

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

VERIZON SERVICES CORP. AND
VERIZON BUSINESS NETWORK SERVICES INC.
Petitioners

v.

STRAIGHT PATH IP GROUP, INC.
(FORMERLY KNOWN AS INNOVATIVE COMMUNICATIONS
TECHNOLOGIES, INC.)
Patent Owner

INTER PARTES REVIEW OF U.S. PATENT NO. 6,131,121
Case IPR No.: *To Be Assigned*

PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 6,131,121
UNDER 35 U.S.C. §§ 311-319 AND 37 C.F.R. § 42.100 *et seq.*

Filed on behalf of Petitioners

By: Rajeev Gupta, Reg. No. 55,873
Darren M. Jiron, Reg. No. 45,777
**FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.**
901 New York Ave., NW
Washington, D.C. 20001-4413
Telephone: 202-408-4000
Facsimile: 202-408-4400

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
II.	MANDATORY NOTICES	3
	A. Real Party-in-Interest.....	3
	B. Related Matters.....	4
	C. Counsel and Service Information.....	6
III.	CERTIFICATION OF GROUNDS FOR STANDING	7
IV.	OVERVIEW OF CHALLENGE AND RELIEF REQUESTED	7
V.	OVERVIEW OF THE '121 PATENT (EX. 1001).....	8
	A. Summary of the Alleged Invention	8
	1. Step 1: Processing Units Obtain Dynamically Assigned IP Addresses.....	9
	2. Step 2: Processing Units Register Their IP Addresses and Identifiers with a Connection Server	9
	3. Steps 3 & 4: First Processing Unit Sends Query to Connection Server, Which Returns IP Address of Second Processing Unit.....	10
	4. Step 5: First Processing Unit Uses Received IP Address to Establish Point-to-Point Communication with Second Processing Unit.....	11
	B. Original Prosecution of the '121 Patent.....	12
	C. Prior <i>Ex Parte</i> Reexamination of the '121 Patent	12
	D. The Sipnet <i>Inter Partes</i> Review for the '704 Patent (Ex. 1010)	13
VI.	OVERVIEW OF THE PRIMARY PRIOR ART REFERENCES.....	14
	A. WINS (Ex. 1003)	14
	1. Step 1: Processing Units Obtain Dynamically Assigned IP Addresses from DHCP Servers	15
	2. Step 2: Processing Units Register Their IP Addresses and Identifiers with the WINS Server	17

3.	Steps 3 & 4: First Processing Unit Sends Query to WINS Server and Receives the IP Address of the Second Processing Unit.....	21
4.	Step 5: First Processing Unit Uses Received IP Address to Establish Point-to-Point Communication with Second Processing Unit.....	22
B.	NetBIOS (Ex. 1004)	22
1.	Step 1: Processing Units Have Assigned IP Addresses	23
2.	Step 2: Processing Units Register Their IP Addresses and Identifiers with the NBNS.....	24
3.	Steps 3 & 4: First Processing Unit Sends Query to the NBNS and Receives the IP Address of the Second Processing Unit.....	25
4.	Step 5: First Processing Unit Uses Received IP Address to Establish Point-to-Point Communications with Second Processing Unit.....	26
VII.	CLAIM CONSTRUCTION	27
A.	“Point-to-Point Communication” (Claims 3, 4, 10, 13); “Point-to- Point Communication Link” (Claims 6, 8, 9, 10-14); “Point-to-Point Communication Connection” (Claim 7).....	28
B.	“[Program Code for / Program Logic Configured to] Transmitting/Transmit to the Server a Network Protocol Address Received By the First Process Following Connection to the Computer Network” (Claims 12-14).....	29
C.	“Connection to the Computer Network” (Claims 3, 9-14) / “Connected to the Computer Network” (Claims 4, 6-8, 12-14)	29
VIII.	SPECIFIC GROUNDS FOR PETITION	30
A.	Ground I: Claims 3, 4, and 6-14 Would Have Been Obvious Over WINS and NetBIOS.	30
1.	Level of Ordinary Skill in the Art.....	30
2.	One Skilled in the Art Would Have Been Motivated to Combine WINS and NetBIOS.	31
3.	Claim 3 (Independent) Should Be Cancelled.....	32
4.	Claim 4 (Depends From Claim 3) Should Be Cancelled.	39

5.	Claim 6 (Independent) Should Be Cancelled.....	43
6.	Claim 7 (Independent) Should Be Cancelled.....	46
7.	Claim 8 (Independent) Should Be Cancelled.....	47
8.	Claim 9 (Independent) Should Be Cancelled.....	49
9.	Claim 10 (Independent) Should Be Cancelled.....	51
10.	Claim 11 (Independent) Should Be Cancelled.....	52
11.	Claim 12 (Independent) Should Be Cancelled.....	53
12.	Claim 13 (Independent) Should Be Cancelled.....	54
13.	Claim 14 (Independent) Should Be Cancelled.....	55
B.	Examples Where Prior Art Teaches Claimed Features	56
IX.	CONCLUSION.....	60

TABLE OF AUTHORITIES

	Page(s)
Federal Cases	
<i>In re ICON Health & Fitness, Inc.</i> , 496 F.3d 1374 (Fed. Cir. 2007)	27
<i>In re Petrus A.C.M. Nuijten</i> , 500 F.3d 1346 (Fed. Cir. 2007)	46
<i>Phillips v. AWH Corp.</i> , 415 F.3d 1303. (Fed. Cir. 2005) (en banc)	27, 28, 29, 30
Commission Cases	
<i>Certain Point-to-Point Network Communication Devices and Products Containing Same</i> , Inv. No. 337-TA-892 (U.S.I.T.C.)	30
Federal Statutes	
35 U.S.C. §§ 102(a) and (b)	6, 7
35 U.S.C. § 102(b)	7, 12
35 U.S.C. § 103	6
35 U.S.C. § 103(a)	12, 30
35 U.S.C. § 314(a)	6
35 U.S.C. § 315	4
35 U.S.C. § 315(c)	1
35 U.S.C. § 325	4
Rules	
Rule 42.22(a)	6
Rule 42.104(a)	6
Rule 42.104(b)(1)-(2)	6

Rule 42.104(b)(4)-(5)	30
-----------------------------	----

Regulations

37 C.F.R. § 42.8(b)(1)	3
37 C.F.R. § 42.100(b)	27
37 C.F.R. § 42.122(b)	1
77 Fed. Reg. 48764 (Aug. 14, 2012)	27

Other Authorities

<i>Microsoft Windows NT Networking Guide</i>	8
<i>Technical Standard - Protocols for X/Open PC Interworking: SMB, Version 2</i>	8
United States International Trade Commission Investigation No. 337- TA-892	29

I. INTRODUCTION

Petitioners Verizon Services Corp. and Verizon Business Network Services Inc. submit this petition for *Inter Partes* Review of U.S. Patent No. 6,131,121 (“the ’121 patent”) (Ex. 1001)¹ concurrently with a Motion for Joinder, under 35 U.S.C. § 315(c) and 37 C.F.R. § 42.122(b), with a pending IPR, *LG Elecs. Inc. v. Straight Path IP Grp., Inc.*, IPR2015-00196, instituted on May 15, 2015.

The ’121 patent is directed to establishing “point-to-point communications” between two processes (e.g., computers) over a computer network. The ’121 inventors did not claim to invent point-to-point communications, which they conceded were already “known in the art.” Instead, they alleged that prior art point-to-point communications were “impractical” when the initiating process did not know the specific network address of the other process; for example, in the case of processes with dynamically assigned addresses that can change over time.

To address that alleged “problem,” the ’121 inventors disclosed a simple look-up feature involving a “server” that tracks the currently assigned network address and other identifying information (e.g., name) of registered processes. In response to a query received from a first process (e.g., using the name of a second process), the server sends the current network address of the second process to the first process,

¹ Petitioners have numbered each page of the Exhibits. All citations are to the numbers added by Petitioners.

and the first process then uses that retrieved address to establish a so-called point-to-point communication with the second process.

But by September 1995 (the claimed priority date of the '121 patent), others had **solved** the **same** problem using the **same** basic lookup feature. For example, in 1994, Microsoft published a user's manual for Version 3.5 of its Windows NT Server software ("WINS") (Ex. 1003). Just like the '121 patent, WINS teaches (1) a name server (WINS implements the NetBIOS protocol) that tracks the current dynamically assigned network address and name of each registered process; and (2) a first process that sends a name query for a second process to the name server, and then uses the network address received in response to the query to establish point-to-point communications with the second process. In fact, on October 11, 2013, the Board instituted *inter partes* review of claims 1-7 and 33-42 of the parent of the '121 patent, U.S. Patent No. 6,108,704 ("the '704 patent"), based on the same WINS reference (Exhibit 1003), and the NetBIOS Technical Standard ("NetBIOS") (Exhibit 1004). *See* Sipnet EU S.R.O. v. Straight Path IP Group, IPR No. 2013-00246. (Ex. 1011.)² And on October 9, 2014, the Board determined that Sipnet had shown by a preponderance

² Although the Sipnet IPR petitioners treated the WINS and NetBIOS references as separate, they can be treated as a single reference because WINS explicitly incorporates the NetBIOS protocol, *see infra* § VI(A). For purposes of this petition, however, Petitioners treat these references as an obviousness combination.

of the evidence that: (i) WINS anticipates claims 1-7 and 32-42 of the '704 patent; (ii) NetBIOS anticipates claims 1-7, 32, and 38-42; and (iii) NetBIOS and WINS render obvious claims 33-37—confirming that WINS and NetBIOS solve the same problem using the same basic features as the '704 and '121 patents. (Ex. 1028 at 25.)

The claims challenged by Petitioners share many limitations with claim 1 of the '704 patent that the Sipnet IPR panel found to be anticipated by both NetBIOS and WINS. For efficiency and consistency, Petitioners ask the Board to assign the Sipnet panel to this petition. As detailed below, WINS and NetBIOS (Ex. 1004) render obvious claims 3, 4, and 6-14 of the '121 patent.

II. MANDATORY NOTICES

A. Real Party-in-Interest

Pursuant to 37 C.F.R. § 42.8(b)(1), Petitioners identify Verizon Services Corp. and Verizon Business Network Services Inc. as the real parties-in-interest. Additionally, Petitioners, out of an abundance of caution in light of prior challenges to the named real parties-in-interest in separate and unrelated IPR petitions, identify Verizon Corporate Resources Group, LLC and Verizon Communications Inc. as real parties-in-interest for the IPR requested by this Petition solely to the extent that Patent Owner contends that these separate legal entities should be named real parties-in-interest in the requested IPR, and Petitioners do so to avoid the potential expenditure of resources to resolve such a challenge. No unnamed entity is funding, controlling, or otherwise has an opportunity to control or direct this Petition or any of

the Petitioners' participation in any resulting IPR. Also, Petitioners note that Verizon Communications Inc. has over 500 affiliated entities; each entity agrees to be estopped under the provisions of 35 U.S.C. §§ 315 and/or 325 as a result of any final written decision in the requested IPR to the same extent that Petitioners are estopped.

B. Related Matters

The following would affect or be affected by a decision in this proceeding:

(1) Petitioners' related *inter partes* review petition contesting the validity of claims 1, 11-12, 14, 16, 19, 22-23, 27, and 30-31 of the '704 patent.

(2) *LG Elecs., Inc., et al. v. Straight Path IP Grp., Inc.* seeking review of the '469 (IPR2015-00198), the '121 (IPR2015-00196), and the '704 (IPR2015-00209) patents (all filed August 1, 2014, and awaiting institution decision) (the "LG/Hulu IPRs").

(3) *Samsung Elecs. Co., Ltd. et al. v. Straight Path IP Grp., Inc.* reviewing '704 patent claims 1, 11-12, 14, 16, 22-23, 27, and 30-31 (IPR2014-01366); '469 patent claims 1-3, 5-6, 9-10, 14, and 17-18 (IPR2014-01367); and '121 patent claims 6, 8, 10-11, and 13-14 (IPR2014-01368) (all instituted March 6, 2015).

(4) *Straight Path IP Grp., Inc. v. Sipnet EU S.R.O.*, No. 15-1212 (Fed. Cir.), appealing the Sipnet IPR Panel decision in IPR2013-00246.

(5) *Sony Corp., et al. v. Straight Path IP Grp., Inc.* seeking review of the '121 patent (IPR2013-00229), the '469 patent (IPR2014-00231), and the '704 patent (IPR2014-00230) (all filed Dec. 5, 2013, terminated on May 2, 2014 on joint motions after Patent Owner's preliminary response, but prior to an institution decision).

(6) *Netflix, Inc., et al. v. Straight Path IP Grp., Inc.* seeking review of certain claims of the '704 patent (IPR2014-01241) (filed August 1, 2014, terminated October 30, 2014 on a joint motion prior to Patent Owner's preliminary response).

(7) *Vonage Holdings Corp., et al. v. Straight Path IP Grp., Inc.* seeking review of certain claims of the '469 patent (IPR2014-01225); the '121 patent (IPR2014-01234); the '365 patent (IPR2014-01224); and the related U.S. Patent No. 6,513,066 (IPR2014-01223) (all filed Aug. 1, 2014, terminated Oct. 30, 2014 on joint motions prior to Patent Owner's preliminary response).

(8) Actions in which Straight Path (or one of its predecessors-in-interest) has asserted the '121 patent, including *Straight Path IP Grp., Inc. v. Verizon Comm'ns, Inc. et al.*, 1-14-cv-07798 (S.D. N.Y.); *Straight Path IP Grp., Inc. v. Cisco Systems, Inc.*, No. 3:14-cv-04312 (N.D. Cal.); *Straight Path IP Grp., Inc. v. AVAYA Inc.*, No. 3:14-cv-04309 (N.D. Cal.); *Straight Path IP Grp., Inc. v. Apple Inc.*, 3-14-cv-04302 (N.D. Cal.); *Straight Path IP Grp., Inc. v. Samsung Elecs. Co., Ltd. et al.*, 6-13-cv-00606 (E.D. Tex.); *Straight Path IP Grp., Inc. v. BlackBerry Ltd. et al.*, 6-14-cv-00534 (E.D. Tex.); *Straight Path IP Grp., Inc. v. Netflix, Inc.*, 6-14-cv-00405 (E.D. Tex.); *Straight Path IP Grp., Inc. v. ZTE Corp. et al.*, 6-13-cv-00607 (E.D. Tex.); *Straight Path IP Grp., Inc. v. Huawei Inv. & Holding Co., Ltd. et al.*, 6-13-cv-00605 (E.D. Tex.); *Straight Path IP Grp., Inc. v. BlackBerry Ltd. et al.*, 6-13-cv-00604 (E.D. Tex.); *Straight Path IP Grp., Inc. v. Toshiba Corp. et al.*, 1-13-cv-01070 (E.D. Va.); *Straight Path IP Grp., Inc. v. Toshiba Corp. et al.*, 3-13-cv-00503 (E.D. Va.); *Straight Path IP Grp., Inc. v. Panasonic Corp. of N. Am. et al.*, 1-13-cv-00935

(E.D. Va.); *Straight Path IP Grp., Inc. v. Sharp Corp. et al.*, 1-13-cv-00936 (E.D. Va.); *Straight Path IP Grp., Inc. v. LG Elecs., Inc. et al.*, 1-13-cv-00933 (E.D. Va.); *Straight Path IP Grp., Inc. v. Sony Corp. et al.*, 2-13-cv-00427 (E.D.Va.); *Straight Path IP Grp., Inc. v. Vizio, Inc. et al.*, 1-13-cv-00934 (E.D.Va.); *Straight Path IP Grp., Inc. v. Sony Corp. et al.*, 1-13-cv-01071 (E.D. Va.); *Innovative Comm’ns Techs., Inc. v. Vivox, Inc.*, 2-12-cv-00007 (E.D. Va.); *Innovative Comm’ns Techs., Inc. v. Stalker Software, Inc. et al.*, 2-12-cv-00009 (E.D. Va.); *Net2phone, Inc. v. Ebay, Inc. et al.*, 2-06-cv-02469 (D.N.J.); *Net2phone, Inc. v. Ebay, Inc., et al.*, 4-10-cv-04090 (W.D. Ark.); and *Point-to-Point Network Communication Devices and Products Containing Same*, Inv. 337-TA-892 (I.T.C.).

(9) Because this petition and Petitioners’ related petition are substantively identical to the petitions underlying the LG/Hulu IPRs, Petitioners are filing Motions for Joinder with the LG/Hulu IPRs. In the alternative, Petitioners request that, for efficiency and consistency, the panel assigned to the LG/Hulu IPRs also be assigned to address this petition and Petitioner’s related petition; or, in the alternative, that the same panel be assigned to this petition and Petitioners’ related petition.

C. Counsel and Service Information

Lead Counsel	Rajeev Gupta (Registration No. 55,873)
Backup Counsel	Darren M. Jiron (Registration No. 45,777)
Service Information	Email:raj.gupta@finnegan.com darren.jiron@finnegan.com Post and hand delivery: FINNEGAN, LLP 901 New York Ave., NW Washington, D.C. 20001 Telephone: 202-408-4000 Facsimile: 202-408-4400

Powers of attorney are submitted with this Petition. Counsel for Petitioners consent to service of all documents via electronic mail.

III. CERTIFICATION OF GROUNDS FOR STANDING

Petitioners certify under Rule 42.104(a) that the '121 patent is available for *inter partes* review and Petitioners are not barred or estopped from requesting *inter partes* review challenging the claims on the grounds identified in this Petition.

IV. OVERVIEW OF CHALLENGE AND RELIEF REQUESTED

Petitioners request cancellation of claims 3, 4, and 6-14 of the '121 patent (“the challenged claims”) as unpatentable under 35 U.S.C. § 103. This Petition, supported by the Declaration of Dr. Bruce M. Maggs (Ex. 1002), demonstrates a reasonable likelihood that the challenged claims are not patentable and that Petitioners will prevail with respect to at least one of them. *See* 35 U.S.C. § 314(a).

Pursuant to Rules 42.22(a) and 42.104(b)(1)-(2), Petitioners’ challenge is based on the following references:

1. WINS (Ex. 1003), which Microsoft Corporation published and publicly distributed to customers no later than September 1994, is prior art under at least 35 U.S.C. §§ 102(a) and (b).³ The September 1994 publication date for WINS is further confirmed, for example, by: (1) Exhibit 1007, a copyright registration notice that lists

³ WINS was before the Board as Exhibit 1004 in the Sipnet IPR (discussed below in Section V(D)), and the Board found WINS to be prior art. Ex. 1028 at 20.

September 19, 1994, as the date of first publication for “Microsoft Windows NT Server, Version 3.5”; (2) Exhibit 1006, a printout of the “TCPIP.HLP” file (bearing a “Date modified” of September 4, 1994) that was distributed with Microsoft Windows NT Server 3.5, bears a 1994 copyright date, and is substantively identical to WINS (except it does not include the “Glossary”); and (3) Exhibit 1008, a book titled *Microsoft Windows NT Networking Guide* that contains the relevant portions (except the “Welcome” and “Glossary” sections) of WINS and was first published in February 1995, as confirmed by the copyright registration notice (Exhibit 1009). The “TCPIP.HLP” file is prior art under at least 35 U.S.C. §§ 102(a) and (b), and the *Microsoft Windows NT Networking Guide* is prior art under at least 35 U.S.C. § 102(b).

2. *Technical Standard - Protocols for X/Open PC Interworking: SMB, Version 2*, including Appendices F and G (RFC Nos. 1001 and 1002) (Ex. 1004, “NetBIOS”) was published and made publicly available in September 1992, and is prior art under at least 35 U.S.C. §§ 102(a) and (b).

V. OVERVIEW OF THE '121 PATENT (EX. 1001)

A. Summary of the Alleged Invention

The '121 patent concedes that, in the prior art, a “first processing unit” or “process”⁴ could establish “point-to-point communications” with a second processing unit using the network address of the second processing unit “in a manner known in

⁴ For convenience, the term “processing unit” is used in Sections V and VI.

the art.” (Ex. 1001, 2:6-8 (“[D]evices interfacing to the Internet and other online services may communicate with each other upon establishing respective device addresses.”); *id.*, 2:32-34, 9:47-51, 10:5-9 (“[P]oint-to-point communications of voice and video signals over the Internet” “may be established as shown in FIGS. 3-4 in a manner known in the art” and “may be conducted in a manner known in the art between the first and second users through the Internet 24”); Ex. 1002 ¶ 34.)

According to the ’121 patent, however, point-to-point communication was “difficult to attain” between devices with “temporary IP addresses” (i.e., dynamically assigned IP addresses that “may be reassigned or recycled” over time). (Ex. 1001, 2:25-46, 2:58-61.) The ’121 patent claimed to solve that supposed “problem” through the basic lookup feature described in connection with Figure 8. (*Id.*, Fig. 8; Ex. 1002 ¶ 35.)

1. Step 1: Processing Units Obtain Dynamically Assigned IP Addresses

When a processing unit “logs on to the Internet . . . , [it] is provided a dynamically allocated IP address by an Internet service provider.” (Ex. 1001, 5:51, 6:53-59, 13:2-5, 13:59-60, 15:27-32, 18:10-14, Fig. 13A; Ex. 1002 ¶ 36.)

2. Step 2: Processing Units Register Their IP Addresses and Identifiers with a Connection Server

After receiving its address, a processing unit “automatically transmits . . . its dynamically allocated IP address to the connection server 26,” which “stores these addresses in the database 34 . . . ” (Ex. 1001, 6:53-66; *id.*, 12:9-12 (“[C]onnection

server 26 . . . timestamps and stores E-mail and IP addresses of logged-in users and processing units in the database 34.”); *id.*, 12:41-46 (connection server provides “directory and information related services.”); *id.*, 18:19-29 (“one-to-one mapping” of “identifier” and “current IP address” for each “client process”); *id.*, 22:50-23:40 (describing “start up”); Ex. 1020 [9/7/99 Amend.] at 19; Ex. 1002 ¶ 37.)

Connection server 26 keeps “relatively current” data concerning the “on-line status” of registered processing units; e.g., it may confirm a registered processing unit remains online after “predetermined time periods, such as a default value of 24 hours....” (Ex. 1001, 7:8-13.) Alternatively, “[w]hen a user logs off or goes offline from the Internet 24, the connection server 26 updates the status of the user in the database 34; for example, by removing the user’s information, or by flagging the user as being offline.” (*Id.*, 7:43-46; Ex. 1002 ¶ 38.)

3. Steps 3 & 4: First Processing Unit Sends Query to Connection Server, Which Returns IP Address of Second Processing Unit

To establish point-to-point communications with a second processing unit, the first processing unit “sends a query . . . to the connection server 26” that includes the e-mail address or “name or alias . . . of a party to be called.” (Ex. 1001, 7:24-29, 11:13-19, 11:59-63, 12:12-16; Ex. 1020 [9/7/99 Amend.] at 19-20.) Connection server 26 then “searches the database 34 to determine whether the [second processing unit] is logged-in by finding any stored information . . . indicating that the [second processing unit] is active and on-line.” (Ex. 1001, 7:2433 (“If the [second processing unit] is

active and on-line . . . the IP address of the [second processing unit] is retrieved from the database 34 and sent to the first [processing unit].”); *id.*, 12:17-20 (“Connection server 26 retrieves the IP address of the specified user from the database 34 . . . and sends the retrieved IP address to the first processing unit 12.”); *id.*, 18:30-19:9, 23:56-24:10 (server looks-up and returns requested IP addresses); *id.*, 23:45-52 (user inputs “callee party’s name or alias” “to obtain the current dynamically assigned Internet Protocol address of the prospective callee”); Ex. 1020 [9/7/99 Amend.] at 20; Ex. 1002 ¶ 39.)

4. Step 5: First Processing Unit Uses Received IP Address to Establish Point-to-Point Communication with Second Processing Unit

After receiving the IP address of the second processing unit from connection server 26, “[t]he first processing unit 12 may then directly establish the point-to-point Internet communications with the [second processing unit] using the IP address of the [second processing unit].” (Ex. 1001, 7:33-36; *id.*, 12:16-21 (received IP address “enable[s] first processing unit to establish point-to-point communications with specified second user.”); Ex. 1002 ¶ 40.)

The ’121 patent does not claim to invent point-to-point communications, or even a new type of point-to-point communications. Rather, it admits the claimed point-to-point communications “may be established as shown in FIGS. 3-4 *in a manner known in the art*” and “*may be conducted in a manner known in the art* between the first and second users through the Internet 24.” (Ex. 1001, 9:47-51,

10:5-9 (emphases added).) The claimed invention also functions the same regardless of “whether the current IP addresses were permanent (i.e. predetermined or preassigned) or temporary (i.e. assigned upon initiation of the point-to-point communication).” (*Id.*, 9:25-28; Ex. 1002 ¶ 41.)

B. Original Prosecution of the ’121 Patent

During prosecution of the ’121 patent, the Examiner rejected all claims, but later issued the ’121 patent after the patentee cancelled claims and amended and added others. (Ex. 1026 [3/4/99 Non-Final Rejection]; Ex. 1020 [9/7/99 Amend.]; Ex. 1023 [8/23/00 Notice of Allowability].) This Petition does not rely on any prior art cited during that prosecution. (Ex. 1002 ¶ 42.)

C. Prior *Ex Parte* Reexamination of the ’121 Patent

In February 2009, a third party requested *ex parte* reexamination of claims 6-14 of the ’121 patent. During that proceeding, the Examiner rejected claims 6-11 under 35 U.S.C. § 102(b) in light of NetBIOS, and rejected claims 12-14 under 35 U.S.C. § 103(a) in light of NetBIOS and RFC 1531 (“Dynamic Host Configuration Protocol” (DHCP)) (Ex. 1027 [8/25/09 Office Action].) The Examiner agreed that NetBIOS describes the same address lookup function claimed in the ’121 patent, but was persuaded by an expert declaration arguing that “bringing dynamic addressing into a NetBIOS type system would create a new set of obstacles that would need to be solved” such that “one of ordinary skill in the art would [not] have been motivated to combine NetBIOS and [DHCP].” (Ex. 1021 [5/17/10 Office Action]; Ex. 1024

[11/25/09 Response]; Ex. 1002 ¶ 43.) During the reexamination, the Patentee amended claims 6-8 to require “dynamically assigned” network protocol addresses. (Ex. 1022 [7/19/10 Amend.] at 2-3.)

The expert declaration failed to note that prior art, including WINS, disclosed using dynamic addressing in a NetBIOS-type system. (*See, e.g.*, Ex. 1003 at 13 (DHCP servers allow “users [to] take advantage of dynamic IP address allocation and management.”); *id.* at 62 (“DHCP offers dynamic configuration of IP addresses for computers.”); *id.* at 41 (“Microsoft Windows networking provides dynamic name resolution for NetBIOS computer names via WINS servers and NetBIOS over TCP/IP.”); *id.* at 67 (“All computers register themselves with the WINS server, which is a NetBIOS Name Server (NBNS) with enhancements.”); Ex. 1002 ¶ 44.)

Relying on the flawed declaration, the Examiner confirmed claims 12-14, and allowed amended versions of claims 6-11, explicitly limited to “dynamically assigned” addresses. (Ex. 1021 [5/17/10 Office Action]; Ex. 1025 [8/16/10 Office Action]; Ex. 1022 [7/19/10 Amend.] at 2-3; Ex. 1002 ¶ 45.)

D. The Sipnet *Inter Partes* Review for the ’704 Patent (Ex. 1010)

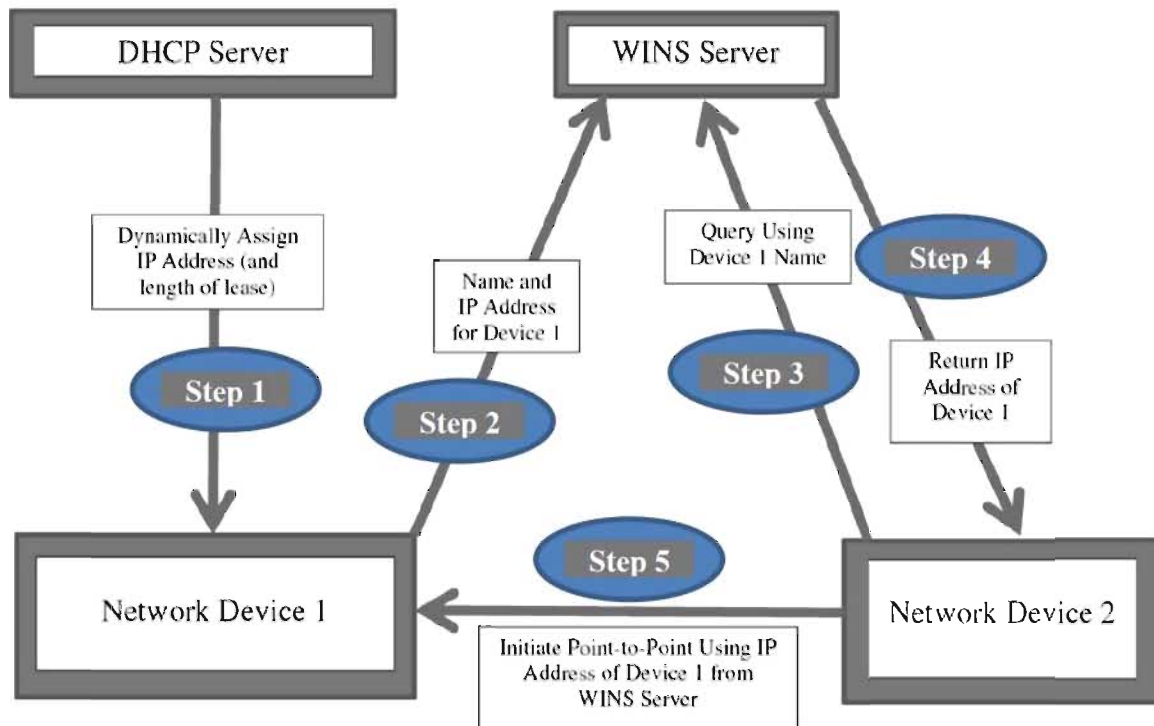
In April 2013, a third party (“Sipnet”) initiated an IPR seeking to cancel claims of the ’704 patent (parent of the ’121 patent), based on, *inter alia*, WINS and NetBIOS. (Ex. 1010.) On October 11, 2013, the Board instituted an IPR for all challenged claims and found a reasonable likelihood that WINS and NetBIOS each anticipated most claims and rendered the others obvious. (Ex. 1011 at 20-21.) On

October 9, 2014, the Board determined that Sipnet had shown by a preponderance of the evidence that claims 1-7 and 32-42 of the '704 patent are unpatentable. (Ex. 1028.)

VI. OVERVIEW OF THE PRIMARY PRIOR ART REFERENCES

A. WINS (Ex. 1003)

In 1994, Microsoft published and publicly distributed WINS, which “describes how to install, configure, and troubleshoot Microsoft TCP/IP on a computer running the Microsoft Windows NT Workstation or Windows NT Server operating system,” including “the software to support new dynamic configuration and name resolution services.” (Ex. 1003 at 12.) WINS teaches how a processing unit (e.g., a computer running the Windows NT or Windows for Workgroups operating system) or process (e.g., the Windows NT or Windows for Workgroups operating system running on a computer) can initiate point-to-point communications with other processing units or processes using the same lookup feature as in the '121 patent:



1. Step 1: Processing Units Obtain Dynamically Assigned IP Addresses from DHCP Servers

WINS explains that each processing unit participating on a Windows NT network “must be assigned a unique IP address,” and that Windows NT networks can use a DHCP server to assign IP addresses “automatically” to processing units at start up—including dynamically. (Ex. 1003 at 57-58, 62 (manually “[a]ssigning and maintaining IP address information can be an administrative burden” and “[DHCP] was established to relieve this administrative burden . . . through centralized management of address allocation” and “dynamic configuration of IP addresses for computers.”); *id.* at 13 (“When DHCP servers are installed on the network, users can take advantage of dynamic IP address allocation and management.”); *id.* at 23 (Windows TCP/IP supports RFC 1541, the “Dynamic Host Configuration Protocol

(DHCP)’’); *id.* at 83 (‘‘A [DHCP] server is a Windows NT Server computer running . . . Microsoft TCP/IP and the DHCP-compatible server software’’ which ‘‘is defined in [RFCs] 1533, 1534, 1541, and 1542.’’); *id.* at 85-120 (DHCP options, including for ‘‘small LAN’’ and ‘‘large enterprise network’’); *id.* at 113 (‘‘Allocation of IP addresses for distribution by DHCP servers can be done dynamically or manually Dynamic allocation allows a client to be assigned an IP address from the free address pool.’’); Ex. 1002 ¶ 48.)

‘‘During system startup,’’ each processing unit wishing to participate on the network sends a ‘‘*discover message*’’ to all DHCP servers, each of which ‘‘responds with an *offer message* containing an IP address and valid configuration information for the client that sent the request.’’ (Ex. 1003 at 63.) After the processing unit chooses one of the DHCP server offers, ‘‘[t]he selected DHCP server sends a *DHCP acknowledgment message* that contains the [IP] address first sent during the discovery stage, plus a valid lease for the address and the TCP/IP network configuration parameters for the client.’’ (*Id.* at 64; *id.* at 84 (‘‘When a DHCP client computer is started, it communicates with a DHCP server to receive the required TCP/IP configuration information . . . includ[ing] at least an IP address and submask plus the lease associated with the configuration.’’); Ex. 1002 ¶ 49.)

The registered processing unit can then participate on the network using the dynamically assigned IP address until either the lease expires or the processing unit leaves the network. (Ex. 1003 at 62 (assigned IP address ‘‘is released automatically for

a DHCP client computer that is removed from a subnet”); *id.* (“The system administrator controls how IP addresses are assigned by specifying *lease* durations, which specify how long a computer can use an assigned IP address before having to renew the lease with the DHCP server.”); *id.* at 63 (“DHCP uses a client-server model and is based on leases for IP addresses.”); *id.* at 64 (“As the lease approaches its expiration date, [the device] attempts to renew its lease with the DHCP server, and is assigned a new address if the current IP address lease cannot be renewed.”); *id.* at 113 (“[D]ynamically allocated addresses can be returned to the free address pool if the client computer is not being used, if it is moved to another subnet, or if its lease expires.”); *id.* at 114 (devices can renew their leases, or if that fails, obtain a new IP address).) The lease period can either be “short” (e.g., “a few days”) or “long” (e.g., “two months”). (*Id.* at 115; Ex. 1002 ¶ 50.)

2. Step 2: Processing Units Register Their IP Addresses and Identifiers with the WINS Server

WINS explains that, “[a]lthough TCP/IP uses IP addresses to identify and reach computers, users typically prefer to use computer names.” (Ex. 1003 at 41; *id.* at 65 (“Computers use IP addresses to identify each other, but users usually find it easier to work with computer names.”).) Thus, WINS teaches that Windows NT requires that each device have a “NetBIOS” name. (*Id.* at 64 (“Configuring Windows NT with TCP/IP requires the IP address and computer name, which are unique identifiers for the computer on the network.”); *id.* at 74 (“Name registration ensures that the

computer's name and IP address are unique for each device.”); *id.* at 262 (“In Windows NT, the computer name is set by choosing the Network icon in Control Panel, and it is a name of up to 15 uppercase characters . . .”); Ex. 1002 ¶ 51.)

WINS further teaches that Windows NT can use Windows Internet Name Service (WINS) servers—which consist of “a Windows NT Server computer running Microsoft TCP/IP and the [WINS] server software” and “contain a dynamic database mapping computer names to IP addresses” in accordance with “a NetBIOS over TCP/IP mode of operation defined in RFC 1001/1002 as p- node.” (Ex. 1003 at 65, 121; *id.* at 151 (WINS database entry including “NetBIOS computer name” and “assigned Internet Protocol address”); *id.* at 20 (“Microsoft TCP/IP includes . . . NetBIOS for establishing logical names and sessions on the network” and “Windows Internet Name Service (WINS) for dynamically registering and querying computer names on an internetwork....”); *id.* at 21 (WINS and NetBIOS services are “[c]ore” services “Integrated with Windows NT”); *id.* at 22 (listing RFC 1001 and 1002 as “NetBIOS Service Protocols” supported by Windows NT); *id.* at 41 (“Microsoft Windows networking provides dynamic name resolution for NetBIOS computer names via WINS servers and NetBIOS over TCP/IP.”); *id.* at 66 (“NetBIOS over TCP/IP is the session-layer network service that performs name-to-IP address mapping for name resolution.”); *id.* at 67 (“The WINS server is responsible for knowing computer names and addresses and for ensuring no duplicate names exist on the network.”); *id.* at 69 (“The WINS protocol is based on and is compatible with the

protocols defined for NBNS in RFCs 1001/1002 ..”); *id.* at 265 (“[A] WINS server can provide name registration services” and is “implemented under RFC 1001/1002 to provide name resolution services for NetBIOS computer names.”); Ex. 1002 ¶ 52.)⁵

WINS teaches that this mapping of names to IP addresses “provides a powerful, new name resolution service for easy, centralized management of computer name-to-IP address resolution in medium and large internetworks.” (Ex. 1003 at 13; *id.* at 69 (“WINS provides a distributed database for registering and querying dynamic computer name-to-IP address mappings in a routed network environment.”); *id.* at 266, 268 (“WINS” is “[a] name resolution service that resolves Windows networking computer names to IP addresses in a routed environment” and “p-node” is “[a] NetBIOS over TCP/IP mode that uses point-to-point communications with a name server to resolve computer names as addresses”); Ex. 1002 ¶ 53.)

WINS also explains that “WINS name resolution is enabled and configured automatically for a computer that is configured with DHCP.” (Ex. 1003 at 40.) After the DHCP server dynamically assigns an IP address, the processing unit transmits its assigned IP address and device name to the WINS server as part of the system startup process. (*Id.* at 65 (“A mechanism must be available on a TCP/IP network to resolve

⁵ WINS teaches that Windows NT also can be used with a Domain Name System (DNS) server, which (unlike WINS) uses “static configuration for computer name-to-IP address mapping.” (Ex. 1003 at 78, 80; Ex. 1002 ¶ 52.)

names to IP addresses. To ensure that both name and address are unique, the Windows NT computer using TCP/IP registers its name and IP address on the network during system startup.”); *id.* at 66 (“When WINS servers are in place on the network, NetBIOS over TCP/IP resolves names on a client computer by communicating with the WINS server.”); *id.* at 67 (“In a p-node environment . . . [a]ll computers register themselves with the WINS server, which is a NetBIOS Name Server (NBNS) with enhancements.”); *id.* at 69 (“WINS provides a distributed database for registering and querying dynamic computer name-to-IP address mappings in a routed network environment.”); *id.* at 72 (“During TCP/IP configuration, the computer’s name is registered with the WINS server, and the IP address of the WINS server . . .”); *id.* at 74 (“The name registration request is sent directly to the WINS server to be added to the database. [If the name is not already in use] WINS accepts the entry and adds it to its local database . . . together with a timestamp, an incremental unique version number, and other information.”); Ex. 1002 ¶ 54.)

WINS also teaches that the WINS/NetBIOS server keeps its mapping of names to IP addresses current to track which users are online. For example, it states that “when dynamic addressing through DHCP results in new IP addresses for computers that move between subnets, the changes are automatically updated in the WINS database.” (Ex. 1003 at 69.) It notes that “[w]henever a computer is shut down properly, it releases its name to the WINS server, which marks the related database

entry as *released* . . . because it knows that the old client is no longer using that name.” (*Id.* at 75.) And “[to] provide[] registration reliability through periodic reregistering of names with the WINS servers,” if a name is not re-registered within the set renewal time, “the WINS server will mark the name as released and available for use.” (*Id.*; Ex. 1002 ¶ 55.)

3. Steps 3 & 4: First Processing Unit Sends Query to WINS Server and Receives the IP Address of the Second Processing Unit

WINS explains that “WINS consists of two components: the WINS server, which handles name queries and registrations, and the client software, which queries for computer name resolution.” (Ex. 1003 at 69; *id.* at 70 (“WINS-enabled computers . . . access the WINS server directly....”); Ex. 1002 ¶ 56.) It further teaches a first processing unit (e.g., a first computer running the Windows NT or Windows for Workgroups operating system) obtains the IP address of a second processing unit (e.g., a second computer running the Windows NT or Windows for Workgroups operating system) by transmitting a “name query request” to the WINS/NetBIOS server, which returns the IP “address mapping” of the second processing unit to the first processing unit. (Ex. 1003 at 72-73 (“[N]ame-to-IP address mapping registered with a WINS server can be provided reliably as a response to a name query.”); *id.* at 67 (“[W]hen NT_PC1 wants to communicate with NT_PC2, it queries the WINS server for the address of NT_PC2 . . . [and] gets the appropriate address from the WINS server.”); *id.* at 122 (WINS servers “support computer name registration and

name resolution,” provide “dynamic name services” in “a NetBIOS namespace,” offer “[c]entralized management of the computer name database,” and “allow[] client computers to easily locate remote systems across local or wide area networks.”); Ex. 1002 ¶ 56.)

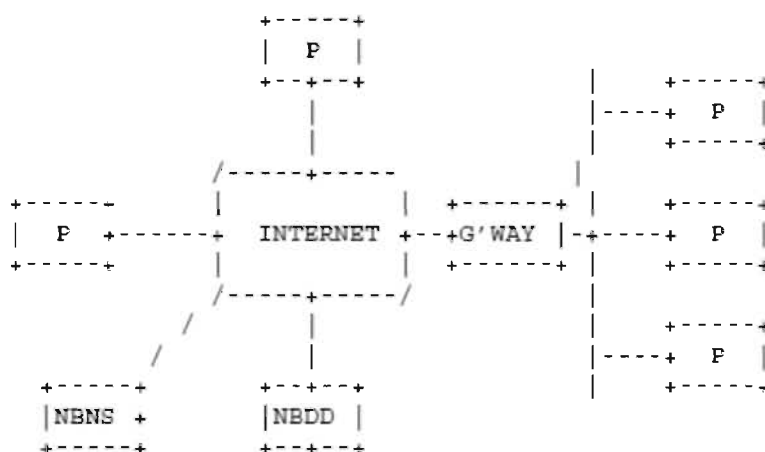
4. Step 5: First Processing Unit Uses Received IP Address to Establish Point-to-Point Communication with Second Processing Unit

The first processing unit can then establish point-to-point communications with the second processing unit by using the IP address received from the WINS/NetBIOS server. (Ex. 1003 at 67 (“[W]hen NT_PC1 wants to communicate with NT_PC2, it queries the WINS server for the address of NT_PC2. When NT_PC1 gets the appropriate address from the WINS server, it goes directly to NT_PC2 without broadcasting.”); *id.* at 266 (“p-node” is “[a] NetBIOS over TCP/IP mode that uses point-to-point communications with a name server to resolve computer names as addresses.”); *id.* at 73 (“If the name is found in the WINS database, the client can establish a session based on the address mapping received from WINS.”); *id.* at 121 (“WINS servers maintain a database that maps computer names to IP addresses, allowing users to easily communicate with other computers while gaining all the benefits of TCP/IP.”); Ex. 1002 ¶ 57.)

B. NetBIOS (Ex. 1004)

RFC 1001 “describes the ideas and general methods used to provide NetBIOS on a TCP and UDP foundation,” and RFC 1002 “contains the detailed packet formats

and protocol specifications for NetBIOS-over-TCP.” (Ex. 1004 at 368, 374, 442.) In 1992, RFCs 1001 and 1002 were published as Appendices F and G to the book attached as Exhibit 1004 (“NetBIOS”). NetBIOS teaches point-to-point communications over a network, including between “[p]oint-to-point (or ‘P’) nodes” using a look-up service called a NetBIOS Name Server (NBNS). (Ex. 1004 at 384-85.) For example, the figure below shows “P”-nodes connected to the “Internet” (2 directly, and 3 through a gateway (“G’WAY”)), each of which can communicate with a NBNS that also is connected to the Internet:



(Ex. 1004 at 389; Ex. 1002 ¶¶ 58-59.)

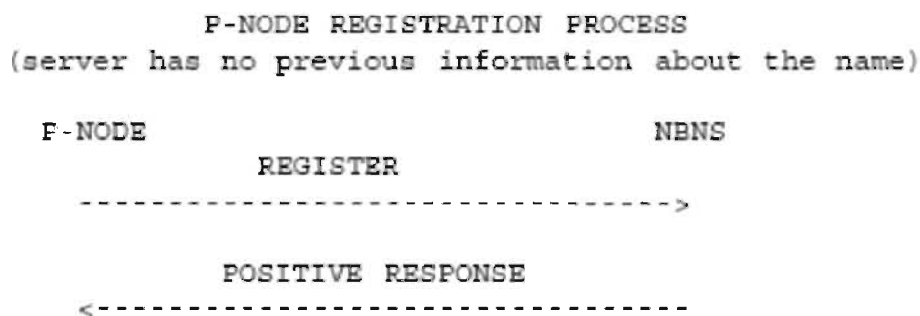
1. Step 1: Processing Units Have Assigned IP Addresses

NetBIOS explains that “[e]very node has a permanent unique name,” and “[a]n IP address may be associated with only one instance of [a p-node].” (Ex. 1004 at 383, 394; *id.* at 416 (“The IP addresses are obtained from the TCP service interface.”); *id.* at 383 (p-node is a “[p]oint-to-point” node).) A person of ordinary skill in the art would understand from this disclosure that a processing unit (e.g., a computer running

software defined by NetBIOS) or process (e.g., software implementing the interface defined by NetBIOS running on a computer) must obtain an IP address to participate on the network. (Ex. 1002 ¶ 60.)

2. Step 2: Processing Units Register Their IP Addresses and Identifiers with the NBNS

NetBIOS teaches that a p-node sends its name (“RR_NAME”) and assigned IP address (“NB_ADDRESS”) to the NBNS, and if the name is unique and accepted, the NBNS confirms registration with a “Positive Response” message:



(Ex. 1004 at 402-03 (“[T]he end-node sends a NAME REGISTRATION REQUEST, [and] the NBNS responds with a POSITIVE NAME REGISTRATION RESPONSE”); *id.* at 430-32 (NAME REGISTRATION REQUEST includes p-node’s name and IP address); Ex. 1002 ¶ 61.)

The NBNS also tracks which nodes remain online. For example, a node is deemed offline after a “[s]ilent release,” which “typically occurs when an end-node fails or is turned off.” (Ex. 1004 at 395; *id.* at 378 (“Implicit name deletion occurs when a station ceases operation. In the case of personal computers, implicit name deletion is a frequent occurrence.”).) A node is treated as offline if it fails to send a

“name refresh message” during a “lifetime” period the NBNS establishes during name registration (which can be short or long, depending on the desired amount of network traffic). (*Id.* at 396 (“Names held by an NBNS are given a lifetime during name registration. The NBNS will consider a name to have been silently released if the end-node fails to send a name refresh message to the NBNS before the lifetime expires. A refresh restarts the lifetime clock.”); *id.* at 400 (“NBNS may impose a ‘time-to-live’ on each name it registers” and a node “must periodically send a status report to the NBNS . . . [to] restart the timers of both the NBNS and . . . node The NBNS may consider that a node has released any name which has not been refreshed within some multiple of name’s time-to-live.”); *id.* at 401 (“NBNS is free to poll nodes . . . to verify that their name status is the same as that registered in the NBNS.”); *id.* at 412-13 (p-nodes are “responsible for sending periodic NAME REFRESH REQUESTs . . . [which] contain[] a single name that has been successfully registered by that node. The interval between such packets is negotiated between the end node and the NBNS server at the time that the name is initially claimed.”). “[E]xplicit release” is when “P nodes send a notification to their NBNS.” (*Id.* at 395; *id.* at 378 (“An explicit name deletion function is specified, so that applications may remove a name.”); Ex. 1002 ¶ 62.)

3. Steps 3 & 4: First Processing Unit Sends Query to the NBNS and Receives the IP Address of the Second Processing Unit

To establish point-to-point communications, a first p-node can learn the IP address of a second p-node by sending a “name query request” to the NBNS that includes the name (“QUESTION_NAME”) of the second p-node. (Ex. 1004 at 447, 458 (query includes name of second p-node); *id.* at 395 (“Name query (also known as ‘resolution’ or ‘discovery’) is the procedure by which the IP address(es) associated with a NetBIOS name are discovered.”); *id.* at 406 (“Name query transactions are initiated by end-nodes to obtain the IP address(es) and other attributes associated with a NetBIOS name.”).) If the second p-node is registered, the NBNS sends a POSITIVE RESPONSE to the first p-node that includes the IP address of the second p-node. (*Id.* at 459; *id.* at 407 (“NBNS answers queries from a P node with a list of IP address and other information [of the queried node]”); *id.* at 422 (“datagram service must perform a name query operation to learn the IP address and the attributes of the destination NetBIOS name”); Ex. 1002 ¶ 63.)

4. Step 5: First Processing Unit Uses Received IP Address to Establish Point-to-Point Communications with Second Processing Unit

After obtaining the second p-node’s IP address from the NBNS, the first p-node can establish “directed (point-to-point) communications” with the second p-node. (Ex. 1004 at 415) (IP addresses “found for the target name . . . may have been acquired using the NetBIOS name query transactions NetBIOS session service transactions, packets, and protocols . . . involve only directed (point-to-point) communications.”); *id.* at 383 (defining p-node as “[p]oint-to-point” node); *id.* at 416

(“An end-node begins establishment of a session to another node by somehow acquiring (perhaps using the name query transactions . . .) the IP address of the node . . . ” and each data packet includes the first and second p-node “IP address” and “NetBIOS name”); *id.* at 419 (point-to-point communications described if “a name query operation was successfully completed by the caller node for the listener’s name.”); Ex. 1002 ¶ 64.)

VII. CLAIM CONSTRUCTION

In an IPR, a claim has its “broadest reasonable construction in light of the specification.” 37 C.F.R. § 42.100(b). Petitioners adopt that standard for this proceeding. But, because the ’121 patent expires on September 25, 2015, likely during the pendency of this IPR, Petitioners also present constructions consistent with the standard expressed in *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312-14. (Fed. Cir. 2005) (en banc). However, under either standard, the claims at issue in this petition are invalid.

This section proposes constructions of, and support for, terms that lack a definition in the specification, which also are broadly interpreted. *See In re ICON Health & Fitness, Inc.*, 496 F.3d 1374, 1379 (Fed. Cir. 2007). Any claim terms not included in this section have their broadest reasonable meaning in light of the specification as commonly understood by those of ordinary skill.⁶ Petitioners propose

⁶ If Straight Path, in an effort to avoid prior art, contends that a claim has a

only a broadest reasonable construction, or no construction, where the broadest reasonable construction is the same as the *Phillips* construction.

A. “Point-to-Point Communication” (Claims 3, 4, 10, 13); “Point-to-Point Communication Link” (Claims 6, 8, 9, 10-14); “Point-to-Point Communication Connection” (Claim 7)⁷

In the Sipnet IPR, the Board construed “point-to-point communication link” to mean “direct communications between two processes over a computer network that are not intermediated by a server.” (Ex. 1011 at 7; Ex. 1028 at 9.) Applying the *Phillips* standard, a district court construed “point-to-point communication” as “communication between two processing units or processes, established *by one of the processing units or processes using the IP or network protocol address of the other processing unit or process*, that is not intermediated by a connection server.” (Ex. 1012 at 13 (emphasis added).) WINS and NetBIOS teach the claimed point-to-point communication under both the construction adopted by the Board in the Sipnet IPR and the construction of the district court.

construction other than its broadest reasonable meaning, it should seek to amend the claim to conform to its position. *See* 77 Fed. Reg. 48764 (Aug. 14, 2012).

⁷ The ’121 patent and file history do not distinguish between “point-to-point communication,” “point-to-point communication link,” and “point-to-point communication connection.”

B. “[Program Code for / Program Logic Configured to] Transmitting/Transmit to the Server a Network Protocol Address Received By the First Process Following Connection to the Computer Network” (Claims 12-14)

The broadest reasonable construction of this limitation means “[program code for/program logic configured to/] transmitting/transmit, to a server, the network protocol address of a first process.” Straight Path previously argued that a similar “received” phrase required *dynamic* assignment of the network protocol address (Ex. 1018 [’704 12/2/97 Amend.]), but the Sipnet IPR Board disagreed, finding the limitation “does not require a dynamic element.” (Ex. 1011 at 10).⁸ That conclusion should apply here.

C. “Connection to the Computer Network” (Claims 3, 9-14) / “Connected to the Computer Network” (Claims 4, 6-8, 12-14)

Under the broadest reasonable construction standard, this phrase should be construed as “on-line, e.g., registered with a server.” This is the construction that the Sipnet Board adopted for this term for the parent ’704 patent. (Ex. 1011 at 5-6; Ex. 1028 at 5-7.) The Board based its conclusion on the specification, which teaches that *registration* with the server establishes a processing unit “as an active on-line party.” (Ex. 1001, 6:60-7:7.) The specification also explains that on-line status information

⁸ The ’121 patent operates the same regardless of “whether the current IP addresses were permanent (i.e. predetermined or preassigned) or temporary (i.e. assigned upon initiation of the point-to-point communication).” (Ex. 1001, 9:25-28.)

need only be “relatively current,” which could include a status correct 24 hours earlier that is no longer correct. (*Id.*, 7:8-13.) The same construction should apply here for the ’121 patent.

Under *Phillips*, the Petitioners named in United States International Trade Commission Investigation No. 337-TA-892 proposed, “registered with the server and not subsequently un-registered.” WINS and NetBIOS teach the claimed limitations under this construction as well.

VIII. SPECIFIC GROUNDS FOR PETITION

A. Ground I: Claims 3, 4, and 6-14 Would Have Been Obvious Over WINS and NetBIOS.

Pursuant to Rule 42.104(b)(4)-(5), the following description, as confirmed in the Maggs Declaration (Ex. 1002), demonstrates in detail that claims 3, 4, and 6-14 would have been obvious under 35 U.S.C. § 103(a) over the combined teachings of WINS and NetBIOS. The discussion below contains non-limiting examples of how the prior art teaches each claimed feature and additional examples are set forth elsewhere in the prior art.

1. Level of Ordinary Skill in the Art

A person of ordinary skill in the art for the ’121 patent would generally have had at least a Bachelor’s of Science degree in Computer Science, or equivalent, and several years of relevant programming experience in networking and/or Voice over Internet Protocol. (Absent a formal degree, a person of ordinary skill in the art would

have additional years of work experience.) Such a person would understand the prior art references described herein. (Ex. 1002 ¶ 71.)

2. One Skilled in the Art Would Have Been Motivated to Combine WINS and NetBIOS.

WINS clearly discloses that NetBIOS (as defined in RFCs 1001 and 1002) is incorporated into, and is an integral part of, the disclosed WINS system. (*See, e.g.*, Ex. 1003 at 65 (defining WINS as a “NetBIOS over TCP/IP mode of operation defined in RFC 1001/1002 as p-node”); *id.* at 66 (section entitled “NetBIOS over TCP/IP and Name Resolution” describing “the modes of NetBIOS over TCP/IP, as defined in RFCs 1001 and 1002 to specify how NetBIOS should be implemented over TCP/IP”); *id.* at 69 (“WINS protocol is based on and is compatible with the protocols defined for NBNS in RFCs 1001/1002, so it is interoperable with any other implementations of these RFCs”).) Indeed, WINS refers to NetBIOS over one hundred times (*id.* at 5, 20, 22, 23, 28, 35, 40, 41, 44, 45, 53, 54, 64-67, 69, 77, 79, 80, 87, 102, 103, 122, 124, 126, 139, 142, 144, 147, 151, 164, 165, 171, 208, 209, 214, 224, 225, 262, 265, 270, 271, 273-77), and refers to NetBIOS protocols RFCs 1001 and 1002 ten times (*id.* at 22, 65, 66, 69, 102, 103, 167, 262, 265). (Ex. 1002 ¶ 73.)

Because WINS (Ex. 1003) expressly teaches that its disclosed system should be combined with NetBIOS RFCs 1001 and 1002 (Ex. 1004), one skilled in the art would have known about and been motivated to combine the references. (Ex. 1002 ¶ 73.)

3. **Claim 3 (Independent) Should Be Cancelled.**

- a. **“A computer program product for use with a computer system capable of executing a first process and communicating with other processes, a directory server process and a mail server process over a computer network, the computer program product comprising a computer usable medium having computer program code embodied in the medium, the program code comprising:”**

To the extent the preamble of claim 3 is limiting, WINS discloses a “computer program product . . . comprising a computer usable medium having computer code embodied in the medium” that is “for use with a computer system.” (Ex. 1003 at 121 (“A WINS server is a Windows NT Server computer running Microsoft TCP/IP and the Windows Internet Name Service (WINS) server software.”); *id.* at 23 (referring to the “Windows NT Server 3.5 compact disc”); *id.* at 155 (“[T]he Windows NT Server installation source . . . can be the Windows NT Server compact disc, the installation floppy disks, or a network directory that contains the master files for Windows NT Server.”); Ex. 1002 ¶ 74.)

WINS further explains that such a computer system can execute a first process (e.g., the Windows NT or Windows for Workgroups operating system running on a first computer) and communicate with a second process (e.g., the Windows NT or Windows for Workgroups operating system running on a second computer), a directory server process (e.g., a WINS/NetBIOS server), and a mail server process (e.g., by using an “electronic mail[] client”) over a computer network (e.g., a LAN,

WAN, or the Internet). (Ex. 1003 at 29 (software includes “graphic connectivity utilities, terminal emulation software, Simple Mail Transfer Protocol (SMTP) and electronic mail clients, network printing utilities, SQL client applications, and corporate client-server applications”); *id.* at 76 (Windows NT computers can be used to “dial in to remotely access . . . networks for services such as . . . electronic mail”); *id.* at 118-19 (explaining use for “a small LAN” and “large enterprise network”); *see* Section VI(A) (Overview of WINS).) It further teaches that “NetBIOS on personal computers includes both a set of services and an exact program interface to those services,” and that “NetBIOS defines a software interface . . . designed to allow an implementation to be built on virtually any type of system where the TCP/IP protocol suite is available.” (Ex. 1004 at 374, 377; *see* Section VI(B) (Overview of NetBIOS); Ex. 1002 ¶ 75.)

b. “A. program code configured to determine the currently assigned network protocol address of the first process upon connection to the computer network;”

WINS discloses that each process on a Windows NT network “must be assigned a unique IP address” (Ex. 1003 at 57), and that software running on an Internet access server (e.g., a Dynamic Host Configuration Protocol (DHCP) server) dynamically assigns IP addresses to processes. (*Id.* at 83 (“A [DHCP] server is a Windows NT Server computer running Microsoft TCP/IP and the DHCP-compatible server software” as “defined in [RFCs] 1533, 1534, 1541, and 1542”); *id.* at 113 (“Allocation of IP addresses for distribution by DHCP servers can be done

dynamically . . . [which] allows a client to be assigned an IP address from the free address pool.”); *id.* at 13, 22, 23, 62, 63, 85-120 (DHCP software and configurations); Ex. 1002 ¶ 76.)

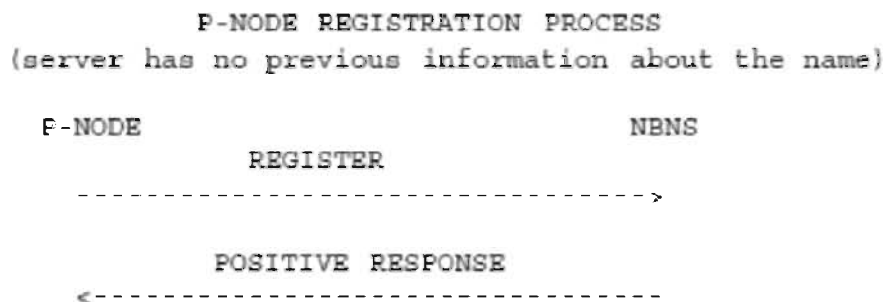
“During system startup,” a computer running the Windows NT or Windows for Workgroups operating system wishing to participate on the network sends a “*discover message*” to multiple DHCP servers, each of which “responds with an *offer message* containing an IP address and valid configuration information for the client.” (Ex. 1003 at 63.) The computer selects the IP address offered by one of the DHCP servers, and “sends a *request message* that identifies the DHCP server for the selected configuration.” (*Id.*) “The selected DHCP server sends a *DHCP acknowledgment message* that contains the [IP] address [for the computer].... After the [computer] receives the acknowledgement, it . . . can now participate on the TCP/IP network and complete its system startup.” (*Id.* at 64; Section (VI)(A)(1).) NetBIOS discloses that p-nodes have “IP addresses . . . obtained from the TCP service interface.” (Ex. 1004 at 416; Section (VI)(B)(1); Ex. 1002 ¶ 77.)

Thus, the prior art teaches program code (e.g., the code underlying the Windows NT or Windows for Workgroups operating system) configured to determine the currently assigned network protocol address of the first process (e.g., the IP address assigned by the DHCP server) upon connection to the network (e.g., at “system startup”).

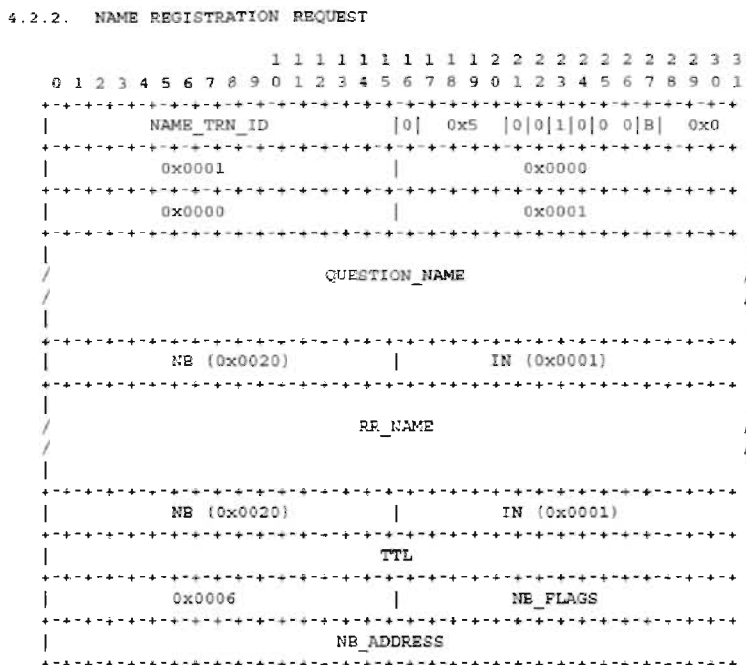
- c. **“B. program code configured to establish a communication connection with the directory server process once the assigned network protocol [sic] of the first process is known;”**

WINS explains that, after the DHCP server dynamically assigns an IP address, the operating system transmits its assigned IP address and name to the WINS server during startup. (Ex. 1003 at 65 (“A mechanism must be available on a TCP/IP network to resolve names to IP addresses. To ensure that both name and address are unique, the Windows NT computer using TCP/IP registers its name and IP address on the network during system startup.”); *id.* at 66 (“NetBIOS over TCP/IP resolves names on a client computer by communicating with the WINS server.”); *id.* at 67 (“In a p-node environment . . . [a]ll computers register themselves with the WINS server, which is a NetBIOS Name Server (NBNS) with enhancements.”); *id.* at 69 (“WINS provides a distributed database for registering and querying dynamic computer name-to-IP address mappings in a routed network environment.”); *id.* at 72 (“During TCP/IP configuration, the computer’s name is registered with the WINS server, and the IP address of the WINS server . . .”); *id.* at 74 (“Name registration ensures that the computer’s name and IP address are unique for each device.”); *id.* (“The name registration request is sent directly to the WINS server to be added to the database.”); *id.* at 151 (database includes “NetBIOS computer name” and “assigned Internet Protocol address”); Section VI(A)(2); Ex. 1002 ¶ 79.)

NetBIOS specifies that the “Name Registration Request” includes “the IP address of the name’s owner [e.g., the p-node registering with the NBNS server].” (Ex. 1004 at 449-50.) “The diagram, below, shows the sequence of events when an end-node [i.e., a p-node] registers a name which is new to the NBNS”:



(*Id.* at 402-03.) NetBIOS also specifies the format for a node name registration (illustrated below), which requires a node to forward to the NBNS its name (“RR_NAME”) and assigned IP address (“NB_ADDRESS”):



(*Id.* at 448-50; *see* Section VI(B)(2); Ex. 1002 ¶ 80.)

Therefore, the prior art discloses program code configured to establish a communication connection with the directory server process (e.g., software running on a first computer sends a name registration request to the WINS/NetBIOS server) once the assigned network protocol address of the first process is known (e.g., after the Windows NT or Windows for Workgroups operating system running on the first computer receives its IP address from the DHCP server during system startup).

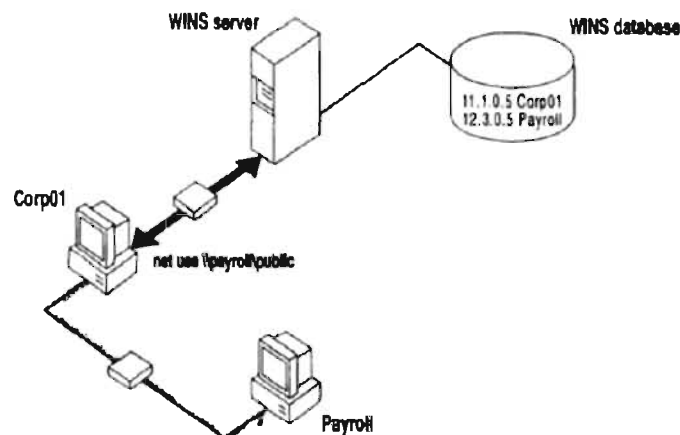
- d. “C. program code configured to forward the assigned network protocol address of the first process to the directory server process upon establishing a communication connection with the directory server process; and”**

As set forth in Section VIII(A)(3)(c) (claim 3[B]), the prior art teaches program (e.g., the code underlying the Windows NT or Windows for Workgroups operating system) code configured to forward the assigned network protocol address of the first process (e.g., the Windows NT or Windows for Workgroups operating system running on a first computer forwards a name registration request containing its IP address) to the directory server process (e.g., WINS/NetBIOS server) upon establishing a communication connection with the directory server process (e.g., during startup). (Ex. 1002 ¶ 82.)

- e. “D. program code configured to establish a point-to-point communication with another process over the computer network”**

WINS teaches how Windows NT software establishes a point-to-point communication with another process over the computer network: “[i]n a p-node

environment,” “when NT_ PC1 [e.g., a computer running the Windows NT or Windows for Workgroups operating system] wants to communicate with NT_PC2 [e.g., another computer running the Windows NT or Windows for Workgroups operating system], it queries the WINS server for the address of NT_PC2 . . . [and] gets the appropriate address from the WINS server....” Ex. 1003 at 67 (emphasis added); *id.* at 266 (defining “p-node” as “[a] NetBIOS over TCP/IP mode that uses point-to-point communications with a name server to resolve computer names as addresses”).) For example, WINS shows how “Corp01” obtains the IP address of “Payroll” using the WINS/NetBIOS server/database, and communicates with Payroll directly without using the WINS/NetBIOS Server:



(Ex. 1003 at 73 (“If the name is found in the WINS database, the client can establish a session based on the address mapping received from WINS.”); *see* Sections (VI)(A)(3) and (A)(4); Ex. 1002 ¶ 83.)

NetBIOS discloses that p-nodes (e.g., implemented by WINS) are “[p]oint- to- point (‘P’) nodes” (Ex. 1004 at 383), and that “[t]he NetBIOS session service,” which

“begins after one or more IP addresses have been found for the target name,” involves “only directed (point-to-point) communications,” beginning with “a TCP connection [being] established with the remote party [e.g., with the second process].” (Ex. 1004 at 415-16; Sections (VI)(B)(3) and (4); Ex. 1002 ¶ 84.)

Thus, prior art teaches program code configured to establish a point-to-point communication with another process (e.g., the Windows NT or Windows for Workgroups operating system running on a first computer can establish point-to-point communication with the Windows NT or Windows for Workgroups operating system running on a second computer using the IP address received from the WINS/NetBIOS server).

4. Claim 4 (Depends From Claim 3) Should Be Cancelled.

a. “The computer program product of claim 3 wherein the program code configured to establish a point-to-point communication further comprises:”

To the extent the preamble of claim 4 is limiting, WINS and NetBIOS teach it for the reasons they teach the limitations of claim 3, as discussed above in Section VIII(A)(3). (Ex. 1002 ¶ 86.)

b. “D.1 program code configured to transmit, from the first process to the directory server process, a query as to whether a second process is connected to the computer network; and”

WINS explains that, “[a]lthough TCP/IP uses IP addresses to identify and reach computers, users typically prefer to use computer names.” (Ex. 1003 at 41; *id.* at

65 (“Computers use IP addresses to identify each other, but users usually find it easier to work with computer names.”).) The Windows NT software assigns a “NetBIOS” name to each process during “setup,” which the process transmits to the WINS/NetBIOS server as a “name registration request” in accordance with “a NetBIOS over TCP/IP mode of operation defined in RFC 1001/1002 as p-node.” (*Id.* at 65, 74, 121; *id.* at 67 (“All computers register themselves with the WINS server, which is a NetBIOS Name Server (NBNS) with enhancements.”); *id.* at 20 (“Microsoft TCP/IP includes . . . NetBIOS for establishing logical names and sessions” and “[WINS] for dynamically registering and querying computer names on an internetwork”); Ex. 1002 ¶ 87.)

WINS further teaches that (1) a WINS server “contain[s] a dynamic database mapping computer names to IP addresses” (Ex. 1003 at 65); (2) software running on a first process requests the WINS/NetBIOS server to retrieve the IP address of a second process by sending “[a] *name query request* . . . to the WINS server” (*id.* at 73); and (3) the WINS/NetBIOS server responds by providing the first process with the IP address of the second process if the name and IP address of the second process are registered in the database. (*Id.* at 73 (“Any name-to-IP address mapping registered with a WINS server can be provided reliably as a response to a name query.”); *id.* at 69 (“WINS provides a distributed database for registering and querying dynamic computer name-to-IP address mappings in a routed network environment.”); *id.* at 72 (“During TCP/IP configuration, the computer’s name is registered with the WINS

server, and the IP address of the WINS server...”); Section (VI)(A)(2); (Ex. 1002 ¶ 88.)

Thus, as explained in Section VIII(A)(3)(e) (claim 3[D]), “when NT_PC1 wants to communicate with NT_PC2, it queries the WINS server for the address of NT_PC2 . . . [and] gets the appropriate address [of NT_PC2] from the WINS server.” (Ex. 1003 at 67; Section (VI)(A)(3); Ex. 1002 ¶ 89.)

WINS also teaches that the WINS server keeps its mapping of names to IP addresses “relatively current” by tracking which users are still connected to the network. (Ex. 1003 at 69 (“[W]hen dynamic addressing through DHCP results in new IP addresses for computers that move between subnets, the changes are automatically updated in the WINS database.”); *id.* at 75 (“[To] provide[] registration reliability through periodic reregistering of names with the WINS servers,” if a process fails to re-register its name within the set renewal time, “the WINS server will mark the name as released and available for use.”); *see* Sections (VI)(A)(1)-(3); Ex. 1002 ¶ 90.)

Similarly, in NetBIOS “[n]ame query transactions are initiated by end-nodes to obtain the IP address(es) and other attributes associated with a NetBIOS name [of a second node]” (Ex. 1004 at 406), and that a “NAME QUERY REQUEST” includes the name (“QUESTION_NAME”) of the second node. (Ex. 1004 at 458.) “Names held by an NBNS are given a lifetime during name registration,” and “[t]he NBNS will consider a name to have been silently released if the end-node fails to send a name refresh message to the NBNS before the lifetime expires.” (Ex. 1004 at 396

(“To detect whether a node has silently released its claim to a name, it is necessary on occasion to challenge that node’s current ownership,” and if that challenge fails “it is assumed that the node has released the name.”); *id.* at 400-01; *see* Sections (VI)(B)(1)-(3); Ex. 1002 ¶ 91.)

NetBIOS further specifies that if the queried name (contained in a “NAME QUERY REQUEST”) is registered with the server directory, the NBNS sends a “POSITIVE NAME QUERY RESPONSE” to the first node. (Ex. 1004 at 395, 407-08, 449, 459.) The “POSITIVE NAME QUERY RESPONSE” includes an “ADDR_ENTRY record ‘representing] an owner of a name” that includes an “NB ADDRESS” corresponding to the IP address of the name of the second computer. (*Id.*) If the queried name is not in the directory (i.e., the second process is not connected to the network), NetBIOS teaches that the NBNS returns a “NEGATIVE NAME QUERY RESPONSE” to the first process, which contains a “Name Error” signifying that “[t]he name requested does not exist” (i.e., the device is not currently connected to the network). (Ex. 1004 at 395, 407, 413, 460; *see* Sections (VI)(B)(1) and (B)(2); Ex. 1002 ¶ 92.)

The prior art thus teaches program code configured to transmit, from the first process (e.g., the Windows NT or Windows for Workgroups operating system running on a first computer) to the directory server process (e.g., a WINS/NetBIOS server), a query (e.g., a WINS or NetBIOS Name Query Request) as to whether a second process (e.g., the Windows NT or Windows for Workgroups operating system

running on a second computer) is connected to the computer network (e.g., if the name and IP address of the second computer is registered in the WINS/NetBIOS server database, the server returns a POSITIVE NAME QUERY RESPONSE with the current IP address of the second process, and if the second process is not registered, the server returns a NEGATIVE NAME QUERY RESPONSE).

c. “D.2 program code configured to receive a network protocol address of the second process from the directory server process, when the second process is connected to the computer network”

As set forth in Section VIII(A)(4)(b) (claim 4[D.1]), WINS and NetBIOS teach program code configured to receive a network protocol address (e.g., IP address) of the second process (e.g., the Windows NT or Windows for Workgroups operating system running on a second computer) from the directory server process (e.g., WINS/NetBIOS server), when the second process is connected to the network (e.g., if the name and IP address of the second process is registered in the WINS/NetBIOS server database, the server informs the first process that the second process is connected to the network by providing a “POSITIVE NAME QUERY RESPONSE” with the current IP address of the second process). (Ex. 1002 ¶ 94.)

5. Claim 6 (Independent) Should Be Cancelled.

a. “A computer program product for use with a computer system capable of executing a first process and connecting to other processes and a server process over a computer network, the computer program product comprising a computer usable medium having computer

readable code means embodied in the medium comprising:”

To the extent the preamble of claim 6 is limiting, WINS and NetBIOS disclose it for the same reasons set forth in Section VIII(A)(3)(a) (claim 3[preamble]). (*Id.* ¶ 96.)

b. “A. program code configured to, following connection of the first process to the computer network, forward to the server process a dynamically assigned network protocol address at which the first process is connected to the computer network;”

WINS and NetBIOS teach this limitation for the reasons in Sections VIII(A)(3)(b)-(d) (claim 3[A-C]). WINS teaches the added requirement of claim 6 for a “dynamically assigned network protocol address” by explaining the DHCP server dynamically assigns IP addresses to processes when they connect to the network, and processes forward their dynamically assigned addresses to the WINS/NetBIOS server during startup. (*E.g.*, Ex. 1003 at 13 (“When DHCP servers are installed on the network, users can take advantage of dynamic IP address allocation and management”); *id.* at 62 (DHCP permits “centralized management of address allocation” through “dynamic configuration of IP addresses for computers”); *id.* at 63 (process obtains network address from DHCP server “[d]uring system startup”); *id.* at 69 (“WINS provides a distributed database for registering and querying dynamic computer name-to-IP address mappings in a routed network environment.”); *id.* at 113 (“Allocation of IP addresses for distribution by DHCP servers can be done

dynamically” which “allows a client to be assigned an IP address from the free address pool.”); *see* Sections (VI)(A)(1) and (A)(2)). As discussed in Sections (VI)(B)(1) and (B)(2), NetBIOS also teaches this feature. (Ex. 1002 ¶ 97.)

WINS and NetBIOS thus teach program code configured to, following connection of the first process to the computer network (e.g., after a first computer running the Windows NT or Windows for Workgroups operating system registers with the DHCP server), forward to the server process (e.g., the WINS/NetBIOS server) a dynamically assigned network protocol address at which the first process is connected to the network (e.g., the IP address that the DHCP server assigned “during system startup”).

- c. **“B. program code configured to query the address server as to whether the second process is connected to the computer network;”**

WINS and NetBIOS teach this limitation for the reasons in Section VIII(A)(4)(b) (claim 4[D.1]). (Ex. 1002 ¶ 99.)

- d. **“C. program code configured to receive a dynamically assigned network protocol address of the second process from the address server, when the second process is connected to the computer network; and”**

WINS and NetBIOS teach this limitation for the reasons in Sections VIII(A)(4)(c) and (5)(b) (claim 4[D.2]; claim 6[A] for “dynamically assigned” aspects of this limitation). (*Id.* ¶ 100.)

- e. **“D. program code configured to respond to the network protocol address of the second process, establish a point-**

to-point communication link with the second process over the computer network”

WINS and NetBIOS teach this limitation for the reasons in Section VIII(A)(3)(e) (claim 3[D]). (*Id.* ¶ 101.)

6. Claim 7 (Independent) Should Be Cancelled.

a. “A computer data signal embodied in a carrier wave comprising:”

To the extent the preamble of claim 7 is limiting, WINS and NetBIOS teach it because one of ordinary skill would understand that a computer data signal embodied in a carrier wave would be created, among other times, during installation of Windows NT or WINS client software. (Ex. 1003 at 155 (“[T]he Windows NT Server installation source ... can be the Windows NT Server compact disc, the installation floppy disks, or a network directory that contains the master files for Windows NT Server.”); Ex. 1002 ¶ 103.)⁹

b. “A. program code configured to, following connection of a first process to a computer network, forward to a server process a dynamically assigned network protocol address at which the first process is connected to the computer network;”

⁹ At least claims 7, 9, and 12 are invalid because they cover a “computer data signal embodied in a carrier wave,” which is unpatentable subject matter. *In re Petrus A.C.M. Nuijten*, 500 F.3d 1346, 1348, 1357 (Fed. Cir. 2007) (claims directed to a “signal” with “embedded data” cover unpatentable subject matter.)

WINS and NetBIOS teach this limitation for the reasons in Sections VIII(A)(3)(b)-(d) (claim 3[A-C]) and VIII(A)(5)(b) (claim 6[A]). (Ex. 1002 ¶ 104.)

- c. **“B. program code configured to query the server process as to whether a second process is connected to the computer network;”**

WINS and NetBIOS teach this limitation for the reasons in Sections VIII(A)(4)(b) and (A)(5)(c) (claims 4[D.1] and 6[B]). (*Id.* ¶ 105.)

- d. **“C. program code configured to receive a dynamically assigned network protocol address of the second process from the server process, when the second process is connected to the computer network; and”**

WINS and NetBIOS teach this limitation for the reasons in Sections VIII(A)(4)(c) and (A)(5)(d) (claims 4[D.2] and 6[C]). (*Id.* ¶ 106.)

- e. **“D. program code, responsive to the network protocol address of the second process, and configured to establish a point-to-point communication connection with the second process over the computer network”**

WINS and NetBIOS teach this limitation for the reasons in Section VIII(A)(3)(e) and (A)(5)(e) (claims 3[D] and 6[D]). (*Id.* ¶ 107.)

7. Claim 8 (Independent) Should Be Cancelled.

- a. **“An apparatus for use with a computer system, the computer system executing a first process operatively coupled over a computer network to a second process and a directory database server process, the apparatus comprising:”**

To the extent the preamble of claim 8 is limiting, WINS and NetBIOS teach it for the reasons in Section VIII(A)(3)(a) (claim 3[preamble]). In addition, WINS and

NetBIOS teach that they are intended for use on an apparatus for use with a computer system. (Ex. 1003 at 12 (“[This manual] describes how to install, configure, and troubleshoot Microsoft TCP/IP on a computer running the Microsoft Windows NT Workstation or Windows NT Server operating system.”); *id.* at 18 (system “can be used to communicate with Windows NT systems, with devices that use other Microsoft networking products, and with non-Microsoft systems”); *id.* at 73 (computers “Corp01” and “Payroll” and WINS server); Ex. 1004 at 374 (NetBIOS intended “for the IBM Personal Computer (PC) and compatible systems” “file servers,” and “virtually any type of system where the TCP/IP protocol suite is available”); *id.* at 375 (NetBIOS used “on TCP networks,” “personal computers,” and “larger hosts”); Ex. 1002 ¶ 109.)

- b. “A. program logic configured to, following connection of the first process to the computer network forward to the address server a dynamically assigned network, protocol address at which the first process is connected to the computer network;”**

WINS and NetBIOS teach this limitation for the reasons in Sections VIII(A)(3)(b)-(d) and (A)(5)(b) (claims 3[A-C], 6[A]). (*Id.* ¶ 110.)

- c. “B. program logic configured to query the address server as to whether the second process is connected to the computer network;”**

WINS and NetBIOS teach this limitation for the reasons in Sections VIII(A)(4)(b) and (A)(5)(c) (claims 4[D.1], 6[B]). (*Id.* ¶ 111.)

- d. **“C. program logic configured to receive a dynamically assigned network protocol address of the second process from the address server, when the second process is connected to the computer network; and”**

WINS and NetBIOS teach this limitation for the reasons in Sections VIII(A)(4)(c) and (A)(5)(d) (claims 4[D.2], 6[C]). (*Id.* ¶ 112.)

- e. **“D. program logic configured to, in response to the network protocol address of the second process, establish a point-to-point communication link with the second process over the computer network”**

WINS and NetBIOS teach this limitation for the reasons in Sections VIII(A)(3)(e), (A)(4), and (A)(5)(e) (claims 3[D], 4, 6[D]). (*Id.* ¶ 113.)

8. Claim 9 (Independent) Should Be Cancelled.

- a. **“A computer data signal embodied in a carrier wave comprising:”**

To the extent the preamble of claim 9 is limiting, it is disclosed for the reasons in Section VIII(A)(6)(a) (claim 7[preamble]). (*Id.* ¶ 115.)

- b. **“a. program code configured to access a directory database, the database having a dynamically assigned network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the dynamically assigned network protocol address of each respective process forwarded to the database following connection to the computer network; and”**

WINS and NetBIOS teach this limitation for the reasons in Sections VIII(A)(3)(b)-(d), (A)(4)(b)-(c), and (A)(5)(b) (claims 3[A-C], 4[D.1- D.2], 6[A]). They teach program code configured to access a directory database (e.g., WINS client

software accesses the WINS/NetBIOS database), the database having a dynamically assigned network protocol address (e.g., the WINS/NetBIOS server stores IP addresses dynamically assigned by the DHCP) for a selected plurality of processes (e.g., instances of the Windows NT or Windows for Workgroups operating system running on computers) having on-line status with respect to the computer network (e.g., the WINS/NetBIOS server database tracks which processes are online, and informs a first process when a second process has online status with respect to the network by providing the current IP address of the second process and/or a POSITIVE NAME QUERY RESPONSE), the dynamically assigned network protocol address of each respective process forwarded to the database following connection to the computer network (e.g., “during system startup,” after the DHCP server dynamically assigns an IP address, the WINS client software forwards the IP address and name to the WINS/NetBIOS server). (Ex. 1002 ¶ 116.)

- c. “b. program code responsive to one of the dynamically assigned network protocol addresses and configured to establish a point-to-point communication link from the first process to the second process over the computer network”**

WINS and NetBIOS teach this limitation for the reasons in Sections VIII(A)(3)(e), (A)(4), and (A)(5)(e) (claims 3[D], 4, 6[D]). They teach program code responsive to one of the dynamically assigned network protocol addresses (e.g., program code responsive to the IP address that the DHCP server dynamically assigned) and configured to establish a point-to-point communication link over the

network from the first process (e.g., the Windows NT or Windows for Workgroups operating system running on a first computer) to the second process (e.g., the Windows NT or Windows for Workgroups operating system running on the second computer) over the computer network (e.g., using TCP/IP). (*Id.* 1 117.)

9. Claim 10 (Independent) Should Be Cancelled.

- a. “In a first computer process operatively coupled over a computer network to a second process and an address server, a method of establishing a point-to-point communication between the first and second processes comprising the steps of:”**

To the extent the preamble of claim 10 is limiting, WINS and NetBIOS teach it for the reasons in Section VIII(A)(3)(e) (claim 3[D]). (*Id.* ¶ 119.)

- b. “a. accessing a directory database, the database having a dynamically assigned network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the dynamically assigned network protocol address of each respective process forwarded to the database following connection to the computer network; and”**

WINS and NetBIOS teach this limitation for the reasons in Sections VIII(A)(3)(b)-(d), (A)(4), and (A)(5)(b) (claims 3[A-C], 4, 6[A]). (*Id.* ¶ 120.)

- c. “b. in response to one of the dynamically assigned network protocol addresses, establish a point-to-point communication link from the first process to the second process over the computer network”**

WINS and NetBIOS teach this limitation for the reasons in Sections VIII(A)(3)(e), (A)(4)(c), and (A)(5)(e) (claims 3[D], 4[D.2], 6[D]). (*Id.* ¶ 121.)

10. Claim 11 (Independent) Should Be Cancelled.

- a. “An apparatus for use with a computer system, the computer system capable of executing a first process connectable over a computer network to a second process and a directory database server process, the apparatus comprising:”**

If the preamble of claim 11 is limiting, WINS and NetBIOS teach it for the reasons in Section VIII(A)(7)(a) (claim 8[preamble]). (*Id.* ¶123.)

- b. “a. program logic configured to access a directory database, the database having a dynamically assigned network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the dynamically assigned network protocol address of each respective process forwarded to the database following connection to the computer network; and”**

WINS and NetBIOS teach this limitation for the reasons in Sections VIII(A)(3)(b)-(d), (A)(4), (A)(5)(b), and (A)(8)(b) (claims 3[A-C], 4, 6[A], 9[a]). (*Id.* ¶ 124.)

- c. “b. program logic responsive to one of the dynamically assigned network protocol addresses and configured to establish a point-to-point communication link from the first process to the second process over the computer network”**

WINS and NetBIOS teach this limitation for the reasons in Sections VIII(A)(3)(e), (A)(4)(c), and (A)(5)(e) (claims 3[D], 4[D.2], 6[D]). (*Id.* ¶ 125.)

11. **Claim 12 (Independent) Should Be Cancelled.**

- a. **“A computer data signal embodied in a carrier wave comprising:”**

To the extent the preamble of claim 12 is limiting, it is disclosed for the reasons in Section VIII(A)(6)(a) (claim 7[preamble]). (*Id.* ¶ 127.)

- b. **“program code for transmitting to a server process, a network protocol address received by a first process following connection to a computer network;”**

WINS and NetBIOS teach this limitation for the reasons in Sections (A)(3)(b)-(d) (claim 3[A-C]). (*Id.* ¶ 128.)

- c. **“program code for transmitting, to the server process, a query as to whether a second process is connected to the computer network;”**

WINS and NetBIOS teach this limitation for the reasons in Section VIII(A)(4)(b) (claim 4[D.1]). (*Id.* ¶ 129.)

- d. **“program code for receiving a network protocol address of the second process from the server process, when the second process is connected to the computer network; and”**

WINS and NetBIOS teach this limitation for the reasons in Section VIII(A)(4)(c) (claim 4[D.2]). (*Id.* ¶ 130.)

- e. **“program code, responsive to the network protocol address of the second process, for establishing a point-to-point communication link between the first process and the second process over the computer network”**

WINS and NetBIOS teach this limitation for the reasons in Sections VIII(A)(3)(e) and (A)(4)(c) (claims 3[D], 4[D.2]). (*Id.* ¶ 131.)

12. **Claim 13 (Independent) Should Be Cancelled.**

- a. **“In a first computer process operatively coupled over a computer network to a second process and an address server, a method of establishing a point-to-point communication between the first and second processes comprising the steps of:”**

To the extent the preamble of claim 13 is limiting, WINS and NetBIOS teach it for the reasons in Sections VIII(A)(3)(a) and (A)(9)(a) (claim 3[preamble], claim 10[preamble]). (*Id.* ¶ 133.)

- b. **“a. transmitting to the server a network protocol address received by the first process following connection to the computer network;”**

WINS and NetBIOS teach this limitation for the reasons in Sections VIII(A)(3)(b)-(d) and (A)(5)(b) (claims 3[A-C], 6[A]). (*Id.* ¶ 134.)

- c. **“b. transmitting, to the server, a query as to whether the second process is connected to the computer network;”**

WINS and NetBIOS teach this limitation for the reasons in Section VIII(A)(4)(b) (claim 4[D.1]). (*Id.* ¶ 135.)

- d. **“c. receiving a network protocol address of the second process from the server, when the second process is connected to the computer network; and”**

WINS and NetBIOS teach this limitation for the reasons in Section VIII(A)(4)(c) (claim 4[D.2]). (*Id.* ¶ 136.)

- e. **“d. program code, responsive to the network protocol address of the second process, for establishing a point-to-point communication link between the first process and the second process over the computer network”**

WINS and NetBIOS teach this limitation for the reasons in Sections VIII(A)(3)(e), (A)(4)(c), and (A)(5)(e) (claims 3[D], 4[D.2], 6[D]). (*Id.* ¶ 137.)

13. Claim 14 (Independent) Should Be Cancelled.

- a. **“An apparatus for use with a computer system, the computer system capable of executing a first process and operatively connectable to a second process and a server process over a computer network, the apparatus comprising:”**

To the extent the preamble of claim 14 is limiting, WINS and NetBIOS teach it for the reasons in Section VIII(A)(3)(a) (claim 3[preamble]). (*Id.* ¶ 139.)

- b. **“program logic configured to transmit to the server a network protocol address received by the first process following connection to the computer network;”**

WINS and NetBIOS teach this limitation for the reasons in Sections (A)(3)(b)-(d) (claim 3[A-C]). (*Id.* ¶ 140.)

- c. **“program logic configured to transmit, to the server, a query as to whether the second process is connected to the computer network;”**

WINS and NetBIOS teach this limitation for the reasons in Section VIII(A)(4)(b) (claim 4[D.1]). (*Id.* ¶ 141.)

- d. “program logic configured to receive a network protocol address of the second process from the server, when the second process is connected to the computer network; and”

WINS and NetBIOS teach this limitation for the reasons in Section VIII(A)(4)(c) (claim 4[D.2]). (*Id.* ¶ 142.)

- e. “program logic, responsive to the network protocol address of the second process, and configured to establish a point-to-point communication link between the first process and the second process over the computer network”

WINS and NetBIOS teach this limitation for the reasons in Sections VIII(A)(3)(e), (A)(4)(c), and (A)(5)(e) (claims 3[D], 4[D.2], 6[D]). (*Id.* ¶ 143.)

B. Examples Where Prior Art Teaches Claimed Features

<u>Claim Limitation</u>	<u>Prior Art</u>
[3-P] A computer program product for use with a computer system capable of executing a first process and communicating with other processes, a directory server process and a mail server process over a computer network, the computer program product comprising a computer usable medium having computer program code embodied in the medium, the program code comprising:	Ex. 1003 at 10-11, 17-20, 23, 29, 118-19; Ex. 1004 at 374, 377
[3-A] A. program code configured to determine the currently assigned network protocol address of the first process upon connection to the computer network;	Ex. 1003 at 13, 22, 57-58, 62-64, 83-115, 118-19; Ex. 1004 at 383, 394, 402, 416, 448-50
[3-B] B. program code configured to establish a communication connection with the directory server process once the assigned network protocol of the first process is known;	Ex. 1003 at 13, 20-22, 40-41, 64-67, 69, 72, 74-75, 79, 121, 151, 262, 265-66; Ex. 1004 at 402-03, 448-50
[3-C] C. program code configured to forward the assigned network protocol address of the first process to the directory server process upon establishing a communication connection with the directory server process; and	See 3-B

<u>Claim Limitation</u>	<u>Prior Art</u>
[3-D] D. program code configured to establish a point-to-point communication with another process over the computer network.	Ex. 1003 at 67, 73, 121-22, 266; Ex. 1004 at 383, 395, 406-07, 415-16, 422, 447, 458
[4-P] The computer program product of claim 3 wherein the program code configured to establish a point-to-point communication further comprises:	See 3-P
[4-A] D.1 program code configured to transmit, from the first process to the directory server process, a query as to whether a second process is connected to the computer network; and	Ex. 1003 at 13, 20-22, 40-41, 64-67, 69, 72-75, 79, 121-22, 151, 262, 26566; Ex. 1004 at 378, 395-96, 400-03, 406-08, 412-13, 422, 447-50, 458-60
[4-B] D.2 program code configured to receive a network protocol address of the second process from the directory server process, when the second process is connected to the computer network.	See 4-A
[6-P] A computer program product for use with a computer system capable of executing a first process and connecting to other processes and a server process over a computer network, the computer program product comprising a computer usable medium having computer readable code means embodied in the medium comprising:	See 3-P
[8-C] C. program logic configured to receive a dynamically assigned network protocol address of the second process from the address server, when the second process is connected to the computer network, and	See 3-A, 3-B, 4-A
[8-D] D. program logic configured to, in response to the network protocol address of the second process, establish a point-to-point communication link with the second process over the computer network.	See 3-D, 4-A
[9-P] A computer data signal embodied in a carrier wave comprising:	See 7-P
[9-A] a. program code configured to access a directory database, the database having a dynamically assigned network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the dynamically assigned network protocol address of each respective process forwarded to the database following connection to the computer network; and	See 3-A, 3-B, 4-A
[9-B] b. program code responsive to one of the dynamically assigned network protocol addresses and configured to establish a point-to-point communication link from the first process to the second process over the computer network.	See 3-D, 4-A

<u>Claim Limitation</u>	<u>Prior Art</u>
[10-P] In a first computer process operatively coupled over a computer network to a second process and an address server, a method of establishing a point-to-point communication between the first and second processes comprising the steps of:	<i>See 3-D</i>
[10-A] a. accessing a directory database, the database having a dynamically assigned network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the dynamically assigned network protocol address of each respective process forwarded to the database following connection to the computer network; and	<i>See 3-A, 3-B, 4-A</i>
[10-B] b. in response to one of the dynamically assigned network protocol addresses, establish a point-to-point communication link from the first process to the second process over the computer network.	<i>See 3-D, 4-A</i>
[11-P] An apparatus for use with a computer system, the computer system capable of executing a first process connectable over a computer network to a second process and a directory database server process, the apparatus comprising:	<i>See 8-P</i>
[11-A] a. program logic configured to access a directory database, the database having a dynamically assigned network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the dynamically assigned network protocol address of each respective process forwarded to the database following connection to the computer network; and	<i>See 3-A, 3-B, 4-A</i>
[11-B] b. program logic responsive to one of the dynamically assigned network protocol addresses and configured to establish a point-to-point communication link from the first process to the second process over the computer network.	<i>See 3-D, 4-A</i>
[12-P] A computer data signal embodied in a carrier wave comprising:	<i>See 7-P</i>
[12-A] program code for transmitting to a server process, a network protocol address received by a first process following connection to a computer network;	<i>See 3-A, 3-B</i>
[12-B] program code for transmitting, to the server process, a query as to whether a second process is connected to the computer network;	<i>See 4-A</i>
[12-C] program code for receiving a network protocol address of the second process from the server process, when the second process is connected to the computer network; and	<i>See 4-A</i>

<u>Claim Limitation</u>	<u>Prior Art</u>
[12-D] program code, responsive to the network protocol address of the second process, for establishing a point-to-point communication link between the first process and the second process over the computer network.	<i>See</i> 3-D, 4-A
[13-P] In a first computer process operatively coupled over a computer network to a second process and an address server, a method of establishing a point-to-point communication between the first and second processes comprising the steps of:	<i>See</i> 3-D
[13-A] a. transmitting to the server a network protocol address received by the first process following connection to the computer network;	<i>See</i> 3-A, 3-B
[13-B] b. transmitting, to the server, a query as to whether the second process is connected to the computer network;	<i>See</i> 4-A
[13-C] c. receiving a network protocol address of the second process from the server, when the second process is connected to the computer network; and	<i>See</i> 4-A
[13-D] d. program code, responsive to the network protocol address of the second process, for establishing a point-to-point communication link between the first process and the second process over the computer network.	<i>See</i> 3-D, 4-A
[14-P] An apparatus for use with a computer system, the computer system capable of executing a first process and operatively connectable to a second process and a server process over a computer network, the apparatus comprising:	<i>See</i> 8-P
[14-A] program logic configured to transmit to the server a network protocol address received by the first process following connection to the computer network;	<i>See</i> 3-A, 3-B
[14-B] program logic configured to transmit, to the server, a query as to whether the second process is connected to the computer network;	<i>See</i> 4-A
[14-C] program logic configured to receive a network protocol address of the second process from the server, when the second process is connected to the computer network; and	<i>See</i> 4-A
[14-D] program logic, responsive to the network protocol address of the second process, and configured to establish a point-to-point communication link between the first process and the second process over the computer network.	<i>See</i> 3-D, 4-A

IX. CONCLUSION

For the foregoing reasons, Petitioners request institution of an *inter partes* review and cancellation of claims 3, 4, and 6-14 of the '121 patent.

Respectfully submitted,

/Rajeev Gupta/

Rajeev Gupta, Reg. No. 55,873

Darren M. Jiron, Reg. No. 45,777

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

901 New York Ave., NW

Washington, D.C. 20001-4413

Telephone: 202-408-4000

Facsimile: 202-408-4400

Table of Exhibits for U.S. Patent 6,131,121 Petition for *Inter Partes* Review

Exhibit	Description
1001	U.S. Patent No. 6,131,121 (“121 patent”)
1002	Declaration of Dr. Bruce M. Maggs (“Maggs Decl”)
1003	Microsoft Windows NT Server Version 3.5 (“WINS”)
1004	Technical Standard: Protocols for X/Open PC Interworking: SMB, Version 2 (“NetBIOS”)
1005	Intentionally Left Blank
1006	Windows NT Server 3.5 TCP/IP Documentation [TCPIP.HLP]
1007	Windows NT Server Copyright Registration
1008	Windows NT Networking Guide
1009	Windows NT Networking Guide Copyright Registration
1010	Petition for Inter Partes Review of U.S. Patent No. 6,108,704, <i>Sipnet EUS.R.O. v. Straight Path IP Group, Inc.</i> (IPR No. 2013-00246) (April 11, 2013)
1011	Institution Decision in <i>Sipnet EU S.R.O. v. Straight Path IP Group, Inc.</i> (IPR No. 2013-00246) (Oct. 11, 2013)
1012	<i>Straight Path IP Group, Inc. v. Bandwidth.com, Inc. et al.</i> , No. 1:13-cv-0932 (E.D.V.A. Feb. 25, 2014) (Dkt. 107, Claim Construction Order)
1013	IETF RFC 1541, October 1993 (“Dynamic Host Configuration Protocol”) (“DHCP”)
1014	IETF RFC 1034, November 1987 (“Domain Names - Concepts And Facilities”) (“Domain Names RFC 1034”)
1015	IETF RFC 1035, November 1987 (“Domain Names - Implementation And Specification”) (“Domain Names RFC 1035”)
1016	IETF RFC 791, September 1981 (“Internet Protocol”)

Exhibit	Description
1017	Curriculum Vitae of Dr. Bruce M. Maggs
1018	Excerpt from File History for U.S. Patent No. No. 6,108,704 (December 2, 1997 Amendment)
1019	U.S. Patent No. 5,159,592 (“Perkins”)
1020	Excerpt from File History for ’121 Patent (Sept. 7, 1999 Amendment)
1021	Excerpt from Reexamination File History for ’121 Patent (May 17, 2010 Office Action)
1022	Excerpt from Reexamination File History for ’121 Patent (July 19, 2010 Amendment)
1023	Excerpt from File History for ’121 Patent (Aug. 23, 2000 Notice of Allowability)
1024	Excerpt from Reexamination for File History for ’121 Patent (Nov. 25, 2009 Response to Non-Final Rejection)
1025	Excerpt from Reexamination for File History for ’121 Patent (Aug. 16, 2010 Notice of Intent to Issue Reexamination Certificate)
1026	Excerpt from File History for ’121 Patent (March 4, 1999 Non-Final Rejection)
1027	Excerpt from Reexamination File History for ’121 Patent (August 25, 2009 Office Action)
1028	Final Written Decision in <i>Sipnet EU S.R.O. v. Straight Path IP Group, Inc.</i> (IPR No. 2013-00246) (Oct. 9, 2014)

CERTIFICATE OF SERVICE

I certify that on June 15, 2015, I caused a copy of this Petition for *Inter Partes* Review, including all attachments, exhibits, and documents therewith, to be served via FedEx Overnight upon the attorneys of record for the Patent Owner at the following address:

DAVIDSON BERQUIST JACKSON & GOWDEY LLP
8300 Greensboro Dr, Suite 500
McLean VA 22102

William Meunier
Matthew Durell
MINTZ, LEVIN, COHEN, FERRIS, GLOVSKY AND POPEO, P.C.
One Financial Center
Boston, Massachusetts 02111
wameunier@mintz.com
mdurell@mintz.com
StraightPathIPRs@mintz.com

Dated: June 15, 2015

Respectfully submitted,

/Rajeev Gupta/

Rajeev Gupta, Reg. No. 55,873
Darren M. Jiron, Reg. No. 45,777
FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.
901 New York Ave., NW
Washington, D.C. 20001-4413
Telephone: 202-408-4000
Facsimile: 202-408-4400