

Petition for *Inter Partes* Review of U.S. Patent No. 10,652,111

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

ARISTA NETWORKS, INC.

Petitioner,

v.

ORCKIT CORPORATION,

Patent Owner.

IPR2024-01239

Attorney Docket No.: 083475.0110

**PETITION FOR *INTER PARTES* REVIEW OF CLAIMS 1-54 OF U.S.
PATENT NO. 10,652,111 B2**

TABLE OF CONTENTS

	Page
PETITIONER’S EXHIBIT LIST.....	VI
I. INTRODUCTION	1
II. MANDATORY NOTICES UNDER 37 C.F.R. 42.8(A)(1)	1
A. Real Party-in-Interest under 37 C.F.R. 42.8(b)(1)	1
B. Related Matters under 37 C.F.R. 42.8(b)(2)	1
C. Lead and Back-Up Counsel under 37 C.F.R. 42.8(b)(3)	2
D. Service Information under 37 C.F.R. 42.8(b)(4).....	3
III. CLAIM LISTING.....	4
IV. REQUIREMENTS UNDER 37 C.F.R. 42.104.....	13
A. Grounds for Standing	13
B. Identification of Challenge and Statement of Precise Relief Requested	13
V. THE ’111 PATENT	14
A. Prosecution History	16
VI. LEVEL OF ORDINARY SKILL IN THE ART	16
VII. CLAIM CONSTRUCTION.....	17
A. The “Controller” Term	17
B. The “Criterion” Terms.....	17
VIII. SUMMARY OF PRIOR ART.....	18
A. U.S. Patent No. 10,097,452 (“Lefebvre”) (EX1005)	19
B. U.S. Patent No. 9,264,301 (“Chua”) (EX1006)	20
C. U.S. Patent No. 9,813,447 (“Rash”) (EX1007)	21

D.	RFC 793 (EX1015)	22
IX.	THE ASSERTED GROUNDS OF INVALIDITY	22
A.	Ground 1: Claims 1-6, 9-15, 18-19, 21-36, 41-42, and 44-54 are rendered obvious by Lefebvre in view of Chua.	22
1.	A POSITA Would Have and Could Have Combined Lefebvre with Chua	23
2.	Independent Claims 1 and 32.....	28
	a) Claim 1	28
	b) Claim 32	39
3.	Dependent Claims 2-6, 9-15, 18-19, 21-31, 33-36, 41-42, and 44-54.....	40
	a) Claims 2 and 33.....	40
	b) Claims 3 and 34.....	42
	c) Claims 4 and 35.....	43
	d) Claims 5 and 36.....	46
	e) Claim 6	47
	f) Claim 9	48
	g) Claim 10	48
	h) Claim 11	49
	i) Claim 12	51
	j) Claim 13	52
	k) Claim 14	53
	l) Claim 15	53
	m) Claims 18 and 41.....	55
	n) Claims 19 and 42.....	55

Petition for *Inter Partes* Review of U.S. Patent No. 10,652,111

o) Claims 21 and 44.....	56
p) Claims 22 and 45.....	57
q) Claims 23 and 46.....	58
r) Claims 24 and 47.....	59
s) Claims 25 and 48.....	59
t) Claims 26 and 49.....	60
u) Claims 27 and 50.....	60
v) Claims 28 and 51.....	61
w) Claims 29 and 52.....	62
x) Claims 30 and 53.....	62
y) Claims 31 and 54.....	63
B. Ground 2: Claims 7, 8, 37, and 38 are rendered obvious by Lefebvre in view of Chua and Rash.	63
1. A POSITA Would Have and Could Have Combined Lefebvre and Chua with Rash	63
2. Dependent Claims 7, 8, 37, and 38	67
a) Claims 7 and 37.....	67
b) Claims 8 and 38.....	68
C. Ground 3: Claims 16-20 and 39-43 are rendered obvious by Lefebvre in view of Chua and RFC 793.	69
1. A POSITA Would Have and Could Have Combined Lefebvre and Chua with RFC 793.....	69
2. Dependent Claims 16-20 and 39-43	70
a) Claims 16 and 39.....	70
b) Claims 17 and 40.....	73

Petition for *Inter Partes* Review of U.S. Patent No. 10,652,111

c) Claims 18 and 41.....	73
d) Claims 19 and 42.....	74
e) Claims 20 and 43.....	74
X. SECONDARY CONSIDERATIONS	77
XI. INSTITUTION SHOULD BE GRANTED	78
A. 35 U.S.C. §314(a).....	78
B. 35 U.S.C. §325(d).....	82
XII. CONCLUSION	82

PETITIONER’S EXHIBIT LIST

Ex[No.]	Description of Documents
1001	U.S. Patent No. 10,652,111 (“’111 Patent”)
1002	Declaration of Paul Min, Ph.D.
1003	Prosecution File History of the ’111 Patent
1004	Reserved
1005	U.S. Patent No. 10,097,452 (“Lefebvre”)
1006	U.S. Patent No. 9,264,301 (“Chua”)
1007	U.S. Patent No. 9,813,447 (“Rash”)
1008	M. Liao, et al., “Design and Evaluation of Deep Packet Inspection System: A Case Study,” IET Networks, pp. 1-8 (2012)
1009	A. W. Moore and K. Papagiannaki, “Toward the Accurate Identification of Network Applications,” In: PAM, Springer, vol. 5, pp. 41-54 (2005)
1010	RFC 3654 (available at https://www.rfc-editor.org/rfc/rfc3654)
1011	W. Xia, et al., “A Survey on Software-Defined Networking,” IEEE Communication Surveys & Tutorials, Vol. 17, No. 1, pp. 27-51 (June 13, 2014)
1012	RFC 3746 (available at https://www.rfc-editor.org/in-notes/rfc3746.txt)
1013	N. McKeown, et al., “OpenFlow: Enabling Innovation in Campus Networks,” ACM SIGCOMM Computer Communication Review, vol. 38, No. 2, pp. 69-74 (April 2008)
1014	Steven Levy, “Going with the Flow: Google’s Secret Switch to the Next Wave of Networking” (April 17, 2012) (available at https://www.wired.com/2012/04/going-with-the-flow-google/)

Petition for *Inter Partes* Review of U.S. Patent No. 10,652,111

Ex[No.]	Description of Documents
1015	RFC 793 (available at https://www.rfc-editor.org/rfc/pdf/rfc793.txt.pdf)
1016	“About the Open Networking Foundation”
1017	“Open Networking Foundation Specifications,” (available at https://opennetworking.org/software-defined-standards/specifications/)
1018	“OpenFlow Switch Specification”
1019	RFC 5810 (available at https://www.rfc-editor.org/rfc/rfc5810.html)
1020	Declaration of Sylvia D. Hall-Ellis, Ph.D.
1021	Federal Court Statistics
1022	Complaint, <i>Orckit Corporation v. Arista Networks, Inc.</i> , Civil Action No. 1-23-cv-00821 (D. Del., Jul. 28, 2023)
1023	Stipulation and Order to Stay Action, <i>Orckit Corporation v. Arista Networks, Inc.</i> , Civil Action No. 1-23-cv-00821 (D. Del., Dec. 4, 2023)
1024	Katherine K. Vidal, Interim Procedure for Discretionary Denials in AIA Post-Grant Proceedings with Parallel District Court Litigation (June 21, 2022)
1025	RFC 2026
1026	U.S. Patent Application Publication No. 2007/0094411 by Mullane (April 26, 2007)
1027	OpenFlow Switch Specification, Version 1.1.0 (available at https://opennetworking.org/wp-content/uploads/2014/10/openflow-spec-v1.1.0.pdf)

I. INTRODUCTION

Arista Networks, Inc. (“Petitioner” or “Arista”) submits this Petition for *inter partes* review (“IPR”) of Claims 1-54 (the “Challenged Claims”) of U.S. Patent No. 10,652,111 (the “’111 Patent”). The ’111 Patent relates to deep packet inspection (DPI) in a software-defined network (SDN). SDNs and DPI were known well before the 2014 filing date of the ’111 Patent. EX1002, ¶¶62-72. Because nothing claimed was novel or nonobvious, Petitioner respectfully requests institution of IPR and a finding that the Challenged Claims are unpatentable.

II. MANDATORY NOTICES UNDER 37 C.F.R. 42.8(a)(1)

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

The real party-in-interest is Arista Networks, Inc. No unnamed entity is funding, controlling, or directing this Petition, or otherwise has had an opportunity to control or direct this Petition.

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

As of the filing date of this Petition, and to the best knowledge of Petitioner, the ’111 Patent is involved in the following:

Orckit Corporation v. Arista Networks, Inc., Case No. 1-23-cv-00821 (D. Del.) (“Delaware Litigation”). The ’111 Patent was first asserted against Petitioner in a Complaint for Patent Infringement filed on July 28, 2023. EX1022. Petitioner was served on August 2, 2023. The Delaware Litigation was stayed on December 4, 2023, pending resolution of IPR2023-00401, IPR2023-00402, and IPR2023-

Petition for Inter Partes Review of U.S. Patent No. 10,652,111

00554. EX1023.

As of the filing date of this Petition, and to the best knowledge of Petitioner, the '111 Patent is or has been involved in the following proceedings:

Orckit Corporation v. Juniper Networks, Inc., Case No. 1-23-cv-00822 (D. Del.);

Orckit Corporation v. Cisco Systems, Inc., Case No. 2-22-cv-00276 (E.D. Tex.);

Cisco Systems, Inc. v. Orckit Corporation, IPR2023-00554 (P.T.A.B.);

Juniper Networks, Inc. v. Orckit Corporation, IPR2024-00037 (P.T.A.B.);

Juniper Networks, Inc. v. Orckit Corporation, IPR2024-00895 (P.T.A.B.);

and

Reexamination Control No. 90/015,261.

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

Petitioner provides the following designation of counsel:

LEAD COUNSEL	BACK-UP COUNSEL
Eliot D. Williams (Reg. No. 50,822) BAKER BOTTS L.L.P. 700 K Street, N.W. Washington, D.C. 20001-5692 Tel: 202-639-1334 eliot.williams@bakerbotts.com	Clarke W. Stavinoha (Reg. No. 71,152) BAKER BOTTS L.L.P. 2001 Ross Avenue, Suite 900 Dallas, TX 75201-2980 Tel: 214-953-6484 clarke.stavinoha@bakerbotts.com Jeremy Taylor (Reg. No. 73,912) BAKER BOTTS L.L.P. 101 California Street, Suite 3200

Petition for *Inter Partes* Review of U.S. Patent No. 10,652,111

	San Francisco, CA 94111 Tel: 415-291-6200 jeremy.taylor@bakerbotts.com Brianna Potter (Reg. No. 76,748) BAKER BOTTS L.L.P. 1001 Page Mill Road Building One, Suite 200 Palo Alto, CA 94304-1007 Tel: 650-739-7556 brianna.potter@bakerbotts.com
--	--

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

A copy of this entire Petition, including all Exhibits and a power of attorney, is being served by PRIORITY MAIL EXPRESS, costs prepaid, to the address of the attorney or agent of record for the '111 Patent at the USPTO: May Patents Ltd. c/o Dorit Shem-Tov, P.O.B 7230, Ramat-Gan, 5217102, ISRAEL; and to the address of the attorney or agent of record for Patent Owner in the Delaware Litigation: Jacob R. Kirkham, KOBRE & KIM LLP, 600 North King Street, Suite 501, Wilmington, DE 19801.

Please address all correspondence to lead and back-up counsel. Petitioner consents to service at lead counsel's address provided above. Petitioner consents to electronic service, provided it is made to all of the following e-mail addresses:

- eliot.williams@bakerbotts.com;
- clarke.stavinoha@bakerbotts.com;
- jeremy.taylor@bakerbotts.com; and

- brianna.potter@bakerbotts.com.

A Power of Attorney is filed concurrently herewith under 37 C.F.R. §42.10(b).

III. CLAIM LISTING

Claim 1	
1[Pre]	A method for use with a packet network including a network node for transporting packets between first and second entities under control of a controller that is external to the network node, the method comprising:
1[a]	sending, by the controller to the network node over the packet network, an instruction and a packet-applicable criterion;
1[b]	receiving, by the network node from the controller, the instruction and the criterion;
1[c]	receiving, by the network node from the first entity over the packet network, a packet addressed to the second entity;
1[d]	checking, by the network node, if the packet satisfies the criterion;
1[e]	responsive to the packet not satisfying the criterion, sending, by the network node over the packet network, the packet to the second entity; and
1[f]	responsive to the packet satisfying the criterion, sending the packet, by the network node over the packet network, to an entity that is included in the instruction and is other than the second entity.
Claim 2	
2	The method according to claim 1, wherein the instruction is ‘probe’, ‘mirror’, or ‘terminate’ instruction, and upon receiving by the network node the ‘terminate’ instruction, the method further comprising blocking, by the network node, the packet from being sent to the second entity and to the controller.
Claim 3	
3	The method according to claim 1, wherein the instruction is a ‘probe’, a ‘mirror’, or a ‘terminate’ instruction, and upon receiving by the network

	node the 'mirror' instruction and responsive to the packet satisfying the criterion, the method further comprising sending the packet, by the network node, to the second entity and to the controller.
Claim 4	
4	The method according to claim 1, wherein the instruction is 'probe', 'mirror', or 'terminate' instruction, and upon receiving by the network node the 'probe' instruction and responsive to the packet satisfying the criterion, the method further comprising: sending the packet, by the network node, to the controller; responsive to receiving the packet, analyzing the packet, by the controller; sending the packet, by the controller, to the network node; and responsive to receiving the packet, sending the packet, by the network node, to the second entity.
Claim 5	
5	The method according to claim 1, further comprising responsive to the packet satisfying the criterion and to the instruction, sending the packet or a portion thereof, by the network node, to the controller.
Claim 6	
6	The method according to claim 5, further comprising storing the received packet or a portion thereof, by the controller, in a memory.
Claim 7	
7	The method according to claim 5, further comprising responsive to the packet satisfying the criterion and to the instruction, sending a portion of the packet, by the network node, to the controller.
Claim 8	
8	The method according to claim 7, wherein the portion of the packet consists of multiple consecutive bytes, and wherein the instruction comprises identification of the consecutive bytes in the packet.
Claim 9	
9	The method according to claim 5, further comprising responsive to receiving the packet, analyzing the packet, by the controller.

Claim 10	
10	The method according to claim 9, further for use with an application server that communicates with the controller, wherein the analyzing comprising sending the packet, by the controller, to the application server, and analyzing the packet by the application server.
Claim 11	
11	The method according to claim 10, wherein the analyzing further comprising sending the packet after analyzing by the application server to the controller, and sending the packet, after receiving from the controller by the network node, to the second entity.
Claim 12	
12	The method according to claim 9, wherein the analyzing comprises applying security or data analytic application.
Claim 13	
13	The method according to claim 9, wherein the analyzing comprises applying security application that comprises firewall or intrusion detection functionality.
Claim 14	
14	The method according to claim 9, wherein the analyzing comprises performing Deep Packet Inspection (DPI) or using a DPI engine on the packet.
Claim 15	
15	The method according to claim 9, wherein the packet comprises distinct header and payload fields, and wherein the analyzing comprises checking part of, or whole of, the payload field.
Claim 16	
16	The method according to claim 1, wherein the packet comprises distinct header and payload fields, the header comprises one or more flag bits, and wherein the packet-applicable criterion is that one or more of the flag bits is set.

Claim 17	
17	The method according to claim 16, wherein the packet is an Transmission Control Protocol (TCP) packet, and wherein the one or more flag bits comprises comprise a SYN flag bit, an ACK flag bit, a FIN flag bit, a RST flag bit, or any combination thereof.
Claim 18	
18	The method according to claim 1, wherein the packet comprises distinct header and payload fields, the header comprises at least the first and second entities addresses in the packet network, and wherein the packet-applicable criterion is that the first entity address, the second entity address, or both match a predetermined address or addresses.
Claim 19	
19	The method according to claim 18, wherein the addresses are Internet Protocol (IP) addresses.
Claim 20	
20	The method according to claim 1, wherein the packet is an Transmission Control Protocol (TCP) packet that comprises source and destination TCP ports, a TCP sequence number, and a TCP sequence mask fields, and wherein the packet-applicable criterion is that the source TCP port, the destination TCP port, the TCP sequence number, the TCP sequence mask, or any combination thereof, matches a predetermined value or values.
Claim 21	
21	The method according to claim 1, wherein the packet network comprises a Wide Area Network (WAN), Local Area Network (LAN), the Internet, Metropolitan Area Network (MAN), Internet Service Provider (ISP) backbone, datacenter network, or inter-datacenter network.
Claim 22	
22	The method according to claim 1, wherein the first entity is a server device and the second entity is a client device, or wherein the first entity is a client device and the second entity is a server device.

Claim 23	
23	The method according to claim 22, wherein the server device comprises a web server, and wherein the client device comprises a smartphone, a tablet computer, a personal computer, a laptop computer, or a wearable computing device.
Claim 24	
24	The method according to claim 22, wherein the communication between the network node and the controller is based on, or uses, a standard protocol.
Claim 25	
25	The method according to claim 24, wherein the standard protocol is according to, based on, or compatible with, an OpenFlow protocol version 1.3.3 or 1.4.0.
Claim 26	
26	The method according to claim 25, wherein the instruction comprises a Type-Length-Value (TLV) structure.
Claim 27	
27	The method according to claim 1, wherein the network node comprises a router, a switch, or a bridge.
Claim 28	
28	The method according to claim 1, wherein the packet network is an Internet Protocol (IP) network, and the packet is an IP packet.
Claim 29	
29	The method according to claim 28, wherein the packet network is an Transmission Control Protocol (TCP) network, and the packet is an TCP packet.
Claim 30	
30	The method according to claim 1, further comprising: receiving, by the network node from the first entity over the packet network, one or more

	additional packets; checking, by the network node, if any one of the one or more additional packets satisfies the criterion; responsive to an additional packet not satisfying the criterion, sending, by the network node over the packet network, the additional packet to the second entity; and responsive to the additional packet satisfying the criterion, sending the additional packet, by the network node over the packet network, in response to the instruction.
Claim 31	
31	The method according to claim 1, wherein the packet network is a Software Defined Network (SDN), the packet is routed as part of a data plane and the network node communication with the controller serves as a control plane.
Claim 32	
32[pre]	A method for use with a packet network including a network node for transporting packets between first and second entities under control of a controller that is external to the network node, the method by the network node comprising:
32[a]	receiving, from the controller, the instruction and the criterion;
32[b]	receiving, from the first entity over the packet network, a packet addressed to the second entity;
32[c]	checking if the packet satisfies the criterion;
32[d]	responsive to the packet not satisfying the criterion, sending over the packet network, the packet to the second entity; and
32[e]	responsive to the packet satisfying the criterion, sending the packet over the packet network, to an entity that is included in the instruction and is other than the second entity.
Claim 33	
33	The method according to claim 32, wherein the instruction is ‘probe’, ‘mirror’, or ‘terminate’ instruction, and upon receiving the ‘terminate’ instruction, the method further comprising blocking, the packet from being sent to the second entity and to the controller.

Claim 34	
34	The method according to claim 32, wherein the instruction is a ‘probe’, a ‘mirror’, or a ‘terminate’ instruction, and upon receiving the ‘mirror’ instruction and responsive to the packet satisfying the criterion, the method further comprising sending the packet to the second entity and to the controller.
Claim 35	
35	The method according to claim 32, wherein the instruction is ‘probe’, ‘mirror’, or ‘terminate’ instruction, and upon receiving the ‘probe’ instruction and responsive to the packet satisfying the criterion, the method further comprising: sending the packet to the controller; receiving the packet, from the controller; and responsive to receiving the packet, sending the packet, to the second entity.
Claim 36	
36	The method according to claim 32, further comprising responsive to the packet satisfying the criterion and to the instruction, sending the packet or a portion thereof to the controller.
Claim 37	
37	The method according to claim 36, further comprising responsive to the packet satisfying the criterion and to the instruction, sending a portion of the packet to the controller.
Claim 38	
38	The method according to claim 37, wherein the portion of the packet consists of multiple consecutive bytes, and wherein the instruction comprises identification of the consecutive bytes in the packet.
Claim 39	
39	The method according to claim 32, wherein the packet comprises distinct header and payload fields, the header comprises one or more flag bits, and wherein the packet-applicable criterion is that one or more of the flag bits is set.

Claim 40	
40	The method according to claim 39, wherein the packet is an Transmission Control Protocol (TCP) packet, and wherein the one or more flag bits comprises comprise a SYN flag bit, an ACK flag bit, a FIN flag bit, a RST flag bit, or any combination thereof.
Claim 41	
41	The method according to claim 32, wherein the packet comprises distinct header and payload fields, the header comprises at least the first and second entities addresses in the packet network, and wherein the packet-applicable criterion is that the first entity address, the second entity address, or both match a predetermined address or addresses.
Claim 42	
42	The method according to claim 41, wherein the addresses are Internet Protocol (IP) addresses.
Claim 43	
43	The method according to claim 32, wherein the packet is an Transmission Control Protocol (TCP) packet that comprises source and destination TCP ports, a TCP sequence number, and a TCP sequence mask fields, and wherein the packet-applicable criterion is that the source TCP port, the destination TCP port, the TCP sequence number, the TCP sequence mask, or any combination thereof, matches a predetermined value or values.
Claim 44	
44	The method according to claim 32, wherein the packet network comprises a Wide Area Network (WAN), Local Area Network (LAN), the Internet, Metropolitan Area Network (MAN), Internet Service Provider (ISP) backbone, datacenter network, or inter-datacenter network.
Claim 45	
45	The method according to claim 32, wherein the first entity is a server device and the second entity is a client device, or wherein the first entity is a client device and the second entity is a server device.

Claim 46	
46	The method according to claim 45, wherein the server device comprises a web server, and wherein the client device comprises a smartphone, a tablet computer, a personal computer, a laptop computer, or a wearable computing device.
Claim 47	
47	The method according to claim 45, wherein the communication with the controller is based on, or uses, a standard protocol.
Claim 48	
48	The method according to claim 47, wherein the standard protocol is according to, based on, or compatible with, an OpenFlow protocol version 1.3.3 or 1.4.0.
Claim 49	
49	The method according to claim 48, wherein the instruction comprises a Type-Length-Value (TLV) structure.
Claim 50	
50	The method according to claim 32, wherein the network node comprises a router, a switch, or a bridge.
Claim 51	
51	The method according to claim 32, wherein the packet network is an Internet Protocol (IP) network, and the packet is an IP packet.
Claim 52	
52	The method according to claim 51, wherein the packet network is a Transmission Control Protocol (TCP) network, and the packet is an TCP packet.
Claim 53	
53	The method according to claim 32, further comprising: receiving, from the first entity over the packet network, one or more additional packets; checking, if any one of the one or more additional packets satisfies the

	criterion; responsive to an additional packet not satisfying the criterion, sending over the packet network, the additional packet to the second entity; and responsive to the additional packet satisfying the criterion, sending the additional packet over the packet network, in response to the instruction.
Claim 54	
54	The method according to claim 32, wherein the packet network is a Software Defined Network (SDN), the packet is routed as part of a data plane and the communication with the controller serves as a control plane.

IV. REQUIREMENTS UNDER 37 C.F.R. 42.104

A. Grounds for Standing

Petitioner certifies that the '111 Patent is available for IPR and that Petitioner is not barred or otherwise estopped.

B. Identification of Challenge and Statement of Precise Relief Requested

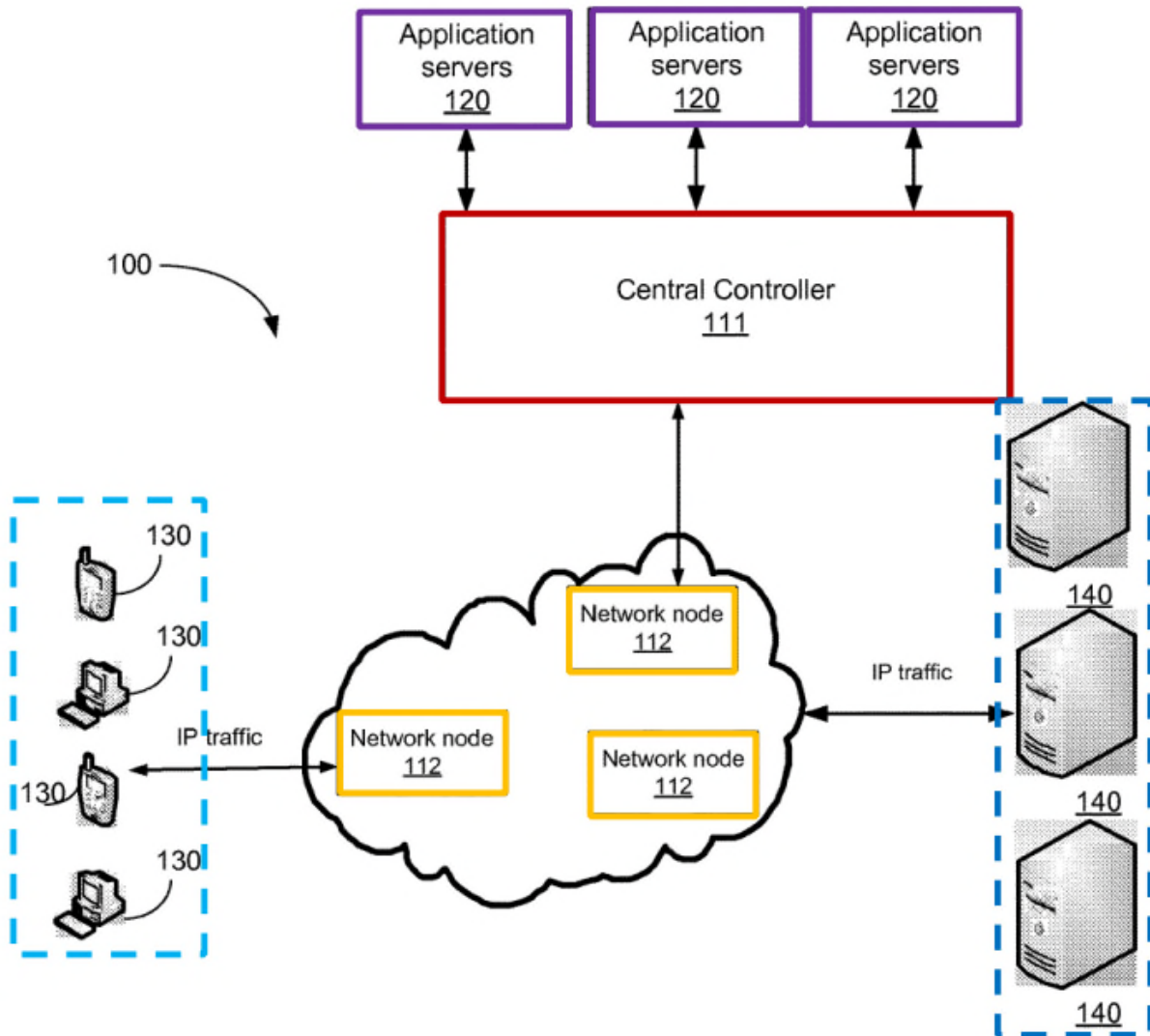
Petitioner asks that the Board institute a trial for *inter partes* review and cancel the Challenged Claims in view of the analysis below for the following grounds:

Ground	'111 Patent Claims	Basis for Challenge
1	1-6, 9-15, 18-19, 21-36, 41-42, 44-54	Obvious under §103 based on U.S. Patent No. 10,097,452 ("Lefebvre") in view of U.S. Patent No. 9,264,301 ("Chua")
2	7-8, 37-38	Obvious under §103 based on Lefebvre and Chua in view of U.S. Patent No. 9,813,447 ("Rash")
3	16-20, 39-43	Obvious under §103 based on Lefebvre and Chua in view of RFC 793

This Petition is supported by the declaration of Paul Min, Ph.D., a qualified technical expert. See EX1002 ("Min"), ¶¶5-22, 53-54.

V. The '111 Patent

The '111 Patent describes a method for DPI in an SDN “performed by a central controller of the SDN.” *See, e.g.*, EX1001, 2:27-30, 2:50-51, 4:5-7 (“[T]he central controller 111 is configured to perform deep packet inspection on designated packets from designated flows or TCP sessions.”); *Min*, ¶¶40-46. The controller 111 instructs network nodes 112 to determine if incoming packets from client devices 130 communicating with destination servers 140 require inspection. EX1001, 4:8-11, FIG. 6 (step S610). The controller 111 and the network nodes 112 communicate using standard protocols such as OpenFlow. EX1001, 3:44-48. The network nodes 112 redirect or mirror packets for inspection by the controller 111. EX1001, 4:15-18. The controller 111 then provides inspected data to application servers 120 for further processing. EX1001, 3:56-61.



EX1001, FIG. 1 (annotated); EX1002, ¶42.

Controller 111 includes memory 313 and processing unit 314. EX1001, 5:33-36. The '111 Patent discloses the controller 111 provides instructions to the network nodes 112. EX1001, 6:10-12, 9:40-46, FIG. 6 (step S610). When the instructions apply to a packet, the network nodes 112 redirect the packet to the controller 111. EX1001, 5:50-59, 6:12-19, 6:27-32, 6:41-52, 9:47-61, FIG. 6. And after inspection, the controller 111 provides the data to the application servers

120. EX1001, 5:36-39; Min, ¶¶43-44.

A. Prosecution History

The '111 Patent was filed as Application 15/126,288 on September 15, 2016 and claims priority to U.S. Provisional Application 61/982,358 filed April 22, 2014. EX1001, Cover. Petitioner assumes without conceding that the '111 Patent has the priority date of April 22, 2014.

During prosecution, the Examiner rejected all claims. EX1003, 271-296. Applicant eventually amended the independent claims to recite that the controller was “external to the network node” and that the packets satisfying the criterion were sent “to an entity that is included in the instruction and is other than the second entity.” EX1003, 397-417; Min, ¶¶47-49. Again, the Examiner rejected all claims as being unpatentable, relying on a new secondary reference. EX1003, 425-451. After the applicant’s response, a Notice of Allowance issued. EX1003, 492-510.

VI. LEVEL OF ORDINARY SKILL IN THE ART

A person of ordinary skill in the art (“POSITA”) as of April 22, 2014, would have had a bachelor’s degree in computer science, computer engineering, or an equivalent field, and two years of professional experience, and a POSITA would have had a working knowledge of hardware and software for packet-switched networking. Min, ¶¶51-54. A person with less or different education but more relevant practical experience, or vice versa, may also meet this standard. Min, ¶52.

VII. CLAIM CONSTRUCTION

Petitioner interprets the claims “in accordance with the ordinary and customary meaning...as understood by one of ordinary skill in the art.” 37 C.F.R. §42.100(b). In the sections that follow, Petitioner provides references from the specification of the ’111 Patent for certain terms. Min, ¶¶56-61.

A. The “Controller” Term

The construction of the term “controller” in claims 1 and 32 has been at issue in other proceedings involving the ’111 Patent. In IPR2024-00895, the petitioner proposed “controller” should be construed to mean “an entity configured to perform deep packet inspection on packets.” *Juniper Networks, Inc. v. Orckit Corp.*, IPR2024-00895, Paper 2, 9-10 (P.T.A.B. May 6, 2024); Min, ¶58.

Additionally, in a district court case involving the ’111 Patent, but that did not involve Petitioner, “controller” was preliminarily construed to mean “an entity that is capable of controlling deep packet inspection.” *Cisco Systems, Inc. v. Orckit Corp.*, IPR2023-00554, Paper 22, 16-17 (P.T.A.B. Jan. 15, 2024); Min, ¶59.

Petitioner submits that “controller” does not require express construction because the prior art relied upon herein teaches the “controller” under any reasonable construction, including those discussed above. Min, ¶60.

B. The “Criterion” Terms

Claim 1 recites “a packet-applicable criterion” and its dependent claims refer to “the criterion” or “the packet-applicable criterion.” Similarly, Claim 32 recites

“the criterion” and its dependent claims refer to “the criterion” or “the packet-applicable criterion.” These terms do not appear in the ’111 Patent’s specification. But the patent discloses how “each network node 112 is configured to determine if an incoming packet requires inspection or not,” and that the “determination is performed based on a set of instructions provided by the controller 111.” EX1001, 4:12-15. As a result of this set of instructions, the packet “is either redirected to the controller 111 or mirrored and a copy thereof is sent to the controller 111” without alteration. EX1001, 4:15-19, 4:61-5:16 (describing type-length-value structures for parsing and identifying packet fields). Further, the ’111 Patent includes a series of examples of probe instructions that consider various TCP flags. EX1001, 4:26-41. The ’111 Patent then states that “each node 112 is configured to receive an incoming packet (either a request from a client device 130 or response from a server 140), analyze the packet’s header, and perform the action (redirect the packet to controller 111 or send to destination server 140) respective of the value of the TCP flag.” EX1001, 4:39-46. Given this description, a POSITA would have understood the “packet-applicable criterion” and “criterion” to include “the conditions, or basis, used to identify packets for actions, such as inspection.” Min, ¶61.

VIII. SUMMARY OF PRIOR ART

Petitioner assumes without conceding that the Challenged Claims are entitled to an April 22, 2014, priority date, and thus are subject to the AIA provisions of 35

U.S.C. §102 and §103.

A. U.S. Patent No. 10,097,452 (“Lefebvre”) (EX1005)

Lefebvre was filed on July 24, 2012 and published on October 17, 2013. EX1005, Cover. Lefebvre is prior art under 35 U.S.C. § 102(a)(1)-(2).

Lefebvre discloses that network operators deploy inline services to “inspect and alter network traffic transiting through their network.” EX1005, 1:19-27; Min, ¶¶78-84. These services typically existed as dedicated appliances with their own costs. EX1005, 1:25-27. As network traffic demands increase, operators seek to selectively direct traffic to these services and minimize costs. EX1005, 1:28-43; Min, ¶80.

Lefebvre discloses that traffic steering is “the action of classifying traffic and directing the different classes of traffic through specific service chains.” EX1005, 1:44-47. Service chains are ordered sets of services. EX1005, 1:44. Lefebvre discloses operators defining service policies specifying traffic classes and the service chains each traffic class traverses. EX1005, 4:41-47. A centralized controller translates these policies into rules for the programmable switches that steer the network traffic through the service chains. EX1005, 4:43-47. Services can exist in multiple service chains and operators can locate and chain these service instances arbitrarily. EX1005, 4:48-53. Lefebvre discloses implementing an SDN using OpenFlow. EX1005, 4:30-40; Min, ¶¶81-82, 84.

Lefebvre implements traffic steering using a two-step process. Min, ¶83.

First, Lefebvre classifies incoming packets and assigns a service path based on predefined policies. EX1005, 6:17-28, Abstract. Then the packets are forwarded to the services in the assigned service path. EX1005, 6:20-21.

B. U.S. Patent No. 9,264,301 (“Chua”) (EX1006)

Chua was filed on March 15, 2013. EX1006, Cover. Chua is prior art under 35 U.S.C. § 102(a)(2).

Chua discloses that OpenFlow permits SDNs to use a standalone controller to “determine the routes and program other devices in the network[.]” EX1006, 1:25-35; *see also* EX1006, 1:45-55 (describing the control plane being “physically separate from the data plane (or forwarding plane) for an SDN”), 1:35-41, 23:62-24:4 (describing OpenFlow as “an example of an SDN protocol”); Min, ¶¶85-87.

Chua’s SDN controller controls the network devices and service devices. EX1006, 6:25-28; Min, ¶88. Using data from the service devices, Chua’s SDN controller makes enforcement decisions including allowing, blocking, mirroring, redirecting, transforming, and encapsulating the packets of the packet flow. EX1006, 7:33-54. Chua discloses that in SDNs, service devices may provide security related functionalities, such as intrusion detection. EX1006, 5:50-6:24.

In addition to controlling the SDN, Chua discloses the controller may execute other programs. EX1006, 6:36-52; Min, ¶89. Chua discloses controllers hosting Layer 3-Layer 7 services instead of external service devices. EX1006, 6:42-52. As an example, Chua discloses the controller performing DPI on packets from client

devices. EX1006, 10:48-52, 31:48-59, 6:45-52 (describing a Layer 3 routing stack executing on the controller, and the controller commanding network devices to update forwarding tables in response).

C. U.S. Patent No. 9,813,447 (“Rash”) (EX1007)

Rash was filed on March 15, 2013. EX1007, Cover. Rash is prior art under 35 U.S.C. § 102(a)(2).

Rash discloses a network that adjusts policies associated with the operation of devices on the network. EX1007, Abstract; Min, ¶¶90-92. Rash discloses “snooping” of the network traffic flows to identify applications operating on the network. *See, e.g.*, EX1007, 6:22-40. Rash discloses using signatures, heuristics, and other techniques including “port values, protocol values, statistics analysis, customized programs dependent on characteristics unique to a program and OSI layer 2 information.” EX1007, 6:50-57.

To process this information, Rash discloses a targeted approach where packets, or portions of packets, are mirrored to a destination other than the original intended destination. *See, e.g.*, EX1007, 7:51-59, 9:3-12, 25:30-55, 26:11-35, 27:31-43, 29:58-62, 35:66-36:2. Rash discloses this helps minimize the data mirrored to identification appliances or other infrastructure devices configured to carry out the application identification function, helping reduce costs. EX1007, 12:59-13:8, 34:43-51; Min, ¶92.

D. RFC 793 (EX1015)

RFC: 793, Transmission Control Protocol, DARPA Internet Program Protocol Specification (“RFC 793”) was published September 1981. EX1015, 1; EX1020, ¶¶43-50, 58; Min, ¶¶93-94.

RFC 793 defines the well-known Transmission Control Protocol. EX1015, 1. RFC 793 would have been widely known to POSITAs in April 2014 and part of the general knowledge of the state of the art. Min, ¶93; EX1020, ¶49; EX1026, ¶0063 (patent application published April 26, 2007 citing RFC 793). “Request for Comments” (RFC) documents were well-known as of the priority date of the ’111 Patent, and were easily accessible to POSITAs. Min, ¶94. RFC documents are published by the Internet Engineering Task Force (IETF). EX1020, ¶¶43-48. RFCs can be obtained from a number of Internet hosts. EX1025, 6. IETF documents are “readily available to a wide audience.” EX1025, 8. IETF participants “shall publicly announce...every activity” relating to the standardization process. EX1025, 26. A POSITA in the field of network communications, including SDNs, would have reasonably relied upon such documents and their content would have been part of the background knowledge of a POSITA. Min, ¶¶93-94.

IX. THE ASSERTED GROUNDS OF INVALIDITY

A. Ground 1: Claims 1-6, 9-15, 18-19, 21-36, 41-42, and 44-54 are rendered obvious by Lefebvre in view of Chua.

For the reasons set forth below, Claims 1-6, 9-15, 18-19, 21-36, 41-42, and

44-54 would have been obvious to a POSITA in view of Lefebvre and Chua. Min, ¶¶73-77, 95-96.

1. A POSITA Would Have and Could Have Combined Lefebvre with Chua

A POSITA would have been motivated and found it obvious to modify Lefebvre with Chua's teachings. Min, ¶¶97-107. Lefebvre discloses techniques for an SDN to steer traffic through inline services. EX1005, 1:13-21, 2:39-47. Lefebvre identifies several types of services that could form part of the service chain, including DPI. EX1005, 1:28-38, 4:48-54. Lefebvre discloses that traffic steering requires classification of incoming packets and assignment of a service path based on predefined policies. EX1005, 6:17-19. Then the packets are forwarded to the next service in the assigned service path. EX1005, 6:20-21. Lefebvre discloses how operators define the traffic classes and service chains for processing traffic flows on a per packet basis. EX1005, 4:41-43, 6:17-19. Lefebvre discloses programming operator instructions into the switches as rules, routing only certain network traffic through the service chains. EX1005, 4:41-47. Lefebvre discloses that at least some of the switches in the SDN use OpenFlow. EX1005, 5:41-44; Min, ¶¶97-98.

Like Lefebvre, Chua discloses an SDN where a controller determines the behavior of devices in the network and how traffic flows through the network. EX1006, 1:25-35. Like Lefebvre, Chua discloses OpenFlow devices in the

network. EX1006, 1:35-41. These network devices provide data to Chua's controller, which then makes network enforcement decisions. EX1006, 6:25-28. Chua's controller considers the traffic on a per-packet or per-flow basis. EX1006, 7:33-54, 10:48-52, 15:9-17. Chua's controller then programs network devices of the SDN to implement these decisions. EX1006, 7:33-54 (describing different programmed actions taken "on packets of a packet flow"), 31:30-34. Chua's service devices include those that execute DPI, for example, along with other techniques including intrusion detection service and firewalls (e.g., web-application firewalls). EX1006, 5:59-67, 10:48-52, 15:4-9, 16:30-33, 31:48-50, Fig. 9 (step 402). Chua also discloses that the "SDN controller 112 itself may host one or more L4-L7 services, rather than (or in addition to) services, provided by service devices 116." EX1006, 6:42-45; *see also* EX1006, 6:36-52. And Chua discloses the controller performing DPI. EX1006, 10:48-52, 15:4-9, 16:30-33, 31:48-50; Min, ¶99.

A POSITA would have been motivated to modify Lefebvre with Chua to help reduce the cost and complexity of the separate dedicated service appliances that Lefebvre discloses existed in the prior art. Min, ¶¶100-104. A POSITA would have been motivated and found it obvious to modify Lefebvre with Chua because both describe SDNs, and both describe DPI as one type of service in an SDN. Min, ¶100. EX1005, 1:28-38; EX1006, 10:48-52, 15:4-9, 16:30-33, 31:48-50, Fig. 9 (step 402). Additionally, both Lefebvre and Chua disclose use of OpenFlow

devices and share similar techniques for classifying the traffic traversing their networks. EX1005, 5:41-44; EX1006, 1:35-41; Min, ¶¶100-101. Lefebvre and Chua are therefore in the same field of endeavor and contain overlapping disclosures with similar purposes. Min, ¶100.

A POSITA would have been motivated to look to Chua's more detailed teachings regarding actions an SDN controller takes in response to information from security-related services like DPI and apply them to Lefebvre. Min, ¶102. Lefebvre mentions DPI as one known service, but focuses its disclosure on actions relevant to traffic steering and the passing of the packet through assigned service chains including redirecting or dropping the packet. See EX1005, 6:17-31, 1:28-38 (discussing DPI); see also EX1005, 8:49-65, Table 1, Table 2 (depicting OpenFlow switch table actions). Lefebvre does not expressly disclose specifics regarding actions resulting from the services processing the traffic along the service path. See generally, Lefebvre. When implementing DPI in an SDN like Lefebvre's, a POSITA would have sought details regarding actions taken after DPI identifies malicious traffic. Min, ¶102. Chua discloses details regarding actions an SDN controller may take, including allowing, blocking, mirroring, redirecting, transforming, or encapsulating the network traffic. EX1006, 7:33-54. A POSITA would therefore have looked to Chua's teachings regarding actions an SDN controller takes in response to information from security-related services like DPI and apply them to Lefebvre when implementing DPI in the SDN. Min, ¶102.

Chua also discloses that the controller may execute other programs instead of external service devices. EX1006, 6:42-45. Lefebvre discloses that services including DPI are typically dedicated, expensive appliances. EX1005, 1:25-27, 1:34-38. Chua, however, discloses programming an SDN controller to perform DPI and other functions performed by service devices. EX1006, 31:48-50, 6:42-45. A POSITA would recognize Chua's teaching means performing DPI and other services is not an expensive task that requires dedicated appliances. Min, ¶103. Similarly, a POSITA would recognize that incorporating services in Lefebvre's controller (e.g., DPI), as Chua teaches, removes the need for the expensive dedicated appliance. Min, ¶103.

Additionally, a POSITA would have also been motivated to modify Lefebvre with Chua's programmable controller teachings because Chua discloses how programs executing on the controller may utilize the SDN controller's existing abilities to implement appropriate updates to the SDN to facilitate the programs' functionality. EX1006, 6:36-52; Min, ¶104. And including within the controller the service device's functionality, as Chua teaches, would remove the need to address network failures between the controller and service device, which Chua describes as a "complex problem[.]" EX1006, 19:1-53 (describing redundancy possibilities for SDN appliances); Min, ¶104. Chua's controller teachings therefore help simplify the SDN and reduce the need to address certain types of failures in the SDN, giving the POSITA further motivation to consider Chua. Min,

¶104.

Implementing Chua's teachings in Lefebvre's controller would have been well within the skill of a POSITA, and the POSITA would have had a reasonable expectation of success. Min, ¶¶105-107. Lefebvre discloses redirecting packets to the appropriate service. EX1005, 8:49-65, Table 1, Table 2 (depicting OpenFlow switch table actions). Chua discloses additional actions an SDN controller may take in response to the information provided by services such as DPI like allowing, blocking, mirroring, redirecting, transforming, or encapsulating the network traffic. EX1006, 7:28-54. Implementing these additional actions in Lefebvre's controller would have been straightforward for a POSITA because the actions taught by Chua build on standard actions many networking devices already implement, and Lefebvre discloses redirecting of packets. See EX1005, 8:49-65, Table 1, Table 2 (depicting OpenFlow switch table actions); Min, ¶105. For example, OpenFlow switches support basic actions including forwarding, encapsulating, and dropping packets. EX1013, 3.

Implementing a DPI service in Lefebvre's controller, as Chua teaches, would also have been straightforward for a POSITA. Min, ¶106. Lefebvre discloses embodiments that exist as a software product executing programs. EX1005, 17:17-37. Chua confirms how controllers include processors and memory. EX1006, 10:67-11:6, 31:60-32:6. Thus, a POSITA would recognize that Lefebvre's controller includes programmable components including processors and memory.

Min, ¶106. Chua discloses “a software development kit (SDK)” for creating programs that can be executed by the SDN controller. EX1006, 6:38-40. As an example, Chua discloses how an SDN controller performs DPI on packets from a client device. EX1006, 31:49-54. Thus, only routine skill would be required to implement Chua’s teachings in Lefebvre. Min, ¶106.

And a POSITA would have had a reasonable expectation of success in modifying Lefebvre with Chua’s controller teachings. Min, ¶107. Because Lefebvre discloses how its embodiments exist as software programs (EX1005, 17:17-37), and Chua discloses the inclusion of an SDK (EX1006, 6:37-47) and example programs executing on its controller including DPI (EX1006, 6:47-52, 31:49-54), a POSITA would have had a reasonable expectation of success in implementing DPI and other services in Lefebvre’s controller, as Chua teaches. Min, ¶107.

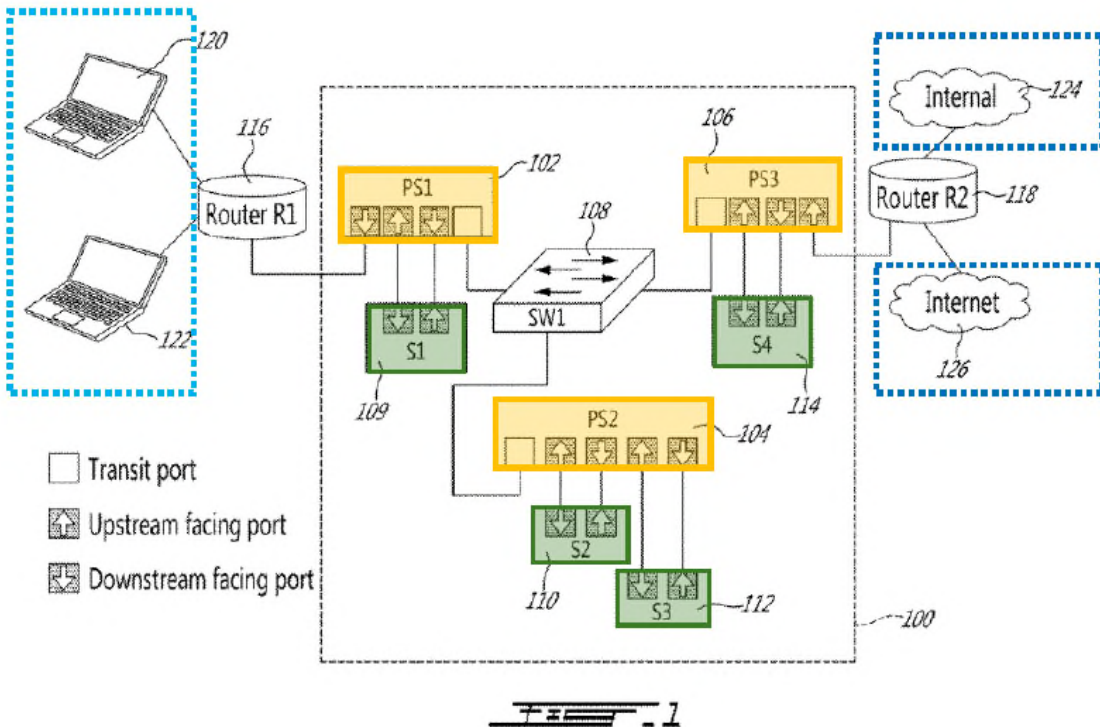
2. Independent Claims 1 and 32

a) Claim 1

- i. 1[Pre]: A method for use with a packet network including a network node for transporting packets between first and second entities under control of a controller that is external to the network node, the method comprising:**

If the preamble is limiting, the Lefebvre-Chua combination renders it obvious. Min, ¶¶108-114. Lefebvre discloses how operators “define service policies that specify traffic classes and the chain of services that each class must traverse” and

how the “policies are translated by the controller into rules that are programmed on the switches in the service network.” EX1005, 4:41-47; *see also* EX1005, 1:13-21; 6:26-28. Annotated FIG. 1 of Lefebvre below highlights certain components.



EX1005, FIG. 1 (annotated); Min, ¶109.

Lefebvre discloses that incoming traffic “always enters the service network 100 via a perimeter switch and through a node port,” and that arriving packets are processed and steered to the next node in the assigned service paths. EX1005, 6:5-11. Lefebvre discloses **user equipment 120, 122** connected to router R1 and the **internal network 124** and the **Internet 126** connected to router R2. EX1005, 5:38-54, 6:13-16. Lefebvre’s user equipment, internal network, and Internet therefore disclose the “packet network” and “first and second entities.” Min, ¶110.

Lefebvre's service network includes **perimeter switches 102, 104, 106** and an inner switch SW1. EX1005, 5:38-44. The network also includes **services 109, 110, 112, 114**. EX1005, 5:44-47. Lefebvre discloses that the **perimeter switches 102, 104, 106** may be OpenFlow switches whereas the inner switch 108 may be "either an OpenFlow switch or a plain Ethernet switch." EX1005, 5:41-44. Each of Lefebvre's **perimeter switches 102, 104, 106** therefore disclose the "network node" of the claims because they, like the '111 Patent's network nodes, communicate with the central controller using OpenFlow. See EX1001, 3:47-48; Min, ¶111.

And even though Lefebvre states that the centralized controller is "not shown in FIG. 1," Lefebvre teaches that its controller is external to the network node because Lefebvre discloses that its centralized controller pushes these policies to the relevant switches. EX1005, 6:28-31 ("These policies can be specified by the operator and pushed to the relevant switches by a centralized controller[.]"). A POSITA would therefore understand that Lefebvre's controller is external to the **perimeter switches 102, 104, 106**, especially given that Lefebvre's system involves an SDN. See, e.g., Min, ¶112; EX1006, 1:27-29 ("In general, software defined networks involve the use of a standalone controller that performs the control functionality for a set of network devices."), 1:46-51 (an SDN "is generally a network of interconnected computing devices having forwarding planes or data planes that can be programmed remotely by one or more controller devices.").

Furthermore, Lefebvre describes using OpenFlow 1.1. EX1005, 4:30-40.

OpenFlow 1.1 uses “an external controller” as shown in Figure 1 below:

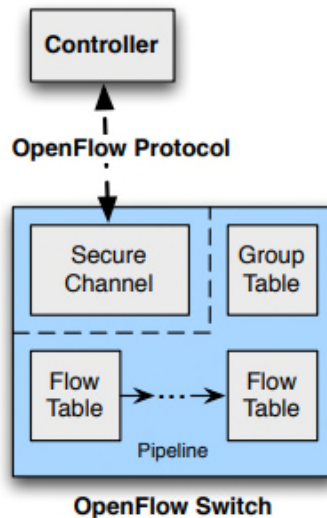


Figure 1: An OpenFlow switch communicates with a controller over a secure connection using the OpenFlow protocol.

EX1027, 3, Fig. 1.

And as discussed in Section IX.A.1, a POSITA would have been motivated and found it obvious to apply Chua’s teachings to Lefebvre’s SDN controller. Min, ¶113. Like Lefebvre, Chua teaches the use of an SDN controller 112 that is external to network devices 108, 110, which Chua teaches may be “switches”:

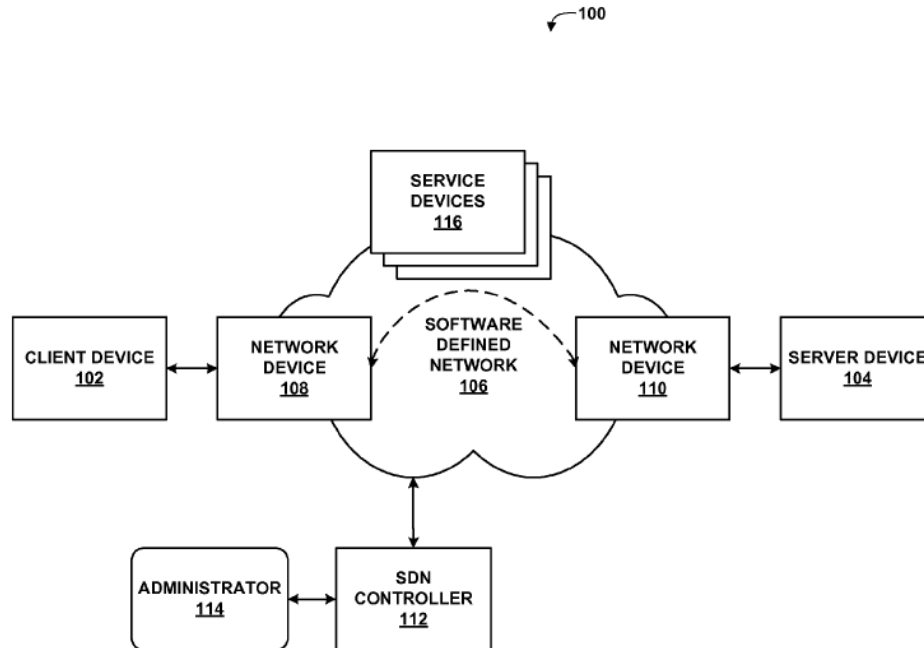


FIG. 1

EX1006, FIG. 1, 5:32-37. As explained above in Section IX.A.1, in the combined system, Lefebvre’s modified controller is programmed to execute DPI, as Chua teaches. Thus, a POSITA would recognize that Lefebvre’s modified controller discloses the claimed “controller” under either of the constructions identified above in Section VII.A. Min, ¶113.

- ii. **1[a]: sending, by the controller to the network node over the packet network, an instruction and a packet-applicable criterion;**

The Lefebvre-Chua combination renders obvious element 1[a]. Min, ¶¶115-120.

The ’111 Patent uses the word “instruction” to describe various commands, including “probe,” “mirroring,” and “termination” instructions. EX1001, 4:23-26; *see also* EX1001, 4:26-46 (describing redirecting, forwarding, and terminating

instructions), 4:47-56 (describing mirroring instructions). As discussed in Section VII.B, a POSITA would understand a “packet-applicable criterion” to include the conditions, or basis, used to identify packets for actions, such as inspection. Min, ¶115.

Lefebvre discloses that the operator defines “service policies that specify traffic classes and the chain of services that each class must traverse.” EX1005, 4:43-45. Lefebvre’s controller programs the rules into the switches of the service network, and the “rules steer the network traffic through the ordered chain of services as specified by the policies.” EX1005, 4:43-47, Table 1, Table 2 (depicting OpenFlow switch table actions), FIG. 5; Min, ¶116. The first step in Lefebvre’s traffic steering process “classifies incoming packets and assigns them a service path based on predefined policies.” EX1005, 6:17-19, 7:35-8:48, FIG. 2, FIG. 5 (step 510). The second step “forwards packets to a ‘next’ service based on its current position along its assigned service path.” EX1005, 6:20-21, FIG. 5 (step 530).

Lefebvre’s traffic steering involving classification of incoming packets to assign them a service path discloses the claimed packet-applicable criterion. Min, ¶117. Lefebvre’s traffic steering policies include subscriber-based policies, application-based policies, and flow-based policies. EX1005, 6:26-31. Lefebvre’s traffic steering mechanism presumes that all traffic traversing the service network is subscriber traffic. EX1005, 5:19-37. But as Lefebvre also discloses,

when the traffic is not subscriber traffic, the SDN functions as a simple router that “does conventional destination-IP-based forwarding” where the traffic does not traverse inline services. EX1005, 14:23-33. Lefebvre also classifies traffic based on the incoming port, the packet’s destination MAC address, the source or destination IP address, the IP address and/or protocol and/or port number, and the position of the packet on the service path. EX1005, 7:35-8:48 (describing “an example methodology for processing and classifying a packet”), Table 1, Table 2 (depicting OpenFlow switch table actions). Lefebvre’s classification techniques disclose the claimed packet-applicable criterion because they are the conditions, or basis, used to identify packets for actions. Min, ¶117. Lefebvre discloses the classification decides which service path the packet will traverse. Lefebvre’s next destination table routes the packet to the appropriate service. EX1005, 8:49-65, Table 1, Table 2 (depicting OpenFlow switch table actions). Like the ’111 Patent’s claimed packet-applicable criterion, Lefebvre’s classification techniques depend on instructions provided by the controller and determine which service chain the packet will traverse. EX1001, 4:12-46.

A POSITA would have also understood that Lefebvre’s traffic steering involving classification of incoming packets to assign them a service path discloses the claimed instruction. Min, ¶118. For example, Lefebvre’s final table, the “next destination table,” matches the packet to the appropriate service. EX1005, 8:49-65, Table 1, Table 2 (depicting OpenFlow switch table actions).

Lefebvre teaches that its “policies can be specified by the operator and pushed to the relevant switches by [the] centralized controller[,]” which a POSITA would have understood to teach sending, by the controller to the network node (e.g., one or more of perimeter switches 102, 104, 106), the instruction and the packet-applicable criterion. EX1005, 6:28-31; Min, ¶119. Chua similarly discloses the controller programs network devices “to perform programmed actions on packets” including allowing, blocking, mirroring, redirecting, transforming, and encapsulating packets. EX1006, 7:28-54. Lefebvre’s controller already redirects. See EX1005, 8:49-65, Table 1, Table 2 (depicting Open-Flow switch table actions). As modified by Chua’s teachings, Lefebvre’s controller’s instructions would include the allowing, blocking, and mirroring packets. Min, ¶119.

iii. 1[b]: receiving, by the network node from the controller, the instruction and the criterion;

The Lefebvre-Chua combination renders obvious element 1[b]. Min, ¶¶121-123. As discussed above for element 1[a], a POSITA would have understood that Lefebvre’s policy-based traffic steering process involving classification of incoming packets to assign them a service path discloses the claimed instruction and criterion. Min, ¶122. Lefebvre’s “policies can be specified by the operator and pushed to the relevant switches by [the] centralized controller[,]” which a POSITA would have understood to teach receiving, by the network node (e.g., one or more of perimeter switches 102, 104, 106) from the controller, the instruction and the criterion as

recited in element 1[b]. EX1005, 6:28-31; Min, ¶122.

- iv. 1[c]: receiving, by the network node from the first entity over the packet network, a packet addressed to the second entity;**

Lefebvre discloses element 1[c]. Min, ¶¶124-126. Lefebvre states that incoming traffic “always enters the service network 100 via a perimeter switch and through a node port.” EX1005, 6:5-8. Lefebvre’s perimeter switches correspond to the claimed network nodes for the reasons discussed for 1[pre]. Min, ¶125. As discussed for 1[pre], Lefebvre discloses **user equipment 120, 122** and the **internal network 124** along with the Internet 126, which teach the “first and second entities.” Min, ¶125. For example, when Lefebvre’s **user equipment 120, 122** communicates with the **internal network 124** using the service network 100, the **user equipment 120, 122** discloses the first entity. EX1005, 5:38-54, 6:8-16; Min, ¶125. And a POSITA would have understood that in doing so, the SDN receives from the **user equipment 120, 122** a packet addressed to the second entity, **internal network 124**. Min, ¶125. Lefebvre confirms this when disclosing the source and destination address fields of the received packet. EX1005, 2:53-64. Similarly, Lefebvre discloses that within the network, at transit ports, packets are forwarded to their destination MAC addresses. EX1005, 5:65-6:12, 7:55-58.

- v. 1[d]: checking, by the network node, if the packet satisfies the criterion;**

Lefebvre discloses element 1[d]. Min, ¶¶127-130. As discussed for

elements 1[a]-1[b], Lefebvre discloses that the operator defines service policies specifying traffic classes and the chain of services the traffic classes must traverse, and that the first step of traffic steering involves the classification of incoming packets. EX1005, 4:43-45, 6:18-19, 7:35-8:48 (describing an example methodology), FIG. 2, FIG. 5 (step 510); Min, ¶128. Lefebvre's policies are implemented by the perimeter switches. EX1005, 4:43-45 ("These policies are translated by the controller into rules that are programmed on the switches in the service network."), 6:5-8. As discussed for elements 1[a]-1[b], Lefebvre's classification techniques disclose the claimed criterion because they are the conditions, or basis, used to identify packets for actions. See EX1005, 8:49-65, Table 1, Table 2 (depicting OpenFlow switch table actions); Min, ¶128.

A POSITA would have understood Lefebvre's classification of incoming packets discloses "checking, by the network node, if the packet satisfies the criterion." Min, ¶129. Lefebvre's classification checks the incoming packet against the claimed criterion and determines the service path the packet will traverse.

vi. 1[e]: responsive to the packet not satisfying the criterion, sending, by the network node over the packet network, the packet to the second entity; and

Lefebvre discloses element 1[e]. Min, ¶¶131-133. Lefebvre assumes that the forwarded traffic is subscriber traffic. EX1005, 14:23-25. As discussed for element 1[a], Lefebvre's distinction between subscriber and non-subscriber traffic

is one type of criterion. Min, ¶132. Lefebvre discloses simply forwarding non-subscriber traffic to its destination. EX1005, 14:25-29; Min, ¶132. Because Lefebvre's non-subscriber traffic does not satisfy the criterion of directing subscriber traffic, the network routes the packets from their source, the claimed first entity, to their destination, the claimed second entity. Min, ¶132.

- vii. 1[f]: responsive to the packet satisfying the criterion, sending the packet, by the network node over the packet network, to an entity that is included in the instruction and is other than the second entity.**

The Lefebvre-Chua combination discloses element 1[f]. Min, ¶¶134-137. As discussed for element 1[a], Lefebvre's classification techniques determine which service chain the traffic traverses. EX1005, 8:49-65, Table 1, Table 2 (depicting OpenFlow switch table actions). A POSITA would have understood that Lefebvre's services are "other than" the intended destination of the packet. Min, ¶135.

Further, as explained in Section IX.A.1, Lefebvre's controller modified by Chua's teachings would include additional actions including mirroring or redirecting the packets to the service chain. Min, ¶136. Chua's mirroring action allows the traffic and also sends a copy of the traffic for deeper inspection or recording. EX1006, 7:43-45, 28:7-32. A POSITA would have understood that both actions, mirroring and redirecting of the packets to the service chain, send the packet to an entity that is "other than the second entity" as claimed. See EX1005, 15:27-46

(disclosing forwarding the packet “to the next service on the assigned service chain” and “assigning a new destination address to the received packet in accordance with the next service to be performed”); Min, ¶136. Further, a POSITA would understand that Lefebvre’s controller, modified by Chua’s teachings, would execute DPI and therefore be the “entity that is included in the instruction,” which would also be a destination “other than the second entity.” Min, ¶136.

b) Claim 32

Independent claim 32 is substantially similar to claim 1 above. Min, ¶233. The primary differences are that claim 32 recites a “method by the network node” and does not recite the step of “sending, by the controller to the network node over the packet network, an instruction and a packet-applicable criterion.” Lefebvre in view of Chua renders obvious independent claim 32 for substantially the same reasons as discussed above for independent claim 1. Min, ¶233.

i. 32[pre]

If the preamble is limiting, the Lefebvre-Chua combination renders it obvious for the reasons discussed for preamble 1[pre]. Min, ¶¶234-235.

ii. 32[a]

The Lefebvre-Chua combination renders obvious element 32[a] for the reasons discussed for elements 1[a]-1[b]. Min, ¶¶236-237.

iii. 32[b]

The Lefebvre-Chua combination renders obvious element 32[b] for the

reasons discussed for element 1[c]. Min, ¶¶238-239.

iv. 32[c]

The Lefebvre-Chua combination renders obvious element 32[c] for the reasons discussed for element 1[d]. Min, ¶¶240-241.

v. 32[d]

The Lefebvre-Chua combination renders obvious element 32[d] for the reasons discussed for element 1[e]. Min, ¶¶242-243.

vi. 32[e]

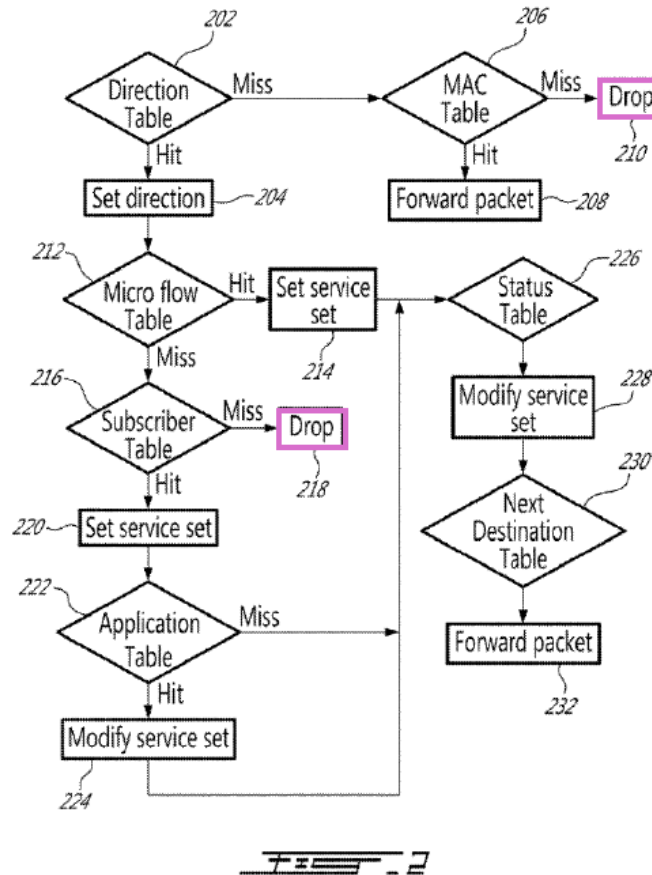
The Lefebvre-Chua combination renders obvious element 32[e] for the reasons discussed above for element 1[f]. Min, ¶¶244-245.

3. Dependent Claims 2-6, 9-15, 18-19, 21-31, 33-36, 41-42, and 44-54

As discussed, claims 1 and 32 are substantially similar. There is also substantial overlap among the subject matter of their respective dependent claims. Accordingly, the following sections address separate claims depending from claims 1 and 32, respectively, together, as appropriate.

a) Claims 2 and 33

The Lefebvre-Chua combination renders obvious claims 2 and 33. Min, ¶¶138-142, 246-247. FIG. 2 of Lefebvre depicts “an example methodology for processing and classifying a packet.” EX1005, 7:37-39.



EX1005, FIG. 2 (annotated); Min, ¶139.

In example embodiments, Lefebvre discloses that when a packet arrives on a transit port, it will “miss” the direction table and proceed to the MAC table 206. EX1005, 7:52-54. If the packet’s destination MAC address does not match the contents of the MAC table, Lefebvre discloses dropping the packet. EX1005, 7:55-58; *see also* EX1005, 8:11-13, FIG. 2 (step 218) (also describing that the default action is to drop the packet when there is a miss in the subscriber table). Lefebvre therefore discloses dropping the packet, but does not explicitly disclose a terminate instruction that results in blocking. Min, ¶140.

As discussed for element 1[a], Chua discloses the SDN controller programs

network devices to perform actions, including blocking. EX1006, 7:35-54. Chua's blocking action explicitly blocks the packets in the packet flow from traversing the SDN. EX1006, 7:35-38, 7:41-42. Chua's SDN controller 112 may perform these actions, including blocking, in response to information from service devices, external triggers or events, or business-level security rules. EX1006, 7:28-38, 16:23-44, 28:7-32, 8:43-47, 17:26-35, 14:39-42; Min, ¶141.

The Lefebvre-Chua combination would therefore include instruction resulting in blocking, which a POSITA would have recognized as the terminate instruction required by claims 2 and 33. Min, ¶141. As discussed in Section IX.A.1, Lefebvre's controller, as modified by Chua's teachings, would implement Chua's actions including blocking. See EX1005, 8:49-65, Table 1, Table 2 (depicting OpenFlow switch table actions); EX1006, 7:33-54 (describing available actions for an SDN controller). In doing so, the modified Lefebvre controller would provide the instruction, which Chua teaches, to the network devices directing when to apply dropping the packet, which Lefebvre discloses. In doing so, the network devices in Lefebvre's SDN would block traffic identified by Lefebvre's controller, which executes DPI as taught by Chua. EX1006, 31:49-54.

b) Claims 3 and 34

The Lefebvre-Chua combination renders obvious claims 3 and 34. Min, ¶¶143-146, 248-249. Lefebvre does not explicitly disclose a mirroring instruction. Min, ¶144. But as discussed for element 1[a], Chua discloses the SDN controller

programs network devices to perform actions, including mirroring. EX1006, 7:28-54. Thus, the Lefebvre-Chua combination would include the claimed mirror instruction. Min, ¶144. As discussed for element 1[f], the Lefebvre-Chua combination renders obvious the claimed “responsive to the packet satisfying the criterion” aspect of claims 1 and 32. Min, ¶144.

Chua’s mirroring allows the traffic and also sends a copy of the traffic for deeper inspection or recording. EX1006, 7:43-45, 28:7-32; Min, ¶144. Chua’s SDN controller may perform these actions, including mirroring, in response to information from service devices or business-level security rules. EX1006, 7:28-38, 16:23-30, 16:36-44, 28:7-32, 14:39-42. As discussed in Section IX.A.1, Lefebvre’s controller, as modified by Chua’s teachings, would implement Chua’s actions including mirroring. See EX1005, 8:49-65, Table 1, Table 2 (depicting OpenFlow switch table actions); EX1006, 7:33-54 (describing available actions for an SDN controller). A POSITA would have understood the network devices in Lefebvre’s SDN would therefore send the packet to Lefebvre’s modified controller as well as the destination (in other words, the second entity). Min, ¶145. And Lefebvre’s controller would execute DPI, as taught by Chua. See EX1006, 31:49-54.

c) Claims 4 and 35

The Lefebvre-Chua combination renders obvious claims 4 and 35. Min, ¶¶147-153, 250-251. The ’111 Patent discloses that the probe instruction includes

matching certain aspects of a packet's header and then redirecting the packet to the controller, and matching certain other aspects of a packet's header, and then redirecting the packet to the destination. EX1001, 4:23-33; *see also* EX1001, 6:30-32, FIG. 3 (step S305).

Lefebvre selectively steers traffic away from expensive services, such as DPI, but does not explicitly disclose a probe instruction. *See, e.g.*, EX1005, 1:34-38. But as discussed for element 1[a], Chua discloses the SDN controller programs network devices to perform actions including redirecting. EX1006, 7:28-54. Because the claimed probe instruction involves redirects, the Lefebvre-Chua combination renders obvious the claimed probe instruction. Min, ¶149. And as discussed for element 1[f], the Lefebvre-Chua combination renders obvious the “responsive to the packet satisfying the criterion” aspect of claims 4 and 35.

Chua's redirect action sends the traffic to another network device for either inspection or to occupy a potential hacker while determining if there is a real security threat. EX1006, 7:46-50. Because Lefebvre's controller, modified by Chua's teachings, executes DPI instead of a service device, the Lefebvre-Chua combination renders obvious the “sending the packet, by the network node, to the controller” and “responsive to receiving the packet, analyzing the packet, by the controller” elements of claim 4, and the analogous element of claim 35 (“sending the packet to the controller”). Min, ¶150.

Further, Chua discloses that if the service device deems the packets malicious,

the controller may instruct the service device to send data back to the switch such as “data of the packet flow after the data has been inspected.” EX1006, 21:35-42, 28:18-32 (disclosing the controller “may program network devices to, in response to an indication that packets of a packet flow include malicious data, forward packets of the packet flow to a destination of the packet flow”). Chua’s SDN controller performs these actions, including redirecting, in response to information from service devices or business-level security rules. EX1006, 7:28-38, 16:23-30, 16:36-44, 28:7-32, 14:39-42. Although Chua discloses this in relation to an intrusion detection system (IDS), a POSITA would have found it obvious that service devices executing DPI would have received similar instructions. Min, ¶151. This is because IDS systems also analyze network traffic for malicious activity but typically do not employ the more rigorous inspection of packets that DPI systems use. Min, ¶151. Because Chua discloses sending data back to the network node, such as “data of the packet flow after the data has been inspected,” the Lefebvre-Chua combination renders obvious the “sending the packet, by the controller, to the network node” element of claim 4 and the analogous element of Claim 35 (“receiving the packet, from the controller”). See EX1006, 21:35-42, 28:18-32; Min, ¶151.

As discussed in Section IX.A.1, Lefebvre’s controller, as modified by Chua’s teachings, would implement Chua’s teachings regarding SDN actions, including redirecting. See EX1005, 8:49-65, Table 1, Table 2 (depicting OpenFlow switch

table actions); EX1006, 7:33-54 (describing available actions for an SDN controller). And Lefebvre’s controller would execute DPI, as Chua teaches. See EX1006, 31:49-54; Min, ¶152. As modified by Chua’s teachings, Lefebvre’s controller would instruct network devices to redirect certain traffic to the controller which would then execute DPI. See EX1006, 7:46-50; Min, ¶152. Once the modified Lefebvre controller completes DPI, the controller sends the inspected packet flow back to the network device for transmission onward to the second entity, as Chua teaches. See EX1006, 21:35-42, 28:18-32. This renders obvious “responsive to receiving the packet, sending the packet, by the network node, to the second entity” element of claim 4 and the analogous element of Claim 35 (“responsive to receiving the packet, sending the packet, to the second entity”). Min, ¶152.

d) Claims 5 and 36

The Lefebvre-Chua combination renders obvious claims 5 and 36. Min, ¶¶154-157, 252-253.

As discussed for 1[f], the Lefebvre-Chua combination renders obvious the “responsive to the packet satisfying the criterion and to the instruction” aspect of claims 5 and 36. Min, ¶155.

As discussed in Section IX.A.1, Lefebvre’s controller, as modified by Chua’s teachings, would implement Chua’s actions including redirecting. See EX1005, 8:49-65, Table 1, Table 2 (depicting OpenFlow switch table actions); EX1006, 7:33-

54 (describing available actions for an SDN controller). And Lefebvre's controller would execute DPI, as taught by Chua. See EX1006, 31:49-54; Min, ¶156. Thus, Lefebvre's modified controller would instruct network devices to redirect certain traffic to the controller which would then execute DPI. And because Lefebvre contemplates steering entire packets, the modified Lefebvre controller would continue to redirect the entire packet. EX1005, 6:17-31; *see also* EX1005, 1:28-38 (discussing DPI); Min, ¶156.

e) Claim 6

The Lefebvre-Chua combination renders obvious dependent claim 6. Min, ¶¶158-163. As discussed above for claim 5, in the combined system, Lefebvre's modified controller would instruct network devices to redirect certain traffic to the controller, which would then execute DPI as taught by Chua. See EX1006, 31:49-54; Min, ¶159. And as discussed in Section IX.A.1, a POSITA would recognize that Lefebvre's controller includes memory. Min, ¶161. It would have been obvious to a POSITA that the process of Lefebvre's modified controller executing DPI on a received packet would involve the controller storing the received packet, or a portion thereof, in memory as claimed. Min, ¶¶160, 162. DPI is a content-based technique that inspects both the header and the payload of a packet to classify network traffic. EX1001, 1:21-29; EX1008, 1-2. A POSITA would have understood that the received packet, or a portion thereof, would be stored in some form of memory by the controller in order to permit the controller to perform DPI.

Min, ¶162.

f) Claim 9

The Lefebvre-Chua combination renders obvious dependent claim 9. Min, ¶¶164-166. As discussed above for claim 5, in the combined system, Lefebvre’s modified controller would instruct network devices to redirect certain traffic to the controller, which would then execute DPI as taught by Chua. See EX1006, 31:49-54; Min, ¶165. A POSITA would have understood that the DPI performed by the modified Lefebvre controller involves “analyzing the packet” as claimed. See, e.g., EX1008, 2 (“DPI technique inspects both the header and the payload of a packet.”); Min, ¶165.

g) Claim 10

The Lefebvre-Chua combination renders obvious claim 10. Min, ¶¶167-172. Claim 10 recites an “application server.” The ’111 Patent describes that “[a]n application server 120 executes, for example, security applications (e.g., Firewall, intrusion detection, etc.), data analytic applications, and so on.” EX1001, 3:56-61.

As discussed above for claims 5 and 9, Lefebvre’s controller, modified by Chua’s teachings, executes DPI instead of a service device. Min, ¶169. Accordingly, the Lefebvre-Chua combination renders obvious the claimed “sending the packet or a portion thereof, by the network node, to the controller” and “responsive to receiving the packet, analyzing the packet, by the controller”

elements of claims 5 and 9. Min, ¶169.

Chua further teaches that its controller sends the packet to other devices based on the DPI. Min, ¶170. For instance, Chua discloses that certain types of traffic “must often be distributed to a variety of appliances or server-based application-aware devices based on potentially deep inspection of incoming (and for certain flow types, outgoing) packets.” EX1006, 15:5-9; *see also* EX1006, 15:9-17. Chua’s appliances or server-based application-aware devices correspond to the claimed “application server that communicates with the controller.” Min, ¶170. Chua’s disclosure that traffic is distributed to appliances or server-based application-aware devices based on DPI (performed by Chua’s controller) teaches sending the packet, by the controller, to the application server as claimed. Min, ¶170.

A POSITA would have understood that packets sent to the appliances or server-based application-aware devices based on DPI would be analyzed, which teaches “analyzing the packet by the application server” as claimed. Min, ¶171. Chua teaches that these devices provide “security” and/or “data analytic” applications consistent with the description in the ’111 Patent. EX1001, 3:56-61. For example, similar to the ’111 Patent, Chua’s SDN controller may determine that a particular packet flow should be inspected for intrusion detection. EX1006, 21:35-37; EX1001, 3:56-61.

h) Claim 11

The Lefebvre-Chua combination renders obvious claim 11. Min, ¶¶173-

175. As discussed for Claim 10, Chua teaches that its controller sends a packet to an application server for analysis (e.g., for intrusion detection) based on DPI. Min, ¶174. Furthermore, as discussed for claims 4 and 35, Chua discloses that the controller may instruct the service device to send data back to the switch, such as data of the packet flow after the data has been inspected. EX1006, 21:35-42, 28:18-32. Because Chua also teaches that the controller can send packets to service devices for analysis (similar to the switches), a POSITA would have understood Chua's teaching of sending data (e.g., data of the packet flow after the data has been inspected) would also be applicable to scenarios in which the controller sends a packet to a device for further inspection (e.g., in the case of Chua's controller distributing traffic to an appliance or server-based application-aware device, such as intrusion detection, based on DPI). EX1006, 15:5-17, 21:35-37; Min, ¶174. Accordingly, the Lefebvre-Chua combination renders obvious "sending the packet after analyzing by the application server to the controller" aspect of Claim 11. See EX1006, 21:35-42, 28:18-32; Min, ¶174.

As discussed in Section IX.A.1, Lefebvre's controller, as modified by Chua's teachings, would implement Chua's teachings regarding SDN actions, including redirecting. See EX1005, 8:49-65, Table 1, Table 2 (depicting OpenFlow switch table actions); EX1006, 7:33-54 (describing available actions for an SDN controller). Thus, as discussed, Lefebvre's modified controller would redirect certain traffic to an application server for analysis (e.g., intrusion detection) based

on the DPI. See EX1006, 15:5-17; Min, ¶175. Once the application server completes the analysis (e.g., intrusion detection), the application server sends the packet to the modified Lefebvre controller, which sends the inspected packet flow back to the network device for transmission onward to the second entity, as Chua teaches. See EX1006, 21:35-42, 28:18-32 (describing programming network devices to “forward packets of the packet flow to a destination of the packet flow.”). This renders obvious the “sending the packet, after receiving from the controller by the network node, to the second entity” element of claim 11. Min, ¶175.

i) Claim 12

The Lefebvre-Chua combination renders obvious claim 12. Min, ¶¶176-179. As discussed for claim 9, a POSITA would have understood that the DPI performed by the modified Lefebvre controller involves “analyzing the packet” as recited in claim 9. Min, ¶177. A POSITA would have understood that DPI is one example of “applying security or data analytic application” as claimed. See, e.g., EX1006, 31:48-54 (describing using DPI to determine whether packets are malicious); Min, ¶177.

Additionally, Chua discloses that “[s]ervice devices 116 may include, for example, intrusion detection service (IDS) devices, intrusion prevention system (IPS) devices, web proxies, web servers, web-application firewalls and the like.” EX1006, 5:59-62. Other examples of services provided by service devices 116 include “denial of service (DoS) protection, distributed denial of service (DDoS)

protection, traffic filtering, wide area network (WAN) acceleration, or other such services” as well as “malware detection devices, network anti-virus devices, network packet capture and analysis devices, honeypot devices, reflector net devices, tar pit devices, domain name service (DNS) and global DNS server devices, mail proxies, and anti-spam devices.” EX1006, 5:62-6:5. Chua explains that service devices 116 are examples of “L4-L7 devices” (EX1006, 14:39-42) and further discloses that “SDN controller itself may host one or more L4-L7 services, rather than (or in addition to) services provided by service devices 116.” EX1006, 6:42-45. Such services, when performed by the SDN controller as taught by Chua, constitute further examples of “applying security or data analytic application” as claimed. Min, ¶179.

j) Claim 13

The Lefebvre-Chua combination renders obvious claim 13. Min, ¶¶180-182. As discussed for claim 12, Chua discloses that “SDN controller itself may host one or more L4-L7 services, rather than (or in addition to) services provided by service devices 116.” EX1006, 6:42-45. Those services include, for example, “intrusion detection service (IDS)” and “firewalls.” EX1006, 5:59-62; *see also* EX1006, 14:39-42 (service devices 116 are examples of “L4-L7 devices”). As discussed in Section IX.A.1, it would have been obvious to modify Lefebvre’s controller to provide such services. Min, ¶181. Such services as “intrusion detection service” and “firewalls,” when performed by the SDN controller, render obvious “wherein

the analyzing comprises applying security application that comprises firewall or intrusion detection functionality” as claimed. Min, ¶181.

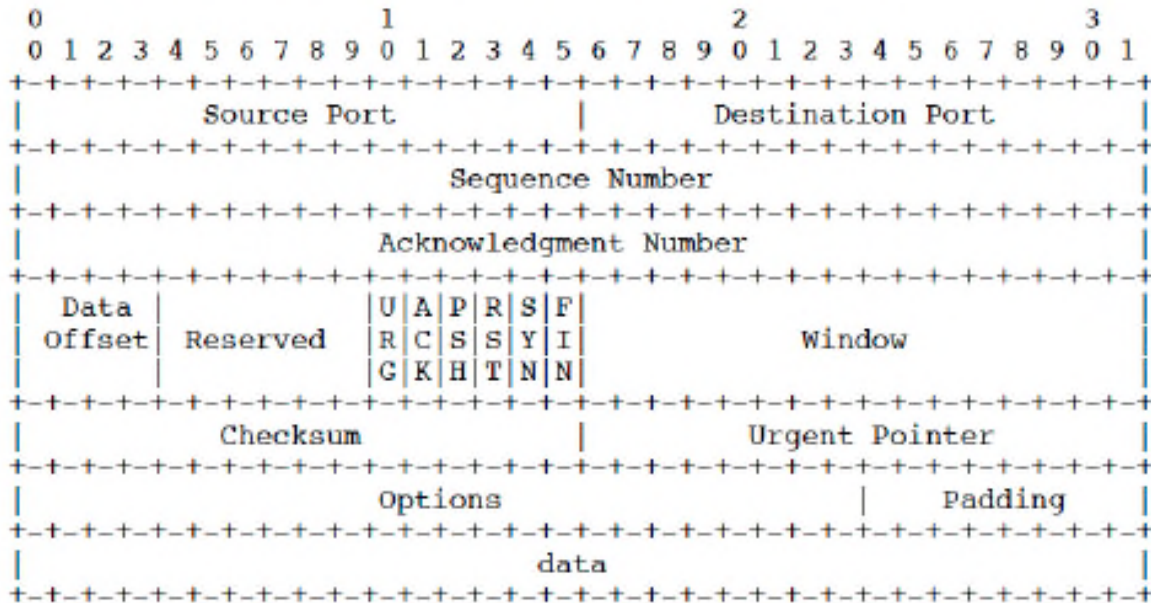
k) Claim 14

The Lefebvre-Chua combination renders obvious claim 14. Min, ¶¶183-185. As discussed for claims 5 and 9, in the combined system, Lefebvre’s modified controller would instruct network devices to redirect certain traffic to the controller, which would then execute DPI as taught by Chua. See EX1006, 31:49-54; Min, ¶184. Thus, the Lefebvre-Chua combination renders obvious “wherein the analyzing comprises performing Deep Packet Inspection (DPI) or using a DPI engine on the packet” as claimed. Min, ¶184.

l) Claim 15

The Lefebvre-Chua combination renders obvious claim 15. Min, ¶¶186-189. A POSITA would have understood that a packet comprises distinct header and payload fields. Min, ¶187. For example, a POSITA would have understood that Transmission Control Protocol (TCP) packets include separate and distinct header and payload fields. RFC 793, published September 1981, defines TCP. See EX1015; see also EX1008, 1 (“A packet is composed with header and payload, and the header carries layering information, such as the IP address is in the layer three and the port number is in the layer four. The payload carries upper layer application protocol and data.”). Figure 3 of RFC 793 below depicts the TCP header format.

TCP Header Format



TCP Header Format

Note that one tick mark represents one bit position.

Figure 3.

EX1015, 15.

As discussed for claim 14, in the combined system, Lefebvre's modified controller would instruct network devices to redirect certain traffic to the controller, which would then execute DPI as taught by Chua. See EX1006, 31:49-54; Min, ¶188. Well before the assumed April 22, 2014, priority date of the '111 Patent, a POSITA would have been aware that DPI analysis involves "checking part of, or whole of, the payload field" of a packet as claimed. Min, ¶188. A POSITA would have understood that DPI is a content-based technique that inspects both the header and the payload of a packet to classify network traffic. Min, ¶188; see also

EX1008, 2 (“DPI technique inspects both the header and the payload of a packet.”).

m) Claims 18 and 41

The Lefebvre-Chua combination renders obvious claims 18 and 41. Min, ¶¶190-193, 254-255. A POSITA would have understood that IP packets include separate and distinct header and payload fields. *See, e.g.*, EX1008, 1 (“A packet is composed with header and payload[.]”); EX1015, 15 (describing “Internet Protocol header”); Min, ¶191. And a POSITA would have understood that IP packets include source and destination addresses. *See* EX1015, 15 (“The Internet Protocol header carries several information fields, including the source and destination host addresses.”); EX1008, 1; Min, ¶191. For instance, Lefebvre discloses associating a service set based on the direction of the packet and the header fields of packets including the “source address, a destination address, a source port, a destination port and a protocol.” EX1005, 2:47-52, 7:59-67, 8:4-30. The source and destination addresses Lefebvre discloses matching against in the header fields are the claimed first and second entities’ addresses in the packet network, and the addresses Lefebvre uses to identify the matching packets are the claimed predetermined addresses. Min, ¶192.

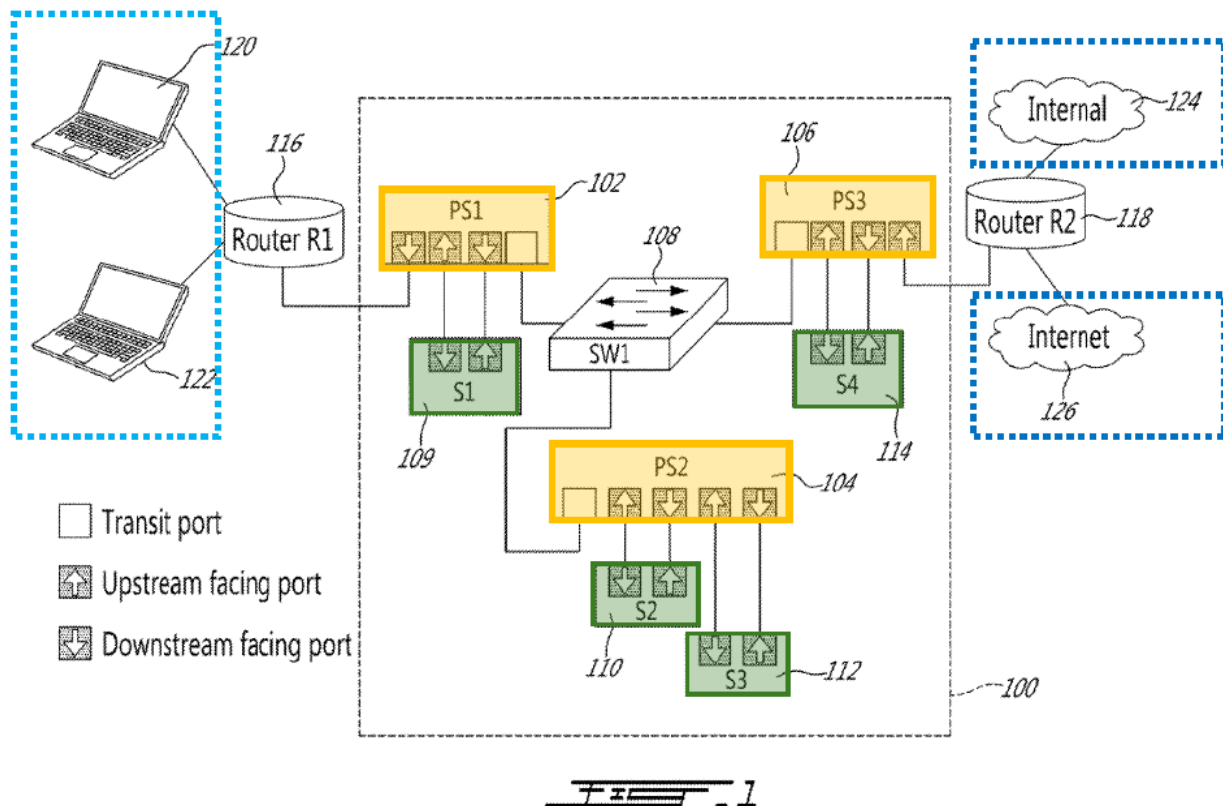
n) Claims 19 and 42

The Lefebvre-Chua combination renders obvious claims 19 and 42. Min, ¶¶194-196, 256-257. As discussed for claims 18 and 41, IP packets include source and destination addresses. Min, ¶195. A POSITA would have understood these

addresses are IP addresses. Min, ¶195.

o) Claims 21 and 44

The Lefebvre-Chua combination renders obvious claims 21 and 44. Min, ¶¶197-199, 258-259. As shown in annotated Figure 1 of Lefebvre below, Lefebvre discloses that the network includes **user equipment 120, 122** connected to router R1 and the **internal network 124** and the **Internet 126** connected to router R2. EX1005, 5:38-54, 6:13-16. A POSITA would therefore understand that Lefebvre's service network is at least a LAN if not a WAN. Min, ¶198. And Lefebvre's service network includes Internet 126. Min, ¶198.

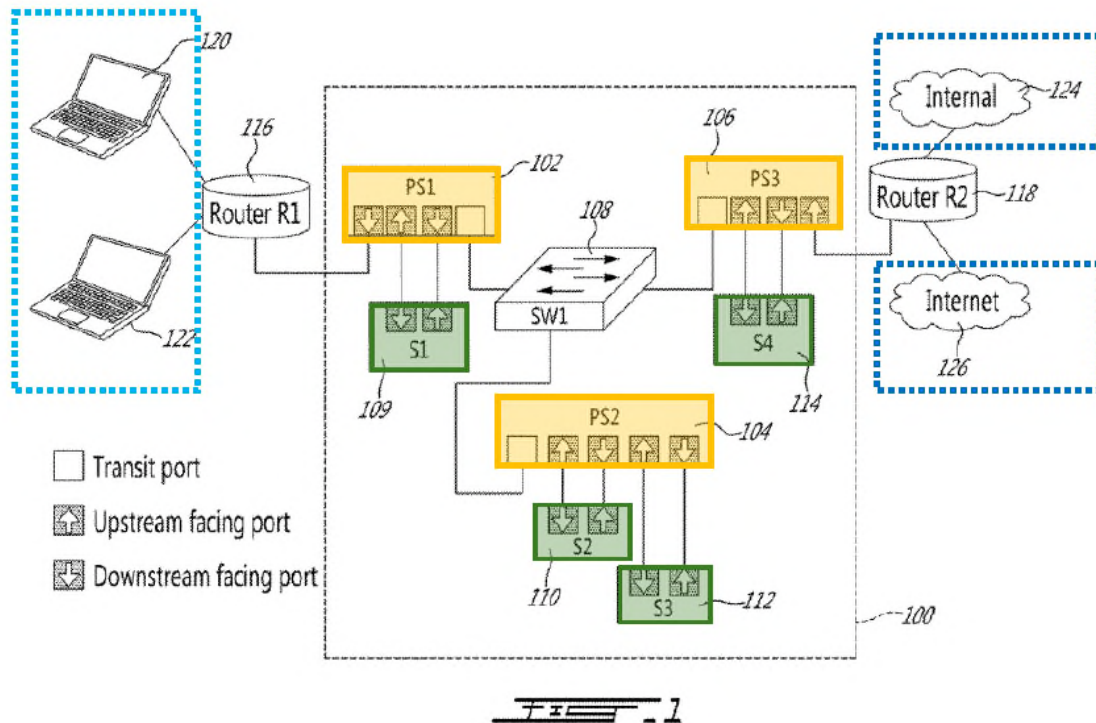


EX1005, FIG. 1 (annotated); Min, ¶198.

p) Claims 22 and 45

The Lefebvre-Chua combination renders obvious dependent claims 22 and 45. Min, ¶¶200-202, 260-261.

As shown in annotated Figure 1 of Lefebvre below, Lefebvre discloses that the network includes **user equipment 120, 122** connected to router R1 and the **internal network 124** and the **Internet 126** connected to router R2. EX1005, 5:38-54, 6:13-16. A POSITA would therefore understand that Lefebvre's **user equipment 120, 122** are client devices. Min, ¶201. And a POSITA would understand that the **internal network 124** and the **Internet 126** include servers. See, e.g., EX1009, 11 (describing the detection of web servers operating on non-standard ports on an internal network); EX1006, 5:51-53 (describing that an SDN “may provide services to network traffic flowing between client device 102 and server device 104”); Min, ¶201.



q) Claims 23 and 46

The Lefebvre-Chua combination renders obvious claims 23 and 46. Min, ¶¶203-205, 262-263. As discussed for claim 22, Lefebvre discloses a service network including **user equipment 120, 122** connected to router R1 and the **internal network 124** and the **Internet 126** connected to router R2. EX1005, 5:38-54, 6:13-16. And Lefebvre depicts **user equipment 120, 122** as laptops. EX1005, FIG. 1; Min, ¶204. A POSITA would therefore understand Lefebvre's **user equipment 120, 122** to teach “the client device comprises . . . a laptop computer” as claimed. Min, ¶204. And a POSITA would understand that the **internal network 124** and the **Internet 126** include servers, like web servers. Min, ¶204; EX1009, 11 (describing the detection of web servers operating on non-standard ports on an internal network).

r) Claims 24 and 47

The Lefebvre-Chua combination renders obvious claims 24 and 47. Min, ¶¶206-209, 264-265. Lefebvre discloses that its SDN uses the OpenFlow protocol. EX1005, 4:30-40. Chua confirms one way to implement an SDN is with OpenFlow. EX1006, 1:35-41; *see also* EX1013. The Open Networking Foundation (ONF) manages the OpenFlow protocol and related technologies. *See, e.g.*, EX1016, 2 (“ONF community standardized the OpenFlow protocol to enable separation of the network control and dataplanes, thus enabling multi-vendor SDN implementations.”); Min, ¶208. A POSITA would have therefore understood the OpenFlow protocol is a standard protocol. Min, ¶208.

s) Claims 25 and 48

The Lefebvre-Chua combination renders obvious dependent claims 25 and 48. Min, ¶¶210-213, 266-267. As discussed for claims 24 and 47, the ONF manages the OpenFlow protocol and related technologies. EX1016, 2; Min, ¶211. Lefebvre discloses that its SDN uses certain features from version 1.1 of the OpenFlow protocol. EX1005, 4:30-40.

The ONF published version 1.3.3 of the OpenFlow protocol in September 2013. EX1017, 3. The ONF published version 1.4.0 of the OpenFlow protocol on October 14, 2013. EX1017, 3; *see also* EX1018. To the extent that OpenFlow 1.1 is not compatible with OpenFlow protocol version 1.3.3 and 1.4.0, a POSITA would have found it obvious to modify Lefebvre’s SDN for compatibility, or to be

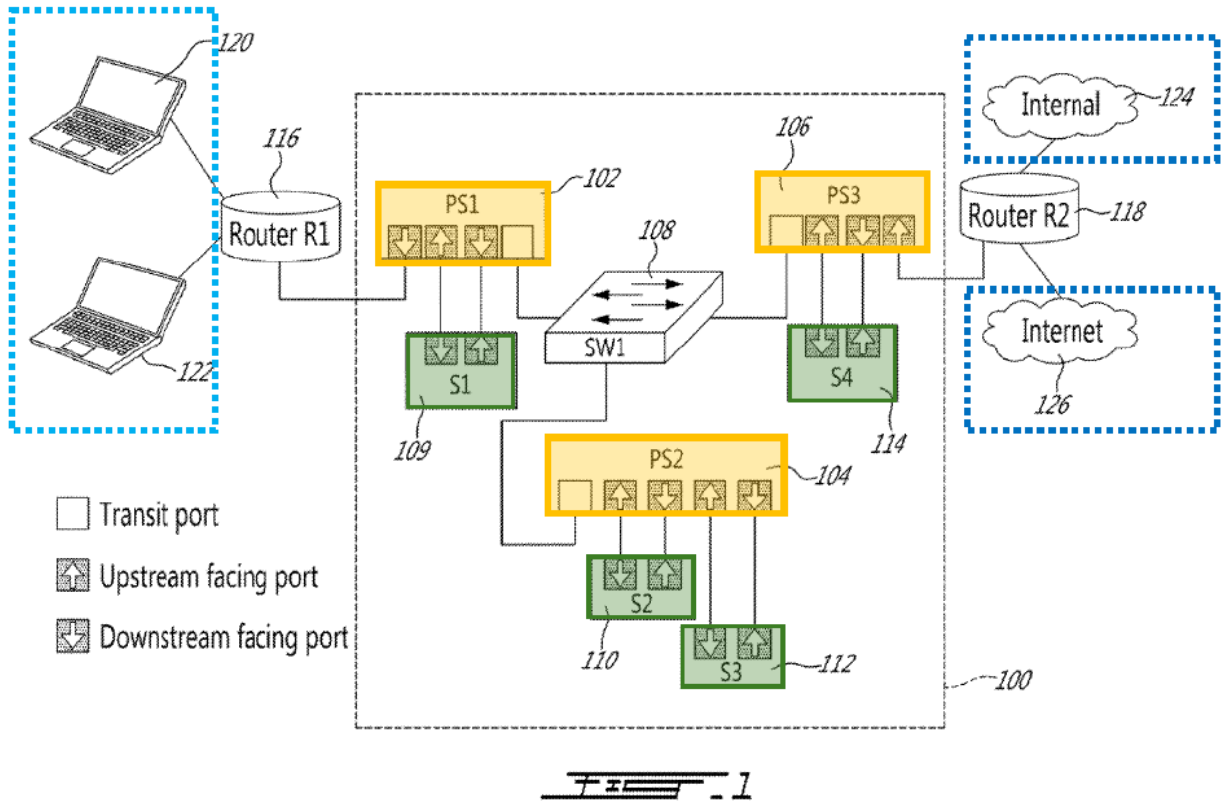
based on, OpenFlow protocol version 1.3.3 or 1.4.0. Min, ¶212.

t) Claims 26 and 49

The Lefebvre-Chua combination renders obvious claims 26 and 49. Min, ¶¶214-216, 268-269. Type-length-value (TLV) structures are well known data structures often used in protocols. Min, ¶215. For example, RFC 5810, published March 2010, discloses that the ForCES protocol makes extensive use of TLVs. EX1019. And OpenFlow protocol version 1.4.0, published October 14, 2013, makes use of TLVs. *See, e.g.*, EX1018, 52. A POSITA would have therefore found it obvious to modify Lefebvre's SDN to utilize instructions with a TLV structure. Min, ¶215.

u) Claims 27 and 50

The Lefebvre-Chua combination renders obvious claims 27 and 50. Min, ¶¶217-220, 270-271. Lefebvre discloses a service network as shown in annotated FIG. 1 below.



EX1005, FIG. 1 (annotated); Min, ¶218. Lefebvre discloses that **perimeter switches 102, 104, 106**, which correspond to the claimed network node, are OpenFlow switches. EX1005, 5:38-54; Min, ¶219. Lefebvre's controller programs **perimeter switches 102, 104, 106** to route traffic through the service chain. See, e.g., EX1005, 5:43-47.

v) Claims 28 and 51

The Lefebvre-Chua combination renders obvious claims 28 and 51. Min, ¶¶221-223, 272-273. Lefebvre's traffic steering process contemplates service policies defined by IP address blocks and IP addresses, for example. See, e.g., EX1005, 6:38-41, 6:46-48, 7:60-66; Min, ¶222. A POSITA would have therefore understood Lefebvre's packet network is an IP network. Min, ¶222.

w) Claims 29 and 52

The Lefebvre-Chua combination renders obvious claims 29 and 52. Min, ¶¶224-226, 274-275. Lefebvre’s traffic steering process contemplates service policies defined by TCP source and destination ports and TCP/UDP flows, for example. EX1005, 6:38-41, 7:60-66. A POSITA would have therefore understood Lefebvre’s packet network is a TCP network with TCP packets. Min, ¶225.

x) Claims 30 and 53

The Lefebvre-Chua combination renders obvious claims 30 and 53. Min, ¶¶227-229, 276-277. Lefebvre discloses steering packet traffic, which a POSITA would have understood discloses the required “receiving, by the network node from the first entity over the packet network, one or more additional packets.” EX1005, 2:30-38; Min, ¶228. As discussed for element 1[a], Lefebvre discloses flow-based policies. Min, ¶228. A POSITA would have understood this discloses “checking, by the network node, if any one of the one or more additional packets satisfies the criterion” because they apply to flows, which are one or more additional packets. EX1005, 6:46-53; Min, ¶228. As discussed for element 1[e], Lefebvre’s non-subscriber traffic does not satisfy the criterion of directing subscriber traffic so the packets are routed to the destination. See EX1005, 14:25-29. A POSITA would have understood this discloses “responsive to an additional packet not satisfying the criterion, sending by the network node over the packet network, the additional packet

to the second entity.” Min, ¶228. And Lefebvre discloses how multiple packets are “processed and steered” to the next node in their assigned service paths. EX1005, 6:5-12, 6:17-25. A POSITA would have therefore understood Lefebvre discloses “responsive to the additional packet satisfying the criterion, sending the additional packet, by the network node over the packet network, in response to the instruction.” Min, ¶228.

y) Claims 31 and 54

The Lefebvre-Chua combination renders obvious claims 31 and 54. Min, ¶¶230-232, 278-279. Lefebvre discloses that its service network is a software defined network (SDN). EX1005, 4:30-40. A POSITA would have understood that in an SDN, the packets are routed as part of the data plane and that communication with the controller serves as the control plane. *See, e.g.*, EX1011, 2; EX1006, 1:46-52; Min, ¶231.

B. Ground 2: Claims 7, 8, 37, and 38 are rendered obvious by Lefebvre in view of Chua and Rash.

A POSITA would have found claims 7, 8, 37, and 38 obvious based on Lefebvre in combination with Chua and Rash. Min, ¶¶280-282.

1. A POSITA Would Have and Could Have Combined Lefebvre and Chua with Rash

A POSITA would have been motivated and found it obvious to modify Lefebvre with Rash’s teachings. Min, ¶¶283-291. Lefebvre discloses techniques for an SDN to steer traffic through inline services. EX1005, 1:13-21, 2:39-47.

Lefebvre discloses how operators can define the traffic classes and service chains for those classes for processing traffic flows on a per packet basis. EX1005, 4:41-43. Lefebvre discloses these operator instructions are programmed into the switches as rules so that the network traffic passes through the ordered chain of services as desired. EX1005, 4:41-47; Min, ¶283.

Lefebvre identifies several types of services that could form part of the service chain, including DPI. EX1005, 1:28-38, 4:48-54. Lefebvre controls the usage of DPI services by routing network traffic through the service only as appropriate, where the services are separate from the controller. Min, ¶284.

Like Lefebvre, Rash discloses packet inspection systems. EX1007, 7:31-34. Rash discloses a packet mirroring function that “provides the capability to select options based on a plurality of criteria to mirror any portion or all of the frames of packets received on the network.” EX1007, 9:1-12, 27:31-43; *see also* EX1007, 9:12-15, 25:30-55, 26:11-35, 29:46-30:17; Min, ¶285. Rash discloses how the mirrored traffic may have a different destination based on certain criteria. EX1007, 35:66-36:2; Min, ¶285. And Rash discloses its dynamic mirroring functionality helps minimize the data mirrored. EX1007, 12:59-13:8, 34:43-51.

A POSITA would have been motivated to modify Lefebvre with Rash to minimize the amount of network capacity used to examine packets, which would help reduce network costs and improve overall efficiency. Min, ¶286. A POSITA would have been motivated and found it obvious to modify Lefebvre with

Rash's teachings because both describe configurable networks, and both describe packet inspection as one type of service available in these networks. EX1005, 1:28-38; EX1007, 9:12-15, 25:30-55, 26:11-35, 29:46-30:17; Min, ¶286. Lefebvre and Rash are therefore in the same field of endeavor and contain overlapping disclosures with similar purposes. Min, ¶286.

Modifying Lefebvre in view of Rash's teachings would have been obvious to a POSITA. Min, ¶287. Lefebvre has three different levels of policies which have different levels of granularity. EX1005, 6:32-48. Similarly, Rash explains that the network control engine uses packet fields including address, protocol, length or byte count, and other fields when making its mirroring determinations. EX1007, 29:53-58.

Although Lefebvre mentions DPI as one type of known service, it considers packets in their entirety and similarly forwards packets in their entirety. *See, e.g.*, EX1005, 6:5-31, 1:28-38 (discussing DPI), 7:42-8:65. Rash provides specific details regarding mirroring packets or just portions of packets. EX1007, 9:1-12, 27:31-43. Rash discloses this targeted approach to mirroring minimizes the data the application identification function must consider. EX1007, 12:59-13:8, 34:43-51. A POSITA would therefore look to Rash's detailed teachings regarding mirroring of packets and portions of packets and apply them to Lefebvre. Min, ¶288.

A POSITA would have been motivated to modify Lefebvre with Rash's

teachings for multiple reasons. Min, ¶289. Lefebvre discloses that services including DPI are typically expensive, dedicated appliances. EX1005, 1:25-27, 1:34-38. By implementing Rash’s mirroring of a portion of a packet instead of the entirety of a packet, a POSITA would recognize the reduction in the amount of network capacity utilized to examine the packets, thereby reducing network costs and improving overall efficiency. Min, ¶289. Rash’s dynamic mirroring function leverages existing capabilities in network devices, minimizing additional costs. EX1007, 30:29-44, 34:37-47; Min, ¶289. Rash’s targeted approach to mirroring also minimizes the data the application identification function must consider. EX1007, 12:59-13:8, 34:43-51; Min, ¶289.

Implementing Rash’s teachings in Lefebvre’s controller would have been well within the skill of a POSITA. Min, ¶290. Lefebvre already redirects entire packets to the appropriate service. EX1005, 8:49-65, Table 1, Table 2 (depicting OpenFlow switch table actions). Similarly, Rash discloses mirroring packets to the network infrastructure that analyzes the mirrored traffic. *See, e.g.*, EX1007, 7:51-57, 27:15-30 (disclosing the “mirrored traffic destination” includes any device “arranged to receive and either transfer or analyze the mirrored frames”). Modifying Lefebvre to forward only a portion of the packet as Rash teaches would have been well within the skill of a POSITA, as Rash confirms. Min, ¶290.

A POSITA would have had a reasonable expectation of success in applying Rash’s teachings to Lefebvre. Min, ¶291. Lefebvre already redirects the entirety

of packets, so a POSITA would need to modify Lefebvre to consider a portion of a packet. See EX1005, 8:49-65, Table 1, Table 2 (depicting OpenFlow switch table actions). Rash's technique allow for "any granularity of the frame of the network traffic" and Rash explains that "[o]ne ordinarily skilled in the art will recognize and appreciate that all or any portion of network traffic, such as all or any available portion of a frame, a packet, or any other protocol data unit, at one or more network protocol layers may be mirrored in practicing the operations of the present invention described herein." EX1007, 27:39-43; see also EX1007, 27:44-53. Because Rash confirms that mirroring portions of a protocol data unit are within a POSITA's knowledge and ordinary skill, a POSITA would have had a reasonable expectation of success in applying Rash's teachings to Lefebvre. Min, ¶291.

2. Dependent Claims 7, 8, 37, and 38

Dependent claims 7 and 8 have substantially the same scope as claims 37 and 38, respectively. Accordingly, the corresponding claims are addressed together in the sections that follow.

a) Claims 7 and 37

The Lefebvre-Chua combination renders obvious the method of claims 5 and 36 (from which Claims 7 and 37 depend) as set forth above. Min, ¶292. The Lefebvre-Chua-Rash combination renders obvious claims 7 and 37. Min, ¶¶292-296, 300-301.

As discussed for element 1[f], the Lefebvre-Chua combination renders

obvious the “responsive to the packet satisfying the criterion and to the instruction” aspect of claims 7 and 37. Min, ¶293. Lefebvre and Chua acknowledge the different aspects of a packet, including the header field. EX1005, 15:54-63, 16:67-17:2; EX1006, 25:57-61. But Lefebvre and Chua do not disclose sending a portion of the packet to the controller.

Rash discloses a packet inspection system that transmits portions of the packet to a destination other than the intended destination. EX1007, 7:51-59, 9:3-12, 25:30-55, 26:11-35, 27:31-43, 29:58-62, 35:66-36:2. Rash discloses a network control manager, controlling the mirroring function, which provides the ability to select options “to mirror any portion or all of the frames of packets.” EX1007, 9:1-15, 27:31-43, 29:7-12, 29:46-30:17.

As discussed in Section IX.B.1, a POSITA would have been motivated and found it obvious to apply Rash’s teachings to Lefebvre’s network. Min, ¶295. Rash discloses that this mirroring (the claimed instruction) sends the portion of packets to “the application identification engine” of the network. *See, e.g.*, EX1007, 7:51-57, 27:15-30 (disclosing the “mirrored traffic destination” includes any device “arranged to receive and either transfer or analyze the mirrored frames”). Thus, Lefebvre’s network, as modified by Rash, would mirror portions of packets to the controller which, as modified by Chua, would execute DPI. Min, ¶295.

b) Claims 8 and 38

The Lefebvre-Chua-Rash combination renders obvious claims 8 and 38.

Min, ¶¶297-299, 302-303. As discussed for claims 7 and 37, Lefebvre, Chua, and Rash disclose sending the packet to the controller. Further, Rash discloses that, “based on a plurality of criteria,” Rash’s mirroring function can “mirror ... the first L bytes.” EX1007, 9:3-12; *see also* EX1007, 29:7-12, 29:46-62. A POSITA would have understood this to mean that Rash teaches mirroring multiple consecutive bytes (e.g., “the first L bytes”), and that the mirroring (the claimed instruction) identifies the consecutive bytes for mirroring. Min, ¶298.

C. Ground 3: Claims 16-20 and 39-43 are rendered obvious by Lefebvre in view of Chua and RFC 793.

A POSITA would have found Claims 16-20 and 39-43 obvious based on Lefebvre in combination with Chua and RFC 793. Min, ¶304.

1. A POSITA Would Have and Could Have Combined Lefebvre and Chua with RFC 793

RFC 793 defines the well-known and widely used Transmission Control Protocol (TCP), which would have been part of the background knowledge of a POSITA in 2014. Min, ¶305. Both Lefebvre and Chua expressly contemplate use of their respective systems with TCP. *See, e.g.*, EX1005, 6:38-41, 7:60-67; EX1006, 18:12-15, 23:52-56. RFC 793 discloses details of TCP that would have been familiar to a POSITA. Min, ¶305. In the context of systems such as Lefebvre and Chua that utilize TCP, a POSITA would have been motivated to look to RFC 793 for implementation details to ensure compliance with the Transmission Control Protocol. Min, ¶305. Additional explanation regarding the rationale and

motivation for combining particular aspects of RFC 793 with Lefebvre and Chua is provided in the analysis that follows.

2. Dependent Claims 16-20 and 39-43

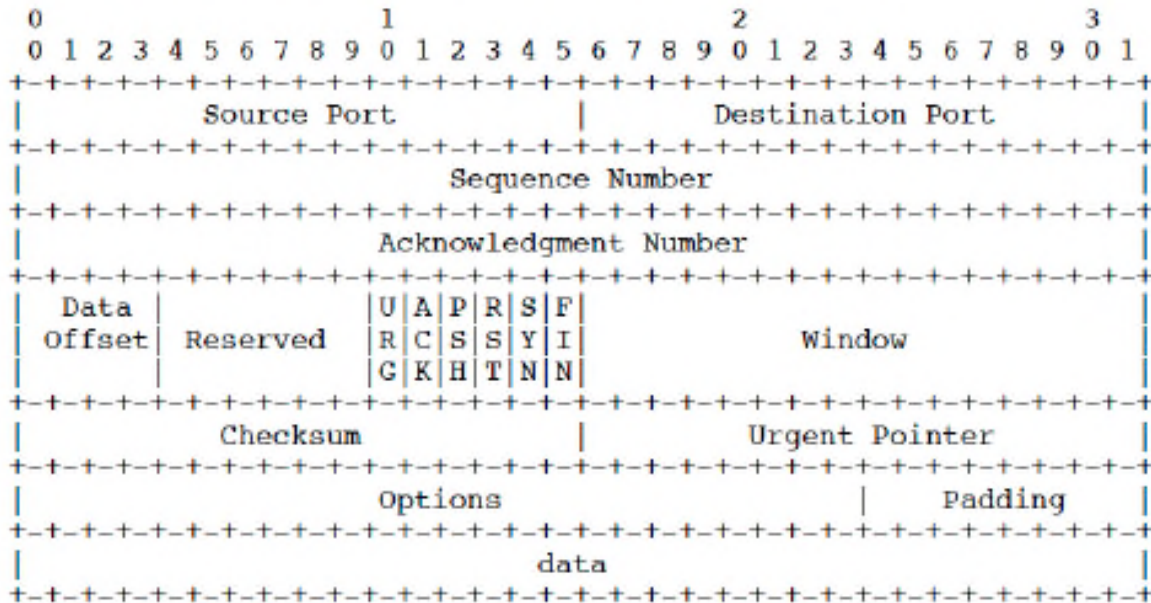
Dependent claims 16-20 have substantially the same scope as claims 39-43, respectively. Accordingly, the corresponding claims are addressed together in the sections that follow.

a) Claims 16 and 39

The Lefebvre-Chua combination renders obvious the method according to claims 1 and 32 (from which claims 16 and 39 depend, respectively) as discussed in Ground 1. Min, ¶306. The Lefebvre-Chua-RFC 793 combination renders obvious claims 16 and 39. Min, ¶¶306-311, 329-330.

A POSITA would have understood that TCP packets include separate and distinct header and payload fields. Min, ¶307. And a POSITA would have further understood that standard TCP packets include flag bits providing additional information related to the state of the connection. Min, ¶307. RFC 793, published September 1981, defines TCP. EX1015. Figure 3 of RFC 793 below depicts the TCP header format. EX1015, 15.

TCP Header Format



TCP Header Format

Note that one tick mark represents one bit position.

Figure 3.

EX1015, 15.

Each TCP header includes source and destination port fields, a sequence number field, along with acknowledgement number, control bits, window, checksum, urgent pointer, and options fields. EX1015, 15-19. And RFC 793 defines optional features that could exist in the options field including no-operation and maximum segment size. EX1015, 18-19. Since RFC 793, other options including selective acknowledgements (SACK) and timestamps have utilized the TCP header options fields. Min, ¶308.

Lefebvre discloses using policies that consider other standard TCP header fields when classifying traffic, including source/destination addresses,

source/destination ports, and protocol fields. *See, e.g.*, EX1005, 15:55-60 (describing “assigning an ordered list of services to be applied to the packet in accordance with at least one header field of the packet” where the field may include “a source address, a destination address, a source port, a destination port or a protocol”); *see also* EX1005, 2:44-64, 3:33-45, 16:1-3, 16:67-17:2, 16:58-61, claims 4, 5, 8, 9, 18, 21; Min, ¶309. Each TCP packet header includes the standard fields defined by RFC 793 such as source/destination ports and addresses, which Lefebvre already considers, and flag bits, which Lefebvre does not indicate it considers when classifying traffic. EX1015, 21; Min, ¶309. Chua similarly confirms the various header fields included with TCP packets including source and destination ports. EX1006, 26:30-36.

Both Lefebvre and Chua discuss several various header fields but do not specifically identify the flag bits of TCP packets. Min, ¶310. But, as RFC 793 confirms, flag bits are part of the TCP standard. It would have been obvious to a POSITA to modify Lefebvre to consider the flag bits. Min, ¶310. A POSITA would have been motivated to modify Lefebvre to consider the status of the flag bits that are included in each packet and provide additional information regarding the connection’s state. Min, ¶310. Modifying Lefebvre to consider these standard fields of each packet’s header would allow Lefebvre’s traffic steering to consider connection state when determining which service chain the packets should traverse. Min, ¶310.

Modifying Lefebvre to consider these flag bits would have been well within the skill of a POSITA. Min, ¶310. Lefebvre already considers other standard aspects of TCP packet headers, so modifying Lefebvre to consider the flag bits in addition to other aspects of the header field would have been well within the ordinary skill of a POSITA. See, e.g., Min, ¶310; EX1005, 15:55-60, 2:44-64, 3:33-45, 16:1-3, 16:67-17:2, 16:58-61, claims 4, 5, 8, 9, 18, 21. Indeed, Chua discloses how the controller programs network devices to perform actions including forwarding based on header data such as source and destination ports. EX1006, 26:30-36.

b) Claims 17 and 40

The Lefebvre-Chua-RFC 793 combination renders obvious claims 17 and 40. Min, ¶¶312-314, 331-332. As discussed for claim 16, a POSITA would have found it obvious to modify Lefebvre to consider TCP packet flag bits. Min, ¶313. As RFC 793 confirms, a POSITA would have also understood that TCP packet flag bits include the SYN, ACK, FIN, and RST flag bits, among others. Min, ¶313. These are some of the standard flag bits defined as part of the TCP standard. EX1015, 22 (describing the defined control bits including SYN, ACK, FIN, and RST).

c) Claims 18 and 41

The Lefebvre-Chua combination renders obvious claims 1 and 32 (from which claims 18 and 41 depend, respectively) as discussed in Ground 1. Min, ¶315. The Lefebvre-Chua-RFC 793 combination renders obvious claims 18 and 41. Min, ¶¶315-318, 333-334.

IP packets include separate and distinct header and payload fields, as well as source and destination addresses. *See, e.g.*, EX1008, 1; Min, ¶316; EX1015, 15. Lefebvre discloses associating a service set based on the direction of the packet and the header fields of packets including the “source address, a destination address, a source port, a destination port and a protocol.” EX1005, 2:47-52, 7:59-67, 8:4-30. The source and destination addresses Lefebvre discloses matching against in the header fields are the claimed first and second entities addresses in the packet network, and the addresses Lefebvre uses to identify the matching packets are the claimed predetermined addresses. Min, ¶317.

d) Claims 19 and 42

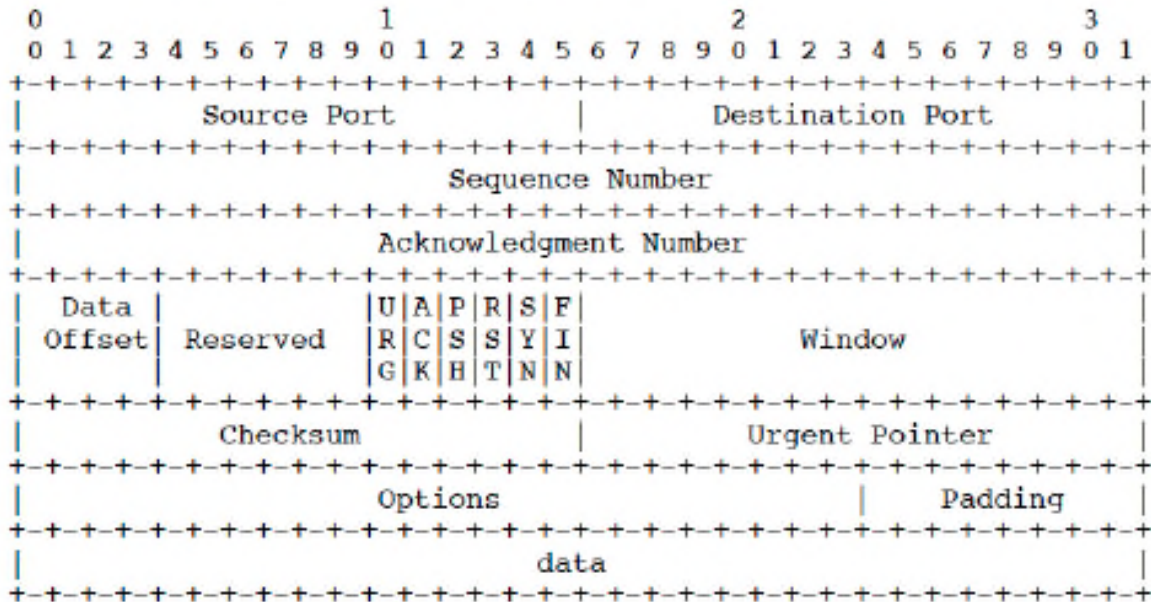
The Lefebvre-Chua-RFC 793 combination renders obvious claims 19 and 42. Min, ¶¶319-321, 335-336. As discussed, IP packets include source and destination addresses. EX1008, 1; EX1015, 15; EX1005, 2:47-52, 7:59-67, 8:4-30; Min, ¶320. A POSITA would have understood these addresses are IP addresses. Min, ¶320.

e) Claims 20 and 43

The Lefebvre-Chua combination renders obvious the method according to claims 1 and 32 (from which claims 20 and 43 depend, respectively) as discussed in Ground 1. Min, ¶322. The Lefebvre-Chua-RFC 793 combination renders obvious claims 20 and 43. Min, ¶¶322-328, 337-338.

As discussed, RFC 793, published September 1981, defines TCP. *See* EX1015. Figure 3 of RFC 793 depicts the TCP header format:

TCP Header Format



TCP Header Format

Note that one tick mark represents one bit position.

Figure 3.

EX1015, 15.

Each TCP header includes source and destination port fields, a sequence number field, along with acknowledgement number, control bits, window, checksum, urgent pointer, and options fields. EX1015, 15-19. And RFC 793 defines optional features that could exist in the options field including no-operation and maximum segment size. EX1015, 18-19; Min, ¶324.

RFC 793 does not disclose a “TCP sequence mask” field, as claims 20 and 43 recite. Min, ¶324. And nothing in the ’111 Patent explains what constitutes the

“TCP sequence mask” field. Min, ¶324. Rather, the ’111 Patent explains how to compute the TCP sequence mask. EX1001, 4:58-60, 6:41-7:40, 9:56-61; Min, ¶324. Regardless of what TCP header field corresponds to “TCP sequence mask,” the Lefebvre-Chua-RFC 793 combination renders claims 20 and 43 obvious. Min, ¶325.

This is because a POSITA would have understood that TCP packets (more typically referred to as TCP segments) include each of the fields required by RFC 793, including source and destination ports, and TCP sequence numbers. Min, ¶325. Lefebvre discloses using the source and destination port and address fields along with the protocol fields, but does not specifically identify the TCP sequence number. *See, e.g.*, EX1005, 15:55-60 (describing “assigning an ordered list of services to be applied to the packet in accordance with at least one header field of the packet” where the field may include “a source address, a destination address, a source port, a destination port or a protocol”).

Regardless, because the sequence number is part of the TCP standard, defined by RFC 793, as are the fields Lefebvre already considers, it would have been obvious to a POSITA to modify Lefebvre to consider the source and destination addresses, the TCP sequence number, and any other fields necessary to constitute the claimed “TCP sequence mask” field. Min, ¶326.

A POSITA would have been motivated to modify Lefebvre to consider these additional standard fields which provide additional information regarding the

session including what portion of the data stream the TCP segment constitutes. Min, ¶327. Lefebvre's traffic steering already considers other standard TCP header fields when classifying traffic, including source and destination addresses, source and destination ports, and protocol fields. *See, e.g.*, EX1005, 15:55-60, 2:44-64, 3:33-45, 16:1-3, 16:67-17:2, 16:58-61, claims 4, 5, 8, 9, 18, 21; Min, ¶327. Each TCP packet header includes the standard fields defined by RFC 793 such as source and destination ports and addresses, which Lefebvre already considers, and sequence numbers, which Lefebvre does not. Min, ¶327. Modifying Lefebvre to consider additional standard fields would allow Lefebvre's traffic steering to consider the details provided by these additional standard fields when classifying traffic and determining which service chain the packets should traverse. Min, ¶327.

Modifying Lefebvre to consider these additional standard fields would have been well within the skill of a POSITA. Min, ¶327. Lefebvre already considers other standard aspects of TCP headers, so parsing and considering additional standard fields would have been well within the ordinary skill of a POSITA. Min, ¶327. Chua confirms this when disclosing how the controller programs network devices to perform actions including forwarding based on standard TCP header data such as source and destination ports. EX1006, 16:27-30; Min, ¶327.

X. SECONDARY CONSIDERATIONS

Petitioner is aware of no evidence of secondary considerations that would meaningfully rebut a finding of obviousness. Min, ¶¶339-340. Petitioner

reserves the right to rebut any purported objective evidence of non-obviousness raised by PO.

XI. INSTITUTION SHOULD BE GRANTED

A. 35 U.S.C. §314(a)

The Board should not exercise its discretion under §314(a) to deny institution. *First*, the *General Plastics* factors favor institution. See *General Plastic Indus. Co., Ltd. v. Canon Kabushiki Kaisha*, IPR2016-01357, Paper 19, 15-16 (P.T.A.B. Sept. 6, 2017) (precedential). This Petition is the first and only petition for IPR of the '111 Patent filed by Arista. Although three other petitions for IPR of the '111 Patent have been filed by different petitioners, that does not warrant denying institution in this case. Specifically, claims 1-9, 12-24, and 27-31 of the '111 Patent (a subset of the claims at issue here) were also challenged in a petition filed by Cisco Systems, Inc. ("Cisco") on February 21, 2023, in IPR2023-00554 (the "Cisco IPR"). See *Cisco Systems, Inc. v. Orckit IP, LLC*, IPR2023-00554, Paper 1 (P.T.A.B. Feb. 21, 2023). The Board instituted review in the Cisco IPR on September 20, 2023, and shortly thereafter a second, substantively identical petition for IPR of the '111 Patent was filed by petitioner Juniper Networks, Inc. ("Juniper") in IPR2024-00037 along with a motion for joinder with the Cisco IPR. See *Juniper Networks, Inc. v. Orckit Corp.*, IPR2024-00037, Paper Nos. 2, 3 (P.T.A.B. Oct. 11, 2023). The Board joined IPR2024-00037 with the Cisco IPR, and those proceedings are ongoing. On May 6, 2024, a third petition challenging claims 32-

54 of the '111 Patent was filed by Juniper in IPR2024-00895 and is still in the preliminary stages. *See Juniper Networks, Inc. v. Orckit Corp.*, IPR2024-00895, Paper No. 2 (P.T.A.B. May 6, 2024).

Because this Petition is the first and only one filed by Arista, and Arista has no significant relationship with Cisco or Juniper, discretionary denial under 314(a) and *General Plastics* is unwarranted. *See Prime Time Toys LLC v. Spin Master, Inc.*, IPR2023-01339, Paper 12, 20-21 (Vidal July 9, 2024) (“Accordingly, where ‘the first and second petitioners are neither the same party, nor possess a significant relationship under *Valve*, *General Plastics* factor one necessarily outweighs the other *General Plastics* factors’ and ‘exercising discretion to deny the Petition is not justified.’”). Although Cisco and Juniper have also been sued for alleged infringement of the '111 Patent, Arista is not a codefendant in district court litigation with Cisco or Juniper, the accused products are not the same, and Cisco and Juniper are competitors of Arista. At most, Arista, Cisco, and Juniper have been accused of infringing the same patent, but “the infringement allegations do not provide a basis for finding that they have a ‘significant relationship.’” *Ford Motor Co. v. Neo Wireless LLC*, IPR2023-00763, Paper 28, 10 (Vidal Mar. 22, 2024).

Second, if the Board applies the analysis in *NHK Spring Co. v. Intri-Plex Techs., Inc.*, IPR2018-00752, Paper 8, 19-20 (P.T.A.B. Sept. 12, 2018) (precedential) or *Apple Inc. v. Fintiv, Inc.*, IPR2020-00019, Paper 11 (P.T.A.B. Mar. 20, 2020) (precedential), those factors taken together support institution.

Factor 1: Potential Stay

The Delaware Litigation was stayed on December 1, 2023 pending final written decisions in the Cisco IPR and two other IPRs also filed by Cisco challenging other asserted patents. EX1023, 1-2. The current stay, and the district court’s demonstrated willingness to stay the litigation pending resolution of parallel PTAB proceedings, strongly weighs against discretionary denial. *See Apple*, IPR2020-00019, Paper 11, 6-7.

Factor 2: Trial Date

The Delaware Litigation was stayed before Arista responded to Patent Owner’s complaint and no trial date has been set. Even if the stay was lifted upon issuance of a FWD in the Cisco IPR (expected by September 20, 2024), the latest statistics show a median time to trial for the District of Delaware as 38 months. EX1021, 14. As such, any trial would not occur until well after a final written decision is reached in this proceeding. Thus, this factor “weighs heavily against exercising discretion to deny institution.” *See, e.g., Microsoft Corp. v. Interdigital Patent Holdings, Inc.*, IPR2024-00537, Paper No. 10, 9 (P.T.A.B. July 15, 2024) (“Because the district court has not set a trial date, and the trial court statistics indicate any trial is likely to occur months after a Final Written Decision in this proceeding, we see little potential for conflicting Board and district court decisions.”).

Factor 3: Investment in the parallel proceeding

The Delaware Litigation was stayed before Arista responded to Patent Owner's complaint. Neither the parties nor the court have expended substantial effort in the parallel proceeding and, given the stay, it would be premature to speculate as to the amount and type of work that will have been completed in the Delaware Litigation when the institution decision is made. Accordingly, this factor weighs against discretionary denial.

Factor 4: Issue Overlap

The Delaware Litigation has barely begun. Any overlap between these two proceedings at this point would be completely speculative. As such, this factor fails to support discretionary denial.

Factor 5: Party Overlap

Petitioner and Patent Owner are parties in the Delaware Litigation.

Factor 6: Other Circumstances Favoring Institution

The merits favor institution as the strong grounds in this Petition demonstrate. "In such cases, the institution of a trial may serve the interest of overall system efficiency and integrity because it allows the proceeding to continue in the event that the parallel proceeding settles or fails to resolve the patentability question presented in the PTAB proceeding." *Google LLC v. Parus Holdings, Inc.*, IPR2020-00846, Paper 9 at 21 (P.T.A.B. Oct. 21, 2020).

Considered as part of a holistic analysis, the *Fintiv* factors favor institution.

B. 35 U.S.C. §325(d)

The references cited by Petitioner were not presented to the Office during prosecution. Thus, the grounds in this Petition based on Lefebvre, Chua, Rash, and RFC 793 were not considered by the Examiner during prosecution.

Furthermore, although Lefebvre, Chua, and Rash are used in the grounds presented in IPR2024-00895 filed by Juniper, that does not warrant denying institution. Arista was sued separately by Patent Owner and is not a party to the litigations involving Cisco and Juniper. Arista has no significant relationship with Cisco or Juniper. Indeed, Arista is a competitor of Cisco and Juniper. Arista is entitled to preserve and present its challenge unless and until the arguments are finally resolved against Patent Owner. *Cf. Blonder-Tongue Labs, Inc. v. University of Ill.*, 402 U.S. 313, 329 (1971) (“litigants...who never appeared in a prior action...may not be collaterally estopped without litigating the issue.”). Furthermore, this Petition also challenges claims 1-31, which are not challenged in IPR2024-00895 filed by Juniper. Accordingly, the Board should decline to exercise its discretion under 35 U.S.C. §325(d).

XII. CONCLUSION

Petitioner respectfully requests institution of IPR and that the Challenged Claims be cancelled as unpatentable pursuant to 35 U.S.C. §318(b).

Petition for *Inter Partes* Review of U.S. Patent No. 10,652,111

Respectfully Submitted,

August 2, 2024

/Eliot D. Williams/

Eliot D. Williams (Reg. No. 50,822)

BAKER BOTTS L.L.P.

700 K Street, N.W

Washington, D.C. 20001-5692

Tel: 202-639-1334

eliot.williams@bakerbotts.com

*Attorneys for Petitioner, Arista Networks,
Inc.*

CERTIFICATE OF SERVICE

In accordance with 37 C.F.R. §§ 42.6(e) and 42.105, the undersigned certifies that on August 2, 2024, a complete and entire copy of the **PETITION FOR *INTER PARTES* REVIEW OF CLAIMS 1-54 OF U.S. PATENT NO. 10,652,111** including exhibits and testimony relied upon were served on Patent Owner via Priority Express Mail at the correspondence address of record for the subject patent and counsel for Patent Owner in the related litigation, as included below:

May Patents Ltd.
c/o Dorit Shem-Tov
P.O.B 7230
Ramat-Gan, 5217102
ISRAEL

Jacob R. Kirkham
KOBRE & KIM LLP
600 North King Street, Suite 501
Wilmington, DE 19801

Date: August 2, 2024

/Eliot D. Williams/
Eliot D. Williams (Reg. No. 50,822)
BAKER BOTTS L.L.P.
700 K Street, N.W
Washington, D.C. 20001-5692
Tel: 202-639-1334
eliot.williams@bakerbotts.com

*Attorneys for Petitioner, Arista Networks,
Inc.*

CERTIFICATION UNDER 37 C.F.R. § 42.24(d)

Pursuant to 37 C.F.R. § 42.24(d), the undersigned hereby certifies that the word count under § 42.24(a)(1) for the foregoing Petition for *Inter Partes* Review totals 13,898 words, within the 14,000 word limit allowed under § 42.24(a)(1)(i).

Date: August 2, 2024

/Eliot D. Williams/
Eliot D. Williams (Reg. No. 50,822)
BAKER BOTTS L.L.P.
700 K Street, N.W
Washington, D.C. 20001-5692
Tel: 202-639-1334
eliot.williams@bakerbotts.com

*Attorneys for Petitioner, Arista Networks,
Inc.*