

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

JUNIPER NETWORKS, INC.,
Petitioner,

v.

MONARCH NETWORKING SOLUTIONS LLC,
Patent Owner.

U.S. Patent No. 8,451,845

“Method of Receiving a Data Packet in an IPv6 Domain, an Associated Device and
an Associated Home Gateway”

IPR2024-00565

**DECLARATION OF DR. SAMRAT BHATTACHARJEE IN SUPPORT OF
PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 8,451,845**

TABLE OF CONTENTS

I.	INTRODUCTION	1
A.	Qualifications.....	1
B.	Materials Considered	4
II.	LEGAL UNDERSTANDINGS	5
A.	Claim Construction.....	5
B.	Anticipation	7
C.	Obviousness.....	7
III.	TECHNOLOGY BACKGROUND.....	9
A.	Internet Protocol Version 4	9
B.	Internet Protocol Version 6	12
C.	IPv4/IPv6 Transition Mechanisms	14
1.	Converting IPv4 Addresses into IPv6 Addresses to Allow IPv4 Hosts to Communicate with IPv6 Hosts.....	14
2.	Using Port Information to Prevent IPv4 Exhaustion.....	16
3.	Combining the Above Two Ideas.....	18
D.	Network Address Translation/Port Address Translation.....	23
IV.	OVERVIEW OF THE '845 PATENT	26
A.	Priority.....	26
B.	Specification	26
V.	LEVEL OF ORDINARY SKILL IN THE ART	28
VI.	CLAIM CONSTRUCTION	29
VII.	IDENTIFICATION OF HOW THE CHALLENGED CLAIMS ARE UNPATENTABLE.....	32

A.	Ground 1: Claims 1, 5-8 Would Have Been Obvious in View of Li and Li-2.....	32
1.	Li (EX1037).....	32
2.	Li-2 (EX1038)	39
3.	The Li-Li-2 Combination.....	45
4.	Motivations to Combine Li and Li-2.....	45
5.	Claim 1	47
6.	Claim 5	75
7.	Claim 6: 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.	79
8.	Claim 7: 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.....	81
9.	Claim 8	82
B.	Ground 2: Claims 1, 5-8 Would Have Been Obvious in View of Li	87
1.	Evidence of Simultaneous Invention Demonstrates the Obviousness of Combining Li and Li-2	87
2.	Motivations to Modify Li.....	90
3.	Claim 1	93
4.	Claim 5	95
5.	Claim 6: A non-transitory computer program product that is at least one of downloadable from a communications	

	network and/or stored on a computer-readable medium and/or executable by a microprocessor, comprising program code instructions for executing the method according to claim 3 of sending a data packet from an IPv6 domain to a destination address of an IPv4 domain.	95
6.	Claim 7: IPv6 domain access equipment according to claim 6, wherein the IPv4 is a public network, said equipment comprising first means for advertising to the IPv4 domain IPv4 addresses of network equipments connected to the IPv6 domain and second means for advertising in the IPv6 domain IPv6 addresses constructed from IPv4 addresses of the IPv4 domain	96
7.	Claim 8	96
C.	Ground 3: Claims 1, 5-8 Would Have Been Obvious in View of Li alone or in view of Li-2 in further view of Paunikar	97
1.	Paunikar (EX1057)	97
2.	The Li-Li-2-Paunikar Combination.....	100
3.	Motivations to Combine Li, Li-2, and Paunikar	100
4.	Claim 1	102
5.	Claim 5	106
6.	Claim 6: A non-transitory computer program product that is at least one of downloadable from a communications network and/or stored on a computer-readable medium and/or executable by a microprocessor, comprising program code instructions for executing the method according to claim 3 of sending a data packet from an IPv6 domain to a destination address of an IPv4 domain.	108
7.	Claim 7: IPv6 domain access equipment according to claim 6, wherein the IPv4 is a public network, said equipment comprising first means for advertising to the IPv4 domain IPv4 addresses of network equipments connected to the IPv6 domain and second means for	

	advertising in the IPv6 domain IPv6 addresses constructed from IPv4 addresses of the IPv4 domain	109
8.	Claim 8	110
D.	Ground 4: Claims 1, 5-8 Would Have Been Obvious in View of Li and Wu	111
1.	Wu (EX1039)	111
2.	The Li-Wu-Paunikar Combination.....	113
3.	Motivations to Combine Li, Wu, and Paunikar	114
4.	Evidence of Simultaneous Invention Demonstrates The Obviousness Of Combining Li And Wu	117
5.	Claim 1	117
6.	Claim 5	120
7.	Claim 6: A non-transitory computer program product that is at least one of downloadable from a communications network and/or stored on a computer-readable medium and/or executable by a microprocessor, comprising program code instructions for executing the method according to claim 3 of sending a data packet from an IPv6 domain to a destination address of an IPv4 domain.	120
8.	Claim 7: IPv6 domain access equipment according to claim 6, wherein the IPv4 is a public network, said equipment comprising first means for advertising to the IPv4 domain IPv4 addresses of network equipments connected to the IPv6 domain and second means for advertising in the IPv6 domain IPv6 addresses constructed from IPv4 addresses of the IPv4 domain	120
9.	Claim 8	121
VIII.	CONCLUSION	122

PETITIONER’S EXHIBIT LIST

EX1001	U.S. Patent No. 8,451,845 to Mohamed Boucadair et al. (“’845 patent”)
EX1002	Prosecution History of U.S. Patent No. 8,451,845 (“’845 Prosecution History”)
EX1003	Declaration of Dr. Samrat Bhattacharjee under 37 C.F.R. §1.68
EX1004	<i>Curriculum Vitae</i> of Dr. Bhattacharjee
EX1005	Declaration of Dr. Sylvia Hall-Ellis under 37 C.F.R. §1.68
EX1006	<i>Curriculum Vitae</i> of Dr. Hall-Ellis
EX1007	Declaration of Alexa Morris under 37 C.F.R. §1.68
EX1008	Declaration of Ron Bonica under 37 C.F.R. §1.68
EX1009	Declaration of Alexa Morris under 37 C.F.R. §1.68
EX1010	<i>Ex Parte</i> Reexamination Communication Transmittal Form, dated Dec. 5, 2023, Reexamination Control No. 90/019,246 (“the ’246 request”).
EX1011	RFC 791, “Internet Protocol Specification” (Sept. 1981), available at https://datatracker.ietf.org/doc/html/rfc791
EX1012	RFC 793, “Transmission Control Protocol” (Sept. 1981), available at https://datatracker.ietf.org/doc/html/rfc793
EX1013	P. Richter et al., “A Primer on IPv4 Scarcity” (Feb. 27, 2015), available at https://arxiv.org/pdf/1411.2649.pdf
EX1014	U.S. Patent Application Publication No. 2006/0146870 to Harvey et al. (“Harvey”)
EX1015	RFC 2460, “Internet Protocol, Version 6 Specification” (Dec. 1998), available at https://datatracker.ietf.org/doc/html/rfc2460
EX1016	RFC 1884, “IP Version 6 Addressing Architecture” (Dec. 1995), available at https://datatracker.ietf.org/doc/html/rfc1884

EX1017	RFC 3587, “IPv6 Global Unicast Address Format” (Aug. 2003), available at https://www.rfc-editor.org/rfc/pdf/rfc3587.txt.pdf
EX1018	Pete Loshin, “IPv6: Theory, Protocol, and Practice” (2d Ed. 2004)
EX1019	U.S. Patent No. 7,391,768 to Samprathi et al. (“Samprathi”)
EX1020	T. Rooney, “IPv4-to-IPv6 Transition Strategies,” INS (Feb. 2007), available at https://www.hpc.mil/images/hpcdocs/ipv6/migration_white_paper_1e5eb697-e7d7-4f62-9452-0121aa0ff0f1.pdf
EX1021	H. Yeom et al., “IP Multiplexing by Transparent Port-Address Translator,” 10th USENIX System Administration Conference (LISA X) (Sept. 29, 1996), available at https://www.usenix.org/conference/lisa-96/ip-multiplexing-transparent-port-address-translator
EX1022	A. Reitz, “The Dynamic Port Reservation Protocol,” Univ. Ill. at Urbana-Champaign (2001), available at http://redefine.dyndns.org/~andyr/documents_and_writings/thesis/docs/areitz_msthesis.pdf
EX1023	Y. Zhu et al., “Stateless Mapping and Multiplexing of IPv4 Addresses in Migration to IPv6 Internet,” Institute of Electrical and Electronics Engineers (“IEEE”) (Nov. 2008), available at https://ieeexplore.ieee.org/document/4698208
EX1024	U.S. Patent No. 8,032,639 to King et al. (“King”)
EX1025	U.S. Patent No. 8,019,889 to Biswas et al. (“Biswas”)
EX1026	X. Li et al., “Bi-Directional Address Borrowing (IVI)” (June 23, 2008)
EX1027	X. Li et al., “Prefix-Specific and Stateless Address Mapping (IVI) for IPv4/IPv6 Coexistence and Transition,” IETF (July 6, 2008), available at https://datatracker.ietf.org/doc/html/draft-xli-behave-ivi-00
EX1028	X. Li et al., “Prefix-Specific and Stateless Address Mapping (IVI) for IPv4/IPv6 Coexistence and Transition,” IETF-72, Dublin (July 29,

	2008), available at https://www.ietf.org/proceedings/72/slides/behave-5.pdf
EX1029	R. Bush et al., “A Better Approach than Carrier-Grade-NAT,” Columbia Univ. (Sept. 2008), available at https://academiccommons.columbia.edu/doi/10.7916/D89S1ZWC
EX1030	U.S. Patent Application Publication No. 2004/0190549 to Huitema et al. (“Huitema”)
EX1031	U.S. Patent No. 7,149,225 to Thubert et al. (“Thubert”)
EX1032	C. Huitema, “Teredo: Tunneling IPv6 over UDP Through NATs,” Internet Engineering Task Force (“IETF”) (Sept. 17, 2002), available at https://www.ietf.org/archive/id/draft-ietf-ngtrans-shipworm-08.txt
EX1033	Petition for <i>Inter Partes</i> Review, <i>Cisco Sys., Inc. v. Monarch Networking Sol’ns LLC</i> , IPR2020-01679, Paper 2 (PTAB Oct. 1, 2020)
EX1034	Decision Settlement Prior to Institution of Trial, <i>Cisco Sys., Inc. v. Monarch Networking Sol’ns LLC</i> , IPR2020-01679, Paper 8 (PTAB Jan. 26, 2021)
EX1035	U.S. Patent No. 7,624,195 to Biswas et al., filed on May 8, 2003
EX1036	Claim Construction Memorandum and Order, <i>Monarch Networking Sol’ns LLC v. Cisco Sys., Inc.</i> , No. 2:20-cv-00015-JRG (E.D. Tex. Jan. 7, 2021)
EX1037	X. Li et al., “Forwarding IPv4 Traffics in Pure IPv6 Backbone with Stateless Address Mapping,” IEEE (Apr. 2006), available at https://ieeexplore.ieee.org/document/1687557 (“Li”)
EX1038	X. Li, “Bi-directional Address Borrowing (IVI),” IETF (June 21, 2008), available at https://www.ietf.org/staging/draft-xli-ivi-00-7430-cancelled.txt (“Li-2”)
EX1039	Certified Translation of CN Patent No. 1525699 to Wu et al. (“Wu”)
EX1040	CN Patent No. 1525699 to Wu et al.

EX1041	R. Bush, “IPv4 / IPv6 Coexistence Mechanisms” (Oct. 26, 2008)
EX1042	Monarch’s Opposition to Juniper’s Motion to Transfer, <i>Monarch Networking Solutions LLC v. Juniper Networks, Inc.</i> , No. 1:23-cv-670-TSE-LRV (E.D. Va. Oct. 3, 2023), ECF No. 52
EX1043	Complaint, <i>Monarch Networking Sol’ns LLC v. Juniper Networks, Inc.</i> , No. 1:23-cv-670-TSE-LRV (E.D. Va. May 23, 2023), ECF No. 1
EX1044	Screenshot IETF Staging Folder, available at https://www.ietf.org/staging/
EX1045	R. Bonica, “draft-kumar-mpls-fec-to-nhlfe-mib-00,” IETF (Jan. 7, 2010), available at https://www.ietf.org/staging/draft-kumar-mpls-fec-to-nhlfe-mib-00-16143-cancelled.txt
EX1046	Correspondence from badra@isima.fr (Oct. 9, 2007), available at https://mailarchive.ietf.org/arch/msg/ngo/6JM4rSBIZM8Ajw02luUG4OqSrmQ/
EX1047	Correspondence from dirk.hugo@t-systems.com (Oct. 12, 2007), available at https://mailarchive.ietf.org/arch/msg/multimob/H8V3v2MvxhYRhjqOKSN7mhM39Q8/
EX1048	Correspondence from dirk.hugo@t-systems.com (Oct. 25, 2007), available at https://mailarchive.ietf.org/arch/msg/multimob/FvyjLiCNdh88V2I3Fm8B21vMtkA/
EX1049	Correspondence from asaeda@sfc.wide.ad.jp (Nov. 12, 2007), available at https://mailarchive.ietf.org/arch/msg/multimob/6Zj-nouoPaLdo46ART2ZGWmJqkE/
EX1050	Correspondence from fluffy@cisco.com (Nov. 13, 2007), available at https://mailarchive.ietf.org/arch/msg/tools-development/CPLnZyjcureXq6j6mII5eTZF1Zo/
EX1051	Correspondence from lowekamp@sipeerior.com (Nov. 14, 2007), available at

	https://mailarchive.ietf.org/arch/msg/sipping/70F4vroEAcKioD214LTt2FJpInk/
EX1052	Correspondence from lowekamp@sipeerior.com (Nov. 17, 2007), available at https://mailarchive.ietf.org/arch/msg/behave/yUKKzuhlhhnKkRRC0V_zGXEkRP4/#
EX1053	Correspondence from schierl@hhi.fhg.de (Nov. 19, 2007), available at https://mailarchive.ietf.org/arch/msg/mmusic/4FxACR25hKzg59oSR-uSoqoSjc0/
EX1054	Correspondence from andrew.dolganow@alcatel-lucent.com (Feb. 11, 2008), available at https://mailarchive.ietf.org/arch/msg/pwe3/ceWfUIVFI-tMV-l_ueo7HcNCums/
EX1055	Correspondence from victor.pascual.avila@gmail.com (Feb. 23, 2008), available at https://mailarchive.ietf.org/arch/msg/p2psip/P71OF43ons3EYw7UKg0_u3M_tww/
EX1056	October 8, 2006 Capture of IETF Datatracker, available at https://web.archive.org/web/20061008141652/https://datatracker.ietf.org/public/pidtracker.cgi
EX1057	US Patent No. 7,752,334 B2 to Paunikar; filed October 15, 2002, published April 15, 2004 (“Paunikar”)
EX1058	Trevor J. Baumgartner et al., Implementation of A Network Address Translation Mechanism Over IPv6 (June 2004)
EX1059	Andrew S. Tanenbaum, Computer Networks (4 th Ed. 2003)
EX1060	RFC 1919, Classical versus Transparent IP Proxies (March 1996), available at https://datatracker.ietf.org/doc/html/rfc1919
EX1061	Unified EPR (Control No. 90/019,246), filed on September 8, 2023
EX1062	RFC 1631, “The IP Network Address Translator (NAT)” (1994), available at https://datatracker.ietf.org/doc/html/rfc1631

EX1063	RFC 3022 (Traditional NAT), dated January 2001, available at https://datatracker.ietf.org/doc/html/rfc3022
EX1064	M. Wasserman, “IPv6-to-IPv6 Network Address Translation (NAT66),” (October 2008), available at https://www.ietf.org/archive/id/draft-mrw-behave-nat66-00.txt
EX1065	https://ipinfo.io/AS24348
EX1066	https://ipinfo.io/AS24349

LIST OF CHALLENGED CLAIMS

Claim	Limitation
1[pre]	A method of receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain, said packet comprising an IPv6 destination address and an IPv6 source address, said method being executed in a home gateway adapted to connect a user terminal to said IPv6 domain, and comprising the following steps:
1[a]	identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address and a destination port number
1[b]	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, 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 modifying the data packet using the regularized address; and
1[c]	routing the modified data packet to its destination.
5[pre]	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:
5[a]	searching an address translation table of the gateway for an entry associating with said constructed destination address a native destination address;
5[b]	if an entry is found, replacing the constructed destination address of the data packet with the native destination address from that entry.
6	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.
7	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.
8[pre]	A device for receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain, said packet comprising an IPv6 destination address and an IPv6 source address, said device for use in a home gateway adapted to connect a user terminal to said IPv6 domain, comprising:
8[a]	means for identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address, and a destination port number;
8[b]	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, 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 modifying the data packet using the regularized address; and
8[c]	means for routing the modified data packet to its destination.

I. INTRODUCTION

1. I have been retained by Quinn Emanuel Urquhart & Sullivan, LLP on behalf of Juniper Networks, Inc. (“Juniper” or “Petitioner”) as an independent expert in this *inter partes* review before the Patent Trial and Appeal Board of the United States Patent and Trademark Office to review claims 1, 5-8 (“Challenged Claims”) of U.S. Patent No. 8,451,845 (“the ’845 patent”). I have been asked by Petitioner to assist in evaluating the claims and the disclosure of the ’845 patent.

2. I understand that the ’845 patent is currently asserted in a litigation by Monarch Networking Solutions LLC (“Monarch” or “Patent Owner”) against Petitioner in *Monarch Networking Solutions LLC v. Juniper Networks, Inc.*, Case No. 3:23-cv-05076-RFL (N.D. Cal.).

A. Qualifications

3. I provide below an overview of my background and qualifications.

4. I earned a Ph.D. in Computer Science from Georgia Institute of Technology in 1999. Before that, I earned Bachelor of Science degrees in Mathematics and Computer Science from Georgia College and State University in 1994. I was a Teaching Assistant at the Georgia Institute of Technology from 1994 to 1995 and an Instructor in 1998.

5. From 1999 through 2005, I was an Assistant Professor at the University of Maryland, College Park, in the Department of Computer Science, and then an

Associate Professor with tenure from 2005 through 2009. In 2006, I was a Visiting Professor at Max Planck Institut für Software Systems in Saarbrücken, Germany. In 2007, I was a Visiting Researcher at AT&T Labs in Florham Park, New Jersey.

6. Since 2009, I have been a tenured Professor at the University of Maryland. My teaching and research have focused on computer networking and other aspects of the technologies pertaining to the '845 patent. As part of my Ph.D. research, I worked on implementing various network protocols in both IPv4 and IPv6. In fact, my IPv6 implementation of Active Networking was one of the first to use the (then yet incomplete) IPv6 implementation in the Linux kernel. I was familiar with the details of IPv4 and IPv6 as part of my dissertation research, and have continuously taught both IPv4 and IPv6 as part of undergraduate and graduate networking courses since 1999, and done research involving both IPv4 and IPv6 since the mid-1990s.

7. I served as a reviewer for ACM/IEEE Transactions on Networking, IEEE Journal on Selected Areas in Communications, Computer Communications Journal (Special Issue on Network Security), ACM Transactions on Computer Systems, Performance Evaluation Journal, Computer Communications Review, European Transactions on Telecommunications, IEEE Transactions on Parallel and Distributed Systems, and ACM Transactions on Internet Technology.

8. I am the author of numerous publications in the field of networking, including journal articles, book chapters, publications in proceedings, technical reports, and invited papers.

9. I have been active in a number of professional organizations and conferences. I have served with the National Science Foundation (NSF) Workshop on Network Testbeds, the NSF Networking Research Panel, the Department of Education High Performance Networking Panel, and as an Evaluator for the Intel Science Talent Search.

10. I have received several honors and awards. These include the Alfred P. Sloan Jr. Fellowship; the Best Paper Award, 14th Annual IEEE International Conference on High Performance Computing (HiPC) (with Vijay Gopalakrishnan, Ruggero Morselli, Peter Keleher, and Aravind Srinivasan); the Best Paper Award, 7th IEEE/ACM Conference on Grid Computing (with Jiksoo Kim, Byomsuk Nam, Peter Keleher, Michael Marsh, and Alan Sussman); and the NSF CAREER Award. I also received Teaching Excellence Awards in 2004, 2008, and 2012.

11. In 2017, along with colleagues from Cornell University and the Max Planck Society in Germany, I started an annual school for introducing undergraduate students from across the world to research. This school is partially supported by the NSF and had student participants from 29 countries.

12. I am also the co-director of a new joint Ph.D program in computer science between the University of Maryland and the Max Planck Society in Germany.

13. Additional details of my education and employment history, professional service, patents, publications, and other testimony are set forth in my current curriculum vitae, which is attached as Petitioner's Exhibit 1004.

B. Materials Considered

14. In forming my opinions, I have considered the '845 patent specification, including the Abstract, the figures, and the claim language itself, as would have been understood by a person having ordinary skill in the art. I have also reviewed the file history of the '845 patent, the exhibits listed within the petition, and any other material cited in the petition and this declaration.

15. In forming my opinions, I have relied on my personal knowledge and professional experience in designing, developing, deploying, or using software related to computer networks, IPv4, IPv6, and related technologies, and on the documents and information referenced in this declaration.

16. I am being compensated for my time spent on this matter at a rate of \$700 per hour, and my compensation is in no way contingent upon the outcome of this matter or on the opinions I offer.

II. LEGAL UNDERSTANDINGS

17. In this section, I describe my understanding of certain legal standards that I have relied upon in forming my opinions set forth in this declaration. I am not an attorney. I have been informed of these legal standards by Petitioner's attorneys.

18. I understand that the claims define the invention. I also understand that an unpatentability analysis is a two-step process. First, the claims of the patent are construed to determine their meaning and scope. Second, after the claims are construed, the content of the prior art is compared to the construed claims.

19. I understand that a claimed invention is only patentable when it is new, useful, and non-obvious in light of the "prior art." That is, the invention, as defined by the claims of the patent, must not be anticipated by or rendered obvious by the prior art.

A. Claim Construction

20. I understand that the United States Patent and Trademark Office interprets claim terms in an *inter partes* review proceeding under the "*Philips*" claim construction standard. I understand that under this standard, the meaning of claim terms is considered from the viewpoint of one of ordinary skill in the art at the time of the alleged invention. For this declaration, I have been asked to apply June 30, 2008 as the priority date for the '845 patent, which is the date I have used for claim

interpretation purposes. However, the interpretations that I provide below would have also been correct if the date of alleged invention was a year earlier or later.

21. I understand that, through the use of “means-plus-function” limitations, patentees may claim functions without reciting structures for performing those functions in the claims. Such limitations are construed to cover the corresponding structure described in the specification. This process requires first determining the function of the means-plus-function limitation, and then determining the corresponding structure disclosed in the specification. A structure is corresponding if the specification or file history clearly links or associates it to the function recited in the claim. I understand that if a term is a “means-plus-function” term, then it must be construed by first identifying the function as set forth in the claims and then identifying the corresponding structure in the specification that performs the recited function.

22. I have been informed that, for the purposes of my analysis in this IPR proceeding, the terms appearing in the claims of the '845 patent should be interpreted according to how a POSITA would have understood the claim terms at the time of the earliest priority date, in light of the patent specification, the prosecution history, and any other relevant intrinsic and extrinsic evidence.

B. Anticipation

23. I understand that a patent claim is anticipated when a single piece of prior art describes every element of the claimed invention, either expressly or inherently, arranged in the same way as in the claim. For inherent anticipation to be found, it is required that the missing descriptive material is necessarily present in the prior art. I understand that, for the purpose of an *inter partes* review, prior art that anticipates a claim can include both patents and printed publications from anywhere in the world.

C. Obviousness

24. I understand that a patent may not be valid even though the invention is not identically disclosed or described in the prior art if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious to a person having ordinary skill in the art in the relevant subject matter at the time the invention was made.

25. I understand that the following factors are evaluated to determine whether the claimed subject matter is obvious: (1) the scope and content of the prior art; (2) the difference or differences, if any, between each claim of the patent and the prior art; and (3) the level of ordinary skill in the art at the time the patent was filed. Unlike anticipation, which allows consideration of only one prior art reference, I

understand that obviousness may be shown by considering more than one prior art reference.

26. I understand that the following rationales may be used to determine whether a piece of prior art can be combined with other prior art or with other information within the knowledge of one of ordinary skill in the art:

- Combining prior art elements according to known methods to yield predictable results;
- Simple substitution of one known element for another to obtain predictable results;
- Use of known techniques to improve similar devices (methods, or products) in the same way;
- Applying a known technique to a known device (method, or product) ready for improvement to yield predictable results;
- “Obvious to try” - choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success;
- Known work in one field of endeavor may prompt variations of it for use in either the same field or a different one based on design incentives or other market forces if the variations would have been predictable to one of ordinary skill in the art; or

- Some teaching, suggestion, or motivation in the prior art that would have led one of ordinary skill to modify the prior art reference or to combine prior art reference teachings to arrive at the claimed invention.

27. Moreover, I understand that certain secondary considerations may be considered in evaluating obviousness in order to prevent hindsight bias. These secondary considerations may include : (1) commercial success; (2) long-felt but unresolved needs; (3) copying of the invention by others in the field; (4) initial expressions of disbelief by experts in the field; (5) failure of others to solve the problem that the inventor solved; and (6) unexpected results. I also understand that evidence of objective indicia of non-obviousness must be commensurate in scope with the claimed subject matter.

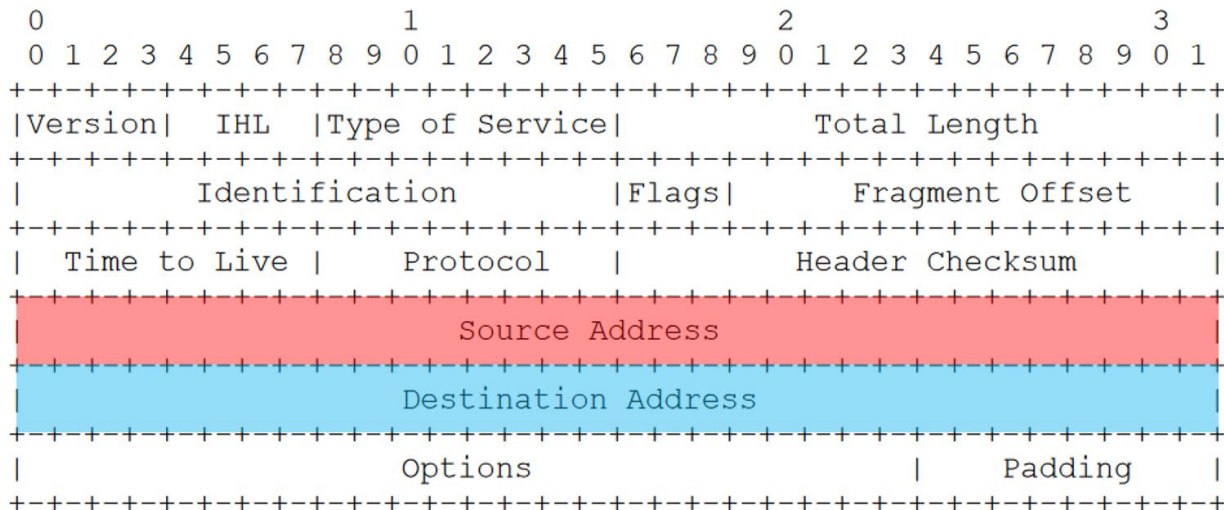
28. Finally, I understand that independently made, simultaneous inventions, made within a comparatively short space of time, are persuasive evidence that the claimed apparatus was the product only of ordinary mechanical or engineering skill. I understand that simultaneous invention can occur as late as one year after the patent's priority date.

III. TECHNOLOGY BACKGROUND

A. Internet Protocol Version 4

29. Internet Protocol Version 4 ("IPv4") is described in RFC 791 (September 1981). EX1011. When a source computer sends an IPv4 packet to a

destination computer, the packet includes an IPv4 header. *Id.*, 17.¹ The format of an IPv4 header is shown below and includes a source address (red) and a destination address (blue) for routing the packet from the source computer to the destination computer. *Id.*, 20. IPv4 addresses are 32 bits long:



Example Internet Datagram Header

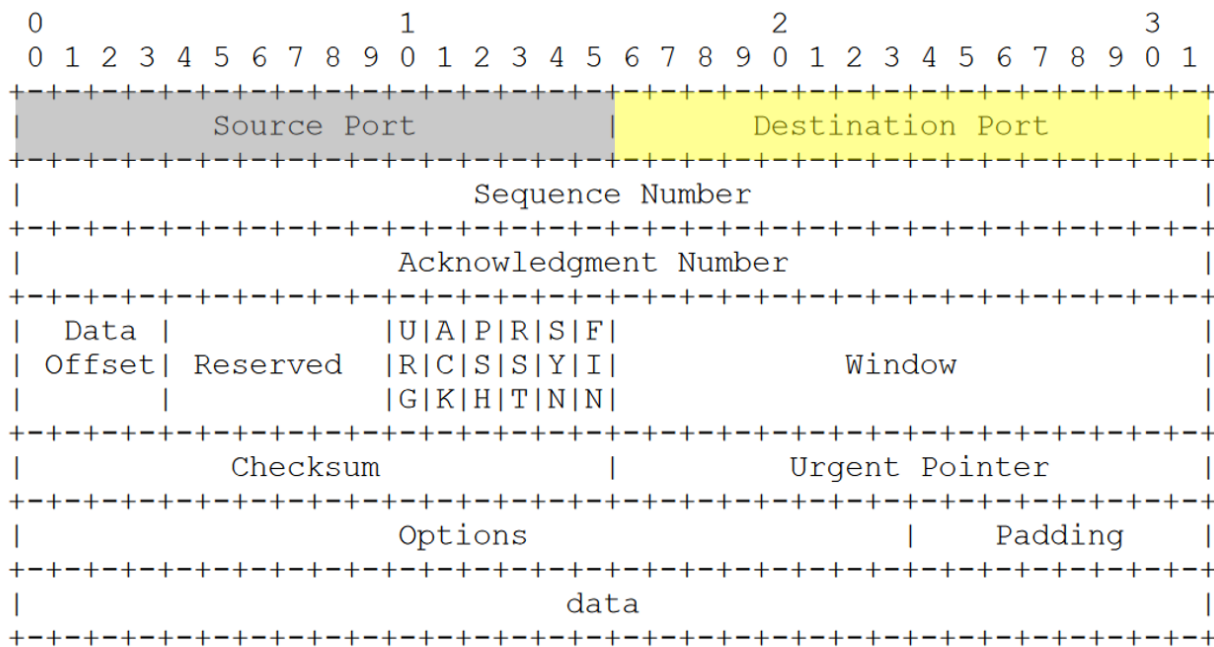
Figure 4.

Id., 17. An IPv4 address is typically written in dotted-quad notation, formatted as four decimal numbers (eight bits each) that are separated by periods, for example: **172.16.254.1**. This representation is for readability purposes; translating a decimal

¹ Exhibit citations—excluding exhibits, like patents, that designate their own paragraph or line numbers—refer to the page stamped on the bottom right of each exhibit, following the words “Juniper Exhibit.”

number to a different format—for example, hexadecimal—does not change the underlying numeric value.

30. An IPv4 packet sent across the Internet typically includes a TCP or a UDP header which contains a 16-bit source port number and a 16-bit destination port number. A TCP header is shown below with a source port number (grey) and destination port number (yellow):



TCP Header Format

Note that one tick mark represents one bit position.

Figure 3.

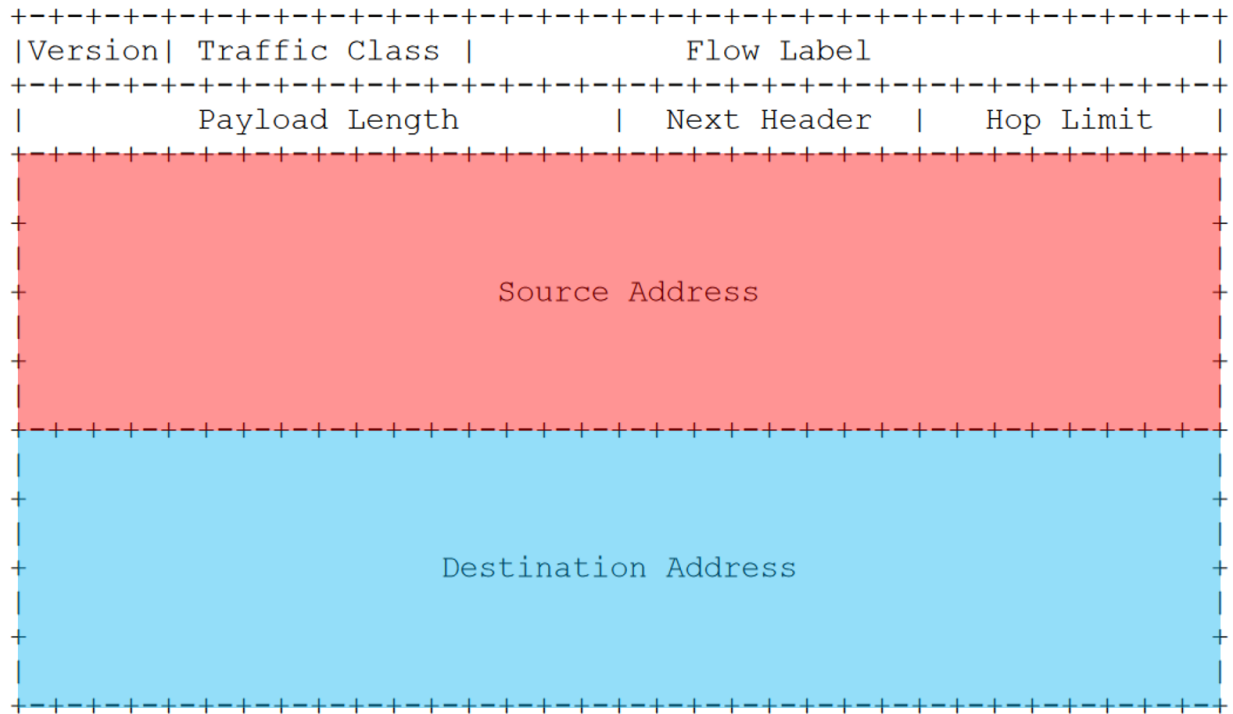
EX1012, 21. The source port number identifies the specific process that sends the data, while the destination port number identifies the process that is to receive it. Email, for instance, might use Simple Mail Transfer Protocol (“SMTP”) on port 25,

while a web server might use Hypertext Transfer Protocol (“HTTP”) on port 80 or HTTP Secure (“HTTPS”) on port 443.

31. The fact that an IPv4 address consists of 32 bits means that there are only ~4 billion available IPv4 addresses. By the time of the ’845 patent, it was well-understood that available IPv4 addresses would soon run out. EX1001, 1:61-62 (“It is commonly accepted in the IP service provider community that IPv4 public addresses are going to run out.”). Indeed, “[t]he Internet engineering community ha[d] long recognized the impending exhaustion of IPv4.” EX1013, 1, 8-9. And this problem had only grown over time “as the number of devices connecting to the Internet continue[d] to increase.” EX1014, ¶1. A new standard—Internet Protocol version 6 (“IPv6”)—was therefore developed to address this issue of IPv4 address exhaustion.

B. Internet Protocol Version 6

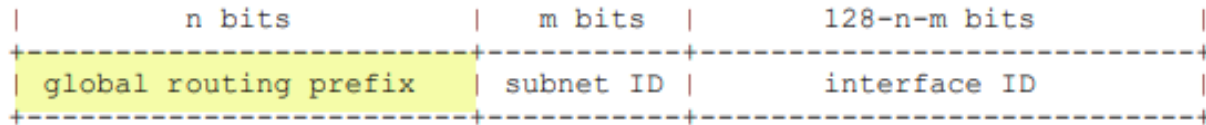
32. IPv6 is described in RFC 2460 (December 1998) and is an evolution of IPv4. EX1015. The IPv6 header is shown below and also includes a source address (red) and a destination address (blue):



Id., 4.

33. An IPv6 address is 128 bits long and is conventionally written using hexadecimal digits within eight groups, each of which is 16 bits. *Id.*, 5. Each group is expressed as four hexadecimal digits and the groups are separated by colons, for example: **FE80:CD00:0000:0CDE:1257:0000:211E:729C**. Further, the example address can be shortened, as the addressing scheme allows the omission of consecutive zeros, designated by two colons “::”. For example, FE80:CD21:0000:0000:0000:0000:211E:729C can be shortened to FE80:CD21::211E:729C. EX1016, 4-5.

34. An IPv6 address begins with a “prefix” (yellow) that is “n bits” long—meaning the prefix is variable in length:



EX1017, 2; EX1018, 176. The prefix can be allocated to individuals, companies, service providers, and any other entity requesting an IPv6 network address space.

EX1018, 176.

C. IPv4/IPv6 Transition Mechanisms

35. The most striking difference in the evolution from IPv4 to IPv6 is the tremendous expansion of the size of the IP address field. Whereas IPv4 uses a 32-bit IP address field, IPv6 uses 128 bits. A 32-bit address field provides a maximum of 2^{32} addresses (~4.2 billion addresses), whereas a 128-bit field provides 2^{128} addresses (~340 trillion trillion trillion addresses). The large number of possible IPv6 addresses helps prevent the problem of Internet addresses running out.

36. IPv6, however, is not inherently compatible with IPv4 and this incompatibility drove the need for transition mechanisms that permitted IPv6 to be deployed alongside IPv4. By the priority date of the '845 patent, myriad transition mechanisms had been proposed and developed.

1. Converting IPv4 Addresses into IPv6 Addresses to Allow IPv4 Hosts to Communicate with IPv6 Hosts

37. One of the most common methods of communicating between IPv4 devices and IPv6 devices involved using an intermediary device to convert an IPv4 address into an IPv6 address.

38. For example, U.S. Patent No. 7,391,768 (“Samprathi”), filed in May 2003, describes a network device **112** that is interposed between an IPv4 FTP client **118** and an IPv6 FTP server **108**:

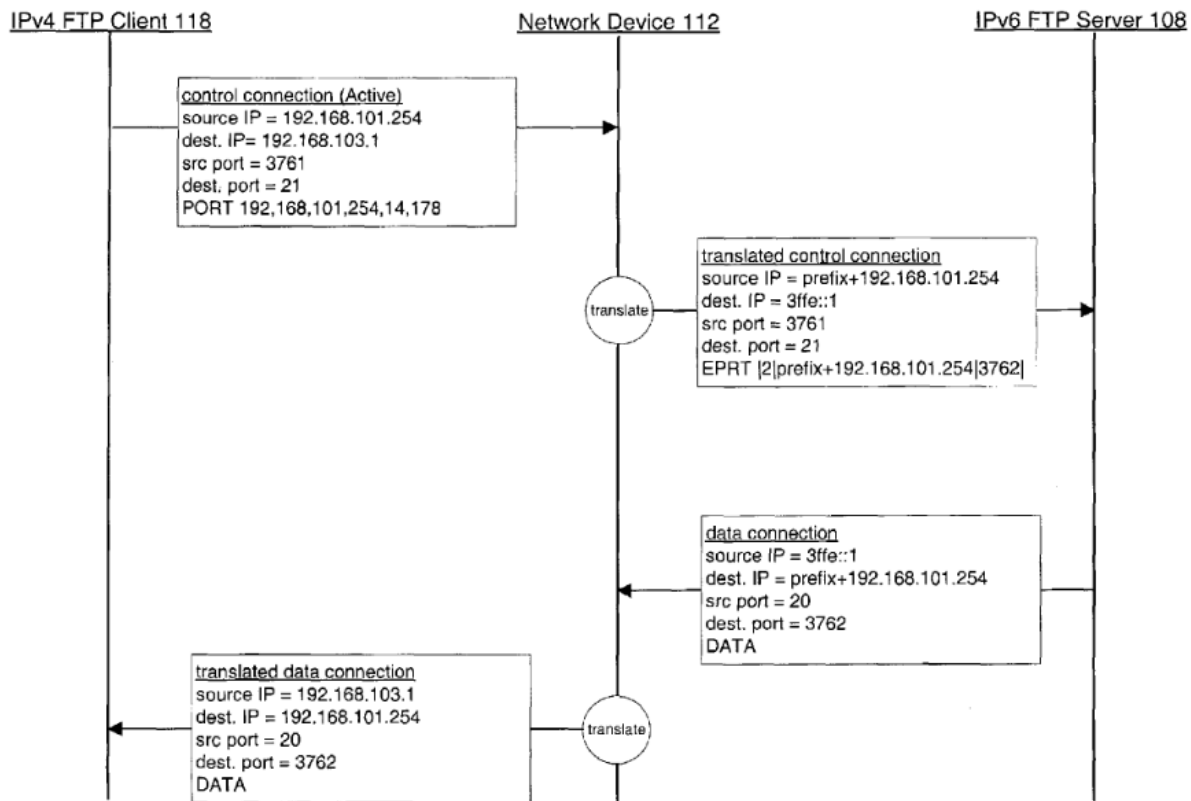


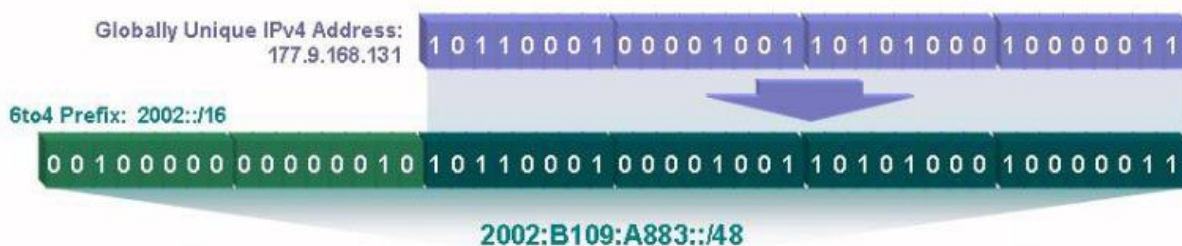
Figure 2A

EX1019, Fig. 2A. When the IPv4 client **118** sends data to the IPv6 server **108**, that data is intercepted by network device **112**. Network device **112** converts the IPv4

source address into an IPv6 source address by concatenating an IPv6 prefix with the IPv4 source address. *Id.*, 6:35-7:9.

39. Similarly, Tim Rooney published “IPv4-to-IPv6 Transition Strategies” in February 2007. EX1020. This document detailed “an IPv6-over-IPv4 tunneling technique” called “6to4,” which constructs an IPv6 address by concatenating a “6to4 prefix” with a “globally unique IPv4 address”:

Figure 8: 6to4 Address Prefix Derivation



Id., 11. The same can be done with a protocol called “6over4.” *Id.*, 15.

2. Using Port Information to Prevent IPv4 Exhaustion

40. Although converting IPv4 addresses into IPv6 addresses solves the compatibility issue—*i.e.*, it allows IPv4 hosts to communicate with IPv6 hosts—it does not solve the IPv4 address exhaustion issue—*i.e.*, IPv4 hosts still need to occupy one of the finite public IPv4 addresses to participate in the network, and eventually the finite number of public IPv4 addresses may run out. This need for both IPv4-and-IPv6 compatibility and IPv4-exhaustion prevention led some persons of ordinary skill in the art (“POSITAs”) to take advantage of the ~65,000 port numbers that hosts can use to access services.

41. As mentioned above, an IPv4 packet that encapsulates a TCP or UDP header includes 16-bit port numbers that identify the specific process or application on the computer sending or receiving the packet. See §III.A. The 16-bit port field provides a maximum of 2^{16} (65,536) unique port numbers. By the priority date of the '845 patent, POSITAs understood that computers typically utilized only a small portion of these port numbers. EX1021, 4 (“Now, we introduce a new technique to re-use address space by transparent port-address translation. We note that there are a lot of ports in a node and only a small portion of them are used at any time.”). Thus, POSITAs proposed solutions that extended the IPv4 address space by combining the 32-bit address with bits from the port number field. *Id.* In this way, multiple devices could share the same 32-bit address with port information distinguishing them. *Id.*; EX1022, 9 (“As such, if you consider a host to be the <IP address, port> pair, then because there are approximately 65,000 possible ports, it is possible to multiplex many hosts on one IP address. This has become quite popular in the marketplace, from large organizations all the way down to small home installations.”); EX1023, 2 (“Intuitively, a host only use[s] a small part of port numbers to connect with others. Therefore, if we limit the port that one host can use, then multiple hosts in an IPv6 network can share a same IPv4 address.”).

42. For example, U.S. Patent No. 8,032,639 (“King”), filed May 23, 2006 and granted October 4, 2011, teaches assigning ranges of port numbers to internal

hosts so that the internal hosts can share an IPv4 address while still allowing external devices to identify hosts based on the port numbers assigned to each host. EX1024, 1:15-4:2.

43. Similarly, U.S. Patent No. 8,019,889 (“Biswas”), filed May 31, 2002 and granted September 13, 2011, also teaches assigning port numbers to internal hosts so that the internal hosts can share an IPv4 address. EX1025, 2:16-4:9.

3. Combining the Above Two Ideas

44. Several prior art references, including those discussed in the invalidity grounds below, disclose combining these two ideas: (1) converting IPv4 addresses into IPv6 addresses to make IPv4 devices interoperable with IPv6 devices and (2) incorporating port numbers into constructed IPv6 addresses to prevent IPv4 address exhaustion.

45. For example, an IPv4-IPv6 coexistence mechanism called “IVI”—a merger of the roman numerals four (IV) and six (VI)—was deployed by the China Education and Research Network (CERNET) in 2006. EX1038, 17 (“The IVI gateway based on the Linux implementation has been deployed between CERNET (IPv4 and partially dual-stack) [CERNET] and CNGI-CERNET2 (pure IPv6) [CERNET2] since March 2006.”). IVI forms an IPv6 address by concatenating an IPv6 prefix with an IPv4 address. *Id.*, 6. And by June 21, 2008, the authors of Li had prepared an Internet Draft specification titled “Bi-Directional Address

Borrowing (IVI)” (EX1038) and presentation (EX1026) that extended IVI by further concatenating a port number with the IPv4 address and prefix:

Address/port multiplexing of the global IPv4 address (1)

- **IVI6 can be address/port multiplexed.**
 - Since public-IPv4 addresses have become a precious resource, borrowing a public-IPv4 address with complete exclusivity in a user login session would be a waste. Therefore, a single borrowed IPv4 IVI address can be used for multiple IPv6 IVI addresses.

EX1026, 24; EX1038, 14.

46. No later than July 6, 2008, one week after the '845 patent's priority date, those same authors submitted an updated draft to the IETF titled “Prefix-Specific and Stateless Address Mapping (IVI) for IPv4/IPv6 Coexistence and Transition” (“IVI Memo”) that continued to propose including a port number when forming the IPv6 address. EX1027, 18-19; *see also* EX1028, 20-22 (July 29, 2008). Various other publications from around the time of the '845 patent similarly proposed extending the IPv4 address space by combining an IPv4 address and port number, and further disclosed using this extended IPv4 address to construct an IPv6 address. EX1029, 2 (“[W]e propose to extend the IPv4 address space by assigning to each customer a single IPv4 address which is extended by ‘stealing’ bits from the port number...”); *see also* §VII.B.1 (discussing simultaneous invention of the '845 patent's disclosures).

47. Further, U.S. Patent Application No. 10/401,083 (“Huitema”), filed on March 27, 2003 and later granted on July 17, 2007 as U.S. Patent No. 7,245,622 also recognizes both of these co-existence and address-depletion problems. EX1030. Huitema explains that “[i]n order for one computing system to communicate with another computing system over a particular network, it is important that each computing system be uniquely identified on that network.” *Id.*, ¶5. Huitema further explains that “IPv4 addresses have become scarcer” and that “the problem associated with a limited number of unique addresses under IPv4 may be addressed by reconfiguring the Internet to operate exclusively under IPv6 instead,” which “is sufficient to provide unique addresses for well into the anticipated future.” *Id.*, ¶¶6-8. But “such a reconfiguration is not practical for the time being,” so “it is commonly accepted that the Internet needs to concurrently work with both IPv6 and IPv4, at least for the near future.” *Id.*, ¶8.

48. Huitema describes the following network configuration:

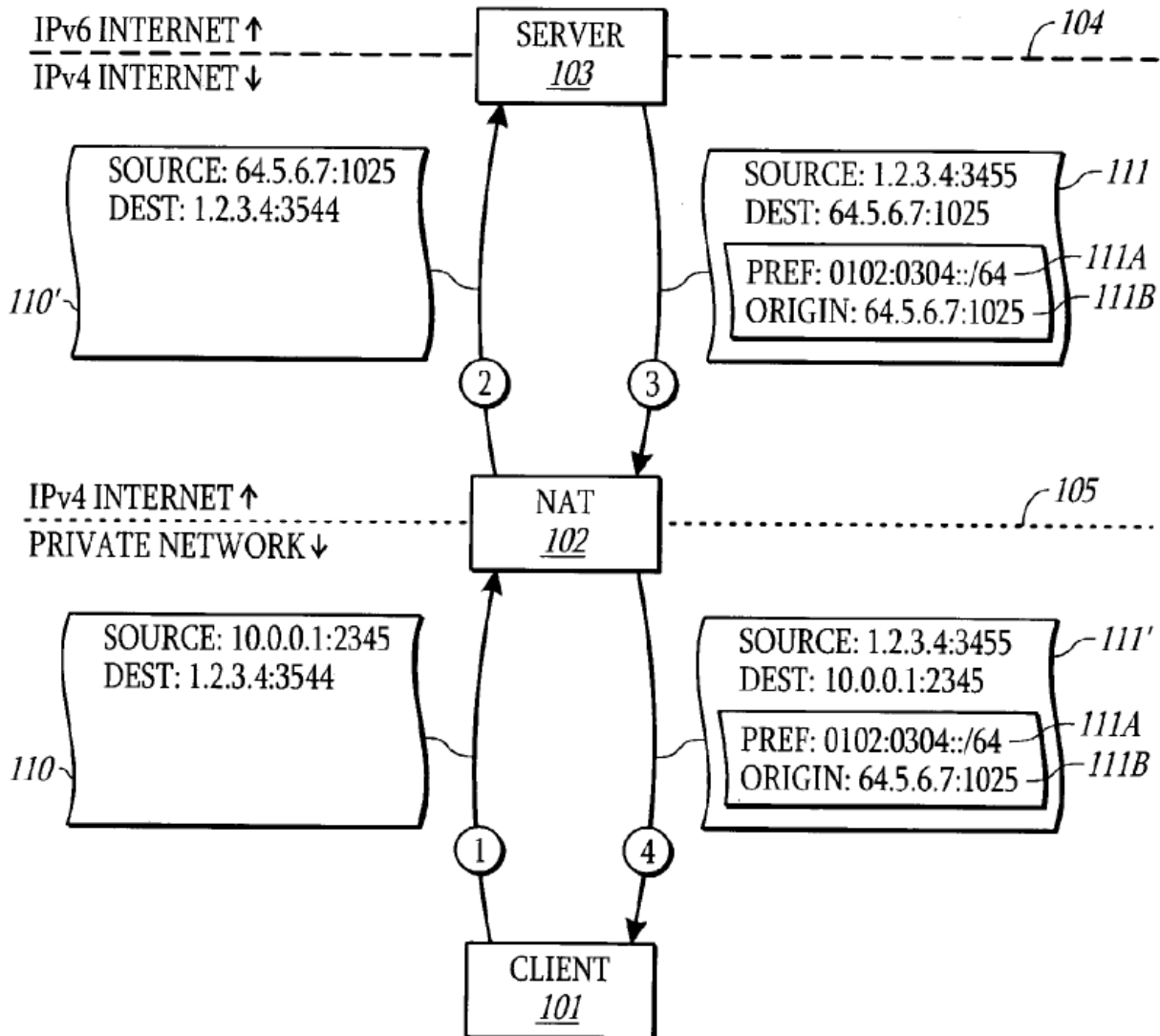


FIG. 1

Id., Fig. 1. In this embodiment, client **101** sits in a private network and needs an IPv6 address “[i]n order to participate in communications with IPv6 computing systems.” *Id.*, ¶49. Client **101** thus requests and receives several pieces of information, (*id.*, ¶¶49-54), in order to construct the following IPv6 address:

PREF:102:304::FBFE:BFFA:F9F8

Id., ¶55. The first portion, “**PREF:102:304**” is an IPv6 prefix that client **101** receives from server **103**. *Id.* The second portion, “**FBFE**” is a port number that client **101** obtains from NAT **102**—port number 1025—converted to hexadecimal and XORed² with hexadecimal number “FFFF.” *Id.* The third portion, “**BFFA:F9F8**” is an IPv4 address that client **101** obtains from NAT **102**—64.5.6.7—converted to hexadecimal and XORed with hexadecimal number “FFFF:FFFF.” *Id.* Accordingly, client **101** constructs an IPv6 address that comprises an IPv6 prefix, a port number, and an IPv4 address. *Id.* In this manner, Huitema is able to address both IPv4-and-IPv6 interoperability and IPv4 address exhaustion.

49. Similarly, U.S. Patent No. 7,149,225 (“Thubert”), filed July 11, 2003 and granted December 12, 2006, teaches using a Teredo address to solve the

² XOR is a binary operation, like an OR operation or an AND operation, that outputs a 1 only if the two input bits are different:

Input Bit #1	Input Bit #2	Output Bit
0	0	0
0	1	1
1	0	1
1	1	0

problems of IPv4-and-IPv6 compatibility and IPv4 address exhaustion. EX1031, 2:30-3:42. Teredo addresses are constructed IPv6 addresses that comprise five components: (1) **an IPv6 prefix**, (2) a server IPv4 address, (3) a flag that designates the type of NAT used, (4) **a port number**, and (5) **the client IPv4 address**:

The Teredo addresses are composed of 5 components:

```

+-----+-----+-----+-----+-----+
| Prefix       | Server IPv4 | Flags | Port | Client IPv4 |
+-----+-----+-----+-----+-----+

```

EX1032, 9.

D. Network Address Translation/Port Address Translation

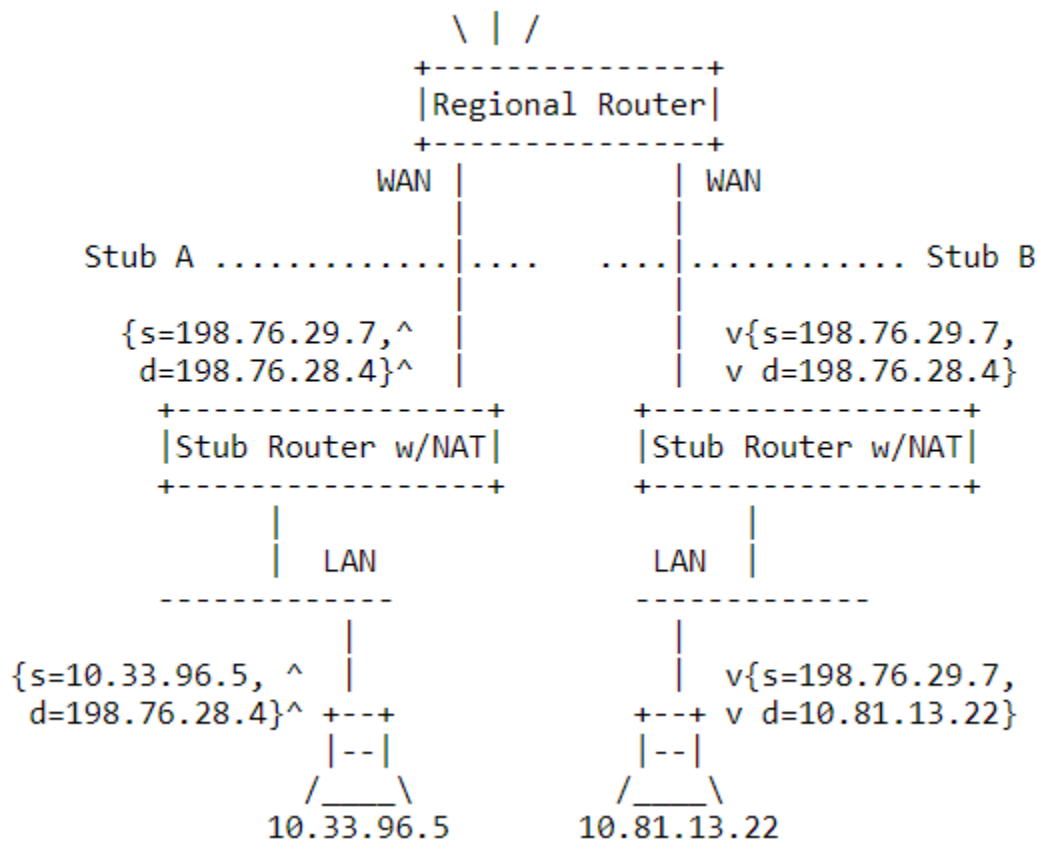
50. Network Address Translation (“NAT”) was developed in order to address “[t]he two most compelling problems facing the IP internet... IP address depletion and scaling in routing.” EX1062 (RFC 1631), 2. By the priority date of the ’845 patent, NAT and its operation was well-known to POSITAs. EX1058, 13-21 (describing well-known forms of NAT, including Basic NAT and NAPT). Indeed, the ’845 patent acknowledges that NAT was well-known in the art by the time of its alleged invention. EX1001, 1:56-60 (“This table is known to the person skilled in the art as a NAT (network address translation) table.”).

51. NAT allows for IP addresses within a local network (*e.g.*, “a stub domain”) to be reused. It performs this process using a network router (*e.g.*, “stub router w/NAT”), wherein the source IP address of outbound data traffic is translated

to a globally unique source address. Inbound traffic can similarly be translated from a globally unique destination address to a reusable local destination address.

52. Figure 2 of RFC 1631 demonstrates this process in its simplest terms.

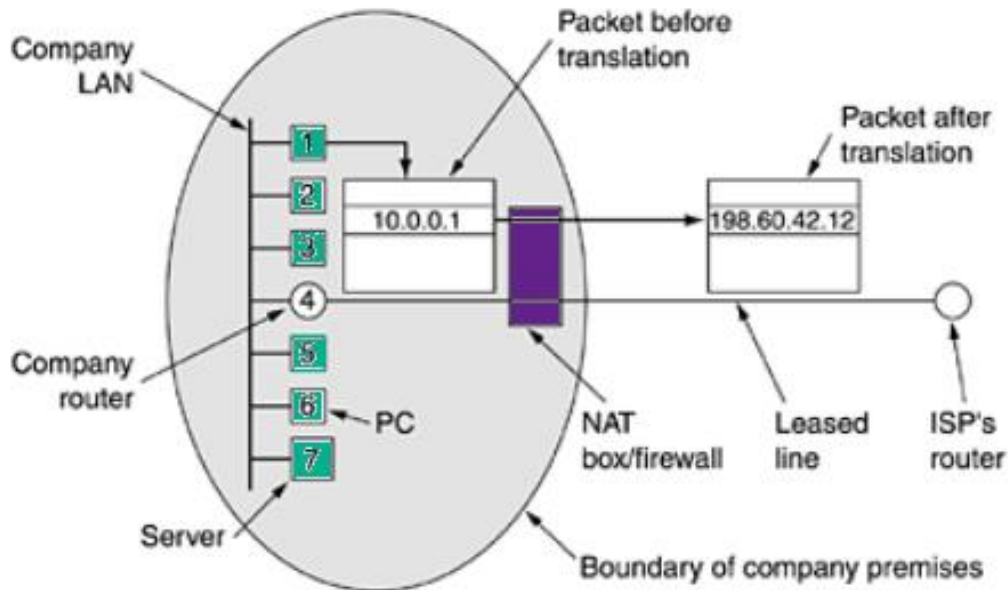
EX. 1062.



53. In figure 2, a data packet is traveling from stub network A (10.33.96.5) to stub network B (10.81.13.22). The data packet uses the global address for the stub B router as its destination. As the data packet departs stub A, stub router A translates the local IP address to a globally unique IP address (198.76.29.7), which travels through a wide area network (WAN) link to stub network B. Stub B router then

translates the packet destination address to the local address (10.81.13.22) to complete the data transfer.

54. The image below shows another example in which a NAT (purple) within a customer network is connected to multiple user terminals (green):



EX1059, 340. “Within the company premises, every machine has a unique address of the form 10.x.y.z” (its private IPv4 address), which is converted to a public IPv4 address (198.60.42.12) by a “NAT” when communicating outside of the internal network. *Id.*; see also *id.*, 341 (“On the way back, packets undergo the reverse mapping.”). While the image above illustrates the NAT and company router as separate devices, POSITAs understood that “[i]t is also possible to integrate the NAT box into the company’s router.” EX1059, 453.

55. Moreover, the examples above illustrates examples of IPv4-to-IPv4 NAT, but by the time of the '845 patent it was also known that NAT could be used to perform IPv6-to-IPV6 translation as well. *See, e.g.*, EX1058, 61 (“This chapter discusses the development and implement of NAT for IPv6”), 89 (“provide NAT functionality for IPv6”); EX1064 (“IPv6-to-IPv6 Network Address Translation (NAT66)”).

IV. OVERVIEW OF THE '845 PATENT

A. Priority

56. I am informed that because the prior art cited in this petition pre-dates the earliest possible priority date of the '845 patent (June 30, 2008), each cited reference herein qualifies as prior art.

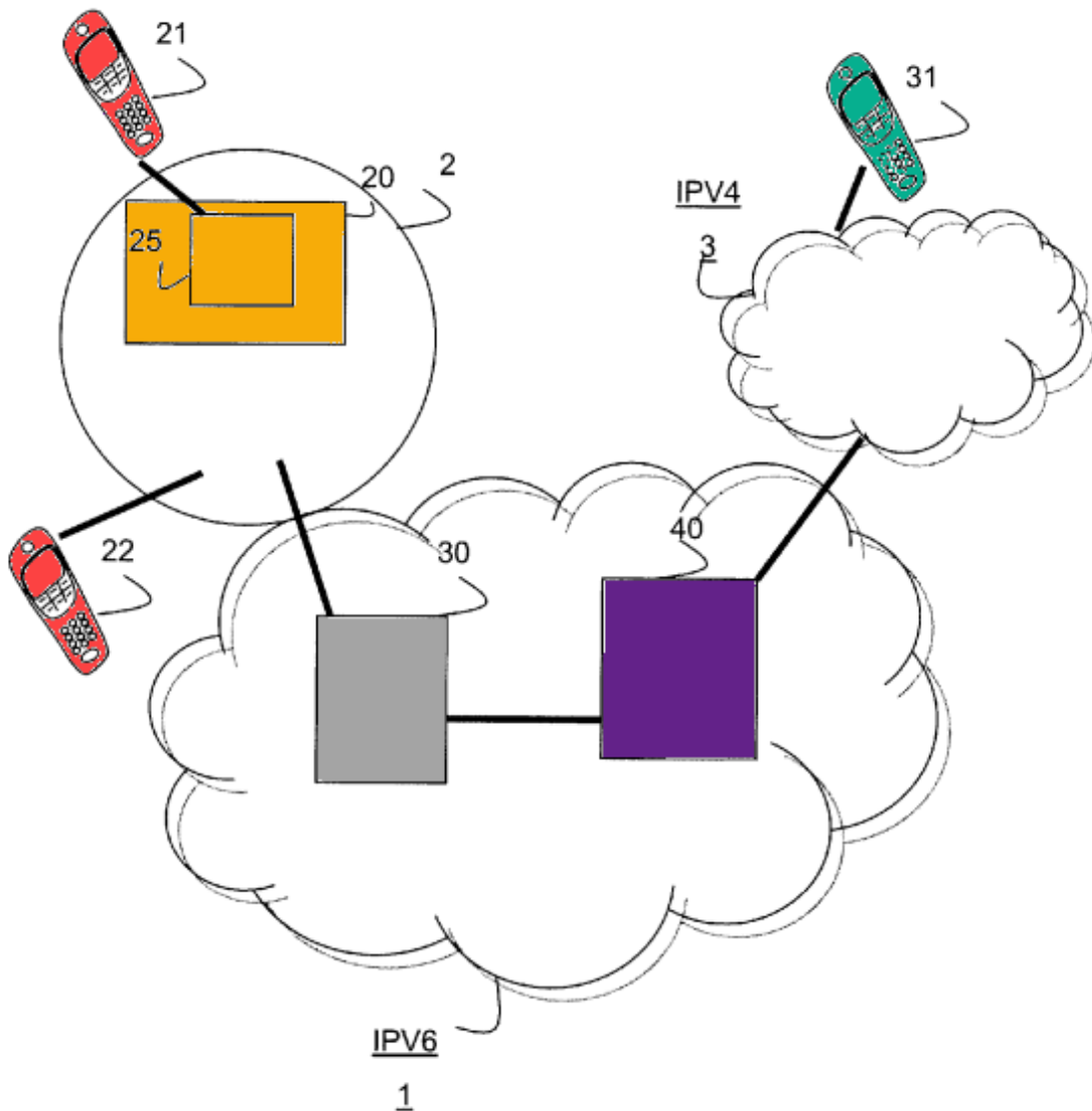
B. Specification

57. The '845 Patent describes an IP network for transmitting data packets from a source terminal identified by a source address to a destination terminal identified by a destination address. EX1001, 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. EX1001, 1:40-48, 4:46-47.

58. 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 packet must be translated to an IPv4 packet, and vice versa. EX1001, 3:18-48.

To facilitate the transition from IPv4 to IPv6 addresses, the '845 Patent discloses that an IPv6 address is constructed by concatenating an IPv6 prefix, an IPv4 address, and a port number. EX1001, 3:7-9, 3:30-48, 7:18-20. In this manner, IPv4 and IPv6 nodes may communicate with one another.

59. For instance, in Figure 1 an IPv4 terminal (green) may communicate with an IPv6 terminal (red). The access node (purple) may receive a packet sent from an IPv4 terminal (green) and construct an IPv6 address by concatenating an IPv6 prefix, an IPv4 address, and a port number. The packet with the constructed address is then routed to a gateway (orange) through an access router (grey) where the constructed address is replaced with a native address and then routed to an IPv6 terminal (red):



EX1001, 11:6-61.

V. LEVEL OF ORDINARY SKILL IN THE ART

60. I understand that a person of ordinary skill in the art (“POSITA”) is a hypothetical person who is presumed to be aware of all pertinent art, thinks along conventional wisdom in the art, and is a person of ordinary creativity—not an automaton. In deciding the level of ordinary skill, I understand that the following

factors may be considered: the levels of education and experience of persons working in the field, the types of problems encountered in the field, and the sophistication of the technology.

61. I understand that the asserted patents must be evaluated from the perspective of a POSITA. In my opinion, the relevant art for this patent relates to the field of computer networks and routing packets in a computer network. I understand that the relevant point in time for determining the qualifications of a POSITA is the time of the alleged invention, which I assume to be the earliest filing date for the patent. Here, I understand that the earliest invention date contended by Patent Owner is June 30, 2008.

62. In my opinion, a POSITA at the time of the invention (*i.e.*, in June 2008) would have had a Bachelor of Science 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. I meet these criteria and have applied it in my analysis.

VI. CLAIM CONSTRUCTION

63. I have been informed that, in a prior district court litigation brought by Patent Owner in Texas, the Texas court construed certain terms found in the '845

patent. EX1036. These constructions are identified below, and I have applied them in my analysis for the purposes of this proceeding. I have also applied Monarch's proposed constructions in my analysis below, to the extent they are different.

Claim(s)	Term	Construction
1, 8	"IPv4 destination address"	"address formatted according to Internet Protocol version 4 indicating a destination"
1, 8	"concatenating"	"appending one item to the end of another so as to form a single unit in a contiguous pattern"
1, 8	"home gateway"	"any equipment for interconnecting a private network and a network operated by a service provider, the private network being either a home network or a business network"
1, 6, 8	"destination port number"	"number indicating a logical point of connection with a destination device"
1, 2, 5, 8	"regularizing"	Plain and ordinary meaning
1, 8	"if necessary"	"if the IPv6 source address is classified as a native address or if the IPv6 destination address is a constructed address"
1, 8	"modifying the data packet using the regularized address"	Plain and ordinary meaning
1, 8	"the replacement process belonging to a group comprising replacement of a constructed address with a native address and replacement of a"	Plain and ordinary meaning

	native address with a constructed address”	
8	“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” Corresponding Structure: “the home gateway 20 that includes a processor programmed to perform the step of identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address, and a destination port number”
8	“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”	Function: “regularizing at least one of the IPv6 source address and the IPv6 destination address of the data packet and modifying the data packet using the regularized address” Corresponding Structure: “the home gateway 20 that includes a processor programmed to perform the step of regularizing at least one IPv6 address when: 1. if the IPv6 source address is a native address, then the program executes the steps of choosing an IPv6 constructed source address whose source port number is in the authorized range assigned to the home gateway 20; and modifying the data packet by replacing the native source address with the constructed source address as illustrated in steps R21 and R22 of Figure 5; or 2. if the IPv6 destination address is a constructed address, then the program executes the steps of searching a translation table of the home gateway 20 and locates a

		native destination address associated with the constructed destination address; verifying that the transport level destination port of the data packet agrees with the destination port number contained in the destination address of the packet; and modifying the data packet by replacing the constructed destination address with the native destination address as illustrated in steps R'21, R'22, and R'23 of Figure 8"
8	"means for routing the IPv6 data packet to its destination"	Function: "routing the IPv6 data packet to its destination" Corresponding Structure: "the home gateway 20 that includes a processor programmed to perform the step of routing the IPv6 data packet to its destination"

VII. IDENTIFICATION OF HOW THE CHALLENGED CLAIMS ARE UNPATENTABLE

A. Ground 1: Claims 1, 5-8 Would Have Been Obvious in View of Li and Li-2

1. Li (EX1037)

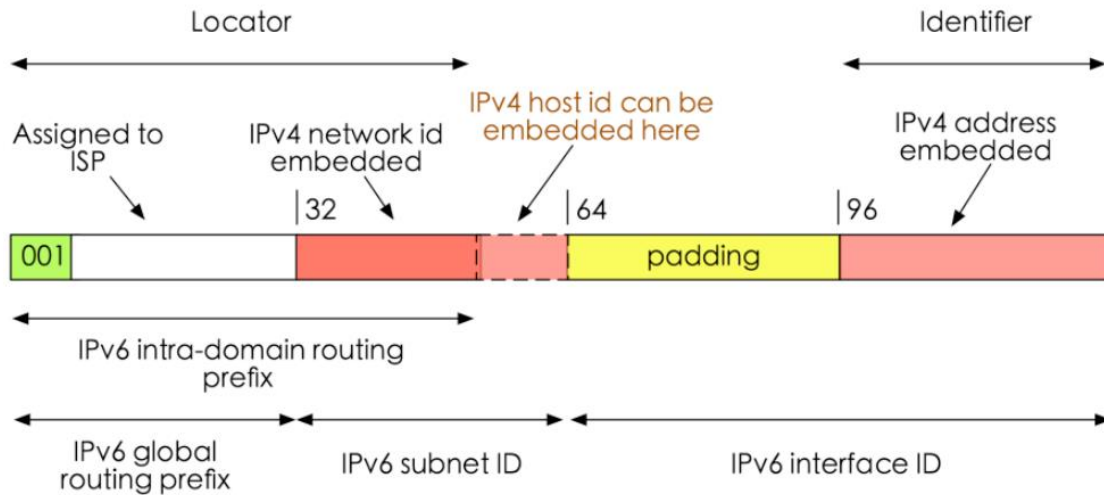
64. Li (EX1037) is a paper entitled "Forwarding IPv4 Traffics in Pure IPv6 Backbone with Stateless Address Mapping" by Xing Li *et al.* that was published by the Institute of Electrical and Electronics Engineers ("IEEE") in April 2006, in connection with a conference that occurred on April 3-7, 2006 in Vancouver, Canada. EX1037. I understand that Li is prior art to the '845 patent because it was publicly accessible more than a year before the priority date of the '845 patent.

65. It is also my opinion that the IEEE is a well-known, reputable organization and its website is known by POSITAs and routinely used by POSITAs

to access technical publications and papers. In my experience, the copyright date that is disclosed on the face of an IEEE paper—in this case, 2006—is the year that the paper was made publicly available and interested persons would have been able to locate Li exercising reasonable diligence. Based on my experience with the standard IEEE practices, Li would have been available to the public no later than the last day of the conference, *i.e.*, April 7, 2006. Because of the IEEE’s eminence in the industry as a source for standards and innovative technical publications, a POSITA would have been a member of the IEEE or, at the very least, would have followed its publications and therefore would have been able to locate Li exercising reasonable diligence.

66. Li concerns the transition between IPv4 and IPv6. EX1037, 1. Li discloses a process for mapping IPv4 packets into IPv6 packets (or vice versa). EX1037, 4 (“An edge router manipulates the IP headers for the packets passing through it from IPv6 to IPv4 or vice versa.”), Fig. 4. The process involves “address mapping,” which constructs an IPv6 source and IPv6 destination address from an IPv4 source and destination address. *Id.*

67. In particular, Li teaches that address mapping constructs an IPv6 address from an IPv4 address by concatenating a 32-bit IPv6 prefix, a 32-bit IPv4 address, and 32 bits of zero-padding. Figure 2 provides an example of how an IPv6 address is constructed from an IPv4 address using Li’s mapping technique:



Id. The constructed IPv6 address begins with an “IPv6 global routing prefix” (bits 0-31), it is then followed by an “IPv4 address” (bits 32-63), and then 32 bits of zero “padding” (bits 64-95). *Id.* For reasons not relevant here, the IPv4 address may also be repeated at the end of the IPv6 address (bits 96-127).

68. Li provides examples of address mapping in Table 1:

TABLE I. SOME ADDRESS MAPPING EXAMPLES

IPv4 prefix/address	IPv6 prefix/address
202.112.0.0/13	2001:da8:ca70::/45
166.111.0.0/16	2001:da8:a66f::/48
166.111.172.1	2001:da8:a66f:ac01::a66f:ac01

Id. (highlighted). In this mapping, Li constructs an IPv6 address in the right column by appending an IPv6 prefix (2001:da8) to the IPv4 address or address range in the left column. *Id.* It is critical to note that IPv4 addresses are typically represented in dotted-quad notation, whereas IPv6 addresses are represented in hexadecimal notation, so while the IPv4 address on the left side appears different from the right side, they are actually the same. For example, the third row in Table 1 contains the IPv4 address 166.111.172.1. *Id.* Translating this address from dotted-quad notation to hexadecimal yields the following result:

$$\begin{array}{ccccccc}
 \mathbf{166} & \mathbf{111} & \mathbf{172} & \mathbf{1} & & & \\
 \downarrow & \downarrow & \downarrow & \downarrow & & & \\
 \mathbf{a6} & \mathbf{6f} & \mathbf{ac} & \mathbf{1} & = & \mathbf{a66f:ac01}
 \end{array}$$

The translation is merely aesthetic, for readability. The translation does not alter the IPv4 address itself—the IPv4 address on the left is the exact same 32-bit address as the one on the right. Thus, the resulting IPv6 address is 2001:da8:a66f:ac01::a66f:ac01,³ where “2001:da8” is the IPv6 prefix (blue),

³ In this example, the IPv4 address is repeated twice in the constructed IPv6 address.

Li explains that issues arise with defining prefixes that cover both networks inside

“a66f:ac01” is the IPv4 address in hexadecimal notation (red), and the notation “::” represents zero-bit padding (yellow), in which bits 64 to 96 are zeros. Thus, Li teaches constructing an IPv6 destination address by concatenating an IPv6 prefix, an IPv4 destination address, and zero padding.

69. Figure 7 of Li depicts the following network configuration as one embodiment that implements its invention:

and outside of the autonomous system. Li therefore reserves the last 32 bits of the IPv6 address for the original IPv4 address in order to define a “common convention” and solve this problem. EX1037, 3. Thus, the IPv4 address “a66f:ac01” is repeated at the end of the constructed IPv6 address (*i.e.*, the last 32 bits) to adhere to this convention. *Id.*

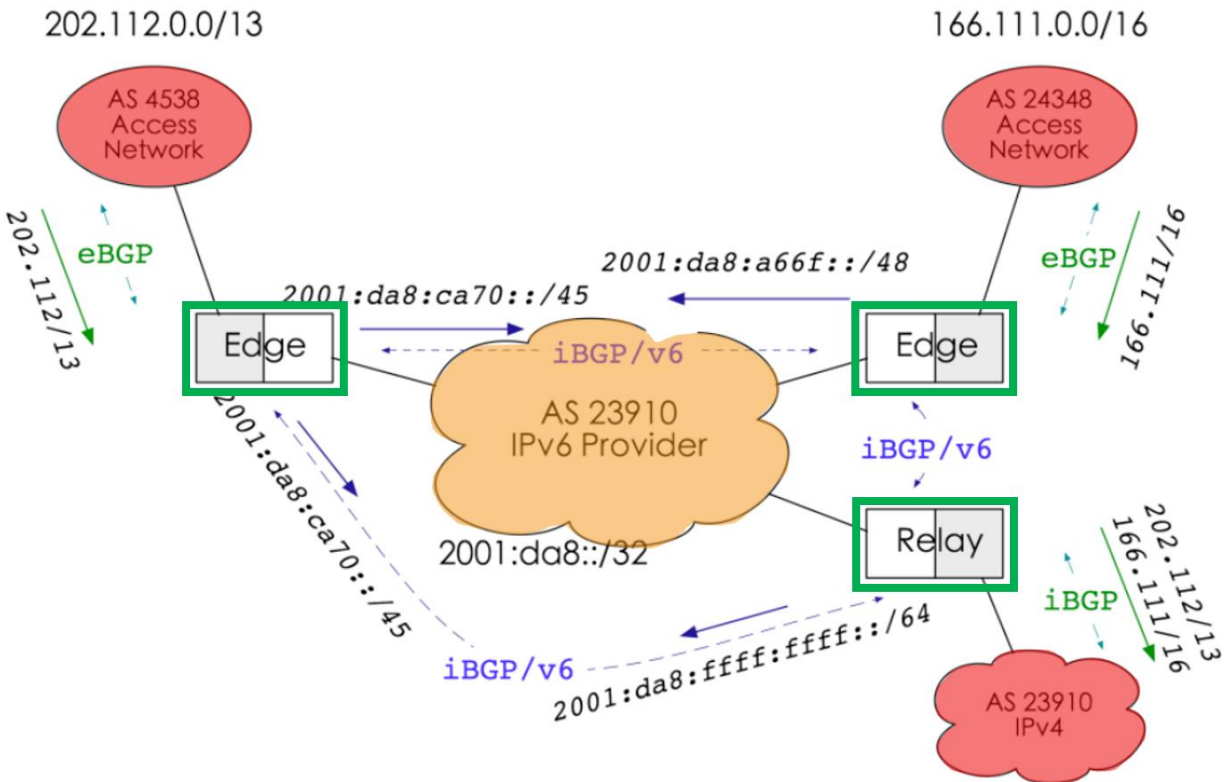


Figure 7. IPv4-over-IPv6 transition for access networks

EX1037, 7 (highlighted). At the center of the network is an IPv6 domain (orange). *Id.*, 3. Surrounding the IPv6 domain are several IPv4-only domains (red), called access networks. *Id.* Edge routers and edge relays (surrounded by green borders) interconnect the IPv6 domain and the IPv4 domains. *Id.* In this embodiment, Li's edge routers are deployed on the boundary of the access network and the provider backbone and implement the address mapping. EX1037, 4 ("An edge router manipulates the IP headers for the packets..."), 4 ("The routers at the edge of IPv4 subnets and the IPv6 backbone perform the functionality of mapping addresses for packets and translating routing information between IPv6 and IPv4."), 7 ("Relays

and edge routers are doing the header mapping as is described in Fig. 4.”), 9 (“The edge router is deployed on the boundary of the access network and the provider backbone. It announces to the backbone an IPv4-mapped IPv6 prefix for the whole access network. It translates headers for all packets coming in or going out of the access network.”).

70. Figure 11 of Li builds upon the architecture in Figure 7 by disclosing that the IPv6 provider network (AS 23910) may be connected to native IPv6 access networks that also supports IPv4 (*e.g.*, AS 24349):

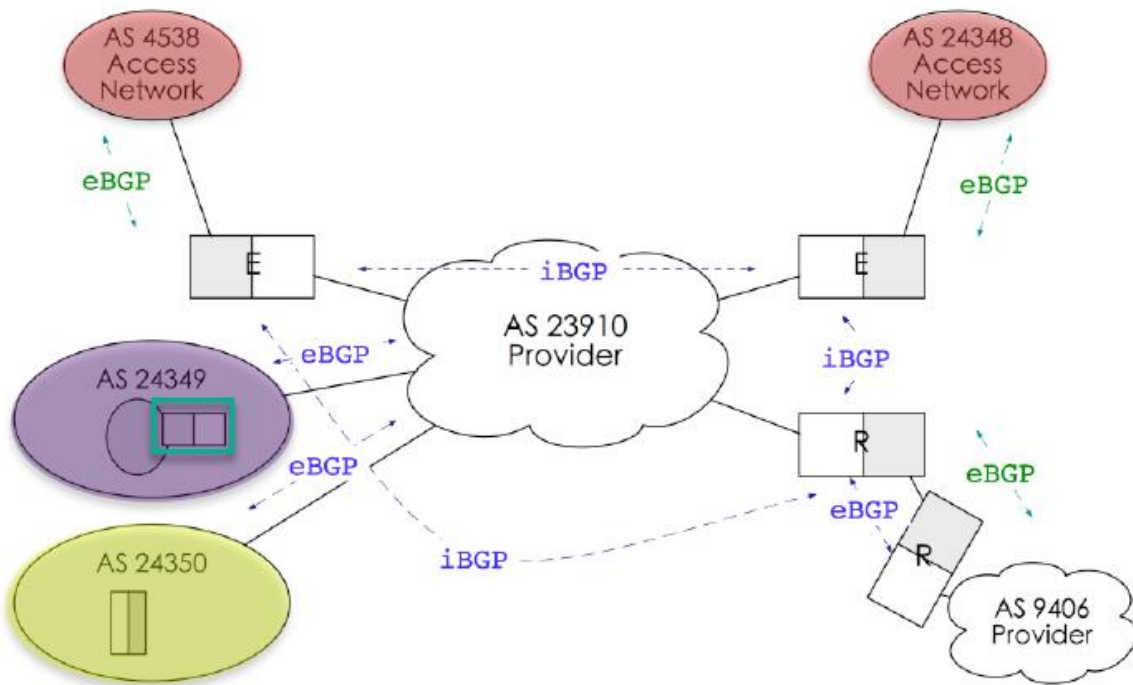


Figure 11. Migration with 3 steps

EX1037, 9 (highlighting added). For instance, Figure 11 shows the “3 typical stages in migration,” wherein the first stage is IPv4-only domains (AS 24348 and AS4538

in red), the second stage is migration to “a native IPv6 access network containing a small group of IPv4 subnets” (AS 24349 in purple) and a third stage is migration to “native IPv6 access network containing separated dual-stack hosts” (AS 24350 in yellow). *Id.*, 9-10. Li discloses that for native IPv6 access networks, a router within the access network (*e.g.*, the router shown in AS 24349 that is surrounded by a green box) may implement the address mapping rather than an edge router on the border of the IPv6 and IPv4 networks. *Id.*, 10 (“The route mapping and header mapping will be performed within the scope of the access network AS 24349”).

71. Li and the '845 patent are analogous art because the references are in the same field of endeavor as each other, *i.e.*, computer networking and the IPv4-to-IPv6 transition. For example, the '845 patent explains that “[t]he field of the invention is that of telecommunications networks, in particular IP telecommunications networks transporting data packets.” EX1001, 1:17-21. Li similarly explains that it is directed to “a new technique of forwarding IPv4 traffics in IPv6-only backbone” and includes “routing” as one of its “keywords.” EX1037, 1.

2. Li-2 (EX1038)

72. On June 21, 2008, the authors of Li extended the foundation provided by Li in an Internet Draft published with the IETF. EX1038 (“Li-2”). I understand

that Li-2 is prior art to the '845 patent because it was publicly accessible before the priority date of the '845 patent.

73. The IETF is an “Internet Engineering Task Force” that was founded in 1986 to “make[] voluntary standards that are often adopted by Internet users, network operators, and equipment vendors.” *Introduction to the IETF*, IETF, available at <https://www.ietf.org/about/introduction/> (last accessed Dec. 13, 2023). The IETF is a well-known, reputable organization and its website is routinely used and known by POSITAs to access technical documents, like Internet Drafts, that are publicly published on the IETF website.

74. The IETF has a “commitment to making [its] documents, [its] Working Group mailing lists, [its] attendance lists, and [its] meeting minutes publicly available on the Internet.” *Introduction to the IETF*, IETF, available at <https://www.ietf.org/about/introduction/> (last accessed Dec. 13, 2023). One type of document that the IETF makes publicly available are Internet Drafts, which are “working documents of the IETF, its Areas, and its Working Groups.” *Internet-Drafts*, IETF, available at <https://www.ietf.org/how/ids/> (last accessed Dec. 13, 2023). The IETF makes Internet Drafts publicly available on its website and provides a way to search Internet Drafts by using the IETF Datatracker. *Id.*

75. As disclosed on the face of the document, Li-2 was submitted on June 21, 2008. EX1038, 1. In my experience, the date that is disclosed on the face of an Internet Draft is the date that the Internet Draft was made publicly available.

76. Li-2 recognizes that “IPv6 deployment in the past 10 years strongly indicate that for a successful transition, the IPv6 hosts need to communicate with the global IPv4 networks.” EX1038, 4. However, current transition methods, like dual-stack hosts do not solve the “IPv4 address depletion problem.” *Id.* And other options, like Network Address Translation “lost the scalability and end-to-end address transparency.” *Id.* Li-2 therefore describes a scheme that converts IPv4 addresses into IPv6 addresses to address these issues. *Id.* Li-2 explains that “[b]ased on this scheme, special IPv6 address blocks are defined, which can communicate with the global IPv6 network directly and with the global IPv4 networks via stateless (or almost stateless) gateways.” *Id.*, 1-2. This addressing scheme confers several benefits, including “end-to-end address transparency, being globally routable, incremental deployable, having multi-homing support and being able to use the global IPv4 addresses effectively.” *Id.*, 2.

77. Like the 2006 Li paper above, (EX1037), Li-2 describes an address mapping in which an IPv6 address is constructed by concatenating a 32-bit IPv6 prefix, a 32-bit IPv4 address, and 55 bits of zero padding:

IVI Address Mapping

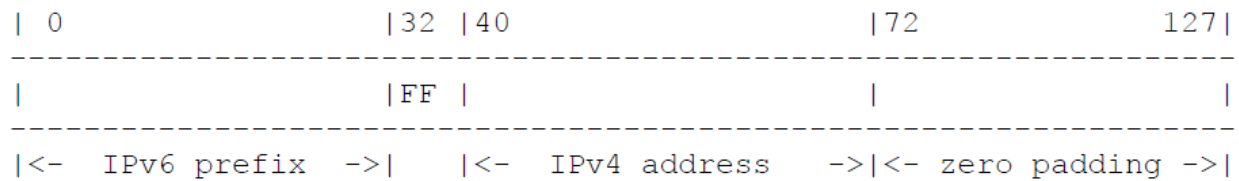


Figure 1

EX1038, 6.

78. However, Li-2 recognizes that this addressing scheme does not solve the problem of IPv4 address depletion. “Since public-IPv4 address is a scarce resource, the effective use of the IPv4 address is important for IVI scheme.” *Id.*, 14. Li-2 therefore expands upon the 2006 paper by introducing an idea of “multiplexing,” specifically “port multiplexing,” to mitigate IPv4 address depletion. *Id.*

79. Port multiplexing works by further replacing the “least significant bits” of the constructed IPv6 address—currently occupied by unused zero padding—with a port number. *Id.* (emphasis added). The relevant embodiment is “1-to-1 mapping,” which means the mapping process “is stateless.” *Id.* Li-2 explains that port multiplexing “increase[s] the utilization ratio of the public IPv4 address” by allowing a “single borrowed IPv4 address” to “be used for multiple IVI6(i) addresses.” *Id.*

80. In fact, just one month later, the authors of Li and Li-2 provided examples of this stateless mapping algorithm in a presentation that they gave in Dublin for an IETF conference. EX1028; EX1007, ¶11 (confirming public accessibility); EX1005, ¶53-57 (same). These examples, which I explain below, further confirm that Li-2 teaches concatenating the destination port number to the IPv6 address by replacing unused zero padding. For example, the authors provided the following examples:

```
202.38.108.2#81 --> IVI61=2001:250:ffca:266c:0200::81#81
202.38.108.2#82 --> IVI61=2001:250:ffca:266c:0200::82#82
202.38.108.2#83 --> IVI61=2001:250:ffca:266c:0200::83#83
202.38.108.2#84 --> IVI61=2001:250:ffca:266c:0200::84#84
```

EX1028, 22 (highlighting added). The IPv6 prefix is the first box, highlighted in blue. EX1038, 6 (“[B]it 0 to bit 31 are the prefix of ISP(i)’s /32.”). The second box, highlighted in red, is a flag that specifies that the IPv6 address has been constructed according to IVI. *Id.* (“[B]it 32 to bit 39 are all one’s as the identifier of IVI.”). The third box, which highlights the next 32 bits in purple, is the IPv4 address from the left side, translated from decimal to hexadecimal. *Id.* (“[B]it 40 to bit 71 are the global IPv4 space (IPG4) presented in hexadecimal format.”). The fourth box, highlighted in yellow, is zero padding. *Id.* (Fig. 1). And the fifth box, highlighted in gray, is the port number that has been taken from the left side of the

translation and concatenated to the end of the constructed IPv6 address, replacing unused zero padding as appropriate. *Id.*, 14.

81. I note that the authors did not translate, perhaps for the sake of convenience, the port number from decimal (*i.e.*, “81”) to hexadecimal (*i.e.*, “51”). However, a POSITA would have understood that the port number extracted from the IPv4 data packet on the left-hand side of the equation is the exact same 16-bit number that is concatenated to form the IPv6 address on the right-hand side of the equation.

82. Finally, as discussed, the double-colon notation—“::”—indicates a string of zeros from the bit preceding the double-colon notation to the bit following the double-colon notation. *See* §III.B. If there is no number following the double-colon notation, then that indicates the zero padding continues to the end of the IPv6 address. Thus, by adding the destination port number to the end of the IPv6 address, that destination port number replaces unused zero padding.

83. Li-2 and the ’845 patent are analogous art because the references are in the same field of endeavor as each other, *i.e.*, computer networking and the IPv4-to-IPv6 transition. For example, the ’845 patent explains that “[t]he field of the invention is that of telecommunications networks, in particular IP telecommunications networks transporting data packets.” EX1001, 1:17-21. And Li-2 is also directed to a system that allows “IPv6 hosts... to communicate with the global IPv4 networks.” EX1038, 4.

3. The Li-Li-2 Combination

84. Li teaches the general network structure and the steps of receiving a data packet from an IPv4 domain in an IPv6 domain. Li further teaches constructing an IPv6 address by concatenating an IPv6 prefix, an IPv4 address, and 32-bits of zero padding, as well as the step of regularizing an address by replacing a constructed address with a native address. Li-2 supplements Li's disclosures by teaching concatenating a destination port number to the end of the constructed IPv6 address (replacing any unused zero padding as necessary) in order to uniquely identify IPv4 hosts and mitigate IPv4 address exhaustion. Thus, the Li-Li-2 combination would implement address mapping to construct IPv6 addresses by concatenating an IPv6 prefix, an IPv4 destination address and a destination port number. The address mapping would be implemented by the edge routers of Li and routers within a native IPv6 network such as the router within AS 24349.

4. Motivations to Combine Li and Li-2

85. A POSITA would have been motivated to combine the teachings of Li and Li-2 and would have had a reasonable expectation of success of doing so.

86. **First**, a POSITA would have been motivated to combine the two references because Li-2 is an extension of Li. The references name the same authors. EX1037, 1; EX1038, 1. Both describe techniques for IPv4 to IPv6 transition that were developed as part of "the China Education and Research Network

(“CERNET”) under the framework of the China Next generation Internet (CNGI) project.” EX1037, 10; EX1038, 17. Indeed, Li describes a gateway implementation for CERNET/CNGI that performs address mapping (EX1037, 10) and Li-2 describes a further implementation of this “IVI gateway” that was “deployed between CERNET... and CNGI-CERNET2” that uses an evolution of this “IVI address mapping” (EX1038, 6, 17). In fact, Li itself notes that its initial system “is just a small step” and “is an extensible foundation” that could be modified further. *Id.*, 2.

87. **Second**, the prior art includes express teachings that would have led a POSITA to combine Li and Li-2. As discussed above, several prior art documents expressed a need to not only make IPv4 domains interoperable with IPv6 domains in order to smooth the transition from IPv4 to IPv6, but also a need to slow IPv4 address exhaustion by sharing a single public IPv4 address among multiple hosts. See §III.C. Based on the information in the prior art, a POSITA would have known that (1) IPv4 addresses could be converted into IPv6 addresses to allow for IPv4-IPv6 interoperability, (see §III.C.1); and (2) port numbers could be used to extend the IPv4 address space by allowing multiple hosts to share a single public IPv4 address, (see §III.C.2). Indeed, Li-2 expressly discloses these teachings and explains that they are the underlying motivation for Li-2’s port multiplexing extension of the original Li paper. EX1038, 4, 14. A POSITA would have had a reasonable expectation of success in combining Li and Li-2, especially because a POSITA did

succeed in combining these references, since Li-2 is an extension of Li authored by the same inventors of the original Li paper.

88. **Third**, the combination merely involves the straightforward application of replacing unused padding in Li's constructed IPv6 address with the destination port number, as taught by Li-2. Li-2 shows that this substitution and its function was known in the art, and describes the function of this substitution. EX1038, 14. And the results would have been predictable, as a POSITA actually *did* make this substitution, as described in Li-2. See EX1038, 14.

89. **Fourth**, this combination would have involved combining prior art elements according to known methods to yield predictable results. Li and Li-2 recite all elements of the '845 claims, as described in the claim-by-claim analysis below. A POSITA could have combined the elements as claimed by known methods, which are described in Li-2. In fact, Li-2 is an extension of the ideas described in the original Li reference and was published by the same authors that published the original Li paper in 2006.

90. **Finally**, Li, Li-2, and the '845 patent are analogous art because the references are in the same field of endeavor as each other, *i.e.*, computer networking and the IPv4-to-IPv6 transition.

5. Claim 1

- (a) 1[pre]: A method of receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain, said packet

comprising an IPv6 destination address and an IPv6 source address, said method being executed in a home gateway adapted to connect a user terminal to said IPv6 domain, and comprising the following steps:

91. To the extent the preamble is limiting, Li discloses 1[pre]: “a method of receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain, said packet comprising an IPv6 destination address and an IPv6 source address, said method being executed in a home gateway adapted to connect a user terminal to said IPv6 domain, and comprising the following steps.” The elements of this limitation are broken down below.

92. ***A method of receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain:*** In this ground, I walk through two examples of Li disclosing a method of receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain.

93. **First**, Li discloses an IPv4 host in the top-left IPv4 access network sends a data packet to a host in the top-right IPv4 access network, as depicted by the black arrow below:

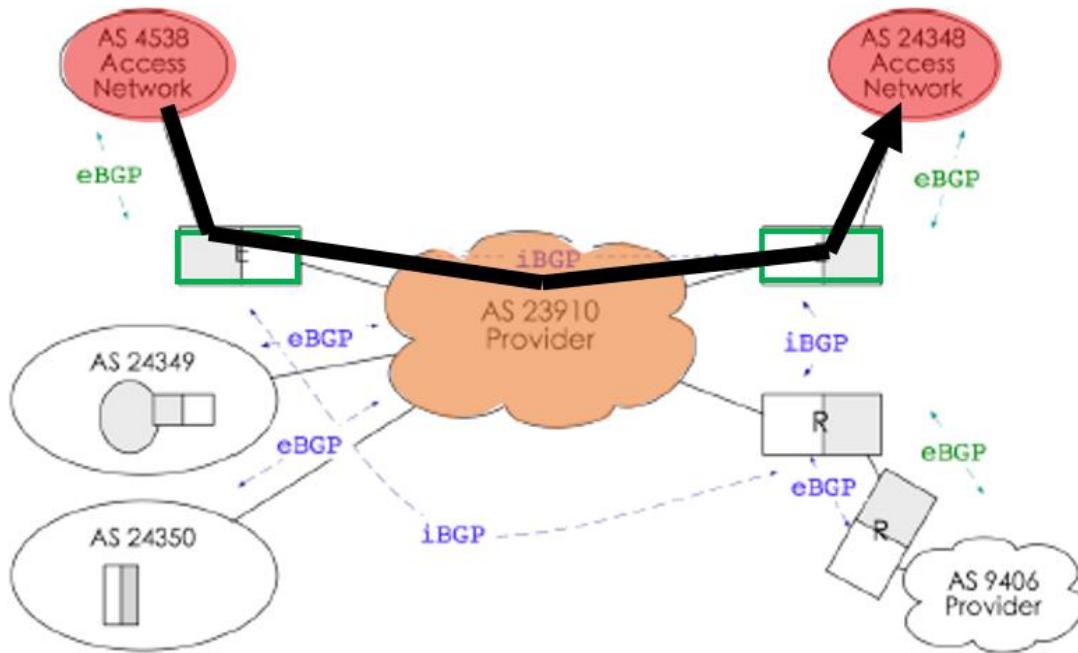


Figure 11. Migration with 3 steps

EX1037, 9 (highlighted and annotated), 7 (Figure 7 illustrating similar architecture). In this example, at the center of the network is an IPv6 domain (orange) that is connected to and routes communications between the IPv4-only domains (red). Edge routers (outlined in green) interconnect the IPv6 and IPv4 domains.

94. Li discloses that the edge routers are configured to receive IPv6 data packets. For instance, Li teaches that the edge router's role is "an access gateway that provides connectivity for a certain IPv6 block." *Id.* 4. It further states that "[a]n edge router is a dual-stack node, extended with the ability of translating routing messages and modifying packet IP versions between IPv6 and IPv4." *Id.* A POSITA

would understand a dual-stack node is a device that processes (sends and receives) both IPv4 and IPv6 packets. *See* EX1037, 4 (“An edge router manipulates the IP headers for the packets passing through it from IPv6 to IPv4 or vice versa.”).

95. A POSITA would understand that the IPv6 data packets are received in an IPv6 domain. Each edge router has an IPv6 interface for receiving IPv6 packets, and utilizes an IPv6 prefix (*e.g.*, 2001:da8) of the IPv6 domain (AS 23910). EX1037, 7, 9. Thus, for instance, the edge router in the top right of Figure 11 receives IPv6 data packets in the IPv6 domain on its IPv6 interface.

96. **Second**, Li also discloses an IPv4 host in the top-left IPv4 access network sends a data packet to a host in a “native IPv6 access networks” (*e.g.*, AS 24349 in purple) via the IPv6 network (orange):

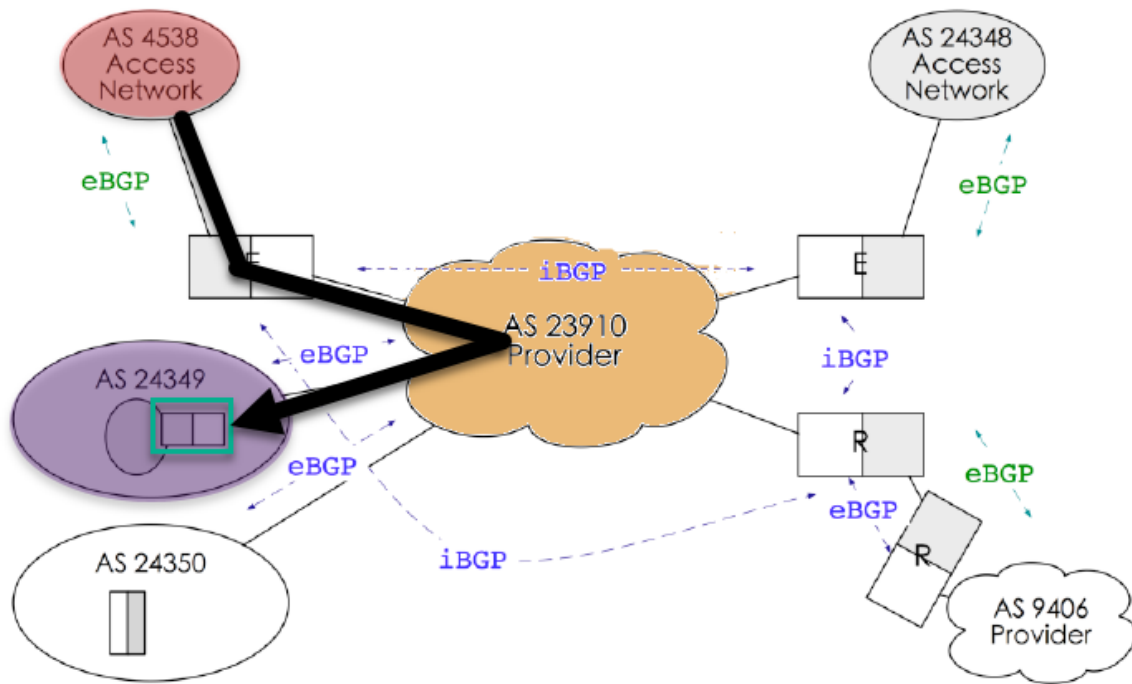
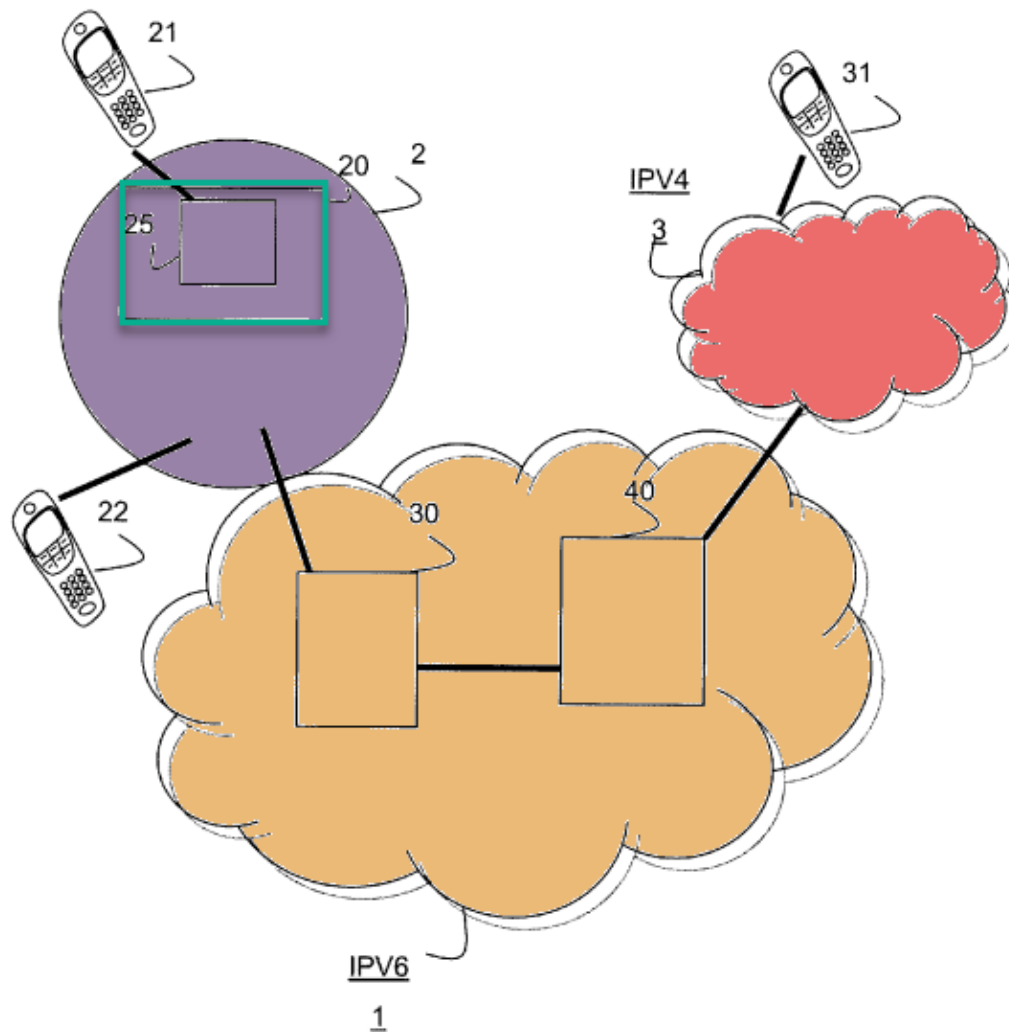


Figure 11. Migration with 3 steps

EX1037, 9 (highlighting added). In this example, the router may be implemented within AS 24349, which Li describes as a “native IPv6 access network containing a small group of IPv4 subnets.” *Id.*, 10 (“The route mapping and header mapping will be performed within the scope of the access network AS 24349”). Because the router is implemented within a “native IPv6 access network” and receives IPv6 packets, a POSITA would understand that the IPv6 data packets are received in an IPv6 domain. In fact, this example is similar to the architecture illustrated in Figure 1 of the ’845 patent, which discloses an IPv4 network (red) connected to an IPv6 network (purple) having a gateway (green) via an IPv6 network (orange):



EX1001, Fig. 1.

97. ***said data packet comprising an IPv6 destination address and an IPv6 source address:*** As discussed above, Li's edge routers receive IPv6 data packets that are routed across a "pure IPv6 backbone" (*e.g.*, AS2391). EX1037, 1 ("IPv4 packets are mapped into IPv6 (or vice versa) automatically at the edge of the IPv6 backbone."). A POSITA would know that data traffic routed across a pure IPv6 network possesses an IPv6 source address and an IPv6 destination address.

98. Indeed, Li discloses that in the outbound direction an edge router (*e.g.*, edge router in the top left of Figure 11) receives an IPv4 packet from an IPv4 domain and uses Li's address mapping to construct an IPv6 data packet containing IPv6 "source and destination addresses" that it routes through the IPv6 backbone (AS 23910). *See* EX1037, 3 ("IPv4 source and destination addresses in a packet can be translated into their IPv6 version"), Table 1 (showing IPv4 addresses being translated to IPv6 addresses). Li illustrates the translation of an IPv4 header to an IPv6 header of a packet, including its "source address" and "destination address":

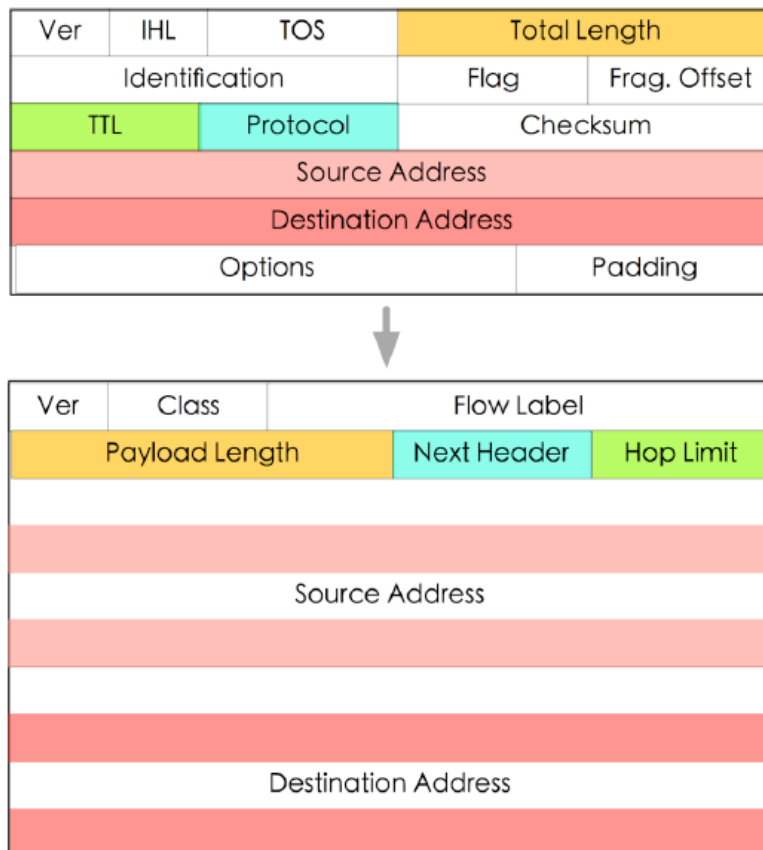


Figure 4. Mapping the header from IPv4 to IPv6

EX1037, 4. Li also teaches that the opposite process occurs in the inbound direction, with the edge router receiving an IPv6 packet with the constructed IPv6 source and destination addresses. EX1037, 4 (“An edge router manipulates the IP headers for the packets passing through it from IPv6 to IPv4 or vice versa.”).

99. ***said method being executed in a home gateway adapted to connect a user terminal to said IPv6 domain, and comprising the following steps:***

100. In the Texas Litigation, Patent Owner agreed and the Eastern District of Texas construed “home gateway” to mean “any equipment for interconnecting a private network and a network operated by a service provider, the private network being either a home network or a business network.” See EX1036, 13.

101. Li discloses a “home gateway” under this construction, for example:

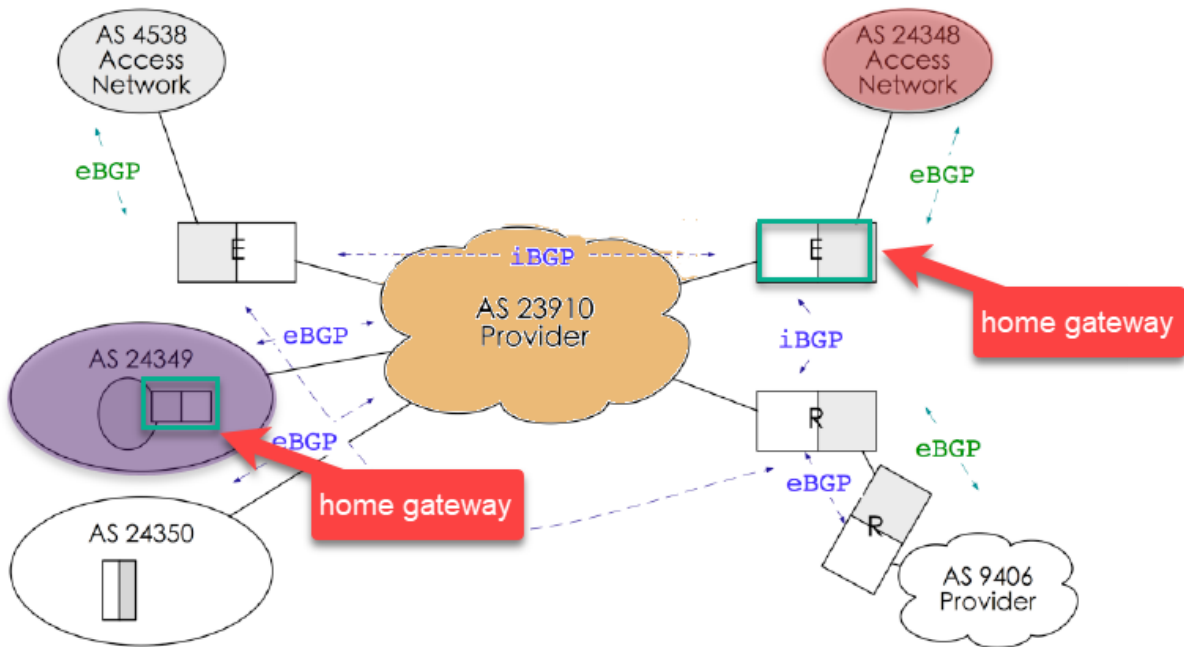


Figure 11. Migration with 3 steps

See EX1037, 4 (“For the IPv6 backbone, the edge router plays the role of an access gateway...”). For instance, the edge routers in the top right of Figure 11 interconnects an IPv4-only network (red) and an IPv6 network (orange). EX1037, 7. Similarly, the edge router within AS 24349 interconnects the IPv6 access network (purple) with the IPv6 network (orange). *Id.* Li teaches that the IPv6 network is that of a service provider. *Id.*, 6 (“access networks send their IPv4 traffics to each another via the pure-IPv6 backbone of their service provider.”). Li also teaches that the connected access networks may be that of a “customer of the provider.” *Id.*, 6 (“IPv4-only customer networks is connected to the IPv6 backbone... Each customer network has an AS number.”), 3, 7.

102. A POSITA would have understood that a home network or the network of a business are examples of a customer network, and that those networks may be implemented as private networks. EX1035 (U.S. Patent No. 7,624,195, filed on May 8, 2003), Fig. 1 (illustrating “edge router” connecting hosts (illustrating edge router 125 connecting home network to provider network 105), 3:42-59 (“edge routers 125a, 125b and 125c may be in the form of customer premises equipment (CPE), such as a cable modem, which may be coupled to one or more hosts 130a, 130b and 130c, such as PCs.”). Thus, Li’s edge routers are “home gateways” under the Texas court’s construction.

103. Moreover, Li’s home gateways are adapted to connect a user terminal to the IPv6 domain. A POSITA would understand that each of the IPv4-only access network AS 24348 (red) and IPv6 access network AS 24349 (purple) include various computers and other host devices (*i.e.*, user terminals). EXS1065-1066 (disclosing that AS 24348 and AS 24349 are networks for Peking University and Tsinghua University).

104. The home gateway in the top right of Figure 11 connects the user terminals that are part of the IPv4-only access network AS 24348 (red) to the IPv6 provider domain AS 23910 (orange) so that they may communicate with other hosts:

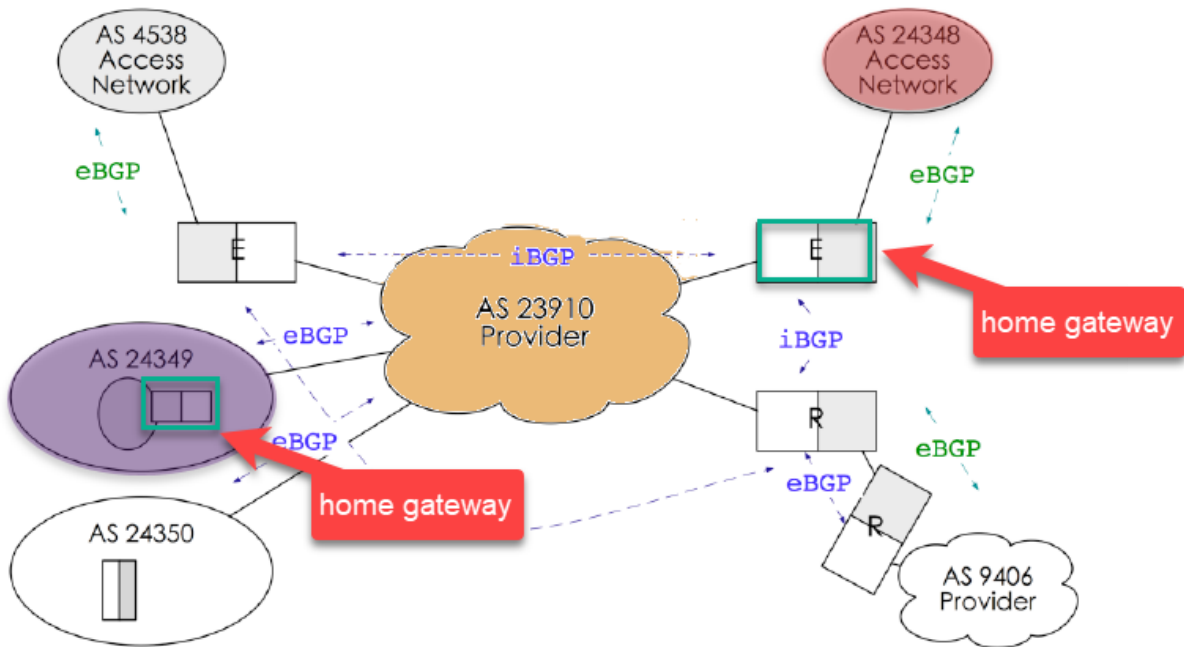


Figure 11. Migration with 3 steps

EX1037, 9 (highlighting added). Indeed, Figure 5 of Li discloses user terminals (e.g., “Host A” and “Host D”) in the IPv4 domain are connected to the IPv6 network via Li’s home gateway (e.g., “edge routers B and C”), such that they can communicate with one another. *Id.*, 5.

105. Similarly, AS 24349 (purple) is a “native IPv6 access network.” *Id.*, 9-10. Thus, the home gateway connects the user terminals to the AS 24349 access network, which is an IPv6 domain. The home gateway also further connects the user terminals to the IPv6 provider domain AS 23910 (orange) so that they may communicate with other hosts over the IPv6 domain. Thus, Li’s home gateways are

adapted to connect hosts (user terminals) to the IPv6 domain (AS 23910 and/or AS 24349).

- (b) 1[a]: identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address and a destination port number

106. Li and Li-2 render obvious limitation 1[a]: “identifying an IPv6 destination address [*e.g.*, Li’s constructed IPv6 address] constructed by concatenating an IPv6 prefix [*e.g.*, Li’s IPv6 global routing prefix], an IPv4 destination address [*e.g.*, Li’s IPv4 destination address], and a destination port number [*e.g.*, Li-2’s destination port number].”

107. When an IPv4 host (*e.g.*, a host within AS 4538) sends an IPv4 packet to a host in access network AS 24348 (red) or AS 24349 (purple), the IPv4 packet is received at the top left edge router that is shown in Figure 11 of Li and its IPv4 header is translated from IPv4 to IPv6 so that the packet may be routed through the IPv6 provider network (orange):

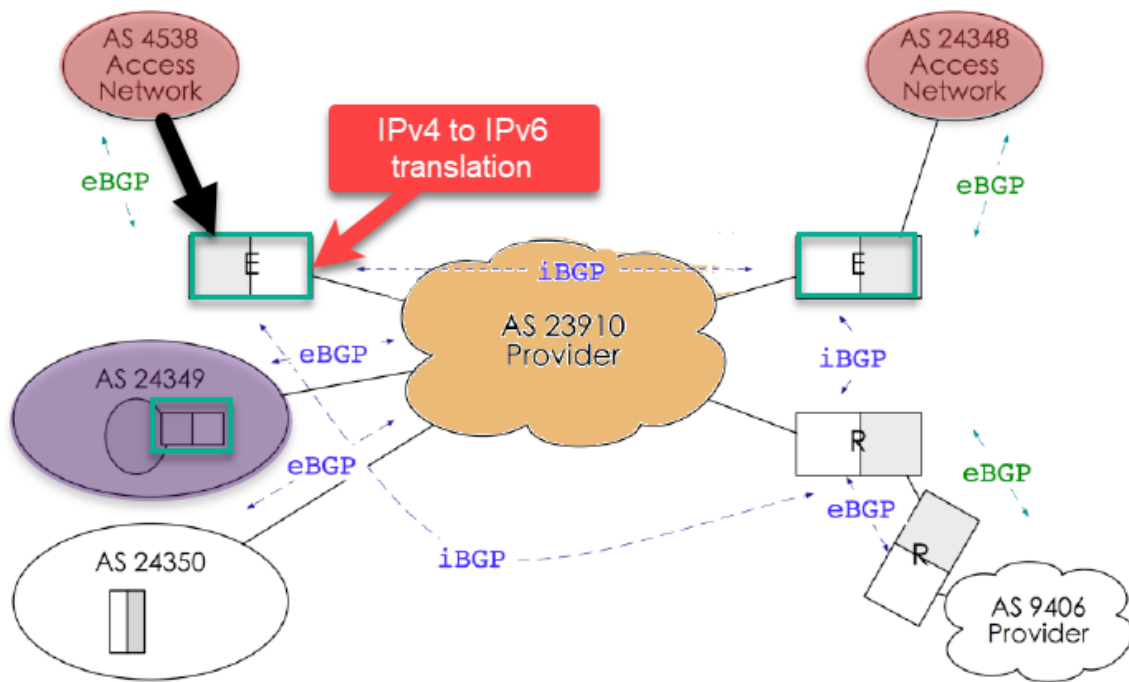


Figure 11. Migration with 3 steps

EX1037, 7, 9. This header translation is illustrated in Figure 4 of Li (reproduced below):

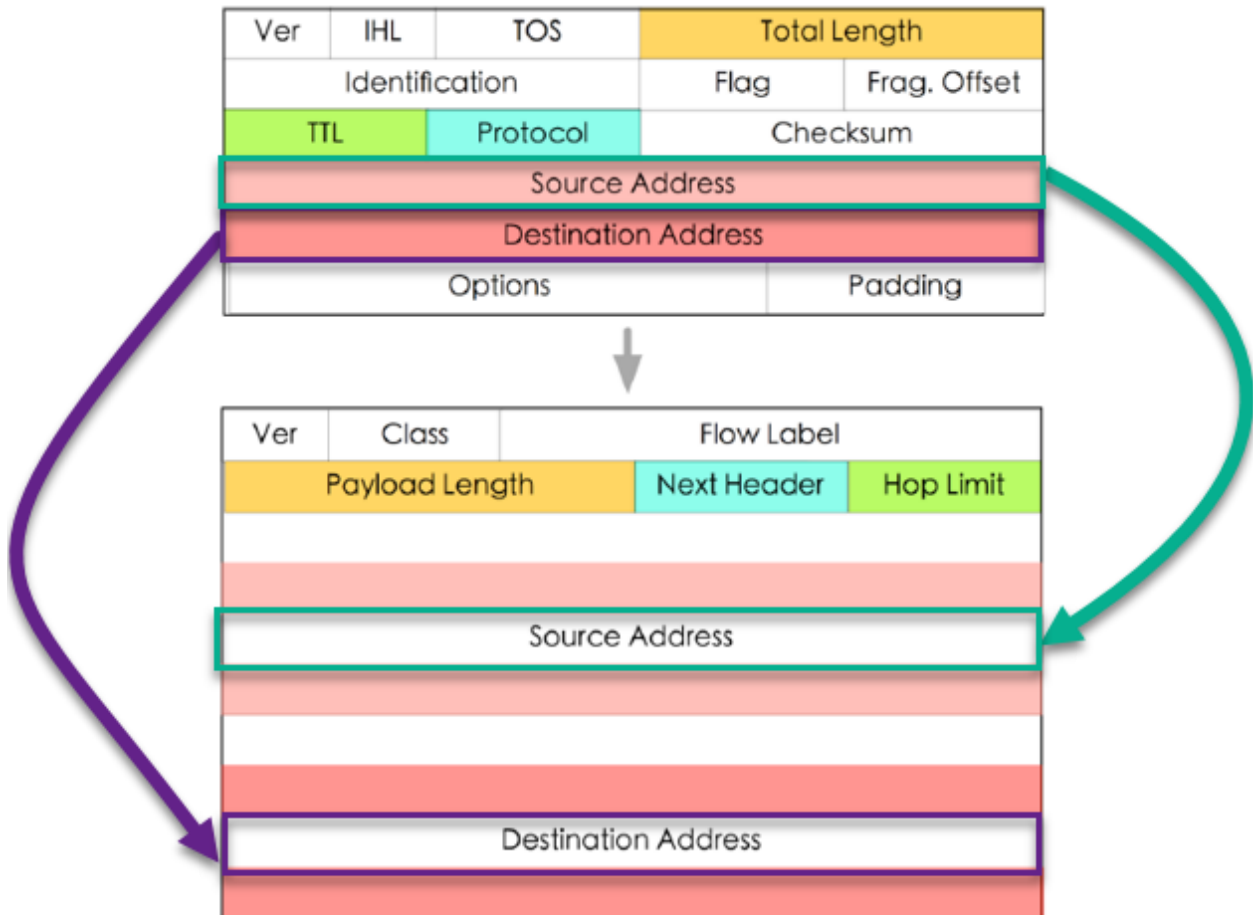


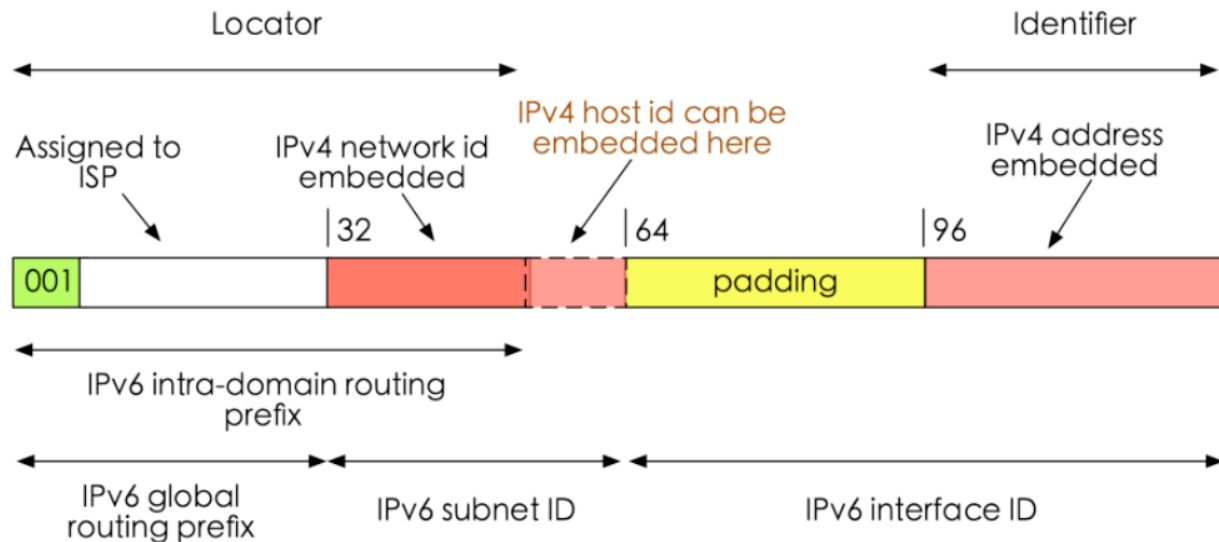
Figure 4. Mapping the header from IPv4 to IPv6

EX1037, 4 (annotated). Among other things, the IPv4 source address is replaced with an IPv6 source address (green) and the IPv4 destination address is replaced with an IPv6 destination address (purple). The IPv6 source and destination addresses are constructed using the address mapping scheme of Li and Li-2, which, as discussed in §VII.A.1-2, concatenates an IPv6 prefix, an IPv4 address and, when modified by Li-2, a port number.

108. Indeed, Li discloses an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address, and zero padding. In

particular, Li describes an “address mapping” method where “IPv4 source and destination addresses in a packet can be translated into their IPv6 version.” EX1037,

3. Figure 2 of Li disclose an IPv6 address constructed with Li’s address mapping:



EX1037, 3. The first 32 bits of the constructed IPv6 address are an IPv6 global routing prefix that is assigned to the IPv6 domain by an “ISP [Internet Service Provider].” *Id.*, 3 (“IPv6 prefixes [are] allocated by the provider”). Thus, Li’s IPv6 global routing prefix corresponds to the ’845 patent’s “IPv6 prefix.” The next 32 bits (up to bit 64)⁴ of the constructed IPv6 address is an IPv4 address. *Id.*, 3 (“IPv4 ... destination address[] in a packet can be translated into their IPv6 version...”), 5 (“IPv4 ... destination address is sometimes copied twice at the translation.”). The

⁴ Or, alternatively, the last 32 bits. EX1037, 3.

32 bits after that (up to bit 96) is “padding,” in which empty space is filled with 0s. *Id.*

109. Li’s “address mapping” involves concatenating—construed by the Eastern District of Texas court to mean “appending one item to the end of another so as to form a single unit in a contiguous pattern”—these pieces of information. EX1036, 13. This is shown in Table 1, which provides examples of Li’s address mapping:

TABLE I. SOME ADDRESS MAPPING EXAMPLES

IPv4 prefix/address	IPv6 prefix/address
202.112.0.0/13	2001:da8:ca70::/45
166.111.0.0/16	2001:da8:a66f::/48
166.111.172.1	2001:da8:a66f:ac01::a66f:ac01

EX1037, 3. The left column contains IPv4 addresses or address ranges, represented in dotted-quad notation. The right column contains translated IPv6 addresses or address ranges, represented in hexadecimal notation. In this example, the IPv6 domain is assigned an IPv6 prefix of 2001:da8. EX1037, 3. In the outbound direction, when an edge router receives a data packet, it concatenates the IPv6 prefix with the IPv4 address shown in the left column.

110. For instance, as Table 1 shows, the third row maps an IPv4 destination address of 166.111.172.1, which corresponds to a host in the IPv4 access network at the top right of Figure 11 (AS 24348), to an IPv6 destination address of 2001:da8:a66f:ac01::a66f:ac01. EX1037, 3; *see also id.*, 7 (Fig. 7 showing AS 24348 hosts have a network prefix of 166.111.0.0/16). The double-colon notation—“::”—in the IPv6 address indicates that the numbers following each colon are zeros; thus, the IPv6 address 2001:da8:a66f:ac01::a66f:ac01 is equivalent to 2001:da8:a66f:ac01:0000:0000:a6f:ac01.

111. The first two groups of this IPv6 address—2001:da8:—are an IPv6 prefix. *See* EX1037, 3. Converting the first portion of the IPv4 address—166—from decimal to hexadecimal results in the hexadecimal number “a6,” converting the second portion of the IPv4 address—111—results in hexadecimal number 6f, converting the third changes 172 to ac, and converting the fourth keeps 1 as 1. Thus, converting “166.111.172.1” from dotted-quad notation to hexadecimal would have resulted in the third and fourth groups of the IPv6 address, “a66f:ac01” (which are also repeated in the seventh and eighth groups, for reasons that are not relevant to this petition). *See* EX1037, 3. And the fifth and sixth groups, “0:0,” are the zero padding shown in Figure 2. EX1037, 3. Thus, Li teaches “identifying an IPv6 destination address constructed by concatenating an IPv6 prefix [*e.g.*, IPv6 global routing prefix], [and] an IPv4 destination address.”

112. Although Li does not expressly disclose further concatenating “a destination port number,” it would have been obvious to do so in view of Li-2. Li-2 teaches a substantially similar address mapping scheme (which it calls “IVI address mapping”) as that disclosed in Li. But Li-2 expands Li by further disclosing that a constructed IPv6 address may also include a destination port number. Specifically, Li-2 describes “port multiplexing”—an extension of Li that constructs an IPv6 address by not only concatenating an IPv6 prefix and IPv4 address but further concatenating a port number. EX1038, 14.

113. A POSITA would have recognized that there is more than sufficient space to further include a 16-bit destination port number in Li’s constructed IPv6 destination address by replacing 16 bits of the zero padding with the destination port number. Indeed, that’s exactly what Li-2 teaches: “The mapping scheme is to use the least significant bits in the IVI6(i) to define the multiple mapping and combine the tran[s]port-layer port number to uniquely perform the mapping from IVI4(i) to IVI6(i).” EX1038, 14. In fact, in a presentation to the IETF describing Li-2 the authors presented examples of Li-2’s stateless mapping algorithm. EX1028. These examples further confirm that Li-2 constructs a destination IPv6 address (orange box on the right) by concatenating (1) an IPv6 prefix with (2) an IPv4 address and (3) a port number in a received data packet (purple box on the left):

```
- 202.38.108.2#81 --> IVI61=2001:250:ffca:266c:0200::81#81
- 202.38.108.2#82 --> IVI61=2001:250:ffca:266c:0200::82#82
- 202.38.108.2#83 --> IVI61=2001:250:ffca:266c:0200::83#83
- 202.38.108.2#84 --> IVI61=2001:250:ffca:266c:0200::84#84
```

EX1028, 22 (highlighted); *see also* §VII.A.2.

114. As Li-2 also explains, the sixteen bit “ff” value between the IPv6 prefix and the IPv4 address in the constructed IPv6 address is an identifier indicating that the address is a constructed “IVI” address. EX1038, 6 (illustrating “FF” between “IPv6 prefix” and “IPv4 address” and stating that it is “the identifier of IVI.”). In other words, in the combined Li-Li-2 combination, Li’s edge router (*e.g.*, the edge router in the top left of Figure 11) would construct an IPv6 destination address by concatenating an IPv6 prefix, an IPv4 destination address, and a destination port number, as recited in this limitation. The edge router on the top left of Figure 11 would then route the data packet with the constructed IPv6 address to another router (*e.g.*, the edge router in the top right of Figure 11 or the access gateway in AS 24349) via the IPv6 provider network (AS 23910). EX1037, 6 (“Fig. 7 summarizes the scenario where access networks send their IPv4 traffics to each another via the pure-IPv6 backbone of their service provider. In the Figure, two IPv4 native ASes are attached to the provider IPv6 AS with the help of edge routers...”), 5 (Figure 5(a) showing Edge Router B sending IPv6 packet to Edge Router C).

115. For instance, in the example just discussed the edge router in the top left of Figure 11 routes the IPv6 packet to the edge router on the top right of Figure 11 that serves a host having the IPv4 address 166.111.172.1 (which is within access network AS 24348 whose network prefix is 166.111.0.0):

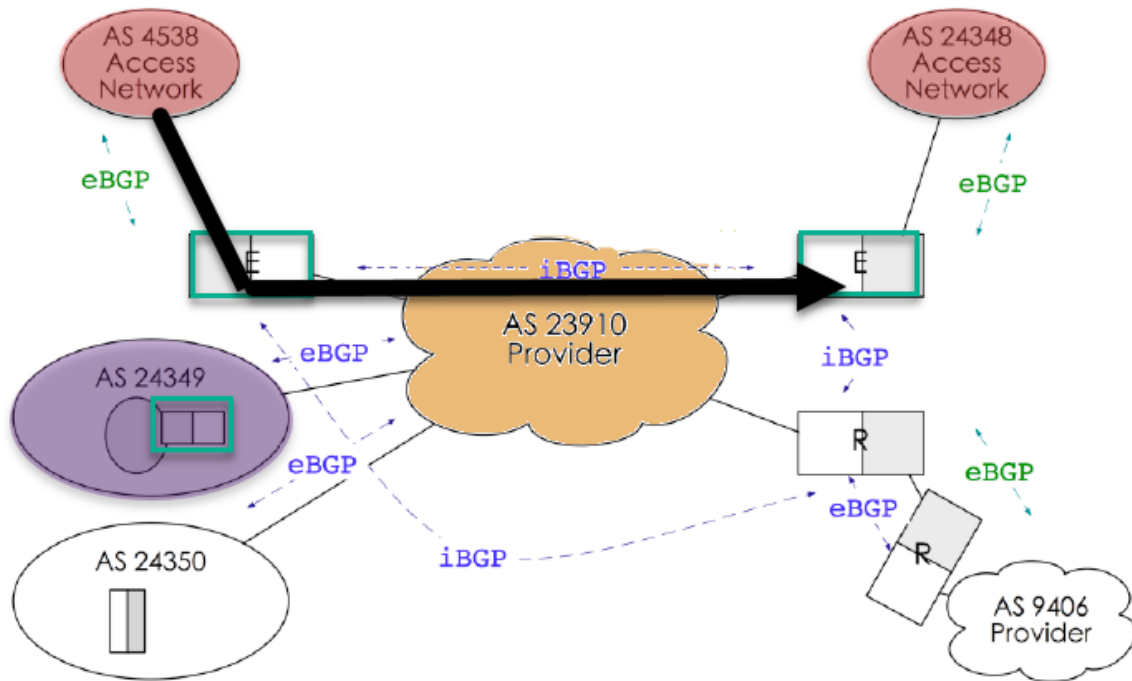


Figure 11. Migration with 3 steps

See *id.*, 7 (Fig. 7 showing AS 24348 hosts have a network prefix of 166.111.0.0/16). The edge router in the top right receives the packet and identifies that it has an IPv6 address that was constructed using the IVI scheme. EX1038, 6 (“bit 32 to bit 39 are all one’s as the identifier of IVI”; Figure 1 illustrating the IVI identifier as “FF”).

116. Alternatively, if the IPv6 packet is destined for a host within the access network AS 24349, then the edge router in the top left of Figure 11 would route the IPv6 packet to the edge router that is within AS 24349, as shown below:

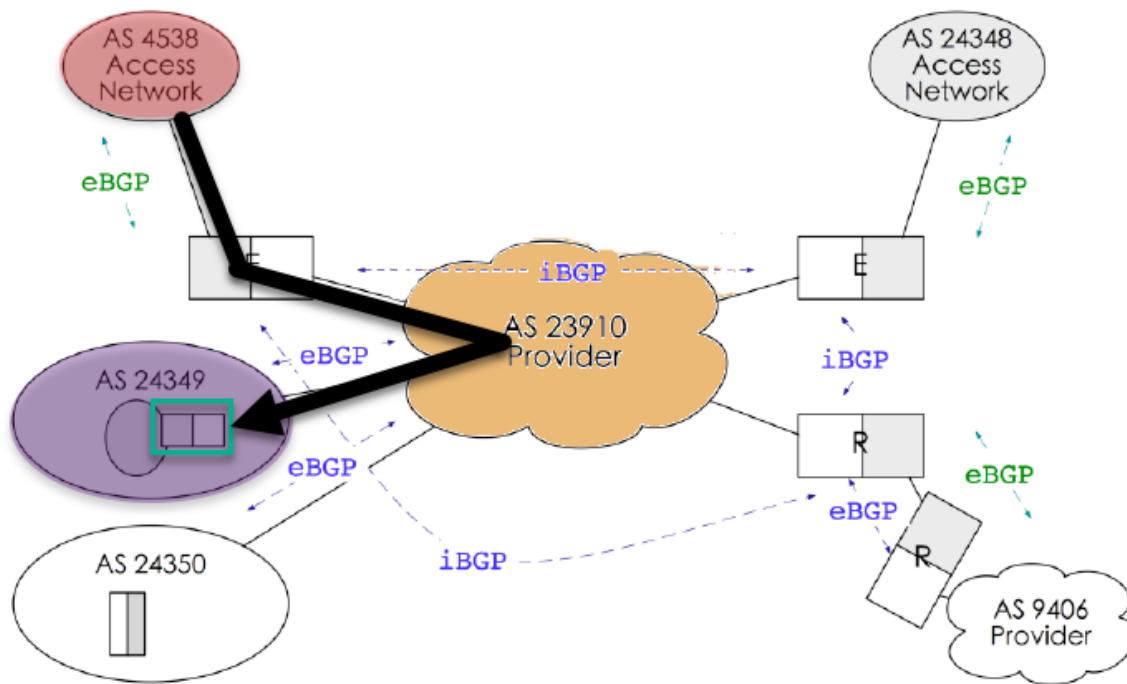


Figure 11. Migration with 3 steps

EX1037, 9 (highlighting added).

- (c) 1[b]: 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, 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 modifying the data packet using the regularized address; and

117. Li and Li-2 disclose and render obvious the limitation 1[b]: “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, 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 modifying the data packet using the regularized address.”

118. This limitation 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. The Eastern District of Texas construed “regularizing” to have its “plain and ordinary meaning.” *See* EX1036, 29-33. More specifically, I understand that in the Texas litigation Patent Owner proposed that the plain meaning of “regularizing” is “conforming [the IP address] to the destination IP domain.” *Id.*, 29. I further understand that Patent Owner stated that the “regularizing” step [1] does *not* require a NAT table and [2] includes changing IPv6 addresses into IPv4 addresses and vice versa. *Id.*, 29 (“Plaintiff further contends that there is no requirement to update a NAT.”), 30 (“Plaintiff replies that Defendant’s narrow interpretation that the claims only cover a packet moving from an IPv6 domain to another IPv6 domain is wrong....Plaintiff argues that the specification also describes the home gateway transforming packets from IPv6 into IPv4 and vice versa.”), 31 (“Plaintiff contends that NAT tables are not a necessary part of

regularizing the address(es), because the claims are not limited to IPv6-to-IPv6 connections.”). I understand that the Eastern District of Texas accepted Patent Owner’s view of the plain meaning. *Id.*, 31 (“The independent claims do not require a NAT table as part of the recited ‘regularizing.’”), 32 (“Defendant contends that the parties have a fundamental dispute regarding whether the claims of the ’845 Patent are limited to only IPv6-to-IPv6 translations, and exclude IPv4-to-IPv6 translations (and vice versa)....The specification indicates that the claims are not as limited as Defendant contends.”). Under the Texas court and Patent Owner’s construction, Li and Li-2 disclose the step of “if necessary, regularizing....”

119. For example in Figure 11 below an IPv4 host in the top-left IPv4 access network sends a data packet to an IPv4 host in the top-right IPv4 access network, as depicted by the black arrow:

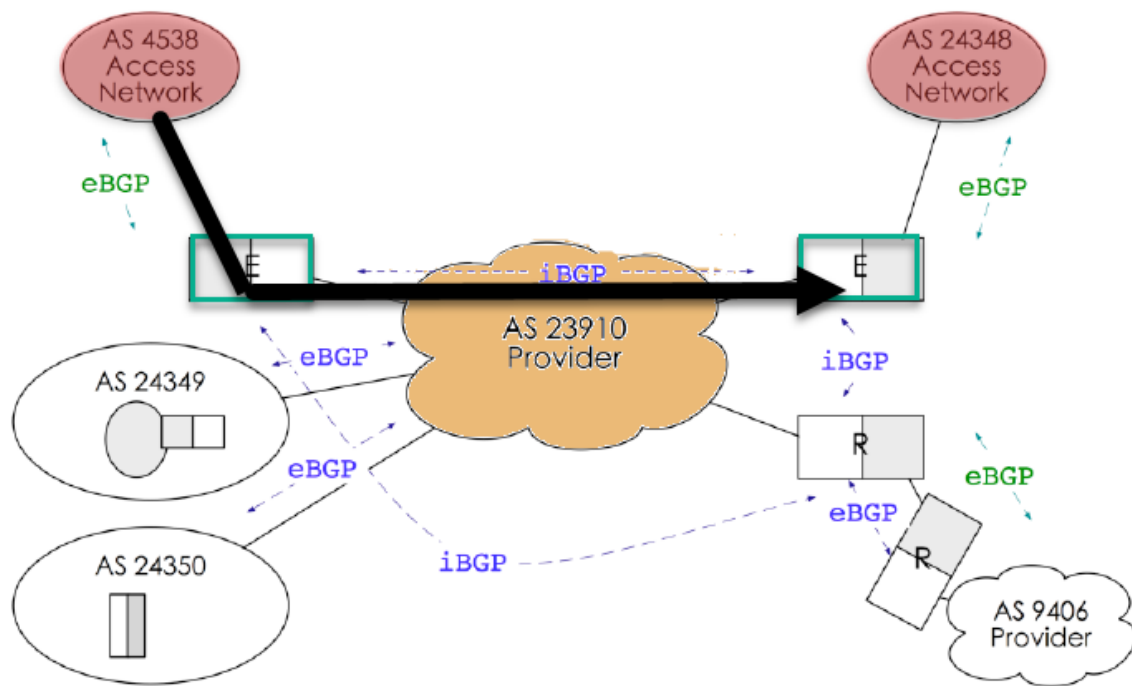


Figure 11. Migration with 3 steps

The edge router in the top right of Figure 11 receives an IPv6 packet with an IPv6 address constructed using the address translation scheme of Li and Li-2 (referred to as the “IVI address mapping”) and identifies that it has an IPv6 address that was constructed using the IVI scheme. EX1038, 6 (“bit 32 to bit 39 are all one’s as the identifier of IVI”; Figure 1 illustrating the IVI identifier as “FF”).

120. Both Li and Li-2 further disclose that the edge router on the top right conforms the IP addresses to the destination IPv4 domain (AS 24348) by replacing the constructed IPv6 addresses with IPv4 addresses. Continuing with the example above, the edge router in the top right receives the IPv6 packet and inverses the IVI address mapping in order to send the packet to the IPv4 domain. EX1037, 7

(highlighted and annotated), 4 (“An edge router manipulates the IP headers for the packets passing through it from IPv6 to IPv4 or vice versa”); *see also* EX1038, 11 (Figure 5 showing IPv6-to-IPv4 translation); EX1028, 19 (explaining that IPv6 to IPv4 translation applies the “inverse” IVI address mapping). For instance, as shown by the red arrow in the image below, the edge router inverses the IVI address mapping for the constructed IPv6 destination address of 2001:da8:a66f:ac01::a66f:ac01 (discussed in connection with Limitation 1[a]) by replacing it with the native destination IPv4 address of 166.111.172.1:

TABLE I. SOME ADDRESS MAPPING EXAMPLES

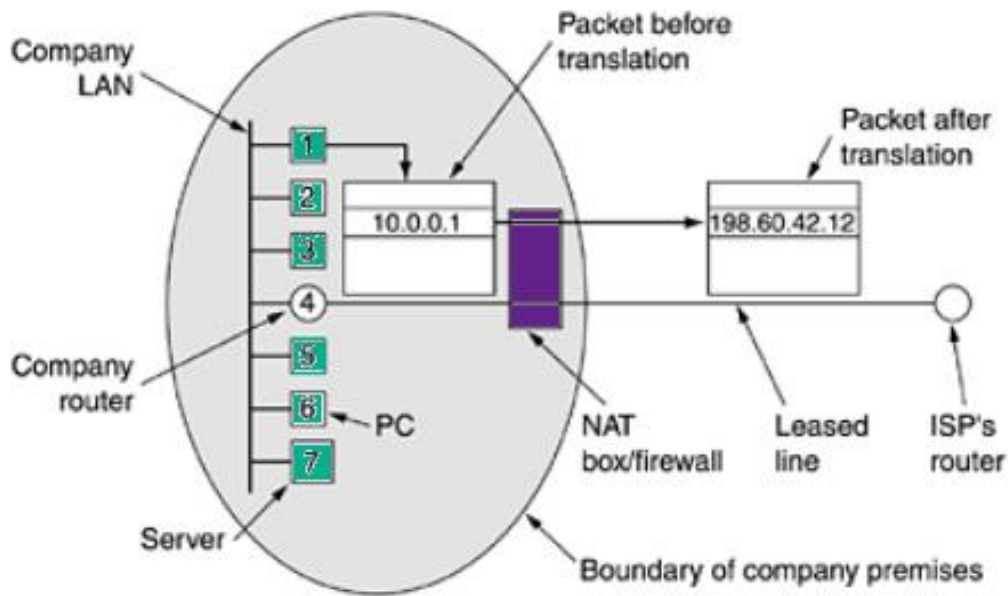
IPv4 prefix/address	IPv6 prefix/address
202.112.0.0/13	2001:da8:ca70::/45
166.111.0.0/16	2001:da8:a66f::/48
166.111.172.1	2001:da8:a66f:ac01::a66f:ac01

EX1037, 3 (red arrow added); *see also* EX1037, 5 (Figure 5(a) showing Edge Router C conforming IPv6 to IPv4 before sending packet to Host D). The edge router then modifies the data packet by using the regularized address (*e.g.*, 166.111.172.1) as the IPv4 destination address in the IPv4 header. *See* EX1037, 4; EX1038, 11 (Figure 5); EX1028, 19; The modified packet would include the destination address of

166.111.172.1, which corresponds to a host in the top-right IPv4 access network, and can be routed to that host.

121. As mentioned above, under the construction proposed by Patent Owner and adopted by the Texas Court, regularizing does *not* require a NAT table. However, Li-2 also teaches that NAT can be integrated into its system as part of the regularizing step. EX1038, 18 (“IPv4 NAT can easily be integrated into the system.”). By the date of the ’845 patent, POSITAs were well-aware of NAT and the benefits it provides for a system like Li. See EX1001, 1:56-60 (“This table is known to the person skilled in the art as a NAT (network address translation) table.”).

122. IPv4 NAT translates public IPv4 addresses to private IPv4 addresses. See §III.D. The image below shows a NAT (purple) within a customer network having multiple user terminals (green):



EX1059, 340. “Within the company premises, every machine has a unique address of the form 10.x.y.z” (its private IPv4 address), which is converted to a public IPv4 address (198.60.42.12) by a “NAT” when communicating outside of the internal network. *Id.*; see also *id.*, 341 (“On the way back, packets undergo the reverse mapping.”). While the image above illustrates the NAT as its own device, POSITAs understood that “[i]t is also possible to integrate the NAT box into the company’s router.” EX1059, 340-341.

123. Thus, in my opinion, it would have been obvious to implement NAT within the system of Li and Li-2. For instance, I explained above that if the IPv6 packet is destined for a host within the access network AS 24349, then the edge router in the top left of Figure 11 would route the IPv6 packet to the gateway that is within AS 24349:

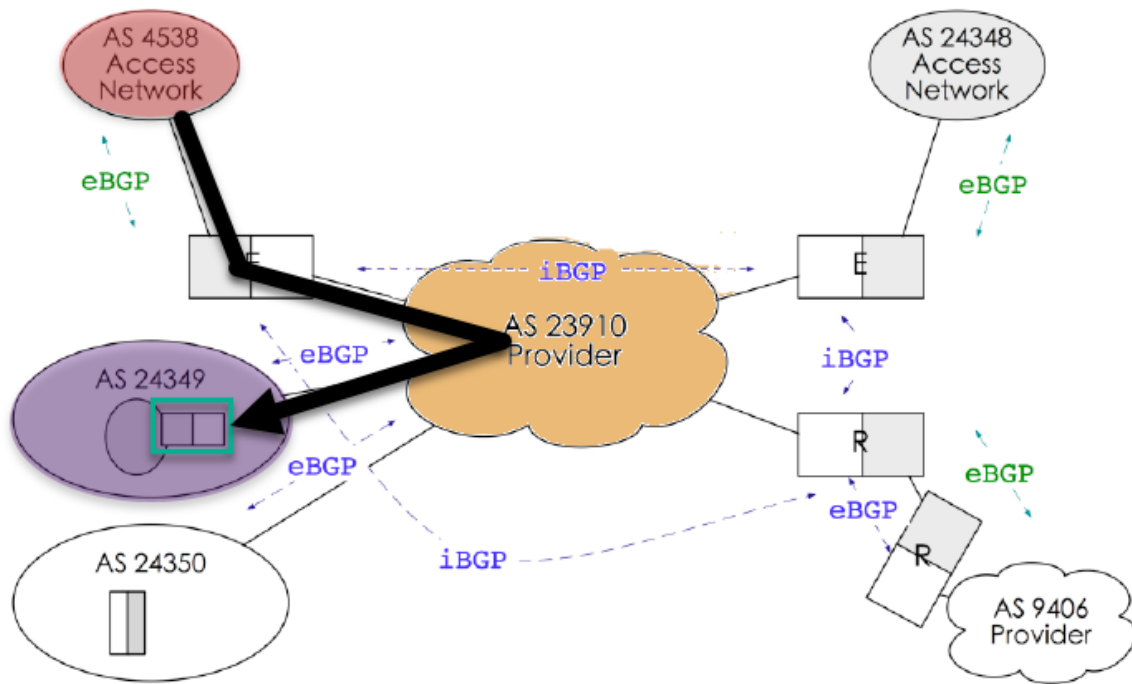


Figure 11. Migration with 3 steps

EX1037, 9 (highlighting added). Li describes AS 24349 as an “IPv6 access network containing a small group of IPv4 subnets”. EX1037, 9-10.

124. Thus, a POSITA would have understood that when communicating with the IPv4 subnets the edge router in AS 24349 (green box) would inverse the IVI mapping scheme to return the public destination IPv4 address embedded in the constructed destination IPv6 address—in the same manner that was just discussed in relation to the IPv4 only domain in the top right (AS 24348). After Li’s router receives and translates the constructed IPv6 address to a native public IPv4 address by reversing the IVI address mapping, “IPv4 NAT can easily be integrated into the system” to further translate the native IPv4 address to a private IPv4 address. See

§III.D. The data packet would then be routed with the private IPv4 address to its destination, namely the IPv4 host that is allocated that private IPv4 address.

(d) 1[c]: routing the modified data packet to its destination.

125. Li further discloses “routing the modified data packet to its destination.” As discussed in the prior limitation, Li and Li-2 disclose regularizing and modifying a data packet. A POSITA would understand Li to teach that the modified data packet is routed to the address in the destination IP address field. Indeed, Figure 5(a) of Li illustrates an example in which the edge router C routes a packet to its destination IP address, namely host D. EX1037, 5.

6. Claim 5

(a) 5[pre]: 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:

126. To the extent the preamble is limiting, Li and Li-2 render obvious limitation 5[pre]: “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”.

127. With reference to Figure 11 below, Li discloses that the router within the native IPv6 access network (AS 24349) receives an IPv6 packet, where the IPv6 packet was originally transmitted as an IPv4 packet from the IPv4 domain (AS 4538):

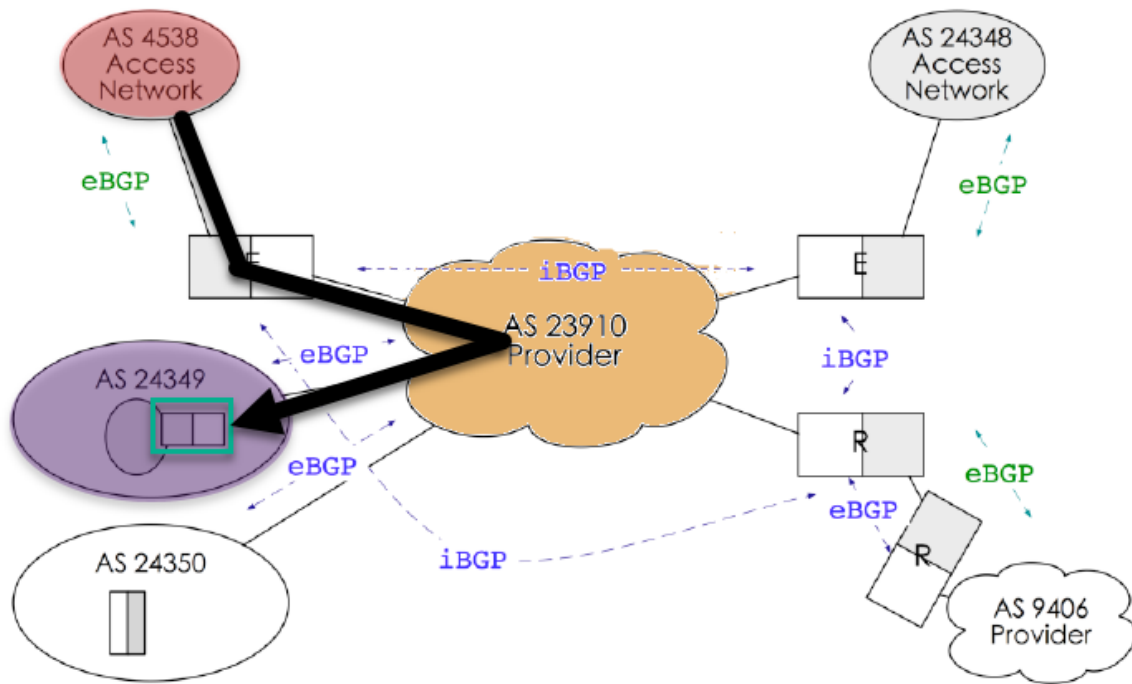


Figure 11. Migration with 3 steps

EX1037, 9 (highlighting added). When the router in the AS 24349 network receives the IPv6 packet, it translates the constructed IPv6 address to a native public IPv4 address by reversing the IVI address mapping and may also integrate IPv4 NAT to further translate the native IPv4 address to a private IPv4 address. See §VII.A.5.c, III.D.. The IPv4 packet is then routed to a host within the AS 24349 that is associated with the private IPv4 address (“user terminal”).

- (b) 5[a]: searching an address translation table of the gateway for an entry associating with said constructed destination address a native destination address;

128. Li and Li-2 disclose and render obvious “searching an address translation table of the gateway for an entry associating with said constructed destination address a native destination address.”

129. Li-2 teaches that after the router in AS 24349 translates the constructed IPv6 address to a native public IPv4 address by reversing the IVI address mapping, “IPv4 NAT can easily be integrated into the system” to further translate the native IPv4 address to a private IPv4 address. EX1038, 18. A POSITA would have understood that IPv4 NAT utilizes an address translation table that would include an association between private IPv4 destination addresses (“native destination addresses”) and public IPv4 addresses contained in the constructed IPv6 destination addresses (“said constructed destination address”), allowing Li’s router to perform the public-to-private IPv4 translation. EX1059, 340 (“When a packet arrives at the NAT box from the ISP, the source port in the TCP header is extracted and used as an index into the NAT box’s mapping table. From the entry located, the internal IP address and original TCP Source port are extracted and inserted into the packet. Then both the IP and TCP checksums are recomputed and inserted into the packet. The packet is then passed to the company router for normal delivery using the 10.x.y.z address.”); *see also* EX1001, 1:56-60 (“This table is known to the person skilled in the art as a NAT (network address translation) table.”).

- (c) 5[b]: if an entry is found, replacing the constructed destination address of the data packet with the native destination address from that entry.

130. Li and Li-2 disclose and render obvious “if an entry is found, replacing the constructed destination address of the data packet with the native destination address from that entry.” As described in connection with Limitation 5[a], Li-2 teaches that after the router in AS 24349 translates the constructed IPv6 address to a native public IPv4 address by reversing the IVI address mapping, “IPv4 NAT can easily be integrated into the system” to translate the public IPv4 destination address embedded in the constructed address to a private IPv4 destination address (the “native destination address”). EX1038, 18; EX1059, 340-341. A POSITA would understand that NAT replaces the public IPv4 address with a private IPv4 address only if an entry is located in its NAT table. EX1038, 18. A POSITA would have understood that IPv4 NAT utilizes an address translation table that would include an association between private IPv4 destination addresses (“native destination addresses”) and public IPv4 addresses contained in the constructed IPv6 destination addresses (“said constructed destination address”), allowing Li’s router to perform the public-to-private IPv4 translation. EX1059, 340 (“When a packet arrives at the NAT box from the ISP, the source port in the TCP header is extracted and used as an index into the NAT box’s mapping table. From the entry located, the internal IP address and original TCP Source port are extracted and inserted into the packet. Then

both the IP and TCP checksums are recomputed and inserted into the packet. The packet is then passed to the company router for normal delivery using the 10.x.y.z address.”); *see also* EX1001, 1:56-60 (“This table is known to the person skilled in the art as a NAT (network address translation) table.”); §III.D.

7. Claim 6: 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.

131. The combination of Li and Li-2 disclose and render obvious “[t]he 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.”

132. For example, Li-2 discloses that the destination port number of the data packet and the destination port number contained in the constructed destination address match. EX1038, 14. In particular, Li discloses that its IVI address mapping uses “the transport-layer port number [i.e., destination port number of the data packet] to uniquely perform the mapping from IVI4(i) to IVI6(i).” *Id.* For instance, the image below illustrates examples of destination IPv6 addresses constructed using Li-2’s IVI address mapping, and confirms that the destination port number in the constructed address is to match the destination port number in the data packet:

IVI61=2001:250:ffca:266c:0200::81#81
IVI61=2001:250:ffca:266c:0200::82#82
IVI61=2001:250:ffca:266c:0200::83#83
IVI61=2001:250:ffca:266c:0200::84#84

EX1028, 22 (highlighted); *see also* §VII.A.2.

133. A POSITA would have also found it obvious to verify that the port numbers match. The port number that is part of the address was obtained from the port number in the transport header. Since the address specifies that it has this particular format, testing this equality to ensure that the address was formed correctly, and that the headers were not tampered with or modified by error en route would have been a routine check that was obvious to implement. Indeed, by the time of the alleged invention of the '845 patent, POSITAs understood that such a comparison would be useful to identify errors and security issues. For instance, Li-2 references RFC 4380. EX1038, 38. RFC 4380 explains that when receiving a constructed address (a Teredo address) a comparison of the port number in the transport layer header and the port number in the constructed address is useful in detecting “spoofed” packets. EX1032 at 28-29 (“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.”). In fact, the '845 patent also states that its system verifies port numbers to protect against

spoofed packets. *See* EX1001, 13:60-67 (“The IPv6 access router 30 verifies that the IPv6 source address IPv6_address_source_ports_range is authorized for the home gateway 20. In other words, it verifies that there is no spoofing, i.e. that the home gateway 20 sends IPv6 packets in its authorized prefix range IPv6_prefix_ports_range. For this the IPv6 access router 30 verifies that the IPv6 source address IPv6_address_source_ports_range of the IPv6 packet received from the gateway 20 is inscribed in the prefix IPv6_prefix_ports_range assigned to this home gateway. This verification is effected in a standard manner in accordance with a principle already used for IPv6 native traffic.”).

8. Claim 7: 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.

134. Li discloses the additional limitation of claim 7. As described above, Li and Li-2 disclose that their method is implemented as part of a gateway (or edge router). EX1037, 3, 7; *see also* EX1038, 17 (“IVI gateway based on the Linux implementation has been deployed”); *see also* EX1037, 5 (describing “in-lab experiments” performed on “5 computers” that were “connected by one over 100Mbps direct Ethernet links.”). A POSITA would have known that a gateway comprises memory (a compute readable medium) and a microprocessor, and that it

stores in memory program code instructions that are executed to perform its operations.

9. Claim 8

- (a) 8[pre]: A device for receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain, said packet comprising an IPv6 destination address and an IPv6 source address, said device for use in a home gateway adapted to connect a user terminal to said IPv6 domain, comprising:

135. To the extent the preamble is limiting, Li teaches “a device for receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain, said packet comprising an IPv6 destination address and an IPv6 source address, said device for use in a home gateway adapted to connect a user terminal to said IPv6 domain.” See §VII.A.5.a.

136. For instance, Li’s edge routers, such as the edge router connecting AS 24348 or the edge router within AS 24349, are a “home gateway” that satisfies limitation 8[pre] for the reasons discussed in connection with claim 1:

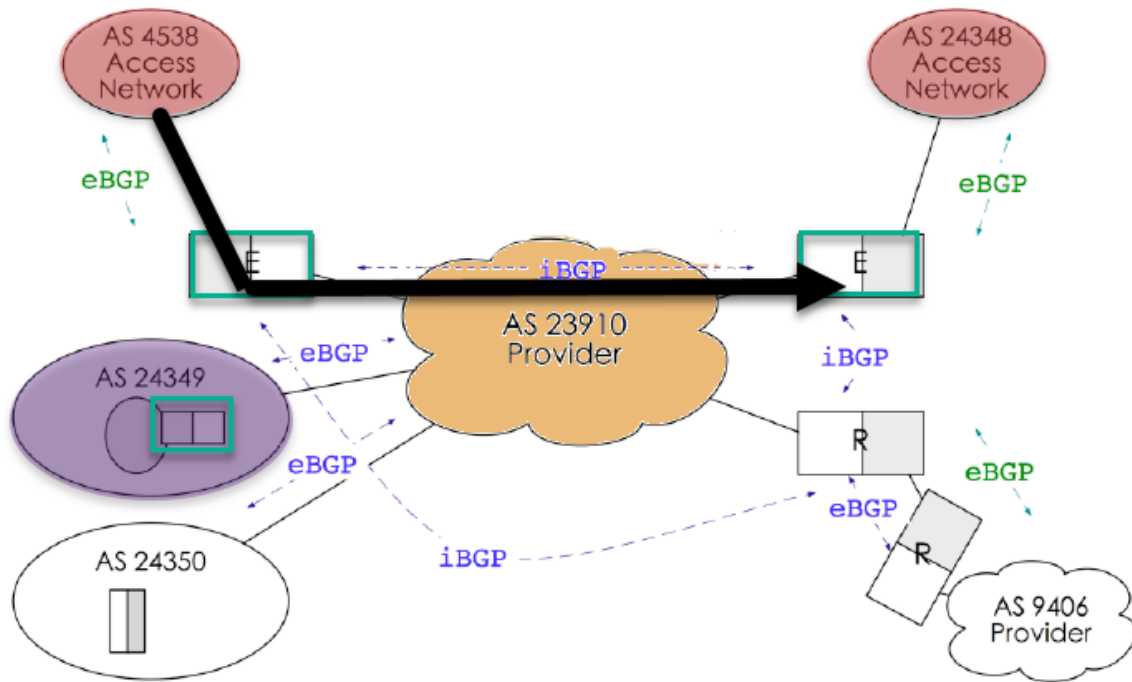


Figure 11. Migration with 3 steps

EX1037, 9 (highlighting added).

- (b) 8[a]: means for identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address, and a destination port number;

137. I have been informed the Eastern District of Texas previously identified limitation 8[a] as means plus function claiming. The court construed this limitation as having the function of “identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address, and a destination port number” with an accompany structure of “the home gateway 20 that includes a processor programmed to perform the step of identifying an IPv6 destination address

constructed by concatenating an IPv6 prefix, an IPv4 destination address, and a destination port number.” EX1036, 83.

138. As discussed in connection with limitation 1[a], Li and Li-2 render obvious a home gateway that includes a processor programmed to perform the step of identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address, and a destination port number.” See §VII.A.5.b. For instance, Li and Li-2 render obvious a home gateway (the edge routers and access gateways in Figure 11) that identifies an IPv6 destination address constructed according to Li’s address mapping, which concatenates an IPv6 prefix, an IPv4 destination address, and a destination port number.” See §VII.A.5.b. Li further discloses that its “gateway” is implemented on computers and using the Linux operating system. EX1037, 3, 7; *see also* EX1038, 17 (“IVI gateway based on the Linux implementation has been deployed”); *see also* EX1037, 5 (describing “in-lab experiments” performed on “5 computers” that were “connected by one over 100Mbps direct Ethernet links.”). A POSITA would, therefore, understand that the gateway includes a processor programmed with computer instruction for identifying the constructed IPv6 destination address.

- (c) 8[b]: 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, 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 modifying the data packet using the regularized address; and

139. I have been informed the Eastern District of Texas previously identified limitation 8[b] as means plus function claiming. The court construed this limitation as having the function of “regularizing at least one of the IPv6 source address and the IPv6 destination address of the data packet and modifying the data packet using the regularized address” with an accompanying structure of “the home gateway 20 that includes a processor programmed to perform the step of regularizing at least one IPv6 address when: (1) if the IPv6 source address is a native address, then the program executes the steps of choosing an IPv6 constructed source address whose source port number is in the authorized range assigned to the home gateway 20; and modifying the data packet by replacing the native source address with the constructed source address as illustrated in steps R21 and R22 of Figure 5; or (2) if the IPv6 destination address is a constructed address, then the program executes the steps of searching a translation table of the home gateway 20 and locates a native destination address associated with the constructed destination address; verifying that the transport level destination port of the data packet agrees with the destination port number contained in the destination address of the packet; and modifying the data packet by replacing the constructed destination address with the native destination address as illustrated in steps R’21, R’22, and R’23 of Figure 8.” EX1036.

140. As discussed in connection with limitation 1[b], Li and Li-2 render obvious a home gateway that includes means for regularizing under the Texas court's construction. *See* §VII.A.5.c. For instance, Li's home gateway identifies IPv6 packets with an IPv6 destination address constructed using the IVI address mapping. EX1038, 6 ("bit 32 to bit 39 are all one's as the identifier of IVI"; Figure 1 illustrating the IVI identifier as "FF"). The home gateway then regularizes the constructed address by inverting the IVI address mapping to obtain the embedded IPv4 address, and may further implement NAT to search a translation table for a native destination address associated with the constructed destination address (a private IPv4 address). *See* §VII.A.5.c. To protect against spoofing, a POSITA would have also found it obvious to verify that the transport level destination port of the data packet agrees with the destination port number contained in the destination address of the packet (*See* §VII.A.7), and to modify the packet by replacing the constructed destination address with the native IPv4 destination address if there is a match (*See* §VII.A.5.(d)).

(d) 8[c]: means for routing the modified data packet to its destination.

141. I have been informed the Eastern District of Texas previously identified limitation 8[b] as means plus function claiming. The court construed this limitation as having the function of "routing the IPv6 data packet to its destination" with an accompanying structure of "the home gateway 20 that includes a processor

programmed to perform the step of routing the IPv6 data packet to its destination.”

EX1036. As discussed in connection with limitation 1[c], Li and Li-2 render obvious a home gateway that includes means for routing under the Texas court’s construction. See §VII.A.5.d.

B. Ground 2: Claims 1, 5-8 Would Have Been Obvious in View of Li

142. The Challenged Claims are also obvious based on Li alone in view of the common knowledge in the art.

1. Evidence of Simultaneous Invention Demonstrates the Obviousness of Combining Li and Li-2

143. Evidence of simultaneous invention indicates that a POSITA would have been motivated to modify Li to form an IPv6 address that includes the destination port number.

144. Li-2 is at minimum evidence of simultaneous invention. As discussed in Ground 1, Li-2 was publicly available as of June 21, 2008 and discloses a port multiplexing technique which is used to construct an IPv6 address containing an IPv6 prefix, a destination IPv4 address, and a destination port number.

145. Similarly, on July 6, 2008, one week after the ’845 patent’s priority date, the authors of Li and Li-2 submitted another IETF publication to the Network Working Group titled “Prefix-Specific and Stateless Address Mapping (IVI) for IPv4/IPv6 Coexistence and Transition” (“IVI Memo”). EX1027; EX1005, ¶¶49-52,

57; EX1007, ¶10. The IVI Memo also discloses using “port multiplexing” to construct an IPv6 address with a destination port number. EX1027, 7-8, 18-20.

146. And then less than one month after that, on July 29, 2008, the authors of Li and Li-2 provided examples of this stateless mapping algorithm in a presentation that they gave in Dublin for an IETF conference. EX1028; EX1007, ¶11 (confirming public accessibility); EX1005, ¶¶53-57 (same). These examples further confirm that Li-2 teaches concatenating the destination port number to the IPv6 address by replacing unused zero padding. For example, the authors provided the following examples:

```
202.38.108.2#81 --> IVI61=2001:250:ffca:266c:0200::81#81
202.38.108.2#82 --> IVI61=2001:250:ffca:266c:0200::82#82
202.38.108.2#83 --> IVI61=2001:250:ffca:266c:0200::83#83
202.38.108.2#84 --> IVI61=2001:250:ffca:266c:0200::84#84
```

EX1028, 22 (highlighting added). The IPv6 prefix is the first box, highlighted in blue. EX1038, 6 (“[B]it 0 to bit 31 are the prefix of ISP(i)’s /32.”). The second box, highlighted in red, is a flag that specifies that the address has been constructed according to IVI. *Id.*, (“[B]it 32 to bit 39 are all one’s as the identifier of IVI.”). The third box, which highlights the next 32 bits in purple, is the IPv4 address from the left side of the translation, converted from decimal to hexadecimal. *Id.* (“[B]it 40 to bit 71 are the global IPv4 space (IPG4) presented in hexadecimal format.”). The third box, highlighted in yellow, is zero padding. *Id.* (Fig. 1). And the fifth box,

highlighted in gray, is the port number that has been taken from the left side of the translation and concatenated to the end of the constructed IPv6 address, replacing unused zero padding as appropriate. *Id.*, 14.

147. I note that the authors did not translate, perhaps for the sake of convenience, the port number from decimal (*i.e.*, “81”) to hexadecimal (*i.e.*, “51”). However, a POSITA would have understood that the port number should be converted to hexadecimal before concatenating it to form the IPv6 address.

148. Accordingly, the later disclosures by the authors of Li and Li-2 further confirm that it was obvious to construct a destination IPv6 address by concatenating (1) an IPv6 prefix, (2) a destination IPv4 address, and (3) a destination port number, as described in the ’845 patent and in the proposed combination.

149. Other groups also disclosed the idea of forming an IPv6 address using an IPv6 prefix, destination IPv4 address, and destination port number. *See* §III.C.3. For instance, in September 2008, Randy Bush *et al.*, published a paper describing an approach called “Address + Port.” EX1029. Bush explained that an IPv4 address could be combined with a port number to form a “A+P pseudo address within an IPv6 address using a well-known IPv6 prefix.” *Id.*, 1-3. In other words, Bush similarly proposed forming an IPv6 address by concatenating an IPv6 prefix, IPv4 address, and port number. In fact, even at the time, Bush’s disclosures were generally known; Bush identified and discussed several related works in the same

field, explaining that “[t]here is a long history of treating port numbers as part of the network address.” *Id.*, 4. Similarly, an October 2008 presentation from Mr. Bush acknowledges that A+P is “like” other, past techniques for extending the address space. EX1041, 16-17; *see also* EX1041, 21 (explaining that IPv6 “dest[ination]” address is formed by combining a well-known prefix (“WKP”) with a “dest[ination]” IPv4 address, wherein the destination IPv4 address is an Address + Port).

- BR receives IPv4 packet w/ src/dest
- Encapsulates in IPv6 packet
 - src = WKP+src
 - dest = WKP+dest
- But note that dest is A+P

EX1041, 21 (highlighted).

2. Motivations to Modify Li

150. A POSITA would have been motivated to modify the teachings of Li and would have had a reasonable expectation of success of doing so, for much of the same reasons described in ground 1. *See* §VII.A.4.

151. **First**, teachings in the art, as well as common knowledge in the art, would have encouraged a POSITA to further concatenate a destination port number

to construct an IPv6 destination address in order to prevent IPv4 address exhaustion. By the time of the '845 patent, it was well-understood that IPv4 addresses would soon run out. EX1001, 1:61-62 ("It is commonly accepted in the IP service provider community that IPv4 public addresses are going to run out."); EX1013, 1, 8-9 ("The Internet engineering community has long recognized the impending exhaustion of IPv4."); EX1014, ¶1 ("The unique IPv4 addresses are becoming scarce as the number of devices connecting to the Internet continues to increase."). POSITAs would have therefore been motivated to improve Li's system by applying known techniques for extending the IPv4 address space, thereby slowing IPv4 address depletion. Specifically, a POSITA would have recognized that port numbers could be used to uniquely identify hosts and allow them to share a single IPv4 address. See §III.C.2-3 (discussing EXS1021-1032, which disclose using port numbers to extend the IPv4 address space and to construct an IPv6 address). Accordingly, a POSITA would have been motivated to modify Li by replacing 16 bits of the unused padding in the constructed IPv6 destination address with a destination port number.

152. A POSITA would have had a reasonable expectation of success in modifying Li and there would have been no unexpected results because the modification merely involves the straightforward application of replacing unused padding in Li's constructed IPv6 address with the destination port number. In fact, the simultaneous invention discussed above further confirms that a POSITA would

have had a reasonable expectation of success of modifying Li. See §§VII.B.1; VII.A.4.

153. **Second**, for the same reasons articulated above, the modification would have been of a known technique (*i.e.*, using port numbers to share an IPv4 address and prevent IPv4 address exhaustion) to a known device (*i.e.*, Li's edge router that already constructs IPv6 addresses by concatenating information, including 32 bits of unused padding) to yield a predictable result (*i.e.*, an edge router that can construct an IPv6 address that includes a destination port number in place of 16 bits of the unused padding). See §VII.A.4.

154. **Third**, this modification would have been obvious because it is a simple substitution of one known element for another to obtain predictable results. Modifying Li would involve merely substituting 16 bits of unused padding in Li's constructed IPv6 address with a 16-bit destination port number to yield a predictable result—a constructed IPv6 address that contains a destination port number. As shown, using port numbers to extend the IPv4 address space, and even to construct an IPv6 address, was known in the art. See §III.C.2-3. And based on this state of the art, a POSITA could have substituted the unused padding for the destination port number, and the results of the substitution would have been predictable.

155. Additionally, Li and the '845 patent are concerned with the interoperability of IPv4 networks and IPv6 networks. The '845 patent explains that

“[t]he invention improves upon the [previously discussed] situation by means of a method of receiving a data packet coming from an IPv4 domain in an IPv6 domain” and is specifically directed to “a period of transition during which IPv6 domains will have to interconnect with IPv4 domains.” EX1001, 2:46-64. Li similarly explains that “at this stage, interoperability of applications in a dual-protocol environment has, replacing connectivity, become the sharpest issue of the IPv4-IPv6 transition.” EX1037, 1.

3. Claim 1

- (a) 1[pre]: A method of receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain, said packet comprising an IPv6 destination address and an IPv6 source address, said method being executed in a home gateway adapted to connect a user terminal to said IPv6 domain, and comprising the following steps:

156. As discussed under Ground 1, Li teaches all the elements of limitation 1[pre]. See §VII.A.5.a.

- (b) 1[a]: identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address and a destination port number

157. Li teaches limitation 1[a] except for concatenating a destination port number. See §VII.A.5.b. A POSITA would have found it obvious to modify Li to replace 16 bits of its unused padding with a destination port number in order to share a single IPv4 address among multiple hosts, thereby slowing IPv4 address exhaustion. See §VII.B.2. Evidence of simultaneous invention shows that not only

would a POSITA have been motivated to make this modification and would have been successful in doing so, but POSITAs at that time were actively modifying Li in such a manner. *See* §VII.B.1.

- (c) 1[b]: 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, 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 modifying the data packet using the regularized address; and

158. Li renders obvious the regularizing step of limitation 1[b]. *See* §VII.A.5.c. Li discloses that its edge routers conform an IP addresses to a destination IPv4 domain by replacing the constructed IPv6 addresses with an IPv4 addresses. *Id.* In view of the general knowledge of a POSITA, it would have been obvious to further regularize the address by using IPv4 NAT, such that the IPv4 address is further translated from a public IPv4 address to a private address IPv4 address. *Id.* A POSITA would have been well-aware of NAT at the time and its operation. *See* §III.D. A POSITA would have also recognized that NAT would have improved a system like Li by, for instance, helping to slow IPv4 exhaustion and allowing Li's access networks to hide their networks internal architecture and addresses. *Id.*; *see also* EX1058, 7 ("NAT features such as address hiding are valuable."); *see also* EX1060 (RFC 1919) at 14 ("Address hiding is generally viewed as a Good Thing, since one of the purposes of deploying proxies is to disclose as little information

about an internetwork as possible.”). Evidence of simultaneous invention shows that not only would a POSITA have been motivated to make this modification and would have been successful in doing so, but POSITAs at the time of invention were actively modifying Li in such a manner. *See, e.g.*, EX1038 (Li-2), 18 (“IPv4 NAT can easily be integrated into the system.”).

(d) 1[c]: routing the modified data packet to its destination.

159. Li renders obvious limitation 1[c]. *See* §VII.A.5.d.

4. Claim 5

(a) 5[pre]: 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:

160. Li renders obvious the additional limitations of claim 5. *See* §VII.A.6.a.

(b) 5[a]: searching an address translation table of the gateway for an entry associating with said constructed destination address a native destination address;

161. Li renders obvious the additional limitation of claim 5[a]. *See* §VII.A.6.b.

(c) 5[b]: if an entry is found, replacing the constructed destination address of the data packet with the native destination address from that entry.

162. Li renders obvious the additional limitation of claim 5[b]. *See* §VII.A.6.c.

5. Claim 6: A non-transitory computer program product that is at least one of downloadable from a communications

network and/or stored on a computer-readable medium and/or executable by a microprocessor, comprising program code instructions for executing the method according to claim 3 of sending a data packet from an IPv6 domain to a destination address of an IPv4 domain.

163. Li renders obvious the additional limitation of claim 6. *See* §VII.A.7.

6. **Claim 7: IPv6 domain access equipment according to claim 6, wherein the IPv4 is a public network, said equipment comprising first means for advertising to the IPv4 domain IPv4 addresses of network equipments connected to the IPv6 domain and second means for advertising in the IPv6 domain IPv6 addresses constructed from IPv4 addresses of the IPv4 domain**

164. Li renders obvious the additional limitation of claim 6. *See* §VII.A.8.

7. **Claim 8**

- (a) 8[pre]: A device for receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain, said packet comprising an IPv6 destination address and an IPv6 source address, said device for use in a home gateway adapted to connect a user terminal to said IPv6 domain, comprising:

165. As discussed under Ground 1, Li teaches all the elements of limitation 8[pre]. *See* §VII.A.9.a.

- (b) 8[a]: means for identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address, and a destination port number;

166. Li teaches limitation 8[a] except for identifying a destination port number. *See* §VII.A.9.b. A POSITA would have found it obvious to modify Li to identify the destination port number of the IPv6 address. *See* §VII.B.1-2.

- (c) 8[b]: 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, 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 modifying the data packet using the regularized address; and

167. Li renders obvious the additional limitation of claim 8[b]. See §VII.A.9.c.

- (d) 8[c]: means for routing the modified data packet to its destination.

168. Li renders obvious the additional limitation of claim 6. See §VII.A.9.d.

C. Ground 3: Claims 1, 5-8 Would Have Been Obvious in View of Li alone or in view of Li-2 in further view of Paunikar

1. Paunikar (EX1057)

169. U.S. Patent 7,752,334 (“Paunikar”) describes a method and system for Network Address Translation (“NAT”) and Network Address Port Translation (“NAPT”) of data packets between a private network and a public network. EX1057, 3:7-12. In particular, Paunikar discloses that “[t]raditionally NAT has two modes of operation—basic NAT and Network Address Port Translation (NAPT)” that “both provide benefits to the network provider and/or network user.” *Id.*, 1:40-2:11. Paunikar discloses an “intelligent network address translator” that can analyze data packets “based on packet type (i.e., protocol and destination address) and decide[],

based on the packet type, what mode of network address translation will be applied.”
Id., 5:18-33.

170. Figure 3 of Paunikar describes an embodiment of the intelligent network address translation:

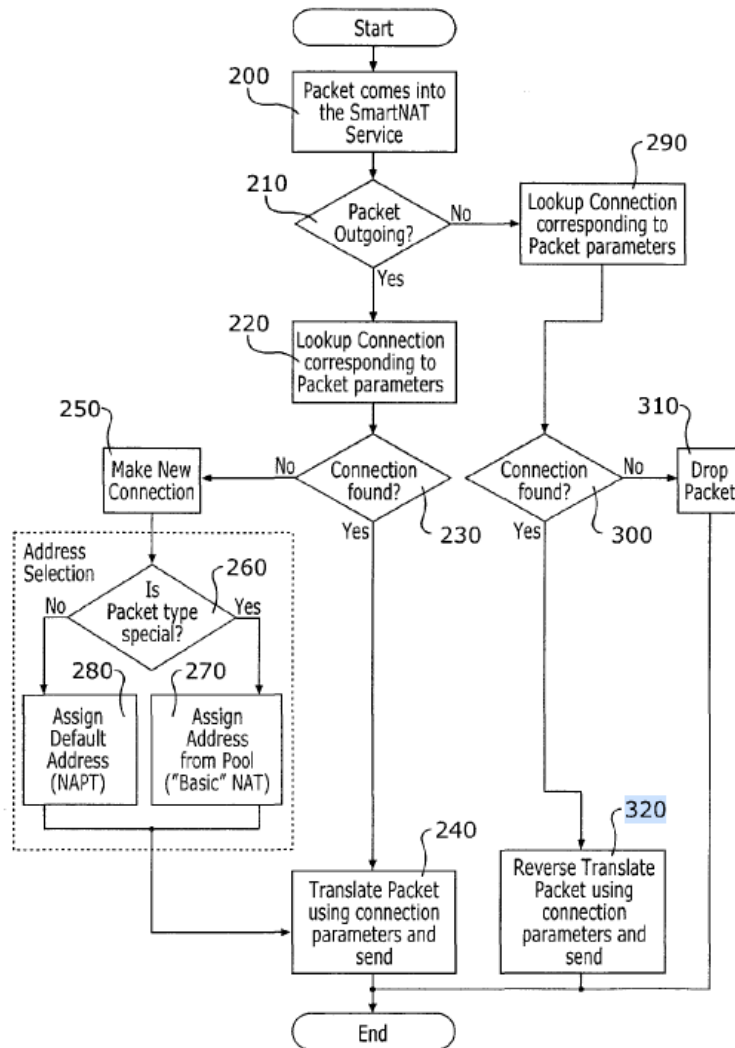


FIG. 3

EX1057, Fig. 13, 8:51-58.

171. A data packet arrives at a SmartNAT service associated with a network gateway which begins analysis of the data packet. The SmartNAT service first considers whether the packet is incoming or outgoing (*i.e.*, does the system need to provide translation to the source address or the destination address). EX1057, 8:53-64. For an outgoing packet, the SmartNAT system evaluates whether the IP address exists in the connection memory. *Id.*, 9:13-15. If a connection is found in the translation table, the system performs a translation in step 240 and completes the connection. *Id.*, 9:7-12. If no connection is found in the memory, the data packet creates a new entry for the source address. *Id.*, 9:13-15. If the packet is “special” (*i.e.*, having characteristics defined by a network administrator), the SmartNAT system performs a basic network address translation and assigns a translated address from the pool of available public IP addresses and enters that connection into the NAT table. Otherwise, the SmartNAT system performs Network Address/Port Translation (NAPT) and assigns the default public IP address with a specific source port. *Id.*, 9:33-40. The operation of basic NAT and NAPT were well-known to POSITAs at the time. EX1058, 13-21 (describing well-known forms of NAT, including Basic NAT and NAPT).

172. For an incoming packet, the data packet travels through the path denoted by steps 290, 300, and 320. At step 290, the system determines if the incoming packet corresponds with an address in the network. *Id.*, 9:41-46. If a

connection to the network is found (*i.e.*, “Yes” at step 300), then the SmartNAT system “reverse translates” the packet and communicates it with the private network. *Id.*, 9:54-57.

2. The Li-Li-2-Paunikar Combination

173. Grounds 1 and 2 demonstrate that Li and Li-2 disclose and render obvious receiving a data packet from an IPv4 domain in an IPv6 domain and identifying an IPv6 address constructed by concatenating an IPv6 prefix, a destination IPv4 address, and a destination port number. Moreover, Li-2 expressly discloses that NAT can be integrated into its system as part of the regularizing step, although it does not describe the operation details for a NAT. Although a POSITA at the time would have been aware of these operation details, it would also have been obvious to combine the teachings of Paunikar regarding network address and port address translation. Thus, in the Li-Li-2-Paunikar combination, Li would operate in its disclosed manner but would further implement Li-2’s teaching of including port numbers in constructed IPv6 addresses. When a data packet with a Li-Li-2 construction interacts with a gateway, Paunikar discloses how the gateway would regularize the data packet.

3. Motivations to Combine Li, Li-2, and Paunikar

174. A POSITA would have been motivated to combine the teachings of Li, Li-2, and Paunikar and would have had a reasonable expectation of success.

175. **First**, the prior art includes express teachings that would have led a POSITA to combine Li, Li-2, and Paunikar. For instance, Li-2 expressly states that “IPv4 NAT can easily be integrated into the system.” EX1038, 18. Moreover, as discussed above, several prior art documents expressed a need to not only make IPv4 domains interoperable with IPv6 domains in order to smooth the transition from IPv4 to IPv6, but also a need to slow IPv4 address exhaustion. *See* §III.A. A POSITA would have known that to address IPv4 address exhaustion NAT functionality could be applied to map public addresses to private networks and vice-versa (*see* §III.D).

176. Further, the prior art explains that NAT provides additional benefits beyond conserving addresses, such as improved security. For instance, Paunikar explains that companies may employ NAT to “ensure security since each outgoing or incoming request must go through a translation process that also offers the opportunity to quality or authenticate the request or match it to previous requests.” EX1057, 1:22-31. Other prior art references similarly explain that NAT improves security by, for instance, “mask[ing] the internal IP addresses of an Intranet.” EX1058, 7 (“While IPv6 does not need NAT for connectivity, other NAT features such as address hiding are valuable.”); *see also* EX1060 (RFC 1919) at 14 (“Address hiding is generally viewed as a Good Thing, since one of the purposes of deploying proxies is to disclose as little information about an internetwork as possible.”).

177. **Second**, this combination would have involved applying a known technique to a known device ready for improvement to yield predictable results. Paunikar discloses the application of NAT functionality to various protocols, to include IPv4 and IPv6. *See generally*, EX1057. By the time of the alleged invention, NAT/NAT-PT were well-known techniques. A POSITA could clearly have applied the techniques from Paunikar to Li's system in order to obtain various benefits, such as masking the internal architecture of a network. The combination would have been straightforward and yielded predictable results, as Li itself indicates that the ability to support NAT devices was a "basic requirement" of its system. EX1038, 18.

4. Claim 1

- (a) 1[pre]: A method of receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain, said packet comprising an IPv6 destination address and an IPv6 source address, said method being executed in a home gateway adapted to connect a user terminal to said IPv6 domain, and comprising the following steps:

178. As discussed under Ground 1, Li teaches all the elements of limitation 1[pre]. *See* §VII.A.5.a.

- (b) 1[a]: identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address and a destination port number

179. Li teaches limitation 1[a] except for concatenating a destination port number. *See* §VII.A.5.b. A POSITA would have found it obvious to modify Li to replace 16 bits of its unused padding with a destination port number in order to share

a single IPv4 address among multiple hosts, thereby slowing IPv4 address exhaustion. See §VII.B.2. Evidence of simultaneous invention shows that not only would a POSITA have been motivated to make this modification and would have been successful in doing so, but POSITAs at that time were actively modifying Li in such a manner. See §VII.B.1.

- (c) 1[b]: 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, 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 modifying the data packet using the regularized address; and

180. To the extent Patent Owner argues that Li or Li-2 does not disclose or render obvious limitation 1[b], this limitation was at least obvious in further view of Paunikar. The combined system would thus perform “if necessary, regularizing at least one of the IPv6 source address and the IPv6 destination address [*e.g.*, Li’s constructed IPv6 destination address] by replacing said at least one IPv6 address [*e.g.*, Paunikar’s NAT and NAPT system replace the constructed IPv6 destination address with an internal IP destination address], 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 modifying the data packet using the regularized address [*e.g.*, the modified packet uses the internal IP destination address].”

181. Li and Li-2 disclose that after receiving an IPv6 packet with an IPv6 destination address, the home gateway (*i.e.*, the edge router or access gateway of Figure 11) reverses the address translation to conform the IPv6 address to a public IPv4 address. *See* §VII.A.5.b. It would have been obvious to further regularize the IPv4 address in view of Paunikar's disclosures regarding NAT.

182. Figure 3 of Paunikar describes Paunikar's regularizing process:

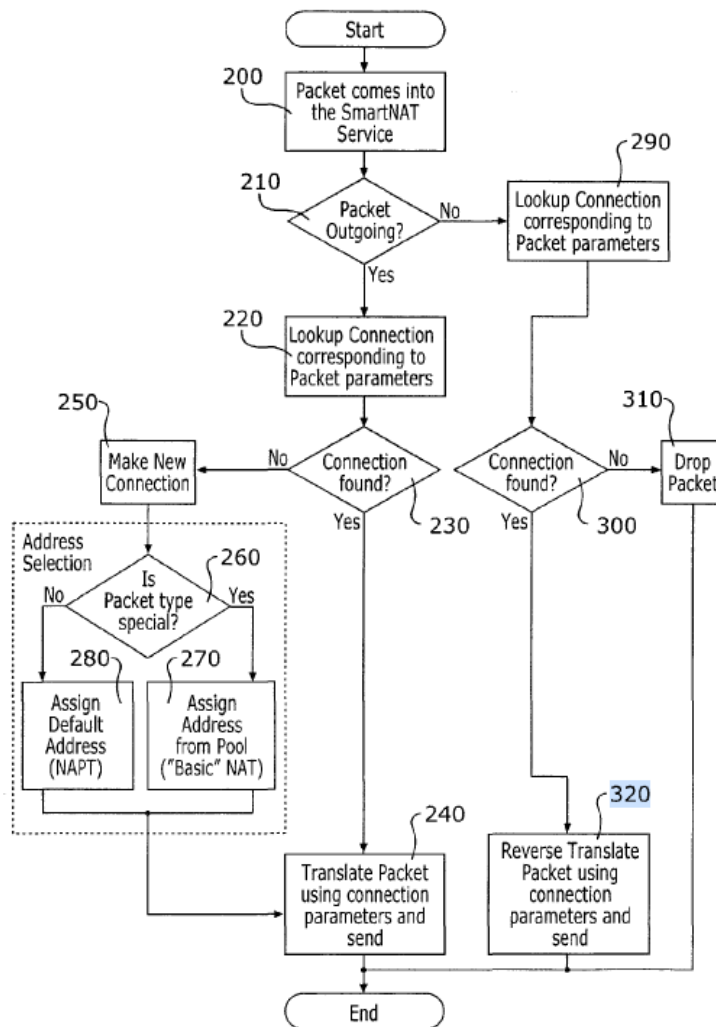


FIG. 3

A data packet arrives at a SmartNAT service associated with a network gateway which begins analysis of the data packet. The SmartNAT service first considers whether the packet is incoming or outgoing (*i.e.*, does the system need to provide translation to the source address or the destination address). EX1057, 8:53-64. For an outgoing packet, the SmartNAT system evaluates whether the IP address exists in the connection memory. 9:13-15. If a connection is found in the translation table, the system performs a translation in Step 240 and completes the connection. 9:7-12. If no connection is found in the memory, the data packet creates a new entry for the source address. 9:13-15. If the packet is “special” (*i.e.*, having characteristics defined by a network administrator), the SmartNAT system performs a basic network address translation and assigns a translated address from the pool of available public IP addresses and enters that connection into the NAT table. Otherwise, the SmartNAT system performs Network Address/Port Translation (NAPT) and assigns the default public IP address with a specific source port. 9:33-40.

183. For an incoming packet, at step 290, the system determines if the incoming packet corresponds with an address in the network (*i.e.*, determine if regularizing the address is necessary). *Id.*, 9:41-46. If a connection to the network is found (*i.e.*, “Yes” at step 300), then the SmartNAT system “reverse translates” (*i.e.*, regularizes the destination packet by replacing the public IPv4 address with a private IP address) the data packet and communicates it with the private IPv4 address

to the destination host. *Id.*, 9:54-57, Fig. 3 (step 320, “reverse translate packet using connection parameters and send”); *see also* Fig. 2 (step 190).

(d) 1[c]: routing the modified data packet to its destination.

184. The combined system of Li, Li-2 and Paunikar discloses and renders obvious limitation 1[c]. *See* §VII.A.5.d.

5. Claim 5

(a) 5[pre]: 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:

185. To the extent the preamble is limiting, Li, Li-2 and Paunikar render obvious limitation 5[pre]: “The method according to claim 1 [discussed *supra* (§VII.A.5)], wherein when the data packet comes from the IPv4 domain [*e.g.*, the top left IPv4 access network in Figure 11 of Li] and goes to said user terminal [*e.g.*, hosts of AS 24348 Access Network or AS 24349], the step of regularizing the destination address comprises the following steps: [Paunikar’s NAT/NAPT translation].”

186. ***The method of claim 1.*** As discussed in Section VII.C.4, the combination of Li, Li-2, and Paunikar disclose all the limitations of the method of claim 1.

187. ***Wherein when the data packet comes from the IPv4 domain and goes to said user terminal.*** As discussed in Section VII.A.5.a, Li teaches an IPv4 domain

connected to IPv4 hosts wherein data packets travel from the IPv4 domain to the IPv4 hosts. EX1037, 6, 7.

188. ***The step of regularizing the destination address comprises the following steps.*** As discussed in Section VII.C.4.c, Paunikar discloses a process for “the step of regularizing the destination address comprises the following steps.

- (b) 5[a]: searching an address translation table of the gateway for an entry associating with said constructed destination address a native destination address;

189. In my opinion, Paunikar teaches limitation 5[a].

190. Figure 3 of Paunikar teaches a system for a SmartNAT translation service for incoming and outgoing data packets (*i.e.*, translating destination or source addresses). EX1057, Fig. 3. For an incoming data packet, after Li inverses the address mapping for the constructed IPv6 address Paunikar may further process the incoming data packet through a lookup connection table to identify an entry associated with the constructed destination address (*i.e.*, the regularizing processes searches the address translation table for an existing entry associated with the destination address). *Id.*, 4:45-51 (“In order to reverse translate the incoming data packets, the system may include a connection lookup table to determine connection parameters for incoming data packets and a reverse translator module that performs reverse network address translation on incoming data packets based on the determined connection parameters in the connection lookup table.”), 9:51-57 (“If an

entry is found in the corresponding connection memory, then the connection parameters and the private address mapped to translation network address are used, at step 320, to reverse translate the data packet back to the original private network address and the reverse translated data packets are then communicated to the private address realm.”). Indeed, Paunikar discloses that its system performs either Basic NAT or NAPT, both of which would involve searching an address translation table of the gateway in this manner. *See also* EX1058, 16-17 (explaining “Address translation and lookup” for “Basic NAT” and “NAPT”); EX1059, 338-341 (same).

- (c) 5[b]: if an entry is found, replacing the constructed destination address of the data packet with the native destination address from that entry.

191. Paunikar teaches limitation 5[b]. Figure 3 of Paunikar discloses, “if an entry is found, replacing the [] destination address of the data packet with the native destination address from that entry.” As the lookup table analyzes the packet parameters of the data packet discussed above, if an entry associated with the destination is found (*e.g.*, “yes” path between blocks 300 and 320 of figure 3), the SmartNAT service translates the destination address (*i.e.*, replaces the destination address with a native destination address). EX1057, 9:51-57 (“If an entry is found in the corresponding connection memory... reverse translate the data packet back to the original private network address”); §VII.C.5.b.

6. Claim 6: A non-transitory computer program product that is at least one of downloadable from a communications

network and/or stored on a computer-readable medium and/or executable by a microprocessor, comprising program code instructions for executing the method according to claim 3 of sending a data packet from an IPv6 domain to a destination address of an IPv4 domain.

192. The combination of Li, Li-2, and Paunikar disclose “[t]he method according to claim 5 [discussed above] 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.”

193. As discussed in connection with Ground 1, a POSITA would have also found it obvious to verify that the destination port numbers match in order to identify errors and security issues, such as spoofed packets. EX1032 at 28-29.

- 7. Claim 7: IPv6 domain access equipment according to claim 6, wherein the IPv4 is a public network, said equipment comprising first means for advertising to the IPv4 domain IPv4 addresses of network equipments connected to the IPv6 domain and second means for advertising in the IPv6 domain IPv6 addresses constructed from IPv4 addresses of the IPv4 domain**

194. Li and Paunikar disclose all the additional limitations of claim 7. Li describes “in-lab experiments” that its inventors performed to verify Li’s invention. EX1037, 5. Specifically, the inventors set up “5 computers” that were “connected by one over 100Mbps direct Ethernet links.” *Id.* A POSITA would have known that these computers comprise memory and a processor, which they use to perform the programmed functions. Similarly, Paunikar teaches the disclosed methods can occur

on “other types of operating systems, computing platforms, computer programs, and/or general-purpose machines.” EX1057, 5:58-61. A POSITA would recognize that the programmed functions are executed by processing program code instructions to perform the steps discussed for claim 1 above.

8. Claim 8

- (a) 8[pre]: A device for receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain, said packet comprising an IPv6 destination address and an IPv6 source address, said device for use in a home gateway adapted to connect a user terminal to said IPv6 domain, comprising:

195. To the extent the preamble is limiting, Li teaches all the limitations of 8[pre]. *See* §VII.A.9. The ’845 patent explains that its disclosed device “includes the hardware elements conventionally found in a standard computer or a dedicated router,” like a processor and memory. EX1001, 12:38-46. Li’s edge router corresponds to such a device. Further, Li discloses all the functions associated with the device. *See* §VII.A.5.

- (b) 8[a]: means for identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address, and a destination port number;

196. Li and Li-2 teach claim limitation 8[a]. *See* §VII.A.9.b. The references renders obvious “the home gateway 20 that includes a processor programed to perform the step of [*e.g., the top-left edge router*] identifying an IPv6 destination address [*e.g., Li’s translated IPv6 address that corresponds to a host in the top-*

right IPv4 access network] constructed by concatenating an IPv6 prefix [*e.g.*, **Li’s IPv6 global routing prefix**], an IPv4 destination address [*e.g.*, **Li’s received IPv4 destination address that corresponds to a host in the top-right IPv4 access network**], and a destination port number [*e.g.*, **Li’s received destination port number in light of Li-2’s teachings**].”

- (c) 8[b]: 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, 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 modifying the data packet using the regularized address; and

197. Li and Paunikar render obvious limitation 8[b]. See §VII.C.4.c.

- (d) 8[c]: means for routing the modified data packet to its destination.

198. The combined system of Li, Li-2 and Paunikar discloses and renders obvious limitation 1[c]. See §VII.C.4.d.

D. Ground 4: Claims 1, 5-8 Would Have Been Obvious in View of Li and Wu

1. Wu (EX1039)

199. Wu teaches a “new tunnel technology in the transition period from IPv4 Internet to IPv6 Internet.” EX1039, 2. Wu describes a Network Address Translation (“NAT”) technology that receives a data packet from an internal network and translates a private address of the data packet into a public address, also called an

“external address.” *Id.*, 9. The NAT further translates a source port number into an external port number. *Id.* However, “the present NAT basically doesn’t allow IPv6-in-IPv4 data packets... to pass.” *Id.*, 10.

200. Wu therefore proposes a “method for automatic tunnel which can traverse NAT” that “aims to solve the mutual communication problem between the IPv6 node and other IPv6 nodes... within NAT field.” *Id.* Specifically, a “Silkroad client”—a dual-stack client that has both IPv4 and IPv6 capabilities—constructs an IPv6 address by concatenating different pieces of information, as shown in Figure 1:

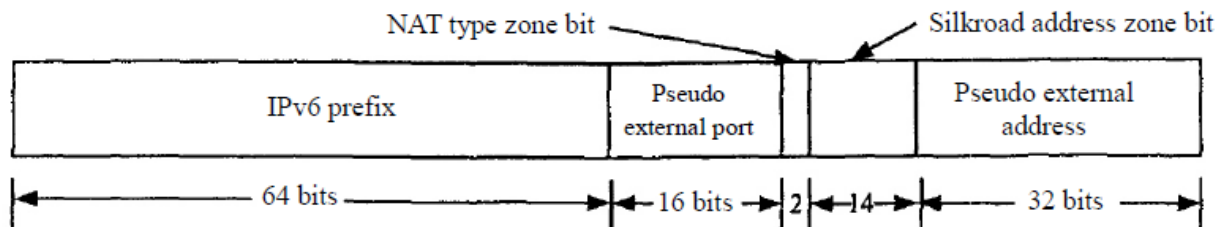


Fig. 1

Id., Fig. 1. The first 64 bits of the constructed IPv6 address correspond to an IPv6 prefix. *Id.*, 11. The next 16 bits correspond to the external port number, which can be obfuscated by using a bit-wise operation. *Id.* The next 2 bits indicate the type of NAT used in the network. *Id.* The next 14 bits are padded with 0s, reserved for a “Silkroad address.” *Id.* And the final 32 bits correspond to the external address, which can be obfuscated by using a bit-wise operation. *Id.*, 12. Accordingly, the

client can construct an IPv6 address by concatenating an IPv6 prefix, a port number, and an IPv4 address, along with other information. *Id.*, 13.

201. Wu and the '845 patent are analogous art because the references are in the same field of endeavor as each other, *i.e.*, computer networking and routing data packets. For example, the '845 patent explains that “[t]he field of the invention is that of telecommunications networks, in particular IP telecommunications networks transporting data packets.” EX1001, 1:17-21. Wu is also directed to changes in “the existing IPv4 networking routing system.” EX1039, 9.

2. The Li-Wu-Paunikar Combination

202. In the Li-Wu-Paunikar combination, Li teaches the general network structure and the steps of receiving a data packet from an IPv4 domain in an IPv6 domain. Li further teaches constructing an IPv6 address by concatenating an IPv6 prefix, an IPv4 address, and 32-bits of zero padding. Wu supplements Li's disclosures by teaching replacing 16 bits of the 32-bit zero padding with a destination port number in order to uniquely identify hosts and mitigate IPv4 address exhaustion. Thus, in the Li-Wu-Paunikar combination, Li would operate in its disclosed manner but would further implement Wu's teaching of including port numbers in constructed IPv6 addresses in order to uniquely identify hosts and prevent IPv4 address exhaustion. When a data packet with a Li-Wu construction

interacts with a gateway, Paunikar discloses how the gateway would regularize the data packet.

3. Motivations to Combine Li, Wu, and Paunikar

203. A POSITA would have been motivated to combine the teachings of Li, Wu, and Paunikar and would have had a reasonable expectation of success in doing so.

204. **First**, the prior art includes express teachings that would have motivated a POSITA to combine the references. By the time of the '845 patent, it was well-understood that IPv4 address would soon run out. EX1001, 1:60-61 (“It is commonly accepted in the IP service provider community that IPv4 public addresses are going to run out.”); 1013, 1, 8-9 (“The Internet engineering community ... recognized the impending exhaustion of IPv4.”). POSITAs would have therefore been motivated to improve Li’s system by applying known techniques for extending the IPv4 address space, thereby slowing IPv4 address depletion. Specifically, a POSITA would have recognized that port numbers could be used to uniquely identify hosts and allow them to share a single IPv4 address. See §III.B (discussing EXS1026-1033, which disclose using port numbers to extend the IPv4 address space). Accordingly, a POSITA would have been motivated to modify Li by replacing 16 bits of the unused padding in the constructed IPv6 destination address with a destination port number, as taught by Wu.

205. A POSITA would have had a reasonable expectation of success in combining Li and Wu and there would have been no unexpected results because the combination merely involves the straightforward application of replacing unused padding in Li's constructed IPv6 address with the destination port number, as taught by Wu.

206. **Second**, a POSITA would have been particularly motivated to apply Wu to Li because the combination would have been of a known technique (*i.e.*, Wu's teaching of constructing an IPv6 address by concatenating a port number) to a known device (*i.e.*, Li's edge router that already constructs IPv6 addresses by concatenating information, including 32 bits of unused padding) to yield a predictable result (*i.e.*, an edge router that can construct an IPv6 address that includes a destination port number in place of 16 bits of the unused padding). As discussed, by the time of the '845 patent, POSITAs recognized the need to extend the IPv4 address space and a well-known technique for doing so was to form an IPv6 address by concatenating IPv4 address and port number. *See* §III.C (discussing EXS1021-1032). A POSITA would have understood that parts of the constructed IPv6 address in Li were unused, as signified by the zero padding, and had no semantic significance. A POSITA would have found it obvious to use this unused space for semantically meaningful information, such as port numbers, as suggested by Wu, as this addition would have provided more information as part of the address. A POSITA would have further

understood that since this part of the address was effectively unused in Li, the functionality provided by Li would not be adversely affected by adding port numbers. The proposed combination thus applies this well-known technique, as disclosed by Wu, to Li's known device, an edge router.

207. **Third**, this combination would have been obvious because it is a simple substitution of one known element for another to obtain predictable results. Combining Li and Wu would involve merely substituting 16 bits of unused padding in Li's constructed IPv6 address with a 16-bit destination port number, as taught by Wu, to yield a predictable result—a constructed IPv6 address that contains a destination port number. As Wu shows, the substituted component—the destination port number—and its function were known in the art. And based on the state of the art, as described above and in the technology background section, a POSITA could have substituted the unused padding for the destination port number, and the results of the substitution would have been predictable. *See* §III.C.2-3.

208. Additionally, Li, Wu, and the '845 Patent are concerned with the interoperability of IPv4 networks and IPv6 networks. The '845 patent explains that “[t]he invention improves upon the [previously discussed] situation by means of a method of receiving a data packet coming from an IPv4 domain in an IPv6 domain” and is specifically directed to “a period of transition during which IPv6 domains will have to interconnect with IPv4 domains.” EX1001, 2:46-64. Li similarly explains

that “at this stage, interoperability of applications in a dual-protocol environment has, replacing connectivity, become the sharpest issue of the IPv4-IPv6 transition.” EX1037, 1. And Wu describes itself as “a tunnel technology in the transition period from IPv4 to IPv6, so as to realize the transition from IPv4 to IPv6.” EX1039, 9.

4. Evidence of Simultaneous Invention Demonstrates The Obviousness Of Combining Li And Wu

209. As discussed above, there are multiple examples of POSITAs simultaneously inventing the '845 claims. See §VII.B.1.

5. Claim 1

- (a) 1[pre]: A method of receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain, said packet comprising an IPv6 destination address and an IPv6 source address, said method being executed in a home gateway adapted to connect a user terminal to said IPv6 domain, and comprising the following steps:

210. As discussed under Ground 1, Li teaches all the elements of limitation 1[pre]. See §VII.A.5.a.

- (b) 1[a]: identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address and a destination port number

211. Li teaches limitation 1[a] with the exception of “a destination port number.” See §VII.A.5.(b). Wu teaches that a constructed IPv6 address may be formed by concatenating the received destination port. Specifically, Wu teaches a “Silkroad client”—which, like the Edge Router in Li, is a dual-stack client that has

both IPv4 and IPv6 capabilities (EX1039, 11)—that constructs an IPv6 address by concatenating an IPv6 prefix, an IPv4 address, and a port number:

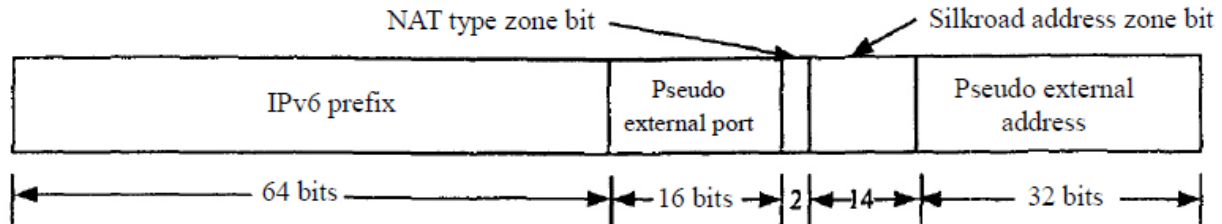
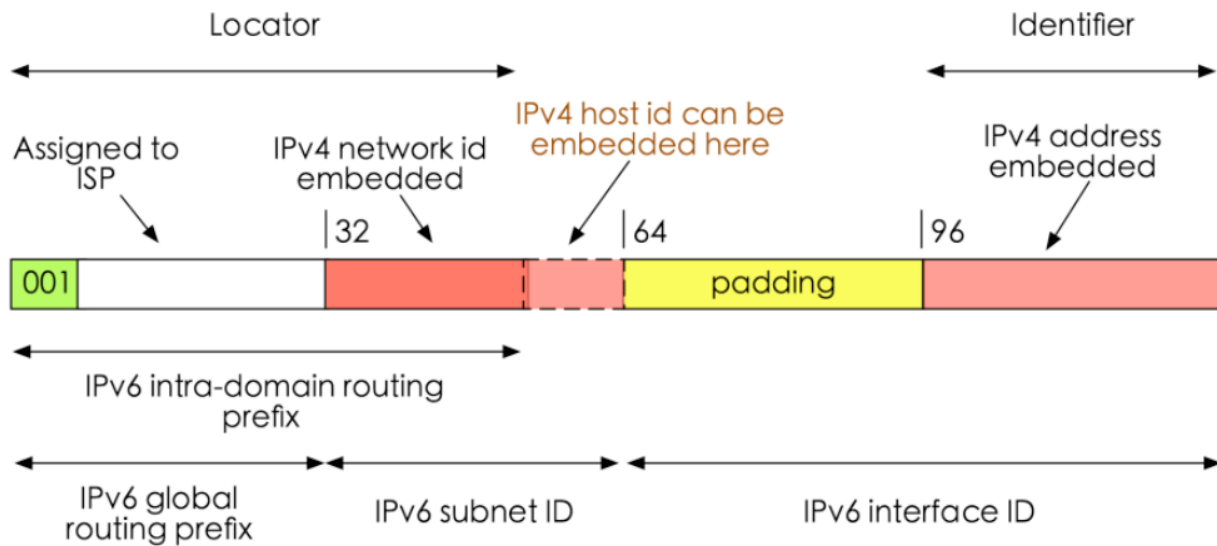


Fig. 1

EX1039, Fig. 1. Wu’s constructed IPv6 address comprises an IPv6 prefix and an IPv4 address (Wu calls the IPv4 address an “external address” or “pseudo external address”), like the constructed IPv6 address described in Li. *Id.*, 9-12. Wu additionally teaches that the constructed IPv6 address may include a port number, called a “pseudo external port number.” *Id.*, 11.

212. Li already teaches that its IPv6 address concatenates 32-bits of zero padding (bits 64-96) following the IPv6 prefix and IPv4 address:



EX1037, 3. Thus, a POSITA would have recognized that there is more than sufficient space to further include a 16 bit destination port number in Li's address by replacing 16 bits of the unused padding. As such, a POSITA would have been motivated to combine Li and Wu and would have found it obvious to use the unused padding portion of Li to embed a port number. See §VII.D.3. Evidence of simultaneous invention shows that not only would a POSITA have been motivated to combine Li and Wu would have been successful in doing so, but POSITAs at that time were actively modifying Li in such a manner. See §VII.D.4.

- (c) 1[b]: 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, 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 modifying the data packet using the regularized address; and

213. Paunikar and Li render obvious limitation 1[b]. See §VII.C.4.c.

(d) 1[c]: routing the modified data packet to its destination.

214. Paunikar teaches limitation 1[c]. See §VII.C.4.d.

6. Claim 5

(a) 5[pre]: 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:

215. To the extent the preamble is limiting, Paunikar renders obvious limitation 5[pre]. See §VII.C.5.a.

(b) 5[a]: searching an address translation table of the gateway for an entry associating with said constructed destination address a native destination address;

216. Paunikar discloses limitation 5[a]. See §VII.C.5.b.

(c) 5[b]: if an entry is found, replacing the constructed destination address of the data packet with the native destination address from that entry.

217. Paunikar discloses limitation 5[b]. See §VII.C.5.c.

7. Claim 6: A non-transitory computer program product that is at least one of downloadable from a communications network and/or stored on a computer-readable medium and/or executable by a microprocessor, comprising program code instructions for executing the method according to claim 3 of sending a data packet from an IPv6 domain to a destination address of an IPv4 domain.

218. Li and Paunikar render obvious claim 6. See §VII.C.6.

8. Claim 7: IPv6 domain access equipment according to claim 6, wherein the IPv4 is a public network, said equipment

comprising first means for advertising to the IPv4 domain IPv4 addresses of network equipments connected to the IPv6 domain and second means for advertising in the IPv6 domain IPv6 addresses constructed from IPv4 addresses of the IPv4 domain

219. Li and Paunikar render obvious claim 7. *See* §VII.C.7.

9. Claim 8

(a) 8[pre]: A device for receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain, said packet comprising an IPv6 destination address and an IPv6 source address, said device for use in a home gateway adapted to connect a user terminal to said IPv6 domain, comprising:

220. Li and Paunikar render obvious limitation 8[pre]. *See* §VII.C.8.a.

(b) 8[a]: means for identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address, and a destination port number;

221. Li teaches the limitation 8[a] except identifying a destination port number. *See* §VII.C.4.c. However, Wu teaches identifying a destination port number. *See* §VII.D.5.b.

(c) 8[b]: 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, 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 modifying the data packet using the regularized address; and

222. Li and Paunikar render obvious limitation 8[b]. *See* §VII.C.8.c.

- (d) 8[c] means for routing the modified data packet to its destination.

223. Paunikar render obvious limitation 8[c]. See §VII.C.8.d.

VIII. CONCLUSION

224. For the reasons stated above, it is my opinion that the Challenged Claims would have been obvious at the time of the invention.

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

* * *

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true. I further declare that these statements were made with the knowledge that willful false statements and the like are punishable by fine or imprisonment, or both, under 18 U.S.C. § 1001.

Respectfully submitted,

Date: February 9, 2024

/s/ Samrat Bhattacharjee
Samrat Bhattacharjee, Ph.D