

DOCKET NO.: 2001417-00163US2

Filed on behalf of

Vonage Holdings Corp., Vonage America, Inc., and Vonage Marketing LLC by:	Netflix, Inc. by:
Grant K. Rowan, Reg. No. 41,278 Victor F. Souto, Reg. No. 33,458 Wilmer Cutler Pickering Hale and Dorr LLP 1875 Pennsylvania Ave., NW Washington, DC 20006 Tel: (202) 663-6000 Email: grant.rowan@wilmerhale.com vic.souto@wilmerhale.com	Stacy S. Chen, Reg. No. 62,609 Matthias Kamber ( <i>pro hac vice</i> pending) Keker & Van Nest LLP 633 Battery St. San Francisco, CA 94111 Tel.: (415) 391-5400 Email: schen@kvn.com mkamber@kvn.com

UNITED STATES PATENT AND TRADEMARK OFFICE

---

**BEFORE THE PATENT TRIAL AND APPEAL BOARD**

---

VONAGE HOLDINGS CORP., VONAGE AMERICA, INC., VONAGE  
MARKETING LLC, and NETFLIX, INC.,  
Petitioner

v.

STRAIGHT PATH IP GROUP, INC.  
Patent Owner

Case IPR No.: *To be assigned*

**PETITION FOR *INTER PARTES* REVIEW OF  
U.S. PATENT NO. 6,701,365  
CHALLENGING CLAIMS 1-3  
UNDER 35 U.S.C. § 312 AND 37 C.F.R. § 42.104**

**TABLE OF CONTENTS**

I.	INTRODUCTION.....	1
II.	MANDATORY NOTICES .....	3
A.	Real Party-in-Interest .....	3
B.	Related Matters .....	3
C.	Counsel and Service Information .....	4
III.	CERTIFICATION OF GROUNDS FOR STANDING.....	5
IV.	OVERVIEW OF CHALLENGE AND RELIEF REQUESTED.....	5
V.	OVERVIEW OF THE '365 PATENT (EX. 1001).....	7
A.	Summary of the Alleged Invention.....	7
1.	Step 1: Processing Units Obtain Dynamically Assigned IP Addresses .....	8
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 '365 patent.....	12
C.	Prior <i>Ex Parte</i> Reexamination of the '365 patent.....	12
D.	The Sipnet <i>Inter Partes</i> Review for the '704 Patent.....	14
VI.	OVERVIEW OF THE PRIMARY PRIOR ART REFERENCES .....	15
A.	The Microsoft Manual (Ex. 1003) .....	15
1.	Step 1: Processing Units Obtain Dynamically Assigned IP Addresses from DHCP Servers .....	16
2.	Step 2: Processing Units Register Their IP Addresses and Identifiers with the WINS Server .....	19
3.	Step 3 & 4: First Processing Unit Sends Query to WINS Server and Receives the IP Address of the Second Processing Unit.....	23

4.	Step 5: First Processing Unit Uses Received IP Address to Establish Point-to-Point Communication with Second Processing Unit .....	24
B.	NetBIOS (Ex. 1004).....	25
1.	Step 1: Processing Units Have Assigned IP Addresses .....	26
2.	Step 2: Processing Units Register Their IP Addresses and Identifiers with the NBNS .....	27
3.	Steps 3 & 4: First Processing Unit Sends Query to the NBNS and Receives the IP Address of the Second Processing Unit .....	29
4.	Step 5: First Processing Unit Uses Received IP Address to Establish Point-to-Point Communications with Second Processing Unit.....	30
VII.	CLAIM CONSTRUCTION .....	30
A.	“Point-to-Point Communication” (Claims 1-3).....	31
B.	“Identifier” (Claims 1-3).....	32
C.	“Connected to the Computer Network” (Claim 3) .....	32
VIII.	SPECIFIC GROUNDS FOR PETITION .....	33
A.	Ground I: Claims 1-3 Would Have Been Obvious Over the Microsoft Manual and NetBIOS.....	33
1.	Level of Ordinary Skill in the Art. ....	33
2.	One Skilled in the Art Would Have Been Motivated to Combine the Microsoft Manual and NetBIOS.....	34
3.	Claim 1 (Independent) Should Be Cancelled. ....	35
4.	Claim 2 (Independent) Should Be Cancelled. ....	47
5.	Claim 3 (Independent) Should Be Cancelled. ....	49
B.	Claim Chart with Examples Showing Where Prior Art Teaches Claimed Features .....	55
IX.	CONCLUSION .....	57

**TABLE OF AUTHORITIES**

**Cases**

<i>In re ICON Health &amp; Fitness, Inc.</i> , 496 F.3d 1374 (Fed. Cir. 2007).....	31
---	----

<i>In re Petrus A.C.M. Nuijten</i> , 500 F.3d 1346 (Fed. Cir. 2007).....	48
---	----

**Statutes**

35 U.S.C. § 314(a) .....	5
37 C.F.R. § 42.100(b) .....	30
37 C.F.R. § 42.104(a).....	5
37 C.F.R. § 42.104(b) .....	5, 33
37 C.F.R. § 42.22(a).....	5
77 Fed. Reg. 48764 (Aug. 14, 2012) .....	31

## I. INTRODUCTION

Vonage Holdings Corp., Vonage America, Inc., Vonage Marketing LLC, and Netflix, Inc. (collectively “Petitioner”) respectfully request *Inter Partes* Review of claims 1-3 of U.S. Patent No. 6,701,365 (“the ’365 patent”) (Ex. 1001)<sup>1</sup> pursuant to 35 U.S.C. §§ 311-19 and 37 C.F.R. § 42.1 *et seq.*

The ’365 patent is directed to establishing “point-to-point communications” between two processing units (e.g., computers) over a computer network. The ’365 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 processing unit did not know the specific network address of the other processing unit; for example, in the case of processing units with dynamically assigned addresses that can change over time.

To address that alleged “problem,” the ’365 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 processing units. In response to a query received from a first processing unit (e.g., using the

---

<sup>1</sup> Petitioner has numbered each page of the Exhibits in the format “Petitioner Vonage Holdings Corp. et al. - Exhibit 10\_\_ - Page \_\_\_\_”. All page number cites are to the numbers added by Petitioner.

name of a second processing unit), the server sends the current network address of the second processing unit to the first processing unit, and the first processing unit then uses that received address to establish point-to-point communication with the second processing unit.

But by September 1995 (the claimed priority date of the '365 patent), others had already *solved* the *same* problem using the *same* basic server lookup feature. For example, in 1994, Microsoft published a user's manual for Version 3.5 of its Windows NT Server software ("the Microsoft Manual") (Ex. 1003). Just like the '365 patent, the Microsoft Manual teaches: (1) a name server (implementing the NetBIOS protocol) that tracks the current dynamically assigned network address and name of each registered processing unit; and (2) a first processing unit that sends a name query for a second processing unit 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 processing unit.

Therefore, as detailed below, the Microsoft Manual and NetBIOS protocol (Ex. 1004) render obvious claims 1-3 of the '365 patent.<sup>2</sup>

---

<sup>2</sup> Because the Microsoft Manual explicitly incorporates the NetBIOS protocol, *see infra* § VIIIA(2), they are a single anticipatory reference. But for purposes of this Petition, Petitioner has treated the references as an obviousness combination.

## II. MANDATORY NOTICES

### A. Real Party-in-Interest

Vonage Holdings Corp., Vonage America, Inc., Vonage Marketing LLC (collectively, “Vonage”) and Netflix, Inc. (“Netflix”) are the real parties-in-interest and are the Petitioner.

### B. Related Matters

Straight Path has asserted the ’469 patent against Vonage in the District of New Jersey (2:14-cv-00502-JLL-JAD) and against Netflix and other parties in the Eastern District of Texas (6:13-cv-00604-KNM; 6:13-cv-00605-KNM; 6:13-cv-00607-KNM; 6:13-cv-00606-KNM, 6:14-cv-00405-KNM; 6:14-cv-00534-KNM). Netflix has moved to intervene in the Eastern District of Virginia (1:13-cv-933-AJT-IDD; 1:13-cv-934-AJT-IDD), and VIZIO has sued Straight Path in the Eastern District of Virginia (No. 2:14-cv-00233-HCM-DEM) concerning the ’704, ’496, and ’121 patents. Petitioner also is filing Petitions for Inter Partes Review of U.S. Patent Nos. 6,108,704 (“the ’704 patent”), 6,009,469 (“the ’469 patent”), 6,513,066 (“the ’066 patent”), and 6,701,365 (“the ’365 patent”); the ’121, ’469, ’066, and ’365 patents all name the same alleged inventors and claim to descend from the ’704 patent.

On October 11, 2013, the Board instituted inter partes review of claims 1-7 and 33-42 of the ’704 patent. See *Sipnet EU S.R.O. v. Straight Path IP Group*, IPR

No. 2013-00246. (Ex. 1011.) For the sake of efficiency and consistency, Petitioner requests that the Board assign the Sipnet panel to address the petition.

**C. Counsel and Service Information**

Lead Counsel for Vonage:	Grant K. Rowan (Registration No. 41,278)  grant.rowan@wilmerhale.com
Lead Counsel for Netflix:	Stacy S. Chen (Registration No. 62,609)  schen@kvn.com
Backup Counsel for Vonage:	Victor F. Souto (Registration No. 33,458)  vic.souto@wilmerhale.com
Backup Counsel for Netflix:	Matthias Kamber ( <i>pro hac vice</i> pending <sup>3</sup> )  mkamber@kvn.com
Post and hand delivery for Vonage:	Wilmer Cutler Pickering Hale and Dorr LLP  1875 Pennsylvania Avenue, NW  Washington, DC 20006
Post and hand delivery for Netflix:	Keker & Van Nest LLP  633 Battery Street  San Francisco, CA 94111

---

<sup>3</sup> Petitioner hereby requests authorization to file a motion for back-up counsel for Netflix to appear *pro hac vice*.



Telephone for Vonage:	202-663-6000
Telephone for Netflix:	415-391-5400
Fax for Vonage:	Fax: 202-663-6363
Fax for Netflix:	Fax: 415-397-7188

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

### **III. CERTIFICATION OF GROUNDS FOR STANDING**

Petitioner certifies under Rule 42.104(a) that the '365 patent is available for *inter partes* review and Petitioner is not barred or estopped from requesting an *inter partes* review challenging the claims on the grounds identified in this Petition.

### **IV. OVERVIEW OF CHALLENGE AND RELIEF REQUESTED**

Petitioner requests cancellation of claims 1-3 of the '365 patent ("the challenged claims") as unpatentable under 35 U.S.C. § 103. This Petition, supported by the accompanying Declaration of Dr. Bruce M. Maggs ("Maggs Decl.") (Ex. 1002), demonstrates a reasonable likelihood that the challenged claims are not patentable and that Petitioner will prevail with respect to at least one challenged claim. *See* 35 U.S.C. § 314(a).

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

1. The Microsoft Manual (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).<sup>4</sup> The September 1994 publication date for the Microsoft Manual is further confirmed, for example, by: (1) Exhibit 1007, which is a copyright registration notice that lists September 19, 1994, as the date of first publication for “Microsoft Windows NT Server, Version 3.5”; (2) Exhibit 1006, which is 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 and bears a 1994 copyright date, and that is substantively identical (except that it does not include the “Glossary” section) to the Microsoft Manual (*see* Decl. of Jason H. Liss Regarding Microsoft Windows NT Server 3.5 (“Liss Decl.”) ¶¶ 4-10); and (3) Exhibit 1008, which is a book titled *Microsoft Windows NT Networking Guide* that contains the relevant portions (aside from the “Welcome” and “Glossary” sections) of the Microsoft Manual at issue here and was first published in February 1995, as confirmed by the copyright registration notice attached as Exhibit 1009 to this Petition (*see* Liss Decl. ¶¶ 11). The “TCPIP.HLP” file is prior art under at least 35 U.S.C. §§ 102(a) and (b) and the

---

<sup>4</sup> The Microsoft Manual (Exhibit 1003) is currently before the Board as Exhibit 1004 in the ongoing Sipnet *inter partes* review for the ’704 patent (discussed further below in § V(D)).

*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 as Appendices F and G (respectively, Internet Engineering Task Force RFC Nos. 1001 (“Protocol Standard for a NetBIOS Service on a TCP/UDP Transport: Concepts and Methods”) and 1002 (“Protocol Standard for a NetBIOS Service on a TCP/UDP Transport: Detailed Specifications”)) (Ex. 1004) (collectively, “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 ’365 PATENT (EX. 1001)**

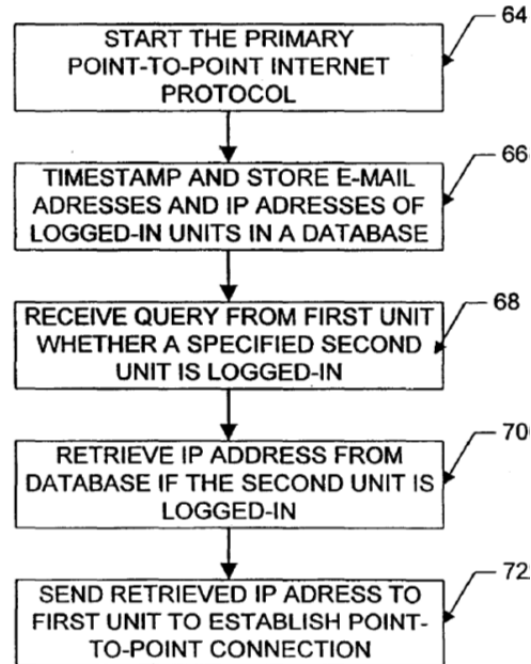
### **A. Summary of the Alleged Invention**

The ’365 patent concedes that, in the prior art, a first processing unit could establish “point-to-point communications” with a second processing unit using the network address assigned to the second processing unit, “in a manner known in the art.” (Ex. 1001, 1:29-31 (“[D]evices interfacing to the Internet and other online services may communicate with each other upon establishing respective device addresses.”); *id.*, 1:55-57, 8:6-8, 8:30-33 (“Permanent IP addresses of users and devices accessing the Internet readily support point-to-point communications of voice and video signals over the Internet,” which “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”).)

According to the '365 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). (*Id.*, 1:48-51, 1:60-63; Ex. 1002 ¶ 34.)

The '365 patent claimed to solve that supposed “problem” through the basic lookup feature described in connection with Figure 8 below:



**FIG. 8**

(Ex. 1001, Fig. 8; Ex. 1002 ¶ 35.)

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

When a “processing unit” (the term used in the specification of the ’365 patent) or “process” (the term used in the claims)<sup>5</sup> “logs on to the Internet ..., the respective [device] is provided a dynamically allocated IP address by a Connection Service Provider.” (Ex. 1001, 5:27-34; 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, 5:35-49; *id.*, 10:15-31 (“[C]onnection server 26 ... timestamp[s] and stor[es] E-mail and IP addresses of logged-in users and processing units in the database 34 in step 66.”); Ex. 1020 [4/19/02 Amendment] at 4 (“One embodiment utilizes a dedicated server which acts as a network address/information directory from which calling processes can obtain information. When a first process connects to the network, the process logs-on to the server and provides the server with the network protocol address under which the first process is currently operating.”); Ex. 1021 [7/17/02 Request] at 2; Ex. 1002 ¶ 37.)

---

<sup>5</sup> For the sake of convenience, and because it is the term used in the specification of the ’365 patent, the term “processing unit” will be used in Sections V and VI of this Petition.

Connection server 26 keeps “relatively current” data concerning the “on-line status” of registered processing units; e.g., it may confirm that a registered processing unit remains online after “predetermined time periods, such as a default value of 24 hours.” (Ex. 1001, 5:50-55.) Alternatively, “[w]hen a user logs off or goes off-line 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.*, 6:17-20; Ex. 1002 ¶ 38.)

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

When the user of a first processing unit wishes to establish point-to-point communications with a second processing unit, the user of the first processing unit “sends a query, including the E-mail address [(or other identifier)] of the callee, to the connection server 26.” (Ex. 1001, 5:66-67.) The query can include, for instance, “the name or alias ... of a party to be called.” (*Id.*, 9:37-44; Ex. 1020 [4/19/02 Amendment] at 4 (“A second process, wishing to establish communications with the first process, connects to the server and request [sic] the network protocol address under which the first process is currently operating.”); Ex. 1021 [7/17/02 Request] at 2; Ex. 1002 ¶ 39.)

Upon receiving the query, connection server 26 “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, 6:1-4.) “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.*, 6:4-8; *id.*, 10:42-45 (connection server “retriev[es] the IP address of the specified user from the database 34 in step 70 if the specified user is logged-in to the Internet”); Ex. 1020 [4/19/02 Amendment] at 4 (“Upon receipt of the network protocol address of the first process, the second process establishes communications with the first process directly, without any intervention from the address/information server.”); Ex. 1021 [7/17/02 Request] at 2; Ex. 1002 ¶ 40.)

**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] may then directly establish point-to-point Internet communications with the [second processing unit] using the IP address of the [second processing unit].” (Ex. 1001, 6:8-11; *id.*, 10:45-47 (IP address received from connection server 26 enables “the first [processing unit] to establish point-to-point Internet communications with the [second processing unit].”); Ex. 1020 [4/19/02 Amendment] at 4 (“Upon receipt of the network protocol address of the first process, the second process establishes communications with the first

process directly, without any intervention from the address/information server.”); Ex. 1021 [7/17/02 Request] at 2; Ex. 1002 ¶ 41.)

The ’365 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, 8:7-8, 8:31-33 (emphases added).) The ’365 patent also functions the same “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., 7:49-52; Ex. 1002 ¶ 42.)

#### **B. Original Prosecution of the ’365 patent**

During the original prosecution of the ’365 patent, the Examiner initially rejected all claims, but then issued the ’365 patent after the patentee amended the claims. (Ex. 1022 [2/13/02 Office Action]; Ex. 1023 [6/20/02 Office Action]; Ex. 1024 [10/8/02 Amendment]; Ex. 1025 [12/31/02 Notice of Allowance].) This Petition does not rely on any prior art cited during the original prosecution. (Ex. 1002 ¶ 43.)

#### **C. Prior *Ex Parte* Reexamination of the ’365 patent**



On February 23, 2009, a third party requested *ex parte* reexamination of the '365 patent. During that proceeding, the Examiner rejected all claims under 35 U.S.C. § 103 in light of NetBIOS and RFC 1531 (*i.e.*, the Dynamic Host Configuration Protocol (DHCP)). (Ex. 1026 [8/14/09 Office Action] at 3-5.) Although the Examiner agreed that NetBIOS describes the same address lookup functionality claimed in the '365 patent, he 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. 1027 [5/11/10 Notice] at 4; Ex. 1002 ¶ 44.)

But the expert declaration failed to note that other prior art, including the Microsoft Manual, disclosed using dynamic addressing in a NetBIOS-type system. (*E.g.*, Ex. 1003 at 67 (“All computers register themselves with the WINS server, which is a NetBIOS Name Server (NBNS) with enhancements.”); *id.* 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,” which “is based on and is compatible with the protocols defined for [NetBIOS Name Server] in RFCs 1001/1002, so it is interoperable with any other implementation of these RFCs.”; *id.* 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 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 65 (“WINS is a NetBIOS over TCP/IP mode of operation defined in RFC 1001/1002 as p-node.”); *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”; “NetBIOS over TCP/IP is the session-layer network service that performs name-to-IP address mapping for name resolution. This section describes the modes of NetBIOS over TCP/IP, as defined in RFCs 1001 and 1002 to specify how NetBIOS should be implemented over TCP/IP.”); Ex. 1002 ¶ 45.)

Relying on the flawed expert declaration, the Examiner confirmed the claims of the ’365 patent without amendment. (Ex. 1027 [5/11/10 Notice] at 6; Ex. 1002 ¶ 46.)

**D. The Sipnet *Inter Partes* Review for the ’704 Patent**

In April 2013, a third party (“Sipnet”) initiated an IPR seeking cancellation of certain claims of the ’704 patent (the parent of the ’365 patent), based on, *inter alia*, the Microsoft Manual and NetBIOS. (Ex. 1010.) On October 11, 2013, the Board instituted an IPR for all challenged claims and found a reasonable likelihood that the Microsoft Manual and NetBIOS each anticipated most of the challenged

claims and that the remaining claims would have been obvious over the Microsoft Manual and NetBIOS. (Ex. 1011 at 20-21.) The Board conducted a hearing on July 11, 2014.

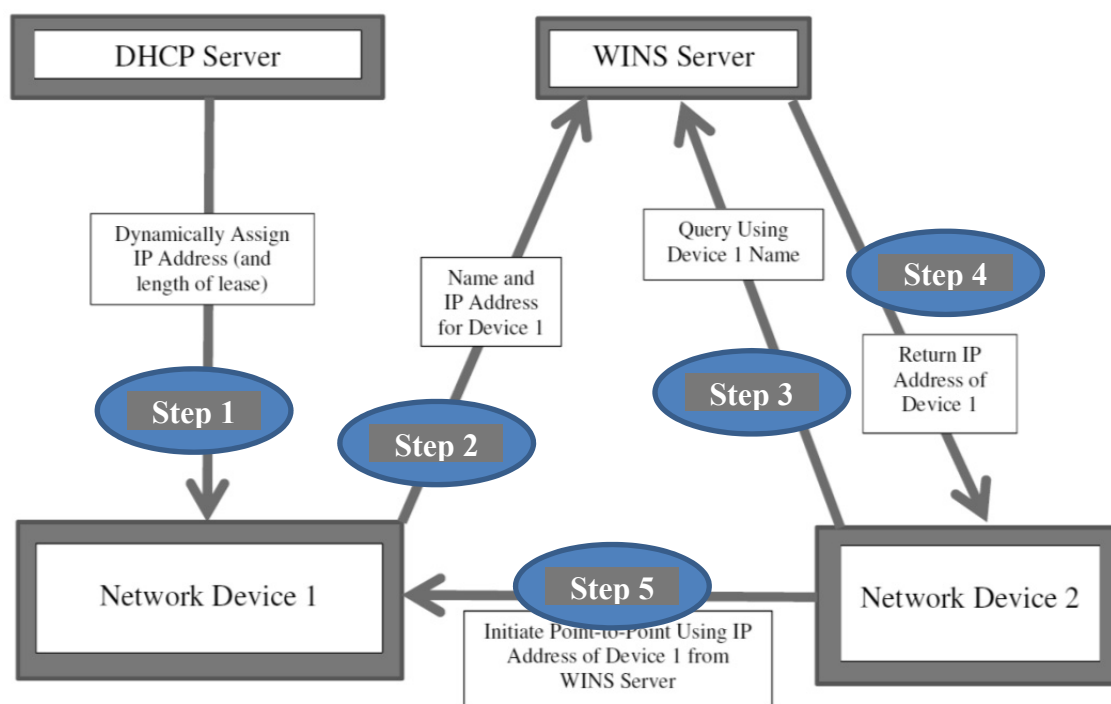
## **VI. OVERVIEW OF THE PRIMARY PRIOR ART REFERENCES**

### **A. The Microsoft Manual (Ex. 1003)**

In 1994, Microsoft published and publicly distributed its Microsoft Windows NT Server TCP/IP manual (“Microsoft Manual”), 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.) As discussed below, the Microsoft Manual teaches how a processing unit (for example, a Windows 3.11 workstation running WINS client software) or process (for example, the Windows 3.11 operating system) can initiate point-to-point communications with other processing units using the same five-step process described in the ’365 patent.<sup>6</sup> (Ex. 1002 ¶ 47.)

---

<sup>6</sup> The Microsoft Manual explains that the Windows NT operating system (v3.5) uses the TCP/IP protocol “to communicate with Windows NT systems, with devices that use other Microsoft networking products, and with non-Microsoft systems, such as UNIX.” (Ex. 1003 at 18, 19 (TCP/IP allows Windows NT to “connect to the Internet” and to “private internetworks”); *id.* at 27-28 (examples of



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

The Microsoft Manual explains that each processing unit participating on a Windows NT network “must be assigned a unique IP address” (Ex. 1003 at 57-58), and that Windows NT networks can use a DHCP server to assign IP addresses “automatically” to processing units at start up—including dynamically (*id.* at 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 compatible non-Microsoft networks); *id.* at 55-56 (outlining history and general operation of TCP/IP).)

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 (explaining 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 software and configuration options, including for “a 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. This configuration information includes 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 processing unit] 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 Microsoft Manual makes clear that the lease period is set by the system administrator, and 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**

The Microsoft Manual 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, the Microsoft Manual teaches that Windows NT is used to assign each device a “NetBIOS” name during “setup.” (*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.)

The Microsoft Manual further teaches that a Windows Internet Name Service (WINS) server consists of “a Windows NT Server computer running Microsoft TCP/IP and the [WINS] server software” and “contain[s] 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 (“In a Windows network, a WINS server can provide name registration services” and is “[t]he server implemented under RFC 1001/1002 to provide name resolution services for NetBIOS computer names.”); Ex. 1002 ¶ 52.)<sup>7</sup>

---

<sup>7</sup> The Microsoft Manual teaches that Windows NT also can be used with a Domain Name System (DNS) server, which, unlike a WINS server, “requires static configuration for computer name-to-IP address mapping.” (Ex. 1003 at 78, 80; Ex. 1002 ¶ 52.)



The Microsoft Manual teaches that this mapping of names to IP addresses performed by the WINS server “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 (defining “WINS” as “[a] name resolution service that resolves Windows networking computer names to IP addresses in a routed environment” and 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”); Ex. 1002 ¶ 53.)

The Microsoft Manual also explains that, in a Windows NT system, “WINS name resolution is enabled and configured automatically for a computer that is configured with DHCP.” (Ex. 1003 at 40.) In such a system, after the DHCP server dynamically assigns an IP address to a processing unit, the 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 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.)

The Microsoft Manual also teaches that the WINS 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.) Similarly, 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.) The Microsoft Manual also explains that, “[to] provide[] registration reliability through periodic reregistering

of names with the WINS servers,” if a network device fails to re-register its name within the set “renewal time,” “the WINS server will mark the name as released and available for use.” (*Id.*; Ex. 1002 ¶ 55.)

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

The Microsoft Manual 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, including proxies, access the WINS server directly.”); Ex. 1002 ¶ 56.)

The Microsoft Manual further teaches that a first processing unit—e.g., the Windows NT or Windows for Workgroups 3.11 operating system running on a first computer—can obtain the IP address of a second processing unit—e.g., a the Windows NT or Windows for Workgroups 3.11 operating system running on a second computer—by transmitting a query to the WINS server containing the name of the second processing unit—which causes the WINS server to return the IP address of the second processing unit. To do so, the “client software” of the first processing unit sends “[a] *name query request*” to the WINS server, which returns the IP “address mapping” of the second processing unit to the first processing unit. (Ex. 1003 at 72-73 (“Any name-to-IP address mapping registered

with a WINS server can be provided reliably as a response to a name query.”); *id.* at 67 (“[W]hen NT\_PC1 wants to communicate with NT\_PC2, it queries the WINS server for the address of NT\_PC2 ... gets the appropriate address from the WINS server, [and] goes directly to NT\_PC2 ....”); *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 ¶ 57.)

**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 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 (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”); *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 ¶ 58.)

## **B. NetBIOS (Ex. 1004)**

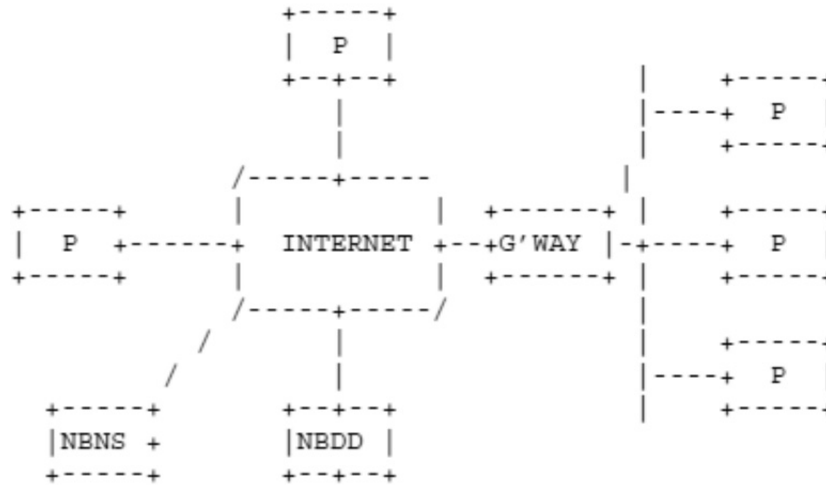
In March 1987, Network Working Group of the IETF published RFC 1001, which “describes the ideas and general methods used to provide NetBIOS on a TCP and UDP foundation,” and RFC 1002, which “contains the detailed packet formats and protocol specifications for NetBIOS-over-TCP.” (Ex. 1004 at 368, 374, 442.)<sup>8</sup> In 1992, RFCs 1001 and 1002 were published as Appendices F and G in *Technical Standard – Protocols for X/Open PC Interworking: SMB, Version 2* (Ex. 1004). (Ex. 1002 ¶ 59.)

NetBIOS teaches point-to-point communications between two or more nodes over a network (including LANs and the Internet), including between “[p]oint-to-point (or ‘P’) nodes” that use a dedicated directory look-up service called a “NetBIOS Name Server” or “NBNS.” (Ex. 1004 at 384-85.) For example, the figure below shows “P”-nodes connected to the “Internet” (two

---

<sup>8</sup> Network Basic Input/Output System (NetBIOS) is a protocol first developed in the early 1980s that allows applications on different computers to communicate across a computer network, such as a local area network or the Internet. (Ex. 1004 at 374, 377.)

directly, and three through a gateway (“G’WAY”)), each of which can communicate with a NetBIOS Name Server (NBNS) that also is connected to the Internet:



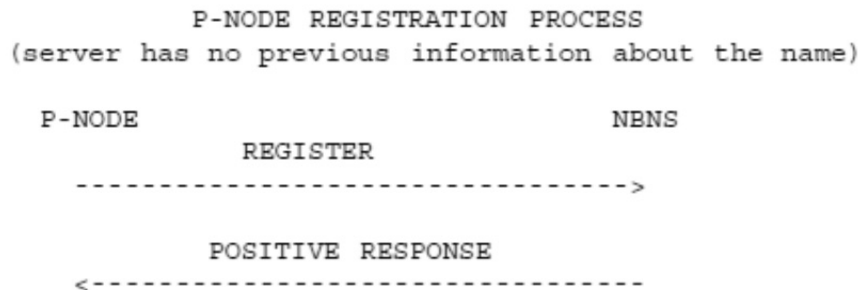
(Ex. 1004 at 389; Ex. 1002 ¶ 60.)

### 1. Step 1: Processing Units Have Assigned IP Addresses

NetBIOS explains that “[e]very node has a permanent unique name” (Ex. 1004 at 394), and that “[a]n IP address may be associated with only one instance of [a p-node].” (Ex. 1004 at 383; *id.* at 416 (“The IP addresses are obtained from the TCP service interface.”); *id.* at 383 (defining p-node as “[p]oint-to-point” node).) In my opinion, a person of ordinary skill in the art would understand from this disclosure that a processing unit (for example, a computer running software implementing the interface defined by NetBIOS) or process (software implementing the interface defined by NetBIOS running on a computer) must obtain an IP address to participate on the network. (Ex. 1002 ¶ 61.)

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

A p-node with an IP address sends its name (“RR\_NAME”) and assigned IP address (“NB\_ADDRESS”) to the NBNS, and if the name is unique and accepted, the NBNS returns a “Positive Response” message confirming registration (as shown below):



(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 (noting “NAME REGISTRATION REQUEST” includes p-node’s name (“RR\_NAME”) and IP address (“NB\_ADDRESS”); Ex. 1002 ¶ 62.)

The NBNS 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.”).) In this scenario, the node is treated as offline if it fails to send a “name refresh message” during a “lifetime” period that the NBNS established 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 (“The 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 (“Each P or M node is 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.”); Ex. 1002 ¶ 63.)

By contrast, “explicit release” occurs when “P nodes send a notification to their NBNS.” (Ex. 1004 at 395; *id.* at 378 (“An explicit name deletion function is specified, so that applications may remove a name.”); Ex. 1002 ¶ 64.)



**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” to the NBNS that includes the name of the second p-node. (Ex. 1004 at 447, 458 (“NAME QUERY REQUEST” includes name of second p-node (“QUESTION\_NAME”)); *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 has previously registered its name with the NBNS, the NBNS provides a “POSITIVE RESPONSE” to the first p-node that includes the IP address of the second p-node. (*Id.* at 459; *id.* at 407 (“An NBNS answers queries from a P node with a list of IP address and other information [for the queried name].”); *id.* at 422 (“To transmit a NetBIOS datagram, the datagram service must perform a name query operation to learn the IP address and the attributes of the destination NetBIOS name.”); Ex. 1002 ¶ 65.)

**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) ("The NetBIOS session service begins after one or more IP addresses have been found for the target name. These addresses may have been acquired using the NetBIOS name query transactions .... NetBIOS session service transactions, packets, and protocols are identical for all end-node types. They 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 ...."); *id.* (each data packet includes the first and second p-node "IP address" and "NetBIOS name"); *id.* at 419 (explaining point-to-point communications if "a name query operation was successfully completed by the caller node for the listener's name"); Ex. 1002 ¶ 66.)

**VII. CLAIM CONSTRUCTION**

In an IPR, a claim is given the "broadest reasonable construction in light of the specification." 37 C.F.R. § 42.100(b). Petitioner adopts that standard for this proceeding, but reserves the right to pursue different constructions in a district court, where different claim construction standards apply.

This section proposes constructions of, and support for, terms that lack a definition in the specification, which also are given a broad interpretation. *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 in the art.<sup>9</sup>

**A. “Point-to-Point Communication” (Claims 1-3)**

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). Applying a different standard, a district court has construed “point-to-point communication” differently, such that communication must be “established by one of the processes using the network protocol address of the other process.” (Ex. 1012 at 11, 13 (“[C]ommunication 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.” (emphasis added))).) The Microsoft Manual and NetBIOS teach the

---

<sup>9</sup> If Straight Path, in an effort to avoid the prior art, contends that a claim has a construction different from its broadest reasonable interpretation, Straight Path should seek to amend the claim to correspond to its contentions in this proceeding. *See* 77 Fed. Reg. 48764 (Aug. 14, 2012).

claimed point-to-point communication under both the construction adopted by the Board in the Sipnet IPR and the construction of the district court.

**B. “Identifier” (Claims 1-3)**

Under the broadest reasonable construction standard, “identifier” should be construed to mean “*information that distinguishes one process from other processes registered with the server.*” In the preferred embodiment, an e-mail address is an identifier that is uniquely associated with a first processing unit and is used to distinguish the first processing unit from other processing units that are registered with the connection server. (Ex. 1001, 6:66-7:3, 7:31-42.) The broadest reasonable construction of “identifier” is not limited to an e-mail address and includes other distinguishing information such as a “name or alias.” (*Id.* at 9:37-44.)

**C. “Connected to the Computer Network” (Claim 3)**

Under the broadest reasonable construction standard, this claim 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 in the parent ’704 patent. (Ex. 1011 at 5-6.)

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, 5:35-49.) The Board’s conclusion is also consistent with the

patent's explanation that on-line status information need only be "relatively current," which could include a status that was correct 24 hours earlier, but might no longer be correct. (*Id.*, 5:50-55.) The broadest reasonable construction of "connected" cannot exclude registration in the database of a server, which is an embodiment in the '365 patent as discussed above.

## **VIII. SPECIFIC GROUNDS FOR PETITION**

### **A. Ground I: Claims 1-3 Would Have Been Obvious Over the Microsoft Manual 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 1-3 would have been obvious under 35 U.S.C. § 103(a) over the combined teachings of the Microsoft Manual and NetBIOS. The discussion below contains non-limiting examples of how the prior art teaches each claimed feature and additional examples are set forth in the claim chart at Section VIII(B) and elsewhere in the prior art.

#### **1. Level of Ordinary Skill in the Art.**

A person of ordinary skill in the art related to the '365 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 ¶ 73.)

**2. One Skilled in the Art Would Have Been Motivated to  
Combine the Microsoft Manual and NetBIOS**

The Microsoft Manual clearly discloses that NetBIOS (as defined in RFCs 1001 and 1002) is incorporated into, and an integral part of, the disclosed WINS system. (*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, the Microsoft Manual 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, 274-77) and refers to NetBIOS protocols RFCs 1001 and 1002 ten times (*id.* at 22, 65, 66, 69, 102, 103, 167, 262, 265). (Ex. 1002 ¶ 75.)

Because the Microsoft Manual (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 ¶ 76.)

**3. Claim 1 (Independent) Should Be Cancelled.**

- a. “A computer program product for use with a server operatively coupled over a computer network to a plurality of processes, the computer program product comprising a computer usable medium having program code embodied thereon the program code comprising:”**

To the extent that the preamble of claim 1 is limiting, the Microsoft Manual discloses a “computer program product ... comprising a computer useable medium having program code embodied thereon” that is “for use with a server.” (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 ¶ 77.)

The Microsoft Manual further explains that such a product is used with a server (e.g., a WINS/NetBIOS server) coupled over a computer network (e.g., a LAN, WAN, or the Internet) to a plurality of processes (e.g., instances of the Windows NT 3.5 or Windows for Workgroups 3.11 operating system running on computers). (Ex. 1003 at 29 (Windows NT Server 3.5 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 24 (“Microsoft TCP/IP-32 for Windows for Workgroups 3.11 ... can be used to provide access for Windows for Workgroups computers to Windows NT, LAN Manager, and other TCP/IP systems. Microsoft TCP/IP-32 includes DHCP and WINS client software.”); *id.* at 118-19 (explaining use for “a small LAN” and “large enterprise network”); *see* Section VI(A) (“The Microsoft Manual (Ex. 1003)”); Ex. 1002 ¶ 78.)

In addition, NetBIOS teaches that “NetBIOS on personal computers includes both a set of services and an exact program interface to those services.” (Ex. 1004 at 377; *id.* at 374 (“NetBIOS defines a software interface . . . [and] has generally been confined to personal computers to date. However, since larger computers are often well suited to run certain NetBIOS applications, such as file servers, this specification has been designed to allow an implementation to be built on virtually any type of system where the TCP/IP protocol suite is available.”); *see generally* Section VI(B) (“NetBIOS (Ex. 1004)”); Ex. 1002 ¶ 79.)

Thus, the prior art teaches a computer program product (e.g., Microsoft Windows NT Server 3.5) for use with a server (e.g., a WINS/NetBIOS server) operatively coupled over a network (e.g., a LAN, WAN, or the Internet) to a plurality of processes (e.g., instances of the Windows NT 3.5 or Windows for Workgroups 3.11 operating system running on computers). Also, the computer



program product comprises a computer usable medium having computer program code embodied in the medium (e.g., the Windows NT Server 3.5 compact disc and memory in the server). (Ex. 1002 ¶ 80.)

b. **“a. program code configured to receive the current network protocol address of one of the processes coupled to the network, the network protocol address being received by said one of the processes from an Internet access server;”**

(1) The Microsoft Manual and NetBIOS disclose “[a] network protocol address being received by said one of the processes from an Internet access server”

The Microsoft Manual discloses that every device participating 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 such as a Dynamic Host Configuration Protocol (DHCP) server assigns IP addresses automatically to network processes—including dynamically assigned addresses. (*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 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.”); *id.* at 13, 22, 62, 63, 85-120 (describing DHCP software and configuration options); Ex. 1002 ¶ 81.)

To do so, “[d]uring system startup,” a computer running WINS client software 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 then 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 acknowledgment, it enters a *bound* state and can now participate on the TCP/IP network and complete its system startup.” (*Id.* at 64; see Section VI(A)(1) (“Processing Units Obtain Dynamically Assigned IP Addresses from DHCP Servers”); Ex. 1002 ¶ 82.)

Similarly, NetBIOS discloses that p-nodes have “IP addresses ... obtained from the TCP service interface.” (Ex. 1004 at 416; see Section VI(B)(1) (“Processing Units Have Assigned IP Addresses”); Ex. 1002 ¶ 83.)

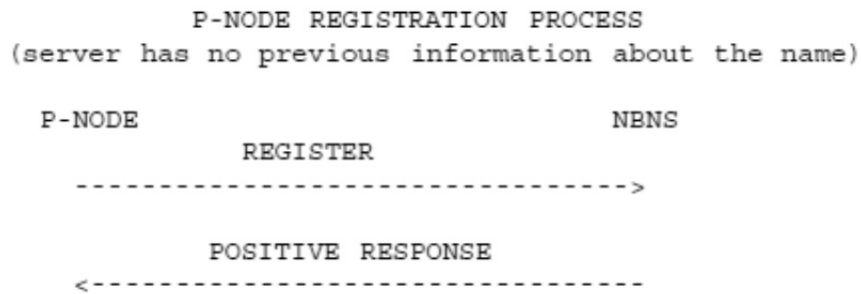
Accordingly, the prior art teaches a network protocol address (e.g., an IP address) being received by said one of the processes (e.g., the Windows NT 3.5 or Windows for Workgroups 3.11 operating system running on a computer registering with the DHCP server) from an Internet access server (e.g., the DHCP server). (Ex. 1002 ¶ 84.)

- (2) The Microsoft Manual and NetBIOS disclose “program code configured to receive the current network protocol address of one of the processes coupled to the network”

As disclosed in the Microsoft Manual, after the DHCP server dynamically assigns an IP address to the Windows NT 3.5 or Windows for Workgroups 3.11 operating system running on the first computer, the WINS client transmits the 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 (“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 (“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 (showing

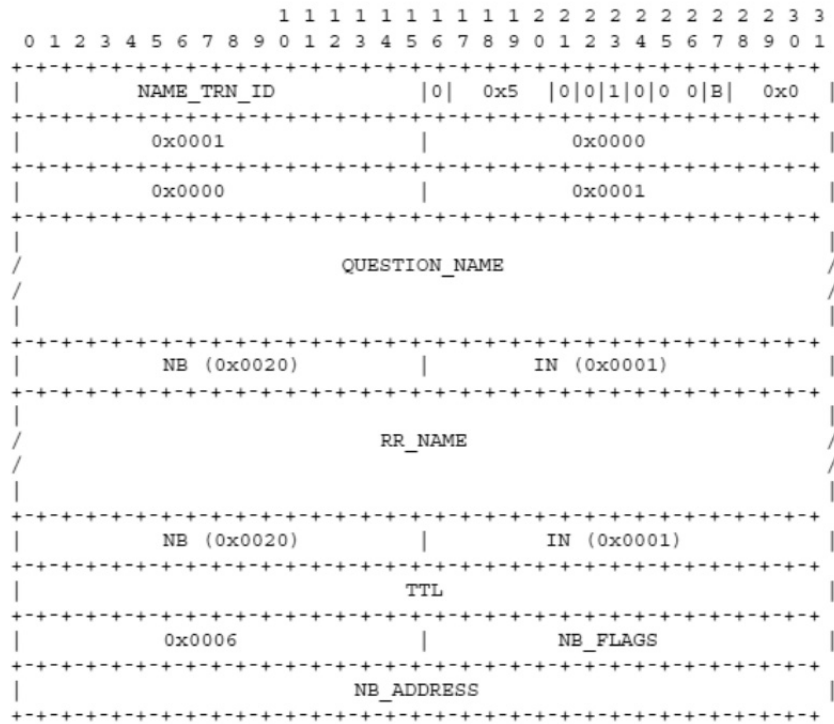
WINS database entry including “NetBIOS computer name” and “assigned Internet Protocol address”); *see* Section VI(A)(2) (“Processing Units Register Their IP Addresses and Identifiers with the WINS Server”); Ex. 1002 ¶ 85.)

NetBIOS specifies further 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”:



(Ex. 1004 at 402-03.) NetBIOS also specifies the format for node name registration (illustrated below), which requires a node to forward to the NBNS its name (“RR\_NAME”) and assigned IP address (“NB\_ADDRESS”):

4.2.2. NAME REGISTRATION REQUEST



(Ex. 1004 at 448-50; *see* Section VI(B)(2) (“Processing Units Register Their IP Addresses and Identifiers with the NBNS”); Ex. 1002 ¶ 86.)

Therefore, the prior art discloses program code (e.g., Windows NT software running on the WINS/NetBIOS server) configured to receive the current network protocol address (e.g., IP address assigned by the DHCP server) of one of the processes (e.g., the the Windows NT 3.5 or Windows for Workgroups 3.11 operating system running on the first computer) coupled to the network. (Ex. 1002 ¶ 87.)

**c. “b. program code configured to receive an identifier associated with said one process; and”**

The Microsoft Manual 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, the disclosed Windows NT software is used to assign a “NetBIOS” name to each computer during “setup,” which the WINS client then 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; Ex. 1002 ¶ 88.)

Accordingly, for these reasons, as well as the reason set forth in Section VIII(A)(3)(b)(1)-(2) in connection with the preamble and limitation “a” of claim 1, the prior art teaches program code (e.g., Windows NT software running on the WINS/NetBIOS server) configured to receive an identifier (e.g., computer name) associated with said one process (e.g., the Windows NT 3.5 or Windows for Workgroups 3.11 operating system running on a computer). (Ex. 1002 ¶ 89.)

- d. **“c. program code configured to receive queries for one of the network protocol address and the associated identifier of said one of the processes from other processes over the computer network at the server, and to allow the establishment of a packet-based point-to-point communication between said one of the processes and one of said other processes.”**

- (1) The Microsoft Manual and NetBIOS disclose “program code configured to receive queries for one of the network protocol address and the associated identifier of said one of the processes from other processes over the computer network at the server”

The Microsoft Manual teaches that: (1) WINS/NetBIOS server software “contain[s] a dynamic database mapping computer names to IP addresses” (Ex. 1003 at 65); (2) WINS client software running on the Windows NT 3.5 or Windows for Workgroups 3.11 operating system on a first computer can request the WINS/NetBIOS server to retrieve the IP address of the Windows NT 3.5 or Windows for Workgroups 3.11 operating system running on a second computer by sending the name of the Windows NT 3.5 or Windows for Workgroups 3.11 operating system running on the second computer to the WINS server in a “*name query request*” (*id.* at 73 (“If the name is found in the WINS database ....”)); and (3) the WINS/NetBIOS server will respond to the name query request by providing the Windows NT 3.5 or Windows for Workgroups 3.11 operating system running on the first computer with the IP address of the Windows NT 3.5 or Windows for Workgroups 3.11 operating system running on the second computer (if the queried

name is currently mapped 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 ....”).) Thus, “when NT\_PC1 [e.g., a first computer running the Windows NT operating system] wants to communicate with NT\_PC2 [e.g., a second computer running the Windows NT operating system], it queries the WINS server for the address of NT\_PC2 [e.g., of the first computer running the Windows NT operating system] ... [and] gets the appropriate address [e.g., of the first computer running the Windows NT operating system] from the WINS server ....” (*Id.* at 67; *see also* Section VI(A)(2) (“Processing Units Register Their IP Addresses and Identifiers (with the WINS Server)”; Section VI(A)(3) (“First Processing Unit Sends Query to WINS Server and Receives the IP Address of the Second Processing Unit”).) (Ex. 1002 ¶ 90.)

NetBIOS similarly specifies that “[n]ame query transactions are initiated by end-nodes [e.g., the first process] to obtain the IP address(es) and other attributes associated with a NetBIOS name [of the second process].” (Ex. 1004 at 406). It also explains that a “NAME QUERY REQUEST” includes the desired name (e.g.,



“QUESTION\_NAME”) of the first process. (Ex. 1004 at 458; *see* Section V(B)(2) (“Processing Units Register Their IP Addresses and Identifiers With the NBNS”) and Section V(B)(3) (“First Processing Unit Sends Query to the NBNS and Receives the IP Address of the Second Processing Unit”); Ex. 1002 ¶ 91.)

Accordingly, the prior art teaches program code configured to receive queries (e.g., Windows NT software running on a WINS/NetBIOS server, which is configured to receive name query requests) for one of the network protocol addresses (e.g., the IP address) and the associated identifier (e.g., the name) of said one of the processes (e.g., the Windows NT or Windows for Workgroups 3.11 operating system running on a first computer) from other processes (e.g. the Windows NT or Windows for Workgroups 3.11 operating system running on a second computer) over the computer network at the server (e.g., the WINS/NetBIOS server). (Ex. 1002 ¶ 92.)

- (2) The Microsoft Manual and NetBIOS disclose “program code configured ... to allow the establishment of a packet-based point-to-point communication between said one of the processes and one of said other processes”

The Microsoft Manual also teaches how the Windows NT software is used to establish a point-to-point communication with another process over the computer network: “[i]n a p-node environment,” “when NT\_PC1 [e.g., a first computer running the Windows NT operating system] wants to communicate with

NT\_PC2 [e.g., a second computer running the Windows NT operating system], 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.” (Ex. 1003 at 67; *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”); Ex. 1002 ¶ 93.)

In addition, the Microsoft Manual explains that the disclosed system supports packet-based communication, for example via TCP/IP, which “is a protocol family that can be used to offer Windows networking capabilities.” (Ex. 1003 at 54.) “IP is a protocol that provides packet delivery for all other protocols within the TCP/IP family[, and] IP packets are not guaranteed to arrive at their destination, nor are they guaranteed to be received in the sequence in which they were sent.... [The] responsibility for the data contained within the IP packet (and the sequencing) is assured only by using higher-level protocols [such as TCP]. TCP supplies a reliable, connection-based protocol over (or encapsulated within) IP. TCP guarantees the delivery of packets .... This reliability makes TCP/IP the protocol of choice for session-based data transmission, client-server applications, and critical services such as electronic mail.” (*Id.* at 55; *see* Section VI(A)(3) (“First Processing Unit Sends Query to WINS Server and Receives the IP Address of the Second Processing Unit”) and Section VI(A)(4) (“First Processing Unit Uses

Received IP Address to Establish Point-to-Point Communication with Second Processing Unit”); Ex. 1002 ¶ 94.)

NetBIOS discloses that p-nodes (such as those implemented by WINS) are “[p]oint-to-point 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 called computer].” (*Id.* at 415-16; Ex. 1002 ¶ 95.)

Accordingly, the Microsoft Manual and NetBIOS teach program code configured to allow the establishment of a packet-based point-to-point communication (e.g., via TCP/IP) between said one of the processes (e.g., the Windows NT or Windows for Workgroups operating system running on a first computer) and one of said other processes (e.g., the Windows NT or Windows for Workgroups operating system running on a second computer). (Ex. 1002 ¶ 96.)

**4. Claim 2 (Independent) Should Be Cancelled.**

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

To the extent that the preamble of claim 2 is limiting, the Microsoft Manual and NetBIOS teach it because one of ordinary skill in the art would understand that a computer data signal embodied in a carrier wave would be created, among other times, during installation of the Windows NT or WINS client software from the

CD-ROM on which it is provided. (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.”).)<sup>10</sup> (Ex. 1002 ¶ 97.)

- b. **“a. program code configured to receive a current network protocol address of a process coupled to a computer network, the network protocol address being received by the process from an Internet access server;”**

The Microsoft Manual and NetBIOS teach this limitation for the reasons set forth above in Section VIII(A)(3)(b) in connection with claim 1, limitation “a.” (Ex. 1002 ¶ 98.)

- c. **“b. program code configured to receive an identifier associated with said one process; and”**

The Microsoft Manual and NetBIOS teach this limitation for the reasons set forth above in Section VIII(A)(3)(c)-(d) in connection with claim 1, limitations “b” and “c.” (Ex. 1002 ¶ 99.)

---

<sup>10</sup> At least claim 2 is invalid because it is directed to 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) (holding claims directed to a “signal” with “embedded data” are not patentable because they do not cover patentable subject matter).

- d. **“c. program code configured to receive queries for one of the network protocol address and the associated identifier of said one process from other processes over the computer network at a connection server, and to allow the establishment of a packet-based point-to-point communication between said one process and one of said other processes.”**

The Microsoft Manual and NetBIOS teach this limitation for the reasons set forth above in Section VIII(A)(3)(d) in connection with claim 1, limitation “c.” (Ex. 1002 ¶ 100.)

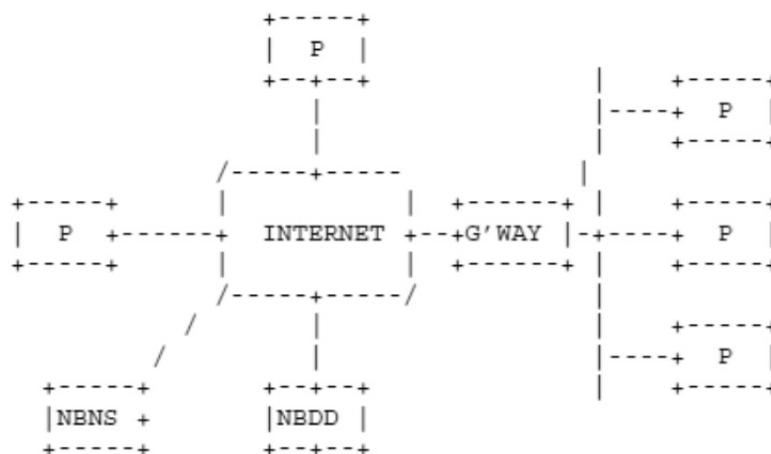
**5. Claim 3 (Independent) Should Be Cancelled.**

- a. **“In a computer system operatively coupled over a computer network to a plurality of processes, a method comprising the steps of:”**

To the extent that the preamble of claim 3 is limiting, the Microsoft Manual and NetBIOS teach it for the same reasons set forth above in Section VIII(A)(3)(a) in connection with the preamble of claim 1. (Ex. 1002 ¶ 102.)

In addition, the Microsoft Manual teaches that a WINS server is used in a networked computer system hardware environment. (*E.g.*, Ex. 1003 at 12 (“[The Microsoft 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 73 (showing computers “Corp01” and “Payroll” and WINS server); Ex. 1002 ¶ 103.)

Similarly, NetBIOS discloses the NBNS operatively coupled over a computer network to a plurality of processes:



(Ex. 1004 at 389; *id.* at 374 (“The NetBIOS service has become the dominant mechanism for personal computer networking. NetBIOS provides a vendor independent interface for the IBM Personal Computer (PC) and compatible systems. . . . NetBIOS has generally been confined to personal computers to date. However, since larger computers are often well suited to run certain NetBIOS applications, such as file servers, this specification has been designed to allow an implementation to be built on virtually any type of system where the TCP/IP protocol suite is available”); *id.* at 375 (“NetBIOS is the foundation of a large body of existing applications. It is desirable to operate these applications on TCP networks and to extend them beyond personal computers into larger hosts. To support these applications, NetBIOS on TCP must closely conform to the services offered by existing NetBIOS systems.”); Ex. 1002 ¶ 104.)

- b. “a. receiving the current network protocol address of a process coupled to the network, the network protocol address being received by the process from an Internet access server;”**

The Microsoft Manual and NetBIOS teach this limitation for the reasons set forth above in Section VIII(A)(3)(b) in connection with claim 1, limitation “a.”  
(Ex. 1002 ¶ 105.)

- c. “b. receiving an identifier associated with said one process;”**

The Microsoft Manual and NetBIOS teach this limitation for the reasons set forth above in Section VIII(A)(3)(c) in connection with claim 1, limitation “b.”  
(Ex. 1002 ¶ 106.)

- d. “c. receiving a query for one of the network protocol address and the associated identifier of said one process from another of the processes over the computer network at a connection server; and”**

The Microsoft Manual and NetBIOS teach this limitation for the reasons set forth above in Section VIII(A)(3)(d) in connection with claim 1, limitation “c.”  
(Ex. 1002 ¶ 107.)

- e. **“d. providing one of the network protocol address and the associated identifier of said one process to a said another process over the computer network, if the said one process is connected to the computer network, and to allow the establishment of a packet-based point-to-point communication between said one process and one of said another processes.”**
  - (1) The Microsoft Manual and NetBIOS disclose “providing one of the network protocol address and the associated identifier of said one process to a said another process over the computer network, if the said one process is connected to the computer network”

The Microsoft Manual and NetBIOS teach this limitation, at least in part, for the reasons set forth above in Section VIII(A)(3)(d) in connection with claim 1, limitation “c.” (Ex. 1002 ¶ 108.)

Furthermore, the Microsoft Manual teaches that the WINS/NetBIOS server keeps its mapping of names to IP addresses relatively current by tracking which users are still connected to the network. 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.) Similarly, it notes that “[w]hen 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.) In addition, the Microsoft Manual describes that, if a network device fails to re-register its name within the set renewal time, “the WINS server



will mark the name as released and available for use.” (*Id.*; *see* Section (VI)(A)(1) (“Processing Units Obtain Dynamically Assigned IP Addresses from DHCP Servers”) and Section (VI)(A)(2) (“Processing Units Register Their IP Addresses and Identifiers with the WINS Server”); Ex. 1002 ¶ 109.)

Similarly, NetBIOS teaches that “[n]ames 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 [e.g., a computer] fails to send a name refresh message to the NBNS before the lifetime expires.” (Ex. 1004 at 396; *id.* (“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. If the node defends the name then the node is allowed to continue possession. Otherwise it is assumed that the node has released the name.”); *id.* at 400-01; Ex. 1002 ¶ 110.)

NetBIOS further specifies that if the queried name (contained in a “NAME QUERY REQUEST”) is found in the server directory (*i.e.*, if the queried node is “online”), the NBNS sends a “POSITIVE NAME QUERY RESPONSE” to the calling node. (Ex. 1004 at 395, 407-08, 413, 449, 459.) The “POSITIVE NAME QUERY RESPONSE includes an “ADDR\_ENTRY record ‘represent[ing] an owner of a name,” and each ADDR\_ENTRY record includes “NB\_ADDRESS,” *i.e.*, the IP address corresponding to the name of the called computer. (*Id.*; Ex. 1002 ¶ 111.)

On the other hand, if the queried name is not found in the directory (*i.e.*, if the called node is “offline”), NetBIOS teaches that the NBNS returns a “NEGATIVE NAME QUERY RESPONSE” to the calling node. (Ex. 1004 at 395, 407, 413, 460.) The “NEGATIVE NAME QUERY RESPONSE” contains a “Name Error” signifying that “[t]he name requested does not exist” (*i.e.*, the device is not currently connected to the computer network). (Ex. 1004 at 460; *see* Section (VI)(B)(1) (“Processing Units Have Assigned IP Addresses”) and Section (VI)(B)(2) (“Processing Units Register Their IP Addresses and Identifiers with the NBNS”); Ex. 1002 ¶ 112.)

Accordingly, the prior art teaches providing one of the network protocol address (e.g., the IP address) and the associated identifier (e.g., the name) of said one process (e.g., the Windows NT 3.5 or Windows for Workgroups 3.11 operating system running on a first computer) to a said another process (e.g., the Windows NT 3.5 or Windows for Workgroups 3.11 operating system running on a second computer) over the computer network (e.g., a TCP/IP network), if the said one process is connected to the computer network (e.g., is registered as online in the WINS/NetBIOS server, the WINS/NBNS server sends a POSITIVE NAME QUERY RESPONSE (with the first IP address) to the second process; and if the first process is not registered as online in the WINS/NetBIOS server, the WINS/NBNS server sends a NEGATIVE NAME QUERY RESPONSE to the

second process)). (Ex. 1002 ¶ 113.)

- (2) The Microsoft Manual and NetBIOS disclose “allow[ing] the establishment of a packet-based point-to-point communication between said one process and one of said another processes”

The Microsoft Manual and NetBIOS teach this limitation for the reasons set forth above in Section VIII(A)(3)(d) in connection with claim 1, limitation “c.” (Ex. 1002 ¶ 114.)

**B. Claim Chart with Examples Showing Where Prior Art Teaches Claimed Features**

<b><u>Claim Limitation</u></b>	<b><u>Prior Art</u></b>
1-P. “A computer program product for use with a server operatively coupled over a computer network to a plurality of processes, the computer program product comprising a computer usable medium having program code embodied thereon the program code comprising:”	Ex. 1003 at 12, 18-19, 23-24, 27-29, 55-56, 105, 118-19, 135; Ex. 1004 at 356, 359, 368, 373-74, 377, 384-85, 389, 442.
1-A. “a. program code configured to receive the current network protocol address of one of the processes coupled to the network, the network protocol address being received by said one of the processes from an Internet access server;”	Ex. 1003 at 13, 20-23, 40-41, 57-58, 62-67, 69, 72, 74-75, 80, 83-115, 118-19, 121, 151, 262, 265-66; Ex. 1004 at 365, 378, 394-96, 398, 400-03, 412-13, 416, 430-32, 448-50.
1-B. “b. program code configured to receive an identifier associated with said one process; and”	Ex. 1003 at 41, 65, 74, 121. <i>See also</i> 1-P, 1-A.
1-C. “c. program code configured to receive queries for one of the network protocol address and the associated identifier of said one of the processes from other processes over the computer network at the server, and to allow the establishment of a packet-based point-to-point communication between said one of the processes and one of said other processes.”	Ex. 1003 at 13, 20-22, 40-41, 54-55, 64-67, 69-70, 72-75, 80, 121-22, 151, 262, 265-266; Ex. 1004 at 368, 374, 378, 383, 395-96, 400-03, 406-07, 412-13, 415-16, 422, 430-32, 438, 447, 458-59.

<b><u>Claim Limitation</u></b>	<b><u>Prior Art</u></b>
2-P. “A computer data signal embodied in a carrier wave comprising:”	Ex. 1003 at 135.
2-A. “a. program code configured to receive a current network protocol address of a process coupled to a computer network, the network protocol address being received by the process from an Internet access server;”	<i>See</i> 1-A.
2-B. “b. program code configured to receive an identifier associated with said one process; and”	<i>See</i> 1-B, 1-C.
2-C. “c. program code configured to receive queries for one of the network protocol address and the associated identifier of said one process from other processes over the computer network at a connection server, and to allow the establishment of a packet-based point-to-point communication between said one process and one of said other processes.”	<i>See</i> 1-C.
3-P. “In a computer system operatively coupled over a computer network to a plurality of processes, a method comprising the steps of:”	Ex. 1003 at 12, 73; Ex. 1004 at 374-75, 389. <i>See also</i> 1-P.
3-A. “a. receiving the current network protocol address of a process coupled to the network, the network protocol address being received by the process from an Internet access server;”	<i>See</i> 1-A.
3-B. “b. receiving an identifier associated with said one process;”	<i>See</i> 1-B.
3-C. “c. receiving a query for one of the network protocol address and the associated identifier of said one process from another of the processes over the computer network at a connection server; and”	<i>See</i> 1-C.
3-D. “d. providing one of the network protocol address and the associated identifier of said one process to a said another process over the computer network, if the said one process is connected to the computer network, and to allow the establishment of a packet-based point-to-point communication between said one process and one of said another processes.”	Ex. 1003 at 13, 23, 57-58, 62-64, 69, 75, 79, 83-115, 118-19; Ex. 1004 at 365, 394-96, 398, 400-01, 407-08, 413, 449, 459-60. <i>See also</i> 1-C.

**IX. CONCLUSION**

For the foregoing reasons, Petitioner requests institution of an *inter partes* review and cancellation of claims 1-3 of the '365 patent.

Respectfully submitted,

/Grant K. Rowan/

---

Grant K. Rowan  
Registration No. 41,278

Victor F. Souto  
Registration No. 33,458

Stacy S. Chen  
Registration No. 62,609

Matthias Kamber  
*pro hac vice* pending

**Table of Exhibits for U.S. Patent 6,701,365 Petition for *Inter Partes* Review**

Exhibit	Description
1001	U.S. Patent No. 6,701,365 (“’365 patent”)
1002	Declaration of Dr. Bruce M. Maggs (“Maggs Decl.”)
1003	Microsoft Windows NT Server Version 3.5 (“Microsoft Manual”)
1004	Technical Standard: Protocols for X/Open PC Interworking: SMB, Version 2 (“NetBIOS”)
1005	Declaration of Jason Liss Regarding Documentation of Microsoft Windows NT Server Version 3.5
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 EU S.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”)

U.S. PATENT 6,701,365  
Petition for *Inter Partes* Review

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 ’365 Patent (Apr. 19, 2002 Amendment)
1021	Excerpt from File History for ’365 Patent (July 17, 2002 Request for Reconsideration)
1022	Excerpt from File History for ’365 Patent (Feb. 13, 2002 Office Action)
1023	Excerpt from File History for ’365 Patent (June 20, 2002 Office Action)
1024	Excerpt from File History for ’365 Patent (Oct. 8, 2002 Amendment)
1025	Excerpt from File History for ’365 Patent (Dec. 31, 2002 Notice of Allowance)
1026	Excerpt from Reexamination File History for ’365 Patent (Aug. 14, 2009 Office Action)
1027	Excerpt from Reexamination File History for ’365 Patent (May 11, 2010 Notice of Intent to Issue Reexamination Certificate)

**CERTIFICATE OF SERVICE**

I hereby certify that on August 1, 2014, I caused a true and correct copy of the foregoing materials:

- Petition for Inter Partes Review of U.S. Patent No. 6,701,365 Under 35 U.S.C. § 312 and 37 C.F.R. § 42.104
- Exhibits 1001-1027
- List of Exhibits for Petition for Inter Partes Review of U.S. Patent No. 6,701,365 (Ex. 1001-1027)
- Powers of Attorney

to be served via Federal Express on the following attorney of record as listed on PAIR:

Davidson Berquist Jackson & Gowdey LLP  
4300 Wilson Blvd., 7th Floor  
Arlington VA 22203

/Victor F. Souto/  
Victor F. Souto  
Registration No. 33,458