UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

AMERICAN AIRLINES, INC.

and

SOUTHWEST AIRLINES CO.,

Petitioners

v.

INTELLECTUAL VENTURES I LLC,

Patent Owner

IPR2025-00786
U.S. Patent No. 7,949,785

PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 7,949,785

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

TABLE OF CONTENTS

TABLE OF AUTHORITIES

Page(s)

CASES

STATUTES

OTHER AUTHORITIES

EXHIBIT LIST

Exhibit No.	Description
1001	U.S. Patent No. 7,949,785 ("'785 Patent")
1002	File History for U.S. Patent No. 7,949,785 (Appl. No. 10/403,818)
1003	U.S. Patent No. 6,970,941 ("Caronni-I")
1004	U.S. Patent No. 7,814,228 ("Caronni-II")
1005	U.S. Patent No. 6,766,371 ("Hipp")
1006	Huitema, C., Network Working Group Request for Comment (RFC): 1383, entitled An Experiment in DNS Based IP Routing ("RFC-1383")
1007	*Intellectual Ventures I LLC et al. v. American Airlines, Inc.*, No. 4:24-cv-980 (E.D. Tex.) – Complaint
1008	*Intellectual Ventures I LLC et al. v. Southwest Airlines Co.*, No. 7:24-cv-277 (W.D. Tex.) – Complaint
1009	*Intellectual Ventures v. Liberty Mutual*, No. 23-cv-525 (E.D. Tex.) – Complaint
1010	*Intellectual Ventures v. Liberty Mutual*, No. 23-cv-525 (E.D. Tex.) – Motion to Dismiss – Voluntary Dismissal (FRCP 41(a))
1011	Declaration of Dr. Erez Zadok ("Zadok Decl.")
1012	Curriculum Vitae of Dr. Erez Zadok
1013	Reserved
1014	Microsoft Computer Dictionary, 3rd ed., 1997, excerpts (Microsoft Computer Dictionary)
1015	Re-Exam 90/0019,519 filed on May 22, 2024 ('785 re-exam)
1016	Reserved
1017	Andrew S. Tanenbaum, *Computer Networks*, 2nd ed., 1988, excerpts (Tanenbaum)
1018	W. Richard Stevens, *TCP/IP Illustrated Volume 1, The Protocols*, 1994, excerpts (Stevens)
1019	William R. Cheswick & Steven M. Bellovin, *Firewalls and Internet Security, Repelling the Wily Hacker*, 1994, excerpts (Cheswick)
1020	RFC 1034, "Domain Names - Concepts and Facilities," P. Mockapetris, November 1987 (RFC-1034)
1021	RFC 1035, "Domain Names - Implementation and Specification," P. Mockapetris, November 1987 (RFC-1035)

1022	RFC 2406, "IP Encapsulating Security Payload (ESP)", S. Kent and R. Atkinson, November 1988 (RFC-2406)
1023	RFC 2131, "Dynamic Host Configuration Protocol," R. Droms, March 1997 (RFC-2131)
1024	Linux DNS Server Administration, Craig Hunt, 2000, excerpts (Hunt)
1025	Claim Chart for '785 Patent, *Intellectual Ventures I LLC et al. v. American Airlines, Inc.*, No. 4:24-cv-980-ALM (E.D. Tex. Nov. 2, 2024) (Exhibit 10 to Complaint, Dkt. 1-11) ('785 claim chart)
1026	Claim Chart for '785 Patent, *Intellectual Ventures I LLC et al. v. Southwest Airlines Co.*, No. 7:24-cv-277 (W.D. Tex. Nov. 2, 2024) (Exhibit 3 to Intellectual Ventures I LLC's and Intellectual Ventures II LLC's Preliminary Infringement Contentions) ('785 claim chart)
1027	Network Working Group Request for Comment (RFC): 2401, entitled *Security Architecture for the Internet Protocol*, by S. Kent *et al.*, 1998 (RFC-2401).
1028	Andrew S. Tanenbaum, *Computer Networks*, 4th ed., 2003, excerpts (Tanenbaum2)

LISTING OF CHALLENGED CLAIMS

Claim 1	
1[pre]	A virtual network system, comprising:
1[a]	a virtual network manager implemented with a first device memory and a first device processor of a first computing device, the virtual network manager configured to register devices in a virtual network that is defined by a domain name, each device in the virtual network being identified to the other devices by a virtual network address that is unique for each device and not directly routable via a public network, the virtual network manager further configured to distribute a virtual network address to a device when the device is registered in the virtual network;
1[b]	a route director implemented with a second device memory and a second device processor of a second computing device, the route director configured to communicate data between the devices that are registered in the virtual network, the data being communicated as encapsulated packets from a source device to a destination device, an encapsulated packet including a first virtual network address that corresponds to the source device and a second virtual network address that corresponds to the destination device; and
1[c]	the virtual network manager further configured to receive a DNS request from the source device, and return a public network address of the route director, a private network address for the destination device, and the second virtual network address that corresponds to the destination device.
Claim 30	
30[pre]	A virtual network manager, comprising:
30[a]	a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;
30[b]	a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to: receive a registration request from an agent associated with a device; distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and
30[c]	a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a

	private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.
Claim 35	
35	The virtual network manager of claim 30 further comprising a join module configured to receive a join request from the agent associated with the device to indicate that the device is connected for data communication within the virtual network, the join module further configured to receive a leave request from the agent associated with the device to indicate that the device will be disconnected from data communication within the virtual network.
Claim 36	
36	The virtual network manager of claim 35 wherein the join module is further configured to provide virtual network addresses to the devices that are registered in the virtual network.
Claim 37	
37	The virtual network manager of claim 35 wherein the join module is further configured to maintain data to associate a virtual network address with a device in the virtual network.
Claim 38	
38[pre]	A virtual network system, comprising:
38[a]	a computing device that includes at least a memory and a processor configured to implement a network manager of a virtual network that is defined by a public domain name, the network manager configured to distribute virtual network addresses to devices that register as members in the virtual network, each device in the virtual network being identified to the other devices by a virtual network address associated with the device;
38[b]	a first virtual network agent associated with a first device that is registered as a member in the virtual network;
38[c]	at least a second virtual network agent associated with at least a second device that is registered as a member in the virtual network;
38[d]	a route director configured to route communications between the first device and the at least second device in the virtual network via the respective first and second virtual network agents, the communications configured for routing as encapsulated packets that include a first virtual network address that is not directly routable corresponding to the first device and a second virtual network address that is not directly routable corresponding to the at least second device; and

38[e]	the network manager includes a DNS server configured to provide authoritative responses for DNS queries in the virtual network, the DNS server further configured to receive a DNS query from the first device and return a network address of the route director, a network address of the second device, and the virtual network address of the second device.
Claim 48	
48[pre]	A computer-implemented method, comprising:
48[a]	receiving registration requests from devices that request to be registered as members of a virtual network that is defined by a domain name having an associated public network address in a public network, each of the devices having an associated private network address;
48[b]	distributing a virtual network address to a device to register the device as a member in the virtual network, each device in the virtual network being identified to the other devices by the virtual network address that is associated with the device;
48[c]	routing communications between the devices that are registered in the virtual network, the communications being routed as encapsulated packets from a source device to a destination device, an encapsulated packet including a first virtual network address that corresponds to the source device and a second virtual network address that corresponds to the destination device; and
48[d]	transmitting a response to a DNS request received from one of the devices that are the members in the virtual network, the response to the DNS request including a public network address of a route director that registers the devices, a public network address of the destination device, and the second virtual network address that corresponds to the destination device.
Claim 62	
62[pre]	One or more processor readable storage media devices comprising processor readable code that, if executed by a computer device, implements a virtual network manager to:
62[a]	receive registration requests from devices that request to be registered as members of a virtual network that is defined by a domain name having an associated public network address in a public network, each of the devices having an associated private network address;
62[b]	distribute a virtual network address to a device to register the device as a member in the virtual network, each device in the virtual network being identified to the other devices by the virtual network address that is associated with the device;

62[c]	manage communications routed between the devices that are registered in the virtual network, the communications routed as encapsulated packets from a source device to a destination device, an encapsulated packet including a first virtual network address that corresponds to the source device and a second virtual network address that corresponds to the destination device; and
62[d]	transmit a response to a DNS request received from one of the devices that are the members in the virtual network, the response to the DNS request including a public network address of the virtual network manager, a public network address of the destination device, and the second virtual network address that corresponds to the destination device.
Claim 75	
75[pre]	A virtual network system, comprising:
75[a]	a computing device that includes at least a memory and a processor configured to implement a virtual network manager having a network interface coupled to a virtual network,
75[b]	the virtual network manager including at least one virtual community definition that is defined by a domain name having an associated public network address and a user set of one or more devices that are registered in the virtual network,
75[c]	each device in the virtual network being identified to the other devices by a virtual network address that is associated with the device,
75[d]	the virtual network manager configured to exchange virtual network information with the one or more devices of the user set, the virtual network being accessible by devices in the user set and devices outside of the user set, and
75[e]	the virtual network manager further configured to receive a DNS request from a source device, and return a public network address of a route director, a private network address for a destination device, and a virtual network address that corresponds to the destination device.
Claim 77	
77	The system of claim 75 wherein the virtual network manager includes a member join module.
Claim 78	
78	The system of claim 77 wherein the member join module provides a virtual network address to a device that is registered as a member of the network.

I. INTRODUCTION

American Airlines, Inc. and Southwest Airlines Co. ("Petitioners") request *inter partes* review ("IPR") of claims 1, 30, 35–38, 48, 62, 75, and 77–78 (the "challenged claims")[1] of U.S. Patent No. 7,949,785 titled "Secure Virtual Community Network System" (the "'785 Patent," Ex-1001).

II. GROUNDS FOR STANDING

Petitioners certify the '785 Patent is available for IPR and Petitioners aren't barred or estopped from requesting IPR on the grounds herein. Petitioners filed this Petition within one-year of service on Intellectual Ventures I of Patent Owner's district court complaint alleging infringement of the '785 Patent. Ex-1007; Ex-1008.

III. IDENTIFICATION OF CHALLENGED GROUNDS

Petitioners request IPR and cancellation of the challenged claims of the '785 Patent on the following grounds:[2]

Ground	Challenged Claims	Basis for Rejection
1	1, 30, 35–38, 48, 62, 75, and 77–78	§ 103: Caronni-I (US 6,970,941B1) in view of Caronni-II (US 7,814,228B2) and in further view of Hipp (US 6,766,371B1)

[1] Petitioners recognize the '785 Patent includes 90 claims; however, this Petition focuses on a smaller subset of claims based on Patent Owner's preliminary Infringement Contentions. Ex-1025; Ex-1026.

[2] All references to 35 U.S.C. §§ 102, 103 are to the pre-AIA statutory framework.

2	1, 30, 35–38, 48, 62, 75, and 77–78	§ 103: Caronni-I in view of Caronni-II and in further view of RFC-1383

The ’785 Patent application (U.S. Patent App. No. 10/403,818) was filed on, and claims priority to, March 31, 2003. Ex-1001, 1. Every prior art reference in the above grounds precedes the ’785 Patent’s priority date.

Reference	Prior Art Date	Status
Caronni-I (Ex-1003)	Priority: December 10, 1999 Filed: December 10, 1999 Published: November 29, 2005	§ 102(e)
Caronni-II (Ex-1004)	Priority: February 13, 2003 Filed: February 13, 2003 Published: August 19, 2004	§ 102(e)
Hipp (Ex-1005)	Priority: October 5, 1999 Filed: October 5, 2000 Published: July 20, 2004	§ 102(e)
RFC-1383 (Ex-1006)	Published: December 1992	§ 102(b)

IV. ’785 PATENT

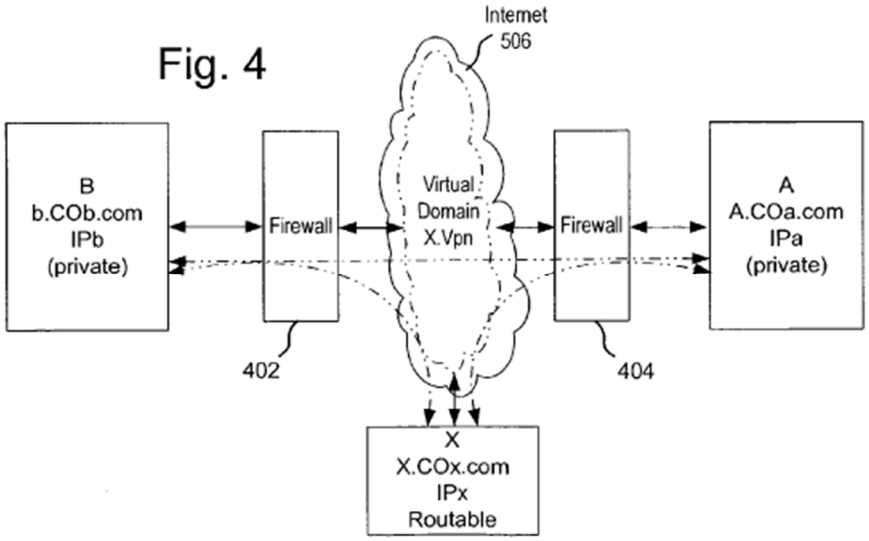
A. Overview

The ’785 Patent claims a “virtual network manager [or system]” comprising four key components. First, it includes a computing device or “network interface configured for data communication.” Ex-1001, 36:37-56. Second, there’s a “register module configured to register devices in a virtual network.” *Id.* Third, the system includes a router or route director. *Id.* The fourth component is a “DNS server configured to receive a DNS request” from a first (source) device and return three addresses to the source, including, for example, “[(i)] a network address associated

with a network route director, [(ii)] a private network address associated with a second [destination] device in the virtual network, and [(iii)] a virtual network address associated with the second [destination] device." *Id.*

The '785 Patent addresses two primary challenges in Internet communication: the IP address depletion problem and the mobility problem. The IP address depletion issue arises from the Internet's popularity, which has led to a shortage of available IP addresses for assigning to devices, particularly limiting communication for devices in private networks. *Id.*, 1:50–55. The "mobility problem" refers to the difficulty in tracking devices that are assigned constantly changing or dynamic IP addresses, complicating communication attempts from other devices on the Internet. *Id.*, 5:10–45. To address these issues, the '785 Patent describes "a need for a system that provides for local and remote entities to communicate and collaborate using the Internet, [a system that] can work with existing NAT devices and firewalls, and allows for devices to move to different physical networks." *Id.*, 2:45–48.

One embodiment discloses a "secure Virtual Community Network or ('VCN')," described as "a private dynamic network which acts as a private LAN for computing devices coupled to public networks or private networks." *Id.*, 8:66–9:3.



Id., Figure 4.

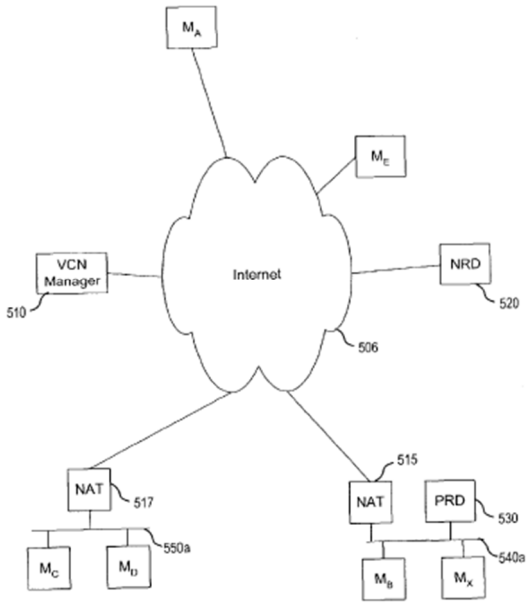
Figure 4 of the '785 Patent illustrates an exemplary Virtual Community Network (VCN), demonstrating the system's capability to connect devices across different network environments. The VCN includes two computers or devices in separate private domains: "computer or device B (host name—b.Cob.com) in a first private domain and computer or device A (host name—a.Coa.com) in a second private domain, both of which are coupled to the Internet by firewall devices 402, 404." *Id.*, 9:9–13. The firewall devices are Network Address Translation (NAT) devices. *Id.*, 9:13–15. Additionally, the VNC incorporates a third "[c]omputer or device X[,]" which "is coupled directly to the Internet and has a public IP address." *Id.*, 9:16–17. The '785 Patent emphasizes the flexibility of this setup, noting that "[m]achines A, B, and X can join the VCN, leave the VCN, or allows other machines in the VCN to communicate with them." *Id.*, 9:18–23. A key feature of this VCN is

that it enables all member devices to interact as if they were part of a single physical local network, facilitating direct communication between devices within the virtual domain, regardless of their actual physical network location. *Id.*, 9:22–27.

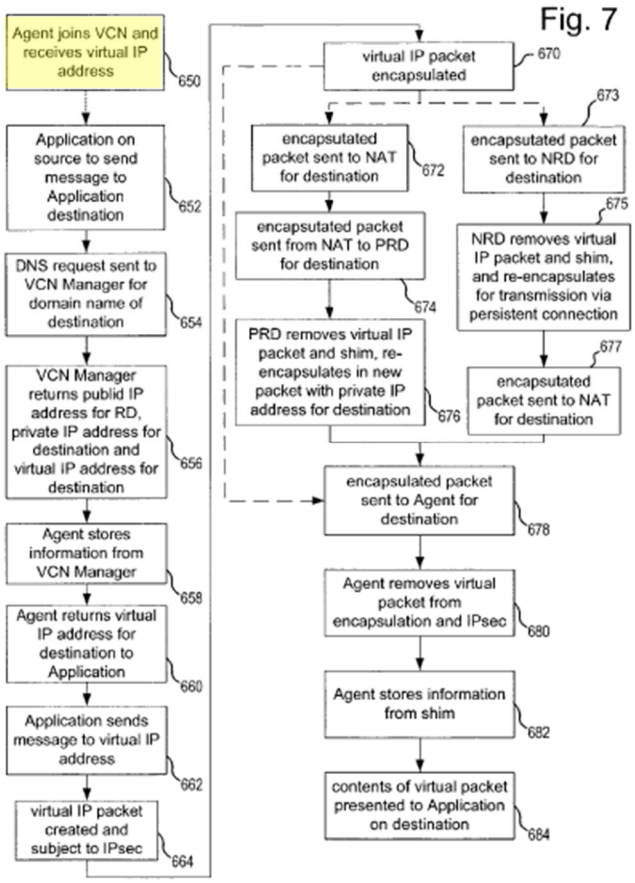
The ’785 Patent indicates “hardware architecture for the machines, server or other devices used to implement the present invention should be well understood to one of average skill in the art.” *Id.*, 9:42–44. Further, the ’785 Patent describes a standard configuration of hardware/software components that are readily available, including “one or more processors, a memory, a mass storage device, a portable storage device, a first network interface, a second network interface, and I/O devices, in communication with each other.” *Id.*, 9:45–48, 9:48–54. The ’785 Patent also specifies that the network interface can either include or be connected to a firewall. *Id.*, 9:57–60. Regarding the network interfaces, the ’785 Patent clarifies that they are internal components of the computer. In cases where the computer also functions as a “router,” it typically “includes two or more network interfaces.” *Id.*, 9:54–55. However, the ’785 Patent acknowledges that “[i]n other embodiments, the computer may include only one network interface.” *Id.*, 9:55–56.

Another example of the VCN system is illustrated in Figure 5A below.

Fig. 5a



Id., Figure 5A. This configuration comprises several key components: a VCN Manager 510, a Network Route Director 520 and/or Private Route Director 530, and multiple user devices (e.g., M_C, M_X, M_A). *Id.*, 10:24–27. "The VCN Manager 510 is a central server or server cluster providing management services, connection services and security services for the VCN." *Id.*, 10:66–11:1. The "various user machines M_n [are] coupled to the Internet 506, as well as other devices." *Id.*, 10:17–19. "In order to establish or participate in communication within the VCN, a device must register with and join the VCN, thereby becoming a member of the VCN." *Id.*, 14:11–14. To join, "[t]he Agent will send a message to [the] VCN Manager 510 and receive a virtual IP address [in return]." *Id.*, 14:14–15. This is shown in step 650 of Figure 7. *Id.*, 14:17–18.



Id., Figure 7 (annotated).

The ’785 Patent describes the virtual IP or virtual network address as a crucial component of the system, designed to uniquely identify devices within the virtual network. This address may be another private network address, selected from a well-defined set of addresses (virtual address realm) to ensure uniqueness across the entire virtual network. *Id.*, 12:48–53. The ’785 Patent provides flexibility in the format of these virtual network addresses, disclosing that they can be represented as a string or comply with either IPV4 or IPV6 standards. *Id.*, 12:40–42 (“Assigning a virtual

address to each peer designated by a DNS string solves the problem of ambiguous endpoints. These virtual addresses may be any legal IPv4 addresses[.]").

As illustrated in Figure 7, when a new member device wants to send a message to another device in the VCN, it begins by "initiat[ing] a DNS request," which is then sent to the VCN Manager 510, as depicted in step 654. *Id.*, 14:32–34. In response to this request, the "VCN Manager 510 returns the public address of the Route Director for the destination, the private address for the destination device and a virtual IP address for the destination device," as shown in step 656. *Id.*, 14:34–38. The Agent for the source device then stores this information in a table or another suitable data structure, which is used to create a virtual IP packet for forwarding to the destination address. *Id.*, 14:48–15:15. For subsequent communications between these devices (i.e., when the destination device wishes to respond to the source or the source device wishes to send additional information), the process of Figure 7 can be repeated. However, the DNS request steps don't need to be performed again since the necessary information has already been stored by both devices. *Id.*, 15:58–67.

B. Prosecution History

The application for the '785 Patent was filed on March 31, 2003, containing 109 claims. Throughout prosecution, the Examiner issued multiple rejections, citing grounds under 35 U.S.C. §§ 101, 102, and 103, based on several prior art references.

Ex-1002, 143 (Non-Final Office Action), 202 (Final Office Action), 391 (Non-Final Office Action), 613 (Final Office Action), 687 (Non-Final Office Action).

In the Examiner's first Non-Final Office Action, the Examiner rejected all claims as anticipated by U.S. Patent Pub. No. 2003/0041136 (Cheline). *Id.*, 143–162. Trying to overcome this rejection, the applicant amended the claims to add language clarifying that the network traffic router "uses router data to route traffic between members of the virtual community;" further, applicant incorporated details about the communication protocol, requiring that "a first member of the virtual community communicates with another member in the virtual community by querying the DNS server to determine a virtual network address and a private network address for the other member, and to determine a public address of the network traffic router." *Id.*, 172–187.

Specifically, applicant argued "the client 102(1) of Cheline does not communicate with the DNS server 120 to obtain a virtual network address and private network address of another client, such as client 102(2), or to obtain a public network address of the service provider system 146, which is asserted by the Examiner to be a network traffic router as claimed." *Id.*, 191.

In the second Final Office Action, the Examiner granted allowance to a portion of the claims, while rejecting pending claims 53–66 and 68–92 as anticipated by U.S. Patent No. 6,631,416 (Bendinelli). *Id.*, 613–618. The Examiner's rationale

for allowance centered on the absence of the prior art teaching "returning a public network address of a route director, a private network address for a destination device, and a virtual network address corresponding to the destination device in response to a DNS request in the context of claims 1, 32, and 93," and that "[s]pecifically, no examples were found in the prior art of record in which both a private address and a virtual address of a destination device would be returned." *Id.*, 616–617. Applicant subsequently filed an amendment on October 21, 2010, amending rejected claims 53–55, 64, and 80 to include a similar DNS request and corresponding response with a triplet of addresses. *Id.*, 658–683.

The Examiner issued another Office Action and rejected claims 1–15, 17–24, 26–36, 41–43, and 80–92 as invalid under § 101, while deeming the remaining claims allowable. *Id.*, 687–691. Applicant traversed the § 101 rejection and amended the claims to clarify that the recited virtual network manager and/or route director are implemented using a memory and a processor of a computer device. *Id.*, 723–750. Following this amendment, the United States Patent and Trademark Office (USPTO) issued a notice of allowance on January 14, 2011. *Id.*, 760–766.

C. Person of Ordinary Skill in the Art

The '785 Patent relates to virtual networking. Ex-1001, Abstract. A person of ordinary skill in the art ("POSITA") relevant to this patent at the time of its invention would've had a bachelor's degree in computer science, computer engineering, or a

related technical field, combined with approximately two years of practical experience in fields of networking such as network virtualization, security, or management. Additional professional experience might substitute for less education and vice versa. Ex-1011, ¶¶ 33–35.

V. CLAIM CONSTRUCTION

Claims should be construed "in accordance with the ordinary and customary meaning of such claim as understood by one of ordinary skill in the art and the prosecution history pertaining to the patent." 37 C.F.R. § 42.100(b); *see Phillips v. AWH Corp.*, 415 F.3d 1303 (Fed. Cir. 2005) (en banc). Petitioners are unaware of any "prior claim construction determination" related to the '785 Patent. 37 C.F.R. § 42.100(b). Claim construction is only required where necessary to resolve a dispute. *See, e.g., Toyota Motor Corp. v. Cellport Sys., Inc.*, IPR2015-00633, Paper 11 at 16 (P.T.A.B. Aug. 14, 2015) (citing *Wellman, Inc. v. Eastman Chem. Co.*, 642 F.3d 1355, 1361 (Fed. Cir. 2011)). Petitioners propose the following claim constructions.

A. All Claims: "virtual network address"

All the claims of the '785 Patent recite a "virtual network address" that is assigned to a device once it is registered in the virtual network. The '785 Patent explains that "[b]y having each member of a VCN use a virtual [network] address, a virtual address realm is created. The virtual address realm is the set of addresses that can be used to identify and send communications to other members of the VCN."

Ex-1001, 12:54–57. Therefore, the "virtual network address" is a name or address that can be used to identify and send communications to other devices in the virtual network.

B. Claims 30: "register module"

Claim 30 recites a "register module" for performing particular functions. The word "module" is a nonce word that substitutes for "means" for purposes of § 112, ¶ 6. *See Williamson v. Citrix Online, LLC*, 792 F.3d 1339, 1350 (Fed. Cir. 2015) (en banc).

30[b]: a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to: receive a registration request from an agent associated with a device; distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Nonce word for "means" function without recitation of structure to perform it

Means-plus-function claims are construed by identifying the claimed function and corresponding structure in the specification for performing it. *See Williamson*, 792 F.3d at 1351. The chart below provides these constructions. *See* 3 C.F.R. § 42.104(b)(3).

Claims	Function	Structure in '785 Patent Specification
30	"receiv[ing] registration request from an agent associated with a device;	"The Registration exchange between the member agent and the VCN Manager is shown in FIG. 12A. When a Member Agent

	distribut[ing] a virtual network address to the device when the device is registered in the virtual network"	565 attempts to register at step 1200, Member Agent 565 will send a registration request packet 1202 to the VCN Manager 510 to start registration. The packet is included in an HTTP wrapper as illustrated at 1204. The registration packet will carry the client Fully Qualified Domain Name (FQDN) and a Diffe-Hellman Key Exchange request to the VCN Manager. The packet further includes packet version information, type and length information, a user authenticator, the length of FQDN in octets, a VCN name offset, the member's FQDN in DNS format, the length of Diffe-Hellman value in octets, and the member's initial Diffe-Hellman value." Ex-1001, 18:17–29.

C. Claims 35–37, 77–78: "join module"

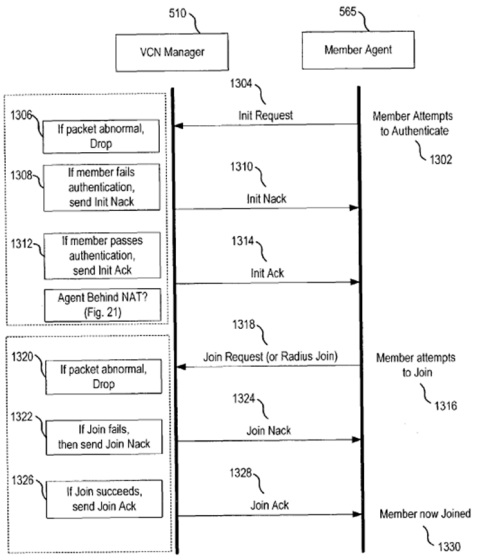
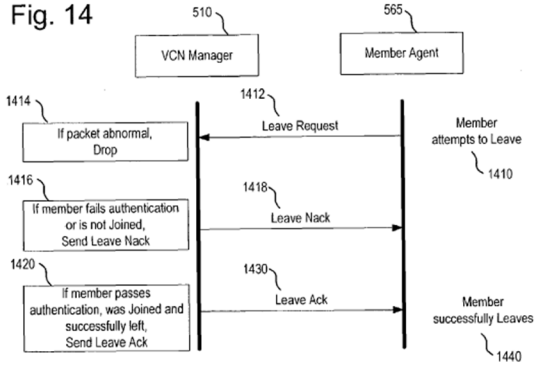
Claims 35–37 and 77–78 recite a "join module" performing the below described functions. *See Williamson*, 792 F.3d at 1350 ("'Module' is a well-known nonce word that can operate as a substitute for 'means' in the context of § 112, para. 6."). Claim set 35–37 and 77–78 below is illustrative.

35: The virtual network manager of claim 30 further comprising a join module configured to receive a join request from the agent associated with the device to indicate that the device is connected for data communication within the virtual network, the join module further configured to receive a leave request from the agent associated with the device to indicate that the device will be disconnected from data communication within the virtual network.	Nonce word for "means" function without recitation of structure to perform it
36: The virtual network manager of claim 35 wherein the join module is further configured	

to provide virtual network addresses to the devices that are registered in the virtual network.	
37: The virtual network manager of claim 35 wherein the join module is further configured to maintain data to associate a virtual network address with a device in the virtual network.	
77: The system of claim 75 wherein the virtual network manager includes a member join module.	
78: The system of claim 77 wherein the member join module provides a virtual network address to a device that is registered as a member of the network.	

Means-plus-function claims are construed by identifying the claimed function and corresponding structure in the specification for performing it. *See Williamson*, 792 F.3d at 1351. The chart below provides these constructions. *See* 3 C.F.R. § 42.104(b)(3).

Claims	Function	Structure in '785 Patent Specification
35	"receiv[ing] [] join [and leave] request[s] from the agent associated with the device [] indicat[ing] that the device is connected [or will be disconnected] from data communication[, respectively"	The member join process is depicted in Figure 13 and the leave process is depicted in Figure 14.

		Fig. 13  Fig. 14  Ex-1001, 20–21 (describing Figures 13 and 14).
36, 78	"provid[ing] virtual network addresses to the devices that are registered in the virtual network"	*Id.*
37	"maintain[ing] data to associate a virtual network address with a device in the virtual network"	*Id.*
77	*N/A*	

D. Other Claims

For the remaining claims and claim terms, no construction is necessary to resolve any disputes identified in this Petition.[3] *Vivid Techs., Inc. v. Am. Sci. & Eng'g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999).

VI. PRIOR ART

A. Caronni-I

Caronni-I, titled "System and Method for Separating Addresses from the Delivery Scheme in a Virtual Private Network," describes "data processing systems and, more particularly, [] a private network using a public-network infrastructure." Ex-1003, 1:57–59. Caronni-I establishes a "'Supernet,' which is a private network that uses components from a public-network infrastructure." *Id.*, 4:38–39. The Supernet allows organizations to utilize public networks for their enterprise networks, reducing the burden on maintaining network infrastructure. *Id.*, 4:40–43.

[3] If trial is instituted, Petitioners reserve the right to address construction of any terms raised at trial, or in the related district court action.

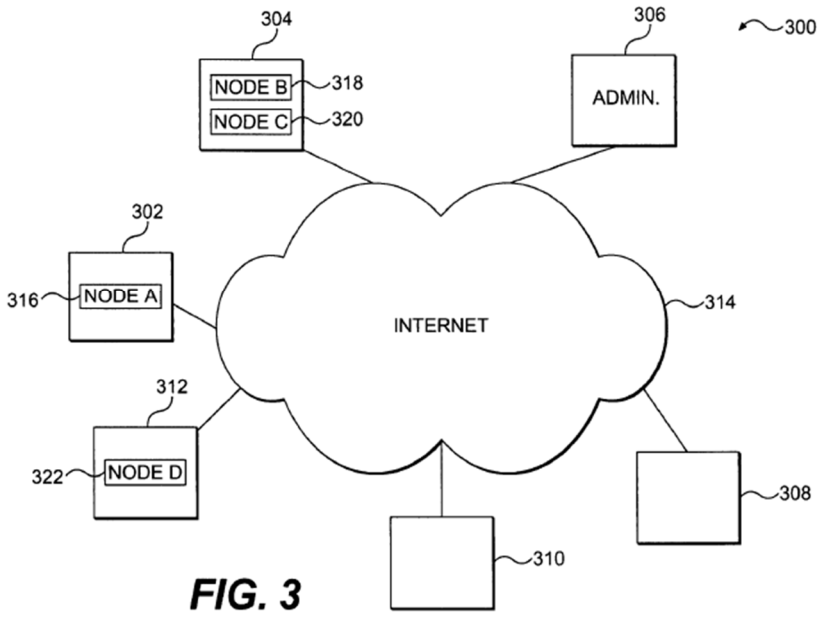


FIG. 3

Id., Figure 3.

Caronni-I's system uses a separate layer that isolates address names of nodes from addressing and delivery schemes. *Id.*, 4:55–59 ("The Supernet contains a modification to the IP packet format that can be used to separate network behavior from addressing. As a result of the modification, any delivery scheme may be assigned to any address, or group of addresses."). This separation provides flexibility in addressing and delivery methods.

Caronni-I describes several features of the Supernet, including secure communication between nodes, address translation, and operating system-level enforcement of node compartmentalization. *Id.*, 3:23–26, 6:6–7, 6:14–16, 6:26–30.

Caronni further discloses channels for communication and resource sharing between nodes. *Id.*, 5:27–30.

Caronni-I's system is implemented by various components, including the Supernet Authentication and Session Daemon (SASD) 540, Virtual Address Resolution Protocol Daemon (VARPD) 548, Supernet Security Layer (SNSL) 542, and Key Management Server (KMS) 550. These components, located on the administrative machine 306 and devices 302–312, work together to provide authentication, address translation, and key management for the Supernet.

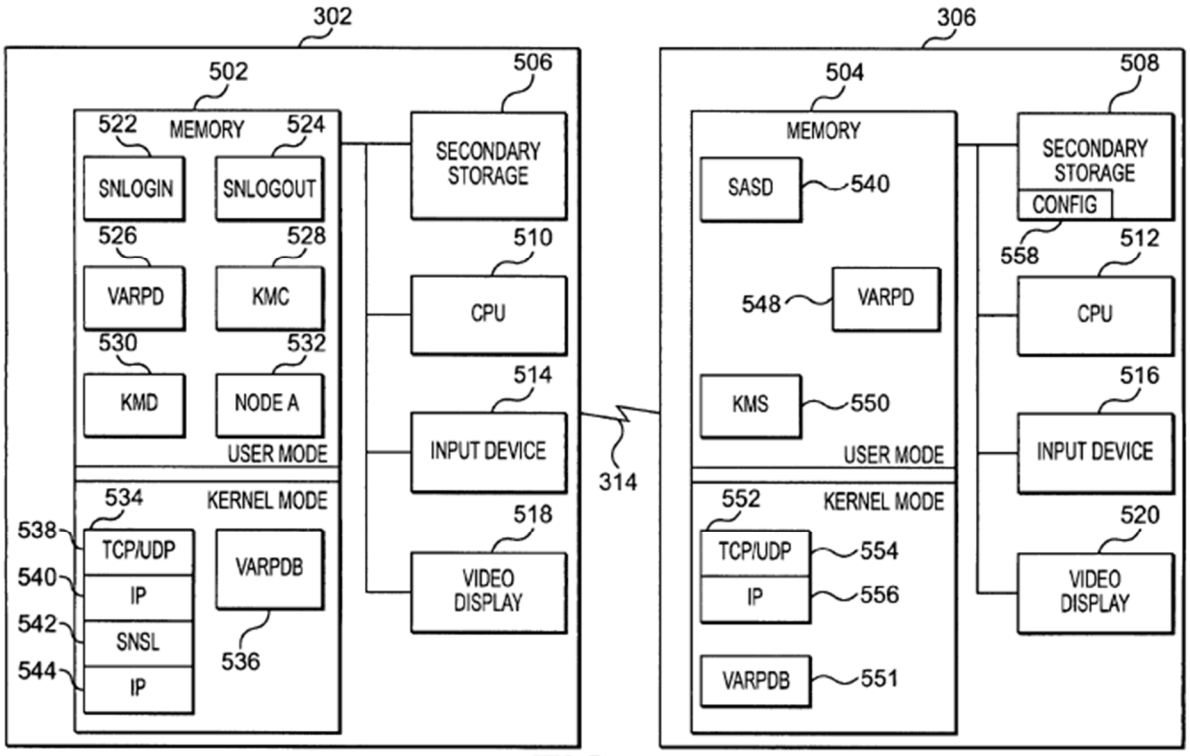


FIG. 5

Id., Figure 5.

The SASD represents the Supernet. *Id.*, 7:3–5. When a node attempts to join a Supernet through the SNlogin Script, SASD validates and authenticates the request and then "creates an address mapping between a node ID and the real address." *Id.*, 9:67–10:18 (SNlogin script "performs a handshaking with SASD to authenticate this information.").[4] This mapping is crucial for translating between the public IP addresses and used on the underlying network and the virtual IP addresses used within the Supernet. *Id.*, 6:20–24.

The SNSL acts as "the conduit for all Supernet communications" and performs critical functions such as address translation, encryption, and authentication. *Id.*, 8:61–64. The SNSL works in conjunction with other components such as the SASD to enable secure communication. *Id.*, 10:45–50.

Caronni-I also describes how packets are encapsulated to enable secure transmission:

> When encrypting the packet, the virtual source node address 642, the virtual destination node address 644, and the data may be encrypted (addressing section 660), but the source and destination real addresses 614, 616 (delivery scheme section 670) are not, so that the real addresses can be used by the public network infrastructure to send the packet to its destination.

[4] Conversely, the SNlogout script handles the process of disconnecting a node from the Supernet. Ex-1003, 12:61–13:5.

Id., 12:10–16. The encapsulation process allows the packet to be routed normally on the public network while keeping the Supernet addressing information confidential.

The encapsulation process involves multiple steps as a packet travels from a source to destination. First, "the SNSL layer [] accesses the VARPDB to obtain the address mapping between virtual source node address 642 and the source real address 614 as well as the virtual destination node address 644 and the destination real address 616." *Id.*, 11:49–53. Next, it "encrypts the packet using the appropriate encryption algorithm and the key previously obtained." *Id.*, 12:8–10. Finally, the SNSL "authenticates the sender to verify that it is the bona fide sender and that the packet was not modified in transit" before passing it to the outer IP layer for transmission. *Id.*, 12:20–22. This encapsulation and security scheme allows the Supernet to provide "heterogeneous addressing functionality" by separating "address names of nodes from addressing schemes and delivery schemes." *Id.*, 4:52–55.

Further, Caronni-I discloses DNS type requests through the operation of the VARP address translation or resolution process:

> [A] VARPD may act as a server by storing all address mappings for a particular Supernet into its associated VARPDB. Second, regardless of its role as a server or not, each VARPD assists in *address translation* for the nodes on its machine. In this role, the VARPD stores into its associated VARPDB the address mappings for its nodes.

Id., 7:22–28 (emphasis added); *see also id.*, 9:47–54.

B. Caronni-II

Caronni-II, titled "System and Method for Using Data Encapsulation in a Virtual Network," describes "a virtual computer network and more particularly to the addressing of messages sent in a virtual network using data encapsulated in higher level protocols." Ex-1004, 1:6–9. Caronni-II introduces an extended virtual address registration process, including higher-level protocol designations, allowing messages to bypass filtering mechanisms. *Id.*, 2:40–44.

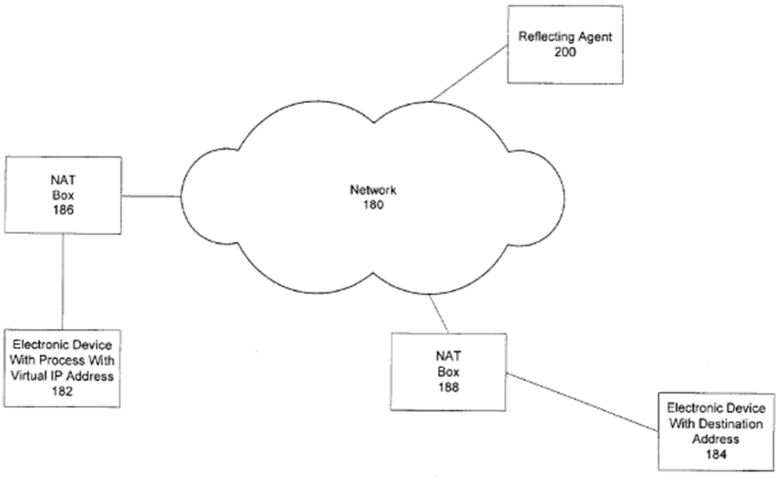
Caronni-II outlines a virtual network overlay on a physical network, where "[a] virtual network is a logical network overlaid on a physical network." *Id.*, 1:13–14. Caronni-II utilizes a virtual address resolution facility that "works with stored virtual address registrations for the virtual network." *Id.*, 4:9–10. These registrations are stored, "such as by storing them in a Virtual Address Resolution Protocol (VARP) lookup table 26." *Id.*, 4:12–14.

The encapsulation of messages in higher-level protocols as specified in the virtual address registration is a feature of Caronni-II. *Id.*, 2:40–44.

Caronni-II's system also provides a method for communication between virtual addresses behind different NAT boxes by introducing "a third-party reflecting agent located at an address outside the NAT box." *Id.*, 8:20–21. The reflecting agent acts as an intermediary, allowing connections between virtual

addresses that would otherwise be unreachable due to NAT restrictions. *Id.*, 8:35–38.

Figure 5



Id., Figure 5.

Caronni-II's system additionally includes DNS type requests through the operation of the VARP address translation process:

> A *resolution request is received* referencing a virtual address destination and the resolution request is resolved using the virtual address resolution facility and the stored associations.

Ex-1004, 3:20–23 (emphasis added); see also *id.*, 5:10–22 ("Each entry in the VARP lookup table …."), Figure 2.

C. Hipp

Hipp, titled "Virtual Network Environment" (VNE), describes "[a] virtual network environment to be used by a set of applications for the express purpose of isolating the applications from other applications on the same node or network[.]"

Ex-1005, Abstract. The VNE is defined as "a collection of IP addresses related to applications that are contained within the[] VNE or have the potential of being placed in the VNE." *Id.*, 2:66–3:2. The system assigns a "unique virtual IP address and virtual hostname to a group of processes that make up the application instance which the instance keeps throughout its execution." *Id.*, 6:1–4. This virtual network identity (VNI) remains with the application regardless of which node it runs on, allowing for transparent movement across machines without modifying the application. *Id.*, 6:4–6.

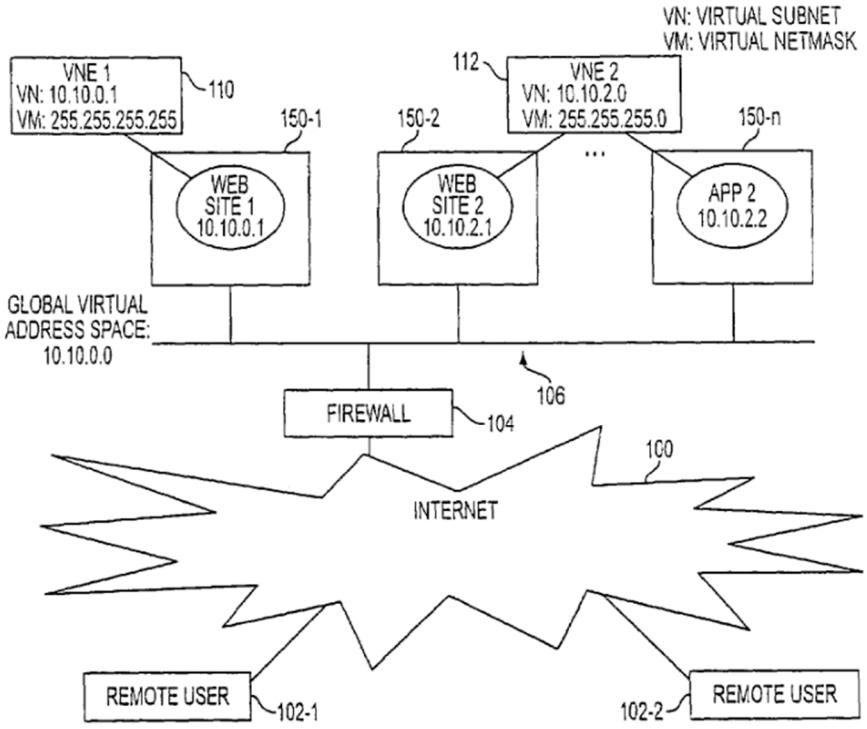


FIG. 1

Id., Figure 1.

Hipp outlines a method where applications are assigned virtual IP addresses and grouped into discrete virtual environments based on these addresses. *Id.*, cl. 1. Applications within the same VNE can communicate with each other, but are prevented from communicating with applications in other VNEs. *Id.*, 3:5–9 ("An application running within one VNE can communicate with another application in the same VNE. However, an application in one VNE cannot communicate with an application in another VNE unless expressly permitted.").

Further, Hipp discloses a domain name server (DNS) in the context of hostname resolution. *Id.*, 6:16–25. "The virtual hostname resolves to the virtual IP address for both the applications registered with the VNI framework as well as those that are not registered. This may require configuration of a name service or OS host configuration files." *Id.*, 6:16–20. "For example, if an application instance used a virtual IP address of 10.10.0.1 and a virtual hostname of host1055, the standard hostname to IP address resolution mechanisms (e.g. DNS or the /etc/hosts file) would have to be preconfigured to resolve a query of host1055 to IP address 10.10.0.1." *Id.*, 6:20–25.

D. RFC-1383

Request For Comment 1383: An Experiment in DNS-Based IP Routing outlines a proposal for utilizing DNS records for IP routing. RFC-1383 seeks to solve

the issue of "routing explosion" on the Internet caused by the increasing number of networks. Ex-1006, 1–2.

RFC-1383 discloses "a scheme that allows for simple routing" that is "complementary with the classic hierarchical routing approach," while providing "an easy to implement and low cost solution for 'multi-homed' domains." *Id.*, 2.

RFC-1383 introduces "RX records" for routing IP packets by associating domain names with preferred gateways by means of "source routing" or "tunneling," or another form of encapsulation protocol. *Id.*, 4. To implement this RX routing, a modified DNS lookup is suggested: the source or an early relay makes the routing decision by querying the DNS for RX records, which list gateways with preference levels. *Id.*, 6, 9. This approach avoids hop-by-hop routing updates across the network and is scalable. *Id.*, 4.

RFC-1383 describes a network interface integrated with a standard IP router, working alongside a DNS query manager. *Id.*, 12–13. When an Internet Control Message Protocol (ICMP) message is received, the query manager updates the local routing table to ensure that any new packets bound for the specified destination are routed through the real-time forwarder. *Id.* Simultaneously, the query manager sends a DNS request to read the RX records for the destination. *Id.* Upon receiving the response, the query manager selects a gateway and provides this information to the real-time forwarder. *Id.*, 11–12.

When the real-time forwarder receives a packet, it checks if a gateway for the destination is available. *Id.*, 12–13. If so, it inserts the necessary source routing information and forwards the packet. *Id.*, 12. Notably, RFC-1383 discloses a way to store and return multiple IP addresses in response to a single query and to identify an address of the destination in the domain part of the DNS request and response. RFC-1383 proposes using "TXT" DNS records to store that information:

> This [TXT] record is designed for easy general purpose extensions in the DNS, and its content is a text string. Each RX record will contain three fields:
>
> - A record identifier, "RX," to distinguish it from other experimental uses of the "TXT" record.
> - A cost indicator, encoded on up to 3 numerical digits.
> - An IP address, encoded as a text string following the 'dot' notation.

Id., 11.

The TXT records described in RFC-1383 are simply a domain field containing a destination address, and a string value comprising comma-delimited fields, including an IP address.

E. Motivation to Combine Caronni-I, Caronni-II, and Hipp

The combined teachings of Caronni-I, Caronni-II, and Hipp create a unified virtual networking system with enhanced addressing and resolution capabilities. A

POSITA would've been motivated to combine Caronni-I, Caronni-II, and Hipp for at least the following reasons. Ex-1011, ¶ 346.

A POSITA would recognize that Caronni-I provides foundational architecture for scalable virtual network environments. The Supernet framework establishes a logical overlay network using administrative nodes for centralized management and regular nodes for endpoint operations. Ex-1003, 4:53–55, 5:11–13 ("The Supernet also includes an administrative node 306 to administer to the needs of the Supernet"). This structure enables virtual addressing (VARPD) through encapsulated packets in an inner and outer IP layer. *Id.*, 7:5–9 ("VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as a node IDs, to the network addresses recognized by the public-network infrastructure, known as the real addresses."). The inner layer includes a header containing the source and destination *virtual* IP addresses. *Id.*, 11:26–30. The outer IP layer performs further encapsulation and includes a header containing the source and destination *real* IP addresses. *Id.*, 9:13–15. Caronni-I further discloses encryption of the encapsulated packets to ensure secure communication between nodes. *Id.*, 12:8–19. The registration process in Caronni-I, where nodes authenticate with administrative nodes (*Id.*, 7:3–5, 10:13–18), creates a scalable framework for dynamic membership management. Ex-1011, ¶¶ 347–355.

Caronni-I also refers to a domain name server (DNS). Caronni-I explains: "In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, *obtains the real address of the SNlogin script by querying network services in a well-known manner* and then registers this information with the VARPD that acts as the server for this Supernet." Ex-1003, 10:13–18 (emphasis added). A POSITA would understand DNS is a well-known network service used to query for real addresses. Ex-1011, ¶¶ 357–363.

Having the same named inventor, Caronni-II naturally extends this architecture by introducing a reflecting agent to manage NAT traversal between private networks. Ex-1004, 8:19–23. While Caronni-II operates at a higher abstraction level than Caronni-I, its disclosure of analogous components—DNS type request, virtual addressing, encapsulation, and node registration—provides overlapping technical objectives. *Id.*, 3:20-23, 3:49–51, 4:11–14. A POSITA would recognize that integrating Caronni-II's reflecting agent with Caronni-I's Supernet structure directly addresses NAT-to-NAT communication challenges. The reflecting agent's ability to map public-facing IP addresses to private virtual addresses aligns with Caronni-I's VARPD system, enabling bidirectional connectivity across NAT boundaries. *Id.*, Figure 5. The Caronni-I/II combination results in the inclusion of the reflecting agent's IP address in Caronni-I's VARPD system. *Id.*, 8:35–38. This combination enhances Caronni-I's virtual addressing scheme by adding NAT

transparency through Caronni-II's reflecting agent, fulfilling a well-documented industry need for scalable hybrid networks. Ex-1011, ¶ 403.

Hipp's DNS-based hostname resolution complements these frameworks by providing DNS resolution of virtual hostnames to virtual IP addresses. Ex-1005, 6:15–26. By incorporating Hipp's DNS methodology into Caronni-I's VARPD operations and Caronni-II's VARP reflecting agent entries, a POSITA would appreciate a DNS capability to resolve a virtual hostname into a virtual IP address, a real IP address and an IP address of a reflecting agent. This integration would leverage Hipp's DNS capability for virtual hostnames to return multiple address records from a single query, aligning with the dual addressing requirements of hybrid virtual/physical networks. The combination creates a cohesive system where DNS resolution triggers context-aware routing decisions—using virtual addresses for internal Supernet traffic and NAT-translated addresses for cross-boundary communication. Ex-1011, ¶¶ 359, 367.

The reasonable expectation of success of this combination stems from the complementary nature of the patents: (1) Caronni-I's structural blueprint and VARP practices, (2) Caronni-II's NAT traversal solution, and (3) Hipp's DNS resolution mechanics. Without extensive experimentation nor undue burden, a POSITA would recognize that DNS enhancements (Hipp) logically apply to virtual addressing systems (Caronni-I/II) given industry-standard practices of coupling name

resolution with routing, while reflecting agents (Caronni-II) provide a well-understood method for bridging NAT-dependent environments. *Id.*, ¶¶ 368–369.

F. Motivation to Combine Caronni-I, Caronni-II, and RFC-1383

The Caronni-I/II and RFC-1383 combination establishes a cohesive virtual networking environment with enhanced addressing and resolution capabilities. A POSITA would have been motivated to combine Caronni-I/II and RFC-1383 for at least the following reasons. Ex-1011, ¶ 397.

Petitioners incorporate by reference Section VI.E. above (regarding combining Caronni-I/II). The Caronni-I/II combination relies on established virtual-to-real address translation principles using VARP resolution. A POSITA would understand the use of well-known DNS practices and would further recognize RFC-1383 as extending these frameworks through a modified DNS functionality to virtual hostnames and multiple real IP addresses returned by a DNS service. *Id.*, ¶¶ 408, 417.

RFC-1383 discloses a way to store and return multiple IP addresses in response to a single query using a text record and identifies an address of the destination in the domain part of the DNS request and response. Ex-1006, 11 (“This [TXT] record is designed for easy general purpose extensions in the DNS, and its content is a text string.”). A POSITA would’ve appreciated that DNS TXT records inherently structure data through a domain field specifying the destination address

and a string value field supporting comma-delimited entries, including an IP address. Further, a POSITA would've understood that when RFC-1383 returns one or more extra addresses in response to a DNS query, those could be easily used as the tunneling/encapsulation addresses to enable, for example, VPN and IPsec services, like the tunneling and encapsulation of the Caronni-I/II combination. A POSITA would have a reasonable expectation of success in using such TXT records to return any number of comma-delimited IP addresses, values, and any information disclosed in the Caronni-I/II combination. Ex-1011, ¶¶ 418–425.

This modification would've been straightforward, requiring no extensive experimentation or imposing undue difficulty, as DNS is a well-established, standardized, and widely utilized system. Furthermore, a POSITA would've had a high likelihood of success since RFC-1383 simply elaborates on one method for returning IP addresses among the several approaches described by Caronni-I/II. *Id.*, ¶ 426.

VII. GROUND 1: Claims 1, 30, 35–38, 48, 62, 75, and 77–78 are obvious in view of Caronni-I in combination with Caronni-II and Hipp

A. The Independent Claims

The independent claims of the '785 Patent (claims 1, 30, 38, 48, 62, and 75) recite substantially overlapping subject matter. For ease of discussion, Petitioners address the independent claims together.

These claims generally disclose a virtual network system or manager, comprising a virtual network with a public domain name, a registration feature or module to register devices on the network, a route director to route data traffic between the devices on the network and a DNS server to exchange network addresses of devices on the network upon request. These limitations, as shown further below, would've been obvious to a POSITA before the time of the filing of the '785 Patent, in light of Caronni-I combined with Caronni-II and Hipp. Ex-1011, ¶ 208.

1. Preamble: virtual network system/manager

Claim	Limitations
1[pre]	**A virtual network system**, comprising:
30[pre]	**A virtual network manager**, comprising:
38[pre]	**A virtual network system**, comprising:
48[pre]	A computer-implemented method, comprising:
62[pre]	One or more processor readable storage media devices comprising process readable code that, if executed by a computer device, implements **a virtual network manager**:
75[pre]	**A virtual network system**, comprising:

To the extent the preambles are limiting, Caronni-I discloses a Supernet, which is a virtual network system designed to manage communications between "a public network having a network infrastructure that is used by a private network over which a plurality of nodes communicate[.]" Ex-1003, cl. 1; Ex-1011, ¶ 210. Caronni-

I further discloses a virtual network manager as an admin node[5] to "to administer to the needs of the Supernet." Ex-1003, 5:11–13; Ex-1011, ¶ 212.

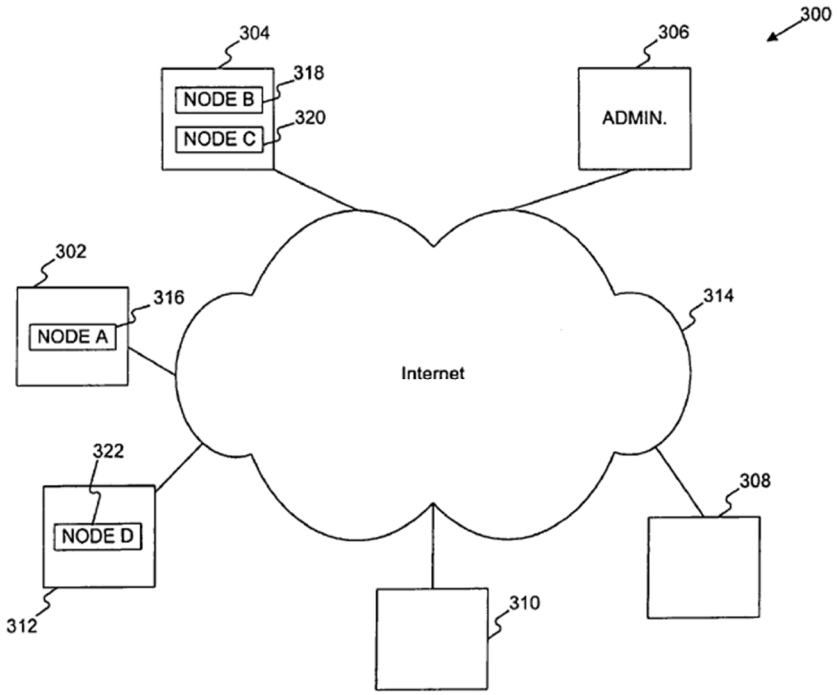


Fig. 3

Ex-1003, Figure 3.

2. A virtual network defined by a domain name having an associated public network address

Claim	Limitations
1[a]	**a virtual network manager implemented with a first device memory and a first device processor of a first computing device**, the virtual network manager configured to register devices in **a virtual network that is defined by a domain name**, …

[5] The '785 Patent explains that "[t]he VCN Manager 510 is a central server or server cluster providing management, connection, and security services for the VCN." Ex-1001, 10:66–11:1.

Claim	Limitations
30[a]	a **network interface configured for data communication** via **a virtual network that is defined by a domain name having an associated public network address**;
38[a]	**a computing device that includes at least a memory and a processor configured to implement a network manager of a virtual network that is defined by a public domain name**, …
48[a]	… **a virtual network that is defined by a domain name having an associated public network address in a public network**, …
62[a]	… **a virtual network that is defined by a domain name having an associated public network address in a public network**, …
75[a]-[b]	**a computing device that includes at least a memory and a processor configured to implement a virtual network manager having a network interface** coupled to a virtual network, **the virtual network manager including at least one virtual community definition that is defined by a domain name having an associated public network address** and a user set of one or more devices that are registered in the virtual network, …

a. A virtual network/manager

Caronni-I discloses a virtual network/community as a Supernet with channels. Caronni-I discloses an "administrative node 306 to administer to the needs of the Supernet." Id., 5:11–13. Further, the administrative node device contains a memory and a processor. "Each device contains similar components, including a memory 502, 504; secondary storage 506, 508; a central processing unit (CPU) 510, 512," where "[m]emory 504 of administrative machine 306 includes the SASD process 540, VARPD 548, and KMS 550 all running in user mode." *Id*., 6:48–55; Ex-1011, ¶¶ 212–214.

b. A network interface

Caronni-I discloses protocol stacks as a component in the administrative node, where the protocol stack is "acting as the conduit for all Supernet communications." Ex-1003, 8:59; *see also id.*, 6:63–68. For instance,

> TCP/IP protocol stack 534 contains a standard TCP/UDP layer 538, two standard IP layers (an inner IP layer 540 and an outer IP layer 544), and a Supernet security layer (SNSL) 542, acting as the conduit for all Supernet communications. To conserve memory, both inner IP layer 540 and outer IP layer 544 may share the same instance of the code of an IP layer.

Id., 8:56–62. Additionally, "KMD 530 receives requests from SNSL 542 of the TCP/IP protocol stack 534 when a packet is received[.]" *Id.*, 8:51–52. Figure 5, illustrated below, shows a TCP/IP protocol stack. Ex-1011, ¶¶ 221–222.

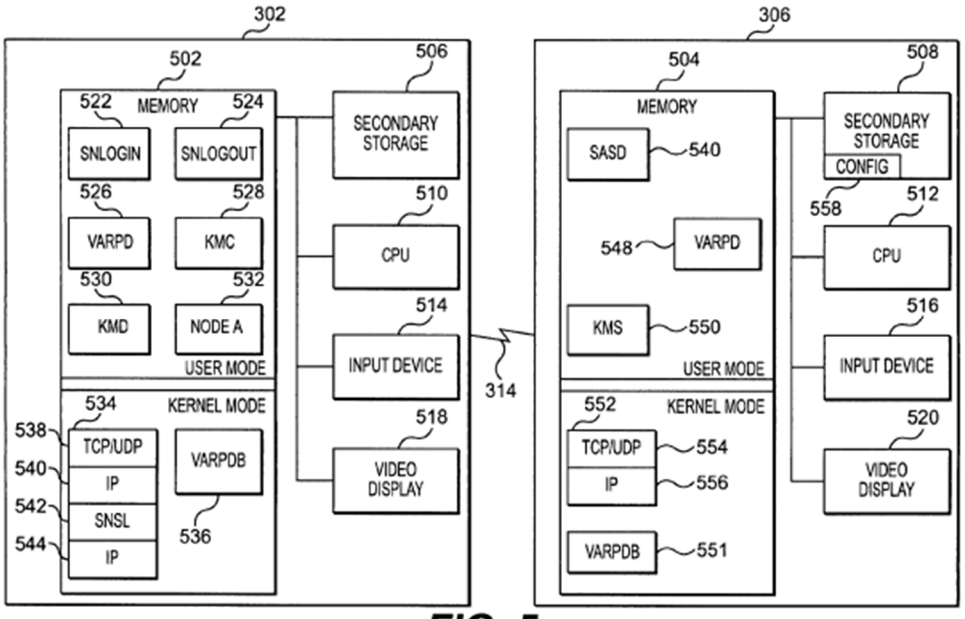


FIG. 5

Ex-1003, Figure 5.

Caronni-I discloses "the nodes of a Supernet may communicate over different transports, such as IP, IPX, X.25, or ATM, as well as different physical layers, such as RF communication, cellular communication, satellite links, or land-based links." Ex-1003, 5:21–25.

A POSITA would understand protocol stacks disclose the claimed network interfaces because they provide (programmable) interfaces and work with actual hardware (physical or virtual) network interface cards. Ex-1011, ¶¶ 223–224.

c. A domain name having an associated public network address

Caronni-I discloses a Supernet defined by a Supernet name and Supernet IDs (channels). Ex-1011, ¶¶ 232–234.

> To configure a Supernet, a System administrator creates a configuration file 558 that is used by SASD 540 when starting or reconfiguring a Supernet. This file may specify: (1) the Supernet name, (2) all of the channels in the Supernet, (3) the nodes that communicate over each channel, (4) the address of the KMS for each channel, (5) the address of the VARPD that acts as the server for the Supernet,

Ex-1003, 8:1–11. Further, the "SASD 540 represents a Supernet." Ex-1003, 7:3. Caronni-I also discloses that nodes in the Supernet community are defined by "all real IP addresses," which are maintained on the VARPDB (*id.*, 11:59–61), having an associated "node ID" including "a Supernet ID (e.g., 0x123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1)." *Id.*, 7:10–13. The Supernet IDs "indicat[e] the channels over which the

process communicates." *Id.*, 11:2–4. Caronni-I further teaches a user may enter a Supernet name when joining a Supernet. "The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password." *Id.*, 9:67–10:2; Ex-1011, ¶¶ 233–234.

Various public addresses are associated with the Supernet, such as a KMS address and/or a VARPD address. Ex-1003, 8:1–11. Moreover, the SASD would have an associated public IP address, as the SASD communicates with various Supernet devices. *Id.*, 7:3–5, 10:5–7. A POSITA would understand that because these administrative node components communicate over the Internet, public IP addresses would be used to communicate with the node components of other Supernet devices. *Id.*, 10:5–7 ("Devices 302 and administrative machine 306 communicate via the Internet 314."). Therefore, Caronni-I discloses a virtual network defined by a domain name having an associated public network address. Ex-1011, ¶¶ 232–235.

To the extent a domain name is a "human-readable address that identifies a specific location on the Internet, such as a website or online service," a POSITA would understand the Supernet name disclosed by Caronni-I is such a domain name. Ex-1015, 568. For example, Caronni-I discloses a system administrator and a user both enter the Supernet name, where the Supernet name would be a human compatible name for ease of entry. Ex-1011, ¶ 238.

Additionally, Hipp discloses a domain name (hostname) having an associated public network address. Ex-1005, 6:1–4. ("Virtualization of network identity is achieved by assigning a unique virtual IP address and virtual hostname to a group of processes that make up the application instance which the instance keeps throughout its execution."). A POSITA would understand the assignment of hostnames and IP addresses to Hipp processes may be applied to the processes of Caronni-I (SASD, VARPD, and KMD) to disclose this limitation. Ex-1011, ¶¶ 243–244.

3. Registration/distribution of a virtual network address to each device which uniquely identifies the device

Claim	Limitations
1[a]	… the virtual network manager **configured to register devices in a virtual network** that is defined by a domain name, **each device in the virtual network being identified to the other devices by a virtual network address that is unique for each device** and **not directly routable via a public network**, **the virtual network manager further configured to distribute a virtual network address to a device when the device is registered in the virtual network**;
30[b]	**a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to: receive a registration request from an agent associated with a device; distribute a virtual network address to the device when the device is registered in the virtual network**, **the device being identified to other devices in the virtual network by the virtual network address**; and
38[a]-[c]	… **the network manager configured to distribute a virtual network addresses to devices that register as members in the virtual network**, **each device in the virtual network being identified to the other devices by a virtual network address associated with the device; a first virtual network agent associated with a first device that is registered as a member in**

	the virtual network; at least a second virtual network agent associated with at least a second device that is registered as a member in the virtual network;
48[a]-[b]	**receiving registration requests from devices that request to be registered as members of a virtual network** that is defined by a domain name having an associated public network address in a public network**, each of the devices having an associated private network address**; **distributing a virtual network address to a device to register the device as a member in the virtual network**, **each device in the virtual network being identified to the other devices by the virtual network address that is associated with the device**;
62[a]-[b]	**receive registration requests from devices that request to be registered as members of a virtual network** that is defined by a domain name having an associated public network address in a public network, **each of the devices having an associated private network address; distribute a virtual network address to a device to register the device as a member in the virtual network, each device in the virtual network being identified to the other devices by the virtual network address that is associated with the device**;
75[b]-[d]	… **a user set of one or more devices that are registered in the virtual network, each device in the virtual network being identified to the other devices by a virtual network address that is associated with the device, the virtual network manager configured to exchange virtual network information with the one or more devices of the user set, the virtual network being accessible by devices in the user set and devices outside of the user set**, and

a. Register module and registration request via an agent

Caronni-I discloses an agent of each Supernet device (SNlogin) that communicates with the SASD on behalf of the device when the device requests to register with the virtual network. Ex-1003, 10:13–18. While "SNlogin 522 is a script used for logging into a Supernet[,]" "one skilled in the art will appreciate that [its]

processing may be performed by another form of software." *Id.*, 6:43–46, 8:30; Ex-1011, ¶¶ 246–248.

Caronni-I also discloses a system for registering and distributing virtual network addresses in a Supernet operating over a public network infrastructure. Caronni-I outlines a process for adding new nodes, from devices 302, 304, and 312 (i.e., a user set), to the Supernet and assigning them unique virtual addresses. Ex-1003, 9:66–67 ("FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet."); *see also id.*, 4:66–5:7, Figures 7A, 7B; Ex-1011, ¶ 249.

When a new node logs into the Supernet, it goes through an authentication process. "The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address." Ex-1003, 9:67–10:3. After successful authentication, the system generates an address mapping for the new node. *Id.*, 10:11–18 ("registers this information with the VARPD that acts as the sever for this Supernet"). A POSITA would understand this initial interaction of the SNlogin script and the SASD of the administrative node is the claimed registration request. *See also* Ex-1004, 3:11–20, 5:10–29, 7:35–39, 9:4–7. Moreover, a POSITA would recognize the administrative node—containing a processor and memory to execute the SASD, VARPD, and KMS processes—is the claimed registration module that performs a registration process to register nodes/devices in a Supernet. Ex-1011, ¶¶ 250–252.

b. Distribute/exchange a virtual address

Caronni-I discloses that after a device is registered to the Supernet, a virtual network address associated with the device is distributed to the device. As described in the registration process above, "SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services[,]" and then "registers this information with the VARPD[.]" Ex-1003, 10:11–18; Ex-1011, ¶ 258.

Additionally, Caronni-I explains that the first time a node sends a packet to another node, it must be supplied with its virtual network address from the administrative node's VARPD.

> The SNSL layer then accesses the VARPDB to obtain the address mapping between *virtual source node address* 642 and the *source real address* 614 as well as the virtual destination node address 644 and the *destination real address* 616 (step 808). If they are not contained in the VARPDB because *this is the first time a packet has been sent from this node* or sent to this destination, the VARPDB accesses the local VARPD to obtain the mapping. When contacted, the *VARPD on the local machine contacts the VARPD that acts as the server for the Supernet to obtain the appropriate address mapping*.

Ex-1003, 11:49–59 (emphasis added). A POSITA would understand the virtual network address is distributed to the node in an exchange of information when it "is the first time a packet has been sent from this node or sent to this destination." *Id.* Inherently, since the virtual network manager is configured to exchange virtual

network information with one or more devices of the user set, then the virtual network is accessible by the devices in the user set. Ex-1011, ¶¶ 258–261.

Caronni-I also discloses a virtual network being accessible by devices outside of the user set. For example, Node C below is a file system manager that "stores the data in an encrypted form so that it is unreadable by others." Ex-1003, 5:44–49.

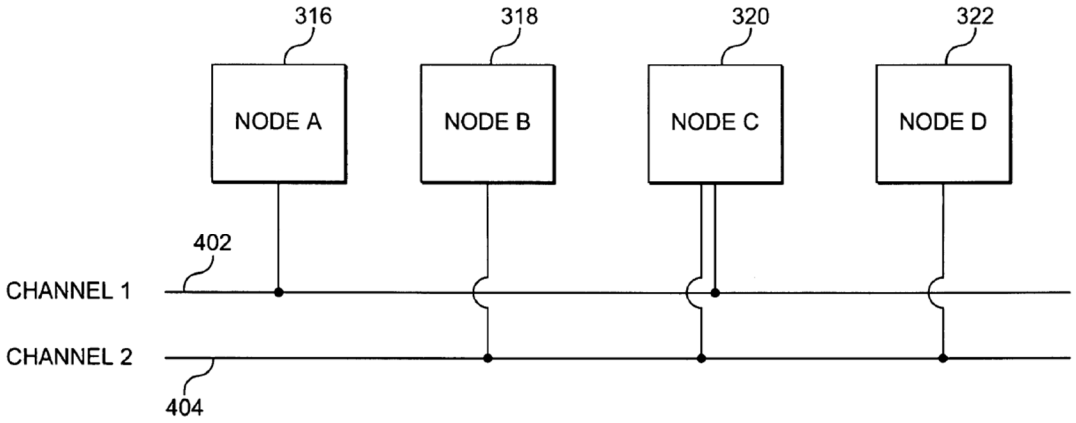


FIG. 4

Id., Figure 4. A POSITA would understand that a portion of Node C's secondary storage is accessible to the user set of channel 1 (Nodes A and C) would also be accessible to the user set of channel 2 (Nodes B, C and D). And further that the user set of Nodes B and D are outside the user set of Node A because, due to the encryption, the data is "unreadable by others." Ex-1011, ¶¶ 290–292.

Furthermore, Hipp discloses a virtual network being accessible by devices outside of the user set if permitted. Ex-1005, 2:11–13 ("However, an application in one VNE cannot communicate with an application in another VNE (unless expressly

permitted)."); *see also* Ex-1003, 5:27–30, 5:36–38, 5:44–49, Figure 4 (disclosing Node C); Ex-1011, ¶¶ 293–299.

c. Identified by a unique virtual network address

A POSITA would recognize that Caronni-I's node ID, being a concatenation of a Supernet ID and a virtual network address, is itself a virtual network address. A POSITA would also understand the virtual network address and the node ID are unique and used to identify a node/device to other nodes/devices in the virtual network. Ex-1011, ¶¶ 262–264.

For instance, the node ID serves as a unique identifier within the Supernet and consists of two parts. "The 'node ID' may include the following: a Supernet ID (e.g., 0x123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1)." Ex-1003, 7:10–13; *see also* Ex-1005, 6:1–4. A POSITA would understand that a node's virtual network address is unique because, "[s]ince the node ID includes a Supernet ID, a node will have more than one node ID when it communicates over more than one channel." Ex-1003, 7:16–18. In other words, virtual network addresses are unique because within a particular channel or Supernet ID, individual nodes must be distinguishable—by their virtual address—from other nodes on that channel. Ex-1011, ¶¶ 265–266.

Moreover, when transmitting data, the system uses modified socket structures containing both virtual and real addresses. Ex-1003, 11:38–40 ("The socket structure

is modified so as to contain an extra data field for Supernet ID 626 and virtual source address 642."). This allows the Supernet Socket Layer (SNSL) to "access[] the VARPDB to obtain the address mapping between virtual source node address 642 and the source real address 614 as well as the virtual destination node address 644 and the destination real address 616," enabling secure cross-network communication without exposing physical infrastructure details. *Id.*, 11:49–53; Ex-1011, ¶¶ 270–271.

d. Associated private network address/not directly routable via a public network

Caronni-I discloses that a Supernet is a private network with an addressing scheme separate from the underlying public network such as the Internet. Ex-1003, 6:7–18. A POSITA would understand the virtual network address of the private Supernet addressing scheme that is separate from the underlying public network wouldn't be routable over the underlying public network. Ex-1011, ¶¶ 272–275.

For example, Caronni-I describes a virtual network address as "a virtual address comprising an IP address (e.g., 10.0.0.1)" and a real address comprising an "IP address (e.g., 10.0.0.2)." Ex-1003, 7:12–13, 7:19–21. A POSITA would recognize these IP addresses, 10.0.0.1 and 10.0.0.2, are reserved or private IP addresses associated with the Supernet's nodes/devices that are not routable over a public network such as the Internet. Ex-1011, ¶¶ 272–273.

Caronni-I discloses that Supernet nodes/devices have an associated private network address.

> VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as a node IDS, to the network addresses recognized by the public-network infrastructure, known as the real addresses.

Ex-1003, 7:5–9. Caronni-II discloses a message is sent using a real IP address destination, where the destination may be behind an edge device (firewall, NAT box, proxy server, packet filtering device, and gateway). Ex-1004, 9:20–24; *see also id.*, 9:31–32, 1:60–2:6. Likewise, Hipp discloses virtual addresses in the context of a virtual network environment (VNE), including reserved or private IP addresses. Ex-1005, 3:19–27; *see also id.*, Figures 1, 3–7; Ex-1011, ¶ 276. Thus, a POSITA would understand that the combination of Caronni-I/II and Hipp discloses an associated private network address. Ex-1011, ¶¶ 279–289.

4. Route director/routing

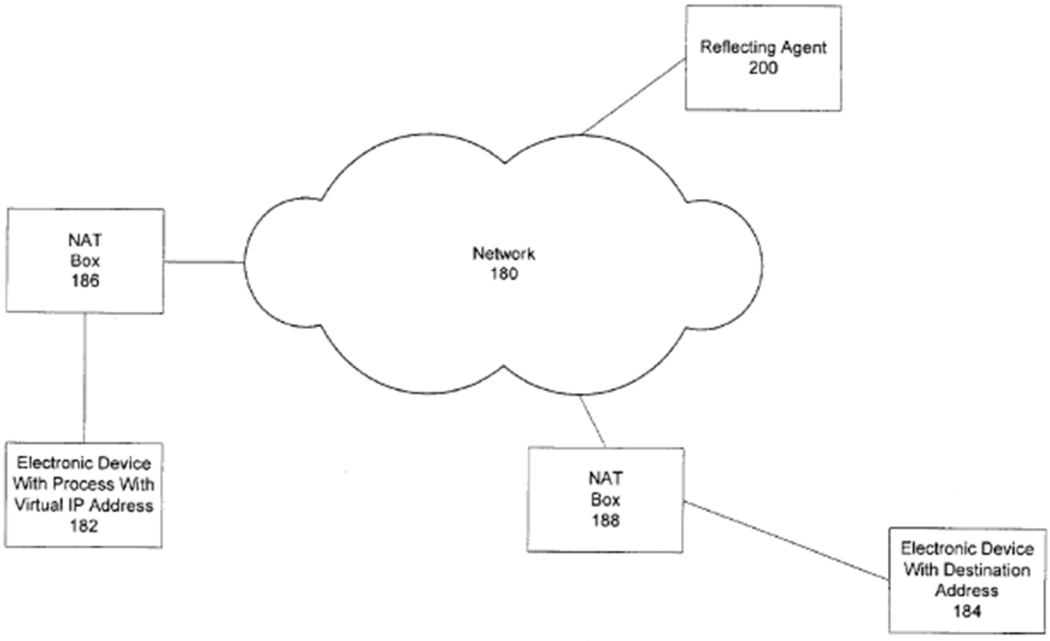
Claim	Limitations
1[b]	**a route director implemented with a second device memory and a second device processor of a second computing device, the route director configured to communicate data between the devices that are registered in the virtual network, the data being communicated as encapsulated packets from a source device to a destination device, an encapsulated packet including a first virtual network address that corresponds to the source device and a second virtual network address that corresponds to the destination device**; and

30[c]	… the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a **network route director**; …
38[d]	**a route director configured to route communications between the first device and the at least second device in the virtual network** via the respective first and second virtual network agents, **the communications configured for routing as encapsulated packets that include a first virtual network address** that is not directly routable **corresponding to the first device** and **a second virtual network address** that is not directly routable **corresponding to the at least second device**; and
48[c]	**routing communications between the devices that are registered in the virtual network, the communications being routed as encapsulated packets from a source device to a destination device, an encapsulated packet including a first virtual network address that corresponds to the source device and a second virtual network address that corresponds to the destination device**; and
62[c]	**manage communications routed between the devices that are registered in the virtual network, the communications routed as encapsulated packets from a source device to a destination device, an encapsulated packet including a first virtual network address that corresponds to the source device and a second virtual network address that corresponds to the destination device**; and
75[e]	the virtual network manager further configured to receive a DNS request from a source device, and return a public address of **a route director**, …

a. Route Director

Caronni-II discloses a reflecting agent 200 that allows communication to a member in a private network. Ex-1001, 25:31–36 ("Route directors allow communication to a member that is in a private network.").

Figure 5



Ex-1004, Figure 5. Ex-1011, ¶ 301.

Caronni-II's reflecting agent and the '785 Patent's route director both act as intermediaries to enable communication between nodes behind different NAT boxes. Ex-1004, 8:22–25 ("Messages that are being sent from the originating virtual address behind a NAT box to a destination which is behind a different NAT box are sent via the reflecting agent intermediary outside the NAT box and reflected to the destination."); *see also* Ex-1001, 12:23–26 ("In order to route packets to a peer in a different addressing realm, the protocol stack knows the address of the appropriate Route Director that server's the peer's realm, the private address of the peer or a NAT address."). Furthermore, the reflecting agent's IP address is added to the VARP

table to enable communication between devices located behind different NAT boxes. Ex-1004, 8:35–38.

> When the destination address is also behind a NAT box, the originator of the message is unable to directly address the destination (since the address of the destination may not be routable in the public Internet). In such a case, the present invention adds an entry to the VARP table for a third-party reflecting agent located at an address outside the NAT box.

Id., 8:13–19. By adding the reflecting agent's real address to the VARP table, messages can be routed through this intermediary to reach their intended destination, circumventing the addressing limitations imposed by NAT boxes. Ex-1011, ¶¶ 302–304. Finally, a POSITA would understand the electronic devices disclosed in Caronni-II and Hipp would be implemented with a memory and processor. Ex-1011, ¶ 304.

b. Encapsulated packets

Caronni-I discloses encapsulated packets that include a first virtual network address corresponding to the source device and a second virtual address that corresponds to the destination device. Ex-1003, 12:8–16; *see also id.*, 2:29–31 (explaining tunneling "refers to encapsulating one packet inside another when packets are transferred between two end points."). Ex-1011, ¶¶ 305–306.

For instance, Caronni-I illustrates packet encapsulation with virtual source and destination address, where a Supernet IP packet 600 is shown. Ex-1003, Figure

6. During the encapsulation process, inner layer 660 receives a packet originating from a source node and includes a virtual source node address 642 and a destination source address 644. *Id.*; *see also id.*, 11:26–30. A POSITA would recognize that these addresses are the claimed first virtual network address that corresponds to the source device and a second virtual network address that corresponds to the destination device. Ex-1011, ¶ 307.

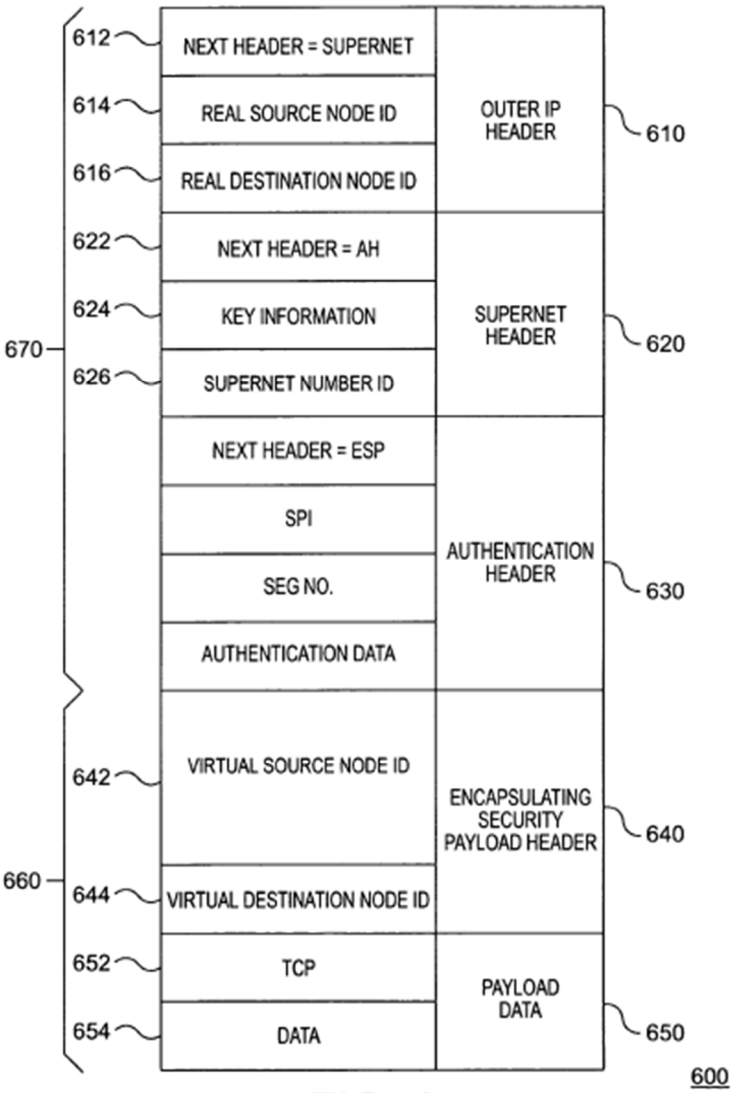


FIG. 6

Ex-1003, Figure 6.

Additionally, Caronni-I explains that "[t]he packet and Supernet ID are then transmitted to the SNSL layer." *Id.*, 11:47–48. Referring to Figure 6 above, "[d]elivery scheme section 670 corresponds to the SNSL layer 542 and is meaningful only to the public-network infrastructure." *Id.*, 9:8–10. The delivery scheme section 670 includes both real source and destination node address for routing the packet 600 over from destination node to source node. *Id.*, 9:17–20 ("Source real address 614 contains the real address of the originating node of Supernet packet 600. Destination real address 616 contains the real address of the destination node of Supernet packet 600."); Ex-1011, ¶¶ 308–310.

Moreover, the VARPD and SNSL work together to encapsulate packets for secure communication between source and destination devices in the Supernet system. The SNSL "act[s] as the conduit for all Supernet communications." Ex-1003, 8:58–59. When sending a packet, the SNSL "accesses the VARPDB to obtain the address mapping between virtual source node address 642 and the source real address 614 as well as the virtual destination node address 644 and the destination real address 616." *Id.*, 11:49–53. The SNSL then "encrypts the packet using the appropriate encryption algorithm and the key previously obtained," ensuring that "the virtual source node address 642, the virtual destination node address 644, and the data may be encrypted (addressing section 660), but the source and destination

real addresses 614, 616 (delivery scheme section 670) are not, so that the real addresses can be used by the public network infrastructure to send the packet to its destination." *Id.*, 12:8–16. This encapsulation allows for secure communication over public networks while maintaining routing functionality. Further, a POSITA would understand the packet originating from a source node and including a virtual source node address and a destination source node address is thus encapsulated into the Supernet packet 600. Ex-1011, ¶¶ 310–312.

Caronni-II also discloses the claimed encapsulation. As shown in Figure 3, an application layer message is encapsulated where, "[m]essages intended for a virtual destination address … are encapsulated in higher level protocols prior to being sent to the edge device." Ex-1004, Abstract; *see also id.*, Figure 3; Ex-1011, ¶¶ 313–315.

5. DNS server, request, and responses

Claim	Limitations
1[c]	**the virtual network manager further configured to receive a DNS request from the source device, and return a public network address of the route director, a private network address for the destination device, and the second virtual network address that corresponds to the destination device**.
30[c]	a **DNS server** for the virtual network, **the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device**.
38[e]	the network manager includes a **DNS server** configured to provide authoritative responses for **DNS queries** in the virtual network, **the DNS server further configured to receive a DNS query from the**

Claim	Limitations
	first device and return a network address of the route director, a network address of the second device, and the virtual network address of the second device.
48[d]	transmitting a response to **a DNS request** received from one of the devices that are the members in the virtual network, **the response to the DNS request including a public network address of a route director that registers the devices, a public network address of the destination device, and the second virtual network address that corresponds to the destination device**.
62[d]	transmit a response to a **DNS request** received from one of the devices that are the members in the virtual network, **the response to the DNS request including a public network address of the virtual network manager, a public network address of the destination device, and the second virtual network address that corresponds to the destination device**.
75[e]	**the virtual network manager further configured to receive a DNS request from a source device, and return a public network address of a route director, a private network address for a destination device, and a virtual network address that corresponds to the destination device**.

a. DNS server for the virtual network

The Caronni-I/II and Hipp combination discloses a DNS server that responds to DNS queries by returning the addresses claimed in the ’785 Patent, including (1) a public network address of a route director/network manager, (2) a network address of the destination device, and (3) virtual network address for a destination device.

Hipp discloses a DNS server for use in a virtual network environment, where a unique virtual hostname is resolved or translated into a unique virtual IP address. Ex-1005, 6:1–4; *see also id.*, 6:16–18 (“The virtual hostname resolves to the virtual IP address for both the applications registered with the VNI framework as well as

those that are not registered."). Hipp proposes that this virtual network DNS server is a standard DNS mechanism that has been preconfigured to resolve virtual hostnames to virtual IP addresses. *Id.*, 6:20–26 ("For example, if an application instance used a virtual IP address of 10.10.0.1 and a virtual hostname of host1055, *the standard hostname to IP address resolution mechanisms* (e.g. *DNS* or the /etc/hosts file) would have to be preconfigured *to resolve a query of host1055 to IP address 10.10.0.1*.") (emphasis added); Ex-1011, ¶¶ 323–324.

Furthermore, Caronni-I contemplates the use of DNS as it discloses that "SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner." Ex-1003, 10:13–16. A POSITA would understand the "querying network services in a well-known manner" is a DNS operation. Moreover, Caronni-I explains that member virtual addresses are known to the member nodes of a specific channel or Supernet ID. *Id.*, 9:30–31. Caronni-I exemplifies this previous knowledge of a member node virtual address in its description of creating a packet for transmission from one node in the Supernet to another. *Id.*, 11:18–32; *see also id.*, Figure 8. A POSITA would understand that higher layer processes in Caronni-I would use DNS services, such as those disclosed by Hipp or those generally known to a POSITA, to initially determine a virtual destination node address. Ex-1011, ¶¶ 319–322.

b. DNS request

Hipp discloses DNS requests or queries. Ex-1005, 6:16–26 ("preconfigured to *resolve a query of host1055 to IP address 10.10.0.1*") (emphasis added). Likewise, Caronni-I and II disclose DNS type requests through the operation of the VARP address translation process. Ex-1003, 9:47–54, 7:22–28; Ex-1004, 3:20–23 ("A resolution request is received …."), 5:10–22 ("Each entry in the VARP lookup table 26 will list the registering virtual IP address 90 and associated information including a real IP address 91, …."). A POSITA would understand both Caronni-I (address translation) and Caronni-II (VARP lookup table) are similar to DNS operations. Ex-1011, ¶¶ 328–330. Thus, a POSITA would understand that the combination of Caronni-I/II and Hipp discloses a DNS request. Ex-1011, ¶¶ 326–330.

c. DNS response

As discussed further below, Caronni-II discloses the address of the reflecting agent is added to the VARP table. Ex-1004, 8:19–21 ("[T]he present invention adds an entry to the VARP table for a third-party reflecting agent located at an address outside the NAT box."). A POSITA would recognize this addition to the VARP table would provide multiple addresses from a single VARP query. Ex-1011, ¶¶ 336–337.

A POSITA would recognize the Caronni-I/II and Hipp combination would disclose a way to store and return multiple IP address in response to a single query.

See supra Section VI.E. This matches the DNS response of the '785 Patent. Ex-1011, ¶¶ 332–333.

i. Public network address of a route director/virtual network manager

Caronni-II discloses the address of the reflecting agent is added to the VARP table. Ex-1004, 8:19–21 ("[T]he present invention adds an entry to the VARP table for a third-party reflecting agent located at an address outside the NAT box."). A POSITA would recognize an address outside of a NAT box would be a public network address. Ex-1011, ¶ 334.

As a part of the VARP address resolution response, the reflecting agent's address would be returned. Ex-1004, 3:20–23 ("A resolution request is received referring a virtual address destination and the resolution request is resolved using the virtual address resolution facility and the stored associations."); Ex-1011, ¶ 335.

Also, Caronni-II discloses "[t]he virtual address resolution facility 24 is used to determine the associations registered with the destination address." Ex-1004, 6:39-41. Therefore, a POSITA would've been motivated to combine Caronni-II and Hipp to add this step of returning the public IP address. Further, a POSITA would've been motivated to combine the systems and methods in Caronni-I/II and Hipp for the reasons discussed above. *See supra* § VI.E; Ex-1011, ¶¶ 336–337.

ii. Network address of the destination device

Caronni-I discloses returning a network address of a destination device as a destination real address. For instance, "[t]he SNSL layer [] accesses the VARPDB to obtain the address mapping between virtual source node address 642 and the source real address 614 as well as the virtual destination node address 644 and the destination real address 616 (step 808)." Ex-1003, 11:49–53; *see also* Ex-1004, Ex-1004, 2:52–62, 4:27–33, 5:24–26; Ex-1011, ¶ 338.

Furthermore, Caronni-I discloses that the real addresses may be public or private network addresses. For example, "VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as [] node IDs, to the network address recognized by the public-network infrastructure, known as the real addresses." Ex-1003, 7:5–9. Further, Caronni-I indicates a real address may be a private address. "The 'real address' is an IP address (e.g., 10.0.0.2)." *Id.*, 7:18–19; *see also supra* Section VII.A.3.d (discussing that IP address 10.0.0.2 is a reserved or private IP addresses); Ex-1011, ¶ 339.

Also, Caronni-II discloses "[e]ach entry in the VARP lookup table 26 will list the registering virtual IP address 90 and associated information including a real IP address 91, a transport protocol designation 92, a port number identification 93 and

an Application layer protocol designation 94." Ex-1004, 5:18–22; Ex-1011, ¶¶ 340–342.

iii. Virtual network address that corresponds to the destination device

Caronni-I discloses returning a virtual network address that corresponds to the destination device. For example, "[e]ach time web client 1102 requests a packet from web server 1104a, the client requests the virtual address of the web server 1004a from computer system 1106." Ex-1003, 13:14–17. As explained above, Hipp also discloses returning a virtual network address that corresponds to the destination device. *See supra* Section VII.5.a–b; *see also* Ex-1005, 4:58–64, 6:16–26; Ex-1011, ¶¶ 343–344. A POSITA would understand that the combination of Caronni-I/II and Hipp discloses a DNS response. Ex-1011, ¶ 345.

B. The Dependent Claims

1. Dependent Claims 35–37 and 77–78

Claims 31–37 depend from Claim 30 and Claims 77–78 depend from Claim 75. These claims generally recite additional limitations regarding a join module receiving a join request and are obvious in view of Caronni-I/II and Hipp.

a. Dependent claims 35 and 77

Petitioners incorporate by reference their analysis to claims 30 and 75.

Further, Caronni-I discloses an administrative node including a join module that is responsive to SNlogin and SNlogout scripts, where these scripts are agents

associated with a device. Caronni-I discloses the first time a node accesses or is accessed over a Supernet, the local VARPDB (part of the device agent) sends a message to the administrative node to obtain its address mappings. Ex-1003, 11:53–59. A POSITA would understand this message is an indication that the device is connected for data communication within the virtual network. *Id.*, 12:64–65; Ex-1011, ¶¶ 371–372.

Further, Caronni-I discloses devices logging out of a Supernet. Ex-1003, 12:62–65. ("[T]he SNlogout script requests a log out from SASD[.]"). When the logout request is received, the SASD removes the node's mapping from the VARPD, "informs the KMS to cancel the registration of the node, and KMS terminates this KMC[.]" *Id.*, 12:65–13:2. A POSITA would understand the log out message from SNlogout is a leave request from the device agent to indicate that the device will be disconnected from data communications. Ex-1011, ¶¶ 373–375.

b. Dependent claims 36 and 78

Petitioners incorporate by reference their analysis to claims 30, 35 and 75.

Moreover, a POSITA would understand the administrator node thus provides a virtual network address to a registered device. A POSITA would further understand only registered devices have addresses in the server VARPD as discussed in Claim 35 with regard to SNlogout. Ex-1011, ¶¶ 376–377.

c. Dependent claim 37

Petitioners incorporate by reference their analysis to claims 30 and 35.

Additionally, Caronni-I's admin node contains a server VARPDB (join module and part of administrative node) that stores and maintains device information, including addresses in mapping tables. Ex-1003, 7:5–9 ("VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as a node IDs, to the network addresses recognized by the public-network infrastructure, known as real addresses."). Ex-1011, ¶¶ 378–379.

VIII. GROUND 2: Claims 1, 30, 35–38, 48, 62, 75, and 77–78 are obvious in view of Caronni-I in combination with Caronni-II and RFC-1383

A. The Independent Claims

The same independent and dependent claims as discussed in Ground 1 are discussed in the same manner below with respect to the Ground 2 references. These limitations would've been obvious to a POSITA before the time of the filing of the '785 Patent, in light of Caronni-I combined with Caronni-II and RFC-1383. Ex-1011, ¶ 380.

1. Preamble: virtual network system or manager

Caronni-I/II disclose this limitation as discussed above in § VII.A.1. Ex-1011, ¶¶ 381–382.

2. A virtual network defined by a domain name having an associated public network address[6]

a. virtual network manager

Caronni-I discloses this limitation as discussed above in § VII.A.2.a. *Id.*

b. A network interface

Caronni-I discloses this limitation as discussed above in § VII.A.2.b. Ex-1011, ¶¶ 383–384.

c. A domain name having an associated public network address

Caronni-I discloses this limitation as discussed above in § VII.A.2.c. *Id.*

3. Registration/distribution of virtual network address to each device which uniquely identifies the device[7]

a. Register module and registration request via an agent

Caronni-I/II disclose this limitation as discussed above in § VII.A.3.a. Ex-1011, ¶¶ 385–386.

b. Distribute/exchange a virtual address

Caronni-I discloses this limitation as discussed above in § VII.A.3.b. *Id.*

c. Identified by a unique virtual network address

Caronni-I discloses this limitation as discussed above in § VII.A.3.c. *Id.*

[6] Petitioners incorporate by reference the claim table included in § VII.A.2.

[7] Petitioners incorporate by reference the claim table included in § VII.A.3.

d. Associated private network address/not directly routable via a public network

Caronni-I discloses this limitation as discussed above in § VII.A.3.d. *Id.*

4. Route director/routing[8]

a. Route Director

Caronni-II discloses this limitation as discussed above in § VII.A.4.a. Ex-1011, ¶¶ 387–388.

b. Encapsulated packets

Caronni-I/II disclose this limitation as discussed above in § VII.A.4.b. *Id.*

5. DNS server, request, and responses[9]

a. DNS server for the virtual network

The Caronni-I/II and RFC-1383 combination discloses a DNS server that responds to DNS queries by returning the addresses claimed in the '785 Patent, including (1) a public network address of a route director/network manager, (2) a network address of the destination device, and (3) virtual network address for a destination device.

Caronni-I discloses this limitation as discussed above in § VII.A.5.a. Ex-1011, ¶ 389.

[8] Petitioners incorporate by reference the claim table included in § VII.A.4.
[9] Petitioners incorporate by reference the claim table included in § VII.A.5.

RFC-1383 discloses a generalized scheme for packet routing using a standard DNS with a defined DNS record. Ex-1006, 2; *see also id.*, 5, 11 ("In the scheme that we propose, the DNS is only accessed once" and "will use the general purpose 'TXT' record."); *supra* § VI.D; Ex-1011, ¶ 391.

Accordingly, a POSITA would understand the Caronni-I/II and RFC-1383 combination discloses a DNS server. *See supra* § VI.F; Ex-1011, ¶ 392.

b. DNS request

RFC-1383 discloses a DNS request:

> [T]he query manager will send a *DNS request*, in order to read the RX *records* corresponding to the destination. After reception of the *response*, it will select a gateway, and pass the information to the real time forwarder.

Ex-1006, 12 (emphasis added); Ex-1011, ¶ 393.

Likewise, Caronni-I/II disclose DNS type requests through the operation of the VARP address translation process. Ex-1003, 9:47–54, 7:22–28; Ex-1004, 3:20–23, 5:10–22. A POSITA would understand that both Caronni-I (address translation) and Caronni-II (VARP lookup table) are similar to DNS operations. Ex-1011, ¶ 328.

c. DNS response

As discussed above in § VI.D, RFC-1383 discloses a way to store and return multiple IP addresses in response to a single query using TXT fields or records. *See also supra* § VI.F. This matches the DNS response of the '785 Patent. In addition,

Caronni-II discloses this limitation as discussed above in § VII.A.5.c. Ex-1011, ¶¶ 394–395.

A POSITA would recognize that the Caronni-I/II and RFC-1383 combination would disclose a way to store and return multiple IP address in response to a single query. *See supra* Section VI.F. This matches the DNS response of the '785 Patent. Ex-1011, ¶¶ 395–396.

i. Public network address of a route director/virtual network manager

Caronni-II discloses this limitation as discussed above in § VII.A.5.c.i.

Further, a POSITA would've been motivated to combine the systems and methods in Caronni-I/II and RFC-1383 for the reasons discussed above. *See supra* § VI.F; Ex-1011, ¶ 396.

ii. Network address of the destination device

Caronni-I/II disclose this limitation as discussed above in § VII.A.5.c.ii. Additionally, RFC-1383, returns an "RX" record with private IP address "10.0.0.7." Ex-1006, 11.

A POSITA would've been motivated to combine the systems and methods in Caronni-I/II and RFC-1383 for the reasons discussed above. *See supra* § VI.F; Ex-1011, ¶ 396.

iii. Virtual network address that corresponds to the destination device

Caronni-I discloses this limitation as discussed above in § VII.A.5.c.iii.

A POSITA would've been motivated to combine the systems and methods in Caronni-I/II and RFC-1383 for the reasons discussed above. *See supra* § VI.F; Ex-1011, ¶ 396.

B. The Dependent Claims

1. Dependent Claims 35–37 and 77–78

Claims 35–37 depend from Claim 30 and Claims 77–78 depend from Claim 75. These claims generally recite additional limitations regarding a join module receiving a join request and are obvious in view of Caronni-I/II as discussed above in §§ VII.B.1.a-c. Ex-1011, ¶¶ 427–436.

IX. DISCRETIONARY DENIAL IS NOT APPROPRIATE

A. *General Plastic*

This is Petitioners' first Petition for IPR of the '785 Patent. The '785 Patent was challenged in one prior IPR filed in 2024 by Liberty Mutual Technology Group, Inc. *et al.* ("Liberty"): *Liberty Mutual Technology Group, Inc. et al v. Intellectual Ventures I LLC*, IPR2025-00201, Paper 3 (P.T.A.B. Nov. 20, 2024). Although *General Plastic* addressed IPR petitions challenging the same patent filed by "the same petitioner," the Board in *Valve Corp. v. Elec. Scripting Prods., Inc.*, IPR2019-00062, Paper 11 (P.T.A.B. Apr. 2, 2019) (precedential) extended *General Plastic*

factor 1 to include IPR petitions challenging the same patent filed by petitioners sharing a "significant relationship … with respect to [the challenged patent]." *Valve*, at 9–10.

Petitioners never had a "significant relationship" with Liberty regarding the '785 Patent. *See Ford Motor Co. v. Neo Wireless LLC*, IPR2023-00763, Paper 28, at 7 (P.T.A.B. Mar. 22, 2024). In *Ford*, the Director vacated and remanded the Board's discretionary denial because the first and second petitioners' "relationship … [was] premised on the allegation that they each infringe the same patent, but with different allegedly infringing products and in different district court proceedings." IPR2023-00763, Paper 28 at 10. The same is true of Petitioners and Liberty. Patent Owner filed suit against Liberty in the Eastern District of Texas on November 15, 2023. Ex. 1009. Patent Owner and Liberty stipulated to dismissal of that case. The order dismissing the case was signed on January 16, 2025. Ex. 1010. Patent Owner filed suit against American and Southwest in the Eastern District of Texas and the Western District of Texas, respectively, on November 2, 2024, alleging infringement of different products. Ex. 1007; Ex-1008. Petitioners and Liberty thus lack the "closely aligned interests with respect to the ['785] [P]atent" necessary for a "significant relationship" to exist. *Ford*, IPR2023-00763, Paper 28 at 9–10.

"[W]here, as here, the first and second petitioners are neither the same party, nor possess a significant relationship under *Valve*, *General Plastic* factor one

necessarily outweighs the other *General Plastic* factors." *Videndum Production Solutions, Inc. v. Rotolight Ltd.*, IPR2023-01218, Paper 12, at 6 (P.T.A.B. Apr. 19, 2024). Moreover, there have been no communications, written or oral, between Petitioners and Liberty relating to the financing, preparation, editing, review, or approval of this Petition. No individuals acting for or on behalf of Liberty participated or assisted in any way with any of these activities. There have been no payments or agreements between Liberty and Petitioners, directly or indirectly, in connection with this Petition or the '785 Patent.

The Board "frequently decline[s] to exercise discretionary denial under *General Plastic* where there is no 'significant relationship' between parties challenging the same patent." *Samsung Bioepis Co. Ltd. v. Regeneron Pharms., Inc.*, IPR2023-00442, Paper 10, at 46 (P.T.A.B. July 19, 2023). The Board should do so here.

B. 35 U.S.C. § 314(a)

Fintiv and the Office's interim guidelines dated June 21, 2022 ("Interim Procedures") favor institution, which best serves the efficiency and integrity of the patent system. Factor 1 is at worst neutral because no request for a stay pending IPR has been filed yet. *Sand Revolution II LLC v. Continental Intermodal Group – Trucking LLC*, IPR2019-01393, Paper 24, at 7 (P.T.A.B. June 16, 2020)

(informative). Nonetheless, Petitioners are considering requesting a stay, which favors institution.

Factor 2 is neutral. Although no trial date is yet set in IV's case against American, the schedule jointly submitted to the district court would provide for a trial more than eighteen months after the filing date of this petition. And although trial is set for July 27, 2026, in the case against Southwest, that date often changes. For example, the most recent time-to-trial data show that the median time to trial for jury trials in the Western District of Texas is 34.9 months. Table T-3—U.S. District Courts–Trials Statistical Tables for the Federal Judiciary (December 31, 2024), U.S. Courts, https://www.uscourts.gov/data-news/data-tables/2024/12/31/statistical-tables-federal-judiciary/t-3 (last visited March 28, 2025). The Board has instituted IPR based on other petitions with similar facts. *Ericsson Inc. v. XR Commc'ns LLC*, IPR2024-00613, Paper 9, 33-34 (P.T.A.B. Oct. 9, 2024).

Factor 3 favors institution because the parties have expended limited resources in the litigations and Petitioners acted expeditiously—*i.e.*, by filing the petition within four months of the original complaints and well before the start of the claim construction process, in the early stages of fact discovery, and before expert discovery. *Sotera Wireless, Inc. v. Masimo Corp.*, IPR2020-01019, Paper 12 at 16-17 (P.T.A.B. Dec. 1, 2020) (precedential).

As to factors 4 and 5, only three claims were asserted in district court so far, so this Petition addresses issues that may not be litigated there, and this weighs against discretionary denial. *See* Ex-1025; *see also* Ex-1026. Moreover, while Petitioners and Patent Owner are the same parties in the district court, this is "far from an unusual circumstance" in IPR and doesn't warrant discretionary denial. *Sand Revolution*, IPR2019-01393, Paper 24 (instituting where parties were the same).

The Board only addresses Factor 6 if Factors 1-5 support discretionary denial, which they do not. *CommScope Techs. LLC. v. Dali Wireless, Inc.*, IPR2022-01242, Paper 23 (P.T.A.B. Feb. 27, 2023) (precedential).

This Petition presents a compelling unpatentability challenge. The cited prior art, none of which was cited by the Examiner in any office action, rebuts the Patent Owner's arguments during prosecution that persuaded the Examiner to allow the claims. Accordingly, the Board shouldn't deny institution under *Fintiv* and the Interim Procedures.

C. 35 U.S.C. § 325(d)

None of the combinations of the prior art references relied on herein under § 103, much less the combination of those teachings as informed by a POSITA's knowledge and expert testimony, were considered by the Examiner during prosecution, nor relied upon in the *ex parte* reexamination request filed by Unified

Patents, LLC (Control No. 90/019,519). *See, e.g., Mylan Pharm. Inc., v. Regeneron Pharm., Inc.*, IPR2022-01226, Paper 22, at 27–29 (P.T.A.B. Jan. 11, 2023). Further, this Petition relies Caronni-I which was not cited or relied upon during prosecution or in the *ex parte* reexamination. Neither Caronni-II,[10] Hipp, nor RFC-1383 were considered during prosecution. Notably, this Petition seeks to review over 10 claims of the '785 Patent, while only two claims (30 and 34) were subject to reexamination. Ex-1015. This Petition thus largely addresses issues that will not be addressed in the reexamination.

Accordingly, this Petition shouldn't be denied under 35 U.S.C. § 325(d).

X. CONCLUSION

Inter partes review of the Challenged Claims is respectfully requested.

[10] Caronni-II was cited by the Examiner during prosecution, as it is listed on the face of the patent, but he did not rely on it as a primary reference.

XI. MANDATORY NOTICES

A. Real Party in Interest

Petitioners identify themselves as real parties in interest.

B. Related Matters

To the best of Petitioners' knowledge, the '785 Patent has been involved in the following related matters:

- *Intellectual Ventures I LLC et al. v. Liberty Mutual Holding Company Inc., et al.*, No. 23-cv-525 (E.D. Tex. filed November 15, 2023)
- *Intellectual Ventures I LLC et al. v. Comerica Incorporated*, No. 23-cv-524 (E.D. Tex. filed November 15, 2023)
- *Intellectual Ventures I LLC et al. v. JP Morgan Chase Bank, National Association et al.*, No. 23-cv-523 (E.D. Tex. filed November 15, 2023)

These cases were consolidated, with the latter Case No. 23-cv-523 as the lead case. The action against Liberty, Comerica, and JP Morgan has been dismissed with prejudice.

The '785 Patent was also at issue in the following IPR proceeding:

- IPR2025-00201

The '785 Patent is also asserted in the following cases:

- *Intellectual Ventures I LLC et al. v. American Airlines, Inc.*, No. 24-cv-980 (E.D. Tex. filed November 2, 2024)

- *Intellectual Ventures I LLC et al. v. Southwest Airlines, Co.*, No. 24-cv-277 (W.D. Tex. filed November 2, 2024)

The '785 Patent is also at issue in the following declaratory judgment case:

- *Assurant, Inc. v. Intellectual Ventures I LLC, et al.*, No. 24-cv-344 (D. Del. Filed March 15, 2024)

The '785 Patent was also the subject of an *ex parte* reexamination request filed by Unified Patents, LLC on May 22, 2024 (Control No. 90/019,519). Reexamination was ordered on June 14, 2024. The Patent Office filed a Notice of Intent to Issue a Reexamination Certificate on March 6, 2025.

Petitioners are not aware of any other civil actions or proceedings filed in connection with the '785 Patent.

C. Lead and Backup Counsel and Service Information

Lead Counsel	Backup Counsel
John B. Campbell (Reg. No. 54,665) jcampbell@McKoolSmith.com MCKOOL SMITH, P.C. 303 Colorado Street Suite 2100 Austin, TX 78701 Tel: (512) 692-8730 Fax: (512) 692-8744	Casey Shomaker (Reg. No. 77,998) cshomaker@mckoolsmith.com McKool Smith, P.C. 300 Crescent Court, Suite 1200 Dallas, TX 75201 Tel: (214) 978-4218 Fax: (214) 978-4044 Emily R. Tannenbaum (Reg. No. 80,655) etannenbaum@McKoolSmith.com MCKOOL SMITH, P.C. 1301 Avenue of the Americas, 32nd Floor New York, New York 10019 Telephone: (212) 402-9400 Telecopier: (212) 402-9444 Keith D. Harden (Reg. No. 74,472) kharden@munckwilson.com S. Wallace Dunwoody (Texas Bar No. 24040838 - admission *pro hac vice* to be requested) wdunwoody@munckwilson.com Michael C. Wilson (Texas Bar No. 21704590 - admission *pro hac vice* to be requested) mwilson@munckwilson.com Munck Wilson Mandala, LLP 2000 McKinney Ave., Ste. 1900 Dallas, Texas 75201 Tel: (972) 628-3600 Fax: (972) 628-3616

Petitioners consent to service by email to the counsel above.

D. Service information

USPTO records show the attorneys having power of attorney over the '785 Patent are Volpe Koenig. This petition is thus being served by Federal Express to the correspondence address for the '785 Patent, Volpe Koenig, 30 South 17th Street, 18th Floor, Philadelphia, PA 19103. Petitioners consent to electronic service at AA_Intellectual_Ventures@mckoolsmith.com.

XII. FEE PAYMENT

The Office is allowed to charge the fees specified by 37 C.F.R. §§ 42.103(a) and 42.15(a) to Deposit Account No. 50-5723.

Respectfully submitted,

/s/John B. Campbell

John B. Campbell
Registration No. 54,665
jcampbell@McKoolSmith.com
303 Colorado Street, Suite 2100
Austin, Texas 78701
Tel: (512) 692-8700
Fax: (512) 692-8744

Casey Shomaker (Reg. No. 77,998)
cshomaker@mckoolsmith.com
McKool Smith, P.C.
300 Crescent Court, Suite 1200
Dallas, TX 75201
Tel: (214) 978-4218
Fax: (214) 978-4044

Emily Tannenbaum
Registration No. 80,655
etannenbaum@McKoolSmith.com

MCKOOL SMITH, P.C.
1301 Avenue of the Americas, 32nd Floor
New York, NY 10019
Telephone: (212) 402-9400
Telecopier: (212) 402-9444

COUNSEL FOR PETITIONER AMERICAN AIRLINES, INC.

-and-

Keith D. Harden (Reg. No. 74,472)
kharden@munckwilson.com
S. Wallace Dunwoody (Texas Bar No. 24040838 - admission *pro hac vice* to be requested)
wdunwoody@munckwilson.com
Michael C. Wilson (Texas Bar No. 21704590 - admission *pro hac vice* to be requested)
mwilson@munckwilson.com
Munck Wilson Mandala, LLP
2000 McKinney Ave., Ste. 1900
Dallas, Texas 75201
Tel: (972) 628-3600
Fax: (972) 628-3616

COUNSEL FOR PETITIONER SOUTHWEST AIRLINES CO.

CERTIFICATE OF SERVICE

I hereby certify, pursuant to 37 C.F.R. Sections 42.6 and 42.106, that a complete copy of the attached **PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 7,949,785 B2**, including all exhibits **(Nos. 1001-1028)** and related materials, are being served by Federal Express on 11th day of April, 2025, the same day as the filing of the above-identified document in the United States Patent and Trademark Office/Patent Trial and Appeal Board, upon Patent Owner by serving the correspondence address of record with the USPTO as follows:

Volpe Koenig
30 South 17th Street, 18th Floor
Philadelphia, PA 19103

A courtesy copy was also sent via electronic mail to the Patent Owner's litigation counsel listed below:

Jonathan K. Waldrop
jwaldrop@kasowitz.com
Darcy L. Jones
djones@kasowitz.com
Marcus A. Barber
mbarber@kasowitz.com
John W. Downing
jdowning@kasowitz.com
Heather S. Kim
hkim@kasowitz.com
ThucMinh Nguyen
tnguyen@kasowitz.com
Jonathan H. Hicks
Jhicks@kasowitz.com
KASOWITZ BENSON TORRES LLP
333 Twin Dolphin Drive, Suite 200
Redwood Shores, California 94065
Telephone: (650) 453-5170
Facsimile: (650) 453-5171

Dated: April 11, 2025

/s/ John B. Campbell
John B. Campbell
Reg. No. 54,665

CERTIFICATION OF WORD COUNT

Pursuant to 37 C.F.R. § 42.24(d), I certify that this petition complies with the type-volume limits of 37 C.F.R. § 42.24(a)(1)(i) because it contains 13,996 words, according to the word-processing system used to prepare this petition, excluding parts of this petition that are exempted by 37 C.F.R. § 42.24(a) (including the table of contents, a table of authorities, mandatory notices, a certificate of service or this certificate word count, appendix of exhibits, and claim listings).

Dated: April 11, 2025

MCKOOL SMITH, P.C.
ATTN: John B. Campbell
303 Colorado Street
Suite 2100
Austin, TX 78701
Tel: (512) 692-8730
Fax: (512) 692-8744

By: */s/ John B. Campbell*
John B. Campbell
Reg. No. 54,665