to an AC or a PW specified in the entry.” EX-1004, [0036], [0039]. Yong discloses that when an NVE supports multiple overlay virtual networks, “the virtual network identifier (VNID) of the packet should be checked and data packets should be forwarded accordingly,” and that an NVE maintains an address mapping table which “may be built previously from control plane.” EX-1009, [0039], [0051].

**6. Modified Flooding Rules Based on Input Port Type—
Including Split-Horizon Forwarding—Were Well Known**

70. By December 2013, the concept of a virtual switching instance having both edge-facing ports and network-facing ports with modified flooding rules based on input port type was well known. Edge-facing ports were variously referred to as edge ports, customer-facing ports, access circuits (ACs), internal interfaces, and customer interface connections. Network-facing ports were variously referred to as inter-switch ports, pseudowires (PWs), overlay interfaces, and service instance connections.

71. The problem of forwarding loops in L2 networks—where packets circulate endlessly between devices, consuming bandwidth and degrading network performance—was also well known. For example, various techniques were developed to address this problem, including the “split-horizon rule,” which was a well-established loop-prevention technique used in L2 networks. EX-1004, [0040], EX-1013, 2:26-32; EX-1006, [0030]. At its most basic, the rule prevents a packet from being sent back out the same type of interface it came in on—ensuring that traffic does not bounce endlessly between devices. Gateway or edge devices that sit between two different types of network connections—such as edge-facing ports (connected to local devices) and network-facing ports (connected to other network sites via tunnels)—apply split-horizon rules to determine which ports should

receive broadcast, unknown unicast, and multicast (BUM) packets, as those types of packets are “flooded” to multiple destinations.

72. One well-known version of the split-horizon rule provides that when a BUM packet is received from a network-facing port, the packet is flooded only to edge-facing ports—not back to other network-facing ports. Ye discloses this, explaining that “PE devices may be fully meshed logically and they implements [sic] horizontal split forwarding to avoid loops. When a PE device receives from a PW associated with a VSI (1) a unicast packet with an unknown MAC destination address, (2) a multicast packet, or (3) a broadcast packet, the PE device floods the packet to all ACs associated with the VSI but not PWs associated with the VSI.” EX-1004, [0040]. Ye also discloses that when a BUM packet is received from an edge-facing port (AC), it is flooded to “to all the ACs and all the PWs associated with the VSI.” *Id.* Ghosh similarly explains that the split-horizon rule is used to avoid data loops in VPLS networks by prohibiting “a router [from] flooding a copy of an L2 packet back out an interface from which the original L2 packet was received.” EX-1013, 2:26-32.

73. Figueira discloses using the split-horizon rule to permit traffic arriving from one type of port to be flooded only to a port of the other type—never to ports of the same type as the input. Figueira explains this rule in the context of L2VPN over PBB networks and VPLS networks:

(1) whenever a frame is received from a Service Instance (which is transported over a PBT trunk), it can only be forwarded to Pseudowires over the MPLS network; and (2) whenever a frame is received from a Pseudowire (which is transported over an MPLS network), it can only be forwarded to Service Instances (which are transported over PBT trunks).... The split horizon rule 511 exemplified here is the common well-known split horizon rule defined in RFC 4762 to avoid loops in the VPLS network.... Split horizon rule 511 states that: (1) whenever a frame is received from a Pseudowire, it can only be forwarded to customer bound interfaces; and (2) whenever a frame is received from a customer interface, it can be forwarded to any one of the other VSI connections. For example, a frame received from a Pseudowire will never be forwarded to another Pseudowire.

EX-1006, [0030].

74. Under this rule, a BUM packet received from an edge-facing connection is flooded only to network-facing connections, and a BUM packet received from a network-facing connection is flooded only to edge-facing connections. *Id.*

VII. CHALLENGES

A. Identification of Challenges

75. Based on my understanding of the law of obviousness discussed above, Claims 1-13 of the '961 Patent are unpatentable as being obvious over Ye

Ground	Claims	Basis	Reference
1	1-13	§ 103	Ye

B. Prior Art

76. I have been informed that Ye, U.S. Patent Application Publication No. 2015/0146727, qualifies as prior art under § 102(a)(1) and/or § 102(a)(2).

C. Ground 1: Claims 1-13

1. The Ye Reference

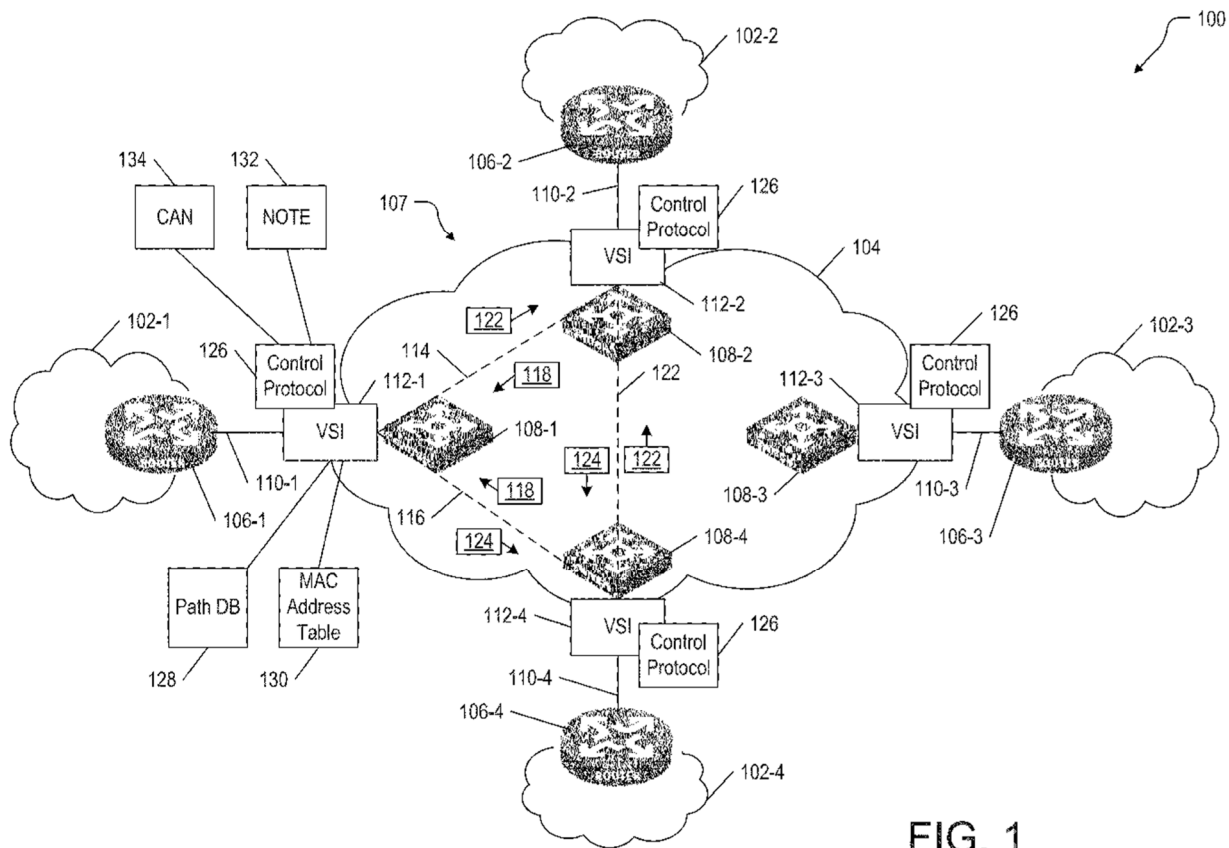
77. Ye is directed to methods and devices for forwarding packets in a Virtual Private LAN Service (VPLS) network—a technology for connecting geographically dispersed Layer 2 (L2) networks into a single, unified virtual LAN over a service provider’s backbone network. EX-1004, [0001], [0026]. Like the ’961 Patent, Ye addresses the problem of connecting multiple L2 networks located at different physical sites so that they operate collectively as if they were a single network. EX-1004, [0001].

78. Ye explains that VPLS developed from a proposal for a multi-protocol label switching (MPLS) L2 virtual private network (L2VPN). EX-1004, [0002]. VPLS provides a multipoint-to-multipoint virtual private network that connects geographically isolated Ethernet-based LANs over a service provider’s MPLS backbone network so the LANs operate collectively as if they were a single LAN. EX-1004, [0002], [0026]. “With VPLS, the technical advantages of Ethernet

technology are combined with the technical advantages of MPLS to emulate all the functionalities of a conventional LAN.” EX-1004, [0002].

79. As illustrated in FIG. 1, a VPLS-capable network (100) includes customer sites (102) connected to a backbone network (104) through provider edge (PE) devices (108). EX-1004, [0048]. Each PE device is connected to one or more customer edge (CE) devices by attachment circuits (ACs). EX-1004, [0029]. Each PE device includes a virtual switch instance (VSI) for each VPLS instance—an Ethernet bridge function entity that forwards packets based on MAC address. EX-1004, [0030]. Pseudo wires (PWs) (114, 116, 120¹) provide bidirectional virtual connections between VSIs on different PE devices. EX-1004, [0031], [0050].

¹ Element 120 is mislabeled as element 122 in FIG. 1, and as a result there are multiple 122 elements and no element 120 in the figure. But a POSITA understood that the 122 label that points at the dotted line connecting VSI 112-2 and VSI 112-4 and spans the gap between PE devices 108-2 and 108-4 is what was meant to be labeled element 120.



80. A PE device of a VPLS instance may automatically discover other PE devices in the same VPLS instance using an automatic discovery protocol, such as the border gateway protocol (BGP), or it may be manually configured. EX-1004, [0033]. A PE device uses a signaling protocol, such as label distribution protocol (LDP) or BGP, to establish a PW with another PE device in the same VPLS instance. EX-1004, [0034]. Under both manual and automatic configuration, each PE device allocates a VSI label for each inbound virtual circuit label-switched paths (VC LSPs), and the associated VPLS instance and PW are identified by a

VPLS ID and a PW ID, respectively. *Id.* Ye discloses that tunnels between PE devices may be MPLS tunnels or generic routing encapsulation (GRE) tunnels, and a tunnel can carry multiple PWs. EX-1004, [0032]. A POSITA understood that GRE tunnels operate over Layer 3 networks using IP addresses.

81. Ye describes that each PE device performs L2 switching—forwarding Ethernet packets based on MAC address. EX-1004, [0026]. When a PE device receives a unicast packet from an AC, it maps the packet to a VSI based on the AC's port number and searches the VSI's MAC address table for an entry matching the MAC destination address. EX-1004, [0036], [0092]. When the MAC destination address is known, the PE device forwards the packet to the AC or PW specified in the matching entry in the VSI's MAC address table. EX-1004, [0039], [0093]-[0095]. When the packet is forwarded through a PW, the PE device adds a VSI label and a tunnel label to encapsulate the packet for forwarding. *Id.*

82. When a PE device receives a packet from an inbound VC LSP of a PW, the PE device removes the tunnel label to expose the VSI label, and based on the VSI label, maps the packet to a VSI and its MAC address table. EX-1004, [0036], [0098]. If the MAC destination address is known, the PE device removes the VSI label and forwards the packet to the path or port identified in the matching entry—adding a new VSI label and tunnel label if forwarding through another PW. EX-1004, [0099]-[0102]. In short, the PE device operates as a relay for inter-site

traffic: it decapsulates the incoming tunnel packet, performs L2 switching based on MAC address to determine the correct outbound path, and re-encapsulates the packet with appropriate labels for transmission to the next destination.

83. Ye discloses that PE devices are fully meshed and implement horizontal split forwarding to avoid loops. EX-1004, [0040]. When a PE device receives either (1) a unicast packet with an unknown MAC destination address, (2) a multicast packet, or (3) a broadcast packet, the PE device floods the packet to all ACs associated with the VSI but not to PWs associated with the VSI. *Id.* When the PE device receives the same type of packet from an AC associated with the VSI, the PE device floods the packet to all the ACs and all the PWs associated with the VSI. *Id.*

84. Ye explains that “[e]ach VSI maintains a MAC address table and supports MAC address learning and aging, and packet forwarding and flooding.” EX-1004, [0035]. When the MAC source address of a received packet is unknown, the PE device creates a new entry in the MAC address table mapping the MAC address to the AC (and VLAN) or outbound VC LSP of the PW. EX-1004, [0037]. Ye further discloses that PE devices may use a first multipoint-to-multipoint label-switched path (MP2MP LSP) to communicate MAC address information to all other PE devices in the backbone network, so that PE devices learn MAC

addresses on the control plane through notification packets rather than flooding the backbone network with unknown packets. EX-1004, [0045], [0053].

85. FIG. 18 illustrates the architecture of a PE device (1800), which includes a receiving module (1810), a searching module (1820), a sending module (1830), a determination module (1840), a path data management module (1850), and a forwarding data management module (1860). EX-1004, [0129]. The receiving module receives packets from ACs and from PWs. EX-1004, [0130]. The PE device includes a tunnel label removing module and a VSI label removing module for processing encapsulated packets. EX-1004, [0133]. The searching module searches the MAC address table of the appropriate VSI for a matching entry. EX-1004, [0128]-[0129]. The sending module floods or forwards packets based on the search results. EX-1004, [0128], [0036]-[0040]. The determination module, path data management module, forwarding data management module handle notification and cancellation packets. EX-1004, [0131], [0135]. The determination module determines if the PE device has a VSI associated with the VPLS instance identified by the notification packet. *Id.* If so, the “[p]ath data management module 1850 adds a new entry recording the relationship among the MAC address, the VSI label, and the selected tunnel to the path database of the VSI.” EX-1004, [0131]. The forwarding data management module then updates the MAC address table associated with that VSI based on that path database. *Id.*

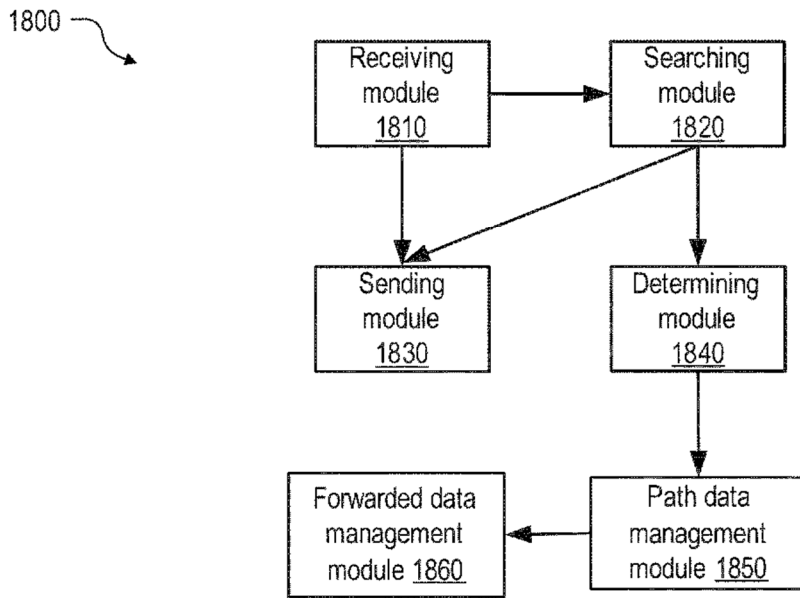


FIG. 18

2. Detailed Application of Ye to the Challenged Claims

Claim 1

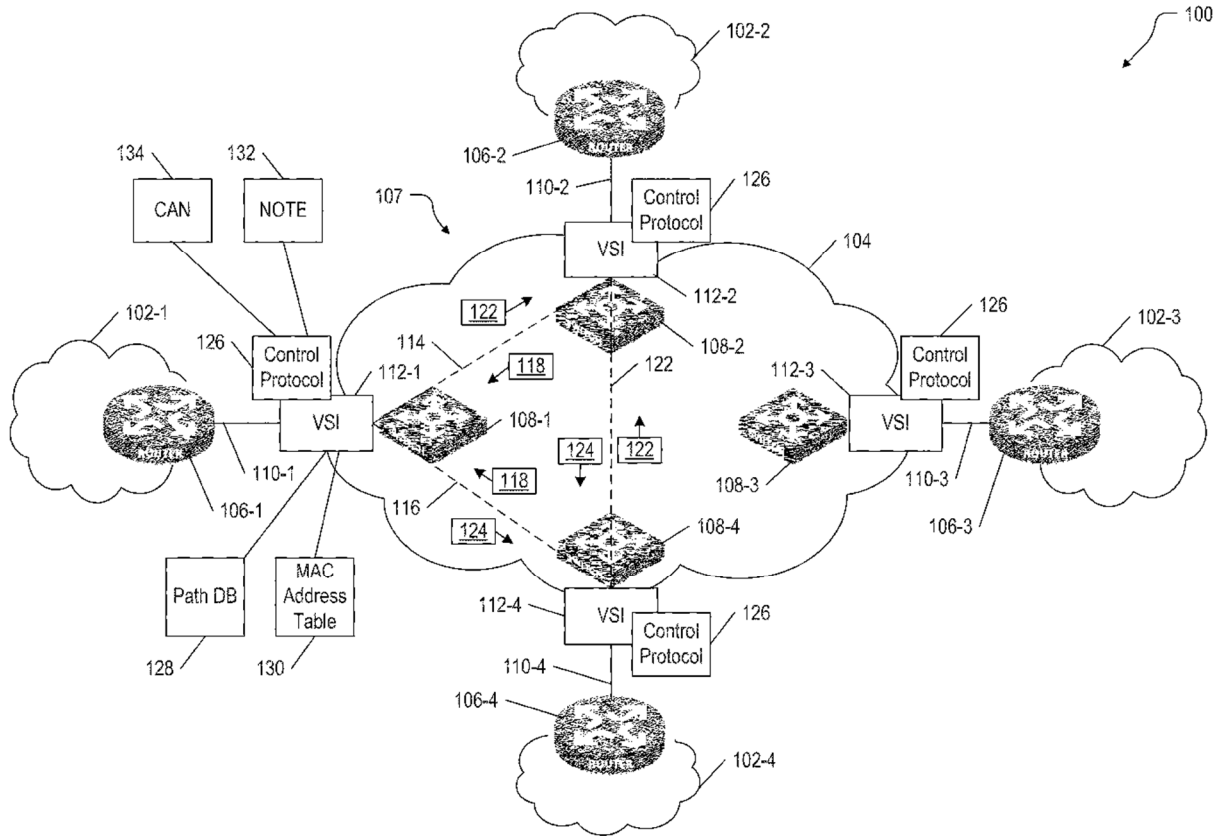
[1.0] A network virtualization apparatus comprising:

86. To the extent the preamble is limiting, Ye discloses this limitation. Ye discloses a Virtual Private LAN Service (VPLS), a Layer 2 multipoint virtual private network (VPN) that connects multiple customer sites into a single bridged domain over a backbone multi-protocol label switching (MPLS) network and emulates an Ethernet bridge by forwarding packets based on media access control (MAC) addresses along with virtual LAN (VLAN) tags. EX-1004, [0026].

87. Ye identifies the elements of a VPLS instance: customer edge (CE) devices at customer sites, provider edge (PE) devices [*network virtualization apparatus*] that connect CE devices to the backbone network and map and forward

packets between private networks through tunnels, and physical/virtual attachment circuits (ACs) that link CE devices to PE devices. EX-1004, [0027]-[0029]. Within each PE device, Ye describes a virtual switch instance (VSI), which provides the Ethernet-layer switching functionality for a given VPLS instance and performs Layer 2 switching. EX-1004, [0030]. VSIs on different PE devices are interconnected by pseudo wires (PWs), each PW being a bidirectional virtual connection between VSIs on two PE device composed of two unidirectional virtual-circuit label-switched paths (VC LSPs) identified by a PW label in opposite directions, and the PWs are carried over tunnels, such as MPLS or Generic Routing Encapsulation (GRE) tunnels. EX-1004, [0031]-[0032]. A POSITA understood that GRE tunnels operate over Layer 3 networks using IP addresses.

88. Thus, a POSITA understood that the PE device disclosed in Ye is a *network virtualization apparatus*.



EX-1004, FIG. 1.

[1.1] a tunnel manager configured to collect tunnel end point information within a network, and to connect each tunnel end point with a tunnel based on the tunnel end point information collected;

89. Ye discloses control-plane functionality using an automatic discovery protocol, such as the border gateway protocol (BGP) on PE devices [*tunnel manager*] that discovers other PE devices in the same VPLS instance [*tunnel manager configured to collect tunnel endpoint information within a network*] and establishes the tunnels and PWs that interconnect them [*tunnel manager ... to*

connect each tunnel end point with a tunnel based on the tunnel end point information collected]. EX-1004, [0032]-[0034], [0053].

90. Ye discloses that a “tunnel” is a “direct channel between PE devices for transparent data transmission between the PE devices. Tunnels are used to carry PWs. A tunnel can be a MPLS tunnel or a generic routing encapsulation (GRE) tunnel. A tunnel can carry multiple PWs.” EX-1004, [0032]. To collect tunnel end point information, Ye discloses that a PE device may be manually configured with addresses of other PE devices in the same VPLS instance, or a PE device can automatically discover other PE devices using an automatic discovery protocol, such as the border gateway protocol (BGP). EX-1004, [0033]. Using the collected end point information, a PE device then uses a signaling protocol, such as label distribution protocol (LDP) or BGP, to establish a PW with another PE device in the same VPLS instance, using tunnels to carry the PWs [*to connect each tunnel end point with a tunnel based on the tunnel end point information collected*]. EX-1004, [0034].

91. Ye further discloses that a first multipoint-to-multipoint label-switched path (MP2MP LSP) connects all PE devices in the backbone network to provide a communication channel for the exchange of protocol packets. EX-1004, [0045], [0053]. Over this first MP2MP LSP, PE devices run a control protocol 126, which includes a VPLSMAC address routing protocol to learn MAC

addresses on the control plane. PEs may also use an automatic discovery protocol, such as BGP, to discover other PE devices of the same VPLS instances. EX-1004, [0053].

92. FIG. 3 illustrates how the system connects the PE devices with labeled tunnel segments over the first MP2MP LSP. EX-1004, [0055]. The first MP2MP LSP service 300 has a PE device 108-3 that acts as a root node 302, and the root node is connected to each PE device node 108 through bidirectional LSP segments. EX-1004, [0056]. When the root node PE device receives a broadcast packet 304 with tunnel label 306 from PE device 108-1, the root node replicates the packet and replaces the ingress tunnel label with leaf tunnel labels 310 and 312 to send the packet to PE leaf nodes 108-2 and 108-4, respectively. EX-1004, [0056]. Each PE device maintains tunnel end point information necessary to reach other PE devices over each individual LSP segments, and it uses that information to direct tunnel traffic to each end point. EX-1004, [0054], [0056].

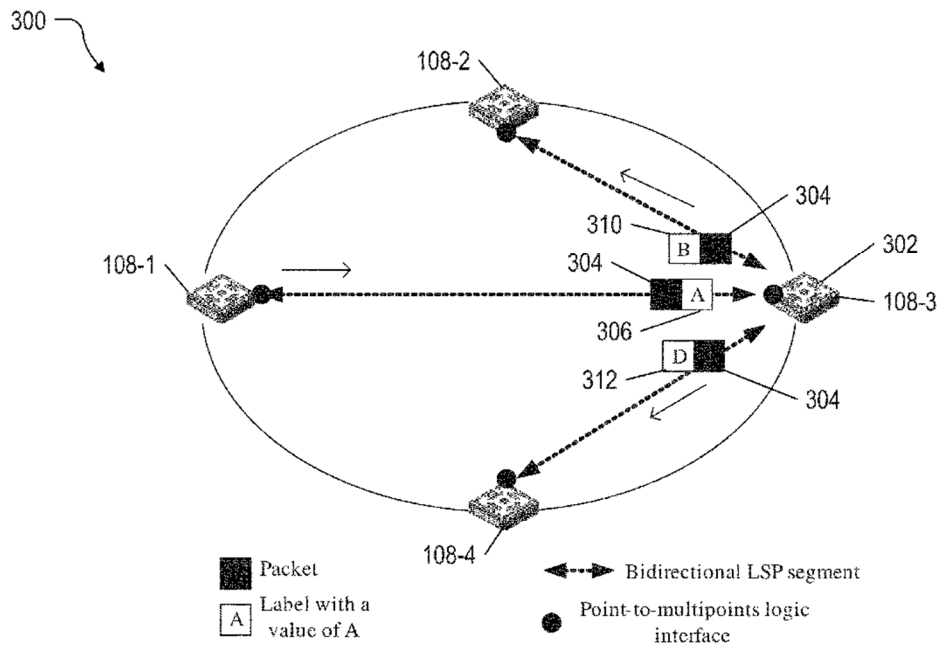


FIG. 3

EX-1004, FIG. 3.

93. Ye's discloses a flowchart in FIG. 5 that further details how this process works: after discovery, the PE device connects each end point with tunnels and PWs; the first MP2MP LSP is created; the VPLS instance is created and PEs automatically discover each other using a discovery protocol; and PE devices communicate VSI labels over the first MP2MP LSP to establish PWs between VSIs. EX-1004, [0059]-[0063].

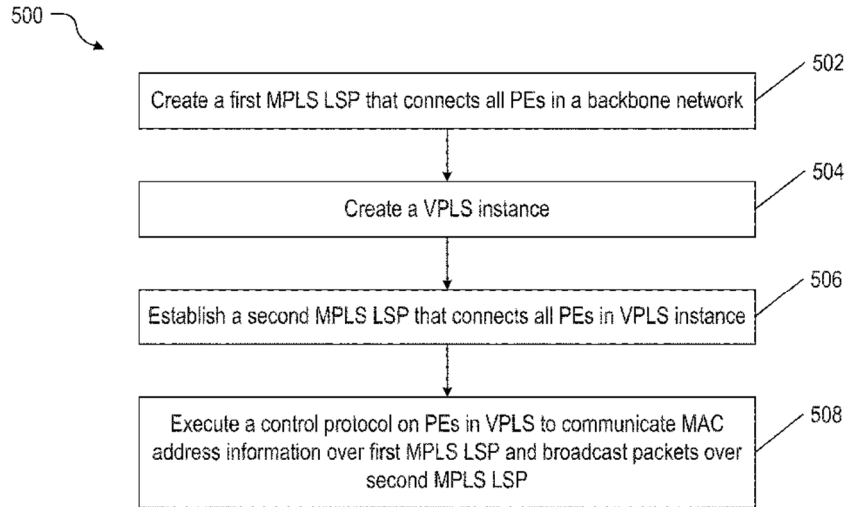


FIG. 5

EX-1004, FIG. 5.

94. Ye's disclosure of a *tunnel manager* is consistent with Patent Owner's assertion in its infringement contentions that use of BGP protocol, *i.e.*, MP-BGP EVPN, reflects a tunnel manager at spine nodes of a fabric:

The Accused Instrumentalities include a tunnel manager configured to collect tunnel end point information within a network, and to connect each tunnel end point with a tunnel based on the tunnel end point information collected. For example, as evidenced below, Cisco ACI Multi-Site exchanges tunnel endpoint information — including each site's Overlay Unicast TEP (OUTEP), Overlay Multicast TEP (O-MTEP), and the EVPN-RID — through MP-BGP EVPN sessions between the spine nodes of the participating fabrics. The Cisco whitepaper explains that “endpoint reachability information is exchanged across sites using a Multiprotocol-BGP (MP-BGP) Ethernet VPN (EVPN) control plane,” and that O-UTEP and O-MTEP are the anycast tunnel-endpoint addresses used by the fabrics. Once this tunnel-endpoint information is exchanged, “all communication...between endpoints connected to different

sites is achieved by establishing site-to-site VXLAN tunnels” that use the collected O-UTEP/O-MTEP addresses as tunnel endpoints. Accordingly, the accused instrumentalities include a “tunnel manager” that collects tunnel endpoint information and creates VXLAN tunnels based on that information.

EX-1016, pp. 6-7 (emphasis added).

95. A POSITA understood that a control-plane protocol, such as BGP, allows a PE device to collect tunnel endpoint information and establish tunnels within a VPLS, and therefore understood the PE devices of Ye include the *tunnel manager*.

[1.2] a tunnel packet end point configured to receive a tunnel packet and process the packet such that an L2 switching may be performed, and to transmit the packet to a domain VSI (virtual switching instance);

96. Ye discloses that a PE device [*tunnel packet end point*] receives a tunnel packet from an inbound VC LSP of a PW, where the packet is an MPLS packet carrying an outer tunnel label and an inner PW/VSI label [*tunnel packet end point configured to receive a tunnel packet*]. EX-1004, [0036], [0099]. Upon receipt, the PE device processes the packet by removing the tunnel label to expose the PW/VSI label. EX-1004, [0036], [0099]. Based on the exposed PW/VSI label, the PE device maps the packet to a VSI of a particular VPLS instance and to that VSI’s MAC address table [*a tunnel packet end point configured to ... process the*

packet such that an L2 switching may be performed]. EX-1004, [0036, 0100].

After this processing, the selected VSI within the PE device performs Layer 2 switching by forwarding packets based on their MAC destination address. EX-1004, [0100]-[0103].

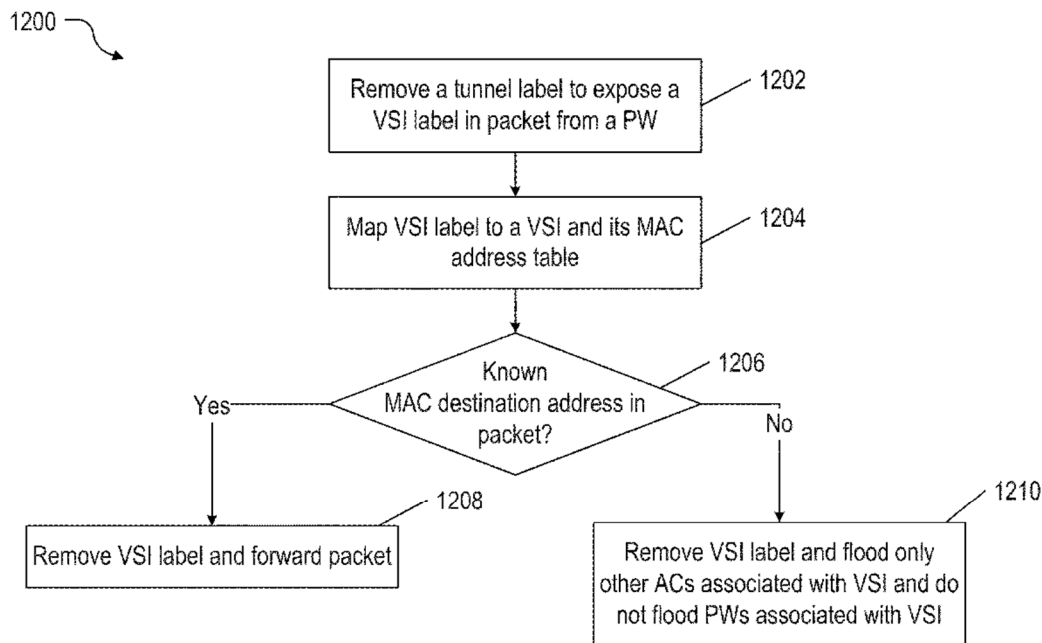


FIG. 12

EX-1004, FIG. 12.

97. Ye also describes PE device modules that perform this process, including modules that remove the tunnel label and the VSI label and that locate the MAC address table of the mapped VSI. EX-1004, [0129]-[0134].

98. Although the above description describes the processing of MPLS packets in an MPLS backbone network, Ye also describes using GRE packets that

operate over a Layer 3 backbone network using IP address information, which would involve the same processing discussed above. Ex-1004, [0032] (“A tunnel can be a MPLS tunnel or a generic routing encapsulation (GRE) tunnel.”), [0078] (“As backbone network 104 has a full mesh of tunnels (MPLS or GRE) between all the PE devices”).

[1.3] the domain VSI configured to perform the L2 switching on the processed packet, and to transmit the packet to a tunnel packet generator; and

99. Ye discloses a VSI [*domain VSI*] is an Ethernet bridge function entity of a VPLS instance on a PE device that forwards packets based on a MAC destination address or a MAC destination address and VLAN tag, which is Layer 2 (L2) switching. EX-1004, [0030]. After the provider edge device (PE) maps an inbound tunnel packet to the appropriate VSI using the exposed VSI/PW label, the VSI consults its MAC address table to determine whether the destination MAC address is known and forwards accordingly [*the domain VSI configured to perform the L2 switching on the processed packet*]. EX-1004, [0036], [0101]-[0103]. Specifically, Ye further describes the PE device-level modules, including a searching module in the VSI that consults the VSI’s MAC address table to perform L2 switching. EX-1004, [0127]-[0129]. Therefore, Ye discloses *the domain VSI configured to perform the L2 switching on the processed packet*.

100. When the VSI's L2 switching involves egress over a PW, Ye discloses that the searching module sends the packet to the sending module of the PE device [*tunnel packet generator*] which encapsulates the packet with the appropriate VSI label and tunnel label and forwards it over the selected tunnel. EX-1004, [0102], [0127]-[0129]; FIG. 18. Therefore, Ye discloses *the domain VSI configured ... to transmit the packet to a tunnel packet generator.*

[1.4] the tunnel packet generator configured to add a tunnel header to the packet for which the L2 switching has been performed to generate a new tunnel packet, and to transmit the new tunnel packet.

101. Ye discloses that when the VSI switching requires sending a packet over a PW, the sending module [*tunnel packet generator*] of the PE device encapsulates the packet [*the packet for which the L2 switching has been performed*] by adding the appropriate VSI/PW and tunnel labels and forwards the packet over the selected tunnel. EX-1004, [0032], [0039], [0102], [0128]. Specifically, Ye discloses that the PE device uses a searching module to find the appropriate destination for the packet using the per-VSI path database containing MAC address tables which house the information required for forwarding. *Id.* The sending module then adds a VSI label and a tunnel label to encapsulate and forward the packet [*the tunnel packet generator configured to add a tunnel header to the packet for which the L2 switching has been performed to generate a new*

tunnel packet, and to transmit the new tunnel packet]. EX-1004, [0035]-[0036], [0039], [0127]-[0132].

Claim 2

[2.1] The apparatus according to claim 1, wherein the domain VSI, when perform the L2 switching, obtains L2 SA (source address) information for the processed packet,

102. Ye discloses this limitation.

103. As set forth above (*see* [1.2] and [1.3]), Ye discloses the *domain VSI* that *perform[s] L2 switching*. Ye discloses that each VSI within a PE “maintains a MAC address table and supports MAC address learning and aging, and packet forwarding and flooding.” EX-1004, [0035]. Ye explains that when a PE device receives a “packet from the inbound VC LSP of a PW,” that packet may include a tunnel label, a PW label, and an Ethernet frame. EX-1004, [0036]. Ye further discloses “an Ethernet frame [] may include a MAC source address, a MAC destination address, a VLAN tag with a VLAN ID, and a payload.” EX-1004, [0036]. Once the PE device processes the packet by removing the tunnel label, “the PE device maps the packet to a VSI of a particular VPLS instance and its MAC address table.” EX-1004, [0036]. Upon receipt of a packet, the VSI either learns the MAC source address by creating a new entry “in the MAC address table that maps the MAC address to the AC (and VLAN) or an outbound VC LSP of the PW” if the source is unknown or by resetting the aging timer if it is known

[obtains L2 SA (source address) information for the processed packet]. EX-1004, [0035], [0037]-[0038].

[2.2] determines whether or not L2 DA (destination address) information of the processed packet is a unicast address already learned, and

104. Ye discloses that after a unicast packet is mapped by the PE device to the appropriate VSI, the system checks whether the MAC destination address is known in the VSI's MAC address table [*determines whether or not L2 DA (destination address) information of the processed packet is a unicast address already learned*]. EX-1004, [0039]. Ye further discloses that when the MAC destination address is known, the unicast packet is forwarded according to the MAC address table entry, which includes both the MAC source address and MAC destination address. EX-1004, [0039].

[2.3] performs a flooding based on a type of an input virtual port, when the L2 DA information is not already learned unicast address, or is multicast or broadcast address;

105. Because PWs and ACs are virtual circuits that serve as the input paths through which packets reach the VSI, and the VSI—as “an Ethernet bridge function entity”—receives and switches those packets at L2, a POSITA understood that a PW or AC functions as an *input virtual port* to the VSI. EX-1004, [0029]-[0031], [0036], [0143]. When the destination of a packet is unknown [*L2 DA information is not already learned unicast address*] or the packet is a multicast or

broadcast, Ye specifies different flooding behaviors based on whether the packet was received by the PE device from a VC LSP of a PW or AC virtual port associated with the VSI [*performs a flooding based on a type of an input virtual port, when the L2 DA information is not already learned unicast address, or is multicast or broadcast address*]. EX-1004, [0026], [0036], [0040].

[2.4] wherein the flooding is performed on only domain virtual ports within the domain VSI, when the type of the input virtual port being an edge virtual port, and the flooding is performed on only edge virtual ports within the domain VSI when the type of the input virtual port being a domain virtual port.

106. Ye discloses that “[w]hen a PE device receives a broadcast or multicast packet at an AC” [*when the L2 DA information is ... multicast or broadcast address and the type of the input virtual port being an edge virtual port*] the PE device broadcasts the packet only to other “PE devices having VSIs in the same VPLS instance” by adding PW labels and tunnel labels to send the broadcast or multicast packet [*wherein the flooding is performed on only domain virtual ports within the domain VSI*]. EX-1004, [0041].

107. Ye also discloses that when the PE device receives a “(1) a unicast packet with an unknown MAC destination address, (2) a multicast packet, or (3) a broadcast packet” packet from an AC associated with a VSI [*when the type of the input virtual port being an edge virtual port*] the PE device floods all ACs and all PWs associated with that VSI. EX-1004, [0040]. A POSITA understood that where

there are no other ACs associated with the VSI connected to the PE device, the PE will flood only PWs associated with the VSI [*wherein the flooding is performed on only domain virtual ports within the domain VSI*].

108. Ye discloses that “PE devices may be fully meshed logically and they implements [sic] horizontal split forwarding to avoid loops.” EX-1004, [0040]. As discussed in the State of the Art section, a POSITA understood that implementing “horizontal split forwarding to avoid loops” is equivalent to employing the split-horizon rule. EX-1006, [0030]; EX-1013, 2:26-32; EX-1014, at 11. Thus, a POSITA understood that Ye taught that *the flooding is performed on only domain virtual ports within the domain VSI, when the type of the input virtual port being an edge virtual port* in order to avoid flooding edge virtual ports to avoid forwarding loops, as disclosed by Ye in accordance with the split-horizon rule.

109. Ye’s disclosure is consistent with the disclosure of the ’961 Patent which explains that modifying the flooding rule to only flood domain virtual ports when the input virtual port is an edge virtual port is “used to prevent generation of a forwarding loop by flooding.” EX-1001, 7:51-57. A POSITA understood that standard ethernet learning procedure using the automatic discovery protocol border gateway protocol (BGP), would result in practicing this limitation because flooding would be restricted according to the split-horizon rule. EX-1004, [0034]; *see also* EX-1006, [0030] (explaining that by following the split-horizon rule, “a frame

received from Service Instance 506 can only be forwarded to Pseudowires 508 and 509 and customer interface 550 and will not be forwarded to another Service Instance, such as Service Instance connection 507. Conversely, a frame received from Pseudowire 508 can only be forwarded to Service Instances 506 and 507 and customer interface connection 550 and will not be forwarded to another Pseudowire, such as Pseudowire connection 509.”).

110. Ye further discloses that when the PE device receives a packet from a PW associated with a VSI [*when the type of the input virtual port being a domain virtual port*] the PE device floods the packet to all ACs associated with that VSI and not to the PWs of that VSI [*the flooding is performed on only edge virtual ports within the domain VSI*]. EX-1004, [0029], [0031], [0040], [0103], FIG. 12.

Claim 3

[3] The apparatus according to claim 2, wherein, the domain VSI, when performing the L2 switching, in response to the L2 DA information being a unicast address already learned, transmits the processed packet to a destination virtual port corresponding to the unicast address, and obtains an L3 address of a destination tunnel end point of the packet from the destination virtual port.

111. Ye discloses this limitation.

112. As set forth above (*see* [1.2], [1.3], and [2.0]), Ye discloses the *domain VSI* that *perform[s] L2 switching*.

113. When the PE device receives a unicast packet, including an Ethernet frame with a MAC source address, a MAC destination address, a VLAN tag with a VLAN ID, and a payload, from an AC, it maps the packet to a VSI of a particular VPLS instance and to that VSI's MAC address table based on the AC's port number. EX-1004, [0036]. When the PE device receives a unicast MPLS packet from an inbound VC LSP of a PW, it removes the outer tunnel label to expose the inner PW label and, based on the PW label, maps the packet to a VSI of a particular VPLS instance and to that VSI's MAC address table. EX-1004, [0036].

114. When the MAC destination address of the unicast packet is known [*in response to the L2 DA information being a unicast address already learned*], the packet is forwarded according to the entry in the VSI's MAC address table: either to an AC or to a PW identified by that entry [*transmits the processed packet to a destination virtual port corresponding to the unicast address*]. EX-1004, [0039]. If the forwarding path is through a PW, the packet is encapsulated with a PW label and a tunnel label and is sent using the known per-VSI address information contained in the VSI's MAC address table [*obtains an L3 address of a destination tunnel end point of the packet from the destination virtual port*]. EX-1004, [0039], [0076]-[0079].

115. The path database and corresponding MAC address tables provide, for that destination, the selected tunnel ID and other address information used to

transmit, such as tunnel label, output interface, and next hop. EX-1004, [0039], [0076]-[0079]. Ye discloses in paragraph [0076] that when a PE device receives a notification packet from another PE device, it creates an entry in the “path database” on the control plane for the corresponding VSI. Table 4 in Ye illustrates the structure of this path database, which includes the following fields: “Path Identifier,” “VSI Label,” and “Tunnel Package” information comprising “Tunnel Label,” “Output interface,” and “Next hop.” EX-1004, [0076]. The PE device then updates the MAC address table on the data plane based on information in this path database. EX-1004, [0076]-[0077].

116. A POSITA understood that, while Ye describes a VPLS embodiment involving MPLS tunnels where L2 information is used for forwarding within the virtual network, Ye expressly discloses that the tunnels between PE devices are not limited to MPLS tunnels. Rather, Ye discloses that “[a] tunnel can be a MPLS tunnel or a generic routing encapsulation (GRE) tunnel.” EX-1004, [0032].

117. A POSITA understood that GRE tunnels operate over Layer 3 networks using IP addresses. GRE was one of the “number of encapsulation protocols available in the industry” prior to the Priority Date for providing overlay network services, and “Internet Protocol (IP) Generic Routing Encapsulation (GRE)” operates by encapsulating traffic in IP packets that traverse a backbone network. EX-1009, [0005]. A POSITA understood that:

MPLS embodiments ... can also use other tunneling protocols to facilitate end-to-end virtualization.... GRE can be used to establish an end-to-end tunnel between two network interface cards. GRE packets are encapsulated within IP and use IP as a delivery protocol. Therefore, when GRE is used, an outer IP header is used outside the GRE header. Effectively, to preserve the layer-2 VM identifying information, the packet is encapsulated with layer-3 headers.

EX-1005, [0066]-[0067].

118. Thus, a POSITA understood that in Ye when the tunnels between PE devices in a VPLS instance are GRE tunnels—as Ye expressly contemplates at paragraph [0032]—that the “Tunnel Package” and “Tunnel Label” information contained in the MAC address table of the VSI would include destination IP address information [*an L3 address of a destination tunnel end point of the packet*] because GRE tunnels operate over L3 networks and require destination IP addresses to forward encapsulated tunnel packets. EX-1004, [0032], [0076]-[0079]; EX-1005, [0066]-[0067], FIG. 6.

119. A POSITA understood that a forwarding path over a PW according to GRE would involve the VSI on the PE device *obtain[ing] an L3 address of a destination tunnel end point of the packet from the destination virtual port* to be able to add labels to *transmit[] the processed packet*. EX-1004, [0039].

Claim 4

[4.1] The apparatus according to claim 1, wherein the tunnel packet end point obtains a VNID (Virtual Network ID) of the tunnel packet and L3 address information of a sending tunnel end point from the tunnel header of the tunnel packet,

120. Ye discloses this limitation.

121. Ye discloses that a packet, arriving “from an inbound VC LSP of a PW,” “may include a tunnel label (an outer label), a PW label (an inner label), and an Ethernet frame.” EX-1004, [0036]. Ye further discloses that the Ethernet frame includes “a VLAN tag with a VLAN ID” [*wherein the tunnel packet end point obtains a VNID (Virtual Network ID) of the tunnel packet*]. EX-1004, [0036]. Ye explains that the PE device [*tunnel packet end point*] “removes the tunnel label to expose the PW label” [*L3 address information of a sending tunnel end point from the tunnel header of the tunnel packet*]. EX-1004, [0036]. A POSITA understood that a tunnel label associated with tunnels operating over an L3 network, such as GRE tunnels, would include L3 address source information such that a PE device would *obtain[] L3 address information of a sending tunnel end point from the tunnel header of the tunnel packet* by receiving the packet and removing the tunnel label. EX-1004, [0036].

[4.2] processes the tunnel packet such that the L2 switching may be performed, and

122. As discussed above, the PE of Ye is configured to *process the tunnel packet such that the L2 switching may be performed* (see [1.2]).

[4.3] transmits the processed packet to the domain VSI corresponding to the VNID through a virtual port associated with the sending tunnel end point.

123. Ye further discloses that the PE device “removes the tunnel label to expose the PW label [and] [b]ased on the PW label, the PE device maps the packet to a VSI of a particular VPLS instance and its MAC address table” [*transmits the processed packet to the domain VSI*]. EX-1004, [0036]. As discussed above in [4.1], the Ethernet frame beneath the PW label includes “a VLAN tag with a VLAN ID” that corresponds to the VPLS instance and its VSI [*corresponding to the VNID*]. EX-1004, [0036]. As discussed above in [2.3], because PWs and ACs are virtual circuits that serve as the input paths through which packets reach the VSI, and the VSI—as “an Ethernet bridge function entity”—receives and switches those packets at L2, a POSITA understood that a PW or AC functions as an *input virtual port* to the VSI. EX-1004, [0029]-[0031], [0036], [0143]. Thus, the processed packet is transmitted to the domain VSI corresponding to the VLAN tag through a virtual port associated with the sending PW using the PW label [*transmits the processed packet to the domain VSI corresponding to the VNID through a virtual port associated with the sending tunnel end point*]. EX-1004, [0031], [0036].

Claim 5

[5] The apparatus according to claim 1, wherein the tunnel packet generator is configured, for the packet for which the L2

switching has been performed, to generate a VN header (virtual network header) including a VNID (Virtual Network ID) corresponding to the domain VSI and a L3 tunnel header based on a destination tunnel end point L3 address obtained from a destination virtual port of the packet for which the L2 switching has been performed, and to combine the VN header and the L3 tunnel header with the processed packet for which the L2 switching has been performed, and transmit the same.

124. Ye discloses this limitation.

125. As described above, Ye discloses L2 switching on a received packet *[for the packet for which the L2 switching has been performed]*, and a tunnel packet generator that adds a tunnel header to the packet for which the L2 switching has been performed, generates a new tunnel packet, and transmits the new tunnel packet *[wherein the tunnel packet generator is configured ... to generate a L3 tunnel header ... and to combine ... L3 tunnel header with the processed packet for which the L2 switching has been performed, and transmit the same]* (see [1.3] and [1.4]).

126. Ye discloses that when a packet with a known MAC address is forwarded through a PW to a virtual port on another PE device, the sending module “adds a VSI label and a tunnel label to encapsulate and forward the unicast packet based on the entry” *[to combine the VN header and the L3 tunnel header with the processed packet for which the L2 switching has been performed, and transmit the same]*. EX-1004, [0039], [0096], [0128]. Ye discloses that “a unique

VSI label is allocated to every VSIs on each PE device” and that “the VSI label corresponds to all inbound VC LSPs in PWs of the same VSI.” EX-1004, [0046]. Ye further discloses that “each PE device communicates a VSI label that the opposing PE device on the other end of the PW is to use to send packets to the VSI of the PE device.” EX-1004, [0063]. Thus, the VSI label is a generated header that includes information that directly identifies and corresponds to a specific VSI, VPLS instance, VLAN tag, and VLAN ID [*to generate a VN header (virtual network header) including a VNID (virtual network ID) corresponding to the domain VSI*]. EX-1004, [0036], [0039], [0063], [0096], [0128]; *see also* EX-1007, pp. 7-16 (a POSITA understood that an encapsulating tunnel end point may add a virtual network identifier that includes a “24 bit value used to designate the individual VXLAN overlay network” and an optional outer VLAN tag to the encapsulation header) (fully incorporated into specification of EX-1015, [0033]).

127. As discussed above in [3], the tunnel label added by the sending module provides an L3 tunnel header based on a destination tunnel end point L3 address obtained from a destination virtual port of the packet for which the L2 switching has been performed [*to generate ... a L3 tunnel header based on a destination tunnel end point L3 address obtained from a destination virtual port of the packet for which the L2 switching has been performed*]. The path database and corresponding MAC address tables provide, for that destination, the selected tunnel

ID and other address information used to transmit the packet. EX-1004, [0039], [0076]-[0079]. Ye discloses that a tunnel “can be a MPLS tunnel or a generic routing encapsulation (GRE) tunnel.” EX-1004, [0032]. A POSITA understood that GRE tunnels operate over L3 networks, and that a forwarding path over a PW according to GRE would involve the sending module obtaining a destination tunnel end point L3 address from the destination virtual port information to generate the L3 tunnel header. EX-1004, [0032], [0039].

Claim 6

[6] The apparatus according to claim 1, further comprising a domain VSI manager configured to receive a VNID (virtual network ID) of a virtual network from a network virtualization manager that manages network virtualization of an entire network, and to generate the domain VSI (domain virtual switching instance) corresponding thereto.

128. Ye discloses that these limitations are implemented in software. Ye discloses that a VPLS instance [*virtual network*] is created and implemented by a *network virtualization manager*. EX-1004, [0033]-[0034], [0060]-[0061]. Ye discloses that a first MP2MP LSP connects all PE devices in the backbone network to provide a communication channel for the exchange of protocol packets. EX-1004, [0053]. Ye explains that a VPLS can be managed through manual configuration of PE devices or by a signaling protocol, such as BGP, sent over the first MP2MP LSP which then connects all PE devices in the same VPLS instance

[*network virtualization manager that manages network virtualization of an entire network*]. EX-1004, [0033]-[0034], [0053]. “Under manual or automatic configuration, each PE device allocates a PW label for an inbound VC LSP that the other PE device is to use to send packets to the PE device. The VPLS instance and the PW are identified in the signaling exchange by a VPLS ID [*VNID (virtual network ID)*] and a PW ID, respectively.” EX-1004, [0034]. As discussed in the State of the Art section, a POSITA understood that this would be accomplished by an external network manager. EX-1017, [0049], [0053].

129. Ye also discloses that “[e]ach PE device of a VPLS instance establishes a VSI for that VPLS instance” [*generate the domain VSI (domain virtual switching instance) corresponding thereto*]. EX-1004, [0035], [0060]-[0061]. For example, Ye discloses that “PE devices 108 create VSIs 112 that are part of VPLS instance 107, associate VSIs 112 with VPLS instance 107 (for example to its VPLS ID), and allocate VSI labels 118, 122, and 124 (FIG. 1) to VSIs 112-1, 112-2, and 112-4 (FIG. 1), respectively.” EX-1004 [0061]. Each VSI on a PE device maintains a MAC address table and supports MAC address learning and aging, and packet forwarding and flooding. EX-1004, [0035]. As illustrated in FIG. 18, Ye further discloses each PE device contains modules that create and maintain the VSI [*domain VSI manager*], including a determination module and path data-management module. EX-1004, FIG. 18, [0129]-[0136]. When a PE

device receives a notification packet containing a VPLS ID, the determination module first determines whether the PE device has a VSI associated with the same VPLS instance, and, if so, the path data management module configures the VSI by adding a new entry that “record[s] the relationship among the MAC address, the VSI label, and the selected tunnel to the path database of the VSI” [*a domain VSI manager configured to receive a VNID (virtual network ID) of a virtual network ... and to generate the domain VSI (domain virtual switching instance) corresponding thereto*]. EX-1004, [0129-0136]. Ye discloses that the PE device creates a VSI for an VPLS instance and configures the VSI to contain path data recording the relationship to other PE devices in the VPLS instance using PE device modules, such as the path data management module, that respond to VPLS IDs, thus a POSITA understood that Ye discloses *a domain VSI manager configured to receive a VNID (virtual network ID) of a virtual network ... and to generate the domain VSI (domain virtual switching instance) corresponding thereto*.

130. Ye’s disclosure of a *domain VSI manager* is consistent with Patent Owner’s assertion in its infringement contentions that a domain VSI manager exists because “the system” is able to create a bridge domain on a physical switch for a particular virtual network instance based on a VNID. EX-1016, p. 57.

Claim 7

[7.0] An operating method of a network virtualization apparatus, the method comprising:

131. Ye discloses this limitation for the same reasons as discussed with respect to limitation [1.0].

[7.1] collecting tunnel end point information within a network virtualization domain, and connecting each tunnel end point with a tunnel based on the tunnel end point information collected;

132. Ye discloses this limitation for the same reasons as discussed with respect to limitation [1.1].

[7.2] receiving a tunnel packet and processing the packet such that a L2 switching may be performed;

133. Ye discloses this limitation for the same reasons as discussed with respect to limitation [1.2].

[7.3] performing the L2 switching on the processed packet;

134. Ye discloses this limitation for the same reasons as discussed with respect to limitation [1.3].

[7.4] adding a tunnel header to the packet for which the L2 switching has been performed to generate a new tunnel packet, and transmitting the new tunnel packet; and

135. Ye discloses this limitation for the same reasons as discussed with respect to limitation [1.4].

[7.5] connecting tunnel end points within the network virtualization domain to other, tunnel end points within another network virtualization network domain with a unicast tunnel using the new tunnel packet.

136. Ye discloses this limitation. Ye discloses that a PE device is connected to one or more CE devices and that all devices are part of a VPLS instance [*network virtualization domain*]. EX-1004, [0048]-[0049]. Ye explains that PE devices contain VSIs that are connected to each other via PWs, and that the PWs are carried through tunnels, which may carry multiple PWs [*connecting tunnel end points within the network virtualization domain to other tunnel end points within another network virtualization domain*]. EX-1004, [0028], [0031]-[0032], [0050]. Ye discloses that a PE device will create a tunnel for a VSI on the PE device to communicate with a VSI on a different PE device and will store the tunnel ID in the path database, which is contained in the MAC address table in the VSI on the PE device. EX-1004, [0053], [0078]. Ye explains that a PE device may receive a “unicast packet from an inbound VC LSP of a PW” and that the packet may include a tunnel label which the PE device removes before “map[ping] the packet to a VSI of a particular VPLS instance and its MAC address table.” EX-1004, [0036]. Ye further explains that if the “MAC destination address of the packet is known, the PE device forwards the packet to an AC or a PW specified in the entry. When the packet is forwarded through a PW, the PE device adds a PW label and a [new] tunnel label to encapsulate and forward the packet.” [*with a unicast tunnel using the new tunnel packet*]. EX-1004, [0039].

Claim 8

[8.1] The method according to claim 7, wherein the performing the L2 switching comprises: obtaining L2 SA (source address) information for the processed packet such that the L2 switching may be performed;

137. Ye discloses this limitation for the same reasons as discussed with respect to limitation [2.1].

[8.2] determining whether or not L2 DA (destination address) information of the processed packet is a unicast address already learned; and

138. Ye discloses this limitation for the same reasons as discussed with respect to limitation [2.2].

[8.3] performing a flooding based on a type of a input virtual port, when the L2 DA information is not the unicast address already learned, or indicates multicast or broadcast address,

139. Ye discloses this limitation for the same reasons as discussed with respect to limitation [2.3].

[8.4] wherein the flooding is performed on only domain virtual ports within a domain VSI when the type of the input virtual port being an edge virtual port, and the flooding is performed on only edge virtual ports within the domain VSI when the type of the input virtual port being a domain virtual port.

140. Ye discloses this limitation for the same reasons as discussed with respect to limitation [2.4].

Claim 9

[9.1] The method according to claim 8, wherein the determining whether or not the L2 DA (destination address) information of

the processed packet is a unicast address already learned further comprises:

141. Ye discloses determining whether or not the L2 DA (destination address) information of the processed packet is a unicast address already learned further comprises for the same reasons as discussed with respect to Claim 8.

[9.2] in response to the L2 DA information being the unicast address already learned, transmitting the processed packet to a destination virtual port corresponding to the unicast address, and obtaining the L3 address of destination tunnel end point of the packet from the destination virtual port.

142. Ye discloses this limitation for the same reasons as discussed with respect to Claim 3.

Claim 10

[10.1] The method according to claim 7, wherein the receiving the tunnel packet and processing the packet such that the L2 switching may be performed comprises: obtaining a VNID (Virtual Network ID) of the tunnel packet and L3 address information of a sending tunnel end point from the tunnel header of the tunnel packet;

143. Ye discloses this limitation for the same reasons as discussed with respect to limitation [4.1].

[10.2] processing the tunnel packet such that the L2 switching may be performed; and

144. Ye discloses this limitation for the same reasons as discussed with respect to limitation [4.2].

[10.3] transmitting the processed packet to the domain VSI corresponding to the VNID through a virtual port associated with the sending tunnel end point.

145. Ye discloses this limitation for the same reasons as discussed with respect to limitation [4.3].

Claim 11

[11] The method according to claim 7, wherein the adding a tunnel header to the packet for which the L2 switching has been performed to generate a new tunnel packet, and transmitting the new tunnel packet comprises: generating, for the packet for which the L2 switching has been performed, a VN header (Virtual Network Header) including a VNID (Virtual Network ID) corresponding to a domain VSI, and an L3 tunnel header based on a destination tunnel end point L3 address obtained from a destination virtual port of the packet for which the L2 switching has been performed; and combining the VN header and the L3 tunnel header with the packet for which the L2 switching has been performed, and transmitting the same.

146. Ye discloses this limitation for the same reasons as discussed with respect to Claim 5.

Claim 12

[12] The method according to claim 7, prior to collecting tunnel end points information within a network, and connecting each tunnel end point with a tunnel based on the tunnel end point information collected, further comprising receiving a VNID (Virtual Network ID) of the virtual network from a network virtualization manager that manages network virtualization of an entire network and generating a domain VSI (Domain Virtual Switching Instance) corresponding thereto.

147. Ye discloses this limitation for the same reasons as discussed with respect to Claim 6.

Claim 13

[13.0] A network virtualization apparatus comprising:

148. Ye discloses this limitation for the same reasons as discussed with respect to limitation [1.0].

[13.1] a tunnel manager configured to collect tunnel end point information within a network, and to connect each tunnel end point with a tunnel based on the tunnel end point information collected;

149. Ye discloses this limitation for the same reasons as discussed with respect to limitation [1.1].

[13.2] a tunnel packet end point configured to receive a tunnel packet and process the packet such that an L2 switching may be performed, and to transmit the packet to a domain VSI (virtual switching instance);

150. Ye discloses this limitation for the same reasons as discussed with respect to limitation [1.2].

[13.3] the domain VSI configured to perform the L2 switching on the processed packet, and to transmit the packet to a tunnel packet generator; and

151. Ye discloses this limitation for the same reasons as discussed with respect to limitation [1.3].

[13.4] the tunnel packet generator configured to add a tunnel header to the packet for which the L2 switching has been performed to generate a new unicast tunnel packet, and to transmit the new unicast tunnel packet.

152. Ye discloses this limitation for the same reasons as discussed with respect to limitation [1.4].

153. Ye further discloses that the generated packet can be a *new unicast packet*. For example, Ye discloses forwarding a unicast packet over a PW encapsulated with a tunnel label. EX-1004, [0128]-[0129].

CONCLUSION AND DECLARATION

The findings and opinions set forth in this declaration are based on my research and analysis to date. I may continue my examinations in view of additional documents or other factual evidence over the course of this proceeding that may necessitate supplementing and/or refining my opinions. I reserve the right to add to, alter, or delete my opinions and my Declaration upon discovery of additional information. I reserve the right to make such changes as necessary.

In signing this declaration, I understand that the declaration will be filed as evidence in a contested case before the Patent Trial and Appeal Board of the United States Patent and Trademark Office. I acknowledge that I may be subject to cross-examination in the case and that cross-examination will take place within the United States. If cross-examination is required of me, I will appear for cross-examination within the United States during the time allotted for cross-examination.

I declare that all statements made herein on my own 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.

[SIGNATURE ON FOLLOWING PAGE]

Dated: March 27 2026

By: A. L. Narasimha Reddy

Dr. A.L. Narasimha Reddy

APPENDIX A – MATERIALS RELIED UPON

<i>Exhibit #</i>	<i>Description</i>
1001	U.S. Patent No. 10,020,961 (“the ’961 Patent”)
1002	Prosecution History of the ’961 Patent (“File History”)
1003	<i>Intentionally Left Blank</i>
1004	U.S. Patent Application Publication No. 2015/0146727 to Ye et al. (“Ye”)
1005	U.S. Patent Application Publication No. 2009/0099602 to Nagapudi et al. (“Nagapudi”)
1006	U.S. Patent Application Publication No. 2008/0247406 to Figueira et al. (“Figueira”)
1007	Internet Engineering Task Force (IETF) Internet-Draft, titled VXLAN: A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks (draft-mahalingam-dutt-dcops-vxlan-00) by Mahalingam et al. (“Mahalingam”)
1008	U.S. Patent Application Publication No. 2013/0311637 to Kamath et al. (“Kamath”)
1009	U.S. Patent Application Publication No. 2014/0086253 to Yong et al. (“Yong”)
1010	U.S. Patent No. 8,166,205 to Farinacci et al. (“Farinacci”)
1011	U.S. Patent Application Publication No. 2013/0250951 to Koganti et al. (“Koganti”)
1012	Internet Engineering Task Force (IETF) Internet-Draft, titled Overlay Transport Virtualization (draft-hasmit-otv-04), by Grover et al. (“Grover”)
1013	U.S. Patent No. 7,649,904 to Ghosh et al. (“Ghosh”)
1014	New H3C Technologies Co., Ltd., VPLS Technology White Paper-6W100 (last updated Apr. 27, 2025)
1015	U.S. Patent Application Publication No. 2013/0266019 to Qu et al. (“Qu”)
1016	OptimNet’s December 5, 2025 Infringement Contentions Patent Infringement Chart for U.S. Patent No. 10,020,961 from <i>OptimNet LLC v. Cisco Systems, Inc.</i> , No. 2-25-cv-00935-JRG (E.D. Tex. case filed Sept. 4, 2025)
1017	U.S. Patent Application Publication No. 2013/0318219 to Kancherla et al. (“Kancherla”)

APPENDIX B – CURRICULUM VITAE

NARASIMHA REDDY
Dept. of Elec. & Comp. Engg.
Texas A & M University
333J Wisenbaker, College Station, TX 77843-3259
<https://cesg.tamu.edu/faculty/narasimha-reddy/>
<https://scholar.google.com/citations?user=en&user=NbHZDW4AAAAJ>
reddy@tamu.edu, 979-845-7598 (O), 979-845-2630 (FAX)

Educational Record:

University of Illinois (Urbana) : Ph.D. in Electrical and Computer Engineering. Aug. 1990.

University of Illinois (Urbana) : M.S. in Electrical and Computer Engineering. May 1987.

Indian Institute of Technology (Kharagpur, India) : Bachelor of Technology (Honors) in Electronics and Electrical Communication Engineering. May 1985.

Thesis Topic:

Ph.D Thesis : Issues in IO architectures for high-performance/ multiprocessor systems.

M.S. Thesis : Algorithm based Fault Detection Techniques for Specific Applications.

B.Tech Thesis (Project) : Studies on Residue Number System for Signal Processing

Honors and Awards:

Cited for one of the most influential papers of first ACM Multimedia Conference

Catedra de Excelencia, University of Carlos III, Madrid, Spain, Aug. 2009-Feb.2010.

Distinguished Achievement Award for teaching (College level), AFS, TAMU, 2005.

Ford Fellow, College of Engg., TAMU 2004, TEES Fellow, College of Engg., TAMU, 2001, Neely Faculty Fellow, College of Engg., TAMU, 1999.

Outstanding Professor Award, Dept. of ECE, TAMU, 2003-04, 2014-15.

Outstanding Professor Award, IEEE Student branch, TAMU, 1997-1998.

NSF Career Award (1996-2000).

5 Patents and an IBM Plateau Invention Achievement Award.

IBM Graduate Fellowship (1988-90).

Presidential Silver Medal, Indian Institute of Technology, Kharagpur, India (May 86).

Research Interests:

Storage Systems, Cybersecurity, Computer Networks and Computer Architecture.

Work Experience:

Associate Dean for Research, College of Engineering, Texas A&M Univ., Apr. 2016 -Present

Associate Agency Director for Strategic Initiatives and Centers, TEES, Aug. 2016 - Present.

Assistant Agency Director for National and Global Initiatives, TEES, June 2014 - Apr. 2016.

J.W. Runyon Professor, Dept. of ECE., Texas A&M University, Oct. 2008-Present.

Professor, Dept. of Elec. and Compu. Engg, Texas A & M University, Sep. 2004 -Sep. 2008.

Associate Professor, Dept. of Elec. Engg, Texas A & M University, Aug. 1995 - Aug. 2004.

Research Staff Member, IBM Almaden Research Center: Aug. 1990 - Aug. 1995.

Professional Activities:

- **Fellow** of the IEEE Computer Society, member of ACM SIGARCH.
- Associate Editor, IEEE/ACM Transactions on Networking, 2014-2016.
- Member, Advisory Board of Center for Hyper Scale Micro Data Center, South Korea.
- External Evaluator, Faculty Promotion Committee, University of Jordan, 2012.
- Program Committee Member, INFLOW Workshop, 2016.
- Program Committee Member, NVM Workshop, 2011, 2015, 2016.
- Program Committee Member, IEEE ANTS Conf., 2014.
- Program Committee Member, Parallel Data Storage Workshop, 2012-2014.
- Member, Information Technology Research Academy Strategy Formulation, India, Oct. 2011
- Review Chair, Los Alamos National Lab Research Information Institute program, Apr. 2011
- Program Committee Member, USENIX HotStorage Conf., 2010-2012.
- External Evaluator, Faculty Search Committee, ICS-FORTH, Crete, Greece, 2009
- Program Co-chair, IEEE Broadnets 2008
- Area Lead, IEEE Infocom 2008
- Program Committee Member, USENIX FAST Conf. 2009
- Panels Co-Chair, COMSWARE 2008
- Program Committee Member, ICDCS 2008, 2013
- Program Committee Member, IEEE ICNP 2008
- Program Committee Member, COMSNETS 2009-2013, 2015
- Program Committee Member, COMSWARE 2007-2008
- Program Committee Member, Int. Workshop Visualization for Computer Security, 2006
- Program Committee Member, CoNEXT Conference, 2006, 2007, 2009.
- Program Committee Member, Int. Workshop on QOS, 2006-2011.
- Program Committee Member, IEEE Infocom, 2005-2015.
- External Reviewer, Faculty promotion Committee, KAIST, Korea, 2003.
- Technical Consultant, EMC Corp., 2001-2003.
- Educational Content Expert on Computer Networking, Jones International University.

- Member, Technical Advisory Board, Netgeo Inc., Mountain View.
- Technical Adviser, TBD Networks.
- Proposal Reviewer, Netherlands NSF.
- Program Committee member, ACM Multimedia Conference, 2002.
- Program Committee member, MCW-2001, Multimedia on Clusters of Workstations, 2001
- Program Committee member, IEEE Int. Conf. on Multimedia Comp. and Systems, 1999
- Panel Chair, IEEE Conf. High Performance Computer Architecture, 1999.
- Program Committee member, High Performance Computing, 1999
- Program Committee member, Int. Conf. on Parallel Processing, 1998.
- Tutorial Chair, IEEE Conf. on Multimedia Computing Systems, 1998.
- Panel Chair, Int. Conf. on Parallel Processing, 1995.
- Program Committee member, Int. Conf. on Parallel Processing, 1995.
- Program track coordinator, Hawaii Int. Conf. on Systems and Sciences, 1994.

Patents

- N. Annapareddy, C-C Chou, C. Tirumalasetty, P. Gratz and A. Abouelwafa, "Faster Page Handling in Computer Systems", Provisional Patent Application, May 2020.
- N. Annapareddy, J. Brady, D. Finney, M. Hartung, M. Ko, J. Menon, and D. Nowlen, Conflict resolution in multi-node communication network, U.S. Patent number: 6,101,194, received Aug. 8, 2000.
- N. Annapareddy, J. Brady, D. Finney, R. Freitas, M. Ko, and M. Rayfield, Method and System for message status reporting in a multi-node communication network, U.S. Patent number: 5,717,862, received Feb. 10, 1998.
- N. Annapareddy, D. Finney, M. Jenkins, L. Kessler, A. Lang, S. Liang, D. Mora, D. Plomgren, P. Urbisci and A. Walls, Interconnection network for a multi-nodal data processing system which exhibits incremental scalability, US Patent number: 5,602,044, received Feb. 11, 1997.
- N. Annapareddy, J. Brady and D. Finney, Adaptive and Dynamic message routing system for multinode wormhole networks, US Patent number: 5,602,839, Received Feb. 11, 1997.
- N. Annapareddy, J. Brady, D. Finney, R. Freitas, M. Hartung, M. Ko, N. Mendelson, J. Menon, D. Nowlen and S-Y Tzou, System and method using chaines structured queues for ordering of message delivery between connected nodes wherein unsuccessful message portion is skipped and retried, US Patent number: 5,577,211, Received Nov. 19, 1996.

Archival Publications

- Mahesh, P., Tiwari, A., Jin, C., Kumar, P.R., Reddy, A. L. N., Bukkapatnam, S., Gupta, N., and Karri, R., "A Survey of Cybersecurity of Digital Manufacturing", Accepted for publication in Proceedings of IEEE, October 2020.

- Tiwari, A., Reddy, A. L. Narasimha, and Bukkaparnam, S., “Cybersecurity assurance in the emerging Manufacturing-as-a-Service (MaaS) paradigm: A lesson from the video streaming industry, ASTM Journal on Smart and Sustainable Manufacturing Systems, SSMS-2020-0066.R1, October 2020.
- Wen, F., Qin. M.¹, Gratz, P., and Reddy, A. L. Narasimha, “Hardware memory management for future mobile hybrid memory systems”, accepted for publication in IEEE Trans. on CAD, Nov. 2020
- Chou, C-C1, Jung, J., Reddy, A. L. Narasimha, Gratz, P., and Voight, D., “Virtualize and Share nonvolatile memories in user space”, CCF Transactions on High Performance Computing, Jan 2020
- Moon, S.¹, and Reddy, A. L. Narasimha, “Does RAID improve lifetime of SSD Arrays?”, accepted for publication at ACM Transactions on Storage., Apr. 2016.
- Fedorov, V.¹, Qiu, S.¹, Reddy, A. L. Narasimha and Gratz, P., “ARI: Adaptive LLC-memory traffic management”, ACM Transactions on Architecture and Code Optimization (TACO), Dec. 2013.
- Wu, X¹, Qiu, S.¹, and Reddy A. L. N., “SCMFS: A file system for Storage Class Memory and its extensions”, accepted for publication in ACM Transactions on Storage, Jan. 2013.
- Wu, X¹ and Reddy A. L. N., “A novel approach to manage hybrid storage system”, Journal of communications, vol.7., July 2012.
- Shin, S., Gu, G., Reddy, A. L. N., “A large-scale study of Conficker”, IEEE Trans. Information Forensics and Security, June 2012.
- Yadav, S¹, Reddy, A. K.¹, Reddy, A. L. Narasimha and Ranjan, S., “Detecting algorithmically generated malicious domain names”, ACM/IEEE Trans. on Networking, 2012.
- Lee, Y-O.¹ and Reddy, A. L. Narasimha, “Algorithms for disjoint path construction for failure recovery and multipath routing”, to appear in Journal on Computer Networks, 2012.
- Garrison, J.¹, and Reddy, A. L. Narasimha, “Umbrella File System: Storage management across heterogeneous devices”, ACM Transactions on Storage, vol.5, no.1, Mar. 2009.
- Kim, S.¹, and Reddy, A. L. Narasimha, “Statistical Techniques for detecting Traffic Anomalies through Packet header data” IEEE/ACM Transactions on Networking (ToN), July 2008.
- Liu, Y.¹, and Reddy, A. L. N., ”Multihoming route control among a group of multihomed stub networks”, Journal on Computer Communications, Nov. 2007.
- Bhandarkar, S.¹, Reddy, A. L. N., Allman, M., Blanton, E., ”Improving the robustness of TCP to non-congestion events”, IETF(Internet Engineering Task Force) RFC 4653, August 2006.
- Kang, S¹., Reddy, A. L. N., “An Approach to Virtual Allocation in Storage Systems”, in ACM Transactions on Storage, Nov. 2006, Vol.2, pages:371-399.
- Kim, S¹., Reddy, A. L. N., “Image based Anomaly detection: Design, Algorithms and Evaluation”, IEEE Journal on Selected Areas of Comm., vol. 24.10, October 2006, 1942-1954.

- Bhandarkar, S.¹, Reddy A. L. N., “LTCP: A layered framework for improving TCP’s performance in high-speed networks”, in ACM CCR, Jan. 2006.
- Kwon, D.W., Ko, M., Vannucci, M., Reddy, A. L. N., and Kim, S.¹, “Wavelet methods for the detection of anomalies and their application to network traffic analysis”, Quality and Reliability Engineering International (QREI) 2006, 22, pages: 1-17.
- Reddy, A. L. N., Wyllie, J., and Wijayarathne, R.¹, “Scheduling for multimedia servers”, invited submission for the premier issue of ACM Transactions on Multimedia, March 2005. Cited for being one of the most influential papers from ACM Multimedia conference 1993.
- Kim, S.¹, and Reddy, A. L. N., “Statistical techniques for detecting traffic anomalies through packet header data”, accepted for publication in ACM/IEEE transactions on Networking, January 2007.
- Garg, A.¹, and Reddy, A. L. N., “Mitigating Denial of service attacks through QOS regulation”, in the special issue of Journal of Microprocessors and Microsystems on secure platforms, May 2004.
- Bhandarkar, S.¹, Sadry, N.¹, Reddy A. L. N., Vaidya, N., “TCP-DCR: A novel protocol for tolerating wireless channel errors”, IEEE Transactions on Mobile Computing, March 2005.
- Zhao, Z.¹, Ametha. J., Swaroop. D. and Reddy A. L. N., “A method for estimating non-responsive traffic at a router”, ACM/IEEE Trans. on Networking, Aug. 2004.
- Yeom, I.¹. and Reddy A. L. N., “Adaptive Marking for Assured Forwarding Service” IEEE Communication Letters, Oct. 2002.
- Yeom, I.¹. and Reddy A. L. N., “Providing consistent delay differentiation” accepted by IEICE Transactions on Communications, Sept. 2002.
- Ma, X.¹. and Reddy A. L. N., “Multiview Storage Systems”, IEEE Trans. on Parallel and Distributed Systems, Apr. 2003.
- Yeom, I.¹. and Reddy A. L. N., “Marking for QOS Improvement”, Elsevier Journal of Computer Communications, pages 35-50, vol. 24, no.1, Jan 2001.
- Yeom, I.¹. and Reddy, A. L. N., “Modeling TCP behavior in a differentiated services network”, ACM/IEEE Transactions on Networking, vol. 9, no.1, pages 31-46, Feb. 2001.
- Ma, G.¹, Khaleel, A.¹. and Reddy, A. L. N., “Evaluation of storage systems based on network-attached disks”, IEEE Transactions on Parallel and Distributed Systems, September 2000, 956-968.
- Wijayarathne, R.¹. and Reddy, A. L. N., “Providing QOS guarantees for disk I/O”, Proc. of ACM/Springer Journal on Multimedia Systems, Jan 2000, vol.8, no.1, pages:57-68.
- Yeom, I.¹. and Reddy, A. L. N. , “ENDE: An End-to-end Network Delay Emulator” , Journal on Multimedia Tools and Applications, Kluwer Publishers, vol. 14, no.3, June 2001.
- Reddy, A. L. N. and Upfal, E., “Real-time communication scheduling in a multicomputer video server”, *Journal on Parallel and Distributed Computing*, Sep. 1999., vol.58, pages: 425-445.

- Reddy, A. L. N. and Wyllie, J., “I/O issues in a multimedia system”, *IEEE Computer*, Mar. 1994, vol.27, no.3, pages:69-74.
- Reddy, A. L. N., Chandy, J. and Banerjee, P., “Design and evaluation of gracefully degradable disk arrays”, *Journal of Parallel and Distributed Computing*, Jan. 1993, vol.17, pages: 28-40.
- Reddy, A. L. N. and Banerjee, P., “Algorithm Based Fault Detection Techniques for Signal Processing Applications”, *IEEE Transactions on Computers*, Oct. 1990, vol.39, no.10, pages: 1304-1308.
- Reddy, A. L. N. and Banerjee, P., “Design, Analysis and Simulation of IO Architectures for Hypercube Multiprocessors” *IEEE Transactions on Parallel and Distributed Systems*, April 1990, vol.1, no.2, pages:140-151.
- Reddy, A. L. N. and Banerjee, P., “An evaluation of multiple-disk IO systems” *IEEE Transactions on Computers*, Dec. 1989, vol. 38, no.12, pages:1680-1690.
- Reddy, A. L. N. and Banerjee, P., “A Study of Parallel Disk Organizations” *ACM Computer Architecture News*, Sept. 1989.
- Abraham, J., Banerjee, P., Chen, C.Y., Fuchs, W.K., Kuo, S. Y. and Reddy, A.L.N., “Fault Tolerance Techniques for Systolic Arrays”, *IEEE Computer*, July 1987, vol.20, no.7, pages:65-85.
- Chakraborti, N.B., Soundararajan, J.S. and Reddy, A. L. N., “An Implementation of Mixed-Radix Conversion for Residue Number Applications”, *IEEE Transactions on computers*, August 1986, vol. C-35, no.8, pages:762-764.

Publication of books or book chapters

- Bhandarkar, S.¹, and Reddy, A. L. N., “TCP in Wireless Networks”, invited book chapter in *Encyclopedia on wireless and mobile communications*, CRC Press, Dec. 2007.
- I. Yeom¹ and A. L. Narasimha Reddy, “Differentiated Services”, invited book chapter in *Wiley’s Encyclopedia on Communications*, 2002.
- A. L. Narasimha Reddy and J. Wyllie, “Disk Scheduling in a Multimedia I/O System”, reproduced as a book chapter in *Readings in Multimedia Computing and Networking*, Morgan Kaufmann Publishers, Editors: Kevin Jeffay and HongJiang Zhang, Summer 2001.
- Reddy, A. L. N. and Wijayarathne R., “Scheduling for integrated service in multimedia systems”, invited book chapter in *Multimedia Handbook*, CRC Press, Dec. 1997.
- Reddy, A. L. N. and Haskin R., “Video Servers”, invited book chapter in *Communications Handbook*, CRC Press, 1996.
- Reddy, A. L. N., “Improving the interactive responsiveness in a video server”, invited book chapter in *Multimedia Information Storage and Management*, Kluwer Academic Publishers, Dec. 1996.
- Reddy, A. L. N., “Scheduling in Multimedia Systems”, invited book chapter in *Design and Applications of Multimedia Systems*, Kluwer Academic Publishers, Aug. 1995.

Rigorously Refereed Conference Publications

- Qin, M.1, Reddy, A. L. Narasimha, Gratz, P., Pitchumani, R., Ki, Y-S., “KVRAID: High Performance, Write Efficient, Update Friendly Erasure Coding Scheme for KV-SSDs”, accepted at ACM Systor Conference., June 2021.
- Wen, F., Qin, M.1, Gratz. P., and Reddy, A. L. Narasimha, “OpenMem: Hardware/Software Cooperative Management for Mobile Memory System”, accepted at ACM/IEEE Design Automation Conference, Dec. 2021.
- Wen, F., Qin, M.1, Gratz. P., and Reddy, A. L. Narasimha, “FPGA-based Hybrid Memory Emulation System”, accepted at Field Programmable Logic and applications Conference, Aug. 2021.
- Qin, M.1, Lee. J-H., Pitchumani, R., Ki, Y-S., Reddy, A. L. Narasimha, Gratz, P., “A Generic FPGA Accelerator for Minimum Storage Regenerating Codes”, Proceedings of the 25th Asia and South Pacific Design Automation Conference, January, 2020.
- Chou, C-C.,1, Chen, Y., Milojicic, D., Reddy, A. L. Narasimha, Gratz, P., “Optimizing Post-Copy Live Migration with System-Level Checkpoint Using Fabric-Attached Memory, 2019 IEEE/ACM Workshop on Memory Centric High Performance Computing (MCHPC), Denver, CO, USA, 2019, pp. 16-24.
- Chou, Chih Chieh¹, Jung, J¹, Reddy, A.L. Narasimha, Gratz, Paul V., and Voight, Doug, “vNVML: An Efficient User Space Library for Virtualizing and Sharing Non-Volatile Memories”, IEEE 35th International Conference on Mass Storage Systems and Technology, 2019.
- Zheng, Z.¹, Webb, A.¹, Reddy, A. L. Narasimha, Bettati, Riccardo, “IoTAegis: A Scalable Framework to Secure the Internet of Things”, Invited Paper at IEEE ICCCN, July 2018.
- Zheng, Z.1, Reddy, A. L. Narasimha, “Safeguarding Building Automation Networks: THE-Driven Anomaly Detector Based on Traffic Analysis”, IEEE ICCCN (Invited Paper), July 2017.
- Fedorov, V¹., Kim, J., Qin, M¹., Reddy, A. L. Narasimha and Gratz, P., “Speculative Paging for future NVM and SSD”, Proc. of MEMSYS Conference, Oct. 2017.
- Zhang, Z¹, Jin, S., Bettati, R., Reddy, A. L. Narasimha, “Securing Cyber-Physical Systems with Adaptive Commensurate Response”, Proc. of IEEE CNS, Oct. 2017.
- Zhang, Z¹, and Reddy, A. L. Narasimha, “Safeguarding Building Automation Networks: THE-Driven Anomaly Detector Based on Traffic Analysis”, Invited Paper at ICCCN 2017, July 2017.
- Zhang, Z¹, and Reddy, A. L. Narasimha, “Towards Improving Data Validity of Cyber-Physical Systems through Path-Redundancy”, 3rd ACM Cyber-Physical Systems Security Workshop, April 2017.
- Kim, J., Pugsley, S., Gratz, P., Reddy, A. L. N., Wilkerson, C., and Chisti, Z., “Path Confidence based Lookahead Prefetching”, MICRO Conf., Oct. 2016. **Best Paper Nomination.**
- Webb, A.¹ and Reddy, A. L. Narasimha, “Finding proxy users at the service using anomaly detection”, IEEE Conf. on Communications and Network Security, Oct. 2016.

- Moon, S¹., and Reddy, A. L. Narasimha, “Adaptive policies for balancing performance and lifetime of mixed SSD arrays through workload sampling”, IEEE Mass Storage Symposium, April 2016
- Fedorov, V.¹, Reddy, A. L. Narasimha and Gratz, P., “Shared Last-Level Caches and The Case for Longer Timeslices”, MEMSYS Conference, Oct. 2015
- Kim, J., Fedorov, V.¹, Gratz, P., and Reddy, A. L. Narasimha, “Dynamic Memory Pressure Aware Ballooning”, MEMSYS Conference, Oct. 2015
- Fedorov, V.¹, Qiu, S.¹, Reddy, A. L. Narasimha and Gratz, P., “ARI: Adaptive LLC-memory traffic management”, HiPEAC Conf., Jan. 2014.
- Moon, S¹ and Reddy, A. L. Narasimha, “Don’t let RAID raid the lifetime of your SSD array”, Proc. of USENIX HotStorage Conf., June 2013.
- Qiu, S¹, and Reddy, A. L. Narasimha, “NVMFS: A hybrid file system for improving random write in NAND-flash SSD”, Proc. of Mass Storage Systems Symposium, May 2013.
- Lee, Y-O¹, and Reddy, A. L. Narasimha, “Multipath Routing for Reducing Network Energy”, IEEE Greencom Conf., Sep. 2012.
- Qui, S¹., Reddy, A. L. Narasimha, “Exploiting superpages in a nonvolatile memory file systems”, IEEE Mass Storage Symposium, Apr. 2012.
- Moon, S¹., Reddy, A. L. Narasimha, “Write amplification due to ECC on flash memory or leave those bit errors alone”, IEEE Mass Storage Symposium, Apr. 2012.
- Wu, X¹., Reddy, A. L. Narasimha, “SCMFS: A file system for storage class memory”, ACM/IEEE Supercomputing Conf. SC11, Nov. 2011.
- Yadav S.¹, Reddy, A. L. Narasimha, “Winning with DNS Failures: Strategies for faster botnet detection”, SecureComm Conf., Sep. 2011.
- Lee, Y-O¹, Rengarajan, B., and Reddy, A. L. Narasimha, “Reaching approximate Wardrop equilibrium at reduced cost of link state updates”, COMSNETS Conf., Jan 2011.
- Qian, B¹., and Reddy, A. L. Narasimha, “Measurement and performance study of PERT for on-demand video streaming”, Pfdnet workshop, Nov. 2010.
- Yadav, S¹, Reddy, A. K.¹, Reddy, A. L. Narasimha and Ranjan, S., “Detecting algorithmically generated malicious domain names”, Proc. of Internet Measurement Conference, Nov. 2010.
- Wu, X¹., Reddy, A. L. Narasimha, “Exploiting concurrency to improve latency and throughput in a hybrid storage system”, **Best Paper Nomination**, IEEE MASCOTS Conf., Aug. 2010.
- Devakota, P¹., Reddy, A. L. Narasimha, “Performance of Quantized Congestion Notification in TCP Incast Scenarios of Data centers”, IEEE MASCOTS Conf., Aug. 2010.
- Lee, Y-Oh¹., Reddy, A. L. Narasimha, “Disjoint Multi-Path Routing and Failure Recovery”, Proc. of ICC, May 2010.

- Yadav, S¹., Reddy, A. L. Narasimha, “DNS Anomaly detection with NS and AS information”, IASTED Conf. on Network Security, Nov. 2010.
- Wu, X¹., Reddy, A. L. Narasimha, “Managing storage space in a flash and disk hybrid storage system”, IEEE MASCOTS Conf., Oct. 2009.
- Podduturi, V., Kim, Y., Shakkottai, S., Reddy, A. L. Narasimha, “MultiTrack: A delay and cost aware P2P Overlay architecture”, ACM Sigmetrics, June 2009.
- Kotla, K¹., Reddy, A. L. N., “Making a delay-based protocol adaptive to heterogeneous environments”, Proc. of IwQOS, June 2008.
- Kang, S¹., Reddy, A. L. N., “User-centric data migration in networked storage systems”, Proc. of IPDPS, April 2008.
- Bhandarkar, S¹., Reddy, A. L. N., Zhang, Y., Loguinov, D., “Emulation of AQM at end hosts”, Proc. ACM Sigcomm, Aug. 2007.
- Bhandarkar, S¹., Reddy, A. L. N., “Robustness to packet reordering in high-speed networks”, Proc. of Pfdnet, Feb. 2007.
- Al-Najjar, C¹., Reddy, A. L. N., “A service provider’s approach to improving performance of aggregate voice-over-IP traffic”, Proc. of IWQOS, June 2006.
- Kim, S¹., Reddy, A. L. N., “Netviewer: A Network traffic visualization and analysis tool”, Proc. of USENIX LISA Conference, Dec. 2005, 185-196.
- Liu, Y¹, and Reddy A. L. N., “Routing among a group of multi-homed stub networks”, Proc. of Globecom, Dec. 2005.
- Bhandarkar, S¹., Reddy, A. L. N., “LTCP: A layered approach to TCP for high-speed networks”, PFLDNet, Feb. 2005.
- Kim, S¹., Reddy, A. L. N., “Modeling network traffic as images”, ICC, May 2005.
- Kim, S¹., Reddy, A. L. N., “A study of analyzing network traffic as images in real-time”, IEEE Infocom, March 2005.
- Kim, S¹., Reddy, A. L. N., “Real-time detection and containment of network attacks using QOS regulation”, Proc. of ICC, May 2005.
- Kang, S¹., Reddy, A. L. N., “Virtual Allocation: A scheme for flexible storage allocation” OASIS workshop at ASPLOS, Oct. 2004.
- Liu, Y¹, and Reddy A. L. N., “A fast rerouting scheme for OSPF/IS-IS networks”, Proc. of ICCCN, Oct. 2004.
- Achanta, P.¹, and Reddy A. L. N., “LRU-FQ: Design and Evaluation of a Partial State Router”, Proc. of ICC 2004, June 2004.
- Zhao, Z¹., Reddy, A. L. N., “Impact of bandwidth-delay product and non-responsive flows on queue management schemes”, Proc. of ICC 2004, June 2004.
- Bhandarkar, S¹., Reddy, A. L. N., “TCP-DCR: Making TCP robust to non-congestion events”, Proc. of Networking 2004, May 2004.

- Kim, S¹., Reddy, A. L. N., and Vannucci, M., “Detecting traffic anomalies at source through aggregate analysis of packet header data”, Proc. of Networking 2004, May 2004.
- Kim, S¹., Reddy, A. L. N., and Vannucci, M., “Detecting traffic anomalies using discrete wavelet transform”, Proc. of Int. Conf. on Information Networking, Feb. 2004.
- Ma, X¹., and Reddy, A. L. N., “SPIRAL: A novel third party transfer scheme for network attached devices”, IEEE Mass Storage Symposium, April 2003.
- Zhao, Z¹., Ametha. J., Swaroop. D. and Reddy A. L. N., “A method for estimating non-responsive traffic at a router”, ACM Conf. on Sigmetrics, June 2002.
- Garg, A¹., and Reddy, A. L. N., “Mitigating Denial of service attacks through QOS regulation”, IwQoS workshop, May 2002.
- Wijayarathne, R¹. and Reddy, A. L. N., “System support for providing integrated services from networked multimedia storage servers”, ACM Conf. on Multimedia, October 2001, 270-279.
- Smitha¹ and Reddy, A. L. N., “LRU-RED: An active queue management scheme to contain high bandwidth flows at a congested router”, in IEEE Globecom Conference, Nov. 2001.
- Smitha¹, Kim, I¹., and Reddy, A. L. N., “Identifying long term high rate flows at a router”, in Proc. of High Performance Computing, Dec. 2001.
- Yeom, I¹., and Reddy, A. L. N., “Adaptive marking for aggregated flows”, in Proc. of IEEE Globecom, Nov. 2001.
- Gopalakrishnan, S¹. and Reddy, A. L. N., “SACRIO: An active buffer management scheme for Differentiated Service Networks”, Proc. of NOSSDAV, June 2001.
- Ma, X¹. and Reddy, A. L. N., “MVSS: MultiView Storage Systems”, Proc. of Int. Conf. on Distributed Computer Systems (ICDCS), Apr. 2001.
- Khaleel, A¹. and Reddy, A. L. N., “Evaluation of Request and data distribution policies in a clustered server”, Proc. of High Performance Computing, Dec. 1999.
- Yeom, I¹. and Reddy, A. L. N., “Realizing throughput guarantees in a differentiated services network”, Proc. of IEEE Int. Conf. on Multimedia Computing and Systems, June 1999.
- Wijayarathne, R¹., Reddy, A. L. N., “Integrated QOS management for disk I/O”, Proc. of IEEE Int. Conf. on Multimedia Computing and Systems, June 1999.
- Yeom, I¹. and Reddy, A. L. N., “Impact of marking strategy on aggregated flows in a differentiated-services network”, Proc. of IEEE International Workshop on QOS (IwQOS 99), June 1999.
- Tong, D¹. and Reddy, A. L. N., “QOS Enhancement with Partial Sate”, Proc. of IEEE International Workshop on QOS (IwQOS 99), June 1999.
- Gupta, S¹. and Reddy, A. L. N., “IPRP: A Client-oriented IP Redirection mechanism”, Proc. of IEEE Infocom Conf., March 1999.
- Wijayarathne, R¹. and Reddy, A. L. N., “Techniques for improving the throughput of VBR streams”, Proc. of ACM/SPIE Multimedia Computing and Networking, Jan. 1999.

- Ma, G¹. and Reddy, A. L. N., “An evaluation of storage systems based on network-attached disks”, Proc. of ICPP Conf., Aug. 1998.
- Reddy, A. L. N., “Effectiveness of caching policies for a web server”, Proc. of High Performance Computing, Dec. 1997.
- Reddy, A.L.N., “Caching strategies for a multimedia server”, Proc. of IEEE Int. conf. on Multimedia Computing and Systems, June 1997.
- Reddy, A. L. N., “Improving Latency in an Interactive Video server”, Proc. of SPIE conf. on Multimedia Computing and Networking, Feb. 1997.
- Reddy, A.L.N., “Movie and communication scheduling on a multiprocessor video server system”, Proc. of *IEEE Int. Conf. on Multimedia Comput. and Systems*, May. 1995.
- Reddy, A.L.N., Wyllie, J., “Disk Scheduling in a Multimedia I/O System”, Proc. of *1st ACM Conf. on Multimedia*, Aug. 1993. Included as a chapter in “A guided tour of multimedia systems and applications”, IEEE Press, 1995. Translated into Japanese by *Nikkei Electronics*, a Japanese technical magazine. To be included as “one of the most influential papers” in ACM Transactions on Multimedia, Applications and Communications.
- Chandy, J., and Reddy, A.L.N., “Failure evaluation of disk array organizations”, Proc. of *Int. Conf. on Distributed Computing Systems*, May 1993.
- Reddy, A.L.N. and Freitas, R., “Fault-tolerance of adaptive routing algorithms in multicomputers”, Proc. of *4th IEEE Symp. on Parallel and Distributed Processing*, Dec. 1992.
- Reddy, A.L.N., “A Study of IO System Organizations”, Proc. of *ACM Int. Symp. on Computer Architecture*, May 1992.
- Reddy, A.L.N., “Reads and Writes: When IOs aren’t the same”, Proc. of Hawaii Int. Symp. on Systems and Sciences, Jan. 1992.
- Reddy, A.L.N., Banerjee, P. and Chen, D.K., “Compiler Support for Parallel I/O Operations”, Proc. of *Int. Conf. on Parallel Processing*, Aug. 1991.
- Reddy, A. L. N. and Banerjee, P., “Gracefully Degradable Disk Arrays”, Proc. of *Int. Symp. on Fault Tolerant Computer Systems*, June 1991.
- Reddy, A. L. N. and Banerjee, P., “A Study of IO Behavior of Scientific Applications on A Multiprocessor”, Proc. of *ACM Int. Symp. on Computer Architecture*, June 1990
- Reddy, A. L. N. and Banerjee, P., “Performance evaluation of multiple-disk IO systems”, Proc. of *Int. Conf. on Parallel Processing*, Aug. 1989.
- Reddy, A. L. N. and Banerjee, P., “IO issues in hypercubes”, Proc. of *ACM Int. Conf. on Supercomputing*, June 1989.
- Reddy, A. L. N., Banerjee, P. and Abrham, S.G., “IO embedding in Hypercubes”, Proc. of *Int. Conf. on Parallel Processing*, August 1988.
- Reddy, A. L. N. and Banerjee, P., “A Fault-Secure Dictionary Machine”, Proc. of *3rd International Conference on Data Engineering*, Feb. 1987.

Other Publications

- Kim, J., Gratz, P., and Reddy, A. L. Narasimha, “ Lookahead Prefetching with Signature Path”, ACM ISCA Workshop on Data Prefetch Competition, June 2015
- Huang, B¹., Wang. S¹., and Reddy, A. L. Narasima, “Who is Hiring Whom: A New Method in Measuring Graduate Programs”, ASEE conference, June 2015.
- Singh, A.¹, and Reddy, A. L. Narasimha, “Multi path PERT”, **Invited Paper** at Int. Conf. on Computer Communications and Networks, July 2013.
- Amaleswaram, A.¹, Reddy, A. L. N., Yadav, S.¹, Gu. G., Yang, C., “CATS: Characterizing Automation of Twitter Spammers”, **Invited Paper** at Int. COMSNETS conf., Jan. 2013.
- Wu, X.¹, and Reddy, A. L. N., “Data organization in a hybrid storage system”, **Invited Paper** at Int. Conf. on Computing, Networking and Communications, ICNC’12, Feb. 2012.
- Yin, Z.¹, Hasari, C., Al-Nuweiri, H., and Reddy, A. L.N., “Improving the Performance of Delay Based Protocol in Delivering Real Time Media via Early Retransmission”, IEEE ICT Conf., Apr. 2011.
- Yin, Z.¹, Al-Nuweiri, H., and Reddy, A. L.N., “Performance of Early retransmission scheme in streaming media”, IEEE ICT Conf., Apr. 2010.
- Kim, S¹., and Reddy, A. L. N., “An evaluation of the effectiveness of measurement-based anomaly detection techniques”, ADSN workshop, held in conjunction with ICDCS, June 2006.
- Vellanki, S¹., and Reddy, A. L. N., “Improving service availability during link failure transients through alternate routing”, Tech. Report, Texas A & M University, Feb. 2003.
- Ma, X¹., and Reddy, A. L. N., “Implementation and Evaluation of an active storage system prototype”, Workshop on novel uses of System Area Networks, Feb. 2002.
- Reddy, A.L.N., “ An evaluation of multiprocessor video server system”, *IBM Confidential report*, June 1994.
- Menon, J., Reddy, A.L.N., Treiber, K. and Tzou, S.Y., “Daisy: A Novel Heirachical Storage Management System, Functional Specification”, *IBM Confidential Report*, 77 pages, Jan 1994.

Recent Presentations:

- **Tutorial** on Network Security at Int. Conference on Information Security, Hyderabad, India, Dec. 2014.
- **Invited talk** at IEEE Communications Socieity, Hyderabad, India, August 2014.
- **Invited talk** at University of Texas at San Antonio, Nov. 2013.
- Presentation at Flash Summit, Aug. 2013
- **Invited talk** at Georgia Tech., May 2013.

- **Keynote talk** at Int. Conf. on Information Technology: New Generations, Las Vegas, April 2013.
- **Invited talk** at COMSNETS Conf., Bangalore, India, Jan. 2013.
- **Invited talk** at Int. Conf. on Networking and Communications, Maui, Feb. 2012.
- **Invited talk** at NVRAMOS 2011, Jeju island, South Korea, Nov. 2011.
- **Invited talks** at Korea University, Ajou University, Samsung and ADD, Korea, Nov. 2011.
- **Keynote talk** at IEEE Int. Symp. on Computers and Communications, Italy, June 2010.
- **Catedra Series talk** at University of Carlos III, Dec. 2009, Madrid, Spain.
- Invited talk at Thomson Research, Paris, France, October 2009.
- Invited talk at Telefonica Research, Barcelona, Spain, September 2009.
- IBM Austin Research Lab **Distinguished Seminar series**, May 1 2008.
- Invited talk at University of Illinois, November 2006.
- Invited talk at Cisco, January 2005.
- Invited talk at University of Birmigham, UK, June 2004.
- Invited talk at Cisco Systems, San Jose, June 2002.
- Invited talk at Juniper Networks, Mountain View, June 2002.
- Invited talk at Univ. of California at Riverside, June 2002.
- Invited talk at Nortel Networks, Santa Clara, June 2000.
- Invited talk at Juniper Networks, Mountain View, June 2000.
- Talk at Internet Engineering Task Force, Washington D.C., Oct. 1999.
- Invited talk at EMC Corp., Boston, Aug. 1999.
- Tutorial on “Scheduling for Multimedia Applications”, presented at IEEE Int. Conf. on Multimedia Computing and Systems, June 1999.
- Tutorial on High-Performance I/O at IEEE/ACM Int. Conf on Computer Architecture, Atlanta, May 1999.
- Invited talk at AT & T Research Center, Florham Park, March 1999.
- Invited talk at Bell Labs Research Center, Murray Hill, March 1999.
- Invited talk at EMC Corp., Boston, Nov. 1997.

Funding:

- “Cooprative Research Agreement”, Army Futures Command, Co-PI, 2019-24.

- “Multi-layer Cybersecurity and Situational Awareness to Enhance Resiliency in Qatars Power Grid, Qatar National Research Foundation, 2020-2025, co-PI.
- “Protecting Building Automation Networks from insider threats”, 2016, QNRF, lead PI.
- EMC, DELL, HPE and Samsung membership participation in NSF I/UCRC Center.
- “NSF I/UCRC Center on Intelligent Storage”, 2014, NSF, lead PI.
- “Equipment Donation”, Viking Technologies, 2013.
- “Emerging memory architectures for big memory applications”, National Science Foundation, 2013-2016, co-PI.
- “Exploiting social communication channels against cyber criminals”, National Science Foundation, 2012-2015.
- “Client interfaces to object storage systems”, Argonne National Lab, 2012-2015.
- “Detecting malicious activities by analyzing social communication contexts”, Qatar National Research Foundation, 2012-2015.
- “Network traffic analysis for botnet detection”, Narus Inc., Research gift, 2009-2010.
- “Design and Analysis of new generation of protocols for triple-play networks”, QNRF, Jan. 1. 2008 - Dec. 31 2010, Co-PI.
- “Dynamic allocation and data distribution in networked storage systems”, National Science Foundation, Oct. 2007 - Sept. 2010, sole PI.
- “Active Data Systems”, National Science Foundation, Sept. 2006 -Aug. 2009, sole PI.
- NSF REU grant, co-PI, 2004-2007.
- Network Elements based on partial state, Intel Corp., gift, Feb. 2003.
- Texas A & M University, “High Availability Infrastructure Laboratory”, TITF Award, Sept. 2002 - Dec. 2003, PI with R. Bettati, M. Vannucci and D. Swaroop.
- “Secure Networked Storage Systems”, National Science Foundation, Research Instrumentation grant, Sept. 2002- Sept. 2005.
- Texas Advanced Technology Program, “Network Architectures based on partial state”, Jan. 2001 - Dec. 2002.
- “Fellowships for Computer Engineering”, Department of Education, Aug. 2001- Aug. 2004, PI with R. Mercer and K. Watson.
- “QOS Enhancement with Partial state”, National Science Foundation, June 2001-June 2004, sole PI.
- “Multiview Storage Systems”, National Science Foundation, May 2001 - May 2004, sole PI.
- Equipment Donation, 3Ware Inc., Nov. 2000.

- “Scalable request-specific file service”, National Science Foundation, Aug. 1999 - Aug. 2002, sole PI.
- Research gift, EMC Corp., 1998.
- “Scalable Multimedia Storage Server”, Texas Higher Education Board, January 1996 - Dec. 1997, sole PI.
- “Scalable Storage Server for Multimedia and Scientific Applications”, National Science Foundation, 1996-2000, sole PI.
- “Development of Novel Systems Software for Multimedia and High-performance computing”, National Science Foundation, 1997-2000, PI along with R. Mercer, P. Cantrell, G. Choi and M. Lu.
- Equipment grant, Tandem Corporation, March 1997.
- Software donation, Microsoft Corporation, April 1997.

Teaching:

Redesigned the course ECEN 449 “Microprocessor Systems Design”, Fall 2007.

Developed a new course ELEN 627 “Multimedia Systems and Networks”.

Developed a freshman undergraduate course ENGR111 “Introduction to Electrical and Computer Engineering”.

Developing a new graduate course on “Network systems security”. Taught in Spring 2004, 2006.

Courses Taught: Undergraduate Introduction to Electrical and Computer Engineering, Undergraduate Computer Architecture, Graduate Network Systems security, Graduate Computer Architecture, Graduate Computer Networks, Graduate Multimedia Storage and Delivery, Graduate Parallel Computer Systems, Graduate I/O and Memory Systems.

Student Advising:

- Chris Baldwin, Ph.D. candidate, Aug 2021-, Storage systems.
- Susav Shresta, Ph.D. candidate, Aug 2021-, Memory systems.
- Chandrahas Tirumalasetty, Ph.D. candidate, Aug 2019-present, Storage Systems.
- Rishabh Singla, Ph.D. candidate, Aug 2017-Present, Network Security.
- Mian Qin, Ph.D. candidate, Aug 2016-Aug 2021, Storage Systems.
- Chih-Chieh Chou, Ph.D., Aug 2014-Jan. 2020, “Optimizing Emerging Memory Systems for Performance”, next at Amazon.
- Zhiyuan Zheng, Ph.D., Aug. 2013 -Dec. 2017, “Protecting Cyber-Physical Systems With Special Emphasis On Building Automation Networks”, next at Pinterest.
- Jaemin Jung, Postdoc, Oct. 2016-18, Storage systems, next at Samsung Research.

- Allen Webb, Ph.D. candidate, Jan. 2013-17, “Addressing Insider Threats From Smart Devices”, next at Google.
- Viacheslav Fedorov, Ph.D., 2010 - 2016, “Memory Management for emerging memory technologies”, later at NXP.
- Jongmin Gim, Postdoc, 2013-2015, Storage Systems Software, Later at Samsung Research Labs, California.
- Sangwhan Moon, Ph.D., 2010 - 2015, “Reliability of SSD Storage Systems”, later at Riverbed Technology.
- Sheng Qui, Ph.D. 2013 “Integration of nonvolatile memory into storage hierarchy”, later at Samsung Research Labs.
- Yong Oh Lee, Ph.D., May 2012, “Improving efficiency and effectiveness of multipath routing in computer networks”, later at Samsung.
- Jagadeesh Dyaberi, Ph.D., Aug. 2011, “Networking and Storage support for on-demand video delivery”, Purdue University, External thesis committee member.
- Sandeep Yadav, Ph.D., 2012, “Scalable techniques for anomaly detection”, Dec. 2012, later at Dumballa.
- Xiaojin Wu, Ph.D., Flash and Disk Hybrid storage systems, Aug. 2011, later at Marvell Semiconductor.
- John Garrison, Ph.D. , “Umbrella File System: Managing heterogeneous storage devices File systems”, Aug. 2009, later at DOD Labs, Tuscaloosa, Alabama.
- Sukwoo Kang, Ph.D., “Flexible space allocation and management in networked storage systems”, May 2007, later at IBM Almaden Research Center.
- Sumitha Bhandarkar, Ph.D., “Congestion control algorithms for TCP in emerging networks”, Aug. 2006, later at Motorola.
- Yong Liu, Ph.D., “Dynamic Routing protocols in multihomed networks”, Dec. 2006, later at EMC Corp.
- Seong Soo Kim, Ph.D., “Real-time analysis of aggregate network traffic”, May 2005, later at Samsung, Seoul, Korea.
- Zhili Zhao, Ph.D., May 2004, “Estimation and control of nonresponsive traffic”, later at Intel.
- Xiaonan Ma, Ph.D., Aug. 2003, “Augmenting functionality at storage systems”, later at IBM Almaden Research Center, San Jose.
- Ravi Wijayarathne, Ph.D., Dec. 2000, “PRISM: A file server architecture for providing integrated services”, later at Quantum, San Jose.
- Ikjun Yeom, Ph.D., Dec. 2000, “Bandwidth assurance in differentiated services networks”, later a faculty member at KAIST, South Korea.
- Daksh Vasistha, M.S., 2015-17, “ Detecting Anomalies in Controller Area Network for Automobiles”, next at Facebook

- Ayush Ruia, M.S., 2015, “Virtualization of nonvolatile memory”, later at Microsoft.
- Bolun Huang, M.S. 2014, “Sentiment based classification of tweeters and university programs”, later at Microsoft.
- Shyam Sambasivan, M.S. “Enhancements to SQLite library to improve performance on mobile platforms”, Aug. 2013, later at Cisco.
- Ankit Singh, M.S., “Multipath Probabilistic Early Response TCP”, Aug. 2012, later at Amazon.
- Cindy Cao, M.S., “SCMFS Performance Enhancement and Implementation on Mobile Platform”, Aug. 2012, later at Microsoft.
- Amit Anand, M.S., “Proactive spammer detection on Twitter”, Aug. 2012, later at Amazon.
- Bin Qian, M.S.. Aug. 2011, “Evaluation of Probabilistic Early Response TCP (PERT) for video streaming and extension with ACK coalescing”, later at Amazon Data Services.
- Ashwath Reddy, M.S., Aug. 2010, Detecting networks employing algorithmically generated names, later at Microsoft.
- Prajjwal Devakota, M.S. Aug. 2010, Performance of Quantized Congestion Notification in TCP Incast in Data Centers, later at Bloomberg.
- Chaitanya Tadi, M.S., May 2009, IP Prefix Hijack Prevention Service, later at Intel.
- Marcus Dunn, M.S., May 2009, A new I/O scheduler for Solid State Devices, later at Cisco.
- Kiran Kotla, M.S., “Adapting the delay-based protocol PERT to heterogeneous environments”, Aug. 2008, later at Cisco.
- Camelia Al-Najjar, M.S., August 2005, thesis on “An approach for improving performance of aggregate voice over IP traffic”, pursuing Ph.D. at Case Western University.
- Saurabh Jain, M.S., May 2005, thesis on “LTCP-RC: RTT compensation technique to scale highspeed protocols to high RTT links”, later at Cisco.
- Sridhar Vellanki, M.S., Feb. 2003, thesis on “Improving service availability through alternate routing”, later at Microsoft.
- Sadry, Nauzad, M.S., Dec. 2002, thesis on “Evaluation of Delayed Congestion Response protocols for tolerating channel errors in wireless networks”, later at Netezza.
- Achanta, Phani, M.S., Dec. 2002, thesis on “Design and Evaluation of Partial State Routers”, later at IBM.
- Smitha, M.S. student, May 2001, thesis on “An active queue management scheme to contain high bandwidth flows at a congested router”, later at Intel India.
- Sumitha Bhandarkar, M.S., August 2001, thesis on “Delayed Congestion Response Protocols”, Ph.D. at TAMU.
- Aman Garg, M.S., August 2001, thesis on “Mitigating Denial of Service using QOS regulation”, later at Tipping Point Technologies, Austin.

- Sai Gopalakrishnan, M.S., May 2000, later at Cisco, thesis on “TRIO: Active resource management in differentiated services networks”.
- Qian Ren, M.S., May 2000, later at Lucent, thesis on “MPEG-aware disk storage system”.
- Marcus Grande, M.S., Dec. 1999, later at IBM, thesis on “Multiported Storage Devices”.
- Adnan Khaleel, M.S., Aug. 1999, later at Motorola, thesis on “Request and data distribution on a clustered server”.
- Deying Tong, M.S., May 1999, later at Cisco, thesis on “Partial state based approaches to QOS enhancement”.
- Sumit Gupta, M.S., Aug. 1998, later at Sun Labs, thesis on “IPRP: A client-oriented IP redirection mechanism and its applications”.
- Ashwin Madhwaraj, M.S., May 1998, later at Cisco, thesis on “Adaptive video transport over networks”.
- Gang Ma, M.S., May 1998, later at EMC Corp., thesis on “Evaluation of storage systems based on network attached storage”.
- Mike Nichols, M.S., May 1998, later at UT Law School, thesis on “Personal informal retrieval agent”.
- Ikjun Yeom, M.S., May 1998, later a faculty member at KAIST, Korea, thesis on “ENDE: An End-to-end Network Delay Emulator”.

Service:

- Member, University IT Research Committee, 2018.
- Chair, Turbo Lab Director Search Committee, 2017.
- Chair, College Cybersecurity Search Committee, 2016-Present.
- Member, University Chief Communications Officer Hiring Committee, 2015-16.
- Member, University level Assoc. Former Students Awards Committee, 2016.
- Member, TAMUQ FASC Committee, 2013.
- Chair, Department Growth Committee, 2013.
- Chair, College Taskforce on National Security, 2013.
- Member, College Research Strategy Subcommittee, 2012.
- Member, Computer Science and Engineering Head Search Committee, 2010-11, 2013-14.
- Member College Awards Committee 2006-2008. 2010-2012.
- Actively participating in improving undergraduate research experience. Co-PI of NSF REU grant. Advised 7 undergraduate students in the last few years.

- Member, Computer Engineering Curriculum Committee, 2003-2012. Led the restructuring of Undergraduate curriculum in Computer Engineering.
- Member, Department Tenure & Promotion Committee, 2004-2006, 2010-2012.
- Member, Electrical and Computer Engineering Department Head Search Committee, 2004.
- Member, Department Advisory committee 2003-2004.
- Chair and Member, department Awards committee 2004-2006.
- Member, Department Awards committee, 1997-2003, 2008-2009.
- Member, Department Distinguished Speakers committee, 1996-1999, 2006-2008.
- Chair and Member, Department Distinguished Speakers committee, 2010-present.
- Member, Department undergraduate studies committee, 1997-2000.
- Member, Department web site design committee, 2000-2004.
- Group leader, Computer Engineering, 2005-2014.

Administrative Experience:

Group Leader, Computer Engineering, 2005-2014:

- Led evaluation, development and rearchitecting the Computer Engineering program, a joint program across two departments of Electrical and Computer Engineering and Computer Science Engineering.
- Led the hiring of about 10 Computer Engineering faculty.
- Led the development of ENGR 111 Freshman Introduction of Electrical and Computer Engineering, primarily to improve retention in Electrical and Computer Engineering programs. Increased retention rates from 45% to about 80%.
- Led the development of several other courses ECEN 449 Microprocessor Systems Design, ECEN 627 Multimedia Systems and Networks, ECEN689 Computer Networks Security.
- Rebuilt several existing courses including ECEN 350 Computer Architecture and ECEN 676 Advanced Computer Architecture, ECEN 602 Computer Networks.
- Contributed to the national ranking of both undergraduate and graduate Computer Engineering programs to top 20 programs in the country.

TEES Assistant Agency Director for National and Global Initiatives, 2014- April 2016:

- Developed and Established a TEES office in Yukatan, Mexico.
- Nurtured relationships in Brazil, France and India. The France relationship resulted in a collaborative center with ENSAM.
- Established realtionships with Kuwait Oil Company which resulted in a \$15 million research contract.

- Developed a strategy to engage with National Labs and Federal Agencies.

**Associate Dean for Research, College of Engineering and TEES Associate Agency
Director for Centers and Strategic Initiatives, April 2016-Present:**

- Established strategic focus on several emerging areas of research including Autonomy, Synthetic Biology, Soft Matter, Cyber Security and Quantum Computing.
- Developed a strategy for supporting large multi-PI proposals and increased large proposal funding from 5-6 per year to about 25 per year.
- Increased TEES Research funding from about \$110 million/year to \$210 million/year from 2016-20.
- Increased Federal funding percentage from 38% to 68% of total TEES funding.
- Nurtured relationships with Army Research Lab (ARL) and facilitated the establishment of ARL-south relationship with A & M. Direct impact: ARL Cooperative Agreements in Synthetic Biology, Electric Power, Materials and Manufacturing, Autonomy and Cybersecurity with about 15 ARL people stationed at Texas A& M.
- Supported establishment of \$65 million ARL/Army Futures Command (AFC) Cooperative agreement on Hypersonics, Autonomous vehicles and Networks.
- Contributed to \$26 million Distributed Autonomy related project funded by Army Research Lab.
- Nurtured and furthered collaborations with Air Force Research Lab, resulting in the establishment of AFRL fellowships at A & M. Direct Impact: AFRL \$4 million Minority Leaders Scholarships in Material Discovery and several projects with sensors, human performance and information directorates.
- Established relationships with Army Engineering and Research Development Center (ERDC). Resulted in \$3 million contract. Exploring further options for joint projects.
- Establishing close collaborative ties with DOE/DOD National Labs. Resulted in Master Research Agreements and joint projects with several labs including Idaho National Lab(INL), Lawrence Livermore National Lab(LLNL), Pacific Northwest National Lab (PNNL) and others.
- Contributed to successful \$40 million proposal for "Offshore Energy Safety Center 2.0" proposal to Department of Interior and Department of Energy.
- Started and organized "TEES Tech. Talk" series to facilitate hosting Labs and Agency researchers and program managers.
- Organized National Labs Day to facilitate interactions of graduate students with National Labs.
- Championed "Remote Ph.D. program for national labs".
- Established "Research Impact Awards" to facilitate a discussion and recognition of impactful research within the College.

- Established External Awards Support office to promote faculty for external professional awards.
- Supported young faculty applications to NSF, DOD/DOE/DARPA CAREER and Young Investigator awards. Increased NSF CAREER awards from 2/year to 8/year.
- Established NSA Masters Program.
- Expanded a weekly "Research Opportunities" bulletin to cover broader research opportunities.
- Interacted with the Research Deans of other colleges and supported inter-college collaborations.
- Established procedures and metrics for evaluating centers within college and reorganized centers after a thorough review of the existing centers. Managed about 30 Centers and Institutes.
- Established support to help graduate student applications for NSF and NDSEG Fellowships.

¹Reddy's Students.