

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

DELL TECHNOLOGIES INC. AND DELL INC.
Petitioners

v.

CLOUD BYTE LLC
Patent Owner

Case IPR2025-01285
Patent No. 9,560,177

PETITION FOR *INTER PARTES* REVIEW

TABLE OF CONTENTS

List of Exhibits.....	vi
Mandatory Notices Under 37 C.F.R. § 42.8	ix
Real Party-in-Interest (37 C.F.R. § 42.8(b)(1))	ix
Related Matters (37 C.F.R. § 42.8(b)(2))	ix
Lead and Back-Up Counsel (37 C.F.R. § 42.8(b)(3))	ix
Service Information (37 C.F.R. § 42.8(b)(4)).....	ix
I. Introduction.....	1
II. Grounds for Standing.....	1
III. Identification of Challenge	1
IV. The Challenged Patent.....	2
A. Patent Specification.....	2
B. Prosecution History	5
C. Patent Claims.....	5
D. Person of Ordinary Skill in the Art	5
V. Claim Construction.....	6
A. “encapsulate”/“encapsulating”/“encapsulation”/ “decapsulate”	7
B. “[duplicate/duplicating] a part of a header of the identified packet as an additional header”	8
VI. Prior Art References	8
A. Rijhsinghani (EX1005)	9
B. RFC-5474 (EX1006).....	15
C. Lean (EX1007).....	16

D.	Computer Networking (EX1008).....	16
VII.	The Challenged Claims Are Invalid As Obvious Based on Rijhsinghani As a primary prior Art reference	17
A.	Ground 1: Rijhsinghani Alone Renders Claims 1, 5-7, 11-13, and 19 Obvious.....	17
1.	Claim 1	17
2.	Claim 5.....	50
3.	Claim 6.....	52
4.	Claim 7	53
5.	Claim 11	57
6.	Claim 12.....	58
7.	Claim 13.....	59
8.	Claim 19.....	60
B.	Ground 2: The Combination of Rijhsinghani and RFC-5474 Renders Claims 2-4, 8-10, 12, 14-18, and 20 Obvious.....	62
1.	Motivation to Combine	63
2.	Claim 2.....	66
3.	Claim 3.....	67
4.	Claim 4.....	68
5.	Claim 8.....	69
6.	Claim 9.....	69
7.	Claim 10.....	69
8.	Claim 12.....	69
9.	Claim 14.....	71

10.	Claim 15.....	72
11.	Claim 16.....	72
12.	Claim 17.....	72
13.	Claim 18.....	72
14.	Claim 20.....	72
C.	Ground 3: The Combination of Rijhsinghani and Lean Renders Claims 2-4, 8-10, 12, 14-18, and 20 Obvious	73
1.	Motivation to Combine	73
2.	Claim 2.....	77
3.	Claim 3.....	78
4.	Claim 4.....	78
5.	Claim 8.....	78
6.	Claim 9.....	78
7.	Claim 10.....	79
8.	Claim 12.....	79
9.	Claim 14.....	80
10.	Claim 15.....	81
11.	Claim 16.....	81
12.	Claim 17.....	81
13.	Claim 18.....	81
14.	Claim 20.....	81
D.	Grounds 4-6: Adding Computer Networking Renders All Challenged Claims Obvious.....	82
VIII.	Discretionary Denial Is Not Warranted.....	84

IX. Conclusion	85
----------------------	----

TABLE OF AUTHORITIES

FEDERAL CASES

<i>Medtronic, Inc. v. Barry</i> , 891 F.3d 1368 (Fed. Cir. 2018)	15, 16, 17
<i>Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co. Ltd. Matal</i> , 868 F.3d 1013 (Fed Cir. 2017)	6
<i>Phillips v. AWH Corp.</i> , 415 F.3d 1303 (Fed. Cir. 2005)(en banc)	6
<i>Valve Corp. v. Ironburg Inventions Ltd.</i> , 8 F.4th 1364 (Fed. Cir. 2021)	15

FEDERAL STATUTES

35 U.S.C.	
§ 102.....	1, 9, 15, 16
§ 103.....	1

FEDERAL RULES AND REGULATIONS

37 C.F.R. § 42.100	6
37 C.F.R. § 42.104	1

LIST OF EXHIBITS

Exhibit	Description
1001	U.S. Patent No. 9,560,177 (the “’177 Patent”)
1002	Prosecution history of U.S. Patent Application No. 15/072,558, which led to the issuance of the ’177 Patent
1003	Declaration of Mark Crovella, Ph.D. in Support of Petition for <i>Inter Partes</i> Review of U.S. Patent No. 9,560,177
1004	<i>Curriculum vitae</i> for Mark Crovella, Ph.D.
1005	U.S. Patent No. 6,526,052 (“Rijhsinghani”)
1006	Request for Comments 5474, “A Framework for Packet Selection and Reporting” (“RFC-5474”)
1007	U.S. Patent Application Publication No. 2008/0031141 (“Lean”)
1008	Kurose and Ross, “Computer Networking: A Top-Down Approach” (4th ed. 2008) (“Computer Networking”)
1009	OpenFlow Switch Specification (version 1.0.0)
1010	Request for Comments 1701, “Generic Routing Encapsulation (GRE)”
1011	Request for Comments 2784, “Generic Routing Encapsulation (GRE)”
1012	Request for Comments 2026, “The Internet Standards Process – Revision 3”
1013	RFC Editor – Publication Process
1014	U.S. Patent No. 6,956,854
1015	U.S. Patent No. 7,389,359
1016	U.S. Patent No. 7,660,259

Exhibit	Description
1017	U.S. Patent No. 8,724,638
1018	U.S. Patent No. 8,125,920
1019	U.S. Patent No. 7,626,938
1020	U.S. Patent No. 6,041,042
1021	U.S. Patent Application Publication No. 2004/0003094
1022	McKeown, “Fast Switched Backplane for a Gigabit Switched Router,” Business Communications Review (Dec. 1997)
1023	Partridge, et al, “A 50-Gb/s IP Router,” IEEE/ACM Transactions on Networking (June 1998)
1024	Claffy, et al., “Application of Sampling Methodologies to Network Traffic Characteristics,” ACM SIGCOMM Computer Communication Review, vol. 23, issue 4 (1993)
1025	Duffield and Grossglauser, “Trajectory Sampling for Direct Traffic Observation,” IEEE/AMC Transactions on Networking, vol. 9, no. 3 (2001)
1026	History of Packet Sampling @ sFlow.org
1027	NetFlow Configuration Guide, Cisco IOS, Release 12.2SX
1028	Experiences with Widespread Packet Sampling
1029	Complaint for Patent Infringement
1030	Cloud Byte’s Preliminary Infringement Contentions with attached Exhibit 6
1031	U.S. District Court – Judicial Caseload Profiles (March 2025)
1032	Assignment (U.S. Patent No. 9,560,177)
1033	Lex Machina (U.S. Patent No. 9,560,177)

Exhibit	Description
1034	Cloud Byte’s Disclosure of Proposed Claim Constructions and Extrinsic Evidence with attached Exhibit A
1035	Internet Archive Declaration of Mina Ching for RFC-5474
1036	Declaration of Su Li

MANDATORY NOTICES UNDER 37 C.F.R. § 42.8

Real Party-in-Interest (37 C.F.R. § 42.8(b)(1))

The real parties-in-interest are petitioner Dell Technologies Inc. and Dell Inc. (“Petitioners”). No unnamed entity is funding, controlling, or directing this petition or has the ability to control or direct this petition or Petitioners’ participation in any resulting *inter partes* review. Petitioners understand based on public information that the challenged patent, U.S. Patent No. 9,560,177 (the “’177 Patent”), is owned by Cloud Byte LLC (“Patent Owner”).

Related Matters (37 C.F.R. § 42.8(b)(2))

Petitioners are aware of the following district court lawsuit involving the ’177 Patent: *Cloud Byte LLC v. Dell Technologies Inc. and Dell Inc.*, Case No. 2:24-cv-00637-JRG (E.D. Tex.).

Lead and Back-Up Counsel (37 C.F.R. § 42.8(b)(3))

Petitioners appoint James L. Day (Reg. No. 72,681) of Farella Braun + Martel LLP as lead counsel and appoint the following attorneys as back-up counsel: Daniel Callaway (Reg. No. 74,267) and Winston Liaw (Reg. No. 78,766) of Farella Braun + Martel LLP.

Service Information (37 C.F.R. § 42.8(b)(4))

Service of any documents to lead and back-up counsel can be made via hand-delivery to Farella Braun + Martel LLP, One Bush Street, Suite 900, San Francisco, California, 94104. Petitioners consent to electronic service to the

following email addresses: jday@fbm.com, dcallaway@fbm.com,
wliaw@fbm.com, and calendar@fbm.com.

I. INTRODUCTION

Dell Technologies Inc. and Dell Inc. (“Petitioners”) respectfully request *inter partes* review of claims 1-20 (the “challenged claims”) of U.S. Patent No. 9,560,177 (the “’177 Patent”) owned by Cloud Byte LLC (“Patent Owner”). Because this petition demonstrates a reasonable likelihood Petitioners will prevail on at least one—and indeed all—of the challenged claims, *inter partes* review should be instituted and the challenged claims canceled as unpatentable.

II. GROUNDS FOR STANDING

Petitioners certify pursuant to 37 C.F.R. § 42.104 that the ’177 Patent is available for *inter partes* review and that Petitioners are not barred or estopped from requesting review of the challenged claims on the grounds identified herein.

III. IDENTIFICATION OF CHALLENGE

Pursuant to 37 C.F.R. § 42.104(b), Petitioners request *inter partes* review of the challenged claims and further request that the challenged claims be canceled as unpatentable for obviousness under 35 U.S.C. § 103¹ based on the following grounds:

¹ The pre-AIA version of Sections 102, 103, and 112 of the Patent Act apply.

Grounds	Claims	Prior Art
1	1, 5-7, 11-13, and 19	Rijhsinghani (EX1005)
2	2-4, 8-10, 12, 14-18, and 20	Rijhsinghani and RFC-5474 (EX1006)
3	2-4, 8-10, 12, 14-18, and 20	Rijhsinghani and Lean (EX1007)
4	1, 5-7, 11-13, and 19	Rijhsinghani and Computer Networking (EX1008)
5	2-4, 8-10, 12, 14-18, and 20	Rijhsinghani, RFC-5474, and Computer Networking
6	2-4, 8-10, 12, 14-18, and 20	Rijhsinghani, Lean, and Computer Networking

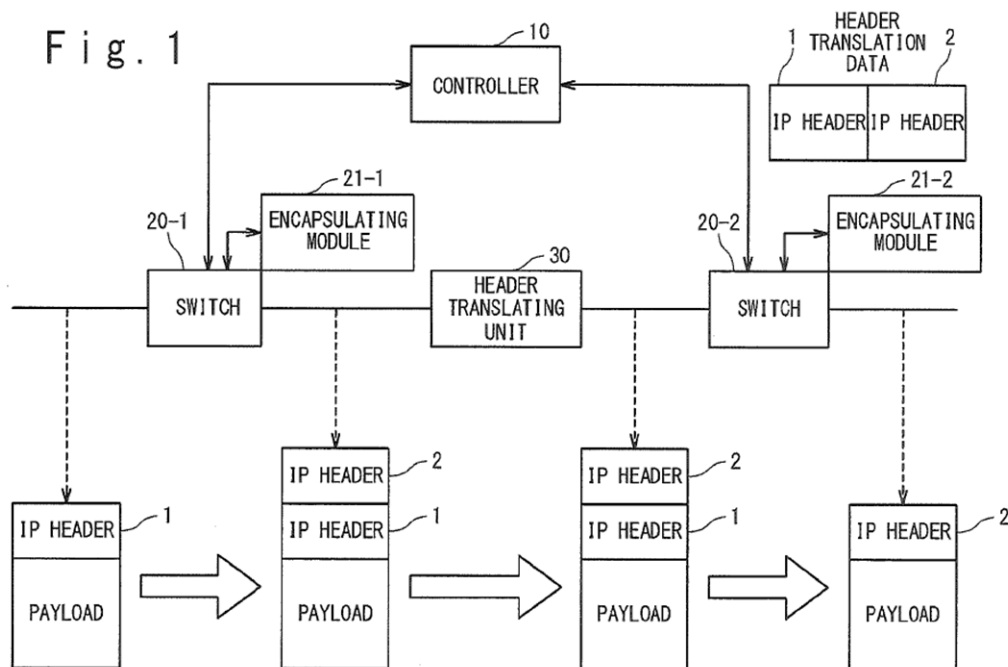
Petitioners’ proposed constructions, the evidence relied upon, and the precise reasons why the challenged claims are unpatentable are provided in the following sections of the petition. A List of Exhibits identifying the specific evidence relied upon is provided above.

IV. THE CHALLENGED PATENT

A. Patent Specification

The ’177 Patent is entitled “Network System and Network Flow Tracing Method.” EX1001, 1. It was issued on January 31, 2017, and claims priority to a Japanese patent application filed February 17, 2011. *Id.*

The '177 Patent “relates to a network system, and especially to a network flow tracing method in a network system.” EX1001, 1:14-16. Figure 1 shows “a network system according to the present invention and change of a packet in the system....” *Id.*, 4:17-19.



'177 Patent, Fig. 1

In the figure, “the network system according to the present invention includes a controller 10, switches 20” (i.e., 20-1 and 20-2), “and a header translating unit 30,” which is network equipment that changes the packet’s header making it difficult to monitor and control the packet flow. *Id.*, 4:44-47; *see also id.*, 2:31-43. The figure illustrates a data packet being forwarded through switch 20-1, header translating unit 30, and switch 20-2. EX1003, ¶43.

The data packet is initially received by switch 20-1 with IP header 1 and payload. EX1001, 6:46-47. Switch 20-1 “hands [the] inputted packet to the encapsulating module 21-1,” which “duplicates the IP header 1..., encapsulates the packet with the same header as the IP header 1 and transfers the packet.” *Id.*, 6:45-46, 6:48-51; *see also* EX1003, ¶44. Next, header translating unit 30 translates “the IP header 1 into the IP header 2.” EX1001, 6:65-67.

“Next, the switch 20-2 hands [the] inputted packet to the encapsulating module 21-2.” EX1001, 7:24-25. If “the inputted packet is an encapsulated packet” (as illustrated), encapsulating module 21-2 “refers to the IP header 2 and the IP header 1...and generates header translation data based on a pair (or set) of them and transmits it to the controller 10.” *Id.*, 7:37-43. The “encapsulating module 21-2 removes the IP header 2...to decapsulate the inputted encapsulated packet, and replaces ‘the IP header 1’ with ‘the IP header 2’....” *Id.*, 7:47-52. “In this way, the encapsulating module 21-2 removes the IP header 1 from the original packet and adds IP header 2 to the payload, and then transfers the packet.” *Id.*, 7:53-55.

By operating as described, “the controller 10 can acquire the header translation data without improving the header translating unit 30, and trace the flow by using the data.” EX1001, 7:58-61.

B. Prosecution History

The examiner initially rejected the application claims under Section 101, but following an examiner interview withdrew the rejection and allowed the claims.

EX1002, 88-94, 145, 149, 151-165, 173-180; EX1003, ¶¶48-53.

C. Patent Claims

The '177 Patent contains 20 claims, including four independent claims (i.e., claims 1, 7, 13, and 19). Claim 1 reads:

[1pre] A switch apparatus, comprising:
[1a] a storage storing a table, the table including rules and actions corresponding to the rules; and
[1b] a controller comprising:
[1c] a memory storing instructions; and
[1d] a processor configured to execute the instructions to:
[1e] receive the rules and the actions from a control apparatus;
[1f] identify, based on the rules, a received packet;
[1g] duplicate a part of a header of the identified packet as an additional header when the identified packet comprises a target of encapsulation;
[1h] encapsulate the identified packet by the additional header;
and
[1i] process, based on the actions, the identified packet.

Claim language for all challenged claims appears in the attached Claim Appendix.

D. Person of Ordinary Skill in the Art

A POSITA would have had an undergraduate degree (or equivalent) in

electrical engineering, computer science, or a comparable subject and approximately two to three years of academic or professional work experience in a technical field with exposure to computer networking and switching/routing technology. EX1003, ¶71. A higher level of education could substitute for less industry experience and more industry experience could substitute for the specific level of education. *Id.* In any event, a POSITA would have been knowledgeable about then-existing network switching and routing technologies and techniques. *Id.*

V. CLAIM CONSTRUCTION

The challenged claims should be construed under the *Phillips* standard. 37 C.F.R. § 42.100(b); *see generally Phillips v. AWH Corp.*, 415 F.3d 1303 (Fed. Cir. 2005)(en banc). The Board, however, need only construe claims when necessary to resolve the underlying controversy. *See Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co.*, 868 F.3d 1013, 1017 (Fed Cir. 2017). Aside from the following terms, Petitioners believe that no express constructions are necessary to determine that the asserted prior art discloses the limitations of the challenged claims.²

² Petitioners reserve the right to present other claim constructions in the related district court litigation as relevant and necessary in that proceeding.

A. “encapsulate”/“encapsulating”/“encapsulation”/“decapsulate”

Claim 1 recites a “switch apparatus” configured to “duplicate a part of a header of the identified packet as an additional header when the identified packet comprises a target of ***encapsulation***” and then to “***encapsulate*** the identified packet by the additional header....” EX1001, 12:34-39.³ The other independent claims recite identical or substantially similar limitations. *Id.*, 13:1-5 (claim 7), 13:30-34 (claim 13), 14:25-29 (claim 19). The concept of “encapsulation” was well-known. EX1003, ¶83. A POSITA would have understood “encapsulate,” “encapsulating,” and “encapsulation” to mean appending an additional header to a data packet. *Id.*, ¶82. This is consistent with the plain meaning of the terms to a POSITA. *Id.*, ¶¶83-89. It is also consistent with the use of the terms in the specification. *Id.*, ¶¶85-88 (citing EX1001, 6:48-63, 8:41-43, 8:49-53, 10:8-17 and EX1010-EX1011).

Similarly, a POSITA would understand “***decapsulate*** the encapsulated packet” in dependent claims 4, 10, and 16 to mean removing the additional header that was appended to the received data packet. EX1003, ¶90.

³ All emphasis is added unless otherwise indicated.

B. “[duplicate/duplicating] a part of a header of the identified packet as an additional header”

Claim 1 recites a “switch apparatus” configured to “duplicate a part of a header of the identified packet as an additional header when the identified packet comprises a target of encapsulation....” EX1001, 12:34-36. The other independent claims recite identical or substantially similar limitations. *Id.*, 13:1-3 (claim 7), 13:30-32 (claim 13), 14:25-27 (claim 19). A POSITA would understand this claim language to mean that when a packet is identified as a target for encapsulation (i.e., when it will have an additional header added), a part of the existing header is duplicated to use as a portion of the new header for encapsulating the packet. EX1003, ¶91. This is consistent with the patent specification, which uses “header” loosely to refer to certain fields in the data packet’s header. *Id.*, ¶92 (citing EX1001, 1:56-60, 3:28-30, 6:20-24, 6:28-30, 8:34-36,). It is also consistent with a POSITA’s understanding of data packet processing. *Id.*, ¶93.

VI. PRIOR ART REFERENCES

The ’177 Patent is a continuation of an application filed December 8, 2011. EX1001, 1. It claims priority to a Japanese application filed on February 17, 2011.

Id. Patent Owner asserts no earlier priority date. EX1030, 10.⁴ The '177 Patent is pre-AIA. See EX1002, 89, 178 (examination was under pre-AIA statutes).

A. Rijhsinghani (EX1005)

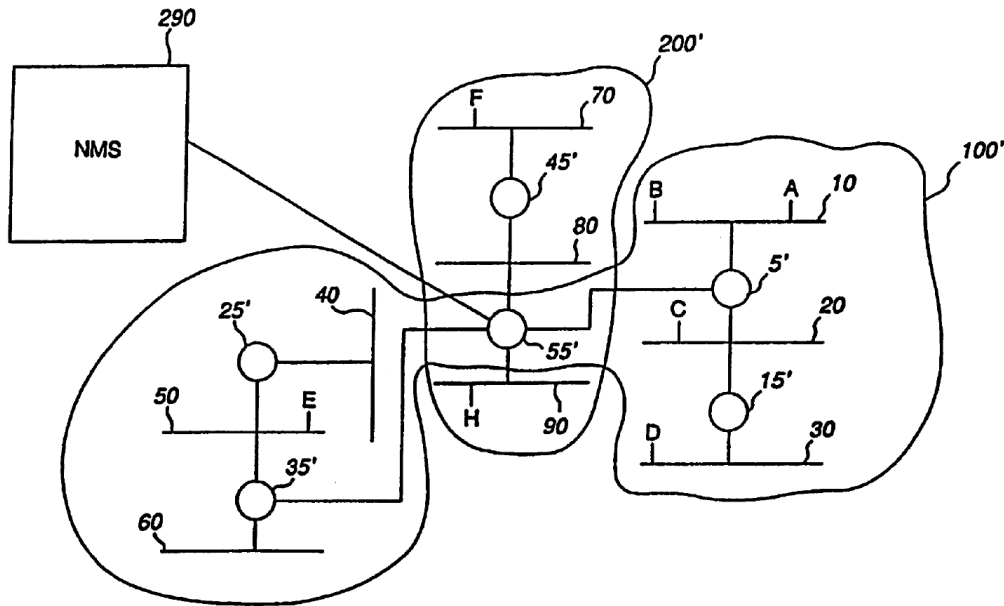
Rijhsinghani is a U.S. patent filed December 23, 1998, entitled “Virtual Local Area Networks Having Rules of Precedence.” EX1005, 1. Rijhsinghani constitutes prior art under at least Section 102(a), (b), and (e)(pre-AIA).

Rijhsinghani discloses “[a] switch for use in a communications system having multiple local area networks interconnected by multiple switches so as to be configurable into different types of virtual local area networks....” EX1005, 1 (Abstract). The switch “detects a communication from the local area network and identifies a virtual local area network over which the communication is to be transmitted based upon rules of precedence for different types of virtual local area networks.” *Id.* “The communication is appended with a VLAN tag representing the identified virtual local area network so as to form a VLAN communication that is directed to the second communication port for transmission over the identified virtual local area network.” *Id.*

Rijhsinghani’s Figure 2 illustrates an exemplary VLAN system. EX1005,

⁴ For purposes of this petition, Petitioners take no position on the foreign priority claim. Petitioner’s prior art antedates the '177 Patent in any event.

2:5-10.



Rijhsinghani, Fig. 2

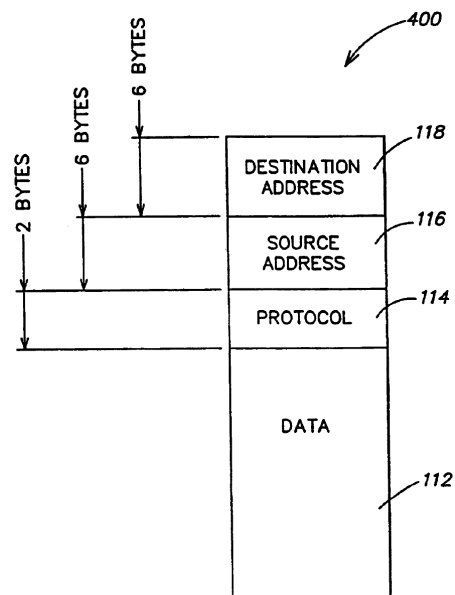
Figure 2 illustrates individual LANs (10-90) connected by switches (5'-55'). *Id.*, 2:8-10. “A network management station (NMS) 290 controls the interconnection of the individual LANs such that LANs can be easily bridged to other LANs on a long term or short term basis without the need to rewire the network.” *Id.*, 2:10-14.

In Figure 2, “the NMS 290 has configured two VLANs by instructing, e.g., programming, and thereby configuring the switches 5'-55' such that LANs 10-60 are bridged together by switches 5', 15', 55', [25' and] 35' to form VLAN 100' and LANs 70-90 are bridged together by switches 45' and 55' to form VLAN 200'.” EX1005, 2:14-19. Rijhsinghani’s “switches 5'-55' are multi-ported and programmable by the NMS 290 such that the network can be configured and

reconfigured in any desired manner by simply changing the switch instructions.”

Id., 2:20-26. “If it becomes desirable to change the network configuration, this is easily accomplished by issuing commands from NMS 290 to the applicable switches 5'-55'.” *Id.*, 2:36-38.

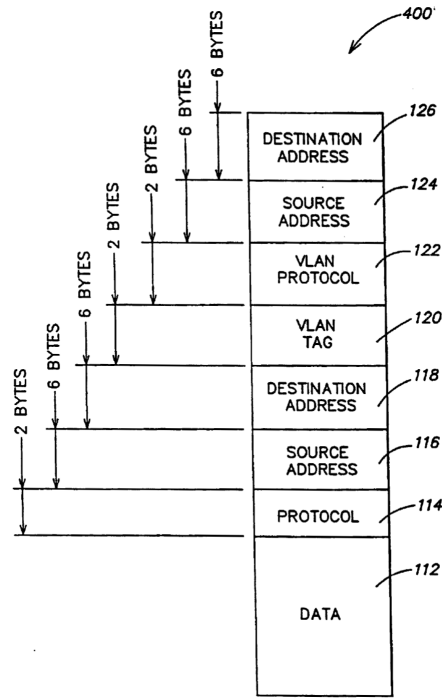
Rijhsinghani’s switches are configured to receive a packet and determine if it should be forwarded to a particular VLAN. EX1003, ¶101. Figure 3 illustrates “a communication packet 400” in a conventional LAN protocol of the type that would be received by one of Rijhsinghani’s switches. EX1005, 2:57-61.



Rijhsinghani, Fig. 3

Communication packet 400 “includes a destination address 118 having six bytes, a source address 116, and message data 112” as well as “an indication of the applicable LAN protocol identifier 114.” *Id.*, 2:57-61.

When a switch receives communication packet 400, it determines the appropriate VLAN (if any) and adds a VLAN header. EX1003, ¶102. Figure 4 illustrates “a VLAN communications packet 400” with a VLAN header added. EX1005, 3:22-25. “The VLAN header is added by the initial switch to which the message packet is directed.” *Id.*, 3:25-26.

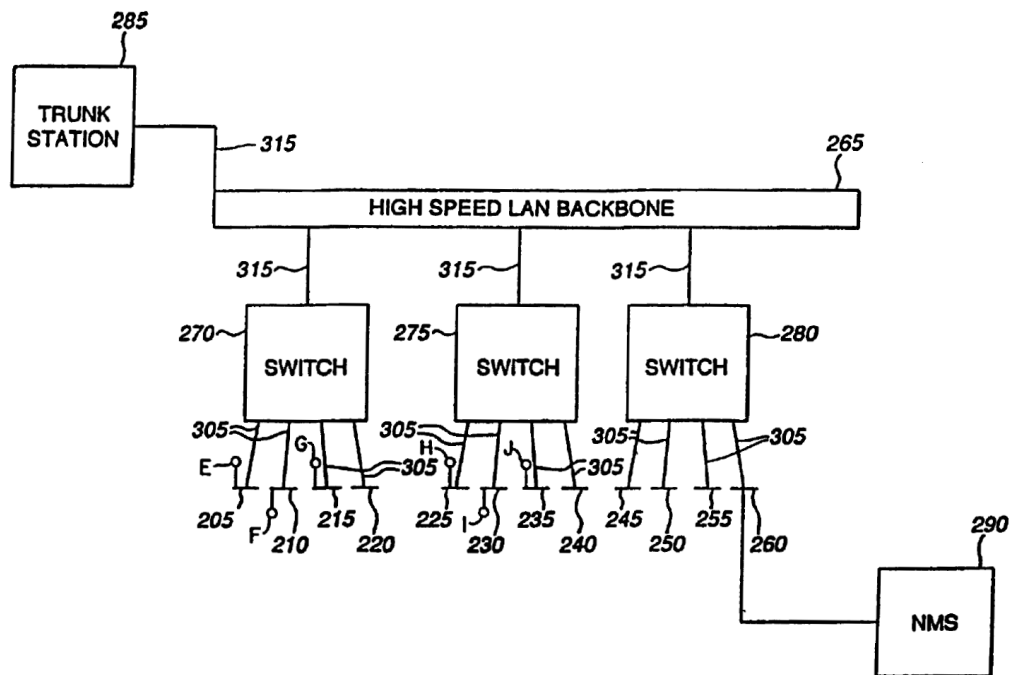


Rijhsinghani, Fig. 4

“The VLAN header, as shown, includes a destination address 126 which is the same address as the destination address 118, a source address 124 which is the same as source address 116, a protocol identifier 122, and a VLAN tag 120 identifying the applicable VLAN.” *Id.*, 3:29-33.

Rijhsinghani’s Figure 5 illustrates “a schematic of a conventional VLAN

system.” EX1005, 2:62.



Rijhsinghani, Fig. 5

“The VLAN system includes LANs 205-260 which are connected by switches 270-280 to a high-speed LAN backbone or trunk 265.” *Id.*, 2:63-65. The NMS 290 is “interconnected to the switches 270-280 via LAN 260,” and “trunk station 285 is connected to the high-speed LAN backbone 265 via a trunk port 315.” *Id.*, 2:65-66, 3:2-3.

Rijhsinghani’s NMS 290 “is capable of configuring different types of VLANs as is understood by those skilled in the art.” EX1005, 4:1-3. “For example, VLANs may be port-based, address-based, protocol-based, port and protocol-based, or address and protocol-based.” *Id.*, 4:3-5. To configure a VLAN,

“the NMS instructs the appropriate switches to identify the VLAN for packets received at the switch.” *Id.*, 4:5-8. This “enables the switch to transmit the packet over the appropriate VLAN.” *Id.*, 4:8-10.

Rijhsinghani states that “[a]lthough it is known to configure different types of VLANs within a VLAN system based upon characteristics such as those previously described, problems arise in attempting to implement such systems.” EX1005, 5:41-45. For example, “under certain circumstances, overlap of VLANs may occur....” *Id.*, 5:45-47; *see also id.*, 4:38-40, Fig. 6. “Overlap occurs when a communication packet received at a switch can be identified with more than one VLAN.” *Id.*, 5:47-49.

Rijhsinghani addresses “overlap” using “rules of precedence.” EX1005, 11:52-57. Switches 270'-280' (illustrated in Figures 7-9) are each “programmed to utilize an order of precedence to identify with certainty the appropriate VLAN for transmission of the received communication.” *Id.*, 9:47-50. The switches are programmed to “tag the communication with the VLAN header, i.e., the VLAN tag, representing the VLAN which is port and protocol-based over any other VLAN.” EX1005, 9:50-58. A port-based VLAN is next in precedence. *Id.*, 10:3-6. In other words, port-and-protocol-based VLANs have the highest precedence, followed next by port-based VLANs, and so on. EX1003, ¶109; EX1001, Fig. 11.

B. RFC-5474 (EX1006)

RFC-5474 is a Request for Comments (RFC) published by the Network Working Group of the Internet Engineering Task Force (IETF) in March 2009, entitled “A Framework for Packet Selection and Reporting.” EX1006, 1.

RFC-5474 constitutes prior art under at least Sections 102(a),(b)(pre-AIA).

RFC-5474 “specifies a framework for the PSAMP (Packet SAMPling) protocol” including functions “to select packets from a stream according to a set of standardized Selectors, to form a stream of reports on the selected packets, and to export the reports to a Collector.” EX1006, 2. RFC-5474 “describes the PSAMP framework for network elements to select subsets of packets by statistical and other methods, and to export a stream of reports on the selected packets to a Collector.” *Id.*, 4. RFC-5474 was readily available to “persons interested and ordinarily skilled in the subject matter or art exercising reasonable diligence [to] locate it.” *Medtronic, Inc. v. Barry*, 891 F.3d 1368, 1380 (Fed. Cir. 2018); EX1003, ¶¶114-115; EX1035, ¶¶2 and 22 (establishing that Internet Archive’s Wayback Machine captured the content of EX1006 no later than November 14, 2010). The Wayback Machine is sufficient evidence of publication and authentication. *Valve Corp. v. Ironburg Inventions Ltd.*, 8 F.4th 1364, 1374 (Fed. Cir. 2021).

C. Lean (EX1007)

Lean is a U.S. patent application publication filed August 1, 2007, and published February 7, 2008. EX1007, 1. Lean constitutes prior art under at least Section 102(a), (b), and (e)(pre-AIA).

Lean discusses “methods and systems for monitoring IP traffic on a high bandwidth IP network,” including “monitoring tunneled IP traffic....” EX1007, [0002]. Figure 1 illustrates “conventional monitoring of traffic traversing a high bandwidth IP network.” *Id.*, [0018]. “Each tap 104 passively copies all IP packets transmitted across Ethernet link 100 and sends the IP packet copy stream 106 to processor 108 for analysis.” *Id.*; *see also* EX1003, ¶¶117-118. Lean also describes several inventive embodiments that similarly copy packets for transmission to remote processors for analysis. EX1007, [0029]-[0034], Figs. 6-7; EX1003, ¶¶120-121.

D. Computer Networking (EX1008)

Computer Networking is a book entitled “Computer Networking: A Top-Down Approach” published in 2008. EX1008, 4; EX1003, ¶122. It constitutes prior art under at least Sections 102(a),(b)(pre-AIA). It was authored by experts in the field of computer networking. EX1008, 5 (“About the Authors”). Engineers and technicians working in the field, as well as students, would reference Computer Networking for an authoritative description of then-existing networking

technology and techniques. EX1003, ¶122. It was readily available to POSITAs “exercising reasonable diligence [to] locate it.” *Medtronic*, 891 F.3d at 1380; EX1003, ¶122.

VII. THE CHALLENGED CLAIMS ARE INVALID AS OBVIOUS BASED ON RIJHSINGHANI AS A PRIMARY PRIOR ART REFERENCE

A. Ground 1: Rijhsinghani Alone Renders Claims 1, 5-7, 11-13, and 19 Obvious

Rijhsinghani discloses and renders obvious to a POSITA all limitations of claims 1, 5-7, 11-13, and 19 for the reasons stated below. EX1003, ¶¶123-243. Rijhsinghani is analogous art, and a POSITA would have naturally considered it, because it is in the same field as the ’177 Patent, namely network communications. EX1003, ¶124; EX1005, 1 (Abstract), 1:8-12, 6:26-35; EX1001, 1:14-16, 4:33-38. Further, both Rijhsinghani and the ’177 Patent contemplate network switches configured to modify packet headers, including by duplicating header fields, to create a new header for encapsulation. EX1003, ¶124; EX1005, 2:57-61, 3:23-45, Figs. 3 and 4; EX1001, 1 (Abstract), 3:22-27.

1. Claim 1

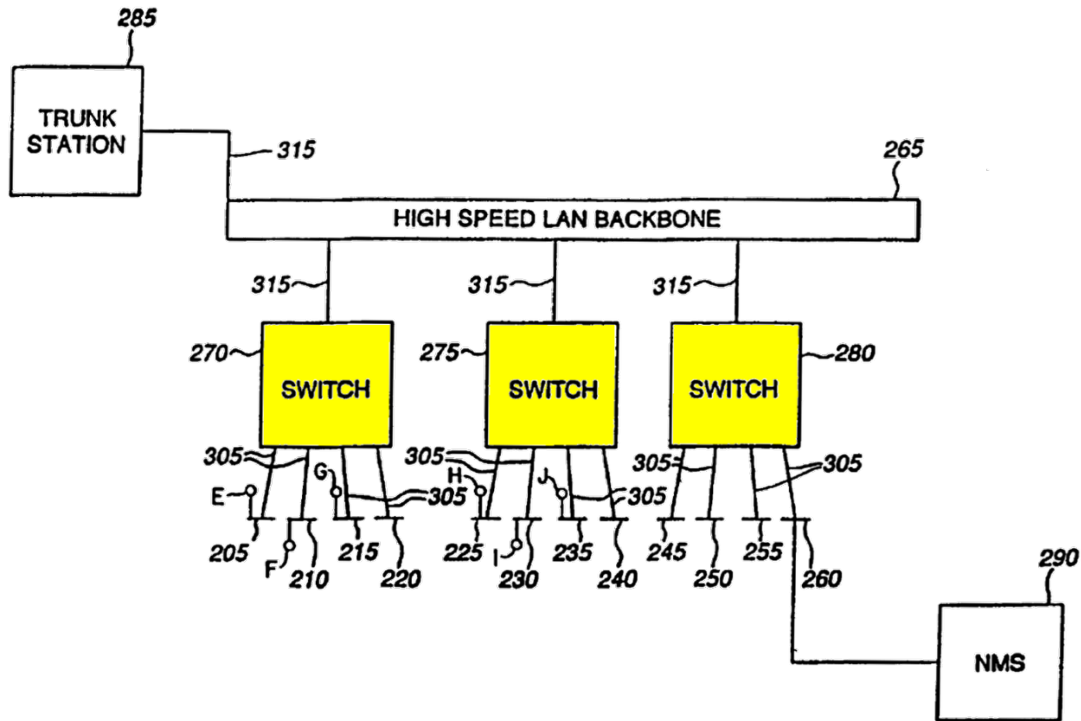
a. [1pre] A switch apparatus, comprising:

Rijhsinghani discloses a *switch apparatus* as recited in the preamble.

EX1003, ¶¶126-131.⁵ For example, Rijhsinghani discloses “[a] **switch** for use in a communications system having multiple local area networks interconnected by multiple **switches** so as to be configurable into different types of virtual local area networks....” EX1005, 1 (Abstract); *see also id.*, 11:54-57 (“**Switches** route communications to addressees, within a VLAN system capable of configuring multiple types of VLANs....”), 12:34-35 (“wherein the network device is a **switch**”), 12:41-42 (same), 12:53-54 (same), 13:13-14 (same), 13:18-19 (same), 13:25-26 (same), 13:37-38 (same), 14:16-17 (same), 14:21-22 (same), 14:28-29 (same), 14:40-41 (same). Rijhsinghani explains that “VLANs allow LANs to be bridged in virtually any desired manner, i.e., independent of physical topology, with **switches** operating at layer 2” meaning that “the **switches** are transparent to the user.” *Id.*, 1:63-66; *see also id.*, 2:8-26, 2:37-56, Fig. 2.

Rijhsinghani’s Figure 5 illustrates a “conventional VLAN system” including switches 270, 275, and 280. EX1005, 2:62-65.

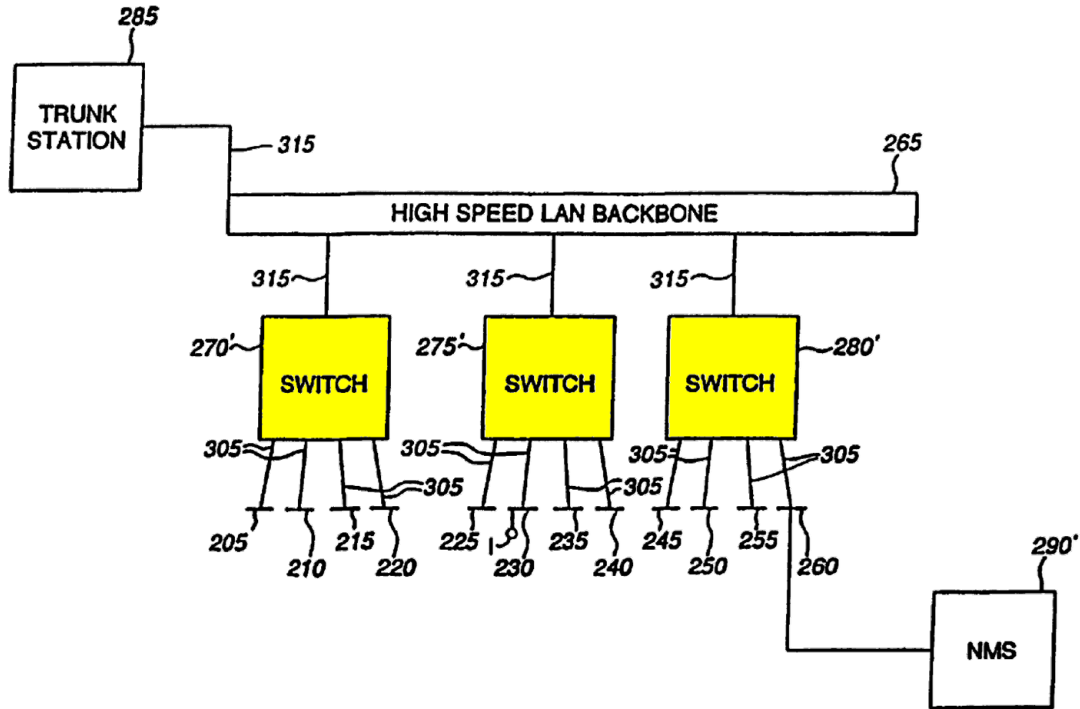
⁵ For ease of review, claim language is presented in italics.



Rijhsinghani, Fig. 5 (highlighted)

“Each switch is capable of interconnecting a LAN connected via an access port 305 with another LAN connected via an access port 305.” *Id.*, 3:8-10. Each switch can also interconnect a LAN “with a LAN connected to another switch by an access port 305 via high-speed LAN backbone 265.” *Id.*, 3:14-17. Each of switches 270, 275, and 280 constitutes a *switch apparatus*. EX1003, ¶128.

Figure 7 illustrates “a VLAN system in accordance with the present invention.” EX1005, 7:33-34. As illustrated, “[t]he network includes multiple Local Area Networks (LANs) 205-260 interconnected by multiple multi-ported reconfigurable switches 270', 275', and 280' all of which are connected by a high speed backbone LAN 265, often referred to as the trunk.” *Id.*, 7:50-55.



Rijhsinghani, Fig. 7 (highlighted)

A POSITA would have understood that switches 270', 275', and 280' operate in the same manner as the switches in the conventional VLAN system of Figure 5, except that the switches in Figure 7 implement Rijhsinghani's rules of precedence.

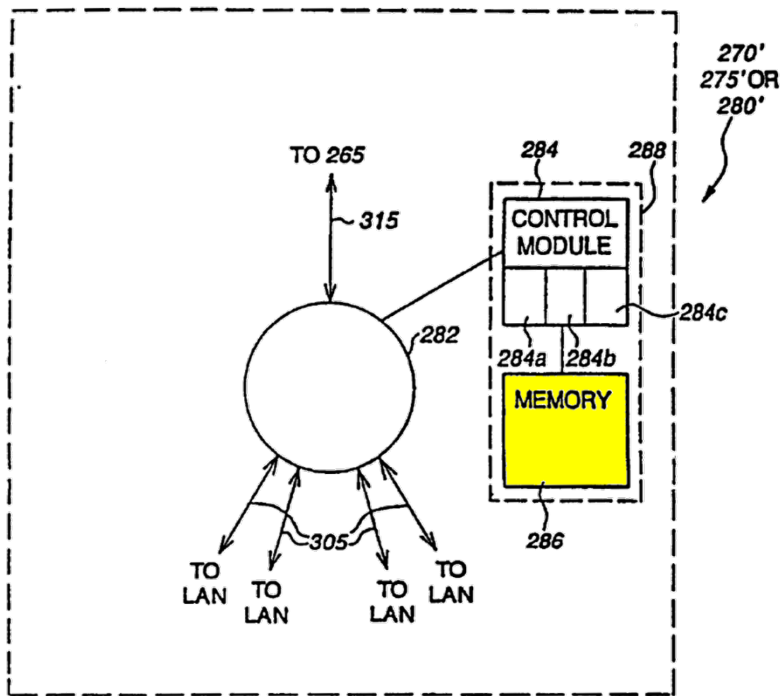
EX1003, ¶130. Each of switches 270', 275', and 280' also constitutes a *switch apparatus* as recited in the preamble. *Id.*, ¶130.

- b. [1a] a storage storing a table, the table including rules and actions corresponding to the rules; and**

Rijhsinghani discloses this limitation. EX1003, ¶¶132-150. In particular, each of Rijhsinghani's switches includes *a storage*. *Id.*, ¶133. Figure 8 “depicts a switch which can be utilized in the VLAN system” described by Rijhsinghani.

EX1005, 7:35-37. As illustrated, “each of the switches 270', 275', and 280'

includes a control console 288 having a control module 284 and a memory 286 for storing and processing control and VLAN configuration instructions.” *Id.*, 8:6-9.



Rijhsinghani, Fig. 8 (highlighted)

Each switch’s memory 286 constitutes *a storage* as recited in claim 1. EX1003, ¶133.

Each memory 286 stores VLAN configuration instructions, including *rules and actions corresponding to the rules*. EX1003, ¶134. As Rijhsinghani explains, “NMS 290’ is capable of configuring VLANs of differing types” (e.g., port-based, address-based, etc.) by instructing “the switches 270’-280’ as to the configuration of the different types of VLANs.” EX1005, 9:30-36. Each switch “is programmed to consider the applicable characteristics of each communication received, via an

access port, in order to determine the appropriate VLAN tag to add to the communication” before transmission. *Id.*, 9:36-42. Thus, each switch “consider[s] the applicable characteristics of each communication received” (i.e., *rules*) to “determine the appropriate VLAN tag to add to the communication before transmission” (i.e., *actions corresponding to the rules*). EX1003, ¶134.⁶

Rijhsinghani’s NMS 290 (or NMS 290’) “is capable of configuring different types of VLANs as is understood by those skilled in the art.” EX1005, 4:1-3. For example, the NMS can configure VLANs that are “port-based.” *Id.*, 4:3-5. To do so, the NMS instructs certain switches “to identify the VLAN for a packet based upon the access port at which the packet is received.” *Id.*, 4:11-14; *see also id.*, 4:5-8 (“[T]he NMS instructs the appropriate switches to identify the VLAN for packets received at the switch.”). In this case, *rules* are based on the access port on which a packet is received. EX1003, ¶136. The switches use access port information (*rules*) to identify the appropriate VLAN so that a VLAN header can be added and the packet forwarded correctly (*actions corresponding to the rules*). *Id.*, ¶136; EX1005, 4:8-10 (“Identifying the appropriate VLAN for a packet enables the switch to transmit the packet over the appropriate VLAN.”), 3:39-45

⁶ Rijhsinghani’s description is similar to the ’177 Patent’s description of rules and actions. EX1003, ¶135 (citing EX1001, 1:52-2:9).

(“The switch determines, based upon instructions previously received from the NMS 290, that the LAN 205 falls within the applicable VLAN and, accordingly, adds the appropriate VLAN header to the packet,” which “is then directed via trunk port 315 to the high-speed backbone LAN 265....”), 9:23-29.

The NMS can also configure VLANs that are address-based, protocol-based, port-and-protocol-based, or address-and-protocol-based. EX1005, 4:3-5, 4:15-37; *see also id.*, 3:39-45, 9:23-29. For each type of VLAN, the *rules* are based on the address, protocol, port-and-protocol, or address-and-protocol information. EX1003, ¶¶136-140. The switches use the particular packet information (*rules*) to identify the appropriate VLAN so that the VLAN header can be added and the packet forwarded correctly (*actions corresponding to the rules*). *Id.*

Rijhsinghani’s “instructions” for configuring VLANs (*rules and actions corresponding to the rules*) “may be stored in the memory 286” (*a storage*) “and utilized by the switch control module 284 in determining which tag to add to a communication received at an access port 305.” EX1005, 9:42-45. Further, a POSITA would have known that the instructions issued to switches to configure VLANs would be stored in *a table*, as claimed, because a table data structure was the most common technique for switches to make packet routing decisions. EX1003, ¶142. Switches commonly used information about a received packet to look up routing information in *a table* stored on the switch. *Id.* (citing EX1008,

95-97, 263-264; EX1009, 4-11; EX1014, 1 (Abstract), 1:38-61, 2:1-26, 2:44-51, 3:37-4:43, Figs. 1B, 2; EX1015, 2:25-60, 5:42-52, 7:34-48, 10:27-47, Fig. 1; EX1016, 1:17-30, 4:24-51, 6:19-43, Figs. 1, 3; EX1017, 1 (Abstract), 1:26-37, 2:66-3:37, 4:57-5:20, Fig. 2). This technique allows network administrators to configure or reconfigure VLANs, as discussed in Rijhsinghani (EX1005, 2:10-14, 2:37-39), by simply updating routing information in tables on the switches.

EX1003, ¶142. Further, it would have been obvious to a POSITA to try implementing Rijhsinghani’s *rules* and *actions* on each switch in *a table* because it was one of a finite number of identified, predictable data structures for this purpose and could have been implemented by a POSITA with more than a reasonable expectation of success. *Id.*, ¶143. Thus, Rijhsinghani discloses to a POSITA *a storage* (memory 286) *storing a table, the table including rules and actions corresponding to the rules* (instructions configuring VLANs). *Id.*, ¶144.

Rijhsinghani discloses this limitation in an additional way based on Rijhsinghani’s rules of precedence. EX1003, ¶¶145-148. These apply when there is “overlap” such that a particular packet received by a switch “can be identified with more than one VLAN.” EX1005, 1 (Abstract), 5:47-49. In Rijhsinghani, “[a] switch control detects a communication from the local area network and identifies a virtual local area network over which the communication is to be transmitted based upon rules of precedence.” *Id.*, 1 (Abstract); *see also id.*, 1:8-12 (“The

present invention relates...particularly to communications systems having various types of virtual local area networks and established rules of precedence for matching a communication packet with a particular virtual local area network.”), 9:47-50 (“Each of the switches 270'-280' is programmed to utilize an order of precedence to identify with certainty the appropriate VLAN for transmission of the received communication.”), 9:50-10:44.

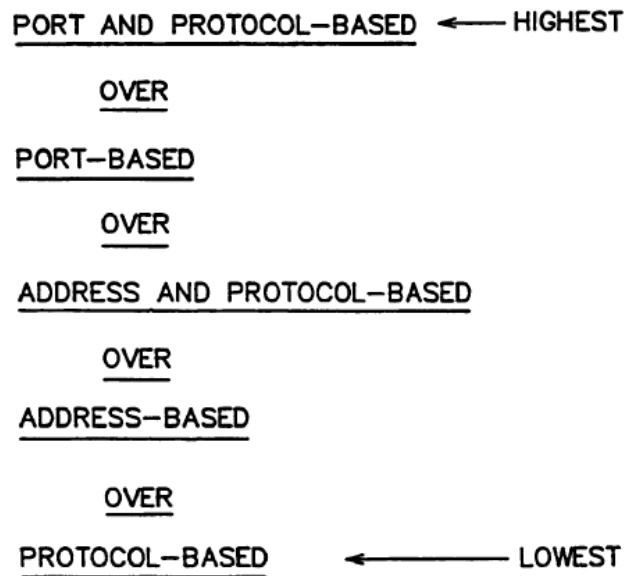
Rijhsinghani’s “rules of precedence” constitute *rules and actions corresponding to the rules*. EX1003, ¶146. In this case, the *rules* include the information associating a packet with one or more types of VLANs and the *actions corresponding to the rules* involve assigning the packet to a particular VLAN, including when there is overlap and the packet could be assigned to more than one VLAN. *Id.*; EX1005, 6:25-35 (“[T]he present invention provides rules of precedence for directing communications within different types of VLANs, in order to provide for predictable and desirable network behavior when there are areas of the network in which there is overlap in VLAN configurations, and to allow conflict resolutions by switches in the VLAN system.”), 10:64-11:4 (“In step 1330, in the case of more than one VLAN match, the applicable switch 270'-280' identifies which of the VLAN type matches has the highest priority based upon the order of precedence....”).

A POSITA would understand that Rijhsinghani’s rules of precedence are

stored in *a storage*, such as the switch's memory 286, in *a table* as claimed.

EX1003, ¶147. For the same reasons discussed above regarding the VLAN configuration instructions, a POSITA would know that the rules of precedence are in a table structure. *Id.* Further, Figure 11 illustrates the rules of precedence in a manner that suggests a table data structure to a POSITA. *Id.*, ¶148. The figure illustrates the order of precedence with port-and-protocol-based VLANs having the highest priority at the top, followed by port-based immediately below, etc.

EX1005, 10:64-11:4.



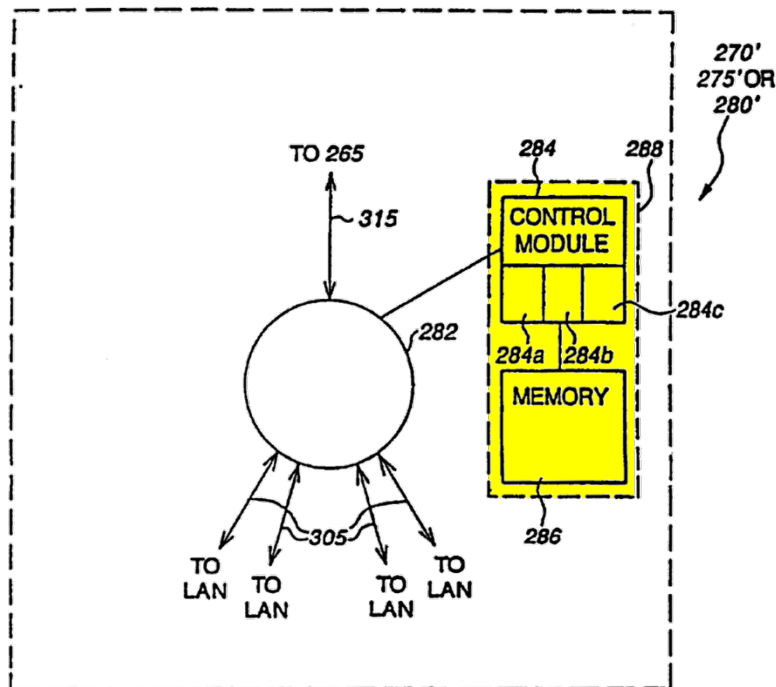
Rijhsinghani, Fig. 11

This figure suggests a table data structure to a POSITA because a table is processed starting with the top row and continuing row-by-row until a match is found. EX1003, ¶148. A table data structure therefore naturally establishes an

order of precedence with the first row having the highest priority (as in Figure 11), followed by rows of decreasing priority, with the lowest-priority at the bottom (also as in Figure 11). *Id.* Further, it would have been obvious to a POSITA to try implementing these *rules* and *actions* on Rijhsinghani's switch in *a table* because it was one of a finite number of identified, predictable data structures for this purpose and would have been implemented by a POSITA with more than a reasonable expectation of success. *Id.*, ¶149. Thus, Rijhsinghani discloses to a POSITA *a storage* (memory 286) *storing a table, the table including rules and actions corresponding to the rules* (rules of precedence). *Id.*, ¶150.

c. [1b] a controller comprising:

Rijhsinghani discloses *a controller*. EX1003, ¶¶151-153. For instance, Figure 8 illustrates Rijhsinghani's switch including control console 288, which constitutes *a controller*. *Id.*, ¶152.



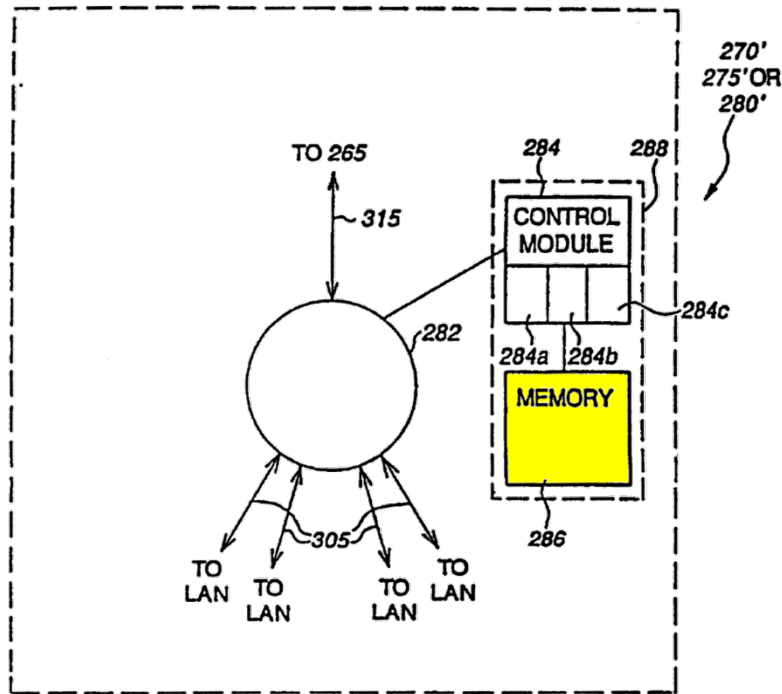
Rijhsinghani, Fig. 8 (highlighted)

“[E]ach of the switches 270', 275' and 280' includes a control console 288 having a control module 284 and a memory 286 for storing and processing control and VLAN configuration instructions.” EX1005, 8:6-9.

d. [1c] a memory storing instructions; and

Rijhsinghani discloses *a memory storing instructions*. EX1003, ¶¶154-156.

Figure 8 illustrates Rijhsinghani's switch including control console 288 (*a controller*) with memory 286, which constitutes *a memory storing instructions* as claimed. *Id.*, ¶155; EX1005, 8:6-9.



Rijhsinghani, Fig. 8 (highlighted)

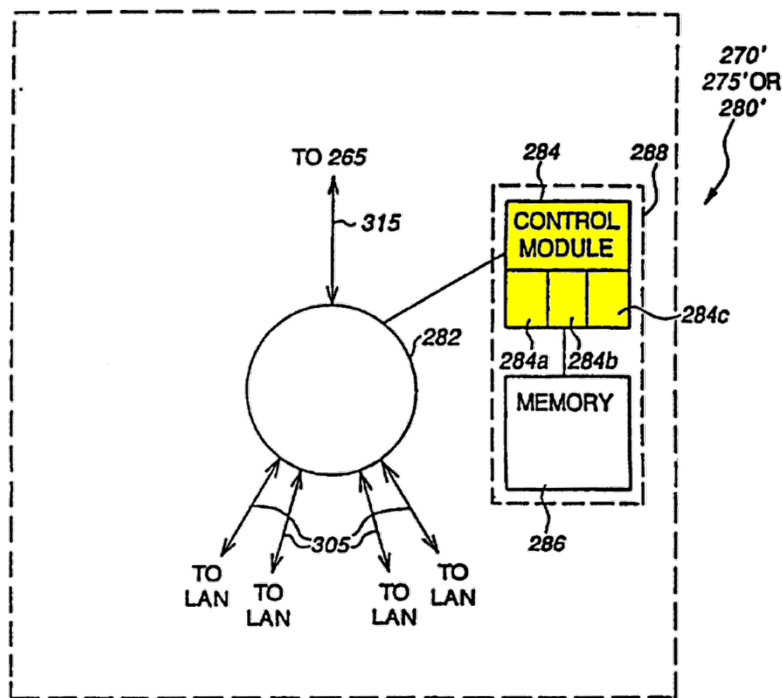
“[E]ach of the switches 270', 275' and 280' includes a control console 288 having...a memory 286 for storing and processing control and VLAN configuration instructions.” EX1005, 8:6-9; *see also id.*, 9:42-46 (“instructions may be stored in the memory 286 of the switch”).⁷

⁷ Memory 286 also constitutes *a storage storing a table, the table including rules and actions corresponding to the rules* as recited in 1a. See Section VII.A.1.b, above. A POSITA would know that switch memory could, and would, hold both instructions to be executed by a processor as well as routing and other types of tables. EX1003, ¶156.

e. [1d] a processor configured to execute the instructions to:

Rijhsinghani discloses *a processor configured to execute the instructions*.

EX1003, ¶¶157-161. Figure 8 illustrates Rijhsinghani’s switch including control console 288 (*a controller*) with control module 284, which discloses *a processor* as claimed. *Id.*, ¶158; EX1005, 8:6-11.



Rijhsinghani, Fig. 8 (highlighted)

“The control module 284 includes a controller 284a to control the switching device 282.” EX1005, 8:11-12. The control module 284 also includes “[a] detector 284b [that] detects a communication packet received from the backbone 265 via a trunk port 315 or from a LAN directly connected to the switch via an access port 305.” *Id.*, 8:12-15. “The control module 284 also includes a tagger

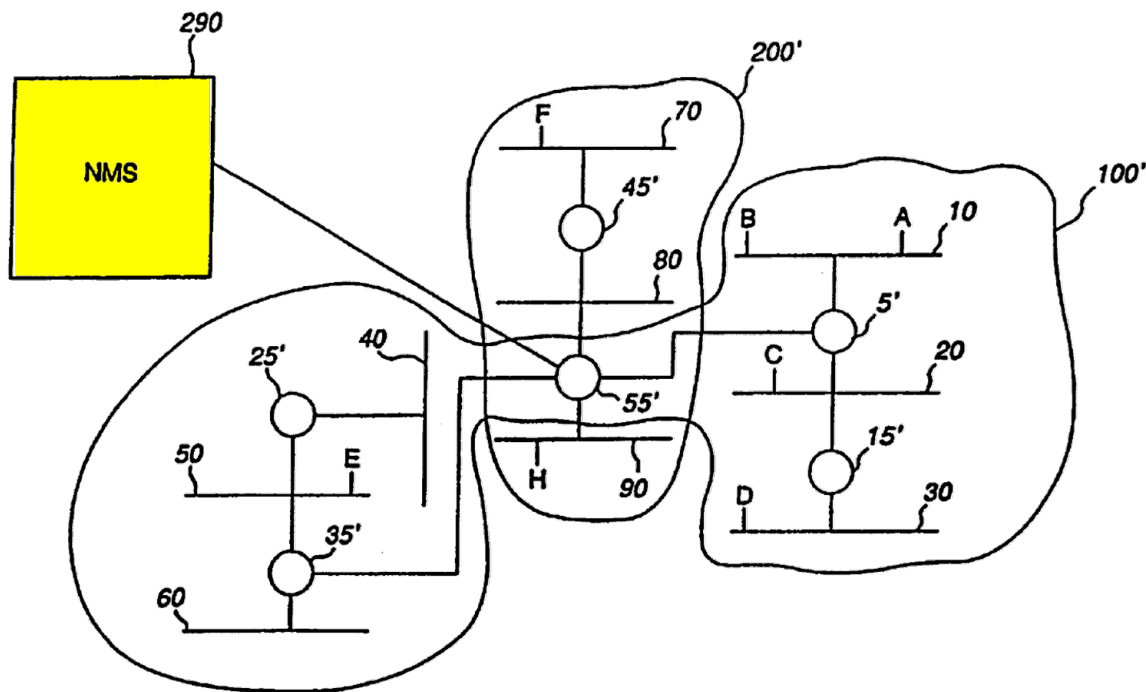
284c for tagging communications received via access port 305 for transmission from one member to another member of a configured VLAN by adding a VLAN header thereto.” *Id.*, 8:60-63.

A POSITA would understand that control module 284 (including controller 284a, detector 284b, and tagger 284c) is software or firmware executed by *a processor* on the switch for “processing control and VLAN configuration instructions.” EX1005, 8:6-9; EX1003, ¶160. Such a person would have known that switches included processors for executing instructions to process and route data packets. EX1003, ¶160; EX1015, 2:32-60, 4:19-36, 5:42-52, Fig. 1; EX1016, 1:45-58, 6:19-27; EX1008, 108; EX1022, 4-5, Fig. 2; EX1023, 3-4, Fig. 2. Thus, Rijhsinghani discloses *a processor configured to execute the instructions* stored in memory 286, as further detailed below. EX1003, ¶161.

f. [1e] receive the rules and the actions from a control apparatus;

Rijhsinghani discloses this limitation. EX1003, ¶¶162-166. In Rijhsinghani, “[a] network management station (NMS) 290 controls the interconnection of the individual LANs such that LANs can be easily bridged to other LANs on a long term or short term basis without the need to rewire the network.” EX1005, 2:10-14. As illustrated in Figure 2, “the NMS 290 has configured two VLANs by instructing, e.g., programming, and thereby configuring the switches 5'-55' such that LANs 10-60 are bridged together by switches 5', 15', 55', 35' to form VLAN

100' and LANs 70-90 are bridged together by switches 45' and 55' to form VLAN 200'." *Id.*, 2:14-19; see also *id.*, 2:62-3:22, Fig. 5.



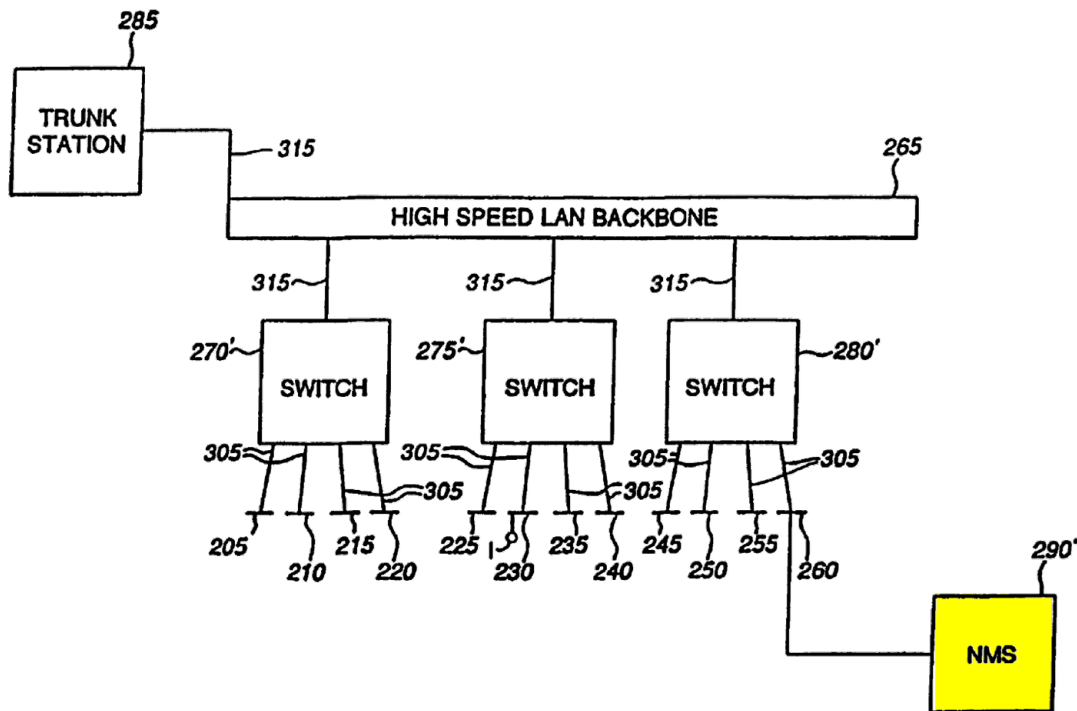
Rijhsinghani, Fig. 2 (highlighted)

Thus, NMS 290 constitutes *a control apparatus* that controls the configuration of Rijhsinghani's switches. EX1003, ¶163.

Further, Rijhsinghani's switches *receive the rules and the actions* from NMS 290 as claimed. "[S]witches 5'-55' are multi-ported and programmable by the NMS 290 such that the network can be configured and reconfigured in any desired manner by simply changing the switch instructions." EX1005, 2:23-26. Rijhsinghani also states that "[i]f it becomes desirable to change the network configuration, this is easily accomplished by issuing commands from NMS 290 to

the applicable switches 5'-55'." *Id.*, 2:36-39. Rijhsinghani explains, based on Figure 2, that "if desired, user H could be easily added to VLAN 100' by simply reconfiguring VLAN 100' from the NMS 290." *Id.*, 2:39-46. The NMS 290 instructs "switch 55' to allow communications to flow between user A-D and E and user H via switch 55', i.e., to include LAN 90 in VLAN 100' and remove it from VLAN 200'." *Id.*, 2:39-46. Rijhsinghani further explains that "the NMS 290 generates and transmits signals to instruct the switches 5'-55' to form the desired VLAN configurations." *Id.*, 2:53-56; *see also id.*, 4:5-10. Thus, the switches *receive the rules and the actions from a control apparatus* (NMS 290). EX1003, ¶165.

In addition, Figure 7 illustrates "a VLAN system in accordance with the present invention." EX1005, 7:33-34. Rijhsinghani states "(NMS) 290', which may be a workstation having network management software loaded thereon, manages the network by configuring the network via the switches 270', 275' and 280' to form one or more virtual local area networks (VLANs)." *Id.*, 7:58-63.



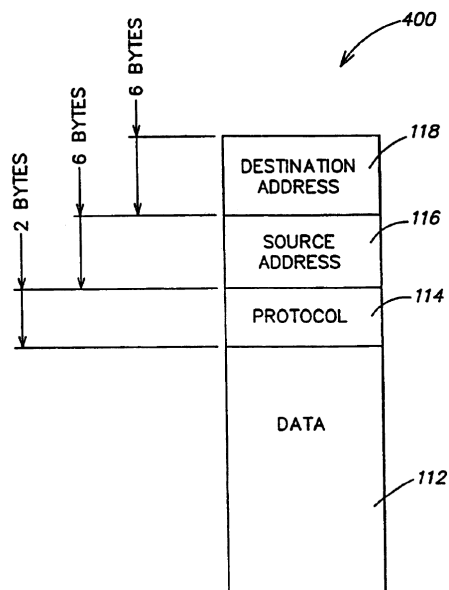
Rijhsinghani, Fig. 7 (highlighted)

Like NMS 290 discussed above, NMS 290' constitutes *a control apparatus* as claimed. EX1003, ¶165.

Rijhsinghani states that switches 270', 275', and 280' include a control console 288 “for storing and processing control and VLAN configuration instructions,” which “may be...transmitted to the switch by the NMS 290'.” EX1005, 8:6-11; *see also id.*, 9:34-36 (“The NMS 290' instructs the switches 270'-280' as to the configurations of the different types of VLANs.”). Again, in this embodiment, Rijhsinghani’s switches *receive the rules and the actions from a control apparatus* (NMS 290'). EX1003, ¶166.

g. [1f] identify, based on the rules, a received packet;

Rijhsinghani discloses this limitation. EX1003, ¶¶167-178. “When the NMS 290 configures a VLAN, the NMS instructs the appropriate switches to identify the VLAN for packets received at the switch.” EX1005, 4:5-8. Figure 3 illustrates “a conventional LAN message packet.” *Id.*, 7:27. “In a conventional LAN protocol, a communication packet 400, as shown in FIG. 3, includes a destination address 118 having six bytes, a source address 116, and message data 112” as well as “an indication of the applicable LAN protocol identifier 114.” *Id.*, 2:57-61.



Rijhsinghani, Fig. 3

In Rijhsinghani, “[a] switch control detects a communication [such as communication 400] from the local area network at the first port [of the switch] and identifies a virtual local area network over which the communication is to be

transmitted based upon rules of precedence for different types of virtual local area networks.” *Id.*, 6:55-60. The packet illustrated in Figure 3 is *a received packet* (i.e., it is an example of a packet in the form received by a switch before a VLAN header has been added). EX1003, ¶168.

In addition, Rijhsinghani discloses switches *identify[ing], based on the rules, a received packet*. EX1003, ¶169. For example, Rijhsinghani states that “[t]he switch determines, based upon instructions previously received from the NMS 290, that the LAN 205 falls within the applicable VLAN....” EX1005, 3:39-43; *see also id.*, 4:5-10 (“When the NMS 290 configures a VLAN, the NMS instructs the appropriate switches to identify the VLAN for packets received at the switch. Identifying the appropriate VLAN for a packet enables the switch to transmit the packet over the appropriate VLAN.”). The “instructions previously received from the NMS 290” include the claimed *rules*, and the *received packet* is *identif[ied]* as associated with a particular VLAN based on those *rules*. EX1003, ¶169; EX1005, 7:1-4 (“Typically, the communication will include at least a source address and a protocol identifier, which the switch control detects, along with the port at which the communication is received, to identify the VLAN.”).

Rijhsinghani also states that “[e]ach of the switches 270'-280' is programmed to consider the applicable characteristics of each communication received, via an access port, in order to determine the appropriate VLAN tag to add

to the communication before transmission via the trunk port 315 to the high speed LAN backbone or trunk 265.” EX1005, 9:37-41. Thus, the switches “consider the applicable characteristics of each communication received” to *identify, based on the rules, a received packet* as one associated with a particular VLAN. EX1003, ¶170.

Additionally, “[e]ach of the switches 270'-280' is programmed to utilize an order of precedence to identify with certainty the appropriate VLAN for transmission of the received communication.” EX1005, 9:45-54. In this way too, the switches identify, based on *the rules*, a received packet as one associated with a particular VLAN. EX1003, ¶171; EX1005, 9:59-10:44 (providing specific examples with reference to Figure 9).

Figure 10 is a flow chart illustrating that the switches *identify, based on the rules, a received packet*. EX1003, ¶172. Figure 10 “summarizes the steps performed at each switch 270'-280' to correctly identify the appropriate VLAN for tagging a communication received from a LAN connected directly thereto by access port 305.” EX1005, 10:45-48. First, “in step 1300, the switch receives a communication, typically in the form of a packet, at an access port 305.” *Id.*, 10:48-50.

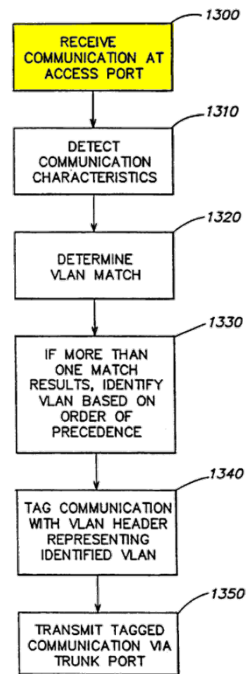


FIG. 10

Rijhsinghani, Fig. 10 (highlighted)

Thus, the switch receives *a received packet*. EX1003, ¶172.

“In step 1310 the communication characteristics are detected by the detector 284b of the control module 288.” EX1005, 10:50-52.

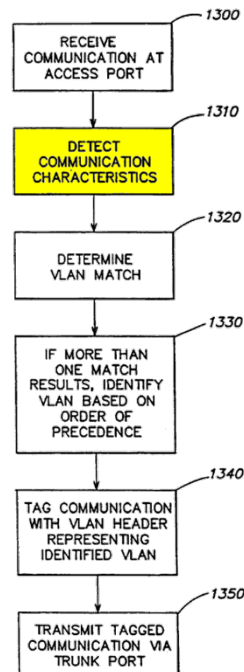


FIG. 10

Rijhsinghani, Fig. 10 (highlighted)

Next, in step 1320, “the control module 288 [*a controller*], in accordance with the programmed instructions stored in the memory 286 [*rules and actions corresponding to the rules*], determines one or more VLAN matches, i.e. determines one or more VLANs with which the communication may be appropriately associated [*identify, based on the rules, a received packet*].”

EX1005, 10:55-59; EX1003, ¶174.

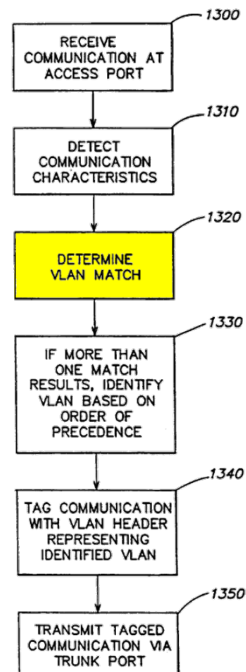


FIG. 10

Rijhsinghani, Fig. 10 (highlighted)

“In step 1330, in the case of more than one VLAN match, the applicable switch 270'-280' identifies which of the VLAN type matches has the highest priority based on the order of precedence....” EX1005, 10:64-11:4.

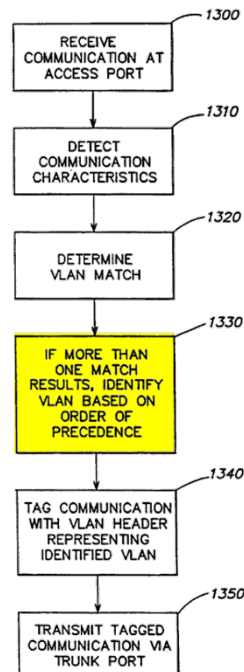


FIG. 10

Rijhsinghani, Fig. 10 (highlighted)

Thus, Figure 10 further discloses that Rijhsinghani’s switches *identify...a received packet* by associating it with a particular VLAN *based on the rules* of precedence used to address overlap. EX1003, ¶175.

In addition, Rijhsinghani’s switches *identify, based on the rules, a received packet* by determining whether the packet is associated with any VLAN. EX1003, ¶176; EX1005, 8:16-18 (“Communications from the backbone 265 may or may not include a VLAN header....”). “For example, communications from a system user which is a member of the default group will not be tagged with a VLAN header by a switch connected via an access port 305 to the LAN on which the system user resides” where the “default group is a group of system users not within any

VLAN.” EX1005, 8:18-23; *see also id.*, 8:23-34, 3:62-63. Rijhsinghani states that “[f]or a communication packet sent by a system user within the default group, the initial switch to which the packet is directed determines that the system user does not fall within any VLAN, and consequently does not add a VLAN header.” *Id.*, 3:63-67. Thus, for default group communications, the switch *identif[ies]...a received packet* as one that is not associated with a VLAN—and therefore does not have a VLAN header added—based on the communication’s characteristics (*based on the rules*). EX1003, ¶176.

Rijhsinghani also states that “where the sender and addressee are each members of a LAN connected to the same switch, there is no need to add a VLAN header to the communication before directing it to an output port 305.” EX1015, 9:13-16; *see also id.*, 9:25-30. Again, in this case, the switch *identif[ies]...a received packet* as one that is not associated with a VLAN—and therefore does not need a VLAN header—based on the communication’s characteristics (*based on the rules*). EX1003, ¶177.

Thus, Rijhsinghani discloses in a number of different ways that the switches *identify, based on the rules, a received packet*. EX1003, ¶178.

h. [1g] duplicate a part of a header of the identified packet as an additional header when the identified packet comprises a target of encapsulation;

Rijhsinghani discloses this limitation. EX1003, ¶¶179-185. Rijhsinghani’s

Figure 4 illustrates “a VLAN communications packet 400’ which is similar to the LAN communication packet 400 depicted in FIG. 3, except that a VLAN header has been added to the packet.” EX1005, 3:22-25. The source and destination addresses used for the VLAN header are duplicates of those in the packet received by the switch. EX1003, ¶180.

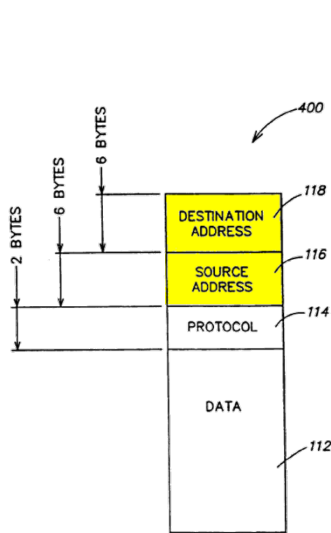


FIG. 3

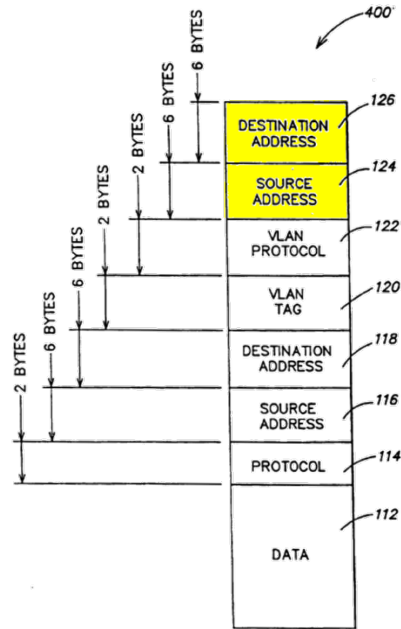


FIG. 4

Rijhsinghani, Figs. 3 and 4 (highlighted)

Rijhsinghani expressly states that “[t]he VLAN header, as shown [in Figure 4], includes a destination address 126 which is the same address as the destination address 118 [and] source address 124 which is the same as source address 116....” EX1005, 3:29-33. Thus, Rijhsinghani’s switches *duplicate a part of a header of the identified packet* (source and destination address). EX1003, ¶180.

Further, the switches do so when the received packet is associated with a VLAN and will have a VLAN header added to it. EX1003, ¶181; EX1005, 3:22-26 (“The VLAN header is added by the initial switch to which the message packet is directed.”), 3:26-29 (“The VLAN header identifies the resulting packet as a ‘VLAN’ or ‘tagged’ packet....”). As discussed for 1f, not all packets are associated with a VLAN. EX1005, 8:16-18. For instance, “communications from a system user which is a member of the default group will not be tagged with a VLAN header....” *Id.*, 8:18-23; *see also id.*, 3:62-67, 8:23-34. Similarly, “where the sender and addressee are each members of a LAN connected to the same switch, there is no need to add a VLAN header to the communication....” *Id.*, 9:13-16; *see also id.*, 9:25-30. Thus, the switch will *duplicate a part of a header of the identified packet...when the identified packet comprises a target of encapsulation* (i.e., when the packet will have an additional header added to it). *See* Section V.A, above (construction of “encapsulation”).

The switch will *duplicate a part of a header of the identified packet as an additional header* as claimed. As discussed above, this limitation means that when a packet is identified as a target for encapsulation (i.e., when it will have an additional header added), the switch duplicates part of the existing header to use as a portion of the added encapsulation header. *See* Section V.B, above (construction of this claim language). Rijhsinghani discloses exactly that, by duplicating part of

the header information in the received packet so that it can be used as part of the VLAN header added to the packet. EX1003, ¶182.

As illustrated in Figures 3 and 4, the received packet's source address 116 and destination address 118 are duplicated for use in the VLAN header as source address 124 and destination address 126. EX1005, 3:29-33. The VLAN header constitutes the *additional header* recited in the claim. EX1003, ¶183.

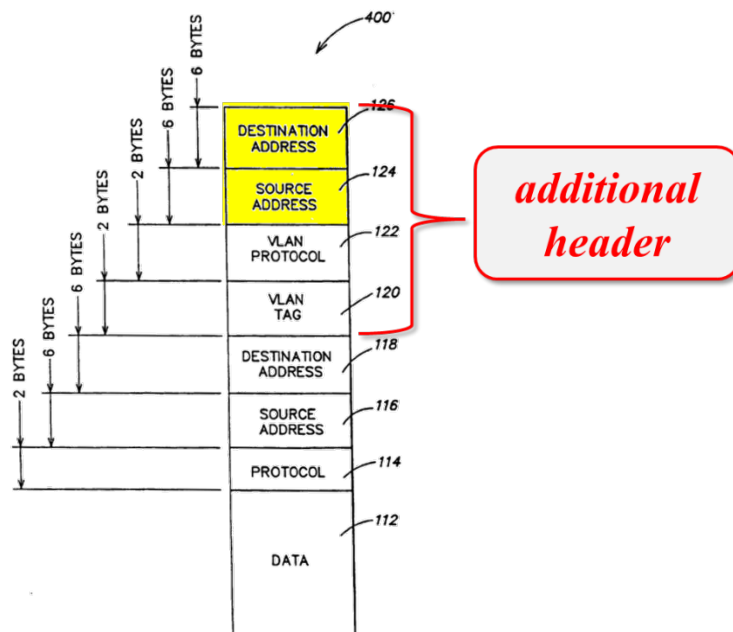


FIG. 4

Rijhsinghani, Fig. 4 (annotated)

Thus, the switch *duplicate[s] a part of a header of the identified packet* (source address and destination address) to use *as an additional header* (in the new VLAN header) *when the identified packet comprises a target of encapsulation* (when the packet will have a VLAN header added). *Id.*, ¶183.

To the extent the Patent Owner may argue that the *additional header* is limited to the duplicated information, then the source and destination addresses within the VLAN header would constitute such an *additional header*. EX1003, ¶184.

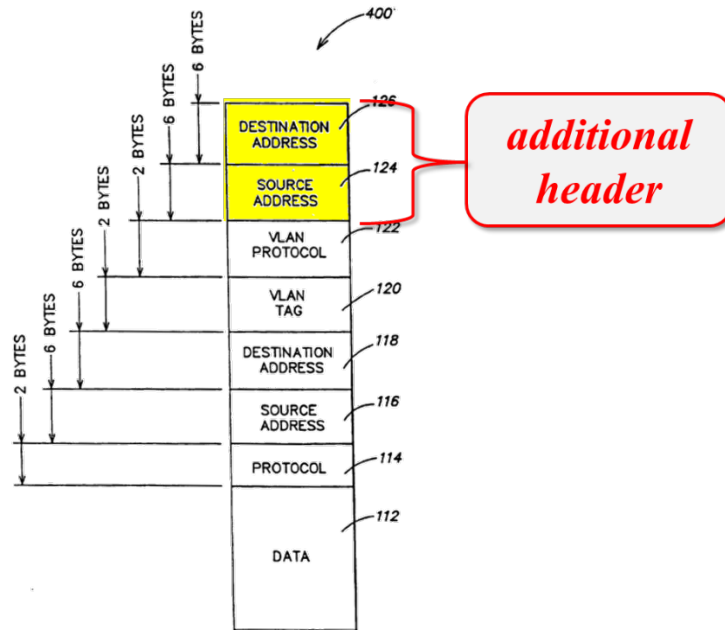


FIG. 4

Rijhsinghani, Fig. 4 (annotated)

Thus, Rijhsinghani's switches *duplicate a part of a header of the identified packet as an additional header when the identified packet comprises a target of encapsulation* as recited in claim 1. EX1003, ¶185.

- i. **[1h] encapsulate the identified packet by the additional header; and**

Rijhsinghani discloses this limitation. EX1003, ¶¶186-189. Rijhsinghani's switches *encapsulate the identified packet by the additional header* by adding a

VLAN header, which Rijhsinghani also refers to as a VLAN tag, to the packet.

EX1003, ¶187; EX1005, 1 (Abstract) (“The communication is appended with a VLAN tag representing the identified virtual local area network so as to form a VLAN communication...”), 3:23-27 (“The VLAN header is added by the initial switch to which the message packet is directed.”), 3:39-43, Figs. 3-4.

Rijhsinghani states that “[a]fter the VLAN has been identified, the switch control adds a VLAN tag representing the identified VLAN to form a VLAN communication.” EX1005, 7:4-7. In particular, the switch’s control module 284 includes “tagger 284c for tagging communications received via an access port 305 for transmission from one member to another member of a configured VLAN by adding a VLAN header thereto.” *Id.*, 8:60-63; *see also id.*, 9:17-22, 9:25-29 (“This is accomplished by identifying communications between LANs within configured VLANs and tagging the communications...with a VLAN header.”), 9:45-54, 9:59-10:44, 11:46-47.

The flow chart of Figure 10 shows that, after the packet is associated with one or more VLANs, “[i]n step 1340, the communication is tagged with the VLAN header representing the highest priority identified VLAN.” EX1005, 11:5-7.

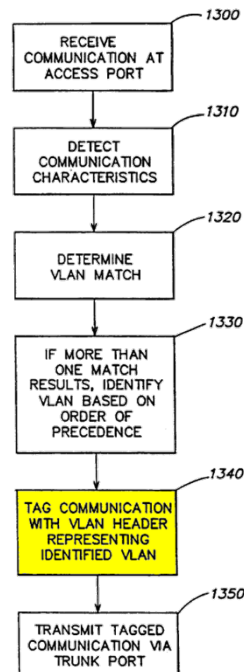


FIG. 10

Rijhsinghani, Fig. 10 (highlighted)

This step constitutes *encapsulat[ing] the identified packet by the additional header* as claimed. EX1003, ¶189; see Section V.A, above (construction of “encapsulate”).

j. [1i] process, based on the actions, the identified packet.

Rijhsinghani discloses this limitation. EX1003, ¶¶190-192. Rijhsinghani’s switches *process...the identified packet* by forwarding it, and they do so *based on the actions* because the packet is forwarded based on its association with a particular VLAN. EX1003, ¶191. Rijhsinghani states that, after the appropriate VLAN is identified and the VLAN header is added, “packet 400’ is then directed via trunk port 315 to the high-speed backbone LAN 265 and detected by

switches 275 and 280.” EX1005, 3:39-45; *see also id.*, 7:1-9 (“After the VLAN has been identified, the switch control adds a VLAN tag representing the identified VLAN to form a VLAN communication. The switch control then directs the VLAN communication to the second communication port for transmission over the identified virtual local area network.”), 11:47-51 (“[T]he tagger 284c of switch 275’ tags the packet with the VLAN header representing VLAN 1100. Last, the switch 275’ transmits the tagged packet via the trunk port 315 onto the high-speed backbone 265 to be forwarded to system user at address V in the conventional manner.”).

The flow chart of Figure 10 shows that, after the packet is tagged with the VLAN header, “[i]n step 1350, the tagged communication is transmitted via the trunk port 315 from the switch and from there forwarded to the appropriate addressee or addressees in the conventional manner.” EX1005, 11:7-10.

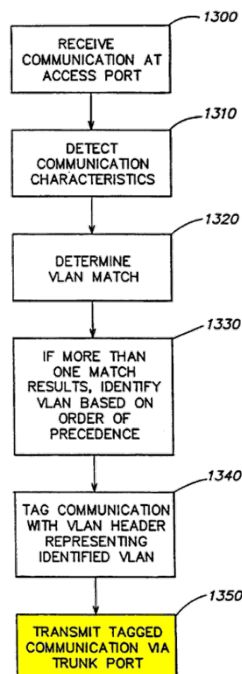


FIG. 10

Rijhsinghani, Fig. 10 (highlighted)

This step constitutes *process[ing]*, based on the actions, the identified packet as claimed. EX1003, ¶192.

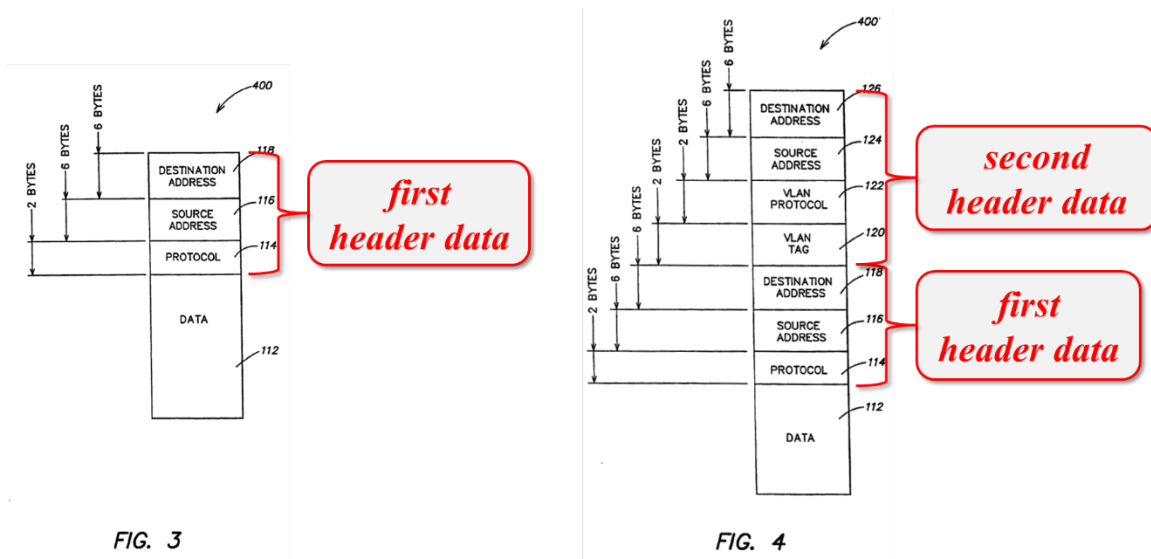
2. Claim 5

Claim 5 reads as follows: *The switch apparatus according to claim 1, wherein the processor is further configured to execute the instructions for translation of first header data of a head of the received packet into second header data.* Rijhsinghani discloses claim 5. EX1003, ¶¶193-195.

As discussed for 1d, Rijhsinghani’s switches include control module 284 that is executed by *the processor*. See Section VII.A.1.e, above. Control module 284 includes detector 284b, which “detects a communication packet

received from the backbone 265 via a trunk port 315 or from a LAN directly connected to the switch via an access port 305.” EX1005, 8:12-15. “The control module 284 also includes a tagger 284c for tagging communications received via an access port 305 for transmission from one member to another member of a configured VLAN by adding a VLAN header thereto.” *Id.*, 8:60-63.

As discussed for 1g, Rijhsinghani’s switch (control module 284) creates the VLAN header using copies of the source address and destination address of the packet received by the switch (by detector 284b). See Section VII.A.1.h, above.



Rijhsinghani, Figs. 3 and 4 (annotated)

Thus, Rijhsinghani’s control module 284, executed by *the processor*, performs a *translation of first header data of a head of the received packet* (including source and destination address from the received packet as in Figures 3 and 4) *into second header data* (VLAN header as in Figure 4), as recited in claim 5. EX1003, ¶195.

3. Claim 6

Dependent claim 6 reads as follows: *The switch apparatus according to claim 5, wherein the instructions to encapsulate the identified packet includes notifying the controller of the first header data before the translation and the second header data after the translation.* Rijhsinghani discloses claim 6. EX1003, ¶¶196-198.

Claim 6 recites *notifying the controller of the first header data and the second header data*. The term *the controller* in claim 6 refers to the *controller* recited in claim 1. In particular, 1b recites *a controller* included in the *switch apparatus*. As discussed for 1b, Rijhsinghani's control console 288 constitutes the *controller* of claim 1. See Section VII.A.1.c, above.

Rijhsinghani's control console 288 (*the controller*) includes detector 284b, which detects a received packet including its source address and destination address (*the first header data before the translation*). EX1005, 8:12-15. The control console 288 (*the controller*) also includes tagger 284c, which adds the VLAN header to the packet (*the second header data after the translation*). EX1003, ¶198; EX1005, 8:60-63. Thus, *the instructions to encapsulate the identified packet includes the detector 284b and tagger 284c notifying the controller (control console 288) of the first header data before the translation (source address and destination address of the received packet) and the second*

header data after the translation (VLAN header). EX1003, ¶198.

4. Claim 7

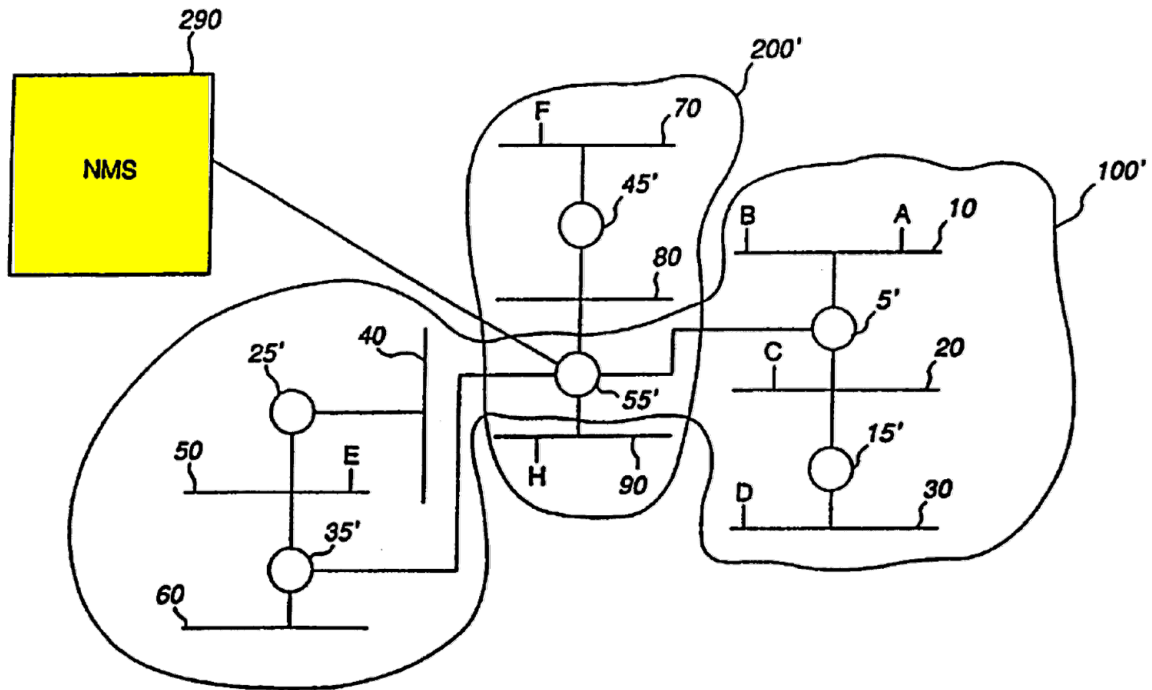
a. A communication system, comprising:

Rijhsinghani discloses a *communication system*, as recited in the preamble. EX1003, ¶¶199-200. For example, Rijhsinghani refers to a switch for use in “a communications system having multiple local area networks interconnected by multiple switches so as to be configurable into different types of virtual local area networks...” EX1005, 1 (Abstract). Under “Field of the Invention,” Rijhsinghani states that “[t]he present invention relates generally to communications networks and more particularly to communications systems having various types of virtual local area networks and established rules of precedence for matching a communication packet with a particular virtual local area network.” *Id.*, 1:8-12; *see also id.*, 6:42-46 (“a switch is provided for use in a virtual communications system”), 7:10-12, 7:49-50 (“FIG. 7 depicts a virtual communications system or network in accordance with the present invention.”).

b. a control apparatus configured to control a plurality of switch apparatuses; and

Rijhsinghani discloses this limitation. EX1003, ¶¶201-204. As explained for 1e, Rijhsinghani discloses a network management station (NMS 290 or NMS 290') constituting *a control apparatus*. *See* Section VII.A.1.f, above. The *control apparatus* is illustrated in Figure 2 in which “NMS 290 has configured two

VLANs by instructing, e.g., programming, and thereby configuring the switches 5'-55' such that LANs 10-60 are bridged together by switches 5', 15', 55', 35' to form VLAN 100' and LANs 70-90 are bridged together by switches 45' and 55' to form VLAN 200'." EX1005, 2:14-19; *see also id.*, 2:62-3:22, Fig. 5.

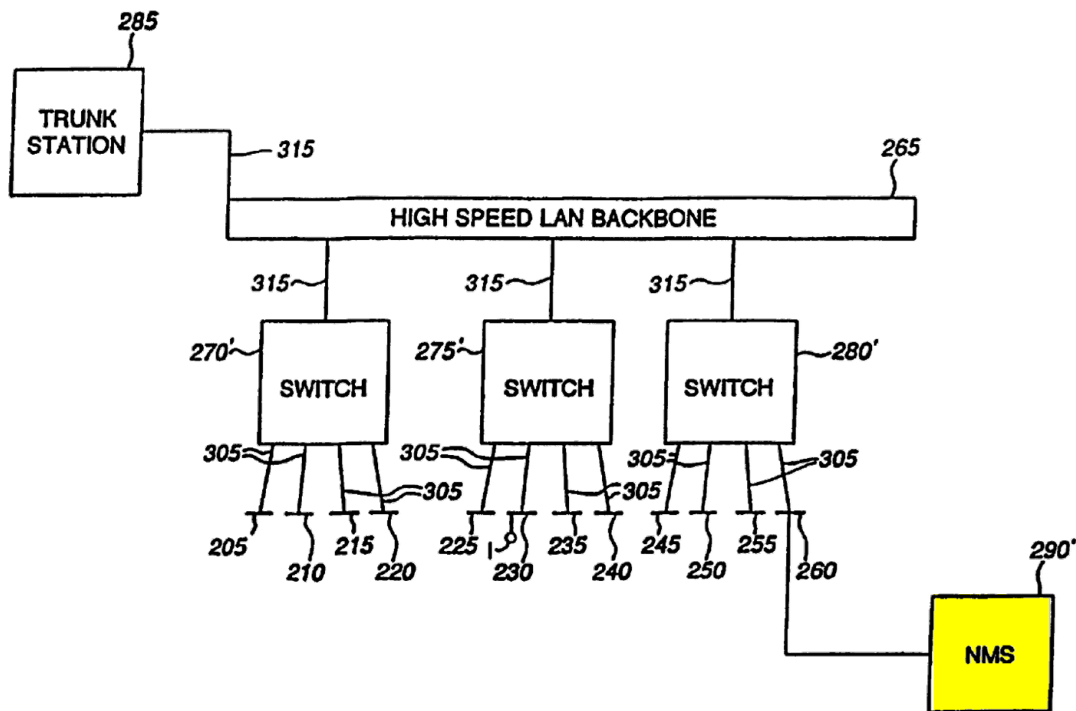


Rijhsinghani, Fig. 2 (highlighted)

Thus, NMS 290 constitutes *a control apparatus configured to control a plurality of switch apparatuses*. EX1003, ¶203.

In addition, Figure 7 illustrates “[a] network management station (NMS) 290’, which may be a workstation having network management software loaded thereon, manages the network by configuring the network via the switches 270’, 275’ and 280’ to form one or more virtual local area network

(VLANs).” EX1005, 7:58-63.



Rijhsinghani, Fig. 7 (highlighted)

Thus, NMS 290' also constitutes *a control apparatus configured to control a plurality of switch apparatuses*. EX1003, ¶204.

c. a switch apparatus of the switch apparatuses comprising:

Rijhsinghani discloses this limitation. EX1003, ¶¶205-209. Rijhsinghani discloses *a switch apparatus* as discussed for 1pre. See Section VII.A.1.a, above. Each switch is one of several that are controlled by a network management station (*control apparatus*) such that each is *a switch apparatus of the switch apparatuses* (i.e., the *plurality of switch apparatuses* controlled by the *control apparatus*).

EX1003, ¶¶207-208 (citing EX1005, 2:14-19, 2:62-65, 3:8-22, 7:49-55,

Figs. 2, 5, 7).

d. a storage storing a table, the table including rules and actions corresponding to the rules; and

This limitation is identical 1a. EX1003, ¶210 (comparing claims 1 and 7).

It is disclosed for at least the reasons discussed for 1a. See Section VII.A.1.b, above.

e. a controller comprising:

This limitation is identical 1b. EX1003, ¶211. It is disclosed for at least the reasons discussed for 1b. See Section VII.A.1.c, above.

f. a memory storing instructions; and

This limitation is identical 1c. EX1003, ¶212. It is disclosed for at least the reasons discussed for 1c. See Section VII.A.1.d, above.

g. a processor configured to execute the instructions to:

This limitation is identical 1d. EX1003, ¶213. It is disclosed for at least the reasons discussed for 1d. See Section VII.A.1.e, above.

h. receive the rules and the actions from a controller;

Rijhsinghani discloses this limitation. EX1003, ¶¶214-219. The reference to *a controller* in this limitation refers to an external controller, such as the *control apparatus* of claim 1 and the first limitation of claim 7, rather than the *controller* included in the *switch apparatus* as recited in a prior limitation of claim 7.

EX1003, ¶215-218. This limitation therefore means that the switch *receive[s] the*

rules and the actions from a controller such as the *control apparatus*, making it equivalent in scope to 1e. *Id.*, ¶219. It is disclosed for at least the reasons discussed for 1e. See Section VII.A.1.f, above.

i. identify, based on the rules, a received packet;

This limitation is identical to 1f. EX1003, ¶220. It is disclosed for at least the reasons discussed for 1f. See Section VII.A.1.g, above.

j. duplicate a part of a header of the identified packet as an additional header when the identified packet comprises a target of encapsulation;

This limitation is identical to 1g. EX1003, ¶221. It is disclosed for at least the reasons discussed for 1g. See Section VII.A.1.h, above.

k. encapsulate the identified packet by the additional header; and

This limitation is identical to 1h. EX1003, ¶222. It is disclosed for at least the reasons discussed for 1h. See Section VII.A.1.i, above.

l. process, based on the actions, the identified packet.

This limitation is identical to 1i. EX1003, ¶223. It is disclosed for at least the reasons discussed for 1i. See Section VII.A.1.j, above.

5. Claim 11

Claim 11's additional limitation is virtually identical to claim 5's. EX1003, ¶65 (comparing claims 5 and 11). It is disclosed as discussed for that claim. See Section VII.A.2, above.

6. Claim 12

Dependent claim 12 reads as follows: *The communication system according to claim 11, wherein the instructions to encapsulate the identified packet includes notifying the controller of first header data before the translation and second header data after the translation.*

Claim 12 recites *notifying the controller*, which is ambiguous because claim 7 (from which claim 12 indirectly depends) recites two different controllers. First, claim 7 recites *a switch apparatus of the switch apparatuses comprising...a controller* (i.e., *a controller* in the *switch apparatus*). Second, claim 7 recites that *the switch apparatus...receive[s] the rules and the actions from a controller*, which is an external controller such as the *control apparatus* of claim 7. See Section VII.A.4.h, above.

Claim 12 is disclosed by the prior art under either construction. If *the controller* of claim 12 is the controller included in the *switch apparatus*, then claim 12 is similar in scope to claim 6. EX1003, ¶67 (comparing claims 6 and 12). Under this construction, claim 12 is disclosed by Rijhsinghani for the reasons discussed for claim 6. See Section VII.A.3, above. Under the other construction, claim 12 is disclosed by the prior art combinations addressed in Sections VII.B.8 and VII.C.8, below.

7. Claim 13

a. A communication method, comprising:

Rijhsinghani discloses a *communication method*, as recited in the preamble, for at least the reasons discussed for 1e-1i. See Sections VII.A.1.f-VII.A.1.j, above; EX1003, ¶229.

b. receiving rules and actions from a controller;

This limitation of claim 13 is virtually identical to the corresponding limitation in claim 7 and disclosed for the same reasons. See Section VII.A.4.h, above; EX1003, ¶230. As in claim 7, the reference in claim 13 to *a controller* means an external controller such as *a control apparatus* of claim 7. See EX1003, ¶¶230-232. Because this limitation of claim 13 recites *receiving the rules and the actions from a controller*, which can be an external controller such as a *control apparatus*, it is similar to (and at least as broad in scope as) 1e and disclosed for the same reasons. See Section VII.A.1.f, above.

c. identifying, based on the rules, a received packet;

This limitation is virtually identical to 1f. EX1003, ¶57 (comparing claims 1 and 13). It is disclosed for at least the reasons discussed for 1f. See Section VII.A.1.g, above.

d. duplicating a part of a header of the identified packet as an additional header when the identified packet comprises a target of encapsulation;

This limitation is virtually identical to 1g. EX1003, ¶234. It is disclosed for

at least the reasons discussed for 1g. *See* Section VII.A.1.h, above.

e. encapsulating the identified packet by the additional header; and

This limitation is virtually identical to 1h. EX1003, ¶235. It is disclosed for at least the reasons discussed for 1h. *See* Section VII.A.1.i, above.

f. processing, based on the actions, the identified packet.

This limitation is virtually identical to 1i. EX1003, ¶236. It is disclosed for at least the reasons discussed for 1i. *See* Section VII.A.1.j, above.

8. Claim 19

a. A non-transitory recording medium recording a program which when executed causes a switch to perform a method, the method comprising:

Rijhsinghani discloses the preamble for at least the reasons discussed for 1c-1i. *See* Sections VII.A.1.d-VII.A.1.j, above; EX1003, ¶¶154-192. In particular, Rijhsinghani discloses switches that include memory 286 (*[a] non-transitory recording medium recording a program*). *See* Section VII.A.1.d, above. And “each of the switches 270', 275' and 280' includes a control console 288 having...a memory 286 for storing and processing control and VLAN configuration instructions.” EX1005, 8:6-9, *see also id.*, 9:42-46 (“instructions may be stored in the memory 286 of the switch”). A processor executes instructions stored in the memory to perform each of the functions recited in 1e to 1i, which mirror the steps recited in claim 19. *See* Sections VII.A.1.f-VII.A.1.j,

above.

b. receiving rules and actions from a controller;

This limitation is identical to the corresponding limitation in claim 13 and virtually identical to the corresponding limitation in claim 1 (limitation 1e).

EX1003, ¶57 (comparing claims 1 and 13) and ¶58 (comparing claims 13 and 19).

It is disclosed for the reasons discussed for those claims. *See* Sections VII.A.1.f and VII.A.7.b, above.

c. identifying, based on the rules, a received packet;

This limitation is identical to the corresponding limitation in claim 13 and virtually identical to the corresponding limitation in claim 1 (limitation 1f).

EX1003, ¶¶57 and 58. It is disclosed for the reasons discussed for those claims.

See Sections VII.A.1.g and VII.A.7.c, above.

d. duplicating a part of a header of the identified packet as an additional header when the identified packet comprises a target of encapsulation;

This limitation is identical to the corresponding limitation in claim 13 and virtually identical to the corresponding limitation in claim 1 (limitation 1g).

EX1003, ¶¶57 and 58. It is disclosed for the reasons discussed for those claims.

See Sections VII.A.1.h and VII.A.7.d, above.

e. encapsulating the identified packet by the additional header; and

This limitation is identical to the corresponding limitation in claim 13 and

virtually identical to the corresponding limitation in claim 1 (limitation 1h).

EX1003, ¶¶57 and 58. It is disclosed for the reasons discussed for those claims.

See Sections VII.A.1.i and VII.A.7.e, above.

f. processing, based on the actions, the identified packet.

This limitation is identical to the corresponding limitation in claim 13 and virtually identical to the corresponding limitation in claim 1 (limitation 1i).

EX1003, ¶¶57 and 58. It is disclosed for the reasons discussed for those claims.

See Sections VII.A.1.j and VII.A.7.f, above.

B. Ground 2: The Combination of Rijhsinghani and RFC-5474 Renders Claims 2-4, 8-10, 12, 14-18, and 20 Obvious

Each of dependent claims 2-4, 8-10, 12, 14-18, and 20 arguably include limitations regarding a switch apparatus sending header information to an external controller. Rijhsinghani discloses communication systems in which a network management station (NMS) configures switches to implement virtual local area networks (VLANs). Rijhsinghani describes the external NMS sending instructions to the network switches but does not expressly discuss the switches sending information to the external NMS. However, RFC-5474 describing the PSAMP protocol discloses a well-known, prior art technique used by switches and other network devices to send header information to an external controller such as Rijhsinghani's NMS. Because a POSITA would have been motivated to implement the PSAMP protocol in Rijhsinghani's network switches as discussed

below, the combination of Rijhsinghani and RFC-5474 renders these claims obvious. EX1003, ¶¶244-282.

1. Motivation to Combine

A POSITA would have considered RFC-5474 because it is in the same field of endeavor as the '177 Patent and Rijhsinghani, namely network communications. EX1003, ¶246; EX1006, 2; EX1005, 1 (Abstract), 1:8-12, 6:26-35; EX1001, 1:14-16, 4:33-38. RFCs are a primary source for understanding standardized techniques used in network communications, as a POSITA would have known. EX1003, ¶246. A POSITA implementing a communications network disclosed by Rijhsinghani would have naturally looked to RFC-5474 (and other RFCs) for standard network communications techniques to permit interoperability within the network and with other networks. *Id.*, ¶246; EX1006, 4 (“The motivation for the PSAMP standard comes from the need for measurement-based support for network management and control across multivendor domains.”); EX1018.

Rijhsinghani itself provides further motivation to consider RFC-5474. Rijhsinghani explains that “LAN network administrators generally attempt to connect together those users who frequently communicate with each other in bridged LANs.” EX1005, 1:45-47. “If, for example, in FIG. 1, user E and user A need to communicate frequently, it would be advantageous to interconnect LAN 10 and LAN 50 via a bridge rather than a router 300,” but “[t]his would require the

rewiring of the system which is costly and may be impracticable under many circumstances, such as, if users A and E will only need to frequently communicate for a limited period of time.” *Id.*, 1:53-60. In contrast, with a VLAN system, as discussed in Rijhsinghani, “the bridging of LANs can be changed as desired without the need to rewire the network.” *Id.*, 1:67-2:1. “A network management station (NMS) 290 controls the interconnection of the individual LANs such that LANs can be easily bridged to other LANs on a long term or short term basis without the need to rewire the network.” *Id.*, 2:10-14. Thus, Rijhsinghani suggests and teaches that network administrators and designers were motivated to understand network traffic in order to configure or reconfigure network elements. EX1003, ¶247.

One common technique used for configuring and reconfiguring networks including LANs and VLANs is called network traffic measurement or analysis. EX1003, ¶248; EX1025, 1 (“Traffic measurement is a critical component for the control and engineering of communication networks.”). Network traffic analysis is based on information provided by network nodes, such as switches and routers, about communications flowing through a network. EX1003, ¶248. By 2011, packet-sampling techniques as disclosed by RFC-5474 were well known and commonly used. *Id.*; EX1024; EX1025; EX1026; EX1028, 3.

A POSITA would have recognized the benefit of implementing switches

compliant with the PSAMP protocol of RFC-5474. EX1003, ¶249. Such a person could have implemented Rijhsinghani’s VLAN system using commercially available switches, which would have allowed network administrators to use Rijhsinghani’s NMS with existing network traffic analysis tools to understand network traffic and to configure VLANs in the system as desired. *Id.*, ¶249; EX1006, 29 (“Packet Sampling is ideally suited to determine the composition of the traffic across a network.”); EX1027, 33-34. RFC-5474 states that the PSAMP framework requirements were “motivated by the dual aims of ubiquitous deployment and utility of the reports for applications.” EX1006, 2.

A POSITA would also recognize that implementing the PSAMP protocol in Rijhsinghani’s network would aid in troubleshooting network problems. EX1003, ¶250. RFC-5474 states that “PSAMP Packet Reports can also be used to diagnose problems whose occurrence is evident from aggregate statistics, per interface utilization and packet loss statistics.” EX1006, 30-31. Additionally, a POSITA would have understood that monitoring network traffic using PSAMP could be used to create summaries of network usage for billing or capacity planning. EX1003, ¶250.

Further, a POSITA could have implemented the PSAMP protocol of RFC-5474 in Rijhsinghani’s network system without undue experimentation and with a reasonable expectation of success. EX1003, ¶¶251.

2. Claim 2

Dependent claim 2 reads as follows: *The switch apparatus according to claim 1, wherein the processor is further configured to execute the instructions to send header information to the control apparatus when the identified packet comprises an encapsulated packet.* The combination of Rijhsinghani and RFC-5474 discloses claim 2. EX1003, ¶¶252-254.

In the combination of Rijhsinghani and RFC-5474, the *switch apparatus* disclosed by Rijhsinghani would be implemented to comply with the PSAMP protocol disclosed by RFC-5474. EX1003, ¶253. Such a *switch apparatus* is a “PSAMP Device” that selects packets and uses them to generate “Packet Reports” that are sent to a remote “Collector,” which would be located at Rijhsinghani’s network management station (*control apparatus*). *Id.*, ¶253; EX1006, 5, 9. Each Packet Report must include “a basic report on the packet, i.e., some number of continuous bytes from the start of the packet, including the packet header (which includes network layer and any encapsulation headers)....” EX1006, 19-20. Thus, in the combination, Rijhsinghani’s *switch apparatus* would be *configured...to send header information to the control apparatus* in compliance with RFC-5474. EX1003, ¶253.

Further, RFC-5474 describes a number of techniques for a PSAMP Device to select packets for reporting. EX1006, 15-17. One option is a technique called

“Property Match Filtering” in which “a packet is selected if a specific field within the packet...equal(s) a predefined value.” *Id.*, 14. RFC-5474 states: “Property match operations should be available for different protocol portions of the packet header” including “encapsulation headers....” *Id.*, 15. Rijhsinghani’s VLAN header (as illustrated in Figure 4) is the “encapsulation header.” EX1003, ¶254. Thus, Rijhsinghani’s switch apparatus employing property match filtering based on the VLAN header (encapsulation header) to select a packet for reporting—as taught by RFC-5474—would *send header information to the control apparatus when the identified packet comprises an encapsulated packet* as recited in claim 2. EX1003, ¶254.

3. Claim 3

Dependent claim 3 reads as follows: *The switch apparatus according to claim 2, wherein the header information includes the additional header.* The combination of Rijhsinghani and RFC-5474 discloses claim 3. EX1003, ¶¶255-256.

As explained for claim 2, RFC-5474 specifies that each Packet Report must include “any encapsulation headers.” EX1006, 19-20. In Rijhsinghani’s VLAN communication packets (illustrated in Figure 4), the “encapsulation header” referred to in RFC-5474 would be the VLAN header, which is or includes the *additional header* of 1g. See Section VII.A.1.h, above. Because the Packet Report

necessarily includes “any encapsulation headers,” *the header information* sent by Rijhsinghani’s switch *includes the additional header*. EX1003, ¶256.

4. Claim 4

Dependent claim 4 reads as follows: *The switch apparatus according to claim 3, wherein the processor is further configured to execute further instructions to decapsulate the encapsulated packet*. The combination of Rijhsinghani and RFC-5474 discloses claim 4. EX1003, ¶¶257-258.

Rijhsinghani describes removing the VLAN header from VLAN communication packets (*decapsulat[ing] the encapsulated packet*). EX1003, ¶258; Section V.A, above (construction of “decapsulate”). Rijhsinghani’s control module 284 includes “a tagger 284c for tagging communications received via an access port 305 for transmission from one member to another member of a configured VLAN by adding a VLAN header thereto.” EX1005, 8:60-63. In addition, “tagger 284c also removes the VLAN header from a communication received from the switch’s trunk port 315 that is to be forwarded to a member of a LAN connected to the switch by an access port.” *Id.*, 8:63-67. “More particularly, the tagger 284c discards the tag by removing the VLAN header from the communication, prior to the communication being transmitted to the appropriate output port 305, i.e., prior to the controller 284a controlling the switching device 282 to transmit the communication from the trunk port 315 to the access

port 305.” *Id.*, 8:67-9:6; *see also id.*, 3:50-54 (“The switch 275 accordingly removes the VLAN header and directs the packet, which now appears as packet 400 of FIG. 3, to LAN 230 over which the member I receives the message.”).

5. Claim 8

Claim 8’s additional limitation is identical to claim 2’s. EX1003, ¶59 (comparing claims 2 and 8). It is disclosed as discussed for that claim. *See* Section VII.B.2, above.

6. Claim 9

Claim 9’s additional limitation is identical to claim 3’s. EX1003, ¶61 (comparing claims 3 and 9). It is disclosed as discussed for that claim. *See* Section VII.B.3, above.

7. Claim 10

Claim 10’s additional limitation is identical to claim 4’s. EX1003, ¶63 (comparing claims 4 and 10). It is disclosed as discussed for that claim. *See* Section VII.B.4, above.

8. Claim 12

As discussed in Section VII.A.6, above, claim 12’s reference to *notifying the controller* is ambiguous and could mean that the *switch apparatus...receive[s] the rules and the actions from a controller*, which is an external controller such as the *control apparatus* of claim 7. The combination of Rijhsinghani and RFC-5474 discloses claim 12 under this construction. EX1003, ¶¶266-270.

As explained for claim 5, Rijhsinghani’s control module 284 translates *first header data before the translation* (including source and destination address in the received packet as in Figures 3 and 4) into *second header data after the translation* (VLAN header in Figure 4). See Section VII.A.2, above. Thus, if *the controller* of claim 12 is located at an external control apparatus—as opposed to in the switch—then claim 12 is similar in scope to claims 2 and 3. EX1003, ¶268. Those claims recite *send[ing] header information to the control apparatus when the identified packet comprises an encapsulation packet* (claim 2) and that *the header information includes the additional header* (claim 3).

According to RFC-5474, each Packet Report sent to the external controller includes “the packet header (which includes network layer and any encapsulation headers)” (EX1006, 19-20), which therefore includes both *first header data before the translation and second header data after the translation*, as shown in the figure below. EX1003, ¶269.

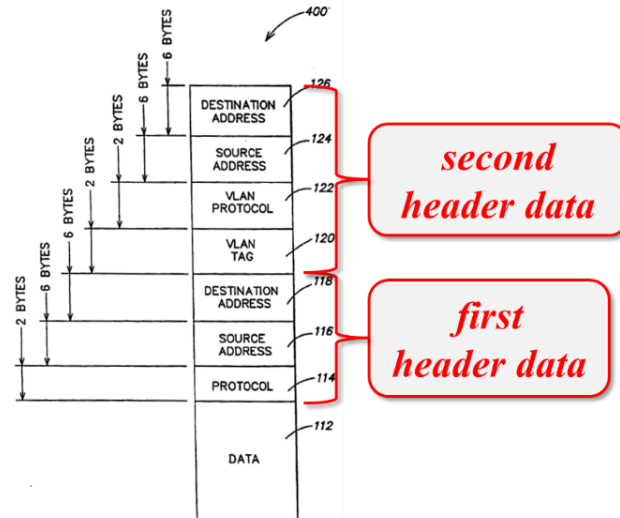


FIG. 4

Rijhsinghani, Fig. 4 (annotated)

Thus, for at least the reasons discussed for claims 2, 3, and 5, the combination of Rijhsinghani and RFC-5474 discloses claim 12 (based on the construction of *the controller* assumed in this section). *Id.*, ¶270; Sections VII.B.2, VII.B.3, and VII.A.2, above.

9. Claim 14

Claim 14’s additional limitation is similar to claim 2’s. EX1003, ¶60 (comparing claims 2 and 14). The primary difference is that claim 2 recites sending header information “to the control apparatus” while claim 14 sends “to a controller.” As in claims 7 and 13, claim 14’s reference to *a controller* means an external controller such as a *control apparatus*. See Sections VII.A.4.h and VII.A.7.b, above. Thus, claim 14 has essentially the same scope as claim 2 and is disclosed for the same reasons. See Section VII.B.2, above.

10. Claim 15

Claim 15's additional limitation is identical to claim 3's. EX1003, ¶62 (comparing claims 3 and 15). It is disclosed as discussed for that claim. See Section VII.B.3, above.

11. Claim 16

Claim 16's additional limitation is virtually identical to claim 4's. EX1003, ¶64 (comparing claims 4 and 16). It is disclosed as discussed for that claim. See Section VII.B.4, above.

12. Claim 17

Claim 17's additional limitation is virtually identical to claim 5's. EX1003, ¶66 (comparing claims 5 and 17). It is disclosed as discussed for that claim. See Section VII.A.2, above.

13. Claim 18

Claim 18's additional limitation is virtually identical to claim 6's. EX1003, ¶68 (comparing claims 6 and 18). It is disclosed as discussed for that claim. See Section VII.A.3, above.

14. Claim 20

Claim 20's additional limitations are virtually identical to the limitations in claims 14 and 15. EX1003, ¶69 (comparing claims 14 and 15 to claim 20). It is disclosed as discussed for those claims. See Sections VII.B.9 and VII.B.10, above.

C. Ground 3: The Combination of Rijhsinghani and Lean Renders Claims 2-4, 8-10, 12, 14-18, and 20 Obvious

As discussed above, each of dependent claims 2-4, 8-10, 12, 14-18, and 20 arguably include limitations regarding a switch sending header information to an external controller. Rijhsinghani describes an external controller—the NMS—sending instructions to the network switches but does not expressly discuss the switches sending information to the NMS. However, Lean describes techniques for switches and other network devices to send copies of packets, including their header information, to an external controller such as Rijhsinghani’s NMS. Because a POSITA would have been motivated to implement Lean’s techniques in Rijhsinghani’s network switches, as discussed below, the combination of Rijhsinghani and Lean renders these claims obvious. EX1003, ¶¶283-326.

1. Motivation to Combine

A POSITA would have considered Lean because it is in the same field as the ’177 Patent and Rijhsinghani, namely network communications. EX1003, ¶285; EX1007, 1 (Abstract), [0002]; EX1005, 1 (Abstract), 1:8-12, 6:26-35; EX1001, 1:14-16, 4:33-38. Further, Lean contemplates monitoring tunneled network traffic including encapsulated packets. EX1007, [0002], [0005], [0007]-[0008]. Both the ’177 Patent and Rijhsinghani also involve transmitting encapsulated packets. EX1005, 3:23-34, Fig. 4; EX1001, 3:31-59, 6:48-54.

Rijhsinghani itself provides further motivation to consider Lean. As

discussed above, Rijhsinghani suggests and teaches that network administrators and designers were motivated to understand network traffic in order to configure or reconfigure network elements. *See* Section VII.B.1, above; EX1003, ¶286. One common technique to configure and reconfigure networks including LANs and VLANs is called network traffic measurement or analysis. EX1003, ¶287; EX1025, 1 (“Traffic measurement is a critical component for the control and engineering of communication networks.”). Network traffic analysis is based on information provided by network nodes, e.g. switches and routers, about communications flowing through a network. EX1003, ¶287. By 2011, systems that passively copied packets traversing a network connection for external traffic analysis, as in Lean, were common and well-known. EX1003, ¶287; EX1019, 1:14-2:19, 2:55-67, 5:31-64, 8:5-11, Figs. 1-4; EX1020, 1 (Abstract), 1:13-2:10, 2:13-27, 3:14-33, Figs. 1-5; EX1021, 1 (Abstract), [0005]-[0019], Fig. 1-3.

Lean discusses several packet-monitoring techniques that, when implemented in Rijhsinghani’s network system, would permit a network administrator to monitor network traffic in order to configure and reconfigure LANs and VLANs in the network system. EX1003, ¶¶288-290. For example, Lean discloses two embodiments of “conventional” packet monitoring systems in Figures 1 and 2. EX1007, [0018]-[0020]. In either, “taps 104” copy packets traversing the network and forward them to an external processor (Figure 1) or

processors (Figure 2) for monitoring and analysis. *Id.*, [0003]-[0004], [0007], [0018]-[0020]. As Lean explains, “[e]ach tap 104 passively copies all IP packets transmitted across Ethernet link 100 and sends the IP packet copy stream 106 to processor 108 for analysis.” *Id.*, [0018]. A POSITA would recognize that either of Lean’s “conventional” monitoring techniques could be implemented in Rijhsinghani’s network system so that network switches would passively copy packets and send them to the NMS where the traffic could be analyzed. EX1003, ¶289. Such a combination would permit a network administrator to configure and reconfigure the LANs and VLANs in Rijhsinghani’s network system. *Id.*⁸

In addition to conventional systems, Lean also describes embodiments of a disclosed invention (e.g., Figures 6 and 7). As in the conventional systems, taps 608 “copy packets that traverse a high bandwidth IP network.” EX1007, [0031]. Copied packets are analyzed and used to segregate traffic at the external packet monitoring system. *Id.*, [0029]-[0035], Figs. 6, 7. Thus, the packet-copying functionality would be the same as in conventional systems, and Lean’s invention would only involve modifications to Rijhsinghani’s external NMS. EX1003, ¶292.

⁸ While Lean identifies certain problems with conventional techniques, Lean does not teach away from the combination presented here. EX1003, ¶291.

Lean describes taps or probes copying packets passing across the network. EX1007, [0018]-[0020], [0031]. A POSITA would have known such functionality could be implemented in Rijhsinghani’s switches. EX1003, ¶290. Indeed, a POSITA would have recognized that locating functionality for copying and forwarding packets within network switches would provide superior visibility to network traffic, as opposed to using separate appliances like taps or probes. *Id.*; EX1026 (“In a switched environment, the most effective place to monitor traffic is within the switch/router, where all the traffic will be seen. Traditional probes will only have a partial view of traffic.”).

A POSITA would have been motivated to incorporate Lean into Rijhsinghani’s network system for reasons stated in Lean. EX1003, ¶293. For example, an “advantage” of Lean’s disclosed invention is “increased IP traffic segregation granularity.” EX1007, [0038]. Lean’s invention permits “deep packet classification” based on packet information that may be at the “application, presentation, and session layer...” *Id.*, [0033]; *see also id.*, [0032]. Thus, implementing Lean’s invention in Rijhsinghani’s network system would allow network administrators to use Lean’s “deep packet classification” to segregate copied packets at a granular level, enhancing traffic monitoring. EX1003, ¶293. Further, a POSITA could have implemented Lean’s packet-monitoring techniques in Rijhsinghani’s network system without undue experimentation and with

reasonable expectation of success. EX1003, ¶294.

2. Claim 2

The combination of Rijhsinghani and Lean discloses claim 2. EX1003, ¶¶295-298. Rijhsinghani’s *switch apparatus* would be modified to perform the packet-copying functionality of the “tap” in Lean’s conventional systems (e.g., Figures 1 and 2) or in Lean’s inventive embodiments (e.g., Figures 6 and 7). *Id.*, ¶296. In Lean’s Figure 1, “tap 104 passively copies all IP packets...and sends the IP packet copy stream 106 to processor 108 for analysis.” EX1007, [0018], Fig. 1. Lean’s conventional system of Figure 2 similarly copies and forwards packets. *Id.*, [0020], Fig. 2. In the disclosed embodiments of Lean’s invention, taps 608 also copy packets traversing the network and send them for analysis on the remote packet monitoring system 600. *Id.*, [0031].

Whether a network designer chose to implement one of Lean’s conventional systems or inventive embodiments in Rijhsinghani’s network system, Rijhsinghani’s *switch apparatus* would be *configured to execute the instructions to send* copies of entire packets including *header information to the control apparatus* (Rijhsinghani’s NMS) including *when the identified packet comprises an encapsulated packet* (Rijhsinghani’s packet with VLAN header as in Figure 4). EX1003, ¶298. The combination of Rijhsinghani with any of Lean’s conventional systems or inventive embodiments therefore discloses claim 2. *Id.*, ¶298.

3. Claim 3

The combination of Rijhsinghani and Lean discloses claim 3. EX1003, ¶¶299-300. As discussed for claim 2, a network designer/administrator implementing Lean’s conventional systems or inventive embodiments would modify Rijhsinghani’s switches to copy and forward entire packets to the external NMS (i.e., to perform the function of Lean’s taps) for traffic analysis. *See* Section VII.C.2, above. Because entire packets are copied and forwarded, *the header information* forwarded to the NMS would necessarily include *the additional header*. EX1003, ¶300.

4. Claim 4

The combination of Rijhsinghani and Lean discloses claim 4. EX1003, ¶¶301-302. As discussed above, Rijhsinghani describes removing the VLAN header from VLAN communication packets (*decapsulat[ing] the encapsulated packet*). *See* Sections VII.B.4 and V.A, above.

5. Claim 8

Claim 8’s additional limitation is identical to claim 2’s. EX1003, ¶59 (comparing claims 2 and 8). It is disclosed as discussed for that claim. *See* Section VII.C.2, above.

6. Claim 9

Claim 9’s additional limitation is identical to claim 3’s. EX1003, ¶61 (comparing claims 3 and 9). It is disclosed as discussed for that claim. *See*

Section VII.C.3, above.

7. Claim 10

Claim 10's additional limitation is identical to claim 4's. EX1003, ¶63 (comparing claims 4 and 10). It is disclosed as discussed for that claim. See Section VII.C.4, above.

8. Claim 12

As discussed in Section VII.B.8, above, claim 12's reference to *notifying the controller* is ambiguous and can refer to an external controller such as the *control apparatus* as in claim 7. See Section VII.A.6, above. As explained for claim 5, Rijhsinghani's control module 284 translates *first header data before the translation* (including source and destination address in the received packet as in Figures 3 and 4) into *second header data after the translation* (VLAN header as in Figure 4). See Section VII.A.2, above. Thus, if *the controller* of claim 12 is located at an external *control apparatus*—as opposed to in the switch—then claim 12 is similar in scope to claims 2 and 3. EX1003, ¶312. Those claims recite *send[ing] header information to the control apparatus when the identified packet comprises an encapsulation packet* (claim 2) and that *the header information includes the additional header* (claim 3).

As discussed for claim 2, a network designer implementing Lean's conventional systems or inventive embodiments would modify Rijhsinghani's

switches to copy and forward entire packets to the external NMS for traffic analysis. See Section VII.C.2, above. Because entire packets are copied and forwarded, the combination discloses *notifying the controller of first header data before the translation and second header data after the translation* as shown below. EX1003, ¶313.

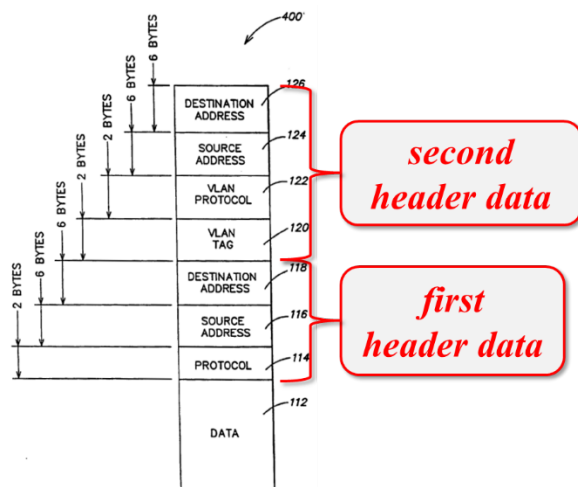


FIG. 4

Rijhsinghani, Fig. 4 (annotated)

For these reasons, and those discussed for claims 2, 3, and 5, the combination of Rijhsinghani and Lean discloses claim 12 (based on the construction of *the controller* assumed in this section). *Id.*, ¶314; see Sections VII.C.2, VII.C.3, and VII.A.2, above.

9. Claim 14

As discussed above, claim 14 is essentially the same in scope as claim 2. See Section VII.B.9, above. It is disclosed for the reasons discussed for claim 2.

See Section VII.C.2, above.

10. Claim 15

Claim 15's additional limitation is identical to claim 3's. EX1003, ¶62 (comparing claims 3 and 15). It is disclosed as discussed for that claim. See Section VII.C.3, above.

11. Claim 16

Claim 16's additional limitation is virtually identical to claim 4's. EX1003, ¶64 (comparing claims 4 and 16). It is disclosed as discussed for that claim. See Section VII.C.4, above.

12. Claim 17

Claim 17's additional limitation is virtually identical to claim 5's. EX1003, ¶66 (comparing claims 5 and 17). It is disclosed as discussed for that claim. See Section VII.A.2, above.

13. Claim 18

Claim 18's additional limitation is virtually identical to claim 6's. EX1003, ¶68 (comparing claims 6 and 18). It is disclosed as discussed for that claim. See Section VII.A.3, above.

14. Claim 20

Claim 20's additional limitations are virtually identical to the limitations in claims 14 and 15. EX1003, ¶69 (comparing claims 14 and 15 to claim 20). It is disclosed as discussed for those claims. See Sections VII.C.9 and VII.C.10, above.

D. Grounds 4-6: Adding Computer Networking Renders All Challenged Claims Obvious

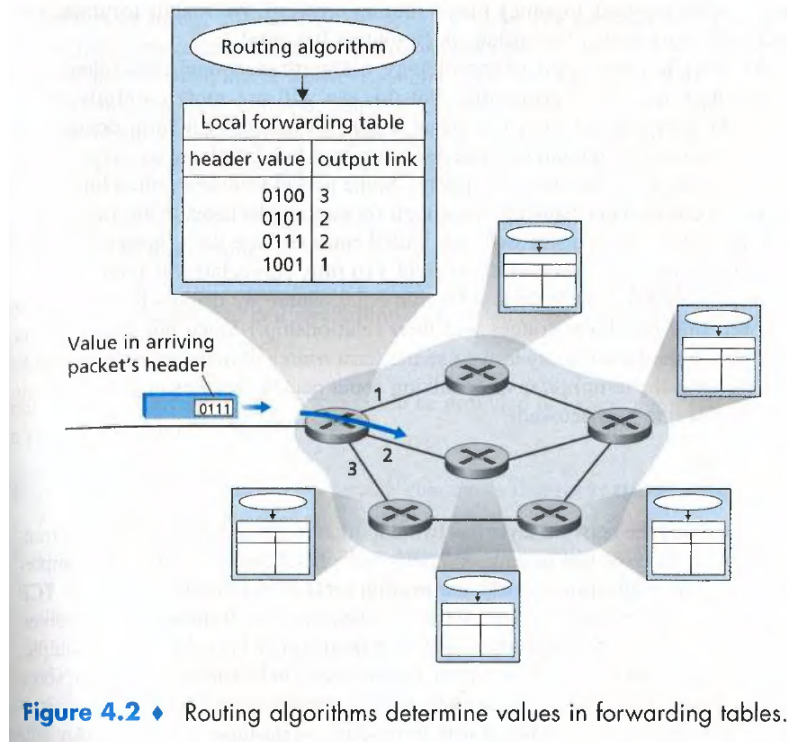
To the extent Patent Owner may argue that Rijhsinghani does not disclose *a table, the table including rules and actions corresponding to the rules* as recited in limitation 1a and the corresponding limitation in claim 7, it would have been obvious to implement Rijhsinghani's switches with table structures including rules and corresponding actions for processing packets as taught by any number of prior art references including Computer Networking (EX1008).

The use of a table to hold rules corresponding to packet-forwarding actions would have been well-known to a POSITA. EX1003, ¶328. As Computer Networking states:

Every router has a **forwarding table**. A router forwards a packet by examining the value of a field in the arriving packet's header, and then using this value to index into the router's forwarding table.

EX1008, 95 (emphasis in original); *see also id.*, 110 ("Given the existence of a forwarding table, table lookup is conceptually simple—we just search through the forwarding table looking for the longest prefix match....").

Computer Networking illustrates an exemplary forwarding table.
EX1008, 96.



Computer Networking, Fig. 4.2

In the illustrated example, the local forwarding table (*a table*) includes a column with header values (*rules*) used to lookup values in a column identifying the output link for forwarding the particular packet (*actions corresponding to the rules*).

EX1003, ¶329.

A POSITA seeking to understand and implement Rijhsinghani’s VLAN system would have naturally looked to a textbook like Computer Networking, which describes and summarizes then-existing network technologies and techniques. EX1003, ¶330; EX1008, 8 (“The book is therefore appropriate for undergraduate courses and for first-year graduate courses. It should also be useful to practitioners in the telecommunications industry.”). Rijhsinghani’s disclosure

presumes basic knowledge of networking technology and techniques. EX1003, ¶330. To the extent a POSITA might need to supplement their basic understanding of networking, Computer Networking would be a natural resource. *Id.*, ¶330.

Further, a POSITA could have implemented the table structures of Computer Networking in Rijhsinghani’s switches without undue experimentation and with high likelihood of success. EX1003, ¶331. Indeed, Computer Networking describes then-existing technology and techniques for networking, using commercially-available switches and other network equipment. *Id.* By implementing Rijhsinghani’s VLAN system using commercially-available equipment, the switches would include forwarding tables as described in Computer Networking and as recited in claims 1 and 7. *Id.*

VIII. DISCRETIONARY DENIAL IS NOT WARRANTED

Pursuant to the Interim Processes for PTAB Workload Management Memorandum (March 26, 2025), discretionary denial issues (if any) will be raised in a separate brief by Patent Owner. If Patent Owner files such a brief, Petitioners will respond in an opposition brief as provided in the Interim Processes for PTAB Workload Management Memorandum.

IX. CONCLUSION

Petitioner respectfully requests institution of review and cancellation of the challenged claims.

Respectfully submitted,

/James L. Day/

James L. Day
Registration No. 72,681
FARELLA BRAUN + MARTEL LLP
One Bush Street, Suite 900
San Francisco, California 94104
jday@fbm.com

CLAIM APPENDIX

[1pre] A switch apparatus, comprising:

[1a] a storage storing a table, the table including rules and actions corresponding to the rules; and

[1b] a controller comprising:

[1c] a memory storing instructions; and

[1d] a processor configured to execute the instructions to:

[1e] receive the rules and the actions from a control apparatus;

[1f] identify, based on the rules, a received packet;

[1g] duplicate a part of a header of the identified packet as an additional header when the identified packet comprises a target of encapsulation;

[1h] encapsulate the identified packet by the additional header; and

[1i] process, based on the actions, the identified packet.

2. The switch apparatus according to claim 1, wherein the processor is further configured to execute the instructions to

send header information to the control apparatus when the identified packet comprises an encapsulated packet.

3. The switch apparatus according to claim 2, wherein the header information includes the additional header.

4. The switch apparatus according to claim 3, wherein the processor is further configured to execute further instructions to decapsulate the encapsulated packet.

5. The switch apparatus according to claim 1, wherein the processor is further configured to execute the instructions for translation of first header data of a head of the received packet into second header data.

6. The switch apparatus according to claim 5, wherein the instructions to encapsulate the identified packet includes notifying the controller of the first header data before the translation and the second header data after the translation.

7. A communication system, comprising:

a control apparatus configured to control a plurality of switch apparatuses; and

a switch apparatus of the switch apparatuses comprising:

a storage storing a table, the table including rules and actions corresponding to the rules; and

a controller comprising:

a memory storing instructions; and

a processor configured to execute the instructions to:

receive the rules and the actions from a controller;

identify, based on the rules, a received packet;

duplicate a part of a header of the identified packet as an additional header when the identified packet comprises a target of encapsulation;

encapsulate the identified packet by the additional header; and

process, based on the actions, the identified packet.

8. The communication system according to claim 7, wherein the processor is further configured to execute the instructions to send header information to the control apparatus when the identified packet comprises an encapsulated packet.

9. The communication system according to claim 8, wherein the header information includes the additional header.

10. The communication system according to claim 9, wherein the processor is further configured to execute further instructions to decapsulate the encapsulated packet.

11. The communication system according to claim 7, wherein the processor is further configured to execute the instructions for translation of the first header data of a head of the received packet into the second header data.

12. The communication system according to claim 11, wherein the instructions to encapsulate the identified packet includes notifying the controller of first header data before the translation and second header data after the translation.

13. A communication method, comprising:

receiving rules and actions from a controller;

identifying, based on the rules, a received packet;

duplicating a part of a header of the identified packet as an additional header when the identified packet comprises a target of encapsulation;

encapsulating the identified packet by the additional header; and

processing, based on the actions, the identified packet.

14. The communication method according to claim 13, further comprising:

sending header information to a controller when the identified packet comprises an encapsulated packet.

15. The communication method according to claim 14, wherein the header information includes the additional header.

16. The communication method according to claim 15, further comprising:

executing further instructions to decapsulate the encapsulated packet.

17. The communication method according to claim 14, further comprising:

translating first header data of a head of the received packet into second header data.

18. The communication method according to claim 17, further comprising:

notifying the controller of the first header data before the translating and the second header data after the translating.

19. A non-transitory recording medium recording a program which when executed causes a switch to perform a method, the method comprising:

receiving rules and actions from a controller;

identifying, based on the rules, a received packet;

duplicating a part of a header of the identified packet as an additional header when the identified packet comprises a target of encapsulation;

encapsulating the identified packet by the additional header; and
processing, based on the actions, the identified packet.

20. The non-transitory recording medium, according to claim 19, the
method further comprising:

sending header information to the controller when the identified
packet comprises an encapsulated packet,

wherein the header information includes the additional header.

CERTIFICATE OF COMPLIANCE

This petition complies with the 14,000 word-count limit under 37 C.F.R. § 42.24(a)(1)(i). The petition contains 13,999 words, excluding the parts exempted by 37 C.F.R. § 42.24(a)(1) (i.e., table of contents, table of authorities, mandatory notices under § 42.8, certificate of service, word count certificate, and appendix of exhibits), as determined by the word count feature of Microsoft Word, which was used to prepare this petition.

/James L. Day/
James L. Day
Registration No. 72,681
Farella Braun + Martel LLP
235 Montgomery Street; 17th Floor
San Francisco, CA 94104

CERTIFICATE OF SERVICE

I hereby certify that on July 14, 2025, I caused a complete and entire copy of this Petition for *Inter Partes* Review and all supporting exhibits to be served by express mail, or means at least as fast and reliable as express mail, on the following counsel of record identified through the USPTO Patent Center.

21254-McGinn Intellectual Property Law Group, PLLC
c/o Sean McGinn, Fredrick Cooperrider, Saryadvinder Sahota, Geoffrey Pechie, or
Ian McGinn
8321 Old Courthouse Road, Suite 200
Vienna, VA 22182-3817

In addition, courtesy copies were sent by electronic means to Patent Owner's litigation counsel at the following addresses:

yurykapgan@quinnemanuel.com
brianmack@quinnemanuel.com
claire@millerfairhenry.com
jonathantse@quinnemanuel.com
melissaneri@quinnemanuel.com
ryangoldstein@quinnemanuel.com
andrea@millerfairhenry.com

/James L. Day/
James L. Day
Registration No. 72,681