

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

CISCO SYSTEMS, INC.,
Petitioner

IPR2020-01679
U.S. Patent No. 8,451,845

**PETITION FOR *INTER PARTES* REVIEW
UNDER 35 U.S.C. § 312 AND 37 C.F.R. § 42.104**

I. INTRODUCTION.....	9
II. MANDATORY NOTICES UNDER 37 C.F.R. § 42.8	9
A. Real Party-in-Interest.....	9
B. Related Matters.....	9
C. Lead and Back-up Counsel and Service Information	9
III. GROUNDS FOR STANDING	10
IV. NOTE	10
V. SUMMARY OF THE '845 PATENT	10
VI. PROSECUTION HISTORY	14
VII. LEVEL OF ORDINARY SKILL IN THE ART	15
VIII. CLAIM CONSTRUCTION.....	16
A. “domain” (Claims 1, 5, 7, and 8).....	16
B. “regularizing at least one of the IPv6 source address and the IPv6 destination address” (claims 1 and 8).....	18
C. “Means for” Limitations.....	19
1. “means for identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address, and a destination port number;”	20
2. “means for regularizing, if necessary, at least one of the IPv6 source address and the IPv6 destination address of the data packet by replacing said at least one IPv6 address,... [and] modifying the data packet using the regularized address;”	21
3. “means for routing the IPv6 data packet to its destination.”	23
IX. RELIEF REQUESTED AND THE REASONS FOR THE REQUESTED	

RELIEF	24
X. IDENTIFICATION OF HOW THE CLAIMS ARE UNPATENTABLE	24
A. Challenged Claims and Statutory Grounds for Challenge	24
B. Discretionary denial under 35 U.S.C. § 325(d) is not appropriate.....	26
C. Discretionary denial under the <i>Fintiv</i> factors is not appropriate.....	26
1. Whether evidence exists that a stay may be granted if a proceeding is instituted	27
2. Proximity of the Court’s trial date to the Board’s projected statutory deadline for a final written decision.....	28
3. Investment in the parallel proceeding by the Court and the parties	29
4. Overlap between issues raised in the petition and in the parallel proceeding	29
5. Whether the petitioner and the defendant in the parallel proceeding are the same party	29
6. Other circumstances that impact the Board’s exercise of discretion, including the merits	30
D. Discretionary denial under <i>General Plastic</i> is not appropriate.....	30
E. Ground 1: Claims 1 and 5-8 are obvious over Ananda, Wetterwald, and RFC4380.....	31
1. Ananda	31
2. Wetterwald	34
3. RFC4380	35
4. Reasons to combine Ananda, Wetterwald, and RFC4380.....	36

5. Claim 1 36

6. Claim 5 77

7. Claim 6 83

8. Claim 7 83

9. Claim 8 85

XI. CONCLUSION 94

CERTIFICATE OF WORD COUNT 95

CERTIFICATE OF SERVICE 96

PETITIONER’S EXHIBIT LIST

Ex.1001	U.S. Patent No. 8,451,845 to Mohamed Boucadair et al. (“’845 patent”)
Ex.1002	Prosecution History of U.S. Patent No. 8,451,845 (“’845 PH”)
Ex.1003	Declaration of Dr. Paul Min under 37 C.F.R. § 1.68
Ex.1004	<i>Curriculum Vitae</i> of Dr. Min
Ex.1005	U.S. Patent Application Publication No. 2004/0107287 to Ananda et al. (“Ananda”)
Ex.1006	Definition of “Domain,” Newton, Harry, “Newton’s Telecom Dictionary,” Sixteenth Expanded Edition, July 2000 (“Newton”)
Ex.1007	Y. Rekhter et al., “Address Allocation for Private Internets,” Network Working Group, Request for Comments: 1918 (Feb. 1996), available at https://tools.ietf.org/html/rfc1918 (“RFC 1918”)
Ex.1008	U.S. Patent No. 7,639,686 to Wetterwald et al. (“Wetterwald”)
Ex.1009	C. Huitema, “Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs),” Internet Engineering Task Force Network Working Group, Request for Comments: 4380 (February 2006), available at https://tools.ietf.org/html/rfc4380 (“RFC4380” or “Teredo”)
Ex.1010	U.S. Patent Application Publication No. 2005/0286553 to Wetterwald et al. (“’553 Publication”)
Ex.1011	<i>Reserved</i>
Ex.1012	“Internet Protocol – DARPA Internet Program Protocol Specification,” IETF Network Working Group, Request for Comments: 791 (J. Postel ed., Sep. 1981), available at https://tools.ietf.org/html/rfc791 (“RFC 791”)
Ex.1013	R. Hinden et al., “IP Version 6 Addressing Architecture,” Network Working Group Request for Comments: 1884 (Dec. 1995), available at https://tools.ietf.org/html/rfc1884 (“RFC 1884”)
Ex.1014	O. Jacobsen et al., “A Glossary of Networking Terms,” Network Working Group, Request for Comments: 1208 (Mar. 1991), available at https://tools.ietf.org/html/rfc1208 (“RFC 1208”)

Ex.1015	“Internet Users’ Glossary,” Network Working Group Request for Comments: 1983 (G. Malkin ed., Aug. 1996), available at https://tools.ietf.org/html/rfc1983 (“RFC 1983”)
Ex.1016	U.S. Patent Application Publication No. 2006/0212572 to Afek et al.
Ex.1017	European Patent No. EP1295454 to Soppera
Ex.1018	U.S. Patent No. 6,792,546 to Shanklin et al.
Ex.1019	Monarch Networking Solution LLC’s and Defendant Cisco Systems, Inc.’s Joint Claim Constructions and Prehearing Statement (P.R. 4-3), <i>Monarch Networking Sols. LLC v. Cisco Sys., Inc. et al.</i> , 2:20-cv-00015 (E.D. Tex. filed September 4, 2020)
Ex.1020	Plaintiff’s Disclosure of Asserted Claims and Infringement Contentions Pursuant to P.R. 3-1, <i>Monarch Networking Sols. LLC v. Cisco Sys., Inc. et al.</i> , 2:20-cv-00015 (E.D. Tex. served May 4, 2020)
Ex.1021	A. Conta et al., “Generic Packet Tunneling in IPv6 Specification,” Network Working Group Request for Comments: 2473 (Dec. 1998), available at https://tools.ietf.org/html/rfc2473 (“RFC 2473”)
Ex.1022	R. Hinden et al., “IP Version 6 Addressing Architecture,” Network Working Group Request for Comments: 2373 (July 1998), available at https://tools.ietf.org/html/rfc2373 (“RFC 2373”)
Ex.1023	R. Hinden et al., “Internet Protocol Version 6 (IPv6) Addressing Architecture,” Request for Comments: 3513 (Apr. 2003), available at https://tools.ietf.org/html/rfc3513 (“RFC 3513”)
Ex.1024	R. Hinden et al., “IP Version 6 Addressing Architecture,” Network Working Group, Request for Comments: 4291 (Feb. 2006), available at https://tools.ietf.org/html/rfc4291 (“RFC 4291”)
Ex.1025	U.S. Patent Application Publication No. 2006/0227792 to Wetterwald et al.
Ex.1026	Amended Docket Control Order, <i>Snik LLC v. Samsung Elecs. Co., Ltd.</i> , No. 2:19-cv-387-JRG, ECF No. 52 (E.D. Tex.) (Gilstrap)
Ex.1027	Docket Control Order, <i>Trioptima AB v. Quantile Techs. Ltd.</i> , No. 2:19-cv-390-JRG, ECF No. 41 (E.D. Tex.) (Gilstrap)
Ex.1028	Second Amended Docket Control Order, <i>GREE, Inc. v. Supercell Oy</i> , No. 2:19-cv-00413-JRG, ECF No. 61 (E.D. Tex.) (Gilstrap)
Ex.1029	Docket Control Order, <i>SpaceTime3D, Inc. v. Samsung Elecs. Co., Ltd.</i> , No. 2:19-cv-00372-JRG, ECF No. 26 (E.D. Tex.) (Gilstrap)

Ex.1030	Third Amended Docket Control Order, <i>Blitzsafe Texas, LLC v. PACCAR Inc.</i> , No. 2:19-cv-00393-JRG-RSP, ECF No. 48 (E.D. Tex.) (Payne)
Ex.1031	Docket Control Order, <i>Blitzsafe Texas, LLC v. Navistar, Inc.</i> , No. 2:19-cv-00403-JRG, ECF No. 29 (E.D. Tex.) (Gilstrap)
Ex.1032	Docket Control Order, <i>Luminati Networks Ltd. v. TESO LT, UAB, et al.</i> , No. 2:19-cv-0395-JRG, ECF No. 65 (E.D. Tex.) (Gilstrap)
Ex.1033	Docket Control Order, <i>Gloria Toussaint, et al. v. Michael Smalling, et al.</i> , No. 2:20-cv-0013-JRG-RSP, ECF No. 8 (E.D. Tex.) (Payne)
Ex.1034	<i>Reserved</i>
Ex.1035	<i>Reserved</i>
Ex.1036	R. Droms, “Dynamic Host Configuration Protocol,” Network Working Group, Request for Comments: 2131 (March 1997), available at https://tools.ietf.org/html/rfc2131 (“RFC 2131”)
Ex.1037	<i>Reserved</i>
Ex.1038	<i>Reserved</i>
Ex.1039	<i>Reserved</i>
Ex.1040	Docket Control Order, <i>Monarch Network Sols. LLC v. Cisco Sys., Inc.</i> , No. 2:20-cv-0015-JRG, ECF. No. 51 (E.D. Tex.) (Gilstrap)
Ex.1041	RFC Editor Webpage, www.rfc-editor.org , as captured by the Internet Archive Wayback Machine on Apr. 30, 2008, available at https://web.archive.org/web/20080430211008/http://www.rfc-editor.org/
Ex.1042	S. Bradner, “The Internet Standards Process – Revision 3,” Request for Comments: 2026 (Oct. 1996), available at https://tools.ietf.org/html/rfc2026 (“RFC 2026”).
Ex.1043	Declaration of Dr. Sylvia Hall-Ellis under 37 C.F.R. § 1.68
Ex.1044	<i>Curriculum Vitae</i> of Dr. Hall-Ellis
Ex.1045	<i>Reserved</i>
Ex.1046	<i>Reserved</i>
Ex.1047	U.S. Patent No. 7,403,995 to Mace et al.
Ex.1048	U.S. Patent No. 7,277,453 to Chin et al.
Ex.1049	U.S. Patent No. 7,764,691 to Sehagal et al.
Ex.1050	U.S. Patent No. 7,720,976 to Chu et al.
Ex.1051	U.S. Patent No. 7,729,347 to Rosberg
Ex.1052	U.S. Patent No. 7,133,405 to Graham et al.

Ex.1053	U.S. Patent No. 6,917,618 to Thubert et al.
Ex.1054	Canadian Patent Application Publication No. CA2446579A1
Ex.1055	PCT Publication No. WO 01/27766
Ex.1056	U.S. Patent No. 7,995,523 to Chen
Ex.1057	U.S. Patent No. 6,957,346 to Kivinen et al.
Ex.1058	U.S. Patent No. 7,080,151 to Borella et al.
Ex.1059	U.S. Patent No. 7,095,738 to Desanti
Ex.1060	U.S. Patent No. 8,045,558 to Droms et al.
Ex.1061	U.S. Patent No. 7,729,312 to Suzuki et al.
Ex.1062	U.S. Patent No. 7,961,725 to Nagarajan et al.
Ex.1063	U.S. Patent No. 7,813,843 to Patel et al.
Ex.1064	U.S. Patent No. 8,688,834 to Fujimoto et al.
Ex.1065	U.S. Patent No. 7,568,040 to Townsley et al.
Ex.1066	U.S. Patent No. 7,738,882 to Dutta et al.
Ex.1067	U.S. Patent No. 7,543,065 to Barkley et al.
Ex.1068	U.S. Patent No. 7,640,338 to Huitema et al.
Ex.1069	U.S. Patent No. 7,502,474 to Kaniz et al.
Ex.1070	U.S. Patent No. 8,245,304 to Chen et al.
Ex.1071	U.S. Patent No. 7,804,848 to Mirtorabi et al.

I. INTRODUCTION

Cisco Systems, Inc. (“Petitioner”) respectfully requests that the Board review and cancel as unpatentable under 35 U.S.C. § 103(a) claims 1 and 5-8 (the “Challenged Claims”) of U.S. Patent No. 8,451,845 (the “’845 patent,” Ex.1001).¹

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

A. Real Party-in-Interest

Under 37 C.F.R. § 42.8(b)(1), Petitioner certifies that the real party-in-interest is Cisco Systems, Inc.

B. Related Matters

As of the filing date of this Petition, and to the best knowledge of Petitioner, the ’845 patent is or has been involved in the following case:

Case Caption	Filed	Number	Court
<i>Monarch Networking Solutions LLC v. Cisco Systems, Inc. and Charter Communications, Inc.</i>	Jan. 21, 2020	2:20-cv-00015	E.D. Tex.

C. Lead and Back-up Counsel and Service Information

<u>Lead Counsel</u> Theodore M. Foster HAYNES AND BOONE, LLP 2323 Victory Ave. Suite 700 Dallas, TX 75219	Phone: (972) 739-8649 Fax: (214) 200-0853 ipr.theo.foster@haynesboone.com USPTO Reg. No. 57,456
---	--

¹ The pre-AIA statutory framework applies to the ’845 patent.

<u>Backup Counsel</u> David L. McCombs HAYNES AND BOONE, LLP 2323 Victory Ave. Suite 700 Dallas, TX 75219 Jonathan R. Bowser HAYNES AND BOONE, LLP 2323 Victory Ave. Suite 700 Dallas, TX 75219	Phone: (214) 651-5533 Fax: (214) 200-0853 david.mccombs.ipr@haynesboone.com USPTO Reg. No. 32,271 Phone: (202) 654-4503 Fax: (214) 200-0853 jon.bowser.ipr@haynesboone.com USPTO Reg. No. 54,574
---	---

Please address all correspondence in this proceeding to lead and back-up counsel. Petitioner consents to service in this proceeding by email at the addresses above.

III. GROUNDS FOR STANDING

Petitioner certifies that the '845 patent is available for IPR and that Petitioner is not barred or estopped from requesting IPR challenging the patent claims.

IV. NOTE

Petitioner cites to exhibits' original page numbers, unless otherwise noted. Emphasis in quoted material has been added throughout, unless otherwise noted.

V. SUMMARY OF THE '845 PATENT

The '845 patent is directed to an Internet Protocol (IP) telecommunications network for transmitting data packets from a source terminal identified by a source address to a destination terminal identified by a destination address. Ex.1001, 1:17-

21. The '845 patent explains that a “home gateway” interconnects a private domain (e.g., a private network such as a home network or a business network) and a public domain. Ex.1001, 1:40-48, 4:46-47.

The conventional version of the IP address system, known as IPv4 networking,² was widely used. It was commonly accepted that public IPv4 addresses would run out. Ex.1001, 1:61-62. The Internet Engineering Task Force (IETF) then developed a new IP addressing protocol referred to as IPv6.³ Ex1001, 1:61-2:6.

IPv6 allows for 128-bit addresses, which are substantially longer than IPv4's 32-bit addresses, and therefore provides a greater number of possible addresses than IPv4. Ex.1001, 7:26-29; Ex.1012, 7 (IPv4); Ex.1013, 3 (IPv6). Example IPv4 and IPv6 addresses are shown below. Ex.1013, § 2.2, 4-5.

² The IPv4 protocol was standardized by the IETF in RFC791 in September 1981. Ex.1012; see Ex.1043, ¶¶ 49-52 (citing Exs.1051-1052).

³ The IPv6 protocol was standardized by the IETF in RFC1884 in December 1995 and revised in later years. Ex.1013; see Ex.1043, ¶¶ 53-56 (citing Exs.1048, 1053); see also Ex.1022-1024; Ex.1043, ¶¶ 69-80 (citing Exs.1058-1063).

Example Addresses	
IPv4 Address	192.159.252.76
IPv6 Address	3FFE:F200:0234:AB00:0123:4567:8901:ABCD

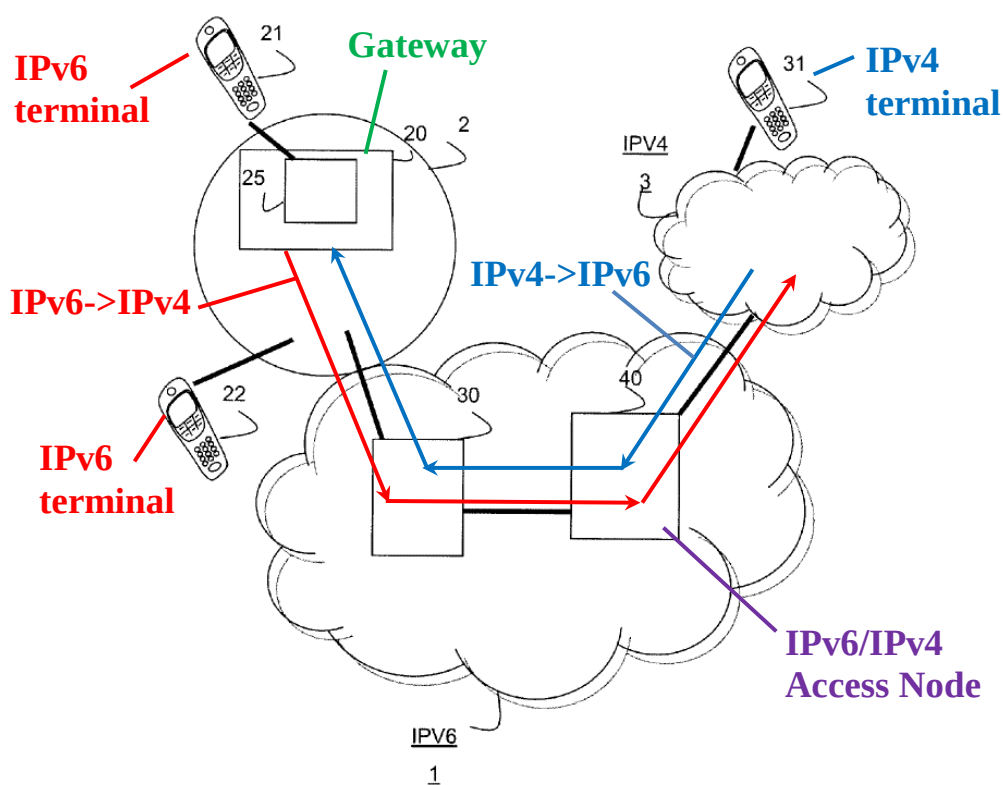
IPv4 and IPv6 addresses have two parts: (1) a prefix identifying the components of a subnetwork (“subnet”) through which a data packet is to be routed to reach its destination (e.g., router, gateway, etc.), and (2) an address of a machine connected to the subnetwork (e.g., a gateway, or the destination). Ex.1001, 7:30-34; Ex.1013, 3-4 (describing a subnet prefix). In the above example IPv6 address, the first 64 bits may be the **IPv6 prefix**, and the last 64 bits may be the **device address**, as shown below.

3FFE:F200:0234:AB00:0123:4567:8901:ABCD

The ’845 patent explains that if an IPv6-configured device in an IPv6 domain transmits an IPv6 packet to an IPv4-configured terminal in an IPv4 domain, the IPv6 packet must be translated to an IPv4 packet, and vice versa. Ex.1001, 3:18-48.

To facilitate transition from IPv4 to IPv6 addresses, the ’845 patent discloses that an IPv6 address is constructed by concatenating an IPv6 prefix (operator prefix), an IPv4 address, and a port number (e.g., port number of a gateway, router or access node connected to the destination terminal). Ex.1001, 3:7-9, 3:30-48,

7:18-20. As shown by the **IPv6-to-IPv4** translation in annotated Figure 1 below, the constructed IPv6 packet allows for the IPv6 packet from an **IPv6 terminal** packet to be transmitted across an IPv6 network via a **gateway** to an **IPv6/IPv4 access node**, which translates the IPv6 packet into an IPv4 packet and forwards the IPv4 packet to an **IPv4 destination terminal**. Ex.1001, 3:34-48, 11:45-61.



Ex.1003, ¶ 38; Ex.1001, Fig. 1 (annotated).

Conversely, the '845 patent discloses that an IPv4 packet from an **IPv4 terminal** is translated into an IPv6 packet⁴ by the **IPv6/IPv4 access node** and

⁴ This operation is known as “4in6” tunneling or “IPv4-in-IPv6” tunneling. See

transmitted across the IPv6 network 1 to an IPv6 terminal via the **gateway**.

Ex.1001, 11:38-61. This represents an **IPv4-to-IPv6** translation, as shown in annotated Figure 1 above.

The '845 patent explains that source and destination IPv6 addresses of an IPv6 packet can be either a native IPv6 address or a constructed IPv6 address, i.e., an address that is constructed as described above, and the gateway can replace the native IPv6 address or constructed IPv6 address. Ex.1001, 3:10-15, 11:6-12.

VI. PROSECUTION HISTORY

The '845 patent was filed as U.S. Application No. 13/501,907 (“’907 Application”) as a U.S. National Stage of PCT/FR2009/051228, which claims priority to French Patent Application No. 0854405, filed on June 30, 2008. Petitioner reserves the right to challenge the priority date of the '845 patent if Patent Owner attempts to rely on the priority date.

Only one Office Action was issued during prosecution of the '845 patent.⁵ Ex.1002, 391-396. The Examiner did not reject the claims on the basis of any

RFC2473 (Ex.1021, 2-7); *see also* Ex.1043, ¶¶ 65-68 (citing Exs.1056-1057).

⁵ In the Office Action, the Examiner cited U.S. 2005/0286553 to Wetterwald et al. (Ex.1010), which is a different reference than Wetterwald (Ex.1008) cited in the ground of unpatentability. Ex.1002, 396 (“Notice of References Cited”).

prior art. *Id.* The applicant presented clarifying amendments to overcome enablement and indefiniteness rejections relating to the construction of the IPv6 destination address, and a clarity objection relating to the “regularizing” limitation. Ex.1002, 393-394, 405-406, 408-409.

The Examiner then allowed the application. Ex.1002, 414-421. The Examiner identified limitations [1.1]/[8.1], [1.2.1]/[8.2.1], [1.2.2]/[8.2.2], and [1.2.3]/[8.2.3] as the allegedly allowable features of claims 1 and 8. Ex.1002, 420. However, as discussed in Section X.E below, these features are rendered obvious by Ananda (Ex.1005), Wetterwald (Ex.1008), and RFC4380 (Ex.1009). There is no record in the prosecution history that the Examiner considered Ananda, Wetterwald or RFC4380 in relation to the challenged claims.

VII. LEVEL OF ORDINARY SKILL IN THE ART

A person of ordinary skill in the art (“POSITA”) in June 2008 would have had a working knowledge of the packet switching art that is pertinent to the ’845 patent, including computer networking, IPv4, IPv6, packet encapsulation, and packet routing. A POSITA would have had a bachelor’s degree in electrical engineering, computer engineering, computer science, or a related subject, and three years of work experience in computer networking, including experience with the IPv4 and IPv6 protocols, or a related field. Lack of professional experience can be remedied by additional education, and vice versa. Ex.1003, ¶¶ 21-22.

VIII. CLAIM CONSTRUCTION

The Challenged Claims are construed “in accordance with the ordinary and customary meaning of such claim as understood by one of ordinary skill in the art and the prosecution history pertaining to the patent.” 37 C.F.R. § 42.100(b).

A. “*domain*” (Claims 1, 5, 7, and 8)

Petitioner does not believe that “*domain*” requires construction. To the extent that Patent Owner argues, or the Board determines, that construction is required, Petitioner submits that an appropriate construction is “network.” Ex.1003, ¶¶ 54-55. However, Petitioner’s detailed claim analysis below shows how the claims are unpatentable under multiple constructions that Patent Owner might advance.

The ’845 patent states that “[a] telecommunications network administered by an operator is also referred to as a domain.” Ex.1001, 1:30-31. The ’845 patent repeatedly suggests that “domain” is essentially a synonym for “network.” For example, the ’845 patent explains that a “home gateway [is provided] between a home network and the public network or IP domain of the operator,” and that the home gateway “generally assigns private IP addresses to the terminals of its home network.” Ex.1001, 1:40-44; Ex.1003, ¶ 56.

Newton’s Telecom Dictionary defines “*domain*” as “the part of a computer network in which the data processing resources are under common control....

Some people think of a domain as a group of computers on a network that share something in common, such as the job they do, the organization type they belong to, or their location.” Ex.1006, 196. The “Glossary of Networking Terms” in RFC 1208 defines “*domain*” as “an administrative partition of a complex distributed system.” Ex.1014, 6. Relatedly, RFC 1208 defines a “subnetwork” as “[a] collection of OSI [Open Systems Interconnection] end systems and intermediate systems under the control of a single administrative domain and utilizing a single network access protocol.” Ex.1014, 16. The “Internet Users’ Glossary” in RFC 1983 notes that “*domain*” is “a heavily overused term,” indicating that it may take on different meanings depending on context, including referring to “collection of hosts and routers, and the interconnecting network(s), managed by a single administrative authority” or to the “routers exchanging routing information within” such a collection. Ex.1015, 17, 4, 48; Ex.1003, ¶ 57; *see also* Ex.1043, ¶¶ 61-64 (citing Exs.1054-1055).

The term “domain” should not be construed to require “common control” or a “single administrative authority” because the claims themselves identify the point of commonality for each “*domain*” by reciting “*IPv4 domain*” or “*IPv6 domain*.” Thus, terminals within an “*IPv4 domain*” communicate using the IPv4 protocol, while terminals within an “*IPv6 domain*” communicate using the IPv6 protocol. Ex.1003, ¶ 58.

In the event Patent Owner may argue otherwise, this Petition shows that features such as “common control” would have been obvious. Ex.1003, ¶ 59.

In summary, no construction is believed necessary for “*domain*,” but to the extent a construction is adopted, the Board should construe “*domain*” as a “network.”

B. “regularizing at least one of the IPv6 source address and the IPv6 destination address” (claims 1 and 8)

By using the phrase “at least one of,” limitations [1.2.1] and [8.2.1] require either (1) regularizing the IPv6 source address by replacing the IPv6 source address, or (2) regularizing the IPv6 destination address by replacing the IPv6 destination address. Ex.1003, ¶ 60.

This interpretation is consistent with the applicant’s claim amendments and remarks during prosecution. In response to a clarity objection, the applicant amended limitations [1.2.1] and [8.2.1] to recite “regularizing at least one of the IPv6 source address and the IPv6 destination address of the data packet by replacing said at least one IPv6 address.” Ex.1002, 405, 406. The applicant explained that “[t]he words ‘at least’ make clear that it can be either the IPv6 destination address or the IPv6 source address. The claims have been amended to spell that out explicitly.” Ex.1002, 408; Ex.1003, ¶ 61.

C. “Means for” Limitations

Claim 8 recites several “means for” limitations. Petitioner submits that the following “means for” terms should be interpreted as means-plus-function limitations under 35 U.S.C. § 112 ¶ 6.⁶ Petitioner identifies the function and structure for each of the “means” terms below, in compliance with 37 C.F.R. § 42.104(b)(3).

Claim 8 recites that the claimed “device [is] for use in a home gateway adapted to connect a user terminal to said IPv6 domain.” Ex.1001, 19:20-21. With reference to annotated Figure 9 below, the ’845 patent describes “a device 25 [] for receiving a data packet in an IPv6 domain.” Ex.1001, 16:56-58. The ’845 patent explains that the device 25 includes “hardware elements conventionally found in a standard computer or a computer router,” including a **processor 25₁**, a “**telecommunications means 25₄** for communicating with the network 1,” and a **read-only memory 25₅**. Ex.1001, 16:55-17:7; Ex.1003; ¶ 64.

⁶ In the related district court litigation, Patent Owner argued that the “means” terms identified in this Section should be construed under 35 U.S.C. § 112 ¶6. Ex.1019, 14-16, 19-20.

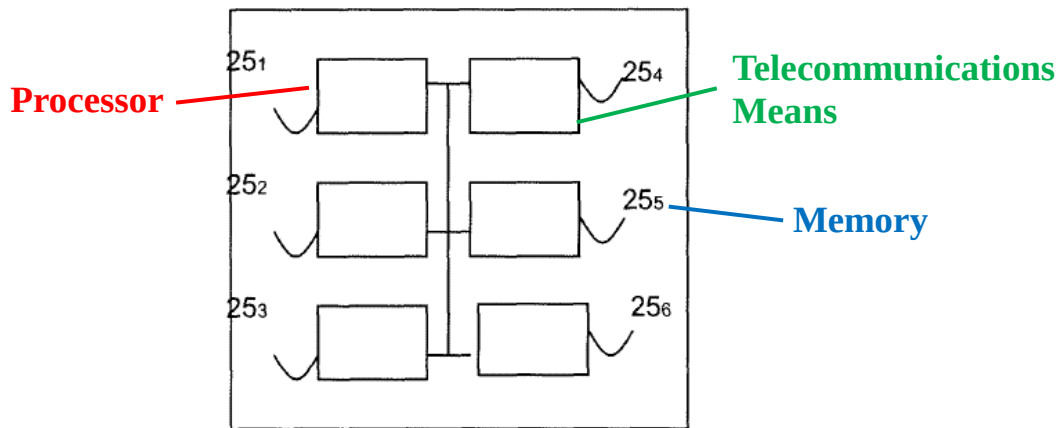


FIG. 9

Ex.1003, ¶ 64; Ex.1001, Fig. 9 (annotated).

1. **“means for identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address, and a destination port number;”**

Function: identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address, and a destination port number.

Structure: a receiver device (e.g., device 25) of the home gateway 20 having a processor 25₁ and memory 25₅, and equivalents. Ex.1001, 10:61-11:4, 16:55-17:7; Ex.1003, ¶ 67.

Explanation: The '845 patent explains that the disclosed method of receiving an IPv6 data packet is “executed by **a receiver device included in a home**

gateway connected to the IPv6 domain, such as the gateway 20,” and includes a step R1 (Fig. 3) of identifying an IPv6 destination address. Ex.1001, 10:61-11:4.

The '845 patent explains that the receiver device of the home gateway 20 determines whether the IPv6 destination address is a constructed IPv6 address based on whether the IPv6 destination address contains a certain IPv6 prefix. Ex.1001, 13:11-16, 10:10-22, 12:40-43, 13:10-14; Ex.1003, ¶¶ 68-69.

2. “means for regularizing, if necessary, at least one of the IPv6 source address and the IPv6 destination address of the data packet by replacing said at least one IPv6 address,... [and] modifying the data packet using the regularized address;”

As described above in Section VIII.B, by using the phrase “at least one of,” the “regularizing” limitation requires either of the following functions (1) or (2):

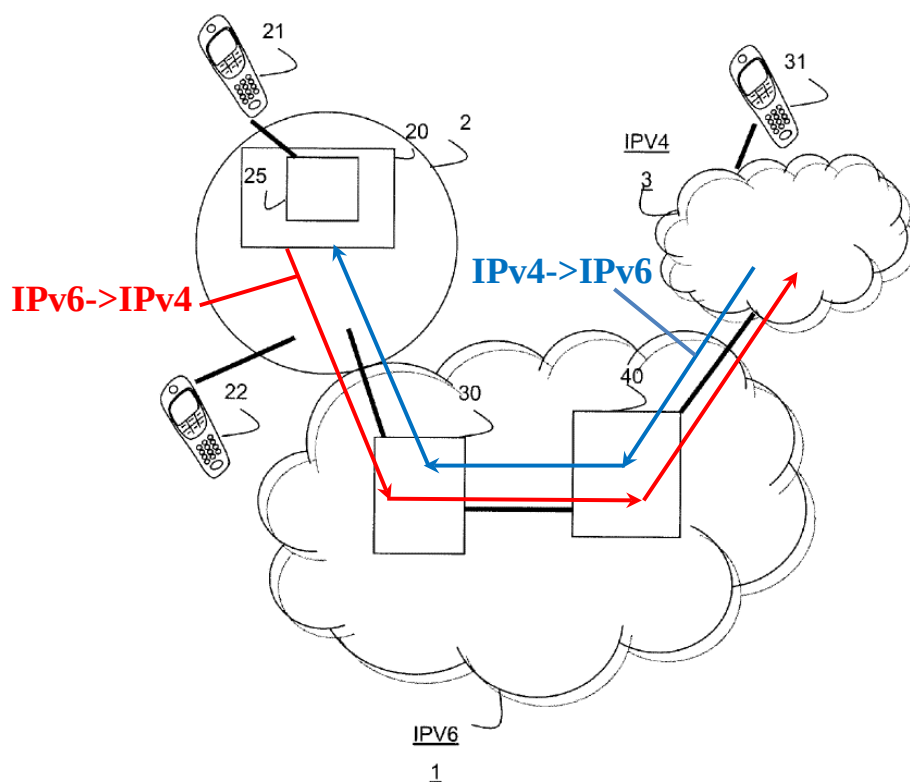
Function: (1) regularizing the IPv6 source address by replacing the IPv6 source address, or (2) regularizing the IPv6 destination address by replacing the IPv6 destination address. Further, the “means for regularizing” includes the function of modifying the data packet using the regularized address.

Structure: the device 25 of the home gateway 20 having a processor 25₁ and a memory 25₅, where the memory 25₅ stores a network address translation table (NAT) for translating between addresses, and equivalents. Ex.1001, 16:55-17:7,

16:18-44;⁷ Ex.1003, ¶ 74.

Explanation: With reference to annotated Figure 1 below, the '845 patent discloses that when a packet is transmitted from an IPv4 domain 3 to the home gateway 20 in the IPv6 domain (**IPv4->IPv6** in annotated Figure 1), the IPv6 source address of the packet is regularized (i.e., replaced) using an address translation table, whereas when a packet is transmitted from an IPv6 terminal in the IPv6 domain 1 via the gateway 20 to the IPv4 network (**IPv6->IPv4** in annotated Figure 1), the IPv6 destination address is regularized (i.e., replaced) using an address translation table. Ex.1001, 3:10-15, 11:7-11, 16:18-44, 16:55-17:7; Ex.1003, ¶¶ 75-76.

⁷ In the related district court litigation, Petitioner has argued that the specification does not identify structure for determining if the regularizing feature is “necessary.” As explained in Section X.E.5[1.2.1] below, Petitioner shows that the conditional “regularizing” features of limitations [1.2.1]-[1.2.3] of claim 1 and limitations [8.2.1]-[8.2.3] of claim 8 are obvious based on the assumption that such features are “necessary.”



Ex.1003, ¶¶ 75-76; Ex.1001, Fig. 1 (annotated).

3. “means for routing the IPv6 data packet to its destination.”

Function: routing the IPv6 data packet to its destination.

Structure: telecommunications means 25₄ of the device 25 of the home gateway 20, and equivalents. Ex.1001, 16:58-63; Ex.1003, ¶ 78.

Explanation: The '845 patent explains that the device 25 of the home gateway 20 includes “the hardware elements conventionally found in a standard computer or a dedicated router, namely a... **telecommunications means 25₄** for communicating with the [IPv6] network 1.” Ex.1001, 16:58-63; Ex.1003, ¶¶ 79-

80.

IX. RELIEF REQUESTED AND THE REASONS FOR THE REQUESTED RELIEF

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

X. IDENTIFICATION OF HOW THE CLAIMS ARE UNPATENTABLE

A. Challenged Claims and Statutory Grounds for Challenge

Ground	Claims	Basis
#1	1 and 5-8	35 U.S.C. § 103 (Pre-AIA) over Ananda, Wetterwald, and RFC4380

U.S. Patent Application Publication No. 2004/0107287 to Ananda et al. (“**Ananda**,” Ex.1005) published on June 3, 2004. Ananda is prior art under 35 U.S.C. §§ 102(a), (b), and (e).

“Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)” by C. Huitema was published by the Internet Engineering Task Force as Request for Comments 4380 (“**RFC4380**,” Ex.1007) in February 2006. IETF publications, including its Requests for Comments and RFC4380, were publicly available and well-known documents to those of skill in the art. Ex.1043, ¶¶ 45-48 (public accessibility of RFC4380) (citing Exs.1049-1050); Ex.1003, ¶ 24 (citing Exs.1007, 1012-1015, 1021-1024, 1036, 1042); Ex.1042, 6, 8, 26. RFC4380 is

prior art under 35 U.S.C. § 102(a) and (b).

U.S. Patent No. 7,639,686 to Wetterwald et al. (“**Wetterwald**,” Ex.1008). Wetterwald was filed on April 7, 2005 and issued on December 29, 2009. Wetterwald is prior art under 35 U.S.C. § 102(e). Wetterwald was also the subject of a pre-grant publication, U.S. Pub. No. 2006/0227792 on October 12, 2006, and is therefore prior art under 35 U.S.C. § 102(a) and (b). *See* Ex. 1025.

Dr. Min’s Declaration and this Petition cite to several documents published by the Internet Engineering Task Force (IETF) as further evidence for various facts. To the extent the Board determines that these IETF documents must qualify as prior art “printed publications” for the purposes for which they are cited, the documents do so qualify. Ex.1003, ¶ 24 (citing Exs.1007, 1012-1015, 1021-1024, 1036, 1042); Ex.1043, ¶¶ 41-44 (citing Exs.1047-1048), 49-88 (citing Exs.1051-1067). The ’845 patent acknowledges that IETF documents—including draft documents—were accessible, available, and known to the public. *See, e.g.*, Ex.1001, 7:46-49 (citing RFC2462), 7:53-58 (citing RFC3315), 7:67-8:5 (citing RFC3633), 9:58-61 (citing RFC3633), Code (56), References Cited (citing RFC2663). The Board has repeatedly found that IETF documents, including RFCs, are “printed publications.” *See, e.g.*, IPR2017-00337, Paper 31 at 46-47 (Requests for Comments are “precisely the type of documents whose main purpose is for public disclosure”); IPR2018-00130, Paper 11 at 30-33 (finding multiple

RFCs to be printed publications).

B. Discretionary denial under 35 U.S.C. § 325(d) is not appropriate

Ananda (Ex.1005) and RFC4380 (Ex.1009) were not considered or cited during prosecution of the '845 patent.

A document related to Wetterwald (Ex.1010, "'553 Publication") was cited by the Examiner but not applied in a rejection or substantively discussed during prosecution. Ex.1002, 396. Petitioner's analysis below references column 8, lines 24-28 of Wetterwald (Ex.1008) for limitations [1.1] and [8.1], and the features of the translation resource 53 and NAT table 54 in Column 6, line 38 to Column 7, line 36 of Wetterwald (Ex.1008) for limitations [1.2.1]-[1.2.3] and [8.2.1]-[8.2.3]. There is no indication that the Examiner considered corresponding disclosure of '553 Publication (Ex.1010, ¶¶ 50, 36, 40-42), nor did the Examiner apply this portion of '553 Publication in rejecting the challenged claims. Thus, to the extent Wetterwald (Ex.1008) was before the Examiner, the relevance of its teachings was overlooked.

C. Discretionary denial under the *Fintiv* factors is not appropriate

The Board balances the six factors in considering denial under § 314. These factors strongly favor institution. *Apple Inc. v. Fintiv, Inc.*, IPR2020-00019, Paper 11 (PTAB Mar. 20, 2020) (precedential).

1. Whether evidence exists that a stay may be granted if a proceeding is instituted

A stay pending the outcome of an IPR in the co-pending district court litigation has not been requested. If an IPR is instituted, a stay would be appropriate under the legal standard in the Eastern District of Texas. *See Cypress Lake Software, Inc. v. Samsung Elecs. Am., Inc.*, No. 6:18-cv-00030 (E.D. Tex. Aug. 28, 2019) (order granting stay). Patent Owner waited for years after the '845 patent issued on May 28, 2013, before filing its litigation Complaint on January 21, 2020. Ex.1001; Ex.1020. Further, Petitioner diligently reviewed prior art and prepared this Petition upon being informed on May 4, 2020 of the claims of the '845 patent asserted by Patent Owner against Petitioner. Ex.1020, 2. Petitioner worked diligently to prepare and file this Petition in about four months from being informed of the asserted claims, and approximately four months before the one-year time period allowed under statute. As such, the litigation is still in its early stages, and a stay pending IPR would be appropriate.

However, given that a motion to stay has not yet been filed, the Board should not infer the outcome of that motion. *Intel Corp. v. VLSI Tech. LLC*, IPR2020-00158, Paper 16, at 7 (PTAB May 20, 2020); *Apple Inc. v. Fintiv, Inc.*, IPR2020-00019, Paper 15, at 12 (PTAB May 13, 2020). Thus, this factor is neutral on discretionary denial.

2. Proximity of the Court’s trial date to the Board’s projected statutory deadline for a final written decision

Fintiv states that this factor does not implicate the decision whether to institute “if the trial date is at or around the same time as the projected statutory deadline.” IPR2020-00019, Paper 11, at 6. The district court’s schedule currently calls for a claim construction hearing to be held on November 20, 2020 and for jury selection for trial to begin May 3, 2021. *See* Ex.1040. A Final Written Decision in this proceeding is likely to issue in or around April 2022. While jury selection is currently scheduled before the expected Final Written Decision, the scheduled May 3, 2021 trial date is uncertain, as the district court has eight other cases scheduled for trial that same day. *See* Ex.1026, 1 (trial date scheduled for May 3, 2021); Ex.1027, 1 (same); Ex.1028, 1 (same); Ex.1029, 1 (same); Ex.1030, 1 (same); Ex.1031, 1 (same); Ex.1032, 1 (same); Ex.1033, 1 (same). It is thus a fair possibility that the scheduled May 3, 2021 trial date involving Petitioner and Patent Owner will be moved back closer to, or even after, the time of the expected Final Written Decision. Because of the uncertainty of the current trial date, this factor weighs against discretionary denial. *See Sand Revolution II, LLC v. Continental Intermodal Grp.—Trucking LLC*, IPR2020-01393, Paper 24 (June 16, 2020) (panel granting institution on rehearing based on finding that uncertainty over district court’s trial date weighed against discretionary denial).

3. Investment in the parallel proceeding by the Court and the parties

The litigation is in its early stages, and the investment in it has been minimal. As mentioned above, a claim construction hearing is currently scheduled for November 20, 2020. *See* Ex.1040, 3. The early stage of the litigation weighs against discretionary denial.

4. Overlap between issues raised in the petition and in the parallel proceeding

The prior art addressed in the Petition is also part of Petitioner's invalidity contentions in the litigation. Instituting a proceeding will allow the Board to address the art, and the issues will be narrowed in the litigation due to the estoppel provisions of 35 U.S.C. § 315(e)(2). To reduce overlap further, if the Board institutes trial, Petitioner will cease asserting in the district court litigation the combination of references on which trial is instituted for the claims on which trial is instituted, while trial is instituted. The most efficient course is for the Board to address the issues before it, and the district court to address only the issues that would remain. Accordingly, this factor weighs against discretionary denial.

5. Whether the petitioner and the defendant in the parallel proceeding are the same party

Petitioner is a defendant in the litigation. Ex.1020. That is true of most Petitioners in IPR proceedings. While the *Fintiv* case indicates that a difference

between the district court defendant and the petitioner may way against discretionary denial, nothing within the *Fintiv* case suggests that the same party between the proceedings weighs in favor of discretionary denial. Accordingly, this factor should not be a basis for denying institution.

6. Other circumstances that impact the Board’s exercise of discretion, including the merits

As discussed in detail below in Section X.E, the prior art presented in this Petition renders the Challenged Claims unpatentable as obvious. The merits of Petitioner’s arguments are strong, and this factor weighs against discretionary denial.

As such, because the *Fintiv* factors are either neutral or weigh against discretionary denial, and because this Petition was filed more than three months before the statutory bar date, institution should not be denied on discretionary factors.

D. Discretionary denial under *General Plastic* is not appropriate

The Board should not exercise its discretion under 35 U.S.C. § 314(a) to deny this Petition. Among other factors, the ’845 patent has not been challenged in any prior IPR petition. None of discretionary institution factors 1-5 in *General Plastic* apply to this Petition. See *General Plastic Indus. Co., Ltd. v. Canon Kabushiki Kaisha*, IPR2016-01357, Paper 19 at 16 (PTAB Sept. 6, 2016) (Section

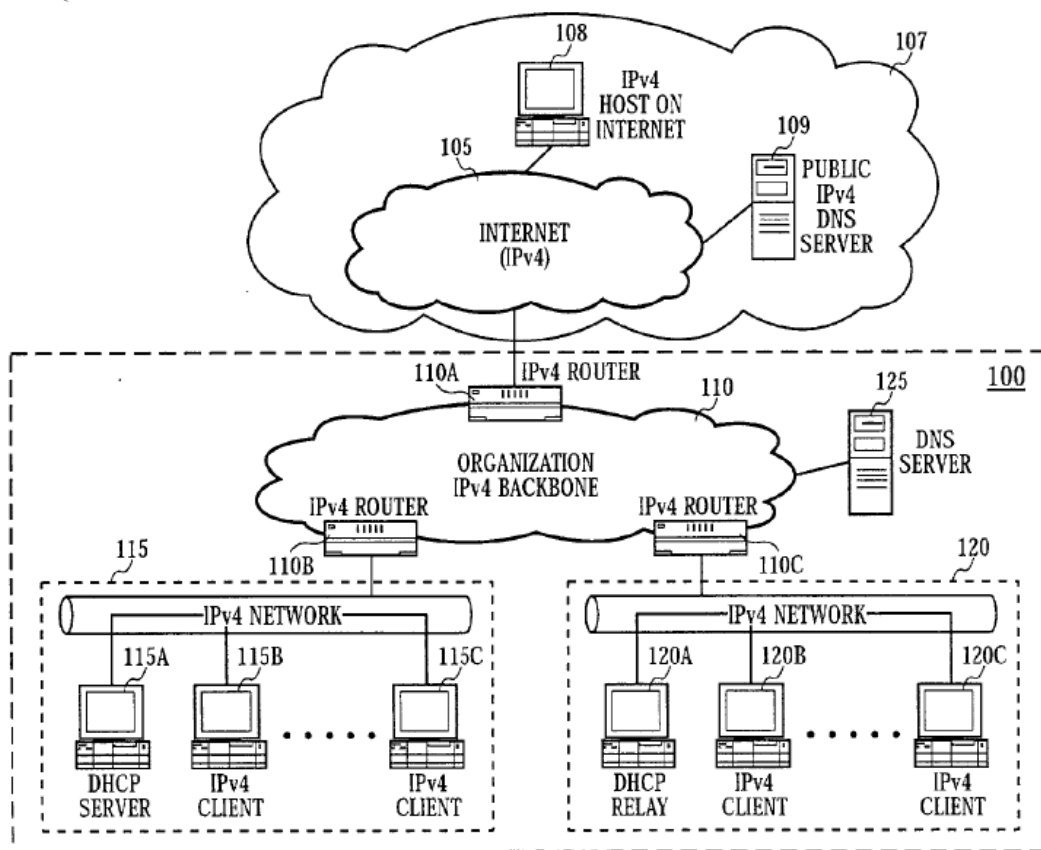
II.B.4.i. precedential).

E. Ground 1: Claims 1 and 5-8 are obvious over Ananda, Wetterwald, and RFC4380

1. Ananda

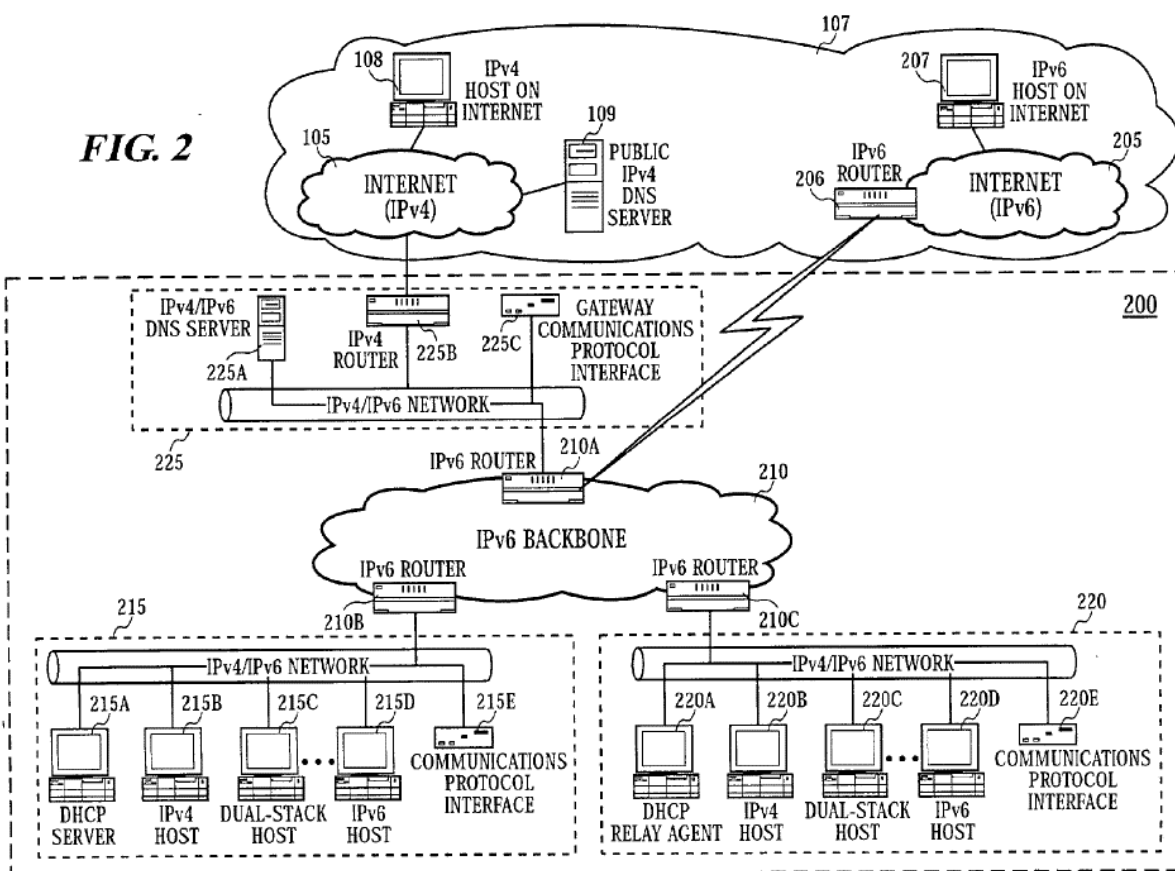
Like the '845 patent, Ananda is directed to enabling devices that are configured for IPv4 networking to communicate across IPv6 networks, and vice versa. Ex.1005, ¶ 17. Ananda explains that computing devices will transition from IPv4 to IPv6 due to the shortage of address space in IPv4, but that transition will be gradual. Ex.1005, ¶¶ 6-9, 17. Ananda notes that “IPv6 and IPv4 communication protocols do not interoperate,” and thus there is “a need for transparent communication between IPv4 and IPv6 on communication networks, particularly among IPv4 nodes over IPv6 communication networks.” Ex.1005, ¶ 9.

With reference to Figure 1 below, Ananda first describes a conventional IPv4 network 100 that includes two distinct subnetworks (“subnets”) 115, 120, which are each connected to a private IPv4 backbone network 110 via respective IPv4 routers 110B, 110C. Ex.1005, ¶¶ 65, 67. The IPv4 backbone network 110 is connected to a public network (IPv4 portion 105 of the Internet 107) via an IPv4 router 110A. Ex.1005, ¶ 67.



Ex.1005, Fig. 1.

Ananda discloses that one way to transition IPv4 network 100 to an IPv6 network is to change all the devices in network 100 to operate according to the IPv6 standard, but that may be cost-prohibitive. Ex.1005, ¶¶ 71-72. Ananda discloses an alternative transition approach, illustrated in Fig. 2 below, that replaces all IPv4 routers 110A-110C in Figure 1 with IPv6 routers 210A-210C and upgrades the IPv4 backbone 110 to an IPv6 backbone 210. Ex.1005, ¶¶ 73, 75-77.



Ex.1005, Fig. 2.

Ananda discloses that the subnet 215 is transitioned to an IPv4/IPv6 network by using a communications protocol interface (CPI) 215E that can communicate according to both the IPv4 and IPv6 protocols, and the IPv4/IPv6 network includes an IPv4 host 215B and an IPv6 host 215D. Ex.1005, ¶ 78. The subnet 220 is similarly transformed to an IPv4/IPv6 network with an IPv4/IPv6 CPI 220E, an IPv4 host 220B, and an IPv6 host 220D. Ex.1005, ¶ 79. Similar to the IPv4 clients 115B, 115C in subnet 115 and the IPv4 clients 120B, 120C in subnet 120 in Figure 1, Ananda discloses that the IPv4 host 215B in subnet 215 and IPv4 host 220B in

subnet 220 in Figure 2 may be assigned private IPv4 addresses. Ex.1005, ¶¶ 83-84.

Like the '845 patent, Ananda discloses that when the IPv4 host 215 transmits an IPv4 data packet, the IPv4 data packet is encapsulated in an IPv6 data packet carrying an IPv6 destination address, as discussed in Section X.E.5[1.0.1] below. Ex.1005, ¶¶ 103, 105, 137-138; Ex.1003, ¶¶ 88-89. Further, like the '845 patent, Ananda discloses that the IPv6 destination address is constructed by combining an IPv6 prefix and an IPv4 address of the destination terminal as discussed in Section X.E.5[1.1] below. Ex.1005, ¶¶ 116, 103, 105-107; Ex.1003, ¶ 90.

2. Wetterwald

Like the '845 patent and Ananda, Wetterwald is directed to providing interoperability between IPv4 nodes and IPv6 nodes across an IPv6 network. Ex.1008, 1:8-14. Wetterwald describes an address assignment scheme that allows an enterprise to assign private IPv4 addresses to IPv4 hosts connected to a private IPv4 network, thus limiting the number of globally-unique IPv4 addresses required for hosts connecting to a public IPv4 network, *e.g.*, the Internet. In particular, Wetterwald explains that concerns arise in the event that an IPv6 node is coupled to a private IPv4 network having a Network Address Translator (NAT). NATs perform a Layer-3 translation of IP-Addresses, so that public Internet addresses map to private IP addresses, as described in detail by RFC1918 (Ex.1007). This

mapping has allowed enterprises to map a large number of private addresses to a limited number of public addresses, thus limiting the number of public addresses required by Internet users. Ex.1008, 1:35-44; Ex.1003, ¶ 91; *see also* Ex.1043, ¶¶ 41-44 (citing Exs.1007, 1047-1048).

3. RFC4380

Like the '845 patent, Ananda and Wetterwald, RFC4380 is directed to providing interoperability between IPv4 nodes and IPv6 nodes across an IPv6 network. Ex.1009, 1. RFC4380 provides interconnection between IPv6 and IPv4 terminals by allowing IPv4 terminals (referred to as a Teredo client) to communicate with an IPv6 network through the use of a Teredo server arranged in an IPv4 network. Ex.1009, 4 (§ 2.3). The Teredo server communicates with a Teredo relay, which can communicate according to the IPv6 protocol—the Teredo relay thus functions as an IPv6 router. Ex.1009, 4 (§ 2.4). To construct an IPv6 packet, the Teredo relay uses an IPv6 prefix together with the IPv4 address of the Teredo server, a port number (referred to as a UDP port), and an IPv4 address of the original IPv4 packet, as shown below. Ex.1009, 11 (§ 4). The UDP port refers to the port number of the Teredo server that is waiting for packets from the IPv4 terminal. Ex.1009, 4 (§ 2.7).

Prefix	Server IPv4	Flags	Port	Client IPv4
--------	-------------	-------	------	-------------

Ex.1009, 11 (§ 4).

RFC4380 thus provides a 64-bit identifier that contains the UDP port and IPv4 address at which an IPv4 client can be reached through the Teredo service. Ex.1009, 5 (§ 2.13). RFC4380 is designed for use when an IPv4 terminal communicates with an IPv4 router that uses network address translation (NAT) to communicate with a public network such as the Internet. Ex.1009, 8 (§ 3.2.1); Ex.1003, ¶¶ 92-93.

4. Reasons to combine Ananda, Wetterwald, and RFC4380

An explanation of the multiple reasons to combine the teachings of Ananda, Wetterwald, and RFC4380 are provided in the detailed claim analysis below.

5. Claim 1

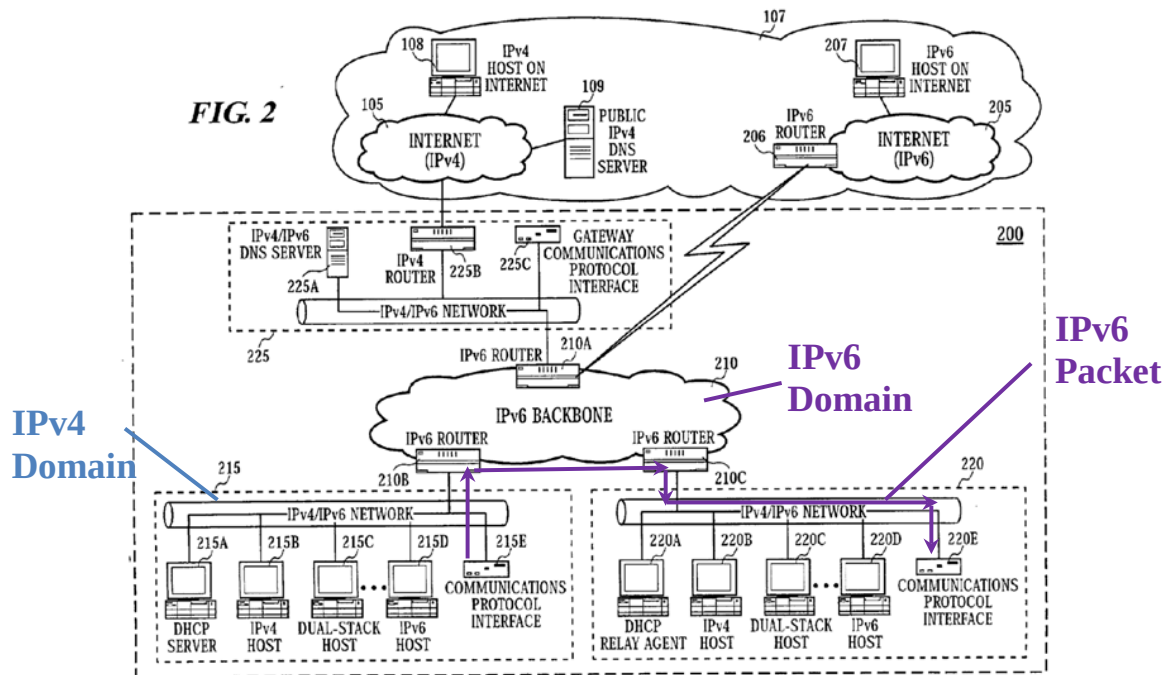
[1.0.1] A method for receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain,

Ananda renders obvious limitation [1.0.1]. Ex.1003, ¶ 95.

Ananda discloses “a method in a communication protocol interface (CPI) for communicating a variety of data packets that comply with a corresponding variety of communication protocols on a communication network,” where the method includes “*receiving a data packet.*” Ex.1005, ¶¶ 35-36, 18. Ananda explains that

CPIs in multi-protocol subnets of the communication network transport IPv4 data packets across an IPv6 communication backbone (“*IPv6 domain*”) encapsulated in IPv6 data packets between the subnets. Ex.1005, ¶ 63; Ex.1003, ¶¶ 96-98.

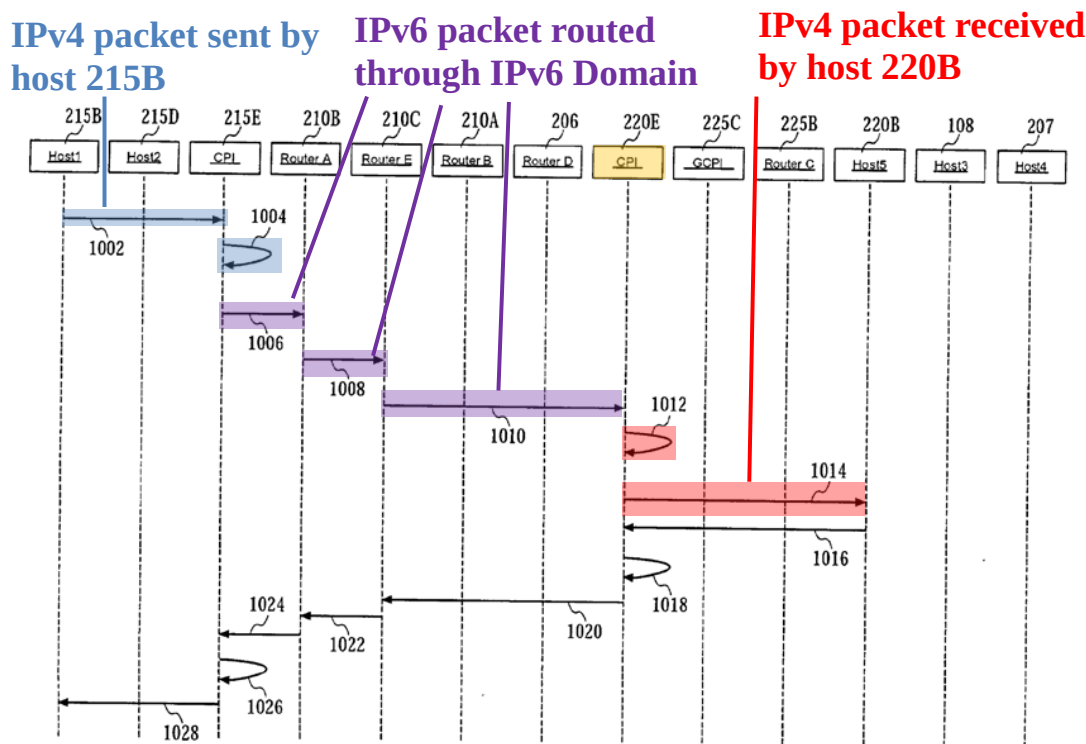
With reference to annotated Figure 2 below, Ananda discloses that the CPI 220E of multi-protocol IPv4/IPv6 subnet 220 receives an IPv6 packet in an IPv6 backbone 210 (“*IPv6 domain*”), which is connected to a multi-protocol IPv4/IPv6 subnet 215 (“*IPv4 domain*”). Ex.1005, ¶¶ 79, 139-142. Ananda discloses that CPI 220E of subnet 220 and CPI 215E of subnet 215 communicate according to both the IPv4 and IPv6 protocols. Ex.1005, ¶¶ 73, 78-79. Ananda discloses that the CPIs 215E, 220E transport IPv4 data packets across the IPv6 communication backbone 210 using IPv6 data packets, thereby allowing the IPv4 hosts 215B, 220B in the subnets 215, 220 to communicate with each other, and to allow multi-protocol IPv4/IPv6 communications between IPv4- and IPv6-configured devices in the subnets 215, 220. Ex.1005, ¶¶ 76-79, Abstract; Ex.1003, ¶¶ 99-101.



Ex.1003, ¶ 102; Ex.1005, Fig. 2 (annotated).

Annotated Figure 2 above shows an IPv6 packet being transmitted from CPI 215E in the subnet 215 (“*IPv4 domain*”) across the IPv6 backbone 210 (“*IPv6 domain*”) to CPI 220E of subnet 220. Ex.1005, ¶¶ 79, 139-142; Ex.1003, ¶ 102.

With reference to annotated Figure 10 below, Ananda describes the sequence of actions when an IPv4 data packet sent by the IPv4 host 215B in subnet 215 (“*IPv4 domain*”) is routed via the IPv6 communication backbone 210 (“*IPv6 domain*”) and CPI 220E to the destination IPv4 host 220B in subnet 220. Ex.1005, ¶ 136; Ex.1003, ¶ 103.



Ex.1003, ¶ 103; Ex.1005, Fig. 10 (annotated).

In step 1002 of Figure 10, the IPv4 host 215B in subnet 215 (“**IPv4 domain**”) sends an IPv4 data packet to the IPv4 host 220B. Ex.1005, ¶ 137. In step 1004, CPI 215E encapsulates the IPv4 packet into an IPv6 packet and transmits it through IPv6 backbone 210 (“**IPv6 domain**”) via routers 210B and 210C (steps 1006 and 1008). Ex.1005, ¶¶ 138-140. In step 1010, the CPI 220E receives the IPv6 packet from router 210C in IPv6 backbone 210 (“**IPv6 domain**”). In step 1012, CPI 220E decapsulates the header of the IPv6 packet to produce the original IPv4 packet transmitted by IPv4 host 215B, and transmits the IPv4 packet to IPv4 host 220B. Ex.1005, ¶¶ 142-143. Ex.1003, ¶ 104.

While petitioner does not believe that “IPv4 domain” or “IPv6 domain” requires claim construction, it would have been obvious to a POSITA that Ananda’s subnets 215 and 220 are each a separate *IPv4 domain* and the IPv6 backbone 210 is an *IPv6 domain* under any reasonable construction of those terms. Ex.1003, ¶ 105. For example, to the extent that *IPv4 domain* is interpreted to refer to an administrative partition of a network that employs the IPv4 protocol, Ananda’s Figures 2 and 8 show subnets 215 and 220 in dotted-line boxes, indicating that they are separate partitions of the overall network. Ananda states that each subnet is a “physically independent network segment.” Ex.1005, ¶ 66. Subnets 215 and 220 also include devices communicating via the IPv4 protocol. Ex.1005, ¶¶ 76, 78-79, 82 (“a subnet can comprise purely IPv4 hosts”). Ex.1003, ¶¶ 105-106.

To the extent that IPv4 domain is interpreted to mean a part of a computer network that uses the IPv4 protocol and in which the data processing resources are under common control, Ananda further explains how each subnet includes a CPI and an IPv6 router that control network traffic flowing to and from the individual subnet. Ex.1005, ¶¶ 78-79. In particular, the IPv4 devices in subnet 215 are under the common control of CPI 215E, which enables multi-protocol communication within the subnet 215 (Ex.1005, ¶ 78), and CPI 215E and IPv6 router 210B exchange IPv6 packets with the IPv6 network 210 on behalf of IPv4 devices in the

subnet 215 (e.g., IPv4 host 215B). Ex.1005, ¶ 137 (step 1002 in Fig. 10), ¶ 150 (step 1028 in Fig. 10). Communications to and from subnet 220 are facilitated separately by IPv6 router 210C and CPE 220E. *See* Ex.1005, ¶¶ 143-144 (steps 1014 and 1016 in Fig. 10). Therefore, a POSITA would have understood that the collection of IPv4 hosts in subnet 215 are under common control and that the common control of subnet 215 is separate from the common control of subnet 220. Thus, subnet 215 and subnet 220 are each a separate *IPv4 domain*. Ex.1003, ¶ 107.

Finally, to the extent that *IPv4 domain* is interpreted to mean a collection of IPv4 hosts, IPv4 routers, and interconnecting network(s) managed by a single administrative authority, Ananda teaches that subnets 215 and 220 are each a “physically independent network segment.” Ex.1005, ¶ 66. Because each subnet is physically independent, it would have been obvious to a POSITA for each subnet to be managed by an independent administrative authority. Ex.1003, ¶¶ 85, 108. Ananda teaches that each subnet has a separate computer configured to provide dynamic host control protocol (DHCP) services (e.g., servers 215A and 220A). Ex.1003, ¶¶ 108, 68, 78. DHCP was a well-known protocol used to administer IPv4 network addresses. *See, e.g.*, Ex.1036, 1 (providing “automatic allocation of reusable network addresses”), 3, 12; Ex.1003, ¶¶ 85, 108. Therefore, a POSITA would have understood that the hosts in subnet 215 are managed separately from the hosts in subnet 220. Thus, subnet 215 and subnet 220 are each a separate *IPv4*

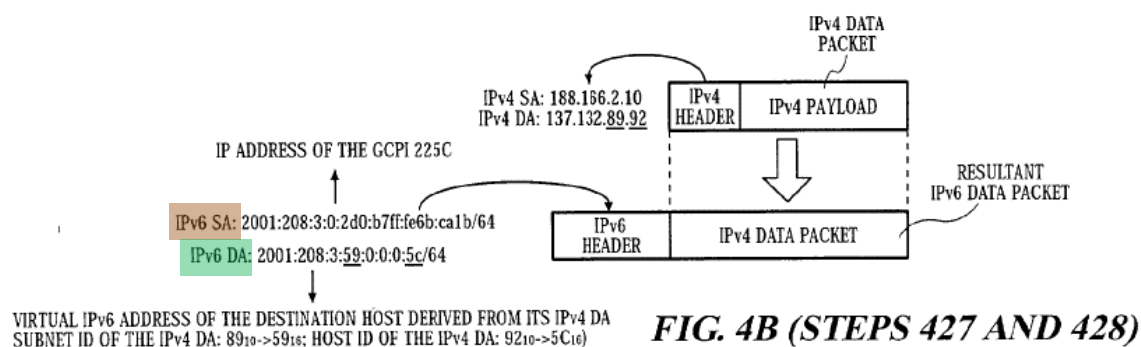
domain. Ex.1003, ¶ 108.

Accordingly, Ananda renders obvious limitation [1.0.1]. Ex.1003, ¶ 109.

[1.0.2] said packet comprising an IPv6 destination address and an IPv6 source address,

Ananda renders obvious limitation [1.0.2]. Ex.1003, ¶ 110.

As described above with respect to limitation [1.0.1], CPI 215E in subnet 215 encapsulates an IPv4 packet received from IPv4 host 215B to produce an IPv6 packet. Ex.1005, ¶ 138. As shown in annotated Figure 4B below, Ananda discloses that the packet transmitted from CPI 215E to CPI 220E includes an IPv6 destination address (“**IPv6 DA**”) and an IPv6 source address (“**IPv6 SA**”) in the header of the IPv6 packet (“*resultant IPv6 data packet*”). Ex.1005, ¶ 138 (IPv6 packet has IPv6 source address of CPI 215E and constructed IPv6 destination address of IPv4 host 220B). Ex.1003, ¶ 111.



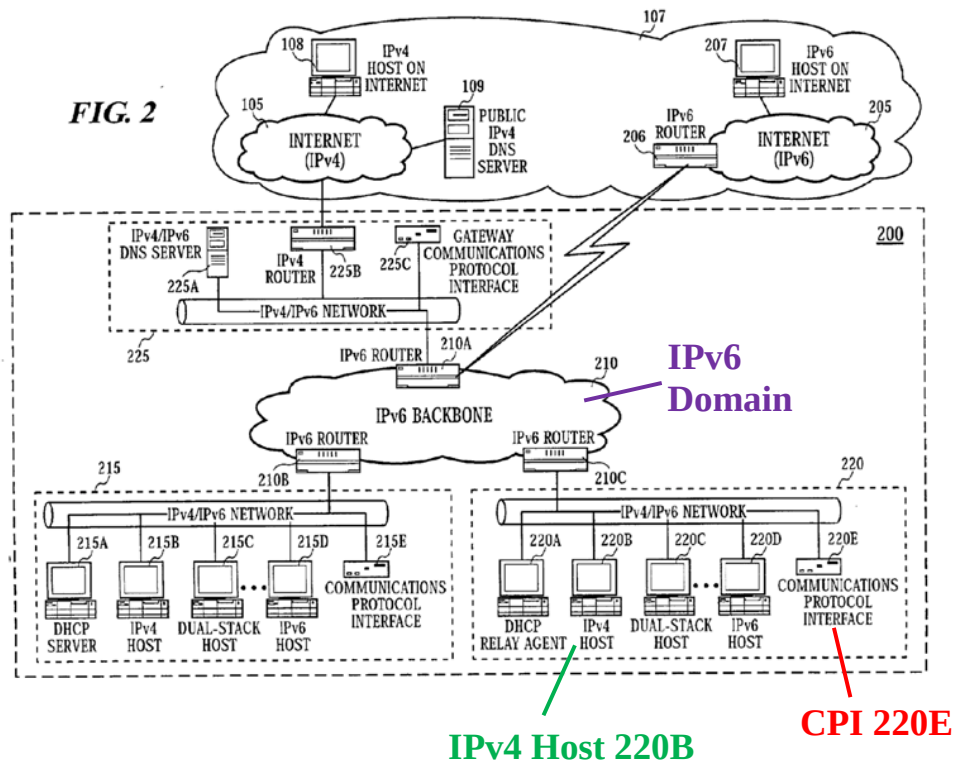
Ex.1003, ¶ 111; Ex.1005, Fig. 4B (annotated).

Thus, Ananda renders obvious limitation [1.0.2]. Ex.1003, ¶ 112.

[1.0.3] said method being executed in a home gateway adapted to connect a user terminal to said IPv6 domain, and comprising the following steps:

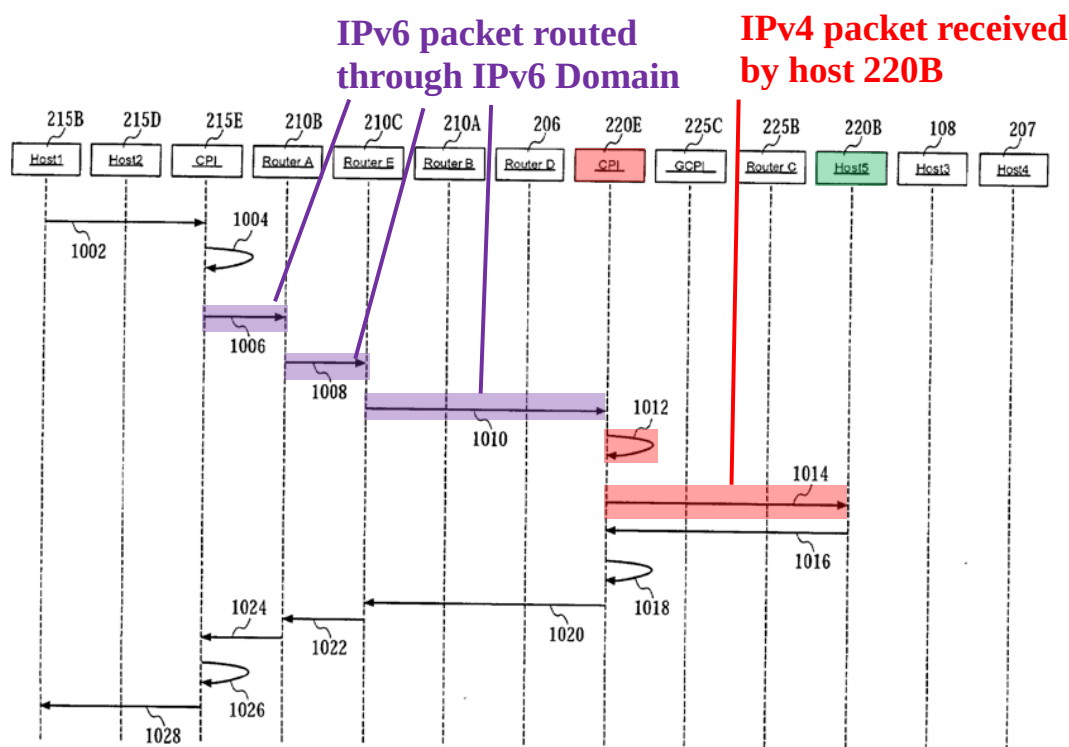
Ananda renders obvious limitation [1.0.3]. Ex.1003, ¶¶ 113.

With reference to annotated Figure 2 below, Ananda discloses that the method is executed in **CPI 220E** (“home gateway”) adapted to connect **IPv4 host 220B** (“user terminal”) to the **IPv6 backbone 210** (“IPv6 domain”). Ananda discloses that the **CPI 220E** (“home gateway”) enables **IPv4 host 220B** (“user terminal”) to communicate with **IPv6 backbone 210** (“IPv6 domain”). Ex.1005, ¶¶ 79, 77-78; *see also* Ex.1005, ¶ 63 (CPIs in multi-protocol subnets transport IPv4 data packets across an IPv6 communication backbone encapsulated in IPv6 data packets between the subnets). Ex.1003, ¶ 114.



Ex.1003, ¶ 114; Ex.1005, Fig. 2 (annotated).

As shown in annotated Figure 10 below, Ananda discloses that **CPI 220E** (“home gateway”), upon receiving an IPv6 packet from CPI 215E through IPv6 backbone 210 (via IPv6 routers 210B and 210C), extracts (decapsulates) the IPv4 packet in the transmitted IPv6 packet and routes the IPv4 packet to **IPv4 host 220B** (“user terminal”). Ex.1005, ¶¶ 138-143; Ex.1003, ¶ 115.



Ex.1003, ¶ 115; Ex.1005, Fig. 10 (annotated).

Thus, Ananda renders obvious limitation [1.0.3]. Ex.1003, ¶ 116.

[1.1] identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address and a destination port number;

Ananda in view of Wetterwald and RFC4380 renders obvious limitation [1.1]. Ex.1003, ¶¶ 117.

With reference to annotated Figure 4 below, Ananda discloses that the IPv6 packet transmitted from CPI 215E includes *an IPv6 destination address that is constructed*. Ex.1005, Fig. 4, step 440, ¶ 138 (construction of IPv6 destination address). Ex.1003, ¶ 118. As described above with respect to limitations [1.0.1] and [1.0.3], the CPI 220E receives the IPv6 packet from the CPI 215E after it is

routed through IPv6 backbone 210. Ex.1005, ¶¶ 138-142. Ananda states that a “determination 651 is made as to whether the IPv6 data packet contains an IPv4 data packet in its payload.” Ex.1005, ¶ 113. Ananda also describes determining “whether the destination address of the IPv6 data packet is within the multi-protocol subnet.” Ex.1005, ¶ 114. Consistent with these teachings and Ananda’s use of virtual IPv6 destination addresses (Ex.1005, ¶¶ 82, 87), it would have been obvious for CPI 220E to identify that the IPv6 destination address included in the IPv6 header is a *constructed* address to facilitate directing the packet to the packet decapsulation process. Ex.1005, ¶ 109, 142; Ex.1003, ¶ 119.

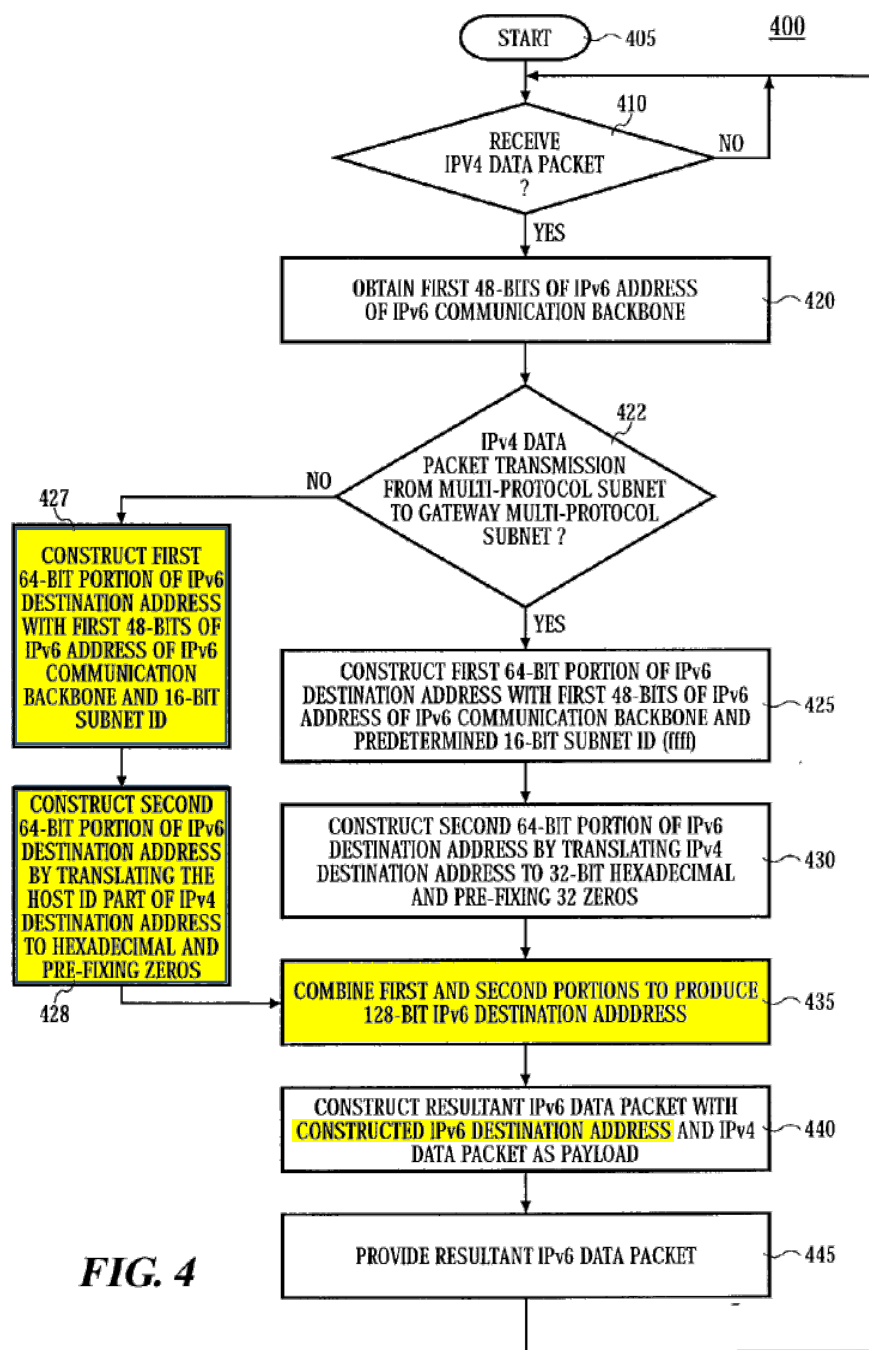


FIG. 4

Ex.1003, ¶ 118; Ex.1005, Fig. 4 (annotated)

Annotated Figure 4 above illustrates an encapsulation process 400 that is performed by the encapsulator 352 of CPI 215E (see Figure 3) of subnet 215, when

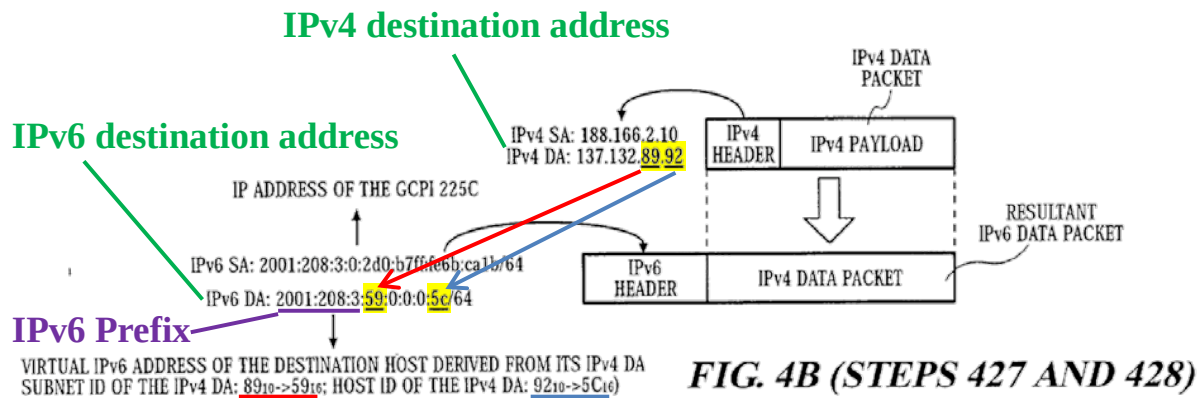
the CPI 215E receives an IPv4 packet from IPv4 host 215B (step 410). Ex.1005, ¶¶ 103, 99 (describing encapsulator 352). Ananda discloses that the CPI 215E supports dynamic encapsulation of IPv4 data packets, resulting in IPv6 data packets, where “the resultant IPv6 or outer encapsulation layer destination address is advantageously derived and **constructed from the destination address of the encapsulated IPv4 data packet**, dynamically.” Ex.1005, ¶ 116; Ex.1003, ¶¶ 120-122.

As shown in annotated Figure 4 above, the encapsulation process 400 of the encapsulator 352 of CPI 215E (see Fig. 3) includes steps 427, 428, and 435 in which an *IPv6 destination address* is *constructed*. Steps 427 and 428 correspond to the situation where CPI 215E and CPI 220E communicate IPv6 packets with each other over IPv6 backbone 210, as discussed above (“No” in step 422). Ex.1005, ¶¶ 103, 106. The IPv6 packet produced by CPI 215E is constructed to include an IPv6 destination address, as discussed below. Ex.1003, ¶ 123.

Ananda explains that in step 427, a first portion of the constructed IPv6 address is formed from a 48-bit IPv6 prefix associated with the IPv6 communication backbone (“*IPv6 prefix*”). Ex.1005, ¶ 106. Ananda explains that in step 428, a second portion of the constructed IPv6 address is formed from the original IPv4 destination address (“*an IPv4 destination address*”) of the IPv4 packet received from IPv4 host 215E. Ex.1005, ¶ 107; Ex.1003, ¶¶ 124-125.

As shown below with reference to annotated Figure 4B, Ananda discloses that the IPv6 destination address (DA) is constructed by combining (1) a 48-bit **IPv6 prefix**,⁸ and (2) the address of the IPv4 destination address. Ex.1005, ¶ 106, Fig. 4B. As shown in annotated Figure 4B below, “89.92” is part of the IPv4 destination address of IPv4 host 220B (“**IPv6 destination address**”). Ex.1005, ¶¶ 83-84. The IPv6 destination address constructed in Figure 4B includes (1) the IPv6 prefix concatenated with (2) two IPv6 address segments that are hexadecimal translations of the decimal values “89” and “92” taken from the IPv4 address of the IPv4 host 220B (i.e., **89 decimal = 59 hexadecimal**, and **92 decimal = 5c hexadecimal**). Ex.1005, ¶¶ 107-108, 138; Ex.1003, ¶¶ 126-127.

⁸ Ananda shows that “2001:208:3” is the IPv6 prefix for both the IPv6 source address for CPI 215E *and* the IPv6 destination address of IPv4 host 220B. Ex.1005, ¶ 138; Ex.1003, ¶ 127.



Ex.1003, ¶ 127; Ex.1005, Fig. 4B (annotated).

To the extent that Patent Owner argues that claim 1 requires incorporating all 32 bits of the IPv4 destination address into the constructed IPv6 destination address, Ananda renders obvious such an interpretation. Ex.1003, ¶ 128. Ananda teaches an alternative approach to constructing an IPv6 address as shown in Figure 4A, in which the constructed IPv6 address includes all 32 bits of the original IPv4 destination address. Ex.1005, ¶¶ 104 (steps 425 and 430 in Figs. 4 and 4A), 127; *see also* Ex.1003, ¶ 129 (explaining alternative approach).

Thus, Ananda discloses that *an IPv6 destination address is constructed by concatenating an IPv6 prefix and an IPv4 destination address*, as shown in Figure 4B above. Ex.1003, ¶ 130.

As discussed above, the CPI 215E encapsulates an IPv4 packet received from IPv4 host 215B into an IPv6 packet that is transmitted to CPI 220E over the IPv6 backbone 210. Ex.1005, ¶¶ 138-141; Ex.1003, ¶ 131.

Wetterwald teaches that it was known, when encapsulating IPv4 packets in IPv6 packets, to include IPv6 extension headers to identify the source and destination port numbers.

If desired, the translator 42 also **may add IPv6 option headers related to the layer 4 TCP/UDP source and destination ports**, as described in detail in RFC 2460.

Ex.1008, 8:24-28; Ex.1003, ¶¶ 134, 132-133, 135-137.

RFC4380 discloses that it was known to include a destination port number in a constructed IPv6 address. The Teredo protocol, for example, was used to provide connectivity and integration between IPv4 and IPv6 networks, like Ananda and Wetterwald. The Teredo protocol defines a “Teredo IPv6 address” that includes “the UDP port and IPv4 address at which a client can be reached.”

Ex.1009 at 5 (§ 2.13); Ex.1003, ¶ 138.

A POSITA would have been motivated to modify Ananda’s IPv6 address construction technique to incorporate a destination port number in the IPv6 address, as taught by Wetterwald and RFC4380, for the same reasons that the Teredo protocol incorporates a port number in an IPv6 address. For example, Teredo protocol uses the port number included in the IPv6 address to detect potentially fraudulent (“spoofed”) traffic:

3) If the source IPv6 address is a Teredo address, the client **compares the mapped IPv4 address and mapped port** in the source address **with the source IPv4 address and source port of the packet....**

5) If the source IPv6 address is a Teredo address, and the mapped IPv4 address and mapped port in the source address do not match the source IPv4 address and source port of the packet, the client checks whether there is an existing "local" entry for that IPv6 address....

6) **In the other cases**, the packet may be accepted, but **the client should be conscious that the source address may be spoofed**; before processing the packet, the client should perform the "direct IPv6 connectivity test" described in Section 5.2.9.

Ex.1009 at 22-23; Ex.1003, ¶ 142.

By comparing the destination port number identified in an IPv4 packet with the destination port number incorporated into an IPv6 destination address, the recipient of Ananda's encapsulated IPv6 packet (e.g., CPI 220E) could potentially detect spoofed traffic. It was well-known in the art that spoofed packets were potentially dangerous and problematic. *See, e.g.*, Ex.1016, ¶ 3 (describing the use of "spoofing" in network attacks); Ex.1017, ¶ 3 ("Attackers hide their location using incorrect or 'spoofed' IP source addresses"); Ex.1018, 1:57-61 ("unauthorized access is associated with 'IP spoofing'"). Thus, the desire to

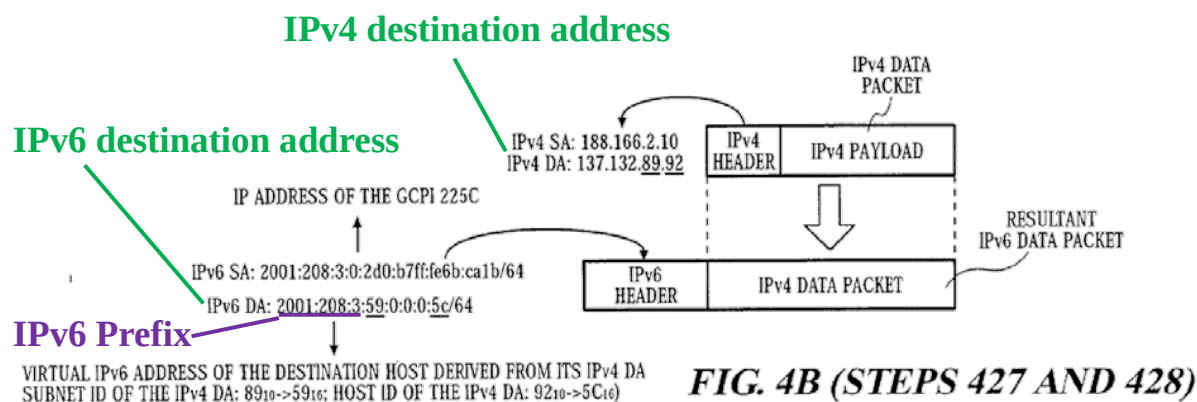
provide some level of protection in Ananda's mixed IPv4 and IPv6 network against spoofed traffic would have motivated a POSITA to incorporate an IPv4 destination port in Ananda's constructed IPv6 destination address. Ex.1003, ¶ 139.

A POSITA would have had an expectation of success in combining the technical features of Ananda, Wetterwald, and RFC4380, as discussed above, because there would have been ample room (e.g., available bits) in Ananda's constructed IPv6 address to include a 16-bit destination port number. As noted above, Ananda teaches prefixing zeros to the IPv4 address (which is 32 bits in length) to create a 64-bit "second portion" of the constructed IPv6 address. Ex.1005, ¶ 107 (step 428 in Fig. 4). Hence, Ananda's constructed IPv6 address includes 32 unallocated bits that Ananda suggests filling with zeros. A destination port number is just 16 bits (Ex.1009, at 12 ("16-bit port number")) and therefore a POSITA would have expected to have had success in including the port number as a substitute for some of the zero bits in Ananda's constructed IPv6 address. Ex.1003, ¶¶ 140-142.

Accordingly, a POSITA would have been motivated to modify Ananda's technique of constructing an IPv6 destination address by concatenating an IPv6 prefix with an IPv4 destination address, as taught by Ananda, and with a destination port number, as taught by Wetterwald and RFC4380. Ex.1003, ¶ 143. This modification represents the combination of prior art elements (constructing an

IPv6 prefix with an IPv4 destination address, as taught by Ananda, and with a destination port number, as taught by Wetterwald and RFC4380) according to known methods (concatenating the IPv6 prefix with the IPv4 destination address and destination port number) to yield the predictable result of including IPv6 extension headers in an IPv6 destination address to identify destination port numbers, as taught by Wetterwald and RFC4380. *See KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 401 (2007); Ex.1003, ¶ 143.

Ananda’s CPI 220E identifies an IPv6 destination address that is constructed as described above. In particular, Ananda discloses that the CPI 220E, upon receiving an IPv6 packet, determines whether the received IPv6 packet contains an IPv6 prefix. The CPI 220E identifies the *IPv6 destination address* included in the IPv6 header because it decapsulates IPv6 packet to produce the original IPv4 packet. Ex.1005, ¶ 142. As shown in annotated Figure 4B below, the IPv6 destination address of the IPv6 packet received by the CPI 220E is constructed to include an IPv6 prefix (“2001:208:3”), which is the IPv6 source address of the CPI 215E and the IPv6 destination address of IPv4 host 220B. Ex.1005, ¶ 106 (step 427 in Figure 4), 138 (step 1004 in Figure 10); Ex.1003, ¶ 144.



Ex.1003, ¶ 144; Ex.1005, Fig. 4B (annotated).

The CPI 220E thus identifies the IPv6 prefix in the IPv6 header, as the IPv6 prefix is the first part of the IPv6 destination address that the CPI 220E strips from the received IPv6 packet. Ex.1005, ¶ 142, 102; Ex.1003, ¶ 145. Further, the CPI 220E determines based on the IPv6 prefix whether the IPv6 packet is a virtual IPv6 address constructed as described above (i.e., “[constructed] IPv6 destination address”). In particular, Ananda discloses that the CPI 220E’s IPv6 input handler 332 determines how to process the IPv6 packet based on the IPv6 destination address. Ex.1005, ¶ 94. Further, Ananda explains that the CPI 220E determines whether the received IPv6 packet encapsulates an IPv4 data packet by “accessing an IPv6 address proxy 340 to **determine whether the destination address of the IPv6 data packet is a virtual IPv6 address of an IPv4 host 215B....**” Ex.1005, ¶ 96; Ex.1003, ¶¶ 146-147.

Thus, Ananda, Wetterwald and RFC4380 render obvious limitation [1.1].

Ex.1003, ¶ 148.

[1.2.1] if necessary, regularizing at least one of the IPv6 source address and the IPv6 destination address of the data packet by replacing said at least one IPv6 address,⁹

Ananda in view of Wetterwald renders obvious limitation [1.2.1] for the following reasons. Ex.1003, ¶ 149.

As described above in Section VIII.B, limitation [1.2.1] requires either (1) regularizing the IPv6 source address by replacing the IPv6 source address, or (2) regularizing the IPv6 destination address by replacing the IPv6 destination address. Ex.1003, ¶ 151.

⁹ The phrase “if necessary” makes limitation [1.2.1] a conditional limitation, because performance of the recited features is predicated on whether the condition “if necessary” is satisfied. Limitations [1.2.2] and [1.2.3] are conditional limitations because they relate to limitation [1.2.1]. See *Ex Parte Schulhauser*, Appeal No. 2013-007847, at 8-10 (PTAB Apr. 28, 2016) (precedential) (conditional steps in method claims need not be carried out when conditions precedent to those steps are not satisfied). Petitioner shows that all features of limitations [1.2.1]-[1.2.3], as well as the corresponding limitations [8.2.1]-[8.2.3] of claim 8, are obvious based on the assumption that such steps are “necessary.” Ex.1003, ¶ 150.

Ananda discloses that the CPI 220E, upon receiving the IPv6 packet, replaces the IPv6 destination address in the packet with an IPv4 destination address. Ex.1005, ¶ 142. The CPI 220E performs this replacement of the IPv6 destination address of the IPv6 packet by decapsulating the IPv6 packet to produce the original IPv4 packet transmitted from IPv4 host 215B. Ex.1005, ¶ 142 (step 1012 in Fig. 10). The decapsulation of the IPv6 packet received by the CPI 220E results in the destination address of the received IPv6 packet being **replaced** with (i.e., translated to) an IPv4 destination address. Ex.1005, ¶¶ 142-143, 137-138; Ex.1003, ¶ 153. In this way, the CPI 220E regularizes the destination address of the packet forwarded to the IPv4 host 220B. Ex.1003, ¶¶ 152-153.

With reference to annotated Figure 3 below, Ananda discloses that when the CPI 220E decapsulates the IPv6 packet to produce the IPv4 packet, an IPv4 input handler 322 of the CPI 220E accesses the CPI 220E's IPv4 routing table 324 to determine where to route the IPv4 packet. Ex.1005, ¶ 142, 90. Thus, Ananda discloses that the CPI 220E accesses its routing table 324 based on the IPv4 destination address of the decapsulated IPv4 packet, before transmitting the decapsulated IPv4 packet to its destination. Ex.1003, ¶ 154.

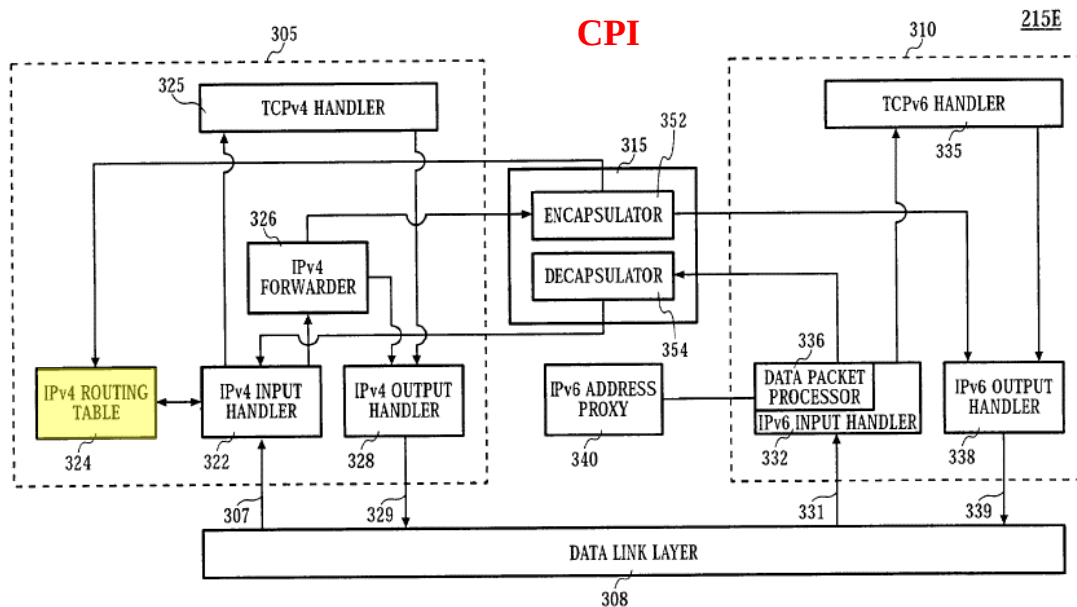


FIG. 3

Ex.1003, ¶ 154; Ex.1005, Fig. 3 (annotated).

Like the '845 patent and Ananda, Wetterwald is directed to providing interoperability between IPv4 nodes and IPv6 nodes across an IPv6 network. Ex.1008, 1:8-14. With reference to annotated Figure 2 below, Wetterwald discloses that an IPv4 terminal 18a in a first **IPv4 domain 12** transmits an IPv4 packet to another IPv4 terminal 22 in a second **IPv4 domain 16**. Ex.1008, 4:60-64, 5:9-13, 6:3-9. Like the '845 patent and Ananda, Wetterwald discloses that the IPv4 packet is translated into an IPv6 packet by an access point 30, the IPv6 packet is transmitted across an **IPv6 domain 14**, and a **translator 50** in clusterhead 32 translates the IPv6 packet to an IPv4 packet before it is routed to the destination IPv4 terminal 22. Ex.1008, 6:6:1-9, 4:28-56 (describing network configuration),

5:9-21 (assignment of IPv4 and IPv6 addresses), 6:38-60 and 7:37-41 (**translator 50** translates IPv6 packet to IPv4 packet); Ex.1003, ¶ 155. A **translation resource 42** of the access point 30 of Wetterwald likewise performs IPv4->IPv6 translation, similar to Ananda's CPI 215E. Ex.1005, ¶ 138; Ex.1008, 6:1-9, 6:16-22. Thus, like the CPI 220E of Ananda, the clusterhead 32 of Wetterwald receives an IPv6 packet in an **IPv6 domain 14** connected to an **IPv4 domain 16**. Ex.1003, ¶¶ 155-158.

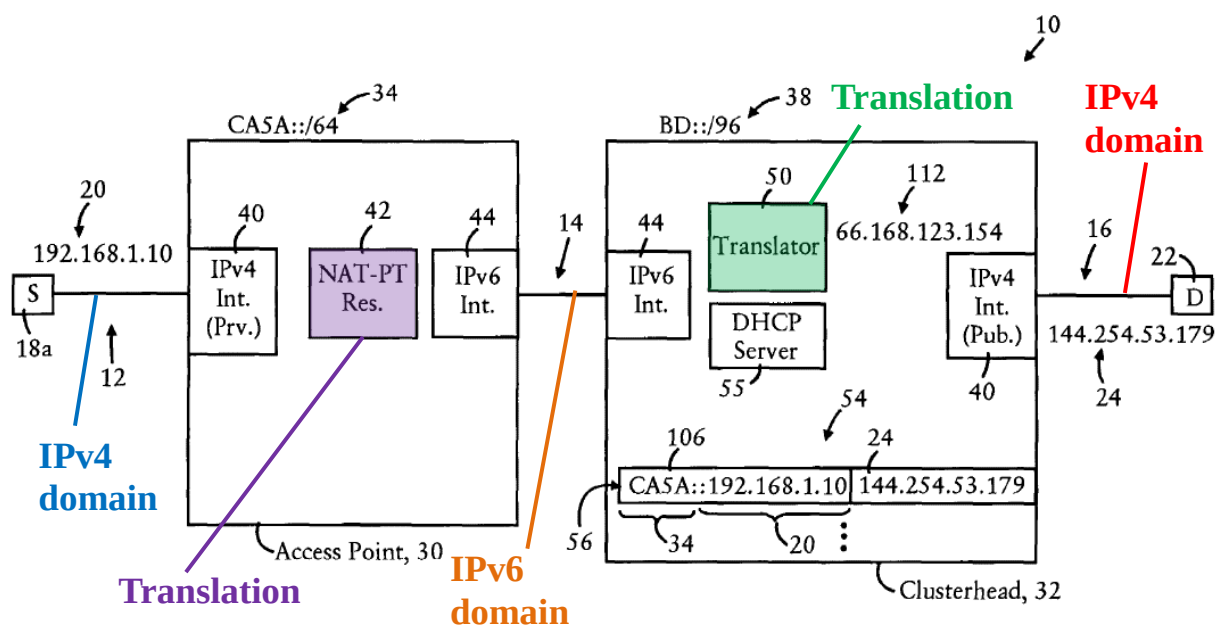


Figure 2

Ex.1003, ¶ 155; Ex.1008, Fig. 2 (annotated).

In addition to performing IPv6->IPv4 translation as shown in annotated Figure 2 above, the **translator 50** also includes an **IPv4 NAT/PAT translation resource 53** that translates between a public IPv4 address and a private IPv4 address, as shown in annotated Figure 5 below. Ex.1008, 7:1-21. Thus, after the

translation resource 51 of the **translator 50** translates a received IPv6 packet into an IPv4 packet (Ex.1008, 6:38-56, 7:37-41), the **translation resource 53** uses a NAT table 54 to translate between public and private IPv4 addresses, as described in RFC1918 (Ex.1007). Ex.1008, 7:1-21, 1:37-40; Ex.1003, ¶ 159.

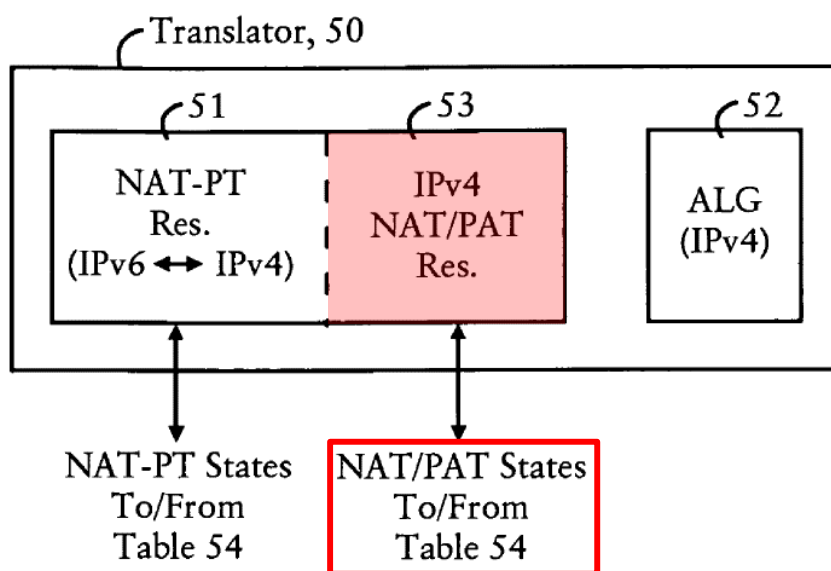


Figure 5

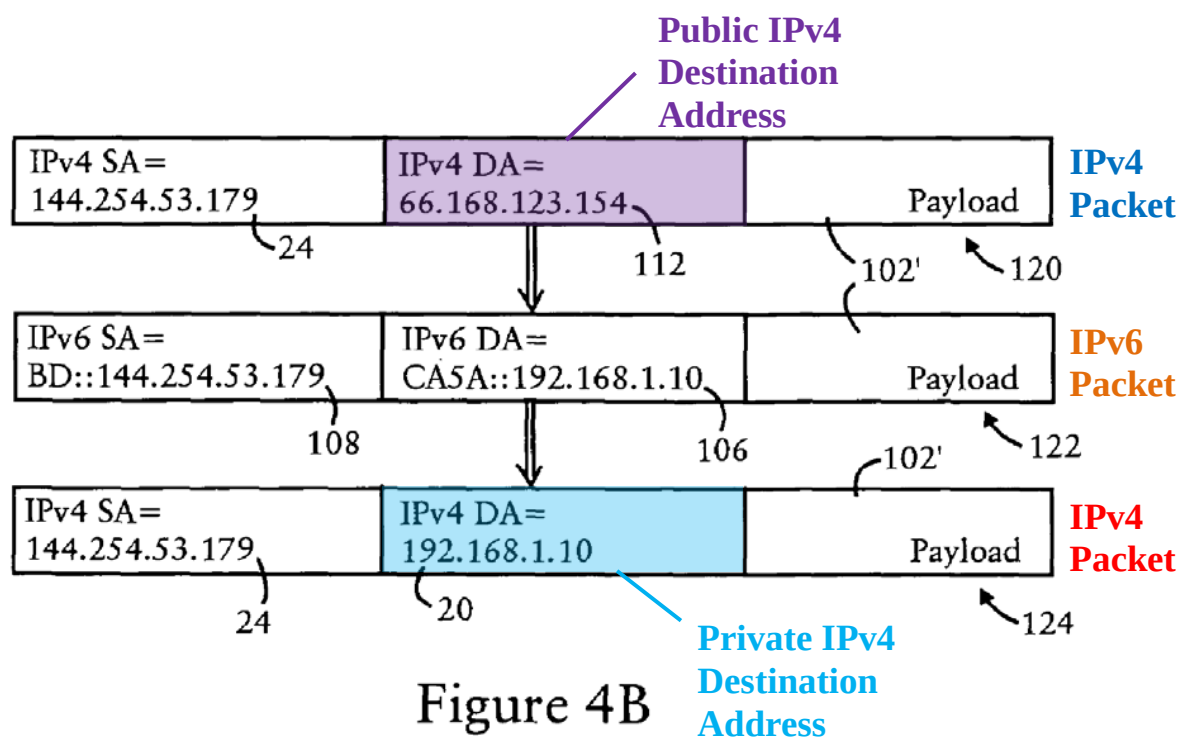
Ex.1003, ¶ 159; Ex.1008, Fig. 5 (annotated).

Using the NAT table 54, Wetterwald’s **translator 50** also performs port address translation (PAT). Ex.1008, 1:7-14. Wetterwald explains that the **translation resource 53** includes PAT functionality that enables the clusterhead 32 to “reuse an assigned public address 112 for multiple connections on the IPv4 network 16,” by translating between the assigned public address 112 and a private

IPv4 address. Ex.1008, 7:1-7; Ex.1003, ¶ 160.

Annotated Figure 4B of Wetterwald below discloses an example of a **public IPv4 destination address 112** (66.168.123.154) contained in the **IPv4 packet 120** that is translated to the **IPv6 packet 122**, and a **private IPv4 destination address 20** contained in the **IPv4 packet 124** that is translated from the **IPv6 packet 122**.

Ex.1008, 6:61-67, 7:30-36 (translation resource 53 has “reverse NAT functionality” corresponding to response packet illustrated in Figure 4B); Ex.1003, ¶ 160.



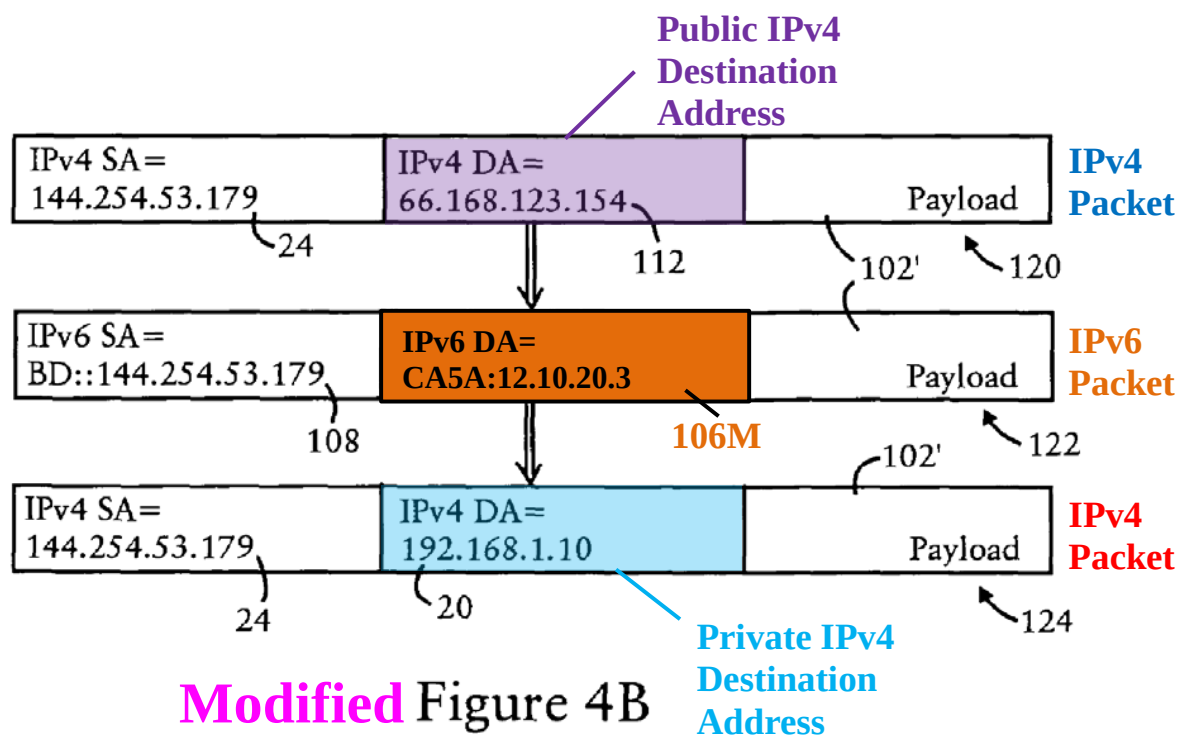
Ex.1003, ¶ 160; Ex.1008, Fig. 4B (annotated).

Wetterwald discloses that the NAT/PAT table entry 54 specifies an

association between (1) the translated IPv4 private and public addresses (e.g., an association between **private IPv4 address 20** (192.168.1.10) and **public IPv4 address 112** (66.168.123.154) 112 in Figure 4B), and (2) the originally-specified and translated TCP/UDP port values (e.g., “80” and “1”). Ex.1008, 7:7-19. Thus, the PAT functionality in the **translation resource 53** enables the gateway to use the same public IPv4 address for 2^{16} (i.e., 65,536) distinct connections. Ex.1008, 7:7-21; Ex.1003, ¶ 161.

In Wetterwald’s Figure 4B above, which shows the response packet, the **private IPv4 address 20** (192.168.1.10) is concatenated with the IPv6 prefix (CA5A::/64) 34 of the access point 30 (see Figure 2) in the IPv6 destination address 106 of the IPv6 packet 122. Wetterwald explains that the clusterhead 32 previously stored the association between the **private IPv4 address 20** and the IPv6 prefix of the access point 30 when the clusterhead received the IPv6 packet 104 shown in Figure 4A. Ex.1008, 8:51-62. A POSITA would have been motivated to follow the guidance in RFC1918 that packets with private addresses should not be forwarded over public links (e.g., IPv6 domain 14 in Wetterwald). As discussed above, Wetterwald’s **translation resource 53** performs public-to-private IPv4 translations as described in RFC1918. Ex.1008, 7:1-3; Ex.1003, ¶ 162. Thus, a POSITA would have been motivated to modify Figure 4B as shown below, where the **modified IPv6 destination address 106M** includes the IPv6 prefix of

the access node 30 (CA5A::/64) concatenated with a public IPv4 address (12.10.20.3) assigned to the access node 30,¹⁰ rather than the **private IPv4 address 20** (192.168.1.10) assigned to the IPv4 host 18a, as illustrated in Figure 2; Ex.1003, ¶¶ 162, 164-165.



Ex.1003, ¶ 162; Ex.1008, Fig. 4B (annotated).

The public-to-private IPv4 translation of the destination address in the IPv4

¹⁰ The public IPv4 address 12.10.20.3 is an address specified by Dr. Min.

Wetterwald does not explicitly assign IPv4 addresses to the access nodes 30.

Ex.1003, ¶ 162 n.2.

packet 124 would then be performed, as shown in modified Figure 4B, and as taught by Wetterwald (i.e., replacement of the public IPv4 destination address (12.10.20.3) assigned to the access node 30 with the private IPv4 address (192.168.1.10) assigned to the IPv4 host 18a). Ex.1008, 7:1-21; Ex.1003, ¶ 163.

Ananda is concerned with the shortage of IPv4 addresses. Ex.1005, ¶ 6. Further, Ananda discloses that the IPv4 host 215B in subnet 215 and IPv4 host 220B in subnet 220 may be assigned private IPv4 addresses. Ex.1005, ¶¶ 84 (IPv4 hosts may be assigned private IPv4 addresses), 82 (IPv4 host 215B and IPv4 host 220B are “purely IPv4.”). Utilizing the public-to-private IPv4 address translation technique of Wetterwald after the CPI 220E has decapsulated a received IPv6 packet into an IPv4 packet would help mitigate the overuse of public IPv4 addresses, because one public IPv4 address can be used to provide communications for 2^{16} (i.e., 65,536) distinct private IPv4 addresses (“*native address[es]*”). Ex.1003, ¶¶ 166-169.

A POSITA would have been motivated to modify Ananda to include a NAT table (like Wetterwald’s NAT 54) to associate public and private IPv4 addresses so that the CPI 220E would perform public-to-private IPv4 translation, just like the **translation resource 53** of Wetterwald. This modification would permit IPv4 host 215B and IPv4 host 220B to use private IPv4 addresses (Ex.1005, ¶ 82 (IPv4 hosts may be assigned private addresses)), and not reveal their private addresses when

transmitting packets outside their respective subnet 215, 220, as disclosed by RFC1918. Ex.1007, § 3, 5 (packets with private source or destination addresses should not be forwarded over public links). Wetterwald's **translation resource 53** performs public-to-private IPv4 translations as described in RFC1918. Ex.1008, 7:1-3; Ex.1003, ¶ 170.

For example, when the IPv4 host 215B transmits an IPv4 packet to IPv4 host 220B (Ex.1005, ¶ 137), the IPv4 host 215B would transmit the IPv4 packet using a public IPv4 address of the host 220B as the destination address if the IPv4 host 220B is using a private address that is unknown to devices outside the subnet 220 (see Ex.1007, § 2, 2-3) (private addresses are ambiguous between different enterprises). In this case, the public IPv4 address of the IPv4 host 220B would be the IPv4 address of the CPI 220E, so that the private IPv4 address of the IPv4 host 220B remains private. Ex.1007, § 3, 5 (packets with private source or destination addresses should not be forwarded over public links); Ex.1003, ¶ 171.

The CPI 215E, when constructing the IPv6 destination address as described above with respect to limitation [1.1] (*see, e.g.*, Ex.1005, ¶¶ 106-107, 105), would then utilize the public IPv4 address of the IPv4 host 220B (i.e., the IPv4 address of the CPI 220E) as the IPv4 destination address when constructing the IPv6 packet. After decapsulating the received IPv6 packet to produce the IPv4 packet transmitted by the CPI 215E, the CPI 220E would perform public-to-private IPv4

translation, like the **translation resource 53** of Wetterwald, to translate the public IPv4 destination address of the decapsulated IPv4 packet (e.g., the public IPv4 address of the CPI 220E) to the private IPv4 address of the IPv4 host 220B. Ex.1008, 7:1-21; Ex.1003, ¶ 172.

Likewise, as explained in Dr. Min's declaration, if IPv4 host 215B is using a private IPv4 address, the CPI 215E may perform a similar private-to-public IPv4 translation, as disclosed by Wetterwald, using the public IPv4 address of the CPI 215E as the source address of the IPv4 packet that the CPI 215E encapsulates into an IPv6 packet. Ex.1005, ¶ 138; Ex.1003, ¶ 173. This modification of Ananda's CPI 215E and CPI 220E using their public IPv4 addresses instead of the private IPv4 address of their respective IPv4 hosts 215B, 220B is consistent with RFC1918. Ex.1007, § 3 (p. 4) (private hosts have access to external services via mediating gateways). Ex.1003, ¶ 174.

As described above, RFC1918 advises that private addresses should not be forwarded over public links. Ex.1007, § 3, 5. Based on this guidance, a POSITA would have been motivated to modify the IPv4 routing technique of Ananda to utilize Wetterwald's public-to-private network address translation so that private IPv4 addresses are not utilized in IPv6 packets transmitted over Wetterwald's IPv6 network 14. Ex.1003, ¶ 175. Further, a POSITA would have been motivated to utilize Wetterwald's public-to-private IPv4 translation in Ananda, because one

public IPv4 address in an IPv4 packet received by Ananda's CPI 220E (e.g., an IPv6 packet that the CPI 220E translates into an IPv4 packet) can be used for 2^{16} private addresses for IPv4 hosts in the subnet 220. Ex.1003, ¶ 176. Ananda explains that the IPv4 hosts (e.g., IPv4 host 220B) are "purely IPv4 hosts" that may be assigned private IPv4 addresses. Ex.1005, ¶¶ 82, 84. This would have involved the combination of prior art elements (Ananda's use of an IPv4 routing table with Wetterwald's NAT/PAT translation resource 53) according to known methods (translating an IPv4 address in the header of an IPv4 packet decapsulated by the CPI 220E into one of 2^{16} private IPv4 addresses) to yield the predictable result of translating a public IPv4 address into a private IPv4 address, as taught by Wetterwald. *See KSR*, 550 U.S. at 401; Ex.1003, ¶ 176.

Furthermore, the obvious combination of Ananda and Wetterwald would not have required any modification to Ananda's network architecture because Ananda discloses that IPv4 host 215B in subnet 215 and IPv4 host 220B in subnets 220 may be assigned private addresses. Ex.1005, ¶ 84; Ex.1003, ¶ 177.

As described above with respect to limitation [1.1], Wetterwald teaches that it was known, when encapsulating IPv4 packets in IPv6 packets, to include IPv6 extension headers to identify the source and destination port numbers (Ex.1008, 8:24-28), and RFC4380 discloses that it was known to include a destination port number in a constructed IPv6 address. Ex.1009, 22-23. Wetterwald discloses that

the translation resource 53 translates between public and private IPv4 addresses based on port values specified in the IPv4 packet. Ex.1008, 7:7-19. Therefore, a POSITA would have had an expectation of success in modifying Ananda's CPI 220E to perform a public-to-private IPv4 translation after decapsulating a received IPv6 packet, because Wetterwald discloses that the public-to-private IPv4 translations are performed based on port values specified in the IPv4 packet. Ex.1008, 7:7-19; Ex.1003, ¶ 178. As described above with respect to limitation [1.1], it would have been obvious to modify Ananda to utilize an IPv6 destination address that is constructed by concatenating an IPv6 prefix, an IPv4 destination address, and a destination port number, as taught by Wetterwald and RFC4380. Thus, a POSITA would have had a reasonable expectation of success in making the above modifications to Ananda because the public-to-private IPv4 translation would be performed based on the port values included in the packet, as taught by Wetterwald. Ex.1008, 7:7-19; Ex.1003, ¶ 179.

Thus, the prior art renders obvious limitation [1.2.1]. Ex.1003, ¶ 180.

[1.2.2] the replacement process belonging to a group comprising replacement of a constructed address with a native address and replacement of a native address with a constructed address, and

Ananda in view of Wetterwald renders obvious limitation [1.2.2]. Ex.1003, ¶ 181.

Initially, it is noted that limitation [1.2.2] recites two features in the

alternative by reciting the phrase “belonging to a group comprising....” Thus, limitation [1.2.2] requires that the replacement process of limitation [1.2.1] comprises (1) replacement of a constructed address with a native address, or (2) replacement of a native address with a constructed address. As described above, Ananda and Wetterwald render obvious limitation [1.2.1]. Ex.1003, ¶¶ 182.

As described above with respect to limitation [1.2.1], Ananda discloses that when the CPI 220E decapsulates the IPv6 packet (step 1012 in Fig. 10) received from the CPI 215E, the CPI 220E replaces the IPv6 destination address with the IPv4 destination address of the IPv4 host 220B. Ex.1005, ¶ 142; Ex.1003, ¶¶ 183.

The IPv6 destination address contained in the IPv6 packet received from the CPI 215E is a constructed address. Ex.1003, ¶¶ 184. In particular, the IPv6 destination address in the IPv6 packet is an IPv6 virtual address of IPv4 host 220B (Ex.1005, ¶ 138) that is constructed in steps 427, 428, 435 and 440 of annotated Figure 4 below. Ex.1005, ¶¶ 106-107, 105; Ex.1003, ¶¶ 184-185.

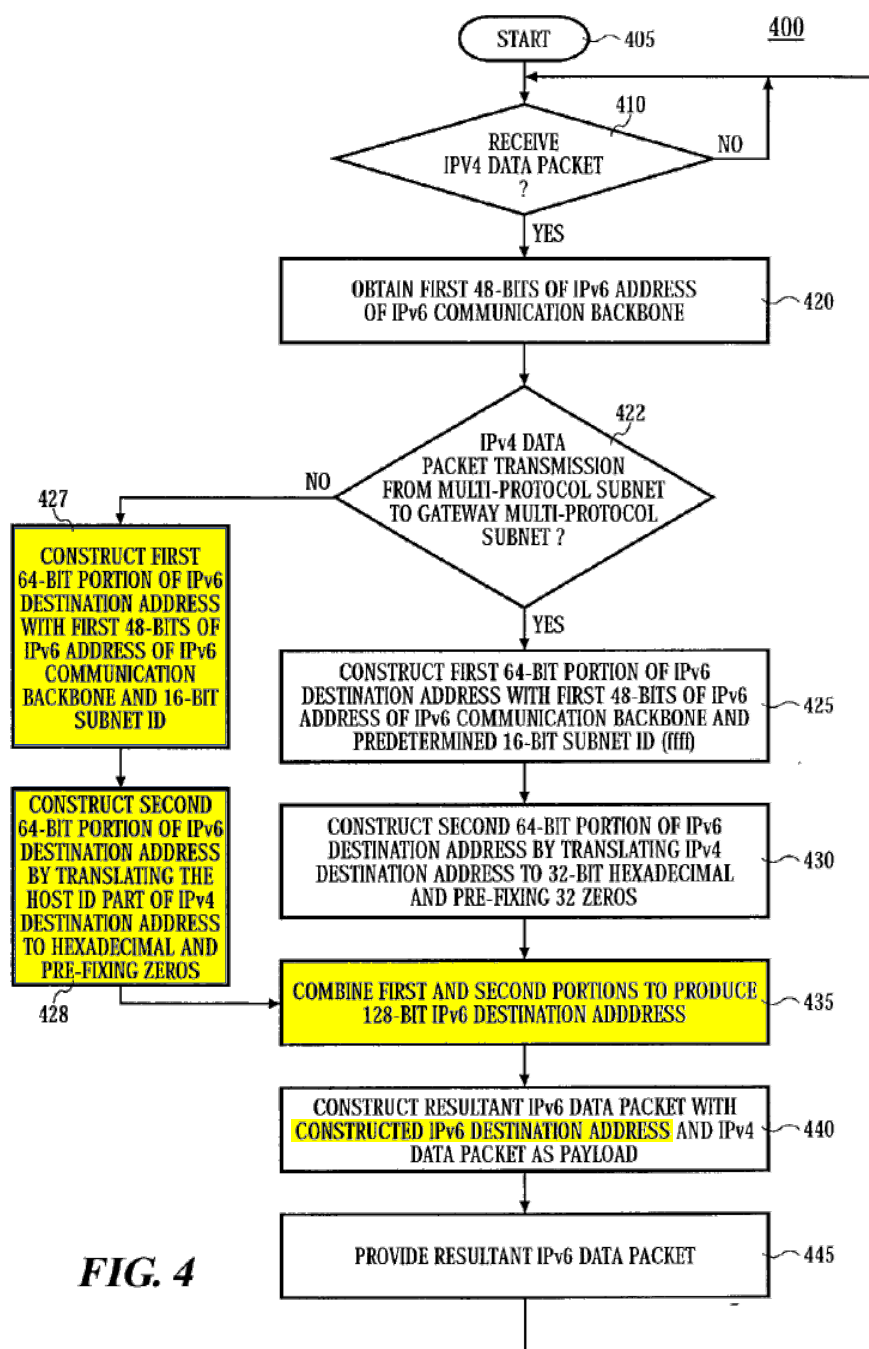


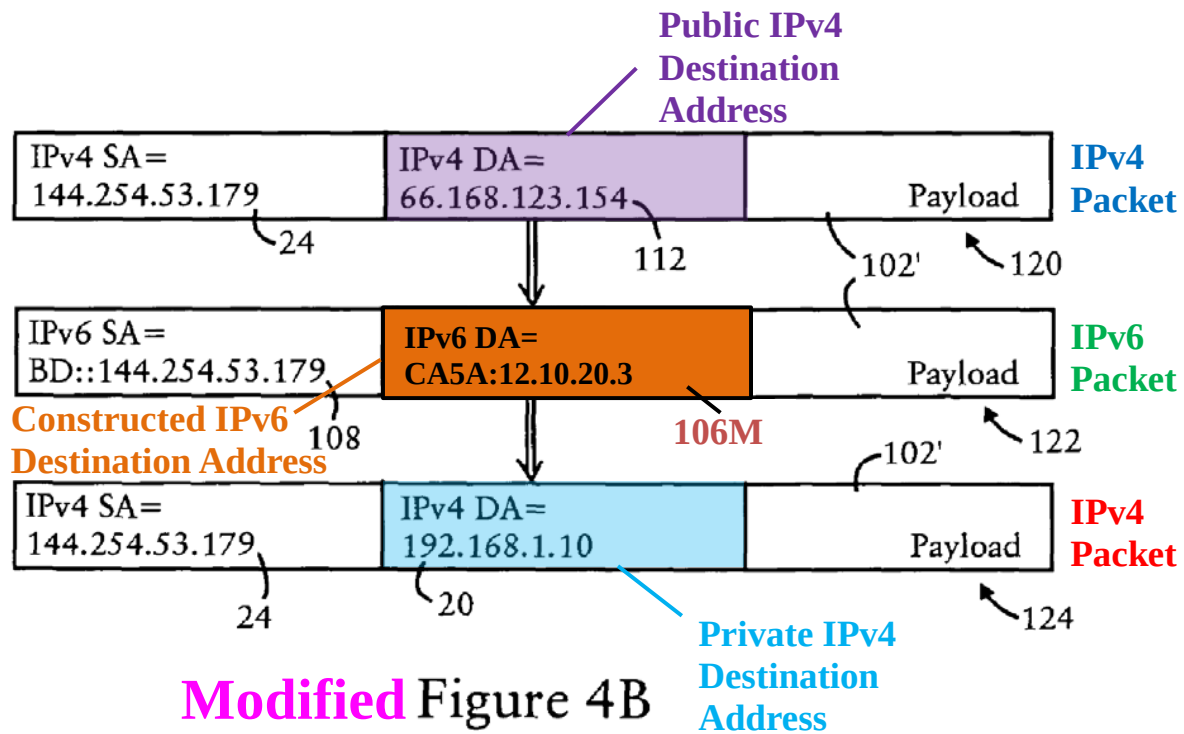
FIG. 4

Ex.1003, ¶ 185; Ex.1005, Fig. 4 (annotated).

As described above with respect to limitation [1.2.1], a POSITA would have been motivated to modify Ananda such that the CPI 220E, after decapsulating the

received IPv6 packet to produce the original IPv4 packet (Ex.1005, ¶ 142) would use a translation resource, like Wetterwald's translation resource 53, to translate the public IPv4 destination address in the decapsulated IPv4 packet to a private IPv4 address of the IPv4 host 220B. Ex.1003, ¶ 186.

This obvious modification would involve replacement of the constructed IPv6 address ("*constructed address*") contained in the IPv6 packet received by the CPI 220E with the private IPv4 address ("*native address*") of the IPv4 host 220B. Annotated Figure 4B of Wetterwald below, as modified as described above with respect to limitation [1.2.1], illustrates replacement of a **constructed IPv6 destination address 106M** with a **private IPv4 destination address 20** in the **IPv4 packet 124** that is translated from the **IPv6 packet 122**. Ex.1008, 6:50-54, 6:61-67, 9:11-32. As shown in annotated Figure 4B, the **private IPv4 destination address 20** in the **IPv4 packet 124** differs from the **public IPv4 destination address 112** contained in the **original IPv4 packet 120** that was translated into the **IPv6 packet 122**. Ex.1008, 6:61-67; Ex.1003, ¶ 187.



Ex.1003, ¶ 187; Ex.1008, Fig. 4B (annotated).

Accordingly, Ananda in view of Wetterwald renders obvious the claimed feature of the replacement process comprising *replacement of a constructed address with a native address*, as recited in limitation [1.2.2]. Ex.1003, ¶ 188.

[1.2.3] modifying the data packet using the regularized address; and

Ananda and Wetterwald render obvious limitation [1.2.3]. Ex.1003, ¶ 189.

As described above with respect to limitation [1.2.1], a POSITA would have been motivated to modify Ananda's CPI 220E to use a translation resource to translate the public IPv4 destination address in the decapsulated IPv4 packet to a private IPv4 address of the IPv4 host 220B. Ex.1003, ¶ 190.

Ananda discloses that when the CPI 220E receives the IPv6 packet transmitted by the CPI 215E, the decapsulator 354 of the CPI 220E decapsulates the IPv6 header to produce the original IPv4 data packet. Ex.1005, ¶ 142. Annotated Figure 3 below shows the decapsulator 354 of the CPI 220E. Ex.1005, ¶¶ 142, 102; Ex.1003, ¶ 191.

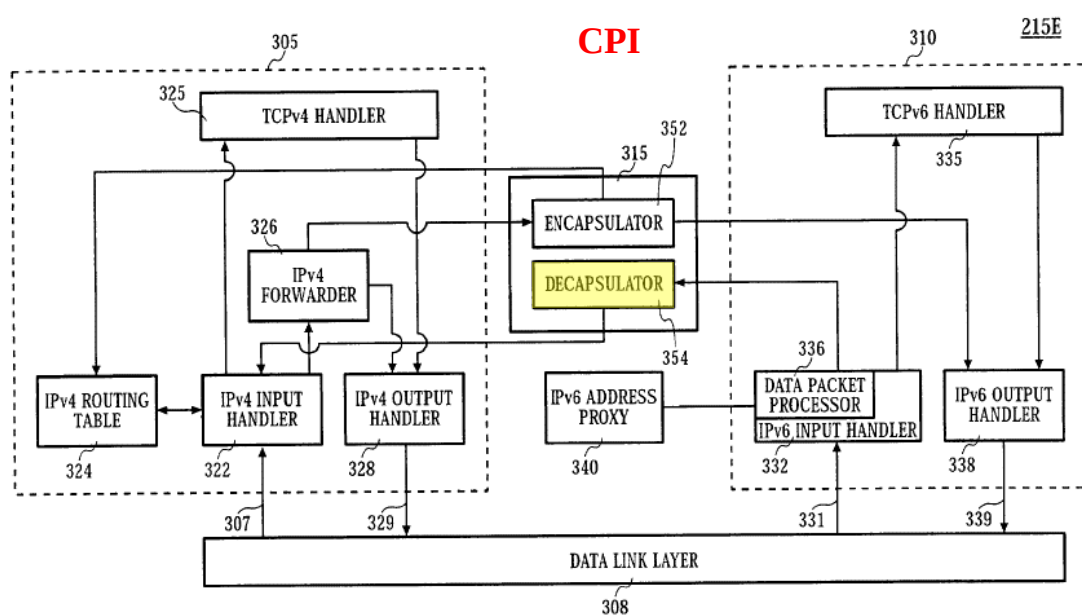


FIG. 3

Ex.1003, ¶ 191; Ex.1005, Fig. 3 (annotated).

Annotated Figure 5 of Ananda below shows the decapsulation process performed by the decapsulator 354. Ananda explains that extracting (step 515) an IPv4 data packet from the payload of an IPv6 data packet involves stripping the IPv6 header from the IPv6 data packet. Ex.1005, ¶ 109. Stripping the IPv6 header from the packet results in modifying the packet. Ex.1003, ¶ 192.

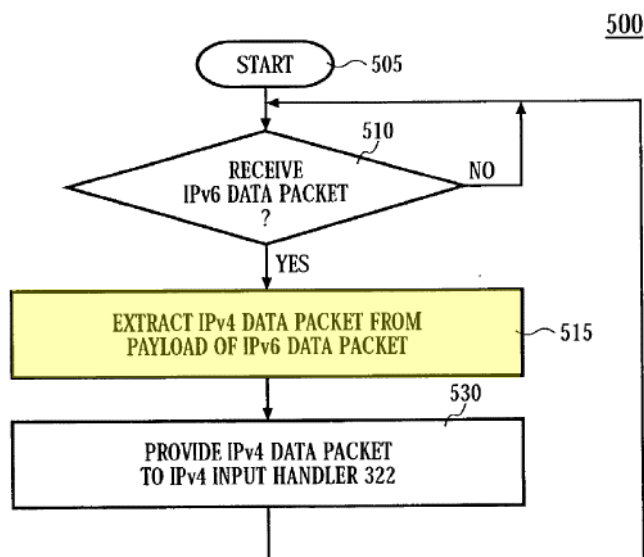
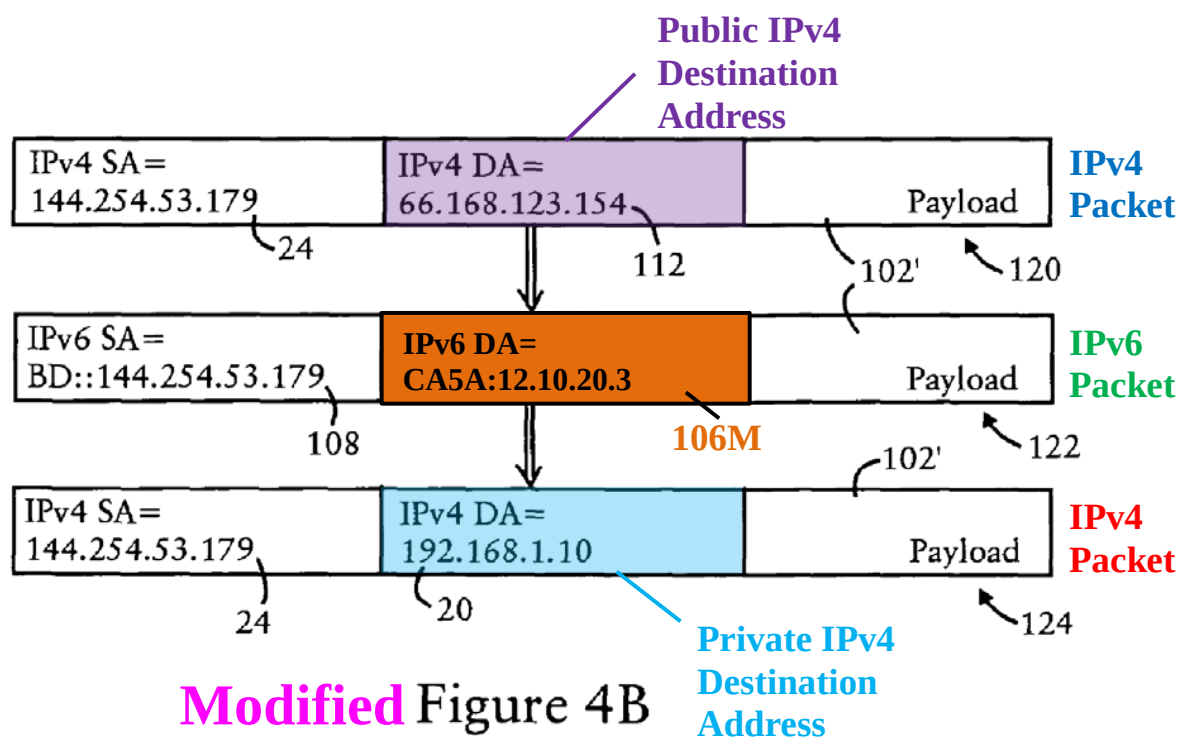


FIG. 5

Ex.1003, ¶ 192; Ex.1005, Fig. 4 (annotated).

Replacing the IPv4 destination address in the decapsulated IPv4 packet with the private IPv4 address of the IPv4 host 220B involves modifying the IPv6 packet with the private IPv4 address of the IPv4 host (“*regularized address*”) because the destination address of the IPv6 packet would be replaced with the private IPv4 address of the IPv4 host 220B, as explained above with respect to limitation [1.2.2]. For example, as shown in annotated Figure 4B below, which is modified as described above with respect to limitation [1.2.1], utilizing Wetterwald’s translation resource 53 in Ananda’s CPI 220E would involve (1) stripping the IPv6 header, as taught by Ananda, and (2) replacing the IPv4 destination address in the

decapsulated IPv4 packet with the private IPv4 destination address of the destination terminal (e.g., IPv4 host 220B), as taught by Wetterwald. Ex.1005, ¶¶142, 109; Ex.1008, 6:61-67, 9:11-32; Ex.1003, ¶ 193.



Ex.1003, ¶ 193; Ex.1008, Fig. 4B (annotated).

As shown in annotated Figure 4B of Wetterwald above, the **private IPv4 destination address 20** (e.g., the private IPv4 address of Ananda's IPv4 host 220B) in the **IPv4 packet 124** differs from the **public IPv4 destination address 112** (e.g., the public IPv4 address of Ananda's CPI 220E) contained in the **original IPv4 packet 120** that was translated into the **IPv6 packet 122**. Ex.1008, 6:61-67; Ex.1003, ¶ 194.

Thus, Ananda in view of Wetterwald renders obvious limitation [1.2.3].

Ex.1003, ¶ 195.

[1.3] routing the IPv6 data packet to its destination.

Ananda renders obvious limitation [1.3]. Ex.1003, ¶ 196.

As shown in annotated Figure 10 below, Ananda discloses that the CPI 220E receives the IPv6 packet transmitted from CPI 215E as it is routed through the IPv6 backbone 210 by routers 210A and 210B. Ex.1005, ¶¶ 138-142 (steps 1006, 1008 and 1010 in Fig. 10). Then, the CPI 220E decapsulates the IPv6 header to produce the IPv4 packet that was encapsulated in the IPv6 packet. Ex.1005, ¶ 142 (step 1012 in Fig. 10). The CPI 220E then routes the IPv4 packet to the IPv4 host 220B, which is the *destination* of the packet. EX1005, ¶ 143 (step 1014); Ex.1003, ¶ 197.

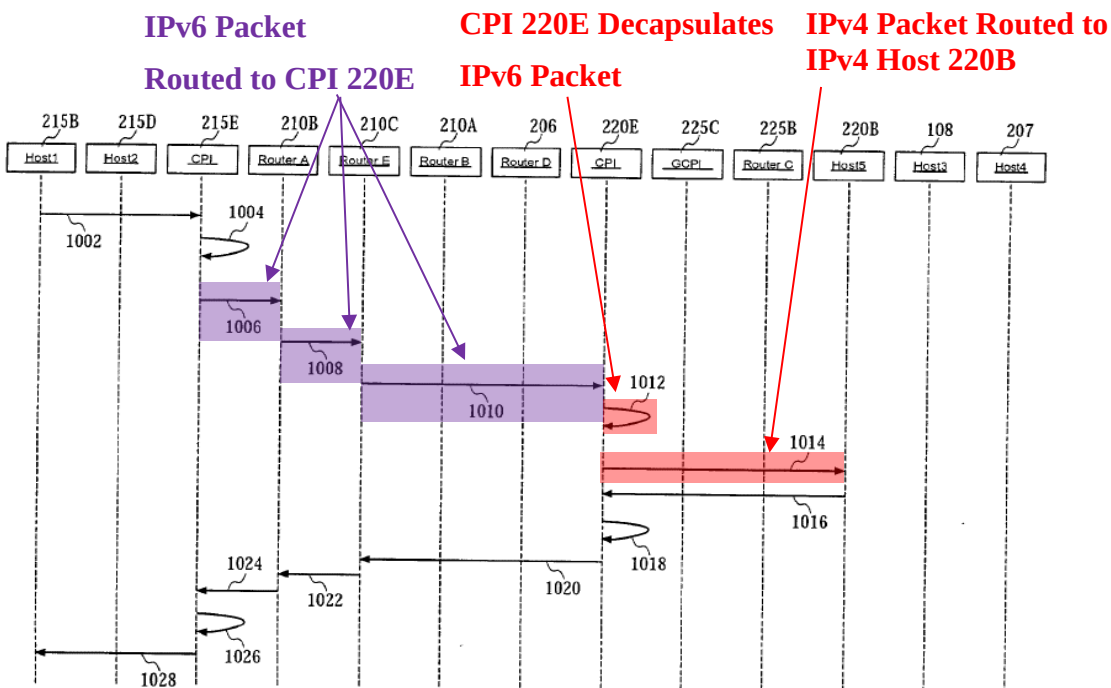


FIG. 10

Ex.1003, ¶ 197; Ex.1005, Fig. 10 (annotated).

By receiving the IPv6 packet, translating the IPv6 packet into an IPv4 packet, and routing the translated IPv4 packet to its destination (the IPv4 host 220B), the CPI 220E routes the IPv6 data packet to its destination. Therefore, Ananda renders obvious limitation [1.3]. Ex.1003, ¶ 198.

6. Claim 5

[5.0] The method according to claim 1, wherein when the data packet comes from the IPv4 domain and goes to said user terminal, the step of regularizing the destination address comprises the following steps:

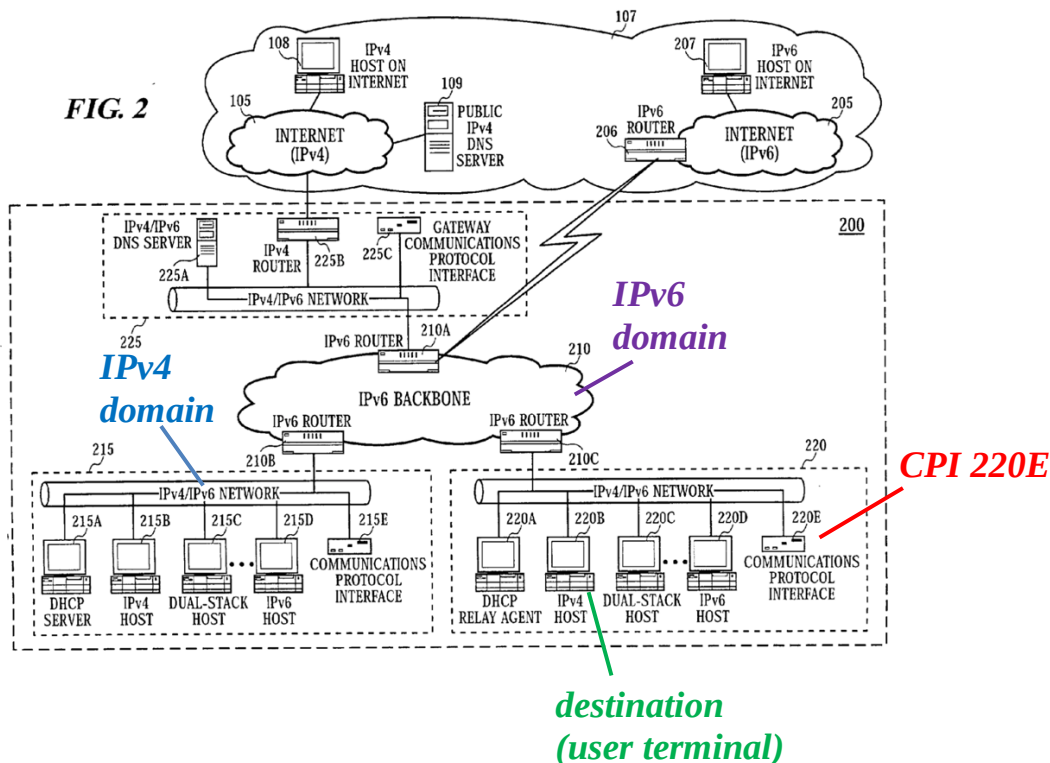
Ananda in view of Wetterwald renders obvious limitation [5.0]. Ex.1003, ¶

199.

With reference to annotated Figure 2 below, Ananda discloses that the CPI 220E of multi-protocol IPv4/IPv6 subnet 220 receives an IPv6 packet in an IPv6 backbone 210 (“*IPv6 domain*”), which is connected to a multi-protocol IPv4/IPv6 subnet 215 (“*IPv4 domain*”), where the IPv6 packet was originally transmitted as an IPv4 packet from the IPv4 host 215B in the subnet 215 (“*IPv4 domain*”).

Ex.1005, ¶¶ 137-142, 79. When the CPI 220E receives an IPv6 packet from CPI 215E, the CPI 220E translates the IPv6 packet into an IPv4 packet and transmits the IPv4 packet to IPv4 host 220B (“*user terminal*”), the destination of the IPv4 packet transmitted by the CPI 215B. Ex.1005, ¶¶ 137, 142; Ex.1003, ¶¶ 200-202.

Thus, Ananda renders obvious limitation [5.0]. Ex.1003, ¶ 203.



Ex.1003, ¶ 200; Ex.1005, Fig. 2 (annotated).

[5.1] searching an address translation table of the gateway for an entry associating with said constructed destination address a native destination address;

As described above with respect to limitations [1.2.1]-[1.2.3], a POSITA would have been motivated to modify Ananda so that the CPI 220E, after decapsulating the received IPv6 packet, utilizes a translation resource to replace a public IPv4 destination address in the decapsulated IPv4 packet with a private IPv4 destination address (e.g., the private IPv4 address of the IPv4 host 220B). Ex.1003, ¶ 207. Wetterwald describes that the translation resource 53 uses a NAT table 54 (*address translation table of the gateway*) to translate between a public IPv4

address and a private IPv4 address, as shown in annotated Figure 5 below.

Ex.1008, 7:1-21. Ex.1003, ¶ 208.

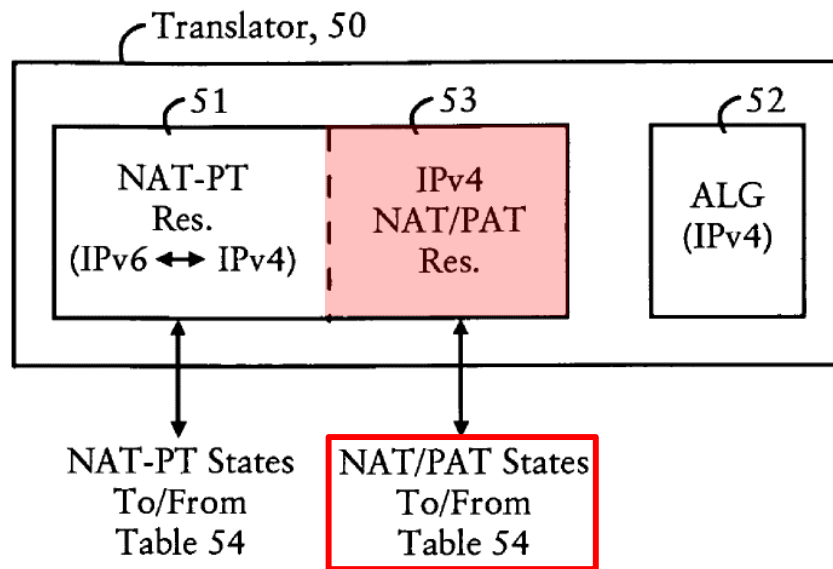


Figure 5

Ex.1003, ¶ 208; Ex.1008, Fig. 5 (annotated).

In the combination, an address translation table (like Wetterwald’s translation table 54) would be incorporated into Ananda’s CPI 220E. The translation table would include an association between private IPv4 addresses (“*native destination address*”) and public IPv4 addresses contained in constructed IPv6 addresses, allowing Ananda’s CPI 220E to perform the public-to-private translation of IPv4 addresses as taught by Wetterwald’s **translation resource 53**. Ex.1008, 7:1-21. In the combination, the constructed IPv6 addresses received by CPI 220E correspond

to the claimed “*said constructed destination address.*” Ex.1003, ¶¶ 209, 204-206.

Accordingly, Ananda in view of Wetterwald renders obvious limitation [5.1].
Ex.1003, ¶ 210.

[5.2] if an entry is found, replacing the constructed destination address of the data packet with the native destination address from that entry.

Ananda and Wetterwald render obvious limitation [5.2]. Ex.1003, ¶ 211.

As described above with respect to limitations [1.2.1] and [5.2], Wetterwald discloses that the **translation resource 53** uses its NAT table 54 to translate between public and private IPv4 addresses. Ex.1008, 7:1-21; Ex.1003, ¶ 212.

As described above with respect to limitations [1.1] and [1.2.2], Ananda discloses that the IPv6 packet is constructed to include hexadecimal translations of the IPv4 destination address contained in the original IPv4 packet. Ex.1005, ¶¶ 106-108, 105 (above-described construction shown in Figure 4B). Further, as described above with respect to limitation [1.2.1], Ananda’s CPI 220E would utilize a translation table, similar to Wetterwald’s NAT 54 shown in annotated Figure 5 below, to perform the public-to-private IPv4 translation after the CPI 220E has decapsulated the IPv6 packet to produce the IPv4 packet. Ex.1005, ¶ 142; Ex.1008, 6:38-67; Ex.1003, ¶ 213.

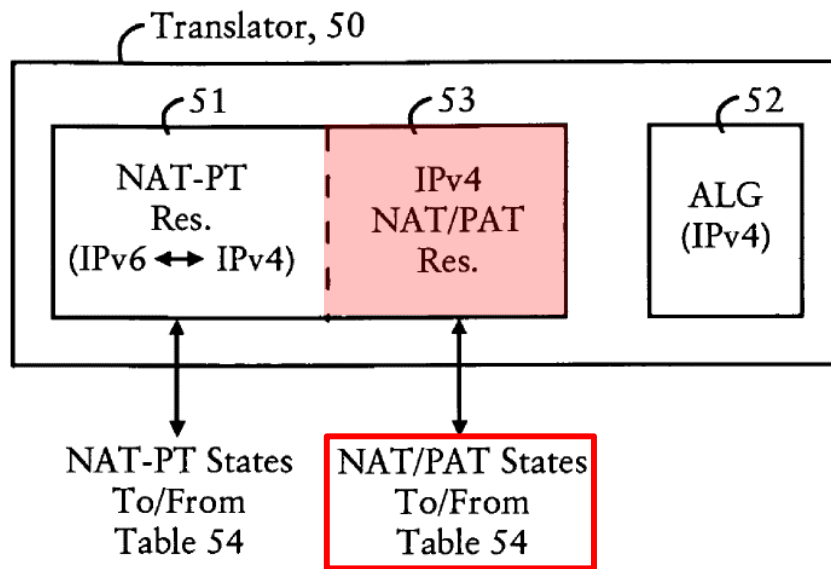


Figure 5

Ex.1003, ¶ 213; Ex.1008, Fig. 5 (annotated).

Therefore, in the obvious modification of Ananda’s CPI220E using a translation resource, Ananda’s CPI 220E would replace the public IPv4 address contained in the constructed IPv6 packet with a private IPv4 address (“*native address*”) of the destination terminal (e.g., IPv4 host 220B), as taught by Wetterwald’s public-to-private IPv4 translation. Ex.1003, ¶¶ 214-215.

Thus, Ananda in view of Wetterwald renders obvious limitation [5.2]. Ex.1003, ¶ 216.

7. Claim 6

[6.0] The method according to claim 5, further comprising a step of verifying that the destination port number of the data packet and the destination port number contained in the constructed destination address match.

Ananda in view of Wetterwald and RFC4380 renders obvious limitation [6.0]. Ex.1003, ¶ 217.

As discussed with respect to limitation [1.2.1], it would have been obvious to a POSITA to compare the destination address and port number values within a constructed IPv6 destination address to the corresponding values in the header of an encapsulated IPv4 packet. Ex.1003, ¶ 218. Such a comparison would be useful, for example, in detecting “spoofed” packets. Ex.1009, 22-23; Ex.1003, ¶ 219. By teaching to compare destination address and port number values of an encapsulated IPv4 packet with the corresponding values in a constructed IPv6 address, the prior art renders obvious the claimed “verifying” step. Ex.1003, ¶ 220.

8. Claim 7

[7.0] A non-transitory computer program product that is at least one of downloadable from a communications network and stored on a computer-readable medium and executable by a microprocessor, comprising program code instructions for executing the method according to claim 1 of receiving a data packet in an IPv6 domain.

As described above in Section X.E.5[1.0.3], the CPI 220E performs the features of the method of claim 1. Ex.1003, ¶¶ 221-222. The CPI 220E “comprises

an IPv4 processor for processing IPv4 specific communication, an IPv6 processor for processing IPv6 specific communication, and a protocol converter for processing communication between the IPv4 and IPv6 processors.” Ex.1005, ¶¶ 19, 63. Therefore, the CPI comprises a *microprocessor*. Ex.1003, ¶¶ 223-224.

Further, Ananda discloses that the CPI comprises various memories. Ex.1005, claims 6, 9 and 13. The memories of Ananda correspond to a *computer-readable medium*. Ex.1003, ¶ 225.

Ananda discloses that the CPIs may be “implemented as software modules within a destination host, a stand-alone unit coupled to a subnet and an IPv6 network,...or be incorporated in a network interface card (NIC).” Ex.1005, ¶ 118. Thus, Ananda discloses executing software (“*program code instructions*”) to perform the functions of the CPI 220E. Ex.1003, ¶¶ 226-227.

As discussed generally regarding claim 1, Ananda, Wetterwald, and RFC4380 render obvious *a method for receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain*. See *supra* Section X.E.5; Ex.1003, ¶ 228.

Wetterwald discloses that the steps for routing data packets in an IPv6 network can be implemented as “executable code stored on a computer readable medium (e.g., floppy disk, hard disk, EEPROM, CD-ROM, etc.).” Ex.1008, 7:55-65. It would have been obvious to a POSITA to implement the teachings of the prior art in this way so that the executable code would be executed by the

processors of Ananda's CPI 220E and thus perform the described packet routing techniques. Ex.1003, ¶¶ 229-230. Accordingly, in view of the disclosure of Wetterwald, a POSITA would have been motivated to modify Ananda to provide a computer-readable medium that stores a computer program product executed by the processors of the CPI 220E, to implement the functions of Ananda's CPI 220E to receive a data packet in an IPv6 domain. Ex.1003, ¶ 231.

Therefore, Ananda, Wetterwald, and RFC4380 render obvious limitation [7.0]. Ex.1003, ¶ 232.

9. Claim 8

[8.0.1] A device for receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain,

Ananda renders obvious limitation [8.0.1]. Ex.1003, ¶ 233.

Limitation [8.0.1] is identical to limitation [1.0.1] above, except that limitation [8.0.1] recites a “device” instead of a “method.” Ananda discloses a device (CPI 220E) for receiving an IPv6 data packet, where the CPI “comprises an IPv4 processor for processing IPv4 specific communication, an IPv6 processor for processing IPv6 specific communication, and a protocol converter for processing communication between the IPv4 and IPv6 processors.” Ex.1005, ¶¶ 19, 63. As described above with respect to limitation [1.0.1], Ananda's CPI 220E receives an IPv6 data packet in an IPv6 domain connected to an IPv4 domain. Ananda's CPI

220E is a device. Ex.1003, ¶ 234.

Ananda describes the structural components of the CPI 215E of subnet 215 in Figure 3 below. Ex.1005, ¶¶ 89-102. However, the description of the structural elements of the CPI 215E applies equally to the CPI 220E of subnet 220. Ex.1003, ¶ 235. For instance, Ananda explains that the CPI 220E uses “its decapsulator 354” (Ex.1005, ¶ 142) and “its encapsulator 352” (Ex.1005, ¶ 144), which refers to the structural components of the CPI 215E illustrated in Figure 3. Dr. Min’s declaration explains in more detail why the CPI 220E has the same structural configuration as the CPI 215E illustrated in Figure 3. Ex.1003, ¶ 235.

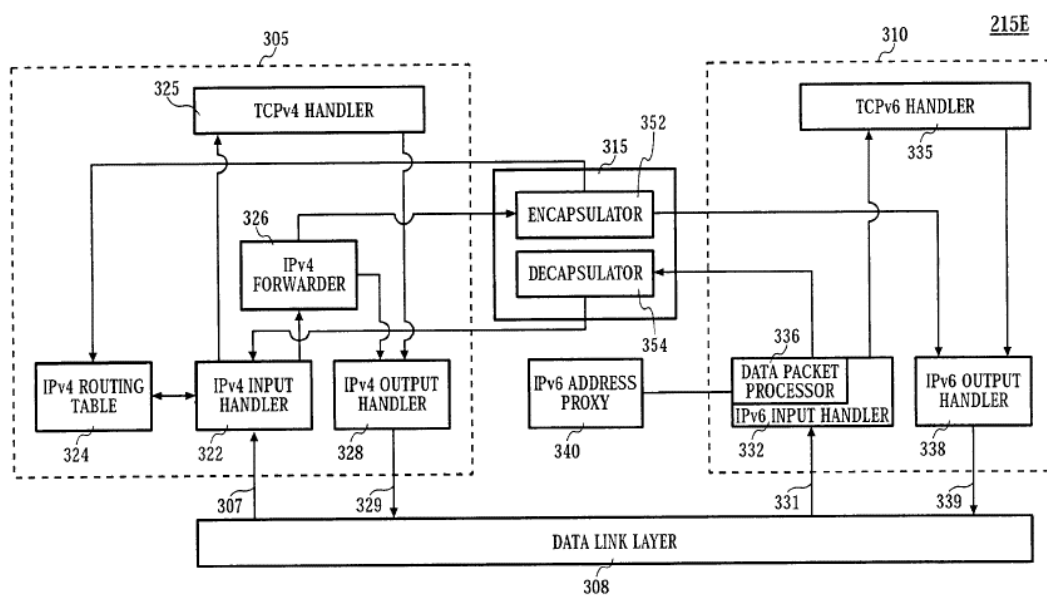
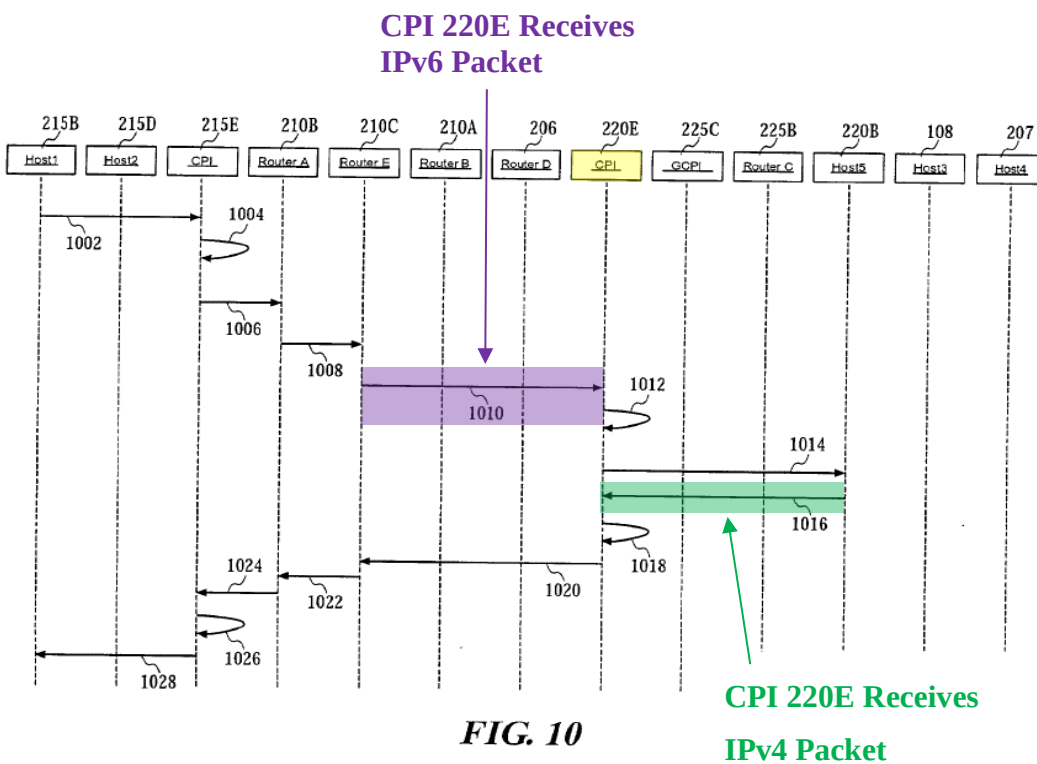


FIG. 3

Ex.1005, Fig. 3.

Because CPI 220E receives IPv6 packets in an IPv6 network (step 1010 in

annotated Figure 10 below) and receives IPv4 packets in an IPv4 network (step 1016 in annotated Figure 10 below), CPI 220E is assigned both IPv4 and IPv6 addresses, which provides further confirmation that CPI 220E is a device. For example, Ananda explains that CPI 220E is assigned an IPv6 address of 2001:208:3:58:202:2dff:fe0D:4561. Ex.1005, ¶ 145; Ex.1003, ¶ 236.



Ex.1003, ¶ 236; Ex.1005, Fig. 4 (annotated).

Thus, Ananda renders obvious limitation [8.0.1]. Ex.1003, ¶ 237.

[8.0.2] said packet comprising an IPv6 destination address and an IPv6 source address,

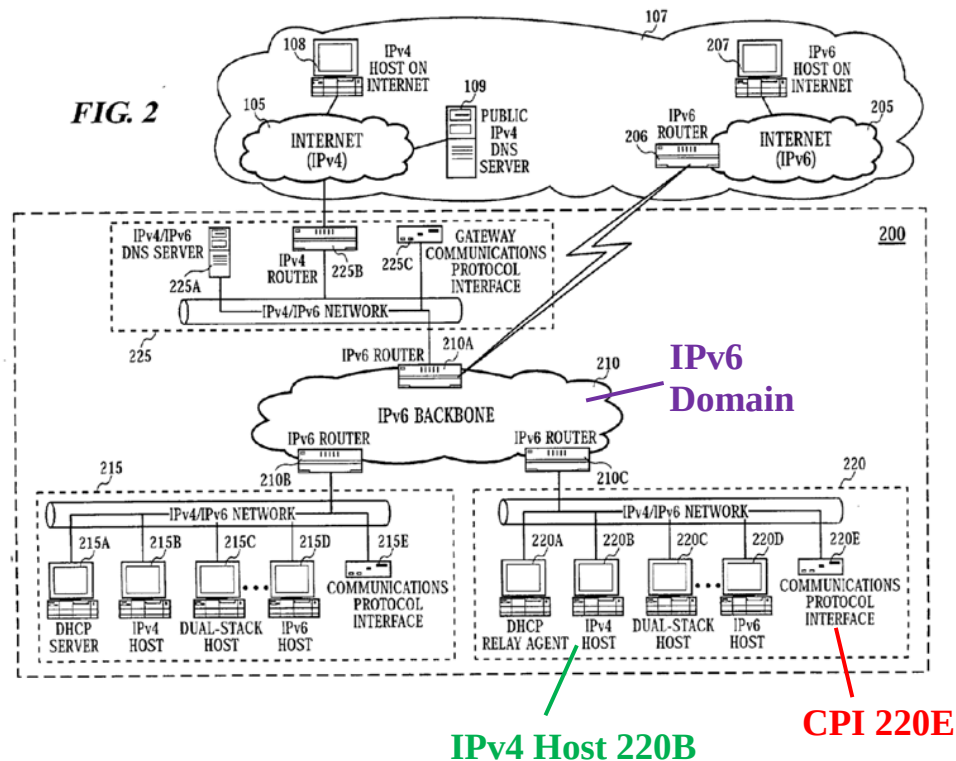
See analysis of limitation [1.0.2], which recites an identical limitation.

Ex.1003, ¶ 238.

[8.0.3] said device for use in a home gateway adapted to connect a user terminal to said IPv6 domain, comprising:

Limitation [8.0.3] is identical to limitation [1.0.3] above, except that limitation [1.0.3] recites that the method is executed in a home gateway, whereas limitation [8.0.3] recites “said device for use in a home gateway.” Ex.1003, ¶ 239.

As described above with respect to limitation [8.0.1], the CPI 220E of Ananda is a *device*. Further, as described above with respect to limitation [1.0.3], Ananda’s **CPI 220E** is a home gateway adapted to connect **IPv4 host 220B** (“*user terminal*”) to communicate with **IPv6 backbone 210** (“*IPv6 domain*”), as shown in annotated Figure 2 below. Ex.1005 ¶¶ 79, 77-78, 63; Ex.1003, ¶ 240.



Ex.1003, ¶ 240; Ex.1005, Fig. 2 (annotated).

Therefore, Ananda renders obvious limitation [8.0.3]. Ex.1003, ¶ 241.

[8.1] means for identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address, and a destination port number;

Limitation [8.1] is identical to limitation [1.1], except that limitation [8.1] recites “means for identifying an IPv6 destination address....,” whereas limitation [1.1] recites “identifying an IPv6 destination address....” Ananda in view of Wetterwald and RFC4380 render obvious limitation [1.1]. Ex.1003, ¶ 242.

Ananda’s CPI 220E is a receiver device because it includes an **IPv4 processor 305** having an IPv4 input handler 322, and an **IPv6 processor 310**

having an IPv6 input handler 332, as shown in annotated Figure 3 below. Ex.1005, ¶¶ 89-90, 94. Ananda discloses that the CPI comprises various memories. Ex.1005, claims 6, 9 and 13. The CPI 220E has the same structure for identifying an IPv6 destination address (i.e., a receiver device having a processor and memory) for limitation [8.1], as discussed above in Section VIII.C.1. Ex.1003, ¶¶ 243-244.

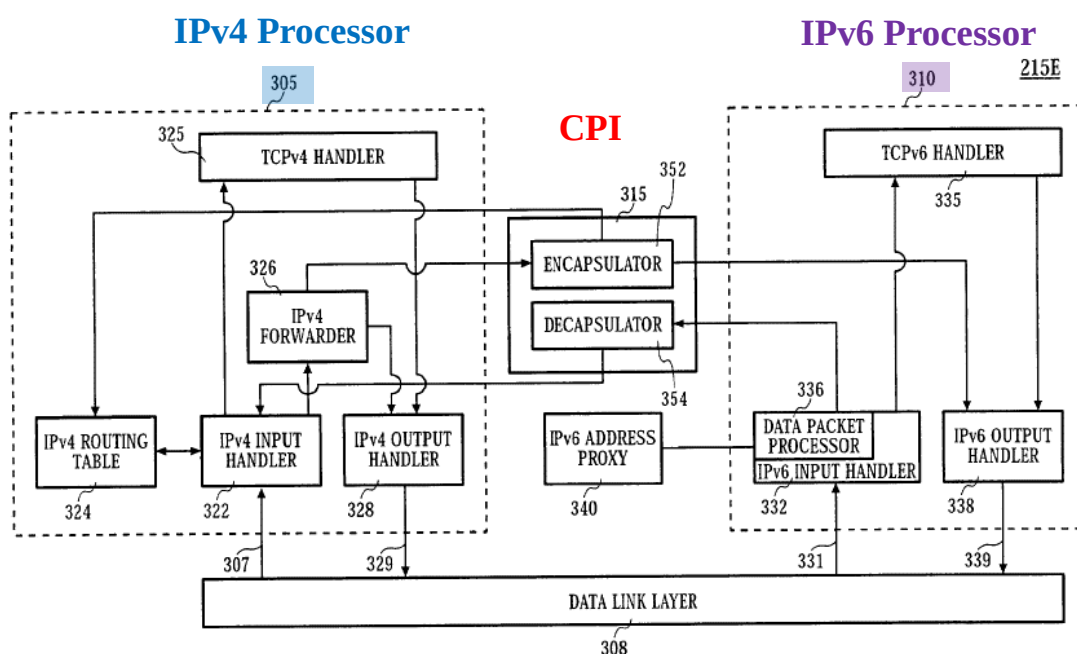


FIG. 3

Ex.1003, ¶ 244; Ex.1008, Fig. 3 (annotated).

Thus, Ananda in view of Wetterwald and RFC4380 renders obvious limitation [8.1]. Ex.1003, ¶ 245.

[8.2.1] means for regularizing, if necessary, at least one of the IPv6 source address and the IPv6 destination address of the data packet by replacing said at least one IPv6 address,

Ananda in view of Wetterwald renders obvious limitation [8.2.1]. Ex.1003, ¶ 246.

Limitation [8.2.1] is identical to limitation [1.2.1], except that limitation [8.2.1] recites “means for regularizing....,” whereas limitation [1.2.1] recites the method step of “regularizing....” Ex.1003, ¶ 247.

As discussed above with respect to limitation [1.2.1], Ananda’s CPI 220E, modified to include NAT features as taught by Wetterwald, regularizes the IPv6 destination address in the received IPv6 packet by replacing the IPv6 destination address with a private IPv4 destination address. The modified CPI 220E would perform a public-to-private IPv4 address translation, like Wetterwald’s translation resource 53, and thus the CPI 220E would include a NAT table stored in the modified CPI 220E’s memory similar to the translation resource 53 as shown in annotated Figure 5 of Wetterwald below. Ex.1008, 6:38-49; Ex.1003, ¶ 248.

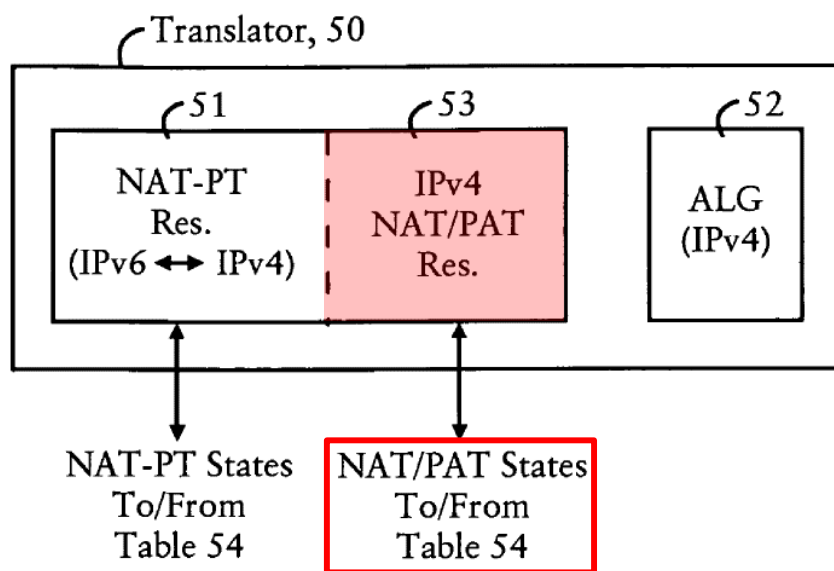


Figure 5

Ex.1003, ¶ 248; Ex.1008, Fig. 5 (annotated).

The modified CPI 220E thus has the same structure as the “means for regularizing” limitation (i.e., a receiver device having a processor and memory that stores a NAT for translating between addresses), as discussed above in Section VIII.C.2. Ex.1003, ¶¶ 249-251.

Thus, Ananda in view of Wetterwald renders obvious limitation [8.2.1].

Ex.1003, ¶ 252.

[8.2.2] the replacement process belonging to a group comprising replacement of a constructed address with a native address and replacement of a native address with a constructed address, and

See analysis of limitation [1.2.2] above, which recites an identical limitation.

Ex.1003, ¶ 253.

[8.2.3] [means for...] modifying the data packet using the regularized address; and

Limitation [8.2.3] is identical to limitation [1.2.3], except that limitation [8.2.3] is part of the “means for regularizing” limitation. Ex.1003, ¶ 254. As described above with respect to limitation [8.2.1], the modified CPI 220E of Ananda has the same structure as the “means for regularizing” limitation.

Ex.1003, ¶ 255.

Thus, Ananda in view of Wetterwald renders obvious limitation [8.2.3].

Ex.1003, ¶ 256.

[8.3] means for routing the IPv6 data packet to its destination.

Limitation [8.3] is identical to limitation [1.3], except that limitation [8.3] recites “means for routing,” whereas limitation [1.3] recites “routing....” As described above, Ananda renders obvious limitation [1.3]. Ex.1003, ¶¶ 257-258. Further, Ananda’s CPI 220E has the same structure as the “means for routing” limitation as discussed above in Section VIII.C.3 (i.e., telecommunications means 25₄ of the device 25). Ex.1003, ¶ 259. The CPI 220E includes IPv4 and IPv6 processors 305, 310 which communicate with the data link layer 308 to route IPv4 and IPv6 packets. Ex.1005, ¶¶ 90, 94-95, Fig. 3. Further, Ananda discloses that the CPIs may be incorporated in a network interface card (NIC). Ex.1005, ¶ 118.

The CPI 220E includes a telecommunications means. Ex.1003, ¶ 259.

Thus, Ananda renders obvious limitation [8.3]. Ex.1003, ¶ 260.

XI. CONCLUSION

Petitioner asks that the Patent Office order an *inter partes* review trial for claims 1 and 5-8 and then cancel these claims as unpatentable.

Respectfully submitted,

October 1, 2020

/Theodore M. Foster/
Theodore M. Foster
Counsel for Petitioner
Registration No. 57,456

CERTIFICATE OF WORD COUNT

Pursuant to 37 C.F.R. § 42.24(d), Petitioner hereby certifies, in accordance with and reliance on the word count provided by the word-processing system used to prepare this Petition, that the number of words in this paper is 13,987. Pursuant to 37 C.F.R. § 42.24(d), this word count excludes the table of contents, table of authorities, mandatory notices under § 42.8, certificate of service, certificate of word count, appendix of exhibits, and any claim listing.

Dated: October 1, 2020

/Theodore M. Foster/
Theodore M. Foster
Lead Counsel for Petitioner
Registration No. 57,456

CERTIFICATE OF SERVICE

The undersigned certifies, under 37 C.F.R. §§ 42.105 and 42.6, that service was made on the Patent Owner as detailed below.

Date of service October 1, 2020

Manner of service Federal Express

Documents served Petition for *Inter Partes* Review, including Exhibit List; Exhibits 1001-1010, 1012-1033, 1036, 1040-1044, and 1047-1071

Persons served Faegre Drinker Biddle & Reath LLP (Chicago)
 Attn: Patent Docket Dept.
 191 N. Wacker Drive, Suite 3700
 Chicago, IL 60606

/Theodore M. Foster/
Theodore M. Foster
Counsel for Petitioner
Registration No. 57,456