



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
90/019,246	09/08/2023	8451845	U022-0010RE	5439
90471	7590	06/03/2024		
Heim, Payne & Chorush, L.L.P. 609 Main St., Suite 3200 Houston, TX 77002			EXAMINER ESCALANTE, OVIDIO	
			ART UNIT	PAPER NUMBER
			3992	
			MAIL DATE	DELIVERY MODE
			06/03/2024	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.



UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

DO NOT USE IN PALM PRINTER

(THIRD PARTY REQUESTER'S CORRESPONDENCE ADDRESS)

LEE & HAYES, P.C.
601 W. RIVERSIDE AVENUE
SUITE 1400
SPOKANE, WASHINGTON 99201

***EX PARTE* REEXAMINATION COMMUNICATION TRANSMITTAL FORM**

REEXAMINATION CONTROL NO. 90/019,246 .

PATENT UNDER REEXAMINATION 8451845 .

ART UNIT 3992 .

Enclosed is a copy of the latest communication from the United States Patent and Trademark Office in the above identified *ex parte* reexamination proceeding (37 CFR 1.550(f)).

Where this copy is supplied after the reply by requester, 37 CFR 1.535, or the time for filing a reply has passed, no submission on behalf of the *ex parte* reexamination requester will be acknowledged or considered (37 CFR 1.550(g)).

DETAILED ACTION

1. This action is in response to the Patent Owner's response filed on the May 8, 2024 and the Supplemental Response filed on May 9, 2024.

Status of the Claims

2. Original claims 1 and 5-13 are rejected.

Response to Arguments

112 6th Paragraph

The Patent Owner states they disagree with the corresponding structures that were identified by the Examiner for each of the claimed means. The Examiner notes that the Patent Owner has not provided their own construction or explained their disagreement with the Examiner's construction.

Therefore, the Examiner will maintain the construction as set forth in the Claim Interpretation Section in the previous Office Action and below.

Home Gateway

The Patent Owner states that the expression "home gateway" refers to any equipment for interconnecting a private network and a network operated by a service provider, the private network being either a home network or a business network. The Patent Owner states that the network operated by the service providers is also referred to as a public network.

The Examiner acknowledges that the '845 Patent discloses "the expression "home gateway" refers to any equipment for interconnecting a private network and a network operated by a service provider, the private network being either a home network or a business network".

Huitema:

PO Argument: Huitema does not describe performing the claimed steps in a home gateway

The Patent Owner states that the expression "home gateway" refers to any equipment for interconnecting a private network and a network operated by a service provider, the private network being

either a home network or a business network. The Patent Owner states that a home gateway is placed “between a home network and the public network or IP domain of the operator”. The Patent Owner further states that a home gateway generally assigns private IP address to the terminal of its home network.

The Patent Owner states that the Examiner points to Huitema’s relay 202 as the home gateway. The Patent Owner explains that the relay 202 interfaces between the IPv6 internet and the IPv4 internet and thus interconnects two public networks (i.e. the IPv6 Internet and the IPv4 Internet), not a public and private network.

The Examiner finds that as set forth in claim 1, the function of a home gateway is: “adapted to connect a user terminal to said IPv6 domain”. The Examiner further finds that claim 8 is not specifically directed to a home gateway but instead is directed to “a device”. Claim 8 recites the “device” as being distinct from a home gateway but that it is ‘for use’ in a home gateway. Thus, the Patent Owner’s arguments with respect to home gateway are not applicable to claim 8.

The Examiner also finds that the Patent Owner is reciting additional functions such as “assigns private IP addresses to the terminals of its home network” or that it is “placed “between a home network and the public network or IP domain of the operator” which are not claimed.

As explained above, the claim (e.g. claim 1) only requires the home gateway to “connect a user terminal to said IPv6 domain” and for executing the claimed method.

The Examiner finds, with respect to Huitema that the user terminal (Fig. 2) which is in a private network, is connected to the IPv6 public network via the Relay 202 (home gateway). Thus, the home gateway of Huitema is configured to connected a user terminal IPv4 Client (private) to the IPv6 (public network). Therefore, the “relay 202” of Huitema performs the same function as defined by the Patent Owner with respect to a “home gateway” and further meets the construction of home gateway as set forth above (i.e. interconnecting a private network and a network operated by a service provider).

PO Argument: Huitema does not describe an IPv6 address constructed by concatenating an IPv6 prefix, an IPv4 destination address, and a destination port number

The Patent Owner states that claims 1, 8 and 11 requires an IPv6 destination address that is constructed from particular components (1) an IPv6 prefix; (2) an IPv4 destination address; and (3) a destination port number.

The Patent Owner states that Huitema teaches constructing an IPv6 address for an IPv4 client that includes five components. The Patent Owner states that the Service field 701 identifies that the IPv6 address corresponds to an IPv4 client, and NAT Flag 703 identifies whether the NAT is a cone NAT. The Patent Owner states that neither value is an IPv6 prefix. The Patent Owner states the IPv6 address shown in Figure 7 seems to suggest the contrary -that no IPv6 prefix is required when constructing the source address of the Huitema client.

The Examiner notes that the Patent Owner has not sufficiently explained why the service field 701 is not a IPv6 prefix. The Examiner, as set forth in the rejection, the TEREDO service prefix is an IPV6 prefix. The Patent Owner states that the Service Field identifies that the IPv6 address corresponds to an IPv4 client. The Examiner notes that the Service Field 701 shows a TEREDO service prefix.

The service field 701 identifies that this IPv6 address corresponds to an IPv4 client. For example, in the IPv6 address "PREF:102:304::FBFE:BFFA:F9F8", the "PREF" represents the service field 701. Since "PREF" represents the service field it is a 'prefix'. In addition, as explained by Huitema that service field 701 identifies that this IPv6 address corresponds to an IPv4 client. Thus, the PREF corresponds to "an IPv6 prefix". The Examiner finds that the Patent Owner has not explained how the claimed IPv6 prefix is different than the IPv6 prefix of Huitema. As explained above, the "PREF" is an IPv6 address and the "PREF" of the IPv6 address represents the service field 701.

Kong and Hinden:

PO Argument: Kong in view of Hinden does not render obvious a port number in a constructed IPv6 address

The Patent Owner states that Kong fails to disclose or teach identifying an IPv6 address constructed in the manner required by the '845 Patent claims because it fails to concatenate "a destination port number" in the IPv6 address.

The Patent Owner explains that when the 64 Translator receives the IPv4 acknowledgement packet, it searches its mapping table for the IPv4 address and IPv4 port of the packet. Id. If a match is found, the packet is translated into an IPv6 acknowledgement packet using the IPv6 address and port information stored in the mapping table and corresponding to the IPv4 address and IPv4 port. The Patent Owner states "This mapping enables the 64 Translator to maintain mapping information without requiring port information to be included in the address. Using these mapping entries and the NAT algorithm, the 64 Translator can translate multiple IPv6 transport identifiers into a single IPv4 address by assigning different port numbers to the different port identifiers. In other words, Kong teaches that port numbers are maintained used separately from the IP addresses to identify the source or destination of a packet."

The Examiner first notes that the function of the 64 Translator is to both allocate a port number as well as maintain the mapping entries. See Table 1. "The functions of the 64 Translator". Thus, a port number is not limited to only being used for mapping as noted by the Patent Owner. In addition, Kong states that the packets include both an IPvX address and port number. Thus, the port number is included as part of the packets. See footnote 1 on page 499. In addition, on page 500, Kong explains that for the sessions from IPv6 network IPv4 network, the sessions are distinguished by the port number. Therefore, the Examiner finds that port numbers are included in the constructed IP addresses.

PO Argument: Kong in view of Hinden does not render obvious a home gateway

The Patent Owner explains that Kong describes the 64 Translator as part of its overall system architecture of its IPv6-IPv4 Translation System using dynamic IP addresses. The Patent Owner states

that as found in Figure 1, the 64 Translator interconnects a native IPv6 Network and a native IPv4 Network. The Patent Owner states that Kong explains that “the IPv4 Network is the legacy Internet,” i.e. a public network. With respect the IPv6 Network, the Patent Owner explains that Kong states that it is “a native IPv6 Network composed [of] IPv6 Hosts and DNSv6 server. The Patent Owner states that this is not a private network such as “a home network or business network”. The Patent Owner maintains that Kong suggests that the native IPv6 Network is a public network or the IP domain of an operator.

The Examiner finds first notes that the Patent Owner’s arguments are not applicable to claim 8 since it is directed to a device “for use” in a home gateway. In addition, the Examiner notes that the 64 Translator connects the IPv4 host on the IPv4 network to the IPv6 network and thus is adapted to connect a user terminal to the IPv6 domain as claimed. The Examiner notes that Kong’s IPv4 host is a desktop computer which is a user terminal. See page 498. The further Examiner notes that as set forth on page 503, Kong discloses that the IPv4 Host “works on an 866MHz Mobile Pentium III with Windows XP”. Therefore, the 64 Translator interconnects the client device on the IPv4 Network to the IPv6 network.

Ananda, Wetterwald and RFC 4380

PO Argument: Ananda in view of Wetterwald and further in view of RFC 4380 does not teach or disclose a home gateway that performs the claimed steps

The Patent Owner states that CPI 220E is not a home gateway, instead Ananda’s CPI 220E connects the IPv4 host to the IPv6 backbone. The Patent Owner states that these are both private networks and in the same domain and thus Ananda does not teach methods for connecting two different domains, and it particular does not teach connections between a private network and a public or operator network.

As set forth above, the Patent Owner’s arguments are not applicable to claim 8 since claim 8 is directed to a device “for use” in a home gateway. Thus, claim 8 does not positively require a home gateway.

With respect to claim 1, the Examiner notes that the rejection of claim 1 states that the CIP 220E (“home gateway”) is adapted to connect IPv4 host 220B (“user terminal”) to the IPv6 backbone 210 (“IPv6 domain”). The Examiner finds that the CPE equipment is located in a subnet (e.g. subnet 215 and 220) which is connected to the IPv6 communication backbone 210. The Examiner also acknowledges that both the IPv6 backbone 201 and the subnets are part of the same communication network 200.

PO Argument: Ananda in view of Wetterwald and further in view of RFC 4380 does not disclose an IPv6 address constructed by concatenating an IPv6 Prefix, an IPv4 destination address and a destination port number

The Patent Owner states that the Examiner contends that Wetterwald teaches that IPv6 extension headers can be included in IPv6 packets “to identify the source and destination port numbers” of an encapsulated IPv4 packet and that RFC 4380 could be combined with Wetterwald to include a destination port number in a constructed address.

The Patent Owner states that Wetterwald’s clusterhead prefix consumes 96 bits and the IPv4 address consumes 32 bites, so there is no room to encode port information the IPv6 destination address without modifying the packet format taught by Wetterwald. The Patent Owner states that because an IPv6 address is limited to 128 bits, there is no room for additional bits in the IPv6 address, such as the bits necessary to identify a port (or port range), when implementing Wetterwald’s packet format.

The Examiner notes that the rejection is based on the combination of Ananda, Wetterwald and RFC4380. The rejection took the position that a POSITA would have been motivated to modify Ananda’s IPv6 address construction technique to incorporate a destination port number in the IPv6 address as taught by Wetterwald and RFC4380. The rejection did not proposed to modify Wetterworld rather it was maintained that Wetterwald teaches that including IPv6 extension headers which identify the source and destination port number were known and that it would have been obvious to include port numbers in Ananda.

Therefore the Examiner does not find the Patent Owner’s arguments persuasive.

The Patent Owner further argues that Wetterwald also does not teach that the IPv6 address would be constructed to include port information, other than to reference the “IPv6 option headers” in RFC 2460.

The Examiner notes that Wetterwald teaches that “the translator 42 also may add IPv6 option headers related to the layer 4 TCP/UDP source and destination ports, as described in detail in RFC 2460”.

The Patent Owner notes that RFC 2460 confirms that any optional information is encoded in separate headers that are distinct from the IPv6 header information where the source and destination IPv6 addresses are located.

The Patent Owner, on pages 56-60 of their response, further argues that a POSITA would not combine Ananda, Wetterwald and RFC 4380.

The Examiner acknowledges that the use of additional optional headers in a portion of an IPv6 packet is separate from the IPv6 headers. Upon further review of the Patent Owner’s arguments with respect to the use of the optional header information as disclosed by Wetterwald and RFC 2380 80. The Examiner finds persuasive that the combination does not disclose or render obvious “identifying an IPv6 destination address constructed by concatenating an IPv6 prefix with an IPv6 destination address and a destination port number as recited in claims 1, 8 and 11.

Claim Interpretation

3. The following is a quotation of 35 U.S.C. 112(f):

(f) Element in Claim for a Combination. – An element in a claim for a combination may be expressed as a means or step for performing a specified function without the recital of structure, material, or acts in support thereof, and such claim shall be construed to cover the corresponding structure, material, or acts described in the specification and equivalents thereof.

The following is a quotation of pre-AIA 35 U.S.C. 112, sixth paragraph:

An element in a claim for a combination may be expressed as a means or step for performing a specified function without the recital of structure, material, or acts in support thereof, and such claim

shall be construed to cover the corresponding structure, material, or acts described in the specification and equivalents thereof.

The claims in this application are given their broadest reasonable interpretation using the plain meaning of the claim language in light of the specification as it would be understood by one of ordinary skill in the art. The broadest reasonable interpretation of a claim element (also commonly referred to as a claim limitation) is limited by the description in the specification when 35 U.S.C. 112(f) or pre-AIA 35 U.S.C. 112, sixth paragraph, is invoked.

As explained in MPEP § 2181, subsection I, claim limitations that meet the following three-prong test will be interpreted under 35 U.S.C. 112(f) or pre-AIA 35 U.S.C. 112, sixth paragraph:

- (A) the claim limitation uses the term “means” or “step” or a term used as a substitute for “means” that is a generic placeholder (also called a nonce term or a non-structural term having no specific structural meaning) for performing the claimed function;
- (B) the term “means” or “step” or the generic placeholder is modified by functional language, typically, but not always linked by the transition word “for” (e.g., “means for”) or another linking word or phrase, such as “configured to” or “so that”; and
- (C) the term “means” or “step” or the generic placeholder is not modified by sufficient structure, material, or acts for performing the claimed function.

Use of the word “means” (or “step”) in a claim with functional language creates a rebuttable presumption that the claim limitation is to be treated in accordance with 35 U.S.C. 112(f) or pre-AIA 35 U.S.C. 112, sixth paragraph. The presumption that the claim limitation is interpreted under 35 U.S.C. 112(f) or pre-AIA 35 U.S.C. 112, sixth paragraph, is rebutted when the claim limitation recites sufficient structure, material, or acts to entirely perform the recited function.

Absence of the word “means” (or “step”) in a claim creates a rebuttable presumption that the claim limitation is not to be treated in accordance with 35 U.S.C. 112(f) or pre-AIA 35 U.S.C. 112, sixth paragraph. The presumption that the claim limitation is not interpreted under 35 U.S.C. 112(f) or pre-AIA

35 U.S.C. 112, sixth paragraph, is rebutted when the claim limitation recites function without reciting sufficient structure, material or acts to entirely perform the recited function.

Claim limitations in this application that use the word “means” (or “step”) are being interpreted under 35 U.S.C. 112(f) or pre-AIA 35 U.S.C. 112, sixth paragraph, except as otherwise indicated in an Office action. Conversely, claim limitations in this application that do not use the word “means” (or “step”) are not being interpreted under 35 U.S.C. 112(f) or pre-AIA 35 U.S.C. 112, sixth paragraph, except as otherwise indicated in an Office action.

1. Functional Phrase #1

means for identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address, and a destination port number

--“**Functional Phrase #1**” or “**FP#1**” – From claim 8

i. 3 Prong Analysis: Invocation Prong (A)

In accordance with Invocation Prong (A), the MPEP states:

(A) The claim limitation uses the term “means” or “step” or a term used as a substitute for “means” that is a generic placeholder (also called a nonce term or a non-structural term having no specific structural meaning) for performing the claimed function...

MPEP 2181.I. – Invocation Prong (A)

The Examiner finds that Functional Phrase #1 uses the term “means” and thus is a generic placeholder for performing the claimed function.

ii. 3 Prong Analysis: Invocation Prong (B)

Based upon review of the Functional Phrase #1, the Examiner finds that for FP#1, the claimed function is:

identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address, and a destination port number

--Function of FP#1

The Examiner finds that because nothing in the patent specification or the prosecution history contradict the plain language as set forth in the Function of FP#1, Function of FP#1 will have their

ordinary and accustomed meaning.

Since Functional Phrase #1 includes the Function of FP#1, the Examiner concludes that, Functional Phrase #1 meets invocation prong (B).

iii. 3 Prong Analysis: Invocation Prong (C)

Based upon a review of Functional Phrase #1, the Examiner finds that Functional Phrase #1 does not contain sufficient structure for performing the entire Function of FP#1.

Because Functional Phrase #1 does not contain sufficient structure for performing the entire claimed function, the Examiner concludes that Functional Phrase #1 meets invocation Prong (C).

Because Functional Phrase #1 meets the three prong analysis as set forth in MPEP § 2181.I, the Examiner concludes that Functional Phrase #1 invokes 112, sixth paragraph.

iv. Corresponding Structure or Materials

“The next step in construing a means-plus-function claim limitation is to look to the specification and identify the corresponding structure for that function.” *In re Aoyama*, 656 F.3d 1293, 1297 (Fed. Cir. 2011) quoting *Golight*, 355 F.3d at 1333. “Under this second step, structure disclosed in the specification is ‘corresponding’ structure only if the specification or prosecution history clearly links or associates that structure to the function recited in the claim.” *Aoyama*, 656 F.3d at 1297 (quoting *Med. Instrumentation & Diagnostics Corp. v. Elekta AB*, 344 F.3d 1205, 1210 (Fed. Cir. 2003)).

Furthermore, if the claimed phase is meant to ‘cover’¹ software, “[i]t is well-established that the corresponding structure for a function performed by a software algorithm is the algorithm itself.” *EON Corp. IP Holdings LLC v. AT&T Mobility LLC*, 785 F.3d 616, 621 (Fed. Cir. 2015). In other words, “[i]f

¹ See 35 U.S.C. § 112 6th ¶ which states in part that “[a]n element in a claim for a combination ... **shall be construed to cover** the corresponding structure, material, or acts described in the specification and equivalents thereof. [Emphasis added.]”

special programming is required for a general-purpose computer to perform the corresponding claimed function, then the default rule requiring disclosure of an algorithm applies.” *Ergo Licensing, LLC v. CareFusion 303, Inc.*, 673 F.3d 1361, 1365 (Fed. Cir. 2012).

The Examiner notes that the ‘845 patent describes “a device 25 [] for receiving a data packet in an IPv6 domain.” See col. 16, lines 56-58. The ‘845 patent explains that the device 25 includes “hardware elements conventionally found in a standard computer or a dedicated computer router,” including a processor 25₁, a telecommunication means 25₄ for communicating with network 1, and a read-only memory 25₅. See col. 16, lines 55-col. 17 line 7.

The Examiner finds that the structure corresponding to the claimed means is: a receiver device (e.g. device 25) of the home gateway 20 having a processor 25₁ and memory 25₅. See col. 10, lines 61 – col. 11, line 4; col. 16, line 55 – col. 17, line 7.

The ‘845 patent explains that the disclosed method of receiving an IPv6 data packet is “executed by a receiver device included in a home gateway connected to the IPv6 domain, such as the gateway 20,” and includes a step R1 (Fig. 3) of identifying an IPv6 destination address. See 10:61-11:4. The ‘845 patent explains that the receiver device of the home gateway 20 determines whether the IPv6 destination address is a constructed IPv6 address based on whether the IPv6 destination address contains a certain IPv6 prefix. See 13:11-16, 10:10-22, 12:40-43, 13:10-14.

2. Functional Phrase #2

means for regularizing, if necessary, at least one of the IPv6 source address and the IPv6 destination address of the data packet by replacing said at least one IPv6 address, the replacement process belonging to a group comprising replacement of a constructed address with a native address and replacement of a native address with a constructed address, and modifying the data packet using the regularized address

--“**Functional Phrase #2**” or “**FP#2**” – From claim 8

i. 3 Prong Analysis: Invocation Prong (A)

In accordance with Invocation Prong (A), the MPEP states:

(B) The claim limitation uses the term “means” or “step” or a term used as a substitute for “means” that is a generic placeholder (also called a nonce term or a non-structural term having no specific structural meaning) for performing the claimed function...

MPEP 2181.I. – Invocation Prong (A)

The Examiner finds that Functional Phrase #2 uses the term “means” and thus is a generic placeholder for performing the claimed function.

ii. 3 Prong Analysis: Invocation Prong (B)

Based upon review of the Functional Phrase #2, the Examiner finds that for FP#2, the claimed function is:

regularizing, if necessary, at least one of the IPv6 source address and the IPv6 destination address of the data packet by replacing said at least one IPv6 address, the replacement process belonging to a group comprising replacement of a constructed address with a native address and replacement of a native address with a constructed address, and modifying the data packet using the regularized address

--Function of FP#2

The Examiner finds that because nothing in the patent specification or the prosecution history contradict the plain language as set forth in the Function of FP#2, Function of FP#2 will have their ordinary and accustomed meaning.

Since Functional Phrase #2 includes the Function of FP#2, the Examiner concludes that, Functional Phrase #2 meets invocation prong (B).

iii. 3 Prong Analysis: Invocation Prong (C)

Based upon a review of Functional Phrase #2, the Examiner finds that Functional Phrase #2 does not contain sufficient structure for performing the entire Function of FP#2.

Because Functional Phrase #2 does not contain sufficient structure for performing the entire claimed function, the Examiner concludes that Functional Phrase #2 meets invocation Prong (C).

Because Functional Phrase #2 meets the three prong analysis as set forth in MPEP § 2181.I, the Examiner concludes that Functional Phrase #2 invokes 112, sixth paragraph.

iv. Corresponding Structure or Materials

“The next step in construing a means-plus-function claim limitation is to look to the specification and identify the corresponding structure for that function.” *In re Aoyama*, 656 F.3d 1293, 1297 (Fed. Cir. 2011) quoting *Golight*, 355 F.3d at 1333. “Under this second step, structure disclosed in the specification is ‘corresponding’ structure only if the specification or prosecution history clearly links or associates that structure to the function recited in the claim.” *Aoyama*, 656 F.3d at 1297 (quoting *Med. Instrumentation & Diagnostics Corp. v. Elekta AB*, 344 F.3d 1205, 1210 (Fed. Cir. 2003)).

Furthermore, if the claimed phase is meant to ‘cover’² software, “[i]t is well-established that the corresponding structure for a function performed by a software algorithm is the algorithm itself.” *EON Corp. IP Holdings LLC v. AT&T Mobility LLC*, 785 F.3d 616, 621 (Fed. Cir. 2015). In other words, “[i]f special programming is required for a general-purpose computer to perform the corresponding claimed function, then the default rule requiring disclosure of an algorithm applies.” *Ergo Licensing, LLC v. CareFusion 303, Inc.*, 673 F.3d 1361, 1365 (Fed. Cir. 2012).

The Examiner notes that the ‘845 patent describes “a device 25 [] for receiving a data packet in an IPv6 domain.” See col. 16, lines 56-58. The ‘845 patent explains that the device 25 includes “hardware elements conventionally found in a standard computer or a dedicated computer router,” including a processor 25₁, a telecommunication means 25₄ for communicating with network 1, and a read-only memory 25₅. See col. 16, lines 55-col. 17 line 7.

² See 35 U.S.C. § 112 6th ¶ which states in part that “[a]n element in a claim for a combination ... **shall be construed to cover** the corresponding structure, material, or acts described in the specification and equivalents thereof. [Emphasis added.]”

The Examiner finds that the structure corresponding to the claimed means is: the device 25 of the home gateway 20 having a processor 25₁ and a memory 25₅, where the memory 25₅ stores a table for translating between addresses, and equivalents. See 16:55-17:7, 16:18-44.

With reference to Figure 1, the '845 patent discloses that when a packet is transmitted from an IPv4 domain 3 to the home gateway 20 in the IPv6 domain, the IPv6 source address of the packet is regularized using an address translation table, whereas when a packet is transmitted from an IPv6 terminal in the IPv6 domain 1 via the gateway 20 to the IPv4 network, the IPv6 destination address is regularized using an address translation table. See 3:10-15, 11:7-11, 16:18-44, 16:55-17:7.

3. Functional Phrase #3

means for routing the modified data packet to its destination.

--“Functional Phrase #3” or “FP#3” – From claim 8

i. 3 Prong Analysis: Invocation Prong (A)

In accordance with Invocation Prong (A), the MPEP states:

(C) The claim limitation uses the term “means” or “step” or a term used as a substitute for “means” that is a generic placeholder (also called a nonce term or a non-structural term having no specific structural meaning) for performing the claimed function...

MPEP 2181.I. – Invocation Prong (A)

The Examiner finds that Functional Phrase #3 uses the term “means” and thus is a generic placeholder for performing the claimed function.

ii. 3 Prong Analysis: Invocation Prong (B)

Based upon review of the Functional Phrase #3, the Examiner finds that for FP#3, the claimed function is:

routing the modified data packet to its destination.

--Function of FP#3

The Examiner finds that because nothing in the patent specification or the prosecution history contradict the plain language as set forth in the Function of FP#3, Function of FP#3 will have their ordinary and accustomed meaning.

Since Functional Phrase #3 includes the Function of FP#3, the Examiner concludes that, Functional Phrase #3 meets invocation prong (B).

iii. 3 Prong Analysis: Invocation Prong (C)

Based upon a review of Functional Phrase #3, the Examiner finds that Functional Phrase #3 does not contain sufficient structure for performing the entire Function of FP#3.

Because Functional Phrase #3 does not contain sufficient structure for performing the entire claimed function, the Examiner concludes that Functional Phrase #3 meets invocation Prong (C).

Because Functional Phrase #3 meets the three prong analysis as set forth in MPEP § 2181.I, the Examiner concludes that Functional Phrase #3 invokes 112, sixth paragraph.

iv. Corresponding Structure or Materials

“The next step in construing a means-plus-function claim limitation is to look to the specification and identify the corresponding structure for that function.” *In re Aoyama*, 656 F.3d 1293, 1297 (Fed. Cir. 2011) quoting *Golight*, 355 F.3d at 1333. “Under this second step, structure disclosed in the specification is ‘corresponding’ structure only if the specification or prosecution history clearly links or associates that structure to the function recited in the claim.” *Aoyama*, 656 F.3d at 1297 (quoting *Med. Instrumentation & Diagnostics Corp. v. Elekta AB*, 344 F.3d 1205, 1210 (Fed. Cir. 2003)).

Furthermore, if the claimed phase is meant to ‘cover’³ software, “[i]t is well-established that the corresponding structure for a function performed by a software algorithm is the algorithm itself.” *EON Corp. IP Holdings LLC v. AT&T Mobility LLC*, 785 F.3d 616, 621 (Fed. Cir. 2015). In other words, “[i]f special programming is required for a general-purpose computer to perform the corresponding claimed function, then the default rule requiring disclosure of an algorithm applies.” *Ergo Licensing, LLC v. CareFusion 303, Inc.*, 673 F.3d 1361, 1365 (Fed. Cir. 2012).

The Examiner notes that the ‘845 patent describes “a device 25 [] for receiving a data packet in an IPv6 domain.” See col. 16, lines 56-58. The ‘845 patent explains that the device 25 includes “hardware elements conventionally found in a standard computer or a dedicated computer router,” including a processor 25₁, a telecommunication means 25₄ for communicating with network 1, and a read-only memory 25₅. See col. 16, lines 55-col. 17 line 7.

The Examiner finds that the structure corresponding to the claimed means is: telecommunications means 25₄ of the device 25 of the home gateway 20. See 16:58-63.

The ‘845 patent explains that the device 25 of the home gateway 20 includes “the hardware elements conventionally found in a standard computer or a dedicated router, namely a... telecommunications means 25₄ for communicating with the network 1.” See 16:58-63.

35 U.S.C. 305

4. Claims 11-13 are rejected under 35 U.S.C. 305 as enlarging the scope of the claim(s) of the patent being reexamined. In 35 U.S.C. 305, it is stated that “[n]o proposed amended or new claim enlarging the scope of a claim of the patent will be permitted in a reexamination proceeding...” A claim presented in a reexamination “enlarges the scope” of the patent claim(s) where the claim is broader than each and every

³ See 35 U.S.C. § 112 6th ¶ which states in part that “[a]n element in a claim for a combination ... **shall be construed to cover** the corresponding structure, material, or acts described in the specification and equivalents thereof. [Emphasis added.]”

Art Unit: 3992

claim of the patent. A claim is broader in scope than the original claims if it contains within its scope any conceivable product or process which would not have infringed the original patent. A claim is broadened if it is broader in any one respect, even though it may be narrower in other respects.

Claim 11 is directed to **a home gateway**. In determining whether “a claim presented in a reexamination “enlarges the scope” of the patent claim(s) where the claim is broader than each and every claim of the patent” the Examiner acknowledges that the original claims the ‘845 patent is directed to:

Claim 1:

“a method for receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain, said packet comprising an IPv6 destination address and an IPv6 source address, said method being executed in a home gateway adapted to connect a user terminal to said IPv6 domain...”

Original Claim 7:

A non-transitory computer program product that is at least one of downloadable from a communications network and stored on a computer-readable medium and executable by a microprocessor, comprising program code instructions for executing the method according to claim 1 of receiving a data packet in an IPv6 domain...”

Original claim 8:

“A device for receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain, said packet comprising an IPv6 destination address and an IPv6 source address, said device for use in a home gateway adapted to connect a user terminal to said IPv6 domain...”

Original claim 9:

A home gateway adapted to connect user terminals of a home network to an IPv6 domain connected to an IPv4 domain...”

The Examiner finds that new claim 11 is directed to a home gateway. Since the claim is an apparatus type of claim is broadened with respect to claims 1 and 7.

In reviewing claim 8 and 9, the Examiner finds that each of claims 8 and 9 both recite a device and a home gateway. Claim 8 recites “A device for receiving an IPv6 data packet in an IPv6 domain

connected to an IPv4 domain.... said device for use in a home gateway adapted to connect a user terminal to said IPv6 domain.” and claim 9 recites “ A home gateway adapted to connect user terminals of a home network to an IPv6 domain connected to an IPv4 domain..... a device according to claim 8 for receiving an IPv6 data packet in the IPv6 domain connected to the IPv4 domain.”

Therefore, new claim 11 must be compared to the original claim 9 to determine whether broadening exists since only claim 9 is specifically directed to a home gateway, which as set forth above is different than the claimed device.” As set forth above, claim 8 is directed to a device and not a home gateway wherein the device includes a plurality of “means for” limitations—the claim device is distinct from a home gateway.

In reviewing claim 9, the Examiner finds that new claim 11 no longer requires “means for obtaining an IPv4 address, a range of port numbers authorized for said gateway, and a range of IPv6 addresses authorized for said gateway, said range of IPv6 addresses being constructed by concatenating an IPv6 prefix, said IPv4 address, and the range of authorized IPv4 port numbers; and a device according to claim 8 for receiving an IPv6 data packet in the IPv6 domain connected to the IPv4 domain.” The Examiner finds that claim 9 recites various means (“means for” limitations) as well as reciting the above identified function (i.e., the limitation directed to a range of port number and addresses) and new claim 11 is not shown to have the same corresponding structure as required by the claimed means.

Therefore, claim 11 is broadened with respect to original claims of the ‘845 patent.

Claim Rejections - 35 USC § 102

5. In the event the determination of the status of the application as subject to AIA 35 U.S.C. 102 and 103 (or as subject to pre-AIA 35 U.S.C. 102 and 103) is incorrect, any correction of the statutory basis (i.e., changing from AIA to pre-AIA) for the rejection will not be considered a new ground of rejection if the prior art relied upon, and the rationale supporting the rejection, would be the same under either status.

The following is a quotation of the appropriate paragraphs of pre-AIA 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for a patent.

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

6. Claim(s) 1, 7, 8 and 11 is/are rejected under pre-AIA 35 U.S.C. 102(a) as being anticipated by Huitema US Patent 7,245,622

Regarding claim 1:

A method for receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain,

Huitema discloses a method (message flow of Figures 3A, 3B) for receiving in IPv6 data packet (response 311) in an IPv6 domain (IPv6 Internet) connected to an IPv4 domain (IPv4 Internet).

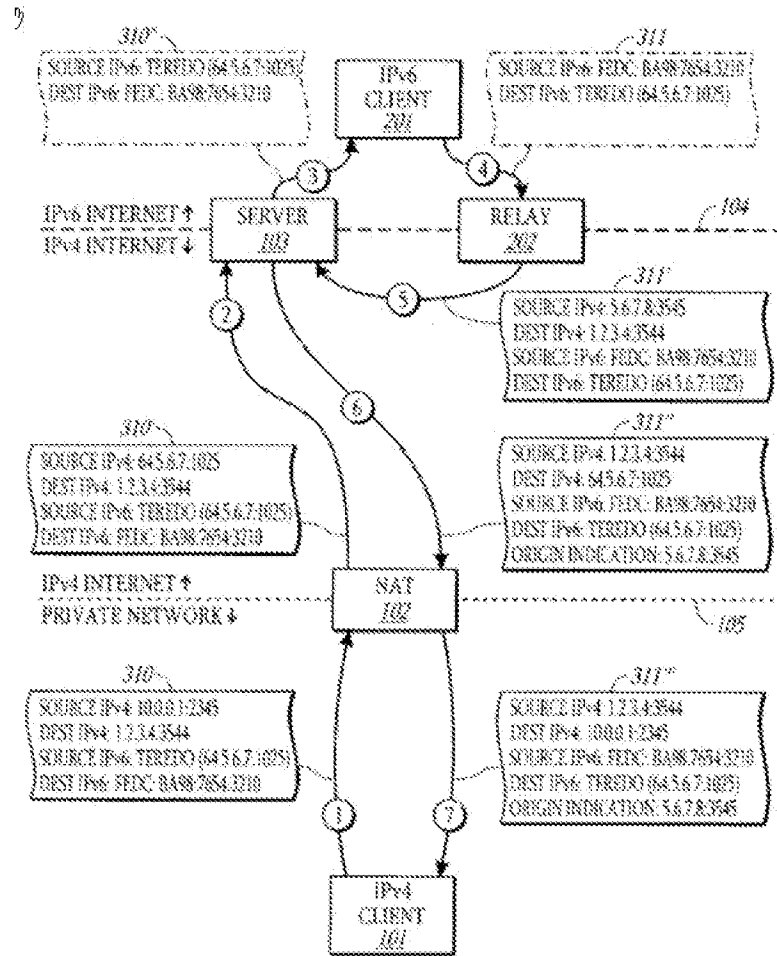
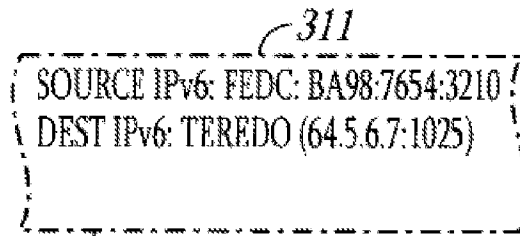


FIG. 3A

As shown above, in Figure 3A (see also 3B), Huitema shows a message flow in which IPv4client 101 communicates with IPv6 client 201. See col. 11, lines 62 – col. 14, line 52. IPv6 client resides in an IPv6 domain (IPv6 Internet), IPv4 client 101 resides in a private network of an IPv4 domain (IPv4 Internet) and relay 202 and server 103 connect to the IPv6 domain to the IPv4 domain. See col. 3, lines 3- col. 5, line 25.

said packet comprising an IPv6 destination address and an IPv6 source address,

As shown in Figure 3A (see also below message 311), the packet (311) comprises an IPv6 destination address (DEST IPv6: TEREDO (64.5.6:1025) and an IPv6 source addresses (SOURCE IPv6: FEDC: BA98:7654:3210). See col. 12, lines 32-40.



See also col. 11, lines 3-6 which explains that “all IPv6 messages are represented with intermittent dashed/dotted lines, as contrasted with IPv4 message which are represented with solid lines.”

See also Figures 2 and 4 which also discloses the packet comprising an IPv6 destination address and IPv6 source address. See col. 1, lines 10-19, col. 14, lines 58 – col. 15, line 3

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

Huitema discloses the method (message flow of Figures 3A and 3B) being executed in a home gateway (relay 202) adapted to connect to a user terminal (IPv4 client 101) or (IPv6 client 201) to said IPv6 domain (IPv6 Internet). The Examiner notes that the relay 202 is a home gateway since it connects IPv4 client 101 on a private network to the IPv6 client 201 on the IPv6 network (i.e. is adapted to connect a user terminal (IPv4 client) to said Ipv6 domain).

The relay 202 connects IPv4 client 101 to the IPv6 Internet because relay 202 sits between the IPv6 and IPv4 Internet and IPv4 client resides in a private network of the IPv4 Internet. See col. 9, lines 45-54 and Fig. 3A.

identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address and a destination port number;

Huitema discloses the relay 202 identifying in response 311 an IPv6 destination address (DEST IPv6) constructed by concatenating an IPv6 prefix (TEREDO service prefix), an IPv4 destination addresses (64.5.6.7) and a destination port number (1025). See Figure 3A and 7

Huitema explains that IPv6 addresses (response 311) have “data structure 700” shown in Figure 7.

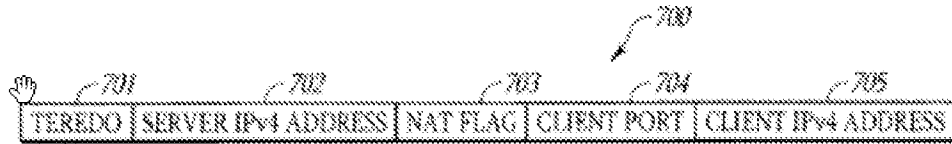
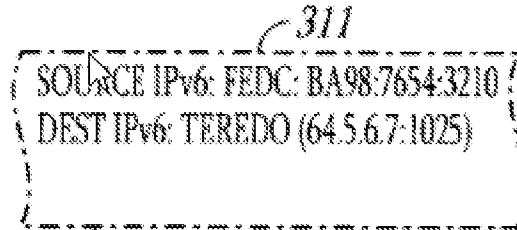


FIG. 7



With respect to IPv6 prefix, Huitema discloses Service field 701 (“PREF”) shows a TEREDO service prefix identifies that this IPv6 address corresponds to an IPv4 client. See col. 8, lines 39-40. Service field 702 shows “[a] service IPv4 address field 702 identifies the IPv4 address of the server 103. For example, the ‘102:304’ in the example IPv6 address is the hexadecimal notation of the address of the server 103 (namely, 1.2.3.4).” See col. 8, lines 42-46.

With respect to IPv4 destination address, Huitema discloses IP address 64.5.6.7 is the IPv4 address of IPv4 client 101. Huitema explains that IPv4 client 101 has “the globally unique IPv4 network address 64.5.6.7”. See col. 7, lines 51-53.

See also Figure 7 and col. 8, lines 51-55 which discloses “client IPv4 address field 705 identifies the IPv4 address of the client 101.”

With respect to destination port number, as shown above in Figure 3A and message 311, Huitema discloses the IPv6 destination address includes port number 1025 concatenated with IPv4 destination address 64.5.6.7. Specifically, port number 1025 is the UDP port number of IPv4 client 101. See col. 8, line 6-10. See also col. 8, lines 46-51

As shown in both Fig. 7 and Figure 3A the IPv6 destination address is constructed by concatenating an IPv6 prefix, an IPv4 destination address and a destination port number.

In addition, Huitema discloses that the relay identifies the IPv6 destination address by detecting the presence of the IPv6 prefix in that address. See col. 12, lines 41, 46-51.

if necessary, regularizing *at least one of the IPv6 source address and the IPv6 destination address of the data packet by replacing said at least one IPv6 address,*

The Examiner first determines that the entire “if necessary” limitation is optional and therefore is not required in this method. The limitation states “if necessary” and thus is not needed to be performed since it is contingent on whether “if necessary” is satisfied. See MPEP 2111.4(II).

Nonetheless, Huitema discloses if necessary (if a TEREDO prefix is detected), regularizing (modifies from IPv6 to IPv4) at least one of the IPv6 source address (::FEDC:BA98:7654:3210) and the IPv6 destination address (TEREDO (64.5.6.7:1025)) of the data packet (response 311) by replacing said at least one IPv6 address (replacing FEDC:BA98:7654:3210 and TEREDO (64.5.6.7:1025) respectively with IPv4 source address 5.6.7.8:3545 and IPv4 destination address 1.2.3.4:3544). See col., 12, lines 34-67. As shown in Fig. 3A response 311 has an IPv6 format and response 311' has an IPv4 format. See col. 11, lines 3-6. The relay regularizes IPv6 format response 311 into IPv4 format response 311'.

Thus, as explained above, Huitema discloses the relay 202 regularizes the **IPv6 source address** (FEDC:BA98:7654:3210) in the header of IPv6 response 311 by replacing the IPv6 source address with a corresponding IPv4 source address (5.6.7.8:3545) in the header of the IPv4 response 311' and moving the IPv6 source address (FEDC:BA98:7654:3210) from the header to the payload in IPv4 response 311'. See col. 12, lines 33-67.

Huitema discloses the relay regularizes the **IPv6 destination address** (TEREDO (64.5.6.7:1025)) by replacing the IPv6 destination address in the header of IPv6 response 311 with a corresponding IPv4 destination (1.2.3.4:3544) in the header of IPv4 response 311' and moving the IPv6 destination address (TEREDO (64.5.6.7:1025)) from the header to the payload in IPv4 response 311'. See col. 12, lines 33-67. See also Figure 6, col. 9, lines 58-64 and col. 11, lines 27-29.

the replacement process belonging to a group comprising replacement of a constructed address with a native address and replacement of a native address with a constructed address, and

Huitema discloses the replacement process (replacing the IPv6 addresses in the header of IPv6 response 311 with corresponding IPv4 addresses to form IPv4 response 311') belonging to a group comprising replacement of a constructed address (TEREDO (64.5.6.7:1025)) with a native address (1.2.3.4:3544) and replacement of a native address (FEDC:BA98:7654:3210) with a constructed address (5.6.7.8:3545), as claimed.

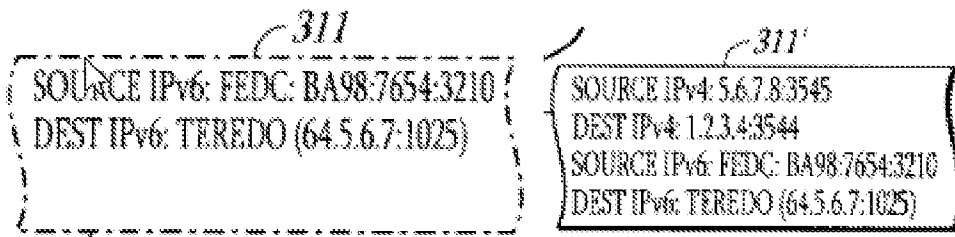
As explained above, in regularizing IPv6 response 311 into IPv4 response 311', Huitema's relay 202 replaces IPv6 destination address TEREDO (6.4.5.6.7:1025) with corresponding IPv4 destination address 1.2.3.4:3544. See 12:31-67. Here, TEREDO (6.4.5.6.7:1025) is a constructed address for IPv4 client 101, because IPv4 client 101 does not have a native IPv6 address. Instead, "client 101 ... constructs its own IPv6 address" from the IPv6 TEREDO service prefix 111a and client 101's own IPv4 address 111B of 64.5.6.7:1025 provided in response 111 received from server 103. See 8:8-26, 35-60, FIGS. 1, 7. And IPv4 destination address 1.2.3.4:3544 that replaces the IPv6 address is a native address because it is native to IPv4 Internet on which IPv4 client 101 resides.

See also Figure 2, col. 11, lines 29-57 and Fig. 4, col. 14, lines 58-67, col. 15, line 66 – col. 16, line 12.

modifying the data packet using the regularized address; and

Huitema's relay 202 modifies the data packet (converts IPv6 response 311 into IPv4 response 311') using the regularized address (using IPv4 destination address 1.2.3.4:3544 and/or IPv4 source address 5.6.7.8:3544 in the header of response 311').

As explained above, Huitema's relay 202 modifies IPv6 response 311 (see below message 311) into IPv4 response 311' (see below message 311') by replacing IPv6 destination address TEREDO (6.4.5.6.7:1025) and IPv6 source address FEDC:BA98:7654:3210 in the header of response 311 respectively with IPv4 destination address 1.2.3.4:3544 and IPv4 source address 5.6.7.8:3544 in the header of response 311. See 12:33-13:3.



See also Figure 2, col. 11, lines 29-57 and Fig. 4, col. 14, lines 58-67, col. 15, line 66 – col. 16, line 12.

routing the IPv6 data packet to its destination.

Huitema's relay 202 routes the IPv6 data packet (response 311, 311' and 311'') to its destination (IPv4 client 101). See Fig. 3A.

See also Figure 2, col. 11, lines 29-57 and Fig. 4, col. 14, lines 58-67, col. 15, line 66 – col. 16, line 12.

Regarding claim 7:

A non-transitory computer program product that is at least one of downloadable from a communications network and stored on a computer-readable medium and executable by a microprocessor, comprising program code instructions for executing the method according to claim 1 of receiving a data packet in an IPv6 domain.

The Examiner incorporates by reference the rejection of claim 1 set forth above for meeting “the method of claim 1 of receiving a data packet in an IPv6 domain.

In addition, Huitema discloses that relay 202 (as well as clients 102, 201; NAT 102; and server 103) “may be structured as... described with respect to the conventional computer 820 of Fig. 8.” See col. 16, lines 36-39. With reference to Figure 8, Huitema discloses a computer 820 (i.e. relay 202) has a non-transitory computer program product (e.g. operating system 835), application programs 836, program modules 837, program data 838) stored on a computer-readable medium (e.g. system memory 822

including RAM 824 and/or ROM 825, hard disk 839, removable hard disk 829 and optical disk 831) and executable by a microprocessor (e.g. processing unit 821). See col. 16, line 39 – col. 17 line 23).

Regarding claim 8:

A device for receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain,

Huitema discloses a device (for receiving in IPv6 data packet (response 311) in an IPv6 domain (IPv6 Internet) connected to an IPv4 domain (IPv4 Internet).

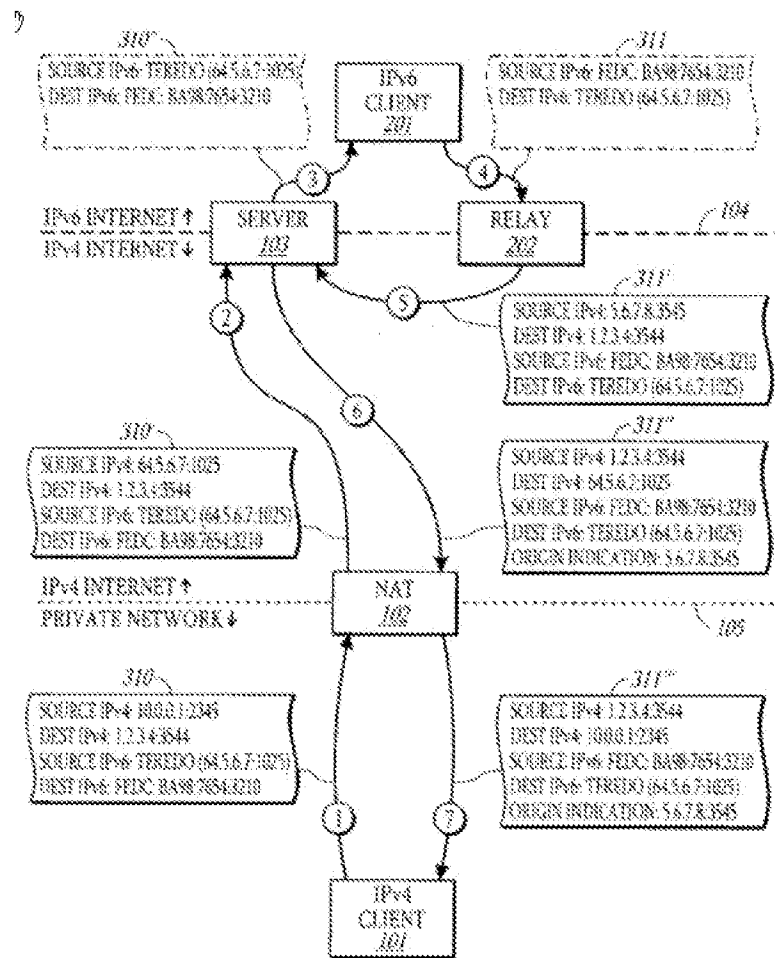


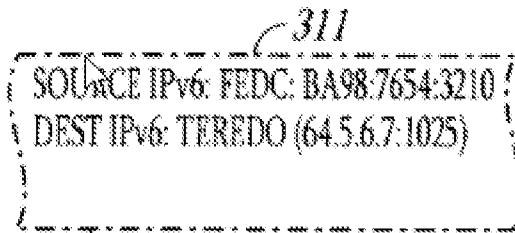
FIG. 3A

As shown above, in Figure 3A (see also 3B), Huitema shows a message flow in which IPv4client 101 communicates with IPv6 client 201. See col. 11, lines 62 – col. 14, line 52. IPv6 client resides in an IPv6 domain (IPv6 Internet), IPv4 client 101 resides in a private network of an IPv4 domain (IPv4

Internet) and relay 202 and server 103 connect to the IPv6 domain to the IPv4 domain. See col. 3, lines 3-
col. 5, line 25.

said packet comprising an IPv6 destination address and an IPv6 source address,

As shown in Figure 3A (see also below message 311), the packet (311) comprises an IPv6
destination address (DEST IPv6: TEREDO (64.5.6:1025) and an IPv6 source addresses (SOURCE IPv6:
FEDC: BA98:7654:3210). See col. 12, lines 32-40.



See also col. 11, lines 3-6 which explains that “all IPv6 messages are represented with
intermittent dashed/dotted lines, as contrasted with IPv4 message which are represented with solid lines.”

See also Figures 2 and 4 which also discloses the packet comprising an IPv6 destination address
and IPv6 source address. See col. 1, lines 10-19, col. 14, lines 58 – col. 15, line 3

**said device for use in a home gateway adapted to connect a user terminal to said IPv6
domain, comprising:**

Huitema discloses the device for use in a home gateway (relay 202) adapted to connect to a user
terminal (IPv4 client 101) or (IPv6 client 201) to said IPv6 domain (IPv6 Internet). The Examiner notes
that the relay 202 is adapted to connect the IPv4 client 101 on a private network to the IPv6 client 201 on
the IPv6 network.

The relay 202 connects IPv4 client 101 to the IPv6 Internet because relay 202 sits between the
IPv6 and IPv4 Internet and IPv4 client resides in a private network of the IPv4 Internet. See col. 9, lines
45-54 and Fig. 3A.

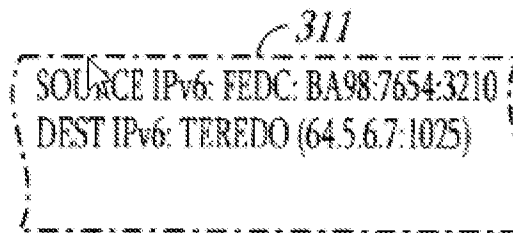
**means for identifying an IPv6 destination address constructed by concatenating an IPv6
prefix, an IPv4 destination address, and a destination port number;**

Huitema discloses the relay 202 identifying in response 311 an IPv6 destination address (DEST IPv6) constructed by concatenating an IPv6 prefix (TEREDO service prefix), an IPv4 destination addresses (64.5.6.7) and a destination port number (1025). See Figure 3A and 7

Huitema explains that IPv6 addresses (response 311) have “data structure 700” shown in Figure 7.



FIG. 7



With respect to IPv6 prefix, Huitema discloses Service field 701 (“PREF”) shows a TEREDO service prefix identifies that this IPv6 address corresponds to an IPv4 client. See col. 8, lines 39-40. Service field 702 shows “[a] service IPv4 address field 702 identifies the IPv4 address of the server 103. For example, the ‘102:304’ in the example IPv6 address is the hexadecimal notation of the address of the server 103 (namely, 1.2.3.4).” See col. 8, lines 42-46.

With respect to IPv4 destination address, Huitema discloses IP address 64.5.6.7 is the IPv4 address of IPv4 client 101. Huitema explains that IPv4 client 101 has “the globally unique IPv4 network address 64.5.6.7”. See col. 7, lines 51-53.

See also Figure 7 and col. 8, lines 51-55 which discloses “client IPv4 address field 705 identifies the IPv4 address of the client 101.”

With respect to destination port number, as shown above in Figure 3A and message 311, Huitema discloses the IPv6 destination address includes port number 1025 concatenated with IPv4 destination address 64.5.6.7. Specifically, port number 1025 is the UDP port number of IPv4 client 101. See col. 8, line 6-10. See also col. 8, lines 46-51

As shown in both Fig. 7 and Figure 3A the IPv6 destination address is constructed by concatenating an IPv6 prefix, an IPv4 destination address and a destination port number.

In addition, Huitema discloses that the relay identifies the IPv6 destination address by detecting the presence of the IPv6 prefix in that address. See col. 12, lines 41, 46-51.

With respect to “means for” and as explained above in the Claim Construction section, the corresponding structure is “a receiver device of the home gateway having a processor and memory”.

As explained above, Huitema teaches that relay (home gateway) has structure of Figure 8’s computer 820, which is a receiver device of the home gateway having a processor and memory.

means for regularizing, if necessary, at least one of the IPv6 source address and the IPv6 destination address of the data packet by replacing said at least one IPv6 address,

With respect to “means for” and as explained above in the Claim Construction section, the corresponding structure is “a device of the home gateway having a processor and memory, wherein the memory stores a table for translating between addresses”.

As explained above, relay 202 is the device of the home gateway having a processor and memory. Additionally, as explained above for dependent claim 5, Huitema discloses that relay 202 has a table for translating between addresses (i.e., converting between IPv6 and IPv4 addresses, and vice versa).

Huitema discloses if necessary (if a TEREDO prefix is detected), regularizes (modifies from IPv6 to IPv4) at least one of the IPv6 source address (::FEDC:BA98:7654:3210) and the IPv6 destination address (TEREDO (64.5.6.7:1025)) of the data packet (response 311) by replacing said at least one IPv6 address (replacing FEDC:BA98:7654:3210 and TEREDO (64.5.6.7:1025) respectively with IPv4 source address 5.6.7.8:3545 and IPv4 destination address 1.2.3.4:3544). See col., 12, lines 34-67. As shown in

Fig. 3A response 311 has an IPv6 format and response 311' has an IPv4 format. See col. 11, lines 3-6.

The relay regularizes IPv6 format response 311 into IPv4 format response 311'.

Thus, as explained above, Huitema discloses the relay 202 regularizes the **IPv6 source address** (FEDC:BA98:7654:3210) in the header of IPv6 response 311 by replacing the IPv6 source address with a corresponding IPv4 source address (5.6.7.8:3545) in the header of the IPv4 response 311' and moving the IPv6 source address (FEDC:BA98:7654:3210) from the header to the payload in IPv4 response 311'. See col. 12, lines 33-67.

Huitema discloses the relay regularizes the **IPv6 destination address** (TEREDO (64.5.6.7:1025)) by replacing the IPv6 destination address in the header of IPv6 response 311 with a corresponding IPv4 destination (1.2.3.4:3544) in the header of IPv4 response 311' and moving the IPv6 destination address (TEREDO (64.5.6.7:1025)) from the header to the payload in IPv4 response 311'. See col. 12, lines 33-67. See also Figure 6, col. 9, lines 58-64 and col. 11, lines 27-29.

the replacement process belonging to a group comprising replacement of a constructed address with a native address and replacement of a native address with a constructed address,

Huitema discloses the replacement process (replacing the IPv6 addresses in the header of IPv6 response 311 with corresponding IPv4 addresses to form IPv4 response 311') belonging to a group comprising replacement of a constructed address (TEREDO (64.5.6.7:1025)) with a native address (1.2.3.4:3544) and replacement of a native address (FEDC:BA98:7654:3210) with a constructed address (5.6.7.8:3545), as claimed.

As explained above, in regularizing IPv6 response 311 into IPv4 response 311', Huitema's relay 202 replaces IPv6 destination address TEREDO (64.5.6.7:1025) with corresponding IPv4 destination address 1.2.3.4:3544. See 12:31-67. Here, TEREDO (64.5.6.7:1025) is a constructed address for IPv4 client 101, because IPv4 client 101 does not have a native IPv6 address. Instead, "client 101 ... constructs its own IPv6 address" from the IPv6 TEREDO service prefix 111a and client 101's own IPv4 address 111B of 64.5.6.7:1025 provided in response 111 received from server 103. See 8:8-26, 35-60, FIGS. 1, 7.

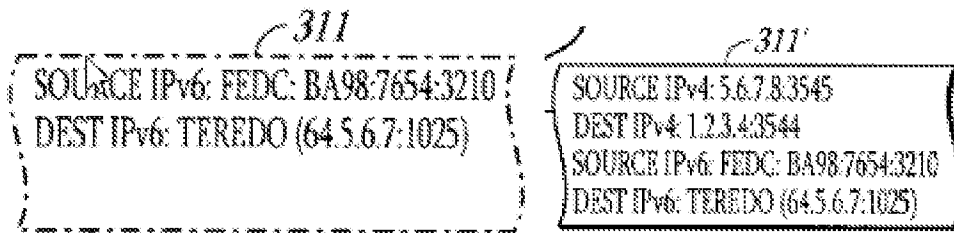
And IPv4 destination address 1.2.3.4:3544 that replaces the IPv6 address is a native address because it is native to IPv4 Internet on which IPv4 client 101 resides.

See also Figure 2, col. 11, lines 29-57 and Fig. 4, col. 14, lines 58-67, col. 15, line 66 – col. 16, line 12.

and modifying the data packet using the regularized address; and

Huitema's relay 202 modifies the data packet (converts IPv6 response 311 into IPv4 response 311') using the regularized address (using IPv4 destination address 1.2.3.4:3544 and/or IPv4 source address 5.6.7.8:3544 in the header of response 311').

As explained above, Huitema's relay 202 modifies IPv6 response 311 (see below message 311) into IPv4 response 311' (see below message 311') by replacing IPv6 destination address TEREDO (6.4.5.6.7:1025) and IPv6 source address FEDC:BA98:7654:3210 in the header of response 311 respectively with IPv4 destination address 1.2.3.4:3544 and IPv4 source address 5.6.7.8:3544 in the header of response 311. See 12:33-13:3.



See also Figure 2, col. 11, lines 29-57 and Fig. 4, col. 14, lines 58-67, col. 15, line 66 – col. 16, line 12.

means for routing the IPv6 data packet to its destination.

With respect to “means for” and as explained above in the Claim Construction section, the corresponding structure is “telecommunication means of the device of the home gateway”.

As discussed above, Huitema's relay 202 discloses the claimed home gateway. Huitema teaches that relay 202 has structure of Figure 8's computer 820, which includes telecommunications means. The computer 820 corresponds to the device of the home gateway, and at least serial port interface 846 and

Art Unit: 3992

modem 854 for communicating on wide area network 852 and/or network interface 853 for communicating on local area network 851 correspond to the telecommunications means. See 17:13-48, FIG. 8.

Huitema's relay 202 routes the IPv6 data packet (response 311, 311' and 311'') to its destination (IPv4 client 101). See Fig. 3A.

See also Figure 2, col. 11, lines 29-57 and Fig. 4, col. 14, lines 58-67, col. 15, line 66 – col. 16, line 12.

Regarding claim 11:

A home gateway, comprising: a processor; and a memory device coupled to the processor, wherein the memory device stores instructions that when executed by the processor cause the home gateway to:

Huitema teaches that relay (home gateway) has structure of Figure 8's computer 820, which is a receiver device of the home gateway having a processor and memory.

receive an IPv6 data packet in an IPv6 domain connected to an IPv4 domain,

Huitema discloses a device (for receiving in IPv6 data packet (response 311) in an IPv6 domain (IPv6 Internet) connected to an IPv4 domain (IPv4 Internet).

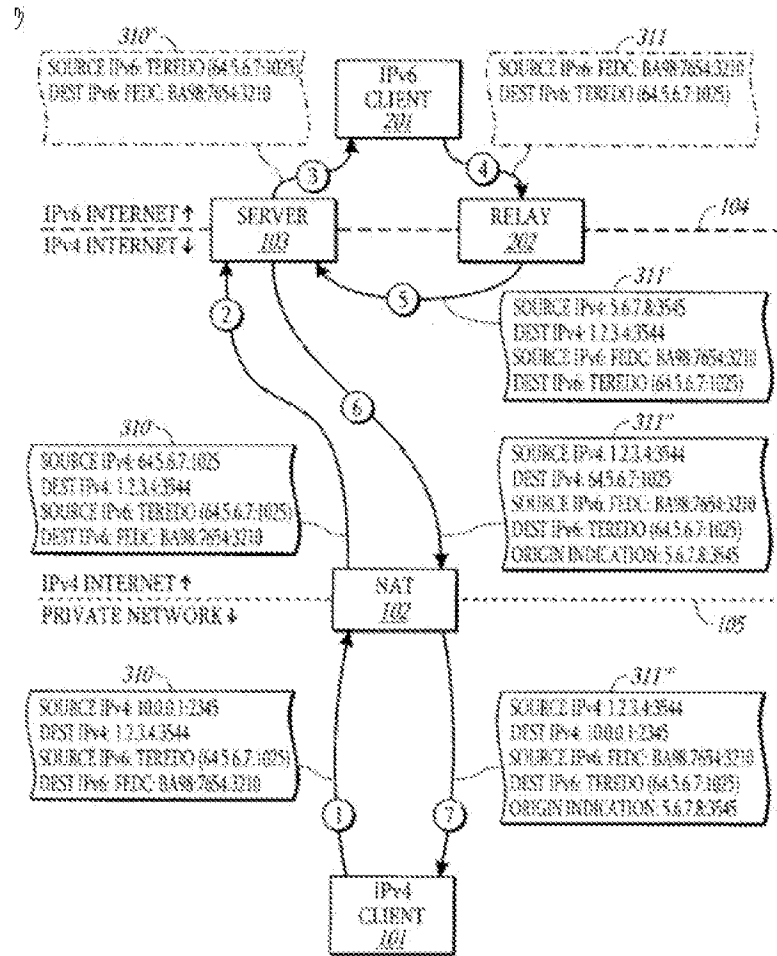
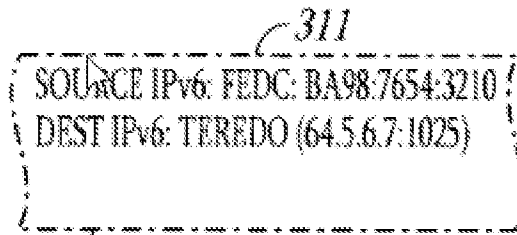


FIG. 3A

As shown above, in Figure 3A (see also 3B), Huitema shows a message flow in which IPv4client 101 communicates with IPv6 client 201. See col. 11, lines 62 – col. 14, line 52. IPv6 client resides in an IPv6 domain (IPv6 Internet), IPv4 client 101 resides in a private network of an IPv4 domain (IPv4 Internet) and relay 202 and server 103 connect to the IPv6 domain to the IPv4 domain. See col. 3, lines 3- col. 5, line 25.

wherein said packet comprises an IPv6 destination address and an IPv6 source address,

As shown in Figure 3A (see also below message 311), the packet (311) comprises an IPv6 destination address (DEST IPv6: TEREDO (64.5.6:1025) and an IPv6 source addresses (SOURCE IPv6: FEDC: BA98:7654:3210). See col. 12, lines 32-40.



See also col. 11, lines 3-6 which explains that “all IPv6 messages are represented with intermittent dashed/dotted lines, as contrasted with IPv4 message which are represented with solid lines.”

See also Figures 2 and 4 which also discloses the packet comprising an IPv6 destination address and IPv6 source address. See col. 1, lines 10-19, col. 14, lines 58 – col. 15, line 3

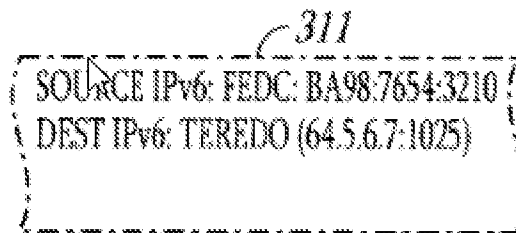
identify an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address and a destination port number;

Huitema discloses the relay 202 identifying in response 311 an IPv6 destination address (DEST IPv6) constructed by concatenating an IPv6 prefix (TEREDO service prefix), an IPv4 destination addresses (64.5.6.7) and a destination port number (1025). See Figure 3A and 7

Huitema explains that IPv6 addresses (response 311) have “data structure 700” shown in Figure 7.



FIG. 7



With respect to IPv6 prefix, Huitema discloses Service field 701 (“PREF”) shows a TEREDO service prefix identifies that this IPv6 address corresponds to an IPv4 client. See col. 8, lines 39-40. Service field 702 shows “[a] service IPv4 address field 702 identifies the IPv4 address of the server 103. For example, the ‘102:304’ in the example IPv6 address is the hexadecimal notation of the address of the server 103 (namely, 1.2.3.4).” See col. 8, lines 42-46.

With respect to IPv4 destination address, Huitema discloses IP address 64.5.6.7 is the IPv4 address of IPv4 client 101. Huitema explains that IPv4 client 101 has “the globally unique IPv4 network address 64.5.6.7”. See col. 7, lines 51-53.

See also Figure 7 and col. 8, lines 51-55 which discloses “client IPv4 address field 705 identifies the IPv4 address of the client 101.”

With respect to destination port number, as shown above in Figure 3A and message 311, Huitema discloses the IPv6 destination address includes port number 1025 concatenated with IPv4 destination address 64.5.6.7. Specifically, port number 1025 is the UDP port number of IPv4 client 101. See col. 8, line 6-10. See also col. 8, lines 46-51

As shown in both Fig. 7 and Figure 3A the IPv6 destination address is constructed by concatenating an IPv6 prefix, an IPv4 destination address and a destination port number.

In addition, Huitema discloses that the relay identifies the IPv6 destination address by detecting the presence of the IPv6 prefix in that address. See col. 12, lines 41, 46-51.

With respect to “means for” and as explained above in the Claim Construction section, the corresponding structure is “a receiver device of the home gateway having a processor and memory”.

As explained above, Huitema teaches that relay (home gateway) has structure of Figure 8’s computer 820, which is a receiver device of the home gateway having a processor and memory.

regularize at least one of the IPv6 source address and the IPv6 destination address of the data packet by replacing said at least one IPv6 address,

With respect to “means for” and as explained above in the Claim Construction section, the corresponding structure is “a device of the home gateway having a processor and memory, wherein the memory stores a table for translating between addresses”.

As explained above, relay 202 is the device of the home gateway having a processor and memory. Additionally, as explained above for dependent claim 5, Huitema discloses that relay 202 has a table for translating between addresses (i.e., converting between IPv6 and IPv4 addresses, and vice versa).

Huitema discloses if necessary (if a TEREDO prefix is detected), regularizes (modifies from IPv6 to IPv4) at least one of the IPv6 source address (::FEDC:BA98:7654:3210) and the IPv6 destination address (TEREDO (64.5.6.7:1025)) of the data packet (response 311) by replacing said at least one IPv6 address (replacing FEDC:BA98:7654:3210 and TEREDO (64.5.6.7:1025) respectively with IPv4 source address 5.6.7.8:3545 and IPv4 destination address 1.2.3.4:3544). See col., 12, lines 34-67. As shown in Fig. 3A response 311 has an IPv6 format and response 311' has an IPv4 format. See col. 11, lines 3-6. The relay regularizes IPv6 format response 311 into IPv4 format response 311'.

Thus, as explained above, Huitema discloses the relay 202 regularizes the **IPv6 source address** (FEDC:BA98:7654:3210) in the header of IPv6 response 311 by replacing the IPv6 source address with a corresponding IPv4 source address (5.6.7.8:3545) in the header of the IPv4 response 311' and moving the IPv6 source address (FEDC:BA98:7654:3210) from the header to the payload in IPv4 response 311'. See col. 12, lines 33-67.

Huitema discloses the relay regularizes the **IPv6 destination address** (TEREDO (64.5.6.7:1025)) by replacing the IPv6 destination address in the header of IPv6 response 311 with a corresponding IPv4 destination (1.2.3.4:3544) in the header of IPv4 response 311' and moving the IPv6 destination address (TEREDO (64.5..6.7:1025)) from the header to the payload in IPv4 response 311'. See col. 12, lines 33-67. See also Figure 6, col. 9, lines 58-64 and col. 11, lines 27-29.

the replacement process belonging to a group comprising replacement of a constructed address with a native address and replacement of a native address with a constructed address,

Huitema discloses the replacement process (replacing the IPv6 addresses in the header of IPv6 response 311 with corresponding IPv4 addresses to form IPv4 response 311') belonging to a group comprising replacement of a constructed address (TEREDO (64.5.6.7:1025)) with a native address (1.2.3.4:3544) and replacement of a native address (FEDC:BA98:7654:3210) with a constructed address (5.6.7.8:3545), as claimed.

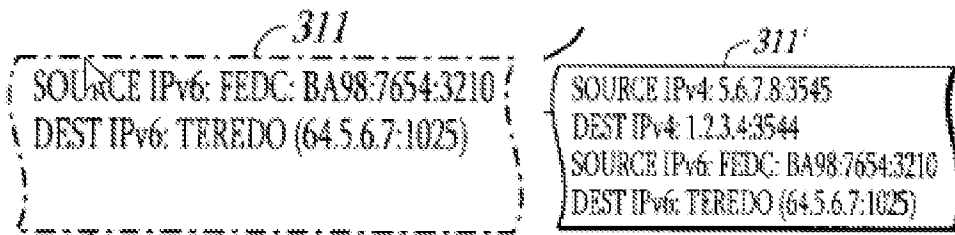
As explained above, in regularizing IPv6 response 311 into IPv4 response 311', Huitema's relay 202 replaces IPv6 destination address TEREDO (6.4.5.6.7:1025) with corresponding IPv4 destination address 1.2.3.4:3544. See 12:31-67. Here, TEREDO (6.4.5.6.7:1025) is a constructed address for IPv4 client 101, because IPv4 client 101 does not have a native IPv6 address. Instead, "client 101 ... constructs its own IPv6 address" from the IPv6 TEREDO service prefix 111a and client 101's own IPv4 address 111B of 64.5.6.7:1025 provided in response 111 received from server 103. See 8:8-26, 35-60, FIGS. 1, 7. And IPv4 destination address 1.2.3.4:3544 that replaces the IPv6 address is a native address because it is native to IPv4 Internet on which IPv4 client 101 resides.

See also Figure 2, col. 11, lines 29-57 and Fig. 4, col. 14, lines 58-67, col. 15, line 66 – col. 16, line 12.

and modifying the data packet using the regularized address; and

Huitema's relay 202 modifies the data packet (converts IPv6 response 311 into IPv4 response 311') using the regularized address (using IPv4 destination address 1.2.3.4:3544 and/or IPv4 source address 5.6.7.8:3544 in the header of response 311').

As explained above, Huitema's relay 202 modifies IPv6 response 311 (see below message 311) into IPv4 response 311' (see below message 311') by replacing IPv6 destination address TEREDO (6.4.5.6.7:1025) and IPv6 source address FEDC:BA98:7654:3210 in the header of response 311 respectively with IPv4 destination address 1.2.3.4:3544 and IPv4 source address 5.6.7.8:3544 in the header of response 311. See 12:33-13:3.



See also Figure 2, col. 11, lines 29-57 and Fig. 4, col. 14, lines 58-67, col. 15, line 66 – col. 16, line 12.

route the IPv6 data packet to its destination.

With respect to “means for” and as explained above in the Claim Construction section, the corresponding structure is “telecommunication means of the device of the home gateway”.

As discussed above, Huitema’s relay 202 discloses the claimed home gateway. Huitema teaches that relay 202 has structure of Figure 8’s computer 820, which includes telecommunications means. The computer 820 corresponds to the device of the home gateway, and at least serial port interface 846 and modem 854 for communicating on wide area network 852 and/or network interface 853 for communicating on local area network 851 correspond to the telecommunications means. See 17:13-48, FIG. 8.

Huitema’s relay 202 routes the IPv6 data packet (response 311, 311’ and 311”) to its destination (IPv4 client 101). See Fig. 3A.

See also Figure 2, col. 11, lines 29-57 and Fig. 4, col. 14, lines 58-67, col. 15, line 66 – col. 16, line 12.

Claim Rejections - 35 USC § 103

7. In the event the determination of the status of the application as subject to AIA 35 U.S.C. 102 and 103 (or as subject to pre-AIA 35 U.S.C. 102 and 103) is incorrect, any correction of the statutory basis (i.e., changing from AIA to pre-AIA) for the rejection will not be considered a new ground of rejection if the prior art relied upon, and the rationale supporting the rejection, would be the same under either status.

The following is a quotation of pre-AIA 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

The factual inquiries for establishing a background for determining obviousness under pre-AIA 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
 2. Ascertaining the differences between the prior art and the claims at issue.
 3. Resolving the level of ordinary skill in the pertinent art.
 4. Considering objective evidence present in the application indicating obviousness or nonobviousness.
8. Claims 5, 6, 12 and 13 is/are rejected under pre-AIA 35 U.S.C. 103(a) as being unpatentable over Huitema in view of Kong et al., “Design and Implementation of IPv6-IPv4 Protocol Translation System Using Dynamic IP Address” (hereinafter Kong)

Regarding claim 5:

The method according to claim 1, wherein when the data packet comes from the IPv4 domain and goes to said user terminal,

The Examiner notes that since this limitation refers back to “the step of regularizing the destination” which is part of an optional limitation then this step need not be performed since the step of regularizing occurs “if necessary”.

Nonetheless, the Examiner notes that Huitema discloses that relay 202 receives the IPv6 response 311, regularizes it into IPv4 response 311’, and routes response 311’ through server 102 (as response 311’) and NAT 102 (as response 311”) to IPv4 client (user terminal) across IPv4 Internet (IPv4 domain). See col. 11, line 62 – col. 14, line 42; fig. 3A.

the step of regularizing the destination address comprises the following steps: searching an address translation table of the gateway for an entry associating with said constructed destination address a native destination address;

As explained above, relay 202 replaces the constructed destination address TEREDO (6.4.5.6.7:1025) in the header of IPv4 response 311 with the native destination address 1.2.3.4:3544 to generate IPv4 response 311'. See 12:37-67.

The Examiner notes that while Huitema does not expressly disclose that relay 202 searches a table associating the IPv6 destination address TEREDO (6.4.5.6.7:1025) with the IPv4 destination address 1.2.3.4:3544, such a table would have been necessary for relay 202 to know the IPv4 destination address from the corresponding IPv6 destination address.

That is, the mechanism that relay 202 uses to replace TEREDO (6.4.5.6.7:1025) (constructed destination address) with the IPv4 destination address 1.2.3.4:3544 is or discloses searching an address translation table. Otherwise, Huitema's system would not work as described to convert response 311 into response 311' because relay 202 could not replace TEREDO (6.4.5.6.7:1025) with 1.2.3.4:3544 without a table associating the two addresses.

The Examiner further notes, Huitema discloses that NAT 102 maintains a "mapping" between IPv4 client 101's public and private addresses to accomplish translation. See 11: 46-50, 12:13-23, 13:19-30, 14: 15- 25, 15:35-65, 16:13-28. These mappings maintained by NAT 102 disclose an address translation table. And, like relay 202 replaces addresses to modify response 311 to response 311', NAT 102 replaces IPv4 client 101's public IPv4 address 64.5.6.7:1025 in response 311'' with client 101's corresponding private IPv4 address 10.0.0.0.1:2345 to generate response 311'. See 13:19-30.

Therefore, it would have been obvious to a person of ordinary skill in the art that the relay 202 is configured to "searching an address translation table of the gateway for an entry associating with said constructed destination address a native destination address". It would have been apparent to a POSITA that relay 102 also needs to maintain a similar table/mapping to replace constructed IPv6 addresses with

IPv4 addresses to convert IPv6 response 211, 311, and 410 in Figures 2, 3A, and 4 respectively into corresponding IPv4 responses 211', 311', and 410' as Huitema teaches.

Nonetheless, Kong explains that its 64 Translator maintains multiple "mapping entries" used for address translation. See, e.g., 499, 501. The mapping entries maintained by the 64 Translator disclose the claimed address translation table. Id.

As Kong explains, when translating the [ConnectIPv4Server]₆ packet to the [ConnectIPv4Server]₄ packet, "[t]he 64 Translator ... translates the [ConnectIPv4Server]₆ into the [ConnectIPv4Server]₄ using the mapping entry." See 501; see also id., 500 (Table 3 explaining that functions of the 64 Translator include "creation of the mapping entry and a translation of the packet from IPv.X to IPvY using that mapping entry"). By using a stored mapping entry to translate the [ConnectIPv4Server]₆ packet into the [ConnectIPv4Server]₄ packet, the 64 Translator searches an address translation table (plurality of mapping entries) for an entry (the mapping entry needed for that particular translation).

Therefore, it would have been obvious to a person of ordinary skill to search an address translation table of the gateway as taught by Huitema's NAT or as taught by Kong.

This same teaching, as set forth above, is set forth in Kong with respect to translating the [ConnectIPv4Server]₆ packet to the [ConnectIPv4Server]₄ packet. Thus, the prior art included each element claimed, with the only different being the lack of actual combination of the elements into the prior art. The Examiner notes that as explained above, Huitema already discloses that it was known to use a NAT ("Network Address Translator") and thus one of ordinary skill in the art would have understood that the relay 202 would also be configured to search an address translation table since it is configured to associate the constructed destination address with a native destination address as explained above. This teaching would have yielded a predictable result to a person of ordinary skill in the art.

One of ordinary skill in the art could have combined the elements as claimed by known methods (the NAT of Huitema or the 64 Translator of Kong). As explained above, one of ordinary skill in the art would have recognized that the results of the combination were predictable since both prior art references both disclose of using translation and mapping.

if⁴ an entry is found, replacing the constructed destination address of the data packet with the native destination address from that entry.

As explained above, Huitema alone or in view of Kong discloses if an entry is found, replacing the constructed destination address (DEST IPv6: TEREDO (6.4.5.6.7:102) of the data packet (response 311) with the native destination address (DEST IPv4: 1.2.3.4:3544) from that entry.

See also Kong as explained above which discloses of translating the [ConnectIPv4Server]₆ packet to the [ConnectIPv4Server]₄ packet.

Regarding claim 6:

The method according to claim 5, further comprising a step of verifying that the destination port number of the data packet⁵ and the destination port number contained in the constructed destination address match.

The Examiner notes that since this claim is dependent upon claim 5 which as set forth above is optional, then claim 6 is likewise optional for the same reasons.

Nonetheless, Figure 3A of Huitema shows that the destination port number (1025) of the data packet (response 311) and the destination port number (1025) contained in the constructed destination address (TEREDO (6.4.5.6.7:102)) match. See col. 12, lines 37-col. 13, line 18.

Regarding claim 12:

The home gateway according to claim 11, wherein the memory device stores further instructions that when executed by the processor that causes the home gateway to: searching an address translation table of the gateway for an entry associating with said constructed destination address a native destination address;

⁴ See MPEP 2111.04(II)

⁵ The Examiner finds that with respect “destination port number” claim 1 only recites “destination port number” once in the limitation reciting identifying “an IPv6 destination address constructed by a destination port number.” Claim 6 recites two “destination port numbers” the first being “the destination port number of the data packet” and the second being “the destination port number contained in the constructed destination”.

As explained above, relay 202 replaces the constructed destination address TEREDO (6.4.5.6.7:1025) in the header of IPv4 response 311 with the native destination address 1.2.3.4:3544 to generate IPv4 response 311'. See 12:37-67.

The Examiner notes that while Huitema does not expressly disclose that relay 202 searches a table associating the IPv6 destination address TEREDO (6.4.5.6.7:1025) with the IPv4 destination address 1.2.3.4:3544, such a table would have been necessary for relay 202 to know the IPv4 destination address from the corresponding IPv6 destination address.

That is, the mechanism that relay 202 uses to replace TEREDO (6.4.5.6.7:1025) (constructed destination address) with the IPv4 destination address 1.2.3.4:3544 is or discloses searching an address translation table. Otherwise, Huitema's system would not work as described to convert response 311 into response 311' because relay 202 could not replace TEREDO (6.4.5.6.7:1025) with 1.2.3.4:3544 without a table associating the two addresses.

The Examiner further notes, Huitema discloses that NAT 102 maintains a "mapping" between IPv4 client 101's public and private addresses to accomplish translation. See 11: 46-50, 12:13-23, 13:19-30, 14: 15- 25, 15:35-65, 16:13-28. These mappings maintained by NAT 102 disclose an address translation table. And, like relay 202 replaces addresses to modify response 311 to response 311', NAT 102 replaces IPv4 client 101's public IPv4 address 64.5.6.7:1025 in response 311'' with client 101's corresponding private IPv4 address 10.0.0.0.1:2345 to generate response 311'. See 13:19-30.

Therefore, it would have been obvious to a person of ordinary skill in the art that the relay 202 is configured to "searching an address translation table of the gateway for an entry associating with said constructed destination address a native destination address". It would have been apparent to a POSITA that relay 102 also needs to maintain a similar table/mapping to replace constructed IPv6 addresses with IPv4 addresses to convert IPv6 response 211, 311, and 410 in Figures 2, 3A, and 4 respectively into corresponding IPv4 responses 211', 311', and 410' as Huitema teaches.

Nonetheless, Kong explains that its 64 Translator maintains multiple “mapping entries” used for address translation. See, e.g., 499, 501. The mapping entries maintained by the 64 Translator disclose the claimed address translation table. Id.

As Kong explains, when translating the [ConnectIPv4Server]₆ packet to the [ConnectIPv4Server]₄ packet, “[t]he 64 Translator ... translates the [ConnectIPv4Server]₆ into the [ConnectIPv4Server]₄ using the mapping entry.” See 501; see also id., 500 (Table 3 explaining that functions of the 64 Translator include “creation of the mapping entry and a translation of the packet from IPv.X to IPvY using that mapping entry”). By using a stored mapping entry to translate the [ConnectIPv4Server]₆ packet into the [ConnectIPv4Server]₄ packet, the 64 Translator searches an address translation table (plurality of mapping entries) for an entry (the mapping entry needed for that particular translation).

Therefore, it would have been obvious to a person of ordinary skill to search an address translation table of the gateway as taught by Huitema’s NAT or as taught by Kong.

This same teaching, as set forth above, is set forth in Kong with respect to translating the [ConnectIPv4Server]₆ packet to the [ConnectIPv4Server]₄ packet. Thus, the prior art included each element claimed, with the only different being the lack of actual combination of the elements into the prior art. The Examiner notes that as explained above, Huitema already discloses that it was known to use a NAT (“Network Address Translator”) and thus one of ordinary skill in the art would have understood that the relay 202 would also be configured to search an address translation table since it is configured to associate the constructed destination address with a native destination address as explained above. This teaching would have yielded a predictable result to a person of ordinary skill in the art.

One of ordinary skill in the art could have combined the elements as claimed by known methods (the NAT of Huitema or the 64 Translator of Kong). As explained above, one of ordinary skill in the art would have recognized that the results of the combination were predictable since both prior art references both disclose of using translation and mapping.

if an entry is found, replacing the constructed destination address of the data packet with the native destination address from the entry.

As explained above, Huitema alone or in view of Kong discloses if an entry is found, replacing the constructed destination address (DEST IPv6: TEREDO (6.4.5.6.7:102) of the data packet (response 311) with the native destination address (DEST IPv4: 1.2.3.4:3544) from that entry.

See also Kong as explained above which discloses of translating the [ConnectIPv4Server]₆ packet to the [ConnectIPv4Server]₄ packet.

Regarding claim 13:

The home gateway according to claim 12, wherein the wherein the memory device stores further instructions that when executed by the processor causes the home gateway to verify that the destination port number of the data packet and the destination port number contained in the constructed destination address match.

Figure 3A of Huitema shows that the destination port number (1025) of the data packet (response 311”) and the destination port number (1025) contained in the constructed destination address (TEREDO (6.4.5.6.7:102)) match. See col. 12, lines 37-col. 13, line 18.

9. Claims 1, 5-8 and 11-13 is/are rejected under pre-AIA 35 U.S.C. 103(a) as being unpatentable over Kong et al., “Design and Implementation of IPv6-IPv4 Protocol Translation System Using Dynamic IP Address” (hereinafter Kong) in view of Hinden, “Internet Protocol, Version 6 (IPv6) Specification” (Exhibit 1029).

Regarding claim 1:

A method for receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain,

Kong discloses a method (Connection Establishment Phase of Figure 2) for receiving an IPv6 data packet ([ConnectIPv4Server]₆) in an IPv6 domain (IPv6 Network) connected to an IPv4 domain (IPv4 Network). See below Figs. 1 and 2.

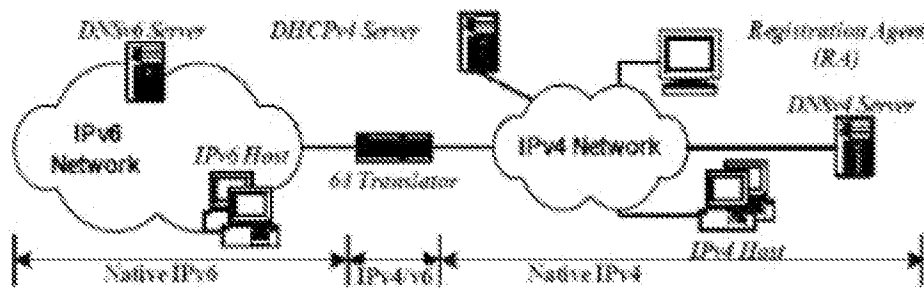


Fig. 1. The overall system architecture using 64TSD

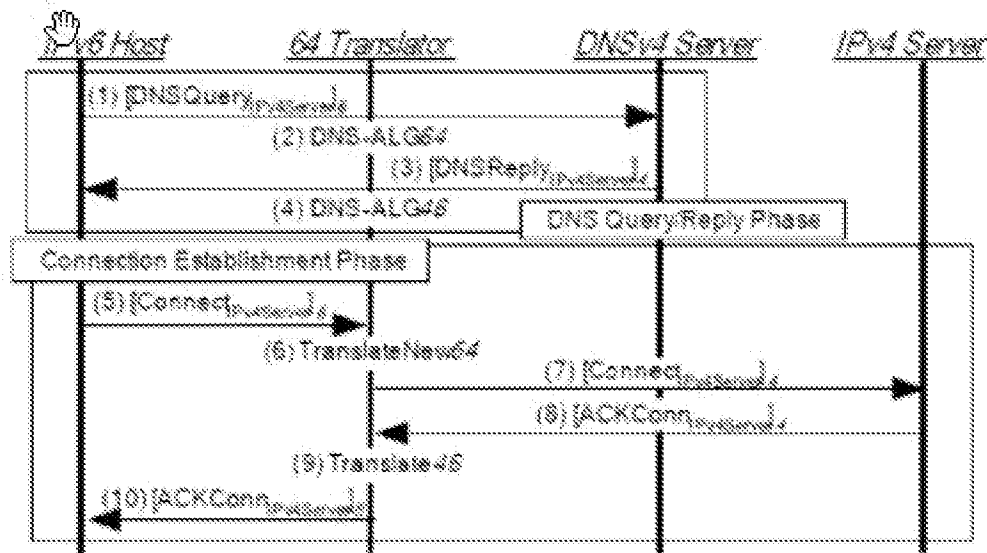


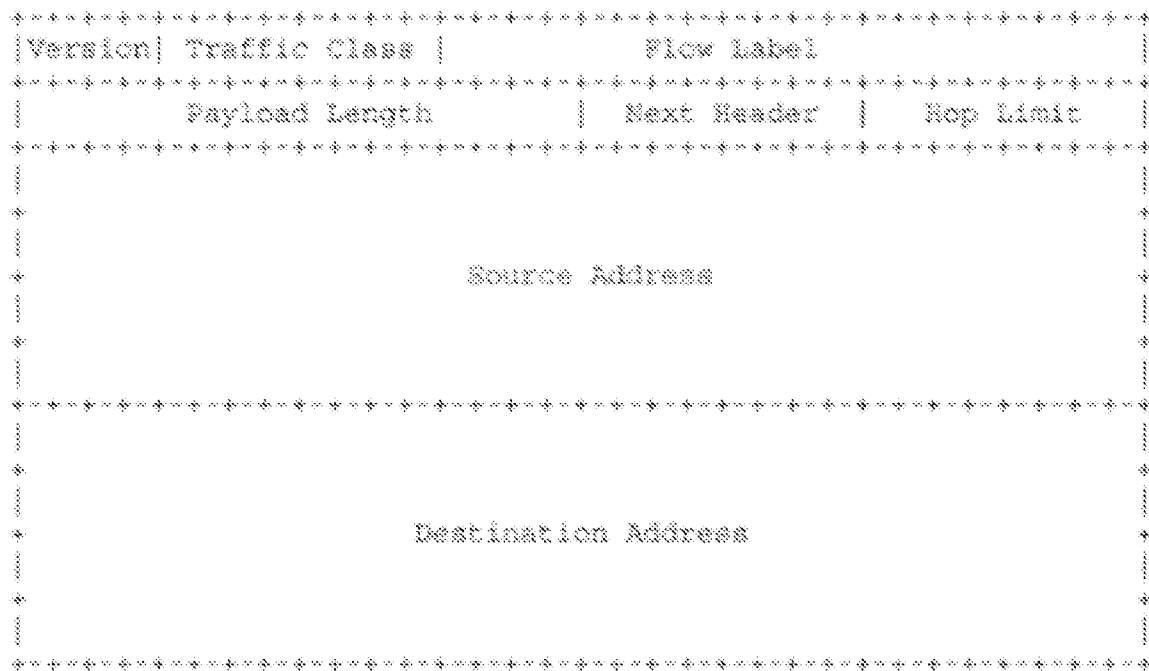
Fig. 2. The procedure of the outbound session

As shown above in Figure 1, an IPv6 host and server in an IPv6 domain (IPv6 network) connects to an IPv4 host and server in an IPv4 domain (IPv4 network) via a 64 Translator. See p 498 (“The overall system architecture using 64TSD consists of IPv6 Network, IPv4 Network and the 64 Translator”); see also pg. 499. The 64 Translator connects the IPv6 network to the IPv4 network. See also pages 500-501.

said packet comprising an IPv6 destination address and an IPv6 source address,

Kong does not specifically disclose the packet “comprising an IPv6 destination address and an IPv6 source address”, however because Kong’s ([ConnectIPv4Server]₆) packet is an IPv6 packet, it necessarily includes an IPv6 destination address and an IPv6 source address.

IPv6 Header Format



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

Kong discloses said method (Connection Establishment Phase of Figure 2) being executed in a home gateway (64 Translator) adapted to connect a user terminal (IPv4 server/host of Figure 1) to said IPv6 domain (IPv6 Network of Figure 1). See page 503

In Figure 1, Kong shows the IPv4 host as a desktop computer (see page 503), which is a user terminal. See p. 498. Additionally, Kong's 64 Translator (home gateway) is adapted to connect the IPv4 host (user terminal) to the IPv6 network (IPv6 domain). See pp, 496, 499- 501

identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address and a destination port number;

Kong discloses that the ([ConnectIPv4Server]₆) (which as explained above includes an IPv6 destination address) (IPv6 data packet) has an IPv6 destination address in the form shown above—i.e., constructed by concatenating an IPv6 prefix (dummy prefix), an IPv4 destination address and a destination port number (<IPv4 address, IPv4 Port>).

Figure 3 of Kong shows “[t]he procedure of the inbound session” in which the IPv4 host connects to the IPv6 network—i.e., communication in the opposition direction of Figure 2's outbound session discussed above. See 501-502. In step (9) of the outbound session, the “64 Translator translates the ([ConnectIPv4Server]₄) to ([ConnectIPv4Server]₆) using the IPv6 address that is created by the concatenation of dummy prefix and IPv4 address.” See 502

See also Table 1. Kong shows the functions of the 64 Translator include allocating a port number for each outbound session and “TCP/UDP port number translation.” See p 499

Kong further discloses identifying the IPv6 destination address in the ([ConnectIPv4Server]₆) packet. For example, Kong explains that “[t]he 64 Translator creates the mapping entry and translates the ([ConnectIPv4Server]₆) into the ([ConnectIPv4Server]₄) using the mapping entry.” See 501, FIG. 2. Kong explains that “[t]he mapping entry is a tuple of... where IPvXAddr and IPvXPort are IPvX address and port number of IPvX packet (X is 4 or 6).” See 499

if necessary, regularizing at least one of the IPv6 source address and the IPv6 destination address of the data packet by replacing said at least one IPv6 address,

The Examiner first determines that the entire “if necessary” limitation is optional and therefore is not required in this method. The limitation states “if necessary” and thus is not needed to be performed since it is contingent on whether “if necessary” is satisfied. It is noted that the claim does not specifically state when this step is needed to be performed.

Kong discloses the 64 Translator, if necessary (if communicating between IPv6 server and IPv4 host), regularizing (translating from IPv6 to IPv4) at least one of the IPv6 source address and the IPv6 destination address of the data packet (addresses in ([ConnectIPv4Server]₆) by replacing (replacing with address in ([ConnectIPv4Server]₄) said at least one IPv6 address. See Fig. 2

In step (6) of Figure 3, “[t]he 64 Translator creates the mapping entry and translates the ([ConnectIPv4Server]₆) into the ([ConnectIPv4Server]₄) using the mapping entry.” See 501. The ([ConnectIPv4Server]₄) packet resulting from the translation is an IPv4 packet because “[Connect_H]_X ... is a packet to connect to host H using IPvX” where X=4. See 500. In Table 1, Kong explains that the translation function of the 64 Translator involves, among other things, “translating a protocol header between IPv6 and IPv4.” See 499.

the replacement process belonging to a group comprising replacement of a constructed address with a native address and replacement of a native address with a constructed address,

Specifically, Kong discloses the replacement process belonging to a group comprising replacement of a constructed address (IPv6 address in ([ConnectIPv4Server]₆) With a native address (IPv4 address in ([ConnectIPv4Server]₄)) and replacement of a native address with a constructed address. See Figure 2.

In step (6) of Figure 2, Kong’s “64 Translator creates the mapping entry and translates the ([ConnectIPv4Server]₆) into the ([ConnectIPv4Server]₄) using the mapping entry,” translating the Connect packet from IPv6 to IPv4. See 501. In this process, the 64 Translator replaces the IPv6 destination address with a corresponding IPv4 destination address. See 501.

Here, the IPv6 destination address in the ([ConnectIPv4Server]₆) packet corresponds to the constructed address because, as Kong explains, an IPv6 address “is created by the concatenation of

dummy prefix and IPv4 address.” See 502. The IPv4 destination address in the ([ConnectIPv4Server]₄) packet, which replaces the IPv6 destination address in ([ConnectIPv4Server]₆) packet, corresponds to the claimed native address. This is because IPv4 addresses are native to the IPv4 network—the destination of the Connect packet.

and modifying the data packet using the regularized address; and

Kong’s 64 Translator modifies the ([ConnectIPv4Server]₆) packet into the ([ConnectIPv4Server]₄) packet by replacing the IPv6 source and destination addresses in the header of the ([ConnectIPv4Server]₆) packet with respective IPv4 source and destination addresses in the header of the resulting ([ConnectIPv4Server]₄) the packet. See 499, 501, FIG. 2. Here, the IPv4 destination address corresponds to the regularized address because it replaces the IPv6 destination address in the regularization (translation) process.

routing the IPv6 data packet to its destination.

Kong discloses routing the IPv6 data packet (the ([ConnectIPv4Server]₆) packet regularized to the ([ConnectIPv4Server]₄) packet) for its destination (IPv4 Server).

In the Figure 2 procedure, Kong explains that “IPv6 Host sends the ([ConnectIPv4Server]₆) through the 64 Translator to the IPv4 Server.” See 501. Specifically, the IPv6 host sends the ([ConnectIPv4Server]₆) packet to the 64 Translator, the 64 Translator “(6) translates the ([ConnectIPv4Server]₆) into the ([ConnectIPv4Server]₄) using the mapping entry,” and then the 64 Translator “sends the translated packet to IPv4 Server.” See 501.

Regarding claim 5:

The method according to claim 1, wherein when the data packet comes from the IPv4 domain and goes to said user terminal, the step of regularizing the destination address comprises the following steps:

The Examiner notes that as set forth above, the step of regularizing” is optional.

As explained above, Kong discloses regularizing the ([Connect_{IPv4Server}]₆) packet when the data packet comes from the IPv4 domain (post-translation by the 64 Translator) and goes to said user terminal (IPv4 server), as claimed. See Fig. 2. See also Figure 4.

searching an address translation table of the gateway for an entry associating with said constructed destination address a native destination address; if an entry is found, replacing the constructed destination address of the data packet with the native destination address from that entry.

Kong discloses searching an address translation table (mapping entries) of the gateway (64 Translator) for an entry (mapping entry) associating with said constructed destination address (IPv6 address in the ([Connect_{IPv4Server}]₆) packet) a native destination address (IPv4 address in the ([Connect_{IPv4Server}]₄) packet). Kong also discloses if an entry (mapping entry) is found, replacing the constructed destination address (IPv6 destination address) of the data packet ([Connect_{IPv4Server}]₆) packet with the native destination address (corresponding IPv4 address in the ([Connect_{IPv4Server}]₄) packet) from that entry.

Kong explains that the 64 Translator maintains multiple “mapping entries” used for address translation. See, e.g., 499, 501, 501.

As Kong explains, when translating the ([Connect_{IPv4Server}]₆) packet to the ([Connect_{IPv4Server}]₄) packet, “[t]he 64 Translator ... translates the ([Connect_{IPv4Server}]₆) into the ([Connect_{IPv4Server}]₄) using the mapping entry.” See 501; see also 500 (Table 3 explaining that functions of the 64 Translator include “creation of the mapping entry and a translation of the packet from IPvX to IPvY using that mapping entry”);

Regarding claim 6:

The method according to claim 5, further comprising a step of verifying that the destination port number of the data packet⁶ and the destination port number contained in the constructed destination address match.

As explained above, in Kong, “an IPv6 address ... is created by the concatenation of dummy prefix and IPv4 address,” and that IPv4 address contains a destination port number. See 501.

As a result, a POSITA would understand that the destination port number in the constructed IPv6 address of the ([ConnectIPv4Server]₆) packet matches the destination port number of the underlying IPv4 address. Kong discloses or suggests verifying that the two port numbers match by looking up the correct mapping entry for the translation using the address information in the packet. See, 501 (“it looks up the mapping entries with <IPv4 address, Pv5 Port>... If the entry is found, it translates...”).

Regarding claim 7:

A non-transitory computer program product that is at least one of downloadable from a communications network and stored on a computer-readable medium and executable by a microprocessor, comprising program code instructions for executing the method according to claim 1 of receiving a data packet in an IPv6 domain.

The Examiner incorporates by reference the rejection of claim 1 under Kong as set forth above for meeting “the method of claim 1 of receiving a data packet in an IPv6 domain.

Kong discloses a non-transitory computer program product (Linux algorithm based on NAT-PT code) that is at least one of downloadable from a communications network and stored on a computer readable medium and executable by a microprocessor (333 MHz Pentium processor of the 64 Translator), comprising program code instructions (Linux algorithm based on NAT-PT code) for executing the

⁶ The Examiner finds that with respect “destination port number” claim 1 only recites “destination port number” once in the limitation reciting identifying “an IPv6 destination address constructed by... **a destination port number.**” Claim 6 recites two “destination port numbers” the first being “the destination port number of the data packet” and the second being “the destination port number **contained in the constructed destination**”.

method according to claim 1 of receiving a data packet ([Connect_{IPv4Server}]₆) in an IPv6 domain (IPv6 network). See page 503

Regarding claim 8:

A device for receiving an IPv6 data packet in an IPv6 domain connected to an IPv4 domain,

Kong discloses a device for receiving an IPv6 data packet ([Connect_{IPv4Server}]₆) in an IPv6 domain (IPv6 Network) connected to an IPv4 domain (IPv4 Network). See below Figs. 1 and 2.

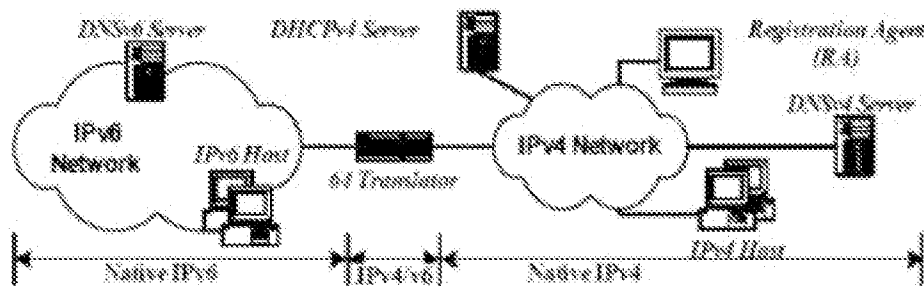


Fig. 1. The overall system architecture using 64TSD

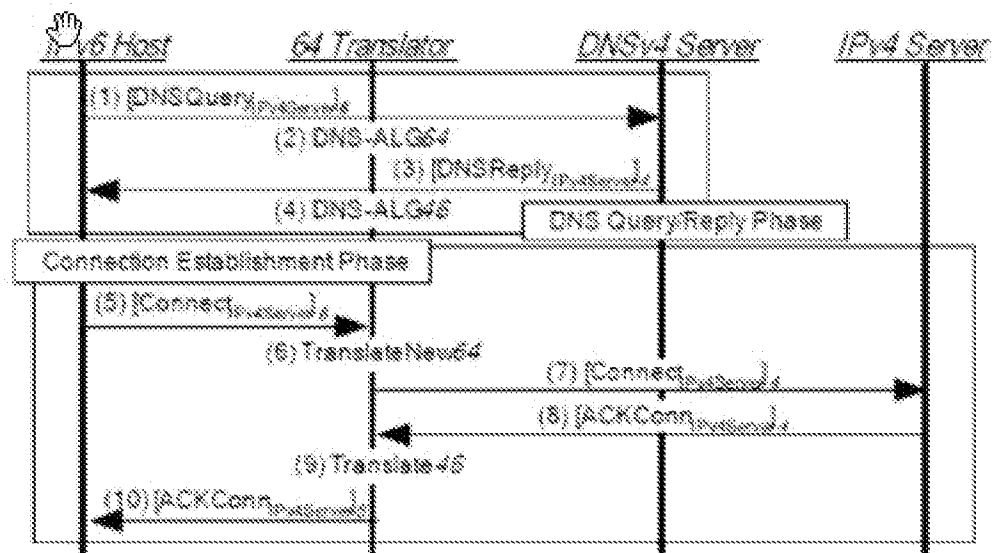


Fig. 2. The procedure of the outbound session

said packet comprising an IPv6 destination address and an IPv6 source address,

For example, as set forth in Deering (Exhibit 1029 – page 4), the required format for the header of an IPv6 packet which compiles with IP Version 6 protocol includes both in IPv6 destination address and an IPv6 source address. See Deering pages 3-4.

[illegible]

Juniper Exhibit 1064, page 56

destination address, communication would be impossible because ([ConnectIPv4Server]₆) packet could not propagate across the IPv6 network from the IPv6 Host to the 64 Translator. Therefore, including having a source and destination in the packet would have been predictable to a person of ordinary skill in the art since IP Version 6 protocol requires an IPv6 packet to include *inter alia* an IPv6 source address and an IPv6 destination address.

said device for use in a home gateway adapted to connect a user terminal to said IPv6 domain, comprising:

Kong discloses said device for use in a home gateway (64 Translator) adapted to connect a user terminal (IPv4 server/host of Figure 1) to said IPv6 domain (IPv6 Network of Figure 1). See page 503

In Figure 1, Kong shows the IPv4 host as a desktop computer (see page 503), which is a user terminal. See p. 498. Additionally, Kong's 64 Translator is adapted to connect the IPv4 host (user terminal) to the IPv6 network (IPv6 domain). See pp, 496, 499- 501

means for identifying an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address, and a destination port number;

Kong's 64 Translator corresponds to the claimed home gateway. Kong implemented the 64 Translator as a Pentium PC, which has a 333 MHz Pentium processor and memory. See 503.

Kong discloses that the ([ConnectIPv4Server]₆) (which as explained above includes an IPv6 destination address) (IPv6 data packet) has an IPv6 destination address in the form shown above—i.e., constructed by concatenating an IPv6 prefix (dummy prefix), an IPv4 destination address and a destination port number (<IPv4 address, IPv4 Port>).

Figure 3 of Kong shows “[t]he procedure of the inbound session” in which the IPv4 host connects to the IPv6 network—i.e., communication in the opposition direction of Figure 2's outbound session discussed above. See 501-502. In step (9) of the outbound session, the “64 Translator translates the ([ConnectIPv4Server]₄) to ([ConnectIPv4Server]₆) using the IPv6 address that is created by the concatenation of dummy prefix and IPv4 address.” See 502

See also Table 1. Kong shows the functions of the 64 Translator include allocating a port number for each outbound session and “TCP/UDP port number translation.” See p 499

Kong further discloses identifying the IPv6 destination address in the ([ConnectIPv4Server]₆) packet. For example, Kong explains that “[t]he 64 Translator creates the mapping entry and translates the ([ConnectIPv4Server]₆) into the ([ConnectIPv4Server]₄) using the mapping entry.” See 501, FIG. 2. Kong explains that “[t]he mapping entry is a tuple of... where IPvXAddr and IPvXPort are IPvX address and port number of IPvX packet (X is 4 or 6).” See 499

means for regularizing, if necessary, at least one of the IPv6 source address and the IPv6 destination address of the data packet by replacing said at least one IPv6 address,

Kong discloses the 64 Translator, if necessary (if communicating between IPv6 server and IPv4 host), regularizing (translating from IPv6 to IPv4) at least one of the IPv6 source address and the IPv6 destination address of the data packet (addresses in ([ConnectIPv4Server]₆) by replacing (replacing with address in ([ConnectIPv4Server]₄) said at least one Ipv6 address. See Fig. 2

In step (6) of Figure 3, “[t]he 64 Translator creates the mapping entry and translates the ([ConnectIPv4Server]₆) into the ([ConnectIPv4Server]₄) using the mapping entry.” See 501. The ([ConnectIPv4Server]₄) packet resulting from the translation is an IPv4 packet because “[ConnectH]_x ... is a packet to connect to host H using IPvX” where X=4. See 500. In Table 1, Kong explains that the translation function of the 64 Translator involves, among other things, “translating a protocol header between IPv6 and IPv4.” See 499.

the replacement process belonging to a group comprising replacement of a constructed address with a native address and replacement of a native address with a constructed address, and

Specifically, Kong discloses the replacement process belonging to a group comprising replacement of a constructed address (IPv6 address in ([ConnectIPv4Server]₆) With a native address (IPv4 address in ([ConnectIPv4Server]₄)) and replacement of a native address with a constructed address. See Figure 2.

In step (6) of Figure 2, Kong's "64 Translator creates the mapping entry and translates the ([ConnectIPv4Server]₆) into the ([ConnectIPv4Server]₄) using the mapping entry," translating the Connect packet from IPv6 to IPv4. See 501. In this process, the 64 Translator replaces the IPv6 destination address with a corresponding IPv4 destination address. See 501.

Here, the IPv6 destination address in the ([ConnectIPv4Server]₆) packet corresponds to the constructed address because, as Kong explains, an IPv6 address "is created by the concatenation of dummy prefix and IPv4 address." See 502. The IPv4 destination address in the ([ConnectIPv4Server]₄) packet, which replaces the IPv6 destination address in ([ConnectIPv4Server]₆) packet, corresponds to the claimed native address. This is because IPv4 addresses are native to the IPv4 network—the ultimate destination of the Connect packet.

modifying the data packet using the regularized address; and

As explained above the 64 Translator is the device of the home gateway having a 333 MHz Pentium processor and memory. Kong discloses that the 64 Translator maintains mapping entries (table) for translating between addresses (i.e., converting between IPv6 and IPv4 addresses, and vice versa).

Kong's 64 Translator modifies the ([ConnectIPv4Server]₆) packet into the ([ConnectIPv4Server]₄) packet by replacing the IPv6 source and destination addresses in the header of the ([ConnectIPv4Server]₆) packet with respective IPv4 source and destination addresses in the header of the resulting ([ConnectIPv4Server]₄) the packet. See 499, 501, FIG. 2. Here, the IPv4 destination address corresponds to the regularized address because it replaces the IPv6 destination address in the regularization (translation) process.

means for routing the IPv6 data packet to its destination.

Kong teaches that "the 64 Translator is implemented on a 333MHz Pentium PC" and that 64 Translator PC communicates with the IPv6 and IPv4 Host PCs via "100Mbps Ethernet," which discloses telecommunications means, See 503. Figure 4 of Kong shows the 64 Translator sitting between IPv6 and IPv4 networks to communicate with the Hosts.

Kong discloses routing the IPv6 data packet (the ([ConnectIPv4Server]₆) packet regularized to the ([ConnectIPv4Server]₄) packet) for its destination (IPv4 Server).

In the Figure 2 procedure, Kong explains that “IPv6 Host sends the ([ConnectIPv4Server]₆) through the 64 Translator to the IPv4 Server.” See 501. Specifically, the IPv6 host sends the ([ConnectIPv4Server]₆) packet to the 64 Translator, the 64 Translator “(6) translates the ([ConnectIPv4Server]₆) into the ([ConnectIPv4Server]₄) using the mapping entry,” and then the 64 Translator “sends the translated packet to IPv4 Server.” See 501.

Regarding claim 11:

A home gateway, comprising:

a processor; and a memory device coupled to the processor, wherein the memory device stores instructions that when executed by the processor cause the home gateway to:

Kong’s 64 Translator corresponds to the claimed home gateway. Kong implemented the 64 Translator as a Pentium PC, which has a 333 MHz Pentium processor and memory. See 503.

receive an IPv6 data packet in an IPv6 domain connected to an IPv4 domain,

Kong discloses a device for receiving an IPv6 data packet ([ConnectIPv4Server]₆) in an IPv6 domain (IPv6 Network) connected to an IPv4 domain (IPv4 Network). See below Figs. 1 and 2.

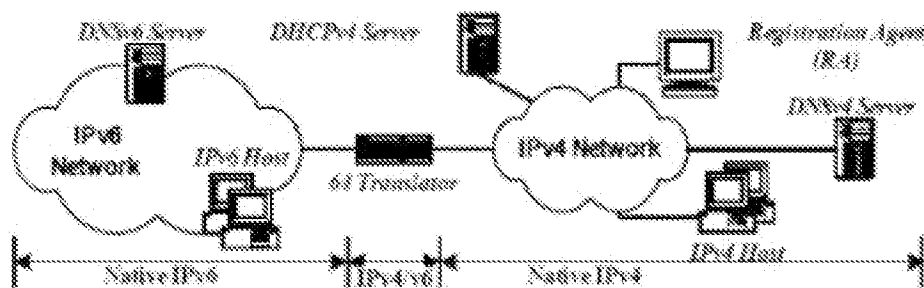


Fig. 1. The overall system architecture using 64TSD

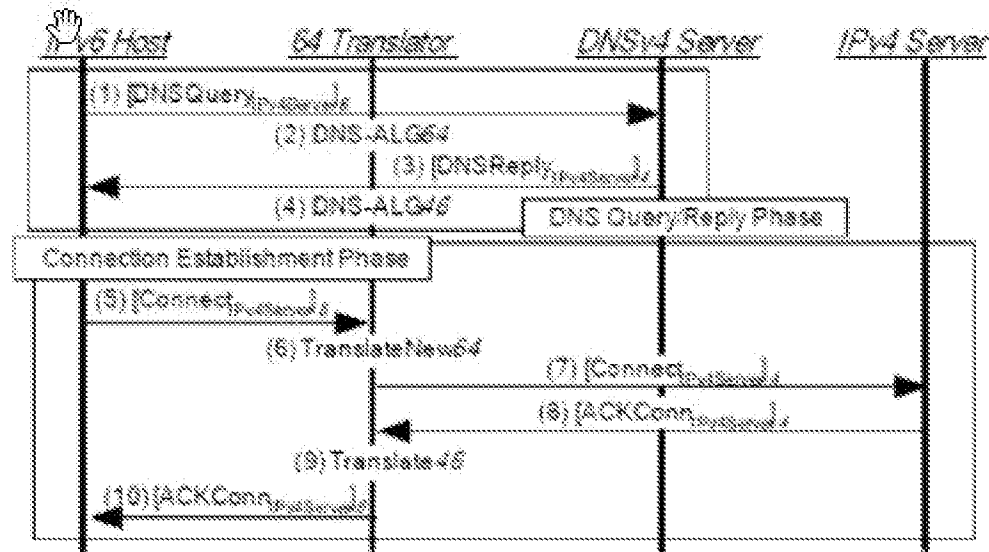


Fig. 2. The procedure of the outbound session

As shown above in Figure 1, an IPv6 host and server in an IPv6 domain (IPv6 network) connects to an IPv4 host and server in an IPv4 domain (IPv4 network) via a 64 Translator. See p 498 (“The overall system architecture using 64TSD consists of IPv6 Network, IPv4 Network and the 64 Translator”); see also pg. 499. The 64 Translator connects the IPv6 network to the IPv4 network. See also pages 500-501.

wherein said packet comprises an IPv6 destination address and an IPv6 source address,

Kong does not specifically disclose the packet “comprising an IPv6 destination address and an IPv6 source address”, however because Kong’s ([ConnectIPv4Server]₆) packet is an IPv6 packet, it necessarily includes an IPv6 destination address and an IPv6 source address.

For example, as set forth in Deering (Exhibit 1029 – page 4), the required format for the header of an IPv6 packet which compiles with IP Version 6 protocol includes both an IPv6 destination address and an IPv6 source address. See Deering pages 3-4.

[illegible]

identify an IPv6 destination address constructed by concatenating an IPv6 prefix, an IPv4 destination address and a destination port number;

Juniper Exhibit 1064, page 62

Kong discloses that the ([Connect_{IPv4Server}]₆) (which as explained above includes an IPv6 destination address) (IPv6 data packet) has an IPv6 destination address in the form shown above—i.e., constructed by concatenating an IPv6 prefix (dummy prefix), an IPv4 destination address and a destination port number (<IPv4 address, IPv4 Port>).

Figure 3 of Kong shows “[t]he procedure of the inbound session” in which the IPv4 host connects to the IPv6 network—i.e., communication in the opposition direction of Figure 2’s outbound session discussed above. See 501-502. In step (9) of the outbound session, the “64 Translator translates the ([Connect_{IPv4Server}]₄) to ([Connect_{IPv4Server}]₆) using the IPv6 address that is created by the concatenation of dummy prefix and IPv4 address.” See 502

See also Table 1. Kong shows the functions of the 64 Translator include allocating a port number for each outbound session and “TCP/UDP port number translation.”. See p 499

Kong further discloses identifying the IPv6 destination address in the ([Connect_{IPv4Server}]₆) packet. For example, Kong explains that “[t]he 64 Translator creates the mapping entry and translates the ([Connect_{IPv4Server}]₆) into the ([Connect_{IPv4Server}]₄) using the mapping entry.” See 501, FIG. 2. Kong explains that “[t]he mapping entry is a tuple of... where IPvXAddr and IPvXPort are IPvX address and port number of IPvX packet (X is 4 or 6).” See 499

regularize at least one of the IPv6 source address and the IPv6 destination address of the data packet by replacing said at least one IPv6 address,

Kong discloses the 64 Translator, if necessary (if communicating between IPv6 server and IPv4 host), regularizing (translating from IPv6 to IPv4) at least one of the IPv6 source address and the IPv6 destination address of the data packet (addresses in ([Connect_{IPv4Server}]₆) by replacing (replacing with address in ([Connect_{IPv4Server}]₄) said at least one Ipv6 address. See Fig. 2

In step (6) of Figure 3, “[t]he 64 Translator creates the mapping entry and translates the ([Connect_{IPv4Server}]₆) into the ([Connect_{IPv4Server}]₄) using the mapping entry.” See 501. The ([Connect_{IPv4Server}]₄) packet resulting from the translation is an IPv4 packet because “[Connect_H]_x ... is a packet to connect to host H using IPvX” where X=4. See 500. In Table 1, Kong explains that the

translation function of the 64 Translator involves, among other things, “translating a protocol header between IPv6 and IPv4.” See 499.

the replacement process belonging to a group comprising replacement of a constructed address with a native address and replacement of a native address with a constructed address,

Specifically, Kong discloses the replacement process belonging to a group comprising replacement of a constructed address (IPv6 address in ([ConnectIPv4Server]₆)) With a native address (IPv4 address in ([ConnectIPv4Server]₄)) and replacement of a native address with a constructed address. See Figure 2.

In step (6) of Figure 2, Kong’s “64 Translator creates the mapping entry and translates the ([ConnectIPv4Server]₆) into the ([ConnectIPv4Server]₄) using the mapping entry,” translating the Connect packet from IPv6 to IPv4. See 501. In this process, the 64 Translator replaces the IPv6 destination address with a corresponding IPv4 destination address. See 501.

Here, the IPv6 destination address in the ([ConnectIPv4Server]₆) packet corresponds to the constructed address because, as Kong explains, an IPv6 address “is created by the concatenation of dummy prefix and IPv4 address.” See 502. The IPv4 destination address in the ([ConnectIPv4Server]₄) packet, which replaces the IPv6 destination address in ([ConnectIPv4Server]₆) packet, corresponds to the claimed native address. This is because IPv4 addresses are native to the IPv4 network—the ultimate destination of the Connect packet.

and modifying the data packet using the regularized address; and

As explained above the 64 Translator is the device of the home gateway having a 333 MHz Pentium processor and memory. Kong discloses that the 64 Translator maintains mapping entries (table) for translating between addresses (i.e., converting between IPv6 and IPv4 addresses, and vice versa).

Kong’s 64 Translator modifies the ([ConnectIPv4Server]₆) packet into the ([ConnectIPv4Server]₄) packet by replacing the IPv6 source and destination addresses in the header of the ([ConnectIPv4Server]₆) packet with respective IPv4 source and destination addresses in the header of the resulting

([ConnectIPv4Server]₄) the packet. See 499, 501, FIG. 2. Here, the IPv4 destination address corresponds to the regularized address because it replaces the IPv6 destination address in the regularization (translation) process.

route the IPv6 data packet to its destination.

Kong teaches that “the 64 Translator is implemented on a 333MHz Pentium PC” and that 64 Translator PC communicates with the IPv6 and IPv4 Host PCs via “100Mbps Ethernet,” which discloses telecommunications means, See 503. Figure 4 of Kong shows the 64 Translator sitting between IPv6 and IPv4 networks to communicate with the Hosts.

Kong discloses routing the IPv6 data packet (the ([ConnectIPv4Server]₆) packet regularized to the ([ConnectIPv4Server]₄) packet) for its destination (IPv4 Server).

In the Figure 2 procedure, Kong explains that “IPv6 Host sends the ([ConnectIPv4Server]₆) through the 64 Translator to the IPv4 Server.” See 501. Specifically, the IPv6 host sends the ([ConnectIPv4Server]₆) packet to the 64 Translator, the 64 Translator “(6) translates the ([ConnectIPv4Server]₆) into the ([ConnectIPv4Server]₄) using the mapping entry,” and then the 64 Translator “sends the translated packet to IPv4 Server.” See 501.

Regarding claim 12:

The home gateway according to claim 11, wherein the memory device stores further instructions that when executed by the processor that causes the home gateway to: searching an address translation table of the gateway for an entry associating with said constructed destination address a native destination address; if an entry is found, replacing the constructed destination address of the data packet with the native destination address from the entry.

Kong discloses searching an address translation table (mapping entries) of the gateway (64 Translator) for an entry (mapping entry) associating with said constructed destination address (IPv6 address in the ([ConnectIPv4Server]₆) packet) a native destination address (IPv4 address in the ([ConnectIPv4Server]₄) packet). Kong also discloses if an entry (mapping entry) is found, replacing the

constructed destination address (IPv6 destination address) of the data packet ([ConnectIPv4Server]₆) packet) with the native destination address (corresponding IPv4 address in the ([ConnectIPv4Server]₄) packet) from that entry.

Kong explains that the 64 Translator maintains multiple “mapping entries” used for address translation. See, e.g., 499, 501, 501.

As Kong explains, when translating the ([ConnectIPv4Server]₆) packet to the ([ConnectIPv4Server]₄) packet, “[t]he 64 Translator ... translates the ([ConnectIPv4Server]₆) into the ([ConnectIPv4Server]₄) using the mapping entry.” See 501; see also 500 (Table 3 explaining that functions of the 64 Translator include “creation of the mapping entry and a translation of the packet from IPvX to IPvY using that mapping entry”);

Regarding claim 13:

The home gateway according to claim 12, wherein the wherein the memory device stores further instructions that when executed by the processor causes the home gateway to verify that the destination port number of the data packet and the destination port number contained in the constructed destination address match.

As explained above, in Kong, “an IPv6 address ... is created by the concatenation of dummy prefix and IPv4 address,” and that IPv4 address contains a destination port number. See 501.

As a result, a POSITA would understand that the destination port number in the constructed IPv6 address of the ([ConnectIPv4Server]₆) packet matches the destination port number of the underlying IPv4 address. Kong discloses or suggests verifying that the two port numbers match by looking up the correct mapping entry for the translation using the address information in the packet. See, 501 (“it looks up the mapping entries with <IPv4 address, Pv5 Port>... If the entry is found, it translates...”).

Conclusion

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the date of this final action.

10. The patent owner is reminded of the continuing responsibility under 37 CFR 1.565(a) to apprise the Office of any litigation activity, or other prior or concurrent proceeding, involving Patent No. 8,451,845 throughout the course of this reexamination proceeding. The Patent Owner is also reminded of the ability to similarly apprise the Office of any such activity or proceeding throughout the course of this reexamination proceeding. See MPEP §§ 2207, 2282 and 2286.

11. Extensions of time under 37 CFR 1.136(a) will not be permitted in these proceedings because the provisions of 37 CFR 1.136 apply only to "an applicant" and not to parties in a reexamination proceeding. Additionally, 35 U.S.C. 305 requires that reexamination proceedings "will be conducted with special dispatch" (37 CFR 1.550(a)). Extension of time in ex parte reexamination proceedings are provided for in 37 CFR 1.550(c).

All correspondence relating to this ex parte reexamination proceeding should be directed:
Electronically: Registered users may submit via Patent Center at <https://patentcenter.uspto.gov/>.

By Mail to: Mail Stop *Ex Parte* Reexam
 Central Reexamination Unit

Commissioner for Patents
United States Patent & Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

By FAX to: (571) 273-9900
Central Reexamination Unit

By hand: Customer Service Window
Randolph Building
401 Dulany Street
Alexandria, VA 22314

For electronic transmissions, 37 CFR 1.8(a)(1)(i)(C) and (ii) states that correspondence (except for a request for reexamination and a corrected or replacement request for reexamination) will be considered timely filed if (a) it is transmitted via the USPTO patent electronic filing system in accordance with 37 CFR 1.6(a)(4), and (b) includes a certificate of transmission for each piece of correspondence stating the date of transmission, which is prior to the expiration of the set period of time in the Office action.

Any inquiry by the patent owner concerning this communication or earlier communications from the Legal Advisor or Examiner, or as to the status of this proceeding, should be directed to the Central Reexamination Unit at telephone number (571) 272-7705.

/Ovidio Escalante/
Ovidio Escalante
Reexamination Specialist
Central Reexamination Unit - Art Unit 3992
(571) 272-7537

Conferees:

/MINH DIEU NGUYEN/
Primary Examiner, Art Unit 3992
/MICHAEL FUELLING/
Supervisory Patent Examiner, Art Unit 3992

Office Action in Ex Parte Reexamination	Control No. 90/019,246	Patent Under Reexamination 8451845	
	Examiner OVIDIO ESCALANTE	Art Unit 3992	AIA (FITF) Status No

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

- a. ☒ Responsive to the communication(s) filed on 08 May 2024.
☐ A declaration(s)/affidavit(s) under **37 CFR 1.130(b)** was/were filed on _____.

b. ☒ This action is made FINAL.

c. ☒ A statement under 37 CFR 1.530 has not been received from the patent owner.

A shortened statutory period for response to this action is set to expire 2 month(s) from the mailing date of this letter. Failure to respond within the period for response will result in termination of the proceeding and issuance of an *ex parte* reexamination certificate in accordance with this action. 37 CFR 1.550(d). **EXTENSIONS OF TIME ARE GOVERNED BY 37 CFR 1.550(c)**. If the period for response specified above is less than thirty (30) days, a response within the statutory minimum of thirty (30) days will be considered timely.

Part I THE FOLLOWING ATTACHMENT(S) ARE PART OF THIS ACTION:

- | | |
|--|---|
| 1. ☐ Notice of References Cited by Examiner, PTO-892. | 3. ☐ Interview Summary, PTO-474. |
| 2. ☐ Information Disclosure Statement, PTO/SB/08. | 4. ☐ _____. |

Part II SUMMARY OF ACTION

- 1a. ☒ Claims 1,5-8 and 11-13 are subject to reexamination.
- 1b. ☐ Claims 2-4 and 9-10 are not subject to reexamination.
2. ☐ Claims _____ have been canceled in the present reexamination proceeding.
3. ☐ Claims _____ are patentable and/or confirmed.
4. ☒ Claims 1,5-8 and 11-13 are rejected.
5. ☐ Claims _____ are objected to.
6. ☐ The drawings, filed on _____ are acceptable.
7. ☐ The proposed drawing correction, filed on _____ has been (7a) ☐ approved (7b) ☐ disapproved.
8. ☐ Acknowledgment is made of the priority claim under 35 U.S.C. 119(a)-(d) or (f).
a) ☐ All b) ☐ Some* c) ☐ None of the certified copies have
1 ☐ been received.
2 ☐ not been received.
3 ☐ been filed in Application No. _____.
4 ☐ been filed in reexamination Control No. _____.
5 ☐ been received by the International Bureau in PCT application No. _____.

* See the attached detailed Office action for a list of the certified copies not received.

9. ☐ Since the proceeding appears to be in condition for issuance of an *ex parte* reexamination certificate except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte* Quayle, 1935 C.D. 11, 453 O.G. 213.
10. ☐ Other: _____

cc: Requester (if third party requester)