IN THE UNITED STATES DISTRICT COURT
FOR THE WESTERN DISTRICT OF TEXAS
MIDLAND/ODESSA DIVISION

INTELLECTUAL VENTURES I LLC and INTELLECTUAL VENTURES II LLC,	§	
Plaintiffs,	§	Civil Action No. 7:24-cv-00277-ADA
v.	§	JURY TRIAL DEMANDED
SOUTHWEST AIRLINES CO.,	§	
Defendant.	§	

DEFENDANT SOUTHWEST AIRLINES CO.'S
PRELIMINARY INVALIDITY CONTENTIONS

Pursuant to the Court's Standing Order Governing Proceedings ("OGP"), Defendant Southwest Airlines Co. ("Defendant" or "Southwest") submits the following Invalidity Contentions regarding United States Patent Nos. 8,332,844 (the '844 Patent), 8,407,722 (the '722 Patent), 7,949,785 (the '785 Patent), 8,027,326 (the '326 Patent), 7,324,469 (the '469 Patent), and 7,257,582 (the '582 Patent) (collectively, the "Asserted Patents"). Southwest makes these Invalidity Contentions based on its present understanding of Plaintiffs Intellectual Ventures I LLC's and Intellectual Ventures II LLC's ("Plaintiffs" or "IV") Preliminary Infringement Contentions (the "Infringement Contentions") and the contended scope of the asserted claims.

The currently asserted claims are identified in the table below (collectively, the "Asserted Claims").

Patent	Asserted Claims[1]
'844 Patent	7 and 11
'722 Patent	14, 16, 17, and 18
'785 Patent	30, 35, and 37
'326 Patent	1, 4, and 18
'469 Patent	24, 25, 26, 28, and 32
'582 Patent	1, 2, 5, and 7

I. PRELIMINARY STATEMENT

These invalidity contentions are preliminary in nature and do not address claim construction or noninfringement. Southwest reserves all rights with respect to such issues, including its position that claims of the Asserted Patents should be construed in a particular manner and/or are not infringed. These invalidity contentions are based on Southwest's current knowledge, understanding, and belief of the Asserted Patents and prior art, of Plaintiffs' infringement theories (as disclosed in the Infringement Contentions), and of the facts and other information available as of the date of these invalidity contentions. Southwest's investigation, discovery, and analysis of information related to this action are ongoing. Additional discovery and/or orders of the Court may require Southwest to amend or supplement these invalidity contentions, and Southwest reserves the right to do so as the case proceeds. These contentions represent Southwest's good-faith effort to provide a comprehensive identification of prior art relevant to this case, but Southwest reserves the right to modify or supplement its prior art list and invalidity contentions at a later time with, or based upon, pertinent information that may be subsequently discovered.

[1] Plaintiffs' Disclosure of Preliminary Infringement Contentions, served February 10, 2025.

A. No Waiver

Nothing in these invalidity contentions is intended, nor should be construed, as a waiver of any claim construction argument, noninfringement position, or argument under Pre-AIA 35 U.S.C. § 101, 102, 103, and/or 112. Southwest's statements herein (including the accompanying claim charts) reflect Southwest's present understanding of the purported scope of the claims as alleged by Plaintiffs in their Infringement Contentions. For the avoidance of doubt, these contentions are not intended to present or argue in favor of any construction of the Asserted Claims, and Southwest reserves the right to present constructions that differ from any construction that may be alleged to underlie these contentions. These invalidity contentions do not constitute an acquiescence to Plaintiffs' interpretation of any claims to the extent they depart from the plain meaning of the Asserted Claims and from the proper scope of the claims as may be subsequently construed by the Court. Southwest disagrees with Plaintiffs' positions as to the purported scope of the Asserted Claims to the extent they seek to expand the scope of the Asserted Claims beyond their proper construction.

The Asserted Claims have not yet been construed by the Court in this case and, thus, Southwest has not yet had the opportunity to compare the Asserted Claims of the Asserted Patents (as construed by the Court) with the prior art. Southwest reserves the right to amend, supplement, or materially modify its invalidity contentions after the claims have been construed by the Court in this case. Southwest also reserves the right to amend, supplement, or materially modify its invalidity contentions in response to any claim construction positions that Plaintiffs may take, and to assert that a claim is indefinite, not enabled, or fails to meet the written description requirement of Pre-AIA 35 U.S.C. § 112 based on any claim construction position Plaintiffs may take in this case or based on any claim construction the Court may adopt in this case.

B. No Admission

Nothing disclosed herein is an admission or acknowledgement that any product accused of infringement by Plaintiffs in its Infringement Contentions (the "Accused Products"), or any of Southwest's other products or services, infringes any of the Asserted Claims.

Plaintiffs do not currently assert that any claim limitation is met under the doctrine of equivalents. However, to the extent that Plaintiffs are later permitted by the Court to amend its contentions to pursue any currently undisclosed doctrine of equivalents theories, Southwest reserves the right to supplement or amend these invalidity contentions to account for such amendments.

To the extent that any of the prior art references disclose the same functionality or feature of any of the Accused Products, Southwest reserves the right to argue that said feature or functionality does not practice any element of any of the Asserted Claims, and to argue, in the alternative, that if said feature or functionality is found to practice any element of any of the Asserted Claims, then the prior art reference demonstrates that the element is not novel, is obvious, and/or is otherwise not patentable.

Attached hereto are representative claim charts that demonstrate how the Asserted Claims of the Asserted Patents are invalid in view of certain prior art. The references cited in the attached claim charts may disclose the limitations of the Asserted Claims expressly and/or inherently. The suggested obviousness combinations may be presented in conjunction with or in the alternative to any contentions regarding anticipation. These obviousness combinations should not be construed to suggest that any reference included in any combination is not anticipatory in its own right.

C. Reservation of Rights

Prior art not included in this disclosure may become relevant. Southwest is currently unaware of the extent, if any, to which Plaintiffs will contend that limitations of the Asserted

Claims are not disclosed in the prior art identified by Southwest. Southwest reserves the right to identify other references that would have made the addition of the allegedly missing limitation to the disclosed device or method obvious or show that the allegedly missing limitation would have been known or readily apparent to one of ordinary skill in the art at the time of the invention in light of the disclosure of the prior art at issue.

Southwest provides these invalidity contentions only for the claims asserted by Plaintiffs, but reserves the right to seek invalidation of all claims in the Asserted Patents.

D. Ongoing Investigation

Southwest's investigation is ongoing, and Southwest reserves the right to amend its disclosures and document production to account for evidence uncovered as its investigation continues. Such amendments include identifying and relying on additional references that may result from Southwest's further search and analysis. Southwest reserves the right to supplement these contentions in light of any additional prior art of which Plaintiffs are aware and did not disclose to Southwest in discovery, or that might be subsequently disclosed by Plaintiffs in response to Southwest's discovery requests. Southwest anticipates issuing subpoenas to third parties believed to have knowledge, documentation and/or corroborating evidence concerning some of the prior art listed herein and/or additional prior art. These third parties include, but are not limited to, the authors, employers of authors, inventors, assignees, or former or current employees of assignees, of the references and/or products identified in these invalidity contentions. Southwest reserves the right to supplement these contentions in light of any newly discovered information produced by these or other third parties from which Southwest may seek discovery.

Additionally, because discovery is ongoing, Southwest reserves the right to present additional prior art references and/or disclosures under 35 U.S.C. §§ 101, 102, and/or 103, located during the course of such discovery or further investigation, and to assert invalidity under Pre-AIA

35 U.S.C. §§ 102(c), (d), or (f), to the extent that such discovery or investigation yields information forming the basis for such invalidity.

Based on Southwest's investigation and knowledge developed to date, and pursuant to the Court's OGP, Southwest: (1) serves, attached as exhibits hereto, charts setting forth where in the prior art references each element of the Asserted Claims are found, (2) identifies any limitations Southwest contends are indefinite or lack written description under § 112, and (3) identifies any claims Southwest contends are directed to ineligible subject matter under § 101. Southwest's § 101 contentions (1) identify the alleged abstract idea, law of nature, and/or natural phenomenon in each challenged claim; (2) identify each claim element alleged to be well-understood, routine, and/or conventional; and (3) to the extent not duplicative of §§ 102/103 prior art contentions, prior art for the contention that claim elements are well-understood, routine, and/or conventional. Southwest has produced with these invalidity contentions the prior art referenced in the invalidity contentions. As for technical documents, including software where applicable, sufficient to show the operation of the accused product(s), IV has accused Southwest of using certain open-source software programs that are not owned or controlled by Southwest but are publicly available here.

https://docs.docker.com/build-cloud/

https://kubernetes.io/docs/home/

https://kafka.apache.org/documentation/

https://hadoop.apache.org/docs/current/

https://spark.apache.org/docs/latest/

IV has not accused specific implementations of those programs at Southwest. To the extent that IV does at some point identify specific uses or implementations at Southwest, Southwest will supplement the production to include any additional available technical details, if any. For the '469

and ‘326 patents, Southwest has produced its contracts but does not have technical documents. Those documents are in the possession, custody and control of Viasat and Anuvu.

II. THE ASSERTED PATENTS

Southwest sets forth a summary of its current understanding of the state of the art for the general subject matter of the Asserted Patents. Southwest reserves the right to rely on each of the prior art references disclosed below and charted in the attached Exhibits. Southwest also reserves the right to rely on the discussions of the state of the art and prior art for the Asserted Patents and their respective file histories, including any prior art cited in the file histories, in explaining the state of the art and the ways that each respective reference corresponds with the Asserted Claims. Southwest further expressly reserves the right to supplement its summary of the state of the art, including for example, by information from any of the authors, named inventors, or assignees of any of the prior art references, by personnel familiar with products based on any of the prior art, or by technical experts retained on behalf of any party.

A. Overview of the ‘844 Patent

The ‘844 Patent initially describes clustered computing involving multiple compute nodes with Operating System (“OS”) software image management, where nodes are activated by diskless booting protocols or remote software installation to local storage, and where a boot image is required for each node in the cluster – and states that storing a boot image for each server results in wasted storage, and copying the entire master image at boot results in long boot times. ‘844 Patent, 1:31-67. The Background also discusses branching-store file systems where a read-only “root” image is accessible by all nodes, and changes made by a node to the root image are stored in a “leaf” image unique to that node – with a filter operating between the nodes and the OS file system(s) that merges the leaf images with the root image and delivers the result to the appropriate

node (which the node views as its own cohesive instance of the application environment). *Id.* 2:13-24.

The 844 patent's system, as shown in Fig. 1, includes a general purpose computing system environment, such as compute node 100, which includes at least one processing unit 102 and memory 104). *Id.* 4:27-36. Nodes may also have additional features, including additional removable 108 and/or non-removable 110 storage, communications connections 112, input devices 114, and output devices 116. *Id.* 4:36-5:10. Memory 104, removable storage 108, and non-removable storage 110 are all computer storage media, which includes "any…medium which can be used to store the desired information and which can be accessed by compute node 100." *Id.* 4:46-55.

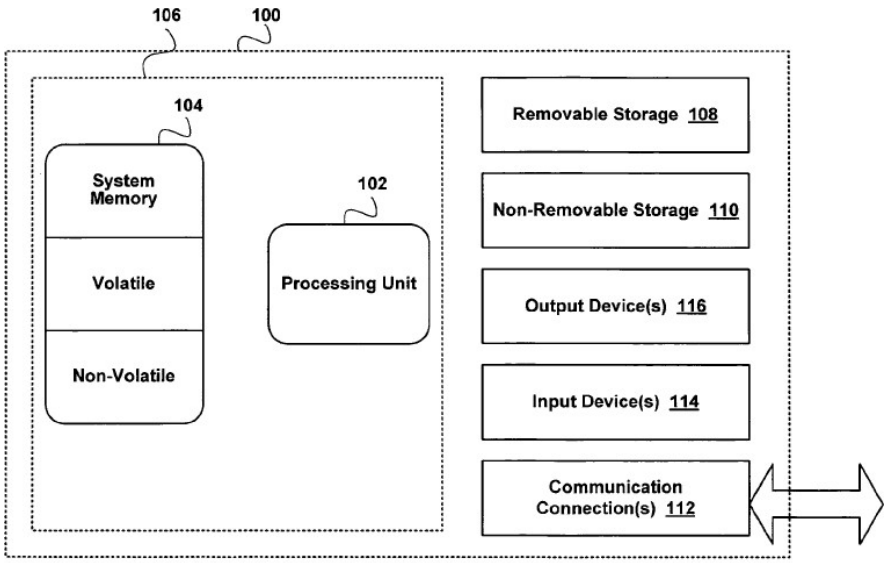


FIG. 1

Fig. 2 diagrams system 200 (e.g., a multi-computer system) having a number of compute nodes 220a-n coupled to a first storage unit 240 and a corresponding second storage unit 250a-n through a corresponding union block device ("UBD") 230a-n. '844 Patent, 5:12-22. To the

compute nodes 220a-n, it appears that they have access to their own version of the application environment. However, a separate boot image is not created and stored for each node. *Id.* 5:22-2.

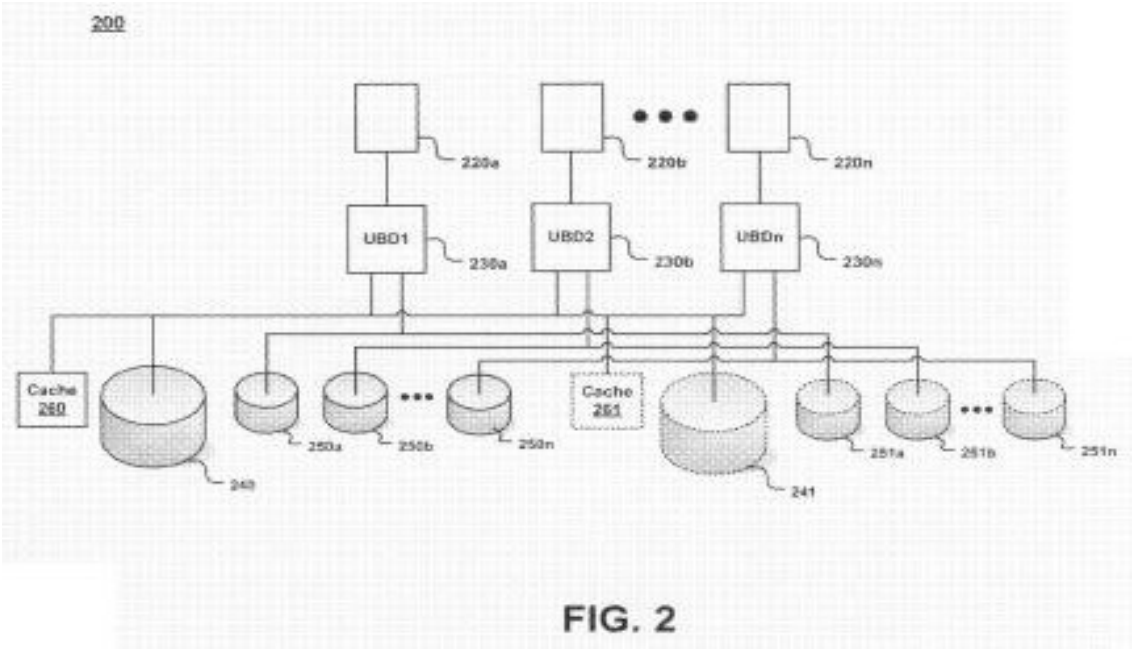


FIG. 2

Instead, a first storage unit 240 stores blocks of the root image. The root image contains data initially common to the compute nodes 220a-n, and is not changed by them. *Id.* 5:27-32. Each compute node 220a-n also has access to a corresponding second storage unit 250a-n for storing a leaf image, which in the challenged claims includes "only additional data blocks not previously contained in said root image and changes made by respective compute nodes to the blocks of [said/the] root image…" or similar language. *Id.* 5:33-34, 10:61-65, 11:32-35, 12:4-8, 12:47-50. The first storage unit 240 and second storage units 250a-n may each be contained on separate physical storage devices, on separate logical spaces on the same storage device, or any combination thereof. *Id.* 5:34-38.

A compute node 220a-n mounts its instantiation of the application environment via its respective UBD 230a-n which, in one embodiment, are "effectively low level drivers that operate as an interface between the first and second storage devices and the file system of each compute

node 220a-n." *Id.* 5:59-64. UBDs 230a-n determine what "leaf image" (from the appropriate second storage unit 250) is needed for portions of the image that their respective compute nodes 220a-n have changed, and locate the portions of the image not changed by their respective node (which may reside in the root image or intermediate images comprising versions of the root image, not depicted in Fig. 2). '844 Patent, 6:4-13. UBDs 230a-n also create a new leaf image on a respective second storage unit 250a-n when their respective compute nodes 220a-n make changes to their instantiations of the image (and may also modify the leaf image in response to their respective compute node's access to its instance of the image). *Id.* 6:13-22.

The blocks of the root image, leaf image, and cache (discussed below) are merged "on the fly" to create the node's image. *Id.* 8:13-15. From the node's perspective, the created image appears as one cohesive image even though a separate and complete image is not actually stored for the node. *Id.* 8:18-55. This saves space without significantly impacting load times. *See id.* 2:51-57.

The '844 patent also claims to further improve speed by caching portions of the root image in a cache 260 (as several nodes in a cluster may often access the same root image data). '844 Patent, 2:59-61, 6:38-43. This may shorten times to load blocks by as much as 90%. *Id.* 6:43-46. Because cache capacity is typically limited, older data can be removed (as the amount approaches a threshold) to make way for new data by techniques "known in the art." *Id.* 7:1-16.

The 844 patent also provides an "added benefit" by which one compute node can index the contents of the (shared) root image and provide the results to the other compute nodes (e.g., by storing results on a shared storage unit or providing the results directly). *Id.* 7:23-31.

B. Overview of the '722 Patent

The '722 patent "pertains ... to transferring information for remotely updating content at client devices through the digital networks." '722 Patent, 1:24-27. For example, claim 14 recites:

14. A method comprising:

providing, using a processing device of an input source, a data representation to a client device, different from the input source, coupled to a routing network, wherein the data representation includes at least one live object recognizable by the client device, and causing the client device to respond to the live object of the data representation by determining an object identifier (ID) of the live object and to register for updates of the live object with the routing network, such that registering the client device with the routing network provides client connection information to a node in the routing network; and

sending, using the processing device of the input source, an update message to the routing network, wherein the update message identifies the live object and contains update data that updates a property of the live object,

wherein a gateway device at the routing network is configured to identify a category of the update message based on the input source, to determine a node type to which the identified category maps, and to route the update message to the node, having the node type, at the routing network,

wherein the node is configured to identify the client device as a registered device and to route the update message to the client device, and

wherein the client device processes the update message upon receipt to update the property of the live object at the client device.

Fig. 1 (annotated below) illustrates the input source (e.g., dynamic content provider 116, information provider 108, server 112 alone or combined), client device 114 (e.g., PC, phone, PDA, web-browsing appliance, other devices), routing network 110, and registry 125 identifying clients who have registered for live objects, and a "web page" (which can be any text, data, and/or programs) to clients 114 over communication links 117 (e.g., network, Internet). *Id.* at 4:11-12, 4:55-5:9, 6:19-30, 6:53-60, 10:14-23. A few servers and clients are shown in Fig. 1, but there could be thousands or even millions. '722 Patent, 4:60-63.

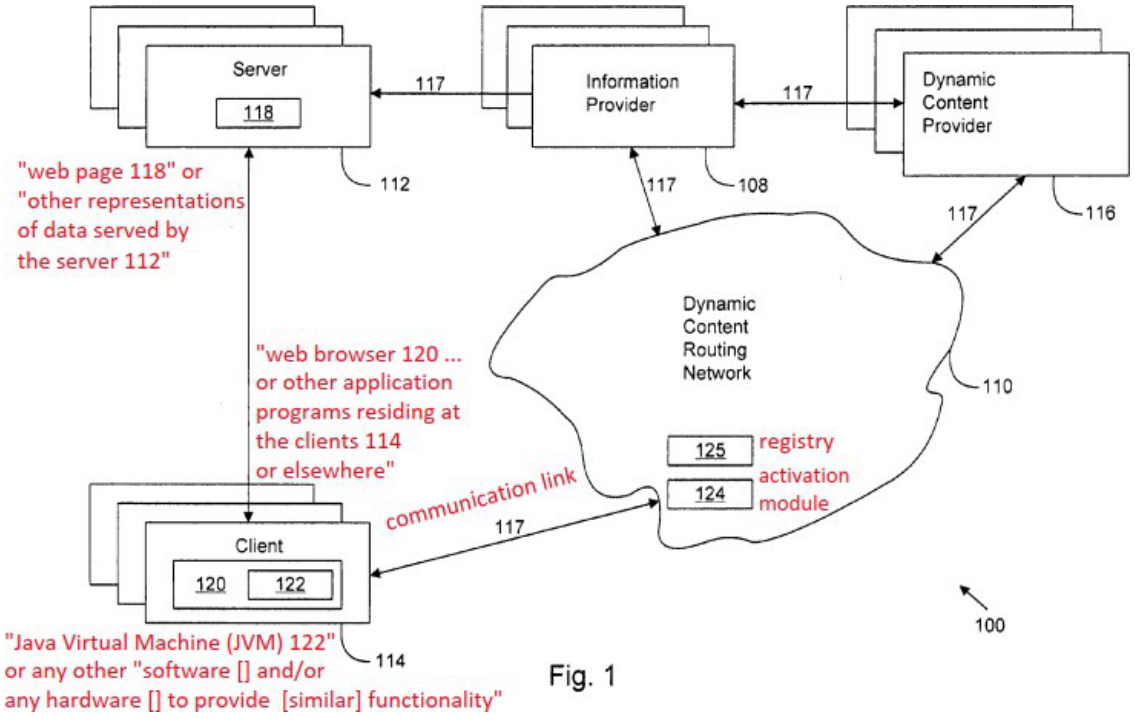


Fig. 1

The "routing network [] routes messages [from the input source (e.g., information/content provider or its server)] containing data for updating properties of live objects to clients displaying web pages or other representations of data." *Id.* at 3:9- 12, 6:22-29; *see also* 11:21-43, 12:24-25, Figs. 2, 3. "The web server that initially provides the web pages to the clients does not need to track which clients are currently displaying the live objects. Instead, the ... input source [] ... simply sends an update message to the routing network" and "the update messages are provided to the clients only when the live object changes." *Id.* at 3:12-18. The "'live objects,' which are designated to be real-time dynamically-updateable objects[, are] identified by an object identifier, or object ID." 5:28-31. The "objectID [] is preferably a string. The string can encode any information ... [and can be] a simple textual and/or numeric identifier" (e.g., "Bang$" for the stock symbol "BANG"). *Id.* at 8:34-37, 8:62-65.

"The clients 114 contact the routing network 110 and register for update messages for the objectIDs [of the live objects] on the web page[/'other application programs residing at the clients 114 or elsewhere']. The routing network 110, in turn, preferably maintains a registry indicating which clients have registered 35 for which object IDs." '722 Patent, 5:32-36, 6:58-59. Input sources send update messages "any time the[y] want[] to update a property of a live object. Each update message preferably identifies a live object [by object ID] and contains data for updating a property of the identified live object. The routing network 110 accesses the registry and determines which clients have registered for the identified object. Then, the routing network 110 routes the update message to the appropriate clients. Upon receipt of an update message, the clients 114 update the specified property of the live object." *Id.* at 5:41-48, 9:59-62. The "property" to update "include[s] any modifiable data related to the [live] object or referenced with respect to the object" (e.g., updates to "stock prices and/or sports scores" or any other data at the client which "may be a simple

numerical or textual value, e.g., '4'"). *Id.* at 7:23-24, 8:57-63, 9:53-56. Fig. 4 shows two differently-formatted webpages 410, 412 with the same "scores object 416" set to the score of the game "42" to "7." *Id.* at 12:37-46, 13:3-18.

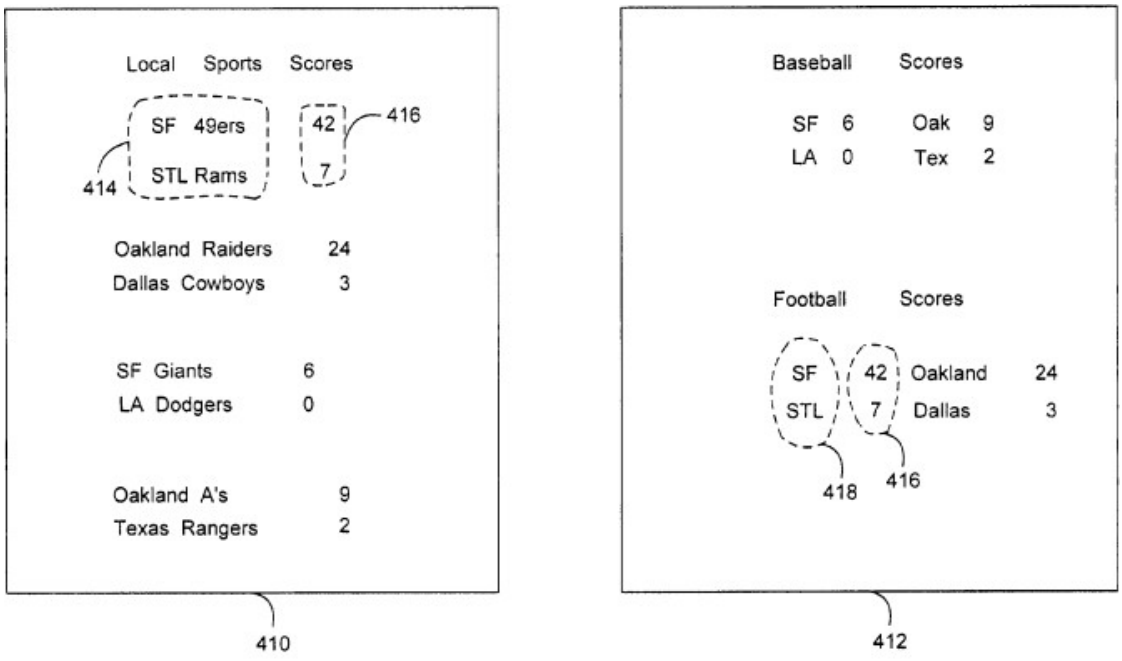


Fig. 4

"[T]o update the score of ... the game on both web pages, the input source 210 simply sends an update message to the routing network 110 specifying the objectID for the scores object 416 and the update to the score property. The routing network 110 routes the update message to the appropriate clients 114, and the clients update the appropriate score regardless of the particular page layout." *Id.* at 13:18-25.

"The routing network is adapted to selectively send messages to the nodes in the network" via a gateway. '722 Patent, 3:21-27. In an embodiment, updates "from an input source are assigned to one or more of N categories" and "the nodes are assigned to one or more of M types, and mappings are created between message categories and node types. Each gateway keeps track of these mappings" (e.g., category 1 data sent to type 1 nodes). *Id.* at 3:39-46; *see also* 18:24-51, Fig.

9. One way the '722 patent teaches assigning messages to categories "is to assign all message from a given input source [(e.g., information/content provider or its server] into a certain category." *Id.* at 18:52-53, 6:22-29. "When a gateway receives messages from input sources, the gateway identifies the categories of the messages and routes the messages to the nodes of the types to which the categories are mapped," and the nodes send the update messages to registered client devices. *Id.* at 3:43-46, 18:66-19:5.

C. Overview of the '785 Patent

The '785 patent claims a "virtual network manager [or system]" that includes several components: (1) a computing device or "network interface configured for data communication," (2) a "register module configured to register devices in a virtual network," (3) a router or route director, and (4) a "DNS server configured to receive a DNS request" from a first (source) device and return three addresses to the source, including "[(i)] a network address associated with a network route director, [(ii)] a private network address associated with a second [destination] device in the virtual network, and [(iii)] a virtual network address associated with the second [destination] device." '785 Patent, 36:37-56 (claim 30).

Representative claim 30 is below:

> [30.0] A virtual network manager, comprising:
>
> [30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;
>
> [30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:
>
> [30.3] receive a registration request from an agent associated with a device;
>
> [30.4] distribute a virtual network address to the device when

> the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and
>
> [30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.

The '785 patent identifies the so-called IP address depletion problem as a challenge it was trying to solve to "enable the rapid creation of secure means that allows local and remote specified entities [computer devices] to communicate and collaborate from any location via a standard Internet connection." '785 Patent, 1:32-37. The '785 patent indicates that, due to its own popularity, the Internet is "running out of [IP] addresses" that can be assigned to each device, thereby limiting the ability for devices in private networks, for example, to communicate or be accessible via the Internet. *Id.* at 1:50-55. The '785 patent also purports to address the "mobility problem," involving devices being assigned constantly changing/dynamic IP addresses, which other devices on the Internet wishing to communicate with them might find difficult to track. *Id.* at 5:10-45. The '785 patent thus describes "a need for a system that provides for local and remote entities to communicate and collaborate using the Internet, [a system that] can work with existing NAT [(Network Address Translation)] devices and firewalls, and allows for devices to move to different physical networks." *Id.* at 2:45-48.

In the "Description of the Related Art" section, the '785 patent admits that NAT solutions where well known the art, and used to transparently translated between private and public IP address:

> One popular solution to the depleting address problem is [NAT]. This concept includes predefining a number of network addresses to be private addresses and public addresses. Public addresses are unique addresses that should only be used by one entity having access to the Internet. That is, no two entities on the Internet should have the same public address. Private addresses are not unique and are typically used for entities not having direct access to the Internet. Private addresses are used in private networks. Private addresses can be used by more than one organization or network. NAT assumes that all of the machines on a network will not need to access the Internet at all times. Therefore, there is no need for each machine to have a public address. A local network can function with a small number of one or more public addresses assigned to a NAT device. The remainder of the machines on the network will be assigned private addresses. Since entities on the network have private addresses, the network is considered to be a private network.

'785 Patent, 1:56-2:7.

One embodiment discloses a "secure Virtual Community Network or ('VCN')," described as "a private dynamic network which acts a private LAN for computing devices coupled to public networks or private networks." '785 Patent, 8:66- 9:5.

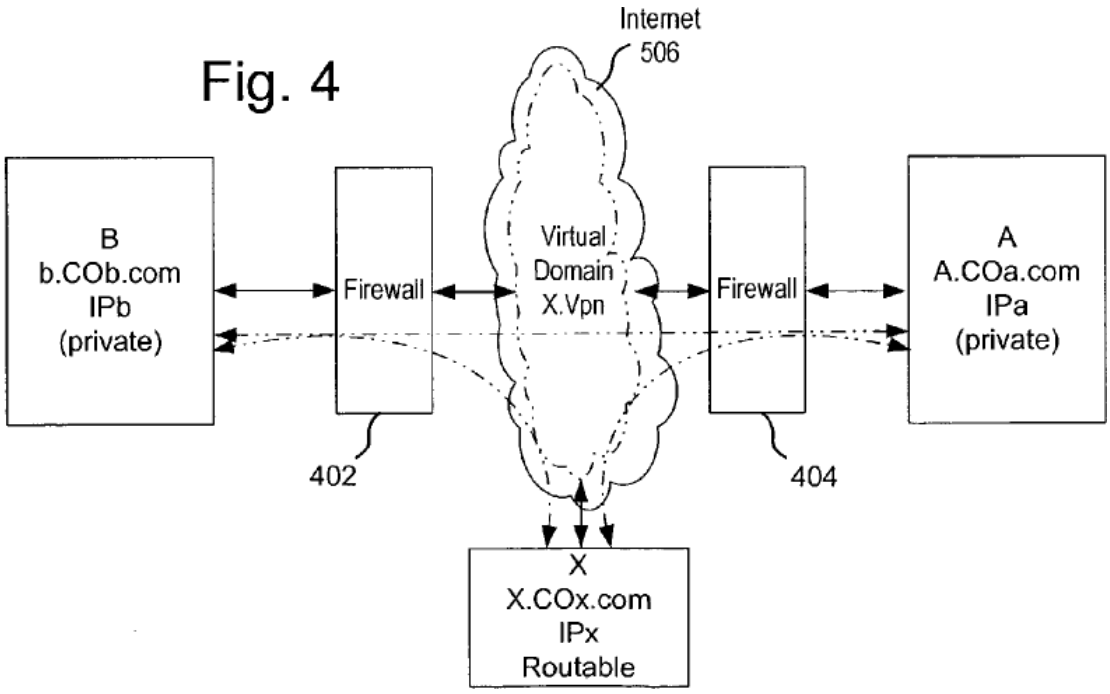


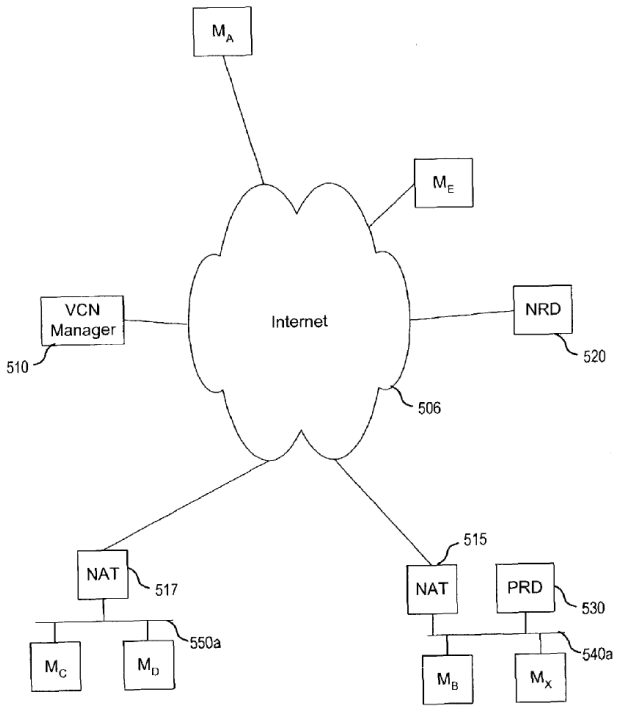
Figure 4 shows an exemplary VCN, which comprises a "computer or device B (host name—b.Cob.com) in a first private domain and computer or device A (host name—a.Coa.com) in a second private domain, both of which are coupled to the Internet by firewall devices 402, 404." *Id.* at 9:9-13. The firewall devices are NAT devices. *Id.* at 9:13-15. Moreover, "[c]omputer or device X is coupled directly to the Internet and has a public IP address." *Id.* at 9:16-17. "Devices A, B, and X can join the VCN, leave the VCN, or allows other devices in the VCN to communicate with them." *Id.* at 9:18-23. "All members of the VCN are accessible as if they were part of a physical local network" (e.g. all in the same private network)—allowing direct communication between devices within the virtual domain. *Id.* at 9:22-27.

The '785 patent states that "hardware architecture for the machines, servers, or other devices used to implement the present invention should be well understood to one of ordinary skill in the art," and outlines standard, readily available hardware/software components, including "one or more processors, a memory, a mass storage device, a portable storage device, a first network interface, a second network interface, and I/O devices, in communication with each other." '785

Patent, 9:42-48; *see id.* 9:48-54. The patent also states the network interface can include or be connected to a firewall. *Id.* at 9:57-60. The network interfaces are inside a computer, and when the computer also functions as a "router," it "includes two or more network interfaces;" *h*owever, "in other embodiments, the computer may include only one network interface." *Id.* at 9:54-56.

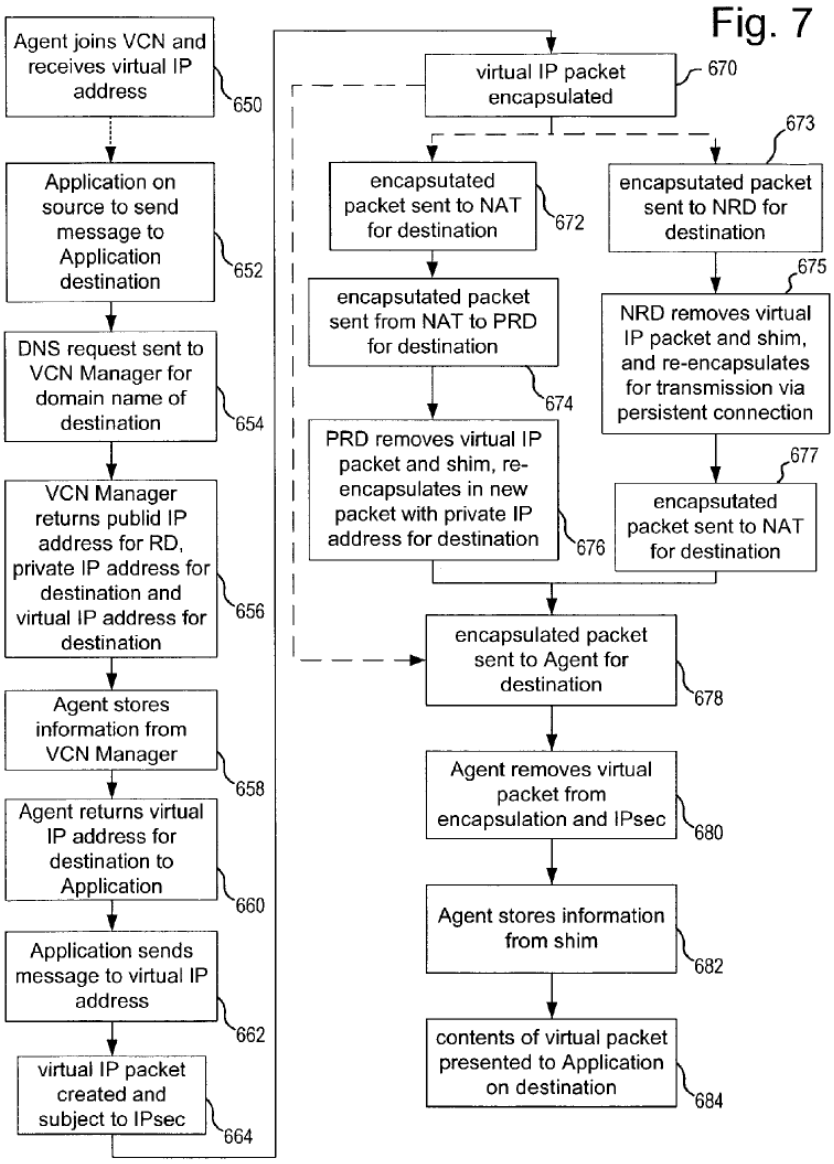
Another example of the system is illustrated in Figure 5A.

Fig. 5a



The VCN includes a VCN Manager 510, a Network Route Director 520 and/or Private Route Director 530, and a number of user devices (e.g., M_C, M_X, M_A etc.). .EX 1001, 10:24-27. "The VCN Manager 510 is a central server or server cluster providing management, connection, and security services for the VCN." ' 785 Patent, 10:66-11:1. The "various user machines, Mn, are coupled to the Internet 506, as well as other devices." '785 Patent, 10:17-20. "In order to establish or participate in communication within the VCN, a device must register with and join the VCN, thereby becoming a member." To join, "[t]he Agent will send a message to the VCN Manager 510

and receive a virtual IP address in return." *Id.* at 14:10-15. This is shown in step 650 of Figure 7. *Id.*



The '785 patent describes the virtual IP or network address as preferably another private network address that is chosen in a well-defined set of addresses (virtual address realm) such that each virtual network address occurs only once across the virtual network and thus is a unique identifier of the device within the network address. *Id.* at 12:50-54 ("each member is assigned a virtual IP address unique in the context of the virtual community and that will not conflict with

any local addresses on its LAN or any public addresses on the Internet."); *Id.* at 12:48-50 ("In general, when choosing virtual addresses it is best to choose addresses that will not be routable to the member."). To that end, the '785 patent describes this virtual network address in broad terms, adding that it can be a string, or an address compliant with either IPv4 or IPv6 standards. *Id.* at 12:40-44 ("Assigning a virtual address to each peer designated by a DNS string solves the problem of ambiguous endpoints ...These virtual addresses may be any legal IPv4 addresses.") As shown in Figure 7, a new member device wanting to send a message to another device in the VCN "will initiate a DNS request," which is then sent to the VCN Manager 510, as shown in step 654. *Id.* at 14:32-34. In response, the "VCN Manager 510 returns the public address of the Route Director for the destination, the private address for the destination device, and a virtual IP address for the destination device," as shown in Step 656. *Id.* at 14:34-38.[2]

This information is then stored in a table, or any other data structure" for creating a virtual IP packet to be forwarded to the destination address. '785 Patent, 14:48-15:15. When the destination wishes to respond to the source, or the sources wishes to send additional information, the process of Figure 7 can be repeated, although the steps of performing the DNS request do not need to be performed again, since the information between the two devices has already been stored by the devices. *Id.* at 15:58-67.

D. Overview of the '326 Patent

The '326 Patent concerns "a system and method for high data rate multi-channel wireless communications" which reuses "existing single channel radio designs" to "achieve very high data rate expanded bandwidth (wide band) WLAN operations." *See* Dkt#1 ¶ 15. Particularly, the '326

[2] The 785 specification notes that the three addresses are not always returned, for instance, in the case where "the destination machine has a public address (and, thus does not use a routing director) then step 656 will only include returning the public IP address for the destination." '785 Patent, 14:40-44.

patent seeks to provide "[a] method for increasing data rates and data throughput in a network" by "partially filling the frequency gap between the first channel and the second channel by adding one or more data subcarriers into the frequency gap such that the one or more guard bands are at least partially filled with at least some of the one or more data subcarriers," "combining the first channel and the second channel using channel bonding with orthogonal frequency division multiplexing," and "transmitting data subcarriers occupying the first channel, the second channel, and the frequency gap in parallel to a receiver." *Id.* at 1:1-24. The patent contemplates "a way to achieve very high data rate expanded bandwidth (wide band) WLAN operations reusing existing single channel radio designs" that "provides a dual-channel form of operation for greater flexibility and performance across multiple platforms." Dkt#1-4 at 7:59-63.

See Figure 4 of the '326 Patent below:

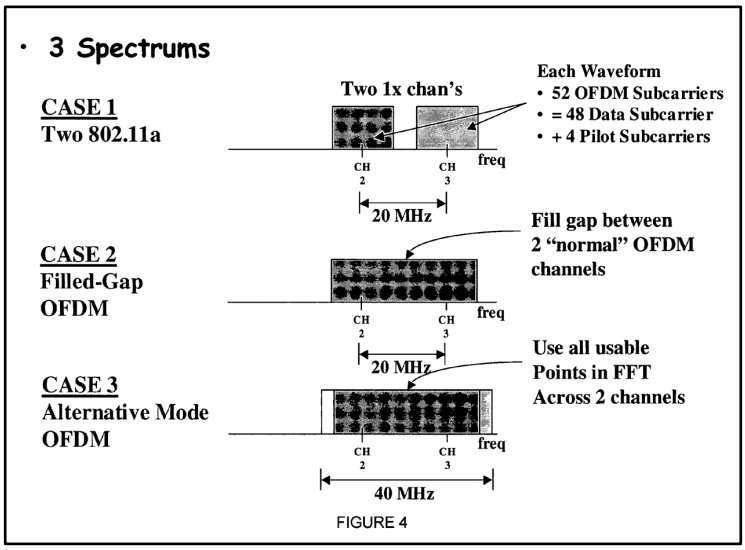


FIGURE 4

E. Overview of the '469 Patent

The '469 Patent concerns installing a satellite distributed Internet hotspot in a "remote location" or "rural area" that "experience[s] high volume transient traffic, such as rest area, restaurants, truck stops, rural hotels, conference centers, motels and state park lodges." '469 Patent at Abstract, 3:52-55. The specification and claims recite placing a "satellite dish 14 for

communicating with the satellite, a subscriber access unit 16 operatively coupled to the satellite dish 14 and a router 18 operatively coupled to the subscriber access unit"—"at the rural location," meaning "anywhere there is 120 volt electricity available or access to the sun for a solar panel and enough space to house the transceiver and mount a satellite dish." *Id*. at 1:39-41, 4:21-25, 4:18-28.

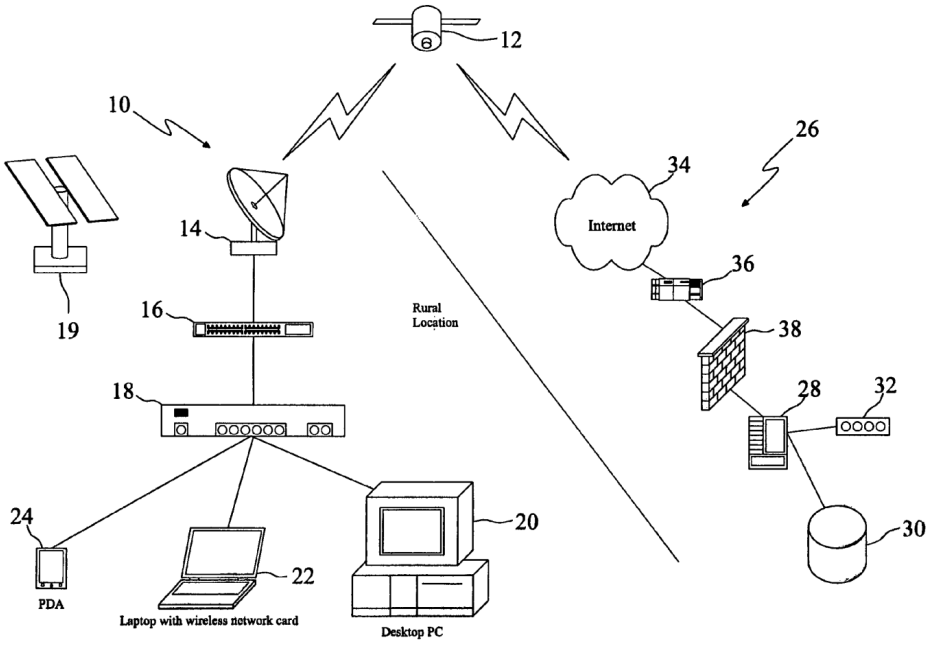


FIG.1

As shown in Figure 1 of the '469 Patent, the invention contemplates connection to the hotspot by "user web-enabled devices, such as PC 20, laptop computer 22 and PDA 24" (*id.* at 4:29-30), upon which

> they are routed through the 'Subscriber Access Unit.' This unit, utilizing radius client, automatically forces the user to first logon via satellite through a third party commercial site that houses the radius server. Internet access is granted upon activation of a valid credit card, or existing valid account.

'469 Patent at 4:2-7; *id.* at 4:32-34 ("The user can access the Hotspot using his own web-ready device to acquire a dynamic IP address").

F. Overview of the '582 Patent

The '582 Patent relates to load balancing using multiple computer processors. '582 Patent, 1:14-17. Particularly, '582 Patent deals with "a way to split the performance of a given task among a plurality of processing units which can all access, directly or indirectly, the input data and the devices on which the output data is to be stored." *Id.* Further, "[t]he principal object of the present invention is to enable the decomposition of a certain type of linear processes that currently use a single computer, into equivalent parallel processes that can efficiently use any number of potentially heterogeneous computers, taking the available capacity of each of these computers into account while optimizing execution." *Id*. at 1:48-54. See FIG. 1 of the '582 Patent, below.

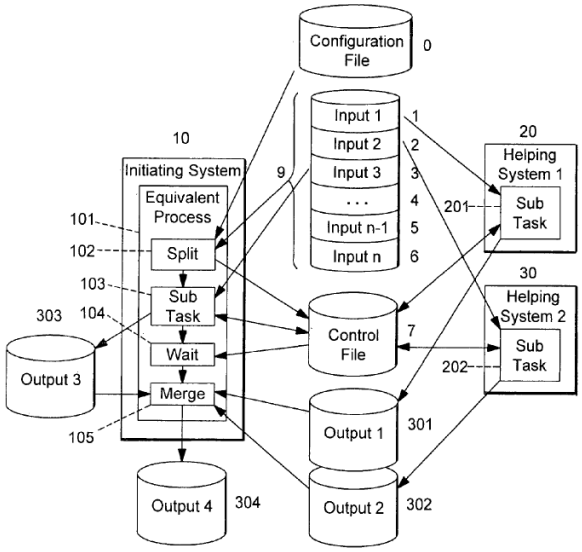


Fig. 1

* * *

Each of the concepts discussed above, as well as many others, were widely known by the alleged invention dates of the Asserted Patents. Accordingly, the Asserted Patents are nothing

more than a recitation of knowledge that was well-known in the industry, and are, at best, a recitation of well-known features and functionalities that took no more than an ordinary level of skill in the art to arrange in a known manner using conventional techniques.

III. GROUNDS OF INVALIDITY UNDER 35 U.S.C. § 101

Pursuant to the Court's OGP, Southwest identifies the following Asserted Claims of the Asserted Patents as ineligible under 35 U.S.C. § 101. In addition to the specific discussions below, the specification and prosecution history for each asserted patent reflect statements and admissions concerning the state of the art at the time of the purported inventions, and the prior art charted and discussed in these contentions for purposes of invalidity under 35 U.S.C. § 103 is relevant for consideration regarding the state of the art at the time of the purported inventions.

A. U.S. Patent No. 8,332,844

1. State of the Art Relevant to the '844 Patent

By the earliest priority date of December 30, 2004, systems administrators and infrastructure engineers working in distributed computing environments had access to a range of well-understood, conventional technologies for provisioning and managing operating system images across multiple compute nodes. These technologies were not only theoretically described but actively implemented and in widespread use across educational institutions, research laboratories, enterprise networks, and government systems. Understanding this state of the art is critical to evaluating the Asserted Claims of the '844 Patent, as it establishes that the techniques described in the patent were already conventional, mature, and widely implemented in the relevant field.

Distributed system architectures at the time were designed to deliver consistent runtime environments to groups of machines—whether stateless thin clients, HPC nodes, or clustered servers—by employing shared root filesystems, overlay filesystems, and caching mechanisms.

These established practices optimized storage, consistency, and performance and were actively documented and deployed in real-world systems.

a. Distributed Image Management in 2004

By December 2004, several foundational techniques in distributed image and storage management were widely recognized, well understood, and routinely implemented across both academic and commercial systems. These included the use of snapshots, copy-on-write (CoW) mechanisms, merging delta changes with base images, shared base images with writable overlays, and efficient virtual machine migration strategies.

One of the most prevalent techniques was snapshotting using Copy-on-Write. Storage systems like NetApp's Write Anywhere File Layout (WAFL) architecture leveraged CoW to create space-efficient, consistent, point-in-time copies of file systems. WAFL enabled the creation of read-only snapshots and writable clones that separated static base data from dynamic changes without duplicating the entire file system. The concept was not limited to NetApp; as early as 1992, the Episode File System integrated snapshots and file versioning with CoW in distributed environments supported by the Distributed Computing Environment (DCE). In the virtualization space, VMware's early GSX and ESX platforms also relied on CoW to manage snapshots of virtual machines, tracking changes made after a snapshot was taken while preserving the original disk image.

Alongside snapshotting, it was routine to merge snapshots with base images to present a unified view of the file system or virtual machine state. For example, VMware systems combined base virtual disk images with delta files to construct a current VM state, a practice commonly used by system administrators by the early 2000s. Similarly, NetApp technologies described in patents such as U.S. Patents 5,819,292 and 6,604,118 disclosed mechanisms for merging base data with CoW-based snapshots to provide a consistent and current data view.

In networked and high-performance computing environments, the use of shared read-only base images with writable overlays was equally well established. Systems such as Sun Microsystems' diskless workstations and the Linux Terminal Server Project (LTSP) enabled multiple clients to boot from a single centralized read-only root image while maintaining isolated writable overlays. Academic and research environments employing Beowulf clusters or Rocks Cluster Toolkit adopted similar strategies, allowing efficient management of large numbers of nodes while minimizing image duplication. NetApp further enabled writable clones of read-only snapshots, as disclosed in U.S. Patent 7,334,095.

The migration and management of virtual machines—including their associated image states and snapshots—was also well understood by 2004. Research by Sapuntzakis *et al.* at Stanford University demonstrated the migration of virtual machines using CoW disks and demand paging to efficiently capture and transfer VM state across systems. These techniques mirrored the commercial practices already in use by companies like VMware and reflected broader industry familiarity with image management in virtualized environments.

b. Real-World Implementations Demonstrating the State of the Art

Several systems operational before or during 2004 embodied the above techniques, demonstrating that they were not only theoretically known but also widely implemented and recognized in practice.

- NetApp's WAFL File System exemplified the state of the art with its CoW-based snapshot architecture. WAFL enabled the creation of read-only snapshots, writable clones, and consistent file system views without duplicating the base image. This architecture was thoroughly documented in both U.S. Patents 5,819,292 and 7,334,095, as well as in technical literature such as "File System Design for an NFS File Server Appliance" by Dave Hitz et al.

- VMware's GSX and ESX server platforms provided commercial-grade virtual machine management, including snapshotting, delta image tracking, and the merging of these deltas with base images. This allowed users to roll back or move between states while minimizing disk usage.
- Sun Microsystems' diskless workstations leveraged centralized read-only images to boot multiple clients across a network, each maintaining private writable layers. These overlays captured client-specific changes while preserving the shared base image intact.
- The Episode File System, developed in the early 1990s, brought snapshotting and version control to distributed file systems using CoW. This system efficiently tracked and stored changes to files across versions in a distributed environment.
- Petal, a distributed virtual disk project from HP Labs, provided dynamic, fault-tolerant storage with support for snapshot creation and management using CoW techniques. Petal's architecture demonstrated the application of distributed block storage principles in combination with image versioning and migration features.

Taken together, these examples show that by December 2004, the claimed techniques for managing images, snapshots, and overlays in distributed systems were neither novel nor inventive—they were standard practice in both the commercial and academic realms.

c. Conclusion:

By the earliest priority date of December 30, 2004, the concepts underlying the Asserted Claims of the '844 patent were well understood, routine, and conventional within the field of distributed system image management. The use of snapshots, CoW, merging base images with

snapshots, and maintaining shared root images with per-client writable overlays were well documented, publicly available, and widely deployed.

These practices were integral to well-established systems such as NetApp WAFL, VMware virtualization environments, Sun Microsystems diskless workstations, and cluster management solutions. Because the techniques described by the Asserted Claims of the '844 Patent were routinely implemented and broadly accepted by December 2004, the Asserted Claims of the '844 patent are both conventional and rendered obvious by the state of the art.

2. The Asserted Claims of the '844 Patent Fail at *Alice* Steps 1 and 2

The '844 Patent, titled "Root Image Caching and Indexing for Block-Level Distributed Application Management," claims methods for organizing and storing information. Dkt#1-1. Claim 7—the only independent claim of the Asserted Claims for the '844 Patent—recites:

> 7. A method for **providing data to a plurality of compute nodes**, comprising:
>
> **storing blocks of a root image** of said compute nodes **on a first storage unit**;
>
> **storing leaf images** for respective compute nodes **on respective second storage units**, said leaf images including only additional data blocks not previously contained in said root image and changes made by respective compute nodes to the blocks of the root image, wherein said leaf images of respective compute nodes do not include blocks of said root image that are unchanged by respective compute nodes; and
>
> **caching blocks of said root image** that have been accessed by at least one of said compute nodes **in a cache memory**.

The specification admits that the claimed components and methods—clustered computing, root-leaf storage systems, and caching mechanisms—were conventional. For example:

- Clustered computing and distributed file systems were established practices (Dkt#1-1 at 1:31–45);
- Root-leaf storage systems were a common method for creating boot images "on the

fly" (*Id.* at 2:14–24); and

- Caching mechanisms, including least-recently-used (LRU) algorithms, were known (*Id.* at 6:38–7:17).

Dependent claim 11 simply adds a generic "merging" step to claim 7—joining the blocks of the root and leaf images to "create cohesive respective application environments." Nonetheless, at their core, claims 7 and 11 are directed to the abstract idea of organizing and storing data using a hierarchical structure of a shared base and individualized overlays. The '844 Patent relies entirely on general-purpose computing devices with generic storage media (*Id.* at 4:27–56), and unclaimed components, such as "Union Block Devices" (UBDs), described as low-level drivers with no disclosed novel functionality (*Id.* at 5:19–64). The Asserted Claims of the '844 Patent likewise employ broad functional language like "providing," "storing," and "caching," without specifying any improved algorithms, hardware, or software. Thus, the Asserted Claims of the '844 Patent merely recite an abstract idea for organizing and storing data without any inventive concept, thereby failing at both steps of the *Alice* test.

a. The Asserted Claims of the '844 Patent recite an abstract idea for organizing and storing data.

The Asserted Claims of the '844 Patent are directed to the abstract idea of organizing and storing information using a "root-leaf" structure. Although applied to a computer network, the method steps—storing root and leaf images and caching accessed blocks—mirror basic data collection and organization practices, analogous to longstanding human activities. The patent attempts to claim the concept of storing some information in a central repository for permanent storage and common use (a root image), storing other non-identical data locally for faster retrieval and personal use (leaf images), and keeping recently used items temporarily on hand (cached). Such abstract ideas for how to organize and store data on computer systems are one of "the 'basic

tools of scientific and technological work' that are free to all men and reserved exclusively to none." *Synopsys, Inc. v. Mentor Graphics Corp.*, 839 F.3d 1138, 1146 (Fed. Cir. 2016) (quoting *Alice*, 573 U.S. at 216). Courts have consistently held such practices abstract. *See Intellectual Ventures I LLC v. Erie Indem. Co.*, 850 F.3d 1315, 1327 (Fed. Cir. 2017) (organizing and accessing data held abstract); *Content Extraction & Transmission LLC v. Wells Fargo Bank, Nat'l Ass'n*, 776 F.3d 1343, 1347 (Fed. Cir. 2014) (data collection and storage held abstract); *Kaavo Inc. v. Amazon.com Inc.*, 323 F. Supp. 3d 630, 641 (D. Del. 2018) ("setting up and managing a cloud computing environment" held abstract); *Versata Software, Inc. v. NetBrain Techs., Inc.*, No. 13-676-LPS-CJB, 2015 WL 5768938, at *7 (D. Del. Sept. 30, 2015) ("representing information in a hierarchy amounts to an abstract idea.").

The specification confirms that the described components and techniques, including root-leaf storage systems and caching mechanisms, were well understood at the time of the invention. Dkt#1-1 at 1:31–45, 6:37–7:5. The Asserted Claims of the '844 Patent provide no specifics on storage structures/hardware, caching algorithms/techniques, or other technical improvements, relying instead on functional language that describes high-level results without specific implementation details. *See ChargePoint, Inc. v. SemaConnect, Inc.*, 920 F.3d 759, 769 (Fed. Cir. 2019) ("Even a specification full of technical details about a physical invention may nonetheless conclude with claims that claim nothing more than the broad law or abstract idea underlying the claims.").

Claim 7's scant details about what data gets saved where are inherently abstract. *See Content Extraction*, 776 F.3d at 1347 ("The concept of data collection, recognition, and storage is undisputedly well-known. Indeed, humans have always performed these functions."). "This type of activity, *i.e.*, organizing and accessing [data], includes longstanding conduct that existed well

before the advent of computers and the Internet." *Erie Indem.*, 850 F.3d at 1327. Claim 7 includes no purportedly novel technical details for cluster computing—just an abstract idea for how to organize and store data. In the end, the claim is little different from a librarian storing different types of books and papers in different sections of a library. Claim 7 therefore fails *Alice* Step 1 because it recites a conventional root-leaf storage system using a high-level abstract concept to organize and store data without any specific technological improvement recited in the claim. Claim 11 fares no better; it merely adds a generic "merging" step—joining the blocks of the root and leaf images to "create cohesive respective application environments"—without describing *how* that merging is performed or any technical rules guiding it. Claim 11 therefore provides no new or inventive mechanism for merging, just the result.

b. The Asserted Claims of the '844 Patent lack an inventive concept.

Under Step 2 of the *Alice* framework, a claim must include an "inventive concept" sufficient to transform an abstract idea into patent-eligible subject matter. *Alice*, 573 U.S. at 217. To qualify, the claim must add something "significantly more" than the abstract idea itself, such as an improvement to the functioning of a computer or a specific application of technology. *See Erie Indem.*, 850 F.3d at 1328. "If a claim's only 'inventive concept' is the application of an abstract idea using conventional and well-understood techniques, the claim has not been transformed into a patent-eligible application of an abstract idea." *BSG Tech LLC v. Buyseasons, Inc.*, 899 F.3d 1281, 1290–91 (Fed. Cir. 2018). "The written description is particularly useful in determining what is well-known or conventional." *Intellectual Ventures I LLC v. Symantec Corp.*, 838 F.3d 1307, 1317 (Fed. Cir. 2016). Moreover, "a claimed invention's use of the ineligible concept to which it is directed cannot supply the inventive concept that renders the invention 'significantly more' than that ineligible concept." *BSG Tech*, 899 F.3d at 1290.

The Asserted Claims of the '844 Patent likewise fail Step 2, in part, because "these claims use generic computers to perform generic computer functions." *Symantec*, 838 F.3d at 1315. The specification concedes that the claimed methods are standard, including storing root images on a first storage unit, leaf images on second storage units, and caching accessed blocks. Dkt#1-1 at 2:14–21, 6:38–67. Thus, even if the specification discloses ideas that could be applied to useful ends, claims 7 and 11 "contain[] no restriction on how the result is accomplished." *Symantec*, at 1316.

The claimed steps are described at a high level of generality, with no new algorithms, configurations, or technological improvements. Generic terms like "providing," "storing," and "caching" describe desired outcomes without specifying any "inventive concept" for how these tasks are performed. *See BSG Tech*, 899 F.3d at 1290-91 ("If a claim's only 'inventive concept' is the application of an abstract idea using conventional and well-understood techniques, the claim has not been transformed into a patent-eligible application of an abstract idea"); *Elec. Power Grp., LLC v. Alstom S.A.*, 830 F.3d 1350, 1356 (Fed. Cir. 2016) (drawing the "distinction between ends sought and particular means of achieving them, between desired results (functions) and particular ways of achieving (performing) them").

Breaking down each element of the Asserted Claims of the '844 Patent reveals how each element is conventional:

Claim 7:

- **"storing blocks of a root image of said compute nodes on a first storage unit"**

  This is conventional. Shared read-only root images were routinely stored on central storage devices, as used in Sun's diskless workstations and LTSP environments. (*See* Hitz et al.; NetApp WAFL).

- **"storing leaf images for respective compute nodes on respective second storage units…"**

  This element is likewise conventional. Systems such as VMware and Episode stored deltas or overlays for individual machines using CoW. These deltas excluded unchanged base data, which was referenced from the shared root image.

- **"caching blocks of said root image that have been accessed…"**

  This element is likewise conventional. Caching mechanisms, including block-level caching and LRU algorithms, were widely used by 2004 in both operating systems and storage appliances (*see* NetApp; LTSP).

Claim 11:

- **"merging said blocks of the root image and said leaf images…"**

  The element recited by Claim 11 is similarly conventional and routine. The merging recited by Claim 11 had been widely implemented in systems such as VMware's and NetApp's platforms.

Courts have invalidated analogous claims. In *Symantec Corp.*, 838 F.3d 1307, the Federal Circuit invalidated claims for managing email messages using conventional computer components. Similarly, the court in *Content Extraction*, 776 F.3d 1343, held claims for data recognition and storage invalid because they relied on routine computing methods. The Asserted Claims of the '844 Patent fall into the same category, as they merely recite routine computing techniques and hardware.

The Asserted Claims of the '844 Patent offer no inventive step, combining conventional techniques predictably to achieve abstract goals. They rely on generic computing elements and

routine data management practices, failing *Alice* Step 2. Thus, The Asserted Claims of the '844 Patent are ineligible under §101.

B. U.S. Patent No. 8,407,722

1. State of the Art Relevant to the '722 Patent

At the time of the purported invention, claimed to have a priority date of December 18, 2000, the field of asynchronous messaging, client registration, and dynamic content routing was well established. Numerous standards, protocols, and technologies had already addressed the challenges of efficiently distributing data updates in networked environments, particularly where client devices needed to receive specific information based on prior registration.

The alleged innovation in the '722 Patent centers around organizing and transmitting data through a method that involves client registration and message routing within a dynamic network architecture. However, as demonstrated below, these concepts were not novel or unconventional as of December 2000.

a. Networking Standards and Protocols (Pre-2000)

One of the foundational concepts purportedly improved by the '722 Patent is asynchronous messaging, where data updates are sent without requiring an immediate response from the client. Asynchronous data routing was not a novel concept at the time of the alleged invention; it had been widely implemented in established systems and protocols well before the priority date.

One of the earliest implementations of dynamic data transmission is IP Multicast, standardized as RFC 1112 in 1989. IP Multicast allows sending a single data transmission to multiple recipients who have joined a specific multicast group. This approach significantly reduced network traffic by ensuring that only registered devices received the transmitted data. The ability to direct data only to interested recipients mirrors the concept of client registration for data updates, as claimed in the '722 Patent.

Similarly, CORBA Event Services, introduced in the late 1990s, provided asynchronous communication between distributed objects. These services allowed clients to register for specific events or updates, forming a fundamental part of distributed application architectures. CORBA's event-driven model effectively addressed the need for targeted data distribution and client registration in networked systems, predating the patent's purported improvements.

Another significant standard, the Java Message Service (JMS), was introduced in 1998. JMS implemented both the publish-subscribe model and point-to-point messaging, allowing clients to subscribe to specific data topics and receive only the relevant updates. This capability directly addressed the problem of selective message delivery, which the '722 Patent later claimed to improve.

Likewise, TIBCO Rendezvous, available by 1998, utilized content-based addressing and routing to manage the efficient delivery of data in real-time applications. This system also incorporated message queuing and subscription mechanisms, showing that targeted data dissemination was a well-understood challenge long before the alleged invention.

b. Message Queuing and Client Registration Systems

During the 1990s, message queuing systems were already employing techniques for client registration and selective message distribution. IBM MQ Series, introduced in the early 1990s, provided robust message queuing with support for both publish-subscribe models and point-to-point communication. These systems enabled client applications to register for specific message types, thereby receiving only the updates of interest. This core functionality—registering for updates and receiving targeted messages—parallels the method claimed in the '722 Patent.

In a similar vein, Gryphon, a content-based routing system introduced in 1998, supported dynamic data routing and client subscriptions. It allowed clients to receive updates based on

content specifications, a method fundamentally similar to the dynamic message routing described in the '722 Patent.

Another early messaging protocol, Session Initiation Protocol (SIP), which was standardized in 1999, offered a subscription and notification mechanism for presence updates. Through SUBSCRIBE and NOTIFY functions, SIP enabled clients to receive real-time notifications based on their expressed interests, thus efficiently handling client-specific data routing.

c. Relevant Prior Art Patents (Pre-2000)

Several patents pre-dating the '722 Patent also addressed dynamic content routing and client registration.

- **US Patent 5,754,939 (Herz et al.)** (issued in **1998**) discloses methods for routing messages to recipients based on profile information. This technique inherently involves selectively delivering messages to clients who registered to receive them, thus predating the '722 Patent's claimed method of client-specific message updates.
- **US Patent 5,845,324 (White et al.)** (issued in **1998**) similarly addresses message routing in networked communication systems, emphasizing methods where client devices receive only the relevant data based on their registrations.
- **US Patent 5,919,247 (Van Hoff et al.)** (issued in **1999**) discusses distributed content update systems that involve client-server architectures to update registered clients. This system reflects the idea of routing updates only to those clients that have expressed a need for specific data, which the '722 Patent claims as an improvement.

d. Real-Time Data and Event Notification Systems

By the late 1990s, event notification systems were leveraging client subscriptions to deliver updates efficiently. For example, Event Notification Services from 1999 utilized client-specific subscriptions to notify registered users of changes or updates, mirroring the client registration and update routing claimed in the '722 Patent.

Another significant example from 1996 is the Dissemination-Based Information Systems research, which focused on efficiently delivering data to registered users. These systems implemented techniques for selectively transmitting updates, thereby optimizing network resources, much like the patent's method of reducing message traffic.

e. Industry Implementations (Pre-2000)

In practice, industries such as finance had long implemented real-time data systems that updated registered clients with new information, such as stock price changes. Middleware systems like IBM MQ and CORBA were commonly used in enterprise environments to route messages efficiently, leveraging client subscriptions and content-based routing well before the priority date of the '722 Patent.

f. Conclusion

The evidence shows that by December 2000, the concepts underlying the '722 Patent—specifically client registration, dynamic content routing, and efficient message dissemination—were well-established in the field. The alleged invention does not introduce a novel technological improvement but rather combines existing methods and standards in a predictable way. The claimed methods represent an incremental combination of known techniques rather than a groundbreaking departure from the established state of the art.

2. The Asserted Claims of the '722 Patent are Ineligible Under 35 U.S.C. § 101

The Asserted Claims (claims 14, 16, 17, and 18) of the '722 Patent recite the abstract idea of receiving information, identifying relevant content, registering interest, and disseminating updates to interested parties. This is a long-standing and fundamental practice in organizing and distributing information—not a technological improvement or specific technical solution. Because the claims implement an abstract idea using only generic networking and data-handling functions without reciting any technological improvement or unconventional hardware, the claims fail both steps of the *Alice* framework.

a. The Asserted Claims of the '722 Patent are Directed to an Abstract Idea

The Asserted Claims of the '722 Patent are directed to the abstract idea of receiving information, identifying relevant content, registering interest, and disseminating updates to interested parties. Claim 14—the only representative independent claim—describes a method for enabling dynamic updates to "live objects" displayed on a client device through a structured communication framework involving a source device, a routing network, and intermediary nodes. An input source sends a data representation to a client device via a routing network; this data includes at least one live object—an identifiable element capable of receiving updates. The client device recognizes the live object, determines its identifier, and registers with the routing network to receive updates about it. This registration includes connection information, which is stored at a node in the routing network. When the input source sends an update message for the live object, the message is routed through a gateway that determines its category and selects the appropriate node type to handle it. The node then identifies the registered client and forwards the message. Upon receipt, the client updates the relevant property of the live object.

Claim 16 adds to claim 14 that the client stores metadata related to the live object, and that the update message modifies this metadata. Claim 17 adds to claim 14 that the update message includes a message type, which the client uses to determine how to apply the update. Claim 18 expands on claim 14 by allowing the client to register for multiple live objects and receive updates for each.

What the Asserted Claims of the '722 Patent effectively describe is a rules-based information forwarding mechanism, based on identifiers and interest registration—a framework that has existed in mailrooms, subscription services, and newsrooms long before the advent of digital networks. Its implementation using computers and a "routing network" merely automates this longstanding human practice.

Stripped of jargon, the Asserted Claims of the '722 Patent describe a familiar, abstract pattern: sending information, having a recipient identify what is relevant, and distributing future updates to those who have expressed interest. This pattern is not new or technological. It is, for example, identical in substance to how a traditional newsroom operates. A journalist (the "input source") provides a summary of several developing stories (the "data representation") to a central bulletin board in the newsroom. Each story on the board is a "live object" in the sense that it may be updated. Editors (the "client devices") review the bulletin board and mark their interest in particular stories. When updates come in from the field, a receptionist (the "gateway device") categorizes the update and determines which editor (the "node") should receive it. The update is then delivered to the relevant editor, who revises the draft story accordingly.

This real-world newsroom example illustrates that the Asserted Claims of the '722 Patent merely recite an abstract method of organizing and delivering information—a practice that has been implemented manually for decades. The fact that the '722 Patent describes this process in the

context of a computing environment does not render it any less abstract. Claims directed to abstract frameworks for information delivery—like updating content in response to user interaction or preference—have repeatedly been held patent-ineligible. *See, e.g.*, *Broadband iTV, Inc. v. Amazon.com, Inc.*, No. 6:20-CV-00921-ADA, 2022 WL 4703425, at *11 (W.D. Tex. Sept. 30, 2022) (J. Albright), *aff'd,* 113 F.4th 1359 (Fed. Cir. 2024); *see also Broadband iTV, Inc. v. Oceanic Time Warner Cable, LLC*, 135 F. Supp. 3d 1175, 1178 (D. Haw. 2015) (claims directed to abstract idea of "using the same hierarchical ordering based on metadata to facilitate the display and locating of video content" ineligible under § 101). Because the Asserted Claims of the '722 Patent focus on the result of an implementing an abstract idea—delivering updated content to interested recipients—rather than a specific technological means of achieving that result, they fail at *Alice* step one.

The specification attempts to present these steps as an improvement over prior art by claiming to enhance network efficiency through dynamic client registration and message routing. It asserts that the use of node specialization and dynamic mapping reduces unnecessary message traffic. The routing network purportedly eliminates the need for the input source to track which clients are displaying the live objects by maintaining a registry of registered clients. The system is described as being useful for applications where content on client devices must be updated dynamically based on changes in data, such as stock prices or sports scores. However, the specification itself reveals that the claimed methods are implemented using conventional networking techniques. The routing network, which consists of nodes and gateways, is built using generic computing components, including single-processor computer systems running common operating systems like Linux. The specification indicates that the system utilizes existing messaging protocols, such as HTTP, and known client-server architectures. Furthermore, the

patent acknowledges that the basic concepts of client registration and update handling were already present in existing methods that included client-driven approaches, applets, and server-driven updates using TCP/IP connections.

b. The Asserted Claims of the '722 Patent Do Not Recite an Inventive Concept

Step 2 of *Alice* cannot save the Asserted Claims of the '722 Patent from ineligibility under 35 U.S.C. § 101. The Asserted Claims of the '722 Patent do no more than implement the abstract idea of selectively distributing content updates to interested recipients using generic computing components performing routine, conventional functions. The purported improvement—routing update messages based on client registration—does not introduce any new computing technology or method of enhancing network efficiency. Instead, it merely applies well-known client-server communication techniques to the problem of distributing data updates. The specification itself discloses that the system utilizes conventional networking components and known data transmission protocols without any significant modification.

Claim 14 recites basic components: an "input source," a "client device," a "routing network," "nodes," and "gateways." But none of these are claimed in a specific or inventive configuration. The specification describes the input source as a web server or dynamic content provider, the client device as a browser-enabled device such as a PC or mobile phone, and the routing network as an internet-based message delivery structure. The "gateway" and "node" elements, while given roles in the message flow, are described in general terms that mirror conventional content delivery mechanisms—such as proxies, relays, load balancers, or intermediate servers.

There is no technical detail in the claim about how messages are classified, how registration is implemented, or how update routing is performed. The claim merely recites the expected result:

that a client receives an update for a live object it previously registered for. Courts have routinely held that such claims, which rely on routine network behavior and leave implementation to the practitioner, do not supply an inventive concept. *See, e.g.*, *Two-Way Media Ltd. v. Comcast Cable Commc'ns, LLC*, 874 F.3d 1329, 1339 (Fed. Cir. 2017) ("The claim uses a conventional ordering of steps—first processing the data, then routing it, controlling it, and monitoring its reception—with conventional technology to achieve its desired result.").

The specification itself confirms that the invention is designed to operate on existing infrastructure and leverages known technologies. Client devices are said to use standard web browsers with existing support for Java, JavaScript, or plug-ins. Update messages are transmitted over HTTP and are parsed using client-side scripts. The routing network is implemented using conventional clusters, nodes, and gateways, and messages are routed using category/type mappings or registry lookups—well-understood design patterns in networked systems. The so-called "live objects" are simply HTML elements or other page components that can be identified by an ID and updated via DOM manipulation or script execution, which common techniques in web development since at least the late 1990s. The use of an "activation module" to register interest in updates is just another name for client-side code that subscribes to a push service—an architecture long practiced in content syndication and notification systems. Nothing in Claim 14 requires, or even enables, a novel technical implementation of these mechanisms.

None of the limitations in claims 16, 17, or 18 inject any inventive concept into the '722 Patent:

- **Claim 16**'s mention of metadata storage and update is a routine feature of client-side state tracking. Web applications and software clients have long-maintained associated metadata (e.g., timestamps, tags, formatting) alongside core content.

- **Claim 17** introduces a message type field used to determine how to process an update. But conditional logic based on message types is a staple of basic programming and messaging protocols. This is no more inventive than a "switch" statement tied to a message header.
- **Claim 18** expands registration to multiple live objects, which is a straightforward extension of the core idea. There is no novel mechanism for batch registration, grouping, or prioritization; the claim merely allows more than one subscription, a feature inherent in any scalable publish-subscribe system.

Thus, the dependent claims likewise apply well-known programming constructs and user-experience features without claiming how any of them are implemented in a novel or unconventional manner.

Furthermore, the combination of client registration and dynamic message routing was not an inventive concept at the time of the alleged invention. Technologies such as IP Multicast, Java Message Service (JMS), and CORBA Event Services were already implementing message routing based on client registration well before the priority date. The combination of these known elements does not produce unexpected results or provide a technological improvement. Here, the claimed method merely takes existing client registration and message routing techniques and applies them in a straightforward, predictable way.

The Asserted Claims of the '722 Patent therefore do not improve any specific computer functionality or solve any technical problem in a new way—they merely implement a conventional information delivery model using abstract functional language and general-purpose computing components. As such, the Asserted Claims of the '722 Patent fail *Alice* step two and are invalid under 35 U.S.C. § 101.

c. Conclusion

The Asserted Claims of the '722 Patent are invalid under 35 U.S.C. § 101 because they fail both prongs of the *Alice* framework. At step one, these claims are directed to the abstract idea of distributing updates about selected content to interested recipients—an organizing principle that has long been practiced by humans and implemented in conventional information systems. These claims describe this idea at a high level of generality, using functional language untethered to any specific improvement in technology or network design. At step two, the claims do not recite any inventive concept sufficient to transform this abstract idea into a patent-eligible application. The components are generic, the operations are conventional, and the implementation relies entirely on known computing techniques. These claims simply automate a familiar content delivery paradigm using routine architecture, offering no technical advancement over the prior art.

C. U.S. Patent No. 7,949,785

The Asserted Claims (claims 30, 35, and 37) of the '785 Patent are invalid under 35 U.S.C. § 101 because they are directed to an abstract idea and lack an inventive concept. The Asserted Claims of the '785 Patent recite a virtual network manager that facilitates secure communications within a virtual community—an abstract concept under *Alice* step 1. Further, the claims use only generic and conventional computing components and processes without introducing any specific technological improvements, and as such, the claims likewise fail to recite an inventive concept necessary to survive *Alice* step 2. Consequently, the Asserted Claims of the '785 Patent fail under *Alice* and are ineligible for patent protection under 35 U.S.C. § 101.

1. State of the Art in March 2003

By March 2003, secure communication and virtual networking were well-established fields, and numerous technologies were routinely used to address the challenges associated with secure data transmission over public networks. The '785 Patent, which claims a virtual network

manager for secure communication in a virtual community, builds on concepts and techniques that were widely known and used by March 2003. To fully understand the context of the invention, it is essential to examine the prevailing technologies and methods that a person of ordinary skill in the art (POSA) would have recognized as routine and conventional.

a. Secure Communication and Virtual Private Networks (VPNs)

One of the most common methods for secure communication over public networks in 2003 was the use of Virtual Private Networks (VPNs). VPNs were routinely deployed to create secure tunnels that allowed private network traffic to be transmitted over the public Internet. These tunnels were established using encryption protocols such as IPsec (Internet Protocol Security), which provided authentication, encryption, and data integrity for IP communications. IPsec was commonly used to create secure point-to-point connections, effectively encapsulating data within an encrypted packet that could be transmitted securely over the Internet.

Other VPN protocols included the Point-to-Point Tunneling Protocol (PPTP), which facilitated the creation of VPNs by encapsulating data within IP packets, and Secure Sockets Layer (SSL)/Transport Layer Security (TLS), which were widely used for web-based secure communication. VPN technology was an integral part of network security practices, with major companies like Cisco offering VPN client software and integrated VPN solutions as part of their networking equipment. The concept of encapsulating private network traffic within a secure channel was routine and had been implemented in numerous commercial and enterprise settings.

Further, the use of VPNs addressed the growing need for secure remote access, allowing employees to connect to a corporate network from offsite locations. By encrypting data and using secure tunneling, VPNs enabled organizations to maintain private communications even when using the public Internet. This approach was considered standard practice for enterprises looking to protect data from unauthorized access.

b. Network Address Translation (NAT) and Address Depletion

During this period, the Internet was facing a well-documented problem: the depletion of available IPv4 addresses. The rapid expansion of Internet-connected devices had exhausted the finite pool of unique public IP addresses. Network Address Translation (NAT) emerged as a practical and widely implemented solution to this problem. NAT allowed multiple devices within a private network to share a single public IP address when accessing the Internet. This was achieved by translating the private IP addresses to the public IP address of a NAT device (such as a router) and tracking the active connections using port numbers.

NAT was particularly beneficial for enterprises with large internal networks, as it significantly reduced the number of public IP addresses needed. NAT devices maintained tables mapping internal (private) addresses to external (public) ones, enabling inbound and outbound traffic to be correctly routed. This technique was so fundamental that most consumer and enterprise routers incorporated NAT functionality by default.

The concept of NAT traversal was also well understood by 2003. Because NAT inherently interfered with direct peer-to-peer communication, solutions like STUN (Session Traversal Utilities for NAT) and UPnP (Universal Plug and Play) were developed to allow devices behind NAT to establish direct connections. These techniques were vital for applications such as VoIP (Voice over IP) and real-time communications, which required consistent end-to-end connectivity despite NAT barriers.

c. Dynamic DNS (DDNS) and Mobility Solutions

Another challenge related to IP address management was the dynamic assignment of IP addresses to devices, especially in mobile or residential contexts. Dynamic DNS (DDNS) provided a solution by automatically updating DNS records when a device's IP address changed. This was crucial for maintaining connectivity to devices that did not have a fixed IP address. DDNS allowed

users to maintain a consistent hostname that would resolve to the device's current IP address, enabling continuous access regardless of network changes.

Dynamic DNS was standardized by RFC 2136, published in 1997, and was commonly used in conjunction with NAT to maintain address stability for dynamically addressed hosts. The emergence of mobile IP protocols, including those outlined in RFC 3344, further addressed the mobility issue by allowing devices to maintain the same IP address while moving between different networks.

d. Peer-to-Peer (P2P) Networking and NAT Traversal

In parallel, peer-to-peer (P2P) networking was becoming increasingly prevalent, with applications such as file sharing (e.g., Gnutella, Napster) and distributed computing. These systems often faced challenges with NAT traversal, as direct device-to-device connections were impeded by NAT devices. To overcome this, P2P applications used techniques like hole punching, where both peers behind NAT would simultaneously open a connection to a third-party server to facilitate direct communication.

These traversal techniques, while not perfect, were necessary for maintaining direct connectivity in decentralized networks. The ability to establish direct communication between peers despite NAT interference was a key feature for P2P systems, and methods to achieve this were well known and routinely employed.

e. Virtual LANs (VLANs) and Virtual IP Addresses

Another well-established concept was the use of Virtual LANs (VLANs) to logically segment a physical network into smaller, manageable sub-networks. VLANs allowed devices to communicate as if they were on the same local network, even if physically separated. This logical grouping of devices was fundamental for both security and network management.

Similarly, the concept of virtual IP addresses was commonly used to represent services or devices within a network. Virtual IPs were not tied to a single physical interface, allowing systems to maintain service availability even if the underlying hardware changed. Load balancers and failover systems routinely used virtual IPs to distribute traffic or provide redundancy.

f. Conclusion

By March 2003, the landscape of network communication and secure virtual networking was marked by mature, well-understood technologies. VPNs provided secure data transmission over public networks, NAT allowed efficient IP address management and connectivity, and dynamic DNS addressed mobility and dynamic IP assignment. P2P networking dealt with NAT traversal challenges using established methods, while VLANs and virtual IPs were commonplace for managing complex network topologies.

The '785 Patent, therefore, builds on these conventional techniques without introducing any novel technological solution. Its reliance on well-known methods for secure communication, NAT traversal, and virtual addressing does not advance the state of the art. Rather, it merely applies conventional networking practices to the problem of secure virtual community formation, which had already been addressed through routine methods available at the time.

2. The Asserted Claims of the '785 Patent Recite an Abstract Idea

The Asserted Claims of the '785 Patent are directed to the abstract idea of membership management and address look-up in a communications network—mere administrative tasks performed every day by receptionists, security personnel, and bouncers. Claim 30, the only independent Asserted Claim of the '785 Patent, recites:

[30.0] A virtual network manager, comprising:

[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;

[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:

[30.3] receive a registration request from an agent associated with a device;

[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and

[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.

Stripped of jargon, the claimed elements are basic administrative functions involved in setting up and maintaining a private communication network—operations that map directly onto longstanding abstract concepts such as maintaining a directory of users, assigning identifiers, and responding to queries with directing information. Claim 35 builds on claim 30 by adding a "join module" that receives join and leave requests—analogous to a sign-in/sign-out log for network members. Claim 37 further adds that the join module maintains a mapping between virtual addresses and devices—a conventional association table akin to a user directory or ARP (address resolution protocol) table.

The Asserted Claims of the '785 Patent all reflect the abstract idea of membership management and address look-up in a communications network—akin to a corporate office

receptionist managing an internal directory. The receptionist receives visitors (devices) requesting access (registration), checks whether they're authorized to enter (authentication), assigns them a visitor badge with an internal extension number (virtual address), and, when asked, tells other employees where to find the visitor (DNS resolution and routing). The receptionist also tracks when the visitor checks in and out (join and leave requests) and keeps a temporary log mapping badge numbers to visitors (address-device association). These are routine administrative functions that merely facilitate communication within a defined space—just as the Asserted Claims do within a virtual network. There is no innovation in how the receptionist performs these tasks, and these tasks have been performed by humans for centuries. Likewise, the Asserted Claims do not improve any underlying networking technology or propose any inventive technical implementation. The Asserted Claims of the '785 Patent are simply automating a familiar organizational role using conventional computing tools, rendering them abstract under *Alice* step one.

Tellingly, the method of originally filed claim 64 (which issued as claim 48) did not include any requirement that the steps of the claim be performed by a computer. On February 4, 2009 (5 years before the *Alice* decision), the Examiner rejected claim 64 under 35 U.S.C. § 101, applying the *Bilski* "machine-or-transformation" test for patentable subject matter that existed in 2009. In other words, without the recitation of a "machine" (*i.e.*, a computer), the claims were not patentable under § 101. In response to the Section 101 rejection, the Applicants amended claim 64 to be "computer-implemented," as none of the originally-filed claims required the use of a computer. Although an amendment requiring that the method be performed by a computer was sufficient in 2009-2010 to overcome the Examiner's Section 101 rejections, this is not the case today. Under current Supreme Court and Federal Circuit law, the *Alice* steps apply.

The '785 Patent Applicants' computer-implemented amendment to independent claim 64 crystallizes two points. First, the amendment demonstrates that the Applicants understood that the method described in the '785 Patent and claimed in the originally-filed claim 64 could be practiced without the use of a computer, *i.e.*, the method could be performed in the human mind, or by a human using a pen and paper. *See CyberSource Corp. v. Retail Decisions, Inc.*, 654 F.3d 1366, 1372–73 (Fed. Cir. 2011) (holding that a claim whose "steps can be performed in the human mind, or by a human using a pen and paper" is directed to an "unpatentable mental process[]"); *Elec. Power Grp.*, 830 F.3d at 1355 ("[C]laims focusing on organizing, characterizing, and/or sorting information are patent-ineligible because they are not significantly differentiable from ordinary mental processes."). Second, this amendment demonstrates that the focus of the claims of the '785 Patent is not on any specific asserted improvement in computer capabilities. *In re TLI Commc'ns L.L.C. Pat. Litig.*, 823 F.3d at 612 ("[T]he claims were not directed to specific improvement to computer functionality. Rather, they are directed to the use of conventional or generic technology in a nascent but well-known environment, without any claim that the invention reflects an inventive solution to any problem presented by combining the two.").

Under *Alice*, a claim that is unpatentable without the recitation of a computer does not become patentable by reciting a computer. That the claims of the '785 Patent merely implement the non-statutory subject matter on a generic computing device is plain from the fact that the originally-filed claims did not require that the claimed systems and methods include or be performed by a computer, respectively. Thus, the language of the Asserted Claims, their prosecution history, and the change in the law demonstrate that the Asserted Claims of the '785 Patent are directed to an abstract administrative idea applied to a communications network.

3. The Asserted '785 Patent Claims Lack an Inventive Concept

The Asserted Claims of the '785 Patent fail at *Alice* step 2 because they do not recite anything beyond generic computer functions performed using conventional components. The specification confirms that the system is built using well-known networking elements (*e.g.*, processors, memory, interfaces, DNS servers, NAT traversal techniques, virtual IP addressing) without any specialized circuitry, protocol, or configuration.

The '785 Patent itself acknowledges that the components and methods it employs were conventional at the time of invention. For example, the specification discusses the use of Network Address Translation (NAT) and virtual addresses to facilitate secure communication, but these techniques were already well-known in the field of networking. The patent does not propose any novel way of utilizing NAT or managing virtual addresses beyond their ordinary and conventional uses.

Claim 30, for instance, merely describes using a DNS server to resolve device addresses within a virtual network, which is a standard function of DNS servers generally. The idea of assigning virtual addresses to devices and resolving these addresses through DNS lookups is not unique or inventive. Instead, the patent recites high-level results—such as secure communication—without specifying how these results are technologically achieved. There is no disclosed protocol, no new addressing scheme, no algorithm for conflict resolution, and no structural innovation in how devices are registered or managed. The system instead relies on existing IP, UDP, NAT, and IPsec technologies. The claims are results-oriented: "register the device," "assign an address," "respond to a DNS query"—without any accompanying inventive technical solution.

The claim elements—processor, memory, network interface, modules—are standard hardware or logical software components performing their expected functions. There is no claim

that the invention requires new hardware, reconfigured components, or even custom protocols. The Asserted Claims of the '785 Patent are therefore equivalent to the claims in *BSG Tech LLC v. BuySeasons, Inc.*, 899 F.3d 1281, 1291 (Fed. Cir. 2018), where the claims at issue merely limited an abstract idea to a particular technological environment and were held ineligible. Similarly, in *ChargePoint, Inc. v. SemaConnect, Inc.*, the court invalidated claims that combined generic networking capabilities with existing concepts without adding any inventive technology. 920 F.3d 759, 774 (Fed. Cir. 2019) ("But network control is the abstract idea itself, and 'a claimed invention's use of the ineligible concept to which it is directed cannot supply the inventive concept that renders the invention 'significantly more' than that ineligible concept.'") (quoting *BSG Tech*, 899 F.3d at 1290). The Asserted Claims of the '785 Patent follow the same pattern by applying well-understood networking techniques without advancing the state of the art.

4. Conclusion

The Asserted Claims of the '785 Patent are invalid under 35 U.S.C. § 101. They are directed to abstract administrative operations for managing virtual networks and fail to recite any inventive concept in either their individual components or overall arrangement. They do not improve any underlying technology, nor do they disclose a novel technical mechanism for achieving the stated goals. As such, they fail both steps of the *Alice* test and are ineligible for patent protection.

5. Admissions in the Specification

a. Network Address Translation (NAT) as Routine and Conventional:

The specification of the '785 Patent explicitly acknowledges that NAT (Network Address Translation) was a well-known and conventional technique at the time: "One popular solution to the depleting address problem is Network Address Translation (NAT). This concept includes predefining a number of network addresses to be private addresses and public addresses. Public addresses are unique addresses that should only be used by one entity having access to the Internet.

That is, no two entities on the Internet should have the same public address. Private addresses are not unique and are typically used for entities not having direct access to the Internet. Private addresses are used in private networks. Private addresses can be used by more than one organization or network. NAT assumes that all of the machines on a network will not need to access the Internet at all times. Therefore, there is no need for each machine to have a public address." 785 Patent at 1:56-62.

b. Virtual Private Network (VPN) as Well-Known:

The patent also identifies VPN technology as an existing and established solution: "One solution currently in use that provides for local and remote specified entities to communicate and collaborate using the Internet is the Virtual Private Network (VPN). The VPN uses additional network software layers to increase security between users in the public realm and those in private realms. For example, some VPNs encrypt packets using IPsec (or other protocols). The encrypted packets are then encapsulated within a standard packet and transmitted across the Internet to the destination. At the destination, the encrypted packet is decrypted." 785 Patent at 4:58-67.

c. Dynamic DNS as Known and Conventional:

Dynamic DNS, which is also mentioned in the patent, is described as an existing technology that predates the invention: "One method of overcoming the mobility problem includes the use of Dynamic DNS. More information about Dynamic DNS can be found in RFC2136 incorporated herein by reference." 785 Patent at 5:11-14.

d. Hardware and System Components as Conventional:

The specification explicitly states that the hardware components required for the invention were standard and understood: "A hardware architecture for the machines, server or other devices used to implement the present invention should be well understood to one of average skill in the art. Suitable hardware includes one or more processors, a memory, a mass storage device, a

portable storage device, a first network interface, a second network interface and I/O devices, in communication with each other." 785 Patent at 9:42-48.

e. IP Addressing and Network Protocols as Routine:

The specification discusses the standard nature of IP addressing and network protocols: "Most machines on the Internet use the TCP/IP (Transmission Control Protocol/Internet Protocol) reference model to send data to other machines on the Internet." 785 Patent at 3:1-3.

D. U.S. Patent No. 8,027,326

Representative Claim 1 recites:

> [1.0] A method for increasing data rates and data throughput in a network, the method comprising:
>
> [1.1] selecting at least a first channel and a second channel, wherein the first channel and the second channel are adjacent without any other channels there between, wherein the first channel and the second channel each have a plurality of data subcarriers, wherein the data subcarriers of the first channel and the data subcarriers of the second channel are separated by a frequency gap corresponding to one or more guard bands between the first and second channels;
>
> [1.2] partially filling the frequency gap between the first channel and the second channel by adding one or more data subcarriers into the frequency gap such that the one or more guard bands are at least partially filled with at least some of the one or more data subcarriers using full spectral synthesis capability of a fast fourier transform or an inverse fast fourier transform;
>
> [1.3] **combining the first channel and the second channel** using channel bonding with orthogonal frequency division multiplexing (OFDM); and
>
> [1.4] transmitting data subcarriers occupying the first channel, the second channel, and the frequency gap in parallel to a receiver.

1. State of the Art Relevant to the '326 Patent

Southwest sets forth a summary of its current understanding of the state of the art for the general subject matter of the '326 Patent. The '326 Patent generally relates to a "method and

system for high data rate multi-channel wireless communications" which reuses "existing single channel radio designs" to "achieve very high data rate expanded bandwidth (wide band) WLAN operations." Broadly, the '326 Patent concerns the development of more efficient Wi-Fi, with I.V.'s Complaint focusing on "hardware and/or software from these providers that support Wi-Fi, including IEEE 802.11n and 802.11ac protocols." Dkt#1 ¶ 93. These Wi-Fi protocols were widely used as of the date of filing:

"802.11ac was finalised in 2013 and you will find it in every major smartphone, laptop and desktop computer and smart television. It succeeds the equally-badly named '802.11n' which has been around since 2007 and brings some major benefits." Kelly, Gordon, "802.11ac vs 802.11n Wi-Fi: What's the Difference?" Forbes, December 30, 2014 *available at* https://www.forbes.com/sites/gordonkelly/2014/12/30/802-11ac-vs-802-11n-wifi-whats-the-difference/.

802.11 was already a popular standard. "IEEE 802.11-based wireless technology is well known by the common name 'Wi-Fi' (Wireless Fidelity) and provides wireless connectivity to the majority of today's digital devices such as mobile phones, laptops, smart TVs, etc. The most commonly used wireless technologies in smart manufacturing, especially discrete manufacturing industries are also IEEE 802.11 based." Liu, Xuejing, et al., "Exploring the Regulatory Mechanisms of Long Non-Coding RNAs in the Pathogenesis of Atherosclerosis," Pubmed Central, August 1, 2022 *available at* https://pmc.ncbi.nlm.nih.gov/articles/PMC9824593/.

Trends in the Wi-Fi industry included improving internet speeds, increasing bandwidth and reducing interference, particularly as the volume of users increased.

2. The Asserted Claims of the '326 Patent are Ineligible Under 35 U.S.C. § 101

The Asserted Claims (claims 1, 4, and 18) of the '326 Patent are directed to the abstract idea of combining adjacent communication channels to increase data throughput in a network. The Asserted Claims of the '326 Patent merely implement this concept using conventional networking techniques and generic computing components, offering no technological innovation or improvement. As such, they are ineligible under 35 U.S.C. § 101 because they are directed to an abstract idea and fail to recite an inventive concept, as required under the two-step framework set forth in *Alice Corp. v. CLS Bank Int'l*, 573 U.S. 208 (2014).

a. The Asserted Claims of the '326 Patent are Directed to an Abstract Idea

The Asserted Claims of the '326 Patent recite the abstract idea of combining channels to increase data throughput (claims 1 and 18) and transmitting data to multiple radios (claim 4), and fail Step 1 of *Alice*. *See Alice*, 573 U.S. at 216-17. These are long-standing and fundamental practices in sending and receiving information over a network—not a technological improvement or specific technical solution. There is nothing to the patent beyond the concept of 'combine two pipes together, including the space between them, to get a bigger pipe.'

Channel bonding and transmitting data to multiple radios is an abstract idea as claims directed to sending and receiving data are abstract in nature. *See, e.g., Affinity Labs of Texas, LLC. v. DIRECTTV, LLC*, 838 F.3d 1253, 1261 (Fed. Cir. 2016) (explaining that patents directed to "the conveyance and manipulation of information using wireless communication and computer technology" do not disclose eligible subject matter). Claims directed to formatting data are also abstract. *See Dropbox, Inc. v. Synchronoss Techs., Inc.*, 815 F. App'x 529, 537 (Fed. Cir. 2020) (finding claims ineligible that were directed, in part, to "formatting data").

b. The ‘326 Patent Claims Lack an Inventive Concept

Step 2 of *Alice* cannot save the Asserted Claims of the ‘326 Patent from ineligibility under 35 U.S.C. § 101. The Asserted Claims of the ‘326 Patent do no more than implement the abstract idea of increasing data throughput by using generic computing components performing routine, conventional functions. Namely, channel bonding (claims 1 and 18) and transmitting data to multiple radios (claim 4) lack an inventive concept because such claims amount only to generic components and fail to demonstrate unconventionality. *See Dropbox*, 815 F. App’x at 537 (finding that “data transfer connection[s] are generic computer-related concepts as they literally are just data transfer modalities” and are thus not inventive concepts).

c. Conclusion

The Asserted Claims of the ‘326 Patent are invalid under 35 U.S.C. § 101. They are directed to the abstract idea of transferring data over a network and fail to recite an inventive concept. Because the claims implement an abstract idea using only generic networking and data-handling functions without reciting any technological improvement or unconventional mechanisms, the claims fail both steps of the *Alice* framework.

E. U.S. Patent No. 7,324,469

The ‘469 Patent, titled “Satellite Distributed High Speed Internet Access,” claims methods for providing an internet hotspot to subscribers in a remote location. Asserted independent Claim 24 of the ‘469 Patent recites:

> 24. An Internet Hotspot comprising:
>
> a satellite dish **communicating with the Internet** via one or more data links with a satellite;
>
> at least one router operatively coupled to the satellite dish;

> a subscriber access unit operatively coupled between the satellite dish and the at least one router, the subscriber access unit **being capable of authenticating** a subscription account associated with a user prior to allowing the user access to the Internet; and
>
> a web-ready device operatively coupled to the at least one router, the web-read device **having a browser application** operating thereon for accessing the Internet;
>
> wherein the satellite dish, at least one router and the subscriber access unit are located in a remote location a experiencing a relatively high volume of transient traffic;
>
> wherein the user **may authenticate** the subscription account and access the Internet at the remote location by **establishing a data connection** between the web-ready device and the router.

(emphasis added).

1. State of the Art Relevant to the '469 Patent

As of September 2003—the earliest claimed priority date of the '469 Patent—wireless and wired Internet hotspots were ubiquitous, and their operation as a pay-for-use business was well known. For example, a 2003 book entitled "Going Wi-Fi" describes that hotpots were "all the rage":

> North America, Singapore, Korea and Japan have led this Wi-Fi trend, offering traveling laptop users temporary wireless Internet access in various types of public gathering spots including, for example, hotels and cafés. According to Garner Inc., the number of HotSpots will grow to more than 71,000 in 2003 from about 1200 in 2001.

Reynolds, Janice, "Going Wi-Fi A Practical Guide to Planning and Building an 802.11 Network," CMP Books, San Francisco (2003) ("Going Wi-Fi") at 4. These thousands of Internet hotspots included monthly subscription-based service as well as some that "allow users to purchase as little as 30 minutes worth of wireless access for the equivalent of a few dollars." *Id*. at 5. Well-established telecommunications providers like T-Mobile and Boingo were, at the time the Asserted Patent was filed, already offering hotspots "from Starbucks to hotels to airports." *Id.* ("and, of course, they plan to charge plenty for it."). Numerous publications describe the availability of hotspots at locations such as hotels, airports, coffee shops, and fast-food restaurants. *See* Ellison,

Carol, "Unwire Everywhere," PC MAGAZINE (Special Wireless Issue 2003) at 97-104 ("Hot spots are cropping up at a remarkable rate to meet the connectivity demands of users as they venture out from their homes and offices.").

The concept of subscription or payment authorization to access Internet hotspots was likewise well established at the time of the filing of the '469 Patent, and many companies offered "turnkey" hotspot "kits" which

> deliver the fundamental services layer by providing standards-based services such as secure authentication, automated configuration, IP routing, firewalls, caching, remote management, usage tracking, and bandwidth shaping.

Going Wi-Fi at 271.

With the advent of satellite Internet connectivity quickly came satellite Internet "hotspots,"—of course, for a fee. In one example, Airpath Wireless offered turnkey hotspot hardware, software, and connectivity services "for Hot Spot providers in need of broadband connectivity in remote or cost-prohibitive Hot Spot locations," such as "marinas, truck-stops, campgrounds, resorts, and hotels located in rural areas." June 25, 2003, Press Release available at https://web.archive.org/web/20030808000849/http://www.mainstreamdata.com:80/content/_news/news0.html. Airpath services included a "Hot Spot Billing Suite" for authentication and subscriber management. Satellite networks were especially suited to implementation of hotspots outside of urban settings, including military bases throughout the world. iDirect Government provided Internet cafes in US military tents and buildings around the world beginning in March 2000. *See* https://www.idirectgov.com/media/1jxdg0c3/idigt_cs_spawar-mwrnet_0215.pdf.

Although the '469 Patent does not mention aircraft or in-transit connectivity—the patent addresses only fixed locations such as "conference centers"—Southwest also addresses the state of the art pertaining to in-flight connectivity in view of the plaintiff's construction of the claims to

encompass in-flight connectivity systems. The Boeing Company announced its in-flight connectivity system, "Connexion," three years before the filing of the provisional application of the '469 patent, in April 2000. Building on its proprietary satellite transceiver antenna technology, Connexion installed a "flat, two-inch antenna and additional wiring [] to each top of aircraft" which "provides enhanced response to directional changes" and which adjusted "to keep in touch with the nearest satellite." Suwanateep at 7-8. Boeing's Connexion system was among the first to offer real-time airborne connectivity. *Id*. at 5.

> The cabin is equipped with onboard server, router and service-analysis equipment and is outfitted for the security enhancement capabilities pursuing for commercial airline customers. Seats are configured for connectivity. Since Internet traffic tends to happen in bursts, the cabin file-server not only allows multiple users to share access to a single physical connection, but also allows them to be on-line simultaneously.

Suwanateep at 9. Connexion by Boeing was first installed on private corporate aircraft, but later rolled out on major commercial airlines including United Airlines and Delta in early 2003. With commercial airline adoption came paid access for travelers using Connexion In-Flight Portal:

Find out about our In-Flight Portal



Before you access the Internet through the Connexion by Boeing service, you will have the option to select a variety of free services and content through our In-Flight Portal.

The In-Flight Portal's selection of top news and entertainment headlines will help keep you up to date, even at 39,000 feet. You can also use the In-Flight Portal to register for the service, view Frequently Asked Questions (FAQs), and contact customer care.

https://web.archive.org/web/20030201115424/http://www.connexionbyboeing.com/index.cfm.

Aircrafts were not the only in-transit systems for Internet access designed to meet growing demand in the early 2000s. In 2002, satellite provider Intelsat partnered with Maritime Telecommunications Network (MTN) to provide connectivity to maritime customers including Norwegian Cruise Lines. *At Sea But Never Out Of Touch*, CRUISE INDUSTRY QUARTERLY,

(Summer 2003) at 25. MTN provided complete hardware and software solutions to cruise ships, including internet cafes and billing options for passenger access.

2. The Asserted Claims of the '469 Patent recite the business practice of placing a paid hotspot in a "transient location"

Each of the Asserted Claims of the '469 Patent are invalid under 35 U.S.C. § 101 because the claims are directed to an abstract idea and fail to describe an inventive concept. *Alice Corp. v. CLS Bank Int'l*, 573 U.S. 208, 216-17 (2014). For example, all of the Asserted Claims of the '469 patent are generally directed to an abstract idea for Section 101 purposes and under Intellectual Venture's apparent constructions. The sole asserted independent claim, claim 24, concerns placing a known composition of commercial components in a "remote location" to establish a subscribers-only internet hotspot. The specification also makes clear that the invention of the Asserted Patent is the idea of placing paid hotspots where users would otherwise not have Internet access:

> The present invention provides rural "Hotspots" (such as Wi-Fi access, for example) to enable wireless and hardwired, satellite distributed Internet access for anyone with a PC or other web-ready device (wireless ready or cabled) and a valid credit card.

'469 Patent at [1:34-38]. The business idea of charging people money for internet access in a "transient location" like a restaurant or motel is just that: a business idea. Paying for services is a "fundamental economic practice long prevalent in our system of commerce" and, "as such, is a patent ineligible abstract idea." *W. Express Bancshares, LLC v. Green Dot Corp*., 816 F. App'x 485, 487 (Fed. Cir. 2020).

Applying this fundamental economic practice to tangible components such as commercially available network equipment does not alter the fact that the claims are directed to "an art or principle in the abstract, and not for any particular method or machinery." *ChargePoint, Inc. v. SemaConnect, Inc.*, 920 F.3d 759, 769 (Fed. Cir. 2019) (quoting *Wyeth v. Stone*, 30 F. Cas. 723 (C.C.D. Mass. 1840)); *ChargePoint*, 920 F.3d at 770 ("In short, the inventors here had the

good idea to add networking capabilities to existing charging stations to facilitate various business interactions. But that is where they stopped, and that is all they patented. We therefore hold that claim 1 is 'directed to' an abstract idea"). The patentee's implementation rests wholly on the existing knowledge of a person of skill in the art: "The Hotspot 10 comprises *all the equipment installed at the rural location that is necessary to provide Internet access via satellite*." '469 patent at 4:18-21 (emphasis added). "The Ethernet wireless access point 40 can employ *a security protocol of a type that is known to persons skilled in the art*, such as the 802.11b security protocol, or any subsequent or future versions of the 802.11 standard." '469 patent at 5:19-25 (emphasis added). The Federal Circuit has "repeatedly found the concept of controlling access to resources via software to be an abstract idea." *Ericsson Inc. v. TCL Commc'n Tech. Holdings Ltd*., 955 F.3d 1317, 1327 (Fed. Cir. 2020); *see also Smartflash LLC v. Apple Inc.*, 680 F. App'x 977 (Fed. Cir. 2017).

3. The Asserted Claims of the '469 Patent lack an inventive concept.

The specification provides no solution to overcome technical challenges associated with "remote"-ness, or a "high volume of transient traffic" or "subscriber access." The claims of the Asserted Patent are results-driven and fail to overcome ineligibility. Each limitation recites generic components which are "capable of" performing connectivity, routing, and authorization using any method whatsoever. As set forth above and in the attached claim charts, paid Internet hotspots were very well known and widely used at the time of the Asserted Patent. Placing a paid Internet hotspot in "a remote location" is insufficient to render the claims eligible where no particular improvement is recited to, for example, teach *how* a subscriber access unit is "capable of" authorizing payments in "a remote location." *Universal Secure Registry LLC v. Apple Inc*., 10 F.4th 1342, 1346 (Fed. Cir. 2021); *Elec. Power Grp., LLC v. Alstom S.A.*, 830 F.3d 1350, 1354 (Fed. Cir. 2016) ("limiting the claims to the particular technological environment of power-grid

monitoring is, without more, insufficient to transform them into patent-eligible applications of the abstract idea at their core."). The "essentially result-focused, functional character of claim language has been a frequent feature of claims held ineligible under § 101, especially in the area of using generic computer and network technology to carry out economic transactions."

Each of the asserted dependent claims likewise recite the use of conventional components or processes to achieve results for which those components are designed. '469 Patent, claim 25 ("a wired data connection" "a wireless data connection"); claim 26 ("a plurality of users **may access** the Internet simultaneously," "at the remote location," "establishing data connections with the router via their web-ready devices"); claim 28 ("the data connections include wireless data connections"); claim 32 ("the wireless connection," "802.11a wireless area network," "802.11b wireless area network," "802.11g wireless area network," "802.11n wireless area network"). *See also CyberSource Corp. v. Retail Decisions, Inc*., 654 F.3d 1366, 1370-73 (Fed. Cir. 2011) (a method of verifying the validity of credit card transactions over the Internet held patent-ineligible as directed to an abstract idea); *buySAFE, Inc. v. Google, Inc*., 765 F.3d 1350, 1354-55 (Fed. Cir. 2014) (guaranteeing performance in an Internet transaction was patent-ineligible); *TLI Communications LLC Patent Litigation*, 823 F.3d 607, 615 (Fed. Cir. 2016) ("vague, functional descriptions of server components are insufficient to transform the abstract idea into a patent-eligible invention."). The claimed steps are described at a high level of generality, using generic terms, confirming the '469 is ineligible under 35 U.S.C. §101.

F. U.S. Patent No. 7,257,582

1. State of the Art Relevant to the '582 Patent in March 2002

The principles underlying the '582 Patent's claims were widely known and implemented in numerous systems prior to the claimed priority date of March 13, 2002. Long before March 2002, distributed computing frameworks had already embraced first-come, first-served dynamic

task assignment, self-scheduling workloads, and efficient data partitioning across multiple processing nodes. These methods were integral to parallel computing, volunteer computing initiatives, and large-scale database management—fields in which researchers had been refining these concepts for over a decade prior to March 2002.

By the late 1990s, high-performance computing ("HPC") systems, parallel databases, and distributed file systems were actively employing automated load balancing techniques, allowing processors to dynamically claim tasks as they became available. The master-worker paradigm, in which a master process assigns work to multiple processors and redistributes tasks based on availability, was a foundational computing concept. Volunteer computing projects like SETI@home (1999) had already demonstrated the real-world application of parallel task distribution and result aggregation, processing enormous datasets by dynamically distributing work units to thousands of participating machines. Similarly, industry-standard MPP (massively parallel processing) databases had been using data partitioning techniques since the 1980s to distribute processing workloads across multiple nodes, enabling efficient parallel query execution.

These well-established techniques, which had been publicly disclosed and successfully deployed in both academic and industry settings, directly contradict any claim that the '582 Patent introduced a novel approach. The state of the art at the time confirms that the patent merely recites preexisting, widely implemented concepts in distributed computing and parallel processing.

a. Dynamic Scheduling in Parallel Processing was Well Known by March 2002.

By the early 1990s, dynamic scheduling of computational workloads was an established technique in parallel computing. Long before March 2002, researchers and industry practitioners had widely implemented first-come, first-served task scheduling to efficiently utilize processor resources. The master-worker paradigm—where a central process partitions a workload and

distributes it dynamically to available processors—was already a fundamental principle of parallel computing systems. Further, the bag-of-tasks model, introduced by Carriero and Gelernter in the late 1980s, demonstrated how worker processes could independently pull new tasks from a shared task pool, ensuring that faster processors were never idle while waiting for slower ones. This dynamic, demand-based task assignment allowed distributed computing systems to scale efficiently and optimize processor utilization.

By 1990, researchers had expanded on this concept through self-scheduling for parallel loops, where processors independently claimed computational iterations rather than following a static, predetermined allocation. Hewlett-Packard's 1994 patent, U.S. Patent No. 5,325,493, further reinforced this technique by describing a system in which processors would automatically request and execute tasks on a first-come, first-served basis, eliminating bottlenecks and ensuring that computational work was dynamically distributed. In that system, each processor, upon finishing its current task, automatically requests the next task, and processed data is eventually recombined in order. The patent emphasizes that this approach maximizes parallel throughput by not waiting for slower processors.

The same principle was later embodied in volunteer computing projects such as SETI@home (1999), which partitioned large astronomical data files into discrete work units and assigned them to available processors across a network. Each participant's system would request the next available task, execute it, and return results asynchronously to be merged into the central database. This demonstrated not only the viability but also the widespread public deployment of dynamically distributed task execution in a distributed computing environment. By 2000, Intel's patent application (US 2002/0040381 A1, filed October 2000) described a multi-processor load-balancing system that placed computational tasks into a queue, where processors would pull the

next available job upon completion of a prior task, ensuring continuous processor utilization. This system functioned precisely as described in the ‘582 Patent: tasks were not assigned in advance but dynamically distributed as processors became available.

These techniques were not confined to academic research—they were actively deployed in high-performance computing, commercial database processing, and networked data analysis. Systems such as IBM LoadLeveler, Sun Grid Engine, and Condor were already widely used in enterprise computing to dynamically distribute jobs among processors. These systems employed task scheduling policies, including first-come, first-served scheduling, priority-based scheduling, and adaptive load balancing, ensuring optimal utilization of processing resources. The ‘582 Patent describes an approach where subtasks are executed in parallel and reassigned based on availability, but this was precisely the methodology employed by Condor—a job scheduling system developed at the University of Wisconsin, which allowed idle processors to claim unassigned tasks dynamically. Similarly, the Globus Toolkit, an essential framework for grid computing, provided mechanisms for distributed task execution across heterogeneous systems, allowing real-time reassignment of workloads based on processor capacity. Google’s 2004 MapReduce paper would later credit these well-established techniques as the foundation for large-scale distributed computing, confirming that the core scheduling method claimed in the ‘582 Patent was well-known and widely implemented before March 2002

The ‘582 Patent itself acknowledges that job scheduling in parallel systems was well understood, admitting that dynamic task execution techniques were already known in the art. The first-come, first served scheduling policy, in particular, was not a novel concept in the context of multi-processor computing; rather, it was a basic scheduling principle applied in numerous high-performance computing environments. U.S. Patent No. 6,334,137 (2001) further reinforces this

point, as it describes a system for controlling the parallel execution of jobs, dynamically registering jobs, and balancing workloads among multiple computing nodes. Similarly, U.S. Patent No. 5,765,166 (1998) highlights a parallel processing approach for multiple hypothesis tracking, which includes task distribution based on processor availability, mirroring the claim structure of the '582 Patent. Thus, dynamic scheduling as described in the '582 Patent was already well established in both commercial products and prior patents, confirming that the claimed invention lacks novelty.

b. Data Partitioning and Shared Processing was Well Known by March 2002.

By March 2002, parallel data processing had been an essential component of high-performance computing for decades. In the 1980s, massively parallel processing (MPP) databases such as Teradata had introduced data partitioning, where large datasets were automatically divided across multiple processors for simultaneous execution. Research projects such as the Gamma database further developed this concept, refining techniques for parallel query execution on partitioned data. These advances laid the foundation for modern distributed data processing systems and demonstrated that splitting an input dataset into independently processed partitions was a well-established practice.

The concept of shared file access across multiple processors was equally well understood. The Andrew File System (AFS), developed in the 1980s, allowed multiple machines to access and process the same file concurrently, enabling parallel computation on shared datasets. By the late 1990s, enterprise computing environments had fully embraced Network Attached Storage (NAS) and Storage Area Networks (SAN), providing high-speed, concurrent access to large datasets by multiple processing nodes. These technologies made it trivial for a cluster of machines to read from and write to a shared file, a key capability claimed in the '582 Patent.

Meanwhile, volunteer computing projects such as SETI@home and distributed.net (1997) were actively employing data partitioning and parallelized computation across thousands of machines worldwide. SETI@home, for example, split radio telescope data into work units, distributed them to participating computers, and recombined results asynchronously, demonstrating large-scale parallel processing on shared datasets. Similarly, distributed.net divided large cryptographic keyspaces into smaller search ranges, assigned them to individual processors, and collected results into a master database—a direct precursor to the distributed workloads envisioned in the '582 Patent.

The process of dividing large computational tasks across multiple processors and merging the results was also well established in high-performance computing (HPC). Systems such as IBM's LoadLeveler and Condor (1990) scheduled jobs across clusters of computers, often breaking down large computational tasks into parallelized workloads. Parallel database management systems, including those developed by Oracle and IBM, routinely split and processed massive tables across multiple processors to accelerate query performance. These well-documented techniques anticipated the purported invention of automatically partitioning an input dataset, distributing partitions across processors, and merging the results into an output dataset.

Further, prior to March 2002, Message Passing Interface (MPI) and Parallel Virtual Machine (PVM) had become the standard for distributed computing, providing detailed methodologies for logically subdividing workloads across multiple nodes. These frameworks enabled data segmentation, real-time allocation of tasks to available processors, and efficient reassembly of processed outputs. Similarly, Parallel Database Management Systems (DBMS), including Oracle Parallel Server and IBM DB2, had implemented partitioning strategies for decades, allowing concurrent query execution across multiple computing nodes. Likewise, U.S.

Patent No. 5,535,393 (1996) discloses a system for optimally distributing workloads among processors while minimizing data contention—fundamentally the same goal the '582 Patent seeks to achieve. U.S. Patent No. 6,304,866 (2001) discusses a multiprocessing system employing on-demand and incremental sub-task allocation, further demonstrating that the concept of dynamically segmenting data was already well known. The notion that file processing can be subdivided logically and executed in a distributed manner was, therefore, a standard principle in computing prior to March 2002.

c. Conclusion

Thus, the '582 Patent fails to introduce any novel aspect in how partitions are defined, stored, or processed. It relies on general functional language, describing the need to logically subdivide records without detailing any new approach or specific improvement over conventional systems. As of the March 2002 earliest possible priority date, the practice of splitting files for parallel execution—whether by byte ranges, record boundaries, or database partitioning schemes—was fully established in the art.

2. The '582 Patent Claims Ineligible Subject Matter Under 35 U.S.C. § 101

The Asserted Claims (claims 1, 2, 5, and 7) of the '582 Patent are invalid under 35 U.S.C. § 101 because they are directed to an abstract idea—specifically, the idea of dividing a large task into subtasks and processing them using available resources on a first-come/first-served basis—and lack any inventive concept that transforms this abstract idea into patent-eligible subject matter. The Asserted Claims of the '582 Patent therefore fail at both steps of the Alice test and are invalid. See *Alice Corp. v. CLS Bank Int'l*, 573 U.S. 208 (2014).

The ‘582 Patent, titled “Method for Managing Distributed Processing of Tasks in a Network,” claims a method for subtask processing in distributed computing systems. Claim 1—the only independent claim of the ‘582 Patent—recites:

> 1. A method of effecting on **a preexisting input file** a computer-executable process **comprised of a plurality of subtasks**, the method comprising the steps of:
>
> (a) automatically determining file allocation and **logically subdividing records of said input file into a plurality of partitions**;
>
> (b) **distributing descriptions of** all of said **partitions to** each of a plurality of **subtask processors**[;]
>
> (c) **simultaneously executing** at least a respective **one of the subtasks** of the computer-executable process in each of at least some of said processors on a respective one of the partitions with each subtask reading and processing the respective partition so as to process the respective partition and produce respective subtask output and;
>
> (d) thereafter **repeating step (c)** in at least some of the subtask processors each **with another unprocessed partition on a first-come/first-served basis**; and
>
> (e) generating at least one output **combining all of the subtask outputs** and reflecting the processing of all of said subtasks.

The patent admits that these steps mostly were well known in the art, including partitioning data, distributing tasks, parallel processing, and aggregating results. Dkt#1-6 at 3:35-47, 3:67-4:15, 4:59-63. Partitioning (step a) relies on conventional techniques, such as byte ranges or track addresses, while subtask distribution (step b) employs generic mechanisms using control files; processing subtasks (step c) involves standard read-process-write operations, and aggregation (step e) is “of course” a conventional merging operation. *Id.* at 3:35-4:15, 4:59-63. The dependent Asserted Claims of the ‘582 Patent fare no better. Claim 2 merely states that a separate processor may perform the partitioning and distribution—an obvious and conventional division of labor that lacks any inventive concept. Claim 5 recites that subtask outputs can be accumulated in arbitrary order but offers no technical detail or mechanism for doing so—just a result. Claim 7 limits the

input file to being stored on a network-attached storage device, which the patent itself identifies as a well-known technology.

The claimed first-come/first-served scheduling (step d) was argued during prosecution as the "primary difference" over the prior art. *See* Ex. A, Mar. 2, 2007 Remarks. According to the applicant, the prior art used "load information to distribute the load between processors," whereas:

> *With the instant invention … load information is not created* …. Instead, the load sharing is done as a byproduct of the fact that *the load-sharing process take parts of the load on a first-come/first-served basis*. A comparison would be to a road intersection where, according to the prior art, there is a traffic light that determines who can go when. The instant invention is more like such an intersection with a four-way stop so that the individual drivers determine who can go and when.

Id. The patentee's admission about the "primary difference" between the claimed invention and the prior art is critical in two ways. First, by pinpointing the "first available" assignment technique as the point of novelty, the inventors have defined what the claimed invention is "directed to" for purposes of the *Alice* Step 1 inquiry. *See Affinity Labs of Tex., LLC v. DIRECTV, LLC*, 838 F.3d 1253, 1257–58 (Fed. Cir. 2016) ("The 'abstract idea' step of the inquiry calls upon us to look at the 'focus of the claimed advance over the prior art' to determine if the claim's 'character as a whole' is directed to excluded subject matter."). Second, by the simple comparison between the invention and the managing of traffic loads at a street intersection, the inventor acknowledged that his "invention" merely applies a longstanding method of organizing human activities to computers.

The methods described in the '582 Patent implement that abstract concept using generic computing components, such as processors and memory, which are standard elements in distributed systems. The claims use simple functional language to describe desired outcomes—partitioning, distributing, processing, scheduling, and aggregating—without detailing how these

operations are performed or improved. Consequently, the Asserted Claims of the '582 Patent fail at both steps of the Alice test and are therefore invalid.

a. The Asserted Claims of the '582 Patent recite an abstract scheduling technique.

The Asserted Claims of the '582 Patent are directed to the abstract idea of dividing a large task into subtasks and processing them using available resources on a first-come/first-served basis. Claim 1 of the '582 Patent—the only independent claim of the '582 Patent—comprises five steps: (a) "logically subdividing records . . . into a plurality of partitions," (b) distributing descriptions of those partitions to multiple "subtask processors," (c) "simultaneously executing" at least one "subtask" on each of the subtask processors, (d) thereafter allocating any remaining "unprocessed partition" to the subtask processors "on a first-come/first-served basis," and (e) aggregating "all of the subtask outputs" into a final combined output. Stripped of jargon, Claim 1 covers the basic concept of splitting up a task and distributing the work, then collecting the results.

Claim 1's distributed task management approach is an abstract idea that humans have long practiced without computers; it is akin to breaking a large job into smaller tasks, handing those tasks out to a group of workers, giving each new work as they free up, and then assembling the final product. The practice of breaking up a large job into discrete tasks, passing out a first round of assignments to team members, instructing them to ask for a new assignment when finished, and then combining the results has been employed by humans since the building of the Egyptian pyramids. The Federal Circuit has consistently held that processes based on such fundamental organizational practices are abstract. *See, e.g.*, *Intellectual Ventures I LLC v. Capital One Bank (USA)*, 792 F.3d 1363, 1367 (Fed. Cir. 2015) (organizing, storing, and retrieving information held abstract); *Intellectual Ventures I LLC v. Symantec Corp.*, 838 F.3d 1307 (Fed. Cir. 2016)

(automatically organizing information into wanted and unwanted categories and discarding the unwanted information held abstract).

"[T]here is a critical difference between patenting a particular concrete solution to a problem and attempting to patent the abstract idea of a solution to the problem in general." *Electric Power Group, LLC v. Alstom S.A.*, 830 F.3d 1350, 1356 (Fed. Cir. 2016). The Asserted Claims of the '582 Patent fall on the latter side of that line: they seek to monopolize the general idea of a solution (parallel task processing with on-the-fly scheduling) rather than a concrete technological solution. The Asserted Claims of the '582 Patent do not improve computer technology or solve a specific technical problem in how computers operate but instead apply conventional and common-sense workload management ideas to distributed computing. Partitioning data, distributing tasks, processing in parallel, and aggregating results are standard operations in distributed computing systems. Dkt#1-6 at 3:35–4:15, 4:59–63. Simply executing a known business/workflow practice on generic computers is not a patentable improvement to those computers.

The '582 Patent's recitation of assigning work using first-come/first-served scheduling—a ubiquitous method of task processing akin to queue management (e.g., a line at a concession stand or a four-way stop)—demonstrates its abstract nature. Claim 1 applies this principle to parallel computer processing without offering any new technological innovation. Moreover, The Asserted Claims of the '582 Patent use abstract functional language to describe desired outcomes—such as "logically subdividing," "distributing," and "aggregating"—without detailing how these steps are performed. This kind of result-oriented claiming that specifies an outcome (efficient parallel processing) without limiting the claim to any particular innovative technique for achieving it is a hallmark of an abstract idea. Thus, the '582 Patent "merely applies a well-known idea using generic computers 'to the particular technological environment of the Internet.'"

Intellectual Ventures I LLC v. Symantec Corp., 838 F.3d 1307, 1314 (Fed. Cir. 2016). In *Two-Way Media*, the Federal Circuit held that reciting a series of steps to achieve a desired result, without specifying how those steps are performed, renders the claim abstract. *See Two-Way Media Ltd. v. Comcast Cable Commc'ns*, LLC, 874 F.3d 1329, 1337–38 (Fed. Cir. 2017) ("routing" and "accumulating records" held abstract and invalid). Even though the specification describes conventional methods for these operations, the Asserted Claims of the '582 Patent do not specify how data is partitioned, how partitioned subtasks are distributed, or how results are aggregated. Dkt#1-6 at 3:35–4:15, 4:59–63. Such high-level, results-oriented language underscores the claim's abstract nature. As the Federal Circuit noted in *Electric Power*, claims that focus on the outcome of data processing without specifying the means to achieve it are abstract and invalid. 830 F.3d at 1356.

IV argues that the Asserted Claims of the '582 Patent are not abstract because they are "directed to an improvement in data processing"—specifically claiming that using multiple computers in parallel "increases the speed of execution of processes and balances the load" on resources. Dkt#28 at 9-10. The claimed method only achieves these benefits by applying the abstract idea of task partitioning and parallel execution on generic computers, rather than by reciting a new technology. Improved speed and load balancing are expected results of running a workload in parallel; they are performance outcomes, not inventive technical means. The '582 Patent claims do not improve the internal operation of the computer processors or change how the network functions—they just use more computing power in a straightforward way. Merely accelerating an existing process via parallelization is akin to the scenario in S*AP America v. InvestPic*, where claims "selecting certain information, analyzing it using mathematical techniques, and reporting or displaying the results of the analysis" were held abstract. 898 F.3d

1161, 1167-1168 (Fed. Cir. 2018). Indeed, the claims of the '582 Patent "fit into the familiar class of claims that do not 'focus ... on [] an improvement in computers as tools, but on certain independently abstract ideas that use computers as tools.'" *Id.* at 1168 (quoting *Electric Power*, 830 F.3d at 1354). Furthermore, IV's reliance on *Enfish, LLC v. Microsoft Corp.*, 822 F.3d 1327 (Fed. Cir. 2016), is misplaced. Unlike *Enfish*, where the claims improved database memory structure, the '582 Patent does not specify any new hardware or improvement to system architecture—it merely applies a standard method of parallelization to a computing process.

Because the Asserted Claims of the '582 Patent of the '582 Patent are directed to a fundamental method for organizing and processing tasks, it falls squarely within the category of claims deemed ineligible under Step 1 of the *Alice* framework.

b. The '582 Patent lacks an inventive concept.

As to *Alice* Step 2, The Asserted Claims of the '582 Patent recite only conventional steps and generic components without introducing any meaningful technical improvements or innovations. The claim elements, individually and in combination, reflect routine operations in distributed computing, as explicitly acknowledged in the specification. Partitioning input data into subsets, distributing tasks to processors, and processing tasks in parallel were standard techniques for optimizing workload distribution at the time. Dkt#1-6 at 3:35-4:15, 4:59-63.

The first-come/first-served scheduling technique, likewise, is a well-established method for processing tasks based on availability, like the applicant's example of the four-way stop sign. The Asserted Claims of the '582 Patent "merely appl[y] [this] well-known idea using generic computers." *Symantec*, 838 F.3d at 1314. The specification does not describe any modification, improvement, or adaptation of this scheduling technique to make it unique or unconventional in the context of distributed computing. The Asserted Claims of the '582 Patent simply apply this known concept predictably to allocate unprocessed subtasks, a principle embedded in all multi-

threaded and distributed computing architectures, and offers no specific mechanisms or technical enhancements. Thus, The Asserted Claims of the '582 Patent are "directed to a result or effect that itself is the abstract idea and merely invoke[s] generic processes and machinery" rather than "a specific means or method that improves the relevant technology." *Yu v. Apple Inc.*, 1 F.4th 1040, 1043 (Fed. Cir. 2021). "What is claimed is simply a generic environment in which to carry out the abstract idea." *Id.*

Confirming the routine nature of the claimed methods, both the specification and the Asserted Claims of the '582 Patent rely entirely on generic computing components, such as processors, memory, and standard network architecture. Dkt#1-6 at 4:27–36; *Trinity Info Media, LLC v. Covalent, Inc.*, 72 F.4th 1355, 1364, 1366–67 (Fed. Cir. 2023) (generic computer components such as "processors" and "memory" insufficient to add an inventive concept). The patent does not propose any novel configurations or functionalities for these components but instead recites conventional "functions in general terms, without limiting them to technical means for performing the functions that are arguably an advance over conventional computer and network technology." *Elec. Power*, 830 F.3d at 1351.

For example, Claim 1's reliance on functional language—such as "automatically determining," "logically subdividing," "distributing," "executing," and "combining"—describes desired outcomes without specifying how those outcomes are achieved. It does not explain how data is partitioned into subsets, how subtasks are allocated, how unallocated tasks are managed, or how results are aggregated into a final output. Dkt#1-6 at 6:47–4:15, 4:59–63. This lack of specificity leaves the claim framed at a high level of abstraction, with no concrete implementation details that could transform the method into a patent-eligible application. The dependent Asserted Claims of the '582 Patent fare no better. Claim 2 merely delegates partitioning and distribution to

another processor, without explaining how that processor performs the task. Claim 5 generically allows output in arbitrary order but lacks any technical detail on how that's achieved. Claim 7 limits the input file to network-attached storage—a conventional environment the patent admits was already well known.

Courts have consistently held analogous claims ineligible under Step 2 of the Alice framework. In *Content Extraction*, 776 F.3d at 1348, the court found "no 'inventive concept'" in the use of generic computer components to perform routine activities. Similarly, in *Symantec Corp.*, 838 F.3d at 1320, the Federal Circuit emphasized that for a computer-implemented invention to be patent-eligible, it must involve more than well-understood, routine, and conventional activities. Like the claims in these cases, the Asserted Claims of the '582 Patent merely apply abstract principles of task management to distributed computing using conventional techniques, which does not suffice to render it patent-eligible.

IV points to the patent's description of using heterogeneous computers and balancing load as evidence that the claims capture a technical innovation. This argument is unpersuasive. As detailed above, the combination of steps in the Asserted Claims of the '582 Patent are entirely conventional—they are the standard recipe for parallel processing (split, distribute, process, repeat, merge). There is no inventive coordination or unconventional synergy arising from the arrangement of these steps; each step does what one would normally expect, and together they achieve the expected outcome of parallel execution. Unlike in *CosmoKey*, where the claim recited a specific authentication method that was not routine, or *Enfish* where a new data structure improved computer functionality, the Asserted Claims of the '582 Patent simply assemble known techniques to carry out an abstract goal. 15 F.4th at 1098-1099; 822 F.3d at 1337. Combining conventional steps in a computer environment, without more, does not create an inventive concept.

IV emphasizes that the patent describes using multiple heterogeneous computers and optimizing execution, but those aspects were already familiar in distributed computing. *See* Section III.F.1 *supra*.

Viewed as a whole, the Asserted Claims of the '582 Patent combine conventional steps in a predictable manner to achieve the abstract goal of managing distributed tasks. They offer no technological advancements or inventive concepts beyond the abstract idea itself. Accordingly, the '582 Patent fails to meet the requirements for patent eligibility under 35 U.S.C. § 101.

IV. THE '844 PATENT IS INVALID UNDER 35 U.S.C. §§ 102 AND/OR 103

A. The '844 Patent Priority Date

The asserted claims of the '844 Patent have a filing date of February 21, 2007. Application No. 11/709,477 (the '477 Application), which matured into the '844 Patent, was filed on February 21, 2007 and was a continuation-in-part of Application No. 11/395,816 (filed March 30, 2006) (now Patent No. 7,721,282) (the '816 Application), which was a continuation-in-part of Application No. 11/026,622 (filed December 20, 2004) (now abandoned) (the '622 Application). Plaintiff's Infringement Contentions allege that the asserted claims of the '844 Patent are entitled to a priority date of December 30, 2004, which is the filing date of the '622 Application.

However, none of the '844 Patent claims are entitled to either priority date, as the '816 and '622 Applications never mention caching or indexing – at least one of which is expressly recited in each claim of the '844 Patent. '844 Patent, 10:55- 14:17. *See Studiengesellschaft Kohle v. Shell Oil Co.*, 112 F.3d 1561, 1564 (Fed. Cir. 1997). Applicants even admitted that "indexing and indexing results are not discussed anywhere within Nguyen" (which matured from a continuation-in-part of the '622 Application). *See* '477 Application, Amendment After Non-Final Rejection dated February 21, 2007. Therefore, the asserted claims of the '844 Patent are only entitled to a filing date of February 21, 2007.

B. Identification of Prior Art that Anticipates or Renders Obvious the Asserted Claims of the '844 Patent

Southwest contends that the prior art references listed below and charted in Exhibits 844-1 – 844-24 anticipate and/or render obvious, alone or in combination, one or more of the asserted claims of the '844 Patent.

Prior Art Reference	Publication Date
"Optimizing the Migration of Virtual Computers," Association for Computing Machinery, Special Interest Group on Operating Systems, Operating Systems Review, Proceedings of the 5th Symposium on Operating Systems Design and Implementation, Volume 36, Issue S1 ("Sapuntzakis")	Published December 31, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b).
U.S. Patent No. 7,089,300 ("Birse")	Filed – October 19, 1999 Issued – August 8, 2006 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No. 2005/0283597 ("Holzmann")	Filed – June 22, 2004 Published – December 22, 2005 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b) and § 102(e).
U.S. Patent Application Publication No. 2002/0034105 ("Kulkarni")	Filed – January 8, 2001 Published – March 21, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b) and § 102(e).
U.S. Patent Application Publication No. 2005/0004925 ("Stahl")	Filed – May 7, 2004 Published – January 6, 2005 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b) and § 102(e).

Prior Art Reference	Publication Date
U.S. Patent Application Publication No. 2006/0168154 ("Zhang")	Filed – November 19, 2004 Published – July 27, 2006 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent No. 8,001,085 ("Kiselev")	Filed – November 25, 2003 Issued – August 16, 2011 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No. 2004/0172509 ("Takeda")	Filed – June 23, 2003 Published – September 2, 2004 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b) and § 102(e).
U.S. Patent No. 6,671,782 ("Menon")	Filed – March 29, 2000 Issued – December 30, 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b) and § 102(e).
U.S. Patent Application Publication No. 2003/0236850 ("Kodama")	Filed – March 21, 2002 Published – December 25, 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b) and § 102(e).
EP Patent No. EP0604013 ("Nelson")	Filed – November 15, 1993 Issued – February 20, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b).
PCT Application Publication No. WO 2006/074869 ("Bayer")	Filed – January 4, 2006 Published – July 20, 2006 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b) and § 102(e).
Shrira, Liuba et al., "Thresher: an efficient storage manager for copy-on-write snapshots," ATEC '06: Proceedings of the	Published at least by May 30, 2006 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(a).

Prior Art Reference	Publication Date
annual conference on USENIX '06 Annual Technical Conference (Shrira)	
Laros, James H. et al., "Implementing Scalable Disk-Less Clusters Using The Network File System (NFS)," ("Laros")	Published at least by January 2003 Retrieved from: https://www.researchgate.net/publication/255223070_Implementing_scalable_disk-less_clusters_using_the_Network_File_System_NFS This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b).
U.S. Patent Application Publication No. 2004/0153639 ("Cherian")	Filed – February 5, 2003 Published – August 5, 2004 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(b) and (e).
U.S. Patent Application Publication No. 2007/0057958 ("Bucher")	Filed – August 9, 2006 Published – March 15, 2007 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent No. 7,197,608 ("Mikuma")	Filed – July 12, 2004 Issued – March 27, 2007 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No. 2005/0283575 ("Kobayashi")	Filed – August 13, 2004 Published – December 22, 2005 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(b) and (e).
U.S. Patent No. 6,453,334 ("Vinson")	Filed – June 16, 1998 Issued – September 17, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(b) and (e).
U.S. Patent No. 5,930,513 ("Taylor")	Filed – June 6, 1996

Prior Art Reference	Publication Date
	Issued – July 27, 1999 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(b) and (e).
U.S. Patent Application Publication No. 2002/0002660 ("Malcolm")	Filed – July 31, 1998 Published – January 3, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(b) and (e).
U.S. Patent Application Publication No. 2002/0091763 ("Shah")	Filed – May 15, 2001 Published – July 11, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(b) and (e).
U.S. Patent No. 7,398,382 ("Rothman")	Filed – December 29, 2004 Issued – July 8, 2008 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent No. 6,317,826 ("McCall")	Filed – October 19, 1998 Issued – November 13, 2001 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(b) and (e).
Southwest Airlines NetApp System – NetApp/Southwest	At least as early as 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a) or (g).
VMware Systems - VMware	At least as early as 2001 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a) or (g).

Plaintiffs assert claims 7 and 11 of the '844 Patent against Southwest in this lawsuit. These claims are invalid because the '844 Patent fails to meet one or more of the requirements for patentability. The individual bases for invalidity are provided below and in Exhibits 844-1 – 844-

24. Each of the prior art references, the underlying work, and/or the underlying apparatus or method qualifies as prior art under one or more sections of Pre-AIA 35 U.S.C. §§ 102 and/or 103.

In an effort to focus the issues, Southwest has cited representative portions of identified references, even where a reference may contain additional disclosures or support for a particular claim element. Each and every disclosure of the same limitation in the same reference is not necessarily identified. In addition, persons of ordinary skill in the art generally read a prior art reference as a whole and in the context of other publications and literature and in the context of their knowledge as persons of ordinary skill in the art. Thus, to understand and interpret any specific statement or disclosure within a prior art reference, such persons would rely on other information within the reference, along with other publications and their general scientific knowledge and experience in the field. Thus, each prior art reference identified herein should be understood in the context of the reference itself in conjunction with anything incorporated by reference or described as relevant technology by the reference, any documents that are referred to for further understanding by the reference, any product embodying the reference, and any testimony by those with knowledge of the reference, such as the named authors and inventors. Southwest may rely upon uncited portions of the prior art references and on other publications and expert testimony to provide context, and as aids to understanding and interpreting the portions that are cited. Southwest may also rely on uncited portions of the prior art references, other disclosed publications, and the testimony of experts to establish that a person of ordinary skill in the art would have been motivated to modify or combine certain of the cited references so as to render the claims obvious.

Some or all of the asserted claims of the ‘844 Patent are invalid as anticipated under Pre-AIA 35 U.S.C. § 102 in view of certain ones of the prior art references identified above as detailed

in the claim charts included in Exhibits 844-1 – 844-24, which identify specific examples of where each limitation of the asserted claims of the '844 Patent is found in the prior art references. As explained above, the cited portions of prior art references identified in the attached claim charts are exemplary in nature and representative of the content and teaching of the prior art references, and should be understood in the context of the reference as a whole and as they would be understood by a person of ordinary skill in the art.

To the extent any item of prior art cited above is deemed not to disclose, explicitly or inherently, any limitation of an asserted claim of the '844 Patent, Southwest reserves the right to argue that any difference between that prior art and the corresponding patent claim would have been either inherent in the art or obvious to a person of ordinary skill in the art, even if Southwest has not specifically denoted that the art is to be combined with the knowledge of a person of ordinary skill in the art.

To the extent Plaintiffs argue that any limitation of the asserted claims of the '844 Patent is not explicitly disclosed by an item of prior art identified above and in Exhibits 844-1 – 844-24, any purported differences are such that the claimed subject matter as a whole would have been obvious to one skilled in the art at the time of the alleged invention, in view of the state of the art and knowledge of those skilled in the art. The item of prior art would, therefore, render the relevant claims invalid as obvious under Pre-AIA 35 U.S.C. § 103.

To the extent any of the foregoing references are found not to anticipate the asserted claims of the '844 Patent, the foregoing references render the asserted claims obvious, either alone or in combination with one or more of the other references identified above. One of ordinary skill in the art would have been motivated to combine the various references cited herein so as to disclose each and every limitation of the asserted claims of the '844 Patent.

The prior art identified in Exhibits 844-1 – 844-24 renders one or more asserted claims of the '844 Patent obvious when the references are read alone or in combination with one or more of each other and/or when read in view of the state of the art and knowledge of those skilled in the art.

To the extent that Plaintiff argues that any asserted claim of the '844 Patent is not disclosed or rendered obvious by any reference charted in Exhibits 844-1 – 844-24, the asserted claims nevertheless would have been obvious in light of one or more other references charted in Exhibits 844-1 – 844-24 and in view of the knowledge, experience, and understanding of persons of ordinary skill in the art.

Each and every reference identified is also relevant to the state of the art at the time of the alleged invention. Any of the references disclosed above may be combined with one another to render obvious (and therefore invalid) each of the asserted claims of the '844 Patent. Southwest may rely upon a subset of the above identified references, or all of the references identified above, including all references in charted in Exhibits 844-1 – 844-24, for purposes of obviousness depending on the Court's claim construction, positions taken by Plaintiffs during this litigation, and further investigation and discovery.

1. Example Combinations of the Prior Art

The table below provides an identification of exemplary combinations of prior art references that, based on Southwest's present understanding of Plaintiff's Infringement Contentions and apparent reading of the claims, render obvious one or more of the asserted claims of the '844 Patent. The combinations of references identified below are non-exhaustive, and additional examples are apparent from other combinations of references charted in Exhibits 844-1 – 844-24. Although certain references charted in Exhibits 844-1 – 844-24 are listed variously in the left or right column below, Southwest does not thereby imply any restriction as to which

reference is primary and which reference is secondary in such a combination. Southwest reserves the right to rely on either identified reference in the identified combinations as a primary and/or secondary reference.

Exhibit	Reference	Combination Reference(s)
844-1	Sapuntzakis	Holzmann, Nelson, Stahl, and/or Kulkarni, and/or one or more references as charted in Exhibits Exhibits 844-1 – 844-24
844-2	Birse	Holzmann, Nelson, Stahl, and/or Kulkarni, and/or one or more references as charted in Exhibits Exhibits 844-1 – 844-24
844-3	Kulkarni	Holzmann, Nelson, and/or Stahl, and/or one or more references as charted in Exhibits 844-1 – 844-24
844-4	Stahl	Holzmann, Nelson, Sapuntzakis, Birse, and/or Kulkarni, and/or one or more references as charted in Exhibits Exhibits 844-1 – 844-24
844-5	Zhang	Holzmann, Nelson, Stahl, Sapuntzakis, Birse, and/or Kulkarni, and/or one or more references as charted in Exhibits 844-1 – 844-24
844-6	Kiselev	Holzmann, Nelson, and/or Stahl, and/or one or more references as charted in Exhibits 844-1 – 844-24
844-7	Takeda	Holzmann, Nelson, and/or Stahl, and/or one or more references as charted in Exhibits 844-1 – 844-24
844-8	Menon	Holzmann, Nelson, Stahl, Sapuntzakis, Birse, and/or Kulkarni, and/or one or more references as charted in Exhibits 844-1 – 844-24
844-9	Kodama	Holzmann, Nelson, Stahl, Sapuntzakis, Birse, and/or Kulkarni, and/or one or

Exhibit	Reference	Combination Reference(s)
		more references as charted in Exhibits 844-1 – 844-24
844-10	Laros	Holzmann, Nelson, Stahl, Sapuntzakis, Birse, and/or Kulkarni, and/or one or more references as charted in Exhibits 844-1 – 844-24
844-11	Shrira	Holzmann, Nelson, Stahl, Sapuntzakis, Birse, and/or Kulkarni, and/or one or more references as charted in Exhibits 844-1 – 844-24
844-12	Bayer	Holzmann, Nelson, Stahl, Sapuntzakis, Birse, and/or Kulkarni, and/or one or more references as charted in Exhibits 844-1 – 844-24
844-13	Cherian	One or more references as charted in Exhibits 844-1 – 844-24
844-14	Bucher	One or more references as charted in Exhibits 844-1 – 844-24
844-15	Mikuma	One or more references as charted in Exhibits 844-1 – 844-24
844-16	Kobayashi	One or more references as charted in Exhibits 844-1 – 844-24
844-17	Vinson	One or more references as charted in Exhibits 844-1 – 844-24
844-18	Taylor	One or more references as charted in Exhibits 844-1 – 844-24
844-19	Malcolm	One or more references as charted in Exhibits 844-1 – 844-24
844-20	Shah	One or more references as charted in Exhibits 844-1 – 844-24
844-21	Rothman	One or more references as charted in Exhibits 844-1 – 844-24

Exhibit	Reference	Combination Reference(s)
844-22	McCall	One or more references as charted in Exhibits 844-1 – 844-24
844-23	Southwest Airlines NetApp System	Holzmann, Nelson, Stahl, Sapuntzakis, Birse, and/or Kulkarni, and/or one or more references as charted in Exhibits 844-1 – 844-24
844-24	VMware Systems	Holzmann, Nelson, Stahl, Sapuntzakis, Birse, and/or Kulkarni, and/or one or more references as charted in Exhibits 844-1 – 844-24

It will be understood that, given the similarities between the art, any of Sapuntzakis, Birse, Kulkarni, Stahl, Zhang, Kiselev, Takeda, Menon, Kodama, Holzmann, Nelson, Cherian, Bucher, Mikuma, Kobayashi, Vinson, Taylor, Malcolm, Shah, Rothman, McCall, Laros, Shrira, Bayer, Southwest Airlines NetApp System, and/or VMware Systems could be combined with one or more of each other, for example based on the disclosures shown in Exhibits 844-1 – 844-24. It would have been obvious to combine these references for reasons that are readily apparent to one of ordinary skill in the art, and for reasons explained in additional detail in the sections that follow.

2. The Claims Recite Well-Known Concepts that Would Have Been Obvious to Implement

As noted above, the asserted claims of the '844 Patent represent nothing more than an arrangement of well-known concepts and features and add nothing to the state of the art at the time of the alleged invention. For each of the concepts represented by the claims, it would have been inherently obvious to add to the teachings of any number of known references and products, including to any of the primary references charted in Exhibits 844-1 – 844-24 to the extent that such reference is alleged to be missing such concept or feature.

3. Additional Reasons to Combine

In addition to the information contained elsewhere in these invalidity contentions, including in Exhibits 844-1 – 844-24, Southwest has identified additional motivations and reasons to combine the various references. Southwest notes that in *KSR Int'l Co. v. Teleflex Inc.*, the Supreme Court held that "[i]n determining whether the subject matter of a patent claim is obvious, neither the particular motivation nor the avowed purpose of the patentee controls. What matters is the objective reach of the claim. If the claim extends to what is obvious, it is invalid under § 103." 550 U.S. 398, 127 S.Ct. 1727, 1742 (2007).

KSR further illustrated several ways in which the subject matter of a patent claim may be shown to be obvious. For instance, "[o]ne of the ways in which a patent's subject matter can be proved obvious is by noting that there existed at the time of invention a known problem for which there was an obvious solution encompassed by the patent's claims." *KSR*, 127 S.Ct. 1727, 1742. The Supreme Court held that it was error to "look only to the problem the patentee was trying to solve." *Id*. Rather, "[t]he question is not whether the combination was obvious to the patentee but whether the combination was obvious to a person with ordinary skill in the art. Under the correct analysis, any need or problem known in the field of endeavor at the time of invention and addressed by the patent can provide a reason for combining the elements in the manner claimed." *Id*. Further, "[c]ommon sense teaches … that familiar items may have obvious uses beyond their primary purposes, and in many cases a person of ordinary skill will be able to fit the teachings of multiple patents together like pieces of a puzzle." *Id.* Further, a showing that a combination of elements was "obvious to try" may show that it was obvious under § 103. *Id.* For instance, the Supreme Court held that "predictable solutions" using a combination of "known options" may render the subject matter of a patent claim obvious.

When there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense. *Id.* The ways in which the subject matter of a patent claim may be shown to be obvious, identified by *KSR*, are merely illustrative. The main thrust of *KSR* was that "[r]igid preventative rules that deny factfinders recourse to common sense"—such as the overturned "teaching, suggestion, or motivation" test—are neither necessary under our case law nor consistent with it." *KSR*, 127 S.Ct. 1727, 1742-43; citing with approval, e.g., *DyStar Textilfarben GmbH & Co. Deutschland KG v. C.H. Patrick Co.*, 464 F. 3d 1356, 1367 (2006) ("Our suggestion test is in actuality quite flexible and not only permits, but requires, consideration of common knowledge and common sense"); *Alza Corp. v. Mylan Labs., Inc.*, 464 F. 3d 1286, 1291 (2006) ("There is flexibility in our obviousness jurisprudence because a motivation may be found implicitly in the prior art. We do not have a rigid test that requires an actual teaching to combine...").

Although Southwest notes that there is no longer a rigid requirement regarding motivation to combine under *KSR*, Southwest provides the following statements regarding motivation to combine. Multiple teachings, suggestions, and/or reasons to modify any of the references and/or to combine any two or more of the references charted in Exhibits 844-1 – 844-24 come from many sources, including the prior art (specific and as a whole), common knowledge, common sense, predictability, expectations, industry trends, design incentives or need, market demand or pressure, market forces, obvious to try, the nature of the problem faced, and/or knowledge possessed by one of ordinary skill. In addition, it would have been obvious to try combining the prior art references identified above because there were only a finite number of predictable solutions and/or because

known work in one field or endeavor prompted variations based on predictable design incentives and/or market forces either in the same field or a different one.

The combination of prior art references identified in these contentions would have been obvious because the combination represents the known potential options with a reasonable expectation of success. Additionally, one of ordinary skill in the art would have been motivated to create combinations identified in these contentions using: known methods to yield predictable results; known techniques in the same way; a simple substitution of one known, equivalent element for another to obtain predictable results; and/or teaching, suggestion, or motivation in the prior art generally. Also, market forces in the industry, and the desire to improve features and performance, would motivate the addition of features to products as they become available, become less expensive, become more commonly used, provide better performance, reduce costs, or predictably achieve other clearly desirable results.

Moreover, in addition to the information contained elsewhere in these contentions, and, in particular, charted in Exhibits 844-1 – 844-24, Southwest has identified additional motivations and reasons to combine the various references cited herein. In particular, multiple teachings, suggestions, and/or reasons to modify any of the references and/or to combine any two or more of the references charted in Exhibits 844-1 – 844-24 come from many sources, including the prior art (specific and as a whole), common knowledge, common sense, predictability, expectations, industry trends, design incentives or need, market demand or pressure, market forces, obvious to try, the nature of the problem faced, and/or knowledge possessed by one of ordinary skill. In addition, it would have been obvious to try combining the prior art references identified above because there were only a finite number of predictable solutions and/or because known work in one field or endeavor prompted variations based on predictable design incentives and/or market

forces either in the same field or a different one. The combination of prior art references identified in these contentions would have been obvious because the claimed combinations represent the known potential options with a reasonable expectation of success.

To the extent that Plaintiffs contend that any of the above-identified references does not expressly or inherently disclose a particular element or limitation, it would have been obvious to a person of ordinary skill in the art to modify the references or to combine their teachings with other prior art references, including the prior art references in these invalidity contentions, in a manner that renders such claims obvious and invalid.

In particular, various references identified above disclose, well before the critical date of the '844 Patent, systems distributed across plurality of compute nodes that provide for creating cohesive application environments by tracking changes to a common image. For example, Sapuntzakis concerns a system architecture for quickly moving the state of a running computer (including its OS, applications and user files on disk, and all data in memory) – called a "capsule" – across a network to another machine. *See* Sapuntazakis, 2. This is achieved by reducing the amount of data sent over the network through use of, for instance, "copy-on-write" ("COW") disk images that track just the updates to the disk-storage component of capsules. *Id*.

As another example, Birse relates to the provision of a OS in a net-booted environment. *See* Birse, 1:18-21. According to one embodiment, a server in a network computer ("NC") system maintains a copy of the OS that cannot be corrupted by ordinary users of the NC system. Additionally, the NC system may preserve user customizations (e.g., preferences) by maintaining individual, user, storage areas. When an NC client boots from the network and accesses a stored copy of the OS from an NC server, the user's preferences are dynamically merged with the system

environment provided to the NC client. *Id.* 2:56-65. Advantageously, because of these features, a user may login to any NC client on the network and have the same user experience. *Id.* 2:65-3:3.

As yet another example, Kulkarni discloses a system comprising a server and a personal digital assistant (PDA). The server is equipped with a processor, memory, and a network interface, enabling communication with remote devices over a network, such as the Internet. The server's memory contains an image. *See* Kulkarni, at [0027]. The image in flash memory serves as the starting point for creating an updated image. A differences file is stored that includes new data sections, data not present in the root image, and references to unchanged sections of the root image. *See* Kulkarni at [0038].

As yet another example, Stahl discloses a read-only repository file system 226 and a shared read-only file system that provides data to multiple computers. It further describes an overlay file system 220, which stores copies of files that have been modified by respective computers. Stahl also discloses a repository FS instance 224 and MapFS instance 216, which serve as a cache configured to store transient, in-memory versions of cached data blocks previously accessed by compute nodes, enabling efficient data access. *See* Stahl at [0026]-[0027].

As yet another example, Zhang describes a data module that stores primary data objects on a computer storage medium. It further discloses replication and placement of data objects in a distributed system, where replicas are created that may include only modified or additional data blocks specific to the respective compute nodes. Zhang at [0034], [0038]. Additionally, Zhang describes an index module on compute nodes that builds and tracks indices of data objects and shares this information with other nodes, enabling efficient access and management of data across the distributed system. *See* Zhang at [0008], [0047].

As yet another example, Kiselev discloses a system for maintaining consistent and up-to-date copies of primary data across a primary node and a secondary node. The primary node stores the primary data, while the secondary node holds snapshots that capture changes made to the primary data using overlay objects. These overlay objects store only the modified data blocks, referencing unchanged blocks from the primary volume. *See* Kiselev at Abstract, 5:26-63; 6:28-63.

As yet another example, Takeda discloses a data processing system comprising a primary storage unit and multiple secondary storage units. The primary storage unit (PVOL 2212) serves as the main data volume, storing data blocks used for updates and journal acquisition. The secondary storage units (SVOL 2214) store journal data and metadata derived from updates made to the primary storage unit. The journal data includes information about updates, such as addresses, timestamps, and content changes. The secondary storage unit (SVOL 2214) does not duplicate unchanged data blocks from the primary storage unit, as journal data represents only changes and additions. Additionally, the storage controller (210) incorporates a cache memory (212) to enhance I/O performance by temporarily storing data blocks. *See* Takeda at [0035]-[0038].

As yet another example, Menon discloses a shared disk system where multiple nodes, each equipped with a computer and an adaptor, interface with shared storage devices. The shared storage devices (10a, b, c) act as the primary repository storing data blocks used by compute nodes. Each adaptor includes a write cache (20a, b, c), which temporarily holds updates or modifications made by individual compute nodes. Additionally, the adaptors feature read caches (18a, b, c) that temporarily store recently accessed data blocks, optimizing data retrieval and enhancing I/O performance for compute nodes. *See* Menon at 3:55-4:61.

As yet another example, Kodama discloses a system where a primary storage unit stores a master copy of data. Kodama explains that replicated volumes store copies of the data from the master volume. The system handles changes to the data, such as modifications and deletions, by writing them to previously unused areas of the storage, with the updated data being replicated to the replicated volumes. *See* Kodama at [0007], [0050]-[0053].

As yet another example, the NetApp system implemented at Southwest Airlines in 2002 exemplifies a robust and efficient method for managing root and leaf image data. The system architecture at Southwest leveraged clustered NetApp filers integrated with Oracle9i RAC to achieve high availability and seamless data management. The key technologies used in this setup included WAFL (Write Anywhere File Layout), SnapMirror, and SnapRestore. These technologies facilitated the management of root and leaf image data, maintained consistent application environments, and supported rapid recovery after system disruptions.

As yet another example, the VMware Systems, introduced in the early 2000s, enabled multiple virtual machines (compute nodes) to run concurrently on a single physical server. The virtual machines share data through root image files stored on centralized storage units and manage data changes via snapshots and redo logs. The systems efficiently provide data to a plurality of compute nodes through virtualization and dynamic resource allocation. *See* VMware GSX Server 3 Virtual Machine Guide, VMware GSX Server 3 Specifications Datasheet, and VMware ESX Server 2.1 Specifications Datasheet.

Various references identified above also disclose, well before the critical date of the ‘844 Patent, caching of data blocks. For example, Holzmann is directed to a system and method for booting multiple servers or other external devices from a storage area network using a single OS image. *See* Holzmann, [0017]. When a boot request is received from an external device, the control

module 103 allocates cache space in the volatile cache memory portion 104 of the disk. The cache is used in the typical manner to optimize future access to data once it has demonstrated some likelihood of being further referenced. As additional servers or other network resources are booted, the cache is used to speed access to the most commonly used data. *Id.* at Abstract, [0021], [0024].

As another example, Nelson discloses a method and system for a caching file server. It stores the data in the cache memory accessed by the first client. A second client does not need to access the data directly from the storage device, the data can be directly accessed by the cache memory. *See* Nelson at [0038]-[0039].

A person of ordinary skill in the art would have been motivated to combine the above references to incorporate the different beneficial technical features provided by each of the above references. Modifying the above references would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results.

* * *

In addition to the specific combinations of prior art and the specific combinations of groups of prior art disclosed above, Southwest reserves the right to rely on any other combination of any prior art references disclosed herein. Southwest further reserves the right to rely upon combinations of prior art references disclosed within the prosecution history of the references cited herein.

The references charted in Exhibits 844-1 – 844-24 disclose the elements of the asserted claims of the '844 Patent explicitly and/or inherently and may also be relied upon to show the state of the art at the time of the alleged invention. The suggested obviousness combinations are provided in the alternative to Southwest's anticipation contentions and are not to be construed to suggest that any reference included in the combinations is not itself anticipatory.

C. OGP: Charts Setting Forth Where in the Prior Art References Each Element of the Asserted Claims Are Found

Pursuant to the Court's OGP, charts setting forth where specifically in each alleged item of prior art each element of each Asserted Claim of the '844 Patent is found are attached as Exhibits 844-1 – 844-24.

D. Other Relevant Prior Art References

In addition to the references identified above and charted in Exhibits 844-1 – 844-24, each of the references below is relevant to the '844 Patent, as it relates to the state of the art. Southwest reserves the right to amend its contentions to rely on any of the below references alone or in combination with any other references identified in these contentions.

Inventor/Author	Country	Issue/Publication No.	Filed Date
C. K. Haun et al.	US	2004/0153526	8/5/2004
James E. King et al.	US	7,529,807	5/5/2004
Yevgeniy Klimenko et al.	US	5,974,547	3/20/1998
Beng-Hong Lim	US	6,795,966	2/4/2000
Julian S. Taylor	US	5,930,513	6/6/1996
Tung M. Nguyen	US	7,870,106	2/2/2006
Pradip Kulkarni	US	7,721,282	5/18/2010
Rauch, et al., "Partition Repositories for Partition Cloning — OS Independent Software Maintenance in Large Clusters of PCs," Proceedings of the IEEE International Conference on Cluster Computing (CLUSTER 2000), DOI: 10.1109/CLUSTR.2000.889066			Pub. November 28, 2000.
Goscinski, et al., "GENESIS: An Efficient, Transparent and Easy to Use Cluster Operating System," *Parallel Computing*, Vol. 28, Issue 4, Pages 557–606			Pub. at least as early as April 2002

V. THE '722 PATENT IS INVALID UNDER 35 U.S.C. §§ 102 AND/OR 103

A. The '722 Patent Priority Date

Application No. 11/396,251 (the '251 Application), which matured into the '722 Patent, was filed on March 30, 2006 and was a continuation of Application No. 10/105,018 (filed March 21, 2002) (now Patent No. 7,051,070) Application No. 10/105,018 is a continuation-in-part of Application No. 10/017,182 (filed Dec. 14, 2001) (now Patent No. 7,043,525).

The '722 Patent further claims the benefit of several provisional applications: Provisional Application No. 60/256,613 (filed Dec. 18, 2000); Provisional Application No. 60/276,847 (filed March 16, 2001); Provisional Application No. 60/278,303 (filed March 21, 2001); Provisional Application No. 60/279,608 (filed March 27, 2001); and Provisional Application No. 60/280,627 (filed March 29, 2001).

Plaintiff's Infringement Contentions allege that the asserted claims of the '722 Patent are entitled to a priority date of December 18, 2000, which is the filing date of the '613 Provisional. However, only one of the provisional applications, the '303 Provisional, filed on March 21, 2001, describes message types and node categories, which are claimed elements in the '722 Patent. Therefore, the asserted claims of the '722 Patent are only entitled to a filing date of March 21, 2001.

B. Identification of Prior Art that Anticipates or Renders Obvious the Asserted Claims of the '722 Patent

Southwest contends that the prior art references listed below and charted in Exhibits 722-1 – 722-11 anticipate and/or render obvious, alone or in combination, one or more of the asserted claims of the '722 Patent.

Prior Art Reference	Publication Date
U.S. Patent No. 5,721,825 ("Lawson")	Filed October 3, 1996 Issued – February 24, 1998 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b) and § 102(e).
U.S. Patent No. 6,480,883 ("Tsutsumitake")	Filed – June 29, 1999 Issued – November 12, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
PCT Patent Application Publication No. WO 99/26127 ("Kostes")	Filed – November 13, 1998 Published – May 27, 1999 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b) and § 102(e).
U.S. Patent No. 6,999,991 ("Ikeda")	Filed – July 24, 2000 Issued – February 14, 2006 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent No. 6,990,591 ("Pearson")	Filed – December 22, 1999 Issued – January 24, 2006 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
European Patent Application Publication No. EP1043671 ("Bird")	Filed – March 17, 2000 Published – October 11, 2000 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(a).
U.S. Patent No. 6,721,740 ("Skinner")	Filed – June 5, 1998 Issued – April 13, 2004 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
United States Patent No. 6,710,702 ("Averbuch")	Filed – November 22, 1999 Issued – March 23, 2004

Prior Art Reference	Publication Date
	This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
United States Patent No. 6,763,384 ("Gupta")	Filed – July 10, 2000 Issued – July 13, 2004 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
United States Patent No. 6,859,829 ("Parupudi")	Filed – February 23, 1999 Issued – February 22, 2005 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
United States Patent No. 6,216,132 ("Chandra")	Filed – November 20, 1997 Issued – April 10, 2001 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
United States Patent No. 6,910,070 ("Mishra")	Filed – January 24, 2000 Issued – June 21, 2005 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
United States Patent No. 6,823,359 ("Heidingsfeld")	Filed – November 21, 2000 Issued – November 23, 2004 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
United States Patent No. 6,757,283 ("Yamanaka")	Filed – January 24, 2000 Issued – June 29, 2004 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
United States Patent No. 7,051,084 ("Hayton")	Filed – November 2, 2000 Issued – May 23, 2006 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).

Prior Art Reference	Publication Date
United States Patent No. 6,633,910 ("Rajan")	Filed – December 14, 1999 Issued – October 14, 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
United States Patent No. 9,300,560 ("Leighton")	Filed – March 4, 2013 Priority - July 20, 2000 Issued – March 29, 2016 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
PointCast System – PointCast Incorporated	At least as early as 1997 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a) or (g).
Castanet System – Marimba, Inc.	At least as early as 1996 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a) or (g).
Active Desktop System – Microsoft Corporation	At least as early as 1998 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a) or (g).

Plaintiffs assert claims 14, 16, 17, and 18 of the '722 Patent against Southwest in this lawsuit. These claims are invalid because the '722 Patent fails to meet one or more of the requirements for patentability. The individual bases for invalidity are provided below and in Exhibits 722-1 - 722-11. Each of the prior art references, the underlying work, and/or the underlying apparatus or method qualifies as prior art under one or more sections of Pre-AIA 35 U.S.C. §§ 102 and/or 103.

In an effort to focus the issues, Southwest has cited representative portions of identified references, even where a reference may contain additional disclosures or support for a particular claim element. Each and every disclosure of the same limitation in the same reference is not

necessarily identified. In addition, persons of ordinary skill in the art generally read a prior art reference as a whole and in the context of other publications and literature and in the context of their knowledge as persons of ordinary skill in the art. Thus, to understand and interpret any specific statement or disclosure within a prior art reference, such persons would rely on other information within the reference, along with other publications and their general scientific knowledge and experience in the field. Thus, each prior art reference identified herein should be understood in the context of the reference itself in conjunction with anything incorporated by reference or described as relevant technology by the reference, any documents that are referred to for further understanding by the reference, any product embodying the reference, and any testimony by those with knowledge of the reference, such as the named authors and inventors. Southwest may rely upon uncited portions of the prior art references and on other publications and expert testimony to provide context, and as aids to understanding and interpreting the portions that are cited. Southwest may also rely on uncited portions of the prior art references, other disclosed publications, and the testimony of experts to establish that a person of ordinary skill in the art would have been motivated to modify or combine certain of the cited references so as to render the claims obvious.

Some or all of the asserted claims of the ‘722 Patent are invalid as anticipated under Pre-AIA 35 U.S.C. § 102 in view of certain ones of the prior art references identified above as detailed in the claim charts included in Exhibits 722-1 - 722-11, which identify specific examples of where each limitation of the asserted claims of the ‘722 Patent is found in the prior art references. As explained above, the cited portions of prior art references identified in the attached claim charts are exemplary in nature and representative of the content and teaching of the prior art references,

and should be understood in the context of the reference as a whole and as they would be understood by a person of ordinary skill in the art.

To the extent any item of prior art cited above is deemed not to disclose, explicitly or inherently, any limitation of an asserted claim of the '722 Patent, Southwest reserves the right to argue that any difference between that prior art and the corresponding patent claim would have been either inherent in the art or obvious to a person of ordinary skill in the art, even if Southwest has not specifically denoted that the art is to be combined with the knowledge of a person of ordinary skill in the art.

To the extent Plaintiffs argue that any limitation of the asserted claims of the '722 Patent is not explicitly disclosed by an item of prior art identified above and in Exhibits 722-1 - 722-11, any purported differences are such that the claimed subject matter as a whole would have been obvious to one skilled in the art at the time of the alleged invention, in view of the state of the art and knowledge of those skilled in the art. The item of prior art would, therefore, render the relevant claims invalid as obvious under Pre-AIA 35 U.S.C. § 103.

To the extent any of the foregoing references are found not to anticipate the asserted claims of the '722 Patent, the foregoing references render the asserted claims obvious, either alone or in combination with one or more of the other references identified above. One of ordinary skill in the art would have been motivated to combine the various references cited herein so as to disclose each and every limitation of the asserted claims of the '722 Patent.

The prior art identified in Exhibits 722-1 - 722-11 renders one or more asserted claims of the '722 Patent obvious when the references are read alone or in combination with one or more of each other and/or when read in view of the state of the art and knowledge of those skilled in the art.

To the extent that Plaintiff argues that any asserted claim of the '722 Patent is not disclosed or rendered obvious by any reference charted in Exhibits 722-1 - 722-11, the asserted claims nevertheless would have been obvious in light of one or more other references charted in Exhibits 722-1 - 722-11 and in view of the knowledge, experience, and understanding of persons of ordinary skill in the art.

Each and every reference identified is also relevant to the state of the art at the time of the alleged invention. Any of the references disclosed above may be combined with one another to render obvious (and therefore invalid) each of the asserted claims of the '722 Patent. Southwest may rely upon a subset of the above identified references, or all of the references identified above, including all references in charted in Exhibits 722-1 - 722-11, for purposes of obviousness depending on the Court's claim construction, positions taken by Plaintiffs during this litigation, and further investigation and discovery.

1. Example Combinations of the Prior Art

The table below provides an identification of exemplary combinations of prior art references that, based on Southwest's present understanding of Plaintiff's Infringement Contentions and apparent reading of the claims, render obvious one or more of the asserted claims of the '722 Patent. The combinations of references identified below are non-exhaustive, and additional examples are apparent from other combinations of references charted in Exhibits 722-1 - 722-11. Although certain references charted in Exhibits 722-1 - 722-11 are listed variously in the left or right column below, Southwest does not thereby imply any restriction as to which reference is primary and which reference is secondary in such a combination. Southwest reserves the right to rely on either identified reference in the identified combinations as a primary and/or secondary reference.

Exhibit	Reference	Combination Reference(s)
722-1	Lawson	Averbuch, Chandra, Mishra, Yamanaka, Hayton, Rajan, Tsutsumitake, Bird, Pearson, PointCast System, Castanet System, and/or Active Desktop System, and/or one or more references as charted in Exhibits 722-1 - 722-11
722-2	Kostes	Averbuch, Chandra, Mishra, Yamanaka, Hayton, Rajan, Tsutsumitake, Bird, Pearson, PointCast System, Castanet System, and/or Active Desktop System, and/or one or more references as charted in Exhibits 722-1 - 722-11
722-3	Ikeda	Averbuch, Chandra, Mishra, Yamanaka, Hayton, Rajan, Tsutsumitake, Bird, Pearson, PointCast System, Castanet System, and/or Active Desktop System, and/or one or more references as charted in Exhibits 722-1 - 722-11
722-4	Skinner	Averbuch, Chandra, Mishra, Yamanaka, Hayton, Rajan, Tsutsumitake, Bird, and/or Pearson, and/or one or more references as charted in Exhibits 722-1 - 722-11
722-5	Gupta	Averbuch, Chandra, Mishra, Yamanaka, Hayton, Rajan, Tsutsumitake, Bird, and/or Pearson, and/or one or more references as charted in Exhibits 722-1 - 722-11
722-6	Parapudi	Averbuch, Chandra, Mishra, Yamanaka, Hayton, Rajan, Tsutsumitake, Bird, and/or Pearson, and/or one or more references as charted in Exhibits 722-1 - 722-11
722-7	Heidingsfeld	Averbuch, Chandra, Mishra, Yamanaka, Hayton, Rajan, Tsutsumitake, Bird, and/or Pearson, and/or one or more references as charted in Exhibits 722-1 - 722-11
722-8	Leighton	Averbuch, Chandra, Mishra, Yamanaka, Hayton, Rajan, Tsutsumitake, Bird, and/or

Exhibit	Reference	Combination Reference(s)
		Pearson, and/or one or more references as charted in Exhibits 722-1 - 722-11
722-9	PointCast System	Lawson, Kostes, Ikeda, Averbuch, Chandra, Mishra, Yamanaka, Hayton, Rajan, Tsutsumitake, Bird, and/or Pearson, and/or one or more references as charted in Exhibits 722-1 - 722-11
722-10	Castanet System	Lawson, Kostes, Ikeda, Averbuch, Chandra, Mishra, Yamanaka, Hayton, Rajan, Tsutsumitake, Bird, and/or Pearson, and/or one or more references as charted in Exhibits 722-1 - 722-11
722-11	Active Desktop System	Lawson, Kostes, Ikeda, Averbuch, Chandra, Mishra, Yamanaka, Hayton, Rajan, Tsutsumitake, Bird, and/or Pearson, and/or one or more references as charted in Exhibits 722-1 - 722-11

It will be understood that, given the similarities between the art, any of Lawson, Kostes, Ikeda, Skinner, Gupta, Parapudi, Heidingsfeld, Averbuch, Chandra, Mishra, Yamanaka, Hayton, Rajan, Tsutsumitake, Bird, Pearson, Leighton, Castanet System, PointCast System, and/or Active Desktop System could be combined with one or more of each other, for example based on the disclosures shown in Exhibits 722-1 - 722-11. It would have been obvious to combine these references for reasons that are readily apparent to one of ordinary skill in the art, and for reasons explained in additional detail in the sections that follow.

2. The Claims Recite Well-Known Concepts that Would Have Been Obvious to Implement

As noted above, the asserted claims of the '722 Patent represent nothing more than an arrangement of well-known concepts and features and add nothing to the state of the art at the time of the alleged invention. For each of the concepts represented by the claims, it would have been

inherently obvious to add to the teachings of any number of known references and products, including to any of the primary references charted in Exhibits 722-1 - 722-11 to the extent that such reference is alleged to be missing such concept or feature.

3. Additional Reasons to Combine

In addition to the information contained elsewhere in these invalidity contentions, including in Exhibits 722-1 - 722-11, Southwest has identified additional motivations and reasons to combine the various references. Southwest notes that in *KSR Int'l Co. v. Teleflex Inc.*, the Supreme Court held that "[i]n determining whether the subject matter of a patent claim is obvious, neither the particular motivation nor the avowed purpose of the patentee controls. What matters is the objective reach of the claim. If the claim extends to what is obvious, it is invalid under § 103." 550 U.S. 398, 127 S.Ct. 1727, 1742 (2007).

KSR further illustrated several ways in which the subject matter of a patent claim may be shown to be obvious. For instance, "[o]ne of the ways in which a patent's subject matter can be proved obvious is by noting that there existed at the time of invention a known problem for which there was an obvious solution encompassed by the patent's claims." *KSR*, 127 S.Ct. 1727, 1742. The Supreme Court held that it was error to "look only to the problem the patentee was trying to solve." *Id*. Rather, "[t]he question is not whether the combination was obvious to the patentee but whether the combination was obvious to a person with ordinary skill in the art. Under the correct analysis, any need or problem known in the field of endeavor at the time of invention and addressed by the patent can provide a reason for combining the elements in the manner claimed." *Id*. Further, "[c]ommon sense teaches … that familiar items may have obvious uses beyond their primary purposes, and in many cases a person of ordinary skill will be able to fit the teachings of multiple patents together like pieces of a puzzle." *Id.* Further, a showing that a combination of elements was "obvious to try" may show that it was obvious under § 103. *Id.* For instance, the Supreme

Court held that "predictable solutions" using a combination of "known options" may render the subject matter of a patent claim obvious.

When there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense. *Id.* The ways in which the subject matter of a patent claim may be shown to be obvious, identified by *KSR*, are merely illustrative. The main thrust of *KSR* was that "[r]igid preventative rules that deny factfinders recourse to common sense"—such as the overturned "teaching, suggestion, or motivation" test—are neither necessary under our case law nor consistent with it." *KSR*, 127 S.Ct. 1727, 1742-43; citing with approval, e.g., *DyStar Textilfarben GmbH & Co. Deutschland KG v. C.H. Patrick Co.*, 464 F. 3d 1356, 1367 (2006) ("Our suggestion test is in actuality quite flexible and not only permits, but requires, consideration of common knowledge and common sense"); *Alza Corp. v. Mylan Labs., Inc.*, 464 F. 3d 1286, 1291 (2006) ("There is flexibility in our obviousness jurisprudence because a motivation may be found implicitly in the prior art. We do not have a rigid test that requires an actual teaching to combine...").

Although Southwest notes that there is no longer a rigid requirement regarding motivation to combine under *KSR*, Southwest provides the following statements regarding motivation to combine. Multiple teachings, suggestions, and/or reasons to modify any of the references and/or to combine any two or more of the references charted in Exhibits 722-1 - 722-11 come from many sources, including the prior art (specific and as a whole), common knowledge, common sense, predictability, expectations, industry trends, design incentives or need, market demand or pressure, market forces, obvious to try, the nature of the problem faced, and/or knowledge possessed by one

of ordinary skill. In addition, it would have been obvious to try combining the prior art references identified above because there were only a finite number of predictable solutions and/or because known work in one field or endeavor prompted variations based on predictable design incentives and/or market forces either in the same field or a different one.

The combination of prior art references identified in these contentions would have been obvious because the combination represents the known potential options with a reasonable expectation of success. Additionally, one of ordinary skill in the art would have been motivated to create combinations identified in these contentions using: known methods to yield predictable results; known techniques in the same way; a simple substitution of one known, equivalent element for another to obtain predictable results; and/or teaching, suggestion, or motivation in the prior art generally. Also, market forces in the industry, and the desire to improve features and performance, would motivate the addition of features to products as they become available, become less expensive, become more commonly used, provide better performance, reduce costs, or predictably achieve other clearly desirable results.

Moreover, in addition to the information contained elsewhere in these contentions, and, in particular, charted in Exhibits 722-1 - 722-11, Southwest has identified additional motivations and reasons to combine the various references cited herein. In particular, multiple teachings, suggestions, and/or reasons to modify any of the references and/or to combine any two or more of the references charted in Exhibits 722-1 - 722-11 come from many sources, including the prior art (specific and as a whole), common knowledge, common sense, predictability, expectations, industry trends, design incentives or need, market demand or pressure, market forces, obvious to try, the nature of the problem faced, and/or knowledge possessed by one of ordinary skill. In addition, it would have been obvious to try combining the prior art references identified above

because there were only a finite number of predictable solutions and/or because known work in one field or endeavor prompted variations based on predictable design incentives and/or market forces either in the same field or a different one. The combination of prior art references identified in these contentions would have been obvious because the claimed combinations represent the known potential options with a reasonable expectation of success.

To the extent that Plaintiffs contend that any of the above-identified references does not expressly or inherently disclose a particular element or limitation, it would have been obvious to a person of ordinary skill in the art to modify the references or to combine their teachings with other prior art references, including the prior art references in these invalidity contentions, in a manner that renders such claims obvious and invalid.

In particular, various references identified above disclose, well before the critical date of the '722 Patent, systems for providing live objects and routing updates to live objects via various nodes based on message and node types to clients registered to receive updates to the live objects.

For example, Ikeda discloses a push service system that routes data to registered client devices through various agents (nodes) depending on the type of node. See Ikeda at Abstract, 19:48-20:20, 20:28-47.

As another example, Lawson discloses "[a] system [and] method for global event notification in a distributed computer environment" in which "a local event registry [is used] to identify local event consumers that should be notified when an event occurs" and "[A] global event registry [] identifies other servers which need notification when an event occurs." These other servers "will then, in turn, notify their local event consumers of the event." *See* Lawson at Abstract.

As yet another example, Kostes, discloses "a system and method for displaying, in near real-time, data published from various computer systems and networks, such as the Internet." 1:22-

24. The system is "web-enabled, providing tight integration with all popular [web] browsers." 30:17-20. In Kostes's "publish/subscribe architecture [(aka ''model-view' architecture')] ... various information sources [(e.g., different sources such as Sportsticker, CNN, Reuters, etc.)], most likely under a contractual agreement with [Kostes's] push server, constantly publish and broadcast data to the push server, using any desired communication method," and a "client computer subscribes to desired types of information available from the push server." 7:12-18, 16:10-13, 2:11-20, 3:9-14.

As yet another example, Tsutsumitake discloses "[a] real-time information transmission system [and method that] can immediately transmit information updated on a server to a client" in a web-enabled environment. *See* Tsutsumitake at Abstract; 1:6-32. A web "client analyzes [a web]page. If an event request is included in the analyzed page ... [a] connection ID [is used by] the server [to] send[] an event to the client each time the event has occurred. The client processes the event and reflects the processed result on a screen of a display" to, e.g., update in real time powerplant data 71 and 72 being monitored via the web. *See* Tsutsumitake at Abstract, 13:31-65.

As yet another example, the PointCast System provided a push/subscribe service in which live webpage objects such as news, financial data, and other content was dynamically updated within a graphical UI. *See* https://web.archive.org/web/19990826121133/http://www.pointcast.com/tour/index.html?hometxt. The PointCast server acted as a gateway, receiving news or stock updates from external sources and associating them with specific content channels. The PointCast server/gateway identified the category of update message and then sent the update message to the appropriate node type. *See* https://www.pcmag.com/encyclopedia/term/pointcast;

https://web.archive.org/web/19990921040538/http://www.pointcast.com/content/index.html?hometxt.

As yet another example, the Castanet System, introduced in 1996, was a family of Java-based delivery systems for publishing and automatically distributing application updates and other published materials via the Internet and intranets." *See* https://www.pcmag.com/encyclopedia/term/marimba.

As yet another example, the Active Desktop System is a Microsoft Windows feature since Windows 95 that allows users to display interactive content, such as web pages, as their desktop wallpaper. *See* https://gustavus.edu/gts/Active_Desktop. Active desktop "provid[ed] a dynamic and personalized computing experience, transforming your static desktop into an interactive hub. By embedding live web content like news feeds or weather updates directly onto your background, it keeps you informed without the need to open multiple applications. This feature enhances convenience and productivity, offering real-time information briefly. Tailoring your desktop with active desktop ensures a workspace that aligns with your preferences, making it a valuable tool for those seeking a more engaging and customized digital environment." *See* https://www.lenovo.com/us/en/glossary/active-desktops/.

Various references identified above also disclose, well before the critical date of the '722 Patent, identifying a message category based on an input source. For example, Bird discloses a publish/subscribe system in which messages are identified according to the sender. *See* Bird at Abstract, [0044]. As another example, Pearson discloses including source identifying information in message headers such as port numbers. *See* Pearson at 18:41-19:18.

A person of ordinary skill in the art would have been motivated to combine the above references to incorporate the different beneficial technical features provided by each of the above

references. Modifying the above references would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results.

* * *

In addition to the specific combinations of prior art and the specific combinations of groups of prior art disclosed above, Southwest reserves the right to rely on any other combination of any prior art references disclosed herein. Southwest further reserves the right to rely upon combinations of prior art references disclosed within the prosecution history of the references cited herein.

The references charted in Exhibits 722-1 - 722-11 disclose the elements of the asserted claims of the '722 Patent explicitly and/or inherently and may also be relied upon to show the state of the art at the time of the alleged invention. The suggested obviousness combinations are provided in the alternative to Southwest's anticipation contentions and are not to be construed to suggest that any reference included in the combinations is not itself anticipatory.

C. OGP: Charts Setting Forth Where in the Prior Art References Each Element of the Asserted Claims Are Found

Pursuant to the Court's OGP, charts setting forth where specifically in each alleged item of prior art each element of each Asserted Claim of the '722 Patent is found are attached as Exhibits 722-1 - 722-11.

D. Additional Prior Art

In addition to the references identified above and charted in Exhibits 722-1 - 722-11, each of the references below is relevant to the '722 Patent, as it relates to the state of the art. Southwest reserves the right to amend its contentions to rely on any of the below references alone or in combination with any other references identified in these contentions.

Inventor/Author	Country	Issue/Publication No.	Filed Date
Robert W. Baynes Jr. et al.	US	2002/0032722	9/12/2001
Simon Antony James Holdsworth et al.	US	7,103,680	4/25/2000
Andrew John Mark Miles et al.	US	6,886,044	4/26/2005
Mark A. Fox et al.	US	6,654,786	8/30/2000
Richard Lee Schwartz et al.	US	5,913,032	9/30/1996
Hemant Madan	US	7,516,196	3/21/2000
Christopher C. Andrews	US	2010/0325562	3/1/2000
Stephen James Paul Todd	US	6,643,682	2/22/2000
Hirokawa Masaru et al.	JP	JPH11353216A	06/08/1998; Pub. 12/24/1999
James P. Richardson et al.	US	6,292,803	11/18/1997
Colin Leonard Bird et al.	UK	GB 2 348 025	3/19/1999; Pub 9/20/2000
M. Hauswirth and M. Jazayeri, "A Component and Communication Model for Push Systems," Proceedings of the ESEC/FSE 99 - Joint 7th European Software Engineering Conference ("Hauswirth")			Pub. at least as early as September 6, 1999.
G. S. Barnes Nelson, "Messaging Systems: SAS® Tools for Internet-Based Communications," Proceedings of the 25th Annual SAS Users Group International Conference (SUGI 25), Paper 288-25 ("Barnes Nelson")			Pub. at least as early as March 19, 1999.
Segall, et al., "Content Based Routing with Elvin4," Proceedings of AUUG2K (No. 39) ("Segall")			Pub. at least as early as June 2000.

VI. THE '785 PATENT IS INVALID UNDER 35 U.S.C. §§ 102 AND/OR 103

A. The '785 Patent Priority Date

Plaintiff's Infringement Contentions allege the asserted claims of the '785 Patent have a filing date of March 21, 2003. However, this appears to be a typographical error. The '785 Patent has a filing date of March 31, 2003. The '785 Patent includes no priority claim.

B. Identification of Prior Art that Anticipates or Renders Obvious the Asserted Claims of the '785 Patent

Southwest contends that the prior art references listed below and charted in Exhibits 785-1 – 785-29 anticipate and/or render obvious, alone or in combination, one or more of the asserted claims of the '785 Patent.

Prior Art Reference	Publication Date
U.S. Patent No. 6,970,941 ("Caronni I")	Filed – December 10, 1999 Issued – November 29, 2005 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent No. 7,814,228 ("Caronni II")	Filed – February 13, 2003 Issued – October 12, 2010 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent No. 6,766,371 ("Hipp")	Filed – October 5, 2000 Issued – July 20, 2004 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No. 2003/0028671 ("Mehta")	Filed – June 10, 2002 Published – February 6, 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
Huitema, C., Network Working Group Request for Comment (RFC): 1383, entitled An Experiment in DNS Based IP Routing ("RFC 1383")	Published December 1992 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b).
U.S. Patent No. 7,133,404 ("Alkhatib")	Filed – August 10, 2001 Issued – November 7, 2006 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No.	Filed – April 19, 2002 Published – December 5, 2002

Prior Art Reference	Publication Date
2002/0184529 (“Foster I”)	This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No. 2003/0208602 (“Bhalla”)	Filed – April 8, 2002 Published – November 6, 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No. 2002/0181395 (“Foster II”)	Filed – April 19, 2002 Published – December 5, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No. 2002/0091859 (“Tuomenoksa”)	Filed – April 11, 2001 Published – July 11, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No. 2004/0177136 (“Chen”)	Filed – March 3, 2003 Published – September 9, 2004 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
European Patent Publication No. EP1098496 (“Engmann”)	Published – October 19, 2000 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b).
U.S. Patent Application Publication No. 2002/0069278 (“Forslow”)	Filed – December 5, 2000 Published – June 6, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent No. 6,701,437 (“Hoke”)	Filed – November 9, 1998 Issued – March 2, 2004 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No. 2002/0116502 (“Iyer”)	Filed – February 22, 2001 Published – August 22, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).

Prior Art Reference	Publication Date
U.S. Patent Application Publication No. 2003/0065785 ("Jain")	Filed – September 28, 2001 Published – April 3, 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No. 2002/0035624 ("Kim")	Filed – July 9, 2001 Published – March 21, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(b) and (e).
U.S. Patent Application Publication No. 20020154635 ("Liu")	Filed – April 23, 2001 Published – October 24, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent No. 8,068,817 ("Viswanath")	Filed – August 27, 2002 Issued – November 29, 2011 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No. 2002/0116523 ("Warrier")	Filed – February 22, 2001 Published – August 22, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No. 2003/0229697 ("Borella")	Filed – June 10, 2002 Published – December 11, 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No. 2003/0084162 ("Johnson")	Filed – October 31, 2001 Published – May 1, 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
PCT Patent Application Publication No. WO2003088625 ("Volz")	Filed – March 28, 2003 Priority – April 8, 2002 Published – October 23, 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No.	Filed – November 20, 2002 Published – May 20, 2004

Prior Art Reference	Publication Date
2004/0098507 ("Thubert")	This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No. 2003/0016636 ("Tari")	Filed – July 15, 2002 Published – January 23, 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent No. 8,107,483 ("Dunk")	Filed – March 12, 2010 Priority – November 27, 2002 Issued – January 31, 2012 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
PCT Patent Application Publication No. WO2002015014 ("Wootton")	Filed – July 31, 2001 Published – February 21, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(b) and (e).
U.S. Patent Application Publication No. 2002/0038382 ("Ryu")	Filed – August 30, 2001 Published – March 28, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(b) and (e).
U.S. Patent Application Publication No. 2003/0172184 ("Kong")	Filed – March 6, 2003 Published – September 11, 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No. 2004/0037260 ("Kakemizu")	Filed – August 7, 2003 Published – February 26, 2004 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
Cisco VPN Routers and Software Systems – Cisco Systems	At least as early as 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a) and (g).
VMware Systems - VMware	At least as early as 2001 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a) and (g).

Plaintiffs assert claims 30, 35, and 37 of the '785 Patent against Southwest in this lawsuit. These claims are invalid because the '785 Patent fails to meet one or more of the requirements for patentability. The individual bases for invalidity are provided below and in Exhibits 785-1 – 785-29. Each of the prior art references, the underlying work, and/or the underlying apparatus or method qualifies as prior art under one or more sections of Pre-AIA 35 U.S.C. §§ 102 and/or 103.

In an effort to focus the issues, Southwest has cited representative portions of identified references, even where a reference may contain additional disclosures or support for a particular claim element. Each and every disclosure of the same limitation in the same reference is not necessarily identified. In addition, persons of ordinary skill in the art generally read a prior art reference as a whole and in the context of other publications and literature and in the context of their knowledge as persons of ordinary skill in the art. Thus, to understand and interpret any specific statement or disclosure within a prior art reference, such persons would rely on other information within the reference, along with other publications and their general scientific knowledge and experience in the field. Thus, each prior art reference identified herein should be understood in the context of the reference itself in conjunction with anything incorporated by reference or described as relevant technology by the reference, any documents that are referred to for further understanding by the reference, any product embodying the reference, and any testimony by those with knowledge of the reference, such as the named authors and inventors. Southwest may rely upon uncited portions of the prior art references and on other publications and expert testimony to provide context, and as aids to understanding and interpreting the portions that are cited. Southwest may also rely on uncited portions of the prior art references, other disclosed publications, and the testimony of experts to establish that a person of ordinary skill in the art

would have been motivated to modify or combine certain of the cited references so as to render the claims obvious.

Some or all of the asserted claims of the '785 Patent are invalid as anticipated under Pre-AIA 35 U.S.C. § 102 in view of certain ones of the prior art references identified above as detailed in the claim charts included in Exhibits 785-1 – 785-29, which identify specific examples of where each limitation of the asserted claims of the '785 Patent is found in the prior art references. As explained above, the cited portions of prior art references identified in the attached claim charts are exemplary in nature and representative of the content and teaching of the prior art references, and should be understood in the context of the reference as a whole and as they would be understood by a person of ordinary skill in the art.

To the extent any item of prior art cited above is deemed not to disclose, explicitly or inherently, any limitation of an asserted claim of the '785 Patent, Southwest reserves the right to argue that any difference between that prior art and the corresponding patent claim would have been either inherent in the art or obvious to a person of ordinary skill in the art, even if Southwest has not specifically denoted that the art is to be combined with the knowledge of a person of ordinary skill in the art.

To the extent Plaintiffs argue that any limitation of the asserted claims of the '785 Patent is not explicitly disclosed by an item of prior art identified above and in Exhibits 785-1 – 785-29, any purported differences are such that the claimed subject matter as a whole would have been obvious to one skilled in the art at the time of the alleged invention, in view of the state of the art and knowledge of those skilled in the art. The item of prior art would, therefore, render the relevant claims invalid as obvious under Pre-AIA 35 U.S.C. § 103.

To the extent any of the foregoing references are found not to anticipate the asserted claims of the '785 Patent, the foregoing references render the asserted claims obvious, either alone or in combination with one or more of the other references identified above. One of ordinary skill in the art would have been motivated to combine the various references cited herein so as to disclose each and every limitation of the asserted claims of the '785 Patent.

The prior art identified in Exhibits 785-1 – 785-29 renders one or more asserted claims of the '785 Patent obvious when the references are read alone or in combination with one or more of each other and/or when read in view of the state of the art and knowledge of those skilled in the art.

To the extent that Plaintiff argues that any asserted claim of the '785 Patent is not disclosed or rendered obvious by any reference charted in Exhibits 785-1 – 785-29, the asserted claims nevertheless would have been obvious in light of one or more other references charted in Exhibits 785-1 – 785-29 and in view of the knowledge, experience, and understanding of persons of ordinary skill in the art.

Each and every reference identified is also relevant to the state of the art at the time of the alleged invention. Any of the references disclosed above may be combined with one another to render obvious (and therefore invalid) each of the asserted claims of the '785 Patent. Southwest may rely upon a subset of the above identified references, or all of the references identified above, including all references in charted in Exhibits 785-1 – 785-29, for purposes of obviousness depending on the Court's claim construction, positions taken by Plaintiffs during this litigation, and further investigation and discovery.

1. Example Combinations of the Prior Art

The table below provides an identification of exemplary combinations of prior art references that, based on Southwest's present understanding of Plaintiff's Infringement

Contentions and apparent reading of the claims, render obvious one or more of the asserted claims of the '785 Patent. The combinations of references identified below are non-exhaustive, and additional examples are apparent from other combinations of references charted in Exhibits 785-1 – 785-29. Although certain references charted in Exhibits 785-1 – 785-29 are listed variously in the left or right column below, Southwest does not thereby imply any restriction as to which reference is primary and which reference is secondary in such a combination. Southwest reserves the right to rely on either identified reference in the identified combinations as a primary and/or secondary reference.

Exhibit	Reference	Combination Reference(s)
785-1	Mehta	RFC 1383, Caronni I, Caronni II, and/or Hipp, and/or one or more references as charted in Exhibits 785-1 – 785-29
785-2	Caronni I	Caronni II, Hipp, Mehta, and/or RFC 1383, and/or one or more references as charted in Exhibits 785-1 – 785-29
785-3	Alkhatib	One or more references as charted in Exhibits 785-1 – 785-29
785-4	Foster I	One or more references as charted in Exhibits 785-1 – 785-29
785-5	Bhalla	One or more references as charted in Exhibits 785-1 – 785-29
785-6	Foster II	One or more references as charted in Exhibits 785-1 – 785-29
785-7	Tuomenoksa	One or more references as charted in Exhibits 785-1 – 785-29
785-8	Chen	One or more references as charted in Exhibits 785-1 – 785-29
785-9	Engmann	One or more references as charted in Exhibits 785-1 – 785-29

Exhibit	Reference	Combination Reference(s)
785-10	Forslow	One or more references as charted in Exhibits 785-1 – 785-29
785-11	Hoke	One or more references as charted in Exhibits 785-1 – 785-29
785-12	Iyer	One or more references as charted in Exhibits 785-1 – 785-29
785-13	Jain	One or more references as charted in Exhibits 785-1 – 785-29
785-14	Kim	One or more references as charted in Exhibits 785-1 – 785-29
785-15	Liu	One or more references as charted in Exhibits 785-1 – 785-29
785-16	Viswanath	One or more references as charted in Exhibits 785-1 – 785-29
785-17	Warrier	One or more references as charted in Exhibits 785-1 – 785-29
785-18	Borella	One or more references as charted in Exhibits 785-1 – 785-29
785-19	Johnson	One or more references as charted in Exhibits 785-1 – 785-29
785-20	Volz	One or more references as charted in Exhibits 785-1 – 785-29
785-21	Thubert	One or more references as charted in Exhibits 785-1 – 785-29
785-22	Tari	One or more references as charted in Exhibits 785-1 – 785-29
785-23	Dunk	One or more references as charted in Exhibits 785-1 – 785-29
785-24	Wootton	One or more references as charted in Exhibits 785-1 – 785-29

Exhibit	Reference	Combination Reference(s)
785-25	Ryu	One or more references as charted in Exhibits 785-1 – 785-29
785-26	Kong	One or more references as charted in Exhibits 785-1 – 785-29
785-27	Kakemizu	One or more references as charted in Exhibits 785-1 – 785-29
785-28	Cisco VPN Routers and Software Systems	Mehta, RFC 1383, Caronni I, Caronni II, and/or Hipp, and/or one or more references as charted in Exhibits 785-1 – 785-29
785-29	VMware Systems	Mehta, RFC 1383, Caronni I, Caronni II, and/or Hipp, and/or one or more references as charted in Exhibits 785-1 – 785-29

It will be understood that, given the similarities between the art, any of Mehta, Caronni I, Caronni II, Hipp, RFC 1383, Alkhatib, Foster I , Bhalla, Foster II, Tuomenoksa, Chen, Engmann, Forslow, Hoke, Iyer, Jain, Kim, Liu, Viswanath, Warrier, Borella, Johnson, Volz, Thubert, Tari, Dunk, Wootton, Ryu, Kong, Kakemizu, Cisco VPN Routers and Software Systems, and/or VMware Systems could be combined with one or more of each other, for example based on the disclosures shown in Exhibits 785-1 – 785-29. It would have been obvious to combine these references for reasons that are readily apparent to one of ordinary skill in the art, and for reasons explained in additional detail in the sections that follow.

2. The Claims Recite Well-Known Concepts that Would Have Been Obvious to Implement

As noted above, the asserted claims of the ‘785 Patent represent nothing more than an arrangement of well-known concepts and features and add nothing to the state of the art at the time of the alleged invention. For each of the concepts represented by the claims, it would have been

inherently obvious to add to the teachings of any number of known references and products, including to any of the primary references charted in Exhibits 785-1 – 785-29 to the extent that such reference is alleged to be missing such concept or feature.

3. Additional Reasons to Combine

In addition to the information contained elsewhere in these invalidity contentions, including in Exhibits 785-1 – 785-29, Southwest has identified additional motivations and reasons to combine the various references. Southwest notes that in *KSR Int'l Co. v. Teleflex Inc.*, the Supreme Court held that "[i]n determining whether the subject matter of a patent claim is obvious, neither the particular motivation nor the avowed purpose of the patentee controls. What matters is the objective reach of the claim. If the claim extends to what is obvious, it is invalid under § 103." 550 U.S. 398, 127 S.Ct. 1727, 1742 (2007).

KSR further illustrated several ways in which the subject matter of a patent claim may be shown to be obvious. For instance, "[o]ne of the ways in which a patent's subject matter can be proved obvious is by noting that there existed at the time of invention a known problem for which there was an obvious solution encompassed by the patent's claims." *KSR*, 127 S.Ct. 1727, 1742. The Supreme Court held that it was error to "look only to the problem the patentee was trying to solve." *Id*. Rather, "[t]he question is not whether the combination was obvious to the patentee but whether the combination was obvious to a person with ordinary skill in the art. Under the correct analysis, any need or problem known in the field of endeavor at the time of invention and addressed by the patent can provide a reason for combining the elements in the manner claimed." *Id*. Further, "[c]ommon sense teaches … that familiar items may have obvious uses beyond their primary purposes, and in many cases a person of ordinary skill will be able to fit the teachings of multiple patents together like pieces of a puzzle." *Id.* Further, a showing that a combination of elements was "obvious to try" may show that it was obvious under § 103. *Id.* For instance, the Supreme

Court held that "predictable solutions" using a combination of "known options" may render the subject matter of a patent claim obvious.

When there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense. *Id.* The ways in which the subject matter of a patent claim may be shown to be obvious, identified by *KSR*, are merely illustrative. The main thrust of *KSR* was that "[r]igid preventative rules that deny factfinders recourse to common sense"—such as the overturned "teaching, suggestion, or motivation" test—are neither necessary under our case law nor consistent with it." *KSR*, 127 S.Ct. 1727, 1742-43; citing with approval, e.g., *DyStar Textilfarben GmbH & Co. Deutschland KG v. C.H. Patrick Co.*, 464 F. 3d 1356, 1367 (2006) ("Our suggestion test is in actuality quite flexible and not only permits, but requires, consideration of common knowledge and common sense"); *Alza Corp. v. Mylan Labs., Inc.*, 464 F. 3d 1286, 1291 (2006) ("There is flexibility in our obviousness jurisprudence because a motivation may be found implicitly in the prior art. We do not have a rigid test that requires an actual teaching to combine...").

Although Southwest notes that there is no longer a rigid requirement regarding motivation to combine under *KSR*, Southwest provides the following statements regarding motivation to combine. Multiple teachings, suggestions, and/or reasons to modify any of the references and/or to combine any two or more of the references charted in Exhibits 785-1 – 785-29 come from many sources, including the prior art (specific and as a whole), common knowledge, common sense, predictability, expectations, industry trends, design incentives or need, market demand or pressure, market forces, obvious to try, the nature of the problem faced, and/or knowledge possessed by one

of ordinary skill. In addition, it would have been obvious to try combining the prior art references identified above because there were only a finite number of predictable solutions and/or because known work in one field or endeavor prompted variations based on predictable design incentives and/or market forces either in the same field or a different one.

The combination of prior art references identified in these contentions would have been obvious because the combination represents the known potential options with a reasonable expectation of success. Additionally, one of ordinary skill in the art would have been motivated to create combinations identified in these contentions using: known methods to yield predictable results; known techniques in the same way; a simple substitution of one known, equivalent element for another to obtain predictable results; and/or teaching, suggestion, or motivation in the prior art generally. Also, market forces in the industry, and the desire to improve features and performance, would motivate the addition of features to products as they become available, become less expensive, become more commonly used, provide better performance, reduce costs, or predictably achieve other clearly desirable results.

Moreover, in addition to the information contained elsewhere in these contentions, and, in particular, charted in Exhibits 785-1 – 785-29, Southwest has identified additional motivations and reasons to combine the various references cited herein. In particular, multiple teachings, suggestions, and/or reasons to modify any of the references and/or to combine any two or more of the references charted in Exhibits 785-1 – 785-29 come from many sources, including the prior art (specific and as a whole), common knowledge, common sense, predictability, expectations, industry trends, design incentives or need, market demand or pressure, market forces, obvious to try, the nature of the problem faced, and/or knowledge possessed by one of ordinary skill. In addition, it would have been obvious to try combining the prior art references identified above

because there were only a finite number of predictable solutions and/or because known work in one field or endeavor prompted variations based on predictable design incentives and/or market forces either in the same field or a different one. The combination of prior art references identified in these contentions would have been obvious because the claimed combinations represent the known potential options with a reasonable expectation of success.

To the extent that Plaintiffs contend that any of the above-identified references does not expressly or inherently disclose a particular element or limitation, it would have been obvious to a person of ordinary skill in the art to modify the references or to combine their teachings with other prior art references, including the prior art references in these invalidity contentions, in a manner that renders such claims obvious and invalid.

In particular, various references identified above disclose, well before the critical date of the '785 Patent, virtual network managers having a network interface and that communicate via a virtual network defined by a domain name having a public network address, and that include a register module to register devices in the virtual network.

For example, Mehta discloses a system designed to enable secure, bi-directional communication between public network devices and wireless devices on private networks. *See* Mehta at Abstract. Mehta discloses an Address Management Proxy System (AMPS) that allocates temporary public IP addresses to wireless devices to facilitate communication across public and private networks without exposing private IP addresses. Mehta at Abstract. Mehta's system includes modified DNS/API servers, Address Proxy/Routers, and an Address Management Data Server to dynamically assign public IP addresses to private network devices as needed. *See* Mehta at [0011]. Metha's AMPS uses techniques like Time to Live (TTL) for each IP mapping, ensuring temporary mappings that enhance security by automatically expiring connections. *See* Mehta at

[0015]. The system manages a pool of public addresses to allocate dynamically, using protocols like TCP/IP and UDP for seamless data exchange. *See* Mehta at [0011], [0024]. AMPS can support various types of networks and devices beyond traditional IP networks, including ATM networks and other wireless technologies. Metha addresses the challenge of secure and efficient communication between disparate network types, enhancing connectivity options without compromising network security. *See* Mehta at [0041].

As another example, RFC 1383 discloses "a scheme that allows for simple routing" that is "complementary with the classic hierarchical routing approach," while providing "an easy to implement and low cost solution for 'multi-homed' domains." *See* RFC 1383 at 2. Specifically, the RFC-1383 solution is described as "a generalization of the 'MX record' scheme used for mail routing." *See* RFC 1383 at 2. RFC-1383 introduces "RX records" for routing IP packets by associating domain names with preferred gateways by means of "source routing" or "tunneling," or another form of encapsulation protocol. *See* RFC 1383 at 4. To implement this RX routing, a modified DNS lookup is suggested: the source or an early relay makes the routing decision by querying the DNS for RX records, which list gateways with preference levels. *See* RFC 1383 at 6, 9. RFC-1383 describes a network interface integrated with a standard IP router, working alongside a DNS query manager. *See* RFC 1383 at 12-13. When an Internet Control Message Protocol (ICMP) message is received, the query manager updates the local routing table to ensure that any new packets bound for the specified destination are routed through the real-time forwarder. *Id.* Simultaneously, the query manager sends a DNS request to read the RX records for the destination. *Id.* Upon receiving the response, the query manager selects a gateway and provides this information to the real-time forwarder. *See* RFC 1383 at 11-12. When the real-time forwarder receives a packet, it checks if a gateway for the destination is available. *See* RFC 1383 at 12-13. If

so, it inserts the necessary source routing information and forwards the packet—either through the general IP routing program or directly to the network interface linked to the intermediate gateway. *See* RFC 1383 at 12.

As another example, Caronni I discloses "data processing systems and, more particularly, [] a private network using a public-network infrastructure." *See* Caronni I at 1:57–59. Caronni I establishes a "'Supernet,' which is a private network that uses components from a public-network infrastructure." *See* Caronni I at 4:38–39. The Supernet allows organizations to utilize public networks for their enterprise networks, reducing the burden on maintaining network infrastructure. *See* Caronni I at 4:40–43. Caronni I's system uses a separate layer that isolates address names of nodes from addressing schemes and delivery schemes. *See* Caronni I at 4:55–59 ("The Supernet contains a modification to the IP packet format that can be used to separate network behavior from addressing. As a result of the modification, any delivery scheme may be assigned to any address, or group of addresses."). This separation provides flexibility in addressing and delivery methods. When a node attempts to join a Supernet through the SNlogin Script, SASD validates and authenticates the request and then "creates an address mapping between a node ID and the real address." *See* Caronni I at 9:67–10:18 (SNlogin script "performs a handshaking with SASD to authenticate this information.").

Caronni II discloses "a virtual computer network and more particularly to the addressing of messages sent in a virtual network using data encapsulated in higher level protocols." *See* Caronni II at 1:6–9. In particular, Caronni II addresses the challenge of sending messages to virtual addresses in networks with packet filtering or network address translation (NAT) devices. *See* Caronni II at 3:54–57, 8:14–25. Caronni II introduces an extended virtual address registration process that includes higher-level protocol designations, allowing messages to bypass filtering

mechanisms. *See* Caronni II at 2:40–44. Caronni II discloses a virtual network overlay on a physical network, where "[a] virtual network is a logical network overlaid on a physical network." *See* Caronni II at 1:13–14. Caronni II utilizes a virtual address resolution facility that "works with stored virtual address registrations for the virtual network." *See* Caronni II at 4:9–10. These registrations are stored in a Virtual Address Resolution Protocol (VARP) lookup table and include not only the virtual IP address corresponding real IP address but also "a designation of a transport protocol and a port address for an interface between the Application layer and the Transport layer." *See* Caronni II at 4:62–65.

As another example, Hipp discloses "[a] virtual network environment to be used by a set of applications for the express purpose of isolating the applications from other applications on the same node or network[.]" *See* Hipp at Abstract. The "Virtual Network Environment" (VNE) is defined as "a collection of IP addresses related to applications that are contained within the[] VNE or have the potential of being placed in the VNE." *See* Hipp at 2:66–3:2. The system assigns a "unique virtual IP address and virtual hostname to a group of processes that make up the application instance which the instance keeps throughout its execution." *See* Hipp at 6:1–4. This virtual network identity (VNI) remains with the application regardless of which node it runs on, allowing for transparent movement across machines without modifying the application. *See* Hipp at 6:4–6.

Further, Hipp demonstrates that using a domain name server (DNS) was well-known and understood well before the '785 Patent. Hipp discloses using a DNS in the context of hostname resolution. *See* Hipp at 6:16–25. "The virtual hostname resolves to the virtual IP address for both the applications registered with the VNI framework as well as those that are not registered. This may require configuration of a name service or OS host configuration files." *See* Hipp at 6:16–20.

"For example, if an application instance used a virtual IP address of 10.10.0.1 and a virtual hostname of host1055, the standard hostname to IP address resolution mechanisms (e.g. DNS or the /etc/hosts file) would have to be preconfigured to resolve a query of host1055 to IP address 10.10.0.1." *See* Hipp at 6:20–25.

As yet another example, the Cisco VPN Routers and Software Systems provided devise and software enabling fast data communication through network interfaces, facilitated adding and removing devices from the network, including supporting VPN tunnels for such devices, and performing address translation. *See, e.g.*, Cisco 7100 Series VPN Routers, Cisco VPN 3000 Series Concentrator Data Sheet, Cisco VPN 3000 Series Concentrator Data Sheet.

As yet another example, the VMware Systems provided a virtual infrastructure that provided dynamic resource allocation, enabling dynamic management of server resources and provisioning through a management console, and provisioning of new resources. *See, e.g.*, VMware ESX Server Specifications, VMware GSX Server Specifications, VMware Enterprises Continue to Embrace VMware Virtual Infrastructure.

A person of ordinary skill in the art would have been motivated to combine the above references to incorporate the different beneficial technical features provided by each of the above references. Modifying the above references would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results.

* * *

In addition to the specific combinations of prior art and the specific combinations of groups of prior art disclosed above, Southwest reserves the right to rely on any other combination of any prior art references disclosed herein. Southwest further reserves the right to rely upon

combinations of prior art references disclosed within the prosecution history of the references cited herein.

The references charted in Exhibits 785-1 – 785-29 disclose the elements of the asserted claims of the '785 Patent explicitly and/or inherently and may also be relied upon to show the state of the art at the time of the alleged invention. The suggested obviousness combinations are provided in the alternative to Southwest's anticipation contentions and are not to be construed to suggest that any reference included in the combinations is not itself anticipatory.

C. OGP: Charts Setting Forth Where in the Prior Art References Each Element of the Asserted Claims Are Found

Pursuant to the Court's OGP, charts setting forth where specifically in each alleged item of prior art each element of each Asserted Claim of the '785 Patent is found are attached as Exhibits 785-1 – 785-29.

D. Other Relevant Prior Art References

In addition to the references identified above and charted in Exhibits 785-1 – 785-29, each of the references below is relevant to the '785 Patent, as it relates to the state of the art. Southwest reserves the right to amend its contentions to rely on any of the below references alone or in combination with any other references identified in these contentions.

Inventor/Author	Country	Issue/Publication No.	Filed Date
Hasan S. Alkhatib et al.	US	6,119,171	Sept. 12, 2000
Hasan S. Alkhatib et al.	US	7,139,828	Nov. 21, 2006
Hasan S. Alkhatib et al.	US	2004/0044778	Aug. 30, 2002
Jan Forslöw	US	7,155,518	January 8, 2001
Barrett Morris Kreiner et al.	US	2008/0232295	November 27, 2002
Anthony J. Li et al.	US	5,473,599	December 5, 1995
Changwen Lui et al.	US	2004/0120295	December 19, 2002
Samir Mehta et al.	WO	2002/102012A2	June 10, 2002
Emily L. Hipp	US	7,146,431	Dec. 5, 2006
Emily L. Hipp	US	7,210,147	Oct. 5, 2000
Can C. Aysan et al.	US	7,379,465	December 7, 2001

Inventor/Author	Country	Issue/Publication No.	Filed Date
Qiaofeng Zhou	US	2002/0138578	January 24, 2001
Ashar Aziz	US	6,643,701	Nov. 17, 1999
Moon-Jeong Choi	US	2002/0040397	October 2, 2001
Germano Caronni et al.	US	6,798,782	Dec. 10, 1999
Germano Caronni et al.	US	7,685,309	Aug. 11, 2005 – priority Dec. 10, 1999
Germano Caronni et al.	US	7,765,581	Dec. 10, 1999
David Spooner	US	8,743,891	Feb. 24, 2004 – priority Aug. 24, 2001
Qiaofend Zhou	US	6,950,879	Jan. 24, 2001
John Keane et al.	US	6,996,628	April 11, 2001
Robert P. St. Peirre et al.	US	7,254,835	Jan. 4, 2002
Robert P. St. Peirre et al.	US	7,890,633	Feb. 13, 2003
James Russell Godwin	US	7,519,721	April 4, 2008 – priority Jan. 17. 2001
Luis F. Cabrera	US	7,899,047	Aug. 13, 2007 – priority Nov. 27, 2001
Werner Lindemann	US	2002/0010796	Jan. 24, 2002
Thomas T. Nguyen	US	2002/0016926	April 26, 2001
Samuel Bendinelli	US	2002/0029276	April 11, 2001
Jheroen P. Dorenbosch	US	2002/0138622	March 21, 2001
John Andrew Aiken JR.	US	2002/0178268	May 22, 2001
Yuefeng Liu	US	2002/0154635	April 23, 2001
Glenn C. Scott	US	2004/0019633	July 24, 2002
Glen Ceniza	US	2002/0186698	June 12, 2001
Germano Caronni et al.	WO	01/43329	Dec. 8, 2000
Germano Caronni et al.	WO	01/45351	Dec. 8, 2000
Germano Caronni et al.	WO	01/43355	Dec. 11, 2000
Germano Caronni et al.	WO	01/43356	Dec. 11, 2000
Germano Caronni et al.	WO	01/43369	Dec. 11, 2000
Germano Caronni et al.	WO	01/43391	Dec. 11, 2000
Holger Karl et al.	WO	2004/075506	Feb. 16, 2004
M. Skye Poier	CA	2316428	Pub. Feb. 18, 2002
Robert P. St. Peirre et al.	UK	2384667	Pub. July 30, 2003
Microsoft Corporation, "Virtual Private Networking," *Microsoft TechNet – Networking & RAS Technology Center*			Pub. Jan. 17, 2000
C. Perkins, "IP Mobility Support for IPv4," *Request for Comments: 3220*, Internet Engineering Task Force			Jan. 18, 2002

Inventor/Author	Country	Issue/Publication No.	Filed Date
C. Perkins, "Mobile IP," *IEEE Communications Magazine*, IEEE Communications Society			Pub. May 1997
G. Caronni et al., "Virtual Enterprise Networks: The Next Generation of Secure Enterprise Networking," Annual Computer Security Applications Conference (2000)			Pub. 2000
VPN 3000 Series Concentrator Getting Started; Cisco Systems - Https://Web.Archive.Org/Web/20030308052045/Http://Www.Cisco.Com/Application/Pdf/En/Us/Guest/Products/Ps2284/C1692/Ccmigration_09186a00801021e2.Pdf			Dated November 2001; Pub. At least by March 8, 2003
Rekhter, Y. et al., Network Working Group Request for Comment (RFC): 1918, entitled Address Allocation for Private Internets			Pub. Feb. 1996
Fox, B. et al., Network Working Group Request for Comment (RFC): 2685, entitled Virtual Private Networks Identifier			Pub. Sept. 1999
Gleeson, B. et al., Network Working Group Request for Comment (RFC): 2764, entitled A Framework for IP Based Virtual Private Networks			Pub. Feb. 2000
Lim, et al., "Customizable virtual private network service with QoS" *Computer Networks*, Volume 36, Issue 1, Pages 137–151, Elsevier Science B.V.			Pub. July 2001
Okazaki, Naonobu et al., "An Efficient Method for Private Network Management Using IP Address Translation," T.IEE Japan, Vo. 121-C, No. 10, 2001			Pub. 2001

Inventor/Author	Country	Issue/Publication No.	Filed Date
Kato, et al., "Mobile IP Using Private IP Addresses," Proceedings of the Sixth IEEE Symposium on Computers and Communications			Pub. July 2001
Zaccone, et al., "Address reuse in the Internet, adjourning or suspending the adoption of IP next generation?," Proceedings of IEEE International Conference on Networks 2000 (ICON 2000) Networking Trends and Challenges in the New Millenium			Pub. 2000
A. Kara, "Secure Remote Access from Office to Home," *IEEE Communications Magazine*, Vol. 39, Issue 10, pp. 68-72			Pub. Oct. 1, 2001
H. Liu, "Mobile Domain Name System An Alternative for Mobile IP," Proceedings of the 8th Annual International Conference on Communications Systems 2002 (ICCS 2002)			Pub. 2002

VII. THE '326 PATENT IS INVALID UNDER 35 U.S.C. §§ 102 AND/OR 103

A. The '326 Patent Priority Date

Plaintiff's Infringement Contentions allege the asserted claims of the '326 Patent have a filing date of January 12, 2004.

B. Identification of Prior Art that Anticipates or Renders Obvious the Asserted Claims of the '326 Patent

Southwest contends that the prior art references listed below and charted in Exhibits 326-1 – 326-11 anticipate and/or render obvious, alone or in combination, one or more of the asserted claims of the '326 Patent.

Prior Art Reference	Publication Date
U.S. Patent Application Publication No. 2003/022473 ("Yamaura")	Filed – February 20, 2003 Published – December 4, 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent No. 6,510,133 ("Uesugi")	Filed – January 21, 1999 Issued – January 21, 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
JP Patent Application Publication No. JP2002319917 ("Mori")	Filed – April 24, 2001 Published – October 31, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b).
JP Patent Application Publication No. JP2003309533 ("Miyoshi")	Filed – April 17, 2002 Published – October 31, 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(a).
JP Patent Application Publication No. JPH11113049 ("Koga")	Filed – September 30, 1997 Published – April 23, 1999 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b).
U.S. Patent No. 7,418,050 ("Gardner")	Filed – February 26, 2003 Issued – August 26, 2008 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
PCT Patent Application Publication No. WO 00/38387 ("Baldemair")	Filed – December 21, 1999 Published – October 10, 2001 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b).
Thompson, D. E., "Modelling Adjacent Channel Interference in 3G Networks," THE INSTITUTE OF	Published – 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).

Prior Art Reference	Publication Date
ELECTRICAL ENGINEERS (2003) ("Thompson")	
Weste, Neil et al., "VLSI for OFDM," IEEE COMMUNICATIONS MAGAZINE (Oct. 1998) ("Weste")	Published – 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(b) and 102(e).
"WLAN Channel Bonding: Causing Greater Problems Than It Solves" ("Texas Instruments White Paper")	Published – September 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
"WLAN Solutions: TNETW1130 Converged Single-Chip MAC and Baseband Processor for IEEE 802.11 a/b/g" ("TI Baseband Processor for IEEE 802.11")	Published – 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).

Plaintiffs assert claims 1, 4, and 18 of the '326 Patent against Southwest in this lawsuit. These claims are invalid because the '326 Patent fails to meet one or more of the requirements for patentability. The individual bases for invalidity are provided below and in Exhibits 326-1 – 326-11. Each of the prior art references, the underlying work, and/or the underlying apparatus or method qualifies as prior art under one or more sections of Pre-AIA 35 U.S.C. §§ 102 and/or 103.

In an effort to focus the issues, Southwest has cited representative portions of identified references, even where a reference may contain additional disclosures or support for a particular claim element. Each and every disclosure of the same limitation in the same reference is not necessarily identified. In addition, persons of ordinary skill in the art generally read a prior art reference as a whole and in the context of other publications and literature and in the context of their knowledge as persons of ordinary skill in the art. Thus, to understand and interpret any specific statement or disclosure within a prior art reference, such persons would rely on other

information within the reference, along with other publications and their general scientific knowledge and experience in the field. Thus, each prior art reference identified herein should be understood in the context of the reference itself in conjunction with anything incorporated by reference or described as relevant technology by the reference, any documents that are referred to for further understanding by the reference, any product embodying the reference, and any testimony by those with knowledge of the reference, such as the named authors and inventors. Southwest may rely upon uncited portions of the prior art references and on other publications and expert testimony to provide context, and as aids to understanding and interpreting the portions that are cited. Southwest may also rely on uncited portions of the prior art references, other disclosed publications, and the testimony of experts to establish that a person of ordinary skill in the art would have been motivated to modify or combine certain of the cited references so as to render the claims obvious.

Some or all of the asserted claims of the ‘326 Patent are invalid as anticipated under Pre-AIA 35 U.S.C. § 102 in view of certain ones of the prior art references identified above as detailed in the claim charts included in Exhibits 326-1 – 326-11, which identify specific examples of where each limitation of the asserted claims of the ‘326 Patent is found in the prior art references. As explained above, the cited portions of prior art references identified in the attached claim charts are exemplary in nature and representative of the content and teaching of the prior art references, and should be understood in the context of the reference as a whole and as they would be understood by a person of ordinary skill in the art.

To the extent any item of prior art cited above is deemed not to disclose, explicitly or inherently, any limitation of an asserted claim of the ‘326 Patent, Southwest reserves the right to argue that any difference between that prior art and the corresponding patent claim would have

been either inherent in the art or obvious to a person of ordinary skill in the art, even if Southwest has not specifically denoted that the art is to be combined with the knowledge of a person of ordinary skill in the art.

To the extent Plaintiffs argue that any limitation of the asserted claims of the '326 Patent is not explicitly disclosed by an item of prior art identified above and in Exhibits 326-1 – 326-11, any purported differences are such that the claimed subject matter as a whole would have been obvious to one skilled in the art at the time of the alleged invention, in view of the state of the art and knowledge of those skilled in the art. The item of prior art would, therefore, render the relevant claims invalid as obvious under Pre-AIA 35 U.S.C. § 103.

To the extent any of the foregoing references are found not to anticipate the asserted claims of the '326 Patent, the foregoing references render the asserted claims obvious, either alone or in combination with one or more of the other references identified above. One of ordinary skill in the art would have been motivated to combine the various references cited herein so as to disclose each and every limitation of the asserted claims of the '326 Patent.

The prior art identified in Exhibits 326-1 – 326-11 renders one or more asserted claims of the '326 Patent obvious when the references are read alone or in combination with one or more of each other and/or when read in view of the state of the art and knowledge of those skilled in the art.

To the extent that Plaintiff argues that any asserted claim of the '326 Patent is not disclosed or rendered obvious by any reference charted in Exhibits 326-1 – 326-11, the asserted claims nevertheless would have been obvious in light of one or more other references charted in Exhibits 326-1 – 326-11 and in view of the knowledge, experience, and understanding of persons of ordinary skill in the art.

Each and every reference identified is also relevant to the state of the art at the time of the alleged invention. Any of the references disclosed above may be combined with one another to render obvious (and therefore invalid) each of the asserted claims of the '326 Patent. Southwest may rely upon a subset of the above identified references, or all of the references identified above, including all references in charted in Exhibits 326-1 – 326-11, for purposes of obviousness depending on the Court's claim construction, positions taken by Plaintiffs during this litigation, and further investigation and discovery.

1. Example Combinations of the Prior Art

The table below provides an identification of exemplary combinations of prior art references that, based on Southwest's present understanding of Plaintiff's Infringement Contentions and apparent reading of the claims, render obvious one or more of the asserted claims of the '326 Patent. The combinations of references identified below are non-exhaustive, and additional examples are apparent from other combinations of references charted in Exhibits 326-1 – 326-11. Although certain references charted in Exhibits 326-1 – 326-11 are listed variously in the left or right column below, Southwest does not thereby imply any restriction as to which reference is primary and which reference is secondary in such a combination. Southwest reserves the right to rely on either identified reference in the identified combinations as a primary and/or secondary reference.

Exhibit	Reference	Combination Reference(s)
326-1	Baldemair	Gardner, Koga, Miyoshi, Mori, Yamaura, Uesegi, Texas Instruments White Paper, TI Baseband Processor for IEEE 802.11, and/or one or more references as charted in Exhibits 326-1 – 326-11.
326-2	Gardner	Miyoshi, Koga, Mori, Uesugi, TI Baseband Processor for IEEE 802.11,

Exhibit	Reference	Combination Reference(s)
		Yamaura, Texas Instruments White Paper, and/or one or more references as charted in Exhibits 326-1 – 326-11.
326-3	Koga	Gardner, Miyoshi, Mori, Uesugi, Yamaura, TI Baseband Processor for IEEE 802.11, Texas Instruments White Paper, and/or one or more references as charted in Exhibits 326-1 – 326-11.
326-4	Miyoshi	Gardner, Koga, Mori, TI Baseband Processor for IEEE 802.11, Texas Instruments White Paper, Yamaura, Uesugi and/or one or more references as charted in Exhibits 326-1 – 326-11.
326-5	Mori	Gardner, Koga, Miyoshi, Yamaura, Texas Instruments White Paper, TI Baseband Processor for IEEE 802.11, Uesugi and/or one or more references as charted in Exhibits 326-1 – 326-11.
326-6	Texas Instruments White Paper	Gardner, Koga, Miyoshi, Mori, TI Baseband Processor for IEEE 802.11, Uesugi, Yamaura, and/or one or more references as charted in Exhibits 326-1 – 326-11.
326-7	Thompson	Gardner, Koga, Miyoshi, Mori, Uesugi, Yamaura, Texas Instruments White Paper, TI Baseband Processor for IEEE 802.11, and/or one or more references as charted in Exhibits 326-1 – 326-11.
326-8	TI Baseband Processor for IEEE 802.11	Yamaura, Uesugi, Mori, Miyoshi, Koga, Gardner, Texas Instruments White Paper and/or one or more references as charted in Exhibits 326-1 – 326-11.
326-9	Uesugi	Gardner, Koga, Miyoshi, Mori, Texas Instruments White Paper, TI Baseband Processor for IEEE 802.11, Yamaura, and/or one or more references as charted in Exhibits 326-1 – 326-11.

Exhibit	Reference	Combination Reference(s)
326-10	Weste	Gardner, Koga, Yamaura, Mori, Miyoshi, Texas Instruments White Paper, TI Baseband Processor for IEEE 802.11, Uesugi, and/or one or more references as charted in Exhibits 326-1 – 326-11.
326-11	Yamaura	Gardner, Koga, Mori, Miyoshi, Texas Instruments White Paper, TI Baseband Processor for IEEE 802.11, Uesugi, and/or one or more references as charted in Exhibits 326-1 – 326-11

It will be understood that, given the similarities between the art, any of Baldemari, Gardner, Koga, Miyoshi, Mori, Texas Instruments White Paper, Thompson, TI Baseband Processor for IEEE 802.11, Uesugi, Weste, and/or Yamaura could be combined with one or more of each other, for example based on the disclosures shown in Exhibits 326-1 – 326-11. It would have been obvious to combine these references for reasons that are readily apparent to one of ordinary skill in the art, and for reasons explained in additional detail in the sections that follow.

2. The Claims Recite Well-Known Concepts that Would Have Been Obvious to Implement

As noted above, the asserted claims of the ‘326 Patent represent nothing more than an arrangement of well-known concepts and features and add nothing to the state of the art at the time of the alleged invention. For each of the concepts represented by the claims, it would have been inherently obvious to add to the teachings of any number of known references and products, including to any of the primary references charted in Exhibits 326-1 – 326-11 to the extent that such reference is alleged to be missing such concept or feature.

3. Additional Reasons to Combine

In addition to the information contained elsewhere in these invalidity contentions, including in Exhibits 326-1 – 326-11, Southwest has identified additional motivations and reasons to combine the various references. Southwest notes that in *KSR Int'l Co. v. Teleflex Inc.*, the Supreme Court held that "[i]n determining whether the subject matter of a patent claim is obvious, neither the particular motivation nor the avowed purpose of the patentee controls. What matters is the objective reach of the claim. If the claim extends to what is obvious, it is invalid under § 103." 550 U.S. 398, 419 (2007).

KSR further illustrated several ways in which the subject matter of a patent claim may be shown to be obvious. For instance, "[o]ne of the ways in which a patent's subject matter can be proved obvious is by noting that there existed at the time of invention a known problem for which there was an obvious solution encompassed by the patent's claims." *KSR*, 127 S.Ct. at 1742. The Supreme Court held that it was an error to "look only to the problem the patentee was trying to solve." *Id*. Rather, "[t]he question is not whether the combination was obvious to the patentee but whether the combination was obvious to a person with ordinary skill in the art. Under the correct analysis, any need or problem known in the field of endeavor at the time of invention and addressed by the patent can provide a reason for combining the elements in the manner claimed." *Id*. Further, "[c]ommon sense teaches … that familiar items may have obvious uses beyond their primary purposes, and in many cases a person of ordinary skill will be able to fit the teachings of multiple patents together like pieces of a puzzle." *Id.* Further, a showing that a combination of elements was "obvious to try" may show that it was obvious under § 103. *Id.* For instance, the Supreme Court held that "predictable solutions" using a combination of "known options" may render the subject matter of a patent claim obvious.

When there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense. *Id.* The ways in which the subject matter of a patent claim may be shown to be obvious, identified by *KSR*, are merely illustrative. The main thrust of *KSR* was that "[r]igid preventative rules that deny factfinders recourse to common sense"—such as the overturned "teaching, suggestion, or motivation" test—are neither necessary under our case law nor consistent with it." *KSR*, 127 S.Ct. at 1742-43; citing with approval, *e.g.*, *DyStar Textilfarben GmbH & Co. Deutschland KG v. C.H. Patrick Co.*, 464 F. 3d 1356, 1367 (2006) ("Our suggestion test is in actuality quite flexible and not only permits, but requires, consideration of common knowledge and common sense"); *Alza Corp. v. Mylan Labs., Inc.*, 464 F. 3d 1286, 1291 (2006) ("There is flexibility in our obviousness jurisprudence because a motivation may be found implicitly in the prior art. We do not have a rigid test that requires an actual teaching to combine . . .").

Although Southwest notes that there is no longer a rigid requirement regarding motivation to combine under *KSR*, Southwest provides the following statements regarding motivation to combine. Multiple teachings, suggestions, and/or reasons to modify any of the references and/or to combine any two or more of the references charted in Exhibits 326-1 – 326-11 come from many sources, including the prior art (specific and as a whole), common knowledge, common sense, predictability, expectations, industry trends, design incentives or need, market demand or pressure, market forces, obvious to try, the nature of the problem faced, and/or knowledge possessed by one of ordinary skill. In addition, it would have been obvious to try combining the prior art references identified above because there were only a finite number of predictable solutions and/or because

known work in one field or endeavor prompted variations based on predictable design incentives and/or market forces either in the same field or a different one.

The combination of prior art references identified in these contentions would have been obvious because the combination represents the known potential options with a reasonable expectation of success. Additionally, one of ordinary skill in the art would have been motivated to create combinations identified in these contentions using: known methods to yield predictable results; known techniques in the same way; a simple substitution of one known, equivalent element for another to obtain predictable results; and/or teaching, suggestion, or motivation in the prior art generally. Also, market forces in the industry, and the desire to improve features and performance, would motivate the addition of features to products as they become available, become less expensive, become more commonly used, provide better performance, reduce costs, or predictably achieve other clearly desirable results.

Moreover, in addition to the information contained elsewhere in these contentions, and, in particular, charted in Exhibits 326-1 – 326-11, Southwest has identified additional motivations and reasons to combine the various references cited herein. In particular, multiple teachings, suggestions, and/or reasons to modify any of the references and/or to combine any two or more of the references charted in Exhibits 326-1 – 326-11 come from many sources, including the prior art (specific and as a whole), common knowledge, common sense, predictability, expectations, industry trends, design incentives or need, market demand or pressure, market forces, obvious to try, the nature of the problem faced, and/or knowledge possessed by one of ordinary skill. In addition, it would have been obvious to try combining the prior art references identified above because there were only a finite number of predictable solutions and/or because known work in one field or endeavor prompted variations based on predictable design incentives and/or market

forces either in the same field or a different one. The combination of prior art references identified in these contentions would have been obvious because the claimed combinations represent the known potential options with a reasonable expectation of success.

To the extent that Plaintiffs contends that any of the above-identified references does not expressly or inherently disclose a particular element or limitation, it would have been obvious to a person of ordinary skill in the art to modify the references or to combine their teachings with other prior art references, including the prior art references in these invalidity contentions, in a manner that renders such claims obvious and invalid.

For example, many of the prior art references charted in Exhibits 326-1 – 326-11 share a goal of increasing the data rates and data throughput in a network. Various references identified in these Preliminary Infringement Contentions disclose, well before the alleged critical date of the '326 patent, increased speed in data transmission: "In order to increase the speed of the communication channel, it is necessary to allocate a wide frequency bandwidth for each transmission channel." Mori at [0029].

For example, Yamaura discloses "a convention radio communication system" that "accords to the scheme called HiSWANa," where "the frequencies of the center carrier are 5.17 GHz, 5.19 GHz, 5.21 GHz, and 5.23 GHz, and a signal band of 20 MHz (including a guard band) is assigned to each carrier." Yamaura at [0005]. A person of ordinary skill in the art would understand that using a HiSWANa (high-speed wireless access network) scheme for optimal guard band usage is comparable to increasing data rate and throughput.

As another example, Mori discloses selecting adjacent channels comprised of data subcarriers and separated by guard bands: "[t]he radio access system put into practical use uses a bandwidth of 100 MHz in a 5 GHz band, and arranges four channels having an interval of 20 MHz

in the 5-GHz band, and performs communication using each channel." Mori at [0003]. Further, Mori discloses partially filling the frequency gap, or guard band, between the two adjacent channels by adding data subcarriers into the guard band: " . . . the spectrum shown in h) increases the number of data carriers by allocating subcarrier data to the guard band between the continuous bands B1 and B2." Mori at [0131-32]. Further still, Mori discloses filling the guard band using fast fourier transform or inverse fast fourier transform: "subcarriers other than the pilot signal for transmitting the information signal are based on a complex signal sequence supplied to the IFFT," " . . . [t]he IFFT circuit 14 performs inverse Fourier transform on the frequency domain information relating to the signal point position for each subcarrier supplied for each of the plurality of operation blocks, and obtains the time domain information for each IFFT circuit block." Mori at [0012], [0051].

As another example, Gardner discloses combining a first and second channel using channel bonding: "[v]arious options for IEEE 802.11a extensions that would support higher data rates have been considered. One option is to use multiple channels to transmit the data stream; for example, data could be multiplexed between two 54 Mbps streams occupying two contiguous 20 MHz channels to obtain a rate of 108 Mbps." Gardner at [2:15-26].

As set forth above, the market and industry for the increase in data rates and data throughput in a network was well established and known by one of ordinary skill in the art. A person of ordinary skill would know, for example, to consider any feature of increasing data rates and throughput already implemented by major industry players, including systems such as Texas Instruments Baseband Processor for IEEE 802.11, which contemplates increasing data throughput and channel bonding.

* * *

In addition to the specific combinations of prior art and the specific combinations of groups of prior art disclosed above, Southwest reserves the right to rely on any other combination of any prior art references disclosed herein. Southwest further reserves the right to rely upon combinations of prior art references disclosed within the prosecution history of the references cited herein.

The references charted in Exhibits 326-1 – 326-11 disclose the elements of the asserted claims of the '326 Patent explicitly and/or inherently and may also be relied upon to show the state of the art at the time of the alleged invention. The suggested obviousness combinations are provided in the alternative to Southwest's anticipation contentions and are not to be construed to suggest that any reference included in the combinations is not itself anticipatory.

C. OGP: Charts Setting Forth Where in the Prior Art References Each Element of the Asserted Claims Are Found

Pursuant to the Court's OGP, charts setting forth where specifically in each alleged item of prior art each element of each Asserted Claim of the '326 Patent is found are attached as Exhibits 326-1 – 326-11.

D. Other Relevant Prior Art References

In addition to the references identified above and charted in Exhibits 326-1 – 326-11, each of the references below is relevant to the '326 Patent, as it relates to the state of the art. Southwest reserves the right to amend its contentions to rely on any of the below references alone or in combination with any other references identified in these contentions.

Inventor/Author	Country	Issue/Publication No.	Filed Date
IEEE STANDARD FOR INFORMATION TECHNOLOGY. ANSI/ IEEE Std 802.11, 1999 Edition, "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY)			Published 1999

Inventor/Author	Country	Issue/Publication No.	Filed Date
specifications: High-speed Physical Layer in the 5 GHz Band"			
IEEE STANDARD FOR INFORMATION TECHNOLOGY. ANSI/ IEEE "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band ANSI/ IEEE "IEEE Std 802.11b-1999."			Published 1999
IEEE STANDARD FOR INFORMATION TECHNOLOGY. ANSI/ IEEE "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 4: Further Higher Data Rate Extension in the 2.4 GHz Band"			Published 2003
Li et al., "802.11n Channel Model Validation"			Published November 2003

VIII. THE '469 PATENT IS INVALID UNDER 35 U.S.C. §§ 102 AND/OR 103

A. The '469 Patent Priority Date

Plaintiff's Infringement Contentions allege the asserted claims of the '469 Patent have a filing date of September 29, 2003. Southwest disputes the alleged priority date of September 29, 2003. In particular, several aspects of the claims are not adequately supported by the written description of the provisional application. In one example, asserted claim 32 recites "[t]he Internet Hotspot of claim 25, wherein the wireless connection is one of an 802.11a wireless area network, an 802.11b wireless area network, an 802.11g wireless area network, **and an 802.11n wireless area network**." But the 802.11g standard was not adopted until 2009, and indeed had not even

been drafted by September 2003. The patentee could not have been in possession of a system employing a standard that did not, at the time, exist. And the patentee made clear that provisional application No. 60/507,022, on which the claimed priority date is premised discloses only the then-available 802.11b, 802.11g, and 802.11a standards:

> Uses existing technology combined with this unique application. Readily available network technology. Utilizes 802.11b, 802.11g or 802.11a in the present embodiment.

09/29/2003 Specification.

Although it disputes the alleged priority date, Southwest nevertheless applies September 29, 2003, as the priority date for the purpose of these Preliminary Invalidity Contentions, as the Asserted Patent is invalid under 35 U.S.C. Sections 102 and 103 even if it were entitled to an artificially early date of priority. To the extent that Intellectual Ventures modifies its alleged priority date, Southwest may respond and challenge that date to the extent required by law to satisfy its burden.

B. Identification of Prior Art that Anticipates or Renders Obvious the Asserted Claims of the '469 Patent

Southwest contends that the prior art references listed below and charted in Exhibits 469-1 – 469-13 anticipate and/or render obvious, alone or in combination, one or more of the asserted claims of the '469 Patent.

Prior Art Reference	Publication Date
Airpath Wireless Hot Spot	At least by June 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a) and (g).
U.S. Patent No. 6,894,990 ("Agarwal")	Filed – October 13, 2000 Issued – May 17, 2005

Prior Art Reference	Publication Date
	This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No. 2002/0138625 ("Bruner")	Filed – March 21, 2001 Published – September 26, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a), § 102(b) and § 102(e).
U.S. Patent No. 6,445,777 ("Clark")	Filed – December 21, 1998 Issued – September 3, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a), (b), (e) and (g).
Connexion by Boeing	At least by June 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a), (b) and (g).
DeSanctis, et al., "Aeronautical Communications for Personal and Multimedia Services via Satellite," AM. INST. OF AERONAUTICAL. AND ASTRONAUTICS ("DeSanctis")	Published at least by April 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a), (b), and (g).
Jahn, et al., "Dimensioning of Aeronautical Satellite Services," AM. INST. OF AERONAUTICAL. AND ASTRONAUTICS ("Jahn I")	Published at least by October 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a) and (g).
Jahn, et al., "Passenger Multimedia Service Concept via Future Satellite Systems" DGLR JAHRBUCH ("Jahn II")	Published – 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a), (b) and (g).
U.S. Patent Application No. 2003/0051041 ("Kalavade")	Filed – August 6, 2002 Published – March 13, 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a), (b), (e), and (g).
Maritime Telecommunications Network Cruise Ship Internet Services	At least by June 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a) and (g).

Prior Art Reference	Publication Date
U.S. Patent Application No. 2007/0008937 ("Mody")	Filed – July 21, 2004 Priority – July 22, 2003 Published – January 11, 2007 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a), (e), and (g).
U.S. Patent Application No. 2003/0055975 ("Nelson")	Filed – June 19, 2001 Published – March 28, 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a), (e) and (g).
U.S. Patent Application No. 2004/0109449 ("Seo")	Filed – August 15, 2003 Published – June 10, 2004 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a), (e), and (g).
Suwanateep, Manasuda, "Inflight Internet Connectivity Technology and Its Application on E-Commerce," U. OF THAILAND (2002) ("Suwanateep")	Published at least by July 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a), (b), and (g).
Tenzing Global	At least by April 2001 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a), (b), and (g).
Wireless Cabin	At least by February 9, 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a) and (g).
U.S. Patent Application Publication No. 2003/0050041 ("Wu")	Filed – September 7, 2001 Published – March 13, 2003 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a), (e), and (g).

Plaintiffs assert claims 24, 25, 26, 28, and 32 of the '469 Patent against Southwest in this lawsuit. These claims are invalid because the '469 Patent fails to meet one or more of the

requirements for patentability. The individual bases for invalidity are provided below and in Exhibits 469-1 – 469-13. Each of the prior art references, the underlying work, and/or the underlying apparatus or method qualifies as prior art under one or more sections of Pre-AIA 35 U.S.C. §§ 102 and/or 103.

In an effort to focus the issues, Southwest has cited representative portions of identified references, even where a reference may contain additional disclosures or support for a particular claim element. Each and every disclosure of the same limitation in the same reference is not necessarily identified. In addition, persons of ordinary skill in the art generally read a prior art reference as a whole and in the context of other publications and literature and in the context of their knowledge as persons of ordinary skill in the art. Thus, to understand and interpret any specific statement or disclosure within a prior art reference, such persons would rely on other information within the reference, along with other publications and their general scientific knowledge and experience in the field. Thus, each prior art reference identified herein should be understood in the context of the reference itself in conjunction with anything incorporated by reference or described as relevant technology by the reference, any documents that are referred to for further understanding by the reference, any product embodying the reference, and any testimony by those with knowledge of the reference, such as the named authors and inventors. Southwest may rely upon uncited portions of the prior art references and on other publications and expert testimony to provide context, and as aids to understanding and interpreting the portions that are cited. Southwest may also rely on uncited portions of the prior art references, other disclosed publications, and the testimony of experts to establish that a person of ordinary skill in the art would have been motivated to modify or combine certain of the cited references so as to render the claims obvious.

Some or all of the asserted claims of the ‘469 Patent are invalid as anticipated under Pre-AIA 35 U.S.C. § 102 in view of certain ones of the prior art references identified above as detailed in the claim charts included in Exhibits 469-1 – 469-13, which identify specific examples of where each limitation of the asserted claims of the ‘469 Patent is found in the prior art references. As explained above, the cited portions of prior art references identified in the attached claim charts are exemplary in nature and representative of the content and teaching of the prior art references, and should be understood in the context of the reference as a whole and as they would be understood by a person of ordinary skill in the art.

To the extent any item of prior art cited above is deemed not to disclose, explicitly or inherently, any limitation of an asserted claim of the ‘469 Patent, Southwest reserves the right to argue that any difference between that prior art and the corresponding patent claim would have been either inherent in the art or obvious to a person of ordinary skill in the art, even if Southwest has not specifically denoted that the art is to be combined with the knowledge of a person of ordinary skill in the art.

To the extent Plaintiffs argue that any limitation of the asserted claims of the ‘469 Patent is not explicitly disclosed by an item of prior art identified above and in Exhibits 469-1 – 469-13, any purported differences are such that the claimed subject matter as a whole would have been obvious to one skilled in the art at the time of the alleged invention, in view of the state of the art and knowledge of those skilled in the art. The item of prior art would, therefore, render the relevant claims invalid as obvious under Pre-AIA 35 U.S.C. § 103.

To the extent any of the foregoing references are found not to anticipate the asserted claims of the ‘469 Patent, the foregoing references render the asserted claims obvious, either alone or in combination with one or more of the other references identified above. One of ordinary skill in the

art would have been motivated to combine the various references cited herein so as to disclose each and every limitation of the asserted claims of the '469 Patent.

The prior art identified in Exhibits 469-1 – 469-13 renders one or more asserted claims of the '469 Patent obvious when the references are read alone or in combination with one or more of each other and/or when read in view of the state of the art and knowledge of those skilled in the art.

To the extent that Plaintiff argues that any asserted claim of the '469 Patent is not disclosed or rendered obvious by any reference charted in Exhibits 469-1 – 469-13, the asserted claims nevertheless would have been obvious in light of one or more other references charted in Exhibits 469-1 – 469-13 and in view of the knowledge, experience, and understanding of persons of ordinary skill in the art.

Each and every reference identified is also relevant to the state of the art at the time of the alleged invention. Any of the references disclosed above may be combined with one another to render obvious (and therefore invalid) each of the asserted claims of the '469 Patent. Southwest may rely upon a subset of the above identified references, or all of the references identified above, including all references charted in Exhibits 469-1 – 469-13, for purposes of obviousness depending on the Court's claim construction, positions taken by Plaintiffs during this litigation, and further investigation and discovery.

1. Example Combinations of the Prior Art

The table below provides an identification of exemplary combinations of prior art references that, based on Southwest's present understanding of Plaintiff's Infringement Contentions and apparent reading of the claims, render obvious one or more of the asserted claims of the '469 Patent. The combinations of references identified below are non-exhaustive, and additional examples are apparent from other combinations of references charted in Exhibits 469-1

– 469-13. Although certain references charted in Exhibits 469-1 – 469-13 are listed variously in the left or right column below, Southwest does not thereby imply any restriction as to which reference is primary and which reference is secondary in such a combination. Southwest reserves the right to rely on either identified reference in the identified combinations as a primary and/or secondary reference.

Exhibit	Reference	Combination Reference(s)
469-1	Agarwal	Airpath Wireless Hot Spot, Bruner, Clark, Connexion by Boeing, DeSanctis, Jahn II, Kalavade, Mody, MTN Cruise Ship Internet Services, Nelson, Seo, Suwanateep, Tenzing Global, Wireless Cabin, Wu, and/or one or more references as charted in Exhibits 469-1 – 469-13.
469-2	Airpath Wireless Hot Spot	Agarwal, Bruner, Clark, Connexion by Boeing, DeSanctis, Jahn II, Kalavade, Mody, MTN Cruise Ship Internet Services, Nelson, Seo, Suwanateep, Tenzing Global, Wireless Cabin, Wu, and/or one or more references as charted in Exhibits 469-1 – 469-13.
469-3	Bruner	Airpath Wireless Hot Spot, Agarwal, Clark, Connexion by Boeing, DeSanctis, Jahn II, Kalavade, Mody, MTN Cruise Ship Internet Services, Nelson, Seo, Suwanateep, Tenzing Global, Wireless Cabin, Wu, and/or one or more other references as charted in Exhibits 469-1 – 469-13.
469-4	Clark	Airpath Wireless Hot Spot, Agarwal, Bruner, Connexion by Boeing, DeSanctis, Jahn II, Kalavade, Mody, MTN Cruise Ship Internet Services, Nelson, Seo, Suwanateep, Tenzing Global, Wireless Cabin, Wu, and/or one or more references as charted in Exhibits 469-1 – 469-13.

Exhibit	Reference	Combination Reference(s)
469-5	Connexion by Boeing	Airpath Wireless Hot Spot, Agarwal, Bruner, Clark, DeSanctis, Jahn II, Kalavade, Mody, MTN Cruise Ship Internet Services, Nelson, Seo, Suwanateep, Tenzing Global, Wireless Cabin, Wu, and/or one or more references as charted in Exhibits 469-1 – 469-13.
469-6	Jahn I	Airpath Wireless Hot Spot, Agarwal, Bruner, Clark, Connexion by Boeing, DeSanctis, Jahn II, Kalavade, Mody, MTN Cruise Ship Internet Services, Nelson, Seo, Suwanateep, Tenzing Global, Wireless Cabin, Wu, and/or one or more references as charted in Exhibits 469-1 – 469-13.
469-7	Jahn II	Airpath Wireless Hot Spot, Agarwal, Bruner, Clark, Connexion by Boeing, DeSanctis, Kalavade, Mody, MTN Cruise Ship Internet Services, Nelson, Seo, Suwanateep, Tenzing Global, Wireless Cabin, Wu, and/or one or more references as charted in Exhibits 469-1 – 469-13.
469-8	Nelson	Airpath Wireless Hot Spot, Agarwal, Bruner, Clark, Connexion by Boeing, DeSanctis, Jahn II, Kalavade, Mody, MTN Cruise Ship Internet Services, Seo, Suwanateep, Tenzing Global, Wireless Cabin, Wu, and/or one or more references as charted in Exhibits 469-1 – 469-13.
469-9	Maritime Telecommunications Network Cruise Ship Internet Services	Airpath Wireless Hot Spot, Agarwal, Bruner, Clark, Connexion by Boeing, DeSanctis, Jahn II, Kalavade, Mody, Nelson, Seo, Suwanateep, Tenzing Global, Wireless Cabin, Wu, and/or one or more references as charted in Exhibits 469-1 – 469-13.
469-10	Suwanateep	Airpath Wireless Hot Spot, Agarwal, Bruner, Clark, Connexion by Boeing, DeSanctis, Jahn II, Kalavade, Mody, MTN Cruise Ship Internet Services,

Exhibit	Reference	Combination Reference(s)
		Nelson, Seo, Tenzing Global, Wireless Cabin, Wu, and/or one or more references as charted in Exhibits 469-1 – 469-13.
469-11	Tenzing Global	Airpath Wireless Hot Spot, Agarwal, Bruner, Clark, Connexion by Boeing, DeSanctis, Jahn II, Kalavade, Mody, MTN Cruise Ship Internet Services, Nelson, Seo, Suwanateep, Wireless Cabin, Wu, and/or one or more references as charted in Exhibits 469-1 – 469-13.
469-12	Wireless Cabin	Airpath Wireless Hot Spot, Agarwal, Bruner, Clark, Connexion by Boeing, DeSanctis, Jahn II, Kalavade, Mody, MTN Cruise Ship Internet Services, Nelson, Seo, Suwanateep, Tenzing Global, Wu, and/or one or more references as charted in Exhibits 469-1 – 469-13.
469-13	Wu	Airpath Wireless Hot Spot, Agarwal, Bruner, Clark, Connexion by Boeing, DeSanctis, Jahn II, Kalavade, Mody, MTN Cruise Ship Internet Services, Nelson, Seo, Suwanateep, Tenzing Global, Wireless Cabin, and/or one or more references as charted in Exhibits 469-1 – 469-13.

It will be understood that, given the similarities between the art, any of Agarwal, Airpath Wireless Hot Spot, Bruner, Clark, Connexion by Boeing, Jahn I, Jahn II, Nelson, Maritime Telecommunications Network Cruise Ship Internet Services, Suwanateep, Tenzing Global, Wireless Cabin, and/or Wu could be combined with one or more of each other, for example based on the disclosures shown in Exhibits 469-1 – 469-13. It would have been obvious to combine these references for reasons that are readily apparent to one of ordinary skill in the art, and for reasons explained in additional detail in the sections that follow.

2. The Claims Recite Well-Known Concepts that Would Have Been Obvious to Implement

As noted above and *supra* II.F the asserted claims of the '469 Patent represent nothing more than an arrangement of well-known concepts and features and add nothing to the state of the art at the time of the alleged invention. For each of the concepts represented by the claims, it would have been inherently obvious to add to the teachings of any number of known references and products, including to any of the primary references charted in Exhibits 469-1 – 469-13 to the extent that such reference is alleged to be missing such concept or feature.

3. Additional Reasons to Combine

In addition to the information contained elsewhere in these invalidity contentions, including in Exhibits 469-1 – 469-13, Southwest has identified additional motivations and reasons to combine the various references. Southwest notes that in *KSR Int'l Co. v. Teleflex Inc.*, the Supreme Court held that "[i]n determining whether the subject matter of a patent claim is obvious, neither the particular motivation nor the avowed purpose of the patentee controls. What matters is the objective reach of the claim. If the claim extends to what is obvious, it is invalid under § 103." 550 U.S. 398, 419 (2007).

KSR further illustrated several ways in which the subject matter of a patent claim may be shown to be obvious. For instance, "[o]ne of the ways in which a patent's subject matter can be proved obvious is by noting that there existed at the time of invention a known problem for which there was an obvious solution encompassed by the patent's claims." *KSR*, 127 S.Ct. 1727, 1742. The Supreme Court held that it was error to "look only to the problem the patentee was trying to solve." *Id*. Rather, "[t]he question is not whether the combination was obvious to the patentee but whether the combination was obvious to a person with ordinary skill in the art. Under the correct analysis, any need or problem known in the field of endeavor at the time of invention and addressed

by the patent can provide a reason for combining the elements in the manner claimed." *Id*. Further, "[c]ommon sense teaches … that familiar items may have obvious uses beyond their primary purposes, and in many cases a person of ordinary skill will be able to fit the teachings of multiple patents together like pieces of a puzzle." *Id.* Further, a showing that a combination of elements was "obvious to try" may show that it was obvious under § 103. *Id.* For instance, the Supreme Court held that "predictable solutions" using a combination of "known options" may render the subject matter of a patent claim obvious.

When there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product is not of innovation but of ordinary skill and common sense. *Id.* The ways in which the subject matter of a patent claim may be shown to be obvious, identified by *KSR*, are merely illustrative. The main thrust of *KSR* was that "[r]igid preventative rules that deny factfinders recourse to common sense"—such as the overturned "teaching, suggestion, or motivation" test—are neither necessary under our case law nor consistent with it." *KSR*, 127 S.Ct. 1727, 1742-43; citing with approval, *e.g.*, *DyStar Textilfarben GmbH & Co. Deutschland KG v. C.H. Patrick Co.*, 464 F. 3d 1356, 1367 (2006) ("Our suggestion test is in actuality quite flexible and not only permits, but requires, consideration of common knowledge and common sense"); *Alza Corp. v. Mylan Labs., Inc.*, 464 F. 3d 1286, 1291 (2006) ("There is flexibility in our obviousness jurisprudence because a motivation may be found implicitly in the prior art. We do not have a rigid test that requires an actual teaching to combine...").

Although Southwest notes that there is no longer a rigid requirement regarding motivation to combine under *KSR*, Southwest provides the following statements regarding motivation to

combine. Multiple teachings, suggestions, and/or reasons to modify any of the references and/or to combine any two or more of the references charted in Exhibits 469-1 – 469-13 come from many sources, including the prior art (specific and as a whole), common knowledge, common sense, predictability, expectations, industry trends, design incentives or need, market demand or pressure, market forces, obvious to try, the nature of the problem faced, and/or knowledge possessed by one of ordinary skill. In addition, it would have been obvious to try combining the prior art references identified above because there were only a finite number of predictable solutions and/or because known work in one field or endeavor prompted variations based on predictable design incentives and/or market forces either in the same field or a different one.

The combination of prior art references identified in these contentions would have been obvious because the combination represents the known potential options with a reasonable expectation of success. Additionally, one of ordinary skill in the art would have been motivated to create combinations identified in these contentions using: known methods to yield predictable results; known techniques in the same way; a simple substitution of one known, equivalent element for another to obtain predictable results; and/or teaching, suggestion, or motivation in the prior art generally. Also, market forces in the industry, and the desire to improve features and performance, would motivate the addition of features to products as they become available, become less expensive, become more commonly used, provide better performance, reduce costs, or predictably achieve other clearly desirable results.

Moreover, in addition to the information contained elsewhere in these contentions, and, in particular, charted in Exhibits 469-1 – 469-13, Southwest has identified additional motivations and reasons to combine the various references cited herein. In particular, multiple teachings, suggestions, and/or reasons to modify any of the references and/or to combine any two or more of

the references charted in Exhibits 469-1 – 469-13 come from many sources, including the prior art (specific and as a whole), common knowledge, common sense, predictability, expectations, industry trends, design incentives or need, market demand or pressure, market forces, obvious to try, the nature of the problem faced, and/or knowledge possessed by one of ordinary skill. In addition, it would have been obvious to try combining the prior art references identified above because there were only a finite number of predictable solutions and/or because known work in one field or endeavor prompted variations based on predictable design incentives and/or market forces either in the same field or a different one. The combination of prior art references identified in these contentions would have been obvious because the claimed combinations represent the known potential options with a reasonable expectation of success.

To the extent that Plaintiffs contends that any of the above-identified references does not expressly or inherently disclose a particular element or limitation, it would have been obvious to a person of ordinary skill in the art to modify the references or to combine their teachings with other prior art references, including the prior art references in these invalidity contentions, in a manner that renders such claims obvious and invalid.

For example, many of the prior art references charted in Exhibits 469-1 – 469-13 share a goal of providing Internet users with airborne, seaborne, or otherwise remote access to the Internet by way of their own PDAs or laptops. Various references identified in these Preliminary Infringement Contentions disclose, well before the alleged critical date of the '469 patent, installation of paid or otherwise access-limited Internet hotspots in remote locations for users who desire access "[w]hen away from home or office." Wu at [0003]; *id.* at [0004] ("Over recent years, the field of wireless communications has enjoyed tremendous growth.").

For example, Wu discloses "a network system capable of providing prepaid wireless remote access service" which "obtains customer-inputted identification information related to the prepaid account" using an "authentication client." Wu at [0001], [0009], [0033] (noting that "virtually any type of prepaid arrangement or prepayment system may be effected" including, in an alternative embodiment, that funds be "deducted immediately and electronically from a credit card, a debit card, a check card, a charge account, or other forms of remote payment systems." Wu further teaches that subscriber access could be easily handled by "a commercially available device such as the ISS-2000 Internet Subscriber Server manufactured by Handlink Technology, Inc." Wu at [0043].

As another example, Bruner discloses "an airline-based system for providing electronic commerce to passengers during flights" and that "it will be understood by those skilled [sic] the art that the invention is generally applicable to any remote vehicle or building including aircraft, ships, trains and hotels." Bruner at [0001]. According to Bruner, "airborne users 13 are connected to the system via the aircraft based user terminals which are connected to an airline-based remote server computer" by way of an Airborne Data Server (ADS).

As another example, Kalavade teaches the use of commercially-available client access servers "at the hotspot, such as, e.g., a Cisco Router or a Nokia Access controller intercepts the session and sends a pre-defined web page to the user. This typically queries the user for a login and a password . . . Once authenticated, the RADIUS server responds with an Access Response to the RADIUS client." Kalavade at [0056], [0207].

As another example, Jahn I and Jahn II disclose a system for "travellers in aircraft on the move" including "wireless access segments in the aircraft cabin, namely a wireless LAN according to IEEE 802.11b standard," a "satellite segment for interconnection of the cabin with the terrestrial

telecom networks," and on-board service integration. Jahn II at 2-3. Jahn explains that, "since IP traffic at IP connection level can be monitored only before the IS stream multiplexing, it is envisaged that AAA functions must be located in the cabin to monitor for instance, volume-based accounting." *Id*. at 8.

As set forth above and *supra* II.F, the market and industry for Internet hotspots was well established and known by one of ordinary skill in the art. A person of ordinary skill would know, for example, to consider any feature of remote Internet hotspot designs already implemented by major industry players, including systems such as Airpath Hot Spot, Connexion by Boeing, or MTN for Cruise Ships. A person of ordinary skill would know that installation of an Internet hotspot could be achieved using "out of the box" hardware and software, and could be then improved upon or customized using any number of known techniques to achieve specific business objectives such as accepting credit cards or offering prepaid or subscription access to a commodity service.

* * *

In addition to the specific combinations of prior art and the specific combinations of groups of prior art disclosed above, Southwest reserves the right to rely on any other combination of any prior art references disclosed herein. Southwest further reserves the right to rely upon combinations of prior art references disclosed within the prosecution history of the references cited herein.

The references charted in Exhibits 469-1 – 469-13 disclose the elements of the asserted claims of the '469 Patent explicitly and/or inherently and may also be relied upon to show the state of the art at the time of the alleged invention. The suggested obviousness combinations are

provided in the alternative to Southwest's anticipation contentions and are not to be construed to suggest that any reference included in the combinations is not itself anticipatory.

C. OGP: Charts Setting Forth Where in the Prior Art References Each Element of the Asserted Claims Are Found

Pursuant to the Court's OGP, charts setting forth where specifically in each alleged item of prior art each element of each Asserted Claim of the '469 Patent is found are attached as Exhibits 469-1 – 469-13.

D. Other Relevant Prior Art References

In addition to the references identified above and charted in Exhibits 469-1 – 469-13, each of the references below is relevant to the '469 Patent, as it relates to the state of the art. Southwest reserves the right to amend its contentions to rely on any of the below references alone or in combination with any other references identified in these contentions.

Inventor/Author	Country	Issue/Publication No.	Filed Date
B. Reilly Barry et al.	US	6,615,258	September 24, 1998
Burt Liebowitz et al.	US	5,757,784	January 4, 1996
David D. Otten et al.	US	7,174,127	September 17, 2002
Jeffrey A. Frisco et al.	US	6,208,307	April 7, 2000
Jeyhan Karaoguz et al.	US	6,885,859	December 9, 2002
John Howard et al.	US	2003/0204610	April 30, 2003
John Thomasson et al.	US	2001/0023435	December 20, 2000
Kazuhiro Hara et al.	US	6,560,221	March 6, 1998
Khalid Monir A. El-Rafie et al.	US	6,968,394	March 21, 2000
Michael Massey	US	2001/0026537	February 23, 2001
Nikhil M. Deshpande et al.	US	2002/0176579	May 24, 2001
Patrick Meenan et al.	US	7,283,505	October 31, 2002
Per Wahlberg	US	2004/0137841	April 19, 2002
Robert Davis et al.	US	2002/0154055	April 15, 2002
Selim Rakib	US	2002/0031120	July 3, 2001
Shailendra Depura et al.	US	2003/0084051	July 26, 2002
Yi Gong et al.	US	2005/0032531	August 6, 2003
Conforto et al., "Suited/GMBS System: Architecture and Mobile Terminal"			January 2001

Inventor/Author	Country	Issue/Publication No.	Filed Date
Holzbock et al., "Land-Mobile Demonstration of Multimedia IP Services in Multi-Segment Access Networks," AM. INST. OF AERONAUTICAL. AND ASTRONAUTICS			At least by May 12, 2002
Holzbock et al., "Evolution of Aeronautical Communications for Personal and Multimedia Services" IEEE COMMS. MAGAZINE			July 2003
Jovanovic et al., "Joint Tactical Radio - Aspects of Standardization," SCIENTIFIC-TECHNICAL REVIEW, vol. III, no. 2			At least by July 2003
Martin, "Communication Satellites (1958-1992)," THE AEROSPACE CORPORATION			At least by December 31, 1991

IX. THE '582 PATENT IS INVALID UNDER 35 U.S.C. §§ 102 AND/OR 103

A. The '582 Patent Priority Date

Plaintiff's Infringement Contentions allege the asserted claims of the '582 Patent have a filing date of March 13, 2002.

B. Identification of Prior Art that Anticipates or Renders Obvious the Asserted Claims of the '582 Patent

Southwest contends that the prior art references listed below and charted in Exhibits 582-1 – 582-6 anticipate and/or render obvious, alone or in combination, one or more of the asserted claims of the '582 Patent.

Prior Art Reference	Publication Date
U.S. Patent No. 6,304,866 ("Chow")	Filed – June 27, 1997 Issued – October 16, 2001

Prior Art Reference	Publication Date
	This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No. 2002/0019844 ("Kurowski")	Filed – January 12, 2001 Published – February 14, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent No. 6,330,583 ("Reiffin")	Filed – September 9, 1994 Issued – December 11, 2001 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent No. 6,212,617 ("Hardwick")	Filed – June 30, 1998 Issued – April 3, 2001 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No. 2002/0103889 ("Markson")	Filed – June 19, 2001 Published – August 1, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent Application Publication No. 2002/0091786 ("Yamaguchi")	Filed – November 1, 2001 Published – July 11, 2002 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(e).
U.S. Patent No. 5,765,146 ("Wolf")	Filed – November 4, 1993 Issued – June 9, 1998 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b) and § 102(e).
EP Patent Application Publication No. EP0377993 ("Lorie")	Published – July 18, 1990 This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b).
P. Maheshwari, "Improving task scheduling for larger grain execution of parallel functional	Published January 8, 1993

Prior Art Reference	Publication Date
programs," [1993] Proceedings of the Twenty-sixth Hawaii International Conference on System Sciences, Wailea, HI, USA, 1993, pp. 594-602 vol.2, doi: 10.1109/HICSS.1993.284066. ("Maheshwari")	This reference is prior art under at least Pre-AIA 35 U.S.C. § 102(b). Retrieved from https://ieeexplore.ieee.org/document/284066
Gamma System - University of Wisconsin-Madison	At least as early as 1990 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a) or (g).
Volcano Parallel Database System - University of Wisconsin-Madison	At least as early as 1990 This reference is prior art under at least Pre-AIA 35 U.S.C. §§ 102(a) or (g).

Plaintiffs assert claims 1, 2, 5, and 7 of the '582 Patent against Southwest in this lawsuit. These claims are invalid because the '582 Patent fails to meet one or more of the requirements for patentability. The individual bases for invalidity are provided below and in Exhibits 582-1 – 582-6. Each of the prior art references, the underlying work, and/or the underlying apparatus or method qualifies as prior art under one or more sections of Pre-AIA 35 U.S.C. §§ 102 and/or 103.

In an effort to focus the issues, Southwest has cited representative portions of identified references, even where a reference may contain additional disclosures or support for a particular claim element. Each and every disclosure of the same limitation in the same reference is not necessarily identified. In addition, persons of ordinary skill in the art generally read a prior art reference as a whole and in the context of other publications and literature and in the context of their knowledge as persons of ordinary skill in the art. Thus, to understand and interpret any specific statement or disclosure within a prior art reference, such persons would rely on other information within the reference, along with other publications and their general scientific

knowledge and experience in the field. Thus, each prior art reference identified herein should be understood in the context of the reference itself in conjunction with anything incorporated by reference or described as relevant technology by the reference, any documents that are referred to for further understanding by the reference, any product embodying the reference, and any testimony by those with knowledge of the reference, such as the named authors and inventors. Southwest may rely upon uncited portions of the prior art references and on other publications and expert testimony to provide context, and as aids to understanding and interpreting the portions that are cited. Southwest may also rely on uncited portions of the prior art references, other disclosed publications, and the testimony of experts to establish that a person of ordinary skill in the art would have been motivated to modify or combine certain of the cited references so as to render the claims obvious.

Some or all of the asserted claims of the '582 Patent are invalid as anticipated under Pre-AIA 35 U.S.C. § 102 in view of certain ones of the prior art references identified above as detailed in the claim charts included in Exhibits 582-1 – 582-6, which identify specific examples of where each limitation of the asserted claims of the '582 Patent is found in the prior art references. As explained above, the cited portions of prior art references identified in the attached claim charts are exemplary in nature and representative of the content and teaching of the prior art references, and should be understood in the context of the reference as a whole and as they would be understood by a person of ordinary skill in the art.

To the extent any item of prior art cited above is deemed not to disclose, explicitly or inherently, any limitation of an asserted claim of the '582 Patent, Southwest reserves the right to argue that any difference between that prior art and the corresponding patent claim would have been either inherent in the art or obvious to a person of ordinary skill in the art, even if Southwest

has not specifically denoted that the art is to be combined with the knowledge of a person of ordinary skill in the art.

To the extent Plaintiffs argue that any limitation of the asserted claims of the '582 Patent is not explicitly disclosed by an item of prior art identified above and in Exhibits 582-1 – 582-6, any purported differences are such that the claimed subject matter as a whole would have been obvious to one skilled in the art at the time of the alleged invention, in view of the state of the art and knowledge of those skilled in the art. The item of prior art would, therefore, render the relevant claims invalid as obvious under Pre-AIA 35 U.S.C. § 103.

To the extent any of the foregoing references are found not to anticipate the asserted claims of the '582 Patent, the foregoing references render the asserted claims obvious, either alone or in combination with one or more of the other references identified above. One of ordinary skill in the art would have been motivated to combine the various references cited herein so as to disclose each and every limitation of the asserted claims of the '582 Patent.

The prior art identified in Exhibits 582-1 – 582-6 renders one or more asserted claims of the '582 Patent obvious when the references are read alone or in combination with one or more of each other and/or when read in view of the state of the art and knowledge of those skilled in the art.

To the extent that Plaintiff argues that any asserted claim of the '582 Patent is not disclosed or rendered obvious by any reference charted in Exhibits 582-1 – 582-6, the asserted claims nevertheless would have been obvious in light of one or more other references charted in Exhibits 582-1 – 582-6 and in view of the knowledge, experience, and understanding of persons of ordinary skill in the art.

Each and every reference identified is also relevant to the state of the art at the time of the alleged invention. Any of the references disclosed above may be combined with one another to render obvious (and therefore invalid) each of the asserted claims of the '582 Patent. Southwest may rely upon a subset of the above identified references, or all of the references identified above, including all references in charted in Exhibits 582-1 – 582-6, for purposes of obviousness depending on the Court's claim construction, positions taken by Plaintiffs during this litigation, and further investigation and discovery.

1. Example Combinations of the Prior Art

The table below provides an identification of exemplary combinations of prior art references that, based on Southwest's present understanding of Plaintiff's Infringement Contentions and apparent reading of the claims, render obvious one or more of the asserted claims of the '582 Patent. The combinations of references identified below are non-exhaustive, and additional examples are apparent from other combinations of references charted in Exhibits 582-1 – 582-6. Although certain references charted in Exhibits 582-1 – 582-6 are listed variously in the left or right column below, Southwest does not thereby imply any restriction as to which reference is primary and which reference is secondary in such a combination. Southwest reserves the right to rely on either identified reference in the identified combinations as a primary and/or secondary reference.

Exhibit	Reference	Combination Reference(s)
582-1	Chow	Reiffin, Kurowski, Yamaguchi, Wolf, Lorie, and/or Markson, and/or one or more references as charted in Exhibits 582-1 – 582-6
582-2	Reiffin	Chow, Kurowski, Yamaguchi, Wolf, Lorie, and/or Markson, and/or one or

Exhibit	Reference	Combination Reference(s)
		more references as charted in Exhibits 582-1 – 582-6
582-3	Hardwick	Reiffin, Kurowski, Yamaguchi, Wolf, Lorie, Chow, and/or Markson, and/or one or more references as charted in Exhibits 582-1 – 582-6
582-4	Maheshwari	Reiffin, Kurowski, Yamaguchi, Wolf, Lorie, Chow, and/or Markson, and/or one or more references as charted in Exhibits 582-1 – 582-6
582-5	Gamma System	Reiffin, Kurowski, Yamaguchi, Wolf, Lorie, Chow, and/or Markson, and/or one or more references as charted in Exhibits 582-1 – 582-6
582-6	Volcano Parallel Database System	Reiffin, Kurowski, Yamaguchi, Wolf, Lorie, Chow, and/or Markson, and/or one or more references as charted in Exhibits 582-1 – 582-6

It will be understood that, given the similarities between the art, any of Chow, Reiffin, Hardwick, Kurowski, Yamaguchi, Wolf, Lorie, Markson, Maheshwari, the Gamma System, and/or the Volcano Parallel Database System could be combined with one or more of each other, for example based on the disclosures shown in Exhibits 582-1 – 582-6. It would have been obvious to combine these references for reasons that are readily apparent to one of ordinary skill in the art, and for reasons explained in additional detail in the sections that follow.

2. The Claims Recite Well-Known Concepts that Would Have Been Obvious to Implement

As noted above, the asserted claims of the ‘582 Patent represent nothing more than an arrangement of well-known concepts and features and add nothing to the state of the art at the time of the alleged invention. For each of the concepts represented by the claims, it would have been inherently obvious to add to the teachings of any number of known references and products,

including to any of the primary references charted in Exhibits 582-1 – 582-6 to the extent that such reference is alleged to be missing such concept or feature.

3. Additional Reasons to Combine

In addition to the information contained elsewhere in these invalidity contentions, including in Exhibits 582-1 – 582-6, Southwest has identified additional motivations and reasons to combine the various references. Southwest notes that in *KSR Int'l Co. v. Teleflex Inc.*, the Supreme Court held that "[i]n determining whether the subject matter of a patent claim is obvious, neither the particular motivation nor the avowed purpose of the patentee controls. What matters is the objective reach of the claim. If the claim extends to what is obvious, it is invalid under § 103." 550 U.S. 398, 127 S.Ct. 1727, 1742 (2007).

KSR further illustrated several ways in which the subject matter of a patent claim may be shown to be obvious. For instance, "[o]ne of the ways in which a patent's subject matter can be proved obvious is by noting that there existed at the time of invention a known problem for which there was an obvious solution encompassed by the patent's claims." *KSR*, 127 S.Ct. 1727, 1742. The Supreme Court held that it was error to "look only to the problem the patentee was trying to solve." *Id*. Rather, "[t]he question is not whether the combination was obvious to the patentee but whether the combination was obvious to a person with ordinary skill in the art. Under the correct analysis, any need or problem known in the field of endeavor at the time of invention and addressed by the patent can provide a reason for combining the elements in the manner claimed." *Id*. Further, "[c]ommon sense teaches … that familiar items may have obvious uses beyond their primary purposes, and in many cases a person of ordinary skill will be able to fit the teachings of multiple patents together like pieces of a puzzle." *Id.* Further, a showing that a combination of elements was "obvious to try" may show that it was obvious under § 103. *Id.* For instance, the Supreme

Court held that "predictable solutions" using a combination of "known options" may render the subject matter of a patent claim obvious.

When there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense. *Id.* The ways in which the subject matter of a patent claim may be shown to be obvious, identified by *KSR*, are merely illustrative. The main thrust of *KSR* was that "[r]igid preventative rules that deny factfinders recourse to common sense"—such as the overturned "teaching, suggestion, or motivation" test—are neither necessary under our case law nor consistent with it." *KSR*, 127 S.Ct. 1727, 1742-43; citing with approval, e.g., *DyStar Textilfarben GmbH & Co. Deutschland KG v. C.H. Patrick Co.*, 464 F. 3d 1356, 1367 (2006) ("Our suggestion test is in actuality quite flexible and not only permits, but requires, consideration of common knowledge and common sense"); *Alza Corp. v. Mylan Labs., Inc.*, 464 F. 3d 1286, 1291 (2006) ("There is flexibility in our obviousness jurisprudence because a motivation may be found implicitly in the prior art. We do not have a rigid test that requires an actual teaching to combine...").

Although Southwest notes that there is no longer a rigid requirement regarding motivation to combine under *KSR*, Southwest provides the following statements regarding motivation to combine. Multiple teachings, suggestions, and/or reasons to modify any of the references and/or to combine any two or more of the references charted in Exhibits 582-1 – 582-6 come from many sources, including the prior art (specific and as a whole), common knowledge, common sense, predictability, expectations, industry trends, design incentives or need, market demand or pressure, market forces, obvious to try, the nature of the problem faced, and/or knowledge possessed by one

of ordinary skill. In addition, it would have been obvious to try combining the prior art references identified above because there were only a finite number of predictable solutions and/or because known work in one field or endeavor prompted variations based on predictable design incentives and/or market forces either in the same field or a different one.

The combination of prior art references identified in these contentions would have been obvious because the combination represents the known potential options with a reasonable expectation of success. Additionally, one of ordinary skill in the art would have been motivated to create combinations identified in these contentions using: known methods to yield predictable results; known techniques in the same way; a simple substitution of one known, equivalent element for another to obtain predictable results; and/or teaching, suggestion, or motivation in the prior art generally. Also, market forces in the industry, and the desire to improve features and performance, would motivate the addition of features to products as they become available, become less expensive, become more commonly used, provide better performance, reduce costs, or predictably achieve other clearly desirable results.

Moreover, in addition to the information contained elsewhere in these contentions, and, in particular, charted in Exhibits 582-1 – 582-6, Southwest has identified additional motivations and reasons to combine the various references cited herein. In particular, multiple teachings, suggestions, and/or reasons to modify any of the references and/or to combine any two or more of the references charted in Exhibits 582-1 – 582-6 come from many sources, including the prior art (specific and as a whole), common knowledge, common sense, predictability, expectations, industry trends, design incentives or need, market demand or pressure, market forces, obvious to try, the nature of the problem faced, and/or knowledge possessed by one of ordinary skill. In addition, it would have been obvious to try combining the prior art references identified above

because there were only a finite number of predictable solutions and/or because known work in one field or endeavor prompted variations based on predictable design incentives and/or market forces either in the same field or a different one. The combination of prior art references identified in these contentions would have been obvious because the claimed combinations represent the known potential options with a reasonable expectation of success.

To the extent that Plaintiffs contend that any of the above-identified references does not expressly or inherently disclose a particular element or limitation, it would have been obvious to a person of ordinary skill in the art to modify the references or to combine their teachings with other prior art references, including the prior art references in these invalidity contentions, in a manner that renders such claims obvious and invalid.

In particular, various references identified above disclose, well before the critical date of the '582 Patent, distributed computing systems that already solved the '582 Patent's goal of decomposing linear computational processes that currently use a single computer into equivalent parallel processes that can use any number of heterogeneous computers.

For example, Chow discloses distributed multi-processing systems and database operations. Chow at 2:42-46. Chow's disclosure provides a multiprocessing system "to perform an aggregate task as quickly as possible, with minimum workload skew among independently operating processing elements." *See* Chow at 2:1-7.

As another example, Reiffin discloses distributed multi-processing systems wherein a large compute-intensive task may be partitioned into subtasks for parallel execution. Reiffin at 1:7-14. Reiffin provides that "a large compute-intensive task may be partitioned into subtasks which are then distributed among a plurality of personal computer or workstations for parallel execution of the subtasks." *See* Reiffin at 1:9-12.

As another example, Kurowski discloses large-scale network-distributed computing systems. Kurowski at [0003]. Kurowski's disclosure provides "[a] distributed computing system achieves a highly distributed environment where very large computation intensive tasks are broken down into thousands of sub-tasks and then distributed to thousands of clients running on a variety of computers across the Internet." *See* Kurowski at Abstract.

As yet another example, Hardwick discloses computer-executable processes following a divide-and-conquer algorithm where the input file is derived from the storage unit and the input is then logically distributed or shared into partitions. The partitions are then distributed to the subtask processors for concurrent processing. *See* Hardwick at Abstract, 10:35-44, claim 1.

As yet another example, Maheshwari discloses a parallel processing system that uses concurrent task partitions, and data is processed in a first-in-first-out (FIFO) manner. *See* Maheshwari at 2-3.

As yet another example, the Gamma System was a parallel relational database system developed at the University of Wisconsin-Madison in the late 1980s. It embodied many concepts that later influenced systems like MapReduce, particularly in how it handled partitioned parallelism for query execution. *See, e.g.*, https://apps.dtic.mil/sti/pdfs/ADA223436.pdf. The Gamma System subdivided data for parallel processing. *See* https://minds.wisconsin.edu/bitstream/handle/1793/11036/file_1.pdf, https://www.cs.ubc.ca/~rap/teaching/504/2022W2/slides/parallel.pdf, https://minds.wisconsin.edu/bitstream/handle/1793/58708/TR635.pdf?sequence=1.

As yet another example, the Volcano Parallel Database System was a parallel database system designed in the late 1980s and early 1990s at the University of Wisconsin–Madison. It served as a foundation for many later research projects and commercial systems. Volcano is best

known not just as a parallel database, but also as an extensible and highly modular query execution engine. Volcano distributes tasks to partitions each with a subtask processor and processes subtasks in parallel. *See* https://paperhub.s3.amazonaws.com/dace52a42c07f7f8348b08dc2b186061.pdf, https://pages.cs.wisc.edu/~shivaram/cs744-readings/volcano.pdf, https://cs-people.bu.edu/mathan/reading-groups/papers-classics/volcano.pdf.

Various references identified above also disclose, well before the critical date of the '582 Patent, file allocation and partitioning, load balancing, using additional processors to carry out file allocation/partitioning and associated distribution, providing an accumulation of output records in an arbitrary order, and input files residing on network attached storage.

For example, Yamaguchi discloses a system in which a program is executed on each of several logical partitions, the load on each of the logical partitions is monitored, and when a logical partition is overloaded, data being distributed by the logical partition is automatically switched for distribution by another logical partition. *See* Yamaguchi at Abstract.

As another example, Wolf discloses using additional processors for performing distributed processing steps. *See* Wolf at 4:63-5:7.

As yet another example, Lorie discloses performing processing in random or arbitrary orders. *See* Lorie at 6:50-56.

As yet another example, Markson discloses a system which logically allocates storage and to the processors and also logically associates logical units from the storage units to those processors, and providing input files on network attached storage. *See* Markson at [0118].

A person of ordinary skill in the art would have been motivated to combine the above references to incorporate the different beneficial technical features provided by each of the above

references. Modifying the above references would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results.

* * *

In addition to the specific combinations of prior art and the specific combinations of groups of prior art disclosed above, Southwest reserves the right to rely on any other combination of any prior art references disclosed herein. Southwest further reserves the right to rely upon combinations of prior art references disclosed within the prosecution history of the references cited herein.

The references charted in Exhibits 582-1 – 582-6 disclose the elements of the asserted claims of the '582 Patent explicitly and/or inherently and may also be relied upon to show the state of the art at the time of the alleged invention. The suggested obviousness combinations are provided in the alternative to Southwest's anticipation contentions and are not to be construed to suggest that any reference included in the combinations is not itself anticipatory.

C. OGP: Charts Setting Forth Where in the Prior Art References Each Element of the Asserted Claims Are Found

Pursuant to the Court's OGP, charts setting forth where specifically in each alleged item of prior art each element of each Asserted Claim of the '582 Patent is found are attached as Exhibits 582-1 – 582-6.

D. Other Relevant Prior Art References

In addition to the references identified above and charted in Exhibits 582-1 – 582-6, each of the references below is relevant to the '582 Patent, as it relates to the state of the art. Southwest reserves the right to amend its contentions to rely on any of the below references alone or in combination with any other references identified in these contentions.

Inventor/Author	Country	Issue/Publication No.	Filed Date
Tuneo Iida et al.	US	6,334,137	August 24, 1999
Chad Grey Thompson	US	2002/0099716	January 25, 2001
Yuichi Aiba	JP	JP20003/30959	November 30, 2000
Dianne L. Steiger et al.	US	US2002/040381	September 28, 2001
Russ W. Herrell et al.	US	5,325,493	March 30, 1993
Christopher L. Reeve	US	5,535,393	June 5, 1995
Roberta L. Gotfried	US	5,765,166	June 9, 1998
S. Kuo, M. Winslett, Y. Chen, Y. Cho, M. Subramaniam and K. Seamons, "Parallel input/output with heterogeneous disks," Proceedings. Ninth International Conference on Scientific and Statistical Database Management (Cat. No.97TB100150), Olympia, WA, USA, 1997, pp. 79-90, doi: 10.1109/SSDM.1997.621154.			Pub. Aug. 11-13, 1997 Retrieved from https://ieeexplore.ieee.org/document/621154
David Morrison et al.	US	5,945,990	November 19, 1996
Shawn B. Smith et al.	WO	03/012696	July 29, 2002
James Mathew Rehg et al.	US	6,480,876	May 28, 1998
"Some Thoughts on Parallel Processing," Lynn D. Yarbough, Communications of the ACM, Vol. 3, Issue 10, page 539, October 1, 1960 (available at: https://dl.acm.org/doi/10.1145/367415.367426).			Oct. 1, 1960
"A Survey of Some Theoretical Aspects of Multiprocessing," J. L. Baer, ACM Computing Surveys (CSUR), Volume 5, Issue 1 Pages 31 – 80, March 1, 1973 (available at: https://dl.acm.org/doi/10.1145/356612.356615)			March 1, 1973
"Parallel Sorting Algorithms for Tightly Coupled Multiprocessors," Michael J.			1988

Inventor/Author	Country	Issue/Publication No.	Filed Date
Quinn, Parallel Computing 6 (1988) 349-357.			
"Removing Skew Effect in Join Operation on Parallel Processors," Ron-Chung Hu, Richard R. Muntz, Computer Science Technical Report, University of California, Los Angeles, June 1989.			June 1989
"Parallel Architectures for Database Systems," A.R. Hurson, L.L. Miller, S.H. Pakzad, M.H. Eich, B. Shirazi, Advances in Computers, Vol. 28, pages 107-151, 1989.			1989
"Parallel Algorithms for Merging and Sorting," Narsingh Deo, Dilip SarKar, Information Sciences, Vol. 56, pages 151-161, 1991.			1991
"Parallel Algorithms for lcast Median of Squares Regression," Chong-Wei Xu and Wei-Kei Shiue, Computational Statistics & Data Analysis, Vol. 16, pages 349-362, 1993.			1993
"On the Average Running Time of Odd-Even Merge Sort," Christine Rub, Max Plank-Institute Fur Informtik, Research Report, April 1995.			April 1995
"Parallel Algorithms," Guy E. Blelloch and Bruce M. Maggs, a chapter in "The Computer Science and Engineering Handbook." CRC Press, 1997, ISBN: 0-8493-2909-4. Available at https://dl.acm.org/doi/pdf/10.5555/1882723.1882748.			1997
"An Approach to Parallelizing Isotonic Regression," Anthony J. Kearsley, Richard A. Tapia, Michael W.Trosset, Applied			1996

Inventor/Author	Country	Issue/Publication No.	Filed Date
Mathematics and Parallel Computing, H. Fisher et. al. (eds.), Physica-Verlag Heidelberg (1996), pages 141-147.			
"Part 1: The Parallel Computing Environment, Alice E. Koniges, Morris A. Jette, and David C. Eder, in "Industrial Strength Parallel Computing," Morgan Kaufmann, (2000). ISBN 9781558605404. Available at: http://wayback.cecm.sfu.ca/PSG/book/intro.html and at https://books.google.com/books?id=mWalawBciCQC&pg=PA1&source=gbs_toc_r&cad=2#v=onepage&q&f=false.			2000
GPFS: A Parallel File System, Jason Barkes, Marcelo R. Barrios, Francis Cougard, Paul G. Crumley, Didac Marin, Hari Reddy, Theerapong Thitayanun, IBM International Technical Support Organization, April 1998.			April 1998
"Analysis of First-Come-First-Serve Parallel Job Scheduling," Uwe Schwiegelshohn and Ramin Yahyapour. Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms (1998). Available at http://dx.doi.org/10.1145/314613.315031.			1998
Bricker, et al., "Condor Technical Summary," University of Wisconsin – Madison Computer Sciences Department			Pub. Jan. 1992
W. Gentzsch, "Sun Grid Engine: Towards Creating a Compute Power Grid," Sun Microsystems, Inc.			2001

Inventor/Author	Country	Issue/Publication No.	Filed Date
O. Sievert, "A Gentle Introduction to Globus," Prepared for Parallel Computation, CSE160/CSE260, Spring 1999			1999
Anderson, et al., "SETI@home Internet Distributed Computing for SETI," ASP Conference Series, Vol. 213			2000
Bamford, et al., "Architecture of Oracle Parallel Server," Proceedings of the 24th Very Large Database (VLDB) Conference New York, USA, 1998			Pub. 1998
J. Howard, "An Overview of the Andrew File System," Carnegie Mellon University			1988
S. Hotovy, "Workload Evolution on the Cornell Theory Center IBM SP2" Cornell Theory Center			1996
Foster et al., "Wide-Area Implementation of the Message Passing Interface," Parallel Computing 24(12),1735-1749			1998
Heymann et al., "Adaptive Scheduling for Master-Worker Applications on the Computational Grid," Department of Computer Sciences University of Wisconsin – Madison			2000
D. Gelernter and N. Carriero, "How to Write Parallel Programs: A Guide to the Perplexed," *ACM Computing Surverys*, Vol. 21, No. 3			Sept. 1989

X. GROUNDS OF INVALIDITY UNDER 35 U.S.C. § 112

Defendants set forth indefiniteness contentions below, but reserve the right to revise, amend, and/or supplement these indefiniteness contentions as part of the claim construction

briefing process as provided for in the Court's scheduling order. The following is provided in a good faith effort to comply with the Court's scheduling order. These contentions shall not be deemed admissions, whether express or implied, regarding the scope of any claims, or the proper constructions of those claims or any terms recited in those claims. Nor do Defendants waive other defenses pursuant to 35 U.S.C. § 112.

A. U.S. Patent No. 8,332,844

Claims 7 and 11 of the 844 Patent (herein, the "Asserted '844 Claims") are invalid under 35 U.S.C. §§ 112(a) and 112(b), respectively, for at least the reasons discussed below.

The Asserted '844 Claims contain subject matter that was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the applicants, at the time the application was filed, had possession of the claimed invention. In addition, the Asserted '844 Claims contain subject matter that was not described in the specification in such a way as to enable one of ordinary skill in the art to make and/or use the invention. Further, the Asserted '844 Claims fail to particularly point out and distinctly claim the invention, such that the claim's scope cannot be ascertained with reasonable certainty.

To the extent that the Asserted 844 Claims are construed broadly to cover undisclosed embodiments, the claims are invalid because the '844 Patent specification does not enable a person skilled in the art to practice the full scope of the purported invention, nor does the 844 Patent specification show the applicants had within their possession the full scope of the Asserted '844 Claims. Further, to the extent that any of the Asserted '844 Claims are construed sufficiently broadly to cover one or more of the various accused products identified in IV's infringement contentions, such claims are invalid pursuant to at least 35 U.S.C. § 112(a).

Southwest contends that the following terms of asserted claims 7 and 11 are indefinite under 35 U.S.C. § 112(b), and/or lack written description support under 35 U.S.C. § 112(a):

1. "only additional data blocks not previously contained in said root image" (Claim 7)

Indefiniteness: This phrase requires that the leaf image contains "only additional" data blocks that were not previously in the root image. However, the patent does not provide a clear definition of what constitutes an "additional" block versus a modified block. If a block is modified rather than newly created, does it still count as "additional"? The claim does not specify how to determine whether a block was previously contained in the root image. For example: If a block has the same content but is stored in a different location, is it still "previously contained"? If a block is partially altered, does it become "additional," or is it still part of the original root image? Without objective boundaries, a POSA is unable to determine whether a system meets this limitation.

Written Description: The specification describes that leaf images store additional data blocks and modifications to the root image, but does not explicitly define "only additional." The term "only additional" appears to exclude modifications, yet the patent acknowledges that leaf images may store modifications.

1. "accessed by at least one of said compute nodes" (Claim 7)

Indefiniteness: This phrase refers to caching blocks that have been "accessed" by at least one compute node, but the term "accessed" is undefined and ambiguous. What type of access qualifies? Read operations only? Write operations? Execution of code within a block? Any attempted access, even if the operation is unsuccessful? The claim does not clarify whether a single access event is sufficient to trigger caching, or if caching occurs only after repeated access.

2. "merging the blocks of said root image with the blocks of respective leaf images" (Claim 11)

Indefiniteness: The claim does not specify what "merging" means in this context. Overwriting blocks in the root image with leaf image data? Appending new blocks without modifying existing ones? Interleaving blocks from the root and leaf images?

3. "to create cohesive respective application environments" (Claim 11)

Indefiniteness: What makes an application environment "cohesive"? Data consistency across all compute nodes? Performance efficiency? Uniform file structure? Without an objective test for determining whether an environment is "cohesive", a POSITA cannot reasonably ascertain the scope of this claim term.

Written Description: The specification never explains what a "cohesive" environment entails.

4. "at an operational level between file systems" (Claim 11)

Indefiniteness: What is meant by "operational level"? Kernel-level operations? Middleware interactions? File system APIs? Without a defined meaning, the phrase is open to multiple interpretations, making the claim scope uncertain.

Written Description: The specification never uses the phrase "operational level," and in the context of the claim, the phrase lacks written description support.

B. U.S. Patent No. 8,407,722

Claims 14, 16, 17 and 18 of the '722 Patent (herein, the "Asserted '722 Claims") are invalid under 35 U.S.C. §§ 112(a) and 112(b), respectively, for at least the reasons discussed below.

The Asserted '722 Claims contain subject matter that was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the applicants, at the time the application was filed, had possession of the claimed invention. In addition, the Asserted

‘722 Claims contain subject matter that was not described in the specification in such a way as to enable one of ordinary skill in the art to make and/or use the invention. Further, the Asserted ‘722 Claims fail to particularly point out and distinctly claim the invention, such that the claim’s scope cannot be ascertained with reasonable certainty.

To the extent that the Asserted ‘722 Claims are construed broadly to cover undisclosed embodiments the claims are invalid because the ‘722 Patent specification does not enable a person skilled in the art to practice the full scope of the purported invention, nor does the ‘722 Patent specification show the applicants had within their possession the full scope of the Asserted ‘722 Claims. Further, to the extent that any of the Asserted ‘722 Claims are construed sufficiently broadly to cover one or more of the various accused products identified in IV’s infringement contentions, such claims are invalid pursuant to at least 35 U.S.C. § 112(a).

Southwest contends that the following terms of asserted claims 14, 16, 17, and 18 are indefinite under 35 U.S.C. § 112(b), and/or lack written description support under 35 U.S.C. § 112(a):

1. “live object recognizable by the client device” (Claim 14)

Indefiniteness: The term “live object” is not explicitly defined in the specification. It is unclear whether “live” refers to real-time data streaming, frequent updates, or simply any data object that can be modified. It is also unclear how the system determines if an object is “live.” Is it based on update frequency, metadata, or another criterion? Additionally, the phrase “recognizable by the client device” is subjective and lacks objective boundaries. Does “recognizable” require predefined software compatibility? Does it mean the client must be capable of identifying the object’s format, or merely receive and process it in some capacity? Without clear technical definitions, a POSA cannot ascertain the exact scope of this term, making the claim indefinite.

Written Description: The specification discusses update messages and object registration, but does not clearly define "live object." If "live object" is meant to be dynamic content, there should be clear criteria for distinguishing it from static content.

2. "gateway device ... configured to identify a category of the update message" (Claim 14)

Indefiniteness: The claim requires the gateway device to "identify a category" of an update message, but the patent does not define what "category" means. Are categories predefined or dynamically generated? Is "category" based on data type, source, priority, or some other factor? Without an objective standard for categorization, a POSITA cannot determine whether a system meets this limitation.

3. "determine a node type to which the identified category maps" (Claim 14)

Indefiniteness: "Node type" is vague and undefined. Mapping a category to a node type suggests a predefined relationship, but there is no indication of how these relationships defined, stored, or modified.

Written Description: The patent describes a hierarchy of nodes, but it does not specify how categories and node types are mapped.

4. "cohesive respective application environments" (Claim 14)

Indefiniteness: The phrase "cohesive respective application environments" is subjective. What makes an application environment "cohesive"? Does "cohesive" require synchronized updates, consistent data availability, or a particular processing order?

Written Description: The specification never explicitly defines what a "cohesive" application environment is.

C. U.S. Patent No. 7,949,785

Claims 30, 35, and 37 of the '785 Patent (herein, the "Asserted '785 Claims") are invalid under 35 U.S.C. §§ 112(a) and 112(b), respectively, for at least the reasons discussed below.

The Asserted '785 Claims contain subject matter that was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the applicants, at the time the application was filed, had possession of the claimed invention. In addition, the Asserted '785 Claims contain subject matter that was not described in the specification in such a way as to enable one of ordinary skill in the art to make and/or use the invention. Further, the Asserted '785 Claims fail to particularly point out and distinctly claim the invention, such that the claim's scope cannot be ascertained with reasonable certainty.

To the extent that the Asserted '785 Claims are construed broadly to cover undisclosed embodiments, the claims are invalid because the '785 Patent specification does not enable a person skilled in the art to practice the full scope of the purported invention, nor does the '785 Patent specification show the applicants had within their possession the full scope of the Asserted '785 Claims. Further, to the extent that any of the Asserted '785 Claims are construed sufficiently broadly to cover one or more of the various accused products identified in IV's infringement contentions, such claims are invalid pursuant to at least 35 U.S.C. § 112(a).

Southwest contends that the following terms of asserted claims 30, 35, and 37 are indefinite under 35 U.S.C. § 112(b), and/or lack written description support under 35 U.S.C. § 112(a):

1. "virtual network manager" (Claim 30)

Indefiniteness: The term "virtual network manager" is not explicitly defined in the specification. Is it a physical device, a software system, or a combination of both? Is it a single entity or a collection of distributed components? How does it differ from existing network management systems?

Written Description: The specification discusses virtual networks and device registration, but does not clearly define or describe a "virtual network manager" as a distinct entity.

2. "receive a registration request from an agent associated with a device" (Claim 30)

Indefiniteness: The term "agent associated with a device" lacks clarity. What qualifies as an agent? How is association with the device determined?

3. "DNS server for the virtual network ... return a network address associated with a network route director" (Claim 30)

Indefiniteness: The phrase "network route director" is not well-defined. Is this a specific device, a logical function, or a dynamic routing protocol? How does it interact with standard routing mechanisms? The claim requires the DNS server to return multiple addresses, including **a** "network address" for the route director, a "private network address" for a second device, and a "virtual network address" for the second device. However, the patent does not specify these addresses are assigned, whether these addresses are static or dynamic, or how conflicts are resolved if multiple addresses exist.

Written Description: The specification does not adequately describe how a DNS server can return multiple types of addresses in a single response.

4. "maintain data to associate a virtual network address with a device in the virtual network" (Claim 37)

Indefiniteness: The term "maintain data" is ambiguous. What kind of data structure is used? Is this a persistent database, a temporary cache, or another storage method? Does this refer to active session tracking or long-term registration?

D. U.S. Patent No. 8,027,326

Claims 1, 4, and 18 of the '326 Patent (herein, the "Asserted '326 Claims") are invalid under 35 U.S.C. §§ 112(a) and 112(b), respectively, for at least the reasons discussed below.

The Asserted '326 Claims contain subject matter that was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the applicants, at the time the application was filed, had possession of the claimed invention. In addition, the Asserted '326 Claims contain subject matter that was not described in the specification in such a way as to enable one of ordinary skill in the art to make and/or use the invention under 35 U.S.C. § 112(a). Further, the Asserted '326 Claims fail to particularly point out and distinctly claim the invention, such that the claim's scope cannot be ascertained with reasonable certainty under 35 U.S.C. § 112(b).

To the extent that the Asserted '326 Claims are construed broadly to cover undisclosed embodiments the claims are invalid because the '326 Patent specification does not enable a person skilled in the art to practice the full scope of the purported invention, nor does the '326 Patent specification show the applicants had within their possession the full scope of the Asserted '326 Claims. Further, to the extent that any of the Asserted '326 Claims are construed sufficiently broadly to cover one or more of the various accused products identified in IV's infringement contentions, such claims are invalid pursuant to at least 35 U.S.C. § 112(a).

Southwest contends that the following terms of asserted claims 1, 4, and 18 are indefinite under 35 U.S.C. § 112(b), and/or lack written description support under 35 U.S.C. § 112(a):

1. "plurality of data subcarriers" (Claims 1 & 18)

Indefiniteness: The term "plurality" lacks definitiveness as to the amount of data subcarriers and therefore lacks a clear quantitative or objective standard. What does "plurality" mean? Why do claims 1 and 18 first recite a "plurality of data subcarriers," then only refer to "data subcarriers" without the "plurality" aspect? Are they trying to quantitatively differentiate between the "plurality of data subcarriers," versus mere "data subcarriers"?

Written Description: The specification does not provide an explanation of what constitutes a "plurality" of data subcarriers.

2. "partially filling the frequency gap" / "partially fill the frequency gap" (Claims 1 & 18)

Indefiniteness: The term "partially filling/fill" lacks a clear quantitative or objective standard. What percentage of the frequency gap must be filled for it to be considered "partial"? Does "partially" refer to uniformly distributed subcarriers or only some regions being filled?

Written Description: The specification does not provide an explanation of what constitutes "partially" filling/filled in a way that distinguishes it from fully filling/filled.

3. "one or more guard bands" (Claims 1 & 18)

Indefiniteness: The phrase "one or more guard bands" is vague. If a system uses multiple guard bands or different sizes of guard bands, does the claim apply to all configurations? Without defining the minimum or maximum required spacing, the claim scope is unclear.

Written Description: The specification does not clearly define how the guard bands are formed or constrained. If guard bands are fundamental to the claimed invention, they should be described with specific examples or constraints.

4. "using full spectral synthesis capability of a fast Fourier transform or an inverse fast Fourier transform" (Claims 1 & 18)

Indefiniteness: The phrase "full spectral synthesis capability" is ambiguous. Does "full" mean that all available subcarriers must be utilized? Does it require a specific mathematical implementation? If different FFT/IFFT algorithms process data differently, how do we determine what qualifies as "full spectral synthesis capability"?

5. "combining the first channel and the second channel using channel bonding with orthogonal frequency division multiplexing (OFDM)" (Claims 1 & 18)

Indefiniteness: The term "combining" is broad and not clearly defined. Does it mean concatenation of two channels? Does it mean joint encoding of data across both channels? Can the method use adaptive or dynamic bonding, or must it be static?

6. "transmitting the first channel and the second channel comprises transmitting to multiple radios" (Claim 4)

Indefiniteness: The phrase "transmitting to multiple radios" is unclear. Does it mean simultaneous transmission to all radios, or can transmission be sequential? Must each radio receive data from both channels, or can one radio receive data from only one channel?

Written Description: The specification does not describe how the first channel and second channel transmit to "multiple radios."

E. U.S. Patent No. 7,324,469

Claims 24, 25, 26, and 32 of the '469 Patent (herein, the "Asserted '469 Claims") are invalid under 35 U.S.C. §§ 112(a) and 112(b), respectively, for at least the reasons discussed below.

The Asserted '469 Claims contain subject matter that was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the applicants, at the time the application was filed, had possession of the claimed invention. In addition, the Asserted '469 Claims contain subject matter that was not described in the specification in such a way as to enable one of ordinary skill in the art to make and/or use the invention. Further, the Asserted '469 Claims fail to particularly point out and distinctly claim the invention, such that the claim's scope cannot be ascertained with reasonable certainty.

To the extent that the Asserted '469 Claims are construed broadly to cover undisclosed embodiments, the claims are invalid because the '469 Patent specification does not enable a person skilled in the art to practice the full scope of the purported invention, nor does the '469 Patent specification show the applicants had within their possession the full scope of the Asserted '469 Claims. Further, to the extent that any of the Asserted '469 Claims are construed sufficiently broadly to cover one or more of the various accused products identified in IV's infringement contentions, such claims are invalid pursuant to at least 35 U.S.C. § 112(a).

Southwest contends that the following terms of asserted claims 24 and 32 are indefinite under 35 U.S.C. § 112(b), and/or lack written description support under 35 U.S.C. § 112(a):

1. An "Internet Hotspot"

Indefiniteness: to the extent the preamble is found to be limiting, it is unclear whether an "Internet Hotspot" is intended to exclude any manner of providing wired or wireless internet access to one or more users. In particular, the specification states that "[t]he present invention provides rural 'Hotspots' (***s**uch as Wi-Fi access, for example*) to enable wireless and hard wired, satellite distributed Internet access for anyone with a PC or other web-ready device (wireless ready or cabled) and a valid credit card." '469 patent at 1:34-37 (emphasis added).

Written Description: to the extent the preamble is found to be limiting, the specification fails to define a "Hotspot" or distinguish a "Hotspot" from any other manner of providing "Wi-Fi access, for example."

2. "subscriber access unit" / "the subscriber access unit being capable of authenticating a subscription account associated with a user prior to allowing the user access to the Internet" (Claim 24)

Indefiniteness: This term is indefinite, as the patent makes no attempt to define a "subscriber access unit" other than its "being capable of authenticating a subscription account associated with a user prior to allowing the user access to the Internet." The specification contains

contradictory descriptions of "authentication," including that "account status is *first* authenticated by the remote server" but that "*authentication of the user's account is performed by a subscriber merchant service*," in which case the described remote subscriber access unit is not "capable of authenticating a subscription account" as recited in the claim. '469 2:9-10 (emphasis added); *id*. at 2:33-34 ("authentication of the user's account is performed by a subscriber merchant service"). The specification's occasional use of capitalization indicates the patentee intended this to be a defined term, but failed to include its definition. *E.g.*, 4:2-3.

Written Description: The specification provides no description of how the subscriber access unit operates to be "capable of authenticating a subscription account associated with a user prior to allowing the user access to the Internet," by example or otherwise.

3. "operatively coupled to," "operatively coupled between" (Claim 24)

Indefiniteness: As used in the patent, the term "operatively coupled" is vague and ambiguous and thus indefinite. In particular, for example, the specification provides that "[a]s used herein, the term 'operatively coupled' refers to any configuration in which two or more devices *interact in any way*, by wired or wireless connections." '469 patent at 4:25-28 (emphasis added). A person of ordinary skill in the art would be unable to determine the scope of the invention, particularly whether any order or configuration of recited components is required, including whether intervening components would render a system within the scope or not.

Written Description: The limitation lacks written description to enable one skilled in the art to practice the invention, instead confirming "operatively coupled" or "operatively coupled between" have no meaning at all as used in the patent.

4. "a web-ready device operatively coupled to the at least one router" (Claim 24)

Indefiniteness: This term is indefinite as used in the claim and in view of the Plaintiff's Preliminary Infringement Contentions, which indicate that "enabling" end-user devices to access the Internet is sufficient to provide, as the claim requires, a "web-ready device."

Written Description: The limitation lacks written description because the specification merely repeats the term "web-ready device" without description.

5. "a remote location experiencing a relatively high volume of transient traffic" (Claim 24)

Indefiniteness: This limitation is indefinite because a person skilled in the art cannot ascertain with reasonable certainty the scope of what is "a remote location," "a relatively high volume" or "transient traffic." The specification provides the examples of "rest areas, restaurants, truck stops, rural hotels, conference centers, motels and state park lodges" ('469 patent at 1:41-44), all of which can and do have wildly different volumes of traffic and wildly different relation to the term "transient." For example, a conference center may house upwards of thousands of attendees who remain in the location for days on end. A truck stop might see anywhere from a handful to several hundred visitors, but for only brief periods of time during rest periods on trucking routes. A restaurant can be located in a busy shopping mall or in a quaint hamlet. Yet all are examples in the specification of "remote location[s] experiencing a relatively high volume of transient traffic." The specification provides no hints as to how remote is "remote," how many users makes a "relatively high volume," and how much or little time connected is sufficient to constitute "transient traffic."

Written Description: The specification provides no description of "a remote location experiencing a relatively high volume of transient traffic" except by inconsistent examples, instead repeatedly reciting only the claim language for this term.

6. "the data connection is one of a wired data connection and a wireless data connection" (Claim 25)

Indefiniteness: This claim is indefinite in view of the specification and claim 24, in which a "user may authenticate the Subscription account and access the Internet at the remote location by establishing **a** data connection," singular. It is unclear whether claim 25 requires the simultaneous use of wired and wireless connections by the user for the same data connection of claim 24. If claim 25 does not refer to "a data connection" recited in claim 24, it lacks antecedent basis.

Written Description: The specification does not describe the use of "a wired data connection and a wireless data connection" for a single data connection as recited by claim 25 and in view of independent claim 24.

7. "the wireless connection is one of an 802.11a wireless area network, an 802.11b wireless area network, an 802.11g wireless area network, and an 802.11n wireless area network" (Claim 32)

Indefiniteness: This claim is indefinite for lack of antecedent basis insofar as "the wireless connection" lacks antecedent basis; claim 25 recites "wherein the data connection is one of a wired data connection and a wireless data connection," not a "wireless connection." It is unclear whether "wireless connection" and "wireless data connection" are intended to be interchangeable, and, if so, how a simultaneously wired and wireless data connection can comprise "one of an 802.11a wireless area network . . ."

Written Description: The specification lacks written description of an "802.11n wireless area network" at least insofar as the 802.11n standard had not yet been drafted by the time of the filing of the provisional application giving rise to the '469 patent. *See supra* VIII.2.

F. U.S. Patent No. 7,257,582

Claims 1, 2, 5, and 7 of the '582 Patent (herein, the "Asserted '582 Claims") are invalid under 35 U.S.C. §§ 112(a) and 112(b), respectively, for at least the reasons discussed below.

The Asserted '582 Claims contain subject matter that was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the applicants, at the time the application was filed, had possession of the claimed invention. In addition, the Asserted '582 Claims contain subject matter that was not described in the specification in such a way as to enable one of ordinary skill in the art to make and/or use the invention. Further, the Asserted '582 Claims fail to particularly point out and distinctly claim the invention, such that the claim's scope cannot be ascertained with reasonable certainty.

To the extent that the Asserted '582 Claims are construed broadly to cover undisclosed embodiments the claims are invalid because the '582 Patent specification does not enable a person skilled in the art to practice the full scope of the purported invention, nor does the '582 Patent specification show the applicants had within their possession the full scope of the Asserted '582 Claims. Further, to the extent that any of the Asserted '582 Claims are construed sufficiently broadly to cover one or more of the various accused products identified in IV's infringement contentions, such claims are invalid pursuant to at least 35 U.S.C. § 112(a).

Southwest contends that the following terms of asserted claims 1, 2, 5, and 7 are indefinite under 35 U.S.C. § 112(b), and/or lack written description support under 35 U.S.C. § 112(a):

1. "automatically determining file allocation and logically subdividing records of said input file into a plurality of partitions" (Claim 1)

Indefiniteness: The term "automatically determining file allocation" is vague and undefined. What criteria define "automatic" allocation? Is the allocation based on file size, content, metadata, or another heuristic? Can manual intervention be involved at any stage and still meet

this limitation? The phrase "logically subdividing records" is also ambiguous. What constitutes a logical subdivision? Does the subdivision require a specific algorithm or data structure? If different methods (e.g., fixed-size vs. content-based) could be used, which ones fall under the claim scope?

2. "distributing descriptions of all of said partitions to each of a plurality of subtask processors" (Claim 1)

Written Description: The specification does not provide examples or implementations for how "descriptions" are structured or distributed.

3. "repeating step (c) in at least some of the subtask processors each with another unprocessed partition on a first-come/first-served basis" (Claim 1)

Indefiniteness: The phrase "first-come/first-served basis" is undefined. The specification does not appear to describe how a first-come/first-served scheduling policy is implemented.

4. "generating at least one output combining all of the subtask outputs and reflecting the processing of all of said subtasks" (Claim 1)

Written Description: The specification does not provide a detailed method for combining outputs (e.g., sorting, appending, aggregating), nor does it address whether outputs must be stored in a specific format or can remain in multiple files.

5. "wherein the output in step (e) is an accumulation of output records from said subtasks in an arbitrary order" (Claim 5)

Indefiniteness: The phrase "arbitrary order" is ambiguous. Does it mean randomized order? Just non-sorted? If two identical executions produce different orderings, does that still meet the claim limitation?

XI. GROUNDS OF INVALIDITY BASED ON OBVIOUSNESS-TYPE DOUBLE PATENTING

A. Obviousness-Type Double Patenting of the '844 Patent in View of U.S. Patent No. 7,721,282

Southwest provides further notice to Plaintiffs that the asserted claims of the '844 Patent are invalid under the doctrine of obviousness-type double patenting ("OTDP") in view of claim 15 of U.S. Patent No. 7,721,282 (the "'282 Patent"). Because the asserted claims of the '844 Patent are directed to the same invention recited by claims presented in the '282 Patent, and because any differences in the asserted claims of the '844 Patent are, at most, obvious modifications of claims in the '282 Patent, the asserted '844 Patent claims are invalid. *See AbbVie Inc. v. Mathilda & Terrence Kennedy Inst. of Rheumatology Tr.*, 764 F.3d 1366, 1373 (Fed. Cir. 2014).

The '282 Patent is available as an OTDP reference against the '844 Patent. The '282 Patent was filed on March 30, 2006, issued on May 18, 2010, and is set to expire on December 19, 2027. The '844 Patent was filed on February 21, 2007, issued on December 11, 2012, and is set to expire on April 6, 2028. Thus, the '282 Patent was filed earlier than the '844 Patent, issued earlier than the '844 Patent, and is set to expire earlier than the '844 Patent.

As is the case here, where "the applicant chooses to file separate applications for overlapping subject matter . . . the doctrine of obviousness-type double patenting ensures that a particular invention (and obvious variants thereof) does not receive an undue patent term extension." *AbbVie*, 764 F.3d at 1373. Therefore, the earlier-filed, earlier-issued, and earlier-expiring '282 Patent qualifies as a double patenting reference.

Given that the '844 Patent and the reference '282 Patent share at least one common inventor and common ownership, determining whether a patent claim is invalid for OTDP entails two steps. *See id.* at 1374. The first step determines the differences between the respective claims of the two patents. The second step determines whether those differences render the later-expiring set of

claims sufficiently different to be patentably distinct. *Id.* "The second part of this analysis is analogous to the obviousness inquiry under 35 U.S.C. § 103 in the sense that if an earlier claim renders obvious or anticipates a later claim, the later claim is not patentably distinct and is thus invalid for obviousness-type double patenting." *UCB, Inc. v. Accord Healthcare, Inc.*, 890 F.3d 1313, 1323 (Fed. Cir. 2018) (citation omitted). The disclosure of the reference patent may be used to determine whether the claims merely define an obvious variation of what is disclosed and claimed in the reference patent. *Abbvie*, 764 F.3d at 1381.

The attached claim chart in Exhibit 844-OTDP-1 demonstrates that, to the extent there are any differences between the asserted claims of the '844 Patent and claims of the '282 Patent, these differences are not patentably distinct. The charts provide reasons why a person of ordinary skill in the art would conclude that the alleged invention defined in the asserted claims of the '844 Patent would have been an obvious variation of the alleged invention defined in the corresponding claim or claims of the '282 Patent.

B. Obviousness-Type Double Patenting of the '844 Patent in View of U.S. Patent No. 7,870,106

Southwest provides further notice to Plaintiffs that the asserted claims of the '844 Patent are invalid under the doctrine of obviousness-type double patenting ("OTDP") in view of claim 1 of U.S. Patent No. 7,870,106 (the "'106 Patent"). Because the asserted claims of the '844 Patent are directed to the same invention recited by claims presented in the '106 Patent, and because any differences in the asserted claims of the '844 Patent are, at most, obvious modifications of claims in the '106 Patent, the asserted '844 Patent claims are invalid. *See AbbVie Inc. v. Mathilda & Terrence Kennedy Inst. of Rheumatology Tr.*, 764 F.3d 1366, 1373 (Fed. Cir. 2014).

The '106 Patent is available as an OTDP reference against the '844 Patent. The '106 Patent was filed on February 2, 2006, issued on January 11, 2011, and is set to expire on May 18, 2027.

The ‘844 Patent was filed on February 21, 2007, issued on December 11, 2012, and is set to expire on April 6, 2028. Thus, the ‘106 Patent was filed earlier than the ‘844 Patent, issued earlier than the ‘844 Patent, and is set to expire earlier than the ‘844 Patent.

As is the case here, where “the applicant chooses to file separate applications for overlapping subject matter . . . the doctrine of obviousness-type double patenting ensures that a particular invention (and obvious variants thereof) does not receive an undue patent term extension.” *AbbVie*, 764 F.3d at 1373. Therefore, the earlier-filed, earlier-issued, and earlier-expiring ‘106 Patent qualifies as a double patenting reference.

Given that the ‘844 Patent and the reference ‘106 Patent share at least one common inventor and common ownership, determining whether a patent claim is invalid for OTDP entails two steps. *See id.* at 1374. The first step determines the differences between the respective claims of the two patents. The second step determines whether those differences render the later-expiring set of claims sufficiently different to be patentably distinct. *Id.* “The second part of this analysis is analogous to the obviousness inquiry under 35 U.S.C. § 103 in the sense that if an earlier claim renders obvious or anticipates a later claim, the later claim is not patentably distinct and is thus invalid for obviousness-type double patenting.” *UCB, Inc. v. Accord Healthcare, Inc.*, 890 F.3d 1313, 1323 (Fed. Cir. 2018) (citation omitted). The disclosure of the reference patent may be used to determine whether the claims merely define an obvious variation of what is disclosed and claimed in the reference patent. *Abbvie*, 764 F.3d at 1381.

The attached claim chart in Exhibit 844-OTDP-2 demonstrates that, to the extent there are any differences between the asserted claims of the ‘844 Patent and claims of the ‘106 Patent, these differences are not patentably distinct. The charts provide reasons why a person of ordinary skill in the art would conclude that the alleged invention defined in the asserted claims of the ‘844

Patent would have been an obvious variation of the alleged invention defined in the corresponding claim or claims of the '106 Patent.

Southwest reserves the right to assert any other double patenting challenges against any of the Asserted Patents.

XII. INVALIDITY CHARTS

Invalidity charts for each item of prior art are attached to these contentions as Exhibits 844-1 – 844-24, Exhibits 844-OTDP-1 – 844-OTDP-2, Exhibits 722-1 – 722-11, Exhibits 785-1 – 785-29, Exhibits 326-1 – 326-11, Exhibits 469-1 – 469-13, and Exhibits 582-1 – 582-6. These invalidity charts identify where in the prior art references each element of the asserted claim is found. While each element of each asserted claim is found in the identified prior art in multiple locations, the attached charts provide examples of citations sufficient to identify at least one such location where each claim limitation is found in each item of prior art. But the specific citations are only exemplary. Accordingly, Southwest and its expert(s) may rely on uncited portions of the prior art references as the citations must be interpreted in light of the entire disclosure of each reference. Additionally, because persons of skill in the art generally would appreciate an item of prior art in the context of other publications, literature, products, and understanding, Southwest and its expert(s) may rely on other publications and expert testimony as aids in understanding and interpreting the cited portions, for providing context to the cited portions, and as additional evidence that the prior art discloses a claimed feature. Southwest may establish what was known to a person having ordinary skill in the art through other publications, products, and/or testimony, and reserves their right to rely on cited and uncited portions of the prior art references, other publications, and/or testimony to establish that a person of skill in the art would have been motivated to combine the references rendering the claims obvious.

Southwest may modify its interpretation of the prior art based on additional analysis by Southwest's expert(s) and/or based on other circumstances that may affect the meaning or application of the claims.

XIII. DOCUMENT PRODUCTION

Southwest has produced the prior art references identified in these invalidity contentions. As noted herein, Southwest is also producing the below listed Exhibits with these invalidity contentions.

Exhibits	Description
844-1 – 844-24	Invalidity Charts for the '844 Patent
722-1 – 722-11	Invalidity Charts for the '722 Patent
785-1 – 785-29	Invalidity Charts for the '785 Patent
326-1 – 326-11	Invalidity Charts for the '326 Patent
469-1 – 469-13	Invalidity Charts for the '469 Patent
582-1 – 582-6	Invalidity Charts for the '582 Patent
844-OTDP-1 – 844-OTDP-2	Invalidity Charts Demonstrating Obviousness-Type Double Patenting for the '844 Patent

Date: April 8, 2025

Respectfully submitted,

/s/ S. Wallace Dunwoody

Michael C. Wilson
Texas Bar No. 21704590
mwilson@munckwilson.com
S. Wallace Dunwoody
Texas Bar No. 24040838
wdunwoody@munckwilson.com
MUNCK WILSON MANDALA, LLP
2000 McKinney Avenue, Suite 1900
Dallas, Texas 75201
(972) 628-3600
(972) 628-3616 fax

David G. Henry
Texas Bar. No. 09479355
dhenry@munckwilson.com
MUNCK WILSON MANDALA, LLP
510 Austin Avenue, Suite 3100
Waco, Texas 76701
(254) 362-2300

Tri T. Truong
Texas Bar No. 24102969
ttruong@munckwilson.com
MUNCK WILSON MANDALA, LLP
807 Las Cimas Parkway, Suite 300
Austin, Texas 78756
737-201-1600

Attorneys for Southwest Airlines Co.

CERTIFICATE OF SERVICE

I hereby certify that on April 8, 2025, a true and correct copy of Defendant Southwest Airlines Co.'s Preliminary Invalidity Contentions and accompanying exhibits were served via electronic mail to counsel of record.

/s/ S. Wallace Dunwoody
S. Wallace Dunwoody

EXHIBITS 785-1 – 785-29

Exhibit 785-1

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent Application Publication No. 2003/0028671 ("Mehta")

Mehta titled "Method and system for two-way initiated data communication with wireless devices" to Samir Narendra Mehta et al. was filed on June 10, 2002 and published on February 6, 2003.[1] Mehta is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. § 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Mehta does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Mehta or combine the teachings of Mehta with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, including but not limited to:

- Huitema, C., Network Working Group Request for Comment (RFC): 1383, entitled An Experiment in DNS Based IP Routing ("RFC 1383");
- U.S. Patent No. 6,970,941 ("Caronni I");
- U.S. Patent No. 7,814,228 ("Caronni II"); and/or
- U.S. Patent No. 6,766,371 ("Hipp").

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

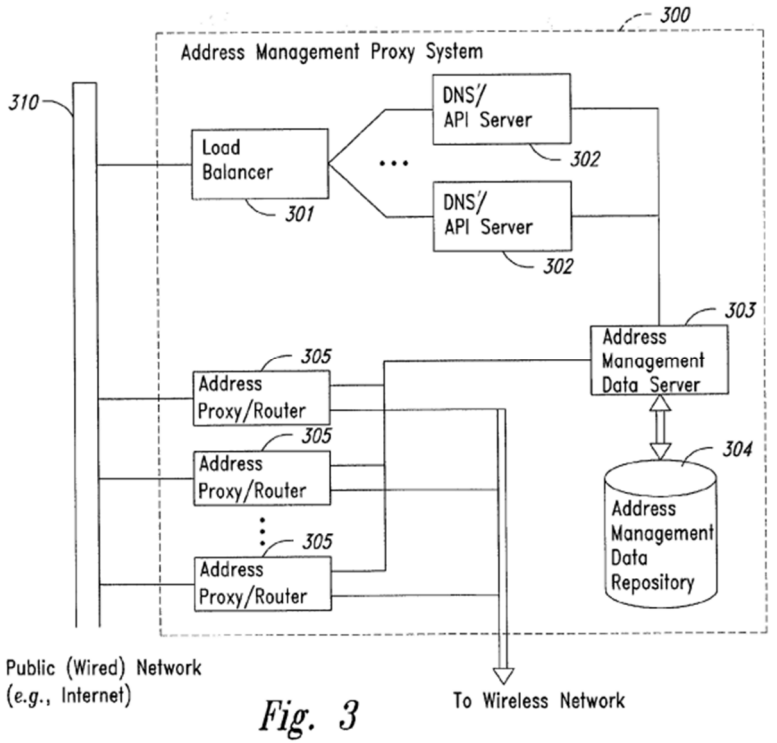
[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

Claim 30	Mehta
[30.0] A virtual network manager, comprising:	To the extent the preamble is limiting, Mehta discloses: "Methods and systems for providing two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP, are provided. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporarily use by a requesting device on a public network to communicate with a wireless device on a wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more data repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the public network to send data to the wireless device." Mehta at Abstract. "The present invention relates to methods and systems for initiating communication with wireless devices and, in particular, to methods and systems for initiating communication with a device on a private network from a device on a public network to achieve virtual end-to-end connectivity." Mehta at [0002]. "Embodiments of the present invention provide computer- and network-based methods and systems for two-way initiated, bi-directional communication with

Claim 30	Mehta
	wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internetwork, such as the Internet, to initiate communication with and send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporary use by a requesting device on an external public network to communicate with a wireless device on a private wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. The mapping of a temporary public address to the private address of a wireless device is maintained and updated transparently by the AMPS using routing tables and other mapping data structures." Mehta at [0011]. "To insure a greater degree of security, according to one embodiment, the AMPS maintains a Time to Live (TTL) parameter with each address mapping. In this way, once the TTL value indicates that the mapping has expired, the AMPS can destroy the mapping, and also any connection. Further, in some embodiments, the AMPS also puts a timestamp in its own mapping tables. After some timeout period based upon the timestamp, the AMPS can destroy the mapping, thereby forcing a new mapping to be initiated on a periodic basis." Mehta at [0015]. "Embodiments of the present invention provide computer- and network-based methods and systems for two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with

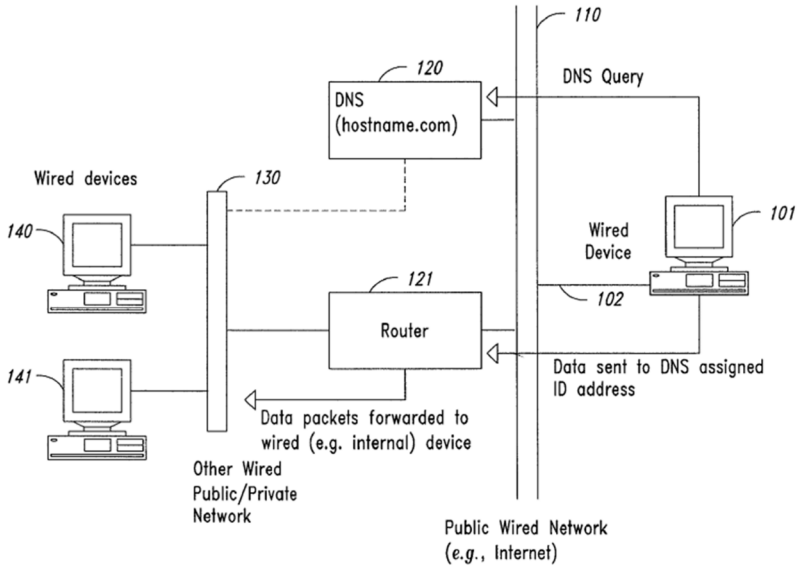
Claim 30	Mehta
	and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices." Mehta at [0024]. "The second approach used to implement the AMPS supports full bi-directional communication through point-to-point connections established, for example, using TCP/IP protocol. (Note that these same techniques also support connection-less UDP bi-directional communication). The second approach can be implemented by providing a modified implementation of a standard UDP or TCP/IP function, "GetHostByName." The GetHostByName API allows a string designation to identify the designated device and returns an IP address data structure. Alternatively, to increase the level of security provided, the AMPS can implement a specialized API to return the dynamically allocated public address that (now) corresponds to the requested wireless device. A disadvantage of the specialized API approach is that the device on the public network (or other device that wishes to obtain a connection to the wireless device) needs to include specialized code in the application on the requesting device." Mehta at [0041].

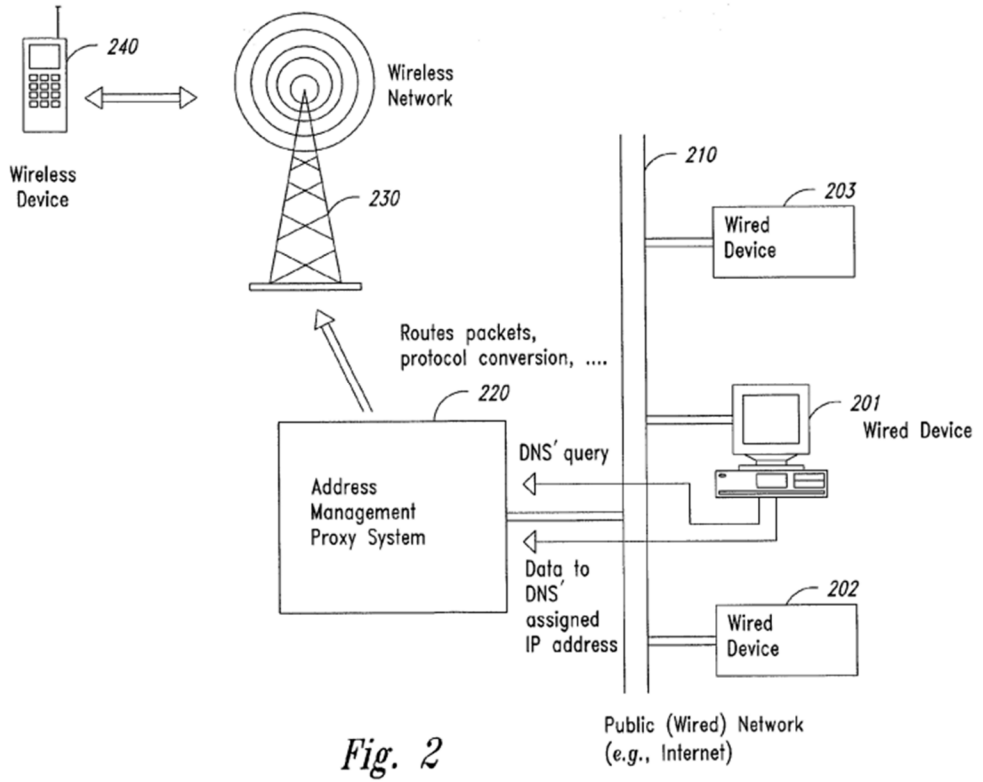
Claim 30	Mehta
	 *Fig. 3* Mehta at FIG. 3. To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Caronni I. Caronni I concerns similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, use of virtual networks was well-known and common. Modifying this reference with Caronni I's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the

Claim 30	Mehta
	disclosures of this reference with that of Caronni I for example, to incorporate this limitation. *See, e.g.*, "Methods and systems consistent with the present invention establish a virtual network on top of current IP network naming schemes. The virtual network uses a separate layer to create a modification to the IP packet format that is used to separate network behavior from addressing. As a result of the modification to the packet format, any type of delivery method may be assigned to any address or group of addresses. The virtual network also maintains secure communications between nodes, while providing the flexibility of assigning delivery methods independent of the delivery addresses." Caronni I at Abstract. "The present invention relates generally to data processing systems and, more particularly, to a private network using a public-network infrastructure." Caronni I at 1:57–59. "Methods and systems consistent with the present invention overcome the shortcomings of existing networks by establishing a "Supernet," which is a private network that uses components from a public-network infrastructure. A Supernet allows an organization to utilize a public-network infrastructure for its enterprise network so that the organization no longer has to maintain a private network infrastructure; instead, the organization may have the infrastructure maintained for them by one or more service providers or other organizations that specialize in such connectivity matters. As such, the burden of maintaining an enterprise network is greatly reduced. Moreover, a Supernet is not geographically restrictive, so a user may plug their device into the Internet from virtually any portal in the world and still be able to use the resources of their private network in a secure and robust manner." Caronni I at 4:36-52.

Claim 30	Mehta
	"Supernets also provide heterogeneous addressing functionality. The Supernet uses a separate layer that isolates address names of nodes from addressing schemes and delivery schemes. The Supernet contains a modification to the IP packet format that can be used to separate network behavior from addressing. As a result of the modification, any delivery scheme may be assigned to any address, or group of addresses." Caronni I at 4:53-59. *See also* Caronni I at Claim 1, FIGS. 3, 5.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	"The present invention relates to methods and systems for initiating communication with wireless devices and, in particular, to methods and systems for initiating communication with a device on a private network from a device on a public network to achieve virtual end-to-end connectivity." Mehta at [0002]. "Embodiments of the present invention provide computer- and network-based methods and systems for two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internetwork, such as the Internet, to initiate communication with and send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporary use by a requesting device on an external public network to communicate with a wireless device on a private wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. The mapping of a temporary public address to the private address of a wireless device is maintained and updated transparently by the AMPS using routing tables and other mapping data structures."

Claim 30	Mehta
	Mehta at [0011]. "Although the techniques of the AMPS are generally applicable to any a wired device communicating with a wireless device, the phrase "public network" (or "wired network") is used generally to imply any type of internetworked environment including a public network or a backbone that is somewhere down the line connected to one or more private or public networks. In addition, although the examples described herein often refer to the Internet, one skilled in the art will recognize that the concepts and inventions described are applicable to other forms and embodiments of internetworking, including, for example ATM type networks. Thus, techniques of the present invention can also be used by one device on a first wireless network to communicate with another wireless device on a second network—each device ends up communicating with the Address Proxy/Router of the other network. This scenario is feasible because each wireless network (or its carrier infrastructure) is connected to a proxy/router that is also connected (via a public address) to a public network. In addition, although a public network is sometimes also referred to herein as a wired network, one skilled in the art will recognize that any network that exposes routable (public) addresses may be implied. Thus, a wireless network with unique public (and routable) address can also employ techniques of the present invention to perform bi-directional communication. Also, one skilled in the art will recognize that terms such as wireless device, phone, handheld, etc., are used interchangeably to indicate any type of wireless device that is capable of operating with the AMPS. In addition, terms may have alternate spellings which may or may not be explicitly mentioned, and one skilled in the art will recognize that all such variations of terms are intended to be included." Mehta at [0033]. "Public address to proxy/router machine table 620 comprises a public network address field 631 and an indication of a functioning proxy/router machine 621. By maintaining such a mapping, the AMPS is able to substitute proxy/router machines for other proxy/router machines to provide a higher degree of robustness. Each

Claim 30	Mehta
	proxy/router machine has a preconfigured set of public network addresses, such as are typically configured by network cards inserted into the proxy/router machine. These address are allocated in a standard fashion through prior purchase or obtaining from an address authorizing authority, currently the Internet Corporation for Assigned Names and Numbers (ICANN). When a machine is inserted for use into the AMPS, indications of these addresses are stored in field 631." Mehta at [0055].  *Fig. 1* Mehta at FIG. 1.

Claim 30	Mehta
	 *Fig. 2* Mehta at FIG. 2. To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Caronni I. Caronni I concerns similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, use of virtual networks was well-known and common. Modifying this reference with Caronni I's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A

Claim 30	Mehta
	person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with that of Caronni I for example, to incorporate this limitation. *See, e.g.*, "Methods and systems consistent with the present invention establish a virtual network on top of current IP network naming schemes. The virtual network uses a separate layer to create a modification to the IP packet format that is used to separate network behavior from addressing. As a result of the modification to the packet format, any type of delivery method may be assigned to any address or group of addresses. The virtual network also maintains secure communications between nodes, while providing the flexibility of assigning delivery methods independent of the delivery addresses." Caronni I at Abstract. "The present invention relates generally to data processing systems and, more particularly, to a private network using a public-network infrastructure." Caronni I at 1:57–59. "Methods and systems consistent with the present invention overcome the shortcomings of existing networks by establishing a "Supernet," which is a private network that uses components from a public-network infrastructure. A Supernet allows an organization to utilize a public-network infrastructure for its enterprise network so that the organization no longer has to maintain a private network infrastructure; instead, the organization may have the infrastructure maintained for them by one or more service providers or other organizations that specialize in such connectivity matters. As such, the burden of maintaining an enterprise network is greatly reduced. Moreover, a Supernet is not geographically restrictive, so a user may plug their device into the Internet from virtually any portal in the world and still be able to use the resources of their private network in a secure and robust manner." Caronni I at 4:36-52.

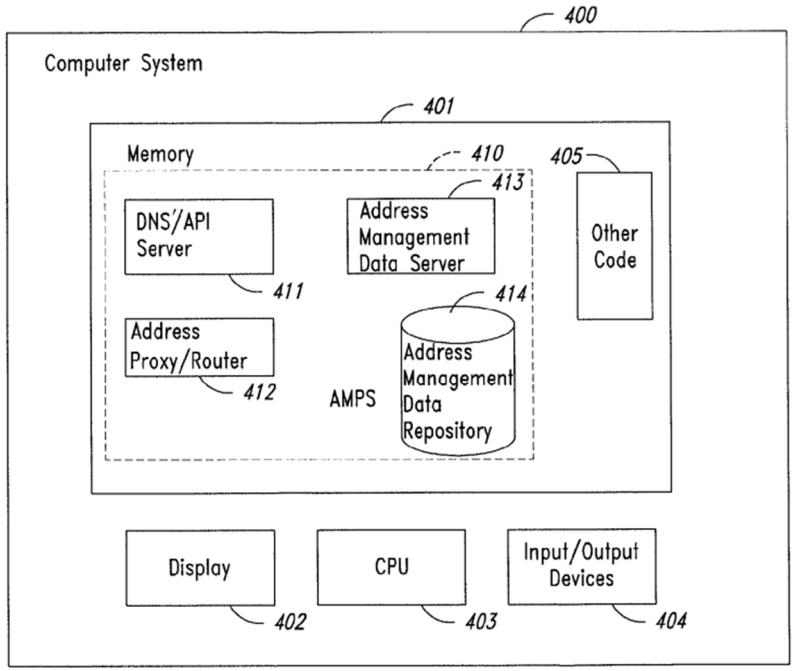
Claim 30	Mehta
	"Supernets also provide heterogeneous addressing functionality. The Supernet uses a separate layer that isolates address names of nodes from addressing schemes and delivery schemes. The Supernet contains a modification to the IP packet format that can be used to separate network behavior from addressing. As a result of the modification, any delivery scheme may be assigned to any address, or group of addresses." Caronni I at 4:53-59. "SNlogin 522 is a script used for logging into a Supernet. Successfully executing this script results in a Unix shell from which programs (e.g., node A 522) can be started to run within the Supernet context, such that address translation and security encapsulation is performed transparently for them and all they can typically access is other nodes on the Supernet. Alternatively, a parameter may be passed into SNlogin 522 that indicates a particular process to be automatically run in a Supernet context. Once a program is running in a Supernet context, all programs spawned by that program also run in the Supernet context, unless explicitly stated otherwise. SNlogout 524 is a script used for logging out of a Supernet. Although both SNlogin 522 and SNlogout 524 are described as being scripts, one skilled in the art will appreciate that their processing may be performed by another form of software. VARPD 526 performs address translation between node IDs and real addresses. KMC 528 is the key management component for each node that receives updates whenever the key for a channel ("the channel key") changes. There is one KMC per node per channel. KMD 530 receives requests from SNSL 542 of the TCP/IP protocol stack 534 when a packet is received and accesses the appropriate KMC for the destination node to retrieve the appropriate key to decrypt the packet. Node A 532 is a Supernet node running in a Supernet context." Caronni I at 8:31-55. "TCP/IP protocol stack 534 contains a standard TCP/UDP layer 538, two standard IP layers (an inner IP layer 540 and an outer IP layer 544), and a Supernet security layer (SNSL) 542, acting as the conduit for all Supernet communications. To conserve memory, both inner IP layer 540 and outer IP layer 544 may share the

Claim 30	Mehta
	same instance of the code of an IP layer. SNSL 542 performs security functionality as well as address translation. It also caches the most recently used channel keys for ten seconds. Thus, when a channel key is needed, SNSL 542 checks its cache first, and if it is not found, it requests KMD 530 to contact the appropriate KMC to retrieve the appropriate channel key." Caronni I at 8:54-67. "FIG. 5 depicts administrative machine 306 and device 302 in greater detail, although the other devices 304 and 308–312 may contain similar components. Device 302 and administrative machine 306 communicate via Internet 314. Each device contains similar components, including a memory 502, 504; secondary storage 506, 508; a central processing unit (CPU) 510, 512; an input device 514, 516; and a video display 518, 520. One skilled in the art will appreciate that these devices may contain additional or different components. Memory 504 of administrative machine 306 includes the SASD process 540, VARPD 548, and KMS 550 all running in user mode. That is, CPU 512 is capable of running in at least two modes: user mode and kernel mode. When CPU 512 executes programs running in user mode, it prevents them from directly manipulating the hardware components, such as video display 518. On the other hand, when CPU 512 executes programs running in kernel mode, it allows them to manipulate the hardware components. Memory 504 also contains a VARPDB 551 and a TCP/IP protocol stack 552 that are executed by CPU 512 running in kernel mode. TCP/IP protocol stack 552 contains a TCP/UDP layer 554 and an IP layer 556, both of which are standard layers well known to those of ordinary skill in the art. Secondary storage 508 contains a configuration file 558 that stores various configuration-related information (described below) for use by SASD 540." Caronni I at 6:44-7:2. *See also* Caronni I at Claim 1, FIGS. 3, 5
[30.2] a memory and a processor to implement a register module configured	"Methods and systems for providing two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less

Claim 30	Mehta
to register devices in a virtual network, the register module further configured to:	protocols, such as, for example, TCP/IP and UDP/IP, are provided. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporarily use by a requesting device on a public network to communicate with a wireless device on a wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more data repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the public network to send data to the wireless device." Mehta at Abstract. "To insure a greater degree of security, according to one embodiment, the AMPS maintains a Time to Live (TTL) parameter with each address mapping. In this way, once the TTL value indicates that the mapping has expired, the AMPS can destroy the mapping, and also any connection. Further, in some embodiments, the AMPS also puts a timestamp in its own mapping tables. After some timeout period based upon the timestamp, the AMPS can destroy the mapping, thereby forcing a new mapping to be initiated on a periodic basis." Mehta at [0015]. "In existing systems, data communication (communication on a data channel) between a wireless device that is connected to a private wireless network and a wired device connected to a public wired network (e.g., the Internet) can only be initiated by the wireless device. Some carriers have assigned fixed public IP

Claim 30	Mehta
	addresses to wireless devices; however, the wireless devices need to then have client programs (e.g., a UDP stack) capability of receiving and handling the incoming communication packets. Moreover, these wireless devices are then part of a public wireless network and not a private wireless network. Since public IP addresses are becoming a scarcer commodity and currently expensive to service a network of multi-million devices, carriers cannot in a practical sense count on having a fixed public IP address for each device on its network. (Although movement to IPv6 will allow more addresses, the current IPv4 definitions are limited and the potential number of wireless users subscribing to larger carriers is very high. In some regions of the world, the current public address scheme is even more limited.) Further, such addressing capability would expose each device to further security risks, because each such device is part of a publicly accessible network and it becomes more difficult to implement and enforce security measures. Thus, assigning private IP addresses to devices is preferred in existing wireless networks over assigning fixed public IP addresses to devices. When wireless networks are private, the locations (addresses) of the wireless devices are intentionally hidden from public view by a carrier system's (or, as referred to in some countries, operator's) infrastructure." Mehta at [0025]. "Example embodiments described herein provide applications, tools, data structures and other support to implement private to public address mappings over one or more wired and wireless networks to be used for bi-directional communication. One skilled in the art will recognize that other embodiments of the methods and systems of the present invention may be used for many other purposes, including to push information and/or data or code from a public network such as the Internet to a wireless device. In addition, although this description primarily refers to "data" as being sent via the networks, one skilled in the art will recognize that all types of data can be communicated using the techniques described herein including, but not limited to, text, graphics, audio, and video." Mehta at [0034].

Claim 30	Mehta
	"FIG. 4 is an example block diagram of a general purpose computer system for practicing embodiments of the Address Management Proxy System. The general purpose computer system 400 may comprise one or more server and/or client computing systems and may span distributed locations. In addition, each block may represent one or more such blocks as appropriate to a specific embodiment or may be combined with other blocks. The various blocks of the Address Management Proxy System 410 may physically reside on one or more machines, which use standard interprocess communication mechanisms to communicate with each other. In the embodiment shown, computer system 400 comprises a computer memory ("memory") 01, a display 402, a Central Processing Unit ("CPU") 403, and Input/Output devices 404. The Address Management Proxy System 410 is shown residing in memory 401. The components of the Address Management Proxy System 410 preferably execute on CPU 403 and manage the address mapping of wireless devices on a wireless network, as described in previous figures, to allow other wired systems to communicate with the wireless devices. Other downloaded code 405 and potentially other data repositories also reside in the memory 410, and preferably execute on one or more CPU's 403. In a typical embodiment, the AMPS 410 includes one or more DNS'/API servers 411, one or more Address Proxy/Routers 412, an Address Management Data Server 413, and Address Management Data Repositories 414. As described earlier, the AMPS may include other data repositories and components, such as a load balancer, depending upon the particular implementation." Mehta at [0036]. "In one embodiment, a timestamp of the mapping between a public network address and a private address is also noted in table 630. After a defined time out, based upon this timestamp, the Address Management Data Server sends a request to the Address Proxy/Router associated with the mapping to unmap the public-to-private address mapping and updates the mapping table 630, thereby returning the associated public address to the pool of unused public network addresses." Mehta at [0056].

Claim 30	Mehta
	 *Fig. 4* Mehta at FIG. 4. To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Caronni I, Caronni II, and/or Hipp. Caronni I, Caronni II, and Hipp concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, registration of devices in a virtual network was a well-known and commonly-used technique. Modifying this reference with Caronni I's, Caronni II's, and/or Hipp's teachings would have only required simple substitution,

Claim 30	Mehta
	would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with those of Caronni I, Caronni II, or Hipp for example, to incorporate this limitation. *See, e.g.*, "To configure a Supernet, a system administrator creates a configuration file 558 that is used by SASD 540 when starting or reconfiguring a Supernet. This file may specify: (1) the Supernet name, (2) all of the channels in the Supernet, (3) the nodes that communicate over each channel, (4) the address of the KMS for each channel, (5) the address of the VARPD that acts as the server for the Supernet, (6) the user IDs of the users who are authorized to create Supernet nodes, (7) the authentication mechanism to use for each user of each channel, and (8) the encryption algorithm to use for each channel. Although the configuration information is described as being stored in a configuration file, one skilled in the art will appreciate that this information may be retrieved from other sources, such as databases or interactive configurations." Caronni I at 8:1-15. "SASD 540 represents a Supernet: there is one instance of an SASD per Supernet, and it both authenticates nodes and authorizes nodes to join the Supernet. VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as a node IDs, to the network addresses recognized by the public-network infrastructure, known as the real addresses. The "node ID" may include the following: a Supernet ID (e.g., 0x123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1). Although the virtual address is described in an IP address scheme, one skilled in the art will appreciate that the virtual address may be any other type addressing scheme, such as an e-mail address, IPX, or IPv6. Since the node ID includes a Supernet ID, a node will have more than one node ID when it communicates over more than one channel. The "real address" is an IP address (e.g., 10.0.0.2) that is globally unique and meaningful to the public-network infrastructure. In a Supernet, one VARPD runs on each machine, and it may play

Claim 30	Mehta
	two roles. First, a VARPD may act as a server by storing all address mappings for a particular Supernet into its associated VARPDB. Second, regardless of its role as a server or not, each VARPD assists in address translation for the nodes on its machine. In this role, the VARPD stores into its associated VARPDB the address mappings for its nodes, and if it needs a mapping that it does not have, it will contact the VARPD that acts as the server for the given Supernet to obtain it. The VARPDB may also decide which virtual address to use in the translation. That is, the VARPDB may associate a virtual address with multiple real addresses or vice versa." Caronni I at 7:3–33. "FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:67–10:18. "After configuring SNSL, SNlogin invokes an operating system call, SETVIN, to cause the SNlogin script to run in a Supernet context (step 720). In Unix, each process has a data structure known as the "proc structure" that contains the process ID as well as a pointer to a virtual memory description of this process. In

Claim 30	Mehta
	accordance with methods and systems consistent with the present invention, the Supernet IDs indicating the channels over which the process communicates as well as its virtual address for this process are added to this structure. By associating this information with the process, the SNSL layer can enforce that this process runs in a Supernet context. Although methods and systems consistent with the present invention are described as operating in a Unix environment, one skilled in the art will appreciate that such methods and systems can operate in other environments. After the SNlogin script runs in the Supernet context, the SNlogin script spawns a Unix program, such as a Unix shell or a service daemon (step 722). In this step, the SNlogin script spawns a Unix shell from which programs can be run by the user. All of these programs will thus run in the Supernet context until the user runs the SNlogout script." Caronni I at 10:63–11:17. "The packet and Supernet ID are then transmitted to the SNSL layer using the modified socket structure (step 806). The SNSL layer then accesses the VARPDB to obtain the address mapping between virtual source node address 642 and the source real address 614 as well as the virtual destination node address 644 and the destination real address 616 (step 808). If they are not contained in the VARPDB because this is the first time a packet has been sent from this node or sent to this destination, the VARPDB accesses the local VARPD to obtain the mapping. When contacted, the VARPD on the local machine contacts the VARPD that acts as the server for the Supernet to obtain the appropriate address mapping. Since the VARPDB maintains all real IP addresses, a remote node may securely communicate with another remote node without reverfication." Caronni I at 11:47–61. *See also* Caronni I at FIGS. 7A and 7B. "In an embodiment of the present invention a virtual network is supported by a physical network and includes a virtual address resolution facility. The virtual

Claim 30	Mehta
	address resolution facility is used to resolve a virtual IP address into a real IP address. A registration request is received at the virtual address resolution facility which includes a real IP address, a port address, a transport protocol designation and an Application layer protocol designation associated with a virtual address destination. The associations are stored using the virtual address resolution facility. A resolution request is received referencing a virtual address destination and the resolution request is resolved using the virtual address resolution facility and the stored associations." Caronni II at 3:11–23. "FIG. 2 depicts a block diagram of the VARP lookup table 26 used by the virtual address resolution facility 24. The VARP lookup table 26 may be stored at any location accessible over the network. The VARP lookup table 26 includes registered virtual destination addresses located in a column 78 and corresponding associations located in a column 79. Entries 80, 82, 84, 86 and 88 include the registered virtual destination address and information associated with the registered virtual destination address. Each entry in the VARP lookup table 26 will list the registering virtual IP address 90 and associated information including a real IP address 91, a transport protocol designation 92, a port number identification 93 and an Application layer protocol designation 94. The associations are provided at the time the virtual destination address is registered in the virtual network. For example, the entry 80 for the virtual IP address 10.0.0.12 (90) includes the association of a real IP address 152.70.0.1 (91), an associated Transport layer protocol designation, UDP 92, an associated port number, 6789 (93), and an Application layer protocol designation of 'none' 94." Caronni II at 5:10–29. "FIG. 4 depicts the sequence of steps followed by the illustrative embodiment of the present invention to process the received encapsulated message. The sequence begins when a process associated with a virtual IP address registers with the virtual address resolution facility 24 (step 140). The registration is stored in a VARP lookup table accessible over the network. Subsequently, a message is received at

Claim 30	Mehta
	the electronic device which bears a MAC/network interface address of the network interface of the electronic device with the designated real IP address listed in the virtual address registration (Step 142). The MAC address header 117 is stripped off at the Link layer and the message and appended headers are passed up to the Network layer 114(Step 144). The Network layer 114 strips off the IP header 115 and identifies the Transport Protocol header 113 underneath. The Network layer passes the message and remaining appended header to the Transport layer 112 (Step 146). The Transport layer 112 strips off the Transport header and then passes the message to the Application layer via the port address of the interface identified in the VARP registration. (Step 148). The receiving process determines whether it is the virtual destination address the message is addressed to (Step 149) and acts accordingly. If the receiving process that is examining the message is the destination address, the process examines the data (Step 150). Alternatively, if the message is not intended for the process executing on the edge device, the process re-routes the message and sends it back down the protocol model stack, on to the local network operating behind the edge device, and on to its intended destination by performing traditional VARP and ARP lookups. Alternatively forwarding the message may be accomplished by reinserting the packet into the IP layer of the networking stack." Caronni II at 7:33–67. *See also* Caronni II at Claim 1, FIGS. 2-4. "FIG. 3 is a data flow diagram illustrating the registration of virtual network environment parameters. The VNE framework 200 is a software module that processes transactions between the applications and the operating system. The VNE parameters are registered with the VNE framework 200 at the time the application is started. The VNE parameters include the application IP address, the virtual network subnet, and the global virtual address subnet. At step 250, the registration harness 220 supplies the application IP address, virtual subnet, and the global virtual address subnet for a $process_x$ to the VNE framework 200. The VNE framework 200 then records the IP address, virtual subnet, and the global virtual

Claim 30	Mehta
	address subnet for the $process_x$ at step 252. The $process_x$ can then spawn additional processes (or create additional objects) at step 254. the new $process_y$ inherits the IP address, virtual subnet, and global virtual address subnet from $process_x$. At step 256, the registration harness 220 launches the application related to $process_y$." Hipp at 3:52-4:2. "Virtualization of network identity is achieved by assigning a unique virtual IP address and virtual hostname to a group of processes that make up the application instance which the instance keeps throughout its execution. This virtual network identity stays with the application instance regardless of which node the application is running on. The framework, in essence, provides a mechanism to create this virtual network identity (VNI) around the application using the virtual network parameters assigned to it. In one embodiment, the virtual network parameters include an IP address and hostname. The framework 200 ensures that the application's instance uses the virtual network parameters, transparently, so that it can be moved across machines, without modifications to the application." Hipp at 6:1-15. *See also* Hipp at 6:16-60, FIG. 3.
[30.3] receive a registration request from an agent associated with a device;	"Methods and systems for providing two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP, are provided. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporarily use by a requesting device on a public network to communicate with a wireless device on a wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and

Claim 30	Mehta
	allocated dynamically to wireless network devices as required. In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more data repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the public network to send data to the wireless device." Mehta at Abstract. "To insure a greater degree of security, according to one embodiment, the AMPS maintains a Time to Live (TTL) parameter with each address mapping. In this way, once the TTL value indicates that the mapping has expired, the AMPS can destroy the mapping, and also any connection. Further, in some embodiments, the AMPS also puts a timestamp in its own mapping tables. After some timeout period based upon the timestamp, the AMPS can destroy the mapping, thereby forcing a new mapping to be initiated on a periodic basis." Mehta at [0015]. "In existing systems, data communication (communication on a data channel) between a wireless device that is connected to a private wireless network and a wired device connected to a public wired network (e.g., the Internet) can only be initiated by the wireless device. Some carriers have assigned fixed public IP addresses to wireless devices; however, the wireless devices need to then have client programs (e.g., a UDP stack) capability of receiving and handling the incoming communication packets. Moreover, these wireless devices are then part of a public wireless network and not a private wireless network. Since public IP addresses are becoming a scarcer commodity and currently expensive to service a network of multi-million devices, carriers cannot in a practical sense count on having a fixed public IP address for each device on its network. (Although movement to IPv6 will allow more addresses, the current IPv4 definitions are limited and the potential number of wireless users subscribing to larger carriers is

Claim 30	Mehta
	very high. In some regions of the world, the current public address scheme is even more limited.) Further, such addressing capability would expose each device to further security risks, because each such device is part of a publicly accessible network and it becomes more difficult to implement and enforce security measures. Thus, assigning private IP addresses to devices is preferred in existing wireless networks over assigning fixed public IP addresses to devices. When wireless networks are private, the locations (addresses) of the wireless devices are intentionally hidden from public view by a carrier system's (or, as referred to in some countries, operator's) infrastructure." Mehta at [0025]. "Example embodiments described herein provide applications, tools, data structures and other support to implement private to public address mappings over one or more wired and wireless networks to be used for bi-directional communication. One skilled in the art will recognize that other embodiments of the methods and systems of the present invention may be used for many other purposes, including to push information and/or data or code from a public network such as the Internet to a wireless device. In addition, although this description primarily refers to "data" as being sent via the networks, one skilled in the art will recognize that all types of data can be communicated using the techniques described herein including, but not limited to, text, graphics, audio, and video." Mehta at [0034]. "In one embodiment, a timestamp of the mapping between a public network address and a private address is also noted in table 630. After a defined time out, based upon this timestamp, the Address Management Data Server sends a request to the Address Proxy/Router associated with the mapping to unmap the public-to-private address mapping and updates the mapping table 630, thereby returning the associated public address to the pool of unused public network addresses." Mehta at [0056].

Claim 30	Mehta
	To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Caronni I, Caronni II, and/or Hipp. Caronni I, Caronni II, and Hipp concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, registration of devices in a virtual network was a well-known and commonly-used technique. Modifying this reference with Caronni I's, Caronni II's, and/or Hipp's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with those of Caronni I, Caronni II, or Hipp for example, to incorporate this limitation. *See, e.g.*, "FIG. 3 depicts a data processing system 300 suitable for use with methods and systems consistent with the present invention. Data processing system 300 comprises a number of devices, such as computers 302–312, connected to a public network, such as the Internet 314. A Supernet's infrastructure uses components from the Internet because devices 302, 304, and 312 contain nodes that together form a Supernet and that communicate by using the infrastructure of the Internet. These nodes 316, 318, 320, and 322 are communicative entities (e.g., processes) running within a particular device and are able to communicate among themselves as well as access the resources of the Supernet in a secure manner. When communicating among themselves, the nodes 316, 318, 320, and 322 serve as end points for the communications, and no other processes or devices that are not part of the Supernet are able to communicate with the Supernet's nodes or utilize the Supernet's resources. The Supernet also includes an administrative node 306 to administer to the needs of the Supernet." Caronni I at 4:66–5:13. "FIG. 5 depicts administrative machine 306 and device 302 in greater detail, although the other devices 304 and 308–312 may contain similar components. Device 302 and administrative machine 306 communicate via Internet 314. Each

Claim 30	Mehta
	device contains similar components, including a memory 502, 504; secondary storage 506, 508; a central processing unit (CPU) 510, 512; an input device 514, 516; and a video display 518, 520. One skilled in the art will appreciate that these devices may contain additional or different components. Memory 504 of administrative machine 306 includes the SASD process 540, VARPD 548, and KMS 550 all running in user mode. That is, CPU 512 is capable of running in at least two modes: user mode and kernel mode. When CPU 512 executes programs running in user mode, it prevents them from directly manipulating the hardware components, such as video display 518. On the other hand, when CPU 512 executes programs running in kernel mode, it allows them to manipulate the hardware components. Memory 504 also contains a VARPDB 551 and a TCP/IP protocol stack 552 that are executed by CPU 512 running in kernel mode. TCP/IP protocol stack 552 contains a TCP/UDP layer 554 and an IP layer 556, both of which are standard layers well known to those of ordinary skill in the art. Secondary storage 508 contains a configuration file 558 that stores various configuration-related information (described below) for use by SASD 540." Caronni I at 6:43–7:2. "Memory 502 of device 302 contains SNlogin script 522, SNlogout script 524, VARPD 526, KMC 528, KMD 530, and node A 532, all running in user mode. Memory 502 also includes TCP/IP protocol stack 534 and VARPDB 536 running in kernel mode." Caronni I at 8:26-30. "FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is

Claim 30	Mehta
	not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:66-10:18. *See also* Caronni I at FIGS. 3, 5, 7A, 7B. "In an embodiment of the present invention a virtual network is supported by a physical network and includes a virtual address resolution facility. The virtual address resolution facility is used to resolve a virtual IP address into a real IP address. A registration request is received at the virtual address resolution facility which includes a real IP address, a port address, a transport protocol designation and an Application layer protocol designation associated with a virtual address destination. The associations are stored using the virtual address resolution facility. A resolution request is received referencing a virtual address destination and the resolution request is resolved using the virtual address resolution facility and the stored associations." Caronni II at 3:11–23. "FIG. 2 depicts a block diagram of the VARP lookup table 26 used by the virtual address resolution facility 24. The VARP lookup table 26 may be stored at any location accessible over the network. The VARP lookup table 26 includes registered virtual destination addresses located in a column 78 and corresponding associations located in a column 79. Entries 80, 82, 84, 86 and 88 include the registered virtual destination address and information associated with the registered virtual destination address. Each entry in the VARP lookup table 26 will list the registering virtual IP address 90 and associated information including a real IP

Claim 30	Mehta
	address 91, a transport protocol designation 92, a port number identification 93 and an Application layer protocol designation 94. The associations are provided at the time the virtual destination address is registered in the virtual network. For example, the entry 80 for the virtual IP address 10.0.0.12 (90) includes the association of a real IP address 152.70.0.1 (91), an associated Transport layer protocol designation, UDP 92, an associated port number, 6789 (93), and an Application layer protocol designation of 'none' 94." Caronni II at 5:10–29. "FIG. 4 depicts the sequence of steps followed by the illustrative embodiment of the present invention to process the received encapsulated message. The sequence begins when a process associated with a virtual IP address registers with the virtual address resolution facility 24 (step 140). The registration is stored in a VARP lookup table accessible over the network. Subsequently, a message is received at the electronic device which bears a MAC/network interface address of the network interface of the electronic device with the designated real IP address listed in the virtual address registration (Step 142). The MAC address header 117 is stripped off at the Link layer and the message and appended headers are passed up to the Network layer 114(Step 144). The Network layer 114 strips off the IP header 115 and identifies the Transport Protocol header 113 underneath. The Network layer passes the message and remaining appended header to the Transport layer 112 (Step 146). The Transport layer 112 strips off the Transport header and then passes the message to the Application layer via the port address of the interface identified in the VARP registration. (Step 148). The receiving process determines whether it is the virtual destination address the message is addressed to (Step 149) and acts accordingly. If the receiving process that is examining the message is the destination address, the process examines the data (Step 150). Alternatively, if the message is not intended for the process executing on the edge device, the process re-routes the message and sends it back down the protocol model stack, on to the local network operating behind the edge device, and on to its intended destination by performing traditional VARP and ARP lookups. Alternatively forwarding the

Claim 30	Mehta
	message may be accomplished by reinserting the packet into the IP layer of the networking stack." Caronni II at 7:33–67. *See also* Caronni II at Claim 1, FIGS. 2-4. "FIG. 3 is a data flow diagram illustrating the registration of virtual network environment parameters. The VNE framework 200 is a software module that processes transactions between the applications and the operating system. The VNE parameters are registered with the VNE framework 200 at the time the application is started. The VNE parameters include the application IP address, the virtual network subnet, and the global virtual address subnet. At step 250, the registration harness 220 supplies the application IP address, virtual subnet, and the global virtual address subnet for a $process_x$ to the VNE framework 200. The VNE framework 200 then records the IP address, virtual subnet, and the global virtual address subnet for the $process_x$ at step 252. The $process_x$ can then spawn additional processes (or create additional objects) at step 254. the new $process_y$ inherits the IP address, virtual subnet, and global virtual address subnet from $process_x$. At step 256, the registration harness 220 launches the application related to $process_y$." Hipp at 3:52-4:2. "Virtualization of network identity is achieved by assigning a unique virtual IP address and virtual hostname to a group of processes that make up the application instance which the instance keeps throughout its execution. This virtual network identity stays with the application instance regardless of which node the application is running on. The framework, in essence, provides a mechanism to create this virtual network identity (VNI) around the application using the virtual network parameters assigned to it. In one embodiment, the virtual network parameters include an IP address and hostname. The framework 200 ensures that the application's instance uses the virtual network parameters, transparently, so that it can be moved across machines, without modifications to the application."

Claim 30	Mehta
	Hipp at 6:1-15. *See also* Hipp at 6:16-60, FIG. 3.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	"Methods and systems for providing two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP, are provided. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporarily use by a requesting device on a public network to communicate with a wireless device on a wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more data repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the public network to send data to the wireless device." Mehta at Abstract. "To insure a greater degree of security, according to one embodiment, the AMPS maintains a Time to Live (TTL) parameter with each address mapping. In this way, once the TTL value indicates that the mapping has expired, the AMPS can destroy the mapping, and also any connection. Further, in some embodiments, the AMPS also puts a timestamp in its own mapping tables. After some timeout period based upon the timestamp, the AMPS can destroy the mapping, thereby forcing a new mapping to be initiated on a periodic basis."

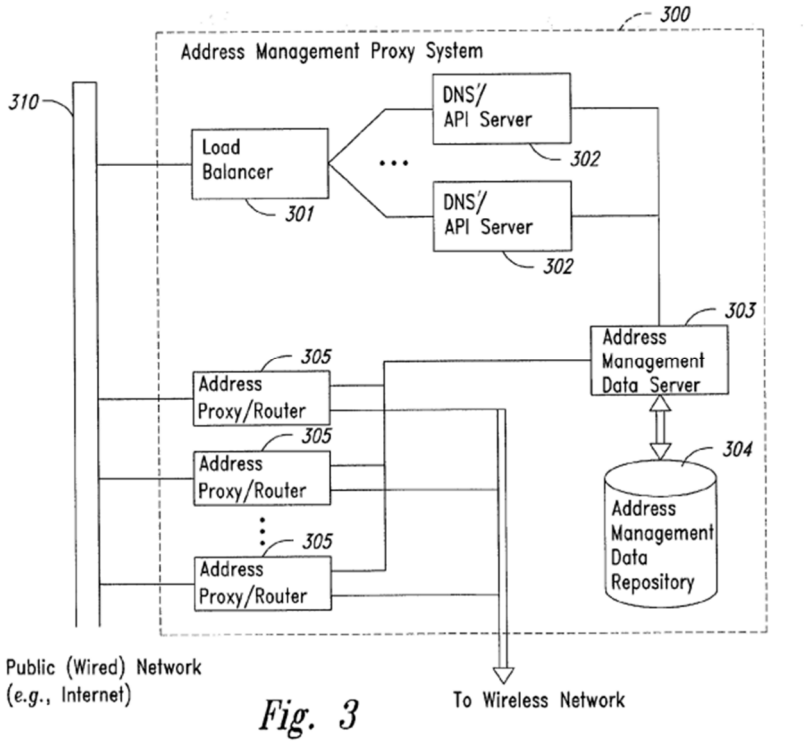
Claim 30	Mehta
	Mehta at [0015]. "In existing systems, data communication (communication on a data channel) between a wireless device that is connected to a private wireless network and a wired device connected to a public wired network (e.g., the Internet) can only be initiated by the wireless device. Some carriers have assigned fixed public IP addresses to wireless devices; however, the wireless devices need to then have client programs (e.g., a UDP stack) capability of receiving and handling the incoming communication packets. Moreover, these wireless devices are then part of a public wireless network and not a private wireless network. Since public IP addresses are becoming a scarcer commodity and currently expensive to service a network of multi-million devices, carriers cannot in a practical sense count on having a fixed public IP address for each device on its network. (Although movement to IPv6 will allow more addresses, the current IPv4 definitions are limited and the potential number of wireless users subscribing to larger carriers is very high. In some regions of the world, the current public address scheme is even more limited.) Further, such addressing capability would expose each device to further security risks, because each such device is part of a publicly accessible network and it becomes more difficult to implement and enforce security measures. Thus, assigning private IP addresses to devices is preferred in existing wireless networks over assigning fixed public IP addresses to devices. When wireless networks are private, the locations (addresses) of the wireless devices are intentionally hidden from public view by a carrier system's (or, as referred to in some countries, operator's) infrastructure." Mehta at [0025]. "FIG. 3 is an example block diagram of components of an example Address Management Proxy System. In one embodiment, the Address Management Proxy System (AMPS) comprises one or more modified DNS'/ API servers 302, one or more Address Proxy/Routers 305, an Address Management Data Server 303 which manages a database or other repositories such as Address Management Data Repository 304, and optionally a load balancer 301. The DNS'/API servers 302 are

Claim 30	Mehta
	either individually connected to a public network 310 or are connected to the load balancer 301 which is in turn connected to public network 310. Similarly, each Address Proxy/Router 305 is also connected to the public network 310, via the routable (public) address to which data from the external network 310 is sent that is destined for the wireless devices. The DNS'/API servers 302 are either modified implementations of a DNS server to add functionality necessary to communicate with wireless devices, or are servers that implement one or more specialized APIs, as will be described further below. The DNS'/API servers 302 use the Address Management Data Server 303 to assist in mapping a unique identifier (e.g., string name) for a wireless device to a public address on public network 310. The pool of public addresses is also maintained by the Address Management Data Serve 303 and Data Repository 304. The Address Management Data server 303 and Address Management Data Repository 304 may be implemented using existing database technology, for example, ODBC technology or, may be implemented as a structure such as a simple text file. One skilled in the art will recognize that any embodiment for storing a set of tables, data, lists, or mappings can be used. Each Address Proxy/Router 305 also uses the Address Management Data Server 303 or equivalent to create and update a series of routing tables that are used to assign public addresses to wireless devices as needed and to update the various mappings between public addresses and the non-routable (private) addresses of the wireless devices. The tables and mappings that are maintained on behalf of the DNS'/API servers 302 and the Address Proxy/Routers 305 by the Address Management Data Server 303 are described below with reference to FIG. 6." Mehta at [0032]. "Although the techniques of the AMPS are generally applicable to any a wired device communicating with a wireless device, the phrase "public network" (or "wired network") is used generally to imply any type of internetworked environment including a public network or a backbone that is somewhere down the line connected to one or more private or public networks. In addition, although the examples described herein often refer to the Internet, one skilled in the art will recognize that the concepts and inventions described are applicable to other forms

Claim 30	Mehta
	and embodiments of internetworking, including, for example ATM type networks. Thus, techniques of the present invention can also be used by one device on a first wireless network to communicate with another wireless device on a second network—each device ends up communicating with the Address Proxy/Router of the other network. This scenario is feasible because each wireless network (or its carrier infrastructure) is connected to a proxy/router that is also connected (via a public address) to a public network. In addition, although a public network is sometimes also referred to herein as a wired network, one skilled in the art will recognize that any network that exposes routable (public) addresses may be implied. Thus, a wireless network with unique public (and routable) address can also employ techniques of the present invention to perform bi-directional communication. Also, one skilled in the art will recognize that terms such as wireless device, phone, handheld, etc., are used interchangeably to indicate any type of wireless device that is capable of operating with the AMPS. In addition, terms may have alternate spellings which may or may not be explicitly mentioned, and one skilled in the art will recognize that all such variations of terms are intended to be included." Mehta at [0033]. "Example embodiments described herein provide applications, tools, data structures and other support to implement private to public address mappings over one or more wired and wireless networks to be used for bi-directional communication. One skilled in the art will recognize that other embodiments of the methods and systems of the present invention may be used for many other purposes, including to push information and/or data or code from a public network such as the Internet to a wireless device. In addition, although this description primarily refers to "data" as being sent via the networks, one skilled in the art will recognize that all types of data can be communicated using the techniques described herein including, but not limited to, text, graphics, audio, and video." Mehta at [0034].

Claim 30	Mehta
	"There are several implementation approaches to the components of the Address Management Proxy System, three of which are described herein. One skilled in the art will recognize that various other approaches and combinations are possible. All three approaches allocate a public (routable) network address for temporary use by a wired device to communicate with a wireless device. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained and allocated dynamically to wireless network devices as required. For example, a typical Class B Internet network address block allows for approximately 65,000 simultaneous connections to wireless devices. Although this number may seem large at first blush, when one considers the number of cell phones and handsets, for example, connected to a carrier, this number can be quite limiting." Mehta at [0039]. "The unique ID table 610 maps unique string names of wireless devices to the private wireless network addresses that have been assigned typically by a carrier's infrastructure. In some embodiments, the carrier infrastructure dynamically allocates, using methods similar to a DHCP protocol, a private wireless network address when the wireless device registers itself with the carrier infrastructure upon being powered on. Thus, the unique ID table 610 may be sparsely filled or entries created dynamically and then deleted dynamically as devices register and unregister with the carrier infrastructure system." Mehta at [0053]. "In one embodiment, a timestamp of the mapping between a public network address and a private address is also noted in table 630. After a defined time out, based upon this timestamp, the Address Management Data Server sends a request to the Address Proxy/Router associated with the mapping to unmap the public-to-private address mapping and updates the mapping table 630, thereby returning the associated public address to the pool of unused public network addresses." Mehta at [0056].

Claim 30	Mehta
	"FIG. 7 is an example flow diagram of an example routine provided by a DNS'/API server of the Address Management Proxy System to return a public address that corresponds to a designated unique identifier. In essence, this routine implements a DNS query or DNS-like query capability for the AMPS using a modified GetHostByName interface or a specialized API, for example GetProxyIP, as described with reference to FIG. 5. In summary, the routine dynamically allocates an appropriate Address Proxy/Router machine to associate with the wireless device and returns the public address of that machine (along with a TTL parameter and potentially other parameters). Specifically, in step 701, the routine determines the private (non-routable) address of the wireless device designated by a string parameter passed as input to the routine. For example, the string parameter may use fields such as "uniqueID.hostname.domain.tld," which specifies a typical hierarchy of person/service on a machine named "hostname" on a domain such as a company's network on a top level domain such as "org." "com," "edu," etc. One skilled in the art will recognize that many other string parameter designations could be used. One mechanism for implementing this routine is to request information from the Address Management Data Repository. In one embodiment, the data repository stores a table that maps unique IDs to private network addresses (see Table 610 in FIG. 6). Preferably, any mechanism that is used by the AMPS stores this data in a secure manner in order to keep the wireless network addresses private. In step 702, the routine retrieves the public network address that corresponds to the private wireless network address of the designated device if one has already been assigned by the AMPS to that device and is still valid. In one embodiment, the data repository stores this mapping information between private wireless network address and public network address (see for example table 630 in FIG. 6). If a public network address has not already been assigned or is not valid, then the routine causes a new public network address to be allocated and that new public address is associated with the private wireless network address. Appropriate tables in the data repository are then updated. In step 703, the routine determines the Address Proxy/Router machine that is associated with the assigned public network address (for example using table 620 in FIG. 6). In step 704, the routine sends a request to the determined proxy/router machine to update its routing tables to map

Claim 30	Mehta
	the determined public network address to the private wireless network address. In step 705, the routine updates information in the data repository to include any other connection related information (e.g., field 634 in Table 630 in FIG. 6) and indicates a Time to Live (TTL) parameter for the public-private address association (e.g., field 633 in Table 630 in FIG. 6). Once all of the tables have been updated in both the proxy/router and in the data repository, the DNS'/API server returns the determined public network address of the associated Address Proxy/Router machine. As described earlier, the public address may be a (hostname, port) pair when a port-based implementation is used." Mehta at [0057].  *Fig. 3*

Claim 30	Mehta
	Mehta at FIG. 3. To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Caronni I, Caronni II and/or Hipp. Caronni I, Caronni II and Hipp concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, distributing virtual network addresses was a well-known and commonly-used technique. Modifying this reference with Caronni I's, Caronni II's, and/or Hipp's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with those of Caronni I, Caronni II, or Hipp for example, to incorporate this limitation. *See, e.g.*, "Fourth, the system provides address translation in a transparent manner. Since the Supernet is a private network constructed from the infrastructure of another network, the Supernet has its own internal addressing scheme, separate from the addressing scheme of the underlying public network. Thus, when a packet from a Supernet node is sent to another Supernet node, it travels through the public network. To do so, the Supernet performs address translation from the internal addressing scheme to the public addressing scheme and vice versa. By separating the addressing schemes, the Supernet creates a flexible delivery scheme that is easily changeable by network software or a system administrator. To reduce the complexity of Supernet nodes, system-level components of the Supernet perform this translation on behalf of the individual nodes so that it is transparent to the nodes. Another benefit of the Supernet's addressing is that it uses an IP-based internal addressing scheme so that preexisting programs require little modification to run within a Supernet." Caronni I at 6:7–25.

Claim 30	Mehta
	"SASD 540 represents a Supernet: there is one instance of an SASD per Supernet, and it both authenticates nodes and authorizes nodes to join the Supernet. VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as a node IDs, to the network addresses recognized by the public-network infrastructure, known as the real addresses. The "node ID" may include the following: a Supernet ID (e.g., 0x123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1). Although the virtual address is described in an IP address scheme, one skilled in the art will appreciate that the virtual address may be any other type addressing scheme, such as an e-mail address, IPX, or IPv6. Since the node ID includes a Supernet ID, a node will have more than one node ID when it communicates over more than one channel. The "real address" is an IP address (e.g., 10.0.0.2) that is globally unique and meaningful to the public-network infrastructure. In a Supernet, one VARPD runs on each machine, and it may play two roles. First, a VARPD may act as a server by storing all address mappings for a particular Supernet into its associated VARPDB. Second, regardless of its role as a server or not, each VARPD assists in address translation for the nodes on its machine. In this role, the VARPD stores into its associated VARPDB the address mappings for its nodes, and if it needs a mapping that it does not have, it will contact the VARPD that acts as the server for the given Supernet to obtain it. The VARPDB may also decide which virtual address to use in the translation. That is, the VARPDB may associate a virtual address with multiple real addresses or vice versa." Caronni I at 7:3–33. "FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively,

Claim 30	Mehta
	the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:66-10:18. "Once inner IP layer 542 receives the packet, a Supernet ID is appended to a socket structure (step 804). The socket structure is modified so as to contain an extra data field for Supernet ID 626 and virtual source address 642. The addition of Supernet ID 626 and virtual address 642 in the socket structure enables the Supernet to communicate with nodes regardless of the delivery scheme used. When the process on node A opens a socket to transmit the packet to inner IP layer 540, the corresponding Supernet ID 626 and virtual source address 642 for that process is included in the socket request. The packet and Supernet ID are then transmitted to the SNSL layer using the modified socket structure (step 806). The SNSL layer then accesses the VARPDB to obtain the address mapping between virtual source node address 642 and the source real address 614 as well as the virtual destination node address 644 and the destination real address 616 (step 808). If they are not contained in the VARPDB because this is the first time a packet has been sent from this node or sent to this destination, the VARPDB accesses the local VARPD to obtain the mapping. When contacted, the VARPD on the local machine contacts the VARPD that acts as the server for the Supernet to obtain the appropriate address mapping. Since the VARPDB maintains all real IP addresses, a remote node may securely communicate with another remote node without reverfication." Caronni I at 11:37–61. *See also* Caronni I at FIGS. 7A and 7B.

Claim 30	Mehta
	"Unfortunately, when the virtual destination address is located on a physical device on the interior of a network which is running a proxy server, firewall, or other packet filtering mechanism, messages that have been sent to a virtual destination address have difficulty getting all the way to their target. The term "interior of a network" refers to devices which are not able to directly access another network without first going through another device on their own network. For example, most local area networks (LANs) access the Internet through a proxy server. Devices other than the proxy server are said to be on the interior of the LAN. The proxy server is referred to as an "edge device" because it is able to directly contact another network without using an intermediary device. "Packet filtering" refers to the filtering of incoming messages or packets by an edge device or process on an edge device so that not all of the packets are permitted to proceed to their destination, they are "filtered out". If the electronic device that is filtering incoming packets, is under the control of the party executing the process associated with the virtual destination address, the device may be configured to allow the packets through to the end destination. However, in many situations, the edge device is not configurable by anyone without system administration privileges. Similarly, if the edge device is a device performing Network Address Translation (a "NAT box"), the NAT box rewrites all outgoing packets from an end user in the interior of the network to make them look like they came directly from the NAT box, and remembers that any traffic coming back from the particular destination address must be mapped back to the originating internal device. Consequently, the responding devices think they are responding to the sending device when they are actually responding to an edge device. In such a case, the packets intended for the virtual destination address on the interior of the physical network may be dropped and not reach their intended destination." Caronni II at 1:60–2:27. "…sending said formatted message, augmented by said determined MAC address destination, said real IP address destination and said transport protocol header, from

Claim 30	Mehta
	a virtual IP address to the real IP address destination indicated in said virtual address destination registration." Caronni II at 9:20–24. "…a process associated with the sending virtual IP address is located on an electronic device outside a firewall and said virtual address destination is located inside the firewall." Caronni II at 9:31–32. *See also* Caronni II at 2:31-3:24, 6:32-54, FIGS. 1-5. "The Virtual Network Environment (VNE) of the present invention is defined by a collection of IP addresses. An application running within one VNE can communicate with another application in the same VNE. However, an application in one VNE cannot communicate with an application in another VNE (unless expressly permitted). These and many other attendant advantages of the present invention will be understood upon reading the following detailed description in conjunction with the drawings." Hipp at 2:8-16. "The VNE is specified at application run time. The VNE is transparent to the application and does not require any modifications to the application. The VNE is defined by subnet of addresses contained within the VNE. For example, all applications within the subnet 10.10.2.0 comprise a VNE. The subnet/netmask specifying such a VNE would be 10.10.2.0/255.255.255.0 and would include the addresses 10.10.2.0 through 10.10.2.255. In this example, an application with IP address 10.10.2.2 would be able to communicate with an application at address 10.10.2.60, but not at 10.10.0.1. Although using a subnet/netmask to specify the VNE is described herein for illustrative purposes, it is to be understood that other methods may be used to accomplish the same mechanism (e.g. an access control list)." Hipp at 3:16-30.

Claim 30	Mehta
	"Virtualization of network identity is achieved by assigning a unique virtual IP address and virtual hostname to a group of processes that make up the application instance which the instance keeps throughout its execution. This virtual network identity stays with the application instance regardless of which node the application is running on. The framework, in essence, provides a mechanism to create this virtual network identity (VNI) around the application using the virtual network parameters assigned to it. In one embodiment, the virtual network parameters include an IP address and hostname. The framework 200 ensures that the application's instance uses the virtual network parameters, transparently, so that it can be moved across machines, without modifications to the application." Hipp at 6:1-15. *See also* Hipp at Figures 1–7.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	"A network that supports IP can be connected to another TCP/IP-based or UDP/IP-based internet or the Internet, by providing a router which forwards data to another router or host machine based upon an IP address. The router (or routing server/service) typically contains a routing table, which determines to which machine (and optionally to which port) to send a datagram, given a destination IP address. The IP address uniquely identifies a router/host machine, and, in a typical TCP/IP network, can be mapped to a string name that identifies, for example, a particular machine as part of a larger domain. (Although referred to herein often as a TCP/IP-based network, one skilled in the art will recognize that the network may also be UDP-based (connectionless), and may support another session management system.)" Mehta at [0008]. "In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more Address Management Data Repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a

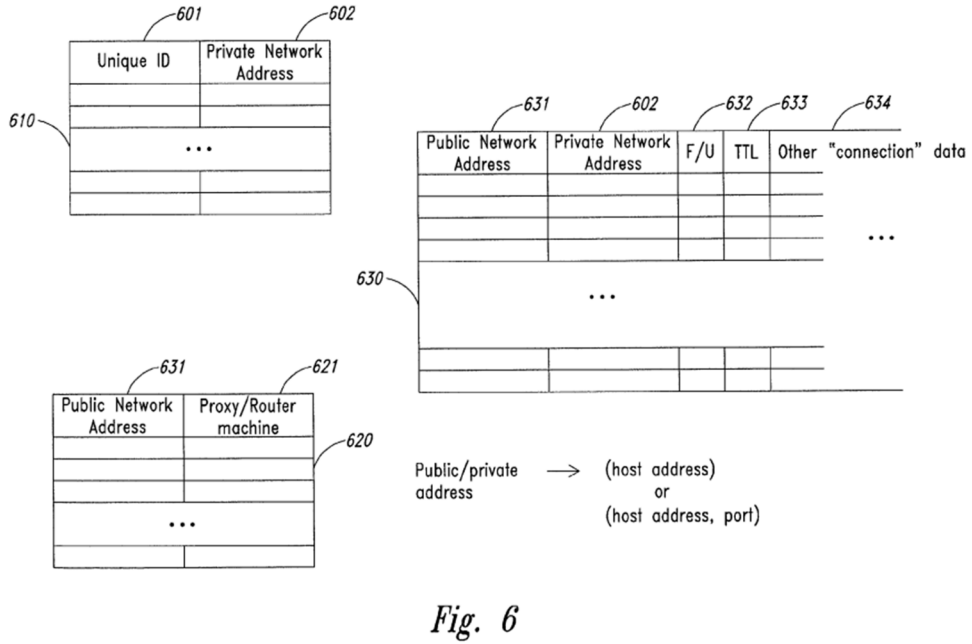
Claim 30	Mehta
	particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the external public network to send data to the wireless device. The temporary public address is, for example, an address associated with one of the Address Proxy/Routers, which are connected to the external public network and have access to the private addresses on the private wireless network. In some cases, the device on the public network that desires to send data to the wireless device uses a connection-based protocol, such as TCP/IP to send data. In other cases the device uses a connection-less protocol, such as UDP (UDP/IP) to send the data." Mehta at [0012]. "Wireless systems on private networks use techniques similar to Network Address Translation (NAT) technology to send data from a wireless device to the public networking world. In a typical carrier infrastructure that uses a private network, a wireless device "registers" itself with the carrier infrastructure when it powers on (or in other circumstances, when the device attempts to initiate data services). The carrier dynamically assigns the wireless device a private non-routable address by means of DHCP (or DHCP-like) server. The information on the mapping of the transient private IP address to the device public IP address is stored in an internal carrier database and managed by carrier services such as a RADIUS server." Mehta at [0026]. "The Address Management Proxy System achieves two way initiated bi-directional communication by implementing a modified DNS server and serving as a proxy/router for devices on the private wireless network as they interface to the public wired internetworking world. In summary, a pool of public addresses is maintained and dynamically distributed among active wireless devices as needed by the AMPS. FIG. 2 is a block diagram of an example Address Management Proxy System used in bi-directional communication with a wireless device. The term "bi-directional" as used herein means that data paths and communication can flow in either direction between two endpoint systems. FIG. 2 shows wired devices

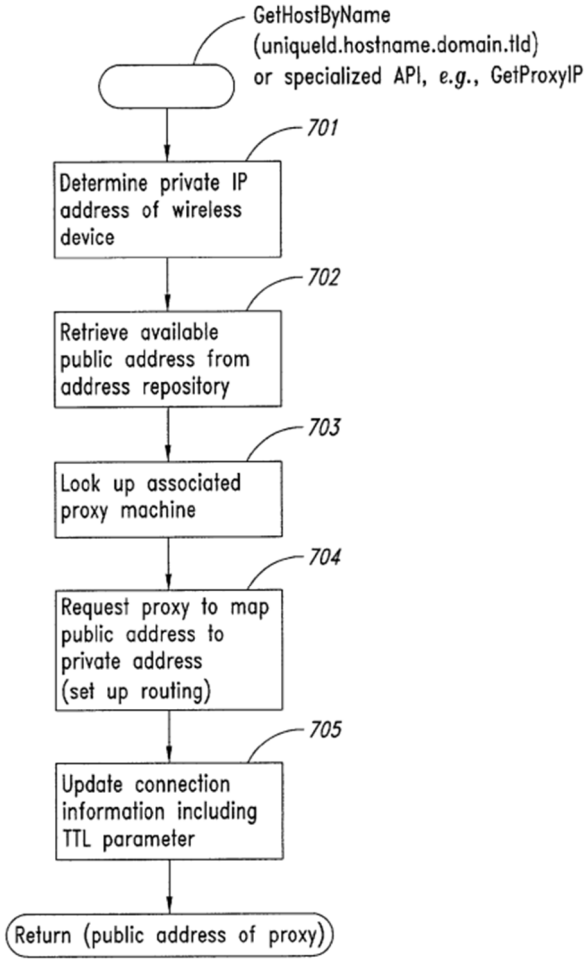
Claim 30	Mehta
	201, 202, and 203 connected to public network 210. One skilled in the art will recognize that these devices could be connected to another private or public network, which is then connected by one or more wired devices to the public network 210, yet still achieve the functionality discussed here. Any such variation provides equivalent functionality and is explicitly contemplated and presumed to be part of the present invention. On the wireless side, the AMPS 220, acting in its capacity as a proxy (and router) for wireless devices, is shown both connected via wire to the public network 210 and via standard carrier infrastructure elements (not shown) to the wireless network 230. It is presumed that the reader has a working knowledge of the elements of a carrier's infrastructure and the basic mechanisms for routing and mechanisms converting packets from a wired network to a wireless network. These may use analog or digital technology and may require protocol conversions in order to send the physical data and transmit it, for example through a satellite, ultimately to the wireless device. Detailed background information on wireless technology and wireless routing mechanisms is described in Stallings, W., Wireless Communications and Networks, Prentice Hall, N.J., 2002, which is herein incorporated by reference in its entirety. In FIG. 2, wireless device 240 is shown connected via various wireless elements (not shown) to the wireless network 230." Mehta at [0030]. "FIG. 6 is an example block diagram of some of the Address Management Proxy System data repository tables used to support routines of the DNS'/API servers and the Address Proxy/Routers. In one embodiment, the Address Management Data Repository comprises three tables: a unique identifier (unique ID) to private address table 610, a public-to-private address table 630, and a public address to proxy/router machine table 620. Although three tables are shown, one skilled in the art will recognize that these tables could contain other data and may be organized differently, including a different number of tables and with different columns or fields. In addition, any technique for storing a table or list of data may be used. To support embodiments that map a host address plus a port designator to a wireless device, the tables are correspondingly modified." Mehta at [0052].

Claim 30	Mehta
	"The unique ID table 610 maps unique string names of wireless devices to the private wireless network addresses that have been assigned typically by a carrier's infrastructure. In some embodiments, the carrier infrastructure dynamically allocates, using methods similar to a DHCP protocol, a private wireless network address when the wireless device registers itself with the carrier infrastructure upon being powered on. Thus, the unique ID table 610 may be sparsely filled or entries created dynamically and then deleted dynamically as devices register and unregister with the carrier infrastructure system." Mehta at [0053]. The public-to-private address table 630 comprises several fields/columns including a public network address 631, a private (wireless) network address 602, a flag 632 that specifies whether the public address stored in field 631 is free or is already used, a Time to Live (TTL) parameter 633, and other connection data 634. In one embodiment, the DNS'/API servers of the AMPS query table 630 to determine a public network address that corresponds to a designated private wireless network address or to allocate an unused public network address (as indicated in field 632) and map the determined unused public address to a private network address stored in field 602. Mehta at [0054]. Public address to proxy/router machine table 620 comprises a public network address field 631 and an indication of a functioning proxy/router machine 621. By maintaining such a mapping, the AMPS is able to substitute proxy/router machines for other proxy/router machines to provide a higher degree of robustness. Each proxy/router machine has a preconfigured set of public network addresses, such as are typically configured by network cards inserted into the proxy/router machine. These address are allocated in a standard fashion through prior purchase or obtaining from an address authorizing authority, currently the Internet Corporation for Assigned Names and Numbers (ICANN). When a machine is inserted for use into the AMPS, indications of these addresses are stored in field 631.

Claim 30	Mehta
	Mehta at [0055]. "In one embodiment, a timestamp of the mapping between a public network address and a private address is also noted in table 630. After a defined time out, based upon this timestamp, the Address Management Data Server sends a request to the Address Proxy/Router associated with the mapping to unmap the public-to-private address mapping and updates the mapping table 630, thereby returning the associated public address to the pool of unused public network addresses." Mehta at [0056]. "FIG. 7 is an example flow diagram of an example routine provided by a DNS'/API server of the Address Management Proxy System to return a public address that corresponds to a designated unique identifier. In essence, this routine implements a DNS query or DNS-like query capability for the AMPS using a modified GetHostByName interface or a specialized API, for example GetProxyIP, as described with reference to FIG. 5. In summary, the routine dynamically allocates an appropriate Address Proxy/Router machine to associate with the wireless device and returns the public address of that machine (along with a TTL parameter and potentially other parameters). Specifically, in step 701, the routine determines the private (non-routable) address of the wireless device designated by a string parameter passed as input to the routine. For example, the string parameter may use fields such as "uniqueID.hostname.domain.tld," which specifies a typical hierarchy of person/service on a machine named "hostname" on a domain such as a company's network on a top level domain such as "org." "com," "edu," etc. One skilled in the art will recognize that many other string parameter designations could be used. One mechanism for implementing this routine is to request information from the Address Management Data Repository. In one embodiment, the data repository stores a table that maps unique IDs to private network addresses (see Table 610 in FIG. 6). Preferably, any mechanism that is used by the AMPS stores this data in a secure manner in order to keep the wireless network addresses private. In step 702, the routine retrieves the public network address that corresponds to the private wireless network address of the designated device if one has already been

Claim 30	Mehta
	assigned by the AMPS to that device and is still valid. In one embodiment, the data repository stores this mapping information between private wireless network address and public network address (see for example table 630 in FIG. 6). If a public network address has not already been assigned or is not valid, then the routine causes a new public network address to be allocated and that new public address is associated with the private wireless network address. Appropriate tables in the data repository are then updated. In step 703, the routine determines the Address Proxy/Router machine that is associated with the assigned public network address (for example using table 620 in FIG. 6). In step 704, the routine sends a request to the determined proxy/router machine to update its routing tables to map the determined public network address to the private wireless network address. In step 705, the routine updates information in the data repository to include any other connection related information (e.g., field 634 in Table 630 in FIG. 6) and indicates a Time to Live (TTL) parameter for the public-private address association (e.g., field 633 in Table 630 in FIG. 6). Once all of the tables have been updated in both the proxy/router and in the data repository, the DNS'/API server returns the determined public network address of the associated Address Proxy/Router machine. As described earlier, the public address may be a (hostname, port) pair when a port-based implementation is used." Mehta at [0057].

Claim 30	Mehta
	 *Fig. 6* Mehta at FIG. 6.

Claim 30	Mehta
	 *Fig. 7*

Claim 30	Mehta
	Mehta at FIG. 7. *See also* Mehta at [0004], [0010], [0032], [0036]-[0038], [0048], [0058]-[0059]. To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as RFC 1383, Caronni I, Caronni II, and/or Hipp. RFC 1383, Caronni I, Caronni II, and/or Hipp concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, use of a server such as a DNS server that returns network addresses was a well-known and commonly-used technique. Modifying this reference with RFC 1383's, Caronni I's, Caronni II's, and/or Hipp's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with those of RFC 1383, Caronni I, Caronni II, or Hipp for example, to incorporate this limitation. *See, e.g.*, "1. Routing, scaling and hierarchies Several recent studies have outlined the risk of "routing explosion" in the current Internet: there are already more than 5000 networks announced in the NSFNET routing tables, more than 7000 in the EBONE routing tables. As these numbers are growing, several problems occur: * The size of the routing tables grows linearly with the number of connected networks; handling this larger tables requires more resources in all "intelligent" routers, in particular in all "transit" and "external" routers that cannot rely on default routes.

Claim 30	Mehta
	* The volume of information carried by the route exchange protocols such as BGP grows with the number of networks, using more network resources and making the reaction to routing events slower. * Explicit administrative decisions have to be exercised by all transit networks administrators which want to implement "routing policies" for each and every additional "multi-homed" network. The current "textbook" solution to the routing explosion problem is to use "hierarchical routing" based on hierarchical addresses. This is largely documented in routing protocols such as IDRP, and is one of the rationales for deploying the CIDR [3] addressing structure in the Internet. This textbook solution, while often perfectly adequate, as a number of inconveniences, particularly in the presence of "multihomed stubs", e.g., customer networks that are connected to more than one service providers. The current proposal presents a scheme that allows for simple routing. It is complementary with the classic "hierarchical routing" approach, but provides an easy to implement and low cost solution for "multi-homed" domains. The solution is a generalization of the "MX record" scheme currently used for mail routing." RFC 1383 at 1-2. "3.1. Loops and relays In the introduction to DNS-IP routing, we mentioned that the packets would be directed towards the access gateway I1 or I2 by means of "source routing" or "tunnelling". This is not, stricto sensu, necessary. One could imagine that the packet would simply be routed "as if it was directed towards I1 or I2". The next relay would, in turn, also access the DNS to get routing information and forward the packet.

Claim 30	Mehta
	Such a strategy would have the advantage of leaving the header untouched and of letting the transit nodes choose the best routing towards the destination, based on their knowledge of the reachability status. It would however have two important disadvantages: - It would oblige all intermediate relays to access the DNS, - It would oblige all these relays to exploit consistently the DNS information. Obliging all intermediate gateways to access the DNS is impractical in the short term: it would mean that we would have to update each and every transit relay before deploying the scheme. It could also have an important performance impact: the "working set" of transit relays is typical much wider than that of stub gateways, and the argument presented previously on the efficiency of caches may not apply. This would perhaps remain impractical even in the long term, as it the volume of DNS traffic could well become excessive. The second argument would apply even if the performance problem had been solved. Suppose that several RX records are registered for a given destination, such as I1 and I2 for Dx in our example, and that a "hop by hop routing" strategy is used. There would be a fair risk that some relays would choose to route the packet towards I1 and some others towards I2, resulting in inefficient routing and the possibility of loops. In order to ensure coherency, we propose that all routing decisions be made at the source, or by one of the first relays near the source." RFC 1383 at 4-5. "3.4. Choosing a gateway A simplification to the previous problem would be to allow only one RX record per destination, thus guaranteeing consistent decisions in the network. This would

Claim 30	Mehta
	however have a number of draw-backs. A single access point would be a single point of failure, and would be connected to only one transit network thus keeping the "customer locking" effect of hierarchical routing. We propose that the RX records have a structure parallel to that of MX records, i.e., that they carry associated with each gateway address a preference identifier. The source host, when making the routing decision based on RX records, should do the following: - List all possible gateways, - Prune all gateways in the list which are known as "unreachable" from the local site, - If the local host is present in the list with a preference index "x", prune all gateways whose preference index are larger than "x" or equal to "x". - Choose one of the gateway in the list. If the list is empty, consider the destination as unreachable. Indeed, these evaluations should not be repeated for each and every packet. The routers should maintain a cache of the most frequently used destinations, in order to speed up the processing." RFC 1383 at 6. "5.1. DNS record In a definitive scheme, it would be necessary to define a DNS record type and the corresponding "RX" format. In order to deploy this scheme, we would then have to teach this new format to the domain name service software. While not very difficult to do, this would probably take a couple of month, and will not be used in the early experimentations, which will use the general purpose "TXT" record.

Claim 30	Mehta
	This record is designed for easy general purpose extensions in the DNS, and its content is a text string. The RX record will contain three fields: - A record identifier composed of the two characters "RX". This is used to disambiguate from other experimental uses of the "TXT" record. - A cost indicator, encoded on up to 3 numerical digits. The corresponding positive integer value should be less that 256, in order to preserve future evolutions. - An IP address, encoded as a text string following the "dot" notation. The three strings will be separated by a single comma. An example of record would thus be: | domain - | type | record | value | |*.27.32.192.in-addr.arpa | IP | TXT | RX, 10, 10.0.0.7| which means that for all hosts whose IP address starts by the three octets "192.32.27" the IP host "10.0.0.7" can be used as a gateway, and that the preference value is 10." RFC 1383 at 11-12. *See also* RFC 1383 at 9, 12-13. "To perform this functionality, D 1 108 utilizes a technique known as tunneling to ensure that the communication between itself and enterprise network 102 is secure in that it cannot be viewed by an interloper. "Tunneling" refers to encapsulating one packet inside another when packets are transferred between two end points (e.g., D 1 108 and VPN software 109 running on firewall 106). The packets may be encrypted at their origin and decrypted at their destination. For example, FIG. 2A depicts a packet 200 with a source Internet protocol (IP) address 202, a destination

Claim 30	Mehta
	IP address 204, and data 206. It should be appreciated that packet 200 contains other information not depicted, such as the source and destination port. As shown in FIG. 2B, the tunneling technique forms a new packet 208 out of packet 200 by encrypting it and adding both a new source IP address 210 and a new destination IP address 212. In this manner, the contents of the original packet (i.e., 202, 204, and 206) are not visible to any entity other than the destination. Referring back to FIG. 1, by using tunneling, remote device D 1 108 may communicate and utilize the resources of the enterprise network 102 in a secure manner." Caronni I at 2:26–46. "SASD 540 represents a Supernet: there is one instance of an SASD per Supernet, and it both authenticates nodes and authorizes nodes to join the Supernet. VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as a node IDs, to the network addresses recognized by the public-network infrastructure, known as the real addresses. The "node ID" may include the following: a Supernet ID (e.g., 0x123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1). Although the virtual address is described in an IP address scheme, one skilled in the art will appreciate that the virtual address may be any other type addressing scheme, such as an e-mail address, IPX, or IPv6. Since the node ID includes a Supernet ID, a node will have more than one node ID when it communicates over more than one channel. The "real address" is an IP address (e.g., 10.0.0.2) that is globally unique and meaningful to the public-network infrastructure. In a Supernet, one VARPD runs on each machine, and it may play two roles. First, a VARPD may act as a server by storing all address mappings for a particular Supernet into its associated VARPDB. Second, regardless of its role as a server or not, each VARPD assists in address translation for the nodes on its machine. In this role, the VARPD stores into its associated VARPDB the address mappings for its nodes, and if it needs a mapping that it does not have, it will contact the VARPD that acts as the server for the given Supernet to obtain it. The VARPDB may also decide which virtual address to use in the translation. That is,

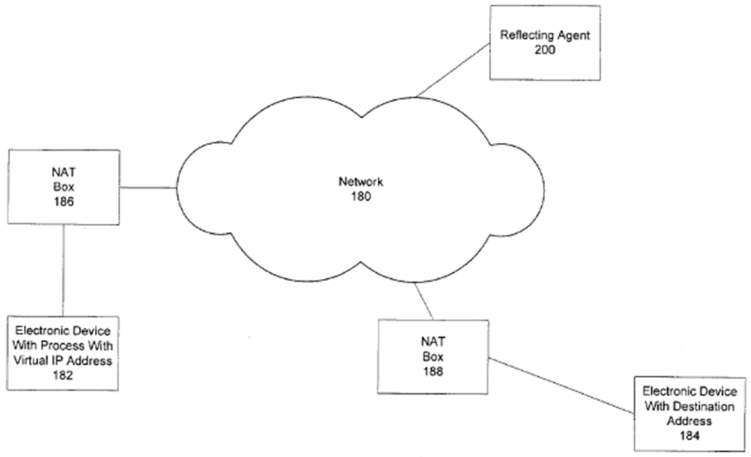
Claim 30	Mehta
	the VARPDB may associate a virtual address with multiple real addresses or vice versa." Caronni I at 7:3–33. "SNSL 542 utilizes VARPDB 536 to perform address translation. VARPDB stores all of the address mappings encountered thus far by SNSL 542. If SNSL 542 requests a mapping that VARPDB 536 does not have, VARPDB communicates with the VARPD 526 on the local machine to obtain the mapping. VARPD 526 will then contact the VARPD that acts as the server for this particular Supernet to obtain it." Caronni I at 9:47–54. "FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:66-10:18. "The packet and Supernet ID are then transmitted to the SNSL layer using the modified socket structure (step 806). The SNSL layer then accesses the VARPDB to obtain the address mapping between virtual source node address 642 and the

Claim 30	Mehta
	source real address 614 as well as the virtual destination node address 644 and the destination real address 616 (step 808). If they are not contained in the VARPDB because this is the first time a packet has been sent from this node or sent to this destination, the VARPDB accesses the local VARPD to obtain the mapping. When contacted, the VARPD on the local machine contacts the VARPD that acts as the server for the Supernet to obtain the appropriate address mapping. Since the VARPDB maintains all real IP addresses, a remote node may securely communicate with another remote node without reverfication." Caronni I at 11:47–61. "After obtaining the address mapping, the SNSL layer determines whether it has been configured to communicate over the appropriate channel for this packet (step 806). This configuration occurs when SNlogin runs, and if the SNSL has not been so configured, processing ends. Otherwise, SNSL obtains the channel key to be used for this channel (step 808). The SNSL maintains a local cache of keys and an indication of the channel to which each key is associated. Each channel key is time stamped to expire in ten seconds, although this time is configurable by the administrator. If there is a key located in the cache for this channel, SNSL obtains the key. Otherwise, SNSL accesses KMD which then locates the appropriate channel key from the appropriate KMC. After obtaining the key, the SNSL layer encrypts the packet using the appropriate encryption algorithm and the key previously obtained (step 810). When encrypting the packet, the virtual source node address 642, the virtual destination node address 644, and the data may be encrypted (addressing section 660), but the source and destination real addresses 614, 616 (delivery scheme section 670) are not, so that the real addresses can be used by the public network infrastructure to send the packet to its destination. By encrypting addressing scheme 660, the Supernet enables data to be transmitted securely and at the same time transparent from delivery scheme used." Caronni I at 11:62-12:19. "Web client 1102 has a virtual address obtained from computer system 1106, as described in FIGS. 7A and 7B. Each time web client 1102 requests a packet from

Claim 30	Mehta
	web server 1104 a, the client requests the virtual address of the web server 1104 a from computer system 1106. If web server 1104 a becomes overloaded (e.g., unable to handle more requests), the overloaded web server spawns new instances of the same web server (web servers 1104 b and 1104 c) and at the same time notifies the computer system 1106. The VARP server of computer 1106 then associates the new instances 104 b and 104 c of the web server with web server's 1104 virtual address. As a result, web client 1102 is not notified of the change and continues to use the same virtual address as previously." Caronni I at 13:13–26. "In one embodiment, a virtual network supported by a physical network includes a virtual address resolution facility. The virtual address resolution facility is used to resolve a virtual IP address into a real IP address. A virtual address is registered with the virtual address resolution facility and the registration includes a real IP address, a port number and a transport protocol designation associated with a virtual address. The virtual address destination is resolved using the virtual address resolution facility and a message is sent from a virtual address in the virtual network to the real IP address indicated in the virtual address destination registration. In one aspect of the embodiment, the registration also includes an Application layer protocol designation." Caronni II at 2:52–64. "In an embodiment of the present invention a virtual network is supported by a physical network and includes a virtual address resolution facility. The virtual address resolution facility is used to resolve a virtual IP address into a real IP address. A registration request is received at the virtual address resolution facility which includes a real IP address, a port address, a transport protocol designation and an Application layer protocol designation associated with a virtual address destination. The associations are stored using the virtual address resolution facility. A resolution request is received referencing a virtual address destination and the resolution request is resolved using the virtual address resolution facility and the stored associations."

Claim 30	Mehta
	Caronni II at 3:11–23. "The physical network 20 is also interfaced with an electronic device 30 located at the edge of a local area physical network 50 which includes devices addressable using IP addresses in the 129.63.1.0/24 range. The electronic device 30 may be a proxy server, gateway, NAT box, or other device and may perform packet filtering. Alternatively, a firewall 51 may be executed in either a hardware or software form to filter packets sent to the local area physical network 50 from the physical network 20. The firewall 51 may be located outside the electronic device 30 or alternatively may be running in software form on the electronic device 30. Those skilled in the art will recognize that a number of different types of networks may be utilized within the scope of the present invention. The networks may be the Internet, an intranet, wide area network (WAN), a local area network (LAN), a satellite network, a wireless network, or some other type of network capable of supporting a virtual network through the devices thereon." Caronni II at 4:17–33. "FIG. 2 depicts a block diagram of the VARP lookup table 26 used by the virtual address resolution facility 24. The VARP lookup table 26 may be stored at any location accessible over the network. The VARP lookup table 26 includes registered virtual destination addresses located in a column 78 and corresponding associations located in a column 79. Entries 80, 82, 84, 86 and 88 include the registered virtual destination address and information associated with the registered virtual destination address. Each entry in the VARP lookup table 26 will list the registering virtual IP address 90 and associated information including a real IP address 91, a transport protocol designation 92, a port number identification 93 and an Application layer protocol designation 94. The associations are provided at the time the virtual destination address is registered in the virtual network. For example, the entry 80 for the virtual IP address 10.0.0.12 (90) includes the association of a real IP address 152.70.0.1 (91), an associated Transport layer protocol designation, UDP 92, an associated port number, 6789 (93), and an Application layer protocol designation of 'none' 94."

Claim 30	Mehta
	Caronni II at 5:10–29. "The illustrative embodiment of the present invention attempts to circumvent problems caused by packet filtering and network address translation occurring at the edge of networks through the use of higher layer protocols. FIG. 3 depicts a block diagram of the encapsulation process used to send a message from a virtual IP address to another virtual IP address located at the interior of a network at which packet filtering and/or network address translation is occurring. The virtual address resolution facility 24 is used to determine the associations registered with the destination address. The original message 111 at the Application layer 110 is formatted in a format specified in the registration of the virtual destination address if one is so indicated (e.g. HTTP). The message 111 is passed down to the Transport layer 112. The Transport layer 112 takes the original message 111 and adds a transport protocol header 113 as specified in the virtual address destination registration. The transport protocol header 113 may be a UDP header, TCP header, an X.25 header, XTP header, AppleTalk header, or other similar transport protocol header. Those skilled in the art will realize that the message may include an IPSEC header, inner IP header, UDP or TCP header and inner payload." Caronni II at 6:32-53. "The illustrative embodiment of the present invention may also be utilized to send a message from a virtual address located behind a NAT box to a destination address on the same virtual network that is behind a different NAT box. Ordinarily, where the destination address is directly connected to the Internet (i.e.: in situations where the destination is not behind a NAT box), a connection, such as a TCP connection, may be established between the originator of the connection behind the NAT box and the destination. The originator's message passes the NAT box and the internal mapping of public address and port number to internal address and port number takes place. Return packets from the destination may then be received following the mapping. When the destination address is also behind a NAT box, the originator of the message is unable to directly address the destination (since the address of the destination may not be routable in the public Internet). In such a case, the present

Claim 30	Mehta
	invention adds an entry to the VARP table for a third-party reflecting agent located at an address outside the NAT box. Messages that are being sent from the originating virtual address behind a NAT box to a destination which is behind a different NAT box are sent via the reflecting agent intermediary outside the NAT box and reflected to the destination. FIG. 5 depicts a block diagram of an environment holding a virtual network in which both the originator and destination of a message are located behind NAT boxes. An electronic device holding an originating process 182 which is part of a virtual network is interfaced with a network 180, such as the Internet, via a NAT box 186. An intended recipient for a message sent from the originating process 180 is an electronic device with a destination address 184 that is also behind a NAT box 188. A reflecting agent 200, the address of which has been added to a VARP table, acts as an intermediary to allow a connection between the originating process 182 and destination address 184." Caronni II at 8:1–38. Figure 5  Caronni II at FIG. 5.

Claim 30	Mehta
	"Beginning at step 270, a client application requests to the VNE framework 200 to connect or send to an address of another application, such as a server application having the address 10.10.2.70: port 9001. At step 272, the VNE framework 200 requests the VNE parameters for the process corresponding to the client application from a process state storage structure. The structure that stores process state is a structure that the operating system uses to store private information about the process. Therefore, it may differ depending on the operating system. The parameters added the process state storage structure as part of the virtual network environment are listed below: typedef struct { ipaddr_t app_address; ipaddr_t virtual_subnet; ipaddr_t virtual_mask; ipaddr_t global_subnet; ipaddr_t global_mask; } vne_param_t; Hipp at 4:44–64. "Virtualization of network identity is achieved by assigning a unique virtual IP address and virtual hostname to a group of processes that make up the application instance which the instance keeps throughout its execution. This virtual network identity stays with the application instance regardless of which node the application is running on. The framework, in essence, provides a mechanism to create this virtual network identity (VNI) around the application using the virtual network parameters assigned to it. In one embodiment, the virtual network parameters include an IP address and hostname. The framework 200 ensures that the

Claim 30	Mehta
	application's instance uses the virtual network parameters, transparently, so that it can be moved across machines, without modifications to the application. The virtual hostname resolves to the virtual IP address for both the applications registered with the VNI framework as well as those that are not registered. This may require configuration of a name service or OS host configuration files. For example, if an application instance used a virtual IP address of 10.10.0.1 and a virtual hostname of host1055, the standard hostname to IP address resolution mechanisms (e.g. DNS or the /etc/hosts file) would have to be preconfigured to resolve a query of host1055 to IP address 10.10.0.1." Hipp at 6:1–26.

Claim 35	Mehta
[35.0] The virtual network manager of claim 30 further comprising a join module configured to receive a join request from the agent associated with the device to indicate that the device is connected for data communication within the virtual network, the join module further configured to receive a leave request from the agent associated with the device to indicate that the device will be disconnected from data communication within the virtual network.	"Wireless systems on private networks use techniques similar to Network Address Translation (NAT) technology to send data from a wireless device to the public networking world. In a typical carrier infrastructure that uses a private network, a wireless device "registers" itself with the carrier infrastructure when it powers on (or in other circumstances, when the device attempts to initiate data services). The carrier dynamically assigns the wireless device a private non-routable address by means of DHCP (or DHCP-like) server. The information on the mapping of the transient private IP address to the device public IP address is stored in an internal carrier database and managed by carrier services such as a RADIUS server." Mehta at [0026]. "FIG. 4 is an example block diagram of a general purpose computer system for practicing embodiments of the Address Management Proxy System. The general purpose computer system 400 may comprise one or more server and/or client computing systems and may span distributed locations. In addition, each block may represent one or more such blocks as appropriate to a specific embodiment or may be combined with other blocks. The various blocks of the Address Management Proxy System 410 may physically reside on one or more machines, which use

Claim 35	Mehta
	standard interprocess communication mechanisms to communicate with each other. In the embodiment shown, computer system 400 comprises a computer memory ("memory") 01, a display 402, a Central Processing Unit ("CPU") 403, and Input/Output devices 404. The Address Management Proxy System 410 is shown residing in memory 401. The components of the Address Management Proxy System 410 preferably execute on CPU 403 and manage the address mapping of wireless devices on a wireless network, as described in previous figures, to allow other wired systems to communicate with the wireless devices. Other downloaded code 405 and potentially other data repositories also reside in the memory 410, and preferably execute on one or more CPU's 403. In a typical embodiment, the AMPS 410 includes one or more DNS'/API servers 411, one or more Address Proxy/Routers 412, an Address Management Data Server 413, and Address Management Data Repositories 414. As described earlier, the AMPS may include other data repositories and components, such as a load balancer, depending upon the particular implementation." Mehta at [0036]. "The unique ID table 610 maps unique string names of wireless devices to the private wireless network addresses that have been assigned typically by a carrier's infrastructure. In some embodiments, the carrier infrastructure dynamically allocates, using methods similar to a DHCP protocol, a private wireless network address when the wireless device registers itself with the carrier infrastructure upon being powered on. Thus, the unique ID table 610 may be sparsely filled or entries created dynamically and then deleted dynamically as devices register and unregister with the carrier infrastructure system." Mehta at [0053]. To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Caronni I. Caronni I concerns similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions,

Claim 35	Mehta
	joining devices with a virtual network was a well-known and commonly-used technique. Modifying this reference with Caronni I's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with that of Caronni I for example, to incorporate this limitation. *See, e.g.*, "FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:67–10:18. "FIG. 10 depicts a flow chart of the steps performed when logging a node out of a Supernet. The first step performed is for the user to run the SNlogout script and to enter a node ID (step 1002). Next, the SNlogout script requests a log out from SASD (step 1004). Upon receiving this request, SASD removes the mapping for this node from the VARPD that acts as the server for the Supernet (step 1006). SASD then informs KMS to cancel the registration of the node, and KMS terminates this KMC (step 1008). Lastly, KMS generates a new channel key for the

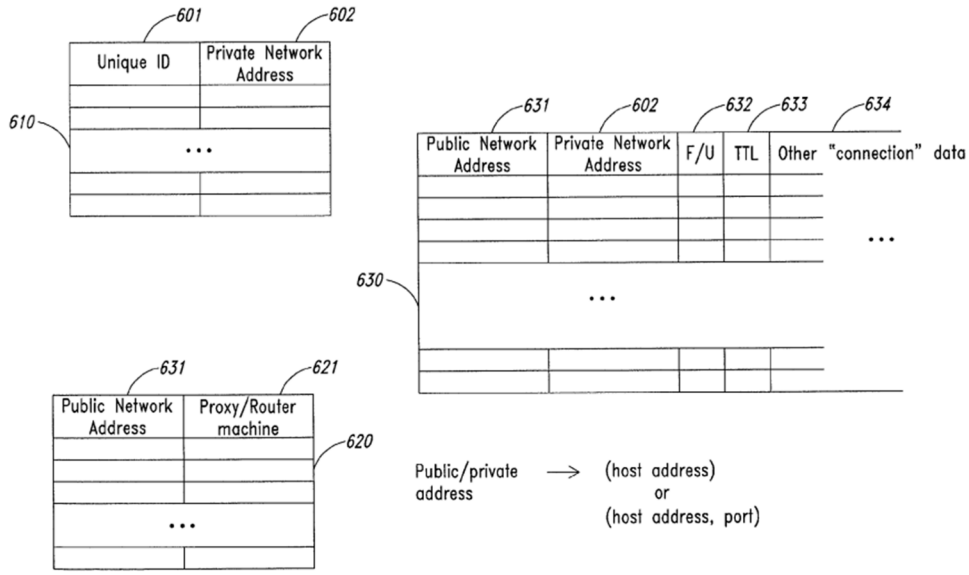
Claim 35	Mehta
	channels on which the node was communicating (step 1010) to reduce the likelihood of an intruder being able to intercept traffic." Caronni I at 12:61–13:5. *See also* Caronni I at FIG. 10.

Claim 37	Mehta
[37.0] The virtual network manager of claim 35 wherein the join module is further configured to maintain data to associate a virtual network address with a device in the virtual network.	"In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more Address Management Data Repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the external public network to send data to the wireless device. The temporary public address is, for example, an address associated with one of the Address Proxy/Routers, which are connected to the external public network and have access to the private addresses on the private wireless network. In some cases, the device on the public network that desires to send data to the wireless device uses a connection-based protocol, such as TCP/IP to send data. In other cases the device uses a connection-less protocol, such as UDP (UDP/IP) to send the data." Mehta at [0012]. "FIG. 3 is an example block diagram of components of an example Address Management Proxy System. In one embodiment, the Address Management Proxy System (AMPS) comprises one or more modified DNS'/ API servers 302, one or more Address Proxy/Routers 305, an Address Management Data Server 303 which manages a database or other repositories such as Address Management Data Repository 304, and optionally a load balancer 301. The DNS'/API servers 302 are either individually connected to a public network 310 or are connected to the load

Claim 37	Mehta
	balancer 301 which is in turn connected to public network 310. Similarly, each Address Proxy/Router 305 is also connected to the public network 310, via the routable (public) address to which data from the external network 310 is sent that is destined for the wireless devices. The DNS'/API servers 302 are either modified implementations of a DNS server to add functionality necessary to communicate with wireless devices, or are servers that implement one or more specialized APIs, as will be described further below. The DNS'/API servers 302 use the Address Management Data Server 303 to assist in mapping a unique identifier (e.g., string name) for a wireless device to a public address on public network 310. The pool of public addresses is also maintained by the Address Management Data Serve 303 and Data Repository 304. The Address Management Data server 303 and Address Management Data Repository 304 may be implemented using existing database technology, for example, ODBC technology or, may be implemented as a structure such as a simple text file. One skilled in the art will recognize that any embodiment for storing a set of tables, data, lists, or mappings can be used. Each Address Proxy/Router 305 also uses the Address Management Data Server 303 or equivalent to create and update a series of routing tables that are used to assign public addresses to wireless devices as needed and to update the various mappings between public addresses and the non-routable (private) addresses of the wireless devices. The tables and mappings that are maintained on behalf of the DNS'/API servers 302 and the Address Proxy/Routers 305 by the Address Management Data Server 303 are described below with reference to FIG. 6." Mehta at [0032]. "One skilled in the art will recognize that the AMPS 410 may be implemented in a distributed environment that is comprised of multiple, even heterogeneous, computer systems and networks. For example, in one embodiment, the DNS'/API servers 411, the Address Proxy/Router components 412, and the Address Management Data Servers 413 with their data repositories 414 are all located in physically different computer systems. In another embodiment, various components of the AMPS 410 are hosted each on a separate server machine and may be remotely located from the tables which are stored in the address management data

Claim 37	Mehta
	repository 414. In addition, under some scenarios, the entire AMPS system 410 may be hosted within a carrier's infrastructure and be completely subsumed by it. Different configurations and locations of programs and data are contemplated for use with techniques of the present invention. In example embodiments, these components may execute concurrently and asynchronously; thus the components may communicate using well-known message passing techniques. One skilled in the art will recognize that equivalent synchronous embodiments are also supported by an AMPS implementation. Also, other steps could be implemented for each routine, and in different orders, and in different routines, yet still achieve the functions of the AMPS." Mehta at [0038]. "FIG. 7 is an example flow diagram of an example routine provided by a DNS'/API server of the Address Management Proxy System to return a public address that corresponds to a designated unique identifier. In essence, this routine implements a DNS query or DNS-like query capability for the AMPS using a modified GetHostByName interface or a specialized API, for example GetProxyIP, as described with reference to FIG. 5. In summary, the routine dynamically allocates an appropriate Address Proxy/Router machine to associate with the wireless device and returns the public address of that machine (along with a TTL parameter and potentially other parameters). Specifically, in step 701, the routine determines the private (non-routable) address of the wireless device designated by a string parameter passed as input to the routine. For example, the string parameter may use fields such as "uniqueID.hostname.domain.tld," which specifies a typical hierarchy of person/service on a machine named "hostname" on a domain such as a company's network on a top level domain such as "org." "com," "edu," etc. One skilled in the art will recognize that many other string parameter designations could be used. One mechanism for implementing this routine is to request information from the Address Management Data Repository. In one embodiment, the data repository stores a table that maps unique IDs to private network addresses (see Table 610 in FIG. 6). Preferably, any mechanism that is used by the AMPS stores this data in a secure manner in order to keep the wireless network addresses private.

Claim 37	Mehta
	In step 702, the routine retrieves the public network address that corresponds to the private wireless network address of the designated device if one has already been assigned by the AMPS to that device and is still valid. In one embodiment, the data repository stores this mapping information between private wireless network address and public network address (see for example table 630 in FIG. 6). If a public network address has not already been assigned or is not valid, then the routine causes a new public network address to be allocated and that new public address is associated with the private wireless network address. Appropriate tables in the data repository are then updated. In step 703, the routine determines the Address Proxy/Router machine that is associated with the assigned public network address (for example using table 620 in FIG. 6). In step 704, the routine sends a request to the determined proxy/router machine to update its routing tables to map the determined public network address to the private wireless network address. In step 705, the routine updates information in the data repository to include any other connection related information (e.g., field 634 in Table 630 in FIG. 6) and indicates a Time to Live (TTL) parameter for the public-private address association (e.g., field 633 in Table 630 in FIG. 6). Once all of the tables have been updated in both the proxy/router and in the data repository, the DNS'/API server returns the determined public network address of the associated Address Proxy/Router machine. As described earlier, the public address may be a (hostname, port) pair when a port-based implementation is used." Mehta at [0057]. "Specifically, in step 801, the routine determines (for example, from the Address Management Data Repository) the private wireless address that corresponds to the invoked public address and the TTL parameter associated with this mapping. These values can be obtained, for example, from the public-to-private address table 630 of FIG. 6. In step 802, the routine determines whether the value of the determined TTL parameter indicates that the mapping has expired, and if so, returns an error, else continues in step 803. In step 803, the routine determines the format required by the target device (the wireless network format). In step 804, the routine determines whether any protocol conversion is necessary and, if so, continues in

Claim 37	Mehta
	step 805 else continues in step 806. Note that protocol conversion for a specific wireless network such as converting the data to an "HTTP" packet may be done by the proxy/router or may be done by some other component within the carrier infrastructure. One skilled in the art will recognize that these are example steps and that different formatting or different protocol conversion routines may be added or omitted as specific to the environment. In step 806, the Address Proxy/Router routine sends the data packet (which has been formatted and whose protocol is converted as necessary) to the determined associated private address of the wireless device, and returns." Mehtat at [0059].  *Fig. 6* Mehta at FIG. 6.

Claim 37	Mehta
	To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Caronni I. Caronni I concerns similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, joining devices with a virtual network was a well-known and commonly-used technique. Modifying this reference with Caronni I's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with that of Caronni I for example, to incorporate this limitation. *See, e.g.*, "SASD 540 represents a Supernet: there is one instance of an SASD per Supernet, and it both authenticates nodes and authorizes nodes to join the Supernet. VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as a node IDs, to the network addresses recognized by the public-network infrastructure, known as the real addresses. The "node ID" may include the following: a Supernet ID (e.g., 0x123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1). Although the virtual address is described in an IP address scheme, one skilled in the art will appreciate that the virtual address may be any other type addressing scheme, such as an e-mail address, IPX, or IPv6. Since the node ID includes a Supernet ID, a node will have more than one node ID when it communicates over more than one channel. The "real address" is an IP address (e.g., 10.0.0.2) that is globally unique and meaningful to the public-network infrastructure. In a Supernet, one VARPD runs on each machine, and it may play two roles. First, a VARPD may act as a server by storing all address mappings for a particular Supernet into its associated VARPDB. Second, regardless of its role as a server or not, each VARPD assists in address translation for the nodes on its machine. In this role, the VARPD stores into its associated VARPDB the address mappings for its nodes, and if it needs a mapping that it does not have, it will contact the VARPD that acts as the server for the given Supernet to obtain it. The

Claim 37	Mehta
	VARPDB may also decide which virtual address to use in the translation. That is, the VARPDB may associate a virtual address with multiple real addresses or vice versa." Caronni I at 7:3–34. "FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:67–10:18. "FIG. 10 depicts a flow chart of the steps performed when logging a node out of a Supernet. The first step performed is for the user to run the SNlogout script and to enter a node ID (step 1002). Next, the SNlogout script requests a log out from SASD (step 1004). Upon receiving this request, SASD removes the mapping for this node from the VARPD that acts as the server for the Supernet (step 1006). SASD then informs KMS to cancel the registration of the node, and KMS terminates this KMC (step 1008). Lastly, KMS generates a new channel key for the channels on which the node was communicating (step 1010) to reduce the likelihood of an intruder being able to intercept traffic." Caronni I at 12:61–13:5.

Claim 37	Mehta
	See also Caronni I at FIG. 10.

Exhibit 785-2

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent No. 6,970,941 ("Caronni I")

Caronni I titled "System and method for separating addresses from the delivery scheme in a virtual private network" to Germano Caronni et al. was filed on December 10, 1999 and issued on November 29, 2005.[1] Caronni I is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Caronni I does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Caronni I or combine the teachings of Caronni I with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, including but not limited to:

- U.S. Patent No. 7,814,228 ("Caronni II");
- U.S. Patent No. 6,766,371 ("Hipp");
- U.S. Patent Application Publication No. 2003/0028671 ("Mehta"); and/or
- Huitema, C., Network Working Group Request for Comment (RFC): 1383, entitled An Experiment in DNS Based IP Routing ("RFC 1383").

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to

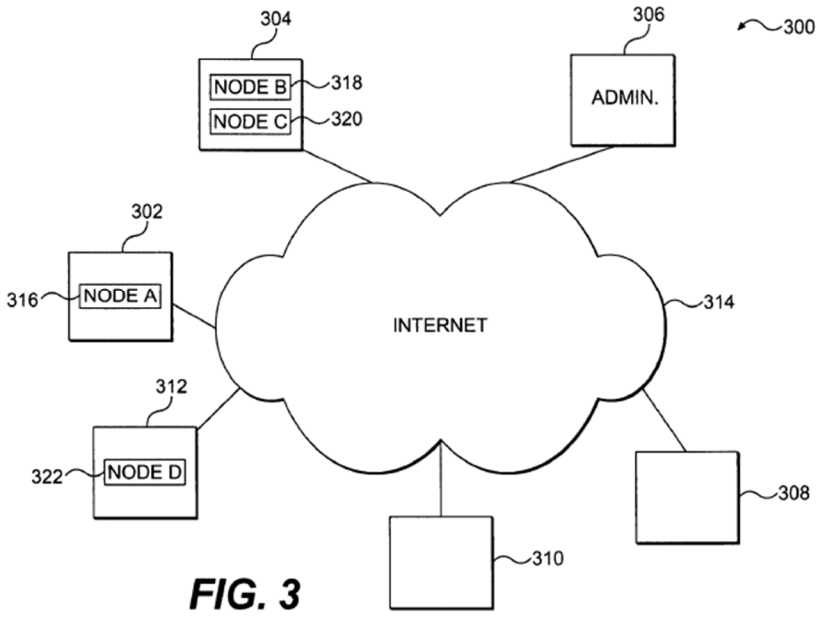
[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

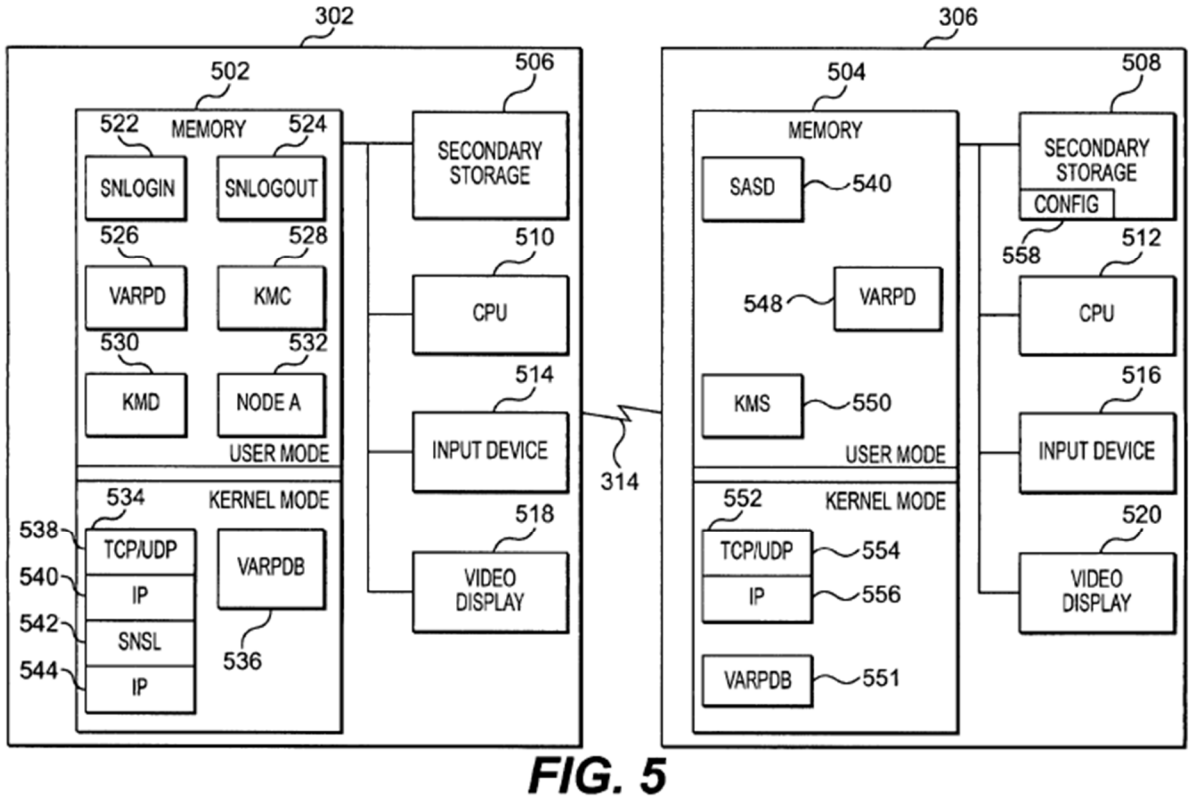
pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

Claim 30	Caronni I
[30.0] A virtual network manager, comprising:	To the extent the preamble is limiting, Caronni I discloses: "Methods and systems consistent with the present invention establish a virtual network on top of current IP network naming schemes. The virtual network uses a separate layer to create a modification to the IP packet format that is used to separate network behavior from addressing. As a result of the modification to the packet format, any type of delivery method may be assigned to any address or group of addresses. The virtual network also maintains secure communications between nodes, while providing the flexibility of assigning delivery methods independent of the delivery addresses." Caronni I at Abstract. "The present invention relates generally to data processing systems and, more particularly, to a private network using a public-network infrastructure." Caronni I at 1:57–59. "Methods and systems consistent with the present invention overcome the shortcomings of existing networks by establishing a "Supernet," which is a private network that uses components from a public-network infrastructure. A Supernet allows an organization to utilize a public-network infrastructure for its enterprise network so that the organization no longer has to maintain a private network infrastructure; instead, the organization may have the infrastructure maintained for them by one or more service providers or other organizations that specialize in such connectivity matters. As such, the burden of maintaining an enterprise network is greatly reduced. Moreover, a Supernet is not geographically restrictive, so a user may plug their device into the Internet from virtually any portal in the world and still be able to use the resources of their private network in a secure and robust manner." Caronni I at 4:36-52. "Supernets also provide heterogeneous addressing functionality. The Supernet uses a separate layer that isolates address names of nodes from addressing schemes and delivery schemes. The Supernet contains a modification to the IP packet format that can be used to

Claim 30	Caronni I
	separate network behavior from addressing. As a result of the modification, any delivery scheme may be assigned to any address, or group of addresses." Caronni I at 4:53-59. "1. A method in a public network having a network infrastructure that is used by a private network over which a plurality of nodes communicate, the private network uses a plurality of delivery schemes to communicate between internal addresses that are suitable for use in communicating among the plurality of nodes in the private network, the method comprising: receiving a request from a user to add a new node to the private network, the request comprising an identifier of the private network and an identifier of the user, the new node for running on a device connected to the network infrastructure, the device having a system-level component used for communicating over the private network; attempting to authenticate the request including the private network identifier and the user identifier; when the request has been authenticated successfully, generating an address mapping for the new node, the address mapping used for mapping between the internal address and an external address suitable for communicating over the network infrastructure of the public network, wherein the internal address is not associated with the delivery scheme; configuring, by the system-level component, the new node in a context such that the new node is capable of communicating with the plurality of nodes on the private network; and sending, by the system-level component, a packet from the new node to a destination one of the plurality of nodes on the private network by accessing the address mapping and adding the external address to the packet and by causing delivery of the packet to the destination node to occur in a secure manner." Caronni I at Claim 1.

Claim 30	Caronni I
	 FIG. 3 Caronni I at FIG. 3.

Claim 30	Caronni I
	 FIG. 5 Caronni I at FIG. 5. To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Mehta. Mehta concerns similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, use of virtual networks was well-known and common. Modifying this reference with Mehta's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results.

Claim 30	Caronni I
	A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with that of Mehta for example, to incorporate this limitation. *See, e.g.*, "Methods and systems for providing two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP, are provided. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporarily use by a requesting device on a public network to communicate with a wireless device on a wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more data repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the public network to send data to the wireless device." Mehta at Abstract. "The present invention relates to methods and systems for initiating communication with wireless devices and, in particular, to methods and systems for initiating communication with a device on a private network from a device on a public network to achieve virtual end-to-end connectivity." Mehta at [0002]. "Embodiments of the present invention provide computer- and network-based methods and systems for two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP. Example embodiments provide an Address Management Proxy System ("AMPS"), which

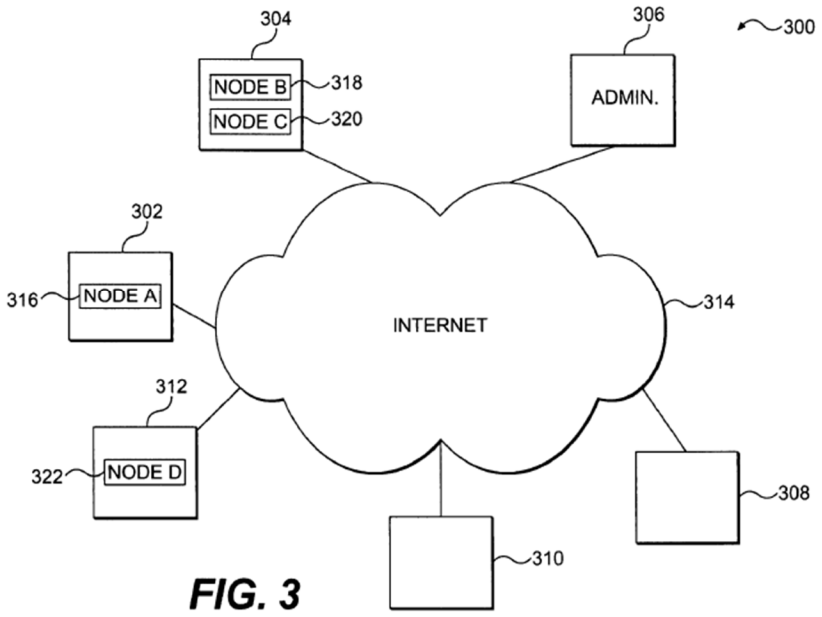
Claim 30	Caronni I
	enables devices and systems connected to a public internetwork, such as the Internet, to initiate communication with and send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporary use by a requesting device on an external public network to communicate with a wireless device on a private wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. The mapping of a temporary public address to the private address of a wireless device is maintained and updated transparently by the AMPS using routing tables and other mapping data structures." Mehta at [0011]. "To insure a greater degree of security, according to one embodiment, the AMPS maintains a Time to Live (TTL) parameter with each address mapping. In this way, once the TTL value indicates that the mapping has expired, the AMPS can destroy the mapping, and also any connection. Further, in some embodiments, the AMPS also puts a timestamp in its own mapping tables. After some timeout period based upon the timestamp, the AMPS can destroy the mapping, thereby forcing a new mapping to be initiated on a periodic basis." Mehta at [0015]. "Embodiments of the present invention provide computer- and network-based methods and systems for two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices." Mehta at [0024]. "The second approach used to implement the AMPS supports full bi-directional communication through point-to-point connections established, for example, using TCP/IP protocol. (Note that these same techniques also support connection-less UDP bi-directional

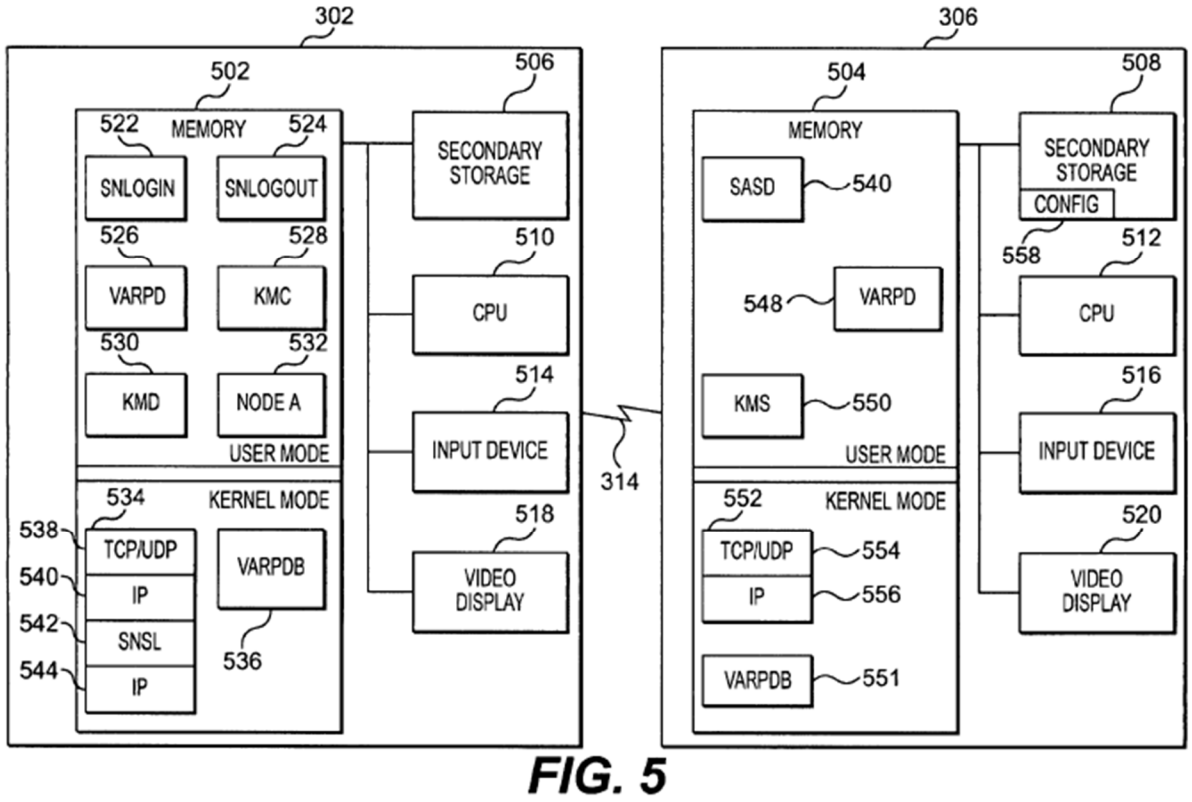
Claim 30	Caronni I
	communication). The second approach can be implemented by providing a modified implementation of a standard UDP or TCP/IP function, "GetHostByName." The GetHostByName API allows a string designation to identify the designated device and returns an IP address data structure. Alternatively, to increase the level of security provided, the AMPS can implement a specialized API to return the dynamically allocated public address that (now) corresponds to the requested wireless device. A disadvantage of the specialized API approach is that the device on the public network (or other device that wishes to obtain a connection to the wireless device) needs to include specialized code in the application on the requesting device." Mehta at [0041]. *See also* Mehta at FIG. 3.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	"Methods and systems consistent with the present invention establish a virtual network on top of current IP network naming schemes. The virtual network uses a separate layer to create a modification to the IP packet format that is used to separate network behavior from addressing. As a result of the modification to the packet format, any type of delivery method may be assigned to any address or group of addresses. The virtual network also maintains secure communications between nodes, while providing the flexibility of assigning delivery methods independent of the delivery addresses." Caronni I at Abstract. "The present invention relates generally to data processing systems and, more particularly, to a private network using a public-network infrastructure." Caronni I at 1:57–59. "Methods and systems consistent with the present invention overcome the shortcomings of existing networks by establishing a "Supernet," which is a private network that uses components from a public-network infrastructure. A Supernet allows an organization to utilize a public-network infrastructure for its enterprise network so that the organization no longer has to maintain a private network infrastructure; instead, the organization may have the infrastructure maintained for them by one or more service providers or other

Claim 30	Caronni I
	organizations that specialize in such connectivity matters. As such, the burden of maintaining an enterprise network is greatly reduced. Moreover, a Supernet is not geographically restrictive, so a user may plug their device into the Internet from virtually any portal in the world and still be able to use the resources of their private network in a secure and robust manner." Caronni I at 4:36-52. "Supernets also provide heterogeneous addressing functionality. The Supernet uses a separate layer that isolates address names of nodes from addressing schemes and delivery schemes. The Supernet contains a modification to the IP packet format that can be used to separate network behavior from addressing. As a result of the modification, any delivery scheme may be assigned to any address, or group of addresses." Caronni I at 4:53-59. "SNlogin 522 is a script used for logging into a Supernet. Successfully executing this script results in a Unix shell from which programs (e.g., node A 522) can be started to run within the Supernet context, such that address translation and security encapsulation is performed transparently for them and all they can typically access is other nodes on the Supernet. Alternatively, a parameter may be passed into SNlogin 522 that indicates a particular process to be automatically run in a Supernet context. Once a program is running in a Supernet context, all programs spawned by that program also run in the Supernet context, unless explicitly stated otherwise. SNlogout 524 is a script used for logging out of a Supernet. Although both SNlogin 522 and SNlogout 524 are described as being scripts, one skilled in the art will appreciate that their processing may be performed by another form of software. VARPD 526 performs address translation between node IDs and real addresses. KMC 528 is the key management component for each node that receives updates whenever the key for a channel ("the channel key") changes. There is one KMC per node per channel. KMD 530 receives requests from SNSL 542 of the TCP/IP protocol stack 534 when a packet is received and accesses the appropriate KMC for the destination node to retrieve the appropriate key to decrypt the packet. Node A 532 is a Supernet node running in a Supernet context." Caronni I at 8:31-55.

Claim 30	Caronni I
	"TCP/IP protocol stack 534 contains a standard TCP/UDP layer 538, two standard IP layers (an inner IP layer 540 and an outer IP layer 544), and a Supernet security layer (SNSL) 542, acting as the conduit for all Supernet communications. To conserve memory, both inner IP layer 540 and outer IP layer 544 may share the same instance of the code of an IP layer. SNSL 542 performs security functionality as well as address translation. It also caches the most recently used channel keys for ten seconds. Thus, when a channel key is needed, SNSL 542 checks its cache first, and if it is not found, it requests KMD 530 to contact the appropriate KMC to retrieve the appropriate channel key." Caronni I at 8:54-67. "FIG. 5 depicts administrative machine 306 and device 302 in greater detail, although the other devices 304 and 308–312 may contain similar components. Device 302 and administrative machine 306 communicate via Internet 314. Each device contains similar components, including a memory 502, 504; secondary storage 506, 508; a central processing unit (CPU) 510, 512; an input device 514, 516; and a video display 518, 520. One skilled in the art will appreciate that these devices may contain additional or different components. Memory 504 of administrative machine 306 includes the SASD process 540, VARPD 548, and KMS 550 all running in user mode. That is, CPU 512 is capable of running in at least two modes: user mode and kernel mode. When CPU 512 executes programs running in user mode, it prevents them from directly manipulating the hardware components, such as video display 518. On the other hand, when CPU 512 executes programs running in kernel mode, it allows them to manipulate the hardware components. Memory 504 also contains a VARPDB 551 and a TCP/IP protocol stack 552 that are executed by CPU 512 running in kernel mode. TCP/IP protocol stack 552 contains a TCP/UDP layer 554 and an IP layer 556, both of which are standard layers well known to those of ordinary skill in the art. Secondary storage 508 contains a configuration file 558 that stores various configuration-related information (described below) for use by SASD 540." Caronni I at 6:44-7:2. "1. A method in a public network having a network infrastructure that is used by a private network over which a plurality of nodes communicate, the private network uses a plurality of

Claim 30	Caronni I
	delivery schemes to communicate between internal addresses that are suitable for use in communicating among the plurality of nodes in the private network, the method comprising: receiving a request from a user to add a new node to the private network, the request comprising an identifier of the private network and an identifier of the user, the new node for running on a device connected to the network infrastructure, the device having a system-level component used for communicating over the private network; attempting to authenticate the request including the private network identifier and the user identifier; when the request has been authenticated successfully, generating an address mapping for the new node, the address mapping used for mapping between the internal address and an external address suitable for communicating over the network infrastructure of the public network, wherein the internal address is not associated with the delivery scheme; configuring, by the system-level component, the new node in a context such that the new node is capable of communicating with the plurality of nodes on the private network; and sending, by the system-level component, a packet from the new node to a destination one of the plurality of nodes on the private network by accessing the address mapping and adding the external address to the packet and by causing delivery of the packet to the destination node to occur in a secure manner." Caronni I at Claim 1.

Claim 30	Caronni I
	 FIG. 3 Caronni I at FIG. 3.

Claim 30	Caronni I
	 FIG. 5 Caronni I at FIG. 5. To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Mehta. Mehta concerns similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, use of virtual networks was well-known and common. Modifying this reference with Mehta's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results.

Claim 30	Caronni I
	A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with that of Mehta for example, to incorporate this limitation. *See, e.g.*, "The present invention relates to methods and systems for initiating communication with wireless devices and, in particular, to methods and systems for initiating communication with a device on a private network from a device on a public network to achieve virtual end-to-end connectivity." Mehta at [0002]. "Embodiments of the present invention provide computer- and network-based methods and systems for two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internetwork, such as the Internet, to initiate communication with and send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporary use by a requesting device on an external public network to communicate with a wireless device on a private wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. The mapping of a temporary public address to the private address of a wireless device is maintained and updated transparently by the AMPS using routing tables and other mapping data structures." Mehta at [0011]. "Although the techniques of the AMPS are generally applicable to any a wired device communicating with a wireless device, the phrase "public network" (or "wired network") is used generally to imply any type of internetworked environment including a public network or a backbone that is somewhere down the line connected to one or more private or public networks. In addition, although the examples described herein often refer to the Internet, one skilled in the art will recognize that the concepts and inventions described are applicable to other forms and embodiments of internetworking, including, for example ATM type

Claim 30	Caronni I
	networks. Thus, techniques of the present invention can also be used by one device on a first wireless network to communicate with another wireless device on a second network—each device ends up communicating with the Address Proxy/Router of the other network. This scenario is feasible because each wireless network (or its carrier infrastructure) is connected to a proxy/router that is also connected (via a public address) to a public network. In addition, although a public network is sometimes also referred to herein as a wired network, one skilled in the art will recognize that any network that exposes routable (public) addresses may be implied. Thus, a wireless network with unique public (and routable) address can also employ techniques of the present invention to perform bi-directional communication. Also, one skilled in the art will recognize that terms such as wireless device, phone, handheld, etc., are used interchangeably to indicate any type of wireless device that is capable of operating with the AMPS. In addition, terms may have alternate spellings which may or may not be explicitly mentioned, and one skilled in the art will recognize that all such variations of terms are intended to be included." Mehta at [0033]. "Public address to proxy/router machine table 620 comprises a public network address field 631 and an indication of a functioning proxy/router machine 621. By maintaining such a mapping, the AMPS is able to substitute proxy/router machines for other proxy/router machines to provide a higher degree of robustness. Each proxy/router machine has a preconfigured set of public network addresses, such as are typically configured by network cards inserted into the proxy/router machine. These address are allocated in a standard fashion through prior purchase or obtaining from an address authorizing authority, currently the Internet Corporation for Assigned Names and Numbers (ICANN). When a machine is inserted for use into the AMPS, indications of these addresses are stored in field 631." Mehta at [0055]. *See also* Mehta at FIGS. 1-2.
[30.2] a memory and a processor to implement a register module configured to	"To configure a Supernet, a system administrator creates a configuration file 558 that is used by SASD 540 when starting or reconfiguring a Supernet. This file may specify: (1) the Supernet name, (2) all of the channels in the Supernet, (3) the nodes that communicate over

Claim 30	Caronni I
register devices in a virtual network, the register module further configured to:	each channel, (4) the address of the KMS for each channel, (5) the address of the VARPD that acts as the server for the Supernet, (6) the user IDs of the users who are authorized to create Supernet nodes, (7) the authentication mechanism to use for each user of each channel, and (8) the encryption algorithm to use for each channel. Although the configuration information is described as being stored in a configuration file, one skilled in the art will appreciate that this information may be retrieved from other sources, such as databases or interactive configurations." Caronni I at 8:1-15. "SASD 540 represents a Supernet: there is one instance of an SASD per Supernet, and it both authenticates nodes and authorizes nodes to join the Supernet. VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as a node IDs, to the network addresses recognized by the public-network infrastructure, known as the real addresses. The "node ID" may include the following: a Supernet ID (e.g., 0x123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1). Although the virtual address is described in an IP address scheme, one skilled in the art will appreciate that the virtual address may be any other type addressing scheme, such as an e-mail address, IPX, or IPv6. Since the node ID includes a Supernet ID, a node will have more than one node ID when it communicates over more than one channel. The "real address" is an IP address (e.g., 10.0.0.2) that is globally unique and meaningful to the public-network infrastructure. In a Supernet, one VARPD runs on each machine, and it may play two roles. First, a VARPD may act as a server by storing all address mappings for a particular Supernet into its associated VARPDB. Second, regardless of its role as a server or not, each VARPD assists in address translation for the nodes on its machine. In this role, the VARPD stores into its associated VARPDB the address mappings for its nodes, and if it needs a mapping that it does not have, it will contact the VARPD that acts as the server for the given Supernet to obtain it. The VARPDB may also decide which virtual address to use in the translation. That is, the VARPDB may associate a virtual address with multiple real addresses or vice versa." Caronni I at 7:3–33.

Claim 30	Caronni I
	"FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:67–10:18. "After configuring SNSL, SNlogin invokes an operating system call, SETVIN, to cause the SNlogin script to run in a Supernet context (step 720). In Unix, each process has a data structure known as the "proc structure" that contains the process ID as well as a pointer to a virtual memory description of this process. In accordance with methods and systems consistent with the present invention, the Supernet IDs indicating the channels over which the process communicates as well as its virtual address for this process are added to this structure. By associating this information with the process, the SNSL layer can enforce that this process runs in a Supernet context. Although methods and systems consistent with the present invention are described as operating in a Unix environment, one skilled in the art will appreciate that such methods and systems can operate in other environments. After the SNlogin script runs in the Supernet context, the SNlogin script spawns a Unix program, such as a Unix shell or a service daemon (step 722). In this step, the SNlogin script spawns a Unix shell from which programs can be run by the user. All of these programs will thus run in the Supernet context until the user runs the SNlogout script." Caronni I at 10:63–11:17.

Claim 30	Caronni I
	"The packet and Supernet ID are then transmitted to the SNSL layer using the modified socket structure (step 806). The SNSL layer then accesses the VARPDB to obtain the address mapping between virtual source node address 642 and the source real address 614 as well as the virtual destination node address 644 and the destination real address 616 (step 808). If they are not contained in the VARPDB because this is the first time a packet has been sent from this node or sent to this destination, the VARPDB accesses the local VARPD to obtain the mapping. When contacted, the VARPD on the local machine contacts the VARPD that acts as the server for the Supernet to obtain the appropriate address mapping. Since the VARPDB maintains all real IP addresses, a remote node may securely communicate with another remote node without reverfication." Caronni I at 11:47–61.

Claim 30	Caronni I

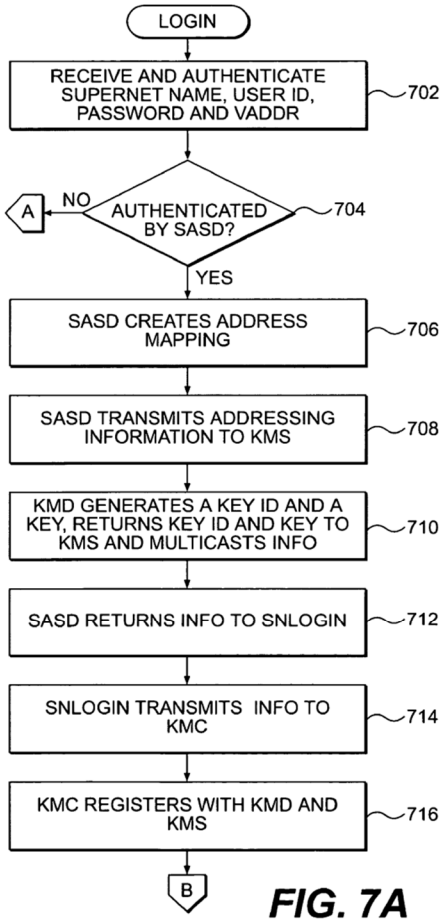


FIG. 7A

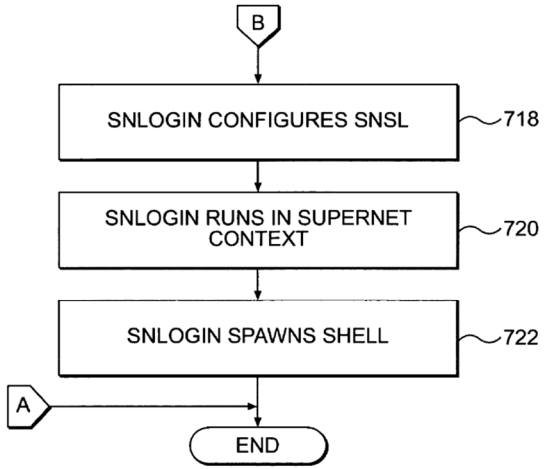


FIG. 7B

Caronni I at FIGS. 7A and 7B.

To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as

Claim 30	Caronni I
	Caronni II, Hipp, and/or Mehta. Caronni II, Hipp, and Mehta concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, registration of devices in a virtual network was a well-known and commonly-used technique. Modifying this reference with Caronni II's, Hipp's, and/or Mehta's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with those of Caronni II, Hipp, or Mehta for example, to incorporate this limitation. *See, e.g.*, "In an embodiment of the present invention a virtual network is supported by a physical network and includes a virtual address resolution facility. The virtual address resolution facility is used to resolve a virtual IP address into a real IP address. A registration request is received at the virtual address resolution facility which includes a real IP address, a port address, a transport protocol designation and an Application layer protocol designation associated with a virtual address destination. The associations are stored using the virtual address resolution facility. A resolution request is received referencing a virtual address destination and the resolution request is resolved using the virtual address resolution facility and the stored associations." Caronni II at 3:11–23. "FIG. 2 depicts a block diagram of the VARP lookup table 26 used by the virtual address resolution facility 24. The VARP lookup table 26 may be stored at any location accessible over the network. The VARP lookup table 26 includes registered virtual destination addresses located in a column 78 and corresponding associations located in a column 79. Entries 80, 82, 84, 86 and 88 include the registered virtual destination address and information associated with the registered virtual destination address. Each entry in the VARP lookup table 26 will list the registering virtual IP address 90 and associated information including a real IP address 91, a transport protocol designation 92, a port number identification 93 and an Application layer protocol designation 94. The associations are provided at the time the virtual destination address is registered in the virtual network. For example, the entry 80 for the virtual IP address 10.0.0.12 (90) includes the association of a real IP address 152.70.0.1 (91), an associated Transport layer protocol designation, UDP 92,

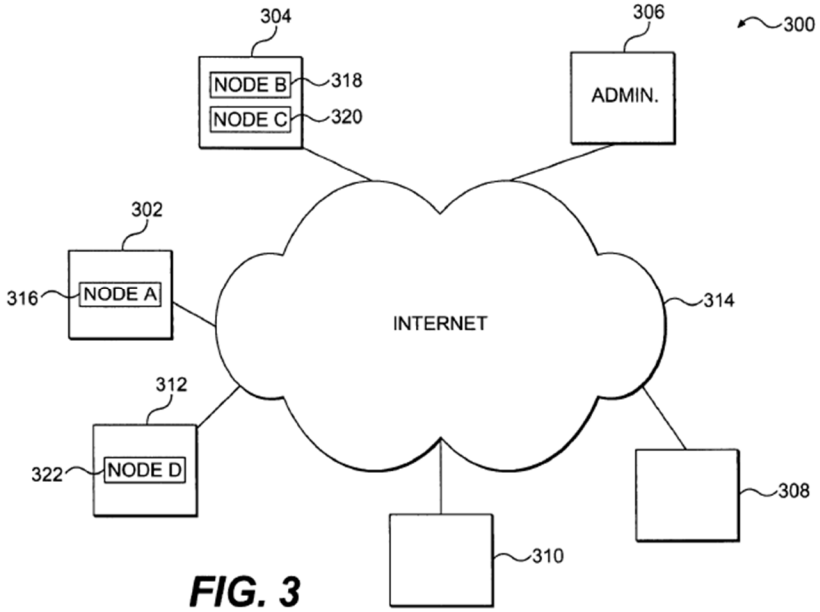
Claim 30	Caronni I
	an associated port number, 6789 (93), and an Application layer protocol designation of 'none' 94." Caronni II at 5:10–29. "FIG. 4 depicts the sequence of steps followed by the illustrative embodiment of the present invention to process the received encapsulated message. The sequence begins when a process associated with a virtual IP address registers with the virtual address resolution facility 24 (step 140). The registration is stored in a VARP lookup table accessible over the network. Subsequently, a message is received at the electronic device which bears a MAC/network interface address of the network interface of the electronic device with the designated real IP address listed in the virtual address registration (Step 142). The MAC address header 117 is stripped off at the Link layer and the message and appended headers are passed up to the Network layer 114(Step 144). The Network layer 114 strips off the IP header 115 and identifies the Transport Protocol header 113 underneath. The Network layer passes the message and remaining appended header to the Transport layer 112 (Step 146). The Transport layer 112 strips off the Transport header and then passes the message to the Application layer via the port address of the interface identified in the VARP registration. (Step 148). The receiving process determines whether it is the virtual destination address the message is addressed to (Step 149) and acts accordingly. If the receiving process that is examining the message is the destination address, the process examines the data (Step 150). Alternatively, if the message is not intended for the process executing on the edge device, the process re-routes the message and sends it back down the protocol model stack, on to the local network operating behind the edge device, and on to its intended destination by performing traditional VARP and ARP lookups. Alternatively forwarding the message may be accomplished by reinserting the packet into the IP layer of the networking stack." Caronni II at 7:33–67. *See also* Caronni II at Claim 1, FIGS. 2-4. "FIG. 3 is a data flow diagram illustrating the registration of virtual network environment parameters. The VNE framework 200 is a software module that processes transactions between the applications and the operating system. The VNE parameters are registered with

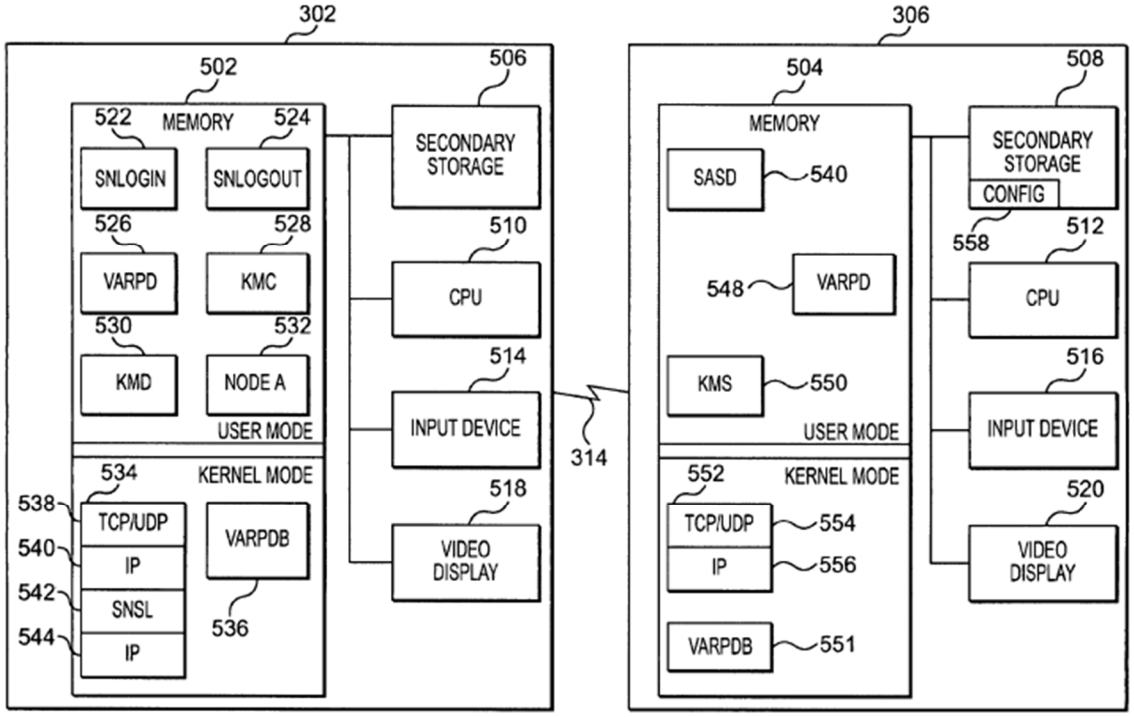
Claim 30	Caronni I
	the VNE framework 200 at the time the application is started. The VNE parameters include the application IP address, the virtual network subnet, and the global virtual address subnet. At step 250, the registration harness 220 supplies the application IP address, virtual subnet, and the global virtual address subnet for a $process_x$ to the VNE framework 200. The VNE framework 200 then records the IP address, virtual subnet, and the global virtual address subnet for the $process_x$ at step 252. The $process_x$ can then spawn additional processes (or create additional objects) at step 254. the new $process_y$ inherits the IP address, virtual subnet, and global virtual address subnet from $process_x$. At step 256, the registration harness 220 launches the application related to $process_y$." Hipp at 3:52-4:2. "Virtualization of network identity is achieved by assigning a unique virtual IP address and virtual hostname to a group of processes that make up the application instance which the instance keeps throughout its execution. This virtual network identity stays with the application instance regardless of which node the application is running on. The framework, in essence, provides a mechanism to create this virtual network identity (VNI) around the application using the virtual network parameters assigned to it. In one embodiment, the virtual network parameters include an IP address and hostname. The framework 200 ensures that the application's instance uses the virtual network parameters, transparently, so that it can be moved across machines, without modifications to the application." Hipp at 6:1-15. *See also* Hipp at 6:16-60, FIG. 3. "Methods and systems for providing two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP, are provided. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporarily use by a requesting device on a public network to communicate with a wireless

Claim 30	Caronni I
	device on a wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more data repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the public network to send data to the wireless device." Mehta at Abstract. "To insure a greater degree of security, according to one embodiment, the AMPS maintains a Time to Live (TTL) parameter with each address mapping. In this way, once the TTL value indicates that the mapping has expired, the AMPS can destroy the mapping, and also any connection. Further, in some embodiments, the AMPS also puts a timestamp in its own mapping tables. After some timeout period based upon the timestamp, the AMPS can destroy the mapping, thereby forcing a new mapping to be initiated on a periodic basis." Mehta at [0015]. "In existing systems, data communication (communication on a data channel) between a wireless device that is connected to a private wireless network and a wired device connected to a public wired network (e.g., the Internet) can only be initiated by the wireless device. Some carriers have assigned fixed public IP addresses to wireless devices; however, the wireless devices need to then have client programs (e.g., a UDP stack) capability of receiving and handling the incoming communication packets. Moreover, these wireless devices are then part of a public wireless network and not a private wireless network. Since public IP addresses are becoming a scarcer commodity and currently expensive to service a network of multi-million devices, carriers cannot in a practical sense count on having a fixed public IP address for each device on its network. (Although movement to IPv6 will allow more addresses, the current IPv4 definitions are limited and the potential number of wireless users subscribing to larger carriers is very high. In some regions of the world, the current public address scheme is even more limited.) Further, such addressing capability would expose each

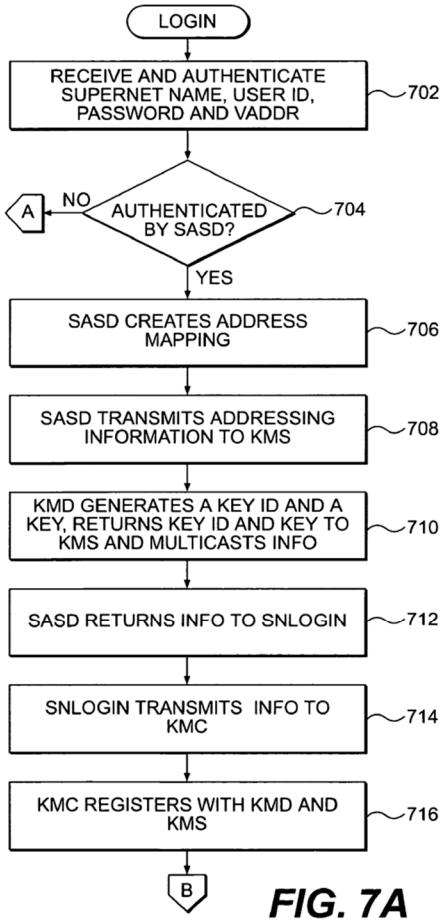
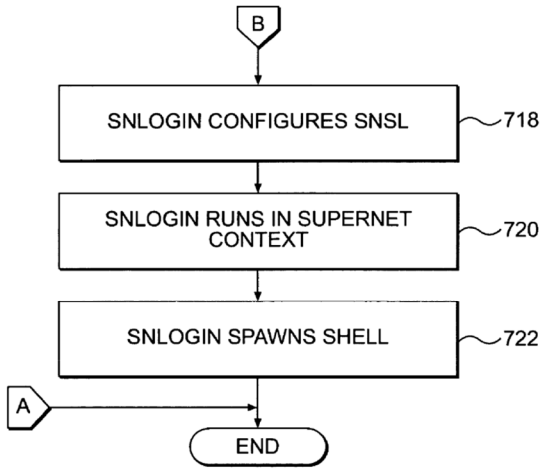
Claim 30	Caronni I
	device to further security risks, because each such device is part of a publicly accessible network and it becomes more difficult to implement and enforce security measures. Thus, assigning private IP addresses to devices is preferred in existing wireless networks over assigning fixed public IP addresses to devices. When wireless networks are private, the locations (addresses) of the wireless devices are intentionally hidden from public view by a carrier system's (or, as referred to in some countries, operator's) infrastructure." Mehta at [0025]. "Example embodiments described herein provide applications, tools, data structures and other support to implement private to public address mappings over one or more wired and wireless networks to be used for bi-directional communication. One skilled in the art will recognize that other embodiments of the methods and systems of the present invention may be used for many other purposes, including to push information and/or data or code from a public network such as the Internet to a wireless device. In addition, although this description primarily refers to "data" as being sent via the networks, one skilled in the art will recognize that all types of data can be communicated using the techniques described herein including, but not limited to, text, graphics, audio, and video." Mehta at [0034]. "FIG. 4 is an example block diagram of a general purpose computer system for practicing embodiments of the Address Management Proxy System. The general purpose computer system 400 may comprise one or more server and/or client computing systems and may span distributed locations. In addition, each block may represent one or more such blocks as appropriate to a specific embodiment or may be combined with other blocks. The various blocks of the Address Management Proxy System 410 may physically reside on one or more machines, which use standard interprocess communication mechanisms to communicate with each other. In the embodiment shown, computer system 400 comprises a computer memory ("memory") 01, a display 402, a Central Processing Unit ("CPU") 403, and Input/Output devices 404. The Address Management Proxy System 410 is shown residing in memory 401. The components of the Address Management Proxy System 410 preferably execute on CPU 403 and manage the address mapping of wireless devices on a wireless network, as described in previous figures, to allow other wired systems to communicate with the wireless devices.

Claim 30	Caronni I
	Other downloaded code 405 and potentially other data repositories also reside in the memory 410, and preferably execute on one or more CPU's 403. In a typical embodiment, the AMPS 410 includes one or more DNS'/API servers 411, one or more Address Proxy/Routers 412, an Address Management Data Server 413, and Address Management Data Repositories 414. As described earlier, the AMPS may include other data repositories and components, such as a load balancer, depending upon the particular implementation." Mehta at [0036]. "In one embodiment, a timestamp of the mapping between a public network address and a private address is also noted in table 630. After a defined time out, based upon this timestamp, the Address Management Data Server sends a request to the Address Proxy/Router associated with the mapping to unmap the public-to-private address mapping and updates the mapping table 630, thereby returning the associated public address to the pool of unused public network addresses." Mehta at [0056]. *See also* Mehta at FIG. 4.
[30.3] receive a registration request from an agent associated with a device;	"FIG. 3 depicts a data processing system 300 suitable for use with methods and systems consistent with the present invention. Data processing system 300 comprises a number of devices, such as computers 302–312, connected to a public network, such as the Internet 314. A Supernet's infrastructure uses components from the Internet because devices 302, 304, and 312 contain nodes that together form a Supernet and that communicate by using the infrastructure of the Internet. These nodes 316, 318, 320, and 322 are communicative entities (e.g., processes) running within a particular device and are able to communicate among themselves as well as access the resources of the Supernet in a secure manner. When communicating among themselves, the nodes 316, 318, 320, and 322 serve as end points for the communications, and no other processes or devices that are not part of the Supernet are able to communicate with the Supernet's nodes or utilize the Supernet's resources. The Supernet also includes an administrative node 306 to administer to the needs of the Supernet." Caronni I at 4:66–5:13.

Claim 30	Caronni I
	 FIG. 3 Caronni I at FIG. 3. "FIG. 5 depicts administrative machine 306 and device 302 in greater detail, although the other devices 304 and 308–312 may contain similar components. Device 302 and administrative machine 306 communicate via Internet 314. Each device contains similar components, including a memory 502, 504; secondary storage 506, 508; a central processing unit (CPU) 510, 512; an input device 514, 516; and a video display 518, 520. One skilled in the art will appreciate that these devices may contain additional or different components. Memory 504 of administrative machine 306 includes the SASD process 540, VARPD 548, and KMS 550 all running in user mode. That is, CPU 512 is capable of running in at least two modes: user mode and kernel mode. When CPU 512 executes programs running in user mode, it prevents them from directly manipulating the hardware components, such as video

Claim 30	Caronni I
	display 518. On the other hand, when CPU 512 executes programs running in kernel mode, it allows them to manipulate the hardware components. Memory 504 also contains a VARPDB 551 and a TCP/IP protocol stack 552 that are executed by CPU 512 running in kernel mode. TCP/IP protocol stack 552 contains a TCP/UDP layer 554 and an IP layer 556, both of which are standard layers well known to those of ordinary skill in the art. Secondary storage 508 contains a configuration file 558 that stores various configuration-related information (described below) for use by SASD 540." Caronni I at 6:43–7:2.  FIG. 5 Caronni I at FIG. 5.

Claim 30	Caronni I
	"Memory 502 of device 302 contains SNlogin script 522, SNlogout script 524, VARPD 526, KMC 528, KMD 530, and node A 532, all running in user mode. Memory 502 also includes TCP/IP protocol stack 534 and VARPDB 536 running in kernel mode." Caronni I at 8:26-30. "FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:66-10:18.

Claim 30	Caronni I
	 FIG. 7A  FIG. 7B Caronni I at FIGS. 7A and 7B. To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Caronni II, Hipp, and/or Mehta. Caronni II, Hipp, and Mehta concern similar subject matter as

Claim 30	Caronni I
	this reference, namely, virtual networks. As set forth in the accompanying contentions, registration of devices in a virtual network was a well-known and commonly-used technique. Modifying this reference with Caronni II's, Hipp's, and/or Mehta's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with those of Caronni II, Hipp, or Mehta for example, to incorporate this limitation. *See, e.g.*, "In an embodiment of the present invention a virtual network is supported by a physical network and includes a virtual address resolution facility. The virtual address resolution facility is used to resolve a virtual IP address into a real IP address. A registration request is received at the virtual address resolution facility which includes a real IP address, a port address, a transport protocol designation and an Application layer protocol designation associated with a virtual address destination. The associations are stored using the virtual address resolution facility. A resolution request is received referencing a virtual address destination and the resolution request is resolved using the virtual address resolution facility and the stored associations." Caronni II at 3:11–23. "FIG. 2 depicts a block diagram of the VARP lookup table 26 used by the virtual address resolution facility 24. The VARP lookup table 26 may be stored at any location accessible over the network. The VARP lookup table 26 includes registered virtual destination addresses located in a column 78 and corresponding associations located in a column 79. Entries 80, 82, 84, 86 and 88 include the registered virtual destination address and information associated with the registered virtual destination address. Each entry in the VARP lookup table 26 will list the registering virtual IP address 90 and associated information including a real IP address 91, a transport protocol designation 92, a port number identification 93 and an Application layer protocol designation 94. The associations are provided at the time the virtual destination address is registered in the virtual network. For example, the entry 80 for the virtual IP address 10.0.0.12 (90) includes the association of a real IP address 152.70.0.1 (91), an associated Transport layer protocol designation, UDP 92,

Claim 30	Caronni I
	an associated port number, 6789 (93), and an Application layer protocol designation of 'none' 94." Caronni II at 5:10–29. "FIG. 4 depicts the sequence of steps followed by the illustrative embodiment of the present invention to process the received encapsulated message. The sequence begins when a process associated with a virtual IP address registers with the virtual address resolution facility 24 (step 140). The registration is stored in a VARP lookup table accessible over the network. Subsequently, a message is received at the electronic device which bears a MAC/network interface address of the network interface of the electronic device with the designated real IP address listed in the virtual address registration (Step 142). The MAC address header 117 is stripped off at the Link layer and the message and appended headers are passed up to the Network layer 114(Step 144). The Network layer 114 strips off the IP header 115 and identifies the Transport Protocol header 113 underneath. The Network layer passes the message and remaining appended header to the Transport layer 112 (Step 146). The Transport layer 112 strips off the Transport header and then passes the message to the Application layer via the port address of the interface identified in the VARP registration. (Step 148). The receiving process determines whether it is the virtual destination address the message is addressed to (Step 149) and acts accordingly. If the receiving process that is examining the message is the destination address, the process examines the data (Step 150). Alternatively, if the message is not intended for the process executing on the edge device, the process re-routes the message and sends it back down the protocol model stack, on to the local network operating behind the edge device, and on to its intended destination by performing traditional VARP and ARP lookups. Alternatively forwarding the message may be accomplished by reinserting the packet into the IP layer of the networking stack." Caronni II at 7:33–67. *See also* Caronni II at Claim 1, FIGS. 2-4. "FIG. 3 is a data flow diagram illustrating the registration of virtual network environment parameters. The VNE framework 200 is a software module that processes transactions between the applications and the operating system. The VNE parameters are registered with

Claim 30	Caronni I
	the VNE framework 200 at the time the application is started. The VNE parameters include the application IP address, the virtual network subnet, and the global virtual address subnet. At step 250, the registration harness 220 supplies the application IP address, virtual subnet, and the global virtual address subnet for a $process_x$ to the VNE framework 200. The VNE framework 200 then records the IP address, virtual subnet, and the global virtual address subnet for the $process_x$ at step 252. The $process_x$ can then spawn additional processes (or create additional objects) at step 254. the new $process_y$ inherits the IP address, virtual subnet, and global virtual address subnet from $process_x$. At step 256, the registration harness 220 launches the application related to $process_y$." Hipp at 3:52-4:2. "Virtualization of network identity is achieved by assigning a unique virtual IP address and virtual hostname to a group of processes that make up the application instance which the instance keeps throughout its execution. This virtual network identity stays with the application instance regardless of which node the application is running on. The framework, in essence, provides a mechanism to create this virtual network identity (VNI) around the application using the virtual network parameters assigned to it. In one embodiment, the virtual network parameters include an IP address and hostname. The framework 200 ensures that the application's instance uses the virtual network parameters, transparently, so that it can be moved across machines, without modifications to the application." Hipp at 6:1-15. *See also* Hipp at 6:16-60, FIG. 3. "Methods and systems for providing two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP, are provided. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporarily use by a requesting device on a public network to communicate with a wireless

Claim 30	Caronni I
	device on a wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more data repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the public network to send data to the wireless device." Mehta at Abstract. "To insure a greater degree of security, according to one embodiment, the AMPS maintains a Time to Live (TTL) parameter with each address mapping. In this way, once the TTL value indicates that the mapping has expired, the AMPS can destroy the mapping, and also any connection. Further, in some embodiments, the AMPS also puts a timestamp in its own mapping tables. After some timeout period based upon the timestamp, the AMPS can destroy the mapping, thereby forcing a new mapping to be initiated on a periodic basis." Mehta at [0015]. "In existing systems, data communication (communication on a data channel) between a wireless device that is connected to a private wireless network and a wired device connected to a public wired network (e.g., the Internet) can only be initiated by the wireless device. Some carriers have assigned fixed public IP addresses to wireless devices; however, the wireless devices need to then have client programs (e.g., a UDP stack) capability of receiving and handling the incoming communication packets. Moreover, these wireless devices are then part of a public wireless network and not a private wireless network. Since public IP addresses are becoming a scarcer commodity and currently expensive to service a network of multi-million devices, carriers cannot in a practical sense count on having a fixed public IP address for each device on its network. (Although movement to IPv6 will allow more addresses, the current IPv4 definitions are limited and the potential number of wireless users subscribing to larger carriers is very high. In some regions of the world, the current public address scheme is even more limited.) Further, such addressing capability would expose each

Claim 30	Caronni I
	device to further security risks, because each such device is part of a publicly accessible network and it becomes more difficult to implement and enforce security measures. Thus, assigning private IP addresses to devices is preferred in existing wireless networks over assigning fixed public IP addresses to devices. When wireless networks are private, the locations (addresses) of the wireless devices are intentionally hidden from public view by a carrier system's (or, as referred to in some countries, operator's) infrastructure." Mehta at [0025]. "Example embodiments described herein provide applications, tools, data structures and other support to implement private to public address mappings over one or more wired and wireless networks to be used for bi-directional communication. One skilled in the art will recognize that other embodiments of the methods and systems of the present invention may be used for many other purposes, including to push information and/or data or code from a public network such as the Internet to a wireless device. In addition, although this description primarily refers to "data" as being sent via the networks, one skilled in the art will recognize that all types of data can be communicated using the techniques described herein including, but not limited to, text, graphics, audio, and video." Mehta at [0034]. "In one embodiment, a timestamp of the mapping between a public network address and a private address is also noted in table 630. After a defined time out, based upon this timestamp, the Address Management Data Server sends a request to the Address Proxy/Router associated with the mapping to unmap the public-to-private address mapping and updates the mapping table 630, thereby returning the associated public address to the pool of unused public network addresses." Mehta at [0056].
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices	"Fourth, the system provides address translation in a transparent manner. Since the Supernet is a private network constructed from the infrastructure of another network, the Supernet has its own internal addressing scheme, separate from the addressing scheme of the underlying public network. Thus, when a packet from a Supernet node is sent to another Supernet node, it travels through the public network. To do so, the Supernet performs address translation

Claim 30	Caronni I
in the virtual network by the virtual network address; and	from the internal addressing scheme to the public addressing scheme and vice versa. By separating the addressing schemes, the Supernet creates a flexible delivery scheme that is easily changeable by network software or a system administrator. To reduce the complexity of Supernet nodes, system-level components of the Supernet perform this translation on behalf of the individual nodes so that it is transparent to the nodes. Another benefit of the Supernet's addressing is that it uses an IP-based internal addressing scheme so that preexisting programs require little modification to run within a Supernet." Caronni I at 6:7–25. "SASD 540 represents a Supernet: there is one instance of an SASD per Supernet, and it both authenticates nodes and authorizes nodes to join the Supernet. VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as a node IDs, to the network addresses recognized by the public-network infrastructure, known as the real addresses. The "node ID" may include the following: a Supernet ID (e.g., 0x123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1). Although the virtual address is described in an IP address scheme, one skilled in the art will appreciate that the virtual address may be any other type addressing scheme, such as an e-mail address, IPX, or IPv6. Since the node ID includes a Supernet ID, a node will have more than one node ID when it communicates over more than one channel. The "real address" is an IP address (e.g., 10.0.0.2) that is globally unique and meaningful to the public-network infrastructure. In a Supernet, one VARPD runs on each machine, and it may play two roles. First, a VARPD may act as a server by storing all address mappings for a particular Supernet into its associated VARPDB. Second, regardless of its role as a server or not, each VARPD assists in address translation for the nodes on its machine. In this role, the VARPD stores into its associated VARPDB the address mappings for its nodes, and if it needs a mapping that it does not have, it will contact the VARPD that acts as the server for the given Supernet to obtain it. The VARPDB may also decide which virtual address to use in the translation. That is, the VARPDB may associate a virtual address with multiple real addresses or vice versa." Caronni I at 7:3–33.

Claim 30	Caronni I
	"FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:66-10:18.

Claim 30	Caronni I

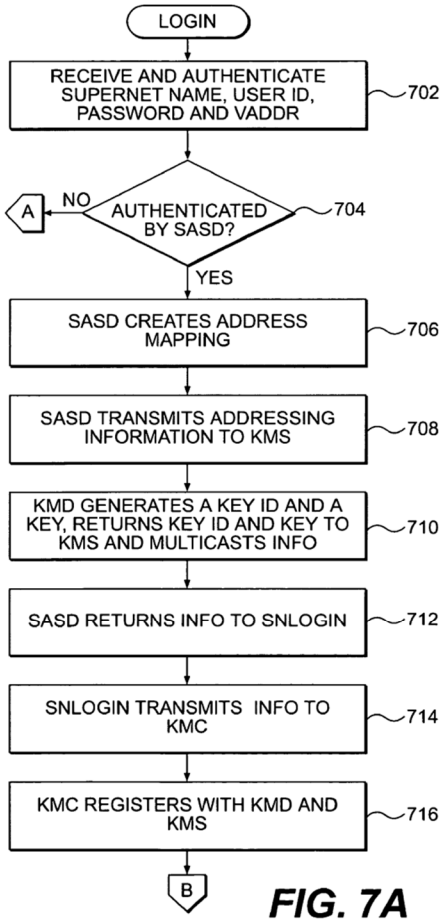


FIG. 7A

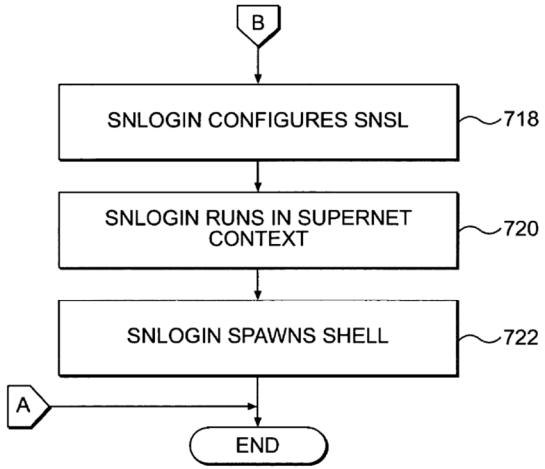


FIG. 7B

Caronni I at FIGS. 7A and 7B.

"Once inner IP layer 542 receives the packet, a Supernet ID is appended to a socket structure (step 804). The socket structure is modified so as to contain an extra data field for Supernet ID 626 and virtual source address 642. The addition of Supernet ID 626 and virtual address 642 in the socket structure enables the Supernet to communicate with nodes regardless of the

Claim 30	Caronni I
	delivery scheme used. When the process on node A opens a socket to transmit the packet to inner IP layer 540, the corresponding Supernet ID 626 and virtual source address 642 for that process is included in the socket request. The packet and Supernet ID are then transmitted to the SNSL layer using the modified socket structure (step 806). The SNSL layer then accesses the VARPDB to obtain the address mapping between virtual source node address 642 and the source real address 614 as well as the virtual destination node address 644 and the destination real address 616 (step 808). If they are not contained in the VARPDB because this is the first time a packet has been sent from this node or sent to this destination, the VARPDB accesses the local VARPD to obtain the mapping. When contacted, the VARPD on the local machine contacts the VARPD that acts as the server for the Supernet to obtain the appropriate address mapping. Since the VARPDB maintains all real IP addresses, a remote node may securely communicate with another remote node without reverfication." Caronni I at 11:37–61. To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Caronni II, Hipp, and/or Mehta. Caronni II, Hipp, and Mehta concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, distributing virtual network addresses was a well-known and commonly-used technique. Modifying this reference with Caronni II's, Hipp's, and/or Mehta's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with those of Caronni II, Hipp, or Mehta for example, to incorporate this limitation. *See, e.g.*, "Unfortunately, when the virtual destination address is located on a physical device on the interior of a network which is running a proxy server, firewall, or other packet filtering mechanism, messages that have been sent to a virtual destination address have difficulty getting all the way to their target. The term "interior of a network" refers to devices which are not able to directly access another network without first going through another device on

Claim 30	Caronni I
	their own network. For example, most local area networks (LANs) access the Internet through a proxy server. Devices other than the proxy server are said to be on the interior of the LAN. The proxy server is referred to as an "edge device" because it is able to directly contact another network without using an intermediary device. "Packet filtering" refers to the filtering of incoming messages or packets by an edge device or process on an edge device so that not all of the packets are permitted to proceed to their destination, they are "filtered out". If the electronic device that is filtering incoming packets, is under the control of the party executing the process associated with the virtual destination address, the device may be configured to allow the packets through to the end destination. However, in many situations, the edge device is not configurable by anyone without system administration privileges. Similarly, if the edge device is a device performing Network Address Translation (a "NAT box"), the NAT box rewrites all outgoing packets from an end user in the interior of the network to make them look like they came directly from the NAT box, and remembers that any traffic coming back from the particular destination address must be mapped back to the originating internal device. Consequently, the responding devices think they are responding to the sending device when they are actually responding to an edge device. In such a case, the packets intended for the virtual destination address on the interior of the physical network may be dropped and not reach their intended destination." Caronni II at 1:60–2:27. "…sending said formatted message, augmented by said determined MAC address destination, said real IP address destination and said transport protocol header, from a virtual IP address to the real IP address destination indicated in said virtual address destination registration." Caronni II at 9:20–24. "…a process associated with the sending virtual IP address is located on an electronic device outside a firewall and said virtual address destination is located inside the firewall." Caronni II at 9:31–32. *See also* Caronni II at 2:31-3:24, 6:32-54, FIGS. 1-5.

Claim 30	Caronni I
	"The Virtual Network Environment (VNE) of the present invention is defined by a collection of IP addresses. An application running within one VNE can communicate with another application in the same VNE. However, an application in one VNE cannot communicate with an application in another VNE (unless expressly permitted). These and many other attendant advantages of the present invention will be understood upon reading the following detailed description in conjunction with the drawings." Hipp at 2:8-16. "The VNE is specified at application run time. The VNE is transparent to the application and does not require any modifications to the application. The VNE is defined by subnet of addresses contained within the VNE. For example, all applications within the subnet 10.10.2.0 comprise a VNE. The subnet/netmask specifying such a VNE would be 10.10.2.0/255.255.255.0 and would include the addresses 10.10.2.0 through 10.10.2.255. In this example, an application with IP address 10.10.2.2 would be able to communicate with an application at address 10.10.2.60, but not at 10.10.0.1. Although using a subnet/netmask to specify the VNE is described herein for illustrative purposes, it is to be understood that other methods may be used to accomplish the same mechanism (e.g. an access control list)." Hipp at 3:16-30. "Virtualization of network identity is achieved by assigning a unique virtual IP address and virtual hostname to a group of processes that make up the application instance which the instance keeps throughout its execution. This virtual network identity stays with the application instance regardless of which node the application is running on. The framework, in essence, provides a mechanism to create this virtual network identity (VNI) around the application using the virtual network parameters assigned to it. In one embodiment, the virtual network parameters include an IP address and hostname. The framework 200 ensures that the application's instance uses the virtual network parameters, transparently, so that it can be moved across machines, without modifications to the application." Hipp at 6:1-15. *See also* Hipp at Figures 1–7.

Claim 30	Caronni I
	"Methods and systems for providing two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP, are provided. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporarily use by a requesting device on a public network to communicate with a wireless device on a wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more data repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the public network to send data to the wireless device." Mehta at Abstract. "To insure a greater degree of security, according to one embodiment, the AMPS maintains a Time to Live (TTL) parameter with each address mapping. In this way, once the TTL value indicates that the mapping has expired, the AMPS can destroy the mapping, and also any connection. Further, in some embodiments, the AMPS also puts a timestamp in its own mapping tables. After some timeout period based upon the timestamp, the AMPS can destroy the mapping, thereby forcing a new mapping to be initiated on a periodic basis." Mehta at [0015]. "In existing systems, data communication (communication on a data channel) between a wireless device that is connected to a private wireless network and a wired device connected to a public wired network (e.g., the Internet) can only be initiated by the wireless device. Some carriers have assigned fixed public IP addresses to wireless devices; however, the wireless devices need to then have client programs (e.g., a UDP stack) capability of receiving

Claim 30	Caronni I
	and handling the incoming communication packets. Moreover, these wireless devices are then part of a public wireless network and not a private wireless network. Since public IP addresses are becoming a scarcer commodity and currently expensive to service a network of multi-million devices, carriers cannot in a practical sense count on having a fixed public IP address for each device on its network. (Although movement to IPv6 will allow more addresses, the current IPv4 definitions are limited and the potential number of wireless users subscribing to larger carriers is very high. In some regions of the world, the current public address scheme is even more limited.) Further, such addressing capability would expose each device to further security risks, because each such device is part of a publicly accessible network and it becomes more difficult to implement and enforce security measures. Thus, assigning private IP addresses to devices is preferred in existing wireless networks over assigning fixed public IP addresses to devices. When wireless networks are private, the locations (addresses) of the wireless devices are intentionally hidden from public view by a carrier system's (or, as referred to in some countries, operator's) infrastructure." Mehta at [0025]. "FIG. 3 is an example block diagram of components of an example Address Management Proxy System. In one embodiment, the Address Management Proxy System (AMPS) comprises one or more modified DNS'/ API servers 302, one or more Address Proxy/Routers 305, an Address Management Data Server 303 which manages a database or other repositories such as Address Management Data Repository 304, and optionally a load balancer 301. The DNS'/API servers 302 are either individually connected to a public network 310 or are connected to the load balancer 301 which is in turn connected to public network 310. Similarly, each Address Proxy/Router 305 is also connected to the public network 310, via the routable (public) address to which data from the external network 310 is sent that is destined for the wireless devices. The DNS'/API servers 302 are either modified implementations of a DNS server to add functionality necessary to communicate with wireless devices, or are servers that implement one or more specialized APIs, as will be described further below. The DNS'/API servers 302 use the Address Management Data Server 303 to assist in mapping a unique identifier (e.g., string name) for a wireless device to a public address on public network 310. The pool of public addresses is also maintained by the Address Management Data Serve 303 and Data Repository 304. The Address

Claim 30	Caronni I
	Management Data server 303 and Address Management Data Repository 304 may be implemented using existing database technology, for example, ODBC technology or, may be implemented as a structure such as a simple text file. One skilled in the art will recognize that any embodiment for storing a set of tables, data, lists, or mappings can be used. Each Address Proxy/Router 305 also uses the Address Management Data Server 303 or equivalent to create and update a series of routing tables that are used to assign public addresses to wireless devices as needed and to update the various mappings between public addresses and the non-routable (private) addresses of the wireless devices. The tables and mappings that are maintained on behalf of the DNS'/API servers 302 and the Address Proxy/Routers 305 by the Address Management Data Server 303 are described below with reference to FIG. 6." Mehta at [0032]. "Although the techniques of the AMPS are generally applicable to any a wired device communicating with a wireless device, the phrase "public network" (or "wired network") is used generally to imply any type of internetworked environment including a public network or a backbone that is somewhere down the line connected to one or more private or public networks. In addition, although the examples described herein often refer to the Internet, one skilled in the art will recognize that the concepts and inventions described are applicable to other forms and embodiments of internetworking, including, for example ATM type networks. Thus, techniques of the present invention can also be used by one device on a first wireless network to communicate with another wireless device on a second network—each device ends up communicating with the Address Proxy/Router of the other network. This scenario is feasible because each wireless network (or its carrier infrastructure) is connected to a proxy/router that is also connected (via a public address) to a public network. In addition, although a public network is sometimes also referred to herein as a wired network, one skilled in the art will recognize that any network that exposes routable (public) addresses may be implied. Thus, a wireless network with unique public (and routable) address can also employ techniques of the present invention to perform bi-directional communication. Also, one skilled in the art will recognize that terms such as wireless device, phone, handheld, etc., are used interchangeably to indicate any type of wireless device that is capable of operating with the AMPS. In addition, terms may have alternate spellings which may or may not be

Claim 30	Caronni I
	explicitly mentioned, and one skilled in the art will recognize that all such variations of terms are intended to be included." Mehta at [0033]. "Example embodiments described herein provide applications, tools, data structures and other support to implement private to public address mappings over one or more wired and wireless networks to be used for bi-directional communication. One skilled in the art will recognize that other embodiments of the methods and systems of the present invention may be used for many other purposes, including to push information and/or data or code from a public network such as the Internet to a wireless device. In addition, although this description primarily refers to "data" as being sent via the networks, one skilled in the art will recognize that all types of data can be communicated using the techniques described herein including, but not limited to, text, graphics, audio, and video." Mehta at [0034]. "There are several implementation approaches to the components of the Address Management Proxy System, three of which are described herein. One skilled in the art will recognize that various other approaches and combinations are possible. All three approaches allocate a public (routable) network address for temporary use by a wired device to communicate with a wireless device. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained and allocated dynamically to wireless network devices as required. For example, a typical Class B Internet network address block allows for approximately 65,000 simultaneous connections to wireless devices. Although this number may seem large at first blush, when one considers the number of cell phones and handsets, for example, connected to a carrier, this number can be quite limiting." Mehta at [0039]. "The unique ID table 610 maps unique string names of wireless devices to the private wireless network addresses that have been assigned typically by a carrier's infrastructure. In some embodiments, the carrier infrastructure dynamically allocates, using methods similar to a DHCP protocol, a private wireless network address when the wireless device registers itself with the carrier infrastructure upon being powered on. Thus, the unique ID table 610 may be

Claim 30	Caronni I
	sparsely filled or entries created dynamically and then deleted dynamically as devices register and unregister with the carrier infrastructure system." Mehta at [0053]. "In one embodiment, a timestamp of the mapping between a public network address and a private address is also noted in table 630. After a defined time out, based upon this timestamp, the Address Management Data Server sends a request to the Address Proxy/Router associated with the mapping to unmap the public-to-private address mapping and updates the mapping table 630, thereby returning the associated public address to the pool of unused public network addresses." Mehta at [0056]. "FIG. 7 is an example flow diagram of an example routine provided by a DNS'/API server of the Address Management Proxy System to return a public address that corresponds to a designated unique identifier. In essence, this routine implements a DNS query or DNS-like query capability for the AMPS using a modified GetHostByName interface or a specialized API, for example GetProxyIP, as described with reference to FIG. 5. In summary, the routine dynamically allocates an appropriate Address Proxy/Router machine to associate with the wireless device and returns the public address of that machine (along with a TTL parameter and potentially other parameters). Specifically, in step 701, the routine determines the private (non-routable) address of the wireless device designated by a string parameter passed as input to the routine. For example, the string parameter may use fields such as "uniqueID.hostname.domain.tld," which specifies a typical hierarchy of person/service on a machine named "hostname" on a domain such as a company's network on a top level domain such as "org." "com," "edu," etc. One skilled in the art will recognize that many other string parameter designations could be used. One mechanism for implementing this routine is to request information from the Address Management Data Repository. In one embodiment, the data repository stores a table that maps unique IDs to private network addresses (see Table 610 in FIG. 6). Preferably, any mechanism that is used by the AMPS stores this data in a secure manner in order to keep the wireless network addresses private. In step 702, the routine retrieves the public network address that corresponds to the private wireless network address of the designated device if one has already been assigned by the AMPS to that

Claim 30	Caronni I
	device and is still valid. In one embodiment, the data repository stores this mapping information between private wireless network address and public network address (see for example table 630 in FIG. 6). If a public network address has not already been assigned or is not valid, then the routine causes a new public network address to be allocated and that new public address is associated with the private wireless network address. Appropriate tables in the data repository are then updated. In step 703, the routine determines the Address Proxy/Router machine that is associated with the assigned public network address (for example using table 620 in FIG. 6). In step 704, the routine sends a request to the determined proxy/router machine to update its routing tables to map the determined public network address to the private wireless network address. In step 705, the routine updates information in the data repository to include any other connection related information (e.g., field 634 in Table 630 in FIG. 6) and indicates a Time to Live (TTL) parameter for the public-private address association (e.g., field 633 in Table 630 in FIG. 6). Once all of the tables have been updated in both the proxy/router and in the data repository, the DNS'/API server returns the determined public network address of the associated Address Proxy/Router machine. As described earlier, the public address may be a (hostname, port) pair when a port-based implementation is used." Mehta at [0057]. *See also* Mehta at FIG. 3.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address	"To perform this functionality, D 1 108 utilizes a technique known as tunneling to ensure that the communication between itself and enterprise network 102 is secure in that it cannot be viewed by an interloper. "Tunneling" refers to encapsulating one packet inside another when packets are transferred between two end points (e.g., D 1 108 and VPN software 109 running on firewall 106). The packets may be encrypted at their origin and decrypted at their destination. For example, FIG. 2A depicts a packet 200 with a source Internet protocol (IP) address 202, a destination IP address 204, and data 206. It should be appreciated that packet 200 contains other information not depicted, such as the source and destination port. As shown in FIG. 2B, the tunneling technique forms a new packet 208 out of packet 200 by encrypting it and adding both a new source IP address 210 and a new destination IP address 212. In this manner, the contents of the original packet (i.e., 202, 204, and 206) are not

Claim 30	Caronni I
associated with the second device.	visible to any entity other than the destination. Referring back to FIG. 1, by using tunneling, remote device D 1 108 may communicate and utilize the resources of the enterprise network 102 in a secure manner." Caronni I at 2:26–46. "SASD 540 represents a Supernet: there is one instance of an SASD per Supernet, and it both authenticates nodes and authorizes nodes to join the Supernet. VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as a node IDs, to the network addresses recognized by the public-network infrastructure, known as the real addresses. The "node ID" may include the following: a Supernet ID (e.g., 0x123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1). Although the virtual address is described in an IP address scheme, one skilled in the art will appreciate that the virtual address may be any other type addressing scheme, such as an e-mail address, IPX, or IPv6. Since the node ID includes a Supernet ID, a node will have more than one node ID when it communicates over more than one channel. The "real address" is an IP address (e.g., 10.0.0.2) that is globally unique and meaningful to the public-network infrastructure. In a Supernet, one VARPD runs on each machine, and it may play two roles. First, a VARPD may act as a server by storing all address mappings for a particular Supernet into its associated VARPDB. Second, regardless of its role as a server or not, each VARPD assists in address translation for the nodes on its machine. In this role, the VARPD stores into its associated VARPDB the address mappings for its nodes, and if it needs a mapping that it does not have, it will contact the VARPD that acts as the server for the given Supernet to obtain it. The VARPDB may also decide which virtual address to use in the translation. That is, the VARPDB may associate a virtual address with multiple real addresses or vice versa." Caronni I at 7:3–33. "SNSL 542 utilizes VARPDB 536 to perform address translation. VARPDB stores all of the address mappings encountered thus far by SNSL 542. If SNSL 542 requests a mapping that VARPDB 536 does not have, VARPDB communicates with the VARPD 526 on the local machine to obtain the mapping. VARPD 526 will then contact the VARPD that acts as the server for this particular Supernet to obtain it."

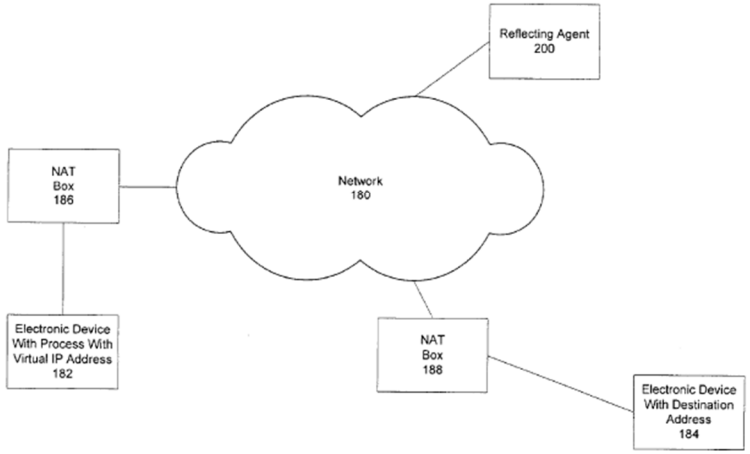
Claim 30	Caronni I
	Caronni I at 9:47–54. "FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:66-10:18. "The packet and Supernet ID are then transmitted to the SNSL layer using the modified socket structure (step 806). The SNSL layer then accesses the VARPDB to obtain the address mapping between virtual source node address 642 and the source real address 614 as well as the virtual destination node address 644 and the destination real address 616 (step 808). If they are not contained in the VARPDB because this is the first time a packet has been sent from this node or sent to this destination, the VARPDB accesses the local VARPD to obtain the mapping. When contacted, the VARPD on the local machine contacts the VARPD that acts as the server for the Supernet to obtain the appropriate address mapping. Since the VARPDB maintains all real IP addresses, a remote node may securely communicate with another remote node without reverfication." Caronni I at 11:47–61. "After obtaining the address mapping, the SNSL layer determines whether it has been configured to communicate over the appropriate channel for this packet (step 806). This configuration occurs when SNlogin runs, and if the SNSL has not been so configured,

Claim 30	Caronni I
	processing ends. Otherwise, SNSL obtains the channel key to be used for this channel (step 808). The SNSL maintains a local cache of keys and an indication of the channel to which each key is associated. Each channel key is time stamped to expire in ten seconds, although this time is configurable by the administrator. If there is a key located in the cache for this channel, SNSL obtains the key. Otherwise, SNSL accesses KMD which then locates the appropriate channel key from the appropriate KMC. After obtaining the key, the SNSL layer encrypts the packet using the appropriate encryption algorithm and the key previously obtained (step 810). When encrypting the packet, the virtual source node address 642, the virtual destination node address 644, and the data may be encrypted (addressing section 660), but the source and destination real addresses 614, 616 (delivery scheme section 670) are not, so that the real addresses can be used by the public network infrastructure to send the packet to its destination. By encrypting addressing scheme 660, the Supernet enables data to be transmitted securely and at the same time transparent from delivery scheme used." Caronni I at 11:62-12:19. "Web client 1102 has a virtual address obtained from computer system 1106, as described in FIGS. 7A and 7B. Each time web client 1102 requests a packet from web server 1104 a, the client requests the virtual address of the web server 1104 a from computer system 1106. If web server 1104 a becomes overloaded (e.g., unable to handle more requests), the overloaded web server spawns new instances of the same web server (web servers 1104 b and 1104 c) and at the same time notifies the computer system 1106. The VARP server of computer 1106 then associates the new instances 104 b and 104 c of the web server with web server's 1104 virtual address. As a result, web client 1102 is not notified of the change and continues to use the same virtual address as previously." Caronni I at 13:13–26. To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Caronni II, Hipp, Mehta, and/or RFC 1383. Caronni II, Hipp, Mehta, and RFC 1383 concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, use of a server such as a DNS server that returns network addresses

Claim 30	Caronni I
	was a well-known and commonly-used technique. Modifying this reference with Caronni II's, Hipp's, Mehta's, and/or RFC 1383's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with those of Caronni II, Hipp, Mehta, or RFC 1383 for example, to incorporate this limitation. *See, e.g.*, "In one embodiment, a virtual network supported by a physical network includes a virtual address resolution facility. The virtual address resolution facility is used to resolve a virtual IP address into a real IP address. A virtual address is registered with the virtual address resolution facility and the registration includes a real IP address, a port number and a transport protocol designation associated with a virtual address. The virtual address destination is resolved using the virtual address resolution facility and a message is sent from a virtual address in the virtual network to the real IP address indicated in the virtual address destination registration. In one aspect of the embodiment, the registration also includes an Application layer protocol designation." Caronni II at 2:52–64. "In an embodiment of the present invention a virtual network is supported by a physical network and includes a virtual address resolution facility. The virtual address resolution facility is used to resolve a virtual IP address into a real IP address. A registration request is received at the virtual address resolution facility which includes a real IP address, a port address, a transport protocol designation and an Application layer protocol designation associated with a virtual address destination. The associations are stored using the virtual address resolution facility. A resolution request is received referencing a virtual address destination and the resolution request is resolved using the virtual address resolution facility and the stored associations." Caronni II at 3:11–23. "The physical network 20 is also interfaced with an electronic device 30 located at the edge of a local area physical network 50 which includes devices addressable using IP addresses in the 129.63.1.0/24 range. The electronic device 30 may be a proxy server, gateway, NAT box,

Claim 30	Caronni I
	or other device and may perform packet filtering. Alternatively, a firewall 51 may be executed in either a hardware or software form to filter packets sent to the local area physical network 50 from the physical network 20. The firewall 51 may be located outside the electronic device 30 or alternatively may be running in software form on the electronic device 30. Those skilled in the art will recognize that a number of different types of networks may be utilized within the scope of the present invention. The networks may be the Internet, an intranet, wide area network (WAN), a local area network (LAN), a satellite network, a wireless network, or some other type of network capable of supporting a virtual network through the devices thereon." Caronni II at 4:17–33. "FIG. 2 depicts a block diagram of the VARP lookup table 26 used by the virtual address resolution facility 24. The VARP lookup table 26 may be stored at any location accessible over the network. The VARP lookup table 26 includes registered virtual destination addresses located in a column 78 and corresponding associations located in a column 79. Entries 80, 82, 84, 86 and 88 include the registered virtual destination address and information associated with the registered virtual destination address. Each entry in the VARP lookup table 26 will list the registering virtual IP address 90 and associated information including a real IP address 91, a transport protocol designation 92, a port number identification 93 and an Application layer protocol designation 94. The associations are provided at the time the virtual destination address is registered in the virtual network. For example, the entry 80 for the virtual IP address 10.0.0.12 (90) includes the association of a real IP address 152.70.0.1 (91), an associated Transport layer protocol designation, UDP 92, an associated port number, 6789 (93), and an Application layer protocol designation of 'none' 94." Caronni II at 5:10–29. "The illustrative embodiment of the present invention attempts to circumvent problems caused by packet filtering and network address translation occurring at the edge of networks through the use of higher layer protocols. FIG. 3 depicts a block diagram of the encapsulation process used to send a message from a virtual IP address to another virtual IP address located at the interior of a network at which packet filtering and/or network address

Claim 30	Caronni I
	translation is occurring. The virtual address resolution facility 24 is used to determine the associations registered with the destination address. The original message 111 at the Application layer 110 is formatted in a format specified in the registration of the virtual destination address if one is so indicated (e.g. HTTP). The message 111 is passed down to the Transport layer 112. The Transport layer 112 takes the original message 111 and adds a transport protocol header 113 as specified in the virtual address destination registration. The transport protocol header 113 may be a UDP header, TCP header, an X.25 header, XTP header, AppleTalk header, or other similar transport protocol header. Those skilled in the art will realize that the message may include an IPSEC header, inner IP header, UDP or TCP header and inner payload." Caronni II at 6:32-53. "The illustrative embodiment of the present invention may also be utilized to send a message from a virtual address located behind a NAT box to a destination address on the same virtual network that is behind a different NAT box. Ordinarily, where the destination address is directly connected to the Internet (i.e.: in situations where the destination is not behind a NAT box), a connection, such as a TCP connection, may be established between the originator of the connection behind the NAT box and the destination. The originator's message passes the NAT box and the internal mapping of public address and port number to internal address and port number takes place. Return packets from the destination may then be received following the mapping. When the destination address is also behind a NAT box, the originator of the message is unable to directly address the destination (since the address of the destination may not be routable in the public Internet). In such a case, the present invention adds an entry to the VARP table for a third-party reflecting agent located at an address outside the NAT box. Messages that are being sent from the originating virtual address behind a NAT box to a destination which is behind a different NAT box are sent via the reflecting agent intermediary outside the NAT box and reflected to the destination. FIG. 5 depicts a block diagram of an environment holding a virtual network in which both the originator and destination of a message are located behind NAT boxes. An electronic device holding an originating process 182 which is part of a virtual network is interfaced with a network 180, such as the Internet, via a NAT box 186. An intended recipient for a message sent from the originating process 180 is an electronic device with a destination

Claim 30	Caronni I
	address 184 that is also behind a NAT box 188. A reflecting agent 200, the address of which has been added to a VARP table, acts as an intermediary to allow a connection between the originating process 182 and destination address 184." Caronni II at 8:1–38. Figure 5  Caronni II at FIG. 5. "Beginning at step 270, a client application requests to the VNE framework 200 to connect or send to an address of another application, such as a server application having the address 10.10.2.70: port 9001. At step 272, the VNE framework 200 requests the VNE parameters for the process corresponding to the client application from a process state storage structure. The structure that stores process state is a structure that the operating system uses to store private information about the process. Therefore, it may differ depending on the operating system. The parameters added the process state storage structure as part of the virtual network environment are listed below:

Claim 30	Caronni I

```
typedef struct {
        ipaddr_t    app_address;
        ipaddr_t    virtual_subnet;
        ipaddr_t    virtual_mask;
        ipaddr_t    global_subnet;
        ipaddr_t    global_mask;
} vne_param_t;
```

Hipp at 4:44–64.

> "Virtualization of network identity is achieved by assigning a unique virtual IP address and virtual hostname to a group of processes that make up the application instance which the instance keeps throughout its execution. This virtual network identity stays with the application instance regardless of which node the application is running on. The framework, in essence, provides a mechanism to create this virtual network identity (VNI) around the application using the virtual network parameters assigned to it. In one embodiment, the virtual network parameters include an IP address and hostname. The framework 200 ensures that the application's instance uses the virtual network parameters, transparently, so that it can be moved across machines, without modifications to the application.
> The virtual hostname resolves to the virtual IP address for both the applications registered with the VNI framework as well as those that are not registered. This may require configuration of a name service or OS host configuration files. For example, if an application instance used a virtual IP address of 10.10.0.1 and a virtual hostname of host1055, the standard hostname to IP address resolution mechanisms (e.g. DNS or the /etc/hosts file) would have to be preconfigured to resolve a query of host1055 to IP address 10.10.0.1."

Hipp at 6:1–26.

> "A network that supports IP can be connected to another TCP/IP-based or UDP/IP-based internet or the Internet, by providing a router which forwards data to another router or host

Claim 30	Caronni I
	machine based upon an IP address. The router (or routing server/service) typically contains a routing table, which determines to which machine (and optionally to which port) to send a datagram, given a destination IP address. The IP address uniquely identifies a router/host machine, and, in a typical TCP/IP network, can be mapped to a string name that identifies, for example, a particular machine as part of a larger domain. (Although referred to herein often as a TCP/IP-based network, one skilled in the art will recognize that the network may also be UDP-based (connectionless), and may support another session management system.)" Mehta at [0008]. "In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more Address Management Data Repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the external public network to send data to the wireless device. The temporary public address is, for example, an address associated with one of the Address Proxy/Routers, which are connected to the external public network and have access to the private addresses on the private wireless network. In some cases, the device on the public network that desires to send data to the wireless device uses a connection-based protocol, such as TCP/IP to send data. In other cases the device uses a connection-less protocol, such as UDP (UDP/IP) to send the data." Mehta at [0012]. "Wireless systems on private networks use techniques similar to Network Address Translation (NAT) technology to send data from a wireless device to the public networking world. In a typical carrier infrastructure that uses a private network, a wireless device "registers" itself with the carrier infrastructure when it powers on (or in other circumstances, when the device attempts to initiate data services). The carrier dynamically assigns the wireless device a private non-routable address by means of DHCP (or DHCP-like) server. The information on the mapping of the transient private IP address to the device public IP

Claim 30	Caronni I
	address is stored in an internal carrier database and managed by carrier services such as a RADIUS server." Mehta at [0026]. "The Address Management Proxy System achieves two way initiated bi-directional communication by implementing a modified DNS server and serving as a proxy/router for devices on the private wireless network as they interface to the public wired internetworking world. In summary, a pool of public addresses is maintained and dynamically distributed among active wireless devices as needed by the AMPS. FIG. 2 is a block diagram of an example Address Management Proxy System used in bi-directional communication with a wireless device. The term "bi-directional" as used herein means that data paths and communication can flow in either direction between two endpoint systems. FIG. 2 shows wired devices 201, 202, and 203 connected to public network 210. One skilled in the art will recognize that these devices could be connected to another private or public network, which is then connected by one or more wired devices to the public network 210, yet still achieve the functionality discussed here. Any such variation provides equivalent functionality and is explicitly contemplated and presumed to be part of the present invention. On the wireless side, the AMPS 220, acting in its capacity as a proxy (and router) for wireless devices, is shown both connected via wire to the public network 210 and via standard carrier infrastructure elements (not shown) to the wireless network 230. It is presumed that the reader has a working knowledge of the elements of a carrier's infrastructure and the basic mechanisms for routing and mechanisms converting packets from a wired network to a wireless network. These may use analog or digital technology and may require protocol conversions in order to send the physical data and transmit it, for example through a satellite, ultimately to the wireless device. Detailed background information on wireless technology and wireless routing mechanisms is described in Stallings, W., Wireless Communications and Networks, Prentice Hall, N.J., 2002, which is herein incorporated by reference in its entirety. In FIG. 2, wireless device 240 is shown connected via various wireless elements (not shown) to the wireless network 230." Mehta at [0030].

Claim 30	Caronni I
	"FIG. 6 is an example block diagram of some of the Address Management Proxy System data repository tables used to support routines of the DNS'/API servers and the Address Proxy/Routers. In one embodiment, the Address Management Data Repository comprises three tables: a unique identifier (unique ID) to private address table 610, a public-to-private address table 630, and a public address to proxy/router machine table 620. Although three tables are shown, one skilled in the art will recognize that these tables could contain other data and may be organized differently, including a different number of tables and with different columns or fields. In addition, any technique for storing a table or list of data may be used. To support embodiments that map a host address plus a port designator to a wireless device, the tables are correspondingly modified." Mehta at [0052]. "The unique ID table 610 maps unique string names of wireless devices to the private wireless network addresses that have been assigned typically by a carrier's infrastructure. In some embodiments, the carrier infrastructure dynamically allocates, using methods similar to a DHCP protocol, a private wireless network address when the wireless device registers itself with the carrier infrastructure upon being powered on. Thus, the unique ID table 610 may be sparsely filled or entries created dynamically and then deleted dynamically as devices register and unregister with the carrier infrastructure system." Mehta at [0053]. The public-to-private address table 630 comprises several fields/columns including a public network address 631, a private (wireless) network address 602, a flag 632 that specifies whether the public address stored in field 631 is free or is already used, a Time to Live (TTL) parameter 633, and other connection data 634. In one embodiment, the DNS'/API servers of the AMPS query table 630 to determine a public network address that corresponds to a designated private wireless network address or to allocate an unused public network address (as indicated in field 632) and map the determined unused public address to a private network address stored in field 602. Mehta at [0054].

Claim 30	Caronni I
	Public address to proxy/router machine table 620 comprises a public network address field 631 and an indication of a functioning proxy/router machine 621. By maintaining such a mapping, the AMPS is able to substitute proxy/router machines for other proxy/router machines to provide a higher degree of robustness. Each proxy/router machine has a preconfigured set of public network addresses, such as are typically configured by network cards inserted into the proxy/router machine. These address are allocated in a standard fashion through prior purchase or obtaining from an address authorizing authority, currently the Internet Corporation for Assigned Names and Numbers (ICANN). When a machine is inserted for use into the AMPS, indications of these addresses are stored in field 631. Mehta at [0055]. "In one embodiment, a timestamp of the mapping between a public network address and a private address is also noted in table 630. After a defined time out, based upon this timestamp, the Address Management Data Server sends a request to the Address Proxy/Router associated with the mapping to unmap the public-to-private address mapping and updates the mapping table 630, thereby returning the associated public address to the pool of unused public network addresses." Mehta at [0056]. "FIG. 7 is an example flow diagram of an example routine provided by a DNS'/API server of the Address Management Proxy System to return a public address that corresponds to a designated unique identifier. In essence, this routine implements a DNS query or DNS-like query capability for the AMPS using a modified GetHostByName interface or a specialized API, for example GetProxyIP, as described with reference to FIG. 5. In summary, the routine dynamically allocates an appropriate Address Proxy/Router machine to associate with the wireless device and returns the public address of that machine (along with a TTL parameter and potentially other parameters). Specifically, in step 701, the routine determines the private (non-routable) address of the wireless device designated by a string parameter passed as input to the routine. For example, the string parameter may use fields such as "uniqueID.hostname.domain.tld," which specifies a typical hierarchy of person/service on a machine named "hostname" on a domain such as a company's network on a top level domain such as "org." "com," "edu," etc. One skilled in the art will recognize that many other string

Claim 30	Caronni I
	parameter designations could be used. One mechanism for implementing this routine is to request information from the Address Management Data Repository. In one embodiment, the data repository stores a table that maps unique IDs to private network addresses (see Table 610 in FIG. 6). Preferably, any mechanism that is used by the AMPS stores this data in a secure manner in order to keep the wireless network addresses private. In step 702, the routine retrieves the public network address that corresponds to the private wireless network address of the designated device if one has already been assigned by the AMPS to that device and is still valid. In one embodiment, the data repository stores this mapping information between private wireless network address and public network address (see for example table 630 in FIG. 6). If a public network address has not already been assigned or is not valid, then the routine causes a new public network address to be allocated and that new public address is associated with the private wireless network address. Appropriate tables in the data repository are then updated. In step 703, the routine determines the Address Proxy/Router machine that is associated with the assigned public network address (for example using table 620 in FIG. 6). In step 704, the routine sends a request to the determined proxy/router machine to update its routing tables to map the determined public network address to the private wireless network address. In step 705, the routine updates information in the data repository to include any other connection related information (e.g., field 634 in Table 630 in FIG. 6) and indicates a Time to Live (TTL) parameter for the public-private address association (e.g., field 633 in Table 630 in FIG. 6). Once all of the tables have been updated in both the proxy/router and in the data repository, the DNS'/API server returns the determined public network address of the associated Address Proxy/Router machine. As described earlier, the public address may be a (hostname, port) pair when a port-based implementation is used." Mehta at [0057]. *See also* Mehta at [0004], [0010], [0032], [0036]-[0038], [0048], [0058]-[0059], FIGS. 6, 7. "1. Routing, scaling and hierarchies Several recent studies have outlined the risk of "routing explosion" in the current Internet: there are already more than 5000 networks announced in the NSFNET routing tables, more

Claim 30	Caronni I
	than 7000 in the EBONE routing tables. As these numbers are growing, several problems occur: * The size of the routing tables grows linearly with the number of connected networks; handling this larger tables requires more resources in all "intelligent" routers, in particular in all "transit" and "external" routers that cannot rely on default routes. * The volume of information carried by the route exchange protocols such as BGP grows with the number of networks, using more network resources and making the reaction to routing events slower. * Explicit administrative decisions have to be exercised by all transit networks administrators which want to implement "routing policies" for each and every additional "multi-homed" network. The current "textbook" solution to the routing explosion problem is to use "hierarchical routing" based on hierarchical addresses. This is largely documented in routing protocols such as IDRP, and is one of the rationales for deploying the CIDR [3] addressing structure in the Internet. This textbook solution, while often perfectly adequate, as a number of inconveniences, particularly in the presence of "multihomed stubs", e.g., customer networks that are connected to more than one service providers. The current proposal presents a scheme that allows for simple routing. It is complementary with the classic "hierarchical routing" approach, but provides an easy to implement and low cost solution for "multi-homed" domains. The solution is a generalization of the "MX record" scheme currently used for mail routing." RFC 1383 at 1-2. "3.1. Loops and relays In the introduction to DNS-IP routing, we mentioned that the packets would be directed towards the access gateway I1 or I2 by means of "source routing" or "tunnelling". This is not,

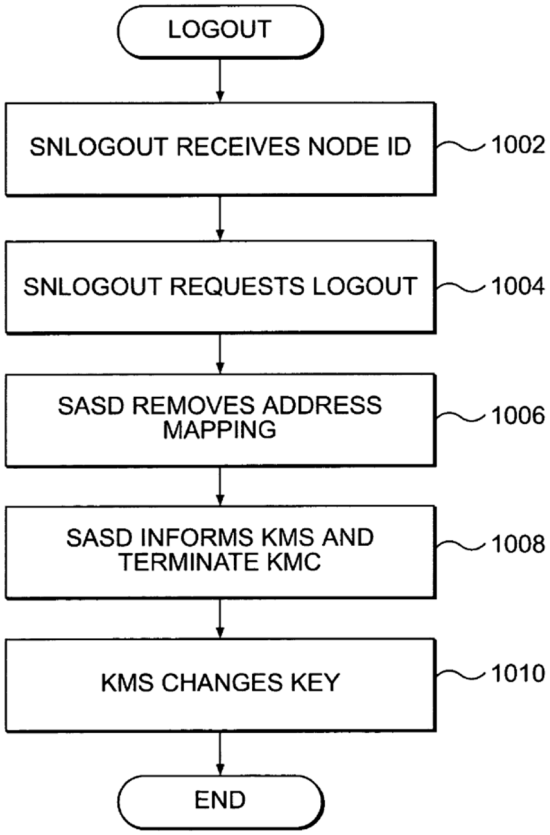
Claim 30	Caronni I
	stricto sensu, necessary. One could imagine that the packet would simply be routed "as if it was directed towards I1 or I2". The next relay would, in turn, also access the DNS to get routing information and forward the packet. Such a strategy would have the advantage of leaving the header untouched and of letting the transit nodes choose the best routing towards the destination, based on their knowledge of the reachability status. It would however have two important disadvantages: - It would oblige all intermediate relays to access the DNS, - It would oblige all these relays to exploit consistently the DNS information. Obliging all intermediate gateways to access the DNS is impractical in the short term: it would mean that we would have to update each and every transit relay before deploying the scheme. It could also have an important performance impact: the "working set" of transit relays is typical much wider than that of stub gateways, and the argument presented previously on the efficiency of caches may not apply. This would perhaps remain impractical even in the long term, as it the volume of DNS traffic could well become excessive. The second argument would apply even if the performance problem had been solved. Suppose that several RX records are registered for a given destination, such as I1 and I2 for Dx in our example, and that a "hop by hop routing" strategy is used. There would be a fair risk that some relays would choose to route the packet towards I1 and some others towards I2, resulting in inefficient routing and the possibility of loops. In order to ensure coherency, we propose that all routing decisions be made at the source, or by one of the first relays near the source." RFC 1383 at 4-5. "3.4. Choosing a gateway

Claim 30	Caronni I
	A simplification to the previous problem would be to allow only one RX record per destination, thus guaranteeing consistent decisions in the network. This would however have a number of draw-backs. A single access point would be a single point of failure, and would be connected to only one transit network thus keeping the "customer locking" effect of hierarchical routing. We propose that the RX records have a structure parallel to that of MX records, i.e., that they carry associated with each gateway address a preference identifier. The source host, when making the routing decision based on RX records, should do the following: - List all possible gateways, - Prune all gateways in the list which are known as "unreachable" from the local site, - If the local host is present in the list with a preference index "x", prune all gateways whose preference index are larger than "x" or equal to "x". - Choose one of the gateway in the list. If the list is empty, consider the destination as unreachable. Indeed, these evaluations should not be repeated for each and every packet. The routers should maintain a cache of the most frequently used destinations, in order to speed up the processing." RFC 1383 at 6. "5.1. DNS record In a definitive scheme, it would be necessary to define a DNS record type and the corresponding "RX" format. In order to deploy this scheme, we would then have to teach this new format to the domain name service software. While not very difficult to do, this would probably take a couple of month, and will not be used in the early experimentations, which will use the general purpose "TXT" record.

Claim 30	Caronni I
	This record is designed for easy general purpose extensions in the DNS, and its content is a text string. The RX record will contain three fields: - A record identifier composed of the two characters "RX". This is used to disambiguate from other experimental uses of the "TXT" record. - A cost indicator, encoded on up to 3 numerical digits. The corresponding positive integer value should be less that 256, in order to preserve future evolutions. - An IP address, encoded as a text string following the "dot" notation. The three strings will be separated by a single comma. An example of record would thus be: domain - type - record - value *.27.32.192.in-addr.arpa - IP - TXT - RX, 10, 10.0.0.7 which means that for all hosts whose IP address starts by the three octets "192.32.27" the IP host "10.0.0.7" can be used as a gateway, and that the preference value is 10." RFC 1383 at 11-12. *See also* RFC 1383 at 9, 12-13.

Claim 35	Caronni I
[35.0] The virtual network manager of claim 30 further comprising a join module configured to receive a join request from the agent associated with the device to indicate that the device is connected for data communication within	"FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this

Claim 35	Caronni I
the virtual network, the join module further configured to receive a leave request from the agent associated with the device to indicate that the device will be disconnected from data communication within the virtual network.	step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:67–10:18. "FIG. 10 depicts a flow chart of the steps performed when logging a node out of a Supernet. The first step performed is for the user to run the SNlogout script and to enter a node ID (step 1002). Next, the SNlogout script requests a log out from SASD (step 1004). Upon receiving this request, SASD removes the mapping for this node from the VARPD that acts as the server for the Supernet (step 1006). SASD then informs KMS to cancel the registration of the node, and KMS terminates this KMC (step 1008). Lastly, KMS generates a new channel key for the channels on which the node was communicating (step 1010) to reduce the likelihood of an intruder being able to intercept traffic." Caronni I at 12:61–13:5.

Claim 35	Caronni I
	 FIG. 10 Caronni I at FIG. 10.

Claim 37	Caronni I
[37.0] The virtual network manager of claim 35 wherein the join module is further configured to maintain data to associate a virtual network address with a device in the virtual network.	"SASD 540 represents a Supernet: there is one instance of an SASD per Supernet, and it both authenticates nodes and authorizes nodes to join the Supernet. VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as a node IDs, to the network addresses recognized by the public-network infrastructure, known as the real addresses. The "node ID" may include the following: a Supernet ID (e.g., 0x123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1). Although the virtual address is described in an IP address scheme, one skilled in the art will appreciate that the virtual address may be any other type addressing scheme, such as an e-mail address, IPX, or IPv6. Since the node ID includes a Supernet ID, a node will have more than one node ID when it communicates over more than one channel. The "real address" is an IP address (e.g., 10.0.0.2) that is globally unique and meaningful to the public-network infrastructure. In a Supernet, one VARPD runs on each machine, and it may play two roles. First, a VARPD may act as a server by storing all address mappings for a particular Supernet into its associated VARPDB. Second, regardless of its role as a server or not, each VARPD assists in address translation for the nodes on its machine. In this role, the VARPD stores into its associated VARPDB the address mappings for its nodes, and if it needs a mapping that it does not have, it will contact the VARPD that acts as the server for the given Supernet to obtain it. The VARPDB may also decide which virtual address to use in the translation. That is, the VARPDB may associate a virtual address with multiple real addresses or vice versa." Caronni I at 7:3–34. "FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively,

Claim 37	Caronni I
	the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:67–10:18. "FIG. 10 depicts a flow chart of the steps performed when logging a node out of a Supernet. The first step performed is for the user to run the SNlogout script and to enter a node ID (step 1002). Next, the SNlogout script requests a log out from SASD (step 1004). Upon receiving this request, SASD removes the mapping for this node from the VARPD that acts as the server for the Supernet (step 1006). SASD then informs KMS to cancel the registration of the node, and KMS terminates this KMC (step 1008). Lastly, KMS generates a new channel key for the channels on which the node was communicating (step 1010) to reduce the likelihood of an intruder being able to intercept traffic." Caronni I at 12:61–13:5.

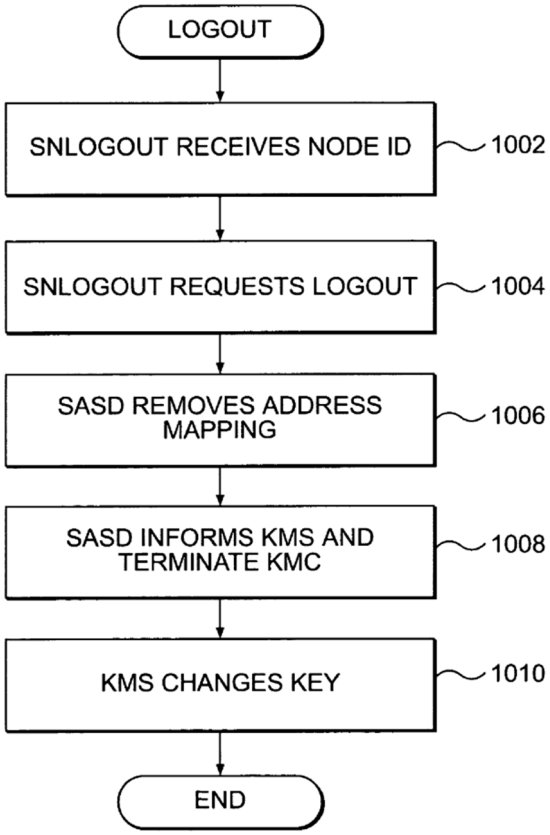
Claim 37	Caronni I
	 FIG. 10 Caronni I at FIG. 10.

Exhibit 785-3

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent No. 7,133,404 ("Alkhatib")

Alkhatib titled "Communication using two addresses for an entity" to Hasan S. Alkhatib was filed on August 10, 2001 and issued on November 7, 2006.[1] Alkhatib is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. § 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

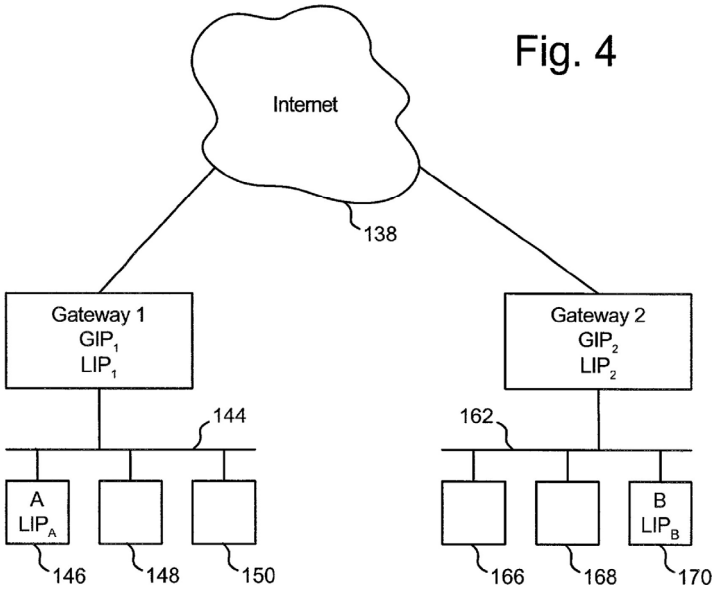
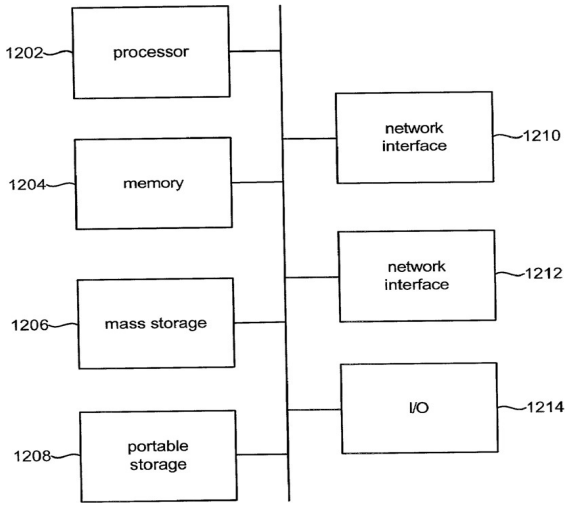
To the extent Plaintiff alleges Alkhatib does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Alkhatib or combine the teachings of Alkhatib with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

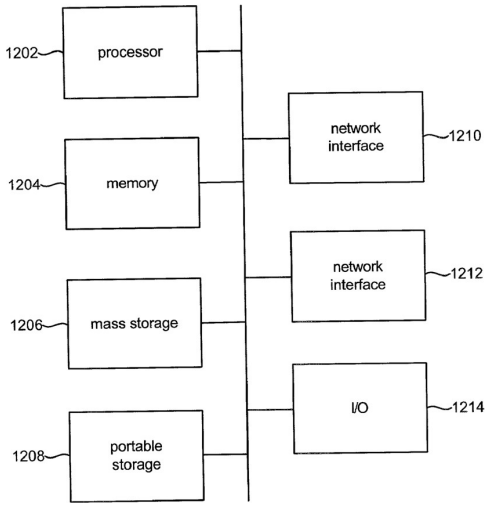
Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	**Prior Art**
[30.0] 30. A virtual network manager, comprising:	Alkhatib at 1:27–28 The present invention is directed to a system for communicating with entities on a network.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Alkhatib at Figure 4

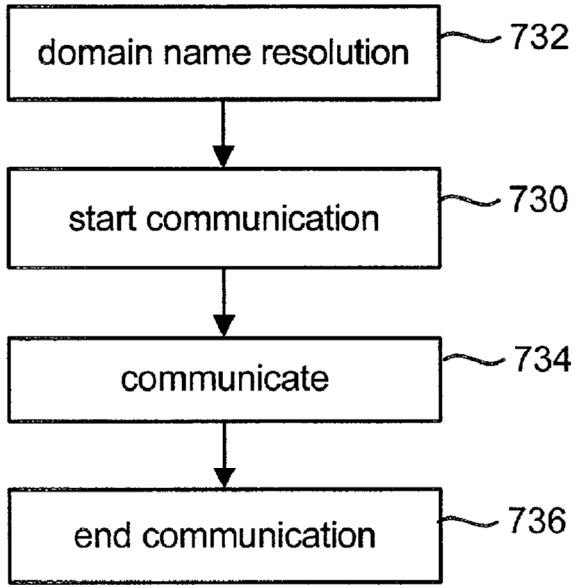
Claims	Prior Art
	Fig. 4  Alkhatib at Figure 25 Fig. 25 

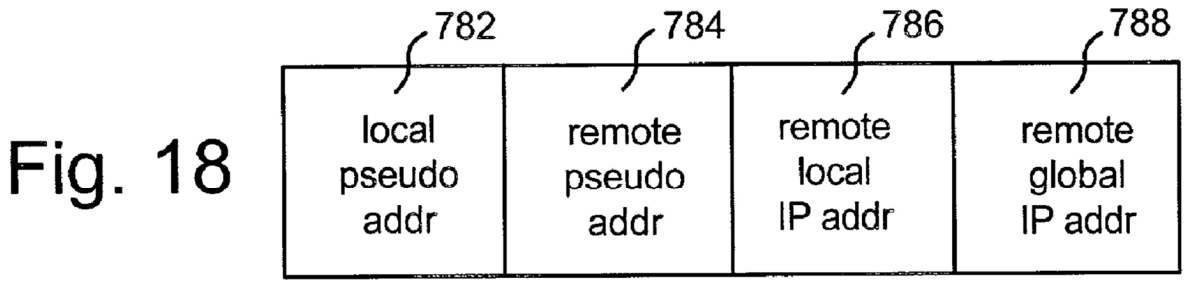
Claims	Prior Art
	Alkhatib at 9:12–14 The second line is a resource record of type A, which indicates that the global address for B.lab.dev.Bcorp.com is GIP_2 Alkhatib at 9:17–19 It is likely that there will be a DNS record, which indicates the IP address for dev.Bcorp.com. Alkhatib at 10:18–22 A must use the domain name resolution process. In prior art systems, A can provide a domain name of B to a domain name resolver and the domain name resolver will provide A with the global IP address of B Alkhatib at 11:37–39 In step 420, the domain name is resolved for a global IP address as described above with respect to FIG. 7. Alkhatib at 13:29–35 A pseudo address in its most generic form is an identification of an entity that is different from the entity's actual address. In one embodiment, the pseudo address is a random address chosen to identify a particular entity. The randomly chosen address is not the entity's actual address. In one alternative, the pseudo address is in the format for IPv4 addresses. Alkhatib at 17:36–39 If the computer is acting as a router, it includes two or more network interfaces. In other embodiments, the computer could include only one network interface. Alkhatib at 18:62–67 A method for communicating, comprising: using a domain name to obtain a first local address for a destination entity and a first global address associated with said destination entity, said destination entity is on a private network that uses said first local address to communicate with said destination entity

Claims	Prior Art
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Alkhatib at Figure 25 Fig. 25  Alkhatib at 10:66–67 Server 224 will keep a table which stores domain names and associated local addresses. Alkhatib at 11:42–43 Gateway 2 stores a table for translating domain names to local addresses. Alkhatib at 13:18–22 More information about pseudo addresses can be found in PSEUDO ADDRESSING, Bruce C. Wootton and Hasan S. Alkhatib, U.S. Application Serial No. 09/637,803, filed on Aug. 11, 2000, incorporated herein by reference. Alkhatib at 14:18–22

Claims	Prior Art
	In step 754, the network software for entity A will chose a pseudo address. In one embodiment, the pseudo address is chosen randomly. In other embodiments, a list of pseudo addresses to choose from can be pre-stored. Alkhatib at 17:26–30 The computer includes a processor 1202, a memory 1204, amass storage device 1206, a portable storage device 1208, a first network interface 1210, a second network interface 1212 and I/O devices 1214.
[30.3] receive a registration request from an agent associated with a device;	Alkhatib at 13:18–22 More information about pseudo addresses can be found in PSEUDO ADDRESSING, Bruce C. Wootton and Hasan S. Alkhatib, U.S. Application Serial No. 09/637,803, filed on Aug. 11, 2000, incorporated herein by reference. Alkhatib at 14:4–10 In step 730, the entity wishing to initiate communication performs a domain name resolution. For purposes of a first example, assume that entity A of FIG. 4 is attempting to initiate communication with entity B. In step 732, the communication is started. In step 734, the two entities communicate with each other. In step 736, the communication ends. Alkhatib at 14:18–22 In step 754, the network software for entity A will chose a pseudo address. In one embodiment, the pseudo address is chosen randomly. In other embodiments, a list of pseudo addresses to choose from can be pre-stored. Alkhatib at 20:1–3 accessing a pseudo address corresponding to said local address and said global address; and using said pseudo address with an application.

Claims	Prior Art
	To the extent that Alkhatib does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Alkhatib , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Alkhatib at 13:18–22 More information about pseudo addresses can be found in PSEUDO ADDRESSING, Bruce C. Wootton and Hasan S. Alkhatib, U.S. Application Serial No. 09/637,803, filed on Aug. 11, 2000, incorporated herein by reference. Alkhatib at 13:29–35 A pseudo address in its most generic form is an identification of an entity that is different from the entity's actual address. In one embodiment, the pseudo address is a random address chosen to identify a particular entity. The randomly chosen address is not the entity's actual address. In one alternative, the pseudo address is in the format for IPv4 addresses. Alkhatib at 13:67–14:2 Note that if application 702 asks network software 704 for its own IP address, then network 704 will respond with the pseudo address. Alkhatib at 14:4–10 In step 730, the entity wishing to initiate communication performs a domain name resolution. For purposes of a first example, assume that entity A of FIG. 4 is attempting to initiate communication with entity B. In step 732, the communication is started. In step 734, the two entities communicate with each other. In step 736, the communication ends. Alkhatib at 14:18–22 In step 754, the network software for entity A will chose a pseudo address. In one embodiment, the pseudo address is chosen randomly. In other embodiments, a list of pseudo addresses to choose from can be pre-stored.

Claims	Prior Art
	Alkhatib at 20:1–3 accessing a pseudo address corresponding to said local address and said global address; and using said pseudo address with an application. To the extent that Alkhatib does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Alkhatib , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Alkhatib at Figure 16 Fig. 16  Alkhatib at Figure 18

Claims	Prior Art
	 Fig. 18 Alkhatib at 3:41–43 The source entity resolves the destination entity's domain name into a global address and a local address. Alkhatib at 3:58–60 Another embodiment of the present invention includes using a domain name to obtain a first local address and a first global address for a destination Alkhatib at 3:60–62 A message is created that includes the first local address, the first global address and a first pseudo address. Alkhatib at 7:58–60 A gateway is an entity that connects a network to the Internet (or another network). Alkhatib at 9:20–30 The first row of data in table 5 (type of HINFO) and third row of data table FIG. 5 (type of MX) are depicted as standard DNS table entries. The second row of the table of FIG. 5 (type is A) is a modification of the standard DNS record. The standard DNS record only includes the global IP address. In one implementation of the present invention, the DNS table structure is changed (e.g. as per FIG. 5) to store both a global IP address and a local IP address. In another embodiment, resource record A only stores the global IP address and a new type of resource record is created to store the local IP address. Alkhatib at 10:22–24 In one alternative of the present invention, the resolving process will return both the global address and the local address for a given domain name. Alkhatib at 13:18–22

Claims	Prior Art
	More information about pseudo addresses can be found in PSEUDO ADDRESSING, Bruce C. Wootton and Hasan S. Alkhatib, U.S. Application Serial No. 09/637,803, filed on Aug. 11, 2000, incorporated herein by reference. Alkhatib at 14:14–18 In step 750, the application software running on entity A requests a domain name resolution. In step 752, the network software on A will initiate a process that will result in the acquisition of global address GIP_2 and a local address LIP_B for B. Alkhatib at 14:18–22 In step 754, the network software for entity A will chose a pseudo address. In one embodiment, the pseudo address is chosen randomly. In other embodiments, a list of pseudo addresses to choose from can be pre-stored. Alkhatib at 14:22–24 the pseudo address is in the same format as typical IPv4 Internet addresses. Alkhatib at 14:24–26 In step 756, the pseudo address chosen in step 754, the global address resolved in step 752 and the local address resolved in step 752 are all stored in a table.

[35.0] 35. The virtual network manager of claim 30 further comprising a join module configured to receive a join request from the agent associated with the device to indicate that the device is connected for data communication within the virtual network, the join module further configured to receive a leave request from the agent associated with the device to indicate that the device will be disconnected from data communication within the virtual network.	Alkhatib at 14:4–10 In step 730, the entity wishing to initiate communication performs a domain name resolution. For purposes of a first example, assume that entity A of FIG. 4 is attempting to initiate communication with entity B. In step 732, the communication is started. In step 734, the two entities communicate with each other. In step 736, the communication ends. Alkhatib at 14:18–22 In step 754, the network software for entity A will chose a pseudo address. In one embodiment, the pseudo address is chosen randomly. In other embodiments, a list of pseudo addresses to choose from can be pre-stored. To the extent that Alkhatib does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Alkhatib , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[37.0] 37. The virtual network manager of claim 35 wherein the join module is further configured to maintain data to associate a virtual network address with a device in the virtual network.	Alkhatib at 10:66–67 Server 224 will keep a table which stores domain names and associated local addresses. Alkhatib at 11:42–43 Gateway 2 stores a table for translating domain names to local addresses. Alkhatib at 14:4–10 In step 730, the entity wishing to initiate communication performs a domain name resolution. For purposes of a first example, assume that entity A of FIG. 4 is attempting to initiate communication with entity B. In step 732, the communication is started. In step 734, the two entities communicate with each other. In step 736, the communication ends. Alkhatib at 14:18–22 In step 754, the network software for entity A will chose a pseudo address. In one embodiment, the pseudo address is chosen randomly. In other embodiments, a list of pseudo addresses to choose from can

	be pre-stored. To the extent that Alkhatib does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Alkhatib , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-4

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent Application Publication No. 2002/0184529 ("Foster I")

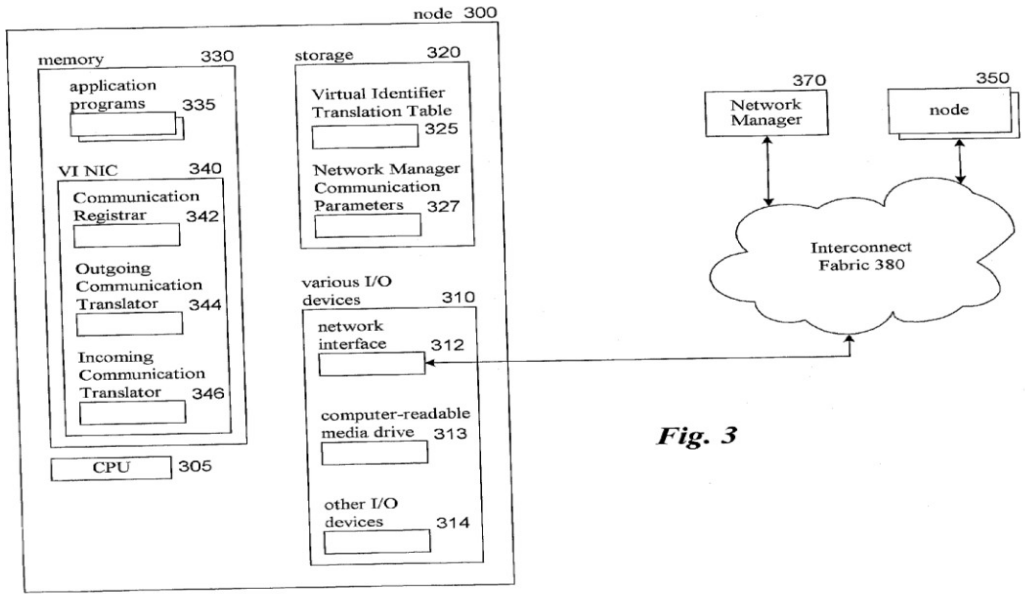
Foster titled "Communicating data through a network" to Michael S. Foster was filed on April 19, 2002 and published on December 5, 2002.[1] Foster is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. § 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Foster does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Foster or combine the teachings of Foster with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Foster at Abstract A method, system, and computer-readable medium for using virtual identifiers to route communications through a network to destinations in an appropriate manner is described. The virtual identifiers can each be assigned to one or more paths through a network to a destination, such as by a network manager for the network. Foster at Figure 3  *Fig. 3*

Claims	Prior Art

	Foster at [0019] FIG. 13 is a block diagram illustrating a distributed network manager in one embodiment. To the extent that Foster does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Foster , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Foster at Abstract A method, system, and computer-readable medium for using virtual identifiers to route communications through a network to destinations in an appropriate manner is described. The virtual identifiers can each be assigned to one or more paths through a network to a destination, such as by a network manager for the network. Foster at [0008] FIGS. 2A-2C illustrate an example of Virtual Identifier Network Interface Controller ("NIC") embodiments using virtual identifiers to inter-communicate through an example Fibre Channel Fabric network. Foster at [0036] Techniques are described below for communicating data through a network to destinations Foster at [0055] Each IFM is assigned a domain address, with the IFMs that are assigned the same domain address being in the same domain. Foster at [0077] Computing device 300 may be connected to other devices that are not illustrated, including one or more networks such as the Internet or via the World Wide Web. To the extent that Foster does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Foster , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Foster at Abstract An appropriate virtual identifier for routing a data communication can also be identified in various ways, such as by registering the data communication with a network manager for the network and receiving an appropriate virtual identifier in response. Foster at [0036] In some embodiments, each node of a network first registers with a network manager for the network (also referred to in some embodiments as a "Management Server") Foster at [0036] In response to the registering, the network manager in some embodiments provides to the node an identifier corresponding to one or more destination nodes that is to be used for routing the data communication to those destinations. Foster at [0037] In particular, when the network manager receives a request to register a data communication (also referred to in some embodiments as an "application login request") from a network node that is to be a source of the data communication, the network manager determines whether to allow the data communication, and in some situations determines one or more appropriate destination nodes for the data. An appropriate destination determination can be made in a variety of ways, such as by selecting appropriate destinations based on a type of the data communication, on information about destinations specified in the request, and/or on other accessible information about nodes, such as nodes that are interested in receiving the data communication. When the data communication is to be allowed, the network manager configures one or more intermediate network routing devices along a path between the source node and the one or more destination nodes in such a manner that the data communication will reach the appropriate destinations. In some embodiments, the network manager selects one or more virtual identifiers to be used with a registered data communication (or with multiple data communications of a specified type), configures each of the intermediate network routing devices along the path to properly handle data communications using those virtual identifiers, and then notifies the source node of the selected virtual identifiers for use with the data communication.

Claims	Prior Art
	Foster at [0047] If the VI NIC registers the data communication with a network manager, this response virtual identifier may be received from the network manager. After identifying this response virtual identifier, the VI NIC associates it with information indicating how to process received data communications that are routed using the response virtual identifier. Foster at claim 1 A method for communicating data through a network from a source node to one or more destination nodes, the network having multiple routing devices and a network manager for configuring those routing devices to forward data, each routing device able to forward data to at least one or more other routing devices based on a current configuration, the method comprising: under control of the source node, receiving data for communication to at least one destination node, registering the communication of the data with the network manager by sending a registration request and receiving in response a transmittal virtual identifier for use with the communication, the transmittal virtual identifier assigned to at least one path through the network, each of the paths from the source node to a destination node and including multiple routing devices; and forwarding the data accompanied by the transmittal virtual identifier to a first routing device included in all of the paths; under control of the network manager and in response to the registration request from the source node, configuring the network to communicate the data by determining at least one path through the network from the source node to a destination node, determining a virtual identifier available for use by all of the routing devices along each of the paths, configuring all but a last of the routing devices along each path to forward data accompanied by the determined virtual identifier to a next routing device along that path, and configuring the last routing device along each path to forward data accompanied by the determined virtual identifier to the destination node for that path; and sending the determined virtual identifier to the source node for use as the transmittal virtual identifier for the data communication; and under control of each of the routing devices along the paths, receiving forwarded data accompanied by the transmittal virtual identifier; determining to forward the received data to a next routing device along the path or to a destination node based on the configuring of the routing device for the transmittal virtual identifier; and forwarding the received data accompanied by the transmittal virtual identifier as determined,

Claims	Prior Art
	so that the destination nodes will receive the data communication sent by the source node based on the configuring of the network for the transmittal virtual identifier. To the extent that Foster does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Foster , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	Foster at Abstract An appropriate virtual identifier for routing a data communication can also be identified in various ways, such as by registering the data communication with a network manager for the network and receiving an appropriate virtual identifier in response. Foster at claim 1 A method for communicating data through a network from a source node to one or more destination nodes, the network having multiple routing devices and a network manager for configuring those routing devices to forward data, each routing device able to forward data to at least one or more other routing devices based on a current configuration, the method comprising: under control of the source node, receiving data for communication to at least one destination node, registering the communication of the data with the network manager by sending a registration request and receiving in response a transmittal virtual identifier for use with the communication, the transmittal virtual identifier assigned to at least one path through the network, each of the paths from the source node to a destination node and including multiple routing devices; and forwarding the data accompanied by the transmittal virtual identifier to a first routing device included in all of the paths; under control of the network manager and in response to the registration request from the source node, configuring the network to communicate the data by determining at least one path through the network from the source node to a destination node, determining a virtual identifier available for use by all of the routing devices along each of the paths, configuring all but a last of the routing devices along each path to forward data accompanied by the determined virtual identifier to a next routing device along that path, and configuring the last routing device along each path to forward data accompanied by the determined virtual identifier to the destination node for that path; and

Claims	Prior Art
	sending the determined virtual identifier to the source node for use as the transmittal virtual identifier for the data communication; and under control of each of the routing devices along the paths, receiving forwarded data accompanied by the transmittal virtual identifier; determining to forward the received data to a next routing device along the path or to a destination node based on the configuring of the routing device for the transmittal virtual identifier; and forwarding the received data accompanied by the transmittal virtual identifier as determined, so that the destination nodes will receive the data communication sent by the source node based on the configuring of the network for the transmittal virtual identifier. To the extent that Foster does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Foster , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Foster at Abstract An appropriate virtual identifier for routing a data communication can also be identified in various ways, such as by registering the data communication with a network manager for the network and receiving an appropriate virtual identifier in response. Foster at [0036] In response to the registering, the network manager in some embodiments provides to the node an identifier corresponding to one or more destination nodes that is to be used for routing the data communication to those destinations. Foster at [0042] In some embodiments, the VI NIC will register some or all outgoing data communications with a network manager for the network, and will receive an appropriate transmittal virtual identifier to be used for that communication from the network manager Foster at claim 1

Claims	Prior Art
	A method for communicating data through a network from a source node to one or more destination nodes, the network having multiple routing devices and a network manager for configuring those routing devices to forward data, each routing device able to forward data to at least one or more other routing devices based on a current configuration, the method comprising: under control of the source node, receiving data for communication to at least one destination node, registering the communication of the data with the network manager by sending a registration request and receiving in response a transmittal virtual identifier for use with the communication, the transmittal virtual identifier assigned to at least one path through the network, each of the paths from the source node to a destination node and including multiple routing devices; and forwarding the data accompanied by the transmittal virtual identifier to a first routing device included in all of the paths; under control of the network manager and in response to the registration request from the source node, configuring the network to communicate the data by determining at least one path through the network from the source node to a destination node, determining a virtual identifier available for use by all of the routing devices along each of the paths, configuring all but a last of the routing devices along each path to forward data accompanied by the determined virtual identifier to a next routing device along that path, and configuring the last routing device along each path to forward data accompanied by the determined virtual identifier to the destination node for that path; and sending the determined virtual identifier to the source node for use as the transmittal virtual identifier for the data communication; and under control of each of the routing devices along the paths, receiving forwarded data accompanied by the transmittal virtual identifier; determining to forward the received data to a next routing device along the path or to a destination node based on the configuring of the routing device for the transmittal virtual identifier; and forwarding the received data accompanied by the transmittal virtual identifier as determined, so that the destination nodes will receive the data communication sent by the source node based on the configuring of the network for the transmittal virtual identifier. To the extent that Foster does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Foster , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS	

Claims	Prior Art
request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Foster at [0038] When the network manager has notified the source node of one or more virtual identifiers to be used for the data communication, the source node uses those virtual identifiers as the indication corresponding to the destination nodes. In other embodiments the destination nodes can be indicated in other manners, such as with network addresses associated with the destination nodes. Foster at [0039] When an intermediate network routing device receives a transmitted data communication, the intermediate network routing device determines how to forward the data communication based on the destination indication and the manner in which the network manager has configured the intermediate network routing device. Foster at [0039] the network manager configures each intermediate network routing device along the path assigned to those virtual identifiers to forward data communications using those virtual identifiers to a next of the intermediate network routing devices along the path (or, for the last intermediate network routing device, to forward the data communication directly to a destination node). Foster at [0043] The assigning of a virtual path identifier to a path can include the configuring of each of one or more intermediate routing devices (e.g., routers or switches) between the source and the destination, such as by the network manager, so that when one of the routing devices receives a data communication that includes the virtual identifier it will forward the communication in an appropriate manner either directly to the destination or instead to a next routing device along the path that is similarly configured. Foster at [0044] In some situations, the indicated data communication may explicitly specify a destination, such as with a destination network address, while in other situations a destination may not be specified, such as when an application is publishing information and is relying on a third party to route the information to one or more current subscribers for that information. Regardless of whether a destination is specified, however,

Claims	Prior Art
	the VI NIC and/or the network manager can select one or more destinations that are appropriate for the indicated data communication, even if the specified destination is not among the selected destinations. Foster at [0047] If the VI NIC registers the data communication with a network manager, this response virtual identifier may be received from the network manager. After identifying this response virtual identifier, the VI NIC associates it with information indicating how to process received data communications that are routed using the response virtual identifier. Foster at claim 40 initiating a data communication of the registered type to the destination device by providing data to a first of the intermediate devices along the path so that the first device can forward the data in a manner based on the registered type; and To the extent that Foster does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Foster , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[35.0] 35. The virtual network manager of claim 30 further comprising a join module configured to receive a join request from the agent associated with the device to indicate that the device is	

Claims	Prior Art

connected for data communication within the virtual network, the join module further configured to receive a leave request from the agent associated with the device to indicate that the device will be disconnected from data communication within the virtual network.	Foster at [0121] A virtual identifier may be removed for various reasons, including when the network manager requests that it be removed, when a routing device or node detects an expiration of the virtual identifier, and/or when the routing device or node detects an error at the physical layer (e.g., due to changing a physical piece of hardware). The network manager may request that a virtual identifier be removed as part of a node's de-registration process. The de-registration may be initiated by the network manager or by the node itself. To the extent that Foster does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Foster , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-5

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent Application Publication No. 2003/0208602 ("Bhalla")

Bhalla titled "System and method for pushing data in an internet protocol network environment" to Rajesh Bhalla et al. was filed on April 8, 2002 and published on November 6, 2003.[1] Bhalla is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Bhalla does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Bhalla or combine the teachings of Bhalla with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Bhalla at [0001] This invention relates in general to the field of network communications and more particularly to a system and method for pushing data in an internet protocol network environment. Bhalla at [0035] The DHCP protocol represents a platform that lets an operator, such as a network administrator for example, to centrally manage (and automate where appropriate) the assignment of IP addressing configurations within a network. To the extent that Bhalla does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Bhalla , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Bhalla at [0001] This invention relates in general to the field of network communications and more particularly to a system and method for pushing data in an internet protocol network environment. Bhalla at [0030] These home domains may also represent corporate domain environments, such as Cisco.com, Intel.com, or IBM.com, for example, which may include an internal domain for end users of one or more mobile stations 12 to register. Such home domains may be private or public, secure or unsecured, and may generally provide a location for data communications, storage, or retrieval.

Claims	Prior Art
	To the extent that Bhalla does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Bhalla , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Bhalla at [0004] A query is received for the network identifier for the end user and, in response to the query, the IP address element is provided such that a querying entity may push data to the mobile station. Bhalla at [0011] FIG. 4 is a flow diagram illustrating a virtual private network (VPN) user registering in a communication system after roaming into a foreign location; Bhalla at [0033] DHCP server 50, DNS database element 54, and network registrar 56 operate to maintain a relationship in a database configuration that is constantly updated in which an NAI is associated with an IP address and/or other suitable parameters and identifies an end user of mobile station 12. Bhalla at [0038] Network registrar 56 manages IP addresses from DHCP server 50 on the basis of the NAIs of end users of one or more mobile stations 12. To the extent that Bhalla does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Bhalla , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	Bhalla at [0037]

Claims	Prior Art
	Mobile station 12 may be continuously registering with any one or more of the sub-networks within IP communications network block 22 depending on its respective location or on any other suitable triggering event. Bhalla at [0044] Home agent 70 cooperates with DHCP server 50 during mobile IP registration in order to assign an IP address to mobile station 12. To the extent that Bhalla does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Bhalla , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Bhalla at [0032] Each of the sub-networks illustrated in IP communications network block 22 comprise a dynamic host configuration protocol server (DHCP) 50, a domain name system (DNS) database element 54, and a network registrar 56 that cooperate in order to address the dynamic addressing feature described above. Bhalla at [0033] In accordance with the teachings of the present invention, one or more end users of one or more mobile stations 12 is assigned a network access identifier (NAI). Bhalla at [0036] DNS database element 54 may store any identification tool, component, object, or element that operates to correlate, recognize, or otherwise associate an end user of mobile station 12 with some unique network identity for communicating data. To the extent that Bhalla does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Bhalla , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server	

Claims	Prior Art
configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Bhalla at [0032] Each of the sub-networks illustrated in IP communications network block 22 comprise a dynamic host configuration protocol server (DHCP) 50, a domain name system (DNS) database element 54, and a network registrar 56 that cooperate in order to address the dynamic addressing feature described above. Bhalla at [0033] DHCP server 50, DNS database element 54, and network registrar 56 operate to maintain a relationship in a database configuration that is constantly updated in which an NAI is associated with an IP address and/or other suitable parameters and identifies an end user of mobile station 12. Bhalla at [0033] DNS database element 54 also allows an application server or device to direct a query to the database (in the appropriate domain) and to retrieve a specific IP address or specific end user information or parameters that are allocated to the NAI associated with an end user of mobile station 12. Bhalla at [0068] Application server 78 issues a DNS query to local DNS database 76 at step 'a'. Local DNS database 76 then forwards this DNS query to DNS database element 54 in step 'b'. At step 'c', the IP address is resolved from the user ID between DHCP server 50 and home agent 70. To the extent that Bhalla does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Bhalla , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[37.0] 37. The virtual network manager of claim 35 wherein the join module is further configured to maintain data to associate a virtual network address with a device in the virtual network.	Bhalla at [0004] A relationship is determined between the network identifier and an IP address element, the relationship being stored in a database.

Claims	Prior Art
	Bhalla at [0033] DHCP server 50, DNS database element 54, and network registrar 56 operate to maintain a relationship in a database configuration that is constantly updated in which an NAI is associated with an IP address and/or other suitable parameters and identifies an end user of mobile station 12. To the extent that Bhalla does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Bhalla , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-6

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent Application Publication No. 2002/0181395 ("Foster II")

Foster II titled "Communicating data through a network so as to ensure quality of service" to Michael S. Foster was filed on April 19, 2002 and published on December 5, 2002.[1] Foster II is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Foster II does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Foster II or combine the teachings of Foster II with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Foster II at Abstract The QOS levels may be selected based on a determined type of the data or the data communication, such as by a network manager after the data communication is registered, and QOS communication parameters are selected for the data communication to ensure that the QOS levels are achieved. Foster II at [0002] The following disclosure relates generally to computer networks, and more particularly to techniques for communicating data through networks. To the extent that Foster II does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Foster II , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Foster II at [0007] FIGS. 2A-2C illustrate an example of Virtual Identifier Network Interface Controller (“NIC”) embodiments using virtual identifiers to inter-communicate through an example Fibre Channel Fabric network. Foster II at [0082] Computing device 300 may be connected to other devices that are not illustrated, including one or more networks such as the Internet or via the World Wide Web.

Claims	Prior Art

	To the extent that Foster II does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Foster II , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Foster II at [0038] In some embodiments, each node of a network first registers with a network manager for the network, such as to identify to the network manager the node and/or one or more hardware ports of the node that are connected to the network. Before a network node communicates data to other network nodes, such as from software executing on that node (e.g., an application program), that node (or software executing on the node) registers that data communication with the network manager, such as in a dynamic manner for each data communication when it is to occur and/or by pre-registering data communications of an indicated type. In response to the registering, the network manager in some embodiments provides to the node an identifier corresponding to one or more destination nodes that is to be used for routing the data communication to those destinations. Foster II at [0087] If it is determined in step 425 that the new indicated data communication needs to be registered, the routine continues to step 430 to send a communication registration notification to the network manager that includes the determined communication type information. The routine then continues to step 435 to receive from the network manager an indication of a pair of transmittal and response virtual identifiers for the data communication, as well as optionally receiving other communication parameters to be used as part of the communication. The routine then continues to step 440 to store the received information in the virtual identifier translation table, as well as to optionally store routing information with which to route a response data communication back to the application that initiated this new data communication. To the extent that Foster II does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Foster II , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	

Claims	Prior Art
	Foster II at [0038] In some embodiments, each node of a network first registers with a network manager for the network, such as to identify to the network manager the node and/or one or more hardware ports of the node that are connected to the network. Before a network node communicates data to other network nodes, such as from software executing on that node (e.g., an application program), that node (or software executing on the node) registers that data communication with the network manager, such as in a dynamic manner for each data communication when it is to occur and/or by pre-registering data communications of an indicated type. In response to the registering, the network manager in some embodiments provides to the node an identifier corresponding to one or more destination nodes that is to be used for routing the data communication to those destinations. Foster II at [0126] When a node registers with the network manager, the network manager authenticates the node. The network manager and the node may use a PKI-based ("Public Key Infrastructure") authentication technique. For example, a node may generate a private and public key pair. The node then provides its public key to the network manager during authorization that may be coordinated by a person who is a network administrator. Once authorized, the node can register with the network manager. To register, the node encrypts its registration request (or a portion of it) using its private key and then sends the encrypted registration request to the network manager. The network manager decrypts the registration request using the node's public key. If the request is correctly decrypted, then the network manager knows it was sent by an authorized node and proceeds with the registration. If, however, the request is not correctly decrypted, then the network manager knows that the request was sent by an imposter (or otherwise unauthorized) node and disregards the registration request. To ensure that a registration request is not intercepted and decrypted by an unauthorized node that has the authorized node's public key, the network manager may generate its own private and public key pair and provide its public key to the authorized node. An authorized node can then further encrypt the registration request with the network manager's public key. In this way, only the network manager can decrypt and recognize the registration request. One skilled in the art will appreciate that these encryption techniques can also be used to protect data communication registrations and other communications sent via the network. In addition, various other authentication techniques may be used during registration. To the extent that Foster II does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Foster II , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Foster II at [0038] In some embodiments, each node of a network first registers with a network manager for the network, such as to identify to the network manager the node and/or one or more hardware ports of the node that are connected to the network. Before a network node communicates data to other network nodes, such as from software executing on that node (e.g., an application program), that node (or software executing on the node) registers that data communication with the network manager, such as in a dynamic manner for each data communication when it is to occur and/or by pre-registering data communications of an indicated type. In response to the registering, the network manager in some embodiments provides to the node an identifier corresponding to one or more destination nodes that is to be used for routing the data communication to those destinations. Foster II at [0039] In some embodiments, the network manager selects one or more virtual identifiers to be used with a registered data communication (or with multiple data communications of a specified type), configures each of the intermediate network routing devices along the path to properly handle data communications using those virtual identifiers, and then notifies the source node of the selected virtual identifiers for use with the data communication. Foster II at [0086] In particular, the routine compares whenever information is used to uniquely identify the destination and/or the manner of transmission for the indicated new data communication, and determines if there is a match for that information already present in the virtual identifier translation table. Those skilled in the art will appreciate that in other embodiments the routine may not make this determination and instead send registration information to the network manager for each new indicated data communication, such as if the network manager provides functionality to determine whether to reuse previously provided transmission information or to instead send new transmission information. Foster II at [0126] When a node registers with the network manager, the network manager authenticates the node. The network manager and the node may use a PKI-based (“Public Key Infrastructure”) authentication

Claims	Prior Art
	technique. For example, a node may generate a private and public key pair. The node then provides its public key to the network manager during authorization that may be coordinated by a person who is a network administrator. Once authorized, the node can register with the network manager. To register, the node encrypts its registration request (or a portion of it) using its private key and then sends the encrypted registration request to the network manager. The network manager decrypts the registration request using the node's public key. If the request is correctly decrypted, then the network manager knows it was sent by an authorized node and proceeds with the registration. If, however, the request is not correctly decrypted, then the network manager knows that the request was sent by an imposter (or otherwise unauthorized) node and disregards the registration request. To ensure that a registration request is not intercepted and decrypted by an unauthorized node that has the authorized node's public key, the network manager may generate its own private and public key pair and provide its public key to the authorized node. An authorized node can then further encrypt the registration request with the network manager's public key. In this way, only the network manager can decrypt and recognize the registration request. One skilled in the art will appreciate that these encryption techniques can also be used to protect data communication registrations and other communications sent via the network. In addition, various other authentication techniques may be used during registration. To the extent that Foster II does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Foster II , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Foster II at Figure 18

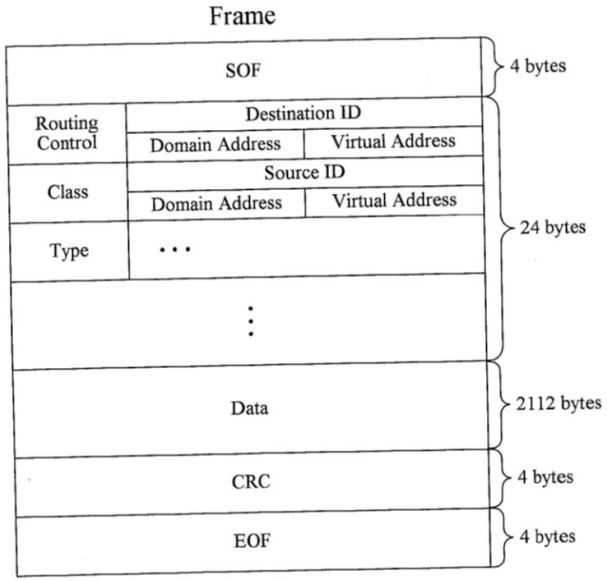
Claims	Prior Art
	 ***Fig. 18*** Foster II at [0005] In addition, it can be difficult for even an authorized source node to transmit data to desired destinations, as the source node must know the appropriate network address or other logical name (e.g., a DNS name) that is assigned or mapped to a destination node in order to perform the transmitting. Foster II at [0060] In one embodiment, a virtual identifier includes both a domain address and a virtual address. To the extent that Foster II does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Foster II , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-7

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent Application Publication No. 2002/0091859 ("Tuomenoksa")

Tuomenoksa titled "Methods and systems for partners in virtual networks" to Mark Tuomenoksa et al. was filed on April 11, 2001 and published on July 11, 2002.[1] Tuomenoksa is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Tuomenoksa does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Tuomenoksa or combine the teachings of Tuomenoksa with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

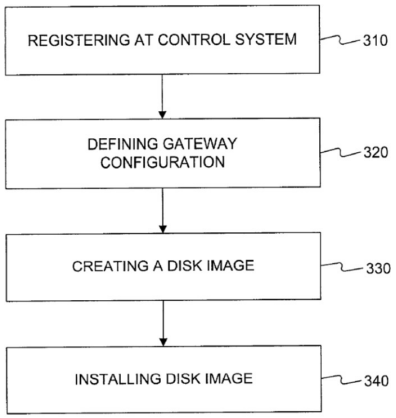
Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

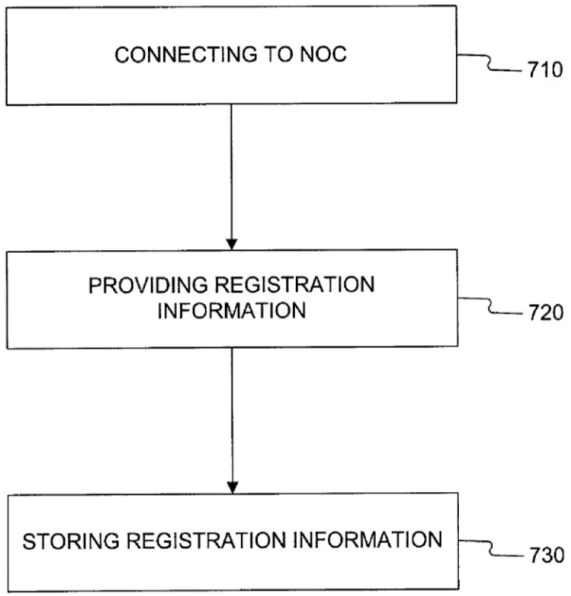
[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Tuomenoksa at [0003] The present invention relates to systems and methods for controlling networks, and in particular, to systems and methods for implementing virtual private networks. Tuomenoksa at [0029] The at least one processor may also monitor each virtual network and alert the user in a customized fashion when events occur in the virtual network. The at least one processor may also monitor quality-of-service (QoS) statistics within the virtual networks, such as the availability, bandwidth, throughput, and latency for each tunnel established through the base network. The at least one processor may further monitor quality-of-service statistics for a network service provider, such as the availability, bandwidth, throughput, and latency for the first and second processors. Tuomenoksa at [0048] FIG. 11C illustrates a flow chart of a method for initially establishing a virtual network, in accordance with methods and systems consistent with the invention Tuomenoksa at [0094] In accordance with an embodiment of the present invention, a prospective user or customer may contact a mediation point or a control system, such as a network operations center via a base network, such as the Internet, and indicate a desire to establish one or more virtual private networks. To the extent that Tuomenoksa does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Tuomenoksa , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art

[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Tuomenoksa at [0008] A virtual private network is "virtual" because it uses a shared or a base network, such as the Internet as its backbone as opposed to a completely private network with dedicated lines. Tuomenoksa at [0018] To address the above and other limitations of the prior art, methods and systems are provided that easily and effectively leverage the power of a shared or a base network, such as the Internet for private connectivity without the complexity, cost, or time associated with setting up traditional virtual private networks. Tuomenoksa at [0119] The output module 230 may include a display 235, a printer 236, and a network interface 238. Tuomenoksa at [0126] This specified configuration information may include a name for the gateway and a name for the virtual private network. This name for the virtual private network will hereinafter be referred to as the virtual private network's domain name. Tuomenoksa at [0266] Referring to Table 1, the network operations center 610 may provide each gateway enabled by the network operations center with one or more of the following: a gateway name, a domain name for the virtual private network, a virtual Internet Protocol (IP) address, and a public IP address visible to the Internet 620. Tuomenoksa at [0331] The gateway 650 may determine the public IP address for the tunnel interface 612 by referring to a routing table in the gateway 650. Alternatively, the gateway 650 may use an Internet/Intranet access device and/or a Dynamic Host Configuration Protocol (DHCP) server. The gateway 650 may also use a domain name server to resolve the real IP address of the tunnel interface driver 612.

Claims	Prior Art
	To the extent that Tuomenoksa does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Tuomenoksa , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Tuomenoksa at Figure 3  FIG. 3 Tuomenoksa at Figure 7

Claims	Prior Art
	 Tuomenoksa at [0019] The additional processor may receive information indicating consent on behalf of the first processor to enabling a tunnel between the first processor and the second processor and information indicating consent on behalf of the second processor to enabling a tunnel between the second processor and the first processor Tuomenoksa at [0027] The additional processor may determine a first virtual address that identifies the first processor in the virtual network and provide the first virtual address to the first processor. When a tunnel between the first processor and the second processor is requested from the additional processor, the additional processor may authenticate the request based on the first virtual address and determine a second virtual address that identifies the second processor in the virtual network. After the additional processor authenticates the request and determines that the first and second processors have indicated a mutual consent for enabling one or more tunnels between the first and second processors, the additional processor may provide the second virtual address to the first processor to enable the requested tunnel between the first and second processors. Tuomenoksa at [0034]

Claims	Prior Art
	FIG. 3 Tuomenoksa at [0042] FIG. 9B Tuomenoksa at [0078] FIG. 34 Tuomenoksa at [0119] The host computers 154 and 155 may each include a processor, such as a computer 200 shown in FIG. 2. The computer 200 may include an input module 205, a central processing unit (CPU) 220, a storage module 250, and an output module 230. Tuomenoksa at [0121] The CPU 220 may also include, for example, one or more of the following: a co-processor, memory, registers, and other processing devices and systems as appropriate. Tuomenoksa at [0124] In one embodiment, a user may serve as an administrator and may register at least one of the gateways 150-153 through control system 175 and/or establish one or more virtual private networks over communication channel 120. Tuomenoksa at [0126] FIG. 3 To the extent that Tuomenoksa does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Tuomenoksa , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	Tuomenoksa at [0027] The additional processor may determine a first virtual address that identifies the first processor in the virtual network and provide the first virtual address to the first processor. When a tunnel between the first

Claims	Prior Art
	processor and the second processor is requested from the additional processor, the additional processor may authenticate the request based on the first virtual address and determine a second virtual address that identifies the second processor in the virtual network. After the additional processor authenticates the request and determines that the first and second processors have indicated a mutual consent for enabling one or more tunnels between the first and second processors, the additional processor may provide the second virtual address to the first processor to enable the requested tunnel between the first and second processors. To the extent that Tuomenoksa does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Tuomenoksa , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Tuomenoksa at Abstract The additional processor may determine a first virtual address for the first processor and a second virtual address for the second processor such that the first and second virtual addresses uniquely identify the first and second processors, respectively, and are routable through the network. Tuomenoksa at Abstract The additional processor may provide to each of the first and second processors the first and second virtual addresses to enable one or more tunnels between the first and the second processors. Tuomenoksa at [0019] The additional processor may determine a first virtual address for the first processor and a second virtual address for the second processor such that the first and second virtual addresses uniquely identify the first and second processors, respectively, and are routable through the network. Tuomenoksa at [0027] The additional processor may determine a first virtual address that identifies the first processor in the virtual network and provide the first virtual address to the first processor. When a tunnel between the first processor and the second processor is requested from the additional processor, the additional processor may authenticate the request based on the first virtual address and determine a second virtual address that

Claims	Prior Art
	identifies the second processor in the virtual network. After the additional processor authenticates the request and determines that the first and second processors have indicated a mutual consent for enabling one or more tunnels between the first and second processors, the additional processor may provide the second virtual address to the first processor to enable the requested tunnel between the first and second processors. Tuomenoksa at [0124] In one embodiment, a user may serve as an administrator and may register at least one of the gateways 150-153 through control system 175 and/or establish one or more virtual private networks over communication channel 120. Tuomenoksa at claim 9 means for determining a first virtual address for the first processor and a second virtual address for the second processor such that the first and second virtual addresses uniquely identify the first and second processors, respectively, and are routable through the network; To the extent that Tuomenoksa does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Tuomenoksa , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Tuomenoksa at [0022] In one embodiment, the at least one additional processor may determine a first virtual address and a first base address for the first processor such that the first virtual address is routable through the virtual network and the first base address is routable through a base network and determine a second virtual address and a second base address for the second processor such that the second virtual address is routable through the virtual network and the second base address is routable through the base network. Tuomenoksa at [0023] Further, methods and systems consistent with yet another aspect of the present invention may enable one or more networks between a first processor and a second processor using at least one additional processor

Claims	Prior Art
	separate from the first and second processors, the first processor and the second processor each identifiable by a name and each independently administered through the additional processor. Tuomenoksa at [0140] Consequently, each gateway may be provided two addresses—a real or public address and a virtual address. Tuomenoksa at [0140] This virtualized addressing may provide virtual connectivity through the Internet 620 and may allow routing of virtual addresses from one address to another. Moreover, this virtualized addressing may facilitate network address translation, port address translation, IP masquerade, and/or IP connection sharing during the process of routing as well as during the dynamic assignment of addresses. Although a virtual address may be used by a gateway to establish one or more tunnels to form a virtual network and/or virtual private network, the network operations center 610 may alternatively provide to each gateway any other address that is capable of enabling any other networks established through or over a base network, such as the Internet 620. Tuomenoksa at [0243] Furthermore, ICMP ping, traceroute traffic, and Domain Name Server (DNS) response traffic (i.e., UDP traffic including responses to a DNS request that originates from a processor on the client side) may also be allowed to reach the client side from the tunnel side. Finally, all other traffic originating from any other source on the tunnel side may be blocked. Tuomenoksa at [0331] The gateway 650 may determine the public IP address for the tunnel interface 612 by referring to a routing table in the gateway 650. Alternatively, the gateway 650 may use an Internet/Intranet access device and/or a Dynamic Host Configuration Protocol (DHCP) server. The gateway 650 may also use a domain name server to resolve the real IP address of the tunnel interface driver 612. Tuomenoksa at [0340] The VPN acknowledgement 10920 may include: the virtual IP address of the gateway 650; the virtual IP address of the network operations center 610; the shared secret of the gateway 650; the public key for the gateway 650; the public key for the network operations center 610; version information of the program code currently used by the gateway 650; and information for establishing an IPSec tunnel consistent with the IPSec standard.

Claims	Prior Art
	To the extent that Tuomenoksa does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Tuomenoksa , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[37.0] 37. The virtual network manager of claim 35 wherein the join module is further configured to maintain data to associate a virtual network address with a device in the virtual network.	Tuomenoksa at [0144] The network operations center 610 may then store the virtual IP address of the gateway with the gateway's name and the authentication information, such as a shared secret for that gateway. Tuomenoksa at [0197] For example, the network operations center 610 may store a partner list for each gateway in a database within the database server 616. This database may store each gateway's name with a corresponding partner list that includes each partner's virtual IP address, public portion of the public key, firewall information, and other stored information. To the extent that Tuomenoksa does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Tuomenoksa , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-8

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent Application Publication No. 2004/0177136 ("Chen")

Chen titled "Method and system for managing a device within a private network using a management device external to the private network" to Weiwen Chen et al. was filed on March 3, 2003 and published on September 9, 2004.[1] Chen is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Chen does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Chen or combine the teachings of Chen with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

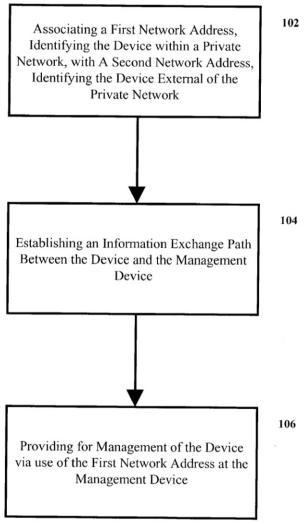
As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Chen at Abstract A method and system are described for managing a device within a private network using a management device external to the private network. To the extent that Chen does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Chen , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Chen at [0020] The IP network 314 can be the Internet. The router 312 can be assigned a globally unique IP address (152.2.117.129). Chen at [0020] Although the management device 310 is shown in FIG. 3 as included in a private network portion 306 of the managed network (e.g., having the assigned private IP address 10.1.1.4), the concepts described herein can be equally applied to a centralized management device operating in a publicly addressable portion of the managed network. To the extent that Chen does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Chen , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module	

Claims	Prior Art

configured to register devices in a virtual network, the register module further configured to:	Chen at Figure 1  *FIG. 1* Chen at [0018] Referring to the router 308 in the first private network 302, a first network address (10.1.1.9) from the private address space can be assigned to an interface (e.g., a device) of the router 308. This first network address uniquely identifies the interface within the private network 302. That is, no other devices (or interfaces) are assigned this address with the private network portion 302. Chen at [0023] The second network address associated with the first network address can be determined by referencing a table stored in memory. Chen at [0039] Moreover, the exemplary embodiments can be considered part of any form of computer readable storage medium having stored therein an appropriate set of computer instructions that would cause a processor to carry out the techniques described herein.

Claims	Prior Art
	Chen at [0044] The steps of a computer program as illustrated in FIG. 1 for managing a device within a private network using a management device external to the private network can be embodied in any computer readable medium for use by or in connection with an instruction execution system, apparatus, or device, such as a computer based system, processor containing system, or other system that can fetch the instructions from the instruction execution system, apparatus, or device and execute the instructions. To the extent that Chen does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Chen , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	Chen at [0018] Referring to the router 308 in the first private network 302, a first network address (10.1.1.9) from the private address space can be assigned to an interface (e.g., a device) of the router 308. This first network address uniquely identifies the interface within the private network 302. That is, no other devices (or interfaces) are assigned this address with the private network portion 302. Chen at [0027] When an SNMPv1 trap is generated by an agent operating on the device 308 in the private network 302, the payload will include the agent-address record corresponding to device's private (or first) network address. Accordingly, the first network address can be extracted directly from the payload portion of the trap along with the device information (e.g., event information). To the extent that Chen does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Chen , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the	

Claims	Prior Art
virtual network by the virtual network address; and	Chen at [0005] According to exemplary embodiments, a first network address, used to uniquely identify the device within the private network, is associated with a second network address used to uniquely identify the device externally of the private network. Chen at [0005] Management of the device is provided for via use of the first network address at the management device. Chen at [0018] Referring to the router 308 in the first private network 302, a first network address (10.1.1.9) from the private address space can be assigned to an interface (e.g., a device) of the router 308. This first network address uniquely identifies the interface within the private network 302. That is, no other devices (or interfaces) are assigned this address with the private network portion 302. Chen at [0023] Instead, the management device 310 can use the second network address (180.75.6.9), associated with the first network address, to uniquely identify the device 308 external of the private network 302. To the extent that Chen does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Chen , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Chen at [0022] If a name-serving device, having software such as Domain Name System (DNS), is configured to map the first network address to a meaningful hostname, then management of the device can be provided via the hostname. Chen at [0030] The routing device is preferably a device 308 arranged within the private network 302 at the boundary between the private and public address space. The routing device can however be physically arranged

Claims	Prior Art
	outside the private network 302 and connected to devices within the private network 302 using firewalls and other suitable security arrangements. To the extent that Chen does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Chen , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-9

Invalidity of U.S. Patent No. 7,949,785 in View of European Patent Publication No. EP1098496 ("Engmann")

Engmann titled "Reversed masking for the acessibility to data nodes in private IPv4 networks" to Steffen Engmann was published on October 19, 2000.[1] Engmann is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. § 102(b) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Engmann does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Engmann or combine the teachings of Engmann with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Engmann at claim 1 Method for connecting devices (CH) of a first and a second packet-oriented communication network, the communication networks being connected to one another via at least one interface (AR), characterized in that a connection-searching device (CH) of the first communication network establishes a connection to an internal device (PH; MH) of the second communication network. To the extent that Engmann does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Engmann , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Engmann at [0009] Private networks can be used with public IP networks, such as the Internet, connected via an access node that then a public address of the official IPv4 address space must receive. Engmann at [0037] Step 9: IP packets created by a private host PH public IP address will always be sent routed via the access node AR (see FIG. 9, arrow c). Engmann at [0047] However, the present concept is not just about communication within a packet-oriented internet or the like applicable. Communication networks, like future cellular ones Mobile networks are also based on IP technology or enable their use.

Claims	Prior Art

	Engmann at claim 1 Method for connecting devices (CH) of a first and a second packet-oriented communication network, the communication networks being connected to one another via at least one interface (AR), characterized in that a connection-searching device (CH) of the first communication network establishes a connection to an internal device (PH; MH) of the second communication network. To the extent that Engmann does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Engmann , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Engmann at [0016] This can be done in particular by the responsible name server both addresses are registered with the access node and preferably only be stored temporarily. Engmann at [0020] A security table enables the release of access to protect data from unauthorized access only for registered external stations. Engmann at [0040] Step 1, which describes the registration of private host names PH, is required in the exemplary procedure the registration procedure is known in itself. Engmann at [0052] To ensure the data security of, for example, one Intranet can be in any suitable for this procedure Router a table is provided in which all external Hosts are registered that connect to a host set up within the network controlled by the access node allowed to. To the extent that Engmann does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Engmann , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art
[30.3] receive a registration request from an agent associated with a device;	Engmann at [0040] Step 1, which describes the registration of private host names PH, is required in the exemplary procedure the registration procedure is known in itself. Engmann at [0046] The case is then the database of the access node AR with the update the actual, currently dynamic IP address every time when a host dials into the private network and assigned a dynamic address. Engmann at [0052] To ensure the data security of, for example, one Intranet can be in any suitable for this procedure Router a table is provided in which all external Hosts are registered that connect to a host set up within the network controlled by the access node allowed to. To the extent that Engmann does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Engmann , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Engmann at [0029] Each host PH in the private network receives one unique host name and a unique IPv4 address from one private IPv4 address space. The operator of the private network leaves the host names for all accessible private hosts PH in register the DNS name server NS. For each host PH will the host name in addition to the IP address of the access node AR inserted into a table tab so that for every setting of a host name for a private host PH an IP address of the access node AR is assigned, which is automatically output can be. In addition, behind each hostname added an entry to announce if a message or notification to the access node AR at a Hostname setting to be sent. This process is in connection with the example shown in FIG easier to understand.

Claims	Prior Art
	Engmann at [0040] Step 1, which describes the registration of private host names PH, is required in the exemplary procedure the registration procedure is known in itself. To the extent that Engmann does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Engmann , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Engmann at [0023] This function is called Domain Name Service (DNS) designated. Engmann at [0024] If a host Ha sends IP packets to another host Hb would like to ask the responsible DNS server or DNS name server NS for the appropriate IP address for the one you want Hostnames as shown in Fig.4. Engmann at [0025] Because of this naming procedure and because of the Fact that hosts Ha changed the name of the host Hb into an IP address have to resolve before they can send IP packets, there is the possibility of the functionality of a DNS name server NS to improve a message to the access node AR, e.g. an access router to send the is responsible for the requested lost Hb. Engmann at [0027] An overview of the method is shown in FIG. 5. In in a first step a corresponding host CH asks which is outside the private network, according to the IP address of the private host PH within the private network through the Inquiries to the name server NS with the hostname of the private Hosts PH. In response, it receives the IP address of the access node AR of the private host PH. The name server NS notifies the access node AR that the corresponding Host CH asked for the private host PH. The corresponding one Host CH sends its IP packets to the address of the access node AR and the access node AR now knows that these packets are intended for the private host PH. The access node AR then exchanges the IP packets

Claims	Prior Art
	between the private and the corresponding host PH, CH by modifying the source and target address fields (src_adresse or dst_adresse) of the IP packets. Between the access node AR and the name server NS become one ns_Notification (CH, PH) or ns-notify (CH, PH) and one ns_confirmation (CH, PH) or ns_ack (CH, PH) exchanged. Engmann at [0029] Each host PH in the private network receives one unique host name and a unique IPv4 address from one private IPv4 address space. The operator of the private network leaves the host names for all accessible private hosts PH in register the DNS name server NS. For each host PH will the host name in addition to the IP address of the access node AR inserted into a table tab so that for every setting of a host name for a private host PH an IP address of the access node AR is assigned, which is automatically output can be. In addition, behind each hostname added an entry to announce if a message or notification to the access node AR at a Hostname setting to be sent. This process is in connection with the example shown in FIG easier to understand. Engmann at [0031] Step 3: After a corresponding external host CH a resolution or definition question (dns question) was sent the name server NS sends a notification to the Access node AR (Fig. 7, arrow b). This notification message contains the IP address of the corresponding external Hosts CH and the hostname of the requested private Hosts PH. Engmann at [0033] Step 5: After receiving the confirmation, the name server replies NS to the resolution question by sending one DNS response that contains the IP address of the access node AR Engmann at [0034] After receiving the notification saves the access node AR this routing information in a database provided for it for a reverse Masking. An example structure for such a database is shown in Fig. 8, in a first column the IP address of the corresponding host CH, in a second column the host name of the private host PH and in one third column is the private address of the private host PH can be saved. Engmann at [0053] The Internet host CH with the fixed public IP address 129.13.130.13 would like to connect to a mobile host MH or a mobile station. He only knows their host names (here 1234567.Provider.com), so that he has to ask the associated name server NS for the address definition. The name server NS sends him the

Claims	Prior Art
	IP address of the access node AR (or UMSC) 132.23.1.254, which is responsible for the mobile host MH. To the extent that Engmann does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Engmann , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[35.0] 35. The virtual network manager of claim 30 further comprising a join module configured to receive a join request from the agent associated with the device to indicate that the device is connected for data communication within the virtual network, the join module further configured to receive a leave request from the agent associated with the device to indicate that the device will be disconnected from data communication within the virtual network.	Engmann at [0016] This can be done in particular by the responsible name server both addresses are registered with the access node and preferably only be stored temporarily. Engmann at [0020] A security table enables the release of access to protect data from unauthorized access only for registered external stations. Engmann at [0052] To ensure the data security of, for example, one Intranet can be in any suitable for this procedure Router a table is provided in which all external Hosts are registered that connect to a host set up within the network controlled by the access node allowed to. To the extent that Engmann does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Engmann , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art

[37.0] 37. The virtual network manager of claim 35 wherein the join module is further configured to maintain data to associate a virtual network address with a device in the virtual network.	Engmann at [0016] This can be done in particular by the responsible name server both addresses are registered with the access node and preferably only be stored temporarily. Engmann at [0019] In the name server is to the known name of the requested private hosts each exactly one Address of the access node of the second communication network registered or determined by this. Engmann at [0020] A security table enables the release of access to protect data from unauthorized access only for registered external stations. Engmann at [0052] To ensure the data security of, for example, one Intranet can be in any suitable for this procedure Router a table is provided in which all external Hosts are registered that connect to a host set up within the network controlled by the access node allowed to. Engmann at [0053] The access node AR stores this information in its routing database in accordance with the example shown in FIG. 8. To the extent that Engmann does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Engmann , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-10

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent Application Publication No. 2002/0069278 ("Forslow")

Forslow titled "Network-based mobile workgroup system" to Jan Forslow was filed on December 5, 2000 and published on June 6, 2002.[1] Forslow is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Forslow does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Forslow or combine the teachings of Forslow with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Forslöw at Abstract The mobile workgroup system is an access management system for mobile users with VPN and firewall functionality inbuilt. To the extent that Forslöw does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Forslöw , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Forslöw at [0001] The present invention relates to data communications in general, and more specifically, the present invention describes a network-based mobile workgroup system that provides secure communication to and within an overlaid workgroup network while applying mobility management for the workgroup members. To the extent that Forslöw does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Forslöw , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Forslöw at [0085]

Claims	Prior Art

	While individual functional blocks and components are shown in many of the figures, those skilled in the art will appreciate these functions may be performed by software modules or programs executed using a suitably programmed digital microprocessor or general purpose computer by individual hardware circuits, by an application specific integrated circuit (ASIC), and/or by one or more digital signaling processors (DSPs). Forslöw at [0110] The domain name system (DNS) 124 is responsible to keep up to date resource records (RRs) for workgroup members (20 and 16) and workgroup resources (28). Forslöw at [0127] The registration portion of the extranet flowchart starts with the mobile client 20 registering with the foreign agent 31 on the insecure AccessNet 84 of Site 2. The foreign agent 31 forwards the registration request across the MVPN tunnel 100 to the home agent 30 for user authentication. If the user authentication is successful, the home agent 30 updates the resource record in the dynamic DNS 124a as well as the dynamic workgroup filter 122b with the allocated mobile client 20 home IP address. The registration portion of the flowchart ends when the home agent 30 sends a registration reply message to the mobile client 20 via the foreign agent 31. Forslöw at [0135] At configuration time, the mobile client 20 registers with the mobile service manager 22 in order to become part of a workgroup To the extent that Forslöw does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Forslöw , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	Forslöw at [0027] Mobile IP is defining a home agent as the anchor point with which the mobile client always has a relationship, and a foreign agent, which acts as the local tunnel-endpoint at the access network where the mobile client is visiting.

Claims	Prior Art
	Forslöw at [0127] The registration portion of the extranet flowchart starts with the mobile client 20 registering with the foreign agent 31 on the insecure AccessNet 84 of Site 2. The foreign agent 31 forwards the registration request across the MVPN tunnel 100 to the home agent 30 for user authentication. If the user authentication is successful, the home agent 30 updates the resource record in the dynamic DNS 124a as well as the dynamic workgroup filter 122b with the allocated mobile client 20 home IP address. The registration portion of the flowchart ends when the home agent 30 sends a registration reply message to the mobile client 20 via the foreign agent 31. Forslöw at [0135] At configuration time, the mobile client 20 registers with the mobile service manager 22 in order to become part of a workgroup To the extent that Forslöw does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Forslöw , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Forslöw at Abstract The mobile user can access the mobile workgroup system over a set of access technologies and select server resources and correspondent nodes to access pending their workgroup membership approvals. Forslöw at [0110] The domain name system (DNS) 124 is responsible to keep up to date resource records (RRs) for workgroup members (20 and 16) and workgroup resources (28). Forslöw at [0127] The registration portion of the extranet flowchart starts with the mobile client 20 registering with the foreign agent 31 on the insecure AccessNet 84 of Site 2. The foreign agent 31 forwards the registration request across the MVPN tunnel 100 to the home agent 30 for user authentication. If the user authentication is successful, the home agent 30 updates the resource record in the dynamic DNS 124a as

Claims	Prior Art
	well as the dynamic workgroup filter 122b with the allocated mobile client 20 home IP address. The registration portion of the flowchart ends when the home agent 30 sends a registration reply message to the mobile client 20 via the foreign agent 31. Forslöw at [0131] The configuration server 146 maintains an IP address pool from which it can allocate addresses to individual mobile clients 20. The configuration server 146 may configure the mobile client 20 with other options such as default DNS, web server, etc. The configuration server 146 updates the directory agent 147 of a new or changed IP address. Forslöw at [0136] If the mobile client 20 is on its home network, the first registration involves a host configuration, in order to provide the mobile client 20 with an IP address. Forslöw at claim 3 a network address identifier (NAI) with which a user of a mobile client is uniquely identified in the mobile VPN system. Forslöw at claim 5 5. The mobile workgroup system of claim 4, wherein the IP address allocated to the mobile client is tied to the network address identifier of the mobile user. To the extent that Forslöw does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Forslöw , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network	Forslöw at [0007] DHCP, Dynamic Host Configuration Protocol. DHCP is an Internet Engineering Task Force (IETF) standard for allocating Internet Protocol addresses to User Systems. User Systems can either be Fixed Hosts or Mobile Systems. The allocation is done each time when the User System is started. The

Claims	Prior Art
address associated with the second device.	allocation is made by a DHCP server to a DHCP client. The DHCP server is controlled by an Internet Service Provider or an IT-department. The DHCP client is a SW embedded in the User System. Forslöw at [0013] The ISP is a notation for the domain providing basic IP configuration services to users, i.e. servers for Domain Name System (DNS) and Dynamic Host Configuration Protocol (DHCP). Forslöw at [0029] This mobility addressing dilemma is handled in mobile IP by allowing the mobile client to be associated with two IP addresses: a static, “home” address and a dynamic, “care-of” address that changes at each new point of attachment to the Internet. Forslöw at [0031] The further delivery by the home agent to the foreign agent requires that each packet intended for the mobile client be modified/extended so that the care-of address appears as the destination IP address. This modification of the packet is sometimes termed a “redirection.” The home agent redirects packets from the home network to the care-of address by constructing a new IP header that contains the mobile client's care-of address as the packet's destination IP address. This new header “encapsulates” the original data packet causing the mobile client's home address to have no effect on the encapsulated packet's routing until it arrives at the care-of address. This encapsulation is commonly known as “tunneling” in the sense that the data packet burrows or tunnels using the new “routing” header through the Internet, while the encapsulated IP header is completely ignored. When the packet arrives at the foreign agent the new “routing” header is removed and the original packet is sent to the mobile client for properly processing by whatever higher level protocol (layer 4) that logically receives it from the mobile client's IP (layer 3) processing layer. Forslöw at [0110] The domain name system (DNS) 124 is responsible to keep up to date resource records (RRs) for workgroup members (20 and 16) and workgroup resources (28). Forslöw at [0128] The client-server session portion of the flowchart starts with the mobile client 20 making a DNS query for the server resource 28 to the DNS 124a across the MVPN tunnel 100. Forslöw at [0131]

Claims	Prior Art
	The directory agent 147 is a location server that maintains a mapping between names (including workgroup members 20 & 16 and resources 28) and IP addresses given by the configuration server 146. The directory agent 147 acts as the end-server in the DNS 124 hierarchy for the workgroups 26. Forslöw at [0136] The mobility manager 148 updates the routing table in the communication controller 149 and the mobile client 20 resource record in the directory agent 147 in order to allow peers, such as the correspondent node 16, to find and initiate sessions to the mobile client 20. As an example of the latter, the correspondent node 16 queries the directory agent 147 for the IP address of the mobile client 20 name. The correspondent node 16 then sends packets towards the mobile client 20. These packets are filtered and routed by the communication controller 149 (to be more correct the forwarding components controlled by the communication controller). As the communication controller 149 has been updated with the workgroup filter to allow the correspondent node 16 to send packets to the workgroup and with the IP address that the mobile client 20 is currently allocated, the communication controller 149 lets the traffic through to the mobile client 20. Forslöw at [0138] The mobile client 20 then sends DHCP inform requests to get information such as default DNS, default Web server, etc. To the extent that Forslöw does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Forslöw , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art

	The registration portion of the extranet flowchart starts with the mobile client 20 registering with the foreign agent 31 on the insecure AccessNet 84 of Site 2. The foreign agent 31 forwards the registration request across the MVPN tunnel 100 to the home agent 30 for user authentication. If the user authentication is successful, the home agent 30 updates the resource record in the dynamic DNS 124a as well as the dynamic workgroup filter 122b with the allocated mobile client 20 home IP address. The registration portion of the flowchart ends when the home agent 30 sends a registration reply message to the mobile client 20 via the foreign agent 31. Forslöw at [0135] At configuration time, the mobile client 20 registers with the mobile service manager 22 in order to become part of a workgroup To the extent that Forslöw does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Forslöw , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[35.0] 35. The virtual network manager of claim 30 further comprising a join module configured to receive a join request from the agent associated with the device to indicate that the device is connected for data communication within the virtual network, the join module further configured to receive a leave request from the agent associated with the device to indicate that the device will be disconnected from data communication within the virtual network.	Forslöw at [0135] At configuration time, the mobile client 20 registers with the mobile service manager 22 in order to become part of a workgroup To the extent that Forslöw does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Forslöw , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-11

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent No. 6,701,437 ("Hoke")

Hoke titled "Method and apparatus for processing communications in a virtual private network" to Mark R. Hoke was filed on November 9, 1998 and issued on March 2, 2004.[1] Hoke is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Hoke does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Hoke or combine the teachings of Hoke with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Hoke at Abstract One embodiment of the present invention provides a computer system for processing communications in a virtual private network. To the extent that Hoke does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Hoke , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Hoke at 1:13–18 The present invention relates to the field of data communications. More specifically, the present invention relates to a device for processing communications and a method of configuring such a device to selectively encrypt communications depending upon whether they are being passed between members of a virtual private network. Hoke at 3:30–33 One embodiment of the present invention is used in a VPN operating over a public data network connected to an organization's private network (e.g., a LAN or WAN). Hoke at 5:59–62 FIG. 1 illustrates a public network 100, such as the Internet, including VPN units 115, 125, 135, 145 and 155 operating under the control of VPN management station 160 in accordance with an embodiment of the present invention.

Claims	Prior Art

	Hoke at 15:24–26 A VPN unit object comprises a number of parameters including, for example, an Internet Protocol (IP) address of the VPN unit. Hoke at 15:31–33 In state 704, the system manager issues commands to create and define group objects for various groups of entities connected to public network 100. To the extent that Hoke does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Hoke , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Hoke at 13:57–61 VPN unit 115 includes processor 600, which is coupled to RAM 602, for executing instructions stored in RAM 602. VPN unit 115 also includes storage memories 604, 608, which take the form of non-volatile flash PROMs in one embodiment of the invention. Hoke at 15:26–30 Before VPN unit objects are created and added to the VPN unit database, however, the system manager authenticates each VPN unit and each VPN unit authenticates the system manager. To the extent that Hoke does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Hoke , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-12

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent Application Publication No. 2002/0116502 ("Iyer")

Iyer titled "Proxy-less packet routing between private and public address realms" to Prakash N. Iyer was filed on February 22, 2001 and published on August 22, 2002.[1] Iyer is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Iyer does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Iyer or combine the teachings of Iyer with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Iyer at [0019] Gateway 16 has a private network interface 15 and a public network interface 17. To the extent that Iyer does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Iyer , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Iyer at [0019] Gateway 16 has a private network interface 15 and a public network interface 17. To the extent that Iyer does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Iyer , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Iyer at [0014] Private network 12 includes private clients, here computers A and B, and gateway 16 Iyer at [0019] Here, gateway 16 also has a processor 62 (FIG. 6) for executing various instructions and tasks such as routing packets.

Claims	Prior Art
	Iyer at [0023] After receipt of network public IP address Pu_G, process 30 registers (305) Pu_G with private and public agents such as private registry 32 and public registry 34 (FIG. 1) as the source (i.e. return) address for private client A. Finally, process 30 un-registers (307) Pu_G with private and public agents, such a registries 32 and 34, and un-reserves Pu_G once application X running on private client A is finished. Iyer at [0029] Computer 60 includes a processor 62, a memory 64 including executable instructions 61, a protocol stack 65 and a mapping table 67, and private and public network interfaces 15 and 17. Processor 62 executes computer instruction of RAM (not shown) to implement processes 30 and 40. Iyer at [0030] Processes 30 and 40 may be implemented in computer programs executing on programmable computers that each include a processor, a storage medium readable by the processor (including volatile and non-volatile memory and/or storage elements and input and output devices. To the extent that Iyer does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Iyer , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	Iyer at [0023] After receipt of network public IP address Pu_G, process 30 registers (305) Pu_G with private and public agents such as private registry 32 and public registry 34 (FIG. 1) as the source (i.e. return) address for private client A. Finally, process 30 un-registers (307) Pu_G with private and public agents, such a registries 32 and 34, and un-reserves Pu_G once application X running on private client A is finished. To the extent that Iyer does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Iyer , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	**Prior Art**
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Iyer at [0015] Private clients A and B share a private addressing scheme known only to members of private network 12 (i.e. the private addressing realm). Each private client has its own private IP address. Here, private client A has the private IP address of Pr_A and private client B has the private IP address of Pr_B. Iyer at [0016] Here, private network 12 also includes a private registry 32. Private clients (e.g. A or B) register an IP addresses in private registry 32 so that applications running on other clients may locate the registered private client. To the extent that Iyer does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Iyer , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[35.0] 35. The virtual network manager of claim 30 further comprising a join module	

Claims	Prior Art
configured to receive a join request from the agent associated with the device to indicate that the device is connected for data communication within the virtual network, the join module further configured to receive a leave request from the agent associated with the device to indicate that the device will be disconnected from data communication within the virtual network.	receive a leave request from the agent associated with the device to indicate that the device will be disconnected from data communication within the virtual network." For example: Iyer at [0023] After receipt of network public IP address Pu_G, process 30 registers (305) Pu_G with private and public agents such as private registry 32 and public registry 34 (FIG. 1) as the source (i.e. return) address for private client A. Finally, process 30 un-registers (307) Pu_G with private and public agents, such a registries 32 and 34, and un-reserves Pu_G once application X running on private client A is finished. To the extent that Iyer does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Iyer , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[37.0] 37. The virtual network manager of claim 35 wherein the join module is further configured to maintain data to associate a virtual network address with a device in the virtual network.	Iyer at [0023] After receipt of network public IP address Pu_G, process 30 registers (305) Pu_G with private and public agents such as private registry 32 and public registry 34 (FIG. 1) as the source (i.e. return) address for private client A. Finally, process 30 un-registers (307) Pu_G with private and public agents, such a registries 32 and 34, and un-reserves Pu_G once application X running on private client A is finished. To the extent that Iyer does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Iyer , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-13

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent Application Publication No. 2003/0065785 ("Jain")

Jain titled "Method and system for contacting a device on a private network using a specialized domain name server" to Nikhil Jain was filed on September 28, 2001 and published on April 3, 2003.[1] Jain is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Jain does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Jain or combine the teachings of Jain with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

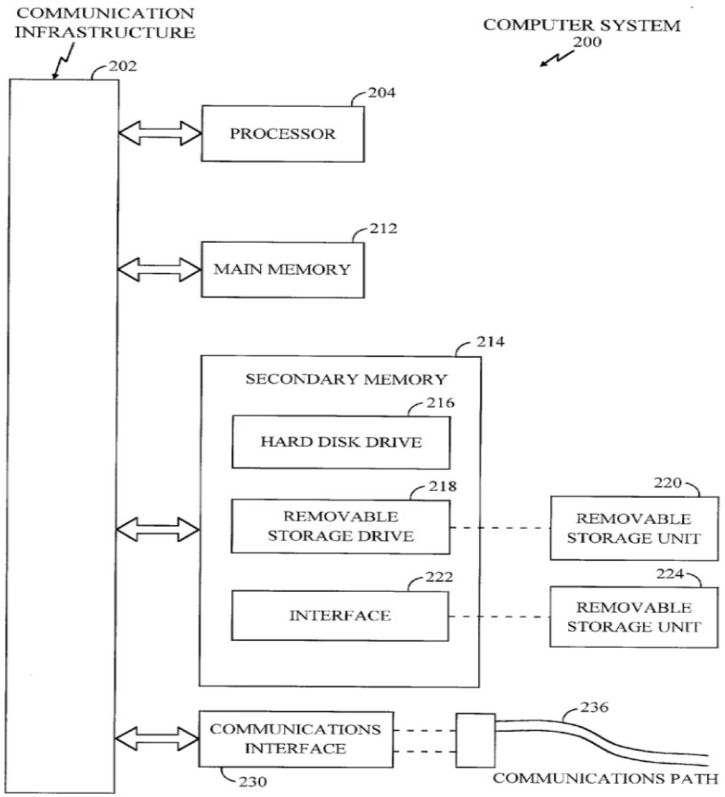
As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

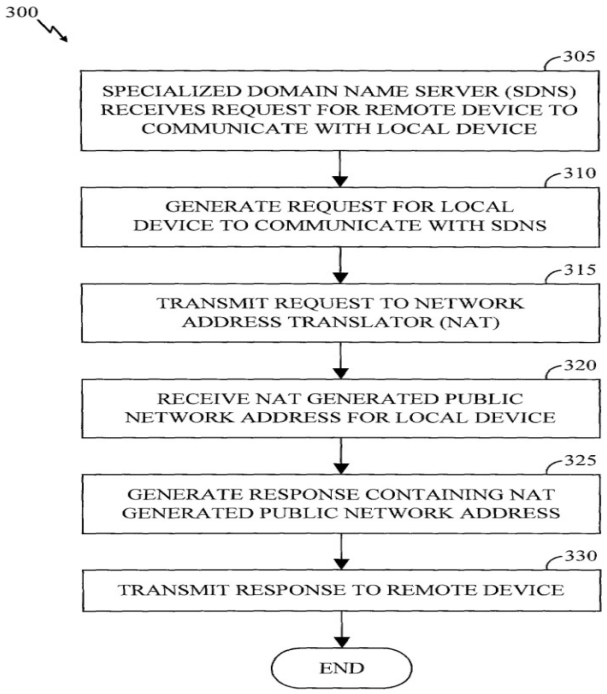
Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Jain at [0002] The present invention relates generally to communications between publicly and privately networked computers or data devices, and more particularly to a system and method for initiating communications with communication devices in a private network from sources outside of the private network. To the extent that Jain does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Jain , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Jain at Abstract The present invention provides a method and system for contacting a local computer on a private network over a public network using a specialized domain name server. Jain at [0002] The present invention relates generally to communications between publicly and privately networked computers or data devices, and more particularly to a system and method for initiating communications with communication devices in a private network from sources outside of the private network. Jain at [0008] The present invention is a method and system for using a specialized domain name server to process requests for remote computers or other connected devices to communicate with local computers or connected devices on a private network.

Claims	Prior Art
	Jain at [0018] The term "public network" refers to the Internet, or other large area or widely accessible network. Jain at [0023] The remote devices 105 are connected to a public network (for example, the Internet) 110. System 100 further comprises a specialized domain name server (SDNS) 115 Jain at [0024] In the disclosed embodiment, remote devices 105 and local devices 125 are shown as general purpose computers 105A and 125A; respectively, wireless communication devices 107, 127 which are in communication with wireless network interfaces 105B, 125B, respectively; and FAX machine 105C and printer 125C. To the extent that Jain does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Jain , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Jain at Figure 2

Claims	Prior Art
	 FIG. 2 Jain at [0008] The requests are initiated by the remote devices and identify the privately networked device, such as computers, by their domain names. In processing communications requests, the specialized domain name server, first receives over a public network, a request for a remote device to communicate with a local device located within the private network. The request uses a domain name assigned to the local device. Next, the specialized domain name server generates a subsequent request for the local device to communicate with the specialized domain name server. This subsequent request is generated in response to receiving the request for the remote device to communicate with the local device. Jain at [0009]

Claims	Prior Art
	After receiving the public network address, the specialized domain name server generates a response to the request for the remote device to communicate with the local device. The response includes the public network address for the local device. Jain at [0018]–[0019] The term "local device" refers to a general purpose computer or any other computer networked device located within the private network. Jain at [0029] Here, SDNS 115 represents any single or multi-processor computer. Single-threaded and multi-threaded computers can be used. Unified or distributed memory systems can be used. Jain at [0030] SDNS 115 includes one or more processors, such as processor 204. One or more processors 204 can execute software and implement all or part of the features of the present invention described herein Jain at [0031] SDNS 115 also includes or is connected to a main memory 212, preferably in the form of random access memory (RAM), and can also include secondary memory 214. Jain at [0037] Computer programs (also called computer control logic) are stored in main memory 212 and/or secondary memory 214. Computer programs can also be received through communications interface 230. Jain at [0037] In particular, the computer programs, when executed, enable the processor 204 to perform the features of the present invention. To the extent that Jain does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Jain , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	

Claims	Prior Art
	Jain at [0008] The requests are initiated by the remote devices and identify the privately networked device, such as computers, by their domain names. In processing communications requests, the specialized domain name server, first receives over a public network, a request for a remote device to communicate with a local device located within the private network. The request uses a domain name assigned to the local device. Next, the specialized domain name server generates a subsequent request for the local device to communicate with the specialized domain name server. This subsequent request is generated in response to receiving the request for the remote device to communicate with the local device. Jain at [0009] After receiving the public network address, the specialized domain name server generates a response to the request for the remote device to communicate with the local device. The response includes the public network address for the local device. To the extent that Jain does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Jain , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Jain at [0009] After receiving the public network address, the specialized domain name server generates a response to the request for the remote device to communicate with the local device. The response includes the public network address for the local device. Jain at [0027] Local devices 125 are each assigned domain names and IP addresses that are unique to the private network 130 in which they are located. Domain names and IP addresses are used to distinguish one computer on a network from another. To the extent that Jain does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Jain , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Jain at Abstract The present invention provides a method and system for contacting a local computer on a private network over a public network using a specialized domain name server. Jain at Figure 3  FIG. 3 Jain at [0028] SDNS 115 is responsible for providing remote devices 105 with the, NAT generated, public network addresses for local devices 125.

Claims	Prior Art
	Jain at [0041] name server recognizes a domain name when a corresponding entry for the domain name is found in the name server's lookup table. Jain at [0042] To generate the subsequent request, SDNS 115 searches a memory or storage element, such one having a lookup table, for the domain name of the local device 125 specified in the communications request and retrieves the local device's corresponding IP address within the private network 130. To the extent that Jain does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Jain , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[35.0] 35. The virtual network manager of claim 30 further comprising a join module configured to receive a join request from the agent associated with the device to indicate that the device is connected for data communication within the virtual network, the join module further configured to receive a leave request from the agent associated with the device to indicate that the device will be disconnected from data communication within the virtual network.	Jain at [0042] To generate the subsequent request, SDNS 115 searches a memory or storage element, such one having a lookup table, for the domain name of the local device 125 specified in the communications request and retrieves the local device's corresponding IP address within the private network 130. To the extent that Jain does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Jain , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-14

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent Application Publication No. 2002/0035624 ("Kim")

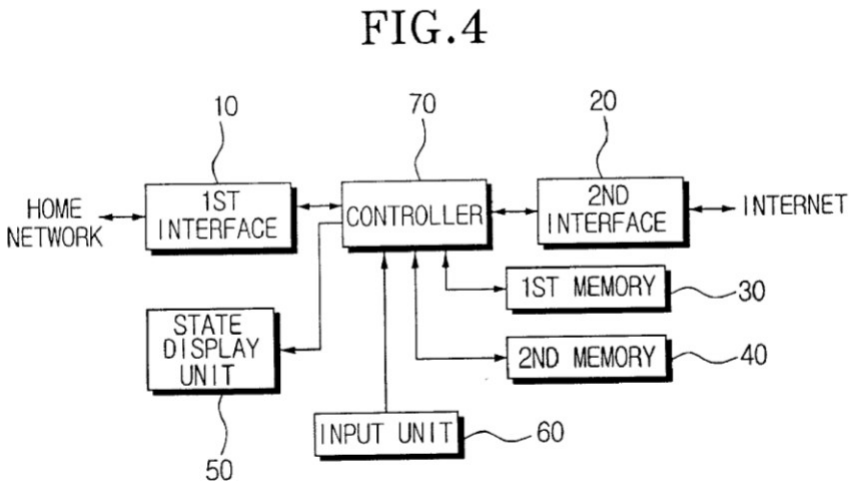
Kim titled "Gateway and a method for operating the same" to Jun-hyeong Kim was filed on July 9, 2001 and published on March 21, 2002.[1] Kim is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(b) and (e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

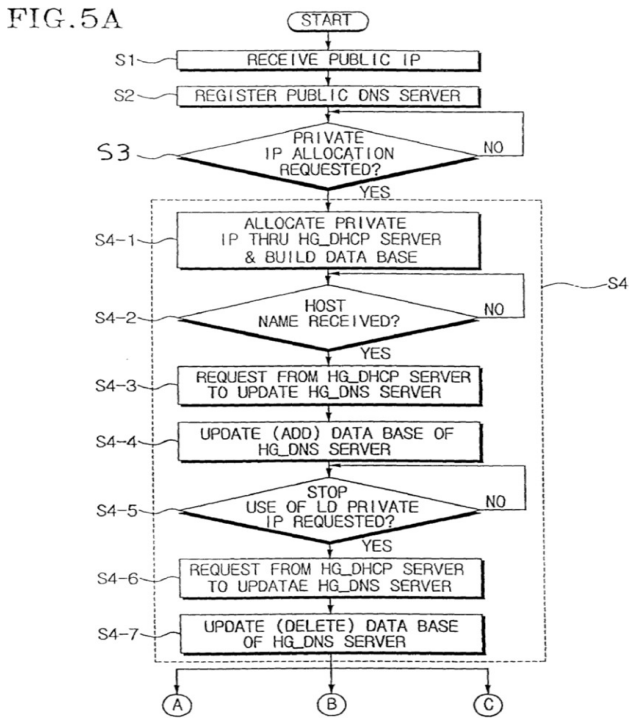
To the extent Plaintiff alleges Kim does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Kim or combine the teachings of Kim with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

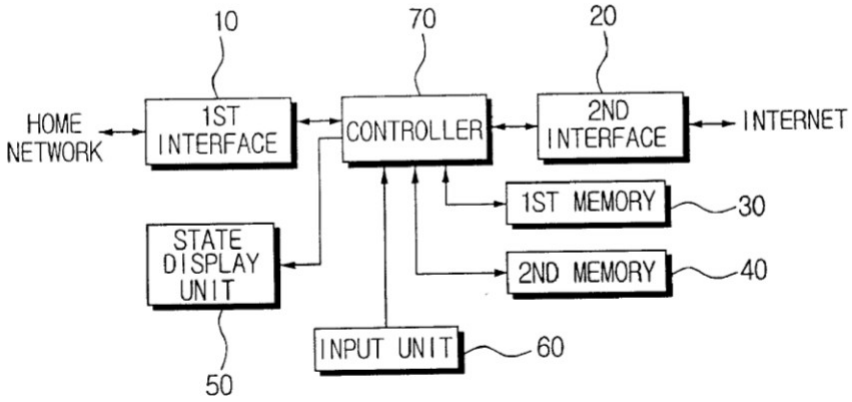
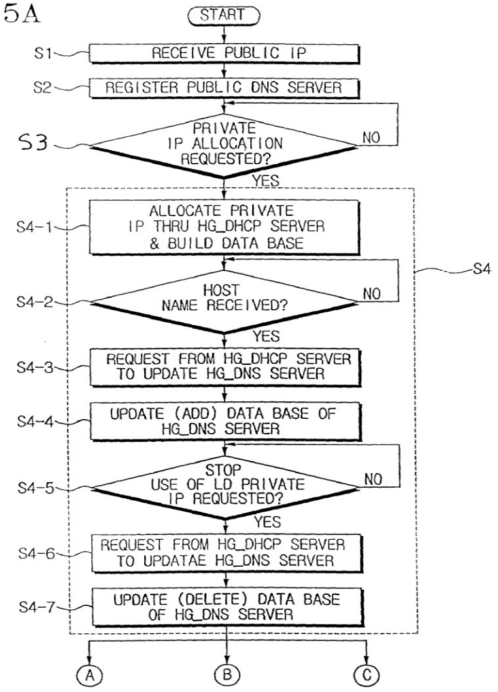
[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Kim at Figure 4  Kim at [0018] In order to solve the above problems, it is an object of the present invention to provide a gateway and a method for operating the same, capable of accessing a network from outside, sharing one public IP address by information appliances connected to the network, and managing the information appliances connected to the network at the entrance at homes or offices in order to enable users lacking relevant knowledge to easily build the network. To the extent that Kim does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kim , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Kim at Figure 5A  Kim at [0002] The present invention relates to a network apparatus and a method for operating the same, and more particularly, to a gateway and a method for operating the same, enabling information appliances connected to an internal network and information appliances connected through an external network to perform mutual data communications. Kim at [0018] In order to solve the above problems, it is an object of the present invention to provide a gateway and a method for operating the same, capable of accessing a network from outside, sharing one public IP address by information appliances connected to the network, and managing the information appliances

Claims	Prior Art
	connected to the network at the entrance at homes or offices in order to enable users lacking relevant knowledge to easily build the network. Kim at [0019] In order to achieve the above object, the gateway according to the present invention includes a first interface for communicating with information appliances connected to an internal network; a second interface for communicating with information appliances connected to an external network; and a controller for, if a control request with respect to any of the information appliances connected to the internal network is received from the information appliances connected to the external network, requesting a function performance to a corresponding appliance according to requested control contents. Kim at [0020] In order to achieve the above object, the method for operating the gateway having a first interface for communicating with information appliances connected to an internal network, a second interface for communicating with information appliances connected to an external network, and a controller for enabling intercommunication between the information appliances connected to the internal and external networks, includes the steps of connecting the Internet upon an initialization of a system, and providing information on the information appliances connected to the internal network if an access request is transmitted from an information appliance connected to the external network; and if a control request with respect to any one of the information appliances connected to the internal network is received, requesting a function performance to the corresponding appliance according to requested control contents. Kim at [0032] HG, standing for a home gateway, indicates a system according to an embodiment of the present invention enabling intercommunications to be made between a home network and the Internet. Kim at [0033] ISP, standing for an Internet service provider, indicates a server in the Internet which provides a public IP address available in the Internet in order for the HG to be connected to the Internet. Kim at [0043] HG is provided with a unique domain name registered in the DNS server. Whenever connected to the Internet, the HG is allocated with a single public IP address from the ISP Kim at [0043]

Claims	Prior Art
	The respective LDs are allocated with respective private IP addresses from the HG, and communicate with one another with the private IP addresses. The LDs access the Internet via the HG, and carry out a control command which is transmitted from the RD via the HG. Kim at [0047] the HG receives a public IP address available in the Internet from the ISP connected to the Internet through the second interface 20 (step S1) and stores the received public IP address in the second memory 40. Kim at [0055] Still further, sharing one public IP address with a plurality of appliances solves the problem of the IP addresses shortage in the Internet. Kim at claim 13 The method as claimed in claim 10, further comprising a step of registering a domain name of the gateway and a public IP address of a system to be associated to each other on an initialization of the system, the domain name being registered in advance in a DNS server authorized in the external network. To the extent that Kim does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kim , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Kim at Figure 4

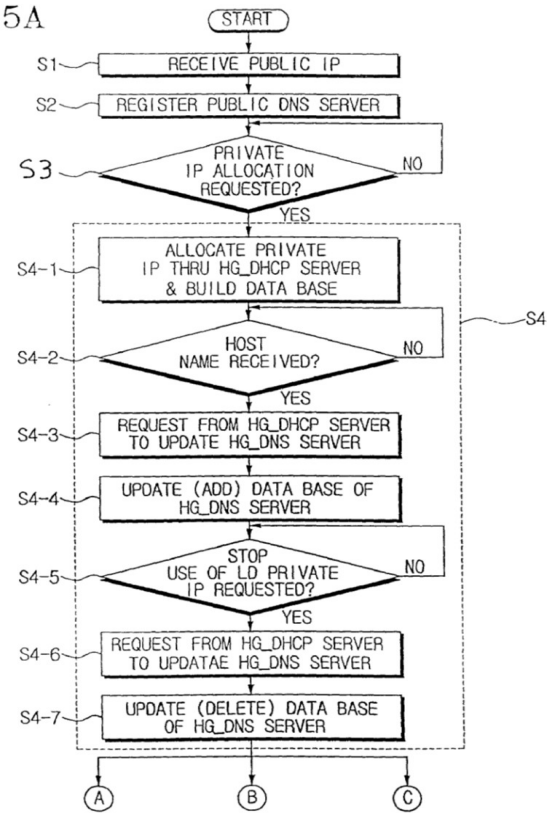
Claims	Prior Art
	FIG.4  Kim at Figure 5A FIG.5A 

Claims	Prior Art
	Kim at [0019] In order to achieve the above object, the gateway according to the present invention includes a first interface for communicating with information appliances connected to an internal network; a second interface for communicating with information appliances connected to an external network; and a controller for, if a control request with respect to any of the information appliances connected to the internal network is received from the information appliances connected to the external network, requesting a function performance to a corresponding appliance according to requested control contents. Kim at [0020] In order to achieve the above object, the method for operating the gateway having a first interface for communicating with information appliances connected to an internal network, a second interface for communicating with information appliances connected to an external network, and a controller for enabling intercommunication between the information appliances connected to the internal and external networks, includes the steps of connecting the Internet upon an initialization of a system, and providing information on the information appliances connected to the internal network if an access request is transmitted from an information appliance connected to the external network; and if a control request with respect to any one of the information appliances connected to the internal network is received, requesting a function performance to the corresponding appliance according to requested control contents. Kim at [0035] LD, standing for a local device, indicates information appliances, for example, an information electronic appliance such as a digital TV and an information terminal such as a computer, which are connected to a home network. Kim at [0036] HG_DHCP, standing for a dynamic host configuration protocol of a home gateway, indicates a protocol for dynamically allocating a private IP address to an LD when the private IP address allocation is requested from the LD, as well as indicates a server for playing a role of the above in the drawings. Kim at [0045] The first memory 30 stores a program for operating a system. Kim at [0047]

Claims	Prior Art
	the HG receives a public IP address available in the Internet from the ISP connected to the Internet through the second interface 20 (step S1) and stores the received public IP address in the second memory 40. Kim at [0049] As stated above, in case an external user wants to control an LD connected to a home network, an application proxy server of the HG is activated as an agent, and the HG performs its function in order for services such as FTP, Telnet, embedded WEB, and the like to be provided. Kim at claim 2 a DHCP server which allocates and manages different private IP addresses in accordance with a private IP address allocation request from the information appliances connected to the internal network, and receives host names from the information appliances allocated with the private IP addresses; a DNS server which builds a database in order for the host names and the private IP addresses to be associated in response to an update request from the DHCP server; and To the extent that Kim does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kim , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	Kim at Figure 5A

Claims	Prior Art

FIG.5A



Kim at [0019]

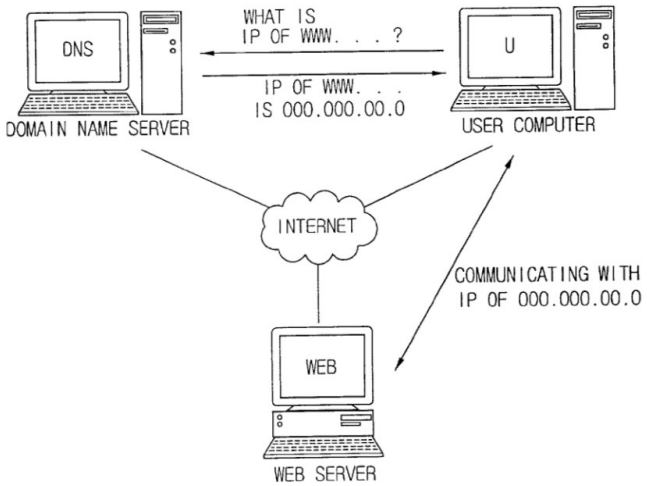
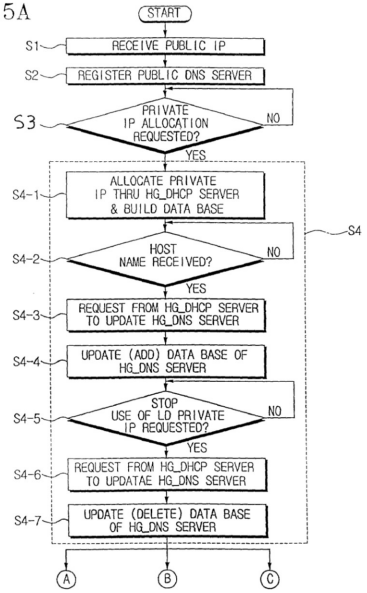
In order to achieve the above object, the gateway according to the present invention includes a first interface for communicating with information appliances connected to an internal network; a second interface for communicating with information appliances connected to an external network; and a controller for, if a control request with respect to any of the information appliances connected to the internal network is received from the information appliances connected to the external network, requesting a function performance to a corresponding appliance according to requested control contents.

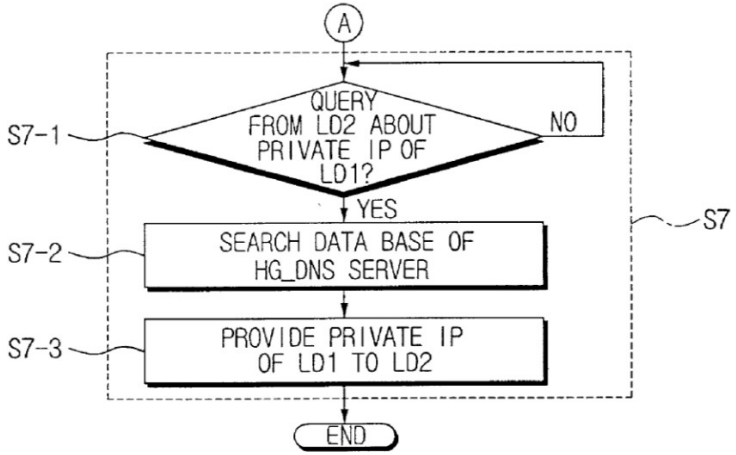
Kim at [0020]

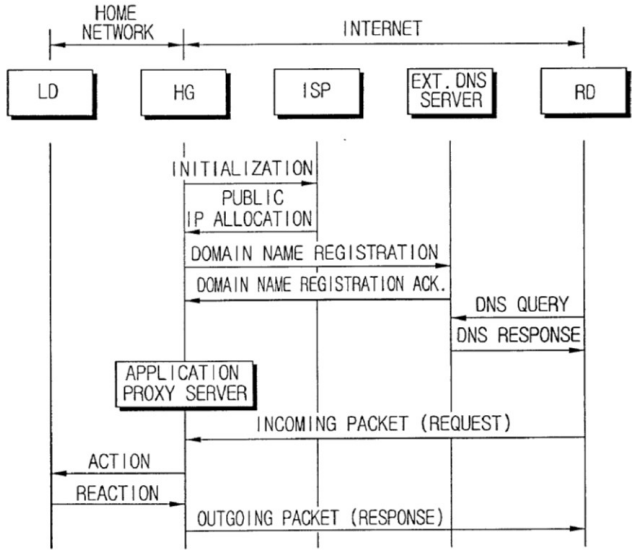
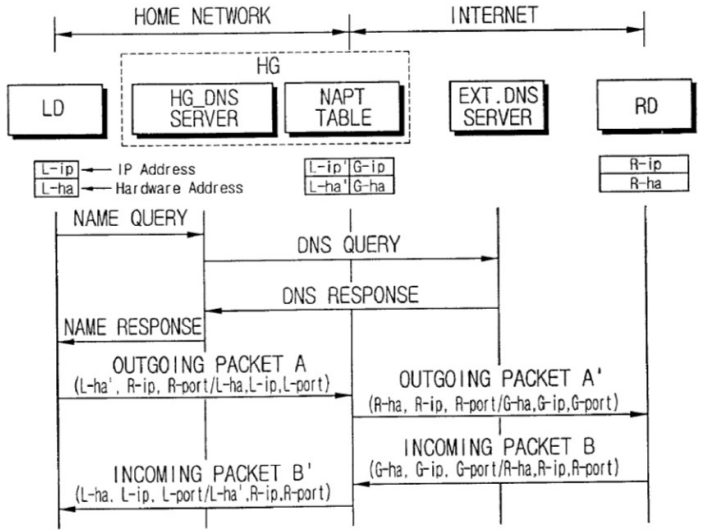
In order to achieve the above object, the method for operating the gateway having a first interface for communicating with information appliances connected to an internal network, a second interface for communicating with information appliances connected to an external network, and a controller for enabling intercommunication between the information appliances connected to the internal and external

Claims	Prior Art
	networks, includes the steps of connecting the Internet upon an initialization of a system, and providing information on the information appliances connected to the internal network if an access request is transmitted from an information appliance connected to the external network; and if a control request with respect to any one of the information appliances connected to the internal network is received, requesting a function performance to the corresponding appliance according to requested control contents. Kim at [0036] HG_DHCP, standing for a dynamic host configuration protocol of a home gateway, indicates a protocol for dynamically allocating a private IP address to an LD when the private IP address allocation is requested from the LD, as well as indicates a server for playing a role of the above in the drawings. Kim at [0039] An application proxy server is a server for playing a role of an agent of LDs when a control request with respect to the LDs connected to a home network occurs from an RD, provides to RDs services such as FTP, TELNET, embedded WEB, and the like, and requests function performances to the LDs. Kim at [0045] The controller 70 also provides information on the LDs connected to the home network if an access request occurs through the Internet from an RD, and, if a control request with respect to either of the LDs connected to the home network is received from the access-requesting RD, requests a function performance suitable for a corresponding LD according to the requested control contents. Kim at [0049] As stated above, in case an external user wants to control an LD connected to a home network, an application proxy server of the HG is activated as an agent, and the HG performs its function in order for services such as FTP, Telnet, embedded WEB, and the like to be provided. To the extent that Kim does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kim , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the	

Claims	Prior Art
virtual network by the virtual network address; and	Kim at [0036] HG_DHCP, standing for a dynamic host configuration protocol of a home gateway, indicates a protocol for dynamically allocating a private IP address to an LD when the private IP address allocation is requested from the LD, as well as indicates a server for playing a role of the above in the drawings. Kim at [0047] That is, since a plurality of home networks exist in the Internet with the respective gateways distinguished by unique domain names of their own, the domain names cannot be arbitrarily chosen by users, but are allocated from the ISP in a manner in which they do not overlap with each other. Kim at claim 2 a DHCP server which allocates and manages different private IP addresses in accordance with a private IP address allocation request from the information appliances connected to the internal network, and receives host names from the information appliances allocated with the private IP addresses; a DNS server which builds a database in order for the host names and the private IP addresses to be associated in response to an update request from the DHCP server; and To the extent that Kim does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kim , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Kim at Figure 1

Claims	Prior Art
	FIG.1  Kim at Figure 5A FIG.5A 

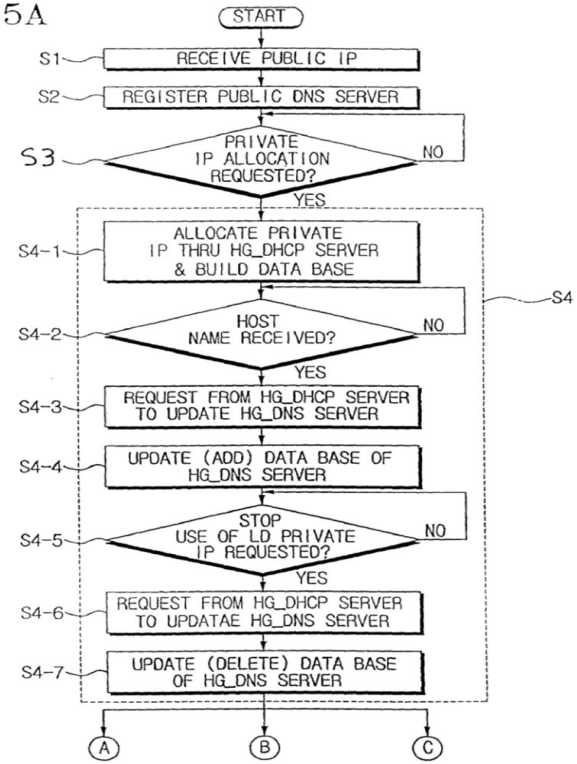
Claims	Prior Art
	Kim at Figure 5D FIG.5D  Kim at Figure 6

Claims	Prior Art
	FIG.6  Kim at Figure 8 FIG.8 

Claims	Prior Art
	Kim at [0012] FIG. 1 is a view for showing a process of the use of a domain name when a general user accesses an Internet communication. Kim at [0013] If a domain name is inputted through a web browser of a user computer U, the web browser requests a Domain Name Server (DNS), an IP address corresponding to the domain name through the Internet, the DNS searches a database of its own in response to the request of the web browser and provides the requested IP address corresponding to the domain name, and the web browser gives and takes data to and from a web server having the corresponding domain name in use of the IP address received. Kim at [0037] HG_DNS, standing for a domain name server of a home gateway, indicates, when an LD makes an inquiry into an address with respect to another LD or an RD, a server for providing a private or a public IP address to the corresponding LD or RD, as well as indicates a server for playing a role of the above in the drawings. Kim at [0043] The respective LDs are allocated with respective private IP addresses from the HG, and communicate with one another with the private IP addresses. The LDs access the Internet via the HG, and carry out a control command which is transmitted from the RD via the HG. Kim at [0045] Further, the controller 70 is allocated with a public IP address through the second interface 20 from an ISP server connected to the Internet upon initializing a system, and registers the allocated public IP address to be associated with a domain name registered in advance in a DNS server connected to the Internet. Furthermore, the controller 70 loads an HG_DHCP server from the first memory 30, allocates different private IP addresses to respective LDs through the first interface 10 and builds a database related to the private IP address allocations, and receives host names from the LDs having the allocated private IP addresses. Kim at [0045] Furthermore, if a home information appliance connected to a home network makes an inquiry about a private IP address through a host name with respect to another home information appliance connected to

Claims	Prior Art
	the same home network, the controller 70 provides the requested private IP address with reference to the database of the HG_DNS server. Kim at claim 2 a DHCP server which allocates and manages different private IP addresses in accordance with a private IP address allocation request from the information appliances connected to the internal network, and receives host names from the information appliances allocated with the private IP addresses; a DNS server which builds a database in order for the host names and the private IP addresses to be associated in response to an update request from the DHCP server; and Kim at claim 4 The gateway as claimed in claim 2, wherein the DNS server builds the database by combining a domain name of the gateway and the host names of the information appliances connected to the internal network at a home, the domain name being registered in advance in an authorized DNS server connected to the external network. To the extent that Kim does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kim , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art
	communicating with information appliances connected to an external network, and a controller for enabling intercommunication between the information appliances connected to the internal and external networks, includes the steps of connecting the Internet upon an initialization of a system, and providing information on the information appliances connected to the internal network if an access request is transmitted from an information appliance connected to the external network; and if a control request with respect to any one of the information appliances connected to the internal network is received, requesting a function performance to the corresponding appliance according to requested control contents. Kim at [0039] An application proxy server is a server for playing a role of an agent of LDs when a control request with respect to the LDs connected to a home network occurs from an RD, provides to RDs services such as FTP, TELNET, embedded WEB, and the like, and requests function performances to the LDs. Kim at [0049] As stated above, in case an external user wants to control an LD connected to a home network, an application proxy server of the HG is activated as an agent, and the HG performs its function in order for services such as FTP, Telnet, embedded WEB, and the like to be provided. To the extent that Kim does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kim , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[35.0] 35. The virtual network manager of claim 30 further comprising a join module configured to receive a join request from the agent associated with the device to indicate that the device is connected for data communication within the virtual network, the join module further configured to receive a leave request from the agent associated with the device to indicate that the device will be disconnected from data	Kim at Figure 5A

Claims	Prior Art
communication within the virtual network.	FIG.5A  Kim at [0019] In order to achieve the above object, the gateway according to the present invention includes a first interface for communicating with information appliances connected to an internal network; a second interface for communicating with information appliances connected to an external network; and a controller for, if a control request with respect to any of the information appliances connected to the internal network is received from the information appliances connected to the external network, requesting a function performance to a corresponding appliance according to requested control contents. Kim at [0020] In order to achieve the above object, the method for operating the gateway having a first interface for communicating with information appliances connected to an internal network, a second interface for communicating with information appliances connected to an external network, and a controller for enabling intercommunication between the information appliances connected to the internal and external networks, includes the steps of connecting the Internet upon an initialization of a system, and providing

Claims	Prior Art
	information on the information appliances connected to the internal network if an access request is transmitted from an information appliance connected to the external network; and if a control request with respect to any one of the information appliances connected to the internal network is received, requesting a function performance to the corresponding appliance according to requested control contents. Kim at [0039] An application proxy server is a server for playing a role of an agent of LDs when a control request with respect to the LDs connected to a home network occurs from an RD, provides to RDs services such as FTP, TELNET, embedded WEB, and the like, and requests function performances to the LDs. Kim at [0045] The controller 70 also provides information on the LDs connected to the home network if an access request occurs through the Internet from an RD, and, if a control request with respect to either of the LDs connected to the home network is received from the access-requesting RD, requests a function performance suitable for a corresponding LD according to the requested control contents. To the extent that Kim does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kim , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[37.0] 37. The virtual network manager of claim 35 wherein the join module is further configured to maintain data to associate a virtual network address with a device in the virtual network.	Kim at [0020] In order to achieve the above object, the method for operating the gateway having a first interface for communicating with information appliances connected to an internal network, a second interface for communicating with information appliances connected to an external network, and a controller for enabling intercommunication between the information appliances connected to the internal and external networks, includes the steps of connecting the Internet upon an initialization of a system, and providing information on the information appliances connected to the internal network if an access request is transmitted from an information appliance connected to the external network; and if a control request with respect to any one of the information appliances connected to the internal network is received, requesting a function performance to the corresponding appliance according to requested control contents.

Claims	Prior Art
	Kim at [0045] The controller 70 also provides information on the LDs connected to the home network if an access request occurs through the Internet from an RD, and, if a control request with respect to either of the LDs connected to the home network is received from the access-requesting RD, requests a function performance suitable for a corresponding LD according to the requested control contents. To the extent that Kim does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kim , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-15

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent Application Publication No. 20020154635 ("Liu")

Liu titled "System and method for extending private networks onto public infrastructure using supernets" to Yuefeng Liu was filed on April 23, 2001 and published on October 24, 2002.[1] Liu is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Liu does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Liu or combine the teachings of Liu with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Liu at Abstract Methods and systems for enabling communications between a first private network and a second private network configured from nodes in a public network. Liu at [0017] To alleviate the problem of geographic restrictiveness, virtual private networks have been developed. Liu at [0020] A Supernet is a private network that uses components from a public-network infrastructure. Liu at [0022] Methods and systems consistent with the principles of the invention enable communications between a first private network and a second private network configured from nodes in a public network. To the extent that Liu does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Liu , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Liu at Abstract Methods and systems for enabling communications between a first private network and a second private network configured from nodes in a public network. Liu at [0015]

Claims	Prior Art
	The present invention relates generally to data processing systems and, more particularly, to extending private networks onto public infrastructure. Liu at [0021] Accordingly, there is a need for a system and method for connecting a pre-existing private network to a private network, such as a Supernet, built on top of public infrastructure. Liu at [0022] Methods and systems consistent with the principles of the invention enable communications between a first private network and a second private network configured from nodes in a public network. Liu at [0047] Since the Supernet is a private network constructed from the infrastructure of another network, the Supernet has its own internal addressing scheme, separate from the addressing scheme of the underlying public network Liu at [0049] Network environment 300 includes a number of devices, such as computers 302-312 (including administrative machine 306), connected to a public network, such as Internet 314. Network environment 300 also includes a number of devices, such as computers 330-334, connected to a private network, such as enterprise network 328. A device, such as computer 324, is connected to both Internet 314 and enterprise network 328. To the extent that Liu does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Liu , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Liu at [0046] A Supernet provides a number of features to ensure secure and robust communication among its nodes. First, the system provides authentication and admission control so that nodes become members of the Supernet under strict control to prevent unauthorized access. Second, the Supernet provides

Claims	Prior Art
	communication security services so that the sender of a message is authenticated and communication between end points occurs in a secure manner by using encryption. Liu at [0049] Network environment 300 includes a number of devices, such as computers 302-312 (including administrative machine 306), connected to a public network, such as Internet 314. Network environment 300 also includes a number of devices, such as computers 330-334, connected to a private network, such as enterprise network 328. A device, such as computer 324, is connected to both Internet 314 and enterprise network 328. Liu at [0051] FIG. 4A is a diagram of an administrative machine in which the features and aspects of the present invention may be implemented. Administrative machine 306 includes a memory 402, secondary storage 404, a central processing unit (CPU) 406, an input device 408, and a video display 410. Liu at [0059] Memory 502 of computer 302 includes SNlogin script 512, SNlogout script 514, VARPD 516, KMC 518, KMD 520, and node A 522, all running in user mode. Memory 502 also includes TCP/IP protocol stack 524 and VARPDB 526 running in kernel mode. Liu at [0061] KMC 518 is the key management component for each node that receives updates whenever the key for a channel ("the channel key") changes. There is one KMC per node per channel. KMD 520 receives requests from SNSL 532 of the TCP/IP protocol stack 524 when a packet is received and accesses the appropriate KMC for the destination node to retrieve the appropriate key to decrypt the packet. To the extent that Liu does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Liu , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	Liu at [0046]

Claims	Prior Art
	A Supernet provides a number of features to ensure secure and robust communication among its nodes. First, the system provides authentication and admission control so that nodes become members of the Supernet under strict control to prevent unauthorized access. Second, the Supernet provides communication security services so that the sender of a message is authenticated and communication between end points occurs in a secure manner by using encryption. Liu at [0061] KMC 518 is the key management component for each node that receives updates whenever the key for a channel ("the channel key") changes. There is one KMC per node per channel. KMD 520 receives requests from SNSL 532 of the TCP/IP protocol stack 524 when a packet is received and accesses the appropriate KMC for the destination node to retrieve the appropriate key to decrypt the packet. To the extent that Liu does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Liu , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Liu at [0025] Thereafter, the computer sends the packet to the router node using the address mapping. The router node forwards the packet to a destination node in the first private network based on an internal address in the packet for the destination node suitable for communicating among nodes in the first private network. Liu at [0046] A Supernet provides a number of features to ensure secure and robust communication among its nodes. First, the system provides authentication and admission control so that nodes become members of the Supernet under strict control to prevent unauthorized access. Second, the Supernet provides communication security services so that the sender of a message is authenticated and communication between end points occurs in a secure manner by using encryption. Liu at [0053] The "node ID" may include the following: a Supernet ID (e.g., 0×123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1).

Claims	Prior Art
	Liu at [0061] KMC 518 is the key management component for each node that receives updates whenever the key for a channel (“the channel key”) changes. There is one KMC per node per channel. KMD 520 receives requests from SNSL 532 of the TCP/IP protocol stack 524 when a packet is received and accesses the appropriate KMC for the destination node to retrieve the appropriate key to decrypt the packet. Liu at [0067] When router node 630 receives a packet on its enterprise network interface, it checks to see if the destination address is logged into the Supernet. If so, then router node 630 proceeds to initiate a transfer of the packet to the appropriate Supernet node via the router node's public network interface. A packet sent from a Supernet node to a device on enterprise network 328 is directed to router node 630, which recognizes that the packet is destined for a device on enterprise network 328. Router node 630 then forwards the packet to the appropriate device. To the extent that Liu does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Liu , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Liu at [0025] Based on the determination, the computer obtains an address mapping corresponding to a router node based on the determination. Liu at [0025] Thereafter, the computer sends the packet to the router node using the address mapping. The router node forwards the packet to a destination node in the first private network based on an internal address in the packet for the destination node suitable for communicating among nodes in the first private network. Liu at [0050]

Claims	Prior Art
	Nodes 316, 318, 320, and 322, and router node 326 together form a Supernet that may communicate by using the infrastructure of Internet 314. Router node 326 is also capable of communicating with computers 330-334 using enterprise network 328. Router node 326 enables devices that are part of enterprise network 328 to communicate with nodes from the Supernet. It is not necessary for those devices to be on a public infrastructure, such as Internet 314. Liu at [0053] The "node ID" may include the following: a Supernet ID (e.g., 0×123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1). Liu at [0053] server Liu at [0053] In this role, the VARPD stores into its associated VARPDB the address mappings for its nodes, and if it needs a mapping that it does not have, it will contact the VARPD that acts as the server for the given Supernet to obtain it. Liu at [0066] Router node 630 enables devices that are part of enterprise network 328 to communicate with nodes from the Supernet. Router node 630 is logged in to the Supernet and operates within the Supernet context. Liu at [0067] When router node 630 receives a packet on its enterprise network interface, it checks to see if the destination address is logged into the Supernet. If so, then router node 630 proceeds to initiate a transfer of the packet to the appropriate Supernet node via the router node's public network interface. A packet sent from a Supernet node to a device on enterprise network 328 is directed to router node 630, which recognizes that the packet is destined for a device on enterprise network 328. Router node 630 then forwards the packet to the appropriate device. Liu at [0070] First, router node 630 receives a packet originating from a node connected to enterprise network 328 (step 702). The packet includes a source node address, a destination node address, and data. Router node 630 accesses the VARPDB to obtain an address mapping corresponding to the destination node address (step 704). Next, VARPDB 620 determines whether the destination node address corresponds to a Supernet

Claims	Prior Art
	node (step 706). For example, VARPDB 620 determines whether it contains a mapping for the destination node address. To the extent that Liu does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Liu , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[35.0] 35. The virtual network manager of claim 30 further comprising a join module configured to receive a join request from the agent associated with the device to indicate that the device is connected for data communication within the virtual network, the join module further configured to receive a leave request from the agent associated with the device to indicate that the device will be disconnected from data	Liu at [0046] The system does so by providing one key per channel and by changing the key for a channel whenever a node joins or leaves the channel Liu at [0053] SASD 412 represents a Supernet: there is one instance of an SASD per Supernet, and it both authenticates nodes and authorizes nodes to join the Supernet.

Claims	Prior Art
communication within the virtual network.	Liu at [0061] KMC 518 is the key management component for each node that receives updates whenever the key for a channel ("the channel key") changes. There is one KMC per node per channel. KMD 520 receives requests from SNSL 532 of the TCP/IP protocol stack 524 when a packet is received and accesses the appropriate KMC for the destination node to retrieve the appropriate key to decrypt the packet. To the extent that Liu does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Liu , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

[37.0] 37. The virtual network manager of claim 35 wherein the join module is further configured to maintain data to associate a virtual network address with a device in the virtual network.	Liu at [0053] The "node ID" may include the following: a Supernet ID (e.g., 0×123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1). Liu at [0053] In this role, the VARPD stores into its associated VARPDB the address mappings for its nodes, and if it needs a mapping that it does not have, it will contact the VARPD that acts as the server for the given Supernet to obtain it. To the extent that Liu does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Liu , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-16

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent No. 8,068,817 ("Viswanath")

Viswanath titled "Virtual address translation to support wireless access to data networks" to Kaartik Viswanath et al. was filed on August 27, 2002 and issued on November 29, 2011.[1] Viswanath is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Viswanath does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Viswanath or combine the teachings of Viswanath with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Viswanath at 2:38–40 According to particular embodiments, elements of system 10 are configured to support access to networks based on network access requests that identify virtual networks. To the extent that Viswanath does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Viswanath , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Viswanath at 1:34–43 According to a particular embodiment, a method for processing network access requests for mobile devices receives a network access request for establishment of a communication link for data communications for a mobile device, with the network access request including a requested network field and a pass-through field. The method determines a requested network from text in the pass-through field, sets the requested network field to the requested network, and determines a gateway linking to the requested network. The method then forwards the network access request to the gateway. Viswanath at 5:27–35 Radius server 32 provides authentication and authorization services. In response to an authentication and authorization request that identifies a user, password and requested network, radius 32 indicates whether the user is authorized and authenticated. For example, using the user identifier and password, radius server 32 can authenticate the identity of the user. Using the user identifier and requested network, radius server 32 can determine whether the user is authorized to access that network.

Claims	Prior Art
	Viswanath at 8:24–26 FIG. 2 illustrates a particular embodiment of mobile device 12 that includes a user interface 50, a wireless interface 52, a processor 54 and a memory 56. To the extent that Viswanath does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Viswanath , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	Viswanath at 5:27–35 Radius server 32 provides authentication and authorization services. In response to an authentication and authorization request that identifies a user, password and requested network, radius 32 indicates whether the user is authorized and authenticated. For example, using the user identifier and password, radius server 32 can authenticate the identity of the user. Using the user identifier and requested network, radius server 32 can determine whether the user is authorized to access that network. Viswanath at 5:36–41 Upon receiving a network access request, gateway 24 uses DHCP server 30 and/or radius server 32 to process the request and determine whether mobile device 12 should be granted access to the requested network. In providing mobile devices 12 access to data networks, gateways 24 maintain routing information for these mobile devices 12. To the extent that Viswanath does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Viswanath , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Viswanath at 5:36–41

Claims	Prior Art
	Upon receiving a network access request, gateway 24 uses DHCP server 30 and/or radius server 32 to process the request and determine whether mobile device 12 should be granted access to the requested network. In providing mobile devices 12 access to data networks, gateways 24 maintain routing information for these mobile devices 12. To the extent that Viswanath does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Viswanath , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Viswanath at 3:5–8 In addition, operator network 16 includes domain name servers (DNS) 26 and 27, a home location register (HLR) 28, a dynamic host configuration protocol (DHCP) server 30, and a radius server 32. Viswanath at 3:13–18 Domain name servers 26 and 27 map domain names to internet protocol (IP) addresses and return these addresses in response to appropriate requests. For example, in response to a request identifying a particular access point name, domain name server 26 returns a list of IP addresses for gateways 24 providing access to the identified access point name. Viswanath at 3:48–50 In response to this request, domain name server 26 returns one or more IP address matching to the access point name in the domain name request. Viswanath at 4:2–6 Thus, domain name server 26 is configured to recognize virtual networks and return the IP address for virtual address translation module 34 in response to domain name requests identifying virtual networks. To the extent that Viswanath does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Viswanath , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art
[35.0] 35. The virtual network manager of claim 30 further comprising a join module configured to receive a join request from the agent associated with the device to indicate that the device is connected for data communication within the virtual network, the join module further configured to receive a leave request from the agent associated with the device to indicate that the device will be disconnected from data communication within the virtual network.	Viswanath at 5:36–41 Upon receiving a network access request, gateway 24 uses DHCP server 30 and/or radius server 32 to process the request and determine whether mobile device 12 should be granted access to the requested network. In providing mobile devices 12 access to data networks, gateways 24 maintain routing information for these mobile devices 12. To the extent that Viswanath does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Viswanath , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[37.0] 37. The virtual network manager of claim 35 wherein the join module is further configured to maintain data to associate a virtual network address with a device in the virtual network.	Viswanath at 5:36–41 Upon receiving a network access request, gateway 24 uses DHCP server 30 and/or radius server 32 to process the request and determine whether mobile device 12 should be granted access to the requested network. In providing mobile devices 12 access to data networks, gateways 24 maintain routing information for these mobile devices 12. To the extent that Viswanath does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Viswanath , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-17

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent Application Publication No. 2002/0116523 ("Warrier")

Warrier titled "Assigning a source address to a data packet based on the destination of the data packet" to Ulhas S. Warrier was filed on February 22, 2001 and published on August 22, 2002.[1] Warrier is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Warrier does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Warrier or combine the teachings of Warrier with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	**Prior Art**
[30.0] 30. A virtual network manager, comprising:	Warrier at [0003] FIG. 1 is a view of a computer network having private and public addressing realms. To the extent that Warrier does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Warrier , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Warrier at Abstract The network has public and private realms separated by an interface device. Warrier at [0040] Virtual network interface card 47 is a software module that provides a public address of gateway 16 to socket interceptor 46 and socket interface 44. To the extent that Warrier does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Warrier , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Warrier at [0017]

Claims	Prior Art
	In this embodiment, gateway 16 is a computer, which belongs to both the public and the private realms and uses interfaces 15 and 17 to connect private realm 12 with the public realm 14. Warrier at [0033] Process 30 assigns (301) a source port 11 for a packet 20 by providing source port number 29 of port 11 to private client A. Source port number 29 is registered with gateway 16 for use in de-multiplexing the addressing information (i.e. address field 24) of reply packets to client A if packet 20 is sent to the public addressing realm. Source port number 29 may also be selected from a pool of source ports previously registered with gateway 16 to increase the efficiency of assigning ports 11 in process 30. Warrier at [0043] Assigned source port number 29 may be stored in a memory table (not shown) accessible to gateway 16 Warrier at [0046] Computer 50 includes a processor 52, a memory 54 including executable instructions 51, a protocol stack 55, a mapping table 57 accessible to gateway 16, a driver 56 serving as virtual interface card and a network interface card 58. Processor 52 executes computer instructions of RAM (not shown) to implement process 30. To the extent that Warrier does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Warrier , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	Warrier at [0033] Process 30 assigns (301) a source port 11 for a packet 20 by providing source port number 29 of port 11 to private client A. Source port number 29 is registered with gateway 16 for use in de-multiplexing the addressing information (i.e. address field 24) of reply packets to client A if packet 20 is sent to the public addressing realm. Source port number 29 may also be selected from a pool of source ports previously registered with gateway 16 to increase the efficiency of assigning ports 11 in process 30. To the extent that Warrier does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Warrier , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Warrier at [0033] Process 30 assigns (301) a source port 11 for a packet 20 by providing source port number 29 of port 11 to private client A. Source port number 29 is registered with gateway 16 for use in de-multiplexing the addressing information (i.e. address field 24) of reply packets to client A if packet 20 is sent to the public addressing realm. Source port number 29 may also be selected from a pool of source ports previously registered with gateway 16 to increase the efficiency of assigning ports 11 in process 30. To the extent that Warrier does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Warrier , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Warrier at [0027] Hence, a private client, here A, has two source addresses; a private source address (i.e. source address 28 and source port 29 in FIG. 2A) and a public source address (i.e. source address 28 and source port 29 in FIG. 2C); depending on the destination of packet 20. The multiple addresses are needed to ensure packets 20D are routed correctly in the public and private realms. Warrier at [0036] If the destination address 26 of packet 20 is public, process 30 retrieves (309) the public address of the gateway for use as the source address 28. Warrier at [0036] If the destination address 26 of packet 20 is private, process 30 retrieves (311) the private address of the client for use as source address 28.

Claims	Prior Art
	To the extent that Warrier does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Warrier , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-18

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent Application Publication No. 2003/0229697 ("Borella")

Borella titled "Method and apparatus for global server load balancing" to Michael S. Borella was filed on June 10, 2002 and published on December 11, 2003.[1] Borella is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Borella does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Borella or combine the teachings of Borella with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Borella at [0001] The present invention relates to data communications. More specifically, it relates to routing of packets between a client device and server devices connected via a network. To the extent that Borella does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Borella , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Borella at [0001] The present invention relates to data communications. More specifically, it relates to routing of packets between a client device and server devices connected via a network. Borella at [0028] FIG. 2 shows an architecture 100 that illustrates an example of the connection of a mobile node (MN) 124 to the public IP network 30. The public IP network 30 includes a mobile node's home agent 110 and a foreign agent 140. The home agent 110 is coupled to the IP network 30 via a communication link 112 and has a globally routable network address of 3.0.0.100 on the network 30. To the extent that Borella does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Borella , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module	

Claims	Prior Art
configured to register devices in a virtual network, the register module further configured to:	Borella at [0012] One exemplary method includes providing a plurality of foreign agents at a data center, and using a Mobile Internet Protocol to dynamically assign a network address to each of the plurality of servers at the data center. The method further includes providing a plurality of home agents at a content owner site. In such an embodiment, using the Mobile Internet Protocol to dynamically assign a network address to each of the plurality of servers at the data center includes selecting a foreign agent for each of the plurality of servers, sending a registration request message comprising a request for a network address assignment from each server to the selected foreign agent, determining a home agent to service the registration request, forwarding the registration request message to the selected home agent, allocating a network address to the server at the home agent, and providing the network address to the server. Borella at [0020] FIG. 5 is a block diagram illustrating a message flow for dynamic server registration using Mobile IP according to one exemplary embodiment; Borella at [0040] According to an exemplary embodiment, when the server 324 boots, the server 324 sends a Mobile IP Registration Request ("MIP RRQ") message 352 to the local foreign agent 318. Borella at [0040] The MIP RRQ message 352 generated at the server 324 includes a request for a dynamic network address assignment. Specifically, the MIP RRQ message 352 includes a home IP address (an IP address to be assigned to the server 324) and a home agent IP address of 0.0.0.0 indicating that the server 324 requests a dynamic network assignment and a dynamic home agent assignment. The MIP RRQ message 352 also includes a network access identifier ("NAI") of the server 324, such as an identifier "server-a.xyz.com," which is a globally unique name. To the extent that Borella does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Borella , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	

Claims	Prior Art
	Borella at [0012] One exemplary method includes providing a plurality of foreign agents at a data center, and using a Mobile Internet Protocol to dynamically assign a network address to each of the plurality of servers at the data center. The method further includes providing a plurality of home agents at a content owner site. In such an embodiment, using the Mobile Internet Protocol to dynamically assign a network address to each of the plurality of servers at the data center includes selecting a foreign agent for each of the plurality of servers, sending a registration request message comprising a request for a network address assignment from each server to the selected foreign agent, determining a home agent to service the registration request, forwarding the registration request message to the selected home agent, allocating a network address to the server at the home agent, and providing the network address to the server. Borella at [0040] According to an exemplary embodiment, when the server 324 boots, the server 324 sends a Mobile IP Registration Request ("MIP RRQ") message 352 to the local foreign agent 318. Borella at [0040] The MIP RRQ message 352 generated at the server 324 includes a request for a dynamic network address assignment. Specifically, the MIP RRQ message 352 includes a home IP address (an IP address to be assigned to the server 324) and a home agent IP address of 0.0.0.0 indicating that the server 324 requests a dynamic network assignment and a dynamic home agent assignment. The MIP RRQ message 352 also includes a network access identifier ("NAI") of the server 324, such as an identifier "server-a.xyz.com," which is a globally unique name. To the extent that Borella does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Borella , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Borella at [0012] One exemplary method includes providing a plurality of foreign agents at a data center, and using a Mobile Internet Protocol to dynamically assign a network address to each of the plurality of servers at the

Claims	Prior Art
	data center. The method further includes providing a plurality of home agents at a content owner site. In such an embodiment, using the Mobile Internet Protocol to dynamically assign a network address to each of the plurality of servers at the data center includes selecting a foreign agent for each of the plurality of servers, sending a registration request message comprising a request for a network address assignment from each server to the selected foreign agent, determining a home agent to service the registration request, forwarding the registration request message to the selected home agent, allocating a network address to the server at the home agent, and providing the network address to the server. Borella at [0040] The MIP RRQ message 352 generated at the server 324 includes a request for a dynamic network address assignment. Specifically, the MIP RRQ message 352 includes a home IP address (an IP address to be assigned to the server 324) and a home agent IP address of 0.0.0.0 indicating that the server 324 requests a dynamic network assignment and a dynamic home agent assignment. The MIP RRQ message 352 also includes a network access identifier ("NAI") of the server 324, such as an identifier "server-a.xyz.com," which is a globally unique name. To the extent that Borella does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Borella , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Borella at [0007] A Domain Name System (DNS) based solution involves a DNS database that includes proximity and load information of potential servers. See RFCs 1034 and 1035. The client device's local DNS includes this information and either proxies or forwards the client's DNS request to the appropriate data center's DNS server based on the proximity or load information. The DNS server then replies with an individual server's IP address that is selected based on the proximity and load information Borella at [0036] The DNS server 306 is coupled to the IP network 30 via a communication link 308, the home agent 310 is coupled to the IP network 30 via a communication link 312, the foreign agent 318 is coupled to the IP

Claims	Prior Art
	network 30 via a communication link 320, the AAA server 314 is coupled to the IP network 30 via a communication link 316, and the server 324 is coupled to the IP network 30 via a communication link 326. Borella at [0037] The content owner then advertises a route to this address through the IP network 30. Thus, when the client device 302 sends a request for the DNS name, the DNS name is resolved to the virtual IP address, and the request is routed to the content owner server site 328, i.e., the DNS 306. Borella at [0061] Referring to FIG. 8, the client device 302 sends a DNS request message 502 to the DNS 306, and the DNS request message 502 includes the domain name "server.xyz.com." When the DNS 306 receives the DNS request message 502, the DNS 306 uses one or more load balancing algorithms to select a server for processing the request. The DNS 306 then sends a DNS response message 504 to the client device 302. According to an exemplary embodiment, the DNS response message 504 includes the IP address of the server 324, i.e., the IP address of "149.112.150.101." To the extent that Borella does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Borella , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art
	Borella at [0031] When the MN 124 receives the agent advertisement message from the foreign agent 140, it will register itself with the foreign agent 140 and with its home agent 110. When the MN 124 registers with the foreign agent 140, the foreign agent 140 will create a routing table entry for the network address 1.0.0.7 of the MN 124. The home agent 110 will also create a routing table entry for the MN 124 that includes the network address 4.0.0.101 for the foreign agent 140 to which the MN 124 is presently connected. Borella at [0040] According to an exemplary embodiment, when the server 324 boots, the server 324 sends a Mobile IP Registration Request ("MIP RRQ") message 352 to the local foreign agent 318. To the extent that Borella does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Borella , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[35.0] 35. The virtual network manager of claim 30 further comprising a join module configured to receive a join request from the agent associated with the device to indicate that the device is connected for data communication within the virtual network, the join module further configured to receive a leave request from the agent associated with the device to indicate that the device will be disconnected from data communication within the virtual network.	Borella at [0027] A mobile node also listens for "agent advertisement" messages from other gateway routers. When a mobile node receives an agent advertisement message indicating that it is now on a foreign subnet, it registers with the foreign gateway router or "foreign agent" and its home agent. The registration with the home agent indicates that the mobile node is away from "home" (i.e., away from its home subnet). The registration with the foreign agent allows the mobile node to receive data on the foreign subnet. To the extent that Borella does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Borella , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art
[37.0] 37. The virtual network manager of claim 35 wherein the join module is further configured to maintain data to associate a virtual network address with a device in the virtual network.	Borella at [0031] When the MN 124 receives the agent advertisement message from the foreign agent 140, it will register itself with the foreign agent 140 and with its home agent 110. When the MN 124 registers with the foreign agent 140, the foreign agent 140 will create a routing table entry for the network address 1.0.0.7 of the MN 124. The home agent 110 will also create a routing table entry for the MN 124 that includes the network address 4.0.0.101 for the foreign agent 140 to which the MN 124 is presently connected. Borella at [0047] Further, in such an embodiment, the home agent 310 may send an update message the DNS 306 every time the server 324 reregisters, and, if the DNS 306 does not receive one or more such updates, the DNS 306 may mark the server 324 as inactive. Borella at [0054] It should be understood that the DNS 306 may store information associating the client device 302 sending application requests with a server, i.e., the server 324 in FIG. 6, selected to process a first application request. In such an embodiment, the DNS 306 may be configured to receive a subsequent DNS request from the client device 302, retrieve the stored information associating the client device 302 with the selected server 324, and then return the IP address of the server 324 so that the subsequent application request is sent to the server 324 identified in the stored information. To the extent that Borella does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Borella , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-19

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent Application Publication No. 2003/0084162 ("Johnson")

Johnson titled "Managing peer-to-peer access to a device behind a firewall" to Bruce L. Johnson et al. was filed on October 31, 2001 and published on May 1, 2003.[1] Johnson is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Johnson does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Johnson or combine the teachings of Johnson with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Johnson at [0037] The system includes a private network 302 such as a Local Area Network (LAN), an organizational intranet, and so on. To the extent that Johnson does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Johnson , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Johnson at [0001] The described subject matter relates to establishing peer-to-peer communication sessions between networked computing devices. Johnson at [0037] The private network is coupled across a different network 310 to a name/address server 312. Johnson at claim 14 An address server as recited in claim 12, wherein the information comprises a public network address, a public port number, a private network address, a private port number, and a substantially unique device name. To the extent that Johnson does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Johnson , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Johnson at [0016] The receiving device receives a request from a different device for the information needed to communicate with the firewall protected device. Responsive to receiving the request and responsive to determining that the communication pathway to the device is valid, the device access information is communicated to the different device. The different device can use the communicated device access information to initiate a peer-to-peer communication session with the firewall protected device. Johnson at [0039] The name/server manages access to the peer-to-peer communication pathways to substantially ensure that any device routes to a device 308 that are provided by the name/address server are valid, regardless of whether network addresses behind the firewall 304 are dynamically assigned to network devices 308, expired and placed into a pool of unused addresses, or reassigned to different devices 308. Johnson at [0041] To accomplish this, the name/address server includes a processor 314 that is coupled to a system memory 315. Johnson at [0057] An outside device 326 can also register its corresponding services (e.g., printing services, digital sender services, and so on) with the address server 312. To the extent that Johnson does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Johnson , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	Johnson at [0057] An outside device 326 can also register its corresponding services (e.g., printing services, digital sender services, and so on) with the address server 312.

Claims	Prior Art
	To the extent that Johnson does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Johnson , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Johnson at [0029] The device name is a name such as an alias name, the device's serial no., LAN ID, and so on, that uniquely identifies the device 308 that is sending the datagram packet (the "source device") to the address server 312. Johnson at [0057] An outside device 326 can also register its corresponding services (e.g., printing services, digital sender services, and so on) with the address server 312. To the extent that Johnson does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Johnson , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Johnson at [0007] The server identifies both the public and private network addresses of each message sent to it. Johnson at [0028] The header includes a private address 112 and a private port number 114. The private address is the device's actual configured address, and the private port is a port that is allocated by the device for use during input/output (I/O) with a device that is not in the same network as the source device.

Claims	Prior Art
	Johnson at [0031] Each block of data 210 includes data such as a substantially unique name 212 (see, the device name 110 of FIG. 1), a private network address 214, a private port no. 316, a public network address 218, a public port no. 220, and a time stamp 322. Johnson at [0033] The public address 218 and public port 220 correspond to the address and port supplied by a gateway device during its address translation procedures in response to an outbound message. Johnson at [0039] The name/address server 312 provides the "outside device" 326 with the ability to initiate and establish peer-to-peer communications with a device 308 that is behind the firewall 304. Johnson at [0043] For instance: (a) the private device can be configured with the address server's network address upon installation in the private network 302; (b) the name of the address server may be known and conventional DNS lookups can be used to acquire the IP address; and/or (c) the address server may be configured via BOOTP parameters or additional DHCP parameters when the device is auto-configured on the network. Johnson at [0053] Responsive to receiving a request from an outside device 326, the address server 312 communicates a predetermined portion of each device access data entry 210 of FIG. 2 in the device access-mapping table 200 to the outside device. The predetermined portion includes the public network address 218 (the public address is the gateway device's 304 network address) the public port number 320, and optionally the substantially unique name 212 of the corresponding private device 308. Johnson at [0066] Such an entry includes a substantially unique name for the private device that communicated the datagram (block 410), the private device's private address, the private device's private port, a public address of the NAT Gateway device, and a public port of the Gateway device. Johnson at claim 14 An address server as recited in claim 12, wherein the information comprises a public network address, a public port number, a private network address, a private port number, and a substantially unique device name.

Claims	Prior Art
	To the extent that Johnson does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Johnson , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-20

Invalidity of U.S. Patent No. 7,949,785 in View of PCT Patent Application Publication No. WO2003088625 ("Volz")

Volz titled "Method and system for enabling connections into networks with local address realms" to Bernie Volz was filed on March 28, 2003, has a priority date of April 8, 2002, and published on October 23, 2003.[1] Volz is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Volz does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Volz or combine the teachings of Volz with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Volz at 8 Yet another object of the invention is to provide a gateway resource manager for supporting enhanced scalability and/or efficient and flexible outside-realm initiated communication. To the extent that Volz does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Volz , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Volz at 1 The present invention generally relates to network communication and more particularly to the issue of providing connectivity between networks of different address realms. Volz at 3 The gateway 30 thus provides the necessary means for connecting hosts within the private realm to the public realm, especially when the number of private hosts is larger than the number of available public IP addresses. Volz at 9 Preferably, an outside node that wants to initiate a new connection to an inside node prepares a user-resource identifier query, such as a DNS (Domain Name Server) query or equivalent, for transfer to a central allocation or addressing mechanism, which determines inside-realm network address information based on the inside node identifier included in the query. The identifier query from the outside node further includes predetermined connection information such as outside node address information and/or

Claims	Prior Art
	inside node port information. This predetermined connection information from the identifier query may then be used as a basis for identifying an outside-realm gateway address suitable for establishing a new dynamic gateway connection state for a flow between the outside node and the inside node through the gateway. The new dynamic gateway connection state is subsequently established at least partly based on the identified outside-realm gateway address, the predetermined connection information from the identifier query and the inside-realm network address information. In this way, the invention provides support for dynamic outside-realm initiated communication. To the extent that Volz does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Volz , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Volz at 9 Preferably, an outside node that wants to initiate a new connection to an inside node prepares a user-resource identifier query, such as a DNS (Domain Name Server) query or equivalent, for transfer to a central allocation or addressing mechanism, which determines inside-realm network address information based on the inside node identifier included in the query. The identifier query from the outside node further includes predetermined connection information such as outside node address information and/or inside node port information. This predetermined connection information from the identifier query may then be used as a basis for identifying an outside-realm gateway address suitable for establishing a new dynamic gateway connection state for a flow between the outside node and the inside node through the gateway. The new dynamic gateway connection state is subsequently established at least partly based on the identified outside-realm gateway address, the predetermined connection information from the identifier query and the inside-realm network address information. In this way, the invention provides support for dynamic outside-realm initiated communication. Volz at 11 The gateway resource manager finally sends a state set-up request to the gateway, which establishes the dynamic gateway connection state.

Claims	Prior Art
	To the extent that Volz does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Volz , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	Volz at 11 The gateway resource manager finally sends a state set-up request to the gateway, which establishes the dynamic gateway connection state. To the extent that Volz does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Volz , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Volz at 24 For example, consider public host B that wants to connect to two private nodes A1 and A2. Host B initiates a connection set-up towards node A1, and the gateway resource manager finds a free outside gateway address, say aOG 1. Simultaneously, host B initiates a connection set-up towards node A2, and the gateway resource manager attempts to identify a useful outside gateway address. Volz at 31 1. Node A1 wants to initiate a communication flow towards node B, which belongs to the outside realm. Node A1 sends a request including the destination network address aOB and port number pB to the gateway (GW). In the IETF RSIP framework (RFC3103), this message could be the "ASSIGN_REQUEST_RSA-IP". 2. The gateway sends a request to the gateway resource manager (GRM). In process X, the GRM manager allocates a network address aOG from the gateway address pool.

Claims	Prior Art
	The gateway typically supplies the destination address and/or destination port as predetermined connection information to the gateway resource manager. In this example, both destination address information and port information are included in the request. To the extent that Volz does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Volz , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Volz at 9 Preferably, an outside node that wants to initiate a new connection to an inside node prepares a user-resource identifier query, such as a DNS (Domain Name Server) query or equivalent, for transfer to a central allocation or addressing mechanism, which determines inside-realm network address information based on the inside node identifier included in the query. The identifier query from the outside node further includes predetermined connection information such as outside node address information and/or inside node port information. This predetermined connection information from the identifier query may then be used as a basis for identifying an outside-realm gateway address suitable for establishing a new dynamic gateway connection state for a flow between the outside node and the inside node through the gateway. The new dynamic gateway connection state is subsequently established at least partly based on the identified outside-realm gateway address, the predetermined connection information from the identifier query and the inside-realm network address information. In this way, the invention provides support for dynamic outside-realm initiated communication. Volz at 13 Efficient use of user-identifier queries, such as DNS-queries, in the overall process of establishing outside-realm initiated connections. Volz at 21 The inventors have recognized that a flexible and dynamic mechanism for enabling outside-realm initiated communication is obtained by letting a user-resource identifier query, such as a DNS-query or equivalent query, initiated from an outside node trigger the set-up of a connection state in the gateway.

Claims	Prior Art
	Volz at 31 1. Node A1 wants to initiate a communication flow towards node B, which belongs to the outside realm. Node A1 sends a request including the destination network address aOB and port number pB to the gateway (GW). In the IETF RSIP framework (RFC3103), this message could be the "ASSIGN_REQUEST_RSA-IP". 2. The gateway sends a request to the gateway resource manager (GRM). In process X, the GRM manager allocates a network address aOG from the gateway address pool. The gateway typically supplies the destination address and/or destination port as predetermined connection information to the gateway resource manager. In this example, both destination address information and port information are included in the request. Volz at 37 As before, the gateway resource manager assigns a public gateway address to the inside node based on predetermined connection information included in the DNS-query and sends a DNS-reply including the assigned address to the requesting outside node. Volz at 58 a user-resource identifier query that includes an inside node identifier as well as predetermined connection information including at least one of outside node address information and inside node port information; To the extent that Volz does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Volz , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-21

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent Application Publication No. 2004/0098507 ("Thubert")

Thubert titled "Mobile IP registration supporting port identification" to Pascal Thubert et al. was filed on November 20, 2002 and published on May 20, 2004.[1] Thubert is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

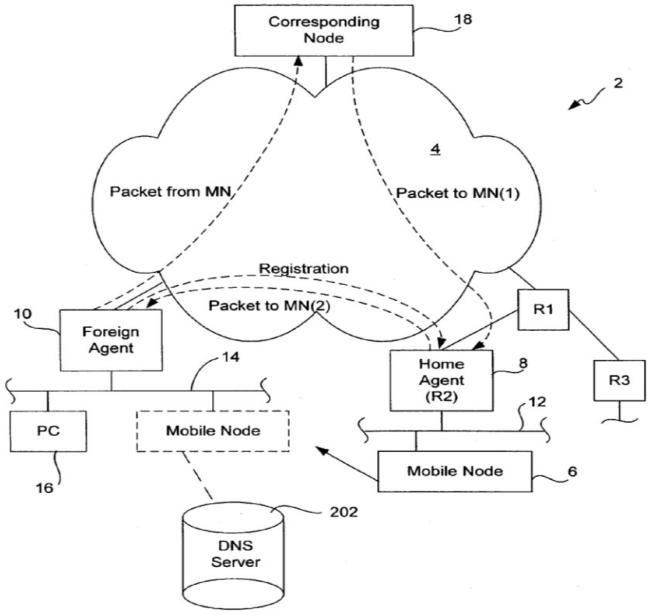
To the extent Plaintiff alleges Thubert does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Thubert or combine the teachings of Thubert with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

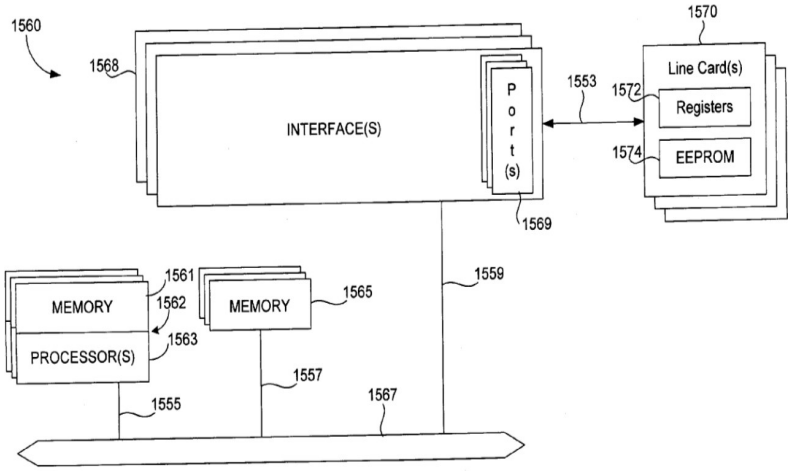
As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

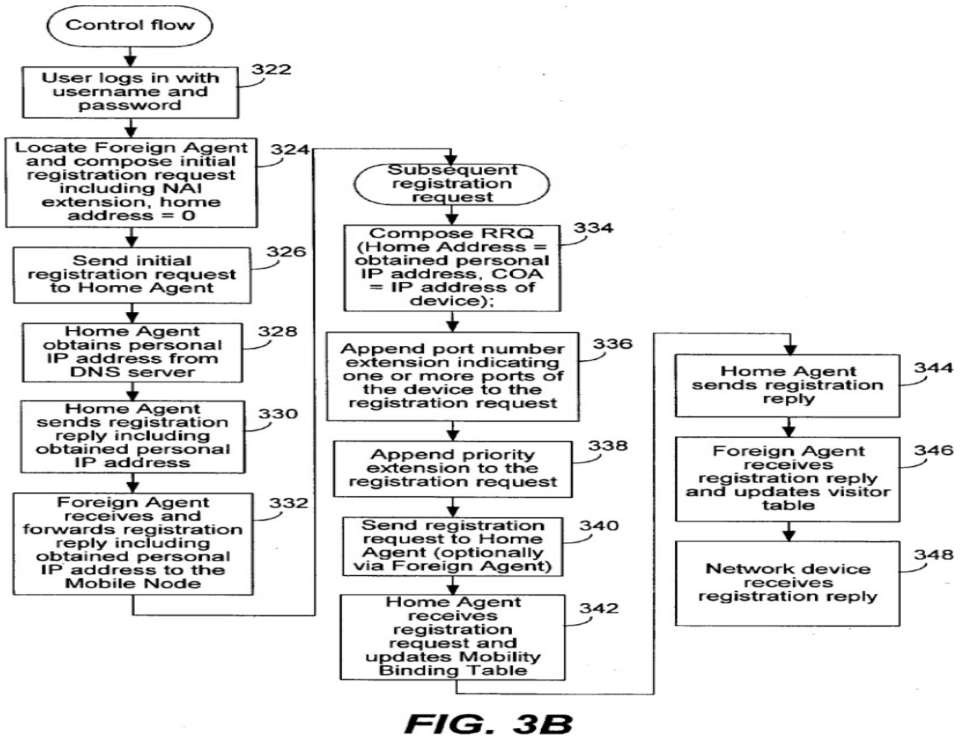
[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Thubert at [0003] The present invention relates to Mobile IP network technology. More particularly, the present invention relates to Mobile IP registration. To the extent that Thubert does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Thubert , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Thubert at [0010] If mobile node 110 wanted to send a message to a correspondent node 155 from its new location, the mobile node 110 would forward a packetized output message 160 through the foreign agent 120 over the Internet 105 to the correspondent node 155 according to standard Internet protocols. Thubert at [0010] The packets of the message from the correspondent node 155 are forwarded over the Internet 105 to the router R2 125 and ultimately to the home agent 115. Thubert at [0060] For example, they can be implemented in an operating system kernel, in a separate user process, in a library package bound into network applications, on a specially constructed machine, or on a network interface card. Thubert at [0061]

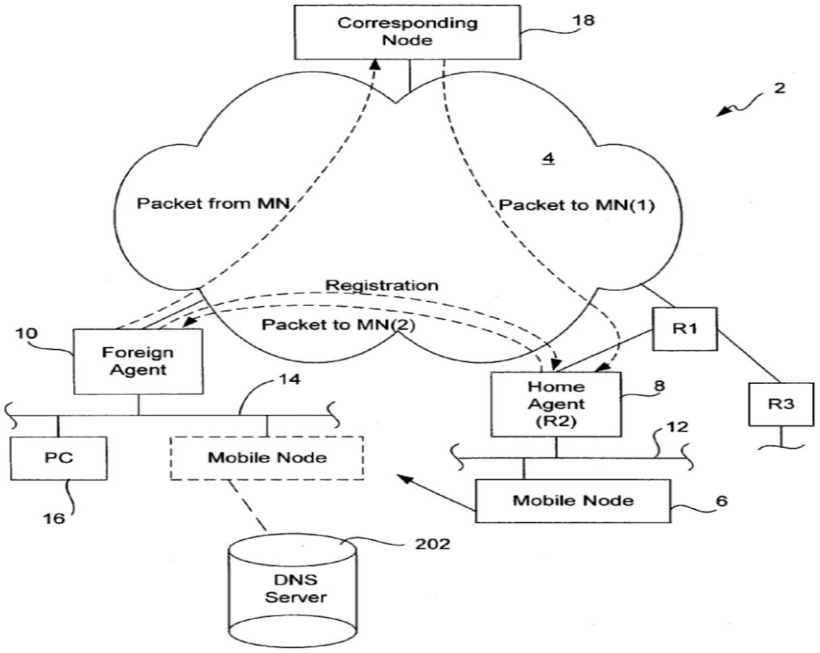
Claims	Prior Art
	Such network devices may have multiple network interfaces including frame relay and ISDN interfaces, for example. To the extent that Thubert does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Thubert , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Thubert at Figure 2  *FIG. 2* Thubert at Figure 11

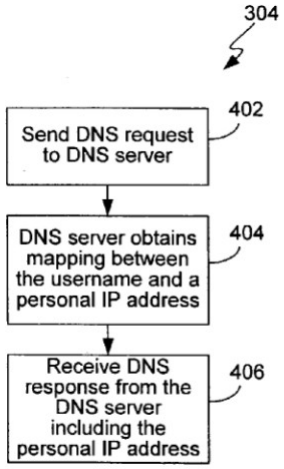
Claims	Prior Art
	 FIG. 11 Thubert at [0009] When mobile node 110 engages with remote network segment 140, it composes a registration request for the home agent 115 to bind the mobile node's 110 current location with its home location. Foreign agent 120 then relays the registration request 150 to home agent 115. During the registration process, the home agent 115 and the mobile node 110 may then negotiate the conditions of the mobile node's 110 attachment to foreign agent 120. For example, the mobile node 110 may request a registration lifetime of 5 hours, but the home agent 115 may grant only a 3 hour period. When the negotiation is successfully completed, home agent 115 updates an internal “mobility binding table” which links the mobile node's 110 current location via its care-of address (e.g., a co-located care-of address or the foreign agent's IP address) to the identity (e.g., home address) of the mobile node 110. Further, if the mobile node 110 registered via foreign agent 120, the foreign agent 120 updates an internal “visitor table” which specifies the mobile node address, home agent address, etc. The home agent's 115 association between a mobile node's home base IP address, its current care-of address, and the remaining lifetime of that association is referred to as a binding. Thubert at [0016] The present invention provides methods and apparatus for processing registration requests. Thubert at [0024]

Claims	Prior Art
	FIGS. 3A and 3B are control flow diagrams illustrating methods of processing a registration request in accordance with various embodiments of the invention. Thubert at [0061] A software or software/hardware hybrid implementation of the techniques of this invention may be implemented on a general-purpose programmable machine selectively activated or reconfigured by a computer program stored in memory. Thubert at [0062] Referring now to FIG. 11, a network device 1500 suitable for implementing the techniques of the present invention includes a master central processing unit (CPU) 1505, interfaces 1510, memory 1515 and a bus 1520. Thubert at [0063] CPU 1505 may include one or more processors such as those from the Motorola family of microprocessors or the MIPS family of microprocessors. In an alternative embodiment, the processor is specially designed hardware for controlling the operations of network device 1500. Thubert at [0066] Regardless of network device's configuration, it may employ one or more memories or memory modules (such as, for example, the memory 1515) configured to store data, program instructions for the general-purpose network operations and/or other information relating to the functionality of the techniques described herein. To the extent that Thubert does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Thubert , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	Thubert at [0043] The registration request is then sent to a Home Agent associated with the user at block 312. Thubert at [0062]

Claims	Prior Art
	The CPU 1505 preferably accomplishes all these functions under the control of software including an operating system (e.g. Windows NT), and any appropriate applications software. To the extent that Thubert does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Thubert , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Thubert at Figure 3B  FIG. 3B Thubert at [0016]

Claims	Prior Art
	A registration request is then composed, where the registration request comprises a home address field including the obtained personal IP address. The registration request is then sent to a Home Agent associated with the username. The registration request that is sent to the Home Agent includes a NAI identifying the username, a care-of address, and a home address field including the obtained IP address. Thubert at [0039] A registration request is then composed at block 306. Specifically, the registration request includes a home address field including the obtained IP address. In addition, the registration request further includes a care-of address field including a care-of address. In accordance with various embodiments, the care-of address is an IP address associated with the network device (e.g., assigned to an interface of the network device). In addition, the registration request preferably includes a NAI extension to the registration request including the NAI (e.g., when registration is performed in Ipv4). By providing the NAI to the Home Agent, the Home Agent may map the NAI to its dynamically assigned personal IP address, enabling the Home Agent to correctly forward data packets or server requests addressed to the username to the appropriate IP address. To the extent that Thubert does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Thubert , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Thubert at Figure 2

Claims	Prior Art
	 FIG. 2 Thubert at Figure 4

Claims	Prior Art
	 FIG. 4 Thubert at [0052] Specifically, when the DNS server receives the DNS request, it resolves the username via a mapping table 502. The DNS server obtains a mapping between the username 504 and a personal IP address 506. The DNS server then provides the personal IP address in a DNS response as described above. To the extent that Thubert does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Thubert , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art
	Thubert at [0007] The Mobile IP process and environment are illustrated in FIG. 1. A Mobile IP environment 100 includes the Internet (or a WAN) 105 over which a mobile node 110 can communicate via mediation by a home agent 115 or a foreign agent 120 To the extent that Thubert does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Thubert , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-22

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent Application Publication No. 2003/0016636 ("Tari")

Tari titled "Communication system, mobile unit database server, mobile radio router, charging method, and vehicle mounted router and agent server therewith" to Kazuyoshi Tari et al. was filed on July 15, 2002 and published on January 23, 2003.[1] Tari is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Tari does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Tari or combine the teachings of Tari with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Tari at [0002] The present invention relates to a mobile wireless communication based on the Internet, and relates in particular to a mobile unit database server that allows a mobile radio router and terminals that can be connected to the mobile radio router to communicate with a host connected to the Internet, a mobile radio router and a method of charging for the services. To the extent that Tari does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Tari , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Tari at [0004] For this reason, it is necessary to provide an interface (gateway portion) between the Internet network and mobile communication networks to perform address conversion. Tari at [0017] The present invention is provided in view of the background information described above, and an object is to provide a network that enables to connect to the Internet without going through a gateway, namely, a mobile radio router, a mobile unit database server, and a radio server operating, which allows a plurality of terminals to process together as one group. To the extent that Tari does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Tari , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Tari at [0038] The present invention also provides a radio communication program containing steps for execution of computing tasks in a communication system comprising: at least one radio server connected to a network; not less than one radio base station connected to the radio server; and a mobile radio router controlling at least not less than one terminal; wherein, when the registration and authentication process of said terminal that is connected to said mobile radio router is carried out, a computer executes the steps of: carrying out authentication based on information in the terminal managing database that stores the identifiers of said terminal, and assigning IP addresses to said terminal based on address information obtained from the home mobile unit database server of said terminal. Tari at [0101] Mobile unit information database 63 associates and registers communicable mobile radio router 5a and terminals 51˜5a as mobile unit information, and stores the results so as to manage the mobile radio router 5a and terminals 51˜5a. To the extent that Tari does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Tari , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	Tari at [0048] The present invention also provides an agent server for use in a communication system comprising: at least one radio server connected to a network; not less than one radio base station connected to the radio server, a vehicle-mounted router according to claims 19 to 23 controlling at least not less than one terminal and connected to said radio base station by radio transmission; and the agent server connected to the network wherein the agent server is provided with: an authentication information database having pre-recorded client authentication information for authenticating a connection client, and vehicle identifier address information for authenticating a vehicle having the vehicle-mounted router; and control devices

Claims	Prior Art
	for performing an authentication process by receiving client information for identifying the connection client and vehicle identifier address information for identifying the vehicle which are sent from the terminal by way of said vehicle-mounted router, and by processing the received client information and vehicle identifier address information according to the client authentication information and the vehicle identifier address information contained in the authentication information database. To the extent that Tari does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Tari , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Tari at [0038] The present invention also provides a radio communication program containing steps for execution of computing tasks in a communication system comprising: at least one radio server connected to a network; not less than one radio base station connected to the radio server; and a mobile radio router controlling at least not less than one terminal; wherein, when the registration and authentication process of said terminal that is connected to said mobile radio router is carried out, a computer executes the steps of: carrying out authentication based on information in the terminal managing database that stores the identifiers of said terminal, and assigning IP addresses to said terminal based on address information obtained from the home mobile unit database server of said terminal. Tari at [0111]–[0113] Also, control section C52 requests a registration to the radio server that manages those radio base stations that operate in the same area as itself, and stores the information received at the time of authentication in the mobile unit database 54. Also, the control section C52 controls registration operation of registration authentication so that determination of whether or not to authenticate a terminal is carried out according to information stored in the pre-registered mobile unit database 54. Also, at the time of registering a terminal, the control section C52 synthesizes and assigns the IP address from the identifier of the terminal and the network address received from the home mobile unit database server when issuing IP addresses to terminals at the time of registration.

Claims	Prior Art
	Tari at [0127] When the radio server RSA receives the registration request and the identifier form the mobile radio router 5a, the identifier sent from the mobile radio router 5a and the random value generated by itself and its own FQDN are sent to the mobile unit database server MDBSA, and the request for authentification is carried out (step S2). Further, the radio server RSA sends the identifier of the mobile radio router 5a and the random value generated by itself to the mobile radio router 5a, and the request for authentification is carried out (step S3). Tari at [0128] When the mobile unit database server MDBSA receives the random value sent from the radio server RSA, the identifier of the mobile radio router 5a, the FQDN of the radio server RSA, and the request for authentification, it is determined whether or not the identifier of the received mobile radio router 5a is stored in the mobile unit information database 63, and a the same time, a calculation is carried out using the identifier of the mobile radio router 5a and the received random value To the extent that Tari does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Tari , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Tari at [0137] Next, address resolution process to be performed after the completion of the process for registration and authentication will be explained with reference to the flowchart shown in FIG. 9. Here, address resolution relates to a process for deducing an IP address that corresponds to FQDN. Tari at [0139] Because the DNS server DNSA does not itself manage FQDN of host C42, the address resolution request which includes the FQDN of the host C42 that is the object of address resolution and the translated IP address of the terminal 51 that is the origin of the request is sent to the DNS server 41. (step S14).

Claims	Prior Art
	Tari at [0140] DNS server 41, upon receiving the address resolution request from the DNS server DNSA, reads the IP address corresponding to the FQDN of the target host C42 for address resolution, and the address resolution response that sets the IP address of the host C42 that was read and the translated IP address of the terminal 51 that is the origin of the request is sent to the DNS server DNSA (step S15). Tari at [0141] DNS server DNSA sends the address resolution response that sets the IP address of host C42 received from DNS server 41 and the translated IP address of the terminal 51 that was the origin of the request to the radio server RSA (step S16). To the extent that Tari does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Tari , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[37.0] 37. The virtual network manager of claim 35 wherein the join module is further configured to maintain data to associate a virtual network address with a device in the virtual network.	Tari at [0108]

Claims	Prior Art
	The mobile unit managing database 34 stores the managing information for each radio communication group in the case that there are a plurality of radio communication groups. In addition, the mobile unit managing database 34 is provided for each radio server. Moreover, the structure of the radio server RSB is identical to the structure to the radio server RSA described above. To the extent that Tari does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Tari , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-23

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent No. 8,107,483 ("Dunk")

Dunk titled "IPv4-to-IPv6 address transitioning methods and apparatus for systems that push information to wireless communication devices" to Craig Allan Dunk was filed on March 12, 2010, has a priority date of November 27, 2002, and issued on January 31, 2012.[1] Dunk is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Dunk does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Dunk or combine the teachings of Dunk with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Dunk at 1:22–24 The present application relates generally to IPv4-to-IPv6 address transitioning methods and apparatus for systems that “push” information to wireless communication devices. To the extent that Dunk ’ does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Dunk ’, those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Dunk at 3:48–51 The present application relates to two related methodologies for facilitating the communication of user information from a plurality of host computers to a plurality of wireless communication devices through a serving network. Dunk at 3:66–4:6 Referring now to FIG. 1, an illustration of a computer system 100 which highlights the components involved in the first technical aspect of the present application, which relates to the serving network's “front end”, is shown. Computer system 100 generally includes a host system 102 and a serving network 106 which communicate through a public network 104 to provide communications for at least one wireless communication device 130. Dunk at 4:11–18 Communications between host system 102 and serving network 106 may be routed through a conventional firewall 112 and public network 104, which may be the Internet. In the present embodiment,

Claims	Prior Art
	host network 110 is an IPv4-addressed network. Public network 104 (e.g. the Internet) may also be an IPv4-addressed network or have components thereof (e.g. routers) that are not yet equipped to handle IPv6 addressing. Dunk at 12:15–18 Also shown in FIG. 4 is a home agent 418 of a virtual home network 416 in serving network 404. Core network 420 and virtual home network 416 are able to communicate information to each other through a router 428. To the extent that Dunk ’ does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Dunk ’, those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Dunk at 3:66–4:6 Referring now to FIG. 1, an illustration of a computer system 100 which highlights the components involved in the first technical aspect of the present application, which relates to the serving network's “front end”, is shown. Computer system 100 generally includes a host system 102 and a serving network 106 which communicate through a public network 104 to provide communications for at least one wireless communication device 130. Dunk at 6:42–54 In particular, a first level of SSH authentication allows any host to connect to a SSH server as long as the password of the account at the server is known. This procedure will encrypt traffic sent via SSH, but it does not in itself provide a strong mechanism to authenticate the host. A second level of SSH authentication relies on a security key mechanism: a key pair is created and the public key is provided to the server. When connecting to an SSH server, the host sends a request to the server for authentication with use of the security keys. The server looks up the public key in a remote home directory at the server and compares both keys. It then sends an encrypted “challenge” to the host, which is decrypted at the host with the private key and sent back to the server. Dunk at 8:14–19

Claims	Prior Art
	Such methods may be embodied in a computer program product which includes a computer storage medium and computer instructions stored on the computer storage medium, where the computer instructions can be executed to perform the methods. Dunk at 13:48–51 In most modern communication devices, controller 506 will be embodied as a central processing unit or CPU running operating system software which is stored in a mobile device memory component (not shown). To the extent that Dunk ’ does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Dunk ’, those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	Dunk at 4:19–21 Host computer 108 has an application program for receiving user information, processing the user information, and displaying the processed information to the end-user. Dunk at 12:15–18 Also shown in FIG. 4 is a home agent 418 of a virtual home network 416 in serving network 404. Core network 420 and virtual home network 416 are able to communicate information to each other through a router 428. To the extent that Dunk ’ does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Dunk ’, those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Dunk at 8:20–36

Claims	Prior Art
	In the following description of the flow diagram of FIG. 2, FIGS. 1-2 are referred to in combination. Host computer 108 initiates a connection to serving network 106 by sending a tunnel request to network entry point device 114 (step 202 of FIG. 2). Network entry point device 114 receives this request from host computer 108 and, in response, selects one of the plurality of tunnel servers 116 to which to direct communications with host computer 108 (step 204 of FIG. 2). In the present example, network entry point device 114 particularly selects tunnel server 118 to which to direct communications with host computer 108. Tunnel server 118 receives this tunnel request from host computer 108 through network entry point device 114 (step 206 of FIG. 2). In response to receiving the tunnel request, tunnel server 118 provides any necessary authentication (step 207 of FIG. 2) and establishes a tunnel connection with host computer 108 (step 208 of FIG. 2) assuming the authentication is successful. Dunk at 9:28–40 The update to the local routing table at tunnel server 118 involves storing an IP address of tunnel server 118 in association with a destination IP address to host computer 108. Tunnel server 118 also uses the DRP to broadcast this updated routing table information to all other tunnel servers 116 and router 126. The broadcasting of updated routing table information may be performed in response to identifying a newly established connection, or it may be performed periodically over time, or both. When updated routing table information is broadcasted from tunnel server 118 and received at other tunnel servers 116 and router 126, other tunnel servers 116 and router 126 update their own local routing tables to reflect the new tunnel connection. Dunk at 15:3–6 Not only are IP addresses handed out, but all related configuration settings like the subnet mask, default router, DNS server, that are required to make TCP/IP work correctly. To the extent that Dunk ’ does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Dunk ’, those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private	Dunk at 8:42–46

Claims	Prior Art
network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Accordingly, network entry point device 114 performs selection of a tunnel server in step 204 in a manner such that all tunnel connections are distributed amongst all tunnel servers 116 (using, for example, a round robin DNS or a local director). Dunk at 12:2–5 Similarly, wireless network 432 and serving network 404 are able to communicate information to each other through a router 430 which is also coupled to core network 420. Dunk at 15:3–6 Not only are IP addresses handed out, but all related configuration settings like the subnet mask, default router, DNS server, that are required to make TCP/IP work correctly. Dunk at 15:9–15 This request is delivered to router 430 of serving network 404 which handles the request. In response, router 430 identifies or determines a temporary IPv6 address to assign to wireless device 408 (step 610 of FIG. 6). Router 430 sends a response message to wireless device 408 which includes this newly assigned temporary IPv6 address (step 612 of FIG. 6). Dunk at 16:23–26 When wireless device 408 leaves IPv4 wireless network 432 and returns to IPv6 wireless network 406, for example, no longer needed are the temporary IPv4 address, the temporary IPv6 address, and Care-Of-Addressing. To the extent that Dunk ’ does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Dunk ’, those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art
	To the extent that Dunk ’ does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Dunk ’, those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

[35.0] 35. The virtual network manager of claim 30 further comprising a join module configured to receive a join request from the agent associated with the device to indicate that the device is connected for data communication within the virtual network, the join module further configured to receive a leave request from the agent associated with the device to indicate that the device will be disconnected from data communication within the virtual network.	Dunk at 9:28–40 The update to the local routing table at tunnel server 118 involves storing an IP address of tunnel server 118 in association with a destination IP address to host computer 108. Tunnel server 118 also uses the DRP to broadcast this updated routing table information to all other tunnel servers 116 and router 126. The broadcasting of updated routing table information may be performed in response to identifying a newly established connection, or it may be performed periodically over time, or both. When updated routing table information is broadcasted from tunnel server 118 and received at other tunnel servers 116 and router 126, other tunnel servers 116 and router 126 update their own local routing tables to reflect the new tunnel connection. Dunk at 16:23–26 When wireless device 408 leaves IPv4 wireless network 432 and returns to IPv6 wireless network 406, for example, no longer needed are the temporary IPv4 address, the temporary IPv6 address, and Care-Of-Addressing. To the extent that Dunk ’ does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Dunk ’, those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[37.0] 37. The virtual network manager of claim 35 wherein the join module is further configured to maintain data to associate a virtual network address with a device in the virtual network.	

Claims	Prior Art
	Dunk at 9:28–40 The update to the local routing table at tunnel server 118 involves storing an IP address of tunnel server 118 in association with a destination IP address to host computer 108. Tunnel server 118 also uses the DRP to broadcast this updated routing table information to all other tunnel servers 116 and router 126. The broadcasting of updated routing table information may be performed in response to identifying a newly established connection, or it may be performed periodically over time, or both. When updated routing table information is broadcasted from tunnel server 118 and received at other tunnel servers 116 and router 126, other tunnel servers 116 and router 126 update their own local routing tables to reflect the new tunnel connection. Dunk at 16:23–26 When wireless device 408 leaves IPv4 wireless network 432 and returns to IPv6 wireless network 406, for example, no longer needed are the temporary IPv4 address, the temporary IPv6 address, and Care-Of-Addressing. To the extent that Dunk ’ does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Dunk ’, those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-24

Invalidity of U.S. Patent No. 7,949,785 in View of PCT Patent Application Publication No. WO2002015014 ("Wootton")

Wootton titled "Pseudo addressing" to Bruce C. Wootton et al. was filed on July 31, 2001 and published on February 21, 2002.[1] Wootton is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(b) and (e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Wootton does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Wootton or combine the teachings of Wootton with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Wootton at Abstract The present invention also allows for communication to be initiated by a source entity (302) outside a private network that is directed to a destination entity (312) with a private address within the private network. Wootton at 1 The present invention is directed to a system for communicating with entities on a network. To the extent that Wootton does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Wootton , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Wootton at 5 One embodiment of the present invention includes a method for communicating. Wootton at 11 Figure 3 also shows a Gateway connected to Internet 138 and to private network 162. Wootton at 14 When an application requests resolution of a domain name, the network software for that entity will contact a DNS server in order to obtain the authoritative resource record that indicates the IP address for the host associated with the domain name. This resource record will be returned to the network software for the entity X in step 352. In step 354, the network software for entity X will chose a pseudo address. In

Claims	Prior Art
	one embodiment, the pseudo address is chosen randomly. In other embodiments, a list of pseudo addresses to choose from can be prestored. In the current example, the pseudo address is in the same format as typical IPv4 Internet addresses. In step 356, the pseudo address chosen in step 354, the IP address resolved in step 352 and the domain name from step 350 are all stored in a table. In step 358, the pseudo address is provided to the application software. To the extent that Wootton does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Wootton , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Wootton at 5 The present invention can be accomplished using hardware, software, or a combination of both hardware and software. The software used for the present invention is stored on one or more processor readable storage media including a hard disk drive, CD-ROM, optical disk, floppy disk, RAM, ROM or other suitable storage device. In alternative embodiments, some or all of the software can be replaced by dedicated hardware including custom integrated circuits, gate arrays, FPGAs, PLDs, and special purpose computers. To the extent that Wootton does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Wootton , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	Wootton at 5 The present invention can be accomplished using hardware, software, or a combination of both hardware and software. The software used for the present invention is stored on one or more processor readable storage media including a hard disk drive, CD-ROM, optical disk, floppy disk, RAM, ROM or other suitable storage device. In alternative embodiments, some or all of the software can be replaced by

Claims	Prior Art
	dedicated hardware including custom integrated circuits, gate arrays, FPGAs, PLDs, and special purpose computers. Wootton at 12 Figure 5 is a block diagram illustrating the use of pseudo addresses. A pseudo address in its most generic form is an identification of an entity that is different from the entity's actual address. In one embodiment, the pseudo address is a random address chosen to identify a particular entity. The randomly chosen address is not the entity's actual address. In one alternative, the pseudo address is in the format for IPv4 addresses. Depicted in Figure 5 are application 302 and network software 304, which are both running on a single computer (in one embodiment). Network software 304 pertains to software at the transport layer, network layer, and other layers. Application software 302 pertains to software at the application layer. Figure 5 also shows application 312 and network software 314, both of which are running on a single computer (in one embodiment). In one example, application 302 communicates with application 312. According to the current invention, both applications 302 and 312 use pseudo addresses to communicate with each other. However, in actuality, application 302 communicates with application 312 using network software 304 and network software 314. Application 302 identifies application 312 to network software 304 using a domain name and a pseudo address. Network software 304 communicates with network software 314 using Internet addresses (e.g. global addresses). In one example, network software 304 and network software 314 use IPv4 addresses. Other address formats can also be used, including IPv6. Application 312 identifies application 302 to network software 314 using a pseudo address and a domain name Wootton at 14 When an application requests resolution of a domain name, the network software for that entity will contact a DNS server in order to obtain the authoritative resource record that indicates the IP address for the host associated with the domain name. This resource record will be returned to the network software for the entity X in step 352. In step 354, the network software for entity X will chose a pseudo address. In one embodiment, the pseudo address is chosen randomly. In other embodiments, a list of pseudo addresses to choose from can be prestored. In the current example, the pseudo address is in the same format as typical IPv4 Internet addresses. In step 356, the pseudo address chosen in step 354, the IP address resolved in step 352 and the domain name from step 350 are all stored in a table. In step 358, the pseudo address is provided to the application software. To the extent that Wootton does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Wootton , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Wootton at 5 The method includes the step of receiving a request to communicate from a first application on a source entity. The request includes a first pseudo address used by the source entity to identify a destination entity. The first pseudo address is used to access a global address. A quantity of information is sent from the source toward the destination entity using the global address. One embodiment of the present invention further includes the steps of receiving the quantity of information at the destination entity, providing at least a subset of the quantity of information to an application on the destination entity and providing the pseudo address to the application on the destination entity. Wootton at 12 Figure 5 is a block diagram illustrating the use of pseudo addresses. A pseudo address in its most generic form is an identification of an entity that is different from the entity's actual address. In one embodiment, the pseudo address is a random address chosen to identify a particular entity. The randomly chosen address is not the entity's actual address. In one alternative, the pseudo address is in the format for IPv4 addresses. Depicted in Figure 5 are application 302 and network software 304, which are both running on a single computer (in one embodiment). Network software 304 pertains to software at the transport layer, network layer, and other layers. Application software 302 pertains to software at the application layer. Figure 5 also shows application 312 and network software 314, both of which are running on a single computer (in one embodiment). In one example, application 302 communicates with application 312. According to the current invention, both applications 302 and 312 use pseudo addresses to communicate with each other. However, in actuality, application 302 communicates with application 312 using network software 304 and network software 314. Application 302 identifies application 312 to network software 304 using a domain name and a pseudo address. Network software 304 communicates with network software 314 using Internet addresses (e.g. global addresses). In one example, network software 304 and network software 314 use IPv4 addresses. Other address formats can also be used, including IPv6. Application 312 identifies application 302 to network software 314 using a pseudo address and a domain name To the extent that Wootton does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Wootton , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Wootton at 2 The Internet uses a Domain Name System (DNS) to convert the domain name to an IP address. Wootton at 14 When an application requests resolution of a domain name, the network software for that entity will contact a DNS server in order to obtain the authoritative resource record that indicates the IP address for the host associated with the domain name. This resource record will be returned to the network software for the entity X in step 352. In step 354, the network software for entity X will chose a pseudo address. In one embodiment, the pseudo address is chosen randomly. In other embodiments, a list of pseudo addresses to choose from can be prestored. In the current example, the pseudo address is in the same format as typical IPv4 Internet addresses. In step 356, the pseudo address chosen in step 354, the IP address resolved in step 352 and the domain name from step 350 are all stored in a table. In step 358, the pseudo address is provided to the application software. Wootton at 17–18 In step 748, the pseudo address chosen for entity B, global address for entity B and local address for entity B are added to the table. Wootton at 20–21 In the embodiment described above with respect to Figure 13, the system first performed a regular domain name resolution to determine the global IP address of the Gateway and then contacts the Gateway to get the local IP address of entity B. In one embodiment, entity A can get the local IP address of entity B from the Gateway by doing a domain name resolution request directly with the Gateway. In another embodiment, entity A can obtain a local address for entity B from the Gateway by using an ICMP echo request. In another embodiment, rather than acquiring the global TP address of the Gateway and the local IP address of entity B in two separate steps, the domain name resolution process can be altered such that when resolving the domain name for entity B (e.g. B.com), the resource record will return with both the global IP address for the Gateway and the local IP address of entity B. In yet another embodiment, the

Claims	Prior Art
	first domain name resolution returns the global address of the Gateway and a second domain name resolution sent to a different entity (other than the Gateway) returns the local address. To the extent that Wootton does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Wootton , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-25

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent Application Publication No. 2002/0038382 ("Ryu")

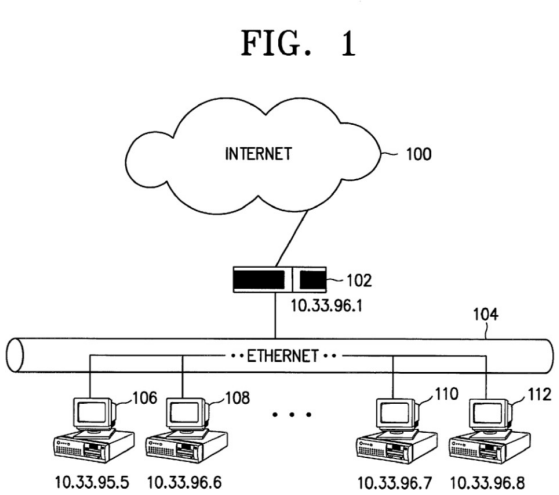
Ryu titled "Method for using a unique IP address in a private IP address domain" to Young-Gyu Ryu was filed on August 30, 2001 and published on March 28, 2002.[1] Ryu is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(b) and (e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

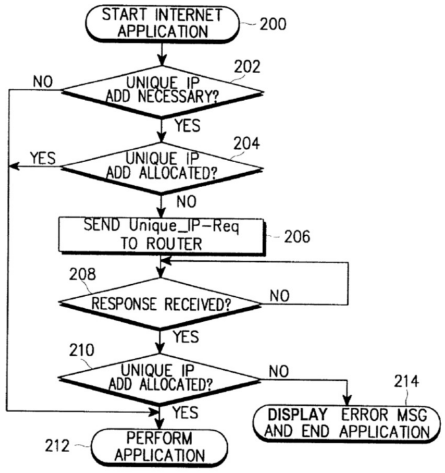
To the extent Plaintiff alleges Ryu does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Ryu or combine the teachings of Ryu with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

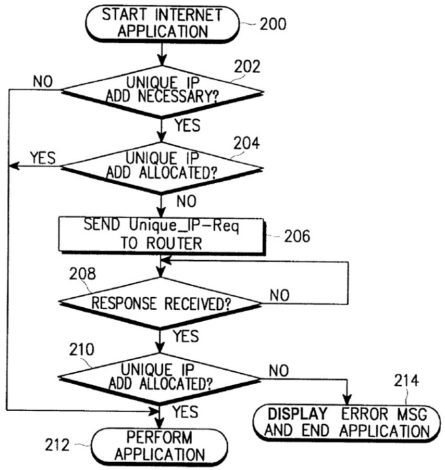
As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Ryu at Figure 1  Ryu at [0022] Specifically, FIG. 1 shows an exemplary case where the router 102 is connected to the Internet 100 and has an IP address of 10.33.96.1 and the LAN devices 106-112 have their IP addresses of 10.33.96.5, 10.33.96.6, 10.33.96.7, and 10.33.96.8, respectively. To the extent that Ryu does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Ryu , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Ryu at Abstract In the method, the LAN device sends a unique IP address request to the router to be allocated a unique IP address, when the LAN device needs a unique IP address. The LAN device returns the allocated unique IP address to the router after using the allocated unique IP address. Ryu at Figure 3  FIG. 2 Ryu at [0013] To achieve the above and other objects, a method for using an IP address on a LAN including a router and a LAN device is provided. In the method, when the LAN device needs a unique IP address, it sends a unique IP address request to the router to be allocated a unique IP address. The LAN device returns the allocated unique IP address to the router after using the allocated unique IP address. To this end, the LAN device supports a procedure for requesting a unique IP address and returning the unique IP address after using it. Further, the router supports a procedure for allocating a unique IP address upon receipt of a

Claims	Prior Art
	unique IP address request from the LAN device, and registering a unique IP address returned from the LAN device for future use. To the extent that Ryu does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Ryu , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Ryu at Abstract In the method, the LAN device sends a unique IP address request to the router to be allocated a unique IP address, when the LAN device needs a unique IP address. The LAN device returns the allocated unique IP address to the router after using the allocated unique IP address. Ryu at Figure 3  FIG. 2 Ryu at [0003]

Claims	Prior Art
	The present invention relates generally to a method for using an IP (Internet Protocol) address, and in particular, to a method for using a unique IP address in a private IP address domain. Ryu at [0013] To achieve the above and other objects, a method for using an IP address on a LAN including a router and a LAN device is provided. In the method, when the LAN device needs a unique IP address, it sends a unique IP address request to the router to be allocated a unique IP address. The LAN device returns the allocated unique IP address to the router after using the allocated unique IP address. To this end, the LAN device supports a procedure for requesting a unique IP address and returning the unique IP address after using it. Further, the router supports a procedure for allocating a unique IP address upon receipt of a unique IP address request from the LAN device, and registering a unique IP address returned from the LAN device for future use. Ryu at [0023] The static unique IP addresses are statically (or fixedly) allocated to the associated LAN devices 106-112. Ryu at [0025] When the LAN device desires to use the unique IP address, there are two possible methods as follows. One method is to statically allocate a unique IP address to one device. To the extent that Ryu does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Ryu , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[35.0] 35. The virtual network manager of claim 30 further comprising a join module configured to receive a join request from the agent associated with the device to indicate that the device is connected for data communication within the virtual network, the join module further configured to receive a leave request from the agent associated with the device to indicate that the device will be	Ryu at Abstract Disclosed is a method for using a unique IP address for a limited period of time in a device connected to a LAN which includes a router and a LAN device. Ryu at Abstract

Claims	Prior Art
disconnected from data communication within the virtual network.	In the method, the LAN device sends a unique IP address request to the router to be allocated a unique IP address, when the LAN device needs a unique IP address. The LAN device returns the allocated unique IP address to the router after using the allocated unique IP address. Ryu at [0013] To achieve the above and other objects, a method for using an IP address on a LAN including a router and a LAN device is provided. In the method, when the LAN device needs a unique IP address, it sends a unique IP address request to the router to be allocated a unique IP address. The LAN device returns the allocated unique IP address to the router after using the allocated unique IP address. To this end, the LAN device supports a procedure for requesting a unique IP address and returning the unique IP address after using it. Further, the router supports a procedure for allocating a unique IP address upon receipt of a unique IP address request from the LAN device, and registering a unique IP address returned from the LAN device for future use. To the extent that Ryu does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Ryu , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[37.0] 37. The virtual network manager of claim 35 wherein the join module is further configured to maintain data to associate a virtual network address with a device in the virtual network.	Ryu at Abstract Disclosed is a method for using a unique IP address for a limited period of time in a device connected to a LAN which includes a router and a LAN device. Ryu at Abstract In the method, the LAN device sends a unique IP address request to the router to be allocated a unique IP address, when the LAN device needs a unique IP address. The LAN device returns the allocated unique IP address to the router after using the allocated unique IP address. Ryu at [0013] To achieve the above and other objects, a method for using an IP address on a LAN including a router and a LAN device is provided. In the method, when the LAN device needs a unique IP address, it sends a unique IP address request to the router to be allocated a unique IP address. The LAN device returns the

Claims	Prior Art
	allocated unique IP address to the router after using the allocated unique IP address. To this end, the LAN device supports a procedure for requesting a unique IP address and returning the unique IP address after using it. Further, the router supports a procedure for allocating a unique IP address upon receipt of a unique IP address request from the LAN device, and registering a unique IP address returned from the LAN device for future use. To the extent that Ryu does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Ryu , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-26

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent Application Publication No. 2003/0172184 ("Kong")

Kong titled "Network-connecting apparatus and method for providing direct connections between network devices in different private networks" to Won-keun Kong was filed on March 6, 2003 and published on September 11, 2003.[1] Kong is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

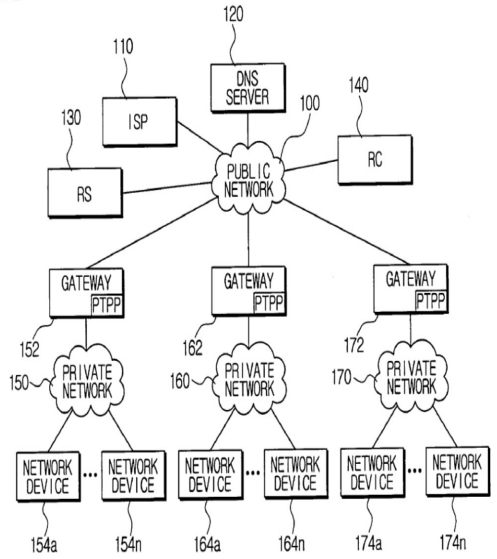
To the extent Plaintiff alleges Kong does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Kong or combine the teachings of Kong with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	**Prior Art**
[30.0] 30. A virtual network manager, comprising:	Kong at [0015] A connection management part is provided for connecting to the second private network based on the acquired public IP address and forming a tunnel with the second private network. To the extent that Kong does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kong , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Kong at Figure 1

Claims	Prior Art
	FIG.1  Kong at [0015] A connection management part is provided for connecting to the second private network based on the acquired public IP address and forming a tunnel with the second private network. Kong at [0044] The public network interface part 205 is connected to the internet 100 through the ISP 110 and provides an interface with a broadband access network. Kong at [0048] The DNS part 230 then registers the public IP address allocated from the ISP 110 and correlates it with a unique domain name of the private network 150. The public IP address and the name are registered with the domain name server 120 connected to the internet 100. Kong at [0051] In order for two hosts to communicate in a virtual network, physical network addresses should be obtained from the respective hosts and then matched with IP addresses.

Claims	Prior Art
	To the extent that Kong does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kong , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Kong at [0017] More specifically the address change part comprises a number acquisition part for receiving a unique tunnel number from the second private network, said tunnel number being allocated to the tunnel. Kong at [0018] In another specific enhancement, a registration request part is provided for requesting a registration of a public IP address from a domain name management server, said public address being allocated from a unique domain name and internet service providing device through a communication network. To the extent that Kong does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kong , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	Kong at [0018] In another specific enhancement, a registration request part is provided for requesting a registration of a public IP address from a domain name management server, said public address being allocated from a unique domain name and internet service providing device through a communication network. To the extent that Kong does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kong , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Claims	Prior Art
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Kong at [0017] More specifically the address change part comprises a number acquisition part for receiving a unique tunnel number from the second private network, said tunnel number being allocated to the tunnel. To the extent that Kong does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kong , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Kong at [0015] In another specific enhancement, the direct connection part comprises an address acquisition part for acquiring a public IP address of the second private network from a domain name management server based on a domain name of the second network device received from the first network device, and for acquiring a private IP address of the second network device based on the received domain name. Kong at [0039] The domain name server 120 links an internet address to a name. The address is denoted by numbers, whereas the name is usually denoted by characters familiar to users. That is, the domain name server 120 stores and provides, to users, relations between continuously varying public IP addresses of the gateways 152, 162, and 172 and unique character names. Because of this, the users can easily recognize the public IP addresses of the gateways 152, 162, and 172 to which the users request a connection based on the names of corresponding gateways 152, 162, and 172. Kong at [0048] The DNS part 230 operates as a DNS server for the private network 150.

Claims	Prior Art
	To the extent that Kong does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kong , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-27

Invalidity of U.S. Patent No. 7,949,785 in View of United States Patent Application Publication No. 2004/0037260 ("Kakemizu")

Kakemizu titled "Virtual private network system" to Mitsuaki Kakemizu was filed on August 7, 2003, has a priority date of August 9, 2002, and published on February 26, 2004.[1] Kakemizu is prior art to the '785 Patent under at least Pre-AIA 35 U.S.C. §§ 102(e) as to claims 30, 35, and 37 (the "Asserted Claims") of the '785 Patent, as discussed in detail in the chart that follows.

To the extent Plaintiff alleges Kakemizu does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Kakemizu or combine the teachings of Kakemizu with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, such as the other prior art identified in Defendant's invalidity contentions.

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc.*, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id.* at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id.* at 421.

Defendant reserves the right to amend or supplement this claim chart at a later date as more fully set forth in the Invalidity Contentions.

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

Claims	Prior Art
[30.0] 30. A virtual network manager, comprising:	Kakemizu at [0002] The present invention relates to a system for implementing a virtual private network between an organization network, which is a private network, and a network accommodating a mobile node, a mobile node, a home agent, and a communications controlling method. To the extent that Kakemizu does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kakemizu , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Kakemizu at [0011] FIGS. 1 and 2 explain a method making a seamless communication with an organization network via a public network such as the Internet, according to a conventional technology. Kakemizu at [0130] A network device in the preferred embodiments according to the present invention is arranged as a security gateway within the communications carrier network or the enterprise network 11. To the extent that Kakemizu does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kakemizu , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.2] a memory and a processor to implement a register module configured to register devices in a	

Claims	Prior Art
virtual network, the register module further configured to:	Kakemizu at [0177] The MA VPN controlling unit 41 sets the VPN information extracted by the AAA protocol processing unit 44 in the VPN information cache 46. If the cache is set or updated for a reference made by a mobile IP protocol processing unit to be described later, a flag indicating that the cache is updated is set in a shared memory. S To the extent that Kakemizu does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kakemizu , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.3] receive a registration request from an agent associated with a device;	Kakemizu at [0027] A home agent according to the present invention is a home agent enabling a communication between a mobile node and a node connected to a private network according to a mobile IP. The home agent comprises: a unit establishing a virtual private network between the mobile node and the home agent; a unit authenticating an access of the mobile node; and a unit notifying the mobile node of information about the virtual private network. Kakemizu at [0126] Additionally, FIG. 6 shows the structure of a location registration request (Reg.Request) message of mobile IP, FIG. 7 shows the structure of an authentication request (AMR: AA Mobile Node Request) message of DIAMETER, and FIG. 8 shows the structure of a home agent registration request (HAR: Home Agent MIP Request) message of the DIAMETER. To the extent that Kakemizu does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kakemizu , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being	

Claims	Prior Art
identified to other devices in the virtual network by the virtual network address; and	Kakemizu at [0030] The first mobile node comprises: an obtaining unit obtaining information of a network to which the mobile node itself currently belongs; and a controlling unit performing a control to transmit a location registration request message to a private address of a router that manages a location of the mobile node if the obtained information of the network indicates a private network, to transmit a location registration request message to a global address of the router if the obtained information of the network indicates a predetermined communications carrier network, or to transmit a location registration request message including a request to set up a communications path having high secrecy to the global address of the router in other cases. Kakemizu at [0113] Either a private address or a global address may be used as an address format within an organization. However, an address used in the present invention is called a "private address" in the sense that it can be communicated only within an organization network. At the same time, an address that can be communicated in a public network is called a "global address". Accordingly, in mobile IP protocol, the "private address" is a home address, which corresponds to a fixed first address, whereas the "global address" is a care-of-address, which corresponds to a second address that can be used to communication. Kakemizu at [0116] The authentication server 18 is a server group that has a name (hereinafter referred to as an AAA) used by the IETF, and performs authentication, authorization, and accounting. The authentication server 18 is configured by an AAA protocol controlling unit extracting VPN information of a user who makes authentication request from a VPN database 17, and notifying an HA 19 of the VPN information with an AAA protocol 21, and an AAA VPN controlling unit extracting VPN information in units of users, and determining a VPN path, in addition to the above described functions. In FIG. 3, the authentication server 18 is arranged in a communications carrier network or an enterprise network 11. AAA protocol 21 Kakemizu at [0120] Each of the VPN data instances is composed of a profile number (ProfileNumber), which is an identifier uniquely representing VPN information, a network identifier (Nai) of a user, a VPN sharing index (vpnshare) indicating whether a security relationship shared by security gateways or a security relationship specific to a user is used, a VPN type (vpnkind), an IP address (destaddr) of a communication destination terminal, an upstream QoS class (upclass), a downstream QoS class (downclass), an upstream

Claims	Prior Art
	SPI (upSPI) used by IPSec, a downstream SPI (downSPI) used by IPSec, and an IP port number (portNumber) used for UDP encapsulation. Kakemizu at [0130] The home agent (HA 19) is an agent that possesses a private address assigned by the enterprise network 12 as a home address. A packet transmitted to the home agent (HA 19), whose destination is the home address of the mobile node (MN 16), is encapsulated and transmitted to a care-of-address of the mobile node (MN 16), which corresponds to the home address. This address correspondence is managed by a table called a mobility binding. Additionally, the HA 19 notifies the mobile node (MN 16) of VPN information by setting a service profile in a location registration reply (Reg.Reply) message. The HA 19 in the preferred embodiments according to the present invention also serves as a VPN gateway function for performing IPSec encapsulation after UDPinIP encapsulation so as to support an IPinIP, IPSec, or an IPSec+UDP packet for which NAT (Network Address Translation) and NAPT (Network Address Port Translation) are performed, and comprises an MA protocol controlling unit (mobile agent protocol controlling unit) analyzing VPN information notified with the AAA protocol and the IP protocol, and an MA VPN controlling unit (Mobile Agent VPN controlling unit) setting up a tunnel at a security level specified by a network kernel based on analyzed VPN information. proxy correspondent node (PCN) 20 To the extent that Kakemizu does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kakemizu , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Kakemizu at [0028] The first router comprises: a unit detecting a care-of-address or a domain of a location registration request transmitted from the mobile node; and a communications controlling unit causing a communication between the mobile node and the node to be made via the router with a communications protocol having low secrecy between the mobile node and the router if a detected care-of-address or domain indicates a network that can guarantee secrecy of a communication, or with a communications protocol having high secrecy between the mobile node and the router if the care-of-address indicates a network that cannot fully guarantee the secrecy of the communication.

Claims	Prior Art
	Kakemizu at [0113] Either a private address or a global address may be used as an address format within an organization. However, an address used in the present invention is called a “private address” in the sense that it can be communicated only within an organization network. At the same time, an address that can be communicated in a public network is called a “global address”. Accordingly, in mobile IP protocol, the “private address” is a home address, which corresponds to a fixed first address, whereas the “global address” is a care-of-address, which corresponds to a second address that can be used to communication. To the extent that Kakemizu does not disclose any elements of this claim limitation, it would have been obvious to a person having ordinary skill in the art to combine, with the teachings of Kakemizu , those elements, as known to them or as disclosed by other prior art identified in the cover pleading.

Exhibit 785-28 to Southwest's Preliminary Invalidity Contentions

Cisco VPN Routers and Software

Cisco VPN Routers and Software was known in this country, used in public, sold or offered for sale no later than 2002, as demonstrated by at least the materials cited herein or the testimony of knowledgeable witnesses and corroborating documents.[1] Specifically, based on information currently available, Cisco VPN Routers and Software was (1) known or used in this country before the alleged invention of the claimed subject matter of the asserted claims, (2) in public use or on sale in this country more than one year before the filing date of the patent, or (3) invented by another who did not abandon, suppress, or conceal, before the alleged invention of the claimed subject matter of the asserted claim. Thus, Cisco VPN Routers and Software anticipates the asserted claims under at least 35 U.S.C. §§ 102(a) or (g).

At least the following documents describe the relevant functionality disclosed by Cisco VPN Routers and Software:

- Cisco 7100 Series VPN Routers;
- Cisco VPN 3000 Series Concentrator Data Sheet;
- CiscoWorks VPN/Security Management Solution 2.1;
- Cisco IPSec Encryption;
- Cisco 2600 and 3600 VPN Router Bundles;
- Cisco SEP Installation Guide;
- CiscoWorks VMS 2.2.

Based on information currently available, the persons or entities involved in the conception and diligent reduction to practice of the ideas in Cisco VPN Routers and Software include at least persons involved in the Cisco VPN Routers and Software and their testing and commercial use, including at least Cisco Systems. The conception of Cisco VPN Routers and Software occurred at least as early as 2002.

Southwest offers this invalidity chart based on its current understanding of Cisco VPN Routers and Software. Southwest's investigation of this system is ongoing. Southwest may rely on materials produced by Plaintiffs or by third parties regarding Cisco VPN Routers and Software and

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

related software to demonstrate invalidity of the asserted claims. Southwest has not had an opportunity to review source code for Cisco VPN Routers and Software. Southwest may rely on source code, when available, implementing the functionality described herein, or other source code implementing substantially similar functionality, to demonstrate the invalidity of the asserted claims.

To the extent Plaintiff alleges Cisco VPN Routers and Software does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify Cisco VPN Routers and Software or combine the teachings of Cisco VPN Routers and Software with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, including but not limited to:

- U.S. Patent Application Publication No. 2003/0028671 ("Mehta");
- Huitema, C., Network Working Group Request for Comment (RFC): 1383, entitled An Experiment in DNS Based IP Routing ("RFC 1383");
- U.S. Patent No. 6,970,941 ("Caronni I");
- U.S. Patent No. 7,814,228 ("Caronni II"); and/or
- U.S. Patent No. 6,766,371 ("Hipp").

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc*., "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id*. at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id*. at 421.

Claims	Cisco VPN Routers and Software
[30.0] A virtual network manager, comprising:	To the extent the preamble is limiting, Cisco VPN Routers and Software discloses: The preamble requires a virtual network manager that manages VPN functionality. The Cisco VPN systems describe themselves as integrated VPN solutions that manage site-to-site connectivity, which aligns with the concept of a virtual network manager. "The Cisco 7100 Series VPN Router is a high-end, integrated VPN solution melding high-speed, industry-leading routing with a comprehensive suite of advanced site-to-site VPN services." (Cisco 7100 Series VPN Routers). "The Cisco VPN 3000 Concentrator Series is a best-of-breed, remote-access VPN solution for enterprise-class deployment." (Cisco VPN 3000 Series Concentrator Data Sheet). "CiscoWorks VMS provides Web-based tools for configuring, monitoring, and troubleshooting enterprise virtual private networks (VPNs), firewalls, and network host-based intrusion detection systems (IDSs)." (CiscoWorks VPN/Security Management Solution 2.1) To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Mehta and/or Caronni I. Mehta and Caronni I concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, use of virtual networks was well-known and common. Modifying this reference with Mehta's and/or Caronni I's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with that of Mehta or Caronni I for example, to incorporate this limitation. *See, e.g.*,

	"Methods and systems for providing two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP, are provided. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporarily use by a requesting device on a public network to communicate with a wireless device on a wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more data repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the public network to send data to the wireless device." Mehta at Abstract. "The present invention relates to methods and systems for initiating communication with wireless devices and, in particular, to methods and systems for initiating communication with a device on a private network from a device on a public network to achieve virtual end-to-end connectivity." Mehta at [0002]. "Embodiments of the present invention provide computer- and network-based methods and systems for two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internetwork, such as the Internet, to initiate communication with and send data to wireless devices connected to a private wireless

	network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporary use by a requesting device on an external public network to communicate with a wireless device on a private wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. The mapping of a temporary public address to the private address of a wireless device is maintained and updated transparently by the AMPS using routing tables and other mapping data structures." Mehta at [0011]. "To insure a greater degree of security, according to one embodiment, the AMPS maintains a Time to Live (TTL) parameter with each address mapping. In this way, once the TTL value indicates that the mapping has expired, the AMPS can destroy the mapping, and also any connection. Further, in some embodiments, the AMPS also puts a timestamp in its own mapping tables. After some timeout period based upon the timestamp, the AMPS can destroy the mapping, thereby forcing a new mapping to be initiated on a periodic basis." Mehta at [0015]. "Embodiments of the present invention provide computer- and network-based methods and systems for two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices." Mehta at [0024]. "The second approach used to implement the AMPS supports full bi-directional communication through point-to-point connections established, for example, using TCP/IP protocol. (Note that these same techniques also support connection-less UDP bi-directional communication). The second approach can be implemented by providing a modified

implementation of a standard UDP or TCP/IP function, "GetHostByName." The GetHostByName API allows a string designation to identify the designated device and returns an IP address data structure. Alternatively, to increase the level of security provided, the AMPS can implement a specialized API to return the dynamically allocated public address that (now) corresponds to the requested wireless device. A disadvantage of the specialized API approach is that the device on the public network (or other device that wishes to obtain a connection to the wireless device) needs to include specialized code in the application on the requesting device."

Mehta at [0041].

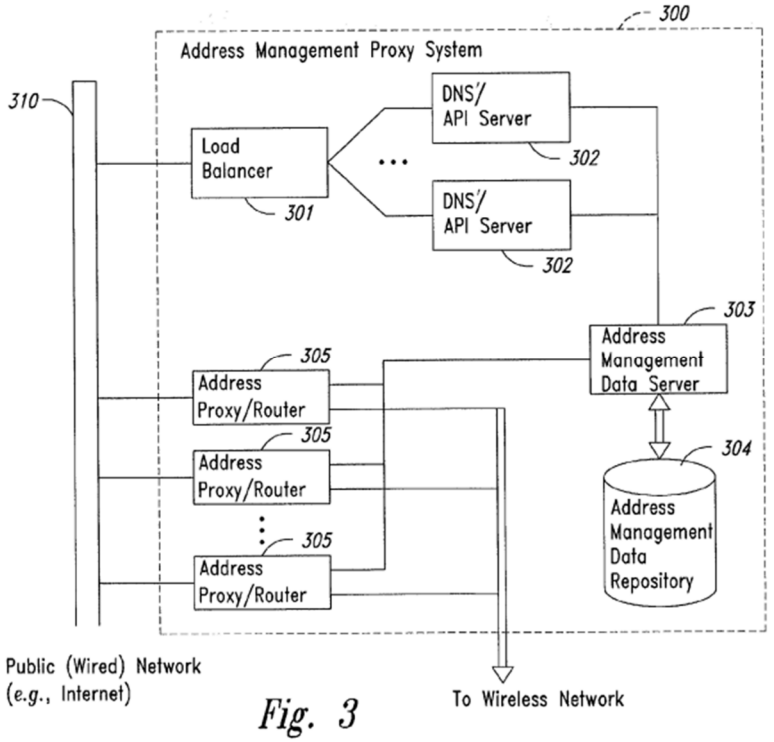


Fig. 3

Mehta at FIG. 3.

	"Methods and systems consistent with the present invention establish a virtual network on top of current IP network naming schemes. The virtual network uses a separate layer to create a modification to the IP packet format that is used to separate network behavior from addressing. As a result of the modification to the packet format, any type of delivery method may be assigned to any address or group of addresses. The virtual network also maintains secure communications between nodes, while providing the flexibility of assigning delivery methods independent of the delivery addresses." Caronni I at Abstract. "The present invention relates generally to data processing systems and, more particularly, to a private network using a public-network infrastructure." Caronni I at 1:57–59. "Methods and systems consistent with the present invention overcome the shortcomings of existing networks by establishing a "Supernet," which is a private network that uses components from a public-network infrastructure. A Supernet allows an organization to utilize a public-network infrastructure for its enterprise network so that the organization no longer has to maintain a private network infrastructure; instead, the organization may have the infrastructure maintained for them by one or more service providers or other organizations that specialize in such connectivity matters. As such, the burden of maintaining an enterprise network is greatly reduced. Moreover, a Supernet is not geographically restrictive, so a user may plug their device into the Internet from virtually any portal in the world and still be able to use the resources of their private network in a secure and robust manner." Caronni I at 4:36-52. "Supernets also provide heterogeneous addressing functionality. The Supernet uses a separate layer that isolates address names of nodes from addressing schemes and delivery schemes. The Supernet contains a modification to the IP packet format that can be used to separate

	network behavior from addressing. As a result of the modification, any delivery scheme may be assigned to any address, or group of addresses." Caronni I at 4:53-59. *See also* Caronni I at Claim 1, FIGS. 3, 5.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	Cisco VPN Routers and Software discloses "a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address." *See, e.g.*, The claim requires a network interface configured for data communication within a virtual network. Cisco VPN systems use Fast Ethernet ports to facilitate such data communication, demonstrating the existence of a network interface. "Dual autosensing 10/100BaseT Fast Ethernet ports." (Cisco 7100 Series VPN Routers) "Cisco VPN 3015-3080—Three auto-sensing, full-duplex 10/100BaseTX Fast Ethernet (public/untrusted, private/trusted and DMZ)." (Cisco VPN 3000 Series Concentrator Data Sheet) "Apply crypto map to interface. The commands below apply the crypto map to the interface." (Cisco IPSec Encryption) "Dual 10/100 Ethernet Router with 2 WIC Slots & 1 NM Slot." (Cisco 2600 and 3600 VPN Router Bundles) To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Mehta and/or Caronni I. Mehta and Caronni I concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, use of virtual networks was well-known

	and common. Modifying this reference with Mehta's and/or Caronni I's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with that of Mehta or Caronni I for example, to incorporate this limitation. *See, e.g.*, "The present invention relates to methods and systems for initiating communication with wireless devices and, in particular, to methods and systems for initiating communication with a device on a private network from a device on a public network to achieve virtual end-to-end connectivity." Mehta at [0002]. "Embodiments of the present invention provide computer- and network-based methods and systems for two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internetwork, such as the Internet, to initiate communication with and send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporary use by a requesting device on an external public network to communicate with a wireless device on a private wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. The mapping of a temporary public address to the private address of a wireless device is maintained and updated transparently by the AMPS using routing tables and other mapping data structures." Mehta at [0011]. "Although the techniques of the AMPS are generally applicable to any a wired device communicating with a wireless device, the phrase "public network" (or "wired network") is used generally to imply any type of internetworked environment including a public network

	or a backbone that is somewhere down the line connected to one or more private or public networks. In addition, although the examples described herein often refer to the Internet, one skilled in the art will recognize that the concepts and inventions described are applicable to other forms and embodiments of internetworking, including, for example ATM type networks. Thus, techniques of the present invention can also be used by one device on a first wireless network to communicate with another wireless device on a second network—each device ends up communicating with the Address Proxy/Router of the other network. This scenario is feasible because each wireless network (or its carrier infrastructure) is connected to a proxy/router that is also connected (via a public address) to a public network. In addition, although a public network is sometimes also referred to herein as a wired network, one skilled in the art will recognize that any network that exposes routable (public) addresses may be implied. Thus, a wireless network with unique public (and routable) address can also employ techniques of the present invention to perform bi-directional communication. Also, one skilled in the art will recognize that terms such as wireless device, phone, handheld, etc., are used interchangeably to indicate any type of wireless device that is capable of operating with the AMPS. In addition, terms may have alternate spellings which may or may not be explicitly mentioned, and one skilled in the art will recognize that all such variations of terms are intended to be included." Mehta at [0033]. "Public address to proxy/router machine table 620 comprises a public network address field 631 and an indication of a functioning proxy/router machine 621. By maintaining such a mapping, the AMPS is able to substitute proxy/router machines for other proxy/router machines to provide a higher degree of robustness. Each proxy/router machine has a preconfigured set of public network addresses, such as are typically configured by network cards inserted into the proxy/router machine. These address are allocated in a standard fashion through prior purchase or obtaining from an address authorizing authority, currently the Internet Corporation for Assigned Names and Numbers (ICANN). When a machine is inserted for use into the AMPS, indications of these addresses are stored in field 631." Mehta at [0055].

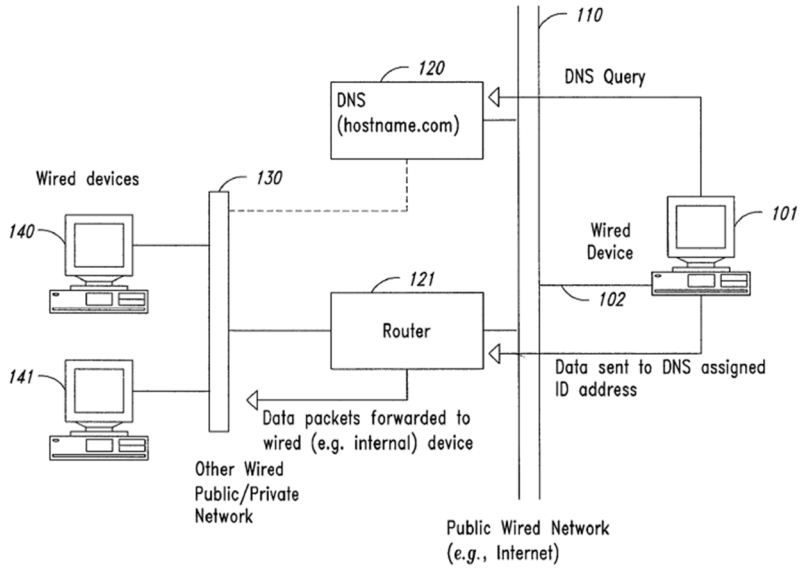


Fig. 1

Mehta at FIG. 1.

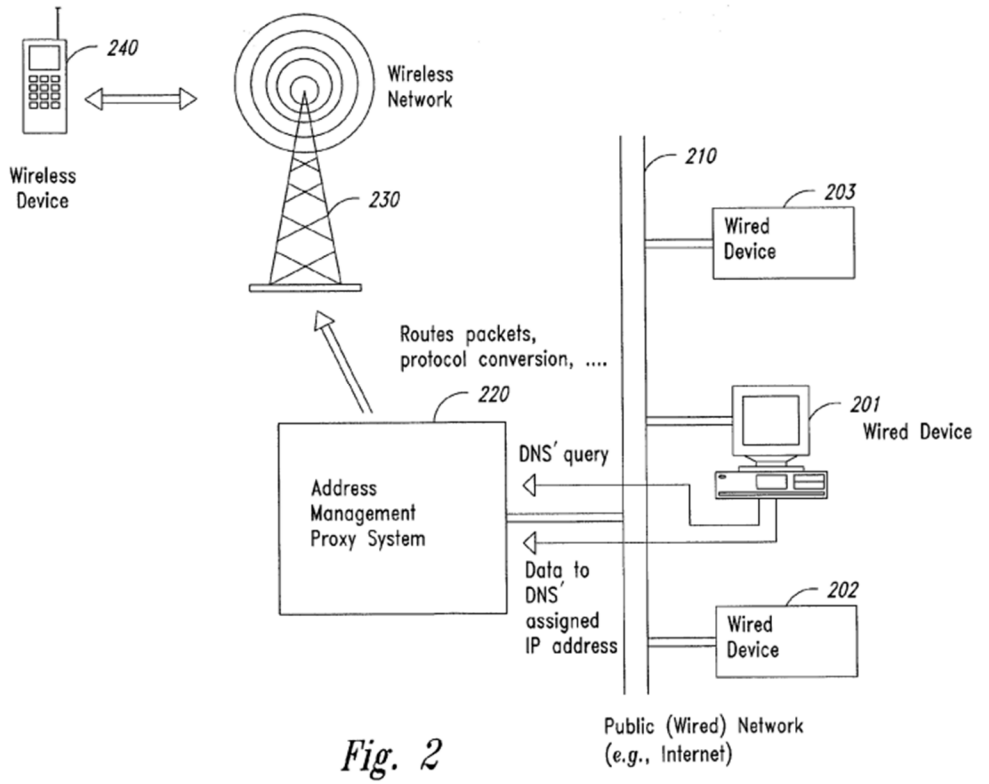


Fig. 2

Mehta at FIG. 2.

> "Methods and systems consistent with the present invention establish a virtual network on top of current IP network naming schemes. The virtual network uses a separate layer to create a modification to the IP packet format that is used to separate network behavior from addressing. As a result of the modification to the packet format, any type of delivery method may be assigned to any address or group of addresses. The virtual network also maintains secure communications between nodes, while providing the flexibility of assigning delivery methods independent of the delivery addresses."

Caronni I at Abstract.

	"The present invention relates generally to data processing systems and, more particularly, to a private network using a public-network infrastructure." Caronni I at 1:57–59. "Methods and systems consistent with the present invention overcome the shortcomings of existing networks by establishing a "Supernet," which is a private network that uses components from a public-network infrastructure. A Supernet allows an organization to utilize a public-network infrastructure for its enterprise network so that the organization no longer has to maintain a private network infrastructure; instead, the organization may have the infrastructure maintained for them by one or more service providers or other organizations that specialize in such connectivity matters. As such, the burden of maintaining an enterprise network is greatly reduced. Moreover, a Supernet is not geographically restrictive, so a user may plug their device into the Internet from virtually any portal in the world and still be able to use the resources of their private network in a secure and robust manner." Caronni I at 4:36-52. "Supernets also provide heterogeneous addressing functionality. The Supernet uses a separate layer that isolates address names of nodes from addressing schemes and delivery schemes. The Supernet contains a modification to the IP packet format that can be used to separate network behavior from addressing. As a result of the modification, any delivery scheme may be assigned to any address, or group of addresses." Caronni I at 4:53-59. "SNlogin 522 is a script used for logging into a Supernet. Successfully executing this script results in a Unix shell from which programs (e.g., node A 522) can be started to run within the Supernet context, such that address translation and security encapsulation is performed transparently for them and all they can typically access is other nodes on the Supernet. Alternatively, a parameter may be passed into SNlogin 522 that indicates a particular process to be automatically run in a Supernet context. Once a program is running in a Supernet

	context, all programs spawned by that program also run in the Supernet context, unless explicitly stated otherwise. SNlogout 524 is a script used for logging out of a Supernet. Although both SNlogin 522 and SNlogout 524 are described as being scripts, one skilled in the art will appreciate that their processing may be performed by another form of software. VARPD 526 performs address translation between node IDs and real addresses. KMC 528 is the key management component for each node that receives updates whenever the key for a channel ("the channel key") changes. There is one KMC per node per channel. KMD 530 receives requests from SNSL 542 of the TCP/IP protocol stack 534 when a packet is received and accesses the appropriate KMC for the destination node to retrieve the appropriate key to decrypt the packet. Node A 532 is a Supernet node running in a Supernet context." Caronni I at 8:31-55. "TCP/IP protocol stack 534 contains a standard TCP/UDP layer 538, two standard IP layers (an inner IP layer 540 and an outer IP layer 544), and a Supernet security layer (SNSL) 542, acting as the conduit for all Supernet communications. To conserve memory, both inner IP layer 540 and outer IP layer 544 may share the same instance of the code of an IP layer. SNSL 542 performs security functionality as well as address translation. It also caches the most recently used channel keys for ten seconds. Thus, when a channel key is needed, SNSL 542 checks its cache first, and if it is not found, it requests KMD 530 to contact the appropriate KMC to retrieve the appropriate channel key." Caronni I at 8:54-67. "FIG. 5 depicts administrative machine 306 and device 302 in greater detail, although the other devices 304 and 308–312 may contain similar components. Device 302 and administrative machine 306 communicate via Internet 314. Each device contains similar components, including a memory 502, 504; secondary storage 506, 508; a central processing unit (CPU) 510, 512; an input device 514, 516; and a video display 518, 520. One skilled in the art will appreciate that these devices may contain additional or different components. Memory 504 of administrative machine 306 includes the SASD process 540, VARPD 548, and KMS 550 all running in user mode. That is, CPU 512 is capable of running in at least two modes: user mode and kernel mode. When CPU 512 executes programs running in user

	mode, it prevents them from directly manipulating the hardware components, such as video display 518. On the other hand, when CPU 512 executes programs running in kernel mode, it allows them to manipulate the hardware components. Memory 504 also contains a VARPDB 551 and a TCP/IP protocol stack 552 that are executed by CPU 512 running in kernel mode. TCP/IP protocol stack 552 contains a TCP/UDP layer 554 and an IP layer 556, both of which are standard layers well known to those of ordinary skill in the art. Secondary storage 508 contains a configuration file 558 that stores various configuration-related information (described below) for use by SASD 540." Caronni I at 6:44-7:2. *See also* Caronni I at Claim 1, FIGS. 3, 5
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	Cisco VPN Routers and Software discloses "a memory and a processor to implement a register module configured to register devices in a virtual network." *See, e.g.*, The claim requires a memory and processor to implement a register module for managing VPN functions. Cisco VPN systems use encryption hardware (e.g., ISM or SEP modules) to enhance processing power for VPN data handling. "Integrated Services Module (ISM) for hardware-based VPN services acceleration, such as high-speed IPSec or MPPE encryption." (Cisco 7100 Series VPN Routers) "IPSec is the next−generation network layer crypto platform for Cisco's security platforms (Cisco IOS® Software, PIX, and so on)." (Cisco IPSec Encryption) "Cisco SEP modules provide hardware-based encryption, ensuring consistent performance throughout the rated capacity." (Cisco VPN 3000 Series Concentrator Data Sheet) "The VPN Concentrator automatically detects the new module during reboot." (Cisco SEP Installation Guide)

	To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Mehta, Caronni I, Caronni II, and/or Hipp. Mehta, Caronni I, Caronni II, and Hipp concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, registration of devices in a virtual network was a well-known and commonly-used technique. Modifying this reference with Mehta's, Caronni I's, Caronni II's, and/or Hipp's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with those of Mehta, Caronni I, Caronni II, or Hipp for example, to incorporate this limitation. *See, e.g.*, "Methods and systems for providing two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP, are provided. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporarily use by a requesting device on a public network to communicate with a wireless device on a wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more data repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the public network to send data to the wireless device." Mehta at Abstract.

	"To insure a greater degree of security, according to one embodiment, the AMPS maintains a Time to Live (TTL) parameter with each address mapping. In this way, once the TTL value indicates that the mapping has expired, the AMPS can destroy the mapping, and also any connection. Further, in some embodiments, the AMPS also puts a timestamp in its own mapping tables. After some timeout period based upon the timestamp, the AMPS can destroy the mapping, thereby forcing a new mapping to be initiated on a periodic basis." Mehta at [0015]. "In existing systems, data communication (communication on a data channel) between a wireless device that is connected to a private wireless network and a wired device connected to a public wired network (e.g., the Internet) can only be initiated by the wireless device. Some carriers have assigned fixed public IP addresses to wireless devices; however, the wireless devices need to then have client programs (e.g., a UDP stack) capability of receiving and handling the incoming communication packets. Moreover, these wireless devices are then part of a public wireless network and not a private wireless network. Since public IP addresses are becoming a scarcer commodity and currently expensive to service a network of multi-million devices, carriers cannot in a practical sense count on having a fixed public IP address for each device on its network. (Although movement to IPv6 will allow more addresses, the current IPv4 definitions are limited and the potential number of wireless users subscribing to larger carriers is very high. In some regions of the world, the current public address scheme is even more limited.) Further, such addressing capability would expose each device to further security risks, because each such device is part of a publicly accessible network and it becomes more difficult to implement and enforce security measures. Thus, assigning private IP addresses to devices is preferred in existing wireless networks over assigning fixed public IP addresses to devices. When wireless networks are private, the locations (addresses) of the wireless devices are intentionally hidden from public view by a carrier system's (or, as referred to in some countries, operator's) infrastructure." Mehta at [0025].

	"Example embodiments described herein provide applications, tools, data structures and other support to implement private to public address mappings over one or more wired and wireless networks to be used for bi-directional communication. One skilled in the art will recognize that other embodiments of the methods and systems of the present invention may be used for many other purposes, including to push information and/or data or code from a public network such as the Internet to a wireless device. In addition, although this description primarily refers to "data" as being sent via the networks, one skilled in the art will recognize that all types of data can be communicated using the techniques described herein including, but not limited to, text, graphics, audio, and video." Mehta at [0034]. "FIG. 4 is an example block diagram of a general purpose computer system for practicing embodiments of the Address Management Proxy System. The general purpose computer system 400 may comprise one or more server and/or client computing systems and may span distributed locations. In addition, each block may represent one or more such blocks as appropriate to a specific embodiment or may be combined with other blocks. The various blocks of the Address Management Proxy System 410 may physically reside on one or more machines, which use standard interprocess communication mechanisms to communicate with each other. In the embodiment shown, computer system 400 comprises a computer memory ("memory") 01, a display 402, a Central Processing Unit ("CPU") 403, and Input/Output devices 404. The Address Management Proxy System 410 is shown residing in memory 401. The components of the Address Management Proxy System 410 preferably execute on CPU 403 and manage the address mapping of wireless devices on a wireless network, as described in previous figures, to allow other wired systems to communicate with the wireless devices. Other downloaded code 405 and potentially other data repositories also reside in the memory 410, and preferably execute on one or more CPU's 403. In a typical embodiment, the AMPS 410 includes one or more DNS'/API servers 411, one or more Address Proxy/Routers 412, an Address Management Data Server 413, and Address Management Data Repositories 414. As described earlier, the AMPS may include other data repositories and components, such as a load balancer, depending upon the particular implementation." Mehta at [0036].

"In one embodiment, a timestamp of the mapping between a public network address and a private address is also noted in table 630. After a defined time out, based upon this timestamp, the Address Management Data Server sends a request to the Address Proxy/Router associated with the mapping to unmap the public-to-private address mapping and updates the mapping table 630, thereby returning the associated public address to the pool of unused public network addresses."

Mehta at [0056].

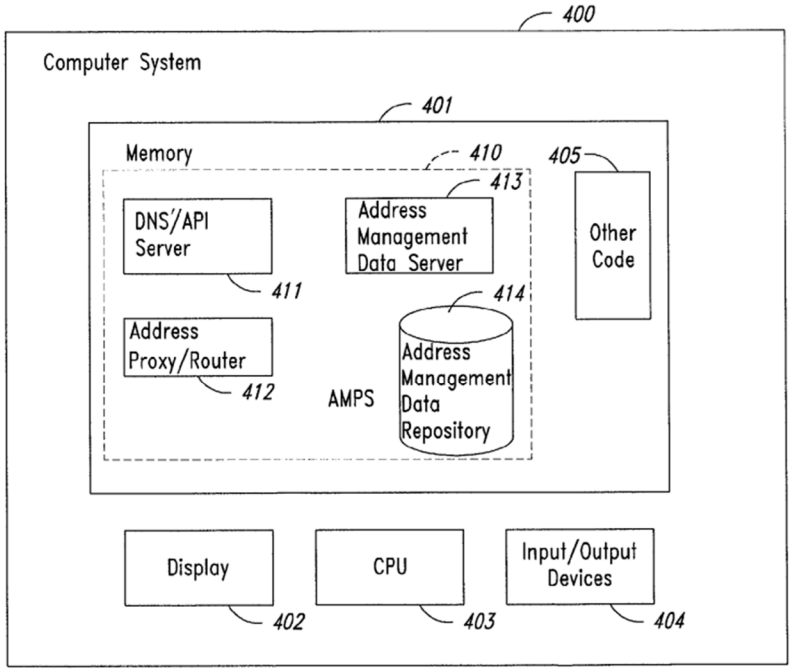


Fig. 4

	Mehta at FIG. 4. "To configure a Supernet, a system administrator creates a configuration file 558 that is used by SASD 540 when starting or reconfiguring a Supernet. This file may specify: (1) the Supernet name, (2) all of the channels in the Supernet, (3) the nodes that communicate over each channel, (4) the address of the KMS for each channel, (5) the address of the VARPD that acts as the server for the Supernet, (6) the user IDs of the users who are authorized to create Supernet nodes, (7) the authentication mechanism to use for each user of each channel, and (8) the encryption algorithm to use for each channel. Although the configuration information is described as being stored in a configuration file, one skilled in the art will appreciate that this information may be retrieved from other sources, such as databases or interactive configurations." Caronni I at 8:1-15. "SASD 540 represents a Supernet: there is one instance of an SASD per Supernet, and it both authenticates nodes and authorizes nodes to join the Supernet. VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as a node IDs, to the network addresses recognized by the public-network infrastructure, known as the real addresses. The "node ID" may include the following: a Supernet ID (e.g., 0x123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1). Although the virtual address is described in an IP address scheme, one skilled in the art will appreciate that the virtual address may be any other type addressing scheme, such as an e-mail address, IPX, or IPv6. Since the node ID includes a Supernet ID, a node will have more than one node ID when it communicates over more than one channel. The "real address" is an IP address (e.g., 10.0.0.2) that is globally unique and meaningful to the public-network infrastructure. In a Supernet, one VARPD runs on each machine, and it may play two roles. First, a VARPD may act as a server by storing all address mappings for a particular Supernet into its associated VARPDB. Second, regardless of its role as a server or not, each VARPD assists in address translation for the nodes on its machine. In this role, the VARPD stores into its associated VARPDB the address mappings for its nodes, and if it needs a mapping that it does not have, it will contact the VARPD that acts as the

	server for the given Supernet to obtain it. The VARPDB may also decide which virtual address to use in the translation. That is, the VARPDB may associate a virtual address with multiple real addresses or vice versa." Caronni I at 7:3–33. "FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:67–10:18. "After configuring SNSL, SNlogin invokes an operating system call, SETVIN, to cause the SNlogin script to run in a Supernet context (step 720). In Unix, each process has a data structure known as the "proc structure" that contains the process ID as well as a pointer to a virtual memory description of this process. In accordance with methods and systems consistent with the present invention, the Supernet IDs indicating the channels over which the process communicates as well as its virtual address for this process are added to this structure. By associating this information with the process, the SNSL layer can enforce that this process runs in a Supernet context. Although methods and systems consistent with the present invention are described as operating in a Unix environment, one skilled in the art will appreciate that such methods and systems can operate in other environments. After the SNlogin script runs in the Supernet context, the SNlogin script spawns a Unix program, such

	as a Unix shell or a service daemon (step 722). In this step, the SNlogin script spawns a Unix shell from which programs can be run by the user. All of these programs will thus run in the Supernet context until the user runs the SNlogout script." Caronni I at 10:63–11:17. "The packet and Supernet ID are then transmitted to the SNSL layer using the modified socket structure (step 806). The SNSL layer then accesses the VARPDB to obtain the address mapping between virtual source node address 642 and the source real address 614 as well as the virtual destination node address 644 and the destination real address 616 (step 808). If they are not contained in the VARPDB because this is the first time a packet has been sent from this node or sent to this destination, the VARPDB accesses the local VARPD to obtain the mapping. When contacted, the VARPD on the local machine contacts the VARPD that acts as the server for the Supernet to obtain the appropriate address mapping. Since the VARPDB maintains all real IP addresses, a remote node may securely communicate with another remote node without reverfication." Caronni I at 11:47–61. *See also* Caronni I at FIGS. 7A and 7B. "In an embodiment of the present invention a virtual network is supported by a physical network and includes a virtual address resolution facility. The virtual address resolution facility is used to resolve a virtual IP address into a real IP address. A registration request is received at the virtual address resolution facility which includes a real IP address, a port address, a transport protocol designation and an Application layer protocol designation associated with a virtual address destination. The associations are stored using the virtual address resolution facility. A resolution request is received referencing a virtual address destination and the resolution request is resolved using the virtual address resolution facility and the stored associations." Caronni II at 3:11–23.

	"FIG. 2 depicts a block diagram of the VARP lookup table 26 used by the virtual address resolution facility 24. The VARP lookup table 26 may be stored at any location accessible over the network. The VARP lookup table 26 includes registered virtual destination addresses located in a column 78 and corresponding associations located in a column 79. Entries 80, 82, 84, 86 and 88 include the registered virtual destination address and information associated with the registered virtual destination address. Each entry in the VARP lookup table 26 will list the registering virtual IP address 90 and associated information including a real IP address 91, a transport protocol designation 92, a port number identification 93 and an Application layer protocol designation 94. The associations are provided at the time the virtual destination address is registered in the virtual network. For example, the entry 80 for the virtual IP address 10.0.0.12 (90) includes the association of a real IP address 152.70.0.1 (91), an associated Transport layer protocol designation, UDP 92, an associated port number, 6789 (93), and an Application layer protocol designation of 'none' 94." Caronni II at 5:10–29. "FIG. 4 depicts the sequence of steps followed by the illustrative embodiment of the present invention to process the received encapsulated message. The sequence begins when a process associated with a virtual IP address registers with the virtual address resolution facility 24 (step 140). The registration is stored in a VARP lookup table accessible over the network. Subsequently, a message is received at the electronic device which bears a MAC/network interface address of the network interface of the electronic device with the designated real IP address listed in the virtual address registration (Step 142). The MAC address header 117 is stripped off at the Link layer and the message and appended headers are passed up to the Network layer 114(Step 144). The Network layer 114 strips off the IP header 115 and identifies the Transport Protocol header 113 underneath. The Network layer passes the message and remaining appended header to the Transport layer 112 (Step 146). The Transport layer 112 strips off the Transport header and then passes the message to the Application layer via the port address of the interface identified in the VARP registration. (Step 148). The receiving process determines whether it is the virtual destination address the message is addressed to (Step 149) and acts accordingly. If the receiving process that is examining the message is the destination address, the process examines the data (Step 150). Alternatively, if

	the message is not intended for the process executing on the edge device, the process re-routes the message and sends it back down the protocol model stack, on to the local network operating behind the edge device, and on to its intended destination by performing traditional VARP and ARP lookups. Alternatively forwarding the message may be accomplished by reinserting the packet into the IP layer of the networking stack." Caronni II at 7:33–67. *See also* Caronni II at Claim 1, FIGS. 2-4. "FIG. 3 is a data flow diagram illustrating the registration of virtual network environment parameters. The VNE framework 200 is a software module that processes transactions between the applications and the operating system. The VNE parameters are registered with the VNE framework 200 at the time the application is started. The VNE parameters include the application IP address, the virtual network subnet, and the global virtual address subnet. At step 250, the registration harness 220 supplies the application IP address, virtual subnet, and the global virtual address subnet for a $process_x$ to the VNE framework 200. The VNE framework 200 then records the IP address, virtual subnet, and the global virtual address subnet for the $process_x$ at step 252. The $process_x$ can then spawn additional processes (or create additional objects) at step 254. the new $process_y$ inherits the IP address, virtual subnet, and global virtual address subnet from $process_x$. At step 256, the registration harness 220 launches the application related to $process_y$." Hipp at 3:52-4:2. "Virtualization of network identity is achieved by assigning a unique virtual IP address and virtual hostname to a group of processes that make up the application instance which the instance keeps throughout its execution. This virtual network identity stays with the application instance regardless of which node the application is running on. The framework, in essence, provides a mechanism to create this virtual network identity (VNI) around the application using the virtual network parameters assigned to it. In one embodiment, the virtual network parameters include an IP address and hostname. The framework 200 ensures

	that the application's instance uses the virtual network parameters, transparently, so that it can be moved across machines, without modifications to the application." Hipp at 6:1-15. *See also* Hipp at 6:16-60, FIG. 3.
[30.3] receive a registration request from an agent associated with a device;	Cisco VPN Routers and Software discloses "receive a registration request from an agent associated with a device." *See, e.g.*, The claim requires receiving a registration request from a device. Cisco VPN concentrators manage client registration and access policies, demonstrating this functionality. "VPN access policies are created and stored centrally in the Cisco VPN 3000 Concentrator and pushed to the client when a connection is established." (Cisco VPN 3000 Series Concentrator Data Sheet). "Centralized role-based Access Control (RBAC) enables organizations to scale access privileges." (CiscoWorks VMS 2.2) To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Mehta, Caronni I, Caronni II, and/or Hipp. Mehta, Caronni I, Caronni II, and Hipp concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, registration of devices in a virtual network was a well-known and commonly-used technique. Modifying this reference with Mehta's, Caronni I's, Caronni II's, and/or Hipp's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with those of Mehta, Caronni I, Caronni II, or Hipp for example, to incorporate this limitation. *See, e.g.*,

	"Methods and systems for providing two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP, are provided. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporarily use by a requesting device on a public network to communicate with a wireless device on a wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more data repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the public network to send data to the wireless device." Mehta at Abstract. "To insure a greater degree of security, according to one embodiment, the AMPS maintains a Time to Live (TTL) parameter with each address mapping. In this way, once the TTL value indicates that the mapping has expired, the AMPS can destroy the mapping, and also any connection. Further, in some embodiments, the AMPS also puts a timestamp in its own mapping tables. After some timeout period based upon the timestamp, the AMPS can destroy the mapping, thereby forcing a new mapping to be initiated on a periodic basis." Mehta at [0015]. "In existing systems, data communication (communication on a data channel) between a wireless device that is connected to a private wireless network and a wired device connected to a public wired network (e.g., the Internet) can only be initiated by the wireless device.

	Some carriers have assigned fixed public IP addresses to wireless devices; however, the wireless devices need to then have client programs (e.g., a UDP stack) capability of receiving and handling the incoming communication packets. Moreover, these wireless devices are then part of a public wireless network and not a private wireless network. Since public IP addresses are becoming a scarcer commodity and currently expensive to service a network of multi-million devices, carriers cannot in a practical sense count on having a fixed public IP address for each device on its network. (Although movement to IPv6 will allow more addresses, the current IPv4 definitions are limited and the potential number of wireless users subscribing to larger carriers is very high. In some regions of the world, the current public address scheme is even more limited.) Further, such addressing capability would expose each device to further security risks, because each such device is part of a publicly accessible network and it becomes more difficult to implement and enforce security measures. Thus, assigning private IP addresses to devices is preferred in existing wireless networks over assigning fixed public IP addresses to devices. When wireless networks are private, the locations (addresses) of the wireless devices are intentionally hidden from public view by a carrier system's (or, as referred to in some countries, operator's) infrastructure." Mehta at [0025]. "Example embodiments described herein provide applications, tools, data structures and other support to implement private to public address mappings over one or more wired and wireless networks to be used for bi-directional communication. One skilled in the art will recognize that other embodiments of the methods and systems of the present invention may be used for many other purposes, including to push information and/or data or code from a public network such as the Internet to a wireless device. In addition, although this description primarily refers to "data" as being sent via the networks, one skilled in the art will recognize that all types of data can be communicated using the techniques described herein including, but not limited to, text, graphics, audio, and video." Mehta at [0034]. "In one embodiment, a timestamp of the mapping between a public network address and a private address is also noted in table 630. After a defined time out, based upon this

	timestamp, the Address Management Data Server sends a request to the Address Proxy/Router associated with the mapping to unmap the public-to-private address mapping and updates the mapping table 630, thereby returning the associated public address to the pool of unused public network addresses." Mehta at [0056]. "FIG. 3 depicts a data processing system 300 suitable for use with methods and systems consistent with the present invention. Data processing system 300 comprises a number of devices, such as computers 302–312, connected to a public network, such as the Internet 314. A Supernet's infrastructure uses components from the Internet because devices 302, 304, and 312 contain nodes that together form a Supernet and that communicate by using the infrastructure of the Internet. These nodes 316, 318, 320, and 322 are communicative entities (e.g., processes) running within a particular device and are able to communicate among themselves as well as access the resources of the Supernet in a secure manner. When communicating among themselves, the nodes 316, 318, 320, and 322 serve as end points for the communications, and no other processes or devices that are not part of the Supernet are able to communicate with the Supernet's nodes or utilize the Supernet's resources. The Supernet also includes an administrative node 306 to administer to the needs of the Supernet." Caronni I at 4:66–5:13. "FIG. 5 depicts administrative machine 306 and device 302 in greater detail, although the other devices 304 and 308–312 may contain similar components. Device 302 and administrative machine 306 communicate via Internet 314. Each device contains similar components, including a memory 502, 504; secondary storage 506, 508; a central processing unit (CPU) 510, 512; an input device 514, 516; and a video display 518, 520. One skilled in the art will appreciate that these devices may contain additional or different components. Memory 504 of administrative machine 306 includes the SASD process 540, VARPD 548, and KMS 550 all running in user mode. That is, CPU 512 is capable of running in at least two modes: user mode and kernel mode. When CPU 512 executes programs running in user mode, it prevents them from directly manipulating the hardware components, such as video

	display 518. On the other hand, when CPU 512 executes programs running in kernel mode, it allows them to manipulate the hardware components. Memory 504 also contains a VARPDB 551 and a TCP/IP protocol stack 552 that are executed by CPU 512 running in kernel mode. TCP/IP protocol stack 552 contains a TCP/UDP layer 554 and an IP layer 556, both of which are standard layers well known to those of ordinary skill in the art. Secondary storage 508 contains a configuration file 558 that stores various configuration-related information (described below) for use by SASD 540." Caronni I at 6:43–7:2. "Memory 502 of device 302 contains SNlogin script 522, SNlogout script 524, VARPD 526, KMC 528, KMD 530, and node A 532, all running in user mode. Memory 502 also includes TCP/IP protocol stack 534 and VARPDB 536 running in kernel mode." Caronni I at 8:26-30. "FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:66-10:18. *See also* Caronni I at FIGS. 3, 5, 7A, 7B.

	"In an embodiment of the present invention a virtual network is supported by a physical network and includes a virtual address resolution facility. The virtual address resolution facility is used to resolve a virtual IP address into a real IP address. A registration request is received at the virtual address resolution facility which includes a real IP address, a port address, a transport protocol designation and an Application layer protocol designation associated with a virtual address destination. The associations are stored using the virtual address resolution facility. A resolution request is received referencing a virtual address destination and the resolution request is resolved using the virtual address resolution facility and the stored associations." Caronni II at 3:11–23. "FIG. 2 depicts a block diagram of the VARP lookup table 26 used by the virtual address resolution facility 24. The VARP lookup table 26 may be stored at any location accessible over the network. The VARP lookup table 26 includes registered virtual destination addresses located in a column 78 and corresponding associations located in a column 79. Entries 80, 82, 84, 86 and 88 include the registered virtual destination address and information associated with the registered virtual destination address. Each entry in the VARP lookup table 26 will list the registering virtual IP address 90 and associated information including a real IP address 91, a transport protocol designation 92, a port number identification 93 and an Application layer protocol designation 94. The associations are provided at the time the virtual destination address is registered in the virtual network. For example, the entry 80 for the virtual IP address 10.0.0.12 (90) includes the association of a real IP address 152.70.0.1 (91), an associated Transport layer protocol designation, UDP 92, an associated port number, 6789 (93), and an Application layer protocol designation of 'none' 94." Caronni II at 5:10–29. "FIG. 4 depicts the sequence of steps followed by the illustrative embodiment of the present invention to process the received encapsulated message. The sequence begins when a process associated with a virtual IP address registers with the virtual address resolution facility 24 (step 140). The registration is stored in a VARP lookup table accessible over the network.

	Subsequently, a message is received at the electronic device which bears a MAC/network interface address of the network interface of the electronic device with the designated real IP address listed in the virtual address registration (Step 142). The MAC address header 117 is stripped off at the Link layer and the message and appended headers are passed up to the Network layer 114(Step 144). The Network layer 114 strips off the IP header 115 and identifies the Transport Protocol header 113 underneath. The Network layer passes the message and remaining appended header to the Transport layer 112 (Step 146). The Transport layer 112 strips off the Transport header and then passes the message to the Application layer via the port address of the interface identified in the VARP registration. (Step 148). The receiving process determines whether it is the virtual destination address the message is addressed to (Step 149) and acts accordingly. If the receiving process that is examining the message is the destination address, the process examines the data (Step 150). Alternatively, if the message is not intended for the process executing on the edge device, the process re-routes the message and sends it back down the protocol model stack, on to the local network operating behind the edge device, and on to its intended destination by performing traditional VARP and ARP lookups. Alternatively forwarding the message may be accomplished by reinserting the packet into the IP layer of the networking stack." Caronni II at 7:33–67. *See also* Caronni II at Claim 1, FIGS. 2-4. "FIG. 3 is a data flow diagram illustrating the registration of virtual network environment parameters. The VNE framework 200 is a software module that processes transactions between the applications and the operating system. The VNE parameters are registered with the VNE framework 200 at the time the application is started. The VNE parameters include the application IP address, the virtual network subnet, and the global virtual address subnet. At step 250, the registration harness 220 supplies the application IP address, virtual subnet, and the global virtual address subnet for a $process_x$ to the VNE framework 200. The VNE framework 200 then records the IP address, virtual subnet, and the global virtual address subnet for the $process_x$ at step 252. The $process_x$ can then spawn additional processes (or create additional objects) at step 254. the new $process_y$ inherits the IP address, virtual subnet,

	and global virtual address subnet from process$_x$. At step 256, the registration harness 220 launches the application related to process$_y$." Hipp at 3:52-4:2. "Virtualization of network identity is achieved by assigning a unique virtual IP address and virtual hostname to a group of processes that make up the application instance which the instance keeps throughout its execution. This virtual network identity stays with the application instance regardless of which node the application is running on. The framework, in essence, provides a mechanism to create this virtual network identity (VNI) around the application using the virtual network parameters assigned to it. In one embodiment, the virtual network parameters include an IP address and hostname. The framework 200 ensures that the application's instance uses the virtual network parameters, transparently, so that it can be moved across machines, without modifications to the application." Hipp at 6:1-15. *See also* Hipp at 6:16-60, FIG. 3.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	Cisco VPN Routers and Software discloses "distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address." *See, e.g.*, The claim requires distributing a virtual network address to registered devices. Cisco VPN systems support large numbers of VPN tunnels, indicating address assignment and management. "Utilizing ISM and ISA hardware encryption acceleration, the Cisco 7100 Series VPN Router can support up to 3000 simultaneous IPSec tunneling sessions with 3DES IPSec encryption performance up to 140 Mbps." (Cisco 7100 Series VPN Routers). "The Cisco VPN 3080 Concentrator is optimized to support large enterprise organizations that demand the highest level of performance combined with support for up to 10,000 simultaneous

	remote access sessions." (Cisco VPN 3000 Series Concentrator Data Sheet). To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Mehta, Caronni I, Caronni II and/or Hipp. Mehta, Caronni I, Caronni II and Hipp concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, distributing virtual network addresses was a well-known and commonly-used technique. Modifying this reference with Mehta's, Caronni I's, Caronni II's, and/or Hipp's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with those of Mehta, Caronni I, Caronni II, or Hipp for example, to incorporate this limitation. *See, e.g.*, "Methods and systems for providing two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP, are provided. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporarily use by a requesting device on a public network to communicate with a wireless device on a wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more data repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the public network to send data to the wireless device."

	Mehta at Abstract. "To insure a greater degree of security, according to one embodiment, the AMPS maintains a Time to Live (TTL) parameter with each address mapping. In this way, once the TTL value indicates that the mapping has expired, the AMPS can destroy the mapping, and also any connection. Further, in some embodiments, the AMPS also puts a timestamp in its own mapping tables. After some timeout period based upon the timestamp, the AMPS can destroy the mapping, thereby forcing a new mapping to be initiated on a periodic basis." Mehta at [0015]. "In existing systems, data communication (communication on a data channel) between a wireless device that is connected to a private wireless network and a wired device connected to a public wired network (e.g., the Internet) can only be initiated by the wireless device. Some carriers have assigned fixed public IP addresses to wireless devices; however, the wireless devices need to then have client programs (e.g., a UDP stack) capability of receiving and handling the incoming communication packets. Moreover, these wireless devices are then part of a public wireless network and not a private wireless network. Since public IP addresses are becoming a scarcer commodity and currently expensive to service a network of multi-million devices, carriers cannot in a practical sense count on having a fixed public IP address for each device on its network. (Although movement to IPv6 will allow more addresses, the current IPv4 definitions are limited and the potential number of wireless users subscribing to larger carriers is very high. In some regions of the world, the current public address scheme is even more limited.) Further, such addressing capability would expose each device to further security risks, because each such device is part of a publicly accessible network and it becomes more difficult to implement and enforce security measures. Thus, assigning private IP addresses to devices is preferred in existing wireless networks over assigning fixed public IP addresses to devices. When wireless networks are private, the locations (addresses) of the wireless devices are intentionally hidden from public view by a carrier system's (or, as referred to in some countries, operator's) infrastructure." Mehta at [0025].

	"FIG. 3 is an example block diagram of components of an example Address Management Proxy System. In one embodiment, the Address Management Proxy System (AMPS) comprises one or more modified DNS'/ API servers 302, one or more Address Proxy/Routers 305, an Address Management Data Server 303 which manages a database or other repositories such as Address Management Data Repository 304, and optionally a load balancer 301. The DNS'/API servers 302 are either individually connected to a public network 310 or are connected to the load balancer 301 which is in turn connected to public network 310. Similarly, each Address Proxy/Router 305 is also connected to the public network 310, via the routable (public) address to which data from the external network 310 is sent that is destined for the wireless devices. The DNS'/API servers 302 are either modified implementations of a DNS server to add functionality necessary to communicate with wireless devices, or are servers that implement one or more specialized APIs, as will be described further below. The DNS'/API servers 302 use the Address Management Data Server 303 to assist in mapping a unique identifier (e.g., string name) for a wireless device to a public address on public network 310. The pool of public addresses is also maintained by the Address Management Data Serve 303 and Data Repository 304. The Address Management Data server 303 and Address Management Data Repository 304 may be implemented using existing database technology, for example, ODBC technology or, may be implemented as a structure such as a simple text file. One skilled in the art will recognize that any embodiment for storing a set of tables, data, lists, or mappings can be used. Each Address Proxy/Router 305 also uses the Address Management Data Server 303 or equivalent to create and update a series of routing tables that are used to assign public addresses to wireless devices as needed and to update the various mappings between public addresses and the non-routable (private) addresses of the wireless devices. The tables and mappings that are maintained on behalf of the DNS'/API servers 302 and the Address Proxy/Routers 305 by the Address Management Data Server 303 are described below with reference to FIG. 6." Mehta at [0032]. "Although the techniques of the AMPS are generally applicable to any a wired device communicating with a wireless device, the phrase "public network" (or "wired network") is used generally to imply any type of internetworked environment including a public network

	or a backbone that is somewhere down the line connected to one or more private or public networks. In addition, although the examples described herein often refer to the Internet, one skilled in the art will recognize that the concepts and inventions described are applicable to other forms and embodiments of internetworking, including, for example ATM type networks. Thus, techniques of the present invention can also be used by one device on a first wireless network to communicate with another wireless device on a second network—each device ends up communicating with the Address Proxy/Router of the other network. This scenario is feasible because each wireless network (or its carrier infrastructure) is connected to a proxy/router that is also connected (via a public address) to a public network. In addition, although a public network is sometimes also referred to herein as a wired network, one skilled in the art will recognize that any network that exposes routable (public) addresses may be implied. Thus, a wireless network with unique public (and routable) address can also employ techniques of the present invention to perform bi-directional communication. Also, one skilled in the art will recognize that terms such as wireless device, phone, handheld, etc., are used interchangeably to indicate any type of wireless device that is capable of operating with the AMPS. In addition, terms may have alternate spellings which may or may not be explicitly mentioned, and one skilled in the art will recognize that all such variations of terms are intended to be included." Mehta at [0033]. "Example embodiments described herein provide applications, tools, data structures and other support to implement private to public address mappings over one or more wired and wireless networks to be used for bi-directional communication. One skilled in the art will recognize that other embodiments of the methods and systems of the present invention may be used for many other purposes, including to push information and/or data or code from a public network such as the Internet to a wireless device. In addition, although this description primarily refers to "data" as being sent via the networks, one skilled in the art will recognize that all types of data can be communicated using the techniques described herein including, but not limited to, text, graphics, audio, and video." Mehta at [0034].

	"There are several implementation approaches to the components of the Address Management Proxy System, three of which are described herein. One skilled in the art will recognize that various other approaches and combinations are possible. All three approaches allocate a public (routable) network address for temporary use by a wired device to communicate with a wireless device. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained and allocated dynamically to wireless network devices as required. For example, a typical Class B Internet network address block allows for approximately 65,000 simultaneous connections to wireless devices. Although this number may seem large at first blush, when one considers the number of cell phones and handsets, for example, connected to a carrier, this number can be quite limiting." Mehta at [0039]. "The unique ID table 610 maps unique string names of wireless devices to the private wireless network addresses that have been assigned typically by a carrier's infrastructure. In some embodiments, the carrier infrastructure dynamically allocates, using methods similar to a DHCP protocol, a private wireless network address when the wireless device registers itself with the carrier infrastructure upon being powered on. Thus, the unique ID table 610 may be sparsely filled or entries created dynamically and then deleted dynamically as devices register and unregister with the carrier infrastructure system." Mehta at [0053]. "In one embodiment, a timestamp of the mapping between a public network address and a private address is also noted in table 630. After a defined time out, based upon this timestamp, the Address Management Data Server sends a request to the Address Proxy/Router associated with the mapping to unmap the public-to-private address mapping and updates the mapping table 630, thereby returning the associated public address to the pool of unused public network addresses." Mehta at [0056]. "FIG. 7 is an example flow diagram of an example routine provided by a DNS'/API server of the Address Management Proxy System to return a public address that corresponds to a

	designated unique identifier. In essence, this routine implements a DNS query or DNS-like query capability for the AMPS using a modified GetHostByName interface or a specialized API, for example GetProxyIP, as described with reference to FIG. 5. In summary, the routine dynamically allocates an appropriate Address Proxy/Router machine to associate with the wireless device and returns the public address of that machine (along with a TTL parameter and potentially other parameters). Specifically, in step 701, the routine determines the private (non-routable) address of the wireless device designated by a string parameter passed as input to the routine. For example, the string parameter may use fields such as "uniqueID.hostname.domain.tld," which specifies a typical hierarchy of person/service on a machine named "hostname" on a domain such as a company's network on a top level domain such as "org." "com," "edu," etc. One skilled in the art will recognize that many other string parameter designations could be used. One mechanism for implementing this routine is to request information from the Address Management Data Repository. In one embodiment, the data repository stores a table that maps unique IDs to private network addresses (see Table 610 in FIG. 6). Preferably, any mechanism that is used by the AMPS stores this data in a secure manner in order to keep the wireless network addresses private. In step 702, the routine retrieves the public network address that corresponds to the private wireless network address of the designated device if one has already been assigned by the AMPS to that device and is still valid. In one embodiment, the data repository stores this mapping information between private wireless network address and public network address (see for example table 630 in FIG. 6). If a public network address has not already been assigned or is not valid, then the routine causes a new public network address to be allocated and that new public address is associated with the private wireless network address. Appropriate tables in the data repository are then updated. In step 703, the routine determines the Address Proxy/Router machine that is associated with the assigned public network address (for example using table 620 in FIG. 6). In step 704, the routine sends a request to the determined proxy/router machine to update its routing tables to map the determined public network address to the private wireless network address. In step 705, the routine updates information in the data repository to include any other connection related information (e.g., field 634 in Table 630 in FIG. 6) and indicates a Time to Live (TTL) parameter for the public-private address association (e.g., field 633 in Table 630 in FIG. 6). Once all of the tables have been updated in both the proxy/router and in

the data repository, the DNS'/API server returns the determined public network address of the associated Address Proxy/Router machine. As described earlier, the public address may be a (hostname, port) pair when a port-based implementation is used."

Mehta at [0057].

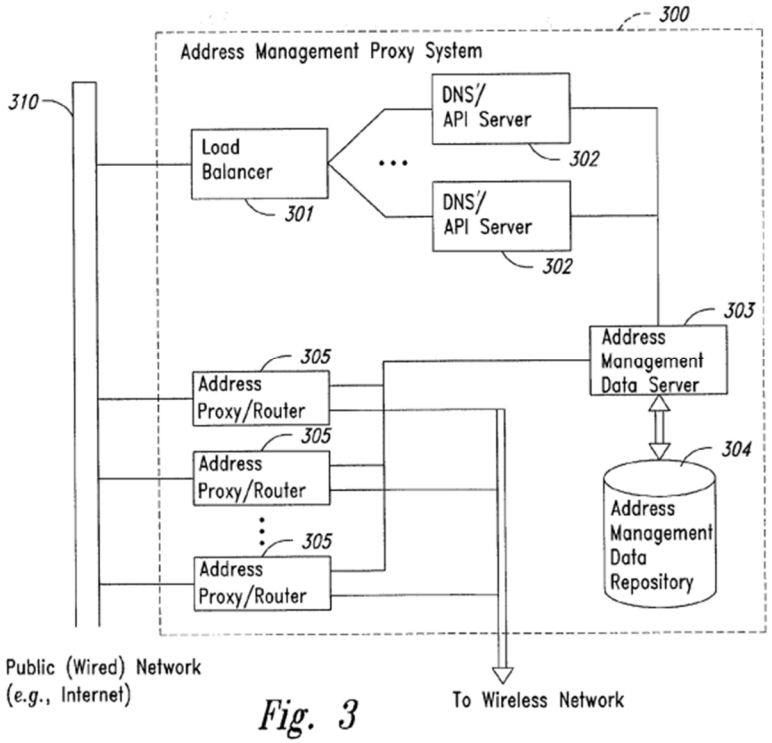


Fig. 3

Mehta at FIG. 3.

"Fourth, the system provides address translation in a transparent manner. Since the Supernet is a private network constructed from the infrastructure of another network, the Supernet has its own internal addressing scheme, separate from the addressing scheme of the underlying public network. Thus, when a packet from a Supernet node is sent to another Supernet node,

	it travels through the public network. To do so, the Supernet performs address translation from the internal addressing scheme to the public addressing scheme and vice versa. By separating the addressing schemes, the Supernet creates a flexible delivery scheme that is easily changeable by network software or a system administrator. To reduce the complexity of Supernet nodes, system-level components of the Supernet perform this translation on behalf of the individual nodes so that it is transparent to the nodes. Another benefit of the Supernet's addressing is that it uses an IP-based internal addressing scheme so that preexisting programs require little modification to run within a Supernet." Caronni I at 6:7–25. "SASD 540 represents a Supernet: there is one instance of an SASD per Supernet, and it both authenticates nodes and authorizes nodes to join the Supernet. VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as a node IDs, to the network addresses recognized by the public-network infrastructure, known as the real addresses. The "node ID" may include the following: a Supernet ID (e.g., 0x123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1). Although the virtual address is described in an IP address scheme, one skilled in the art will appreciate that the virtual address may be any other type addressing scheme, such as an e-mail address, IPX, or IPv6. Since the node ID includes a Supernet ID, a node will have more than one node ID when it communicates over more than one channel. The "real address" is an IP address (e.g., 10.0.0.2) that is globally unique and meaningful to the public-network infrastructure. In a Supernet, one VARPD runs on each machine, and it may play two roles. First, a VARPD may act as a server by storing all address mappings for a particular Supernet into its associated VARPDB. Second, regardless of its role as a server or not, each VARPD assists in address translation for the nodes on its machine. In this role, the VARPD stores into its associated VARPDB the address mappings for its nodes, and if it needs a mapping that it does not have, it will contact the VARPD that acts as the server for the given Supernet to obtain it. The VARPDB may also decide which virtual address to use in the translation. That is, the VARPDB may associate a virtual address with multiple real addresses or vice versa." Caronni I at 7:3–33.

	"FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:66-10:18. "Once inner IP layer 542 receives the packet, a Supernet ID is appended to a socket structure (step 804). The socket structure is modified so as to contain an extra data field for Supernet ID 626 and virtual source address 642. The addition of Supernet ID 626 and virtual address 642 in the socket structure enables the Supernet to communicate with nodes regardless of the delivery scheme used. When the process on node A opens a socket to transmit the packet to inner IP layer 540, the corresponding Supernet ID 626 and virtual source address 642 for that process is included in the socket request. The packet and Supernet ID are then transmitted to the SNSL layer using the modified socket structure (step 806). The SNSL layer then accesses the VARPDB to obtain the address mapping between virtual source node address 642 and the source real address 614 as well as the virtual destination node address 644 and the destination real address 616 (step 808). If they are not contained in the VARPDB because this is the first time a packet has been sent from this node or sent to this destination, the VARPDB accesses the local VARPD to obtain the mapping. When contacted, the VARPD on the local machine contacts the VARPD that acts as the server for the Supernet to obtain the appropriate address mapping. Since the

	VARPDB maintains all real IP addresses, a remote node may securely communicate with another remote node without reverfication." Caronni I at 11:37–61. *See also* Caronni I at FIGS. 7A and 7B. "Unfortunately, when the virtual destination address is located on a physical device on the interior of a network which is running a proxy server, firewall, or other packet filtering mechanism, messages that have been sent to a virtual destination address have difficulty getting all the way to their target. The term "interior of a network" refers to devices which are not able to directly access another network without first going through another device on their own network. For example, most local area networks (LANs) access the Internet through a proxy server. Devices other than the proxy server are said to be on the interior of the LAN. The proxy server is referred to as an "edge device" because it is able to directly contact another network without using an intermediary device. "Packet filtering" refers to the filtering of incoming messages or packets by an edge device or process on an edge device so that not all of the packets are permitted to proceed to their destination, they are "filtered out". If the electronic device that is filtering incoming packets, is under the control of the party executing the process associated with the virtual destination address, the device may be configured to allow the packets through to the end destination. However, in many situations, the edge device is not configurable by anyone without system administration privileges. Similarly, if the edge device is a device performing Network Address Translation (a "NAT box"), the NAT box rewrites all outgoing packets from an end user in the interior of the network to make them look like they came directly from the NAT box, and remembers that any traffic coming back from the particular destination address must be mapped back to the originating internal device. Consequently, the responding devices think they are responding to the sending device when they are actually responding to an edge device. In such a case, the packets intended for the virtual destination address on the interior of the physical network may be dropped and not reach their intended destination." Caronni II at 1:60–2:27.

	"…sending said formatted message, augmented by said determined MAC address destination, said real IP address destination and said transport protocol header, from a virtual IP address to the real IP address destination indicated in said virtual address destination registration." Caronni II at 9:20–24. "…a process associated with the sending virtual IP address is located on an electronic device outside a firewall and said virtual address destination is located inside the firewall." Caronni II at 9:31–32. *See also* Caronni II at 2:31-3:24, 6:32-54, FIGS. 1-5. "The Virtual Network Environment (VNE) of the present invention is defined by a collection of IP addresses. An application running within one VNE can communicate with another application in the same VNE. However, an application in one VNE cannot communicate with an application in another VNE (unless expressly permitted). These and many other attendant advantages of the present invention will be understood upon reading the following detailed description in conjunction with the drawings." Hipp at 2:8-16. "The VNE is specified at application run time. The VNE is transparent to the application and does not require any modifications to the application. The VNE is defined by subnet of addresses contained within the VNE. For example, all applications within the subnet 10.10.2.0 comprise a VNE. The subnet/netmask specifying such a VNE would be 10.10.2.0/255.255.255.0 and would include the addresses 10.10.2.0 through 10.10.2.255. In this example, an application with IP address 10.10.2.2 would be able to communicate with an application at address 10.10.2.60, but not at 10.10.0.1. Although using a subnet/netmask to specify the VNE is described herein for illustrative purposes, it is to be understood that other methods may be used to accomplish the same mechanism (e.g. an access control list)." Hipp at 3:16-30.

	"Virtualization of network identity is achieved by assigning a unique virtual IP address and virtual hostname to a group of processes that make up the application instance which the instance keeps throughout its execution. This virtual network identity stays with the application instance regardless of which node the application is running on. The framework, in essence, provides a mechanism to create this virtual network identity (VNI) around the application using the virtual network parameters assigned to it. In one embodiment, the virtual network parameters include an IP address and hostname. The framework 200 ensures that the application's instance uses the virtual network parameters, transparently, so that it can be moved across machines, without modifications to the application." Hipp at 6:1-15. *See also* Hipp at Figures 1–7.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	Cisco VPN Routers and Software discloses "a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device." *See, e.g.*, Cisco systems used routing protocols and address translation, which relates to address management. "Routing Protocols: RIP, RIP2, OSPF, Static, Automatic endpoint discovery, Network Address Translation (NAT), Classless Interdomain Routing (CIDR)." (Cisco VPN 3000 Series Concentrator Data Sheet). To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Mehta, RFC 1383, Caronni I, Caronni II, and/or Hipp. Mehta, RFC 1383, Caronni I, Caronni II, and/or Hipp

	concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, use of a server such as a DNS server that returns network addresses was a well-known and commonly-used technique. Modifying this reference with Mehta's, RFC 1383's, Caronni I's, Caronni II's, and/or Hipp's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with those of Mehta, RFC 1383, Caronni I, Caronni II, or Hipp for example, to incorporate this limitation. *See, e.g.*, "A network that supports IP can be connected to another TCP/IP-based or UDP/IP-based internet or the Internet, by providing a router which forwards data to another router or host machine based upon an IP address. The router (or routing server/service) typically contains a routing table, which determines to which machine (and optionally to which port) to send a datagram, given a destination IP address. The IP address uniquely identifies a router/host machine, and, in a typical TCP/IP network, can be mapped to a string name that identifies, for example, a particular machine as part of a larger domain. (Although referred to herein often as a TCP/IP-based network, one skilled in the art will recognize that the network may also be UDP-based (connectionless), and may support another session management system.)" Mehta at [0008]. "In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more Address Management Data Repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the external public network to send data to the wireless device. The temporary public address is, for example, an address associated with one of the Address Proxy/Routers, which are connected to the external public network and have access to the private addresses on the private wireless network. In some cases, the device on the public network that desires to send data to the

	wireless device uses a connection-based protocol, such as TCP/IP to send data. In other cases the device uses a connection-less protocol, such as UDP (UDP/IP) to send the data." Mehta at [0012]. "Wireless systems on private networks use techniques similar to Network Address Translation (NAT) technology to send data from a wireless device to the public networking world. In a typical carrier infrastructure that uses a private network, a wireless device "registers" itself with the carrier infrastructure when it powers on (or in other circumstances, when the device attempts to initiate data services). The carrier dynamically assigns the wireless device a private non-routable address by means of DHCP (or DHCP-like) server. The information on the mapping of the transient private IP address to the device public IP address is stored in an internal carrier database and managed by carrier services such as a RADIUS server." Mehta at [0026]. "The Address Management Proxy System achieves two way initiated bi-directional communication by implementing a modified DNS server and serving as a proxy/router for devices on the private wireless network as they interface to the public wired internetworking world. In summary, a pool of public addresses is maintained and dynamically distributed among active wireless devices as needed by the AMPS. FIG. 2 is a block diagram of an example Address Management Proxy System used in bi-directional communication with a wireless device. The term "bi-directional" as used herein means that data paths and communication can flow in either direction between two endpoint systems. FIG. 2 shows wired devices 201, 202, and 203 connected to public network 210. One skilled in the art will recognize that these devices could be connected to another private or public network, which is then connected by one or more wired devices to the public network 210, yet still achieve the functionality discussed here. Any such variation provides equivalent functionality and is explicitly contemplated and presumed to be part of the present invention. On the wireless side, the AMPS 220, acting in its capacity as a proxy (and router) for wireless devices, is shown both connected via wire to the public network 210 and via standard carrier infrastructure elements (not shown) to the wireless network 230. It is presumed that the

	reader has a working knowledge of the elements of a carrier's infrastructure and the basic mechanisms for routing and mechanisms converting packets from a wired network to a wireless network. These may use analog or digital technology and may require protocol conversions in order to send the physical data and transmit it, for example through a satellite, ultimately to the wireless device. Detailed background information on wireless technology and wireless routing mechanisms is described in Stallings, W., Wireless Communications and Networks, Prentice Hall, N.J., 2002, which is herein incorporated by reference in its entirety. In FIG. 2, wireless device 240 is shown connected via various wireless elements (not shown) to the wireless network 230." Mehta at [0030]. "FIG. 6 is an example block diagram of some of the Address Management Proxy System data repository tables used to support routines of the DNS'/API servers and the Address Proxy/Routers. In one embodiment, the Address Management Data Repository comprises three tables: a unique identifier (unique ID) to private address table 610, a public-to-private address table 630, and a public address to proxy/router machine table 620. Although three tables are shown, one skilled in the art will recognize that these tables could contain other data and may be organized differently, including a different number of tables and with different columns or fields. In addition, any technique for storing a table or list of data may be used. To support embodiments that map a host address plus a port designator to a wireless device, the tables are correspondingly modified." Mehta at [0052]. "The unique ID table 610 maps unique string names of wireless devices to the private wireless network addresses that have been assigned typically by a carrier's infrastructure. In some embodiments, the carrier infrastructure dynamically allocates, using methods similar to a DHCP protocol, a private wireless network address when the wireless device registers itself with the carrier infrastructure upon being powered on. Thus, the unique ID table 610 may be sparsely filled or entries created dynamically and then deleted dynamically as devices register and unregister with the carrier infrastructure system." Mehta at [0053].

	The public-to-private address table 630 comprises several fields/columns including a public network address 631, a private (wireless) network address 602, a flag 632 that specifies whether the public address stored in field 631 is free or is already used, a Time to Live (TTL) parameter 633, and other connection data 634. In one embodiment, the DNS'/API servers of the AMPS query table 630 to determine a public network address that corresponds to a designated private wireless network address or to allocate an unused public network address (as indicated in field 632) and map the determined unused public address to a private network address stored in field 602. Mehta at [0054]. Public address to proxy/router machine table 620 comprises a public network address field 631 and an indication of a functioning proxy/router machine 621. By maintaining such a mapping, the AMPS is able to substitute proxy/router machines for other proxy/router machines to provide a higher degree of robustness. Each proxy/router machine has a preconfigured set of public network addresses, such as are typically configured by network cards inserted into the proxy/router machine. These address are allocated in a standard fashion through prior purchase or obtaining from an address authorizing authority, currently the Internet Corporation for Assigned Names and Numbers (ICANN). When a machine is inserted for use into the AMPS, indications of these addresses are stored in field 631. Mehta at [0055]. "In one embodiment, a timestamp of the mapping between a public network address and a private address is also noted in table 630. After a defined time out, based upon this timestamp, the Address Management Data Server sends a request to the Address Proxy/Router associated with the mapping to unmap the public-to-private address mapping and updates the mapping table 630, thereby returning the associated public address to the pool of unused public network addresses." Mehta at [0056].

	"FIG. 7 is an example flow diagram of an example routine provided by a DNS'/API server of the Address Management Proxy System to return a public address that corresponds to a designated unique identifier. In essence, this routine implements a DNS query or DNS-like query capability for the AMPS using a modified GetHostByName interface or a specialized API, for example GetProxyIP, as described with reference to FIG. 5. In summary, the routine dynamically allocates an appropriate Address Proxy/Router machine to associate with the wireless device and returns the public address of that machine (along with a TTL parameter and potentially other parameters). Specifically, in step 701, the routine determines the private (non-routable) address of the wireless device designated by a string parameter passed as input to the routine. For example, the string parameter may use fields such as "uniqueID.hostname.domain.tld," which specifies a typical hierarchy of person/service on a machine named "hostname" on a domain such as a company's network on a top level domain such as "org." "com," "edu," etc. One skilled in the art will recognize that many other string parameter designations could be used. One mechanism for implementing this routine is to request information from the Address Management Data Repository. In one embodiment, the data repository stores a table that maps unique IDs to private network addresses (see Table 610 in FIG. 6). Preferably, any mechanism that is used by the AMPS stores this data in a secure manner in order to keep the wireless network addresses private. In step 702, the routine retrieves the public network address that corresponds to the private wireless network address of the designated device if one has already been assigned by the AMPS to that device and is still valid. In one embodiment, the data repository stores this mapping information between private wireless network address and public network address (see for example table 630 in FIG. 6). If a public network address has not already been assigned or is not valid, then the routine causes a new public network address to be allocated and that new public address is associated with the private wireless network address. Appropriate tables in the data repository are then updated. In step 703, the routine determines the Address Proxy/Router machine that is associated with the assigned public network address (for example using table 620 in FIG. 6). In step 704, the routine sends a request to the determined proxy/router machine to update its routing tables to map the determined public network address to the private wireless network address. In step 705, the routine updates information in the data repository to include any other connection related information (e.g., field 634 in Table 630 in FIG. 6) and indicates

a Time to Live (TTL) parameter for the public-private address association (e.g., field 633 in Table 630 in FIG. 6). Once all of the tables have been updated in both the proxy/router and in the data repository, the DNS'/API server returns the determined public network address of the associated Address Proxy/Router machine. As described earlier, the public address may be a (hostname, port) pair when a port-based implementation is used."

Mehta at [0057].

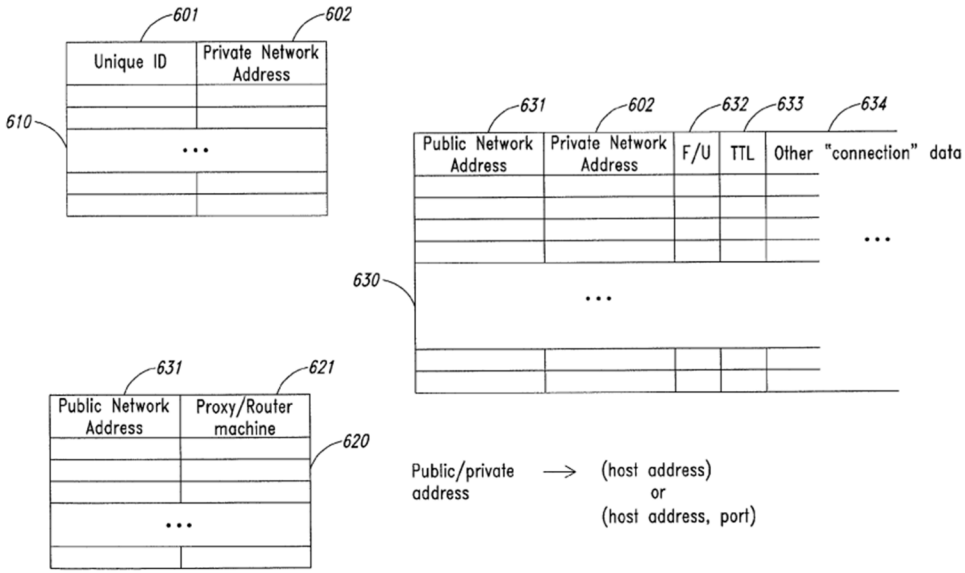


Fig. 6

Mehta at FIG. 6.

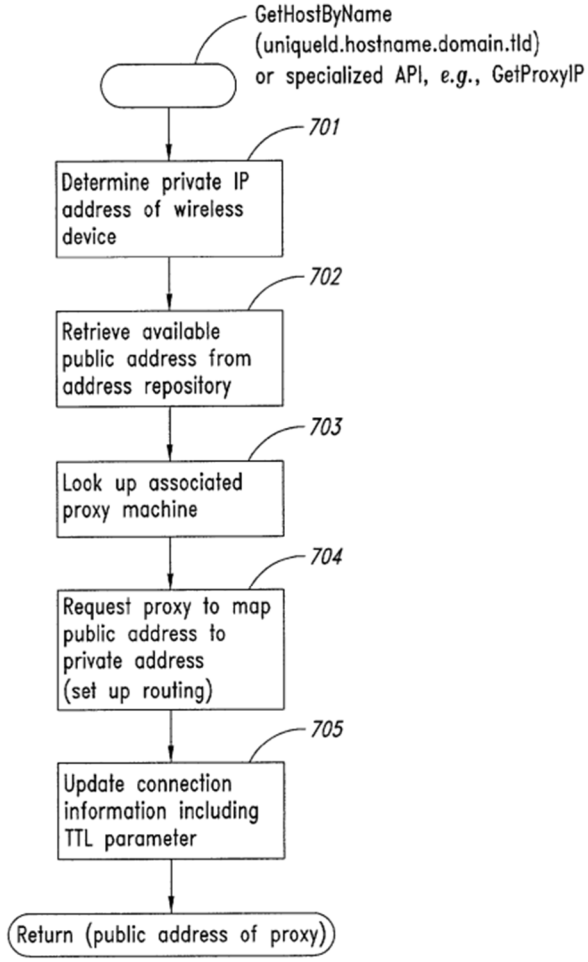

GetHostByName
(uniqueId.hostname.domain.tld)
or specialized API, e.g., GetProxyIP
701
Determine private IP address of wireless device
702
Retrieve available public address from address repository
703
Look up associated proxy machine
704
Request proxy to map public address to private address (set up routing)
705
Update connection information including TTL parameter
Return (public address of proxy)

Fig. 7

	Mehta at FIG. 7. *See also* Mehta at [0004], [0010], [0032], [0036]-[0038], [0048], [0058]-[0059]. "1. Routing, scaling and hierarchies Several recent studies have outlined the risk of "routing explosion" in the current Internet: there are already more than 5000 networks announced in the NSFNET routing tables, more than 7000 in the EBONE routing tables. As these numbers are growing, several problems occur: * The size of the routing tables grows linearly with the number of connected networks; handling this larger tables requires more resources in all "intelligent" routers, in particular in all "transit" and "external" routers that cannot rely on default routes. * The volume of information carried by the route exchange protocols such as BGP grows with the number of networks, using more network resources and making the reaction to routing events slower. * Explicit administrative decisions have to be exercised by all transit networks administrators which want to implement "routing policies" for each and every additional "multi-homed" network. The current "textbook" solution to the routing explosion problem is to use "hierarchical routing" based on hierarchical addresses. This is largely documented in routing protocols such as IDRP, and is one of the rationales for deploying the CIDR [3] addressing structure in the Internet. This textbook solution, while often perfectly adequate, as a number of inconveniences, particularly in the presence of "multihomed stubs", e.g., customer networks that are connected to more than one service providers.

	The current proposal presents a scheme that allows for simple routing. It is complementary with the classic "hierarchical routing" approach, but provides an easy to implement and low cost solution for "multi-homed" domains. The solution is a generalization of the "MX record" scheme currently used for mail routing." RFC 1383 at 1-2. "3.1. Loops and relays In the introduction to DNS-IP routing, we mentioned that the packets would be directed towards the access gateway I1 or I2 by means of "source routing" or "tunnelling". This is not, stricto sensu, necessary. One could imagine that the packet would simply be routed "as if it was directed towards I1 or I2". The next relay would, in turn, also access the DNS to get routing information and forward the packet. Such a strategy would have the advantage of leaving the header untouched and of letting the transit nodes choose the best routing towards the destination, based on their knowledge of the reachability status. It would however have two important disadvantages: - It would oblige all intermediate relays to access the DNS, - It would oblige all these relays to exploit consistently the DNS information. Obliging all intermediate gateways to access the DNS is impractical in the short term: it would mean that we would have to update each and every transit relay before deploying the scheme. It could also have an important performance impact: the "working set" of transit relays is typical much wider than that of stub gateways, and the argument presented previously on the efficiency of caches may not apply. This would perhaps remain impractical even in the long term, as it the volume of DNS traffic could well become excessive. The second argument would apply even if the performance problem had been solved. Suppose that several RX records are registered for a given destination, such as I1 and I2 for

	Dx in our example, and that a "hop by hop routing" strategy is used. There would be a fair risk that some relays would choose to route the packet towards I1 and some others towards I2, resulting in inefficient routing and the possibility of loops. In order to ensure coherency, we propose that all routing decisions be made at the source, or by one of the first relays near the source." RFC 1383 at 4-5. "3.4. Choosing a gateway A simplification to the previous problem would be to allow only one RX record per destination, thus guaranteeing consistent decisions in the network. This would however have a number of draw-backs. A single access point would be a single point of failure, and would be connected to only one transit network thus keeping the "customer locking" effect of hierarchical routing. We propose that the RX records have a structure parallel to that of MX records, i.e., that they carry associated with each gateway address a preference identifier. The source host, when making the routing decision based on RX records, should do the following: - List all possible gateways, - Prune all gateways in the list which are known as "unreachable" from the local site, - If the local host is present in the list with a preference index "x", prune all gateways whose preference index are larger than "x" or equal to "x". - Choose one of the gateway in the list. If the list is empty, consider the destination as unreachable.

Indeed, these evaluations should not be repeated for each and every packet. The routers should maintain a cache of the most frequently used destinations, in order to speed up the processing."

RFC 1383 at 6.

"5.1. DNS record

In a definitive scheme, it would be necessary to define a DNS record type and the corresponding "RX" format. In order to deploy this scheme, we would then have to teach this new format to the domain name service software. While not very difficult to do, this would probably take a couple of month, and will not be used in the early experimentations, which will use the general purpose "TXT" record.

This record is designed for easy general purpose extensions in the DNS, and its content is a text string. The RX record will contain three fields:

- A record identifier composed of the two characters "RX". This is used to disambiguate from other experimental uses of the "TXT" record.

- A cost indicator, encoded on up to 3 numerical digits. The corresponding positive integer value should be less that 256, in order to preserve future evolutions.

- An IP address, encoded as a text string following the "dot" notation.

The three strings will be separated by a single comma. An example of record would thus be:

domain	type	record	value
*.27.32.192.in-addr.arpa	IP	TXT	RX, 10, 10.0.0.7

which means that for all hosts whose IP address starts by the three octets "192.32.27" the IP host "10.0.0.7" can be used as a gateway, and that the preference value is 10."

	RFC 1383 at 11-12. *See also* RFC 1383 at 9, 12-13. "To perform this functionality, D 1 108 utilizes a technique known as tunneling to ensure that the communication between itself and enterprise network 102 is secure in that it cannot be viewed by an interloper. "Tunneling" refers to encapsulating one packet inside another when packets are transferred between two end points (e.g., D 1 108 and VPN software 109 running on firewall 106). The packets may be encrypted at their origin and decrypted at their destination. For example, FIG. 2A depicts a packet 200 with a source Internet protocol (IP) address 202, a destination IP address 204, and data 206. It should be appreciated that packet 200 contains other information not depicted, such as the source and destination port. As shown in FIG. 2B, the tunneling technique forms a new packet 208 out of packet 200 by encrypting it and adding both a new source IP address 210 and a new destination IP address 212. In this manner, the contents of the original packet (i.e., 202, 204, and 206) are not visible to any entity other than the destination. Referring back to FIG. 1, by using tunneling, remote device D 1 108 may communicate and utilize the resources of the enterprise network 102 in a secure manner." Caronni I at 2:26–46. "SASD 540 represents a Supernet: there is one instance of an SASD per Supernet, and it both authenticates nodes and authorizes nodes to join the Supernet. VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as a node IDs, to the network addresses recognized by the public-network infrastructure, known as the real addresses. The "node ID" may include the following: a Supernet ID (e.g., 0x123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1). Although the virtual address is described in an IP address scheme, one skilled in the art will appreciate that the virtual address may be any other type addressing scheme, such as an e-mail address, IPX, or IPv6. Since the node ID includes a Supernet ID, a node will have more than one node ID when it communicates over more than one channel. The "real address" is an IP address (e.g., 10.0.0.2) that is globally unique and

	meaningful to the public-network infrastructure. In a Supernet, one VARPD runs on each machine, and it may play two roles. First, a VARPD may act as a server by storing all address mappings for a particular Supernet into its associated VARPDB. Second, regardless of its role as a server or not, each VARPD assists in address translation for the nodes on its machine. In this role, the VARPD stores into its associated VARPDB the address mappings for its nodes, and if it needs a mapping that it does not have, it will contact the VARPD that acts as the server for the given Supernet to obtain it. The VARPDB may also decide which virtual address to use in the translation. That is, the VARPDB may associate a virtual address with multiple real addresses or vice versa." Caronni I at 7:3–33. "SNSL 542 utilizes VARPDB 536 to perform address translation. VARPDB stores all of the address mappings encountered thus far by SNSL 542. If SNSL 542 requests a mapping that VARPDB 536 does not have, VARPDB communicates with the VARPD 526 on the local machine to obtain the mapping. VARPD 526 will then contact the VARPD that acts as the server for this particular Supernet to obtain it." Caronni I at 9:47–54. "FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file."

	Caronni I at 9:66-10:18. "The packet and Supernet ID are then transmitted to the SNSL layer using the modified socket structure (step 806). The SNSL layer then accesses the VARPDB to obtain the address mapping between virtual source node address 642 and the source real address 614 as well as the virtual destination node address 644 and the destination real address 616 (step 808). If they are not contained in the VARPDB because this is the first time a packet has been sent from this node or sent to this destination, the VARPDB accesses the local VARPD to obtain the mapping. When contacted, the VARPD on the local machine contacts the VARPD that acts as the server for the Supernet to obtain the appropriate address mapping. Since the VARPDB maintains all real IP addresses, a remote node may securely communicate with another remote node without reverfication." Caronni I at 11:47–61. "After obtaining the address mapping, the SNSL layer determines whether it has been configured to communicate over the appropriate channel for this packet (step 806). This configuration occurs when SNlogin runs, and if the SNSL has not been so configured, processing ends. Otherwise, SNSL obtains the channel key to be used for this channel (step 808). The SNSL maintains a local cache of keys and an indication of the channel to which each key is associated. Each channel key is time stamped to expire in ten seconds, although this time is configurable by the administrator. If there is a key located in the cache for this channel, SNSL obtains the key. Otherwise, SNSL accesses KMD which then locates the appropriate channel key from the appropriate KMC. After obtaining the key, the SNSL layer encrypts the packet using the appropriate encryption algorithm and the key previously obtained (step 810). When encrypting the packet, the virtual source node address 642, the virtual destination node address 644, and the data may be encrypted (addressing section 660), but the source and destination real addresses 614, 616 (delivery scheme section 670) are not, so that the real addresses can be used by the public network infrastructure to send the packet to its destination. By encrypting addressing scheme 660, the Supernet enables data to be transmitted securely and at the same time transparent from delivery scheme used." Caronni I at 11:62-12:19.

	"Web client 1102 has a virtual address obtained from computer system 1106, as described in FIGS. 7A and 7B. Each time web client 1102 requests a packet from web server 1104 a, the client requests the virtual address of the web server 1104 a from computer system 1106. If web server 1104 a becomes overloaded (e.g., unable to handle more requests), the overloaded web server spawns new instances of the same web server (web servers 1104 b and 1104 c) and at the same time notifies the computer system 1106. The VARP server of computer 1106 then associates the new instances 104 b and 104 c of the web server with web server's 1104 virtual address. As a result, web client 1102 is not notified of the change and continues to use the same virtual address as previously." Caronni I at 13:13–26. "In one embodiment, a virtual network supported by a physical network includes a virtual address resolution facility. The virtual address resolution facility is used to resolve a virtual IP address into a real IP address. A virtual address is registered with the virtual address resolution facility and the registration includes a real IP address, a port number and a transport protocol designation associated with a virtual address. The virtual address destination is resolved using the virtual address resolution facility and a message is sent from a virtual address in the virtual network to the real IP address indicated in the virtual address destination registration. In one aspect of the embodiment, the registration also includes an Application layer protocol designation." Caronni II at 2:52–64. "In an embodiment of the present invention a virtual network is supported by a physical network and includes a virtual address resolution facility. The virtual address resolution facility is used to resolve a virtual IP address into a real IP address. A registration request is received at the virtual address resolution facility which includes a real IP address, a port address, a transport protocol designation and an Application layer protocol designation associated with a virtual address destination. The associations are stored using the virtual address resolution facility. A resolution request is received referencing a virtual address

	destination and the resolution request is resolved using the virtual address resolution facility and the stored associations." Caronni II at 3:11–23. "The physical network 20 is also interfaced with an electronic device 30 located at the edge of a local area physical network 50 which includes devices addressable using IP addresses in the 129.63.1.0/24 range. The electronic device 30 may be a proxy server, gateway, NAT box, or other device and may perform packet filtering. Alternatively, a firewall 51 may be executed in either a hardware or software form to filter packets sent to the local area physical network 50 from the physical network 20. The firewall 51 may be located outside the electronic device 30 or alternatively may be running in software form on the electronic device 30. Those skilled in the art will recognize that a number of different types of networks may be utilized within the scope of the present invention. The networks may be the Internet, an intranet, wide area network (WAN), a local area network (LAN), a satellite network, a wireless network, or some other type of network capable of supporting a virtual network through the devices thereon." Caronni II at 4:17–33. "FIG. 2 depicts a block diagram of the VARP lookup table 26 used by the virtual address resolution facility 24. The VARP lookup table 26 may be stored at any location accessible over the network. The VARP lookup table 26 includes registered virtual destination addresses located in a column 78 and corresponding associations located in a column 79. Entries 80, 82, 84, 86 and 88 include the registered virtual destination address and information associated with the registered virtual destination address. Each entry in the VARP lookup table 26 will list the registering virtual IP address 90 and associated information including a real IP address 91, a transport protocol designation 92, a port number identification 93 and an Application layer protocol designation 94. The associations are provided at the time the virtual destination address is registered in the virtual network. For example, the entry 80 for the virtual IP address 10.0.0.12 (90) includes the association of a real IP address 152.70.0.1 (91), an associated Transport layer protocol designation, UDP 92, an associated port number, 6789 (93), and an Application layer protocol designation of 'none' 94."

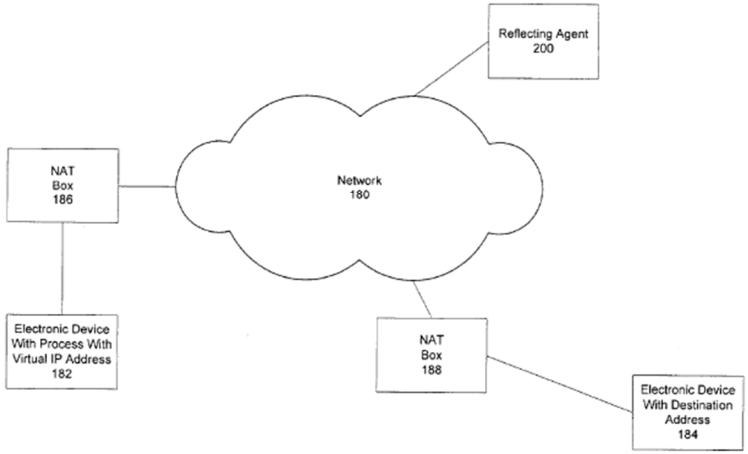
	Caronni II at 5:10–29. "The illustrative embodiment of the present invention attempts to circumvent problems caused by packet filtering and network address translation occurring at the edge of networks through the use of higher layer protocols. FIG. 3 depicts a block diagram of the encapsulation process used to send a message from a virtual IP address to another virtual IP address located at the interior of a network at which packet filtering and/or network address translation is occurring. The virtual address resolution facility 24 is used to determine the associations registered with the destination address. The original message 111 at the Application layer 110 is formatted in a format specified in the registration of the virtual destination address if one is so indicated (e.g. HTTP). The message 111 is passed down to the Transport layer 112. The Transport layer 112 takes the original message 111 and adds a transport protocol header 113 as specified in the virtual address destination registration. The transport protocol header 113 may be a UDP header, TCP header, an X.25 header, XTP header, AppleTalk header, or other similar transport protocol header. Those skilled in the art will realize that the message may include an IPSEC header, inner IP header, UDP or TCP header and inner payload." Caronni II at 6:32-53. "The illustrative embodiment of the present invention may also be utilized to send a message from a virtual address located behind a NAT box to a destination address on the same virtual network that is behind a different NAT box. Ordinarily, where the destination address is directly connected to the Internet (i.e.: in situations where the destination is not behind a NAT box), a connection, such as a TCP connection, may be established between the originator of the connection behind the NAT box and the destination. The originator's message passes the NAT box and the internal mapping of public address and port number to internal address and port number takes place. Return packets from the destination may then be received following the mapping. When the destination address is also behind a NAT box, the originator of the message is unable to directly address the destination (since the address of the destination may not be routable in the public Internet). In such a case, the present invention adds an entry to the VARP table for a third-party reflecting agent located at an address outside the NAT box. Messages that are being sent from the originating virtual address behind a NAT box to a

destination which is behind a different NAT box are sent via the reflecting agent intermediary outside the NAT box and reflected to the destination.

FIG. 5 depicts a block diagram of an environment holding a virtual network in which both the originator and destination of a message are located behind NAT boxes. An electronic device holding an originating process 182 which is part of a virtual network is interfaced with a network 180, such as the Internet, via a NAT box 186. An intended recipient for a message sent from the originating process 180 is an electronic device with a destination address 184 that is also behind a NAT box 188. A reflecting agent 200, the address of which has been added to a VARP table, acts as an intermediary to allow a connection between the originating process 182 and destination address 184."

Caronni II at 8:1–38.

Figure 5



Caronni II at FIG. 5.

"Beginning at step 270, a client application requests to the VNE framework 200 to connect or send to an address of another application, such as a server application having the address

10.10.2.70: port 9001. At step 272, the VNE framework 200 requests the VNE parameters for the process corresponding to the client application from a process state storage structure. The structure that stores process state is a structure that the operating system uses to store private information about the process. Therefore, it may differ depending on the operating system. The parameters added the process state storage structure as part of the virtual network environment are listed below:

```
typedef struct {
        ipaddr_t    app_address;
        ipaddr_t    virtual_subnet;
        ipaddr_t    virtual_mask;
        ipaddr_t    global_subnet;
        ipaddr_t    global_mask;
} vne_param_t;
```

Hipp at 4:44–64.

"Virtualization of network identity is achieved by assigning a unique virtual IP address and virtual hostname to a group of processes that make up the application instance which the instance keeps throughout its execution. This virtual network identity stays with the application instance regardless of which node the application is running on. The framework, in essence, provides a mechanism to create this virtual network identity (VNI) around the application using the virtual network parameters assigned to it. In one embodiment, the virtual network parameters include an IP address and hostname. The framework 200 ensures that the application's instance uses the virtual network parameters, transparently, so that it can be moved across machines, without modifications to the application.

The virtual hostname resolves to the virtual IP address for both the applications registered with the VNI framework as well as those that are not registered. This may require configuration of a name service or OS host configuration files. For example, if an

	application instance used a virtual IP address of 10.10.0.1 and a virtual hostname of host1055, the standard hostname to IP address resolution mechanisms (e.g. DNS or the /etc/hosts file) would have to be preconfigured to resolve a query of host1055 to IP address 10.10.0.1." Hipp at 6:1–26.
[35.0] The virtual network manager of claim 30 further comprising a join module configured to receive a join request from the agent associated with the device to indicate that the device is connected for data communication within the virtual network, the join module further configured to receive a leave request from the agent associated with the device to indicate that the device will be disconnected from data communication within the virtual network.	Cisco VPN Routers and Software discloses "a join module configured to receive a join request from the agent associated with the device to indicate that the device is connected for data communication within the virtual network, the join module further configured to receive a leave request from the agent associated with the device to indicate that the device will be disconnected from data communication within the virtual network." *See, e.g.*, The claim requires receiving a join request from a device. Cisco VPN systems support pre-configured client logons, which aligns with the concept of automated joining. "The client can be pre-configured for mass deployments and the initial logons require very little user intervention." (Cisco VPN 3000 Series Concentrator Data Sheet). To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Mehta and/or Caronni I. Mehta and Caronni I concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, joining devices with a virtual network was a well-known and commonly-used technique. Modifying this reference with Mehta's and/or Caronni I's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with that of Mehta or Caronni I for example, to incorporate this limitation. *See, e.g.*,

	"Wireless systems on private networks use techniques similar to Network Address Translation (NAT) technology to send data from a wireless device to the public networking world. In a typical carrier infrastructure that uses a private network, a wireless device "registers" itself with the carrier infrastructure when it powers on (or in other circumstances, when the device attempts to initiate data services). The carrier dynamically assigns the wireless device a private non-routable address by means of DHCP (or DHCP-like) server. The information on the mapping of the transient private IP address to the device public IP address is stored in an internal carrier database and managed by carrier services such as a RADIUS server." Mehta at [0026]. "FIG. 4 is an example block diagram of a general purpose computer system for practicing embodiments of the Address Management Proxy System. The general purpose computer system 400 may comprise one or more server and/or client computing systems and may span distributed locations. In addition, each block may represent one or more such blocks as appropriate to a specific embodiment or may be combined with other blocks. The various blocks of the Address Management Proxy System 410 may physically reside on one or more machines, which use standard interprocess communication mechanisms to communicate with each other. In the embodiment shown, computer system 400 comprises a computer memory ("memory") 01, a display 402, a Central Processing Unit ("CPU") 403, and Input/Output devices 404. The Address Management Proxy System 410 is shown residing in memory 401. The components of the Address Management Proxy System 410 preferably execute on CPU 403 and manage the address mapping of wireless devices on a wireless network, as described in previous figures, to allow other wired systems to communicate with the wireless devices. Other downloaded code 405 and potentially other data repositories also reside in the memory 410, and preferably execute on one or more CPU's 403. In a typical embodiment, the AMPS 410 includes one or more DNS'/API servers 411, one or more Address Proxy/Routers 412, an Address Management Data Server 413, and Address Management Data Repositories 414. As described earlier, the AMPS may include other data repositories and components, such as a load balancer, depending upon the particular implementation."

	Mehta at [0036]. "The unique ID table 610 maps unique string names of wireless devices to the private wireless network addresses that have been assigned typically by a carrier's infrastructure. In some embodiments, the carrier infrastructure dynamically allocates, using methods similar to a DHCP protocol, a private wireless network address when the wireless device registers itself with the carrier infrastructure upon being powered on. Thus, the unique ID table 610 may be sparsely filled or entries created dynamically and then deleted dynamically as devices register and unregister with the carrier infrastructure system." Mehta at [0053]. "FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:67–10:18. "FIG. 10 depicts a flow chart of the steps performed when logging a node out of a Supernet. The first step performed is for the user to run the SNlogout script and to enter a node ID (step 1002). Next, the SNlogout script requests a log out from SASD (step 1004). Upon receiving this request, SASD removes the mapping for this node from the VARPD that acts as the server for the Supernet (step 1006). SASD then informs KMS to cancel the registration of the

	node, and KMS terminates this KMC (step 1008). Lastly, KMS generates a new channel key for the channels on which the node was communicating (step 1010) to reduce the likelihood of an intruder being able to intercept traffic." Caronni I at 12:61–13:5. *See also* Caronni I at FIG. 10.
[37.0] The virtual network manager of claim 35 wherein the join module is further configured to maintain data to associate a virtual network address with a device in the virtual network.	Cisco VPN Routers and Software discloses "wherein the join module is further configured to maintain data to associate a virtual network address with a device in the virtual network." *See, e.g.*, The claim requires maintaining data associating a virtual network address with a device. Cisco systems manage address associations through centralized policy management and VPN encryption modules. "VPN access policies are created and stored centrally in the Cisco VPN 3000 Concentrator and pushed to the client when a connection is established." (Cisco VPN 3000 Series Concentrator Data Sheet). "The VPN encryption modules handle a variety of IPSec-related tasks, including encryption, hashing, key exchange, storage of security associations." (Cisco 2600 and 3600 VPN Router Bundles). To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Mehta and/or Caronni I. Mehta and Caronni I concerns similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, joining devices with a virtual network

	was a well-known and commonly-used technique. Modifying this reference with Mehta's and/or Caronni I's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with that of Mehta or Caronni I for example, to incorporate this limitation. *See, e.g.*, "In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more Address Management Data Repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the external public network to send data to the wireless device. The temporary public address is, for example, an address associated with one of the Address Proxy/Routers, which are connected to the external public network and have access to the private addresses on the private wireless network. In some cases, the device on the public network that desires to send data to the wireless device uses a connection-based protocol, such as TCP/IP to send data. In other cases the device uses a connection-less protocol, such as UDP (UDP/IP) to send the data." Mehta at [0012]. "FIG. 3 is an example block diagram of components of an example Address Management Proxy System. In one embodiment, the Address Management Proxy System (AMPS) comprises one or more modified DNS'/ API servers 302, one or more Address Proxy/Routers 305, an Address Management Data Server 303 which manages a database or other repositories such as Address Management Data Repository 304, and optionally a load balancer 301. The DNS'/API servers 302 are either individually connected to a public network 310 or are connected to the load balancer 301 which is in turn connected to public network 310. Similarly, each Address Proxy/Router 305 is also connected to the public network 310, via the routable (public) address to which data from the external network 310 is sent that is destined for the wireless devices. The DNS'/API servers 302 are either modified implementations of a DNS server to add functionality necessary to communicate with

	wireless devices, or are servers that implement one or more specialized APIs, as will be described further below. The DNS'/API servers 302 use the Address Management Data Server 303 to assist in mapping a unique identifier (e.g., string name) for a wireless device to a public address on public network 310. The pool of public addresses is also maintained by the Address Management Data Serve 303 and Data Repository 304. The Address Management Data server 303 and Address Management Data Repository 304 may be implemented using existing database technology, for example, ODBC technology or, may be implemented as a structure such as a simple text file. One skilled in the art will recognize that any embodiment for storing a set of tables, data, lists, or mappings can be used. Each Address Proxy/Router 305 also uses the Address Management Data Server 303 or equivalent to create and update a series of routing tables that are used to assign public addresses to wireless devices as needed and to update the various mappings between public addresses and the non-routable (private) addresses of the wireless devices. The tables and mappings that are maintained on behalf of the DNS'/API servers 302 and the Address Proxy/Routers 305 by the Address Management Data Server 303 are described below with reference to FIG. 6." Mehta at [0032]. "One skilled in the art will recognize that the AMPS 410 may be implemented in a distributed environment that is comprised of multiple, even heterogeneous, computer systems and networks. For example, in one embodiment, the DNS'/API servers 411, the Address Proxy/Router components 412, and the Address Management Data Servers 413 with their data repositories 414 are all located in physically different computer systems. In another embodiment, various components of the AMPS 410 are hosted each on a separate server machine and may be remotely located from the tables which are stored in the address management data repository 414. In addition, under some scenarios, the entire AMPS system 410 may be hosted within a carrier's infrastructure and be completely subsumed by it. Different configurations and locations of programs and data are contemplated for use with techniques of the present invention. In example embodiments, these components may execute concurrently and asynchronously; thus the components may communicate using well-known message passing techniques. One skilled in the art will recognize that equivalent synchronous embodiments are also supported by an AMPS implementation. Also, other steps could be

	implemented for each routine, and in different orders, and in different routines, yet still achieve the functions of the AMPS." Mehta at [0038]. "FIG. 7 is an example flow diagram of an example routine provided by a DNS'/API server of the Address Management Proxy System to return a public address that corresponds to a designated unique identifier. In essence, this routine implements a DNS query or DNS-like query capability for the AMPS using a modified GetHostByName interface or a specialized API, for example GetProxyIP, as described with reference to FIG. 5. In summary, the routine dynamically allocates an appropriate Address Proxy/Router machine to associate with the wireless device and returns the public address of that machine (along with a TTL parameter and potentially other parameters). Specifically, in step 701, the routine determines the private (non-routable) address of the wireless device designated by a string parameter passed as input to the routine. For example, the string parameter may use fields such as "uniqueID.hostname.domain.tld," which specifies a typical hierarchy of person/service on a machine named "hostname" on a domain such as a company's network on a top level domain such as "org." "com," "edu," etc. One skilled in the art will recognize that many other string parameter designations could be used. One mechanism for implementing this routine is to request information from the Address Management Data Repository. In one embodiment, the data repository stores a table that maps unique IDs to private network addresses (see Table 610 in FIG. 6). Preferably, any mechanism that is used by the AMPS stores this data in a secure manner in order to keep the wireless network addresses private. In step 702, the routine retrieves the public network address that corresponds to the private wireless network address of the designated device if one has already been assigned by the AMPS to that device and is still valid. In one embodiment, the data repository stores this mapping information between private wireless network address and public network address (see for example table 630 in FIG. 6). If a public network address has not already been assigned or is not valid, then the routine causes a new public network address to be allocated and that new public address is associated with the private wireless network address. Appropriate tables in the data repository are then updated. In step 703, the routine determines the Address Proxy/Router machine that is associated with the assigned public network address (for example using table 620 in FIG.

	6). In step 704, the routine sends a request to the determined proxy/router machine to update its routing tables to map the determined public network address to the private wireless network address. In step 705, the routine updates information in the data repository to include any other connection related information (e.g., field 634 in Table 630 in FIG. 6) and indicates a Time to Live (TTL) parameter for the public-private address association (e.g., field 633 in Table 630 in FIG. 6). Once all of the tables have been updated in both the proxy/router and in the data repository, the DNS'/API server returns the determined public network address of the associated Address Proxy/Router machine. As described earlier, the public address may be a (hostname, port) pair when a port-based implementation is used." Mehta at [0057]. "Specifically, in step 801, the routine determines (for example, from the Address Management Data Repository) the private wireless address that corresponds to the invoked public address and the TTL parameter associated with this mapping. These values can be obtained, for example, from the public-to-private address table 630 of FIG. 6. In step 802, the routine determines whether the value of the determined TTL parameter indicates that the mapping has expired, and if so, returns an error, else continues in step 803. In step 803, the routine determines the format required by the target device (the wireless network format). In step 804, the routine determines whether any protocol conversion is necessary and, if so, continues in step 805 else continues in step 806. Note that protocol conversion for a specific wireless network such as converting the data to an "HTTP" packet may be done by the proxy/router or may be done by some other component within the carrier infrastructure. One skilled in the art will recognize that these are example steps and that different formatting or different protocol conversion routines may be added or omitted as specific to the environment. In step 806, the Address Proxy/Router routine sends the data packet (which has been formatted and whose protocol is converted as necessary) to the determined associated private address of the wireless device, and returns." Mehtat at [0059].

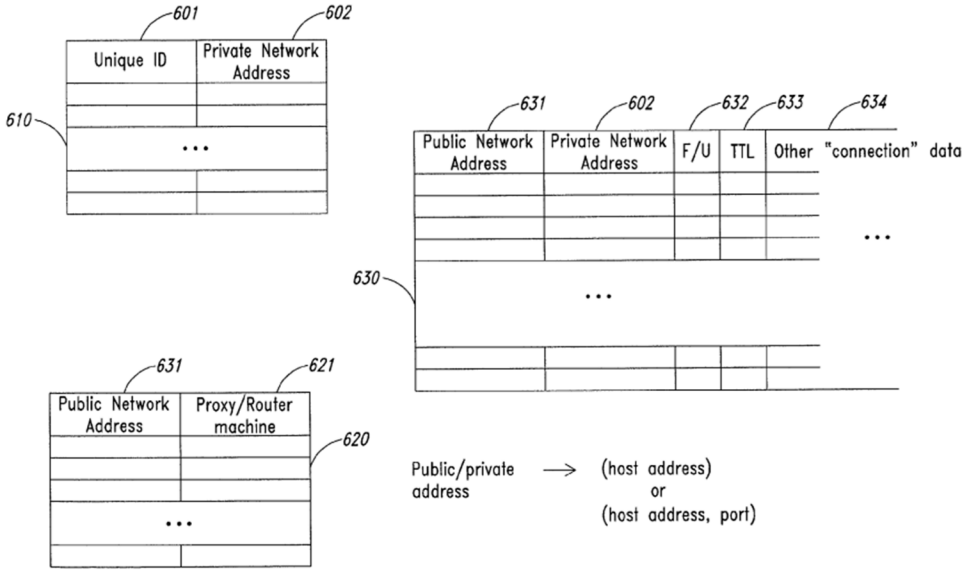


Fig. 6

Mehta at FIG. 6.

"SASD 540 represents a Supernet: there is one instance of an SASD per Supernet, and it both authenticates nodes and authorizes nodes to join the Supernet. VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as a node IDs, to the network addresses recognized by the public-network infrastructure, known as the real addresses. The "node ID" may include the following: a Supernet ID (e.g., 0x123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1). Although the virtual address is described in an IP address scheme, one skilled in the art will appreciate that the virtual address may be any other type addressing scheme, such as an e-mail address, IPX, or IPv6. Since the node ID includes a Supernet ID, a node will have more than one node ID when it communicates over more than one channel. The "real address" is an IP address (e.g., 10.0.0.2) that is globally unique and

	meaningful to the public-network infrastructure. In a Supernet, one VARPD runs on each machine, and it may play two roles. First, a VARPD may act as a server by storing all address mappings for a particular Supernet into its associated VARPDB. Second, regardless of its role as a server or not, each VARPD assists in address translation for the nodes on its machine. In this role, the VARPD stores into its associated VARPDB the address mappings for its nodes, and if it needs a mapping that it does not have, it will contact the VARPD that acts as the server for the given Supernet to obtain it. The VARPDB may also decide which virtual address to use in the translation. That is, the VARPDB may associate a virtual address with multiple real addresses or vice versa." Caronni I at 7:3–34. "FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:67–10:18. "FIG. 10 depicts a flow chart of the steps performed when logging a node out of a Supernet. The first step performed is for the user to run the SNlogout script and to enter a node ID (step 1002). Next, the SNlogout script requests a log out from SASD (step 1004). Upon receiving this request, SASD removes the mapping for this node from the VARPD that acts as the server for the Supernet (step 1006). SASD then informs KMS to cancel the registration of the

	node, and KMS terminates this KMC (step 1008). Lastly, KMS generates a new channel key for the channels on which the node was communicating (step 1010) to reduce the likelihood of an intruder being able to intercept traffic." Caronni I at 12:61–13:5. *See also* Caronni I at FIG. 10.

Exhibit 785-29 to Southwest's Preliminary Invalidity Contentions

VMware Systems

VMware Systems was known in this country, used in public, sold or offered for sale no later than 2001, as demonstrated by at least the materials cited herein or the testimony of knowledgeable witnesses and corroborating documents.[1] Specifically, based on information currently available, VMware Systems was (1) known or used in this country before the alleged invention of the claimed subject matter of the asserted claims, (2) in public use or on sale in this country more than one year before the filing date of the patent, or (3) invented by another who did not abandon, suppress, or conceal, before the alleged invention of the claimed subject matter of the asserted claim. Thus, VMware Systems anticipates the asserted claims under at least 35 U.S.C. §§ 102(a) or (g).

At least the following documents describe the relevant functionality disclosed by VMware Systems:

- VMware GSX Server 3.1 Administration Guide;
- VMware Delivers Open Interface to Virtual Infrastructure;
- VMware Enterprises Continue to Embrace VMware Virtual Infrastructure;
- VMware ESX Server Specifications;
- VMware GSX Server Specifications;
- VMware GSX Server 3.1 Virtual Machine Guide; and
- VMware Scripting API User's Manual.

Based on information currently available, the persons or entities involved in the conception and diligent reduction to practice of the ideas in VMware Systems include at least persons involved in the VMware Systems and their testing and commercial use, including at least VMware. The conception of VMware Systems occurred at least as early as 2001.

Southwest offers this invalidity chart based on its current understanding of VMware Systems. Southwest's investigation of this system is ongoing. Southwest may rely on materials produced by Plaintiffs or by third parties regarding VMware Systems and related software to demonstrate

[1] Defendant's invalidity charts, in some instances, rely at least in part on Plaintiff's apparent positions regarding the scope of its claims for purposes of asserting infringement. Nothing in these claim charts should be understood as an admission relating to infringement, either literally or under the doctrine of equivalents, or as an admission relating to Defendant's understanding of the proper interpretation or scope of the Asserted Claims. Defendant reserves the right to rely on additional citations or sources of evidence that may also be applicable, or that may become applicable in light of any Court Order on claim construction, changes in Plaintiff's infringement contentions, or information obtained during discovery as the case proceeds.

invalidity of the asserted claims. Southwest has not had an opportunity to review source code for VMware Systems. Southwest may rely on source code, when available, implementing the functionality described herein, or other source code implementing substantially similar functionality, to demonstrate the invalidity of the asserted claims.

To the extent Plaintiff alleges VMware Systems does not disclose any particular limitation of the Asserted Claims of the '785 Patent, either expressly or inherently, any purported differences are such that the claimed subject matter as a whole would have been obvious in view of the knowledge of one skilled in the art. It would have further been obvious to a person of ordinary skill in the art as of the priority date of the '785 Patent to modify VMware Systems or combine the teachings of VMware Systems with other prior art in a manner that would have rendered the Asserted Claims invalid as obvious, including but not limited to:

- U.S. Patent Application Publication No. 2003/0028671 ("Mehta");
- Huitema, C., Network Working Group Request for Comment (RFC): 1383, entitled An Experiment in DNS Based IP Routing ("RFC 1383");
- U.S. Patent No. 6,970,941 ("Caronni I");
- U.S. Patent No. 7,814,228 ("Caronni II"); and/or
- U.S. Patent No. 6,766,371 ("Hipp").

As the United States Supreme Court held in *KSR Int'l Co. v. Teleflex, Inc*., "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." 550 U.S. 398, 416 (2007). The Supreme Court further held: "[w]hen a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, § 103 likely bars its patentability. For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill." *Id*. at 417. The Supreme Court further held: "[w]hen there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense." *Id*. at 421.

Claims	VMware Systems
[30.0] A virtual network manager, comprising:	To the extent the preamble is limiting, VMware Systems discloses: The preamble requires a virtual network manager that manages virtual network resources. VMware's virtualization technology acts as a centralized control platform for managing virtual machines, consolidating resources, and automating management tasks. "VMware® GSX Server™ is virtual infrastructure for enterprise IT administrators who want to consolidate servers and streamline development and testing operations." (VMware GSX Server 3.1 Administration Guide) "VMware, Inc., the global leader in virtual infrastructure software for x86-based systems, today announced the availability of the VMware Virtual Infrastructure SDK. The SDK provides standards-based interfaces to control VMware virtual infrastructure." (VMware Delivers Open Interface to Virtual Infrastructure) "Virtual infrastructure provides a layer of abstraction between the computing, storage and networking hardware and the software that runs on it." (VMware Enterprises Continue to Embrace VMware Virtual Infrastructure) To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Mehta and/or Caronni I. Mehta and Caronni I concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, use of virtual networks was well-known and common. Modifying this reference with Mehta's and/or Caronni I's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with that of Mehta or Caronni I for example, to incorporate this limitation. *See, e.g.*,

	"Methods and systems for providing two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP, are provided. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporarily use by a requesting device on a public network to communicate with a wireless device on a wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more data repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the public network to send data to the wireless device." Mehta at Abstract. "The present invention relates to methods and systems for initiating communication with wireless devices and, in particular, to methods and systems for initiating communication with a device on a private network from a device on a public network to achieve virtual end-to-end connectivity." Mehta at [0002]. "Embodiments of the present invention provide computer- and network-based methods and systems for two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internetwork, such as the Internet, to

	initiate communication with and send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporary use by a requesting device on an external public network to communicate with a wireless device on a private wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. The mapping of a temporary public address to the private address of a wireless device is maintained and updated transparently by the AMPS using routing tables and other mapping data structures." Mehta at [0011]. "To insure a greater degree of security, according to one embodiment, the AMPS maintains a Time to Live (TTL) parameter with each address mapping. In this way, once the TTL value indicates that the mapping has expired, the AMPS can destroy the mapping, and also any connection. Further, in some embodiments, the AMPS also puts a timestamp in its own mapping tables. After some timeout period based upon the timestamp, the AMPS can destroy the mapping, thereby forcing a new mapping to be initiated on a periodic basis." Mehta at [0015]. "Embodiments of the present invention provide computer- and network-based methods and systems for two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices." Mehta at [0024]. "The second approach used to implement the AMPS supports full bi-directional communication through point-to-point connections established, for example, using TCP/IP protocol. (Note that these same techniques also support connection-less UDP bi-directional

communication). The second approach can be implemented by providing a modified implementation of a standard UDP or TCP/IP function, "GetHostByName." The GetHostByName API allows a string designation to identify the designated device and returns an IP address data structure. Alternatively, to increase the level of security provided, the AMPS can implement a specialized API to return the dynamically allocated public address that (now) corresponds to the requested wireless device. A disadvantage of the specialized API approach is that the device on the public network (or other device that wishes to obtain a connection to the wireless device) needs to include specialized code in the application on the requesting device."

Mehta at [0041].

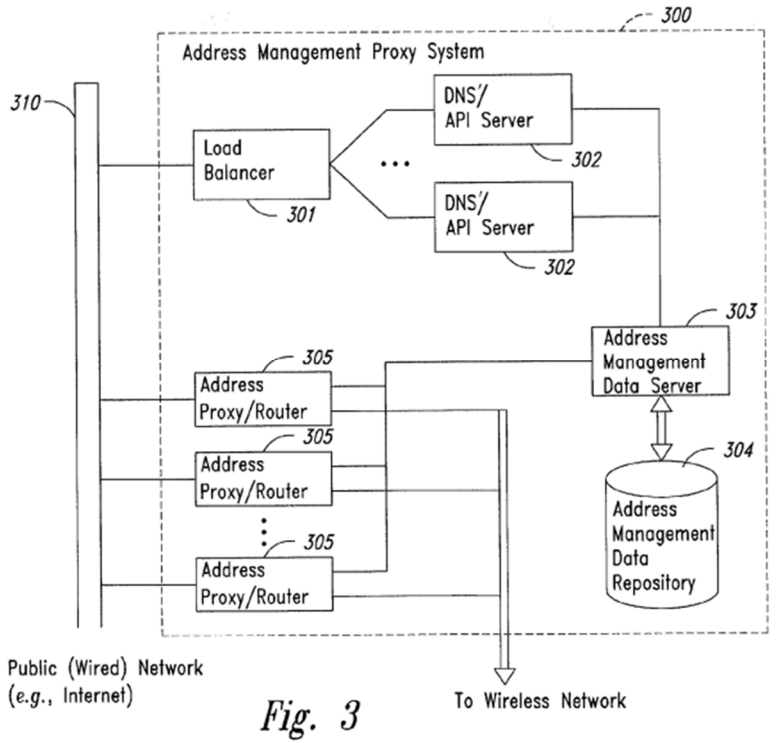


Fig. 3

	Mehta at FIG. 3. "Methods and systems consistent with the present invention establish a virtual network on top of current IP network naming schemes. The virtual network uses a separate layer to create a modification to the IP packet format that is used to separate network behavior from addressing. As a result of the modification to the packet format, any type of delivery method may be assigned to any address or group of addresses. The virtual network also maintains secure communications between nodes, while providing the flexibility of assigning delivery methods independent of the delivery addresses." Caronni I at Abstract. "The present invention relates generally to data processing systems and, more particularly, to a private network using a public-network infrastructure." Caronni I at 1:57–59. "Methods and systems consistent with the present invention overcome the shortcomings of existing networks by establishing a "Supernet," which is a private network that uses components from a public-network infrastructure. A Supernet allows an organization to utilize a public-network infrastructure for its enterprise network so that the organization no longer has to maintain a private network infrastructure; instead, the organization may have the infrastructure maintained for them by one or more service providers or other organizations that specialize in such connectivity matters. As such, the burden of maintaining an enterprise network is greatly reduced. Moreover, a Supernet is not geographically restrictive, so a user may plug their device into the Internet from virtually any portal in the world and still be able to use the resources of their private network in a secure and robust manner." Caronni I at 4:36-52. "Supernets also provide heterogeneous addressing functionality. The Supernet uses a separate layer that isolates address names of nodes from addressing schemes and delivery schemes. The Supernet contains a modification to the IP packet format that can be used to separate

	network behavior from addressing. As a result of the modification, any delivery scheme may be assigned to any address, or group of addresses." Caronni I at 4:53-59. *See also* Caronni I at Claim 1, FIGS. 3, 5.
[30.1] a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address;	VMware Systems discloses "a network interface configured for data communication via a virtual network that is defined by a domain name having an associated public network address." *See, e.g.*, VMware systems, such as GSX and ESX Server, support virtual Ethernet NICs and virtual network interfaces, which provide data communication in virtual environments. "Networking: Up to four virtual Ethernet NICs. Each virtual NIC may be high-performance VMware virtual NIC or AMD® PCnet™-PCI II compatible virtual NIC." (VMware ESX Server Specifications) "Ethernet Card: Up to four virtual Ethernet cards. AMD® PCnet™ -PCI II compatible. PXE ROM version 2.0." (VMware GSX Server 3.1 Administration Guide) "Virtual Networking and File Sharing: Nine virtual Ethernet switches (three reserved for bridged, host-only and NAT networking)." (VMware GSX Server Specifications) To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Mehta and/or Caronni I. Mehta and Caronni I concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, use of virtual networks was well-known and common. Modifying this reference with Mehta's and/or Caronni I's teachings would have only required simple substitution, would not have required extensive experimentation, and would

	have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with that of Mehta or Caronni I for example, to incorporate this limitation. *See, e.g.*, "The present invention relates to methods and systems for initiating communication with wireless devices and, in particular, to methods and systems for initiating communication with a device on a private network from a device on a public network to achieve virtual end-to-end connectivity." Mehta at [0002]. "Embodiments of the present invention provide computer- and network-based methods and systems for two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internetwork, such as the Internet, to initiate communication with and send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporary use by a requesting device on an external public network to communicate with a wireless device on a private wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. The mapping of a temporary public address to the private address of a wireless device is maintained and updated transparently by the AMPS using routing tables and other mapping data structures." Mehta at [0011]. "Although the techniques of the AMPS are generally applicable to any a wired device communicating with a wireless device, the phrase "public network" (or "wired network") is used generally to imply any type of internetworked environment including a public network or a backbone that is somewhere down the line connected to one or more private or public networks. In addition, although the examples described herein often refer to the Internet, one

	skilled in the art will recognize that the concepts and inventions described are applicable to other forms and embodiments of internetworking, including, for example ATM type networks. Thus, techniques of the present invention can also be used by one device on a first wireless network to communicate with another wireless device on a second network—each device ends up communicating with the Address Proxy/Router of the other network. This scenario is feasible because each wireless network (or its carrier infrastructure) is connected to a proxy/router that is also connected (via a public address) to a public network. In addition, although a public network is sometimes also referred to herein as a wired network, one skilled in the art will recognize that any network that exposes routable (public) addresses may be implied. Thus, a wireless network with unique public (and routable) address can also employ techniques of the present invention to perform bi-directional communication. Also, one skilled in the art will recognize that terms such as wireless device, phone, handheld, etc., are used interchangeably to indicate any type of wireless device that is capable of operating with the AMPS. In addition, terms may have alternate spellings which may or may not be explicitly mentioned, and one skilled in the art will recognize that all such variations of terms are intended to be included." Mehta at [0033]. "Public address to proxy/router machine table 620 comprises a public network address field 631 and an indication of a functioning proxy/router machine 621. By maintaining such a mapping, the AMPS is able to substitute proxy/router machines for other proxy/router machines to provide a higher degree of robustness. Each proxy/router machine has a preconfigured set of public network addresses, such as are typically configured by network cards inserted into the proxy/router machine. These address are allocated in a standard fashion through prior purchase or obtaining from an address authorizing authority, currently the Internet Corporation for Assigned Names and Numbers (ICANN). When a machine is inserted for use into the AMPS, indications of these addresses are stored in field 631." Mehta at [0055].

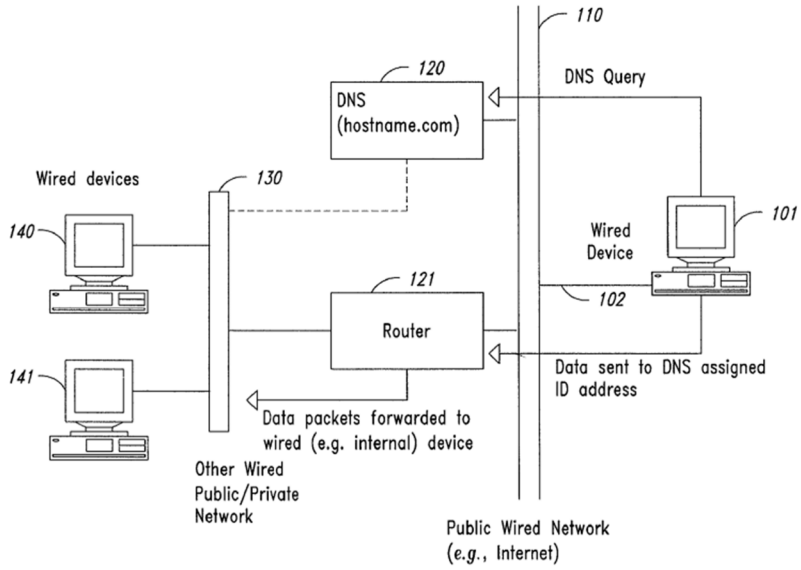


Fig. 1

Mehta at FIG. 1.

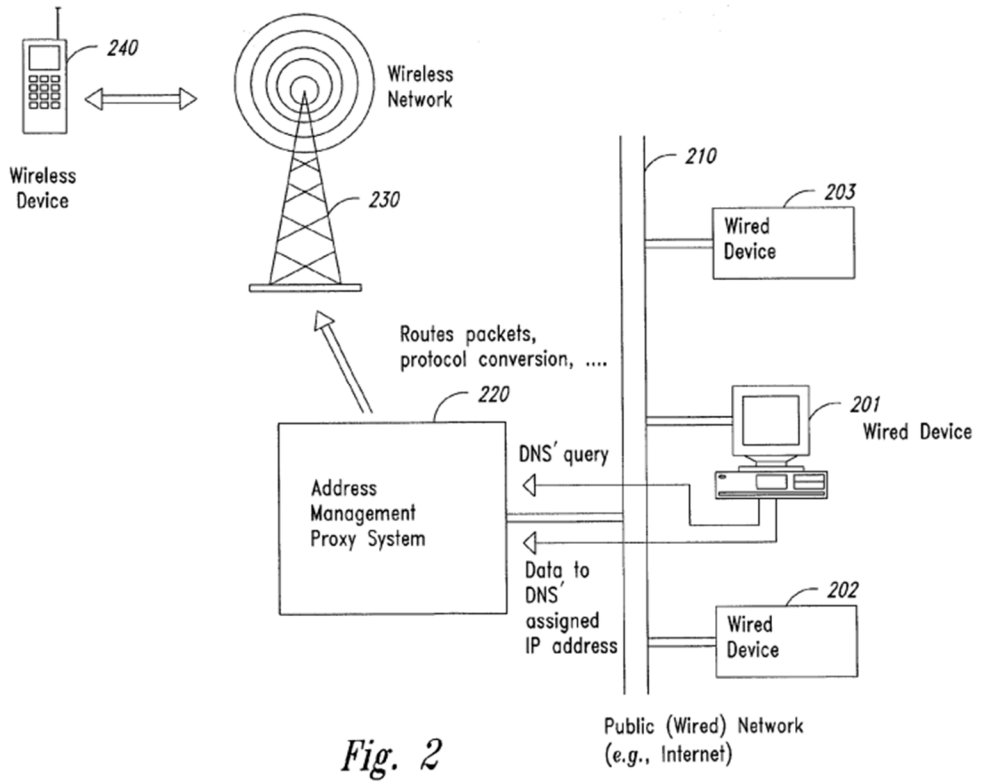


Fig. 2

Mehta at FIG. 2.

> "Methods and systems consistent with the present invention establish a virtual network on top of current IP network naming schemes. The virtual network uses a separate layer to create a modification to the IP packet format that is used to separate network behavior from addressing. As a result of the modification to the packet format, any type of delivery method may be assigned to any address or group of addresses. The virtual network also maintains secure communications between nodes, while providing the flexibility of assigning delivery methods independent of the delivery addresses."

Caronni I at Abstract.

	"The present invention relates generally to data processing systems and, more particularly, to a private network using a public-network infrastructure." Caronni I at 1:57–59. "Methods and systems consistent with the present invention overcome the shortcomings of existing networks by establishing a "Supernet," which is a private network that uses components from a public-network infrastructure. A Supernet allows an organization to utilize a public-network infrastructure for its enterprise network so that the organization no longer has to maintain a private network infrastructure; instead, the organization may have the infrastructure maintained for them by one or more service providers or other organizations that specialize in such connectivity matters. As such, the burden of maintaining an enterprise network is greatly reduced. Moreover, a Supernet is not geographically restrictive, so a user may plug their device into the Internet from virtually any portal in the world and still be able to use the resources of their private network in a secure and robust manner." Caronni I at 4:36-52. "Supernets also provide heterogeneous addressing functionality. The Supernet uses a separate layer that isolates address names of nodes from addressing schemes and delivery schemes. The Supernet contains a modification to the IP packet format that can be used to separate network behavior from addressing. As a result of the modification, any delivery scheme may be assigned to any address, or group of addresses." Caronni I at 4:53-59. "SNlogin 522 is a script used for logging into a Supernet. Successfully executing this script results in a Unix shell from which programs (e.g., node A 522) can be started to run within the Supernet context, such that address translation and security encapsulation is performed transparently for them and all they can typically access is other nodes on the Supernet. Alternatively, a parameter may be passed into SNlogin 522 that indicates a particular process to be automatically run in a Supernet context. Once a program is running in a Supernet

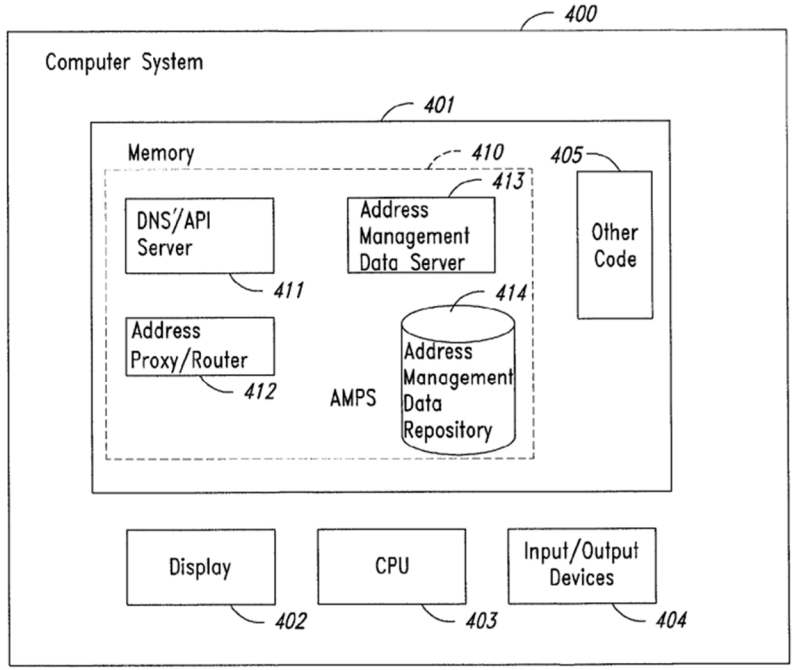
	context, all programs spawned by that program also run in the Supernet context, unless explicitly stated otherwise. SNlogout 524 is a script used for logging out of a Supernet. Although both SNlogin 522 and SNlogout 524 are described as being scripts, one skilled in the art will appreciate that their processing may be performed by another form of software. VARPD 526 performs address translation between node IDs and real addresses. KMC 528 is the key management component for each node that receives updates whenever the key for a channel ("the channel key") changes. There is one KMC per node per channel. KMD 530 receives requests from SNSL 542 of the TCP/IP protocol stack 534 when a packet is received and accesses the appropriate KMC for the destination node to retrieve the appropriate key to decrypt the packet. Node A 532 is a Supernet node running in a Supernet context." Caronni I at 8:31-55. "TCP/IP protocol stack 534 contains a standard TCP/UDP layer 538, two standard IP layers (an inner IP layer 540 and an outer IP layer 544), and a Supernet security layer (SNSL) 542, acting as the conduit for all Supernet communications. To conserve memory, both inner IP layer 540 and outer IP layer 544 may share the same instance of the code of an IP layer. SNSL 542 performs security functionality as well as address translation. It also caches the most recently used channel keys for ten seconds. Thus, when a channel key is needed, SNSL 542 checks its cache first, and if it is not found, it requests KMD 530 to contact the appropriate KMC to retrieve the appropriate channel key." Caronni I at 8:54-67. "FIG. 5 depicts administrative machine 306 and device 302 in greater detail, although the other devices 304 and 308–312 may contain similar components. Device 302 and administrative machine 306 communicate via Internet 314. Each device contains similar components, including a memory 502, 504; secondary storage 506, 508; a central processing unit (CPU) 510, 512; an input device 514, 516; and a video display 518, 520. One skilled in the art will appreciate that these devices may contain additional or different components. Memory 504 of administrative machine 306 includes the SASD process 540, VARPD 548, and KMS 550 all running in user mode. That is, CPU 512 is capable of running in at least two modes: user mode and kernel mode. When CPU 512 executes programs running in user

	mode, it prevents them from directly manipulating the hardware components, such as video display 518. On the other hand, when CPU 512 executes programs running in kernel mode, it allows them to manipulate the hardware components. Memory 504 also contains a VARPDB 551 and a TCP/IP protocol stack 552 that are executed by CPU 512 running in kernel mode. TCP/IP protocol stack 552 contains a TCP/UDP layer 554 and an IP layer 556, both of which are standard layers well known to those of ordinary skill in the art. Secondary storage 508 contains a configuration file 558 that stores various configuration-related information (described below) for use by SASD 540." Caronni I at 6:44-7:2. *See also* Caronni I at Claim 1, FIGS. 3, 5
[30.2] a memory and a processor to implement a register module configured to register devices in a virtual network, the register module further configured to:	VMware Systems discloses "a memory and a processor to implement a register module configured to register devices in a virtual network." *See, e.g.*, VMware systems dynamically allocate CPU, memory, network bandwidth, and disk I/O, using processing resources to manage registration and network functions. "VMware GSX Server guarantees server resources for CPU, memory, network bandwidth, and disk I/O at optimum performance levels." (VMware GSX Server 3.1 Virtual Machine Guide) "The VMware Virtual Infrastructure SDK provides comprehensive interfaces to: Create, delete, copy and clone virtual machines." (VMware Delivers Open Interface to Virtual Infrastructure) "The RegisterVm method registers a virtual machine on a server where vmName is a string specifying the virtual machine's configuration file name." (VMware Scripting API User's Manual) To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This

	reference further renders the limitation obvious in light of other references, such as Mehta, Caronni I, Caronni II, and/or Hipp. Mehta, Caronni I, Caronni II, and Hipp concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, registration of devices in a virtual network was a well-known and commonly-used technique. Modifying this reference with Mehta's, Caronni I's, Caronni II's, and/or Hipp's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with those of Mehta, Caronni I, Caronni II, or Hipp for example, to incorporate this limitation. *See, e.g.*, "Methods and systems for providing two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP, are provided. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporarily use by a requesting device on a public network to communicate with a wireless device on a wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more data repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the public network to send data to the wireless device." Mehta at Abstract. "To insure a greater degree of security, according to one embodiment, the AMPS maintains a Time to Live (TTL) parameter with each address mapping. In this way, once the TTL value

	indicates that the mapping has expired, the AMPS can destroy the mapping, and also any connection. Further, in some embodiments, the AMPS also puts a timestamp in its own mapping tables. After some timeout period based upon the timestamp, the AMPS can destroy the mapping, thereby forcing a new mapping to be initiated on a periodic basis." Mehta at [0015]. "In existing systems, data communication (communication on a data channel) between a wireless device that is connected to a private wireless network and a wired device connected to a public wired network (e.g., the Internet) can only be initiated by the wireless device. Some carriers have assigned fixed public IP addresses to wireless devices; however, the wireless devices need to then have client programs (e.g., a UDP stack) capability of receiving and handling the incoming communication packets. Moreover, these wireless devices are then part of a public wireless network and not a private wireless network. Since public IP addresses are becoming a scarcer commodity and currently expensive to service a network of multi-million devices, carriers cannot in a practical sense count on having a fixed public IP address for each device on its network. (Although movement to IPv6 will allow more addresses, the current IPv4 definitions are limited and the potential number of wireless users subscribing to larger carriers is very high. In some regions of the world, the current public address scheme is even more limited.) Further, such addressing capability would expose each device to further security risks, because each such device is part of a publicly accessible network and it becomes more difficult to implement and enforce security measures. Thus, assigning private IP addresses to devices is preferred in existing wireless networks over assigning fixed public IP addresses to devices. When wireless networks are private, the locations (addresses) of the wireless devices are intentionally hidden from public view by a carrier system's (or, as referred to in some countries, operator's) infrastructure." Mehta at [0025]. "Example embodiments described herein provide applications, tools, data structures and other support to implement private to public address mappings over one or more wired and wireless networks to be used for bi-directional communication. One skilled in the art will recognize that other embodiments of the methods and systems of the present invention may be used for

	many other purposes, including to push information and/or data or code from a public network such as the Internet to a wireless device. In addition, although this description primarily refers to "data" as being sent via the networks, one skilled in the art will recognize that all types of data can be communicated using the techniques described herein including, but not limited to, text, graphics, audio, and video." Mehta at [0034]. "FIG. 4 is an example block diagram of a general purpose computer system for practicing embodiments of the Address Management Proxy System. The general purpose computer system 400 may comprise one or more server and/or client computing systems and may span distributed locations. In addition, each block may represent one or more such blocks as appropriate to a specific embodiment or may be combined with other blocks. The various blocks of the Address Management Proxy System 410 may physically reside on one or more machines, which use standard interprocess communication mechanisms to communicate with each other. In the embodiment shown, computer system 400 comprises a computer memory ("memory") 01, a display 402, a Central Processing Unit ("CPU") 403, and Input/Output devices 404. The Address Management Proxy System 410 is shown residing in memory 401. The components of the Address Management Proxy System 410 preferably execute on CPU 403 and manage the address mapping of wireless devices on a wireless network, as described in previous figures, to allow other wired systems to communicate with the wireless devices. Other downloaded code 405 and potentially other data repositories also reside in the memory 410, and preferably execute on one or more CPU's 403. In a typical embodiment, the AMPS 410 includes one or more DNS'/API servers 411, one or more Address Proxy/Routers 412, an Address Management Data Server 413, and Address Management Data Repositories 414. As described earlier, the AMPS may include other data repositories and components, such as a load balancer, depending upon the particular implementation." Mehta at [0036]. "In one embodiment, a timestamp of the mapping between a public network address and a private address is also noted in table 630. After a defined time out, based upon this timestamp, the Address Management Data Server sends a request to the Address

	Proxy/Router associated with the mapping to unmap the public-to-private address mapping and updates the mapping table 630, thereby returning the associated public address to the pool of unused public network addresses." Mehta at [0056].  *Fig. 4* Mehta at FIG. 4. "To configure a Supernet, a system administrator creates a configuration file 558 that is used by SASD 540 when starting or reconfiguring a Supernet. This file may specify: (1) the

	Supernet name, (2) all of the channels in the Supernet, (3) the nodes that communicate over each channel, (4) the address of the KMS for each channel, (5) the address of the VARPD that acts as the server for the Supernet, (6) the user IDs of the users who are authorized to create Supernet nodes, (7) the authentication mechanism to use for each user of each channel, and (8) the encryption algorithm to use for each channel. Although the configuration information is described as being stored in a configuration file, one skilled in the art will appreciate that this information may be retrieved from other sources, such as databases or interactive configurations." Caronni I at 8:1-15. "SASD 540 represents a Supernet: there is one instance of an SASD per Supernet, and it both authenticates nodes and authorizes nodes to join the Supernet. VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as a node IDs, to the network addresses recognized by the public-network infrastructure, known as the real addresses. The "node ID" may include the following: a Supernet ID (e.g., 0x123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1). Although the virtual address is described in an IP address scheme, one skilled in the art will appreciate that the virtual address may be any other type addressing scheme, such as an e-mail address, IPX, or IPv6. Since the node ID includes a Supernet ID, a node will have more than one node ID when it communicates over more than one channel. The "real address" is an IP address (e.g., 10.0.0.2) that is globally unique and meaningful to the public-network infrastructure. In a Supernet, one VARPD runs on each machine, and it may play two roles. First, a VARPD may act as a server by storing all address mappings for a particular Supernet into its associated VARPDB. Second, regardless of its role as a server or not, each VARPD assists in address translation for the nodes on its machine. In this role, the VARPD stores into its associated VARPDB the address mappings for its nodes, and if it needs a mapping that it does not have, it will contact the VARPD that acts as the server for the given Supernet to obtain it. The VARPDB may also decide which virtual address to use in the translation. That is, the VARPDB may associate a virtual address with multiple real addresses or vice versa." Caronni I at 7:3–33.

	"FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:67–10:18. "After configuring SNSL, SNlogin invokes an operating system call, SETVIN, to cause the SNlogin script to run in a Supernet context (step 720). In Unix, each process has a data structure known as the "proc structure" that contains the process ID as well as a pointer to a virtual memory description of this process. In accordance with methods and systems consistent with the present invention, the Supernet IDs indicating the channels over which the process communicates as well as its virtual address for this process are added to this structure. By associating this information with the process, the SNSL layer can enforce that this process runs in a Supernet context. Although methods and systems consistent with the present invention are described as operating in a Unix environment, one skilled in the art will appreciate that such methods and systems can operate in other environments. After the SNlogin script runs in the Supernet context, the SNlogin script spawns a Unix program, such as a Unix shell or a service daemon (step 722). In this step, the SNlogin script spawns a Unix shell from which programs can be run by the user. All of these programs will thus run in the Supernet context until the user runs the SNlogout script." Caronni I at 10:63–11:17.

	"The packet and Supernet ID are then transmitted to the SNSL layer using the modified socket structure (step 806). The SNSL layer then accesses the VARPDB to obtain the address mapping between virtual source node address 642 and the source real address 614 as well as the virtual destination node address 644 and the destination real address 616 (step 808). If they are not contained in the VARPDB because this is the first time a packet has been sent from this node or sent to this destination, the VARPDB accesses the local VARPD to obtain the mapping. When contacted, the VARPD on the local machine contacts the VARPD that acts as the server for the Supernet to obtain the appropriate address mapping. Since the VARPDB maintains all real IP addresses, a remote node may securely communicate with another remote node without reverfication." Caronni I at 11:47–61. *See also* Caronni I at FIGS. 7A and 7B. "In an embodiment of the present invention a virtual network is supported by a physical network and includes a virtual address resolution facility. The virtual address resolution facility is used to resolve a virtual IP address into a real IP address. A registration request is received at the virtual address resolution facility which includes a real IP address, a port address, a transport protocol designation and an Application layer protocol designation associated with a virtual address destination. The associations are stored using the virtual address resolution facility. A resolution request is received referencing a virtual address destination and the resolution request is resolved using the virtual address resolution facility and the stored associations." Caronni II at 3:11–23. "FIG. 2 depicts a block diagram of the VARP lookup table 26 used by the virtual address resolution facility 24. The VARP lookup table 26 may be stored at any location accessible over the network. The VARP lookup table 26 includes registered virtual destination addresses located in a column 78 and corresponding associations located in a column 79. Entries 80, 82,

	84, 86 and 88 include the registered virtual destination address and information associated with the registered virtual destination address. Each entry in the VARP lookup table 26 will list the registering virtual IP address 90 and associated information including a real IP address 91, a transport protocol designation 92, a port number identification 93 and an Application layer protocol designation 94. The associations are provided at the time the virtual destination address is registered in the virtual network. For example, the entry 80 for the virtual IP address 10.0.0.12 (90) includes the association of a real IP address 152.70.0.1 (91), an associated Transport layer protocol designation, UDP 92, an associated port number, 6789 (93), and an Application layer protocol designation of 'none' 94." Caronni II at 5:10–29. "FIG. 4 depicts the sequence of steps followed by the illustrative embodiment of the present invention to process the received encapsulated message. The sequence begins when a process associated with a virtual IP address registers with the virtual address resolution facility 24 (step 140). The registration is stored in a VARP lookup table accessible over the network. Subsequently, a message is received at the electronic device which bears a MAC/network interface address of the network interface of the electronic device with the designated real IP address listed in the virtual address registration (Step 142). The MAC address header 117 is stripped off at the Link layer and the message and appended headers are passed up to the Network layer 114(Step 144). The Network layer 114 strips off the IP header 115 and identifies the Transport Protocol header 113 underneath. The Network layer passes the message and remaining appended header to the Transport layer 112 (Step 146). The Transport layer 112 strips off the Transport header and then passes the message to the Application layer via the port address of the interface identified in the VARP registration. (Step 148). The receiving process determines whether it is the virtual destination address the message is addressed to (Step 149) and acts accordingly. If the receiving process that is examining the message is the destination address, the process examines the data (Step 150). Alternatively, if the message is not intended for the process executing on the edge device, the process re-routes the message and sends it back down the protocol model stack, on to the local network operating behind the edge device, and on to its intended destination by performing traditional

	VARP and ARP lookups. Alternatively forwarding the message may be accomplished by reinserting the packet into the IP layer of the networking stack." Caronni II at 7:33–67. *See also* Caronni II at Claim 1, FIGS. 2-4. "FIG. 3 is a data flow diagram illustrating the registration of virtual network environment parameters. The VNE framework 200 is a software module that processes transactions between the applications and the operating system. The VNE parameters are registered with the VNE framework 200 at the time the application is started. The VNE parameters include the application IP address, the virtual network subnet, and the global virtual address subnet. At step 250, the registration harness 220 supplies the application IP address, virtual subnet, and the global virtual address subnet for a $process_x$ to the VNE framework 200. The VNE framework 200 then records the IP address, virtual subnet, and the global virtual address subnet for the $process_x$ at step 252. The $process_x$ can then spawn additional processes (or create additional objects) at step 254. the new $process_y$ inherits the IP address, virtual subnet, and global virtual address subnet from $process_x$. At step 256, the registration harness 220 launches the application related to $process_y$." Hipp at 3:52-4:2. "Virtualization of network identity is achieved by assigning a unique virtual IP address and virtual hostname to a group of processes that make up the application instance which the instance keeps throughout its execution. This virtual network identity stays with the application instance regardless of which node the application is running on. The framework, in essence, provides a mechanism to create this virtual network identity (VNI) around the application using the virtual network parameters assigned to it. In one embodiment, the virtual network parameters include an IP address and hostname. The framework 200 ensures that the application's instance uses the virtual network parameters, transparently, so that it can be moved across machines, without modifications to the application." Hipp at 6:1-15.

	See also Hipp at 6:16-60, FIG. 3.
[30.3] receive a registration request from an agent associated with a device;	To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Mehta, Caronni I, Caronni II, and/or Hipp. Mehta, Caronni I, Caronni II, and Hipp concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, registration of devices in a virtual network was a well-known and commonly-used technique. Modifying this reference with Mehta's, Caronni I's, Caronni II's, and/or Hipp's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with those of Mehta, Caronni I, Caronni II, or Hipp for example, to incorporate this limitation. *See, e.g.*, "Methods and systems for providing two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP, are provided. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporarily use by a requesting device on a public network to communicate with a wireless device on a wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more data repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private

	address of the wireless device. The public address is then usable by the device on the public network to send data to the wireless device." Mehta at Abstract. "To insure a greater degree of security, according to one embodiment, the AMPS maintains a Time to Live (TTL) parameter with each address mapping. In this way, once the TTL value indicates that the mapping has expired, the AMPS can destroy the mapping, and also any connection. Further, in some embodiments, the AMPS also puts a timestamp in its own mapping tables. After some timeout period based upon the timestamp, the AMPS can destroy the mapping, thereby forcing a new mapping to be initiated on a periodic basis." Mehta at [0015]. "In existing systems, data communication (communication on a data channel) between a wireless device that is connected to a private wireless network and a wired device connected to a public wired network (e.g., the Internet) can only be initiated by the wireless device. Some carriers have assigned fixed public IP addresses to wireless devices; however, the wireless devices need to then have client programs (e.g., a UDP stack) capability of receiving and handling the incoming communication packets. Moreover, these wireless devices are then part of a public wireless network and not a private wireless network. Since public IP addresses are becoming a scarcer commodity and currently expensive to service a network of multi-million devices, carriers cannot in a practical sense count on having a fixed public IP address for each device on its network. (Although movement to IPv6 will allow more addresses, the current IPv4 definitions are limited and the potential number of wireless users subscribing to larger carriers is very high. In some regions of the world, the current public address scheme is even more limited.) Further, such addressing capability would expose each device to further security risks, because each such device is part of a publicly accessible network and it becomes more difficult to implement and enforce security measures. Thus, assigning private IP addresses to devices is preferred in existing wireless networks over assigning fixed public IP addresses to devices. When wireless networks are private, the locations (addresses) of the wireless devices are intentionally hidden from public view by a carrier system's (or, as referred to in some countries, operator's) infrastructure."

	Mehta at [0025]. "Example embodiments described herein provide applications, tools, data structures and other support to implement private to public address mappings over one or more wired and wireless networks to be used for bi-directional communication. One skilled in the art will recognize that other embodiments of the methods and systems of the present invention may be used for many other purposes, including to push information and/or data or code from a public network such as the Internet to a wireless device. In addition, although this description primarily refers to "data" as being sent via the networks, one skilled in the art will recognize that all types of data can be communicated using the techniques described herein including, but not limited to, text, graphics, audio, and video." Mehta at [0034]. "In one embodiment, a timestamp of the mapping between a public network address and a private address is also noted in table 630. After a defined time out, based upon this timestamp, the Address Management Data Server sends a request to the Address Proxy/Router associated with the mapping to unmap the public-to-private address mapping and updates the mapping table 630, thereby returning the associated public address to the pool of unused public network addresses." Mehta at [0056]. "FIG. 3 depicts a data processing system 300 suitable for use with methods and systems consistent with the present invention. Data processing system 300 comprises a number of devices, such as computers 302–312, connected to a public network, such as the Internet 314. A Supernet's infrastructure uses components from the Internet because devices 302, 304, and 312 contain nodes that together form a Supernet and that communicate by using the infrastructure of the Internet. These nodes 316, 318, 320, and 322 are communicative entities (e.g., processes) running within a particular device and are able to communicate among themselves as well as access the resources of the Supernet in a secure manner. When communicating among themselves, the nodes 316, 318, 320, and 322 serve as end points for the communications, and no other processes or devices that are not part of the Supernet are

	able to communicate with the Supernet's nodes or utilize the Supernet's resources. The Supernet also includes an administrative node 306 to administer to the needs of the Supernet." Caronni I at 4:66–5:13. "FIG. 5 depicts administrative machine 306 and device 302 in greater detail, although the other devices 304 and 308–312 may contain similar components. Device 302 and administrative machine 306 communicate via Internet 314. Each device contains similar components, including a memory 502, 504; secondary storage 506, 508; a central processing unit (CPU) 510, 512; an input device 514, 516; and a video display 518, 520. One skilled in the art will appreciate that these devices may contain additional or different components. Memory 504 of administrative machine 306 includes the SASD process 540, VARPD 548, and KMS 550 all running in user mode. That is, CPU 512 is capable of running in at least two modes: user mode and kernel mode. When CPU 512 executes programs running in user mode, it prevents them from directly manipulating the hardware components, such as video display 518. On the other hand, when CPU 512 executes programs running in kernel mode, it allows them to manipulate the hardware components. Memory 504 also contains a VARPDB 551 and a TCP/IP protocol stack 552 that are executed by CPU 512 running in kernel mode. TCP/IP protocol stack 552 contains a TCP/UDP layer 554 and an IP layer 556, both of which are standard layers well known to those of ordinary skill in the art. Secondary storage 508 contains a configuration file 558 that stores various configuration-related information (described below) for use by SASD 540." Caronni I at 6:43–7:2. "Memory 502 of device 302 contains SNlogin script 522, SNlogout script 524, VARPD 526, KMC 528, KMD 530, and node A 532, all running in user mode. Memory 502 also includes TCP/IP protocol stack 534 and VARPDB 536 running in kernel mode." Caronni I at 8:26-30. "FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet

	name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:66-10:18. *See also* Caronni I at FIGS. 3, 5, 7A, 7B. "In an embodiment of the present invention a virtual network is supported by a physical network and includes a virtual address resolution facility. The virtual address resolution facility is used to resolve a virtual IP address into a real IP address. A registration request is received at the virtual address resolution facility which includes a real IP address, a port address, a transport protocol designation and an Application layer protocol designation associated with a virtual address destination. The associations are stored using the virtual address resolution facility. A resolution request is received referencing a virtual address destination and the resolution request is resolved using the virtual address resolution facility and the stored associations." Caronni II at 3:11–23. "FIG. 2 depicts a block diagram of the VARP lookup table 26 used by the virtual address resolution facility 24. The VARP lookup table 26 may be stored at any location accessible over the network. The VARP lookup table 26 includes registered virtual destination addresses located in a column 78 and corresponding associations located in a column 79. Entries 80, 82,

	84, 86 and 88 include the registered virtual destination address and information associated with the registered virtual destination address. Each entry in the VARP lookup table 26 will list the registering virtual IP address 90 and associated information including a real IP address 91, a transport protocol designation 92, a port number identification 93 and an Application layer protocol designation 94. The associations are provided at the time the virtual destination address is registered in the virtual network. For example, the entry 80 for the virtual IP address 10.0.0.12 (90) includes the association of a real IP address 152.70.0.1 (91), an associated Transport layer protocol designation, UDP 92, an associated port number, 6789 (93), and an Application layer protocol designation of 'none' 94." Caronni II at 5:10–29. "FIG. 4 depicts the sequence of steps followed by the illustrative embodiment of the present invention to process the received encapsulated message. The sequence begins when a process associated with a virtual IP address registers with the virtual address resolution facility 24 (step 140). The registration is stored in a VARP lookup table accessible over the network. Subsequently, a message is received at the electronic device which bears a MAC/network interface address of the network interface of the electronic device with the designated real IP address listed in the virtual address registration (Step 142). The MAC address header 117 is stripped off at the Link layer and the message and appended headers are passed up to the Network layer 114(Step 144). The Network layer 114 strips off the IP header 115 and identifies the Transport Protocol header 113 underneath. The Network layer passes the message and remaining appended header to the Transport layer 112 (Step 146). The Transport layer 112 strips off the Transport header and then passes the message to the Application layer via the port address of the interface identified in the VARP registration. (Step 148). The receiving process determines whether it is the virtual destination address the message is addressed to (Step 149) and acts accordingly. If the receiving process that is examining the message is the destination address, the process examines the data (Step 150). Alternatively, if the message is not intended for the process executing on the edge device, the process re-routes the message and sends it back down the protocol model stack, on to the local network operating behind the edge device, and on to its intended destination by performing traditional

	VARP and ARP lookups. Alternatively forwarding the message may be accomplished by reinserting the packet into the IP layer of the networking stack." Caronni II at 7:33–67. *See also* Caronni II at Claim 1, FIGS. 2-4. "FIG. 3 is a data flow diagram illustrating the registration of virtual network environment parameters. The VNE framework 200 is a software module that processes transactions between the applications and the operating system. The VNE parameters are registered with the VNE framework 200 at the time the application is started. The VNE parameters include the application IP address, the virtual network subnet, and the global virtual address subnet. At step 250, the registration harness 220 supplies the application IP address, virtual subnet, and the global virtual address subnet for a $process_x$ to the VNE framework 200. The VNE framework 200 then records the IP address, virtual subnet, and the global virtual address subnet for the $process_x$ at step 252. The $process_x$ can then spawn additional processes (or create additional objects) at step 254. the new $process_y$ inherits the IP address, virtual subnet, and global virtual address subnet from $process_x$. At step 256, the registration harness 220 launches the application related to $process_y$." Hipp at 3:52-4:2. "Virtualization of network identity is achieved by assigning a unique virtual IP address and virtual hostname to a group of processes that make up the application instance which the instance keeps throughout its execution. This virtual network identity stays with the application instance regardless of which node the application is running on. The framework, in essence, provides a mechanism to create this virtual network identity (VNI) around the application using the virtual network parameters assigned to it. In one embodiment, the virtual network parameters include an IP address and hostname. The framework 200 ensures that the application's instance uses the virtual network parameters, transparently, so that it can be moved across machines, without modifications to the application." Hipp at 6:1-15.

	See also Hipp at 6:16-60, FIG. 3.
[30.4] distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address; and	VMware Systems discloses "distribute a virtual network address to the device when the device is registered in the virtual network, the device being identified to other devices in the virtual network by the virtual network address." *See, e.g.*, VMware systems provide dynamic resource allocation, enabling dynamic management of server resources and provisioning through a management console, which can involve network address assignment. "System resources are dynamically allocated to any operating system based on need, providing mainframe-class capacity utilization and control of server resources." (VMware ESX Server Specifications) "Provision Servers Rapidly: Pre-configured virtual machine servers can be built once quickly and deployed anywhere immediately." (VMware GSX Server Specifications) "By implementing a virtual infrastructure, IT organizations can provision new services and change the amount of resources dedicated to a software service simply by interacting with a management console." (VMware Enterprises Continue to Embrace VMware Virtual Infrastructure) To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Mehta, Caronni I, Caronni II and/or Hipp. Mehta, Caronni I, Caronni II and Hipp concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, distributing virtual network addresses was a well-known and commonly-used technique.

	Modifying this reference with Mehta's, Caronni I's, Caronni II's, and/or Hipp's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with those of Mehta, Caronni I, Caronni II, or Hipp for example, to incorporate this limitation. *See, e.g.*, "Methods and systems for providing two-way initiated, bi-directional communication with wireless devices using connection-based or connection-less protocols, such as, for example, TCP/IP and UDP/IP, are provided. Example embodiments provide an Address Management Proxy System ("AMPS"), which enables devices and systems connected to a public internet, such as the Internet, to initiate communication with and to send data to wireless devices connected to a private wireless network, without exposing the non-routable private addresses of these wireless devices. The AMPS allocates a public (routable) network address for temporarily use by a requesting device on a public network to communicate with a wireless device on a wireless network. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained by the AMPS and allocated dynamically to wireless network devices as required. In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more data repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the public network to send data to the wireless device." Mehta at Abstract. "To insure a greater degree of security, according to one embodiment, the AMPS maintains a Time to Live (TTL) parameter with each address mapping. In this way, once the TTL value indicates that the mapping has expired, the AMPS can destroy the mapping, and also any connection. Further, in some embodiments, the AMPS also puts a timestamp in its own mapping tables. After some timeout period based upon the timestamp, the AMPS can destroy the mapping, thereby forcing a new mapping to be initiated on a periodic basis."

	Mehta at [0015]. "In existing systems, data communication (communication on a data channel) between a wireless device that is connected to a private wireless network and a wired device connected to a public wired network (e.g., the Internet) can only be initiated by the wireless device. Some carriers have assigned fixed public IP addresses to wireless devices; however, the wireless devices need to then have client programs (e.g., a UDP stack) capability of receiving and handling the incoming communication packets. Moreover, these wireless devices are then part of a public wireless network and not a private wireless network. Since public IP addresses are becoming a scarcer commodity and currently expensive to service a network of multi-million devices, carriers cannot in a practical sense count on having a fixed public IP address for each device on its network. (Although movement to IPv6 will allow more addresses, the current IPv4 definitions are limited and the potential number of wireless users subscribing to larger carriers is very high. In some regions of the world, the current public address scheme is even more limited.) Further, such addressing capability would expose each device to further security risks, because each such device is part of a publicly accessible network and it becomes more difficult to implement and enforce security measures. Thus, assigning private IP addresses to devices is preferred in existing wireless networks over assigning fixed public IP addresses to devices. When wireless networks are private, the locations (addresses) of the wireless devices are intentionally hidden from public view by a carrier system's (or, as referred to in some countries, operator's) infrastructure." Mehta at [0025]. "FIG. 3 is an example block diagram of components of an example Address Management Proxy System. In one embodiment, the Address Management Proxy System (AMPS) comprises one or more modified DNS'/ API servers 302, one or more Address Proxy/Routers 305, an Address Management Data Server 303 which manages a database or other repositories such as Address Management Data Repository 304, and optionally a load balancer 301. The DNS'/API servers 302 are either individually connected to a public network 310 or are connected to the load balancer 301 which is in turn connected to public network 310. Similarly, each Address Proxy/Router 305 is also connected to the public

	network 310, via the routable (public) address to which data from the external network 310 is sent that is destined for the wireless devices. The DNS'/API servers 302 are either modified implementations of a DNS server to add functionality necessary to communicate with wireless devices, or are servers that implement one or more specialized APIs, as will be described further below. The DNS'/API servers 302 use the Address Management Data Server 303 to assist in mapping a unique identifier (e.g., string name) for a wireless device to a public address on public network 310. The pool of public addresses is also maintained by the Address Management Data Serve 303 and Data Repository 304. The Address Management Data server 303 and Address Management Data Repository 304 may be implemented using existing database technology, for example, ODBC technology or, may be implemented as a structure such as a simple text file. One skilled in the art will recognize that any embodiment for storing a set of tables, data, lists, or mappings can be used. Each Address Proxy/Router 305 also uses the Address Management Data Server 303 or equivalent to create and update a series of routing tables that are used to assign public addresses to wireless devices as needed and to update the various mappings between public addresses and the non-routable (private) addresses of the wireless devices. The tables and mappings that are maintained on behalf of the DNS'/API servers 302 and the Address Proxy/Routers 305 by the Address Management Data Server 303 are described below with reference to FIG. 6." Mehta at [0032]. "Although the techniques of the AMPS are generally applicable to any a wired device communicating with a wireless device, the phrase "public network" (or "wired network") is used generally to imply any type of internetworked environment including a public network or a backbone that is somewhere down the line connected to one or more private or public networks. In addition, although the examples described herein often refer to the Internet, one skilled in the art will recognize that the concepts and inventions described are applicable to other forms and embodiments of internetworking, including, for example ATM type networks. Thus, techniques of the present invention can also be used by one device on a first wireless network to communicate with another wireless device on a second network—each device ends up communicating with the Address Proxy/Router of the other network. This scenario is feasible because each wireless network (or its carrier infrastructure) is connected

	to a proxy/router that is also connected (via a public address) to a public network. In addition, although a public network is sometimes also referred to herein as a wired network, one skilled in the art will recognize that any network that exposes routable (public) addresses may be implied. Thus, a wireless network with unique public (and routable) address can also employ techniques of the present invention to perform bi-directional communication. Also, one skilled in the art will recognize that terms such as wireless device, phone, handheld, etc., are used interchangeably to indicate any type of wireless device that is capable of operating with the AMPS. In addition, terms may have alternate spellings which may or may not be explicitly mentioned, and one skilled in the art will recognize that all such variations of terms are intended to be included." Mehta at [0033]. "Example embodiments described herein provide applications, tools, data structures and other support to implement private to public address mappings over one or more wired and wireless networks to be used for bi-directional communication. One skilled in the art will recognize that other embodiments of the methods and systems of the present invention may be used for many other purposes, including to push information and/or data or code from a public network such as the Internet to a wireless device. In addition, although this description primarily refers to "data" as being sent via the networks, one skilled in the art will recognize that all types of data can be communicated using the techniques described herein including, but not limited to, text, graphics, audio, and video." Mehta at [0034]. "There are several implementation approaches to the components of the Address Management Proxy System, three of which are described herein. One skilled in the art will recognize that various other approaches and combinations are possible. All three approaches allocate a public (routable) network address for temporary use by a wired device to communicate with a wireless device. In one embodiment, a pool of public addresses, for example, public IP addresses, is maintained and allocated dynamically to wireless network devices as required. For example, a typical Class B Internet network address block allows for approximately 65,000 simultaneous connections to wireless devices. Although this number

	may seem large at first blush, when one considers the number of cell phones and handsets, for example, connected to a carrier, this number can be quite limiting." Mehta at [0039]. "The unique ID table 610 maps unique string names of wireless devices to the private wireless network addresses that have been assigned typically by a carrier's infrastructure. In some embodiments, the carrier infrastructure dynamically allocates, using methods similar to a DHCP protocol, a private wireless network address when the wireless device registers itself with the carrier infrastructure upon being powered on. Thus, the unique ID table 610 may be sparsely filled or entries created dynamically and then deleted dynamically as devices register and unregister with the carrier infrastructure system." Mehta at [0053]. "In one embodiment, a timestamp of the mapping between a public network address and a private address is also noted in table 630. After a defined time out, based upon this timestamp, the Address Management Data Server sends a request to the Address Proxy/Router associated with the mapping to unmap the public-to-private address mapping and updates the mapping table 630, thereby returning the associated public address to the pool of unused public network addresses." Mehta at [0056]. "FIG. 7 is an example flow diagram of an example routine provided by a DNS'/API server of the Address Management Proxy System to return a public address that corresponds to a designated unique identifier. In essence, this routine implements a DNS query or DNS-like query capability for the AMPS using a modified GetHostByName interface or a specialized API, for example GetProxyIP, as described with reference to FIG. 5. In summary, the routine dynamically allocates an appropriate Address Proxy/Router machine to associate with the wireless device and returns the public address of that machine (along with a TTL parameter and potentially other parameters). Specifically, in step 701, the routine determines the private (non-routable) address of the wireless device designated by a string parameter passed as input to the routine. For example, the string parameter may use fields such as

	"uniqueID.hostname.domain.tld," which specifies a typical hierarchy of person/service on a machine named "hostname" on a domain such as a company's network on a top level domain such as "org." "com," "edu," etc. One skilled in the art will recognize that many other string parameter designations could be used. One mechanism for implementing this routine is to request information from the Address Management Data Repository. In one embodiment, the data repository stores a table that maps unique IDs to private network addresses (see Table 610 in FIG. 6). Preferably, any mechanism that is used by the AMPS stores this data in a secure manner in order to keep the wireless network addresses private. In step 702, the routine retrieves the public network address that corresponds to the private wireless network address of the designated device if one has already been assigned by the AMPS to that device and is still valid. In one embodiment, the data repository stores this mapping information between private wireless network address and public network address (see for example table 630 in FIG. 6). If a public network address has not already been assigned or is not valid, then the routine causes a new public network address to be allocated and that new public address is associated with the private wireless network address. Appropriate tables in the data repository are then updated. In step 703, the routine determines the Address Proxy/Router machine that is associated with the assigned public network address (for example using table 620 in FIG. 6). In step 704, the routine sends a request to the determined proxy/router machine to update its routing tables to map the determined public network address to the private wireless network address. In step 705, the routine updates information in the data repository to include any other connection related information (e.g., field 634 in Table 630 in FIG. 6) and indicates a Time to Live (TTL) parameter for the public-private address association (e.g., field 633 in Table 630 in FIG. 6). Once all of the tables have been updated in both the proxy/router and in the data repository, the DNS'/API server returns the determined public network address of the associated Address Proxy/Router machine. As described earlier, the public address may be a (hostname, port) pair when a port-based implementation is used." Mehta at [0057].

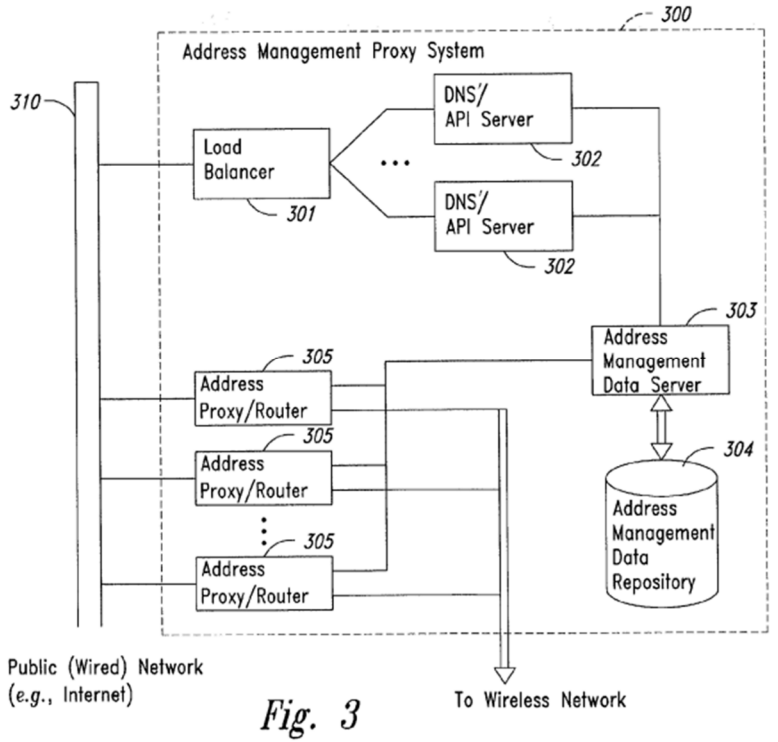


Fig. 3

Mehta at FIG. 3.

"Fourth, the system provides address translation in a transparent manner. Since the Supernet is a private network constructed from the infrastructure of another network, the Supernet has its own internal addressing scheme, separate from the addressing scheme of the underlying public network. Thus, when a packet from a Supernet node is sent to another Supernet node, it travels through the public network. To do so, the Supernet performs address translation from the internal addressing scheme to the public addressing scheme and vice versa. By separating the addressing schemes, the Supernet creates a flexible delivery scheme that is easily changeable by network software or a system administrator. To reduce the complexity of Supernet nodes, system-level components of the Supernet perform this translation on

	behalf of the individual nodes so that it is transparent to the nodes. Another benefit of the Supernet's addressing is that it uses an IP-based internal addressing scheme so that preexisting programs require little modification to run within a Supernet." Caronni I at 6:7–25. "SASD 540 represents a Supernet: there is one instance of an SASD per Supernet, and it both authenticates nodes and authorizes nodes to join the Supernet. VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as a node IDs, to the network addresses recognized by the public-network infrastructure, known as the real addresses. The "node ID" may include the following: a Supernet ID (e.g., 0x123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1). Although the virtual address is described in an IP address scheme, one skilled in the art will appreciate that the virtual address may be any other type addressing scheme, such as an e-mail address, IPX, or IPv6. Since the node ID includes a Supernet ID, a node will have more than one node ID when it communicates over more than one channel. The "real address" is an IP address (e.g., 10.0.0.2) that is globally unique and meaningful to the public-network infrastructure. In a Supernet, one VARPD runs on each machine, and it may play two roles. First, a VARPD may act as a server by storing all address mappings for a particular Supernet into its associated VARPDB. Second, regardless of its role as a server or not, each VARPD assists in address translation for the nodes on its machine. In this role, the VARPD stores into its associated VARPDB the address mappings for its nodes, and if it needs a mapping that it does not have, it will contact the VARPD that acts as the server for the given Supernet to obtain it. The VARPDB may also decide which virtual address to use in the translation. That is, the VARPDB may associate a virtual address with multiple real addresses or vice versa." Caronni I at 7:3–33. "FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this

	information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:66-10:18. "Once inner IP layer 542 receives the packet, a Supernet ID is appended to a socket structure (step 804). The socket structure is modified so as to contain an extra data field for Supernet ID 626 and virtual source address 642. The addition of Supernet ID 626 and virtual address 642 in the socket structure enables the Supernet to communicate with nodes regardless of the delivery scheme used. When the process on node A opens a socket to transmit the packet to inner IP layer 540, the corresponding Supernet ID 626 and virtual source address 642 for that process is included in the socket request. The packet and Supernet ID are then transmitted to the SNSL layer using the modified socket structure (step 806). The SNSL layer then accesses the VARPDB to obtain the address mapping between virtual source node address 642 and the source real address 614 as well as the virtual destination node address 644 and the destination real address 616 (step 808). If they are not contained in the VARPDB because this is the first time a packet has been sent from this node or sent to this destination, the VARPDB accesses the local VARPD to obtain the mapping. When contacted, the VARPD on the local machine contacts the VARPD that acts as the server for the Supernet to obtain the appropriate address mapping. Since the VARPDB maintains all real IP addresses, a remote node may securely communicate with another remote node without reverfication." Caronni I at 11:37–61. *See also* Caronni I at FIGS. 7A and 7B.

	"Unfortunately, when the virtual destination address is located on a physical device on the interior of a network which is running a proxy server, firewall, or other packet filtering mechanism, messages that have been sent to a virtual destination address have difficulty getting all the way to their target. The term "interior of a network" refers to devices which are not able to directly access another network without first going through another device on their own network. For example, most local area networks (LANs) access the Internet through a proxy server. Devices other than the proxy server are said to be on the interior of the LAN. The proxy server is referred to as an "edge device" because it is able to directly contact another network without using an intermediary device. "Packet filtering" refers to the filtering of incoming messages or packets by an edge device or process on an edge device so that not all of the packets are permitted to proceed to their destination, they are "filtered out". If the electronic device that is filtering incoming packets, is under the control of the party executing the process associated with the virtual destination address, the device may be configured to allow the packets through to the end destination. However, in many situations, the edge device is not configurable by anyone without system administration privileges. Similarly, if the edge device is a device performing Network Address Translation (a "NAT box"), the NAT box rewrites all outgoing packets from an end user in the interior of the network to make them look like they came directly from the NAT box, and remembers that any traffic coming back from the particular destination address must be mapped back to the originating internal device. Consequently, the responding devices think they are responding to the sending device when they are actually responding to an edge device. In such a case, the packets intended for the virtual destination address on the interior of the physical network may be dropped and not reach their intended destination." Caronni II at 1:60–2:27. "…sending said formatted message, augmented by said determined MAC address destination, said real IP address destination and said transport protocol header, from a virtual IP address to the real IP address destination indicated in said virtual address destination registration." Caronni II at 9:20–24.

	"…a process associated with the sending virtual IP address is located on an electronic device outside a firewall and said virtual address destination is located inside the firewall." Caronni II at 9:31–32. *See also* Caronni II at 2:31-3:24, 6:32-54, FIGS. 1-5. "The Virtual Network Environment (VNE) of the present invention is defined by a collection of IP addresses. An application running within one VNE can communicate with another application in the same VNE. However, an application in one VNE cannot communicate with an application in another VNE (unless expressly permitted). These and many other attendant advantages of the present invention will be understood upon reading the following detailed description in conjunction with the drawings." Hipp at 2:8-16. "The VNE is specified at application run time. The VNE is transparent to the application and does not require any modifications to the application. The VNE is defined by subnet of addresses contained within the VNE. For example, all applications within the subnet 10.10.2.0 comprise a VNE. The subnet/netmask specifying such a VNE would be 10.10.2.0/255.255.255.0 and would include the addresses 10.10.2.0 through 10.10.2.255. In this example, an application with IP address 10.10.2.2 would be able to communicate with an application at address 10.10.2.60, but not at 10.10.0.1. Although using a subnet/netmask to specify the VNE is described herein for illustrative purposes, it is to be understood that other methods may be used to accomplish the same mechanism (e.g. an access control list)." Hipp at 3:16-30. "Virtualization of network identity is achieved by assigning a unique virtual IP address and virtual hostname to a group of processes that make up the application instance which the instance keeps throughout its execution. This virtual network identity stays with the application instance regardless of which node the application is running on. The framework, in essence, provides a mechanism to create this virtual network identity (VNI) around the

	application using the virtual network parameters assigned to it. In one embodiment, the virtual network parameters include an IP address and hostname. The framework 200 ensures that the application's instance uses the virtual network parameters, transparently, so that it can be moved across machines, without modifications to the application." Hipp at 6:1-15. *See also* Hipp at Figures 1–7.
[30.5] a DNS server for the virtual network, the DNS server configured to receive a DNS request from a first device in the virtual network, and return a network address associated with a network route director, a private network address associated with a second device in the virtual network, and a virtual network address associated with the second device.	To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Mehta, RFC 1383, Caronni I, Caronni II, and/or Hipp. Mehta, RFC 1383, Caronni I, Caronni II, and/or Hipp concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, use of a server such as a DNS server that returns network addresses was a well-known and commonly-used technique. Modifying this reference with Mehta's, RFC 1383's, Caronni I's, Caronni II's, and/or Hipp's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with those of Mehta, RFC 1383, Caronni I, Caronni II, or Hipp for example, to incorporate this limitation. *See, e.g.*, "A network that supports IP can be connected to another TCP/IP-based or UDP/IP-based internet or the Internet, by providing a router which forwards data to another router or host machine based upon an IP address. The router (or routing server/service) typically contains a routing table, which determines to which machine (and optionally to which port) to send a datagram, given a destination IP address. The IP address uniquely identifies a router/host machine, and, in a typical TCP/IP network, can be mapped to a string name that identifies, for example, a particular machine as part of a larger domain. (Although referred to herein often as a TCP/IP-based network, one skilled in the art will recognize that the network may also be UDP-based (connectionless), and may support another session management system.)"

	Mehta at [0008]. "In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more Address Management Data Repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the external public network to send data to the wireless device. The temporary public address is, for example, an address associated with one of the Address Proxy/Routers, which are connected to the external public network and have access to the private addresses on the private wireless network. In some cases, the device on the public network that desires to send data to the wireless device uses a connection-based protocol, such as TCP/IP to send data. In other cases the device uses a connection-less protocol, such as UDP (UDP/IP) to send the data." Mehta at [0012]. "Wireless systems on private networks use techniques similar to Network Address Translation (NAT) technology to send data from a wireless device to the public networking world. In a typical carrier infrastructure that uses a private network, a wireless device "registers" itself with the carrier infrastructure when it powers on (or in other circumstances, when the device attempts to initiate data services). The carrier dynamically assigns the wireless device a private non-routable address by means of DHCP (or DHCP-like) server. The information on the mapping of the transient private IP address to the device public IP address is stored in an internal carrier database and managed by carrier services such as a RADIUS server." Mehta at [0026]. "The Address Management Proxy System achieves two way initiated bi-directional communication by implementing a modified DNS server and serving as a proxy/router for devices on the private wireless network as they interface to the public wired internetworking world. In summary, a pool of public addresses is maintained and dynamically distributed

	among active wireless devices as needed by the AMPS. FIG. 2 is a block diagram of an example Address Management Proxy System used in bi-directional communication with a wireless device. The term "bi-directional" as used herein means that data paths and communication can flow in either direction between two endpoint systems. FIG. 2 shows wired devices 201, 202, and 203 connected to public network 210. One skilled in the art will recognize that these devices could be connected to another private or public network, which is then connected by one or more wired devices to the public network 210, yet still achieve the functionality discussed here. Any such variation provides equivalent functionality and is explicitly contemplated and presumed to be part of the present invention. On the wireless side, the AMPS 220, acting in its capacity as a proxy (and router) for wireless devices, is shown both connected via wire to the public network 210 and via standard carrier infrastructure elements (not shown) to the wireless network 230. It is presumed that the reader has a working knowledge of the elements of a carrier's infrastructure and the basic mechanisms for routing and mechanisms converting packets from a wired network to a wireless network. These may use analog or digital technology and may require protocol conversions in order to send the physical data and transmit it, for example through a satellite, ultimately to the wireless device. Detailed background information on wireless technology and wireless routing mechanisms is described in Stallings, W., Wireless Communications and Networks, Prentice Hall, N.J., 2002, which is herein incorporated by reference in its entirety. In FIG. 2, wireless device 240 is shown connected via various wireless elements (not shown) to the wireless network 230." Mehta at [0030]. "FIG. 6 is an example block diagram of some of the Address Management Proxy System data repository tables used to support routines of the DNS'/API servers and the Address Proxy/Routers. In one embodiment, the Address Management Data Repository comprises three tables: a unique identifier (unique ID) to private address table 610, a public-to-private address table 630, and a public address to proxy/router machine table 620. Although three tables are shown, one skilled in the art will recognize that these tables could contain other data and may be organized differently, including a different number of tables and with different columns or fields. In addition, any technique for storing a table or list of data may be

	used. To support embodiments that map a host address plus a port designator to a wireless device, the tables are correspondingly modified." Mehta at [0052]. "The unique ID table 610 maps unique string names of wireless devices to the private wireless network addresses that have been assigned typically by a carrier's infrastructure. In some embodiments, the carrier infrastructure dynamically allocates, using methods similar to a DHCP protocol, a private wireless network address when the wireless device registers itself with the carrier infrastructure upon being powered on. Thus, the unique ID table 610 may be sparsely filled or entries created dynamically and then deleted dynamically as devices register and unregister with the carrier infrastructure system." Mehta at [0053]. The public-to-private address table 630 comprises several fields/columns including a public network address 631, a private (wireless) network address 602, a flag 632 that specifies whether the public address stored in field 631 is free or is already used, a Time to Live (TTL) parameter 633, and other connection data 634. In one embodiment, the DNS'/API servers of the AMPS query table 630 to determine a public network address that corresponds to a designated private wireless network address or to allocate an unused public network address (as indicated in field 632) and map the determined unused public address to a private network address stored in field 602. Mehta at [0054]. Public address to proxy/router machine table 620 comprises a public network address field 631 and an indication of a functioning proxy/router machine 621. By maintaining such a mapping, the AMPS is able to substitute proxy/router machines for other proxy/router machines to provide a higher degree of robustness. Each proxy/router machine has a preconfigured set of public network addresses, such as are typically configured by network cards inserted into the proxy/router machine. These address are allocated in a standard fashion through prior purchase or obtaining from an address authorizing authority, currently

	the Internet Corporation for Assigned Names and Numbers (ICANN). When a machine is inserted for use into the AMPS, indications of these addresses are stored in field 631. Mehta at [0055]. "In one embodiment, a timestamp of the mapping between a public network address and a private address is also noted in table 630. After a defined time out, based upon this timestamp, the Address Management Data Server sends a request to the Address Proxy/Router associated with the mapping to unmap the public-to-private address mapping and updates the mapping table 630, thereby returning the associated public address to the pool of unused public network addresses." Mehta at [0056]. "FIG. 7 is an example flow diagram of an example routine provided by a DNS'/API server of the Address Management Proxy System to return a public address that corresponds to a designated unique identifier. In essence, this routine implements a DNS query or DNS-like query capability for the AMPS using a modified GetHostByName interface or a specialized API, for example GetProxyIP, as described with reference to FIG. 5. In summary, the routine dynamically allocates an appropriate Address Proxy/Router machine to associate with the wireless device and returns the public address of that machine (along with a TTL parameter and potentially other parameters). Specifically, in step 701, the routine determines the private (non-routable) address of the wireless device designated by a string parameter passed as input to the routine. For example, the string parameter may use fields such as "uniqueID.hostname.domain.tld," which specifies a typical hierarchy of person/service on a machine named "hostname" on a domain such as a company's network on a top level domain such as "org." "com," "edu," etc. One skilled in the art will recognize that many other string parameter designations could be used. One mechanism for implementing this routine is to request information from the Address Management Data Repository. In one embodiment, the data repository stores a table that maps unique IDs to private network addresses (see Table 610 in FIG. 6). Preferably, any mechanism that is used by the AMPS stores this data in a secure manner in order to keep the wireless network addresses private. In step 702, the routine retrieves the public network address that corresponds to the private wireless network

	address of the designated device if one has already been assigned by the AMPS to that device and is still valid. In one embodiment, the data repository stores this mapping information between private wireless network address and public network address (see for example table 630 in FIG. 6). If a public network address has not already been assigned or is not valid, then the routine causes a new public network address to be allocated and that new public address is associated with the private wireless network address. Appropriate tables in the data repository are then updated. In step 703, the routine determines the Address Proxy/Router machine that is associated with the assigned public network address (for example using table 620 in FIG. 6). In step 704, the routine sends a request to the determined proxy/router machine to update its routing tables to map the determined public network address to the private wireless network address. In step 705, the routine updates information in the data repository to include any other connection related information (e.g., field 634 in Table 630 in FIG. 6) and indicates a Time to Live (TTL) parameter for the public-private address association (e.g., field 633 in Table 630 in FIG. 6). Once all of the tables have been updated in both the proxy/router and in the data repository, the DNS'/API server returns the determined public network address of the associated Address Proxy/Router machine. As described earlier, the public address may be a (hostname, port) pair when a port-based implementation is used." Mehta at [0057].

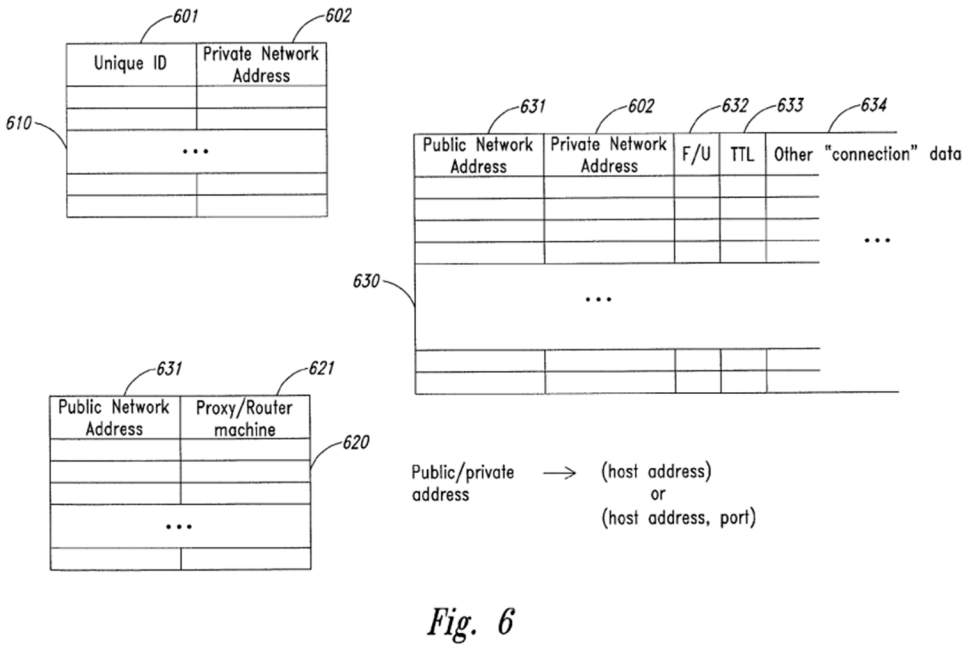


Fig. 6

Mehta at FIG. 6.

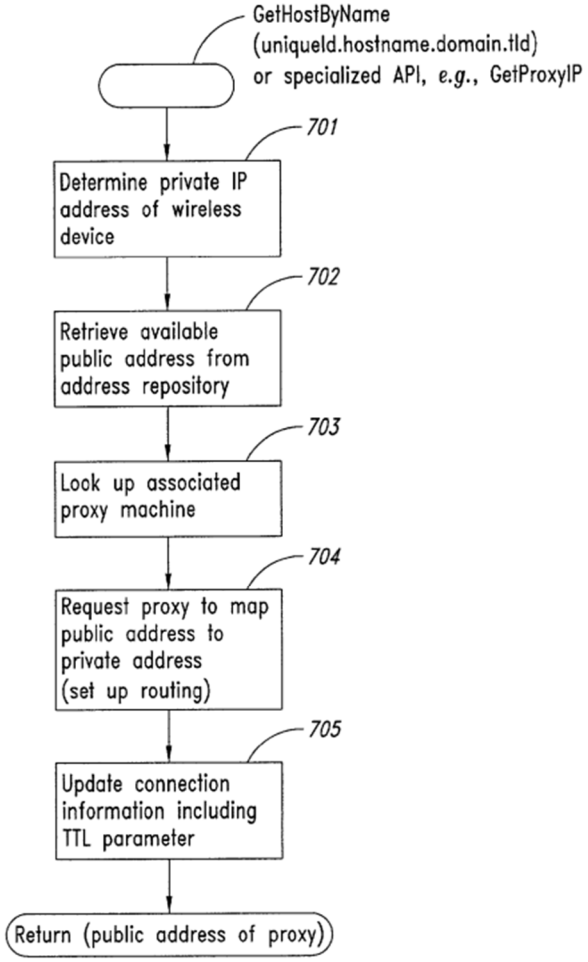

GetHostByName
(uniqueId.hostname.domain.tld)
or specialized API, e.g., GetProxyIP
701
Determine private IP address of wireless device
702
Retrieve available public address from address repository
703
Look up associated proxy machine
704
Request proxy to map public address to private address (set up routing)
705
Update connection information including TTL parameter
Return (public address of proxy)

Fig. 7

	Mehta at FIG. 7. *See also* Mehta at [0004], [0010], [0032], [0036]-[0038], [0048], [0058]-[0059]. "1. Routing, scaling and hierarchies Several recent studies have outlined the risk of "routing explosion" in the current Internet: there are already more than 5000 networks announced in the NSFNET routing tables, more than 7000 in the EBONE routing tables. As these numbers are growing, several problems occur: * The size of the routing tables grows linearly with the number of connected networks; handling this larger tables requires more resources in all "intelligent" routers, in particular in all "transit" and "external" routers that cannot rely on default routes. * The volume of information carried by the route exchange protocols such as BGP grows with the number of networks, using more network resources and making the reaction to routing events slower. * Explicit administrative decisions have to be exercised by all transit networks administrators which want to implement "routing policies" for each and every additional "multi-homed" network. The current "textbook" solution to the routing explosion problem is to use "hierarchical routing" based on hierarchical addresses. This is largely documented in routing protocols such as IDRP, and is one of the rationales for deploying the CIDR [3] addressing structure in the Internet. This textbook solution, while often perfectly adequate, as a number of inconveniences, particularly in the presence of "multihomed stubs", e.g., customer networks that are connected to more than one service providers.

	The current proposal presents a scheme that allows for simple routing. It is complementary with the classic "hierarchical routing" approach, but provides an easy to implement and low cost solution for "multi-homed" domains. The solution is a generalization of the "MX record" scheme currently used for mail routing." RFC 1383 at 1-2. "3.1. Loops and relays In the introduction to DNS-IP routing, we mentioned that the packets would be directed towards the access gateway I1 or I2 by means of "source routing" or "tunnelling". This is not, stricto sensu, necessary. One could imagine that the packet would simply be routed "as if it was directed towards I1 or I2". The next relay would, in turn, also access the DNS to get routing information and forward the packet. Such a strategy would have the advantage of leaving the header untouched and of letting the transit nodes choose the best routing towards the destination, based on their knowledge of the reachability status. It would however have two important disadvantages: - It would oblige all intermediate relays to access the DNS, - It would oblige all these relays to exploit consistently the DNS information. Obliging all intermediate gateways to access the DNS is impractical in the short term: it would mean that we would have to update each and every transit relay before deploying the scheme. It could also have an important performance impact: the "working set" of transit relays is typical much wider than that of stub gateways, and the argument presented previously on the efficiency of caches may not apply. This would perhaps remain impractical even in the long term, as it the volume of DNS traffic could well become excessive. The second argument would apply even if the performance problem had been solved. Suppose that several RX records are registered for a given destination, such as I1 and I2 for

	Dx in our example, and that a "hop by hop routing" strategy is used. There would be a fair risk that some relays would choose to route the packet towards I1 and some others towards I2, resulting in inefficient routing and the possibility of loops. In order to ensure coherency, we propose that all routing decisions be made at the source, or by one of the first relays near the source." RFC 1383 at 4-5. "3.4. Choosing a gateway A simplification to the previous problem would be to allow only one RX record per destination, thus guaranteeing consistent decisions in the network. This would however have a number of draw-backs. A single access point would be a single point of failure, and would be connected to only one transit network thus keeping the "customer locking" effect of hierarchical routing. We propose that the RX records have a structure parallel to that of MX records, i.e., that they carry associated with each gateway address a preference identifier. The source host, when making the routing decision based on RX records, should do the following: - List all possible gateways, - Prune all gateways in the list which are known as "unreachable" from the local site, - If the local host is present in the list with a preference index "x", prune all gateways whose preference index are larger than "x" or equal to "x". - Choose one of the gateway in the list. If the list is empty, consider the destination as unreachable.

Indeed, these evaluations should not be repeated for each and every packet. The routers should maintain a cache of the most frequently used destinations, in order to speed up the processing."

RFC 1383 at 6.

"5.1. DNS record

In a definitive scheme, it would be necessary to define a DNS record type and the corresponding "RX" format. In order to deploy this scheme, we would then have to teach this new format to the domain name service software. While not very difficult to do, this would probably take a couple of month, and will not be used in the early experimentations, which will use the general purpose "TXT" record.

This record is designed for easy general purpose extensions in the DNS, and its content is a text string. The RX record will contain three fields:

- A record identifier composed of the two characters "RX". This is used to disambiguate from other experimental uses of the "TXT" record.

- A cost indicator, encoded on up to 3 numerical digits. The corresponding positive integer value should be less that 256, in order to preserve future evolutions.

- An IP address, encoded as a text string following the "dot" notation.

The three strings will be separated by a single comma. An example of record would thus be:

domain	type	record	value
*.27.32.192.in-addr.arpa	IP	TXT	RX, 10, 10.0.0.7

which means that for all hosts whose IP address starts by the three octets "192.32.27" the IP host "10.0.0.7" can be used as a gateway, and that the preference value is 10."

	RFC 1383 at 11-12. *See also* RFC 1383 at 9, 12-13. "To perform this functionality, D 1 108 utilizes a technique known as tunneling to ensure that the communication between itself and enterprise network 102 is secure in that it cannot be viewed by an interloper. "Tunneling" refers to encapsulating one packet inside another when packets are transferred between two end points (e.g., D 1 108 and VPN software 109 running on firewall 106). The packets may be encrypted at their origin and decrypted at their destination. For example, FIG. 2A depicts a packet 200 with a source Internet protocol (IP) address 202, a destination IP address 204, and data 206. It should be appreciated that packet 200 contains other information not depicted, such as the source and destination port. As shown in FIG. 2B, the tunneling technique forms a new packet 208 out of packet 200 by encrypting it and adding both a new source IP address 210 and a new destination IP address 212. In this manner, the contents of the original packet (i.e., 202, 204, and 206) are not visible to any entity other than the destination. Referring back to FIG. 1, by using tunneling, remote device D 1 108 may communicate and utilize the resources of the enterprise network 102 in a secure manner." Caronni I at 2:26–46. "SASD 540 represents a Supernet: there is one instance of an SASD per Supernet, and it both authenticates nodes and authorizes nodes to join the Supernet. VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as a node IDs, to the network addresses recognized by the public-network infrastructure, known as the real addresses. The "node ID" may include the following: a Supernet ID (e.g., 0x123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1). Although the virtual address is described in an IP address scheme, one skilled in the art will appreciate that the virtual address may be any other type addressing scheme, such as an e-mail address, IPX, or IPv6. Since the node ID includes a Supernet ID, a node will have more than one node ID when it communicates over more than one channel. The "real address" is an IP address (e.g., 10.0.0.2) that is globally unique and

	meaningful to the public-network infrastructure. In a Supernet, one VARPD runs on each machine, and it may play two roles. First, a VARPD may act as a server by storing all address mappings for a particular Supernet into its associated VARPDB. Second, regardless of its role as a server or not, each VARPD assists in address translation for the nodes on its machine. In this role, the VARPD stores into its associated VARPDB the address mappings for its nodes, and if it needs a mapping that it does not have, it will contact the VARPD that acts as the server for the given Supernet to obtain it. The VARPDB may also decide which virtual address to use in the translation. That is, the VARPDB may associate a virtual address with multiple real addresses or vice versa." Caronni I at 7:3–33. "SNSL 542 utilizes VARPDB 536 to perform address translation. VARPDB stores all of the address mappings encountered thus far by SNSL 542. If SNSL 542 requests a mapping that VARPDB 536 does not have, VARPDB communicates with the VARPD 526 on the local machine to obtain the mapping. VARPD 526 will then contact the VARPD that acts as the server for this particular Supernet to obtain it." Caronni I at 9:47–54. "FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file."

	Caronni I at 9:66-10:18. "The packet and Supernet ID are then transmitted to the SNSL layer using the modified socket structure (step 806). The SNSL layer then accesses the VARPDB to obtain the address mapping between virtual source node address 642 and the source real address 614 as well as the virtual destination node address 644 and the destination real address 616 (step 808). If they are not contained in the VARPDB because this is the first time a packet has been sent from this node or sent to this destination, the VARPDB accesses the local VARPD to obtain the mapping. When contacted, the VARPD on the local machine contacts the VARPD that acts as the server for the Supernet to obtain the appropriate address mapping. Since the VARPDB maintains all real IP addresses, a remote node may securely communicate with another remote node without reverfication." Caronni I at 11:47–61. "After obtaining the address mapping, the SNSL layer determines whether it has been configured to communicate over the appropriate channel for this packet (step 806). This configuration occurs when SNlogin runs, and if the SNSL has not been so configured, processing ends. Otherwise, SNSL obtains the channel key to be used for this channel (step 808). The SNSL maintains a local cache of keys and an indication of the channel to which each key is associated. Each channel key is time stamped to expire in ten seconds, although this time is configurable by the administrator. If there is a key located in the cache for this channel, SNSL obtains the key. Otherwise, SNSL accesses KMD which then locates the appropriate channel key from the appropriate KMC. After obtaining the key, the SNSL layer encrypts the packet using the appropriate encryption algorithm and the key previously obtained (step 810). When encrypting the packet, the virtual source node address 642, the virtual destination node address 644, and the data may be encrypted (addressing section 660), but the source and destination real addresses 614, 616 (delivery scheme section 670) are not, so that the real addresses can be used by the public network infrastructure to send the packet to its destination. By encrypting addressing scheme 660, the Supernet enables data to be transmitted securely and at the same time transparent from delivery scheme used." Caronni I at 11:62-12:19.

	"Web client 1102 has a virtual address obtained from computer system 1106, as described in FIGS. 7A and 7B. Each time web client 1102 requests a packet from web server 1104 a, the client requests the virtual address of the web server 1104 a from computer system 1106. If web server 1104 a becomes overloaded (e.g., unable to handle more requests), the overloaded web server spawns new instances of the same web server (web servers 1104 b and 1104 c) and at the same time notifies the computer system 1106. The VARP server of computer 1106 then associates the new instances 104 b and 104 c of the web server with web server's 1104 virtual address. As a result, web client 1102 is not notified of the change and continues to use the same virtual address as previously." Caronni I at 13:13–26. "In one embodiment, a virtual network supported by a physical network includes a virtual address resolution facility. The virtual address resolution facility is used to resolve a virtual IP address into a real IP address. A virtual address is registered with the virtual address resolution facility and the registration includes a real IP address, a port number and a transport protocol designation associated with a virtual address. The virtual address destination is resolved using the virtual address resolution facility and a message is sent from a virtual address in the virtual network to the real IP address indicated in the virtual address destination registration. In one aspect of the embodiment, the registration also includes an Application layer protocol designation." Caronni II at 2:52–64. "In an embodiment of the present invention a virtual network is supported by a physical network and includes a virtual address resolution facility. The virtual address resolution facility is used to resolve a virtual IP address into a real IP address. A registration request is received at the virtual address resolution facility which includes a real IP address, a port address, a transport protocol designation and an Application layer protocol designation associated with a virtual address destination. The associations are stored using the virtual address resolution facility. A resolution request is received referencing a virtual address

	destination and the resolution request is resolved using the virtual address resolution facility and the stored associations." Caronni II at 3:11–23. "The physical network 20 is also interfaced with an electronic device 30 located at the edge of a local area physical network 50 which includes devices addressable using IP addresses in the 129.63.1.0/24 range. The electronic device 30 may be a proxy server, gateway, NAT box, or other device and may perform packet filtering. Alternatively, a firewall 51 may be executed in either a hardware or software form to filter packets sent to the local area physical network 50 from the physical network 20. The firewall 51 may be located outside the electronic device 30 or alternatively may be running in software form on the electronic device 30. Those skilled in the art will recognize that a number of different types of networks may be utilized within the scope of the present invention. The networks may be the Internet, an intranet, wide area network (WAN), a local area network (LAN), a satellite network, a wireless network, or some other type of network capable of supporting a virtual network through the devices thereon." Caronni II at 4:17–33. "FIG. 2 depicts a block diagram of the VARP lookup table 26 used by the virtual address resolution facility 24. The VARP lookup table 26 may be stored at any location accessible over the network. The VARP lookup table 26 includes registered virtual destination addresses located in a column 78 and corresponding associations located in a column 79. Entries 80, 82, 84, 86 and 88 include the registered virtual destination address and information associated with the registered virtual destination address. Each entry in the VARP lookup table 26 will list the registering virtual IP address 90 and associated information including a real IP address 91, a transport protocol designation 92, a port number identification 93 and an Application layer protocol designation 94. The associations are provided at the time the virtual destination address is registered in the virtual network. For example, the entry 80 for the virtual IP address 10.0.0.12 (90) includes the association of a real IP address 152.70.0.1 (91), an associated Transport layer protocol designation, UDP 92, an associated port number, 6789 (93), and an Application layer protocol designation of 'none' 94."

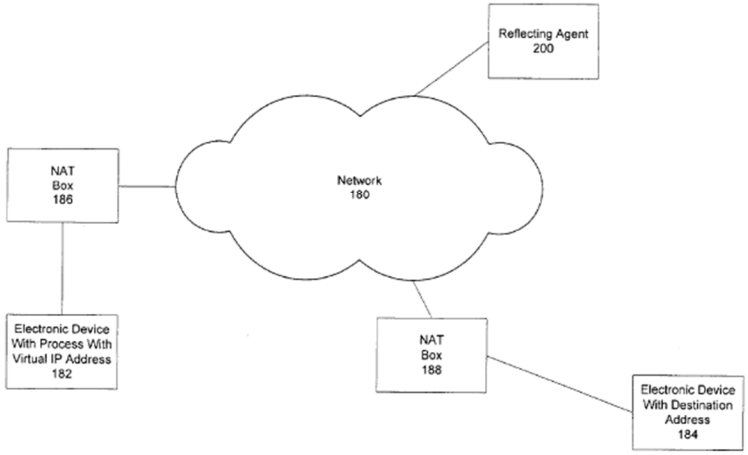
	Caronni II at 5:10–29. "The illustrative embodiment of the present invention attempts to circumvent problems caused by packet filtering and network address translation occurring at the edge of networks through the use of higher layer protocols. FIG. 3 depicts a block diagram of the encapsulation process used to send a message from a virtual IP address to another virtual IP address located at the interior of a network at which packet filtering and/or network address translation is occurring. The virtual address resolution facility 24 is used to determine the associations registered with the destination address. The original message 111 at the Application layer 110 is formatted in a format specified in the registration of the virtual destination address if one is so indicated (e.g. HTTP). The message 111 is passed down to the Transport layer 112. The Transport layer 112 takes the original message 111 and adds a transport protocol header 113 as specified in the virtual address destination registration. The transport protocol header 113 may be a UDP header, TCP header, an X.25 header, XTP header, AppleTalk header, or other similar transport protocol header. Those skilled in the art will realize that the message may include an IPSEC header, inner IP header, UDP or TCP header and inner payload." Caronni II at 6:32-53. "The illustrative embodiment of the present invention may also be utilized to send a message from a virtual address located behind a NAT box to a destination address on the same virtual network that is behind a different NAT box. Ordinarily, where the destination address is directly connected to the Internet (i.e.: in situations where the destination is not behind a NAT box), a connection, such as a TCP connection, may be established between the originator of the connection behind the NAT box and the destination. The originator's message passes the NAT box and the internal mapping of public address and port number to internal address and port number takes place. Return packets from the destination may then be received following the mapping. When the destination address is also behind a NAT box, the originator of the message is unable to directly address the destination (since the address of the destination may not be routable in the public Internet). In such a case, the present invention adds an entry to the VARP table for a third-party reflecting agent located at an address outside the NAT box. Messages that are being sent from the originating virtual address behind a NAT box to a

destination which is behind a different NAT box are sent via the reflecting agent intermediary outside the NAT box and reflected to the destination.

FIG. 5 depicts a block diagram of an environment holding a virtual network in which both the originator and destination of a message are located behind NAT boxes. An electronic device holding an originating process 182 which is part of a virtual network is interfaced with a network 180, such as the Internet, via a NAT box 186. An intended recipient for a message sent from the originating process 180 is an electronic device with a destination address 184 that is also behind a NAT box 188. A reflecting agent 200, the address of which has been added to a VARP table, acts as an intermediary to allow a connection between the originating process 182 and destination address 184."

Caronni II at 8:1–38.

Figure 5



Caronni II at FIG. 5.

"Beginning at step 270, a client application requests to the VNE framework 200 to connect or send to an address of another application, such as a server application having the address

10.10.2.70: port 9001. At step 272, the VNE framework 200 requests the VNE parameters for the process corresponding to the client application from a process state storage structure. The structure that stores process state is a structure that the operating system uses to store private information about the process. Therefore, it may differ depending on the operating system. The parameters added the process state storage structure as part of the virtual network environment are listed below:

```
typedef struct {
        ipaddr_t    app_address;
        ipaddr_t    virtual_subnet;
        ipaddr_t    virtual_mask;
        ipaddr_t    global_subnet;
        ipaddr_t    global_mask;
} vne_param_t;
```

Hipp at 4:44–64.

"Virtualization of network identity is achieved by assigning a unique virtual IP address and virtual hostname to a group of processes that make up the application instance which the instance keeps throughout its execution. This virtual network identity stays with the application instance regardless of which node the application is running on. The framework, in essence, provides a mechanism to create this virtual network identity (VNI) around the application using the virtual network parameters assigned to it. In one embodiment, the virtual network parameters include an IP address and hostname. The framework 200 ensures that the application's instance uses the virtual network parameters, transparently, so that it can be moved across machines, without modifications to the application.

The virtual hostname resolves to the virtual IP address for both the applications registered with the VNI framework as well as those that are not registered. This may require configuration of a name service or OS host configuration files. For example, if an

	application instance used a virtual IP address of 10.10.0.1 and a virtual hostname of host1055, the standard hostname to IP address resolution mechanisms (e.g. DNS or the /etc/hosts file) would have to be preconfigured to resolve a query of host1055 to IP address 10.10.0.1." Hipp at 6:1–26.
[35.0] The virtual network manager of claim 30 further comprising a join module configured to receive a join request from the agent associated with the device to indicate that the device is connected for data communication within the virtual network, the join module further configured to receive a leave request from the agent associated with the device to indicate that the device will be disconnected from data communication within the virtual network.	To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This reference further renders the limitation obvious in light of other references, such as Mehta and/or Caronni I. Mehta and Caronni I concern similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, joining devices with a virtual network was a well-known and commonly-used technique. Modifying this reference with Mehta's and/or Caronni I's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with that of Mehta or Caronni I for example, to incorporate this limitation. *See, e.g.*, "Wireless systems on private networks use techniques similar to Network Address Translation (NAT) technology to send data from a wireless device to the public networking world. In a typical carrier infrastructure that uses a private network, a wireless device "registers" itself with the carrier infrastructure when it powers on (or in other circumstances, when the device attempts to initiate data services). The carrier dynamically assigns the wireless device a private non-routable address by means of DHCP (or DHCP-like) server. The information on the mapping of the transient private IP address to the device public IP address is stored in an internal carrier database and managed by carrier services such as a RADIUS server." Mehta at [0026].

	"FIG. 4 is an example block diagram of a general purpose computer system for practicing embodiments of the Address Management Proxy System. The general purpose computer system 400 may comprise one or more server and/or client computing systems and may span distributed locations. In addition, each block may represent one or more such blocks as appropriate to a specific embodiment or may be combined with other blocks. The various blocks of the Address Management Proxy System 410 may physically reside on one or more machines, which use standard interprocess communication mechanisms to communicate with each other. In the embodiment shown, computer system 400 comprises a computer memory ("memory") 01, a display 402, a Central Processing Unit ("CPU") 403, and Input/Output devices 404. The Address Management Proxy System 410 is shown residing in memory 401. The components of the Address Management Proxy System 410 preferably execute on CPU 403 and manage the address mapping of wireless devices on a wireless network, as described in previous figures, to allow other wired systems to communicate with the wireless devices. Other downloaded code 405 and potentially other data repositories also reside in the memory 410, and preferably execute on one or more CPU's 403. In a typical embodiment, the AMPS 410 includes one or more DNS'/API servers 411, one or more Address Proxy/Routers 412, an Address Management Data Server 413, and Address Management Data Repositories 414. As described earlier, the AMPS may include other data repositories and components, such as a load balancer, depending upon the particular implementation." Mehta at [0036]. "The unique ID table 610 maps unique string names of wireless devices to the private wireless network addresses that have been assigned typically by a carrier's infrastructure. In some embodiments, the carrier infrastructure dynamically allocates, using methods similar to a DHCP protocol, a private wireless network address when the wireless device registers itself with the carrier infrastructure upon being powered on. Thus, the unique ID table 610 may be sparsely filled or entries created dynamically and then deleted dynamically as devices register and unregister with the carrier infrastructure system." Mehta at [0053].

	"FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:67–10:18. "FIG. 10 depicts a flow chart of the steps performed when logging a node out of a Supernet. The first step performed is for the user to run the SNlogout script and to enter a node ID (step 1002). Next, the SNlogout script requests a log out from SASD (step 1004). Upon receiving this request, SASD removes the mapping for this node from the VARPD that acts as the server for the Supernet (step 1006). SASD then informs KMS to cancel the registration of the node, and KMS terminates this KMC (step 1008). Lastly, KMS generates a new channel key for the channels on which the node was communicating (step 1010) to reduce the likelihood of an intruder being able to intercept traffic." Caronni I at 12:61–13:5. *See also* Caronni I at FIG. 10.
[37.0] The virtual network manager of claim 35 wherein the join module is further	To the extent this reference does not expressly or inherently disclose this limitation, it would have been obvious in view of the background knowledge of a person of ordinary skill in the art. This

configured to maintain data to associate a virtual network address with a device in the virtual network.	reference further renders the limitation obvious in light of other references, such as Mehta and/or Caronni I. Mehta and Caronni I concerns similar subject matter as this reference, namely, virtual networks. As set forth in the accompanying contentions, joining devices with a virtual network was a well-known and commonly-used technique. Modifying this reference with Mehta's and/or Caronni I's teachings would have only required simple substitution, would not have required extensive experimentation, and would have yielded predictable results. A person of ordinary skill in the art would have further understood how to modify the disclosures of this reference with that of Mehta or Caronni I for example, to incorporate this limitation. *See, e.g.*, "In one embodiment, the AMPS comprises one or more modified DNS/API servers, one or more Address Proxy/Routers, an Address Management Data Server, one or more Address Management Data Repositories, and optionally a load balancer. The AMPS DNS'/API server receives a request from a device on a public network for a particular wireless device, and returns an appropriate temporary public address, which is internally mapped to the private address of the wireless device. The public address is then usable by the device on the external public network to send data to the wireless device. The temporary public address is, for example, an address associated with one of the Address Proxy/Routers, which are connected to the external public network and have access to the private addresses on the private wireless network. In some cases, the device on the public network that desires to send data to the wireless device uses a connection-based protocol, such as TCP/IP to send data. In other cases the device uses a connection-less protocol, such as UDP (UDP/IP) to send the data." Mehta at [0012]. "FIG. 3 is an example block diagram of components of an example Address Management Proxy System. In one embodiment, the Address Management Proxy System (AMPS) comprises one or more modified DNS'/ API servers 302, one or more Address Proxy/Routers 305, an Address Management Data Server 303 which manages a database or other repositories such as Address Management Data Repository 304, and optionally a load balancer 301. The DNS'/API servers 302 are either individually connected to a public network 310 or are connected to the load balancer 301 which is in turn connected to public network 310. Similarly, each Address Proxy/Router 305 is also connected to the public

	network 310, via the routable (public) address to which data from the external network 310 is sent that is destined for the wireless devices. The DNS'/API servers 302 are either modified implementations of a DNS server to add functionality necessary to communicate with wireless devices, or are servers that implement one or more specialized APIs, as will be described further below. The DNS'/API servers 302 use the Address Management Data Server 303 to assist in mapping a unique identifier (e.g., string name) for a wireless device to a public address on public network 310. The pool of public addresses is also maintained by the Address Management Data Serve 303 and Data Repository 304. The Address Management Data server 303 and Address Management Data Repository 304 may be implemented using existing database technology, for example, ODBC technology or, may be implemented as a structure such as a simple text file. One skilled in the art will recognize that any embodiment for storing a set of tables, data, lists, or mappings can be used. Each Address Proxy/Router 305 also uses the Address Management Data Server 303 or equivalent to create and update a series of routing tables that are used to assign public addresses to wireless devices as needed and to update the various mappings between public addresses and the non-routable (private) addresses of the wireless devices. The tables and mappings that are maintained on behalf of the DNS'/API servers 302 and the Address Proxy/Routers 305 by the Address Management Data Server 303 are described below with reference to FIG. 6." Mehta at [0032]. "One skilled in the art will recognize that the AMPS 410 may be implemented in a distributed environment that is comprised of multiple, even heterogeneous, computer systems and networks. For example, in one embodiment, the DNS'/API servers 411, the Address Proxy/Router components 412, and the Address Management Data Servers 413 with their data repositories 414 are all located in physically different computer systems. In another embodiment, various components of the AMPS 410 are hosted each on a separate server machine and may be remotely located from the tables which are stored in the address management data repository 414. In addition, under some scenarios, the entire AMPS system 410 may be hosted within a carrier's infrastructure and be completely subsumed by it. Different configurations and locations of programs and data are contemplated for use with techniques of the present invention. In example embodiments, these components may execute

	concurrently and asynchronously; thus the components may communicate using well-known message passing techniques. One skilled in the art will recognize that equivalent synchronous embodiments are also supported by an AMPS implementation. Also, other steps could be implemented for each routine, and in different orders, and in different routines, yet still achieve the functions of the AMPS." Mehta at [0038]. "FIG. 7 is an example flow diagram of an example routine provided by a DNS'/API server of the Address Management Proxy System to return a public address that corresponds to a designated unique identifier. In essence, this routine implements a DNS query or DNS-like query capability for the AMPS using a modified GetHostByName interface or a specialized API, for example GetProxyIP, as described with reference to FIG. 5. In summary, the routine dynamically allocates an appropriate Address Proxy/Router machine to associate with the wireless device and returns the public address of that machine (along with a TTL parameter and potentially other parameters). Specifically, in step 701, the routine determines the private (non-routable) address of the wireless device designated by a string parameter passed as input to the routine. For example, the string parameter may use fields such as "uniqueID.hostname.domain.tld," which specifies a typical hierarchy of person/service on a machine named "hostname" on a domain such as a company's network on a top level domain such as "org." "com," "edu," etc. One skilled in the art will recognize that many other string parameter designations could be used. One mechanism for implementing this routine is to request information from the Address Management Data Repository. In one embodiment, the data repository stores a table that maps unique IDs to private network addresses (see Table 610 in FIG. 6). Preferably, any mechanism that is used by the AMPS stores this data in a secure manner in order to keep the wireless network addresses private. In step 702, the routine retrieves the public network address that corresponds to the private wireless network address of the designated device if one has already been assigned by the AMPS to that device and is still valid. In one embodiment, the data repository stores this mapping information between private wireless network address and public network address (see for example table 630 in FIG. 6). If a public network address has not already been assigned or is not valid, then the routine causes a new public network address to be allocated and that new public address is

	associated with the private wireless network address. Appropriate tables in the data repository are then updated. In step 703, the routine determines the Address Proxy/Router machine that is associated with the assigned public network address (for example using table 620 in FIG. 6). In step 704, the routine sends a request to the determined proxy/router machine to update its routing tables to map the determined public network address to the private wireless network address. In step 705, the routine updates information in the data repository to include any other connection related information (e.g., field 634 in Table 630 in FIG. 6) and indicates a Time to Live (TTL) parameter for the public-private address association (e.g., field 633 in Table 630 in FIG. 6). Once all of the tables have been updated in both the proxy/router and in the data repository, the DNS'/API server returns the determined public network address of the associated Address Proxy/Router machine. As described earlier, the public address may be a (hostname, port) pair when a port-based implementation is used." Mehta at [0057]. "Specifically, in step 801, the routine determines (for example, from the Address Management Data Repository) the private wireless address that corresponds to the invoked public address and the TTL parameter associated with this mapping. These values can be obtained, for example, from the public-to-private address table 630 of FIG. 6. In step 802, the routine determines whether the value of the determined TTL parameter indicates that the mapping has expired, and if so, returns an error, else continues in step 803. In step 803, the routine determines the format required by the target device (the wireless network format). In step 804, the routine determines whether any protocol conversion is necessary and, if so, continues in step 805 else continues in step 806. Note that protocol conversion for a specific wireless network such as converting the data to an "HTTP" packet may be done by the proxy/router or may be done by some other component within the carrier infrastructure. One skilled in the art will recognize that these are example steps and that different formatting or different protocol conversion routines may be added or omitted as specific to the environment. In step 806, the Address Proxy/Router routine sends the data packet (which has been formatted and whose protocol is converted as necessary) to the determined associated private address of the wireless device, and returns." Mehtat at [0059].

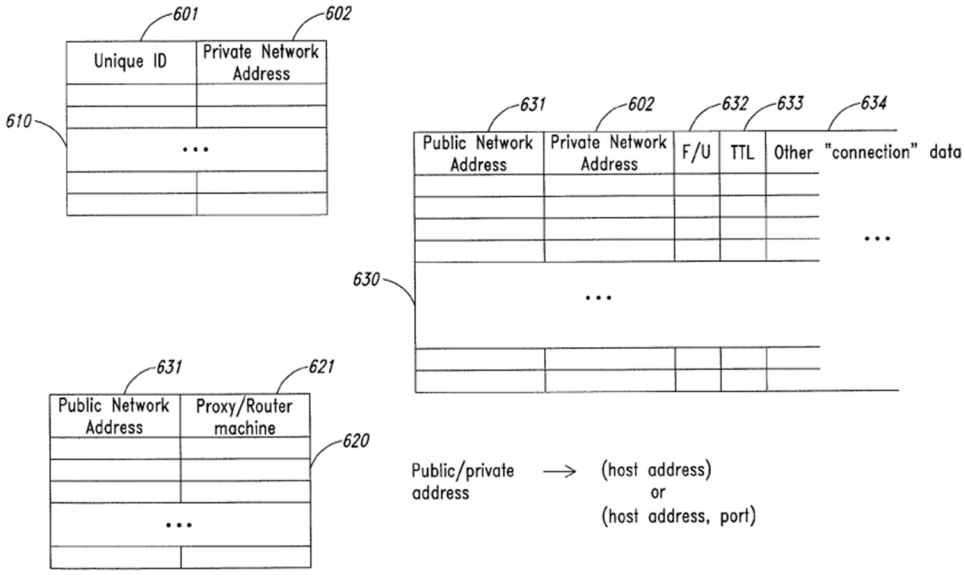


Fig. 6

Mehta at FIG. 6.

"SASD 540 represents a Supernet: there is one instance of an SASD per Supernet, and it both authenticates nodes and authorizes nodes to join the Supernet. VARPD 548 has an associated component, VARPDB 551, into which it stores mappings of the internal Supernet addresses, known as a node IDs, to the network addresses recognized by the public-network infrastructure, known as the real addresses. The "node ID" may include the following: a Supernet ID (e.g., 0x123), reflecting a unique identifier of the Supernet, and a virtual address, comprising an IP address (e.g., 10.0.0.1). Although the virtual address is described in an IP address scheme, one skilled in the art will appreciate that the virtual address may be any other type addressing scheme, such as an e-mail address, IPX, or IPv6. Since the node ID includes a Supernet ID, a node will have more than one node ID when it communicates over more than

	one channel. The "real address" is an IP address (e.g., 10.0.0.2) that is globally unique and meaningful to the public-network infrastructure. In a Supernet, one VARPD runs on each machine, and it may play two roles. First, a VARPD may act as a server by storing all address mappings for a particular Supernet into its associated VARPDB. Second, regardless of its role as a server or not, each VARPD assists in address translation for the nodes on its machine. In this role, the VARPD stores into its associated VARPDB the address mappings for its nodes, and if it needs a mapping that it does not have, it will contact the VARPD that acts as the server for the given Supernet to obtain it. The VARPDB may also decide which virtual address to use in the translation. That is, the VARPDB may associate a virtual address with multiple real addresses or vice versa." Caronni I at 7:3–34. "FIGS. 7A and 7B depict a flow chart of the steps performed when a node joins a Supernet. The first step performed is that the user invokes the SNlogin script and enters the Supernet name, their user ID, their password, and a requested virtual address (step 702). Of course, this information depends on the particular authentication mechanism used. Upon receiving this information, the SNlogin script performs a handshaking with SASD to authenticate this information. In this step, the user may request a particular virtual address to be used, or alternatively, the SASD may select one for them. Next, if any of the information in step 702 is not validated by SASD (step 704), processing ends. Otherwise, upon successful authentication, SASD creates an address mapping between a node ID and the real address (step 706). In this step, SASD concatenates the Supernet ID with the virtual address to create the node ID, obtains the real address of the SNlogin script by querying network services in a well-known manner, and then registers this information with the VARPD that acts as the server for this Supernet. This VARPD is identified in the configuration file." Caronni I at 9:67–10:18. "FIG. 10 depicts a flow chart of the steps performed when logging a node out of a Supernet. The first step performed is for the user to run the SNlogout script and to enter a node ID (step 1002). Next, the SNlogout script requests a log out from SASD (step 1004). Upon receiving this request, SASD removes the mapping for this node from the VARPD that acts as the

	server for the Supernet (step 1006). SASD then informs KMS to cancel the registration of the node, and KMS terminates this KMC (step 1008). Lastly, KMS generates a new channel key for the channels on which the node was communicating (step 1010) to reduce the likelihood of an intruder being able to intercept traffic." Caronni I at 12:61–13:5. *See also* Caronni I at FIG. 10.