

UNITED STATES PATENT AND TRADEMARK OFFICE

CISCO SYSTEMS, INC.

Petitioner,

v.

UMBRA TECHNOLOGIES LTD.

Patent Owner.

Case No. IPR2024-00497

U.S. Patent No. 10,630,505

PATENT OWNER'S TRIAL DEMONSTRATIVE

Mail Stop PATENT BOARD
Patent Trial and Appeal Board
U.S. Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
Submitted Electronically via PTAB E2E

PETITIONER
Cisco Systems, Inc.,

v.

PATENT OWNER
UMBRA Technologies Ltd.

U.S. Patent No. 11,108,595

Inter Partes Review No.: IPR2024-00270

PETITIONER
Cisco Systems, Inc.,

v.

PATENT OWNER
UMBRA Technologies Ltd.

U.S. Patent No. 10,630,505

Inter Partes Review No.: IPR2024-00497

Patent Owner UMBRA's Combined Trial Presentation

June 4, 2025

PART 1:

The '595 Patent

- The '595 Patent's Background
- Hankins Does Not Disclose “*A List of Available Servers*”:
 - That is Received in Response to an Endpoint’s “*request for a list of available servers*”
 - That an Endpoint Device “Could Utilize to Build a Tunnel”
 - That is “*Dynamically Order[ed] . . . Based on*” the Endpoint
- Kommula Fails as a Prior Art Reference
- The Petition Fails to Address Claims 19 and 20

- The '595 Patent's Background
- Hankins Does Not Disclose “*A List of Available Servers*”:
 - That is Received in Response to an Endpoint’s “*request for a list of available servers*”
 - That an Endpoint Device “Could Utilize to Build a Tunnel”
 - That is “*Dynamically Order[ed] . . . Based on*” the Endpoint
- Kommula Fails as a Prior Art Reference
- The Petition Fails to Address Claims 19 and 20

Background of UMBRA U.S. Patent 11,108,595 (“the ’595 patent”)

- The ’595 patent claims at issue (15-20) resulted from the pioneering efforts of the UMBRA inventors in the 2010s in areas such as the management of a global virtual network connection via a tunnel between an endpoint and an access point server controlled by a control server.
- The improvements overcame information exchange limitations in the tunnel building process.
- The improvements resolved, for example, problems tied to the finite number of public and private IP address, which influences which endpoint devices can connect to which access point servers. (See Ex. 1001, the ’595 patent, at 15:46-49.) By publishing address availability and dynamically managing availability to avoid conflict, the same address can be used for multiple planned connections. (*Id.* at 15:50-67.)

The '595 Dispute is Focused on Ground 2: Claim 15

- Patent Owner focused the '595 dispute to claims 15-20 by statutorily disclaiming claims 1-14, but did not concede the veracity of any of Petitioner's allegations:

C. The Remaining Claims (1-14) Have Been Statutorily Disclaimed

Concurrently herewith, Patent Owner has filed a statutory disclaimer of claims 1-14 under 37 C.F.R. § 1.321(a). Under 37 C.F.R. § 42.107(e), no *inter partes* review may be instituted based on disclaimed claims.

Patent Owner does not by way of the disclaimer admit the allegations made regarding claims 1-14; indeed, many of the allegations are deeply flawed. With the disclaimer of claims 1-14, however, all remaining claims depend from claim 15, for which the Petition fails to make a *prima facie* case of obviousness.

(Paper 6, POPR, at 15.)

Claim 15 Recites a Network Sfor Managing a GVN Composed of Access Points TO BE Connected to Endpoints via Tunnels, Directed by a Control Server

Claim 15 of the '595 patent recites:

[15.0] A network system for managing a global virtual network, comprising:

[15.1] an access point server configured to connect to one or more endpoint devices via one or more first tunnels; and

[15.2] a control server connected to the access point server via a second tunnel, the control server storing one or more records associated with each of the one or more endpoint devices, wherein the control server is configured to:

[15.3] receive a request for a list of available servers from a first endpoint device among the one or more endpoint devices, the list of available servers including the access point server;

[15.5] retrieve the list of available servers; retrieve the one or more records associated with the first endpoint device;

[15.5] dynamically order the list of available servers for the first endpoint device based on the one or more records associated with the network device; and

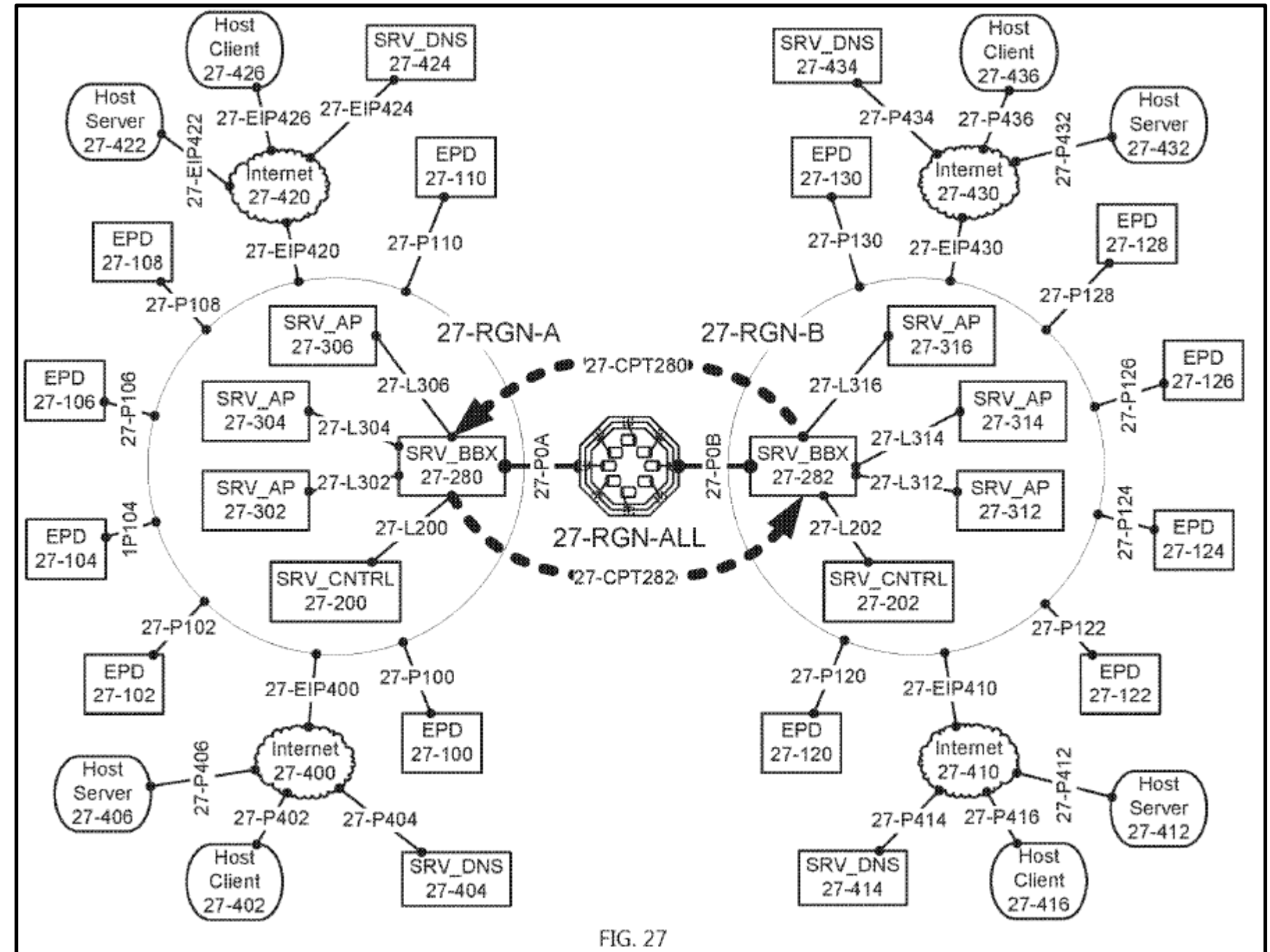
[15.6] transmit the ordered list of available servers to the first endpoint device.

(EX 1001, '595 patent, at 34:4-24.)

The '595 Patent Provides a Mechanism for Balancing Demand for Resources to Build Tunnels

- By supplying each endpoint device (EPD), in an array of numerous EPDs, with a dynamically customized list of multiple *potential* access point servers (SRV_APs) *for tunnel building* in the overlay network, the EPD is given dynamically ordered choices which “balanc[e] demand for resources by . . . EPDs wanting to build[] tunnels against available resources of SRV_APs which they would connect with.” (Ex. 1001, '595 patent, 40-43.)

(See also, Paper 2, Pet., at 9; Ex. 1003, Reddy Decl., at ¶ 43; Paper 21, POR, at 3 (citing Ex. 1001, '595 patent, at 2:30-36, 11:2-26, 14:46-48, 15:43-67, claim 15; Ex. 2002, Beck Decl., at ¶ 31.)



(Ex. 1001, '595 patent, at Fig. 27; see also, Paper 21, POR, at 3.)

- The '595 Patent's Background
- Hankins Does Not Disclose “*A List of Available Servers*”:
 - That is Received *in Response to* an Endpoint’s “*request for a list of available servers*”
 - That an Endpoint Device “*Could Utilize to Build a Tunnel*”
 - That is “*Dynamically Order[ed] . . . Based on*” the Endpoint
- Kommula Fails as a Prior Art Reference
- The Petition Fails to Address Claims 19 and 20

Element [15.3] Requires a Control Server Configured to “*Receive a Request for a List of Available Servers . . .*”

Claim 15 of the '595 patent recites:

[15.0] A network system for managing a global virtual network, comprising:

[15.1] an access point server configured to connect to one or more endpoint devices via one or more first tunnels; and

[15.2] a control server connected to the access point server via a second tunnel, the control server storing one or more records associated with each of the one or more endpoint devices, wherein ***the control server is configured to:***

[15.3] ***receive a request for a list of available servers from a first endpoint device*** among the one or more endpoint devices, the list of available servers including the access point server;

[15.4] retrieve the list of available servers; retrieve the one or more records associated with the first endpoint device;

[15.5] dynamically order the list of available servers for the first endpoint device based on the one or more records associated with the network device; and

[15.6] transmit the ordered list of available servers to the first endpoint device.

(EX 1001, '595 patent, at 34:4-24.)

The “*List of Available Servers*” is Central to Claim 15 Because it Is the Item: (i) Requested, (ii) Retrieved, (iii) Dynamically Ordered, and (iv) Transmitted

[15.0] A network system . . . comprising:

[15.1] an access point server configured to connect to one or more endpoint devices via one or more first tunnels; and

[15.2] a control server . . . storing one or more records associated with each of the one or more endpoint devices, wherein the control server is configured to:

[15.3] *receive a request for a list of available servers* from a first endpoint device among the one or more endpoint devices, the list of available servers including the access point server;

[15.4] ***retrieve the list of available servers***; retrieve the one or more records associated with the first endpoint device;

[15.5] ***dynamically order the list of available servers*** for the first endpoint device based on the one or more records associated with the network device; and

[15.6] ***transmit the ordered list of available servers*** to the first endpoint device.

(EX 1001, '595 patent, at 34:4-24.)

trivial. The “list” is repeatedly referenced throughout claim 15 of the '595 patent. (See, e.g., Ex. 1001 at 34:13, 34:15, 34:17, 34:20, and 34:23.) It is the subject of the request that is “*receive[d]*” by the control server from a “first endpoint device”. It is what is “*retrieve[d]*” by that control server. It is what is “*dynamically order[ed]*” by that control server. And it is, once ordered, what is “*transmit[ted]*” by that control server back to the first endpoint device issuing the initial request for the list. (Ex. 1001 at 34:12-24.) One could not simply substitute the word “*services*” in place of the claimed “*list of available servers*” in claim 15 of the '595 patent and yield a claim that would make any reasonable sense.

(Paper 21, POR, at 10.)

Petitioner Conflates “a Request for a List of Available Servers” with a Request for “Services Available from a Gateway”

- Petitioner lacks support to equate “a request for a list of available servers” with a request for “services available from a gateway.”

[15.3] wherein the control server is configured to receive a request for a list of available servers from a first endpoint device among the one or more endpoint devices, the list of available servers including the access point server;

Hankins discloses that the controller receives a request from a gateway 311

(*endpoint device*) for a list of available servers including tunnel servers (*access*

point server). For example, Hankins discloses that gateway 311 in one logical

broadcast domain, may request services available from gateway 321 in another

logical broadcast domain. *EX-1005* 12:9-16. The controller identifies potential

tunnel servers that are available to provide a tunnel between gateway 311 to

gateway 321. For example, as shown in Annotated FIG. 2, tunnel servers 360 and

370 can be used to establish communications between gateway 311 and 321. Thus,

a POSITA understood that a request from gateway 311 for services available from

gateway 321 is a request for available tunnel servers that can provide a tunnel

between gateway 311 and 321. This is consistent with the disclosure of the '595

(Paper 2, Pet., at 45-46.)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Conflates “a Request for a List of Available Servers” with a Request for “Services Available from a Gateway”

- Petitioner lacks support to equate “a request for a list of available servers” with a request for “services available from a gateway.”

[15.3] wherein the control server is configured to receive a request for a list of available servers from a first endpoint device among the one or more endpoint devices, the list of available servers including the access point server;

Hankins discloses that the controller receives a request from a gateway 311

(*endpoint device*) for a list of available servers including tunnel servers (*access point server*). For example, Hankins discloses that gateway 311 in one logical

broadcast domain, may request services available from gateway 321 in another

logical broadcast domain. *EX-1005* 12:9-16. The controller identifies potential

tunnel servers that are available to provide a tunnel between gateway 311 to

gateway 321. For example, as shown in Annotated FIG. 2, tunnel servers 360 and

370 can be used to establish communications between gateway 311 and 321. Thus,

a POSITA understood that a request from gateway 311 for services available from

gateway 321 is a request for available tunnel servers that can provide a tunnel

between gateway 311 and 321. This is consistent with the disclosure of the '595

- The Petition’s cited portion of Hankins does not even mention a controller or a server, let alone a “*controller [that] identifies potential tunnel servers that are available to provide a tunnel*” in response to a gateway’s request for services.

(Paper 2, Pet., at 45-46.)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Conflates “a Request for a List of Available Servers” with a Request for “Services Available from a Gateway”

- Petitioner lacks support to equate “a request for a list of available servers” with a request for “services available from a gateway.”

[15.3] wherein the control server is configured to receive a request for a list of available servers from a first endpoint device among the one or more endpoint devices, the list of available servers including the access point server;

Hankins discloses that the controller receives a request from a gateway 311

(*endpoint device*) for a list of available servers including tunnel servers (*access point server*). For example, Hankins discloses that gateway 311 in one logical

broadcast domain, may request services available from gateway 321 in another

logical broadcast domain. *EX-1005* 12:9-16. The controller identifies potential

tunnel servers that are available to provide a tunnel between gateway 311 to

gateway 321. For example, as shown in Annotated FIG. 2, tunnel servers 360 and

370 can be used to establish communications between gateway 311 and 321. Thus,

a POSITA understood that a request from gateway 311 for services available from

gateway 321 is a request for available tunnel servers that can provide a tunnel

between gateway 311 and 321. This is consistent with the disclosure of the '595

- The Petition’s cited portion of Hankins does not even mention a controller or a server, let alone a “***controller [that] identifies potential tunnel servers that are available to provide a tunnel***” in response to a gateway’s request for services.

In one example illustrative embodiment, use of a network device, for example, may allow network devices communicating via their respective logical broadcast domains, for example, to discover and/or request services available via network devices of another logical broadcast domain while still communicating via their respective logical broadcast domains, potentially at least partially with less complexity, traffic and/or expense than conventional routing. In this

(Ex. 1005, Hankins, at 12:9-16.)

(Paper 2, Pet., at 45-46.)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner Conflates “a Request for a List of Available Servers” with a Request for “Services Available from a Gateway”

- Petitioner lacks support to equate “a request for a list of available servers” with a request for “services available from a gateway.”

[15.3] wherein the control server is configured to receive a request for a list of available servers from a first endpoint device among the one or more endpoint devices, the list of available servers including the access point server;

Hankins discloses that the controller receives a request from a gateway 311 (endpoint device) for a list of available servers including tunnel servers (access

...

370 can be used to establish communications between gateway 311 and 321. Thus,
a POSITA understood that a request from gateway 311 for services available from gateway 321 is a request for available tunnel servers that can provide a tunnel between gateway 311 and 321. This is consistent with the disclosure of the '595 Patent that when a device makes a request to a server, what it is really asking for is a list of available servers, and specifically an IP and port which it could utilize to in the build a tunnel. EX-1001 23:11-24; EX-1003 ¶¶ 139, 140.

- The Petition’s support for what “a POSITA understood,” is a misstatement of the ’595 patent, which refers to “a request to a server availability module,” not a “a request to a server.”

(Paper 2, Pet., at 45-46.)

Petitioner Conflates “a Request for a List of Available Servers” with a Request for “Services Available from a Gateway”

- Petitioner lacks support to equate “a request for a list of available servers” with a request for “services available from a gateway.”

[15.3] wherein the control server is configured to receive a request for a list of available servers from a first endpoint device among the one or more endpoint devices, the list of available servers including the access point server;

Hankins discloses that the controller receives a request from a gateway 311 (endpoint device) for a list of available servers including tunnel servers (access

. . .

370 can be used to establish communications between gateway 311 and 321. Thus, a POSITA understood that a request from gateway 311 for services available from gateway 321 is a request for available tunnel servers that can provide a tunnel between gateway 311 and 321. This is consistent with the disclosure of the '595 Patent that when a device makes a request to a server, what it is really asking for is a list of available servers, and specifically an IP and port which it could utilize to in the build a tunnel. EX-1001 23:11-24; EX-1003 ¶¶ 139, 140.

(Paper 2, Pet., at 45-46.)

- The Petition’s support for what “a POSITA understood,” is a misstatement of the ’595 patent, which refers to “a request to a server availability module,” not a “a request to a server.”

When a device makes a request to the server availability module, what it is really asking for is a list of available servers, and specifically an IP and port which it could utilize to in the present or in the future to build a tunnel to. Each (Ex. 1001, ’595 patent, at 23:17-20.)

Petitioner Conflates “a Request for a List of Available Servers” with a Request for “Services Available from a Gateway”

- Petitioner lacks support to equate “a request for a list of available servers” with a request for “services available from a gateway.”

[15.3] wherein the control server is configured to receive a request for a list of available servers from a first endpoint device among the one or more endpoint devices, the list of available servers including the access point server;

Hankins discloses that the controller receives a request from a gateway 311 (endpoint device) for a list of available servers including tunnel servers (access

. . .

370 can be used to establish communications between gateway 311 and 321. Thus, a POSITA understood that a request from gateway 311 for services available from gateway 321 is a request for available tunnel servers that can provide a tunnel between gateway 311 and 321. This is consistent with the disclosure of the '595 Patent that when a device makes a request to a server, what it is really asking for is a list of available servers, and specifically an IP and port which it could utilize to in the build a tunnel. EX-1001 23:11-24; EX-1003 ¶¶ 139, 140.

(Paper 2, Pet., at 45-46.)

- The Petition’s support for what “a POSITA understood,” is a misstatement of the '595 patent, which refers to “a request to a server availability module,” not a “a request to a server”.

When a device makes a request to the server availability module, what it is really asking for is a list of available servers, and specifically an IP and port which it could utilize to in the present or in the future to build a tunnel to. Each (Ex. 1001, '595 patent, at 23:17-20.)

- It is also improper to “rel[y] on the [challenged] patent itself as [a] roadmap for putting . . . pieces of a 'jigsaw puzzle' together” to assess obviousness. *TQ Delta, LLC v. Cisco Sys.*, 942 F.3d 1352, 1360 (Fed. Cir. 2019). (Paper 21, POR, at 13.)

Hankins Does Not Say Tunnels Are Provisioned in Response to a Gateway Device's Request for Services from Another Gateway Device

- Hankins explains that “**available services** via a network device may be determined based . . . on which ports are open,” i.e., “transmitting signal packets to **request a connection with a network device port**. A network device may respond, for example, by transmitting signal packets indicating an available connection.”

Likewise, in another embodiment, available services via a network device may be determined based at least in part on which ports are open on a network device. Port scanning a network device may determine which ports may be open by transmitting signal packets to request a connection with a network device on a port. A network device may respond, for example, by transmitting signal packets indicating an available connection. See, for example, RFC 793—Transmission Control Protocol, available, for example, from the Internet Engineering Task Force (IETF), at www.ietf.org. Signal packets comprising responses may imply that a port may be available.

(Ex. 1005, Hankins, at 10:23-34.)

Hankins Does Not Say Tunnels Are Provisioned in Response to a Gateway Device's Request for Services from Another Gateway Device

- Hankins explains that “**available services** via a network device may be determined based . . . on which ports are open,” i.e., “transmitting signal packets to **request a connection with a network device port**. A network device may respond, for example, by transmitting signal packets indicating an available connection.”

Likewise, in another embodiment, available services via a network device may be determined based at least in part on which ports are open on a network device. Port scanning a network device may determine which ports may be open by transmitting signal packets to request a connection with a network device on a port. A network device may respond, for example, by transmitting signal packets indicating an available connection. See, for example, RFC 793—Transmission Control Protocol, available, for example, from the Internet Engineering Task Force (IETF), at www.ietf.org. Signal packets comprising responses may imply that a port may be available.

- Hankins contemplates network devices will, themselves, determine the availability of services from other network devices, by transmitting directly to those devices.

(Ex. 1005, Hankins, at 10:23-34.)

Hankins Does Not Say Tunnels Are Provisioned in Response to a Gateway Device's Request for Services from Another Gateway Device

- Hankins explains that “**available services** via a network device may be determined based . . . on which ports are open,” i.e., “transmitting signal packets to **request a connection with a network device port**. A network device may respond, for example, by transmitting signal packets indicating an available connection.”

Likewise, in another embodiment, available services via a network device may be determined based at least in part on which ports are open on a network device. Port scanning a network device may determine which ports may be open by transmitting signal packets to request a connection with a network device on a port. A network device may respond, for example, by transmitting signal packets indicating an available connection. See, for example, RFC 793—Transmission Control Protocol, available, for example, from the Internet Engineering Task Force (IETF), at www.ietf.org. Signal packets comprising responses may imply that a port may be available.

(Ex. 1005, Hankins, at 10:23-34.)

- Hankins contemplates network devices will, themselves, determine the availability of services from other network devices, by transmitting directly to those devices.
- By the time this “*available services*” inquiry takes place, the network device is already able to transmit to another network device, i.e., there is no need to request a server.

Hankins Does Not Say Tunnels Are Provisioned in Response to a Gateway Device's Request for Services from Another Gateway Device

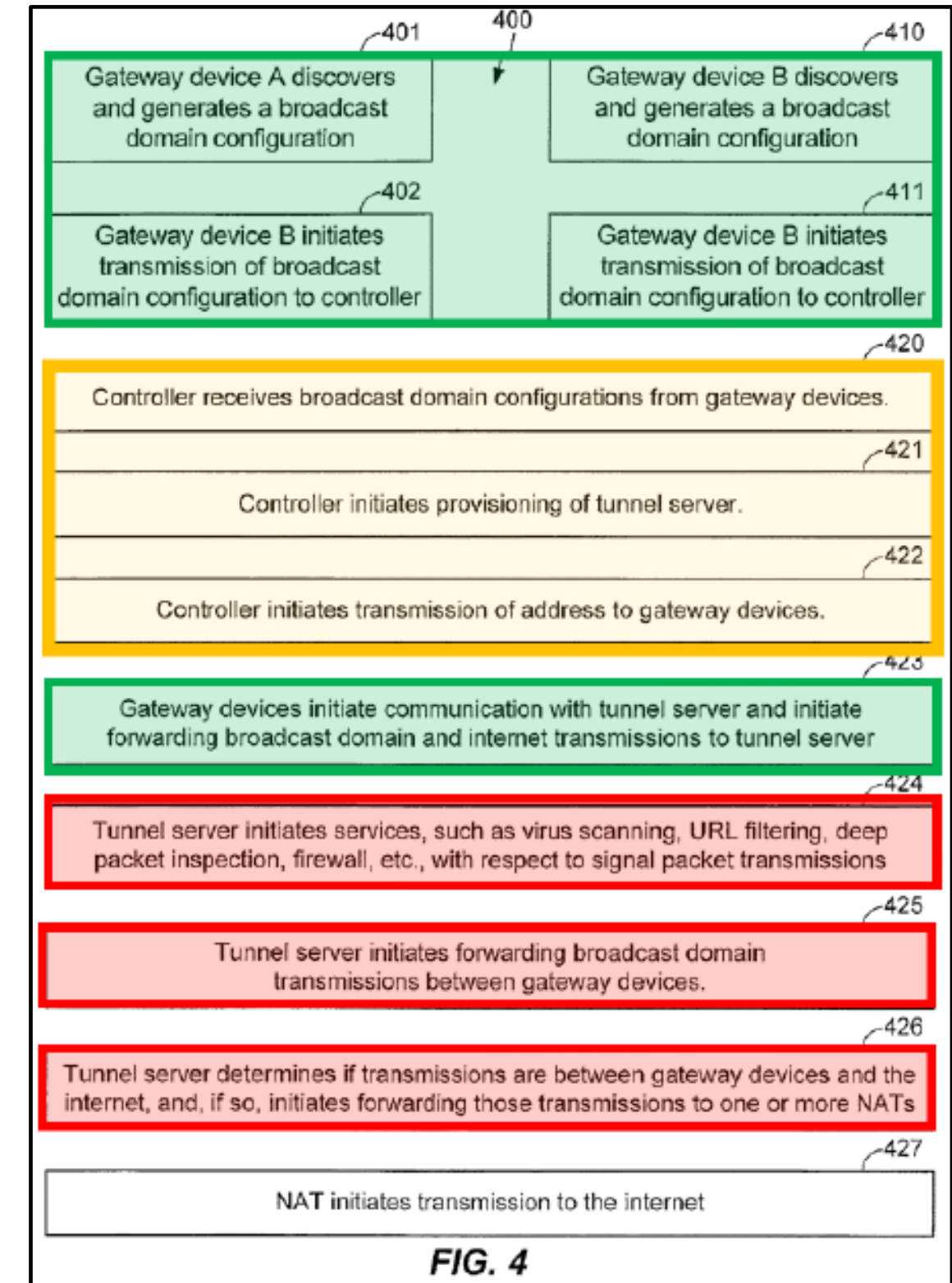
- *Nothing* suggests Hankins' *provisioning is done in response to a gateway's request for the services*. To the contrary, Hankins' tunnel server provisioning stems from receipt of broadcast domain configurations.

communication via more than one tunnel server. Gateway devices 311, 321, 331, and 341 may generate broadcast configurations for their respective logical broadcast domains which may be used by controller to identify a list of available tunnel servers. For example, controller 350 may provision tunnel server one 360 and initiate instructions to gateway devices 311, 321, 331, and 341 to communicate with tunnel server one 360. *EX-1004* 24:47-55; *EX-1003* ¶ 143.

(Paper 2, Pet., at 47.)

Hankins Transmits a Server Address at the Time of Provisioning, Not in Response to Request for Services

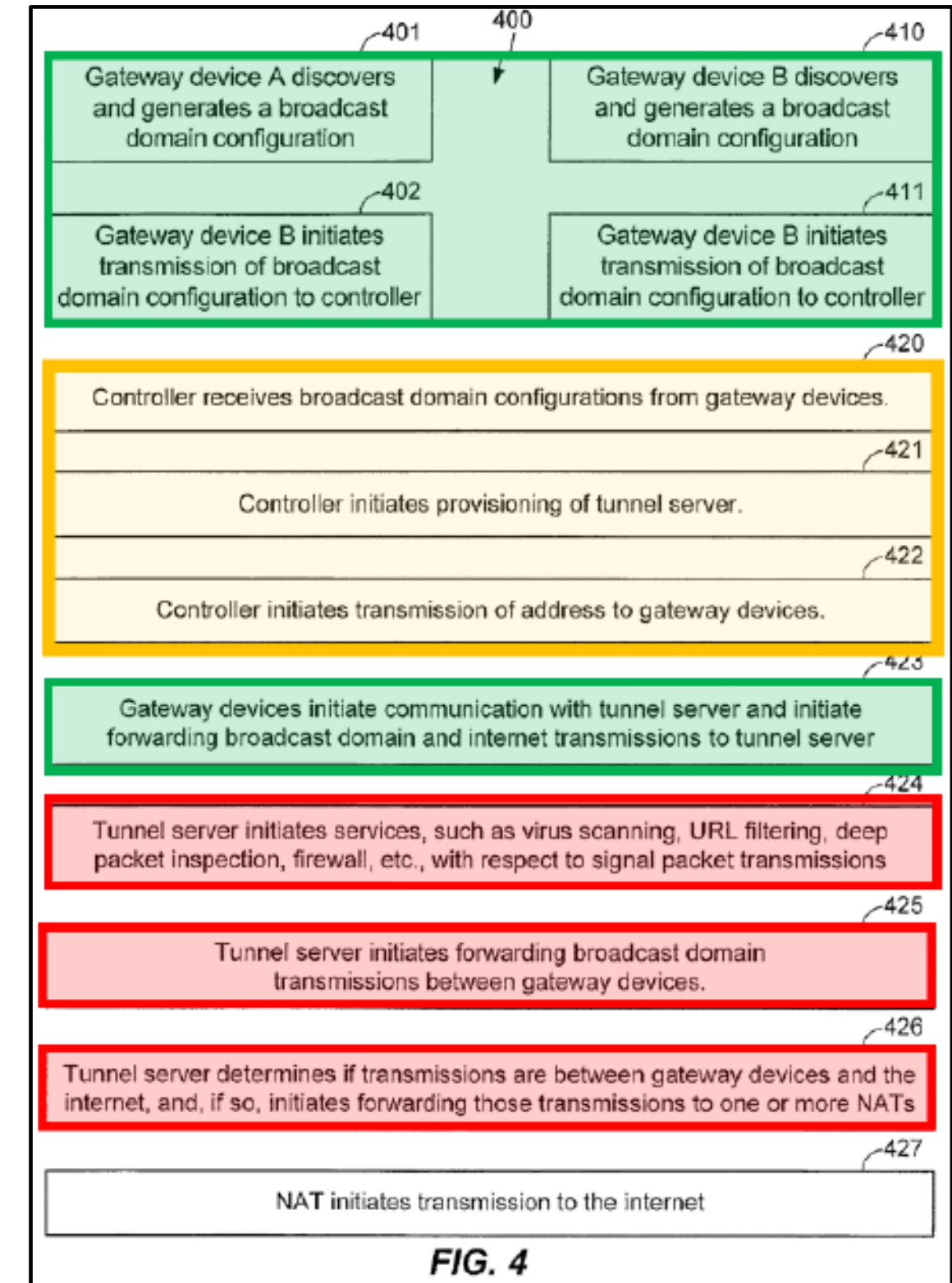
- Hankins provides each gateway with the network addresses for tunnel servers after provisioning. Thus, “provisioning” is more than merely giving an address. And there is **NO** suggestion that those addresses are given *in response to a gateway’s “request” for services*. (Paper 6, POPR, at 4 (citing Ex. 1005, Hankins, at 19:32-35; 25:22-26); see also, Ex. 1005, Hankins, at 28:56-61.)



(Paper 2, Pet., at 40 (annotating Ex. 1005, Hankins, Fig. 4).)

Hankins Transmits a Server Address at the Time of Provisioning, Not in Response to Request for Services

- Hankins provides each gateway with the network addresses for tunnel servers after provisioning. Thus, “provisioning” is more than merely giving an address. And there is **NO** suggestion that those addresses are given *in response to a gateway’s “request” for services*. (Paper 6, POPR, at 4 (citing Ex. 1005, Hankins, at 19:32-35; 25:22-26); see also, Ex. 1005, Hankins, at 28:56-61.)
- In fact, Hankins Figure 4 indicates that tunnel server provisioning is done *in response to receiving broadcast domain configurations*, rather than a gateway’s request for service.

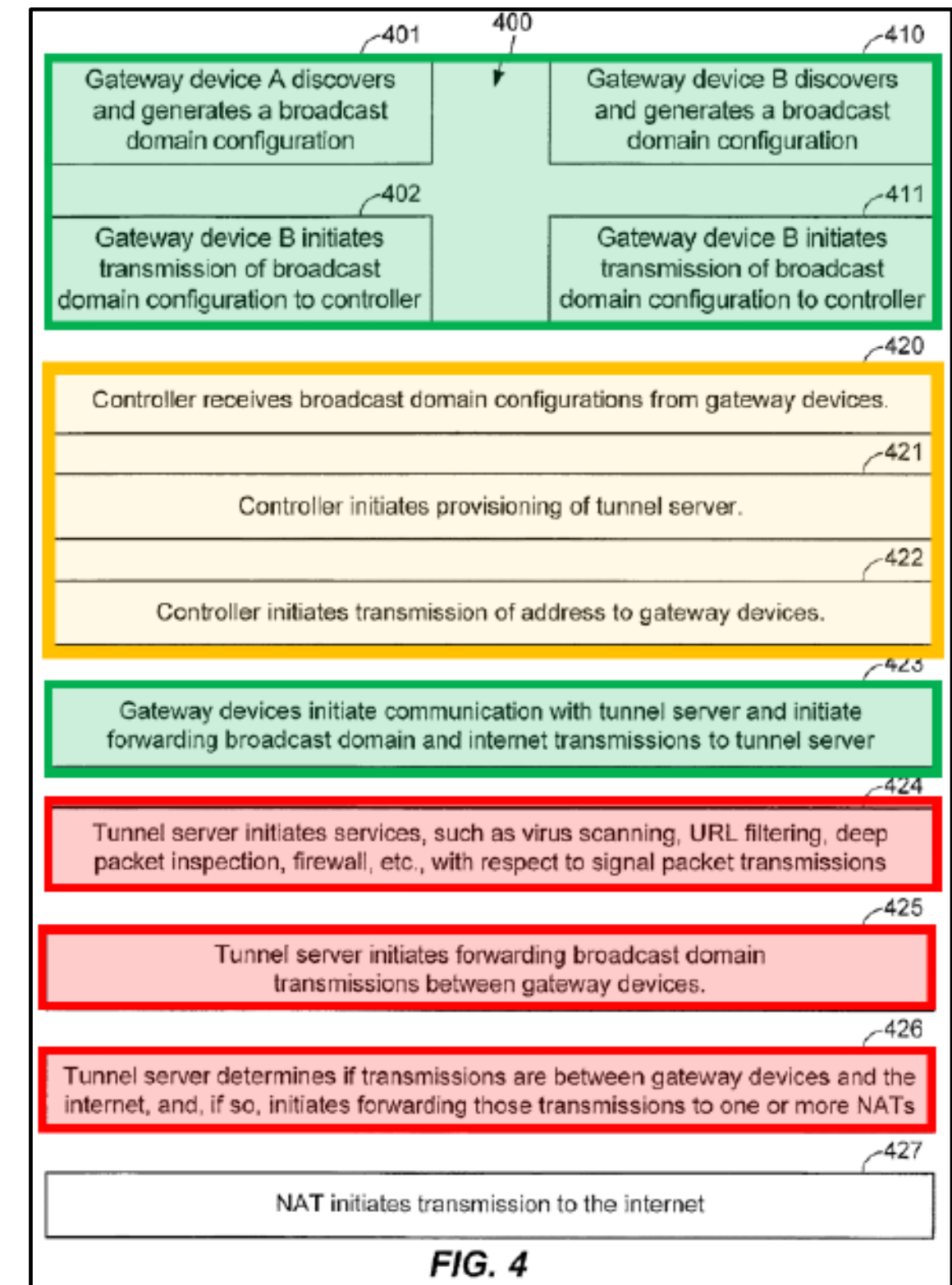


(Paper 2, Pet., at 40 (annotating Ex. 1005, Hankins, Fig. 4).)

Hankins Transmits a Server Address at the Time of Provisioning, Not in Response to Request for Services

- Hankins provides each gateway with the network addresses for tunnel servers after provisioning. Thus, “provisioning” is more than merely giving an address. And there is **NO** suggestion that those addresses are given *in response to a gateway’s “request” for services*. (Paper 6, POPR, at 4 (citing Ex. 1005, Hankins, at 19:32-35; 25:22-26); see also, Ex. 1005, Hankins, at 28:56-61.)
- In fact, Hankins Figure 4 indicates that tunnel server provisioning is done *in response to receiving broadcast domain configurations*, rather than a gateway’s request for service.
- Once those tunnel servers are “provisioned” to a gateway, it is illogical to suggest that the gateway would “request a list of available servers” to which it is already provisioned.

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE



(Paper 2, Pet., at 40 (annotating Ex. 1005, Hankins, Fig. 4).)

Petitioner Relies on Hankins, which Fails to Disclose [15.3], a Control Server Configured to: “*Receive a Request for a List of Available Servers . . .*”

- Petitioner relies on conclusory assertions of what “*a POSITA understood*” to equate a gateway-to-gateway request for services with the claimed “*request for a list of available servers.*”

gateway 321. For example, as shown in Annotated FIG. 2, tunnel servers 360 and 370 can be used to establish communications between gateway 311 and 321. Thus, a POSITA understood that a request from gateway 311 for services available from gateway 321 is a request for available tunnel servers that can provide a tunnel between gateway 311 and 321. This is consistent with the disclosure of the '595

(Paper 2, Pet., at 46.)

26:55-65. Finally, Hankins discloses that the provisioning of the tunnel servers by the controller via execution of one or more software applications EX-1005 25:20-22. Based on this disclosure, a POSITA understood that Hankins discloses that a request for services from a network device (e.g., content from a server) would be sent from a gateway (endpoint device) to a controller, that would use a software module to provision one or more tunnel servers to connect the gateway with another gateway to receive the services. Thus, Hankins discloses limitation [15.3].

(Paper 22, Reply, at 4.)

Petitioner Relies on Hankins, which Fails to Disclose [15.3], a Control Server Configured to: “*Receive a Request for a List of Available Servers . . .*”

- Petitioner relies on conclusory assertions of what “*a POSITA understood*” to equate a gateway-to-gateway request for services with the claimed “*request for a list of available servers.*”

gateway 321. For example, as shown in Annotated FIG. 2, tunnel servers 360 and 370 can be used to establish communications between gateway 311 and 321. Thus, a POSITA understood that a request from gateway 311 for services available from gateway 321 is a request for available tunnel servers that can provide a tunnel between gateway 311 and 321. This is consistent with the disclosure of the '595

(Paper 2, Pet., at 46.)

- Conclusory POSITA assertions are improper. (See Paper 23, Sur-reply, at 4 (citing *Innogenetics, N.V. v. Abbott Labs.*, 512 F.3d 1363, 1373 (Fed. Cir. 2008).)

26:55-65. Finally, Hankins discloses that the provisioning of the tunnel servers by the controller via execution of one or more software applications EX-1005 25:20-22. Based on this disclosure, a POSITA understood that Hankins discloses that a request for services from a network device (e.g., content from a server) would be sent from a gateway (endpoint device) to a controller, that would use a software module to provision one or more tunnel servers to connect the gateway with another gateway to receive the services. Thus, Hankins discloses limitation [15.3].

(Paper 22, Reply, at 4.)

Petitioner Relies on Hankins, which Fails to Disclose [15.3], a Control Server Configured to: “*Receive a Request for a List of Available Servers . . .*”

- Petitioner relies on conclusory assertions of what “*a POSITA understood*” to equate a gateway-to-gateway request for services with the claimed “*request for a list of available servers.*”

gateway 321. For example, as shown in Annotated FIG. 2, tunnel servers 360 and 370 can be used to establish communications between gateway 311 and 321. Thus, a POSITA understood that a request from gateway 311 for services available from gateway 321 is a request for available tunnel servers that can provide a tunnel between gateway 311 and 321. This is consistent with the disclosure of the '595

(Paper 2, Pet., at 46.)

- Conclusory POSITA assertions are improper. (See Paper 23, Sur-reply, at 4 (citing *Innogenetics, N.V. v. Abbott Labs.*, 512 F.3d 1363, 1373 (Fed. Cir. 2008).)

26:55-65. Finally, Hankins discloses that the provisioning of the tunnel servers by the controller via execution of one or more software applications EX-1005 25:20-22. Based on this disclosure, a POSITA understood that Hankins discloses that a request for services from a network device (e.g., content from a server) would be sent from a gateway (endpoint device) to a controller, that would use a software module to provision one or more tunnel servers to connect the gateway with another gateway to receive the services. Thus, Hankins discloses limitation [15.3].

(Paper 22, Reply, at 4.)

- This POSITA assertion is not only absent from in the Petition, it lacks evidentiary support.

Petitioner's Fails to Offer Proper Evidence of the “*List*” in the Prior Art

Petitioner's expert, Dr. Reddy, fails to identify a prior art device *responsive* to “*a request for a list of available servers.*”

139. Hankins discloses that the controller receives a request from a gateway 311 (*endpoint device*) for a list of available servers including tunnel servers (*access point server*). For example, Hankins discloses that gateway 311 in one logical broadcast domain, may request services available from gateway 321 in another logical broadcast domain. *EX-1005* 12:9-16.

140. The controller identifies potential tunnel servers that are available to provide a tunnel between gateway 311 to gateway 321. For example, as shown in Annotated FIG. 2, tunnel servers 360 and 370 can be used to establish communications between gateway 311 and 321. Thus, a POSITA understood that a request from gateway 311 for services available from gateway 321 is a request

for available tunnel servers that can provide a tunnel between gateway 311 and 321. This is consistent with the disclosure of the '595 Patent that when a device makes a request to a server, what it is really asking for is a list of available servers, and specifically an IP and port which it could utilize to in the build a tunnel. *EX-1001* 23:11-24.

- Dr. Reddy fails to cite a disclosure in Hankins describing any device responsive to “*a request for a list of available servers.*”
- Instead, Dr. Reddy relies on the assumed knowledge of a POSITA, but fails to explain why a POSITA would understand a gateway's request for services from another gateway to be a request for available tunnel servers, let alone a *list* of available servers.
- Dr. Reddy's only support for his POSITA assertion is a novel feature of the '595 patent itself, describing a “*server availability module,*” not a server or a generic “request to a server.”

(EX 1003, Reddy Decl., at ¶ 139-140.) DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Petitioner's Only Evidence of the “*List*” in the Prior Art is Not Reliable

- The “***server availability module***” is a novel ‘*advantage*’ offered by the ’595 patent, not something that was already known in the prior art. (See Paper 2, at 45-56; EX 1001, 595 patent, at 23:17-20; Paper 21 at 12; Paper 23 at 7.)

FIG. 32 illustrates a simple example databases schema for relating and storing connectivity information from both tests and production environment usage.

Those experienced in the art will see its advantages. The tables, the fields, labels and other information are provided for reference only and may differ in the various production deployments.

The Server Availability 32-280 module examines current allocation of IP address and port combinations while taking account actual use, load, and other factors of the live production environment. It further analyzes the potential impact of assigned but standby allocations to predict the impact of those resource demands going live.

When a device makes a request to the server availability module, what it is really asking for is a list of available servers, and specifically an IP and port which it could utilize to in the present or in the future to build a tunnel to. Each

(EX 1001, 595 patent, at 23:4-20.)

- The '595 Patent's Background
- Hankins Does Not Disclose “*A List of Available Servers*”:
 - That is Received in Response to an Endpoint’s “*request for a list of available servers*”
 - That an Endpoint Device “Could Utilize to Build a Tunnel”
 - That is “*Dynamically Order[ed] . . . Based on*” the Endpoint
- Kommula Fails as a Prior Art Reference
- The Petition Fails to Address Claims 19 and 20

Petitioner Admits the “*List*” in [15.3] Is at the Stage of Pre-Tunnel Formation, *i.e.*, Provided to Allow Endpoint Devices “To Form Tunnels” Not Yet Built

- Petitioner admits that the “*list of available servers*” recited in [15.3] is to allow endpoints *to form tunnels*.

In another aspect, some of the claims of the '595 Patent are directed to a system in which a controller dynamically orders a list of available access point servers, which can be used to form tunnels between first and second endpoint devices. For example, FIG. 35 of the '595 Patent illustrates the network system

(Paper 2, Pet., at 3.)

The Challenged Claims of Ground 2, Claims 15-18, are very similar to the claims of Ground 1, but are directed to a server that dynamically orders a list of available access point servers which can be used to form tunnels between first and second endpoint devices. Pet. 37-38 (citing EX-1005 25:7-25; 5 24:39-57, 25:15-55). The Petition demonstrates how Hankins, alone or in combination with Kommula discloses Claims 15-18.

(Paper 22, Reply, at 1-2.)

- Thus, the “*one or more first tunnels*” of claim 15—through which “an access point server [is] *configured to connect* to one or more endpoint devices”—***have not yet been formed*** when the first endpoint requests the server availability list from the control server.

Petitioner Says the “*List of Available Servers*” is What Exists Prior to Tunnel Building, *i.e.*, What Endpoints “Could Utilize . . . to Build a Tunnel . . .”

- Here, again, Petitioner acknowledges that requesting “*a list of available servers*,” as recited in [15.3], is really asking for what an endpoint device “could utilize . . . to build a tunnel to the available server.”
- Thus, the list is provided prior to the “*first tunnels*” being built/formed.

The '595 Patent describes that in one embodiment, in order to build a tunnel, certain information is needed. *EX-1001* 10:65-11:3. For example, a server availability module examines current allocation of IP address and port combinations while taking account actual use, load, and other factors of the live production environment. It further analyzes the potential impact of assigned but standby allocations to predict the impact of those resource demands going live. When a device makes a request to the server availability module, what it is really asking for is a list of available servers, and specifically an IP and port which it could utilize to in the present or in the future to build a tunnel to the available server. Each tunnel has specific information which needs to be known to each peer pair in the tunnel building process as defined by tables under tunnel information. *EX-1001* 23:11-24; *EX-1003* ¶ 41.

(Paper 2, Pet., at 8; *see also*, Paper 9, Inst. Dec., at 16 (citing Paper 2 at 46; Ex. 1001, 23:11-24; Ex. 1003 ¶¶ 139, 140); Paper 22, Reply, at 2.)

Petitioner Ignores that the “*List of Available Servers*” are Servers to Which the First Endpoint Has Not Yet Built/Formed a Tunnel

- Hankins is silent as to tunnel forming/building.
- Petitioner does not even attempt to explain when tunnel forming/building occurs in Hankins, *i.e.*, whether a tunnel between its “gateway” and its “tunnel server” *already exists* when the gateway requests services from another gateway.
- Because tunnels to the tunnel servers already exist at the time of a services request, Hankins’s controller will not receive a gateway request for “a list of available access point servers, *which can be used to form tunnels between first and second endpoint devices.*” (Paper 2, Pet., at 3; see also, Paper 22, Reply, at 1-2.)

- The '595 Patent's Background
- Hankins Does Not Disclose “*A List of Available Servers*”:
 - That is Received in Response to an Endpoint’s “*request for a list of available servers*”
 - That an Endpoint Device “Could Utilize to Build a Tunnel”
 - That is “*Dynamically Order[ed] . . . Based on*” the Endpoint
- Kommula Fails as a Prior Art Reference
- The Petition Fails to Address Claims 19 and 20

Element [15.5] Requires a Control Server Configured to “*Dynamically Order the List of Available Servers . . . Based on*” the Context of the Endpoint Device

Claim 15 of the '595 patent recites:

[15.0] A network system for managing a global virtual network, comprising:

[15.1] an access point server configured to connect to one or more endpoint devices via one or more first tunnels; and

[15.2] a control server connected to the access point server via a second tunnel, the control server storing one or more records associated with each of the one or more endpoint devices, wherein ***the control server is configured to:***

[15.3] receive a request for a list of available servers from a first endpoint device among the one or more endpoint devices, the list of available servers including the access point server;

[15.4] retrieve the list of available servers; retrieve the one or more records associated with the first endpoint device;

[15.5] **dynamically order the list of available servers** for the first endpoint device **based on the one or more records** associated with the network device; and

[15.6] transmit the ordered list of available servers to the first endpoint device.

(EX 1001, '595 patent, at 34:4-24.)

The Petition Recognizes that “*the Network Device*” Recited in Element [15.5] is “*the First Endpoint Device*”

[15.0] A network system . . . comprising:
[15.1] an access point server configured to connect to one or more endpoint devices via one or more first tunnels; and
[15.2] a control server . . . storing *one or more records associated with each* of the one or more *endpoint devices*, wherein the control server is configured to:
[15.3] receive a request for a list of available servers from *a first endpoint device* among the one or more endpoint devices, the list of available servers including the access point server;
[15.4] retrieve the list of available servers; retrieve *the one or more records associated with the first endpoint device*;
[15.5] dynamically order the list of available servers for *the first endpoint device based on the one or more records associated with the network device*; and
[15.6] transmit the ordered list of available servers to the first endpoint device.

(EX 1001, '595 patent, at 34:4-24.)

- By mapping the claimed “the network device” and “the first endpoint device” to the same “*gateway device*” of Hankins, the Petition looks past a typographic error in claim 15, and acknowledges that “*the network device*” in [15.5] is “*the first endpoint device*.”

[15.5] *dynamically order the list of available servers for the first endpoint device based on the one or more records associated with the network device; and*

Hankins discloses several examples of this limitation. Hankins discloses

dynamically ordering a list of available tunnel servers for the gateway based on the

one or more records associated with the gateway that provide contextual

information. For example, gateway 311 is provided with a list of tunnel servers

(Paper 2, Pet., at 48; *see also*, Paper 22, POR, at n.7.)

Petitioner's Expert Cites Nothing in the Prior Art for the '595 Patent's “*Custom-Made List*,” Recited as “*Dynamically Order[ed] . . . Based on*,” in Element [15.5]

According to Petitioner's expert, the '595 patent describes the “*server availability module*” as providing “*a custom-made list of IP addresses and ports . . . contextually designed to provide best options for endpoint devices to utilize*”:

48. The '595 Patent describes that the server availability module's main benefit is to catalog the resources usage, IP addresses and ports of access point servers and to look at which devices, such as end point devices that would like to connect to server access points to build tunnels with them. The server availability module calculates and compiles a custom-made list of IP addresses and ports which are contextually designed to provide best options for endpoint devices to utilize. The contextual information includes connectivity tests based on current and

(EX 1003, Reddy Decl., at ¶ 48.)

- This “*custom-made*” list is the element [15.5] “*dynamically order[ed] list of available servers for the first endpoint device based on the one or more records associated with the [first endpoint] device.*”
- Hankins has no such “*custom-made*” or “*contextual*” *list*. (Paper 21, POR, at 13-14.)

Petitioner Offers No Expert Support for a “POSA” Assertion Regarding [15.5]

Regarding element [15.5], Petitioner relies on assertions of what “*a POSA understood*” as to the ordering of tunnel servers in Hankins to improve performance.

packets via tunnel sever two 370. Pet. 48 (citing EX-1005 25:56-67). Based on these disclosures of Hankins, a POSA understood that controller 350 would provision tunnel servers one 360 and two 370 to gateway 311 and 321 with tunnel server one 360 being the primary path and tunnel server two 370 being the secondary path because that achieves one of the express goals of Hankins to “provision tunnel servers in a manner to improve performance and/or reduce costs without human intervention.” EX-1005 25:34-55.

• • •

Pet. 49 (citing EX-1005 26:21-48). Based on these disclosures of Hankins, a POSA understood that controller 350 would provision tunnel servers to the gateways in a specific order to ensure load balancing across the tunnel servers because that achieves one of the express goals of Hankins to “provision tunnel servers in a manner to improve performance and/or reduce costs without human intervention.” EX-1005 25:34-55.

(Paper 22, Reply, at 8 (quoting Ex. 1005, Hankins, at 25:34-55).)

Petitioner Offers No Expert Support for a “POSA” Assertion Regarding [15.5]

Regarding element [15.5], Petitioner relies on assertions of what “*a POSA understood*” as to the ordering of tunnel servers in Hankins to improve performance.

packets via tunnel sever two 370. Pet. 48 (citing EX-1005 25:56-67). Based on these disclosures of Hankins, a POSA understood that controller 350 would provision tunnel servers one 360 and two 370 to gateway 311 and 321 with tunnel server one 360 being the primary path and tunnel server two 370 being the secondary path because that achieves one of the express goals of Hankins to “provision tunnel servers in a manner to improve performance and/or reduce costs without human intervention.” EX-1005 25:34-55.

• • •

Pet. 49 (citing EX-1005 26:21-48). Based on these disclosures of Hankins, a POSA understood that controller 350 would provision tunnel servers to the gateways in a specific order to ensure load balancing across the tunnel servers because that achieves one of the express goals of Hankins to “provision tunnel servers in a manner to improve performance and/or reduce costs without human intervention.” EX-1005 25:34-55.

- These “POSA” assertions are not in the Petition and are not supported by an expert declaration.

(Paper 22, Reply, at 8 (quoting Ex. 1005, Hankins, at 25:34-55).)

Petitioner Offers No Expert Support for a “POSA” Assertion Regarding [15.5]

Regarding element [15.5], Petitioner relies on assertions of what “*a POSA understood*” as to the ordering of tunnel servers in Hankins to improve performance.

packets via tunnel sever two 370. Pet. 48 (citing EX-1005 25:56-67). Based on these disclosures of Hankins, a POSA understood that controller 350 would provision tunnel servers one 360 and two 370 to gateway 311 and 321 with tunnel server one 360 being the primary path and tunnel server two 370 being the secondary path because that achieves one of the express goals of Hankins to “provision tunnel servers in a manner to improve performance and/or reduce costs without human intervention.” EX-1005 25:34-55.

. . .

Pet. 49 (citing EX-1005 26:21-48). Based on these disclosures of Hankins, a POSA understood that controller 350 would provision tunnel servers to the gateways in a specific order to ensure load balancing across the tunnel servers because that achieves one of the express goals of Hankins to “provision tunnel servers in a manner to improve performance and/or reduce costs without human intervention.” EX-1005 25:34-55.

- These “POSA” assertions are not in the Petition and are not supported by an expert declaration.
- These “POSA” assertions show that Hankins does not sufficiently disclose the concept in element [15.5].

(Paper 22, Reply, at 8 (quoting Ex. 1005, Hankins, at 25:34-55).)

Hankins Discloses the Controller Controls Tunnel Server Selection, Rather than the Gateway Selecting from a Customized, “*Dynamically Order[ed] List*”

With respect to element [15.5], Petitioner relies on Hankins’s disclosures of a controller “*provisioning*” a tunnel server for its gateway:

1005 25:34-45; 26:49-27:9. Hankins specifically describes that that the controller “may provision tunnel servers in a manner to improve performance and/or reduce costs without human intervention.” EX-1005 25:34-55. The Petition identifies several examples of Hankins provisioning more than one tunnel server and dynamically ordering the servers. For example, Hankins describes that gateway 311 is provisioned with tunnel server one 360 and tunnel server two 370, based on information obtained from the network devices (gateway, tunnel servers, etc.) such as proximity of the gateway device to tunnel server one 360. Pet. 48 (citing EX-1005 24:39-57, 25:15-55). Normally, when gateway device 311 transmits signal

(Paper 22, Reply, at 7.)

- As noted above, “***provisioning***” (i.e., assigning) a tunnel server ***does not give the first endpoint device a list of available servers*** which it “***could utilize to build a tunnel to the available server.***” (Paper 2, Pet., at 8; *see also, id.* at 3 (“*can be used **to form tunnels***”)); Paper 22, Reply, at 1-2 (same).)

Petitioner's Examples of a Gateway Switching Between Tunnel Servers Are Static, Not Dynamic, and Are Tunnels Already Formed as of the Request

The Petition's next cited example of a Hankins gateway communicating with multiple tunnel servers—to which tunnels were already formed when services were requested—is by a *static* (non-dynamic) scheme of switching between existing tunnels.

By way of another example, Hankins discloses an embodiment where broadcast domains communicate through multiple tunnel servers. Each nth packet goes to the nth tunnel server. Hankins describes that this approach can improve security, fault tolerance, redundancy and/or load balancing of tunnel servers. *EX-1005* 26:21-48. A POSITA understood that providing load balancing and fault tolerance means that a list of tunnel services would be provided in preference of order. *EX-1003* ¶¶ 149, 150.

(Paper 2, Pet., at 49.)

Petitioner's Examples of a Gateway Switching Between Tunnel Servers Are Static, Not Dynamic, and Are Tunnels Already Formed as of the Request

The Petition's next cited example of a Hankins gateway communicating with multiple tunnel servers—to which tunnels were already formed when services were requested—is by a *static* (non-dynamic) scheme of switching between existing tunnels.

By way of another example, Hankins discloses an embodiment where broadcast domains communicate through multiple tunnel servers. Each nth packet goes to the nth tunnel server. Hankins describes that this approach can improve security, fault tolerance, redundancy and/or load balancing of tunnel servers. *EX-1005* 26:21-48. A POSITA understood that providing load balancing and fault tolerance means that a list of tunnel services would be provided in preference of order. *EX-1003* ¶¶ 149, 150.

- This “Nth” scheme does not involve selecting from a list of available servers which *could be used to form a tunnel* to the available server.

(Paper 2, Pet., at 49.)

Petitioner's Examples of a Gateway Switching Between Tunnel Servers Are Static, Not Dynamic, and Are Tunnels Already Formed as of the Request

The Petition's next cited example of a Hankins gateway communicating with multiple tunnel servers—to which tunnels were already formed when services were requested—is by a *static* (non-dynamic) scheme of switching between existing tunnels.

By way of another example, Hankins discloses an embodiment where broadcast domains communicate through multiple tunnel servers. Each nth packet goes to the nth tunnel server. Hankins describes that this approach can improve security, fault tolerance, redundancy and/or load balancing of tunnel servers. *EX-1005* 26:21-48. A POSITA understood that providing load balancing and fault tolerance means that a list of tunnel services would be provided in preference of order. *EX-1003* ¶¶ 149, 150.

- This “Nth” scheme does not involve selecting from a list of available servers which *could be used to form a tunnel* to the available server.
- This tunnel-selection scheme is *neither customized* (based on the first endpoint device), *nor dynamically ordered*.

(Paper 2, Pet., at 49.)

Petitioner's Examples of a Gateway Switching Between Tunnel Servers Are Static, Not Dynamic, and Are Tunnels Already Formed as of the Request

The Petition's next cited example of a Hankins gateway communicating with multiple tunnel servers—to which tunnels were already formed when services were requested—is by a *static* (non-dynamic) scheme of switching between existing tunnels.

By way of another example, Hankins discloses an embodiment where broadcast domains communicate through multiple tunnel servers. Each nth packet goes to the nth tunnel server. Hankins describes that this approach can improve security, fault tolerance, redundancy and/or load balancing of tunnel servers. *EX-1005* 26:21-48. A POSITA understood that providing load balancing and fault tolerance means that a list of tunnel services would be provided in preference of order. *EX-1003* ¶¶ 149, 150.

(Paper 2, Pet., at 49.)

- This “Nth” scheme does not involve selecting from a list of available servers which *could be used to form a tunnel* to the available server.
- This tunnel-selection scheme is *neither customized* (based on the first endpoint device), *nor dynamically ordered*.
- Neither Petitioner nor its expert explains this POSITA assertion re: “*preference of order*” or how it relates to a gateway.

Petitioner's Examples of a Gateway Switching Between Tunnel Servers Are Static, Not Dynamic, and Are Tunnels Already Formed as of the Request

The Petition's final example from Hankins of a gateway communicating with multiple tunnel servers—to which tunnels were already formed when services were requested—is by a static (non-dynamic) scheme of switching between existing tunnels based on an aspect of the server, not the endpoint.

Hankins further describes that different tunnel servers could provide different services. For example, one tunnel server may be better suited for use in file serving and the other tunnel server could be better suited for message serving. Hankins describes that the provisioning of tunnel servers may be generated based on the capability of the tunnel servers and thus a POSITA understood that Hankins orders the availability of the tunnel servers on the needs of the gateways in the broadcast domains. For example, if the needs include file service more than messaging, the tunnel server handling file service would be ranked above the tunnel server for message servicing. *EX-1003* ¶ 151.

(Paper 2, Pet., at 49.)

Petitioner's Examples of a Gateway Switching Between Tunnel Servers Are Static, Not Dynamic, and Are Tunnels Already Formed as of the Request

The Petition's final example from Hankins of a gateway communicating with multiple tunnel servers—to which tunnels were already formed when services were requested—is by a static (non-dynamic) scheme of switching between existing tunnels based on an aspect of the server, not the endpoint.

Hankins further describes that different tunnel servers could provide different services. For example, one tunnel server may be better suited for use in file serving and the other tunnel server could be better suited for message serving. Hankins describes that the provisioning of tunnel servers may be generated based on the capability of the tunnel servers and thus a POSITA understood that Hankins orders the availability of the tunnel servers on the needs of the gateways in the broadcast domains. For example, if the needs include file service more than messaging, the tunnel server handling file service would be ranked above the tunnel server for message servicing. *EX-1003* ¶ 151.

- This specialized-server scheme does not involve selecting from a list of available servers which ***could be used to form a tunnel*** to the available server.

(Paper 2, Pet., at 49.)

Petitioner's Examples of a Gateway Switching Between Tunnel Servers Are Static, Not Dynamic, and Are Tunnels Already Formed as of the Request

The Petition's final example from Hankins of a gateway communicating with multiple tunnel servers—to which tunnels were already formed when services were requested—is by a static (non-dynamic) scheme of switching between existing tunnels based on an aspect of the server, not the endpoint.

Hankins further describes that different tunnel servers could provide different services. For example, one tunnel server may be better suited for use in file serving and the other tunnel server could be better suited for message serving. Hankins describes that the provisioning of tunnel servers may be generated based on the capability of the tunnel servers and thus a POSITA understood that Hankins orders the availability of the tunnel servers on the needs of the gateways in the broadcast domains. For example, if the needs include file service more than messaging, the tunnel server handling file service would be ranked above the tunnel server for message servicing. *EX-1003* ¶ 151.

- This specialized-server scheme does not involve selecting from a list of available servers which *could be used to form a tunnel* to the available server.
- This tunnel-selection scheme is *neither customized* (based on the first endpoint device), *nor dynamically ordered*.

(Paper 2, Pet., at 49.)

Petitioner's Examples of a Gateway Switching Between Tunnel Servers Are Static, Not Dynamic, and Are Tunnels Already Formed as of the Request

The Petition's final example from Hankins of a gateway communicating with multiple tunnel servers—to which tunnels were already formed when services were requested—is by a static (non-dynamic) scheme of switching between existing tunnels based on an aspect of the server, not the endpoint.

Hankins further describes that different tunnel servers could provide different services. For example, one tunnel server may be better suited for use in file serving and the other tunnel server could be better suited for message serving. Hankins describes that the provisioning of tunnel servers may be generated based on the capability of the tunnel servers and thus a POSITA understood that Hankins orders the availability of the tunnel servers on the needs of the gateways in the broadcast domains. For example, if the needs include file service more than messaging, the tunnel server handling file service would be ranked above the tunnel server for message servicing. *EX-1003* ¶ 151.

(Paper 2, Pet., at 49.)

- This specialized-server scheme does not involve selecting from a list of available servers which *could be used to form a tunnel* to the available server.
- This tunnel-selection scheme is *neither customized* (based on the first endpoint device), *nor dynamically ordered*.
- A gateway's ephemeral “*need*” for a particular type service is not a “*store[d] record[] associated with*” the gateway, as recited in claim 15.

- The '595 Patent's Background
- Hankins Does Not Disclose “*A List of Available Servers*”:
 - That is Received in Response to an Endpoint’s “*request for a list of available servers*”
 - That an Endpoint Device “Could Utilize to Build a Tunnel”
 - That is “*Dynamically Order[ed] . . . Based on*” the Endpoint
- Kommula Fails as a Prior Art Reference
- The Petition Fails to Address Claims 19 and 20

Kommula, Directed to a “DNS Server,” Is Not Analogous to the Access Point Server and Tunnel Elements of the Transport Network recited in Claim 15

- The Petition and Reply oversimplify how Hankins and Kommula are analogous art to '595 patent.

other. POR 1-2, 15-17. The Petition sets forth the reason why Kommula and

Hankins are analogous:

Hankins and Kommula are analogous art. Both Hankins and Kommula are directed to identifying available servers to provide a communication path through a network. EX-1005 12:9-16, 24:39-57, 25:15-55; EX-1004 [0008], [0019], [0026], [0027]; EX-1003 ¶ 154. Both Hankins and Kommula use contextual information to identify available servers such as the proximity of the server to the endpoint device or gateway. EX-1005 25:7-25; 26:55-65; EX-1004 [0009]; EX-1003 ¶ 155

Pet. 50.

(Paper 22, Reply, at 10 (citing Paper 2, Pet., at 50.))

Kommula, Directed to a “DNS Server,” Is Not Analogous to the Access Point Server and Tunnel Elements of the Transport Network recited in Claim 15

- The Petition and Reply fail to demonstrate the compatibility of the Kommula DNS server to the claimed “access point servers” to which “tunnels” are connect in the transport network of the ’595 patent, as explained in Patent Owner’s Response.

With respect to Kommula, the facts are clear and admitted by Petitioner.

Kommula is in the DNS server art, not the transport network art. (Pet. at 20-21.)

The “tunnel” configuration required by the claims of the ’595 patent is directed to a

“transport network.” (Ex. 2002 at ¶ 34.) In response to a DNS request from a local

DNS server 30 to DNS proxy 12 (Fig. 1), Kommula orders the IP addresses for

different host servers that could serve an application request. (Id. at 34.) This list

will never include an access point server for an overlay network as required by claim

15, as DNS servers merely resolve URLs for application servers. Claim 15,

however, requires a “tunnel” configuration through which access is provided to the

network via access point servers. (Ex. 2002 at ¶ 34.) The DNS list is merely a list

of application server IP addresses, with no instruction as to how a network access

should be performed to reach those servers.

Kommula Cannot Be Combined with Hankins

- Petitioner says its proposed modification of Hankins to incorporate Kommula is “***not based on physical incorporation of Kommula's DNS server***,” but points to specific physical structures within Kommula to implement the concept of element [15.5] which are not identified in Hankins. (See Paper 23, Sur-Reply, at 11-12.)

device, all as recited in Claim 1. Pet. 1-2 (citing EX-1001 32:49- 61). Kommula discloses a global server load balancing switch that in combination with a domain name server provides IP addresses of available host servers to a client program which requested access to an application hosted on one of the host servers. The available host services are based on a selected set of performance metrics. When the domain name server (DNS) resolves a host name in a query and returns one or more IP addresses, the GSLB switch returns a ranked or weighted list of IP addresses to the requesting computer. Pet. 20-21 (citing EX-1004 [0008]; EX-1003

(Paper 22, Reply, at 1.) ¶ 71). Prior to institution, Patent Owner disclaimed Claims 1-14.

Kommula Cannot Be Combined with Hankins

- Petitioner also undermines its “*not based on physical incorporation*” statement by relying on a physical structure within Kommula to implement the “ranked or weighted list” Petition correlates to the “*dynamically order[ed] list*” in element [15.5]. (See Paper 23, Sur-Reply, at 11-12.)

device, all as recited in Claim 1. Pet. 1-2 (citing EX-1001 32:49- 61). Kommula discloses a global server load balancing switch that in combination with a domain name server provides IP addresses of available host servers to a client program which requested access to an application hosted on one of the host servers. The available host services are based on a selected set of performance metrics. When the domain name server (DNS) resolves a host name in a query and returns one or more IP addresses, the GSLB switch returns a ranked or weighted list of IP addresses to the requesting computer. Pet. 20-21 (citing EX-1004 [0008]; EX-1003 ¶ 71). Prior to institution, Patent Owner disclaimed Claims 1-14.

- Petitioner fails to identify the “**switch**” in Hankins, let alone a “**global server load balancing (GSLB) switch**,” into which these allegedly *non-physical* elements of Kommula could be incorporated to modify Hankins to arrive at claim 15. (Paper 23, Sur-Reply, at 11-12.)

(Paper 22, Reply, at 1.)

- The '595 Patent's Background
- Hankins Does Not Disclose “*A List of Available Servers*”:
 - That is Received *in Response to* an Endpoint’s “*request for a list of available servers*”
 - That an Endpoint Device “*Could Utilize to Build a Tunnel*”
 - That is “*Dynamically Order[ed] . . . Based on*” the Endpoint
- Kommula Fails as a Prior Art Reference
- The Petition Fails to Address Claims 19 and 20

The Petition Fails to Address Claim 19

- The Petition relies on its analysis of claim 7 to challenge the validity of claim 19.

d. Claim 19

The network system of claim 15, the one or more records are retrieved based on a unique device identity associated with the first endpoint device.

Hankins as modified by Kommula discloses this limitation for the same reason as discussed with respect to Claim 7. *EX-1003* ¶ 170.

(Paper 2, Pet., at 53.)

- The Petition's analyses of claims 1 and 7 (alluding to its discussion of limitation [1.3]) only address Kommula. **Neither mentions Hankins**, let alone how or why a POSITA would be motivated to modify Hankins to incorporate the alleged teachings of Kommula to arrive at claim 19.

The Petition Fails to Address Claim 20

- The Petition relies on its analysis of claim 4 to challenge the validity of claim 20.

e. Claim 20

The network system of claim 15, wherein the control server receives tunnel information associated with the one or more first tunnels via the global virtual network, and wherein dynamically ordering the list of available servers is further based on the tunnel information.

Hankins as modified by Kommula discloses this limitation for the same

reason as discussed with respect to Claim 4. *EX-1003* ¶ 171.

(Paper 2, Pet., at 54.)

- The Petition also fails to offer evidence tying claim 4's ordering based on "*anticipated level of demand [for a server]*" with claim 20's ordering based on "*the tunnel information*".

- The Petition's analyses of claims 1 and 4 only address Kommula. **Neither mentions Hankins**, let alone how or why a POSITA would be motivated to modify Hankins to incorporate the alleged teachings of Kommula to arrive at claim 20.

PART 2:

The '505 Patent

- The '505 Patent's Background
- The '505 Patent's an “*End-to-End Tunnel*”
Is Not Disclosed in Hankins

- The '505 Patent's Background
- The '505 Patent's an “*End-to-End Tunnel*”
Is Not Disclosed in Hankins

Background of UMBRA U.S. Patent 10,630,505 (“the ’505 patent”)

- The ’505 patent claims at issue (1-20) resulted from the pioneering efforts of the UMBRA inventors in the 2010s in areas such as the management of a global virtual network where a control server uses a plurality of intermediate access point servers (SRV_APs) to form an end-to-end tunnel, connecting endpoint devices (EPDs) at opposing ends of a wide area network (WAN).
- The improvements overcame problems associated with “badly formed or incorrect routing policies governing middle devices between origin and destination” and other causes of “looping,” which, because of “too many hops and too high of latency,” result in “either time outs or dropped packets.” (Ex. 1001, the ’595 patent, at 26:6-21.)

Background of UMBRA U.S. Patent 10,630,505 (“the ’505 patent”)

- The ’505 patent’s improvements include utilizing multiple SRV_APs over a WAN to peer directly with each other and join intermediate tunnels across the internet so that a client sending data packets through a tunnel via an EPD can view the full transit of packets to a destination as occurring within a single end-to-end tunnel formed to reach an opposing EPD, despite the existence of intermediate SRV_AP hops. (*See* Ex. 1001, the ’595 patent, at 33:43-34:11.)
- “The key point is that the Tapestry of the GVN allows for end to end links of fabrics . . . even though there may be some dissimilar segments at the physical level” (*See id.* at 34:28-32.)

Background of UMBRA U.S. Patent 10,630,505 (“the ’505 patent”)

- According to the unrefuted testimony of Dr. Micah Beck, Patent Owner’s expert, a benefit of the claimed end-to-end tunnel is to avoid the need to un-encode, process, and re-encode tunnel traffic at each intermediate access point:

end-to-end tunnels from point-to-point tunnels. Data packets sent through an end-to-end tunnel travel through intermediate access point servers (e.g., AP-1), but *the inner tunnel headers of the data packets are not un-encoded at the access point servers*. Conversely, data packets sent through two back-to-back point-to-point tunnels exit a first tunnel at an access point server, where the data packets are unencoded and processed before being reencoded for a second tunnel and forwarded.

(EX 2002, Beck Decl., at ¶ 31.)

- The '505 Patent's Background
- The '505 Patent's an “*End-to-End Tunnel*”
Is Not Disclosed in Hankins

Elements [1.3] and [1.4] of the '505 Patent Require Forming and Selecting One of a Plurality of “*End-to-End Tunnels*”

Claim 1 of the '505 patent recites:

[1.0] A network system comprising:

[1.1] a first device;

[1.2] a second device; and

[1.3] *a plurality of intermediate access point servers that form a plurality of end-to-end tunnels connecting the first device and the second device;*
and

[1.4] a control server, wherein the control server receives information from at least one of the plurality of intermediate access point servers, and wherein *the control server selects one of the plurality of end-to-end tunnels for communication between the first and second devices based on the information.*

(EX 1001, '505 patent, at 58:13-24.)

All Claims of the '505 Patent Involve Selecting an “*End-to-End Tunnel*”

Each independent claim of the '505 patent involves forming a plurality of “*end-to-end tunnels*” from a plurality of intermediate access point servers, and selecting one “*end-to-end tunnel*” to connect two network devices.

The invention claimed is:

1. A network system comprising:
a first device;
a second device; and
a plurality of intermediate access point servers that form a plurality of end-to-end tunnels connecting the first device and the second device; and
a control server, wherein the control server receives information from at least one of the plurality of intermediate access point servers, and wherein the control server selects one of the plurality of end-to-end tunnels for communication between the first and second devices based on the information.

(EX 1001, '505 patent, at 58:13-24.)

11. A method comprising:
establishing, by one or more computer processors, a plurality of end-to-end tunnels between a first device and a second device, the plurality of end-to-end tunnels being configured to carry network traffic via a plurality of intermediate access point servers;
receiving, by the one or more computer processors, information from at least one of the plurality of intermediate access point servers; and

selecting, by the one or more computer processors, one of the plurality of end-to-end tunnels for communication between the first and second devices based on the information.

(EX 1001, '505 patent, at 58:59-59:4.)

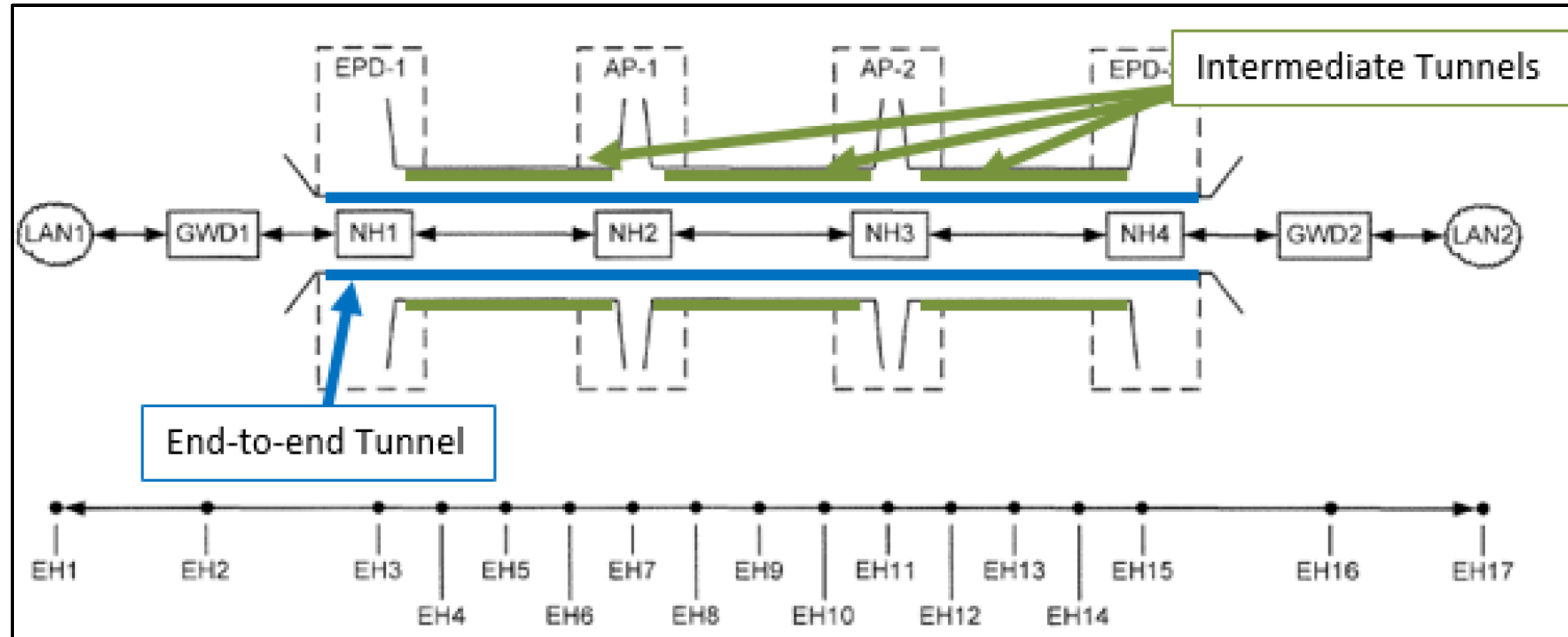
14. A system comprising:
a memory storing instructions; and
one or more processors coupled to the memory, the one or more processors being configured to execute the instructions, the instructions causing the one or more processors to perform operations comprising:
establishing a plurality of end-to-end tunnels between a first device and a second device, the plurality of end-to-end tunnels being configured to carry network traffic via a plurality of intermediate access point servers;
receiving information from at least one of the intermediate access point servers; and

selecting one of the plurality of end-to-end tunnels for communication between the first and second devices based on the information.

(EX 1001, '505 patent, at 59:16-60:3.)

An “Intermediate Tunnel” is Different than an “*End-to-End Tunnel*”

An “*intermediate tunnel*” is different than an “*end-to-end tunnel*”.



(Paper 22, Sur-Reply, at 5 (annotating Fig. 34 of the '505 patent (EX 1001)).)

An “Intermediate Tunnel” is Different than an “*End-to-End Tunnel*”

An “*intermediate tunnel*” is different than an “*end-to-end tunnel*”.

communication path. The communication path may comprise one or more intermediate tunnels connecting each endpoint device to one or more intermediate access point servers and one or more control servers.

(EX 1001, '505 patent, at 4:5-8.)

- The '505 patent refers to a tunnel "connecting [an] endpoint device to" an intermediate access point as an “*intermediate tunnel*”.

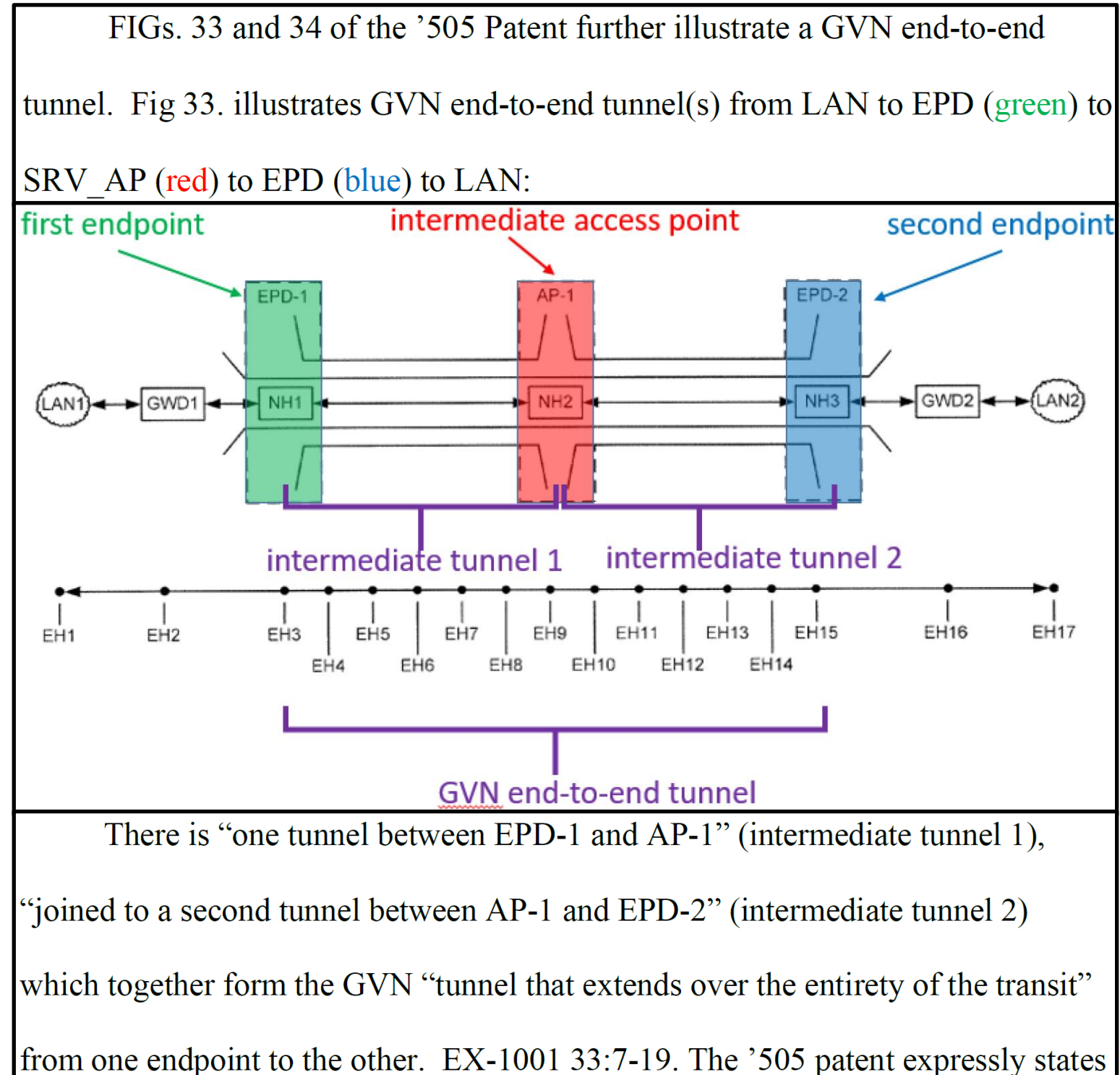
FIG. 34 illustrates a GVN Tunnel from LAN to EPD to SRV_AP to SRV_AP to EPD to LAN. This variation illustrates an end-to-end GVN tunnel(s) between edges of two LANs via two (or possibly more) SRV_APs. The difference

(EX 1001, '505 patent, at 33:25-28.)

- The '505 patent refers to a tunnel running "between edges of two LANs via two [or more] SRV_APs" to connect endpoint devices (EPDs) at each LAN as an “*end-to-end tunnel*”.

An “Intermediate Tunnel” is Different than an “*End-to-End Tunnel*”

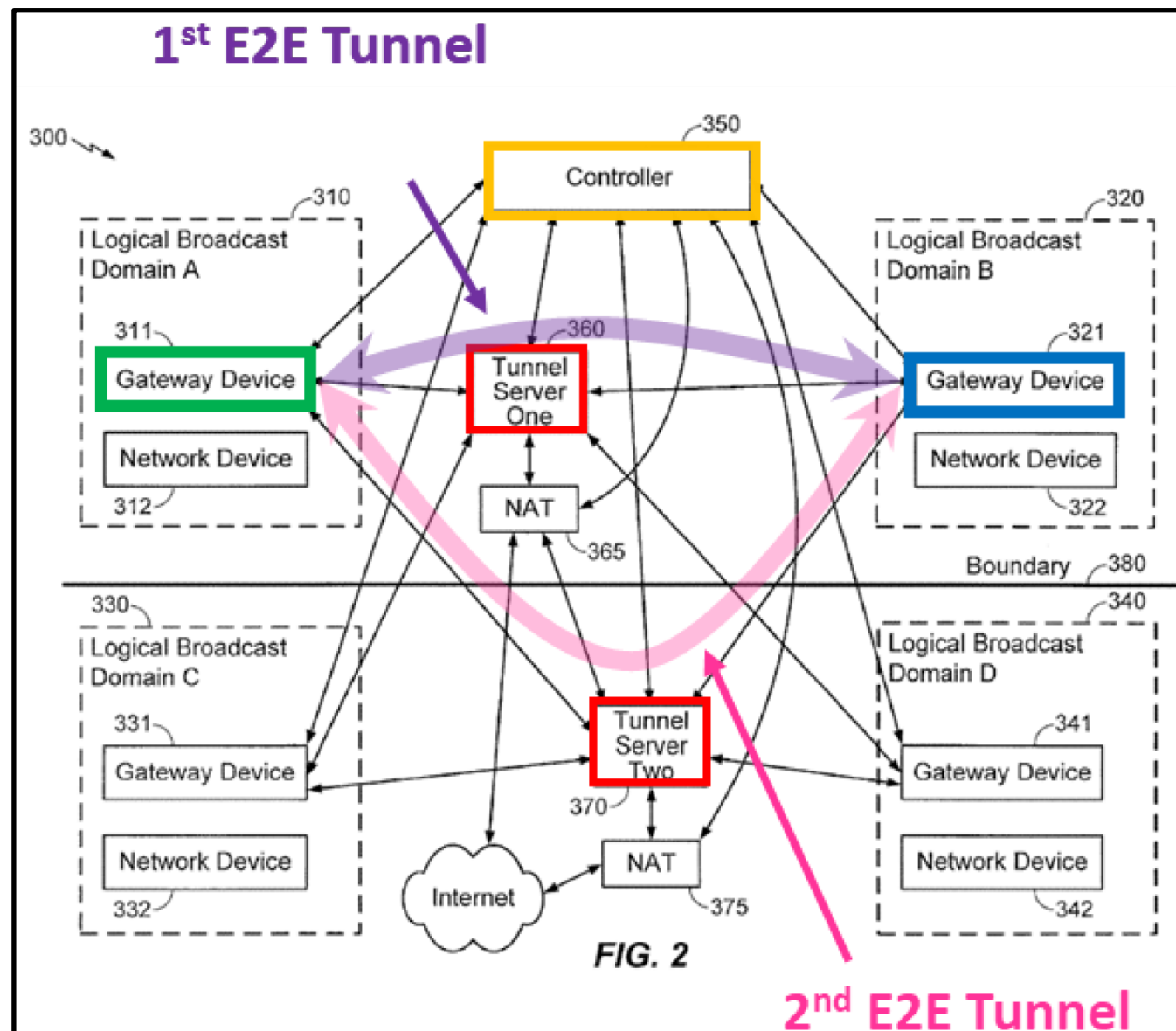
Petitioner acknowledges that an “intermediate tunnel” is different than an “*end-to-end tunnel*”.



(Paper 22, Reply, at 10, (annotating Fig. 33 of '595 patent).)

Two Intermediate Tunnels Do Not Automatically Form an “*End-to-End Tunnel*”

Petitioner assumes that a series of adjacent intermediate tunnels will automatically form an “*end-to-end tunnel*”.



Hankins discloses the system illustrated in FIG. 2 includes a first gateway device 311 (**green**) and a second gateway device 321 (**blue**), a plurality of access point servers including tunnel server one (**red**) and tunnel server two (**red**) that each form an end-to-end tunnel (**purple** and **pink**) between the devices, and a central server controller 350 (**gold**) that selects one of the end-to-end tunnels for communication between the devices based upon information the controller 350 receives from at least one of the tunnel servers. EX-1004 Fig. 2; EX-1003 ¶33.

(Paper 2, Pet., at 5.)

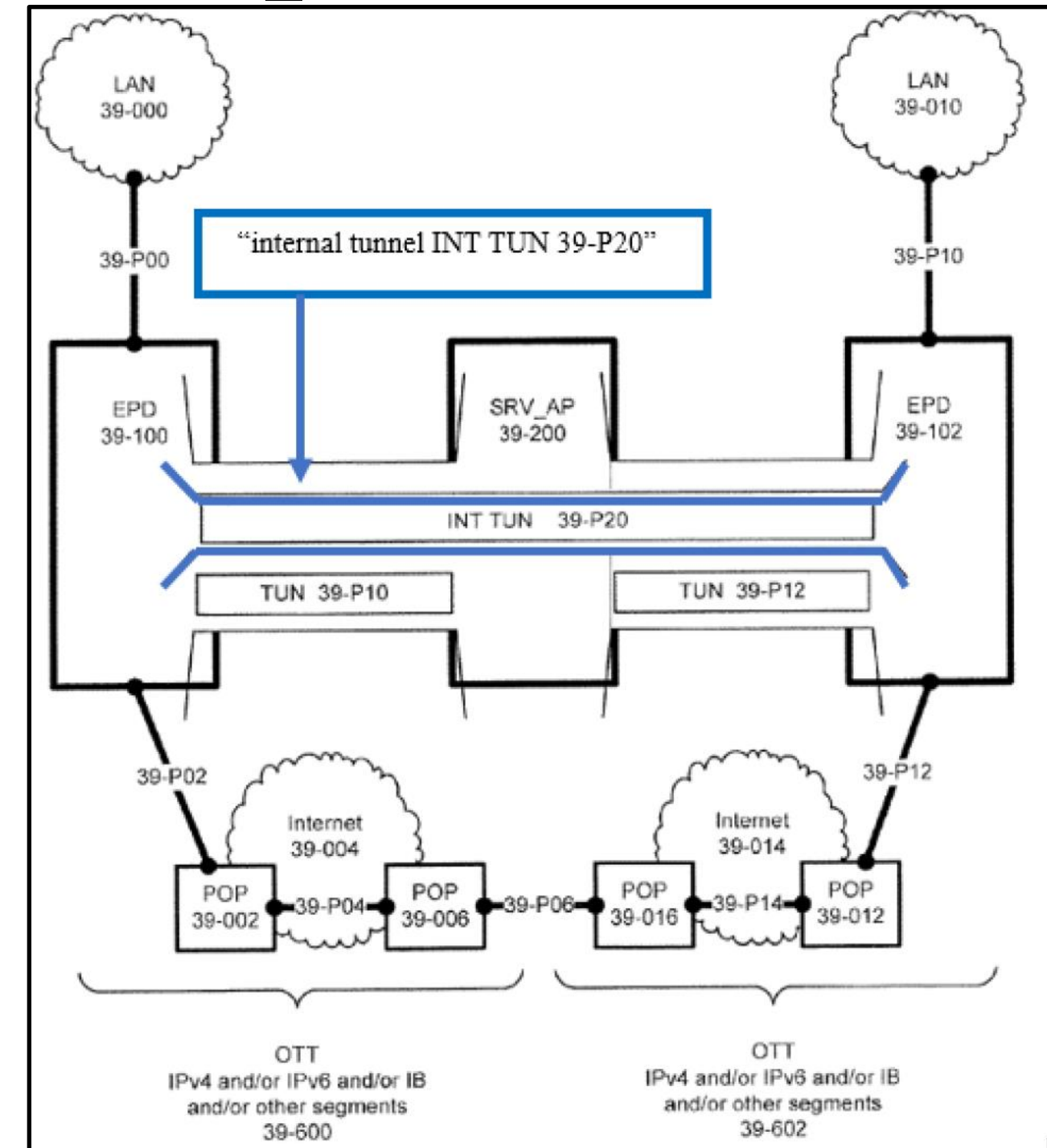
This is incorrect!

Two Intermediate Tunnels Do Not Automatically Form an “*End-to-End Tunnel*”

Petitioner wrongly asserts that the internal, end-to-end tunnel of Figure 39 “*is simply the aggregation of both other tunnels ‘joined at SRV_AP 39-200’*”:

Even the embodiment that the POR relies on to assert a single tunnel cannot be a plurality of tunnels does not support the POR’s argument. FIG. 39 clearly states that internal tunnel 39-P20 is simply the aggregation of both other tunnels ‘joined at SRV AP 39-200.’ EX-1001 36:41-48; EX-1012 ¶16.

(Paper 22, Reply, at 17.)



(Paper 24, Sur-Reply, at 11 (annotating EX 1001, '505 patent, Fig. 39.)

Two Intermediate Tunnels Do Not Automatically Form an “*End-to-End Tunnel*”

Petitioner wrongly asserts that the internal, end-to-end tunnel of Figure 39 “*is simply the aggregation of both other tunnels ‘joined at SRV_AP 39-200’*”:

Even the embodiment that the POR relies on to assert a single tunnel cannot be a plurality of tunnels does not support the POR’s argument. FIG. 39 clearly states that internal tunnel 39-P20 is simply the aggregation of both other tunnels ‘joined at SRV AP 39-200.’ EX-1001 36:41-48; EX-1012 ¶16.

(Paper 22, Reply, at 17.)

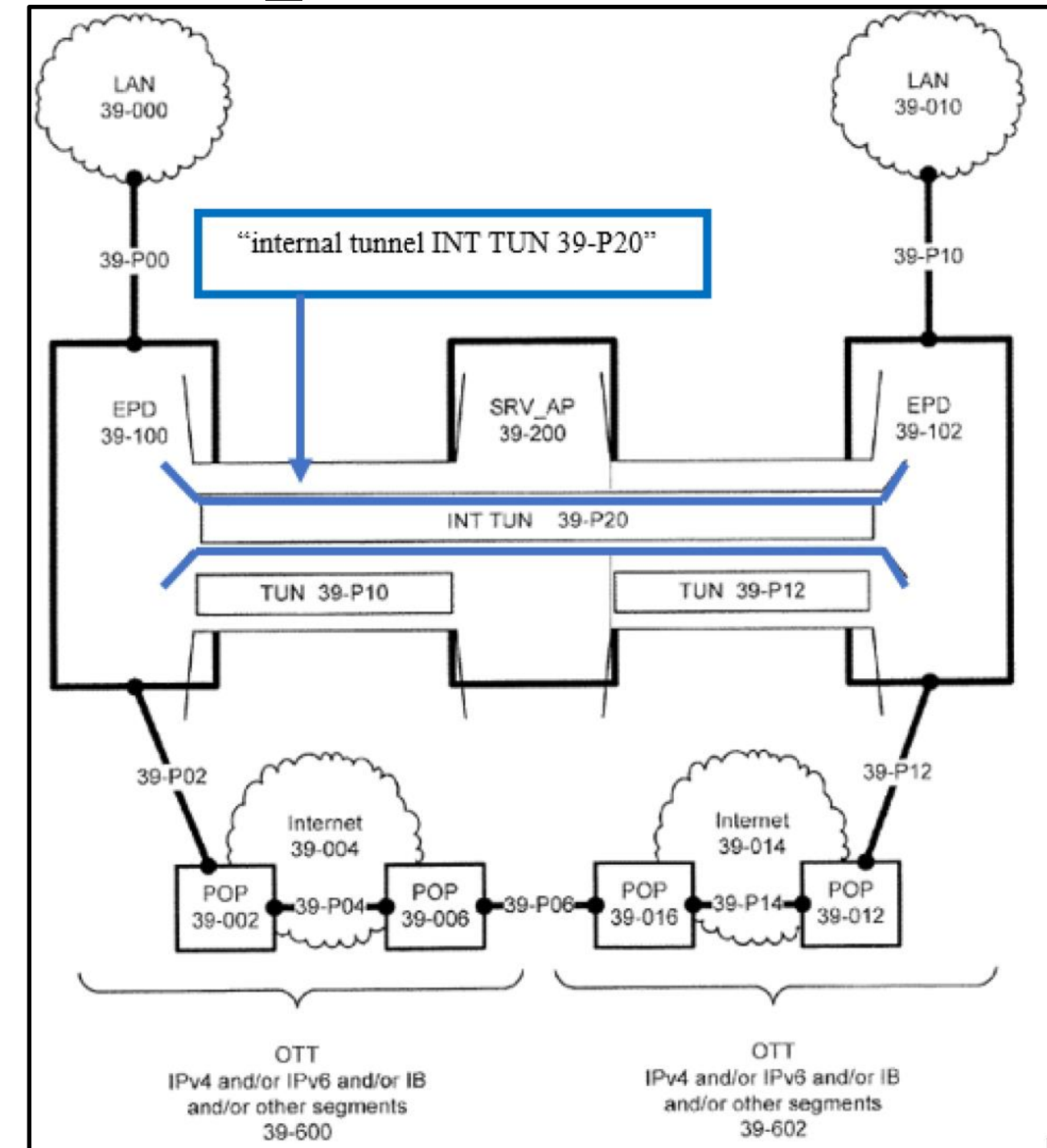
But the ‘595 patent makes clear that the end-to-end tunnel in Figure 39 is optional.

tunnel to extend *within and through the point-to-point tunnels*. The ‘505 patent’s description of the “internal tunnel” in Figure 39 further reinforces that such internal tunnel paths do not automatically arise whenever two intermediate tunnels meet at an access point server (SRV_AP):

EPD 39-100 builds a tunnel TUN 39-P10 between itself and SRV_AP 39-200. EPD 39-102 also builds a tunnel TUN 39-P12 between itself and SRV_AP 39-200. One or both of these tunnels may or may not be encrypted or secured. There can also be another tunnel, internal tunnel INT TUN 39-P20 run through both of the other tunnels, joined at the SRV AP 39-200 through which traffic can flow. This tunnel can be the communications path through which the WAN is built.

(Paper 24, Sur-Reply, at 10.)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE



(Paper 24, Sur-Reply, at 11 (annotating EX 1001, ‘505 patent, Fig. 39.)

An “*End-to-End Tunnel*” Is a “*Singular Tunnel [in] the Neutral Third Layer of the GVN*”

The advantage of the '595 patent is to form an “*end-to-end tunnel*,” which the '595 patent describes as a "singular tunnel" of the "Third Layer of the GVN . . . "

While FIGS. 33, 34 and 35 all illustrate the joining of tunnels at AP hops, this is viewed as a single tunnel by client devices within LAN1 and LAN2. This singular tunnel represents the neutral Third Layer of the GVN within which it is possible to run all traffic that would normally transit over the internet, including TCP, UDP, and other protocols, plus other tunnels such as IPSec, OpenVPN, PPTP, or other.

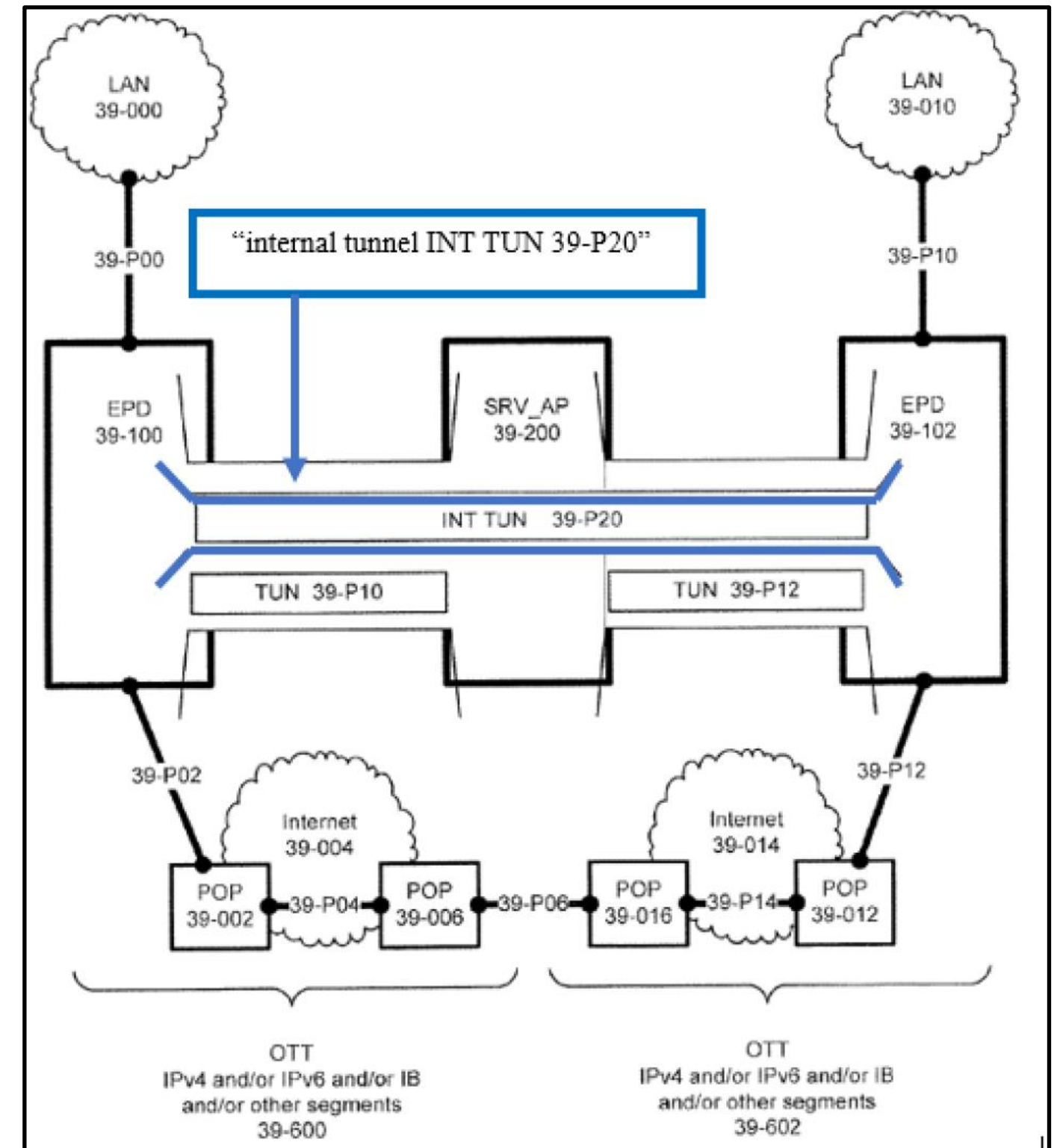
(EX 1001, '505 patent, at 34:1-7.)

Two Intermediate Tunnels Do Not Automatically Form an “*End-to-End Tunnel*”

Petitioner’s expert, Dr. Reddy, even admits that a series of adjacent intermediate tunnels do not automatically form a single “*end-to-end tunnel*”

18 Q. Okay. So you could have tunnels 39P10
19 and 39P12 without necessarily having tunnel 39P20?
20 A. Absolutely.

(Ex. 2003, 2025-03-13 Reddy Dep., 68:18-20;
see also, Paper 24, Sur-Reply, at 14.)



(Paper 24, Sur-Reply, at 11 (annotating
EX 1001, '505 patent, Fig. 39.)

Two Intermediate Tunnels Do Not Automatically Form an “*End-to-End Tunnel*”

Petitioner fails to meet its burden to show why the intermediate tunnels of Hankins automatically form a single “*end-to-end tunnel*.”

to do so. The burden remains on Petitioner. Thus, Petitioner must reconcile why the '505 patent describes and illustrates a difference between (i) two intermediate point-to-point tunnels that terminate at a same access point and (ii) a path that is a single, end-to-end tunnel running “*on top of*” (*i.e.*, “*within*”) those intermediate tunnels. Not only has Petitioner failed to address this distinction, it fails to identify prior art showing the latter rather than merely the former.

(Paper 24, Sur-Reply, at 17.)

Two Intermediate Tunnels Do Not Automatically Form an “*End-to-End Tunnel*”

According to the unrefuted testimony of Dr. Micah Beck, Patent Owner’s expert, packets traversing the ’595 patent’s an end-to-end tunnel “***are not un-encoded at the access point servers.***”

end-to-end tunnels from point-to-point tunnels. Data packets sent through an end-to-end tunnel travel through intermediate access point servers (e.g., AP-1), but *the inner tunnel headers of the data packets are not un-encoded at the access point servers*. Conversely, data packets sent through two back-to-back point-to-point tunnels exit a first tunnel at an access point server, where the data packets are unencoded and processed before being reencoded for a second tunnel and forwarded.

By contrast, packets traversing back-to-back point-to-point tunnels will be unencoded then reencoded at the access point server.

(EX 2002, Beck Decl., at ¶ 31.)

Two Intermediate Tunnels Do Not Automatically Form an “*End-to-End Tunnel*”

Petitioner fails to explain (nor can it) how signal packets traversing the Hankins tunnel server by having encoding (e.g., “***encapsulation***”) removed at the mid-point “tunnel server”, and then reapplied (“***re-encapsulate[d]***”) for forwarding, could be considered an “***end-to-end tunnel***” rather than back-to-back intermediate tunnels. (Paper 24, Sur-Reply, at 17-18.)

However, in this embodiment, a tunnel server may likewise remove encapsulation (e.g., referred to as termination) to determine where to forward a signal packet and re-encapsulate it for forwarding. It is noted that the encapsulation is

(EX 1004, Hankins, at 13:12-15.)

encapsulated. However, in an embodiment, a tunnel server may remove encapsulation (e.g., referred to as termination) to determine where to forward a signal packet and re-encapsulate it for forwarding so that it may be routed via intermediate network devices. Likewise, in an embodiment,

(EX 1004, Hankins, at 30:7-11.)

Two Intermediate Tunnels Do Not Automatically Form an “*End-to-End Tunnel*”

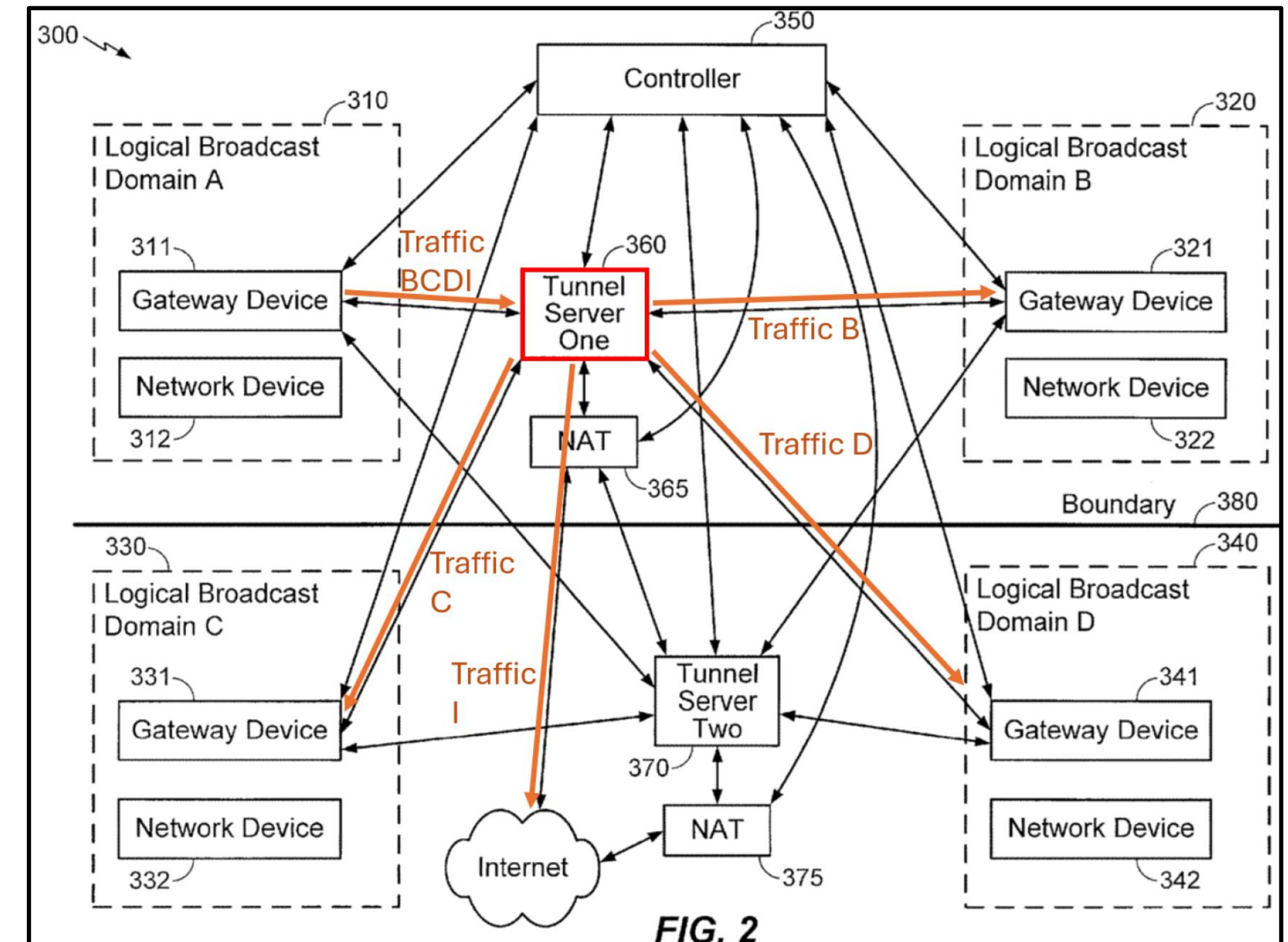
Petitioner’s theory that Hankin’s intermediate tunnels necessarily join to form an *end-to-end* tunnel ignores the various paths a packet may take at the other end of a Hankins’ tunnel server.

And Dr. Reddy certainly does not explain why the arrows Petitioner relies on to indicate traffic arriving at a Hankins tunnel server in a tunnel must leave on a specifically paired/joined tunnel coming out of that tunnel server. Notably, Hankins shows six different double-sided arrows pointing to/from each tunnel server, but nowhere suggests that a pair of these arrows will get joined to the exclusion of all other arrows. (Ex. 1004 at Fig. 2.) As described in more detail

... (the three other gateways and the Internet). Petitioner has not, and cannot, explain how the two connections can be dedicated to reaching only one of those four destinations. Hankins unequivocally states that the connection to either tunnel server can be used to reach any of the four destinations: “... four logical broadcast domains may be in communication via more than one tunnel server.”

(Ex. 1004 at 24:45-47.) Accordingly, that connection cannot be part of an “end-to-

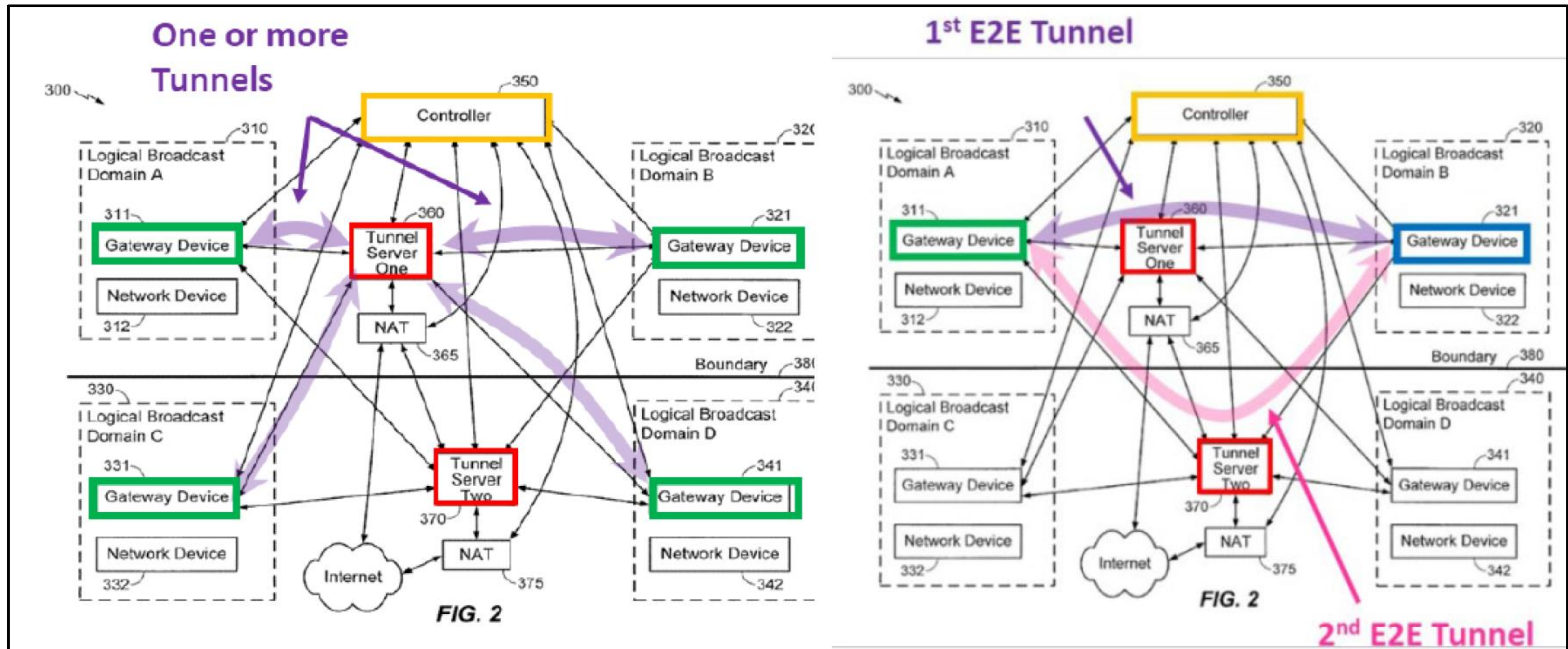
(Paper 24, Sur-Reply, at 18, 21.)



(Paper 24, Sur-Reply, at 24 (annotating EX 1004, Hankins, Figure 2.)

Two Intermediate Tunnels Do Not Automatically Form an “*End-to-End Tunnel*”

Petitioner’s annotated tunnel labelling of Hankins has been inconsistent between the IPRs on the ’595 patent and ’505 patent.



(See IPR2024-0027 (’595) Paper 23, Sur-Reply, at 8, (comparing IPR2024-00270 (’595) Paper 24, Paper 2, Pet., at 42, with IPR-2024-00497 (’505) Paper 22, Reply, at 21).)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Thank You

CERTIFICATE OF SERVICE

The undersigned certifies that the foregoing document was served on June 3, 2025, in its entirety by filing this document through the Patent Trial and Appeal Case Tracking System (P-TACTS) as well as providing a courtesy copy via e-mail to the following attorneys of record for the Petitioner listed below:

LEAD COUNSEL	BACKUP COUNSEL
Patrick D. McPherson DUANE MORRIS LLP 901 New York Ave N.W. Suite 700 East Washington, DC 20001 Tel: 202-776-5214 PDMcPherson@duanemorris.com	Patrick Muldoon Joseph Powers D. Joseph English DUANE MORRIS LLP 901 New York Avenue N.W. Suite 700 East Washington, DC 20001 Tel: 202-776-5214 PCMuldoon@duanemorris.com JAPowers@duanemorris.com DJEnglish@duanemorris.com

/ James Lennon /
James Lennon