UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

PALO ALTO NETWORKS, INC.,
Petitioner,

v.

QOMPLX LLC,
Patent Owner.

Case IPR2026-00362
Patent 11,539,663

PATENT OWNER'S PRELIMINARY RESPONSE

TABLE OF CONTENTS

Page

I. INTRODUCTION 1

II. THE PETITION FAILS TO SHOW A REASONABLE LIKELIHOOD OF PREVAILING FOR ANY CHALLENGED CLAIM (ALL CLAIMS, ALL GROUNDS). 4

A. Petitioner Fails To Prove That Its Combination Discloses Or Renders Obvious A "Midserver." (All Claims). 4

1. The '663 Patent And Its Claims Center Around A "Midserver." 6

2. "Midserver" Is Lexicographically Defined To Require "A Server … Which Runs One Or More Containerized Services[.]" 7

3. Petitioner Fails To Apply The Lexicographic Definition Or Demonstrate That The Server/Hardware Running Maturana's Cloud Agent "Runs One Or More Containerized Services[]." 9

B. Petitioner Fails To Prove That Its Combination Discloses Or Renders Obvious "A Single Data Stream" (All Claims). 13

C. Petitioner Fails To Prove That Its Combination Discloses Or Renders Obvious "establish[ing] a secure connection with an external network" (All Claims). 17

1. Petitioner Relies Upon Maturana's Physical Device Address Methodology For The "establish[ing] a secure connection" Limitation. 19

2. The Combined System Relies On Virtualization That Breaks Maturana's Hardware-Specific Device Identification. 22

3. Petitioner Fails To Demonstrate That The Maturana/Kashyap Combined System "establishes a secure connection." 25

D. Petitioner Fails To Demonstrate That A POSITA Would Be Motivated To Combine Maturana/Kashyap. 28

1. Petitioner Fails To Demonstrate Motivation Because The Proposed Combination Undermines Maturana's Means Of Establishing A Secure Connection. 29

2. Petitioner's Asserted Motivation Seeks To Achieve The Very Condition Maturana Is Designed To Prevent. 31

III. CONCLUSION 33

TABLE OF AUTHORITIES

Page(s)

COURT DECISIONS

Alnylam Pharms., Inc. v. Moderna, Inc.,
138 F.4th 1326 (Fed. Cir. 2025) 8

Arctic Cat Inc. v. Bombardier Recreational Prods.,
876 F.3d 1350 (Fed. Cir. 2017) 30

Arctic Cat Inc. v. Polaris Indus.,
795 F. App'x 827 (Fed. Cir. 2019) 30

In re Gordon,
733 F.2d 900 (Fed. Cir. 1984) 30

Henny Penny Corp. v. Frymaster LLC,
938 F.3d 1324 (Fed. Cir. 2019) 30

In re ICON Health & Fitness, Inc.,
496 F.3d 1374 (Fed. Cir. 2007) 30

Intelligent Bio-Sys., Inc. v. Illumina Cambridge Ltd.,
821 F.3d 1359 (Fed. Cir. 2016) 13

In re Magnum Oil Tools Int'l, Ltd.,
829 F.3d 1364 (Fed. Cir. 2016) 15

ParkerVision, Inc. v. Vidal,
88 F.4th 969 (Fed. Cir. 2023) 8

Plas-Pak Indus. v. Sulzer Mixpac AG,
600 F. App'x 755 (Fed. Cir. 2015) 30

In re Ratti,
270 F.2d 810 (CCPA 1959) 30

Sinorgchem Co. Shandong v. Int'l Trade Comm'n,
511 F.3d 1132 (Fed. Cir. 2007) 8

Tec Air, Inc. v. Denso Mfg. Mich., Inc.,
192 F.3d 1353 (Fed. Cir. 1999) 30

AGENCY DECISIONS

Garmin Int'l, Inc. v. Logantree, LP,
IPR2018-00564, Paper 25 (Aug. 2, 2019) 25

Xerox Corp. v. Bytemark, Inc.,
IPR2022-00624, Paper 9 (Aug. 24, 2022) (precedential).....15

REGULATIONS

37 C.F.R. § 42.104(b)(3).....13

37 C.F.R. § 42.104(b)(4).....13

37 C.F.R. § 42.65(a).....15

EXHIBIT LIST	
2001	Declaration of Nathan Lowenstein in Support of Notice of Intent
2002	Declaration of Colette Woo in Support of Notice of Intent
2003	Decision Denying Institution, *Microsoft Corp. v. Qomplx LLC*, IPR2026-00183, Paper 17 (June 12, 2026) [-00183 DDI]
2004	Defendant Palo Alto Networks, Inc.'s Preliminary Invalidity Contentions, *Qomplx LLC v. Palo Alto Networks, Inc.*, No. 2:25-cv-00913 (E.D. Tex.) (served Jan. 12, 2026) [Preliminary Invalidity Contentions]
2005	Petition for Inter Partes Review of U.S. Patent No. 11,539,663, *Microsoft Corp. v. Qomplx LLC*, IPR2026-00183, Paper 2 (filed Jan. 14, 2026) [-00183 Pet.]
2006	Declaration of Sam Malek, Ph.D.
2007	*What is a data stream?*, TENCENT CLOUD TECHNICAL ENCYCLOPEDIA, https://www.tencentcloud.com/techpedia/108341 (accessed Feb. 23, 2026) [Tencent-Encyclopedia]
2008	*What is data streaming?*, SPOTFIRE GLOSSARY, https://www.spotfire.com/glossary/what-is-data-streaming [Spotfire-Glossary]
2009	*What is Virtualization?*, AMAZON WEB SERVICES, https://aws.amazon.com/what-is/virtualization/ [AWS-Virtualization]
2010	Stephanie Susnjara & Ian Smalley, *What is Virtualization?*, IBM https://www.ibm.com/think/topics/virtualization [IBM-Virtualization]

2011	*What is a Hypervisor? The Ultimate Guideto Virtualization and Virtual Machine Monitors* (SC Insights), SCALE COMPUTING (Dec. 11, 2024), https://www.scalecomputing.com/resources/what-is-a-hypervisor [ScaleComputing-Hypervisor]
2012	*Understanding MAC Addresses: A Beginner's Guide* (Technnology Help Desk), UNIVERSITY OF PITTSBURGH, https://services.pitt.edu/TDClient/33/Portal/KB/ArticleDet?ID=1773 [Pitt-MAC-Address];
2013	*Virtual Machine MAC Address*, BROADCOM (last updated Dec. 19, 2024), https://techdocs.broadcom.com/us/en/vmware-cis/vsphere/vsphere-sdks-tools/6-5/web-services-sdk-programming-guide/virtual-machine-configuration/configuring-a-virtual-machine/networks-1/virtual-machine-mac-address.html [Virtual Machine MAC Address]
2014	*Windows Hyper-V server has a default limit of 256 dynamic MAC addresses*, MICROSOFT (last updated Feb. 12, 2016), https://learn.microsoft.com/en-us/troubleshoot/windows-server/virtualization/default-limit-256-dynamic-mac-addresses [Windows Hyper-V server)
2015	*Scenario: Manage the network configuration of virtual machines*, MICROSOFT (last updated Aug. 30, 2016), https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2012-r2-and-2012/dn425042(v=ws.11) [Scenario: Manage the Network Configuration of Virtual Machines]
2016	*MAC Address Assignment from vCenter Server* (VMware vSphere 7.0 Technical Documentation), BROADCOM (last updated Jan. 19, 2026), https://techdocs.broadcom.com/us/en/vmware-cis/vsphere/vsphere/7-0/vsphere-networking/mac-addresses/mac-address-generation.html [MAC Address Assignment]
2017	Microsoft's Opening Claim Construction Brief, *Qomplx LLC v. Microsoft Corp.*, No. 1:25-cv-01383 (W.D. Tex.) (Mar. 23, 2026) [Microsoft Opening Claim Construction Brief]

2018	Qomplx's Responsive Claim Construction Brief, *Qomplx LLC v. Microsoft Corp.*, No. 1:25-cv-01383 (W.D. Tex.) (Apr. 13, 2026) [Qomplx Responsive Claim Construction Brief]

All emphases in quotations are added unless otherwise stated.

I. INTRODUCTION

The Director has already considered the ground presented in Petitioner's "copy-cat" Petition and denied institution on the merits. The Petition seeks to join Microsoft's IPR2026-00183. On June 12, 2026, the Director denied institution in IPR2026-00183 based on "review of the merits," not discretionary considerations. Ex. 2003 [-00183 DDI] 2. PAN admits that its "petition is a 'copycat' of Microsoft's petition." Pet., 2. PAN identifies no materially different argument or evidence that could justify departing from the Director's determination, much less reaching the opposite result. Thus, for the same reasons as in IPR2026-00183, the Board should reject this Petition.

Petitioner advances a single obviousness ground based on the combination of Maturana (Ex. 1004) and Kashyap (Ex. 1005). Petitioner fails to demonstrate a reasonable likelihood of prevailing over this ground for at least three independently sufficient reasons.

First, Petitioner never shows that its combination discloses or renders obvious the claimed "midserver." The '663 patent, to a significant degree, centers on a "midserver," and that term is expressly and lexicographically defined in a section titled "Definitions":

> "***Midserver***" as used herein ***means a server*** that functions as an interface between an external network and a cloud-based service, and

> ***which runs one or more containerized services*** that perform one or more of: collecting, aggregating, analyzing, filtering, transforming, and securely transmitting data.

Ex. 1001 ['663] 12:66-13:4. Under this definition, a "midserver" must, *inter alia*, "run[] one or more containerized services." Petitioner asserts that the server/hardware platform executing Maturana's cloud agent is a "midserver" (Pet., 40) but does not apply this express definition or even attempt to show that this server "runs one or more containerized services," as the definition requires. Indeed, when Petitioner and its expert discuss this definition—and even then only in a section providing an overview of the '663 patent—they actively omit the requirement that the "midserver" "run[] one or more containerized services." *Compare* Pet., 18 and Ex. 1003 [Crovella-Decl.] ¶ 45 *with* Ex. 1001, 12:66-13:4; *see also* Section II.A.3, *infra* (including comparison table demonstrating omission). Because Petitioner has not shown that a lexicographically defined term is disclosed or rendered obvious by its proposed combination, institution should be denied. *See* Section II.A.

Second, Petitioner fails to prove that its ground discloses or renders obvious the limitation requiring the midserver "retransmit[] the received data … as a single ***data stream***." Petitioner contends this limitation is satisfied by Maturana's aggregation of data into a single packet that is uploaded to the cloud. Petitioner,

however, does not demonstrate that uploading a single ***packet*** qualifies as retransmitting the data as a "single data ***stream***." Nor does Petitioner identify any intrinsic or extrinsic evidence supporting that implicit assumption. At a minimum, nothing in the record suggests that a single upload packet, standing alone, would satisfy this limitation. Petitioner instead offers only conclusory assertions, which are insufficient to carry its burden. *See* Section II.B.

Third, Petitioner does not demonstrate that its proposed Maturana/Kashyap combination "establish[es] a secure network connection" as required by the claims. Petitioner relies on Maturana's handshaking methodology to argue that the combination discloses or renders obvious "establish[ing] a secure connection." That methodology depends on verifying the identity of the particular physical device attempting to access the cloud platform through its device address. In the proposed combination, however, Petitioner implements Maturana's cloud agent as a virtual appliance under Kashyap. In such a virtual environment, device identifiers such as MAC addresses cannot be relied upon to identify a particular device because they are assigned by software and changeable rather than tied to a particular physical machine. Petitioner never explains how Maturana's device-specific authentication process could function in that modified system or establish the claimed secure connection. *See* Section II.C.

Finally, Petitioner fails to demonstrate that a POSITA would have been

motivated to make the proposed combination. The proposed combination would eliminate Maturana's means of establishing a secure connection and is premised on enabling the very condition—movement of the cloud agent across devices—that Maturana is designed to detect and prevent. *See* Section II.D.

Because, as the Director has already determined, Petitioner's sole ground fails to demonstrate that the proposed combination discloses or renders obvious multiple independent claim limitations, and because a POSITA would not be motivated to make such a combination, Petitioner has not shown a reasonable likelihood of prevailing on any challenged claim. Institution should therefore be denied as it was in IPR2026-00183.

II. THE PETITION FAILS TO SHOW A REASONABLE LIKELIHOOD OF PREVAILING FOR ANY CHALLENGED CLAIM (ALL CLAIMS, ALL GROUNDS).

A. Petitioner Fails To Prove That Its Combination Discloses Or Renders Obvious A "Midserver" (All Claims).

Both challenged claims require a "midserver." Indeed, to a significant degree, the '663 patent centers around a "midserver." *See* Section II.A.1. There can be no question what a "midserver" is because it is lexicographically defined in the '663 patent. Indeed, in a section titled "Definitions," the '663 patent states that:

> "Midserver" as used herein means a server that functions as an interface between an external network and a cloud-based service, and ***which runs***

> ***one or more containerized services*** that perform one or more of: collecting, aggregating, analyzing, filtering, transforming, and securely transmitting data.

Ex. 1001 ['663] 12:66-13:4. Thus, a "midserver" is, *inter alia*, "a server … [that] runs one or more containerized services …" *See id.*; *see also* Section II.A.2.

Petitioner does not apply this governing definition when mapping the "midserver" limitation. In its Maturana/Kashyap combination, Petitioner contends that the server/hardware platform executing Maturana's cloud agent is a "midserver." *See* Pet., 40. Petitioner, however, does not attempt to show that this server/hardware "runs one or more containerized services," much less containerized services that "perform one or more of: collecting, aggregating, analyzing, filtering, transforming, and securely transmitting data." Indeed, Petitioner not only fails to apply the lexicographic definition; when it discusses that passage, it omits the portion requiring the server to run "one or more containerized services." *See* Section II.A.3. Nor does Petitioner contend—let alone demonstrate—that Maturana's system inherently runs containerized services or that it would have been obvious to modify Maturana to do so.

Because Petitioner has not shown that a required element is disclosed or rendered obvious by its combination, it has not demonstrated a reasonable likelihood of prevailing. Institution should therefore be denied.

1. The ’663 Patent And Its Claims Center Around A “Midserver.”

The ’663 patent addresses challenges associated with collecting and transmitting data from large business enterprises to cloud-based services. As described in its “Discussion of the State of the Art,” large enterprises may have thousands of computing devices and peripherals—many of which are mobile—sending heterogeneous data over separate connections. Ex. 1001 [’663] 7:12-42. This environment can create numerous problems, including unreliable data collection methods, poor standardized support for connection-oriented protocols, security risks from unfiltered or poorly filtered data, bandwidth concerns from constantly streaming data, and systems that did not scale well when adding new data sources or performing flexible ad hoc queries. *Id.*, 7:22-42. Existing approaches also resulted in excessive data transfer, lack of context for data and data sources, and expensive, disorganized log management. *Id.*

The inventors addressed these challenges in part by introducing a midserver to collect, aggregate, analyze, transform, and securely transmit data between the enterprise environment and the cloud. *Id.*, 7:60-66. The midserver is not incidental; it is a key player in the operation of the disclosed systems and is required in each of the claims. *See, e.g.*, *id.*, Title, Abstract, 7:60-8:33 (summary of the invention), 8:38-9:24 (describing Figures 1-8), cls., 1-6. The specification and claims make clear that the invention is directed to systems and methods built

around a midserver.

2. <u>"Midserver" Is Lexicographically Defined To Require "A Server … Which Runs One Or More Containerized Services[.]"</u>

The "midserver," which lies at the heart of the '663's inventions, is lexicographically defined in a section titled "Definitions":

Definitions

"Bastion host" as used herein means a computer that is deliberately exposed on a public network as the primary or only node or the network exposed to the outside world. A bastion host processes and filters all incoming traffic and prevents malicious traffic from entering the network.

"Ingestion" as used herein means the transfer of data into a cloud-based service.

"Midserver" as used herein means a server that functions as an interface between an external network and a cloud-based service, and which runs one or more containerized services that perform one or more of: collecting, aggregating, analyzing, filtering, transforming, and securely transmitting data. A midserver may also be configured as a bastion host.

See Ex. 1001 ['663] 12:58-13:5. The '663 patent, thus, defines "midserver" as follows:

> "Midserver" as used herein means ***a server*** that functions as an interface between an external network and a cloud-based service, and ***which runs one or more containerized services*** that perform one or more of: collecting, aggregating, analyzing, filtering, transforming, and securely transmitting data. A midserver may also be configured as a bastion host.

See Ex. 1001 ['663] 12:66-13:5.

Here, the patentee has acted as its own lexicographer. "Midserver" appears in quotation marks, in a section titled "Definitions," and is defined using the phrase "as used herein." Each of these features is recognized by the Federal Circuit as

strong evidence of lexicography. *Alnylam Pharms., Inc. v. Moderna, Inc.*, 138 F.4th 1326, 1333 (Fed. Cir. 2025) (term lexicographically defined where "the sentence in question appears under the title 'Definitions.'"); *id.* (finding term lexicographically defined where "the term to be defined [] is set off in quotation marks."); *see also Sinorgchem Co. Shandong v. Int'l Trade Comm'n*, 511 F.3d 1132, 1136 (Fed. Cir. 2007) ("The term . . . is set off by quotation marks—often a strong indication that what follows is a definition."); *ParkerVision, Inc. v. Vidal*, 88 F.4th 969, 976 (Fed. Cir. 2023) (the term "as used herein" generally "conveys an intent for [that sentence] to be definitional."). [1]

[1] In Microsoft's district court action—but not in IPR2026-00183 or the instant Petition—Microsoft argues that a claim amendment negates the lexicographic definition of "midserver." Microsoft, however, identifies no statement in the prosecution history disclaiming or redefining the term. *See* Ex. 2017 [Microsoft Opening Claim Construction Brief] 6-11; Ex. 2018 [Qomplx Responsive Claim Construction Brief] 3-10. Microsoft cites no authority supporting its contention that the mere removal of claim language—without any accompanying disclaimer—precludes a lexicographic definition. And here, the removed language simply tracked the specification's definition. Where a term is

Thus, to qualify as a "midserver," a server must "run[] one or more containerized services."

3. Petitioner Fails To Apply The Lexicographic Definition Or Demonstrate That The Server/Hardware Running Maturana's Cloud Agent "Runs One Or More Containerized Services[]."

As explained above, to qualify as a "midserver," a server must "run[] one or more containerized services." Ex. 1001 ['663] 12:66-13:4. Petitioner does not apply the lexicographic definition in mapping the "midserver" limitation. Instead, Petitioner expressly states that it "applies the plain and ordinary meaning of all claim terms below." Pet., 21; *see also id.*, 40 (addressing "midserver" limitation without mentioning or applying lexicographic definition).

Petitioner contends that the server/hardware platform executing Maturana's cloud agent "is a midserver" because it allegedly performs certain functions that align with descriptions of the midserver in the '663 patent's specification:

---

expressly defined, it would be redundant and confusing to restate that definition in the claim itself. Eliminating such duplicative language reflects routine claim-drafting practice, not any intent to alter claim scope. Microsoft's position would penalize the patentee for avoiding redundancy and confusion. In any event, this argument was readily foreseeable and could have been raised in the Petition, where Petitioner instead proceeded on an unexplained "plain and ordinary meaning."

> [T]he midserver of the '663 patent is used to 'collect, aggregate, analyze, transform, and securely transmit data from a multitude of computing devices and peripherals at an external network to a cloud-based service.' [] The '663 patent describes a midserver as 'an interface between an external network and a cloud-based service' and '[u]sing the midserver as a gateway to the cloud-based service.' The server hosting the cloud agent of Maturana meets this description ….

Pet. 40 (citing Ex. 1001 ['663] 7:60-66, 8:5-7, 8:17-19, 11:28-29).

Petitioner's argument does not apply the patent's lexicographic definition and does not demonstrate that it is satisfied. While Petitioner does contend that Maturana's "server hosting the cloud agent" acts as an "interface between an external network and a cloud-based service," (Pet., 40) Petitioner does not contend—let alone demonstrate—that Maturana's server "runs ***one or more containerized services***," as the definition requires. *See* Pet., 40-42. Nor does Petitioner contend it would be inherent or obvious for Maturana's server, or anything else in the combination, to "run[] one or more containerized services."

Indeed, the term "containerized" does not appear ***anywhere in the Petition or Petitioner's expert declaration***.[2] Moreover, when discussing the '663 patent's

[2] While the term "container" appears once in the Petition (and twice in the expert declaration), those discussions have nothing to do with either "containerized

specification, Petitioner cites and quotes from the lexicographic definition but omits the requirement that the "server … run[] one or more containerized services" as the comparison below reveals:

The Petition And Its Declarant Omit The Requirement That The Midserver "Run[] One Or More Containerized Services"	
Petition	**'663's lexicographic definition**
The '663 patent describes "midserver" as "a server that functions as an interface between an external network and a cloud-based service," and performs standard operations such as "collecting, aggregating, analyzing, filtering, transforming, and securely transmitting data." EX1001, 12:66-13:5. Pet., 18; *see also* Ex. 1003 [Crovella-Decl.] ¶ 45 (same)	"Midserver" as used herein means a server that functions as an interface between an external network and a cloud-based service, ***and which runs one or more containerized services*** that perform one or more of: collecting, aggregating, analyzing, filtering, transforming, and securely transmitting data.

services" or limitation 1.2 where Petitioner attempts to demonstrate that the "midserver" limitation is satisfied by Maturana. *See* Pet., 58 (alluding once to a "container," not "containerized services," and only in connection with limitation 1.6); *see also* Ex. 1003 [Crovella-Decl.] ¶ 68 (same), 15 (discussing VM containers when discussing the state of the art). Thus, neither the Petition nor the expert discuss containerized services and no attempt is made to show that Maturana's server runs any containerized services.

Thus, it cannot be said that Petitioner and its declarant accidentally overlooked the lexicographic definition. They both cited and discussed it and yet omitted the clear requirement that the "server … run[] one or more containerized services."

Because the patent's lexicographic definition requires that a midserver run "one or more containerized services," a system lacking that feature cannot qualify as a "midserver." Petitioner and its expert do not apply the patent's express definition and do not demonstrate that Maturana's server runs one or more containerized services as required. Consequently, Petitioner has failed to show a reasonable likelihood of prevailing as to any claim.

Moreover, Petitioner's failure to show that this limitation is met by its combination is not something that can be remedied later. The rules governing IPRs are clear. Petitioner must "make its case" in its petition, including presenting all evidence necessary to establish a prima facie case of unpatentability. As the Federal Circuit has explained:

> Unlike district court litigation—where parties have greater freedom to revise and develop their arguments over time and in response to newly discovered material—***the expedited nature of IPRs bring with it an obligation for petitioners to make their case in their petition to institute***. While the Board's requirements are strict ones, they are

> requirements of which petitioners are aware when they seek to institute an IPR.

Intelligent Bio-Sys., Inc. v. Illumina Cambridge Ltd., 821 F.3d 1359, 1369 (Fed. Cir. 2016). Further, the failure to address and demonstrate a clear definition is satisfied is a further violation of the requirements that the petition state "how the challenged claim is to be construed" and explain "how the construed claim is unpatentable." 37 C.F.R. §§ 42.104(b)(3)-(4).

Having failed to show that its combination discloses or renders obvious a "midserver" as expressly defined in the patent, Petitioner has not demonstrated a reasonable likelihood of prevailing on any challenged claim. Institution must therefore be denied.

B. Petitioner Fails To Prove That Its Combination Discloses Or Renders Obvious "A Single Data Stream" (All Claims).

Each challenged claim requires "retransmit[ting] the received data over the secure connection as a single ***data stream***." Ex. 1001 ['663] cl. 1, cl. 4. In its Maturana/Kashyap combination, Petitioner contends that Maturana's aggregation of data from multiple sensors into a "***single*** cloud upload packet" satisfies this limitation. Pet., 58-59.

Petitioner relies upon Maturana packaging a compressed data file into "***a*** data packet" that is pushed to the cloud platform:

> [s]pecifically, the cloud agent [of Maturana] "packages the compressed data file 412 into a data packet and pushes the data packet to the cloud platform" (*retransmit the received data over the secure connection as a single data stream*). EX1004, [0052]; *id.*, [0076]. The data is provided as a single data stream because "data from multiple sensors measuring related aspects of an automation system can be identified and *aggregated into a single cloud upload packet* by cloud agent 202." EX1004, [0054] (emphasis added), EX1003, ¶100.

Pet., 59 (emphasis added by Petitioner).

Petitioner's theory therefore depends on an implicit construction of "single data stream" under which a "***single*** cloud upload ***packet***" qualifies as a "data ***stream***." Petitioner, however, never articulates or supports such a construction; it maintains that no construction is necessary and purports to "appl[y] the plain and ordinary meaning of all claim terms." Pet., 21. Nor does Petitioner identify any intrinsic or extrinsic evidence suggesting that a single, discrete upload packet would qualify as a "data stream."

Instead, Petitioner cites only paragraph 100 of the Crovella Declaration, which merely parrots the Petition's conclusory assertion.[3] *Compare* Ex. 1003

[3] Petitioner also cites Maturana's paragraphs 52, 76 and 54 (see block quote above), but these passages merely describe how Maturana's system operates and

[Crovella-Decl.] ¶ 100 *with* Pet., 59. Obviousness, however, cannot be shown by "mere conclusory statements." *See In re Magnum Oil Tools Int'l, Ltd.*, 829 F.3d 1364, 1380 (Fed. Cir. 2016). Similarly, conclusory expert testimony that merely parrots the Petition "adds little to the conclusory assertion for which it is offered to support[.]" *See Xerox Corp. v. Bytemark, Inc.*, IPR2022-00624, Paper 9, 15-16 (Aug. 24, 2022) (precedential); *see also* 37 C.F.R. § 42.65(a).

Petitioner nowhere explains why a "single data packet" qualifies as a "single data stream," nor does it identify any teaching in Maturana describing retransmission as a data stream. That failure is dispositive. Petitioner bears the burden to demonstrate that its proposed combination satisfies each claim limitation, but offers no evidence—intrinsic or extrinsic—supporting its implicit view that a single, discrete upload constitutes a "data stream." Having failed to meet that burden, institution should be denied.

Even if the Director were to look beyond Petitioner's unsupported assertion, there is no evidentiary basis to conclude that uploading a "single data packet" constitutes a "data stream." Extrinsic materials, for example, describe "data streams" in the context of data conveyed over time, not a single, discrete

---

transmits data; they do not address whether such transmission constitutes a "data stream."

transmission. *See* Ex. 2007 [Tencent-Encyclopedia] 1 ("A data stream is a continuous flow of data… Data streams are essentially sequences of data elements made available over time."); Ex. 2008 [Spotfire-Glossary] 1 ("A data stream consists of a series of data elements ordered in time… A data stream can be visualized as an endless conveyor belt, carrying data elements and continuously feeding them into a data processor."). At a minimum, this usage is inconsistent with Petitioner's assumption that uploading a single data packet would constitute a "single data stream."

Nor does the '663 patent support any such assumption. For example, the '663 patent's "Discussion of the State of the Art" references "bandwidth concerns with ***constantly*** streaming data" and it being "unclear from a data stream what portion of the data results in triggering of certain security alerts …" Ex. 1001 ['663] 7:25-30, 7:46-50. The specification likewise refers to "continuous streaming." *See id.*, 17:7-63.

The patent's description of Figure 6 further magnifies Petitioner's omission. Once a secure VPN connection is established, "all communication is done over the VPN," and containerized services "forward data using transport layer security (TLS)." Ex. 1001 ['663] 16:48-56. The '663 patent, in this regard, describes forwarding data over an established connection, not merely packaging data into a one-time upload and pushing it upstream. By contrast, Maturana's cloud agent

closes its communication channel after an upload is completed. *See* Ex. 1004 [Maturana] [0076] (confirming data channel "automatically close[s] upon completion of the data transfer").

Quite simply, Petitioner identifies no disclosure in Maturana describing retransmission as a "data stream," and offers no technical explanation or evidence for its implicit position that a single upload packet would qualify as one. Nor does Petitioner identify any intrinsic or extrinsic evidence supporting Petitioner's position. Thus, the Director should not simply presume that uploading a single data packet satisfies the requirement of retransmission "as a single data stream."

Petitioner therefore fails to show that its combination discloses or renders obvious retransmission "as a single data stream."

C. Petitioner Fails To Prove That Its Combination Discloses Or Renders Obvious "establish[ing] a secure connection with an external network" (All Claims).

Each claim requires "establish[ing] a secure connection with an external network." *See* Ex. 1001 ['663] cls. 1, 4. Petitioner's showing on this limitation fails because it relies on a security mechanism from Maturana that depends on verifying the identity of a particular physical device, while simultaneously proposing a combination with Kashyap's virtual-appliance implementation that defeats Maturana's device-specific verification methodology.

More specifically, the Petition relies on Maturana's registry-based handshake mechanism for "establish[ing] a secure connection." That mechanism depends on use of a physical device address, such as a MAC address, to confirm that the particular physical device seeking access to the cloud platform is authorized to do so. If the cloud agent is improperly moved to a different device, access is denied. In that respect, Maturana functions much like fingerprint authentication: before access is granted, the system verifies that the requesting physical device is authorized to connect. *See* Section II.C.1.

Petitioner, however, does not rely on Maturana alone. Instead, it proposes implementing Maturana's cloud agent as a standardized virtual appliance under Kashyap—which uses a virtual machine environment. In that environment, the device identifier presented by the system is assigned and changeable, rather than a fixed hardware-specific identifier of a particular physical machine. *See* Section II.C.2.

That modification breaks the premise on which Maturana's handshake methodology depends. Maturana's security mechanism is not merely concerned with identifying a device, but with confirming that the requesting device is the same previously registered physical device. Kashyap's virtualized environment eliminates that guarantee by allowing the cloud agent to be instantiated across different physical machines while presenting software-assigned, changeable

identifiers. The Petition never explains how Maturana's device-specific authentication requirement could function in that modified system or how the combined system would still "establish[] a secure connection" as claimed. *See* Section II.C.3.

Because Petitioner does not show that its relied-upon security mechanism survives its proposed modification, it has not demonstrated that the combination discloses or renders obvious "establish[ing] a secure connection with an external network."

1. Petitioner Relies Upon Maturana's Physical Device Address Methodology For The "establish[ing] a secure connection" Limitation.

For the "establish[ing] a secure connection with an external network" limitation, Petitioner relies exclusively on Maturana's handshaking methodology:

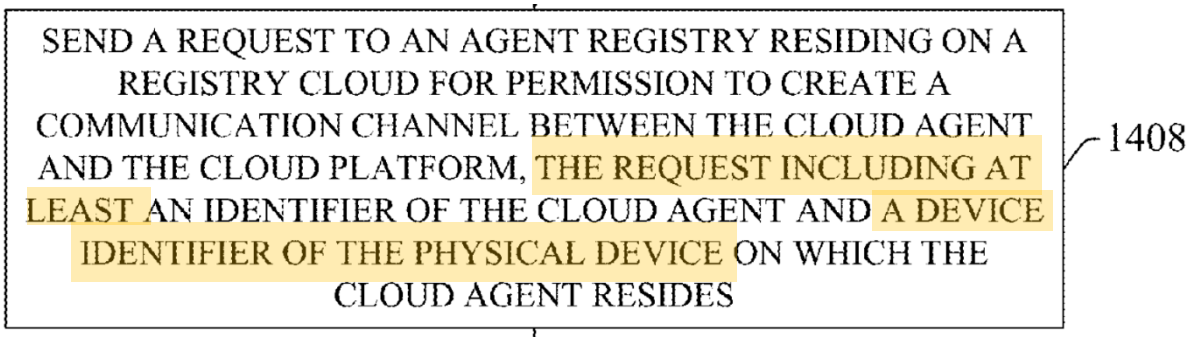
> Maturana discloses establishing a secure network connection to an external network, referred to by Maturana as "the cloud," through a "methodology for securely connecting a cloud agent to a cloud platform by handshaking with an agent registry" (*establish a secure network connection to an external network*). EX1004, [***0021***], [0089].

Pet., 49-50.

Petitioner's reliance upon Maturana's handshaking methodology is unmistakable. Petitioner states that Maturana "securely connect[s] a cloud agent to a cloud platform ***by handshaking*** with an agent registry," and further provides a

parenthetical emphasizing that the handshaking methodology is relied upon to "establish a secure network connection to an external network." The Petition further cites paragraph 21 of Maturana which describes Maturana's Figure 14. *See* Pet., 50 (citing Ex. 1004 [Maturana] [0021] ("FIG. **14** is a flowchart of an example methodology for securely connecting a cloud agent to a cloud platform by handshaking with an agent registry.")), *see also* Pet., 50 (citing Ex. 1004 [Maturana] [0089]).

Under Maturana's Figure 14 process, the cloud agent sends a request for permission to create a communication channel to an agent registry. Ex. 1004 [Maturana], [0090]; *see also id.*, Fig. 14. This request "includ[es] at least an identifier of the cloud agent and ***a device identifier of the physical device***":



Id., Fig. 14 (step 1408); *see also id.*, [0090] ("The device identifier may be, for example, a MAC address ***or other physical address for the device***."). Permission to utilize the communication channel is based upon that physical device identifier. *Id.*, [0091].

As Maturana confirms, determining whether to grant permission to utilize the channel is "based on information provided in the agent request." *Id.*, [0075]. The agent registry uses the information provided "to confirm that the physical address of the computing device 808 from which the request was received is associated with the particular cloud agent" that made the request. *Id.* "By confirming that the connection request for Cloud Agent 1 has been received from the previously registered computing device 808, the agent registry ensures that Cloud Agent 1 cannot be used to establish connectivity to the cloud platform if improperly moved or copied to another physical device." *Id.* Thus, the secure connection mechanism relied upon by Petitioner is expressly tied to the physical device on which the cloud agent executes.

Petitioner confirms that its combination relies on this same validation process—dependent on a physical address:

> Maturana contemplates that a cloud agent may be "moved to a different computing device," with registry checks denying communication if the cloud agent is unregistered on the different device. EX1004, [0075]. Maturana thus teaches enforcing cloud agent identity per deployment to ensure secure communication. EX1004, [0074]-[0076].

Pet., 35-36. The Petition therefore relies on Maturana's device-specific handshake to satisfy the "secure connection" limitation.

2. The Combined System Relies On Virtualization That Breaks Maturana's Hardware-Specific Device Identification.

Petitioner does not contend that Maturana alone renders the claims obvious. Instead, it relies on a Maturana/Kashyap combination that combines Maturana's "on-premise cloud agent" with Kashyap's "automated ***virtual***-appliance deployment and configuration techniques" to "implement Maturana's cloud agent as a standardized, easily deployable cloud-agent service." Pet., 28-29. In doing so, Petitioner shifts Maturana's cloud agent from a model in which the secure handshake depends on a hardware-specific device identifier to Kashyap's model in which the cloud agent executes inside a virtual machine.

A virtual machine, however, is not tied to any particular physical device. It is a software-defined environment that operates through virtualized hardware interfaces, not direct access to the underlying hardware. *See* Ex. 2009 [AWS-Virtualization] 2-3; Ex. 2010 [IBM-Virtualization] 7. Unlike the "bare metal" systems that are the focus of Maturana, a hypervisor mediates access to the physical machine and presents each virtual machine with virtual hardware interfaces that emulate physical components. *See* Ex. 2009 [AWS-Virtualization] 4; Ex. 2011 [ScaleComputing-Hypervisor] 4. Accordingly, virtual machines typically do not expose the immutable hardware identifiers of the underlying physical device.

A physical network device typically has a manufacturer-assigned MAC address that remains fixed. *See* Ex. 2012 [Pitt-MAC-Address] 1-3. By contrast, in a virtualized environment identifiers such as MAC addresses are typically software-assigned values associated with virtual network interfaces rather than the physical machine itself. *See* Ex. 2006 [Malek-Decl.] ¶ 63. As Dr. Malek explains, numerous documents confirm that device identifiers (such as MAC addresses) are assigned and changeable in virtual environments:

> In a virtual environment, like Kashyap's, the MAC addresses (or other physical device identifiers) typically are not the MAC address associated with the physical device. Rather, most virtualization platforms ***assign*** MAC addresses to the network interface cards used by virtual machines rather than relying on hardware network interface cards built into a specific physical computing device. *See, e.g.*, Ex. 2013 [Virtual Machine MAC Address] 1 (discussing a "MAC address-generation algorithm" that generates MAC addresses that are assigned); *see also* Ex. 2014 [Windows Hyper-V server] 1 ("The Hyper-V server is configured to provide Dynamic MAC addresses to the guest machines."). For example, documentation for VMware environments explains that "from virtual machine creation … systems assign each virtual network interface its own unique MAC address." *See also* Ex. 2015 [Scenario: Manage the Network Configuration of Virtual Machines] 3 ("The default option is to use a dynamic MAC address, which means that Hyper-V will generate a MAC address for the network adaptor."). Moreover, such assigned MAC addresses are

> readily changeable. Ex. 2016 [MAC Address Assignment] 1 (explaining that MAC addresses are generated and confirming that "[w]hen a virtual machine is powered on again, it might acquire a different MAC address.").
>
> As these documents confirm, in a virtualized environment, the MAC address is assigned and may be changed as desirable by the virtualization platform. It is not necessarily tied to any particular physical computing device.

Ex. 2006 [Malek-Decl.] ¶¶ 64-65.

Thus, in the combined system, the physical device address is no longer the actual device address for the physical machine but is, instead, an assigned and changeable value. Because such identifiers are assigned and changeable, they do not identify any particular physical device.

Petitioner offers no response to this fundamental problem. Instead, it asserts that Maturana and Kashyap are "compatible" because Maturana's cloud agent is purportedly executable on "virtual machine(s)." Pet., 35-36 (citing Ex. 1004 [0042]). But the cited paragraph of Maturana only states that certain "aspects" and "components" of Maturana's system are executable on a virtual machine. *See* Ex. 1004 [Maturana] [0042]. Nothing in Maturana suggests that its handshake mechanism—which depends on identifying a particular physical device—can be

implemented in a virtualized environment where the identifier is not tied to any particular physical device.

Nor does Petitioner explain how Maturana's handshake process could function when the device identifier is no longer tied to a specific physical machine. A general disclosure that "aspects" of a system may be implemented on a virtual machine does not support applying virtualization to every component of that system, particularly where doing so would undermine the mechanism itself. *See Garmin Int'l, Inc. v. Logantree, LP*, IPR2018-00564, Paper 25, 20-21 (Aug. 2, 2019).

3. Petitioner Fails To Demonstrate That The Maturana/Kashyap Combined System "establishes a secure connection."

The Petition never explains how Maturana's device-specific method of establishing a secure connection survives the shift from Maturana's hardware-specific architecture to Kashyap's virtualized platform, where device identifiers are no longer tied to particular physical machines. As Dr. Malek explains, deploying the cloud agent in this context breaks the device-specific authentication mechanism on which Maturana's handshake depends:

> Petitioner contends that a POSITA would implement Maturana's process for granting certification—verifying the physical address of the cloud agent device through handshaking with an agent registry—as part of Kashyap's virtual appliance deployment. Petitioner, however, does

> not explain how such a process would "establish[] a secure connection" as required by the claims—nor would a POSITA understand it to do so.
>
> Deploying Maturana's cloud agent as a virtual appliance fundamentally compromises the trust assumptions underlying Maturana's physical-device validation. Maturana's handshake process, again, relies upon identifying a particular, unique device. In a virtual machine environment, however, traditional "physical" device identifiers such as MAC addresses are not inherently tied to a unique physical device. Instead, these identifiers are software-defined or hypervisor-managed values that can be assigned, duplicated, or changed by the virtualization platform. Accordingly, a virtual machine presenting a MAC address or other device identifier does not demonstrate that the request originated from a particular physical device in the manner required by Maturana's security process.

Ex. 2006 [Malek-Decl.] ¶¶ 69-70. As Dr. Malek explains, in the combined system, the device identifier supplied during the handshake would therefore no longer provide the hardware-specific assurance that Maturana relies upon to determine whether the requesting device is authorized to access the cloud platform. In such an environment, the registry would be validating a software-defined and changeable identifier rather than confirming the identity of a specific physical device.

Petitioner nevertheless relies on this same handshake protocol to satisfy the "secure connection" limitation, even though the protocol's device-specific

verification mechanism no longer functions in the modified system. As Dr. Malek further explains:

> In a virtualized system, any access to a "physical device address" is mediated by the hypervisor. *See* Ex. 1005 [Kashyap] [0032]; Ex. 2011 [ScaleComputing-Hypervisor] 4-6. The hosted virtual machine does not inherently possess a hardware identity; it receives assigned attributes through a software abstraction layer that is configurable by the virtualization platform. Even if a host MAC address or other hardware identifier is exposed to the guest, it is an identifier that is assigned and changeable by the hypervisor and therefore cannot be relied upon to identify any particular physical device.

Ex. 2006 [Malek-Decl.] ¶ 73.

A POSITA therefore would not understand the combined Maturana/Kashyap system to establish a secure connection in the manner required by the claims. Maturana's security mechanism depends on verifying that a specific cloud agent is operating on a particular registered physical device, but Kashyap's virtual appliance architecture allows that agent to be instantiated across different hosts while presenting software-assigned device identifiers. Worse yet, the software assigned identifier is written in sand; it can be changed at any time. Because the Petition never explains how Maturana's device-specific authentication process would function in that modified environment, it fails to demonstrate that the

proposed combination discloses or renders obvious "establish[ing] a secure connection to an external network."

D. Petitioner Fails To Demonstrate That A POSITA Would Be Motivated To Combine Maturana/Kashyap.

Petitioner also fails to demonstrate that a POSITA would have been motivated to combine Maturana and Kashyap, for two independent reasons. First, as explained in Section II.C, Petitioner's proposed combination would eliminate Maturana's ability to verify that a connection request originates from a registered physical device, thereby undermining the security function Maturana is designed to provide. Petitioner does not explain why a POSITA would adopt a modification that deprives the system of its ability to ensure a secure connection. *See* Section II.D.1. Second, Petitioner asserts that a POSITA would be motivated to enable portability of the cloud agent across devices. But this is directly contrary to Maturana's system design, which makes clear that such movement is a condition that must be detected and prevented. Petitioner provides no basis to conclude that a POSITA would be motivated to combine Maturana and Kashyap in contravention of Maturana's system design. *See* Section II.D.2. Each defect independently demonstrates that Petitioner has failed to show that a POSITA would have been motivated to adopt the proposed combination.

1. Petitioner Fails To Demonstrate Motivation Because The Proposed Combination Undermines Maturana's Means Of Establishing A Secure Connection.

Petitioner fails to demonstrate that a POSITA would have been motivated to combine Maturana and Kashyap in the manner proposed because the modification Petitioner advances would undermine Maturana's means of establishing a secure connection. As explained, implementing Maturana's cloud agent within Kashyap's virtualized environment eliminates the hardware-specific device identification on which Maturana's handshake depends. *See* Section II.C, *supra*. The proposed combination therefore eliminates Maturana's ability to verify that a connection request originates from the previously registered physical device.

Petitioner's proposed modification cuts directly against Maturana's design. Under Maturana's approach, the handshake uses the device identifier to confirm that the request originated from the specific physical device previously registered with the system. *See* Ex. 1004 [Maturana] [0075], [0090]-[0091]. Under Petitioner's Kashyap-based implementation, however, the identifier is no longer tied to a particular physical machine, but instead is assigned and changeable within the virtualized environment. The handshake would therefore no longer ensure that the cloud agent seeking to connect is operating on the registered physical device. *See* Section II.C, *supra*. The proposed combination leaves Maturana's handshake unable to perform device-specific verification.

Petitioner identifies no benefit to discarding this essential safeguard, nor does it explain why a POSITA seeking secure cloud connectivity would adopt a modification that makes it more difficult to verify the identity of the connecting device. The Federal Circuit has made clear that the Board must weigh the benefits and drawbacks of a proposed modification in assessing motivation to combine. *Arctic Cat Inc. v. Polaris Indus.*, 795 F. App'x 827, 833 (Fed. Cir. 2019); *Henny Penny Corp. v. Frymaster LLC*, 938 F.3d 1324, 1331-32 (Fed. Cir. 2019); see also *Arctic Cat Inc. v. Bombardier Recreational Prods.*, 876 F.3d 1350, 1363 (Fed. Cir. 2017). Here, the drawback is material and disables the system's ability to establish a secure connection.

Nor would a POSITA be motivated to modify a reference in a way that renders it inoperable for its intended purpose or changes its principle of operation. *Plas-Pak Indus. v. Sulzer Mixpac AG*, 600 F. App'x 755, 759 (Fed. Cir. 2015); *In re Gordon*, 733 F.2d 900, 902 (Fed. Cir. 1984); *In re ICON Health & Fitness, Inc.*, 496 F.3d 1374, 1382 (Fed. Cir. 2007); *Tec Air, Inc. v. Denso Mfg. Mich., Inc.*, 192 F.3d 1353, 1360 (Fed. Cir. 1999); *In re Ratti*, 270 F.2d 810, 813 (CCPA 1959). Maturana's handshaking methodology is designed to ensure that only a verified instance of a cloud agent running on the proper physical device may establish a connection to the cloud platform. Petitioner's proposed virtualization removes the hardware-specific basis for that verification, in contravention of Maturana's

design.

A POSITA therefore would not have been motivated to combine Maturana and Kashyap in the manner Petitioner proposes.

2. Petitioner's Asserted Motivation Seeks To Achieve The Very Condition Maturana Is Designed To Prevent.

Petitioner also fails to demonstrate that a POSITA would have been motivated to combine Maturana and Kashyap because its asserted rationale is fundamentally inconsistent with Maturana's system design. Maturana is designed to ensure that a cloud agent cannot be moved across devices in a manner that enables unauthorized access. Petitioner's proposed combination, by contrast, depends on enabling that very movement.

Petitioner contends that Maturana and Kashyap are "align[ed]" and "naturally compatible," relying on Maturana's statement that a cloud agent may be "moved to a different computing device." *See* Pet., 31-32, 35-36 (citing Ex. 1004 [Maturana] [0075]). From this, Petitioner asserts that a POSITA would have been motivated to implement the cloud agent in a more portable, virtualized form. *Id.*

That theory, however, cannot be reconciled with Maturana's system design. Maturana's authentication mechanism depends on verifying that a connection request originates from a specific, previously registered computing device so that "Cloud Agent 1 ***cannot be used to establish connectivity to the cloud platform if***

improperly moved or copied to another physical device." Ex. 1004 [Maturana] [0075]. Thus, Maturana treats movement of the cloud agent as a condition that must be detected and blocked, not as a feature to be enabled.

Petitioner's reliance on Maturana's passing reference to "virtual machine(s)" (Pet., 30) only underscores this mismatch. That disclosure is generic, stating only that unspecified "aspects" or "components" may be executed in a virtual machine, without addressing how Maturana's system would operate in such an environment. Ex. 1004 [Maturana] [0042]. It does not explain—nor does Petitioner provide any evidence—how Maturana's device-specific authentication could be implemented in a virtualized system. A generalized statement that certain "aspects" may run in a virtual machine does not support applying virtualization to the system as a whole, particularly where doing so would defeat its core operating assumption.

Petitioner's asserted motivation therefore seeks to achieve the very condition Maturana is designed to prevent. As discussed in the immediately preceding section, the law requires weighing the benefits and drawbacks of a proposed combination and makes clear that a POSITA would not be motivated to interfere with a primary reference's principle of operation. *See* Section II.D.1 (citing caselaw). Here, Petitioner ignores both principles and fails to demonstrate that a POSITA would be motivated to modify Maturana in a manner that facilitates movement of a cloud agent across devices when Maturana's disclosed system is

specifically designed to ensure that such movement does not enable unauthorized connections.

III. CONCLUSION

Thus, as the Director has already found, Petitioner fails to demonstrate a reasonable likelihood of success as to any of the challenged claims, and the Petition should be denied.

Respectfully submitted,

/ Kenneth J. Weatherwax /

Kenneth J. Weatherwax (Reg. No. 54,528)
Nathan Lowenstein, *pro hac vice*
Colette Woo, *pro hac vice*
LOWENSTEIN & WEATHERWAX LLP

Date: August 19, 2026

CERTIFICATE OF COMPLIANCE WITH TYPE-VOLUME LIMITS

This Patent Owner Preliminary Response (the "POPR") consists of 6,609 words, excluding table of contents, table of authorities, certificate of service, this certificate, or table of exhibits. The POPR complies with the type-volume limitation of 14,000 words as mandated in 37 C.F.R. § 42.24. In preparing this certificate, the undersigned has relied on the word count of the word-processing system used to prepare the paper (Microsoft Word).

Respectfully submitted,

/ Michael Cosley /

Date: August 19, 2026

CERTIFICATE OF SERVICE

The undersigned hereby certifies that the following documents were served by electronic service, by agreement between the parties, on the date below:

PATENT OWNER PRELIMINARY RESPONSE

EXHIBITS 2006-2018

The names and addresses of the parties being served are as follows:

James M. Glass	jimglass@quinnemanuel.com
Quincy Lu	quincylu@quinnemanuel.com
Brian Biddinger	brianbiddinger@quinnemanuel.com
Jeremy Thomas	jeremythomas@quinnemanuel.com

Respectfully submitted,

/ Michael Cosley /

Date: August 19, 2026